UNIVERSITÉ DE MONTRÉAL

CONCEPTION, PROTOTYPAGE ET ÉVALUATION D'UN SYSTÈME POUR L'EXPLORATION AUDIO-TACTILE ET SPATIALE DE PAGES WEB PAR DES UTILISATEURS NON-VOYANTS

GRÉGORY PETIT

DÉPARTEMENT DE MATHÉMATIQUES ET DE GÉNIE INDUSTRIEL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION

DU DIPLÔME DE DOCTORAT

(GÉNIE INDUSTRIEL)

AOÛT 2013

© Grégory Petit, 2013.

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée:

CONCEPTION, PROTOTYPAGE ET ÉVALUATION D'UN SYSTÈME POUR L'EXPLORATION AUDIO-TACTILE ET SPATIALE DE PAGES WEB PAR DES UTILISATEURS NON-VOYANTS

présentée par : PETIT Grégory

en vue de l'obtention du diplôme de : Philosophiae Doctor

a été dûment acceptée par le jury d'examen constitué de :

M. DESMARAIS Michel C., Ph.D., président

M. ROBERT Jean-Marc, Doctorat, membre et directeur de recherche

Mme DUFRESNE Aude, Ph.D., membre et codirectrice

M. MABILLEAU Philippe, Ph.D., membre

M. BELLIK Yacine, Doctorat, membre

À mon beau-père

REMERCIEMENTS

En premier lieu, j'aimerais chaleureusement remercier mon directeur et ma codirectrice de thèse, Jean-Marc Robert et Aude Dufresne. J'ai pu apprécier la passion d'Aude pour le domaine de l'accessibilité ainsi que sa capacité à se battre pour les choses qui lui tenaient à cœur. J'ai pu bénéficier de la rigueur de Jean-Marc pour la méthodologie et la qualité de la rédaction. Leur apport a largement dépassé l'encadrement de cette thèse et je leur exprime ma plus profonde gratitude.

Je remercie sincèrement Michel Desmarais, Yacine Bellik et Philippe Mabilleau d'avoir accepté de faire partie de mon jury. Je me considère chanceux de faire évaluer mon travail par un jury multidisciplinaire.

Je souhaite remercier le Ministère des services gouvernementaux du Québec pour avoir participé au financement de ma recherche. J'aimerais d'ailleurs remercier la MATI et son directeur Jacques Raynauld qui m'ont grandement soutenu au début de ma thèse et m'ont aidé dans l'obtention du financement de ma recherche.

J'aimerais remercier tout particulièrement Gabriel Michel pour m'avoir fait connaître le domaine de l'ergonomie et tout particulièrement l'accessibilité. Gaby, tu m'as donné une ouverture d'esprit que je n'avais pas avant d'intégrer ta formation très ouverte sur le monde.

Je remercie Vincent Lévesque et ses collègues de TactileLabs pour leur aide précieuse au début de cette recherche. Je remercie Anne Jarry, Jean-Marie d'Amour et l'INLB pour m'avoir guidé au sein de la population non-voyante, ainsi qu'Aurélie Jonquet pour son aide à la fin de ma thèse. Je remercie aussi Marc Bourdeau pour son aide en statistiques dans les derniers instants de ma rédaction.

J'aimerais remercier François Courtemanche pour nos nombreuses discussions sur nos thèses respectives, et sur tous les autres sujets plus ou moins sérieux. Je remercie aussi mes collègues *pédauques* du CITÉ : Claire, Karine, et Emma. Je vous ai rencontrées trop tard; vous avez ensoleillé mes deux dernières années de doctorat.

Je remercie mes amis de Montréal et d'ailleurs (pardon à ceux et celles que j'oublie) : Illia, Jess, Simon, Caro, Delph, Anaïs, Romain, Jocelyn, Alex, Etienne, Philippe, Élien, Seb, Arnaud, Saïd,

Gwen, Tom, Marion, Lety, Christine et Geneviève. Vous avez tous joué un rôle au cours de mon doctorat et je vous en remercie infiniment.

J'aimerais remercier ce jeune élève non-voyant d'une école secondaire pour avoir fait germer l'idée de cette thèse dans ma tête. Je ne me rappelle malheureusement plus de son nom mais sans lui, il est fort probable que je n'aurais jamais pensé à travailler sur un système audio-tactile améliorant l'accessibilité du Web pour les personnes non-voyantes. J'en profite pour remercier aussi les personnes non-voyantes qui ont participé à l'expérimentation décrite dans cette thèse.

Enfin, j'aimerais remercier ma famille pour son soutien. Maman, Alain, vous m'avez permis de m'épanouir, même si vous avez dû me laisser partir très loin. Vous m'avez inculqué les valeurs nécessaires pour réussir dans ce monde un peu bizarre. Je vous dois énormément, ainsi qu'à ma petite sœur Léa et mon grand frère Ludo. Je vous aime fort de l'autre côté de l'océan.

RÉSUMÉ

L'accès à l'information a drastiquement changé depuis l'apparition des nouvelles technologies et du monde en ligne. Il est maintenant possible d'accéder à une multitude d'informations en tout temps et en tout lieu. Cette apparente facilité d'accès à l'information est cependant très loin de la réalité des personnes ayant un handicap pour qui l'arrivée des nouvelles technologies et du monde en ligne a créé de nouvelles situations de handicap. La présente thèse se concentre sur les situations de handicap rencontrées par les personnes non-voyantes, au cours de l'exploration de pages Web. Heureusement, des technologies adaptées sont actuellement disponibles pour les personnes non-voyantes qui désirent accéder au monde du Web : les lecteurs d'écran. Ceux-ci permettent une exploration linéaire de la page à l'aide d'un retour sonore généré par une synthèse vocale. Cette adaptation améliore grandement l'accès au monde du Web mais engendre son lot de frustrations. Celles-ci sont principalement liées au non-respect des lignes de conduite d'accessibilité dans plusieurs sites Web et à la présentation strictement linéaire de l'information par les lecteurs d'écran. L'objectif de notre recherche est d'améliorer l'accès au Web pour les utilisateurs non-voyants, en leur proposant une alternative à l'exploration linéaire : l'exploration spatiale de pages Web, c.-à-d. à l'aide de retours tactiles et sonores. Par l'intermédiaire de l'exploration spatiale, nous souhaitons passer outre certains problèmes d'accessibilité dans les pages Web, en donner une meilleure idée globale, et permettre de mieux associer les informations de cette page entre elles. Notre hypothèse principale est la suivante :

L'exploration spatiale d'un site Web est plus efficace et plus efficiente que l'exploration linéaire pour des utilisateurs non-voyants.

L'exploration spatiale se fait par l'intermédiaire du fureteur d'écran multimodal TactoWeb que nous avons développé. TactoWeb est contrôlé par le Tactograph, un appareil générant des sensations tactiles sous forme de vibrations et d'ondulations qui dépendent de l'emplacement du curseur sur la page Web. Les retours sonores sont une combinaison entre une synthèse vocale et un ensemble d'audicônes. Cette approche multimodale permet de recréer les liens entre les différents éléments d'une page Web qui pourraient disparaître au cours de la linéarisation de l'information. Avant de concevoir notre système, nous avons identifié les points forts et les points faibles des lecteurs d'écrans. La thèse présente ensuite le processus de conception de notre fureteur multimodal du point de vue logiciel, ainsi que l'amélioration du matériel utilisé. Un

tutoriel d'apprentissage a été conçu afin de guider les utilisateurs non-voyants dans l'exploration spatiale. Nous procédons ensuite à l'évaluation de notre système à l'aide d'une étude comparative. Cette étude prend la forme d'une expérimentation comparant JAWS, un lecteur d'écran permettant l'exploration linéaire, avec TactoWeb, notre système d'exploration spatiale. Il est évident que les deux outils ont différents degrés de maturité. En effet, JAWS existe depuis presque 20 ans alors que TactoWeb n'en est qu'à sa première version. Cette différence de maturité peut clairement influer sur les résultats de notre évaluation en faveur de l'exploration linéaire avec JAWS. À cela s'ajoute le fait que nos participants avaient plusieurs années d'expérience avec JAWS alors qu'ils faisaient de l'exploration spatiale avec TactoWeb pour la première fois et pour une courte période.

L'étude porte sur 14 participants non-voyants et met en relief les différents aspects de l'exploration spatiale qui permettent de réduire les situations de handicap rencontrées au cours de l'exploration de pages Web. Pour ce faire, nous avons étudié les taux de succès, la difficulté rencontrée et le temps d'exécution pour huit tâches effectuées selon les deux types d'exploration. Ces tâches sont divisées en deux types en fonction de leur but : rechercher une information dans un site Web et remplir un formulaire. De plus, chacun de ces sous-ensembles de tâches contient deux tâches à effectuer dans des sites accessibles, et deux tâches à effectuer dans des sites non-accessibles. Chaque participant a donc effectué quatre tâches avec chacun des deux outils (JAWS et TactoWeb), soit une tâche de chaque type, dans des sites accessibles ou non. Cela nous a permis d'étudier les deux types d'exploration dans quatre situations différentes et d'observer leurs avantages et inconvénients dans chacune de ces situations. De plus, nous avons observé s'il y avait des différences dans l'utilisation des deux types d'exploration en fonction de la période d'apparition de la cécité chez nos participants (cécité de naissance ou tardive).

L'étude a infirmé notre hypothèse principale. Les deux types d'exploration sont aussi efficaces l'un que l'autre pour ce qui est de la capacité des participants à réaliser les tâches et l'exploration linéaire est plus efficiente avec un temps d'exécution plus court. Tel que mentionné ci-dessus, cette différence d'efficience peut s'expliquer par le fait que les participants avaient beaucoup d'expérience avec l'outil d'exploration linéaire et aucune avec l'outil d'exploration spatiale, à part la durée de notre tutoriel (moyenne de 38 min) et la durée de l'expérience elle-même (entre deux et trois heures). De plus, la différence de maturité entre les deux outils est à considérer. Notre hypothèse principale est infirmée en tenant compte de ces biais mais il pourrait en être

autrement lorsque TactoWeb aura une plus grande maturité et lorsque les participants auront plus d'expérience avec notre outil. Néanmoins, malgré les différences d'expérience des participants et de maturité entre les deux outils, nous obtenons une efficacité similaire, ce qui est un gros point positif pour l'exploration spatiale. Cette dernière semble donc demander beaucoup moins de temps d'apprentissage que l'exploration linéaire. De plus, l'exploration spatiale génère moins de difficulté dans les tâches de remplissage de formulaire et a un temps d'exécution similaire à l'exploration linéaire dans les tâches de ce type, effectuées dans des sites non-accessibles. D'ailleurs, les écarts de temps d'exécution entre les deux types d'exploration sont globalement plus réduits dans les tâches effectuées dans des sites non-accessibles, par rapport à celles effectuées dans des sites accessibles. Même si l'exploration spatiale améliore l'accès aux sites Web pour les personnes non-voyantes dans certaines situations de handicap (non-respect de l'ordre logique de lecture dans le code, non-respect de l'association entre un champ et son étiquette dans les formulaires), notre outil TactoWeb peut être amélioré. L'exploration horizontale devrait être plus guidée et la qualité de la synthèse vocale grandement améliorée.

Enfin, si on regarde globalement les résultats de notre expérimentation, on se rend compte que l'exploration linéaire semble plus pertinente lorsqu'il s'agit de naviguer entre les différentes pages d'un même site Web, et que l'exploration spatiale semble plus adaptée lorsqu'il faut explorer dans une même page Web. Les deux types d'explorations semblent donc complémentaires.

ABSTRACT

The emergence of new technologies and the online world changed the way we access information. It is now possible to access any information, at any time, and in any place. This apparent ease of access to information is however far from the reality of people with disabilities. The emergence of new technologies and the online world have created new situations of handicap for them. This thesis focuses on situations of handicap faced by blind people when they browse Web pages. Fortunately, appropriate technologies are currently available for blind people wishing to access the World Wide Web: screen readers. These systems allow a linear exploration of a Web page, using audio feedback generated by speech synthesis. This adaptation greatly improves Web accessibility but also creates a lot of frustration. This frustration is mainly produced by non-compliance with accessibility guidelines in several Web sites, as well as the strictly linear presentation of information by screen readers. The goal of our research is to improve access to the Web for blind users, offering them an alternative to linear exploration: spatial exploration of Web pages, i.e. with tactile and audible feedbacks. Through spatial exploration, we want to override some accessibility issues in Web pages, give a better overall picture of the pages, and give a better connection between linked information in these pages. Our main hypothesis is:

Spatial exploration of a Web site is more effective and more efficient than linear exploration for blind users.

Spatial exploration is done through TactoWeb, a multimodal Web browser we developed. TactoWeb is controlled by the Tactograph, a tactile feedback device producing undulations and vibrations, depending on where the cursor is on the Web page. Audio feedback is a combination between speech synthesis and a set of earcons. This multimodal approach allows the user to recreate connections between the different elements composing a Web page that could have disappeared during the linearization of the information. Before designing our system, we identified the strengths and weaknesses of screen readers. The thesis presents the process of designing our multimodal Web browser, and improving the hardware we used. A training tutorial was designed to guide blind users in spatial exploration. Then, we evaluate our system using a comparative study. This study takes the form of an experiment comparing JAWS, a screen reader using linear exploration, with TactoWeb, our browser allowing spatial exploration. It is obvious the degree of maturity of each tool is different. JAWS actually exists since 1995 whereas we are

still using the first version of TactoWeb. This difference of maturity could affect the results of our evaluation in favour of the linear exploration with JAWS. In addition, one must remind that our participants have several years of experience with JAWS whereas they will use space exploration with TactoWeb for the first time during the tutorial (average of 38 min) and the experiment (between two and three hours).

The study involves 14 blind participants and highlights the different aspects of space exploration that reduce handicap situations encountered when browsing Web pages. To do so, we studied the success rate, the difficulty and the execution time for eight tasks performed with both types of exploration. These tasks are divided into two types according to their purpose: finding information in a Web site and filling out a form. Moreover, each of these subsets includes two tasks performed in accessible Web sites and two in non-accessible Web sites. Each type of task, in an accessible Web site or not, has been performed by each participant, with each tool (JAWS and TactoWeb). This allowed us to study the two types of exploration in four different situations, observing their advantages and disadvantages for each of these situations. Moreover, we observed whether there were differences between congenital and late blind, depending of the type of exploration used.

The study invalidated our main hypothesis. The two types of exploration are in fact as effective as each other, but linear exploration is more efficient thanks to a shorter execution time. This efficiency difference could be explained by the fact that participants had much more experience with the linear exploration tool than with the spatial exploration tool, which is limited to the time they used our tutorial (average of 38 min) and to the duration of the experiment. Moreover, one must take into account the difference of maturity between the two tools. Our main hypothesis is not validated but it could be different when TactoWeb will have greater maturity and when participants will have more experience with our tool. However we obtain a similar effectiveness despite the difference of user experience among the participants and the difference of maturity between the two tools, and this is a major positive point for spatial exploration. Learning spatial exploration seems to take a lot less time than learning linear exploration. Moreover, spatial exploration generates less difficulty when filling out a form, as well as a similar execution time as the linear exploration for tasks of this type is made in non-accessible Web sites. Also, differences in execution time between the two types of exploration are generally smaller in the tasks performed in non-accessible Web sites than those made in accessible Web sites. Even if

spatial exploration improves Web accessibility for blind people in some situations of handicap (non-compliance with the logical reading order in the code, no association between a field and its label in forms), the TactoWeb browser can be improved. The horizontal exploration could be more guided and the quality of speech synthesis greatly improved.

Finally, if we look at the overall results of our experiment, we realize that linear exploration seems more relevant when it comes to navigating between the different pages of the same Web site, and spatial exploration seems more relevant when exploring in a single Web page. So the two types of exploration seem to be complementary.

TABLE DES MATIÈRES

DÉDICACE III

REMERCIEMENTS IV

RÉSUMÉ VI

ABSTRACT IX

TABLE DES MATIÈRES XII

LISTE DES TABLEAUX XVII

LISTE DES FIGURES XX

LISTE DES SIGLES ET ABRÉVIATIONS XXIII

LISTE DES DÉFINITIONS XXV

LISTE DES ANNEXES XXVI

INTRODUCTION 1

CHAPITRE 1 REVUE DE LITTÉRATURE SUR LES INTERFACES HUMAIN-MACHINE, L'ACCESSIBILITÉ POUR LES UTILISATEURS NON-VOYANTS ET L'INTERACTION TACTILE 4

1.1 Interfaces Humain-Machine 4

1.1.1 Utilisabilité 4

1.1.2 Interaction multimodale 5

1.2 Accessibilité aux utilisateurs non-voyants 8

1.2.1 Handicap visuel 9

1.2.1.1 La perception visuelle 9

1.2.1.2 Le handicap visuel 11

1.2.1.3 Image mentale et handicap visuel de naissance ou de tardif 12

1.2.2 Technologie adaptée 13

1.2.2.1 Technologies relatives au Web et accessibilité 13

1.2.2.2 Lignes directrices pour rendre le contenu Web accessible 15

1.2.2.3 Lecteurs d'écran et navigateur Web 16

1.2.2.4 Limites des technologies actuelles favorisant l'accessibilité au Web 19

1.3 Interaction tactile 21

1.3.1 Le sens du toucher 21

1.3.1.1 La main 21

1.3.1.2 Mouvement 24

1.3.2 Les outils d'interaction liés au sens du toucher 24

1.3.2.1 Les outils à retours tactiles 25

1.3.2.2 Les outils haptiques à retour d'effort 28

1.3.2.3 Exploration haptique et reconnaissance d'objets 30

1.3.2.4 Le Tactograph et MaskGen 31

1.4 Synthèse 33

CHAPITRE 2 ÉTUDE EMPIRIQUE DES TECHNOLOGIES ADAPTÉES POUR LE WEB À L'USAGE DES UTILISATEURS NON-VOYANTS 35

2.1 Méthodologie 35

2.2 JAWS 37

2.2.1 Exploration linéaire avec JAWS 37

2.2.2 Problème liés au non-respect des lignes directrices d'accessibilité 39

2.2.3 Interaction avec JAWS 41

2.3 VoiceOver 41

2.3.1 VoiceOver pour Mac OS 42

2.3.1.1 Exploration linéaire 42

2.3.1.2 Exploration spatiale 44

2.3.2 VoiceOver pour appareils mobiles 46

2.4 La souris iFeel 48

2.5 Synthèse 48

CHAPITRE 3 PROBLÉMATIQUE, OBJECTIF ET HYPOTHÈSES 50

3.1 Objectif de la recherche 51

3.2 Problèmes d'accessibilité pouvant être réglés par une exploration spatiale 52

3.3 Hypothèses 53

CHAPITRE 4 TACTOWEB : UN FURETEUR AUDIO-TACTILE À L'USAGE DES PERSONNES NON-VOYANTES 57

4.1 Matériel utilisé 58

4.1.1 Amélioration du Tactograph 58

4.1.1.1 Le Tactograph au début de la thèse 58

4.1.1.2 Modification de la surface d'exploration 59

4.1.1.3 La coquille du module tactile 61

4.1.2 Plateforme de développement 62

4.2 Architecture et fonctionnalités 64

4.2.1 Architecture 64

4.2.2 Récupération et traitement des données HTML et CSS 65

4.2.2.1 Récupération du code source 65

4.2.2.2 Analyseur syntaxique CSSTidy et CSSContent 66

4.2.2.3 Analyseur syntaxique HTMLCXX et TactoElement 67

4.2.2.4 Association entre TactoElement et CSSContent 68

4.2.3 Affichage des TactoElement : dimensions et localisation 70

4.2.3.1 Relation par rapport au TactoElement parent 72

4.2.3.2 Adaptation de la dimension pour TactoWeb 73

4.2.3.3 Propriétés CSS spécifiques à la localisation des éléments Web 75
4.2.4 Retour tactile 76
4.2.4.1 Création des masques graphiques.....76
4.2.4.2 Librairie de rendu tactile 78
4.2.4.3 Grammaire tactile utilisée 78
4.2.5 Retour sonore 82
4.2.5.1 Audicônes et SDL 82
4.2.5.2 Synthèse vocale et eSpeak.....83
4.2.6 Autres fonctionnalités.....84
4.2.6.1 Boutons.....84
4.2.6.2 TactoColor.....85
4.3 Tutoriel 86
4.3.1 Objectifs d'apprentissage 87
4.3.2 Activités 88
4.4 Synthèse et contribution 90
CHAPITRE 5 ÉTUDE COMPARATIVE DE LA PERFORMANCE HUMAINE BASÉE SUR L'EXPLORATION SPATIALE VS L'EXPLORATION LINÉAIRE DE SITES WEB 92
5.1 Méthodologie 92
5.1.1 Participants 92
5.1.2 Tâches.....94
5.1.2.1 Tâches de recherche d'information 97
5.1.2.2 Tâches de remplissage de formulaire 99
5.1.3 Variables étudiées.....101
5.1.4 Procédure expérimentale 103
5.2 Recueil des données 105

5.3 Analyse des résultats 106

5.3.1 Analyses des tâches 106

5.3.1.1 Analyses globales 108

5.3.1.2 Analyse des tâches de recherche d'information 112

5.3.1.3 Analyse des tâches de remplissage de formulaire 121

5.3.2 Analyse des questions fermées du questionnaire 131

5.3.3 Analyse des questions ouvertes du questionnaire 136

5.3.3.1 Évaluation subjective de JAWS 136

5.3.3.2 Évaluation subjective de TactoWeb 138

5.3.3.3 Comparaison subjective des outils 140

5.4 Synthèse et contributions 141

CONCLUSION 146

BIBLIOGRAPHIE 153

ANNEXES 162

LISTE DES TABLEAUX

Tableau 5-1 : Caractéristiques des participants 94

Tableau 5-2 : Répartition des tâches en fonctions des variables indépendantes 96

Tableau 5-3 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution de l'ensemble des tâches en fonction de l'outil 110

Tableau 5-4 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches dans les sites accessibles en fonction de l'outil 110

Tableau 5-5 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches dans les sites non-accessibles en fonction de l'outil 111

Tableau 5-6 : Test-t pur échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution des tâches en fonction de l'apparition de la cécité chez les participants utilisant JAWS 112

Tableau 5-7 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution des tâches en fonction de l'apparition de la cécité chez les participants utilisant TactoWeb 112

Tableau 5-8 : Test-t pour échantillons appariées sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution dans les tâches de recherche d'information en fonction de l'outil 114

Tableau 5-9 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches de recherche d'information dans les sites accessibles en fonction de l'outil utilisé 116

Tableau 5-10 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour la tâche de recherche d'information S1 en fonction de l'outil utilisé 116

Tableau 5-11 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour la tâche de recherche d'information S2 en fonction de l'outil utilisé.....117

Tableau 5-12 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches de recherche d'information dans des sites non-accessibles en fonction de l'outil utilisé.....119

Tableau 5-13 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour la tâche de recherche d'information S3 en fonction de l'outil utilisé.....119

Tableau 5-14 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour la tâche de recherche d'information S4 en fonction de l'outil utilisé.....120

Tableau 5-15 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches de remplissage de formulaire en fonction de l'outil et de l'accessibilité.....123

Tableau 5-16 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution des tâches de remplissage de formulaire en fonction de l'apparition de la cécité chez les participants utilisant TactoWeb 124

Tableau 5-17 : Test-t pour échantillons appariés sur les moyennes des taux de succès et des temps d'exécution pour des tâches de remplissage de formulaires dans des sites accessibles en fonction de l'outil utilisé.....126

Tableau 5-18 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution de la tâche de remplissage de formulaire Q1 en fonction de l'outil utilisé.....126

Tableau 5-19 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution de la tâche de remplissage de formulaire Q2 en fonction de l'outil utilisé.....126

Tableau 5-20 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches de remplissage de formulaires dans des sites non-accessibles en fonction de l'outil.....129

Tableau 5-21 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour la tâche de remplissage de formulaire Q3 en fonction de l'outil 130

Tableau 5-22 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution de la tâche de remplissage de formulaire Q4 en fonction de l'outil 130

Tableau 5-23 : Test-t (échantillons appariés) sur les moyennes des réponses aux questions fermées du questionnaire portant sur le taux de satisfaction à l'égard de l'outil 133

LISTE DES FIGURES

Figure 1-1 : Exemples d'afficheurs Braille : (a) BrailleNote et (b) Brailliant 19

Figure 1-2 : L'homunculus de Penfield 23

Figure 1-3 : Divers outils tactiles : (a) le « Cat », (b) le VTD, (c) l'Optacon, (d) le « three-dimensional (3-D) tactile display device » et (e) le « Braille Window System ». 27

Figure 1-4 : Les actionneurs STReSS² (a) et le Tactograph (b) 31

Figure 1-5 : Expérimentation de MaskGen avec le Tactograph.....33

Figure 2-1 : Utilisation du rotor de VoiceOver sous Mac OS.....44

Figure 2-2 : Pavé tactile d'un MacBook (a), Magic TrackPad (b) et tablette graphique Wacom Intuos 4 (c).....45

Figure 2-3 : Utilisation du rotor sous iOS 47

Figure 4-1 : Seconde version du Tactograph (a) et image tactile adaptée d'un manuel scolaire (b) 59

Figure 4-2 : Nouvelle version du Tactograph sans la coquille du module tactile 60

Figure 4-3 : Maquette en pâte à modeler de la coquille du Latero.....61

Figure 4-4 : Premier modèle 3D (a) et modèle final 3D (b et c) de la coquille du module tactile du Tactograph.....62

Figure 4-5 : Version actuelle du Tactograph.....62

Figure 4-6 : Architecture de TactoWeb.....64

Figure 4-7 : Localisation d'un TactoElement par rapport à son parent dans le cas où le parent n'a qu'un enfant direct 72

Figure 4-8 : Localisation des enfants directs d'un TactoElement ayant une largeur fixe.....73

Figure 4-9 : Localisation des enfants directs d'un TactoElement n'ayant pas une largeur fixe 73

Figure 4-10 : Menu présenté sous forme de liste (a), son adaptation sans adaptation de largeur dans TactoWeb (b) et avec adaptation de largeur dans TactoWeb (c).....74

Figure 4-11 : Fonctionnement de la propriété CSS « float » en fonction de sa valeur : (a) « none », (b) « left » et (c) « right ». 75

Figure 4-12 : Modification des masques graphiques à partir d'un extrait de page Web.....77

Figure 4-13 : Création des rendus tactiles (d) à partir des masques graphiques (a, b et c). 80

Figure 4-14 : Page Web originale du programme d'étude en science de la nature du Collège André-Grasset de Montréal 81

Figure 4-15 : Représentation de la page Web du programme d'étude en science de la nature du Collège André-Grasset de Montréal dans TactoWeb.....81

Figure 4-16 : Différences entre TactoWeb (a) et TactoColor (b).....86

Figure 4-17 : Page Web du tutoriel (a) et son affichage dans TactoWeb (b).....89

Figure 4-18 : Page Web du tutoriel pour les formulaires (a) et son affichage dans TactoWeb (b)90

Figure 5-1 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b), et des temps d'exécution (c) en fonction de l'outil utilisé et de l'accessibilité des sites Web pour l'ensemble des tâches.....109

Figure 5-2 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b), et des temps d'exécution (c) pour les tâches de recherche d'information en fonction de l'outil et de l'accessibilité 113

Figure 5-3 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b) et des temps d'exécution (c) pour les tâches de recherche d'information dans des sites accessibles en fonction de l'outil 115

Figure 5-4 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b) et des temps d'exécution (c) pour les tâches de recherche d'information dans des sites non-accessibles en fonction de l'outil.....118

Figure 5-5 : Histogrammes présentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b) et des temps d'exécution (c) pour les tâches de remplissage de formulaire en fonction de l'outil et de l'accessibilité 122

Figure 5-6 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b) et des temps d'exécution (c) pour les tâches de remplissage de formulaire dans des sites accessibles en fonction de l'outil.....125

Figure 5-7 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b) et des temps d'exécution (c) pour les tâches de remplissage de formulaire dans des sites non-accessibles en fonction de l'outil 128

Figure 5-8 : Histogramme présentant les moyennes des réponses aux questions fermées du questionnaire sur la satisfaction à l'égard des deux outils 132

Figure 5-9 : Histogramme présentant les moyennes des réponses aux questions fermées spécifiques à TactoWeb.....136

LISTE DES SIGLES ET ABRÉVIATIONS

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
BANQ	Bibliothèque et archives nationales du Québec
CER	Comité d'Éthique de la Recherche
CRIR	Centre de Recherche Interdisciplinaire en Réadaptation
CSS	Cascading Style Sheet
DMLA	Dégénérescence Maculaire Liée à l'Âge
DOM	Document Object Model
DOS	Disk Operating System
HTML	HyperText Markup Language
IHM	Interfaces Humain-Machine
INLB	Institut Nazareth et Louis Braille
MATI	Maison des technologies de formation et d'apprentissage Roland-Giguère
OSM	OffScreen Mode
PDF	Portable Document Format
SGQRI	Standard du Gouvernement du Québec sur les Ressources Informationnelles
SMIL	Synchronized Multimedia Integration Language
STReSS²	Stimulator of Tactile Receptors by Skin Stretch
SVG	Scalable Vector Graphics
Test-t	Test-t de Student
URL	Uniform Resource Locator
W3C	World Wide Web Consortium

WAI	Web Accessibility Initiative
WCAG	Web Content Accessibility Guidelines
Web AIM	Web Accessibility in Mind
XHTML	eXtensible HyperText Markup Language
XML	eXtensible Markup Language
XSL	eXtensible Stylesheet Language

LISTE DES DÉFINITIONS

Audicône	Nom commun féminin. Son symbolisant un type élément, soit un type de TactoElement dans le cas de TactoWeb.
CSSContent	Représentation d'une règle CSS dans TactoWeb.
Élément Web	Représentation d'un élément composant une page Web (ex. : hyperlien, image, paragraphe, titre, champ de formulaire, etc.).
Padding	Marge interne d'un élément Web.
TactoElement	Représentation audio-tactile d'un élément Web dans TactoWeb.
Tactograph	Appareil tactile composé d'un module générant des sensations tactiles en fonction de sa position sur une surface d'exploration.
TactoWeb	Fureteur audio-tactile à l'usage des personnes non-voyantes.
Touches VO	Combinaison des touches « CTRL + ALT » utilisées dans VoiceOver.

LISTE DES ANNEXES

ANNEXE 1 – LISTE DES TACTOELEMENT 162

ANNEXE 2 – TUTORIEL TACTOWEB 166

ANNEXE 3 – ANNONCE DE RECRUTEMENT 171

ANNEXE 4 – CERTIFICAT D'ÉTHIQUE DE RECHERCHE 172

ANNEXE 5 – QUESTIONNAIRE DE RECRUTEMENT 174

ANNEXE 6 – IMPRESSION D'ÉCRAN ET RENDUS DANS TACTOWEB POUR LES TÂCHES EXPÉRIMENTALES 175

ANNEXE 7 – QUESTIONNAIRE POST-EXPÉRIMENTATION 187

INTRODUCTION

La perception du monde qui nous entoure se fait majoritairement par la vision. La répartition de la perception de notre environnement entre nos cinq sens est déséquilibrée. Les auteurs ne s'entendent d'ailleurs pas sur les chiffres exacts de cette répartition. Certains énoncent que 80% de ce que nous percevons se fait par la vision (Seiderman, Marcus, & Hapgood, 1989) et d'autres 83% (Rosenblum, 2011). Mais tous s'accordent sur le fait que plus des trois quarts de la perception du monde qui nous entoure se fait par l'intermédiaire de notre vision alors que nos quatre autres sens se partagent l'information restante. On dit qu'une image vaut mille mots. Que se passe-t-il lorsque nous sommes privés de cette image? Que se passe-t-il lorsque nous sommes privés de notre sens le plus important pour percevoir notre environnement?

La privation de la vision entraîne de nombreuses situations de handicap, que ce soit dans la vie de tous les jours ou dans la vie professionnelle. L'accès à l'information est une de ces situations de handicap. L'information est de plus en plus numérique. Accéder numériquement à de l'information se fait principalement en ligne par l'intermédiaire d'un ordinateur ou d'un appareil mobile, les yeux figés sur l'écran. Nous naviguons sur nos sites Web favoris et consultons nos courriels tous les jours. Nous effectuons aussi de plus en plus d'achats en ligne. L'accès à l'information en ligne nous semble essentiel. Mais s'il l'est, il devrait l'être pour tout le monde, indépendamment des contraintes et des limitations que peuvent avoir les utilisateurs. Cette préoccupation nous a amenés à nous intéresser au domaine de l'accessibilité en général, et à l'accessibilité du Web en particulier. Même s'il est impossible d'éliminer totalement les situations de handicap, il faut au moins tenter d'en réduire le nombre et la gravité.

Au cours des dernières années, notre intérêt pour l'accessibilité s'est concentré sur la situation des personnes non-voyantes. Avant de travailler sur cette thèse, nous travaillions déjà sur un projet de recherche impliquant des personnes non-voyantes. Il s'agissait de l'accès aux images de manuels scolaires pour des élèves non-voyants (Petit, Dufresne, Lévesque, & Hayward, 2008). Cet accès se faisait grâce des graphiques rafraîchissables qu'on pouvait explorer à l'aide d'un appareil tactile et sonore. Au cours d'une expérimentation servant à tester cet appareil, un des élèves a indiqué que cela serait « cool » d'utiliser le même appareil pour naviguer sur Internet, afin d'avoir une expérience proche de celle des personnes voyantes. C'est de là que l'idée de cette thèse a germé.

Les personnes non-voyantes ont accès au monde numérique grâce à des logiciels d'adaptation appelés lecteurs d'écran. Ces derniers leur permettent entre autres de naviguer sur des sites Web et d'en explorer les pages en lisant le contenu de l'écran comme si on lisait un livre. L'apprentissage de cette technologie de réadaptation peut être fastidieux, surtout pour les personnes non-voyantes tardives. En plus d'apprendre à se servir d'un clavier sans retour visuel, la façon d'explorer les sites Web change complètement puisque le lecteur d'écran présente l'information de façon linéaire. L'utilisateur est alors obligé de suivre un même chemin pour accéder au contenu de la page Web, sans avoir accès à la structure spatiale de la page. Cette linéarisation peut compliquer l'assimilation de l'information car les sites Web sont conçus pour être explorés de façon spatiale (selon deux dimensions) sur un écran. La frustration engendrée par l'utilisation de ces lecteurs d'écran a d'ailleurs été mentionnée dans plusieurs études (Lazar, Allen, Kleinman, & Malarkey, 2007; Leporini & Paternò, 2004; Web AIM, 2013). Le fait de ne pas avoir le même accès que les personnes voyantes était ce que l'élève cité plus haut reprochait aux logiciels de lecture d'écran. Comment donner le même accès? Nous nous sommes basés sur l'idée de cet élève pour tenter de répondre à cette question : utiliser un appareil tactile et sonore pour explorer spatialement des pages Web.

Le but de cette recherche est de savoir si l'exploration spatiale des pages Web est possible pour les personnes non-voyantes, et dans ce cas, si elle donne un meilleur accès à l'information que l'exploration linéaire proposée par les lecteurs d'écran. Cette thèse propose un outil permettant d'évaluer le potentiel de l'exploration spatiale à améliorer l'accès au monde du Web pour les personnes non-voyantes. Cela améliorerait grandement l'expérience utilisateur des personnes non-voyantes en termes de temps d'apprentissage et de satisfaction. Pour réaliser ce but, nous avons conçu, développé et évalué un système tactile et sonore permettant l'exploration spatiale de pages Web. Ce système est composé d'un logiciel appelé TactoWeb, faisant office de fureteur audio-tactile, et d'un appareil tactile appelé Tactograph, permettant de déplacer le curseur dans le fureteur.

Cette thèse est structurée en cinq chapitres. Le premier chapitre présente une revue de littérature sur les trois domaines pertinents à notre recherche. Le premier domaine, le plus général, porte sur les interfaces humain-machine. Comme le système que nous développons comporte une interface multimodale, il était important de comprendre les différentes caractéristiques de ce type d'interface. De plus, il fallait tester son utilisabilité auprès des utilisateurs, d'où une section sur ce

sujet. Le second domaine porte sur l'accessibilité de l'information aux personnes non-voyantes, notamment lors de l'exploration de pages Web. Il s'agit du principal domaine de notre recherche. Le troisième domaine porte sur les interactions tactiles, en se concentrant sur le sens du toucher, puis sur les types d'outils générant des retours tactiles.

Le deuxième chapitre présente une étude empirique que nous avons réalisée sur les technologies adaptées pour le Web, à l'usage des utilisateurs non-voyants. Nous nous sommes intéressés en particulier à JAWS, le lecteur d'écran le plus utilisé par les personnes non-voyantes, ainsi qu'à VoiceOver, un lecteur d'écran plus récent proposant aussi l'exploration spatiale.

Le troisième chapitre présente la problématique de la thèse, les objectifs et nos hypothèses de recherche.

Le quatrième chapitre présente le processus de développement de notre système permettant l'exploration spatiale à l'aide de retours tactiles et sonores. C'est la partie de la thèse qui a nécessité le plus de temps de travail. Nous présentons les logiciels et le matériel utilisés. Nous décrivons le processus de conception de notre fureteur TactoWeb ainsi que ses fonctionnalités. Afin de permettre aux utilisateurs de se familiariser avec l'exploration spatiale, nous avons aussi développé un tutoriel.

Le cinquième chapitre présente la validation de notre système avec des personnes non-voyantes. Pour faire cette validation, nous avons comparé la performance et la satisfaction humaine lorsqu'il y a exploration spatiale avec TactoWeb et le Tactograph, et lorsqu'il y a exploration linéaire avec JAWS. Le chapitre présente la méthodologie et les résultats de cette étude.

Enfin, dans la conclusion, nous montrons les différentes contributions de notre recherche à l'avancement des connaissances et nous proposons quelques pistes de recherche pour la suite de notre recherche.

CHAPITRE 1 REVUE DE LITTÉRATURE SUR LES INTERFACES HUMAIN-MACHINE, L'ACCESSIBILITÉ POUR LES UTILISATEURS NON-VOYANTS ET L'INTERACTION TACTILE

Avant de nous plonger dans n'importe quelle recherche, il est important de faire le tour des différents domaines qui lui sont associés. Dans ce chapitre, nous présentons ainsi certains concepts, théories, méthodes et travaux relatifs à notre recherche. Nous allons tout d'abord nous concentrer sur le domaine des Interfaces Humain-Machine (IHM) dont fait partie le thème de l'accessibilité dans la section 1.1. Il est important d'avoir des notions globales sur les IHM avant de nous focaliser sur l'accessibilité. Dans la section 1.2, nous nous concentrons sur l'accessibilité des IHM, tout particulièrement du Web, pour les utilisateurs non-voyants qui est le domaine principal de notre recherche. Enfin, nous nous intéressons dans la section 1.3 aux interactions tactiles qui représentent le moyen de répondre à notre problématique.

1.1 Interfaces Humain-Machine

Le domaine des IHM est très vaste. Il est important de s'attarder sur certains concepts essentiels pour poser les fondements de ce projet de recherche.

1.1.1 Utilisabilité

Le premier concept des IHM auquel nous allons nous intéresser est l'utilisabilité. Il fut abordé la première fois par Gould et Lewis en 1981 (Gould & Lewis, 1985). Mais c'est depuis les travaux de Nielsen (Nielsen, 1993), considéré comme un expert mondial de l'utilisabilité, que l'utilisabilité a commencé à prendre de l'importance. La norme standard ISO 9241-11 (Standardization, 1998) a été mise en place en 1998 et sert de définition officielle à l'utilisabilité :

> *L'utilisabilité est le degré selon lequel un produit peut être utilisé, par des utilisateurs identifiés, pour atteindre des buts définis, avec efficacité, efficience et satisfaction, dans un contexte d'utilisation spécifié.*

Pour mesurer l'utilisabilité d'un produit, on peut regarder les trois critères énoncés dans la définition ci-dessus : l'efficacité, l'efficience et la satisfaction.

L'efficacité se rapporte à la complétion des objectifs fixés par l'utilisateur lorsqu'il effectue une activité (Baccino, Bellino, & Colombi, 2005). Elle ne peut donc pas être mesurée si les objectifs n'ont pas été clairement définis préalablement. Elle peut se mesurer par le taux de réussite de l'activité ou le taux d'erreurs.

L'efficience se rapporte à la capacité cognitive sollicitée lors de la production d'une tâche (Baccino et al., 2005). En effet, plus l'effort cognitif demandé pour la tâche est important, plus l'efficience est faible. On peut l'évaluer en étudiant la charge de travail demandée pour la complétion de la tâche. L'efficience se mesure par la vitesse d'exécution de la tâche et par le nombre d'étapes requises pour effectuer cette tâche. Un autre élément est toutefois important à prendre en compte lors de la mesure de l'efficience. L'efficience est très liée à la volonté de réussite de l'utilisateur. En effet, celui-ci peut décider d'allouer moins de ressources cognitives à une tâche s'il considère que celle-ci n'a pas besoin d'être complétée parfaitement. Cet aspect de volonté de l'utilisateur rend ce critère de l'utilisabilité plus difficile à mesurer.

La satisfaction correspond au niveau de confort et au plaisir de l'utilisateur lorsqu'il effectue une activité (Baccino et al., 2005). Elle implique une réaction affective lors de l'utilisation d'un outil. Cela ouvre la porte à la prise en compte des émotions de l'utilisateur. Ce critère est difficile à évaluer à cause de sa nature subjective et c'est pourquoi, au moyen d'entrevue ou de questionnaire, on demande à l'utilisateur de rendre compte de ce qu'il ressent au cours d'une activité. On peut aussi estimer le niveau de confort et le plaisir de l'utilisateur en mesurant les signaux physiologiques produits par ce dernier. Ces mesures sont obtenues de manière continue et en temps réel, sans interrompre l'interaction entre l'utilisateur et le système (Ganglbauer, Schrammel, Deutsch, & Tscheligi, 2009). Même si cette méthode est objective, elle est assez intrusive pour l'utilisateur et les signaux physiologiques restent complexes à interpréter.

La mesure de l'utilisabilité est essentielle dans notre projet de recherche ; en effet, elle va nous permettre de comparer l'efficacité de notre système par rapport aux systèmes existants en tenant compte du niveau de qualité des interfaces utilisateurs de ces systèmes.

1.1.2 Interaction multimodale

Le second concept des IHM nous intéressant pour notre recherche est l'interaction multimodale. D'après Nigay et Coutaz (Nigay & Coutaz, 1995), une interaction implique un couple (p, L), soit

un périphérique p et un langage d'interaction L. Le périphérique est l'artéfact du système qui envoie et/ou retourne une information (Nigay, 2004). Par exemple, une souris et un clavier sont des périphériques d'entrée, un écran et des hautparleurs sont des périphériques de sortie, et un appareil à retour d'effort, comme un contrôleur Playstation3 (Sony, 2013) ou l'appareil haptique Phantom Omni (Sensable, 2013), est un périphérique d'entrée et de sortie en même temps. Quant au langage, il s'agit d'un ensemble d'expressions devant respecter des règles. Les entrées et sorties de ces règles résultent de l'interaction du système avec des périphériques (Nigay & Coutaz, 1995).

Nous avons cinq modalités sensorielles : la vue, l'ouïe, l'odorat, le toucher et le goût. Ce sont, en quelque sorte, les périphériques des êtres humains car ce sont eux qui nous permettent de percevoir (entrée) et d'agir (sortie). La vision est la modalité perceptive la plus développée chez l'homme. Par contre, lorsque nous voulons agir avec le monde extérieur, la principale modalité utilisée est le toucher.

Une interaction multimodale contient au moins deux modes d'interactions entrantes et différentes, coordonnées dans le système (Oviatt, 2003) ; ce qui implique l'utilisation d'au moins deux modalités différentes. Avec le développement des nouvelles technologies, les interfaces multimodales sont en continuelle progression. Elles permettent d'améliorer l'expérience utilisateur et l'utilisabilité des interfaces (Neto, Bittar, Fortes, & Felizardo, 2009). Elles expriment de manière très fidèle les traits de l'interaction humain-machine car elles se greffent directement au comportement naturel de l'être humain (Oviatt, 2003). Les premières interfaces multimodales combinaient le pointage et la parole (« Put That Here », (Bolt, 1980)) pour donner une alternative au contrôle classique clavier/souris. Des recherches ont aussi été faites pour associer la parole, combinaison de phonèmes, avec les mouvements des lèvres, des visèmes, dans le but de créer des avatars encore plus proches de la réalité. À la fin des années 90, on a commencé à s'intéresser au regard, aux gestes et aux expressions faciales pour rendre l'interaction encore plus naturelle et mobile. Enfin, des interfaces multimodales impliquant plus de deux modalités ont fait leur apparition, notamment grâce à l'émergence de la biométrie.

Les interfaces multimodales comportent de nombreux avantages (Oviatt, 2003). Elles sont flexibles car on peut choisir et/ou combiner plusieurs types d'entrées. Le choix de ces modalités est devenu très important pour éviter l'abus de l'usage d'une seule modalité donc sa saturation,

comme cela peut être le cas avec les interfaces graphiques. En effet, les interfaces graphiques, dont le Web pour l'instant, sont des interfaces unimodales ce qui rend la communication non-naturelle (Neto et al., 2009). Les tâches effectuées avec une interface multimodale paraissent plus rapides et plus simples par rapport à une interface unimodale (Oviatt, 2003), une interface graphique par exemple, car les commandes y sont plus directes (Oviatt & Kuhn, 1998).

Les interfaces multimodales sont efficaces pour gérer et prévenir les erreurs. En effet, l'utilisateur peut choisir la modalité avec laquelle il est le plus à l'aise et le moins enclin à faire des erreurs. De plus, le système peut aussi inférer sur ce que fait l'utilisateur pour rendre des situations moins ambigües. Il retrouvera ainsi le bon sens d'une modalité en se référant au contexte indiqué naturellement par l'interaction multimodale (Oviatt, 2003). Une interface multimodale peut aussi être utile dans le cas où une personne est dans l'incapacité d'utiliser un de ses sens, ce qui est le cas pour notre recherche.

Les informations véhiculées par les différentes modalités peuvent être redondantes pour cibler plus de types d'utilisateurs. Mais la majorité des interfaces multimodales véhicule des informations complémentaires pour chacune des modalités pour rendre l'interaction plus naturelle (Oviatt, 2003). Pour développer des interfaces multimodales, on peut se baser sur l'approche de Stanciulescu (Stanciulescu, Limbourg, Vanderdonckt, Michotte, & Montero, 2005), valable en fait pour n'importe quelle IHM. Celui-ci propose de diviser le cycle de vie d'une interface en quatre étapes : le modèle de la tâche, le modèle du domaine, le modèle de l'interface abstraite et le modèle de l'interface concrète. Ces modèles, leurs éléments, et leur transformation sont spécifiés grâce à un fichier XML, puis modélisés graphiquement.

Il est important de tenir compte des sciences cognitives lorsqu'on veut produire une interface multimodale (Oviatt, 2003). En effet, il faut pouvoir coordonner correctement les différentes modalités sensorielles pour ne pas provoquer de problème dans la perception inter-sensorielle. Les vitesses des retours sensoriels doivent être respectées pour donner l'impression que le système est complètement multimodal et obtenir une simulation de plus en plus réelle. Ceci est très important pour que l'interaction multimodale soit la plus proche d'une communication interpersonnelle naturelle, qui est toujours multimodale (paroles, gestes, etc.). Néanmoins, même si l'utilisateur préfère globalement la multimodalité, il n'est pas certain qu'il l'utilise tout le

temps. En effet, il peut alterner entre multimodalité et unimodalité, en fonction du type d'action qu'il doit effectuer (Oviatt, DeAngeli, & Kuhn, 1997).

Un préjugé courant à l'égard les interfaces multimodales est que les modalités sensorielles doivent être exactement synchronisées, alors que ce n'est pas le cas dans la réalité car les gestes précèdent la parole, les lèvres précèdent légèrement la parole, etc. Et ceci est encore plus faux pour les interfaces multimodales (Oviatt, 2003). Il faut étudier la cascade temporelle des évènements et les différences individuelles dans la façon d'agir, pouvant être liées à la culture, pour identifier la façon d'agir dominante. C'est là un des principaux défis à relever pour développer une interface multimodale. Un autre défi est d'en abaisser le coût. En effet, l'ajout de la multimodalité à une interface rend souvent son coût beaucoup plus élevé car les périphériques sont généralement plus chers que les périphériques utilisés pour les interfaces graphiques. De plus, le coût d'un système est bloquant pour n'importe quelle technologie voulant se démocratiser.

Enfin, Dufresne et al. ont pu comparer l'utilisation de la multimodalité (sonore et tactile) à l'utilisation séparée de ces deux modalités (Dufresne, Martial, & Ramstein, 1995). Les participants non-voyants devaient effectuer des tâches basiques de manipulation d'interfaces graphiques. Au cours de cette étude, les tâches ont clairement été mieux réussies lors de l'emploi des deux modalités en même temps (sonore et haptique), surtout par rapport à l'utilisation de la modalité sonore seule. Ces résultats s'appliquent directement à notre recherche car le Web est aussi une interface graphique. Ainsi, l'utilisation combinée des modalités tactile et sonore est plus efficace qu'une interaction strictement sonore, comme c'est le cas pour les lecteurs d'écran grandement utilisés par les utilisateurs non-voyants pour accéder au Web. Néanmoins, ces lecteurs d'écrans donnent lieu à une exploration linéaire alors que l'expérimentation de Dufresne et al. donnait lieu à une exploration spatiale.

1.2 Accessibilité aux utilisateurs non-voyants

Penchons-nous maintenant sur un aspect beaucoup moins répandu dans les IHM et sur lequel notre recherche est principalement axée : l'accessibilité, tout particulièrement pour les utilisateurs non-voyants.

Si on prête attention aux interfaces actuelles pour le grand public, on constate que la modalité visuelle y est omniprésente. En effet, les interfaces modernes sont toutes graphiques et se

retrouvent dans beaucoup de systèmes interactifs que nous utilisons depuis le début des années 80 (Edwards, 2008; Smith, Irby, Kimball, Verplank, & Harslem, 1989), y compris la grande majorité des sites Web (Barreto, 2008). Notre interaction avec les ordinateurs est donc basée sur le retour visuel que les écrans nous fournissent. Les alternatives à ce type d'interaction sont peu présentes et rarement assez démocratisées pour toucher le grand public.

1.2.1 Handicap visuel

La vision est la modalité perceptive la plus utilisée par l'être humain. Elle nous permet d'observer le monde qui nous entoure ainsi que les objets et les êtres vivants qui en font partie. Elle nous permet aussi d'avoir une idée des contraintes de l'environnement, comme la position et la distance entre les objets, la luminosité, etc. Ce sens prend tellement d'importance dans notre système sensoriel, qu'il peut influencer les autres sens (Massaro, 1998). On peut se demander comment une personne ayant perdu l'utilisation de la vue peut toujours interagir activement avec le monde qui l'entoure. On parle d'ailleurs généralement de situation de handicap, et non pas d'une personne ayant un handicap.

La perte de la vision affecte un grand nombre de personnes et peut prendre différentes formes en fonction de son origine et de son impact sur l'acuité et le champ visuel (Hollins, 1989; Warren, 1978). En 2006, on estimait que le nombre de personnes atteintes de cécité complète au Canada était de 0,24% (75 000 personnes) de la population et que 0,71% (220 000 personnes) des Canadiens possédaient une vision réduite, donc étaient malvoyants (voir « 1.2.1.2 Le handicap visuel ») (Maberley et al., 2006). Il y avait donc 295 000 personnes non-voyantes ou malvoyantes au Canada; ces chiffres sont loin d'être négligeables.

1.2.1.1 La perception visuelle

Trois systèmes de l'humain nous permettent d'utiliser des IHM (Card, Newell, & Moran, 1983) :

- Le système perceptif : réception du message venant de la machine;
- Le système moteur : contrôle de l'action de l'utilisateur;
- Le système cognitif : représentation et connexion entre les deux autres systèmes.

Ces trois systèmes doivent être opérationnels pour qu'on puisse naviguer normalement sur le Web. Mais si un de ces systèmes est défaillant, il faut pouvoir donner une alternative à l'utilisateur pour qu'il puisse tout de même accéder au Web.

La vision est très sollicitée par le Web actuel. En effet, l'information est majoritairement accessible via le canal visuel à partir de points lumineux, les pixels, sur un écran. Ce stimulus parvient à l'œil à travers la cornée, la lumière se distribue ensuite sur la rétine, puis génère un signal électrique qui est acheminé jusqu'au cerveau.

La lumière est courbée en deux étapes (Barreto, 2008). La première réfraction a lieu dans la pupille lorsque la lumière passe de l'air à la cornée, grâce au cristallin. Ce dernier modifie sa forme grâce au muscle ciliaire. La seconde a lieu dans l'humeur aqueuse qui doit être la plus sphérique possible pour que la lumière ne soit pas trop déviée. La lumière va être de plus en plus déviée si l'objet est de plus en plus proche.

L'image rétinienne se forme ensuite de manière très complexe (Barreto, 2008). La lumière est distribuée depuis l'écran et projetée sur la rétine contenant des photorécepteurs (PR) : les cônes et les bâtonnets. Un point sur l'écran correspond à un point sur la rétine, mais à une plus petite échelle et l'image est inversée horizontalement et verticalement. Si un point n'est pas capté, on obtient l'illumination d'une zone étendue ce qui provoque une zone de floue.

Chaque type de cônes est sensible à une seule région spectrale (bleu : 450 nm de longueur d'onde; vert : 575 nm de longueur d'onde; rouge : 600 nm de longueur d'onde).

La perception visuelle a quatre fonctions majeures :

- L'acuité visuelle : capacité à distinguer deux points différents proches l'un de l'autre et situés le plus loin possible. La norme est de 25 secondes d'arc. Une personne ayant une acuité visuelle de 20/20 sur l'échelle de Snellen a une vision parfaite. Une personne ayant une acuité de 20/30 doit se mettre à 20 pieds pour voir correctement ce qu'elle devrait normalement voir à 30 pieds (Martini & Nath, 2008).
- La sensibilité au contraste : capacité à discerner les subtiles différences de niveau de gris (Ginsburg & Hendee, 1993). Cela peut s'évaluer grâce à l'échelle de Pelli-Robson (Pelli, Robson, & Wilkins, 1988).

- Le champ visuel : aire visuelle vue par un œil à un instant t. Cela se mesure par périmétrie, où on amène doucement une source lumineuse vers le champ visuel et on identifie quand l'utilisateur la voit (Guyton & Hall, 2006).
- La perception des couleurs. Elle peut se mesurer à l'aide de planches pseudoisochromatiques ou avec un test d'ordonnancement de Farnsworth.

Les développeurs exploitent au maximum les résolutions, les tailles et les couleurs disponibles. Cela implique une large variété de conditions pouvant provoquer une baisse de l'accessibilité, surtout s'il y a handicap visuel.

1.2.1.2 Le handicap visuel

En premier lieu, il est important de définir la terminologie qui sera employée tout au long de cette thèse concernant le handicap visuel.

Il est très commun d'avoir une réfraction imparfaite de la lumière (myopie ou hypermétropie) ou d'avoir des difficultés à faire la mise au point sur certains objets avec l'arrivée de l'âge (presbytie). En effet, passé 60 ans, on peut avoir du mal à faire la mise au point sur des objets plus près que 83 cm alors qu'on peut le faire pour des objets à 15-20 cm lorsqu'on est plus jeune. Dans ce cas-ci, on parle de baisse de vision qui peut être totalement corrigée par des verres correcteurs (lunettes ou verres de contact).

Dans notre recherche, nous allons nous intéresser à la population ayant subi la perte de la vision et que nous dénommons « utilisateurs non-voyants ». L'*American Foundation for the Blind* définit le terme « perte de la vision » comme étant un handicap visuel persistant, même lors de l'utilisation de verres correcteurs (American Foundation for the Blind, 2013). Cette définition s'applique aussi au terme « vision réduite » que l'on retrouve dans certaines littératures (Maberley et al., 2006). Le terme « cécité » se rapporte à une acuité visuelle inférieure à 20/200 sur une échelle de Snellen ou à un champ de vision inférieur à 20°, et ceci pour les deux yeux (Maberley et al., 2006). Dans ces cas, on parle de personnes « fonctionnellement aveugles ». Il existe aussi des personnes qui ne sont pas fonctionnellement aveugles mais qui ont tout de même besoin d'outils d'adaptation. Ces personnes nommées « malvoyantes » ont une acuité visuelle inférieure à 20/70, avec la meilleure correction possible, ou une perte de 140° dans le champs de vision (American Foundation for the Blind, 2013).

D'après l'*Organisation mondiale de la santé*, les trois causes principales de handicap visuel sont la rétinopathie diabétique, le glaucome et la DMLA (Dégénérescence Maculaire Liée à l'Âge) (World Health Organization, 2013). Il est intéressant de souligner que deux de ces trois causes n'interviennent pas à la naissance ce qui expliquerait le fait qu'il y ait plus de personnes ayant un handicap visuel tardif que de naissance (Maberley et al., 2006).

La rétinopathie diabétique est un problème de rétine lié au diabète. Sa forme la plus grave provoque l'apparition de vaisseaux sanguins anormaux sur la rétine. Ces vaisseaux se brisent facilement ce qui provoque des hémorragies dans la rétine (Association des médecins ophtalmologistes du Québec, 2013c). Le glaucome se caractérise par une trop grande pression à l'intérieur de l'œil. Cela provoque une atrophie du nerf optique et cela cause une perte conséquente du champ visuel (Association des médecins ophtalmologistes du Québec, 2013d). La DMLA se caractérise par la dégénérescence de la macula avec l'âge. La macula est la partie centrale de la rétine et la plus sensible. C'est elle qui permet la vision de précision. Sa dégénérescence provoque une distorsion des lignes droites dans la vision centrale ce qui la rend floue (Association des médecins ophtalmologistes du Québec, 2013b). Enfin, une autre cause de handicap visuel importante est la cataracte. Celle-ci se manifeste par une opacification du cristallin ce qui provoque une baisse de la luminosité de la vision qui peut devenir totale (Association des médecins ophtalmologistes du Québec, 2013a).

1.2.1.3 Image mentale et handicap visuel de naissance ou tardif

Il est aussi important de différencier le handicap visuel de naissance (congénital) et le handicap visuel tardif car cela peut influencer la façon dont les personnes non-voyantes vont interagir avec les outils d'adaptation. Dans le cas d'un handicap visuel de naissance, l'utilisateur n'aura jamais eu conscience de l'apport de la vision. Son système sensoriel est seulement composé des quatre autres sens primaires et sa représentation mentale du monde qui l'entoure est différente par rapport à une personne disposant de ces cinq sens. Cet utilisateur aura grandi avec son handicap et avec les technologies d'adaptation appropriées et sera plus à l'aise avec elles. Néanmoins, une personne ayant un handicap visuel de naissance ne peut pas se faire une image mentale d'un objet au premier sens du terme, c.-à-d. avoir une représentation visuelle exacte de cet objet en tête. Elle se fait tout de même une « image mentale » qui peut être plus ou moins déformée de la réalité en se servant de ses autres sens, en particulier le sens du toucher. Cette image mentale haptique

(Hollins, 1989) permet à une personne non-voyante de naissance d'avoir une idée de la forme, de la taille et de la localisation d'un objet dans l'espace.

Pour une personne ayant un handicap visuel apparu tardivement, la représentation mentale d'un objet est différente d'une personne ayant un handicap visuel de naissance : elle est beaucoup plus similaire à celle d'une personne voyante. En effet, celle-ci a été privée d'un sens qu'elle avait l'habitude d'utiliser. La représentation mentale du monde qui l'entoure n'est donc pas vraiment altérée et fonctionne comme si elle n'avait aucun problème de vision car elle est née ainsi. Pour ces utilisateurs, l'adoption et l'utilisation des technologies adaptées sont généralement plus difficiles. Néanmoins, ils sont toujours capables de se faire une image mentale d'un objet, même des décennies après avoir perdu la vue (Hollins, 1989). Certains peuvent même se faire une image mentale d'un objet qu'ils n'auraient jamais vu si on leur décrit correctement. Ils peuvent aussi avoir une image mentale lorsqu'ils rêvent. Mais la faculté de se faire une image mentale diminue tout de même avec le temps pour être remplacée au fur et mesure par une image mentale haptique (Hollins, 1989).

1.2.2 Technologie adaptée

Dans cette section nous nous intéressons à l'accessibilité du Web pour les personnes non-voyantes.

1.2.2.1 Technologies relatives au Web et accessibilité

Le Web et les nouvelles technologies sont en continuel changement (Kirkpatrick, 2006). En partie à cause de cela, l'accessibilité pour les non-voyants est souvent laissée de côté ce qui frustre les utilisateurs ainsi que les développeurs. Les moyens alternatifs d'accès sont très rares.

Beaucoup de technologies constituent maintenant le Web. Les principales sont l'HTML et le XHTML (eXtensible HyperText Markup Language) gérant le contenu et la structure de la page Web. En ce qui concerne le HTML, l'accessibilité a fait son apparition en 1997 avec HTML 3.2. Ce dernier a l'avantage d'être bien compatible avec la navigation clavier ainsi qu'avec les logiciels de lecture d'écran (JAWS depuis février 2002, Window-Eyes depuis septembre 2003 et Home Page Reader depuis le printemps 2001). XHTML est une version moins permissive au niveau de la syntaxe que HTML mais reste sensiblement très similaire. Il est amené à disparaitre en 2014 avec la sortie d'HTML5 (W3C, 2013a).

Le CSS (Cascading Style Sheet) gère en partie la présentation des pages Web. D'ailleurs, avec l'apparition de HTML5, on s'attend à ce que toute la présentation soit gérée par CSS car beaucoup d'attributs et paramètres vont disparaitre de code HTML(W3C, 2013b). CSS est très utile pour l'accessibilité car il affecte l'apparence sans toucher au contenu (Edwards, 2008; Kirkpatrick, 2006). Un seul fichier CSS permet de gérer un site entier. Le code CSS est composé de plusieurs règles. Une règle CSS peut s'appliquer à un type de balise HTML, un identifiant, une classe, ou n'importe quelle combinaison de ces trois paramètres.

CSS est compatible avec la fonctionnalité des navigateurs Web gérant la taille des polices. De plus, l'utilisateur peut choisir d'utiliser son propre fichier CSS pour n'importe quel site Web. Il fonctionne très bien avec les lecteurs d'écran et est maintenant compatible avec le format Flash et les fichiers vectoriels (SVG). Il faut néanmoins faire attention à ce que le CSS ne soit utilisé que pour sa fonctionnalité principale : la mise en page et le formatage. Certaines propriétés CSS permettent aussi d'ajouter du contenu qui n'est pas accessible avec les technologies adaptées actuelles (voir Chapitre 2).

Le JavaScript gère en partie le comportement de la page Web. Il est de plus en plus utilisé pour créer du contenu et des interfaces. AJAX (Asynchronous JavaScript and XML) en est la parfaite illustration avec l'arrivée du Web 2.0. Néanmoins, cela rend des interfaces dynamiques moins accessibles (Kirkpatrick, 2006). Les nouvelles générations de lecteurs d'écran gèrent de mieux en mieux le JavaScript mais celui-ci pose encore des problèmes pour signaler que le contenu de la page a changé. De plus, la navigation au clavier est souvent abandonnée avec cette technologie. Pour les anciennes générations de lecteurs d'écran, il vaut mieux proposer du contenu alternatif via la balise <noscript>.

En ce qui concerne le Flash, celui-ci a longtemps été inaccessible. Mais depuis l'arrivée de Flash 6, de grands progrès ont été faits pour l'accessibilité, notamment avec Flash 8 (Kirkpatrick, 2006). Flash est néanmoins très utile pour les utilisateurs ayant des problèmes d'apprentissage car il permet une expérience dynamique et multi-sensorielle. Flash supporte la navigation au clavier, les légendes (sous-titres), l'ajustement du contraste et de la taille de la police mais cela reste assez difficile à implémenter pour les développeurs.

En ce qui concerne PDF (Portable Document Format), ce format peut être accessible si le fichier a été correctement généré, ce qui n'est pas forcément le cas avec les logiciels de création de fichiers PDF autres que celui d'Adobe.

Le SMIL (Synchronized Multimedia Integration Langage) est considéré comme une technologie accessible (Kirkpatrick, 2006). C'est une architecture de présentation multimédia gérant la localisation d'éléments et leur occurrence dans le temps. Il supporte les lecteurs de vidéos RealPlayer et QuickTime Player. Il aide beaucoup l'accessibilité même si ce n'est pas son but original. Il permet d'ajouter des sous-titres, des descriptions sonores, des sources audio alternatives. Cela permet d'avoir la même présentation sous plusieurs versions.

La majorité des lecteurs vidéo permettent d'inclure des sous-titres, des avatars, des images et des pistes sonores secondaires. Ils ont souvent leur propre format pour les sous-titres ce qui ne favorise pas la compatibilité, même si la principale difficulté reste la synchronisation de ces sous-titres. Ils supportent généralement bien le contrôle au clavier.

Les images vectorielles commencent tout doucement à se démocratiser. Les fichiers SVG ont de grandes possibilités mais elles sont souvent mal exploitées (Kirkpatrick, 2006). Il est possible d'ajouter des descriptions alternatives à ce genre de fichier mais aucune technologie assistée ne permet d'y accéder.

La technologie Java peut aussi être accessible en utilisant l'API (Application Programming Interface) Accessibility API ainsi que l'outil Java Access Bridge, permettant de rendre les informations accessibles.

1.2.2.2 Lignes directrices pour rendre le contenu Web accessible

Le contenu Web à rendre accessible contient le texte, les images et les parties audio et vidéo des sites Web (Thatcher, 2006a). En effet, une personne ayant un handicap doit pouvoir accéder au contenu entier de la page Web. Pour ce faire, la page doit être compatible avec les technologies d'assistance. Plusieurs lignes directrices ont été définies dans ce but par diverses organisations.

La WAI (Web Accessibility Initiative) du W3C (World Wide Web Consortium) a mis en place les lignes directrices WCAG (Web Content Accessibility Guidelines). Dans sa version 1.0, WCAG contient 14 lignes directrices incluant 65 points de contrôle ayant trois priorités différentes allant de la plus stricte (P1) à la plus souple (P3) (W3C, 2013c). Mais récemment, la

version 2.0 de WCAG est sortie avec 13 lignes directrices à suivre (W3C, 2013d). Mais cette fois, elles sont classées par importance du processus de développement à faire en plus de la page Web pour valider la ligne. Le niveau 1 correspond au niveau minimum d'accessibilité qui doit être applicable facilement alors que le niveau 3 correspond à un niveau additionnel qui n'est pas obligatoirement applicable.

Une autre organisation, l'US ACCESS BOARD, a mis en place des lignes directrices (la Section 508) qui sont obligatoires pour les technologies d'informations numériques du gouvernement fédéral américain (Section 508, 2013). La Section 508 contient 16 clauses dont beaucoup sont semblables à WCAG 1.0.

Enfin, le gouvernement du Québec a adopté en 2011 ses propres lignes directrices concernant l'accessibilité des sites Web : SGQRI (Standard du Gouvernement du Québec sur les Ressources Informationnelles) (Secrétariat du Conseil du Trésor, 2013). Celles-ci doivent être appliquées dans tous les sites publiques du Québec, ainsi que ceux du réseau de l'éducation et de la santé, depuis le 10 mai 2012. Elles sont basées sur les lignes définies dans WCAG 2.0 et reprennent leurs trois niveaux de priorité.

Tous ces ensembles de lignes directrices concernant l'accessibilité des sites Web sont très similaires. On y retrouve, entre autres, l'importance d'avoir un code HTML correspondant à l'ordre séquentiel logique de lecture, du texte alternatif pour chaque élément non-textuel, et une cohérence entre la signification de chaque élément Web et la balise HTML qui lui est associée. Lorsqu'on regarde le nombre et le détail des différentes lignes de conduite à suivre, on peut facilement penser qu'il est laborieux de les appliquer à un site Web. Néanmoins, la majorité d'entre elles relève de la logique. Si on tient compte de ces lignes directrices dès le début de la conception d'un site Web, la charge de travail supplémentaire n'est pas si élevée.

1.2.2.3 Lecteurs d'écran et navigateur Web

Les premiers lecteurs d'écran datent de DOS (Disk Operating System) et lisaient un écran d'une résolution de 80x25 pixels, directement à partir du code ASCII (American Standard Code for Information Interchange) (Thatcher, 2006b). Ils ont dû devenir plus intelligents avec l'apparition des interfaces graphiques. Pour ce faire, un mode de lecture spécial a été créé : l'offscreen mode (OSM). Il correspond en fait à une base de données contenant tout le texte affiché à l'écran. L'information est ainsi lue depuis l'OSM et non plus depuis l'écran.

Avec l'arrivée du Web, les lecteurs d'écrans ont encore dû s'adapter. Pour ce faire, les navigateurs génèrent un « Document Object Model » (DOM). Le DOM est un arbre de type XML contenant tous les composants de la page Web. L'information des pages est ainsi lue depuis le DOM puis présentée sous forme de synthèse vocale ou sur une plage Braille rafraîchissable (Thatcher, 2006b). Dans les deux cas, une linéarisation de l'information est effectuée pour l'adapter à une seule dimension à partir de l'affichage de la page Web sur l'écran. La lecture se fait dans l'ordre du code source, en éliminant les balises HTML mais en gardant quelques attributs comme « alt » ou « title ». La lecture se fait de gauche à droite puis de haut en bas. L'avantage des lecteurs d'écrans est qu'ils sont compatibles avec tout le système d'exploitation et pas spécifiques à un seul navigateur.

Les utilisateurs non-voyants utilisent surtout Internet Explorer, plutôt que Firefox ou Google Chrome, car le DOM y est plus accessible. De plus, ils ne lisent généralement pas la page en entier mais naviguent entre les titres, paragraphes, tableaux, etc. Les lecteurs d'écran les plus utilisés sont JAWS (Freedom Scientific, 2013a), NVDA (NV Access, 2013), Window-Eyes (GW Micro, 2013) et SuperNova (Dolphin, 2013) sur Windows, VoiceOver (Apple, 2013) pour Mac, Gnopernicus (OATS, 2013) et Emacspeak (Raman, 2013) pour Unix. Leurs commandes au clavier sont souvent complexes car elles ne doivent pas entrer en conflit avec les raccourcis clavier du navigateur et du système d'exploitation. Il faut donc beaucoup de pratique pour les maîtriser. Des alternatives à la complexité de ces commandes sont d'utiliser des navigateurs parlants ou textuels. Le lecteur d'écran le plus utilisé actuellement est JAWS.

Un des navigateurs parlants les plus connus était Home Screen Reader (Thatcher, 2006b). Il était disponible en six langues (il n'est plus distribué) et disposait d'un écran divisé en quatre parties : le navigateur graphique, le navigateur textuel, l'historique et une fenêtre d'information. Le texte lu est montré dans le navigateur textuel et le mot lu est surligné. L'exemple le plus probant de navigateur textuel est Lynx (Wikipedia, 2013) qui n'est utilisable que via le clavier. Ultra rapide, il utilise les couleurs pour indiquer la structure mais n'est pas compatible avec CSS, JavaScript, et Java.

Lorsque les logiciels de lecture d'écran n'étaient pas très performants, certains outils permettaient de corriger des problèmes d'accessibilité des sites Web par l'intermédiaire d'un serveur proxy (Farhat & Bellik, 1999; Mankoff, Dey, Batra, & Moore, 2002). Ces outils permettaient

d'améliorer l'accessibilité d'une page Web de façon transparente, ou avec l'aide d'un webmestre. Ils pouvaient par exemple créer automatiquement les listes des liens, des images et des titres de sections afin que l'on puisse naviguer plus rapidement dans un page Web (Farhat & Bellik, 1999). Un projet plus récent permet aussi de corriger certains problèmes d'accessibilité à l'aide d'un module d'extension de fureteur (Colas, Monmarché, & Slimane, 2009). Ce module permet entres autres de reformater automatiquement les tableaux qui sont utilisés comme mise en page pour une meilleure exploration dans la page, ou d'ajouter des textes de description pour les zones actives d'une image-map.

Pour les utilisateurs malvoyants, il existe des logiciels d'agrandissement d'écran. L'un d'entre eux est directement fourni avec Windows depuis la sortie de la version XP : l'outil « Loupe ». Ces logiciels permettent généralement un agrandissement de tout ce qui est affiché de 1,5 à 32 fois. Néanmoins, plus l'agrandissement est important, moins il y a d'informations affichées à l'écran. Ainsi, on n'a plus de vue d'ensemble et on peut avoir des problèmes pour identifier le contexte de l'information affichée. De plus, les informations sont souvent difficiles à trouver, surtout pour l'exploration et les formulaires. Une solution est de conserver des architectures constantes, comme les menus par exemple (Thatcher, 2006b). La majorité de ces agrandisseurs d'écran offrent tout de même de séparer l'écran en deux pour garder une partie de l'écran avec l'affichage original. Certains d'entre eux, comme ZoomText (AI Squared, 2013) et Magic (Freedom Scientific, 2013b), lissent même l'information agrandie pour éviter leur pixellisation.

Il existe des outils permettant de vérifier l'accessibilité d'une page Web. Certains sites Web, comme Lynx Viewer, permettent de voir la page Web de manière linéaire comme si on utilisait le navigateur Lynx. Le fait de voir une page Web affichée de manière linéaire permet d'identifier des problèmes d'accessibilité, sans avoir recours à un logiciel de lecture d'écran. Un autre site, Wave, permet d'afficher la page Web avec des icônes identifiant les erreurs, avertissements ou succès relatifs à l'accessibilité.

Les navigateurs intègrent aussi certains outils utiles à l'accessibilité comme la désactivation des images ou l'agrandissement de la police. Des barres d'outils et des favelets existent aussi pour tester l'accessibilité comme la « Web Accessibility Toolbar » (Vision Australia, 2013), la « Section 508 Toolbar » (RampWeb, 2013) et le NCAM Accessibility QA Favelet (National Center for Accessible Media, 2013). Ces derniers ne sont disponibles qu'en anglais.

Enfin, la majorité des lecteurs d'écran sont compatibles avec des afficheurs Braille. Ceux-ci peuvent être moins contraignants que la synthèse vocale qui peut facilement devenir ennuyante. Néanmoins, les afficheurs Braille ne font que retranscrire la même information pouvant être donnée par la synthèse vocale. Par exemple, l'entreprise Humanware (HumanWare, 2013) fournit deux genres d'afficheurs Braille : BrailleNote (Figure 1-1a) qui permet aussi de prendre des notes en Braille grâce à un ordinateur intégré, et des tablettes Braille plus classiques comme BrailleConnect et Brailliant (Figure 1-1b).

(a) (b)

Figure 1-1 : Exemples d'afficheurs Braille : (a) BrailleNote et (b) Brailliant

1.2.2.4 Limites des technologies actuelles favorisant l'accessibilité au Web

Plusieurs études ont mis en relief certaines limites des technologies actuelles, et tout particulièrement dans l'utilisation de lecteurs d'écran pour naviguer sur le Web, provoquant la frustration des utilisateurs non-voyants (Lazar et al., 2007; Leporini & Paternò, 2004; Web AIM, 2013). La principale limite est l'absence de vue globale de la page Web (Lazar et al., 2007; Leporini & Paternò, 2004) ce qui provoque l'accumulation de petits problèmes d'exploration et de compréhension dans la page.

À partir d'une étude sur 100 utilisateurs non-voyants de lecteur d'écran, Lazar et al. ont déterminé que les principales causes de frustration liées à l'utilisation de lecteurs d'écran étaient :

a) une mauvaise mise en page gêne le retour sonore du lecteur d'écran,
b) un conflit entre l'application et le lecteur d'écran,
c) des formulaires mal conçus et mal libellés,
d) l'absence de texte alternatif pour les images,

e) les hyperliens ambigus,
f) les fichiers PDF inaccessibles,
g) les pannes de lecteurs d'écran.

Parmi ces sept points, trois (a, c et e) sont directement liés à la linéarité de l'exploration avec les lecteurs d'écran : c.-à-d. que l'utilisateur est obligé de suivre un chemin unique préétabli. Trois autres points (b, f et g) sont directement liés à des problèmes technologiques et n'ont rien à voir avec la navigation ou l'exploration. Il en est de même pour l'enquête menée par Web AIM (Web Accessibility in Mind) sur 665 utilisateurs non-voyants de lecteurs d'écran (Web AIM, 2013). En effet, cette enquête récente (octobre 2009) montre que les hyperliens et boutons ambigus, ainsi que les formulaires trop complexes font partie des éléments les plus problématiques lors de l'accès à une page Web pour les non-voyants. Les hyperliens et les boutons peuvent être ambigus car il est parfois difficile de savoir à quelles informations ils sont associés et quel est leur véritable but (Leporini & Paternò, 2004). En ce qui concerne les formulaires, c'est souvent le manque d'association entre les étiquettes et les champs de formulaire qui pose des problèmes d'accessibilité. Des problèmes concernant le rafraichissement automatique de la page ont été aussi mis en relief (Lazar et al., 2007). En effet, à chaque rafraichissement, le lecteur d'écran recommence la lecture au début si rien n'a été indiqué dans le code de faire autrement. L'ensemble de ces problèmes provoque une perte de 30,4% du temps passé devant un ordinateur (Lazar et al., 2007).

On peut donc se demander comment surpasser ces limites. L'exploration linéaire (un seul chemin d'exploration possible) semble le principal élément à l'origine de ces limites. En effet, cette exploration linéaire est justifiée par l'emploi d'une seule modalité de sortie : le son (on ne peut pas vraiment considérer la tablette Braille comme une seconde modalité de sortie pertinente vu que celle-ci ne fait que répéter en Braille le texte dit avec le son). N'utiliser que le son comme sortie justifie aussi le fait que le code CSS ne soit pas traité par les lecteurs d'écran. Mais sans le traitement du CSS, on perd accès aux différentes informations de mise en page de la page Web qui peuvent aider sa compréhension.

L'ajout d'une seconde modalité d'appoint pourrait régler les problèmes liés à la linéarité de l'exploration et diminuera donc la frustration des utilisateurs non-voyants. Ainsi, cet ajout donnera une meilleure vue globale de la page Web et fera ressortir plus de relations entre les

différents éléments la composant. Voilà pourquoi nous proposons une navigation et une exploration audio-tactile et spatiale, donc non-linéaire. Pour ce faire, intéressons-nous maintenant plus particulièrement à l'interaction tactile et à ce qu'elle apportera à l'accessibilité du Web pour les non-voyants.

1.3 Interaction tactile

Nous venons de nous intéresser à la population bien spécifique des personnes non-voyantes. La perte de la vue provoque néanmoins le développement des autres sens, tout particulièrement le toucher. En effet, on a longtemps pensé qu'il n'y avait pas de différence entre les voyants et les non-voyants en ce qui concerne l'acuité tactile (Hollins, 1989). Mais de récentes recherches ont prouvé que le sens du toucher était plus développé chez des non-voyants que chez des voyants du même âge (Goldreich & Kanics, 2003; Stevens, Foulke, & Patterson, 1996). Ainsi, de nombreuses recherches sur les personnes non-voyantes utilisent la modalité tactile pour essayer de combler le manque de la vue en augmentant l'interaction tactile. C'est pour cette raison que nous allons maintenant nous intéresser à ce type d'interaction.

1.3.1 Le sens du toucher

Le toucher est le principal sens utilisé pour agir. De plus, il est la seule modalité bidirectionnelle. L'action et la perception y sont vraiment très imbriquées. Vous ne pouvez pas toucher sans ressentir et vous ne pouvez pas ressentir sans toucher.

1.3.1.1 La main

La main est indissociable du sens du toucher. Elle en est l'organe primaire (Katz, 1989). Pour connaître le développement de la main, il faut prendre en compte ses racines archéologiques, zoologiques et anthropologiques. De plus, nous pouvons aussi porter notre attention sur la phylogénie (étude de la formation et de l'évolution des organismes vivants en vue d'établir leur parenté) et sur l'ontogénie (étude du développement d'un organisme depuis sa conception jusqu'à sa mort) pour expliquer comment la main est devenue un outil aussi utile.

Il y a eu trois développements majeurs dans l'évolution de la structure et des fonctions de la main (Jones & Lederman, 2006). En effet, celle-ci est passée d'un moyen de locomotion à un organe de manipulation. Le premier développement correspond à l'apparition de la main à cinq doigts.

Le second consiste à l'amincissement et au déplacement dorsal des griffes qui se sont transformées en ongles. Et le dernier est le développement d'une surface tactile ultrasensible en dessous des doigts. Mais ces trois évolutions auraient laissé notre main « primitive » si un dernier changement radical n'était pas apparu : le développement du pouce opposable. Cette apparition a encouragé le déplacement bipède et l'emploi de la main comme outil.

Ce pouce opposable permet le contact direct entre les parties inférieures de nos doigts. Il peut bouger de 45° autour de son axe longitudinal et permet la manipulation et l'exploration de petits objets. Contrairement à ce que l'on pourrait croire, cette caractéristique ne concerne pas tous les primates. En effet, les colobes et les atèles sont dépourvus de pouces (Napier, 1993). Ce qui rend unique la main humaine par rapport à la main des primates est la surface extensible de l'extrémité de nos doigts ainsi que la taille idéale entre le pouce et l'index pour que ceux-ci puissent correctement interagir. De plus, contrairement aux autres primates, mais conjointement avec les chevaux, nos phalanges distales sont compressées pour mieux distribuer la pression lorsque nous attrapons un objet, puis mieux la dissiper lorsque nous le relâchons.

La main est un merveilleux instrument qui sert pour toute sorte de choses. Elle permet surtout d'identifier des objets et d'en extraire toute sorte d'information (texture, poids, forme, etc.). Sa dextérité de mouvement est aussi impressionnante, comme en témoigne la dextérité avec laquelle les personnes sourdes et muettes l'utilisent pour le langage de signes (Jones & Lederman, 2006). Mais la main est aussi un outil de créativité. Les arts de la danse, de la sculpture, de la peinture, ainsi que la musique (pour des instruments) font appel à ses talents. Et ce n'est pas quelque chose de nouveau : l'art avec la main est apparu il y a environ 28 000 ans (Jones & Lederman, 2006).

Révész s'est intéressé à la perception de la forme des objets avec la main, opposée à la perception par la vision (Révész, 1950). C'est en observant des utilisateurs non-voyants qu'il a prouvé qu'il était aussi facile d'identifier la forme d'un objet avec la main qu'avec les yeux. Cela a été repris par Gibson qui a mis en valeur le fait que la main est utilisée pour des fonctions de perception, ainsi que pour des fonctions d'exécution (Gibson, 1962) ; ce qui fait de la main un organe sensoriel tout à fait unique chez l'être humain. De plus, contrairement à nos autres organes sensoriels, la main peut améliorer sa rapidité et sa précision de mouvement, comme le montre l'étude de Woodworth avec les opérateurs de Morse (Woodworth, 1899).

Jones et Lederman définissent quatre différentes fonctions de la main (Jones & Lederman, 2006). La sensation tactile est la première d'entre elles et correspond à l'effet du contact entre une main immobile et une surface ou un objet. Dans cette fonction, la main y est toujours passive. La sensation haptique active implique un mouvement volontaire de la main sur une surface ou un objet. La préhension consiste à attraper un objet et à le tenir dans sa main. Enfin, les mouvements habiles non préhensibles correspondent à des mouvements de la main sans aucune interaction avec des surfaces ou des objets. Cela inclut par exemple le langage des mains lorsque nous parlons.

C'est grâce à ces caractéristiques que les interfaces tangibles ont fait leur apparition, permettant une exploration totale et libre de l'interface avec la main, notamment pour les utilisateurs non-voyants.

Une dernière chose est intéressante à noter. La sensibilité sensorielle varie d'un endroit à l'autre du corps mais c'est dans la main que le seuil de discrimination tactile est le plus petit. Les mécanorécepteurs de nos doigts peuvent détecter des signaux tactiles d'une fréquence de 15 Hz jusqu'à 1000 Hz (Boff, Kaufman, & Thomas, 1988). De plus, notre cerveau alloue énormément de capacités à nos mains. En effet, une topographie des représentations du corps humain sur le cortex sensoriel a été faite et donne cet étrange bonhomme, l'homunculus de Penfield (Figure 1-2) (Crichton & Crichton, 1994).

Figure 1-2 : L'homunculus de Penfield

1.3.1.2 Mouvement

Le contrôle du mouvement de la main est assuré par le fonctionnement de 29 muscles qui se situent en majorité dans l'avant-bras. Ce sont les muscles intrinsèques de la main (Jones & Lederman, 2006).

La notion de « sensation musculaire » est apparue il y a plus de 100 ans, puis fut remplacée par les termes « proprioception » (Sherrington, 1906) et « kinesthésie » (Bastian, 1887). La proprioception se rapporte aux récepteurs dans les muscles, tendons et jointures. Cela concerne donc les mouvements et les tensions de notre corps (Webster, 1985). La kinesthésie ajoute à la proprioception les récepteurs de la peau et ceux de l'oreille interne. Elle permet de nous transmettre toutes les informations concernant la position actuelle de tous les éléments de notre corps (Burdea, 1996). Tous ces concepts permettent la sensation musculaire qui contient les sensations de fatigue, de douleur, de poids et de résistance, ainsi que de mouvement et de position. Ce sens est souvent considéré comme le sixième sens.

Il est important de faire la différence entre le toucher actif, impliquant une série de sensations musculaires, et le toucher passif, impliquant les points tactiles (Sherrington, 1906). Le toucher actif se rapporte donc plus à de l'exploration tactile, c.-à-d. que la personne est en train d'effectuer des mouvements (ex. : lecture du Braille). Le toucher passif se rapporte à une sensation tactile lorsque la personne n'est pas en mouvement. Dans ce dernier cas, c'est l'objet d'interaction qui est en mouvement et non pas l'utilisateur (ex. : une fourmi se promenant sur notre bras).

Il faut aussi ajouter la sensation haptique active qui donne beaucoup d'importance au retour d'effort (Weber, 1978). Cette sensation est la résistance rencontrée lors de nos mouvements. Par exemple, il est plus facile d'évaluer le poids d'un objet en le soulevant, donc en impliquant aussi les muscles, qu'en le posant seulement sur notre main, donc seulement avec le sens tactile.

1.3.2 Les outils d'interaction liés au sens du toucher

Nous allons maintenant nous intéresser aux différents outils existants qui utilisent le sens du toucher comme modalité d'interaction de sortie. De nombreux outils utilisant le toucher comme modalité d'interaction d'entrée sont disponibles pour le grand public : l'iPhone en est le parfait exemple. Mais cela n'est pas le cas pour les outils générant un retour tactile qui eux ne sont pas vraiment disponibles pour le grand public.

Dans cette thèse, nous allons discerner deux catégories d'outils liés au sens du toucher : ceux qui génèrent seulement des retours tactiles et ceux qui génèrent aussi un retour d'effort.

1.3.2.1 Les outils à retours tactiles

Les outils à retour tactile se rapportent au toucher actif ou passif. Ils peuvent théoriquement générer des sensations tactiles sur n'importe quel endroit du corps mais sont le plus souvent utilisés avec la main. Nous allons volontairement mettre de côté les supports papier en relief (PolymorpheDesign, 2009) car ils ne sont pas pertinents dans le cadre d'une recherche sur le monde du Web qui est énormément dynamique. Nous passons aussi sur les afficheurs Braille qui ont déjà été décrits dans la section 1.2.2.3. Nous allons plutôt nous intéresser aux dispositifs tactiles permettant de générer des schémas plus complexes et qui pourront être utiles pour la représentation tactile d'une page Web. Ces outils permettent généralement de générer des images tactiles numériques par l'intermédiaire de picots ou lamelles tactiles.

Tout d'abord, parlons d'un outil pouvant représenter d'une manière assez sommaire des graphiques tactiles. Le « Cat » (Figure 1-3a) est un petit appareil qui était normalement prévu seulement pour le braille (Ebina, Igi, Miyake, & Takahashi, 1998). Il est composé de 40 cellules tactiles contenant chacune 16 picots organisés sous forme de matrice 8 x 2. Grâce à cet appareil, les concepteurs ont pu représenter des graphiques tactiles en les transformant en série de points. Il faut préciser que ce type d'outil reste limité en ce qui concerne le rendu tactile car les cellules brailles ne sont pas très précises et les picots qui les composent n'ont que deux états différents possibles : rentrés ou sortis. Comparés à des graphiques sur papier en relief, les résultats obtenus avec le « Cat » étaient bons (Ebina et al., 1998) car les images étaient simplistes, ce qui est son principal défaut.

Le VTD (Virtual Tactile Display) (Maucher, Schemmel, & Meier, 2000) permet d'explorer des images plus complexes que le dispositif présenté ci-dessus. Ce dispositif (Figure 1-3b) est composé d'une surface de 164 x 159 mm qui peut être explorée par un appareil composé de quatre cellules Braille du même type que pour le « Cat », soit 48 actionneurs au total. Ces quatre cellules fixées entre elles peuvent se déplacer à l'aide de trois rails, deux positionnés horizontalement et un autre pour le déplacement vertical. Il est ainsi possible d'explorer facilement des figures mais celles-ci doivent rester simples car les cellules de braille ne peuvent pas donner de rendu tactile très fin. Les résultats obtenus avec cet appareil ont montré que le

dispositif était efficace pour reconnaître des formes simples, comme des cercles, des croix ou des triangles car tous les participants ont pu reconnaître ces formes (Maucher et al., 2000). De plus, ces formes étaient reconnues assez rapidement par les utilisateurs non-voyants (10 secondes en moyenne). Néanmoins, tout comme le « Cat », le rendu tactile reste minimaliste à cause de l'emploi de cellules Braille.

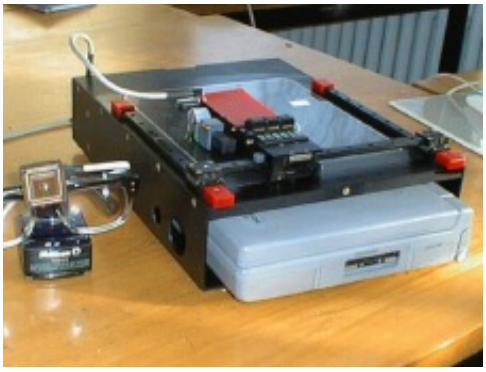
La cellule tactile peut aussi être indépendante de la surface d'exploration. Par exemple, l'outil Tactos (Gapenne, Rovira, Ali Ammar, & Leney, 2003) est composé d'une tablette PC et d'une cellule tactile sur le côté. La cellule tactile ne se déplace pas mais ses actionneurs bougent en fonction de l'endroit où est le stylet sur la tablette. L'utilisateur se sert donc d'une main pour tenir le stylet et le déplacer sur la tablette, et l'autre main pour sentir le retour tactile sur la cellule. Ce même genre de système fut utilisé pour l'Optacon (Goldish & Taylor, 1974), un outil qui fut très utilisé ces 30 dernières années pour faciliter la lecture de livres et de journaux n'étant pas traduits en Braille (Figure 1-3c). En effet, l'utilisateur peut diriger une mini-caméra sur le texte qu'il souhaite lire avec une main comme s'il lisait en Braille un texte en parcourant les lignes de la main. Un afficheur tactile composé de picots agencés sous forme de matrice 24 x 6 est aussi placé sous l'autre main. La caméra reconnait ainsi les différents caractères parcourus et la sensation tactile appropriée est envoyée à l'afficheur.

Un autre outil est le « three-dimensional (3-D) tactile display device » (Figure 1-3d) développé par Shinohara (Shinohara, Shimizu, & Mochizuki, 1998). Cet outil est composé d'une grande tablette (20 x 17 cm) où une matrice de 64x64 picots a été placée de manière à former un hexagone. Chaque picot peut se soulever de 0,1 mm à 10 mm pour représenter une image en relief. Cet appareil prend assez de place mais permet une exploration du graphisme tactile avec les deux mains et en utilisant tous les doigts de chaque main ce qui améliore la compréhension globale de l'image. Ce dispositif est beaucoup plus précis que les ceux précédemment cités et a même été commercialisé.

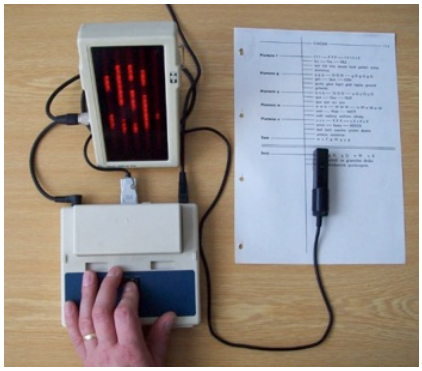
Un dernier outil permet aussi l'exploration à deux mains d'une surface tactile : le « Braille Window System » (Figure 1-3e) du projet HyperBraille (Prescher, Weber, & Spindler, 2010). Contrairement à l'outil développé par Shinohara en 1998, ce système est composé des mêmes picots utilisés par les cellules Braille et qui n'ont donc que deux états (rentrés ou sortis). Néanmoins, ces picots sont très rapprochés et forment une matrice de 120x60 picots

particulièrement grande. De plus, la surface d'exploration est aussi sensible au toucher, ce qui lui permet de reconnaître des commandes gestuelles. Cet outil permet d'afficher des images tactiles mais aussi des interfaces graphiques. Comme c'est souvent le cas, l'affichage tactile demande un travail manuel préalable afin d'adapter les images et les interfaces graphiques. De plus, le retour généré est exclusivement tactile. Le texte des interfaces graphiques est affiché en Braille ce qui peut affecter la forme et la disposition des éléments de l'interface graphique.

(a)



(b)



(c)

(d)

(e)

Figure 1-3 : Divers outils tactiles : (a) le « Cat », (b) le VTD, (c) l'Optacon, (d) le « three-dimensional (3-D) tactile display device » et (e) le « Braille Window System ».

À part l'Optacon et l'appareil de Shinohara, ces dispositifs sont tous restés au stade de la recherche et n'ont jamais été mis au service du grand public. Cela peut s'expliquer par le fait que le grand public trouve peut-être peu utiles les outils existants ayant un retour tactile. C'est peut-être aussi parce que les outils générant un retour tactile destinés au grand public n'avaient pas forcément une utilisation très pertinente. Le parfait exemple est la souris iFeel de Logitech (Logitech, 2013). Cette souris, ressemblant à une souris tout à fait normale, génère des pulsations tactiles en fonction des actions effectuées avec celle-ci. Sa fonctionnalité principale était son intégration à Windows et aux navigateurs Web. Par exemple, une vibration est émise lorsque la souris passe au dessus d'un hyperlien ou d'une icône. La souris iFeel ne permet pas de lire en plus le texte situé en dessous de son curseur. Elle n'a malheureusement pas été adoptée par le grand public qui l'a plus considérée comme un gadget que comme un outil vraiment utile.

1.3.2.2 Les outils haptiques à retour d'effort

Le mot « haptique » vient du mot grec « Ηαπτειν » qui veut dire « toucher » et se rapporte à tout ce qui concerne le toucher et le retour d'effort. On l'associe souvent au système somesthésique qui est un de nos systèmes sensoriels se rapportant à toutes les sensations ressenties au cours du toucher (pression, chaleur, douleur, etc.) (Kandel, Schwartz, & Jessell, 2000). On associe donc les outils haptiques à la sensation haptique active, soit le retour d'effort. On n'associe généralement pas les outils haptiques aux outils générant des sensations tactiles (voir 1.3.2.1).

Un système à retour d'effort est un appareil bidirectionnel, vu qu'on peut agir tout en ressentant le retour d'effort. Ce retour est produit en combinant l'action de moteurs et de capteurs de déplacement. Cette combinaison doit produire un retour approprié par rapport à ce qu'attend l'utilisateur (Hayward & MacLean, 2007). En effet, l'haptique est souvent utilisé conjointement avec d'autres sens : ce sont des interactions multimodales (voir 1.1.2). Elle peut ainsi renforcer une tâche ou servir à des tâches additionnelles si les autres canaux sensoriels sont saturés (MacLean & Hayward, 2008). Il faut donc que le retour d'effort soit correctement coordonné avec les autres sens comme il l'est dans notre vie de tous les jours. Par exemple, un joueur de baseball voulant frapper correctement la balle avec un bâton va se placer d'une certaine manière en fonction du type de lancer de balle de son adversaire. Ce placement va l'aider à mieux absorber le choc généré par le contact entre la balle et le bâton afin de la renvoyer le plus loin possible, ou à l'endroit le plus inaccessible pour les autres joueurs. Le joueur coordonne ainsi sa position, ses

gestes, l'orientation et la force du mouvement du bâton en fonction de ce qu'il observe du lancer de son adversaire. La force du retour d'effort d'un outil haptique doit donc être minutieusement contrôlée pour refléter le plus possible la réalité. Pour aider ce contrôle, il vaut mieux que la base de l'appareil soit stationnaire. Si cela n'est pas possible, il faut essayer de compenser les éventuels mouvements de l'appareil qui pourraient gêner son interaction avec l'utilisateur. De plus, les capteurs doivent avoir une résolution suffisante pour pouvoir gérer les mouvements les plus fins.

Sur le marché de l'haptique, les outils les plus répandus dans le monde de la recherche sont les bras à retour d'effort comme le Phantom Omni de Sensable qui permet une exploration haptique en trois dimensions (Sensable, 2013). Ce genre de systèmes à retour d'effort est utilisé, par exemple, pour la téléopération. Cela permet ainsi de remplacer le bras et la main d'un opérateur, par exemple, qui peut effectuer son travail à des milliers de kilomètres de l'appareil. On peut ainsi toucher des objets virtuels et obtenir la même sensation que s'ils étaient réels. Ce genre de système commence sérieusement à percer dans le monde de la chirurgie pour simuler des opérations dans un cadre d'apprentissage (Scharver, Evenhouse, Johnson, & Leigh, 2004). Certains outils haptiques utilisés dans le monde de la recherche permettent aussi une exploration en deux dimensions avec l'aide de retours d'effort. C'est le cas du Pantograph (Ramstein & Hayward, 1994) qui permet entre autres de simuler des situations de gravité nulle.

Tout comme les outils tactiles, très peu d'outils haptiques ont été commercialisés pour le grand public. On peut considérer qu'un seul d'entre eux a vraiment trouvé son public : le contrôleur à retour d'effort pour les jeux vidéo de courses, par exemple les volants de Thrustmaster (Thrustmaster, 2013). Ce genre de système reste assez basique pour le retour d'effort. Néanmoins, des systèmes beaucoup plus poussés ont aussi été proposés au grand public mais ils n'ont connu que des échecs jusqu'à maintenant. L'exemple le plus probant est la souris Wingman de Logitech qui génère un retour d'effort sur deux dimensions (WingMan Team, 2013). Tout comme la souris iFeel, ce fut un tel échec qu'elle n'est plus vraiment supportée par son constructeur. En effet, c'est une équipe parallèle à Logitech qui s'en occupe avec un site Web complètement indépendant. Néanmoins, un autre outil à retour d'effort est sorti pour le grand public en 2006 en ciblant le marché des jeux-vidéo : le Novint Falcon (Novint, 2013). Ce dernier, tout comme le Phantom Omni, permet de toucher des objets virtuels comme s'ils existaient réellement. Même si cet appareil a connu un meilleur départ que la Wingman, on ne peut pas vraiment dire qu'il ait été un succès. Même s'il pouvait apporter plus de réalité aux jeux-vidéo, il

n'a été supporté que par très peu de compagnies de jeux-vidéo et fut jugé trop dispendieux. Son développement est néanmoins toujours supporté par Novint.

1.3.2.3 Exploration haptique et reconnaissance d'objets

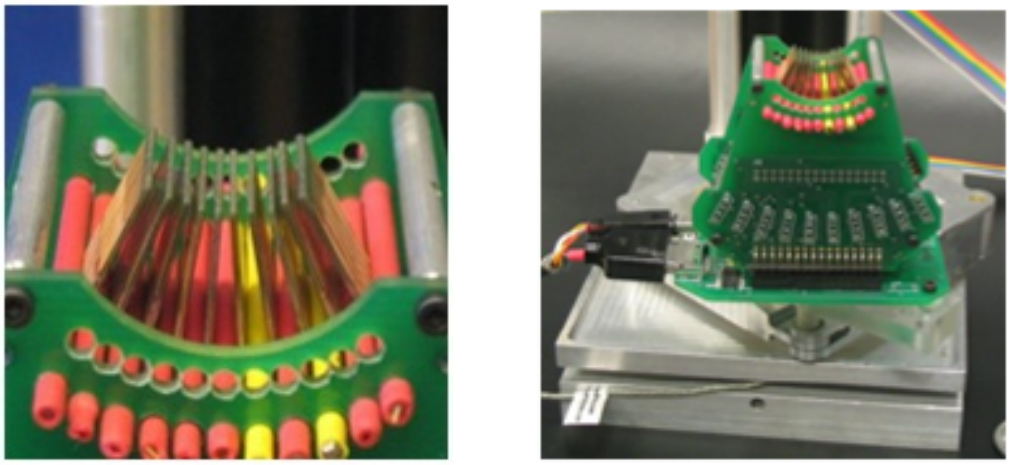
En ce qui concerne l'utilité de l'exploration haptique pour les utilisateurs non-voyants, Graupp et al. résument très bien la situation en se concentrant justement sur la perception haptique d'objets en trois dimensions et de graphismes tactiles en deux dimensions (Graupp, Gladstone, & Thompson, 2002). La perception haptique est très efficace lorsqu'il s'agit d'explorer des scènes de tous les jours et reconnaître des objets en trois dimensions (96% d'objets reconnus lors de l'expérimentation menée par Klatzky et al.) (Klatsky, Lederman, & Metzger, 1985) ; il en est de même chez les utilisateurs non-voyants (Graupp et al., 2002). Mais la perception haptique est moins adéquate pour une exploration en deux dimensions (Lederman, Klatsky, Chataway, & Summers, 1990). En effet, lors de l'expérimentation menée par Lederman et al., des images tactiles représentant des objets de la vie courante en relief étaient explorées par des utilisateurs voyants ayant les yeux bandés. Seulement un tiers de ces images ont été correctement identifiées (Lederman et al., 1990). Et c'est encore plus problématique pour des utilisateurs non-voyants où, après une exploration de ce même genre d'images tactiles en relief, seulement un quart d'entre elles étaient correctement identifiées (Graupp et al., 2002). Néanmoins, ces études sont concentrées sur des représentations d'objets en deux dimensions pouvant être complexes.

Pour notre recherche où une exploration en deux dimensions est nécessaire puisqu'il s'agit d'explorer le monde du Web, une exploration haptique en trois dimensions est exclue. Nous sommes obligés d'avoir une exploration en deux dimensions à cause de la nature même du Web qui est présenté en deux dimensions. Même si l'exploration haptique d'objets représentés en deux dimensions est plus difficile, la représentation des éléments composant une page Web est moins complexe que celle des objets utilisés dans les études décrites dans le paragraphe précédent. En effet, l'organisation et la forme de ces éléments Web sont presque exclusivement géométriques, sous la forme d'une grille. L'exploration se fera donc principalement de manière horizontale et verticale, ce qui n'est pas le cas lors de l'exploration d'objets issus de la vie courante.

1.3.2.4 Le Tactograph et MaskGen

Intéressons-nous maintenant à un autre outil tactile, plus récent, permettant toute sorte de rendus tactiles assez fins : le Tactograph (Figure 1-4b). Celui-ci a été développé au laboratoire Haptics de l'Université McGill de Montréal. Il est composé d'un module tactile, où est inséré l'appareil tactile STReSS² (Stimulator of Tactile Receptors by Skin Stretch), monté sur un appareil haptique d'exploration sur deux dimensions, le Pantograph (décrit dans la section précédente).

Le Tactograph est vraiment différent des autres types d'outils tactiles que l'on utilise d'habitude car il stimule le bout de notre doigt en étirant latéralement la peau. Le STReSS² possède une cellule de rendu tactile de 10,8 x 12 mm (Figure 1-4a) ce qui correspond environ à la surface de peau au bout de notre index (Wang & Hayward, 2006). Pour déformer latéralement la peau, 60 actionneurs (6 x 10) bougent de droite à gauche pour produire des sensations de vibration, d'ondulation, etc. (Figure 1-4a). La distance entre chaque actionneur est de 1,8×1,2 mm et chacun d'eux peut se courber de 0,1 mm vers la gauche ou vers la droite. Le STReSS² est ensuite intégré dans une protection en mousse noire, puis monté sur un charriot permettant d'explorer une surface de 11,3 x 6 cm. Grâce à ce dispositif, il est donc possible d'explorer une surface, tout en ayant un rendu tactile adapté à l'endroit où l'on se trouve sur cette surface.



(a) (b)

Figure 1-4 : Les actionneurs STReSS² (a) et le Tactograph (b)

Le principal avantage de cet appareil est sa cellule tactile. En effet, celle-ci permet d'avoir un rendu tactile beaucoup plus fin qu'un ensemble de cellules braille. Cela reste aussi plus précis que le « three-dimensional (3-D) tactile display device » (Shinohara et al., 1998) car dans le cas

du Tactograph, il est possible d'avoir des textures différentes. C'est donc pour ses rendus tactiles plus fins et plus variés que nous avons choisi cet outil pour développer notre système permettant l'exploration audio-tactile, donc spatiale, de pages Web par des utilisateurs non-voyants. Le Tactograph est contrôlé de manière logicielle par un programme C++ très complexe mais très solide. Celui-ci permet de contrôler les 60 actionneurs et de générer des rendus tactiles très différents et très fins, qu'ils soient statiques (vibrations, ondulations unilatérales ou radiales fixes, formes vectorielles, points) ou dynamiques (ondulations unilatérales ou radiales en mouvement, rendus tactiles différents en fonction de la vitesse du mouvement, rendus tactiles en fonction de la direction du mouvement).

Grâce à cet outil, l'équipe du laboratoire Haptics a déjà créé quelques petites applications comme un jeu de mémoire (Wang, Lévesque, Pasquero, & Hayward, 2006). De plus, nous avons aussi développé une application permettant l'exploration tactile et sonore d'images provenant de manuels scolaires, MaskGen, et une expérimentation a permis de mettre de l'avant la qualité des rendus tactiles de l'appareil et son utilité pour explorer du contenu par des utilisateurs non-voyants (Petit, Dufresne, Lévesque, Hayward, & Trudeau, 2008).

MaskGen permet d'automatiser certaines étapes de la transformation des images qui doivent être transcrites en images audio-tactiles. Celui-ci, développé en Java, facilite la définition du graphique tactile et produit automatiquement un fichier XML qui est directement utilisable par le Tactograph. Néanmoins, les images doivent d'abord être simplifiées avec un logiciel d'édition d'image avant de pouvoir être traitées par MaskGen. L'image simplifiée doit ainsi contenir des bordures noires et les zones à l'intérieur de ces bordures doivent être en blanc. MaskGen extrait automatiquement les zones composant l'image simplifiée. On peut alors attribuer un rendu tactile spécifique, ajuster les paramètres et associer un retour audio à chacune de ces zones. Les retours audio peuvent être générés automatiquement par synthèse vocale en français et en anglais. Toutes ces informations se retrouvent ensuite dans le fichier XML pour être ensuite exploitées par le Tactograph. Quatre images venant d'un manuel scolaire d'histoire de secondaire 2 ont été adaptées avec MaskGen et furent testées au cours de l'expérimentation (voir Figure 1-5).

Figure 1-5 : Expérimentation de MaskGen avec le Tactograph

Le Tactograph, associé à MaskGen, possède ainsi plusieurs applications utiles pour les utilisateurs non-voyants et c'est la raison pour laquelle nous l'avons utilisé pour notre recherche. Mais c'est surtout parce qu'il nous semble être le plus adapté à une exploration audio-tactile en deux dimensions que nous l'utilisons et que nous nous sommes orientés vers une exploration spatiale du Web.

De plus, l'idée originale d'utiliser le Tactograph pour naviguer sur le Web nous est venue lors de l'expérimentation sur MaskGen, au cours d'une session à l'INLB (Institut Nazareth et Louis Braille). Un jeune participant nous avait indiqué qu'il aimerait bien utiliser cet appareil pour aller sur Internet.

Au début de cette thèse, un autre point intéressant était que le Tactograph exploitait des fichiers XML générés par MaskGen et que les pages Web sont principalement produites par du code HTML, un langage semblable au XML pour la syntaxe. Nous pensions donc utiliser cette similitude pour explorer des pages Web mais nous nous sommes rendu compte par la suite que cela ne serait pas possible avec les ressources dont nous disposions (voir 4.1.2.)

1.4 Synthèse

Cette revue de littérature vient de présenter un tour d'horizon des domaines des IHM, de l'accessibilité aux utilisateurs non-voyants et de l'interaction tactile.

En ce qui concerne les IHM, nous nous intéressons à l'utilisabilité des interfaces utilisateurs car nous devrons en tenir compte lors des tests avec les personnes non-voyantes visant à comparer la performance et la satisfaction humaine liées à l'utilisation de notre système et d'un système de référence. Nous examinons aussi les interactions multimodales car nous sommes d'avis que ce type d'interaction peut améliorer l'accessibilité du Web actuel. Utiliser des interfaces multimodales donne entre autres une plus grande flexibilité à un système et permet ainsi d'éviter le problème de saturation sensorielle qui est causé par l'emploi d'une seule modalité (Oviatt, 2003).

Nous passons ensuite en revue ce qui touche au domaine de l'accessibilité du Web pour les personnes non-voyantes. Il est important de connaître la population visée par notre recherche de même que les caractéristiques, avantages et inconvénients des différentes technologies adaptées qui leurs sont disponibles. Ces informations nous permettent d'explorer comment notre recherche peut palier aux différents problèmes d'accessibilité rencontrés par les personnes non-voyantes au cours de leur utilisation de l'Internet.

Enfin, nous nous focalisons sur l'interaction tactile pour mieux comprendre comment utiliser ce genre d'interaction dans notre recherche. Il était primordial de savoir comment fonctionne le sens du toucher et comment il pouvait palier en partie l'absence du sens de la vision. Nous avons aussi passé en revue les différents outils tactiles permettant d'adapter du contenu graphique et qui pourraient nous inspirer pour notre recherche. Beaucoup d'entre eux sont axés sur les cellules Braille mais nous avons décidé de nous orienter vers un outil générant des sensations tactiles plus précises et plus proches de la réalité : le Tactograph.

CHAPITRE 2 ÉTUDE EMPIRIQUE DES TECHNOLOGIES ADAPTÉES POUR LE WEB À L'USAGE DES UTILISATEURS NON-VOYANTS

Dans notre revue de littérature, nous avons identifié une certaine frustration vis-à-vis des technologies d'adaptation les plus employées par les utilisateurs non-voyants : les lecteurs d'écran. Cette frustration est en partie causée par le fait que ces lecteurs d'écran ne reproduisent pas l'organisation spatiale lorsque l'utilisateur explore une page Web en n'offrant qu'une exploration linéaire. Il est donc important d'identifier clairement les problèmes posés par l'exploration linéaire. Néanmoins, il y a sans doute du bon dans celle-ci et c'est pour cela qu'il est aussi pertinent d'identifier les avantages de ce type d'exploration.

Pour ce faire, nous avons procédé à une étude empirique des technologies d'adaptation en nous focalisant sur deux outils : un lecteur d'écran classique qui fait une présentation linéaire et un autre lecteur d'écran qui permet à l'utilisateur de faire une exploration spatiale. La méthodologie employée au cours de cette étude empirique est présentée dans la section 2.1. Cette étude comprend les différentes observations que nous avons faites au cours de l'utilisation des deux outils. Les observations sur le premier lecteur d'écrans, JAWS, sont présentées dans la section 2.2 alors que les observations sur le second, VoiceOver sont décrites dans la section 2.3. L'étude comparative aborde aussi brièvement la souris tactile iFeel dans la section 2.4.

2.1 Méthodologie

Pour cette étude empirique, nous avons retenu le logiciel de lecture d'écran le plus utilisé sur le marché, JAWS (Freedom Scientific, 2013a), ainsi qu'un logiciel d'écran plus récent qui intègre des notions d'exploration spatiale, VoiceOver (Apple, 2013). Nous voulions à la base évaluer VoiceOver car il est de plus en plus utilisé, notamment sur les appareils mobiles de la gamme Apple (iPod touch, iPhone, iPad). Il a grandement permis de rendre ce genre d'appareils mobiles accessibles aux personnes non-voyantes. Le fait qu'il soit aussi le seul lecteur d'écran grand public rendant possible l'exploration spatiale grâce à une interaction tactile en faisait un outil très pertinent à évaluer.

Cette étude empirique comprend une évaluation technique et une évaluation ergonomique. L'évaluation technique consistait à utiliser les différentes fonctionnalités des deux outils pour explorer trois sites, en assumant que ces fonctionnalités étaient maîtrisées. Cela nous a permis d'observer comment les deux lecteurs d'écran se comportaient en situation réelle. Ces trois sites n'ont servi qu'à établir le contexte et le but de l'étude empirique n'était en aucun cas d'évaluer leur accessibilité. Néanmoins, nous avons pu observer que les problèmes liés au non-respect de certaines lignes directrices d'accessibilité pouvaient être accentués par l'exploration linéaire.

L'évaluation ergonomique portait sur les différentes interactions possibles entre un utilisateur et les deux outils; il s'agissait donc d'une évaluation de l'interface plutôt que des fonctionnalités des outils. On a examiné l'interaction avec le clavier pour les deux outils et avec le pavé ou l'écran tactile pour VoiceOver. Les commentaires de plusieurs personnes non-voyantes ont été aussi pris en compte pour cette évaluation ergonomique.

Pour l'évaluation technique, trois sites Web furent testés en faisant les tâches suivantes :

- Se rendre sur le portail international depuis la page d'accueil du gouvernement du Québec[1]
    - Se rendre sur le portail de la France depuis le portail International.
    - Retourner à l'accueil et trouver les services aux citoyens québécois.
    - Accéder au système de changement d'adresse.
    - Retourner à l'accueil, trouver le formulaire de contact par courriel puis le remplir.
- Se rendre sur le site du Collège André-Grasset[2]
    - Trouver des renseignements sur le programme en sciences de la nature.
    - Depuis la page du programme en sciences de la nature, se rendre dans les conditions d'admission du collège.
- Se rendre sur le site du journal Le Point[3]
    - Trouver l'adresse postale pour contacter le journal.

---

[1] http://www.gouv.qc.ca/portail/quebec/pgs/commun/

[2] http://www.grasset.qc.ca/

[3] http://www.lepoint.fr/

- Depuis la page de contact, trouver le formulaire d'abonnement à la « newsletter » puis le remplir.

La raison pour laquelle nous avons choisi ces trois sites est que le premier suit parfaitement les lignes directrices d'accessibilité du Web, le second ne les suit pas du tout et le troisième a quelques problèmes d'accessibilité, surtout avec son formulaire. Le premier site nous a donc permis d'observer une expérience idéale avec les deux outils, le second site nous a permis d'observer une expérience plutôt problématique et le troisième site nous a donné une expérience assez mixte, proche de la majorité des sites.

2.2 JAWS

Nous avons testé la version 12 de JAWS en Français sur Internet Explorer 8. JAWS fonctionne aussi correctement sur Google Chrome et Firefox mais est compatible avec Internet Explorer depuis beaucoup plus longtemps. De plus, Freedom Scientific développe JAWS en collaboration avec Microsoft, ce qui réduit les problèmes d'interaction entre Internet Explorer et ce lecteur d'écran. Il faut noter que nous n'avons pas utilisé d'afficheur Braille avec JAWS. Nous avons néanmoins pu observer le retour qui serait généré sur l'afficheur Braille grâce au logiciel de visionneuse Braille fourni avec le lecteur d'écran.

Avant de nous lancer dans l'utilisation de JAWS, deux points positifs le concernant sont à noter: sa portabilité et son haut degré de personnalisation des paramètres. L'utilisateur peut paramétrer de nombreux facteurs comme ceux de la voix utilisée pour la synthèse vocale (choix de la voix, du débit, de l'intonation, etc.), le degré d'information énoncé pour les types d'éléments Web, la taille des blocs de texte, le comportement par défaut dans les formulaires et le fait de sauter le texte répété dans une nouvelle page. Il est aussi possible d'installer JAWS directement sur une clé USB. Cela permet de pouvoir utiliser le lecteur d'écran sur n'importe quel ordinateur tout en conservant sa propre personnalisation des paramètres. Cette portabilité est très utile lorsqu'une personne non-voyante doit utiliser un ordinateur dans un lieu public ou chez des connaissances.

2.2.1 Exploration linéaire avec JAWS

Intéressons-nous maintenant à l'utilisation de JAWS. Lorsqu'un utilisateur explore une page Web avec ce lecteur d'écran dans un site qu'il ne connaît pas, il est fort probable qu'il aille tout

d'abord explorer la page intégralement avec les flèches directionnelles haut et bas du clavier. Ces flèches permettent de déplacer le curseur virtuel que JAWS crée pour lire les pages Web. En faisant ceci, l'utilisateur accède à une information linéarisée, un peu comme si on déroulait une bobine de fil. Si la page est conçue selon l'ordre logique de lecture, c.-à-d. que le code HTML est conforme à cet ordre, il n'y a pas de perte de contexte d'information au court de l'exploration linéaire. Si ce n'est pas le cas, il est possible que l'utilisateur puisse avoir des difficultés à comprendre la page et qu'il soit difficile d'associer les différentes informations pertinentes entre elles. Un paragraphe décrivant une image qui ne serait pas lue juste après ou juste avant cette image pourrait poser problème. Un menu principal présenté visuellement au début de la page mais lu plutôt vers la fin avec JAWS serait aussi problématique. Ce genre de problème fut rencontré sur le site du Collège André-Grasset.

Il est néanmoins possible de prendre des raccourcis au cours de cette exploration linéaire qui pourrait paraître longue. En effet, l'utilisation de certains raccourcis clavier permet de filtrer le contenu de la page Web. Par exemple, on peut déplacer le curseur virtuel entre les différents titres d'une page Web en appuyant sur la touche « T », entre les hyperliens avec la touche « L » ou entre les paragraphes avec la touche « P ». Il est aussi possible d'afficher une fenêtre contenant tous les éléments du même type (« INSERT + F7 » pour la liste des liens par exemple). Ce genre de raccourcis permet de grandement raccourcir le temps d'exploration d'une page Web, surtout si celle-ci est dans un site où l'utilisateur a l'habitude d'aller. L'utilisateur peut aussi placer des marqueurs dans les sites Web où il a l'habitude d'aller, ce qui lui permet de se déplacer rapidement vers ce marqueur chaque fois qu'il visite ce site. Ce genre de marqueur peut par exemple être très utile dans les sites Web de nouvelles.

La navigation dans un site Web connu se retrouve ainsi grandement accélérée. Par exemple, un utilisateur peut faire « sauter » le curseur virtuel entre les titres ou les liens des pages d'accueil, des sections ou des sous-sections pour aller plus vite au cours de la navigation. Il peut ensuite reprendre une lecture pas à pas, avec les flèches directionnelles du clavier, une fois rendu dans la page qu'il souhaitait atteindre. Cela correspond plus alors à de l'exploration.

En ce qui concerne les formulaires, le curseur virtuel de JAWS passe directement en mode « formulaire » quand il atteint un champ. Ce passage automatique est indiqué par une audicône qui notifie l'utilisateur qu'il peut entrer de l'information dans le champ sans appuyer sur une

touche du clavier supplémentaire. Ce passage automatique peut bien sûr être désactivé et l'utilisateur devra appuyer sur la touche « ENTRÉE» pour activer le mode « formulaire » une fois qu'il est rendu sur un champ, puis sur la touche « ECHAP » pour le désactiver.

Comme n'importe quel élément de la page Web, JAWS linéarise les formulaires. Il faut donc que l'ordre logique de lecture soit respecté dans le code HTML. Si cet ordre ne l'est pas, les formulaires peuvent tout de même être accessibles avec JAWS si l'association entre les champs et leur étiquette est effectuée dans le code HTML. Dans ce cas, la linéarisation de l'information n'est pas bloquante. De plus, si l'ordre logique de lecture et l'association champ/étiquette ne sont pas faits correctement, l'utilisateur peut passer au mode de curseur JAWS. Ce mode de curseur permet de prendre le contrôle du curseur de la souris et de le déplacer horizontalement ou verticalement avec les flèches directionnelles. Ce mode reste tout de même assez difficile à utiliser et ralentit grandement l'exploration.

Il est en de même pour les tableaux. En effet, les tableaux sont faits pour être lus dans deux dimensions, ce qui ne paraît pas naturel avec la linéarisation de l'information. Si on utilise le curseur virtuel, le tableau est lu ligne par ligne. Dans ce cas, les en-têtes de ligne et de colonne doivent être correctement balisés dans le code HTML. Si ce n'est pas fait correctement, JAWS est incapable d'associer une cellule aux en-têtes de ligne et de colonne respectifs. Ce problème est clairement lié à la linéarisation de l'information.

Néanmoins, JAWS permet une exploration du tableau en deux dimensions grâce aux raccourcis clavier « ALT + CTRL + flèche directionnelle ». L'utilisateur peut ainsi se déplacer de case en case en choisissant la direction de son déplacement avec les flèches directionnelles. Le comportement par défaut reste tout de même la linéarisation du tableau et son exploration avec le curseur virtuel, ce qui peut provoquer des problèmes de compréhension (voir 5.3.1.2).

2.2.2 Problèmes liés au non-respect des lignes directrices d'accessibilité

En plus des problèmes liés à la linéarisation de l'information, il en existe de nombreux autres liés au non-respect des lignes directrices d'accessibilité. Nous n'allons pas tous les traiter car le but de ce chapitre n'est pas de faire une revue exhaustive des problèmes d'accessibilité pouvant se produire dans un site Web. De plus, tous ne concernent par les utilisateurs non-voyants. Les

problèmes les plus pénibles pour eux sont ceux supprimant l'accès à du contenu ou supprimant le contexte d'une information. Nous allons donc nous focaliser sur ces derniers.

En ce qui concerne la disparition de contenu, nous pouvons citer les sous-menus apparaissant lorsqu'on déplace la souris au-dessus d'un élément d'un menu. Ces ensembles de liens sont généralement cachés via des attributs CSS. Le problème est qu'il existe plusieurs façons de cacher du contenu via CSS et que toutes ne sont pas compatibles avec JAWS. Par exemple, un élément Web auquel on aurait appliqué l'attribut CSS « display : none » ne sera pas lu avec JAWS alors qu'il l'aurait été si on avait utilisé l'attribut « visibility : hidden ».

Par contre, les contenus cachés peuvent aussi améliorer la navigation des sites avec JAWS quand ils sont bien utilisés. Ajouter des ancres cachées permettant de sauter le menu principal pour aller directement au contenu est très pratique pour éviter d'avoir à relire le menu à chaque page du site Web. On peut aussi avoir ce même genre d'ancres cachées à la fin du contenu pour retourner au menu principal de navigation. Il s'agit encore d'un mécanisme permettant de rendre l'utilisation de JAWS plus rapide, mais cette fois au niveau de l'exploration de la page.

Parmi les autres contenus qui pourraient disparaitre à cause du non-respect des lignes de conduite d'accessibilité, on retrouve l'oubli des textes de remplacement des contenus non-textuels comme les images ou les vidéos. En effet, une image apportant de l'information supplémentaire au contenu doit toujours avoir un texte alternatif contenant cette information. C'est le cas des icônes, des logos ou des graphiques. Il en est de même pour les contenus vidéo qui devraient toujours contenir une piste sonore d'audiodescription, ou un contenu textuel équivalent, en plus de sa piste sonore originale.

En ce qui concerne la perte du contexte du contenu, ce problème est causé par l'utilisation de propriétés CSS à la place des balises HTML. Certains développeurs créent des règles CSS pour simuler des titres ou des listes alors qu'ils devraient utiliser les balises HTML correspondantes pour les indiquer. En effet, ce sont les balises HTML qui indiquent à JAWS le contexte des éléments Web, et non pas les règles CSS qui ne sont pas prises en compte par le lecteur d'écran. Ces contextes sont lus ensuite textuellement par JAWS, sans utilisation d'audicônes.

L'inverse est aussi problématique. Certains développeurs se servent des balises de titre pour formater du texte afin de le rendre plus petit ou plus grand. Ceci n'est pas la fonction de ces

balises et les utiliser seulement pour du formatage pose des problèmes de compréhension lors de l'utilisation de JAWS.

2.2.3 Interaction avec JAWS

Le premier problème rencontré lorsqu'une personne non-voyante veut apprendre à utiliser JAWS, est qu'elle doit préalablement apprendre à se servir tactilement d'un clavier d'ordinateur. En effet, l'interaction avec JAWS se fait exclusivement avec un clavier. Même si de plus en plus de personnes écrivent au clavier sans le regarder, le repère visuel est toujours important lors de son utilisation. Cette observation est encore plus pertinente pour les personnes non-voyantes tardives qui perdent la vision à un âge assez élevé. Ces personnes ne sont pas toujours les plus à l'aise avec les nouvelles technologies. L'apprentissage tactile du clavier peut ainsi prendre plusieurs mois, et ceci sans avoir commencé à apprendre l'utilisation de JAWS.

Une fois que le clavier est maîtrisé tactilement, il faut apprendre à utiliser un ordinateur avec ce seul périphérique. Cet apprentissage implique la mémorisation et l'utilisation d'un nombre très important de raccourcis clavier. D'ailleurs, le manuel fourni lors de l'achat de JAWS ne contient que des raccourcis clavier, sur un total de 19 pages. De plus, la majorité de ces raccourcis impliquent l'utilisation de deux, voire trois touches du clavier en même temps.

Néanmoins, il est possible d'ajouter des guides sur le clavier sous la forme de bulles autocollantes. Ces guides permettent d'identifier les touches les plus utilisées parmi les raccourcis de JAWS. Ils sont souvent placés sur les touches « CTRL », « ENTRÉE », « INSERT », « F » et « J ».

2.3 VoiceOver

À la différence des autres logiciels de lecture de d'écran, VoiceOver est directement intégré dans le système d'exploitation, que ce soit dans Mac OS X sur les ordinateurs Apple, ou dans iOS dans les appareils mobiles iPod Touch, iPhone et iPad (depuis la version 4 d'iOS). Le fait que le lecteur d'écran soit développé par les mêmes personnes développant l'ensemble du système d'exploration favorise son interaction avec ce système. Tout comme JAWS, nous n'avons pas utilisé d'afficheur Braille avec VoiceOver au cours de l'évaluation mais nous avons pu observer le retour Braille grâce à l'interface visuelle.

2.3.1 VoiceOver pour Mac OS

VoiceOver a d'abord été testé sur Mac OS X Lion avec le navigateur Web Safari, même si Google Chrome et Firefox sont maintenant supportés par ce lecteur d'écran, avec encore quelques problèmes pour ce dernier. En effet, VoiceOver ne fonctionnait à ses débuts qu'avec Safari. Normalement, pour qu'une application soit parfaitement compatible avec VoiceOver, Apple recommande qu'elle soit développée via leur API Cocoa, ce qui n'est pas le cas pour Google Chrome et Firefox. Pour éviter tout problème lié à la non-utilisation de cette API, nous avons donc préféré effectuer l'étude avec Safari.

2.3.1.1 Exploration linéaire

VoiceOver est avant tout un logiciel de lecture d'écran linéaire. Ses fonctionnalités de base sont donc très proches de celles de JAWS. Cette exploration linéaire contient les mêmes avantages et défauts que ceux de JAWS (voir 2.2.1) et reste tributaire du respect des lignes directrices d'accessibilité (voir 2.2.2). Il y a néanmoins des différences entre les deux outils concernant cette exploration linéaire.

Une des premières choses qu'on observe lorsqu'on commence à utiliser VoiceOver est la complexité plus importante de ses raccourcis clavier par rapport à JAWS. Ceux-ci exigent au minimum une combinaison de trois touches et peuvent même aller jusqu'à cinq touches. Ces raccourcis sont tous basés sur les touches « CTRL + OPTION » aussi appelées touches « VO » pour VoiceOver. Néanmoins, le fait de toujours utiliser la même base de touches du clavier peut rendre les raccourcis plus faciles à mémoriser.

Par rapport à JAWS, VoiceOver ajoute la notion d'interaction avec des éléments. En effet, VoiceOver interagit par défaut avec le fureteur Web, et non directement avec la page alors que c'est le contraire pour JAWS. Il faut donc choisir d'interagir avec la page pour l'explorer, à l'aide du raccourci « VO + 🡻 ». Ce système d'interaction peut aussi jouer des mauvais tours au cours de l'exploration d'une page Web. Sur la page d'accueil du gouvernement du Québec, il y a un bloc de liens qui n'est pas lu si on ne choisit pas d'interagir avec ce bloc. VoiceOver indique seulement qu'il s'agit d'un « groupe » et il faut attendre quelques secondes sur le bloc de liens pour obtenir automatiquement de l'aide et qu'on nous indique qu'il est possible d'interagir avec celui-ci. Il est donc possible de manquer des liens importants à cause de cette notion d'interaction

avec un groupe qui n'est pas évidente. Par contre, le fait que VoiceOver aide l'utilisateur quand celui-ci ne bouge plus est un point positif. Cela permet de se faire guider quand on semble perdu avec l'exploration linéaire.

D'autres problèmes sont aussi présents par rapport à JAWS. Par exemple, VoiceOver ne reconnaît pas automatiquement la langue d'une page Web même si elle est indiquée au début du code HTML. Il faut changer manuellement la langue de la synthèse vocale quand on se rend compte que la langue de la page Web ne correspond pas à la langue par défaut de la synthèse. Un autre problème concerne les champs de formulaire supplémentaires qui apparaissent lorsqu'un autre champ est modifié (ex. : le champ « Confirmer adresse » qui apparait quand on remplit le champ « Adresse courriel » dans le formulaire de contact du gouvernement du Québec). Ces champs ne sont pas accessibles via l'exploration linéaire de VoiceOver. Il faut recharger la page si on veut y accéder. Enfin, VoiceOver a des problèmes avec les tableaux complexes c.-à-d. ayant au moins deux lignes ou deux colonnes d'en-tête, même quand les lignes directrices d'accessibilité sont respectées. L'association entre les cellules et leurs en-têtes respectifs ne se fait pas correctement.

Tout comme JAWS, VoiceOver permet de filtrer la page en fonction du type d'élément Web pour accélérer la navigation dans l'ensemble du site. Par contre, afficher une fenêtre regroupant les mêmes éléments Web du même type ne fonctionne pas de la même façon. Alors qu'il y a un raccourci par type d'élément Web dans JAWS, il n'y en a qu'un dans VoiceOver « VO + U ». Celui-ci permet d'afficher cette fenêtre et les éléments Web affichés à l'intérieur dépendent du type sélectionné avec le rotor.

Le rotor est un outil spécifique à VoiceOver utilisable seulement avec un pavé tactile. Ce pavé tactile supporte des commandes gestuelles multipoints qui permettent de contrôler VoiceOver, telles que la rotation de deux doigts qui contrôle le rotor (Figure 2-1). En faisant cette rotation, on peut donc sélectionner plusieurs types d'éléments Web (titres, liens, champs de formulaires, etc.) et ce sont les éléments Web de ce type qui sont affichés dans la fenêtre déclenchée par le raccourci « VO + U ». Il est aussi possible de naviguer entre les éléments Web du type sélectionné par le rotor en faisant un geste vers le haut ou le bas avec un seul doigt sur le pavé tactile.

Figure 2-1 : Utilisation du rotor de VoiceOver sous Mac OS

Le pavé tactile supporte aussi d'autres commandes gestuelles multipoints pouvant inclure jusqu'à l'utilisation de trois doigts. Par exemple, le déplacement du curseur virtuel se fait avec des gestes latéraux à un doigt. L'interaction avec les éléments se fait par contre avec des gestes latéraux à deux doigts.

Pour résumer, VoiceOver semble moins efficace que JAWS pour l'exploration linéaire mais cela peut s'expliquer par le fait qu'il est beaucoup plus « jeune » que JAWS. Néanmoins, l'intégration des commandes gestuelles multipoints le rend assez intéressant à utiliser.

2.3.1.2 Exploration spatiale

VoiceOver permet aussi de faire une exploration spatiale d'une page Web. Cette exploration spatiale se fait par l'intermédiaire d'une interaction tactile mais sans générer de retours tactiles. Le curseur virtuel est en fait associé au curseur de la souris. VoiceOver lit donc l'élément Web qui se situe en dessous du curseur de la souris. Cette exploration spatiale peut être absolue ou relative. En utilisant une souris, elle est relative car il n'y a aucun référentiel fixe. On peut jouer sur la vitesse de déplacement du curseur de la souris pour gagner en précision. Mais le fait de ne pas avoir de retour tactile rend compliqué le contrôle du curseur.

Si on utilise un pavé tactile (interne sur les MacBook ou externe avec le Magic TrackPad) ou une tablette graphique (Figure 2-2), l'exploration spatiale devient absolue. La surface de la tablette ou du pavé tactile fait office de référence en se rapportant à la surface de l'écran. L'utilisateur peut donc explorer spatialement l'écran avec son doigt.

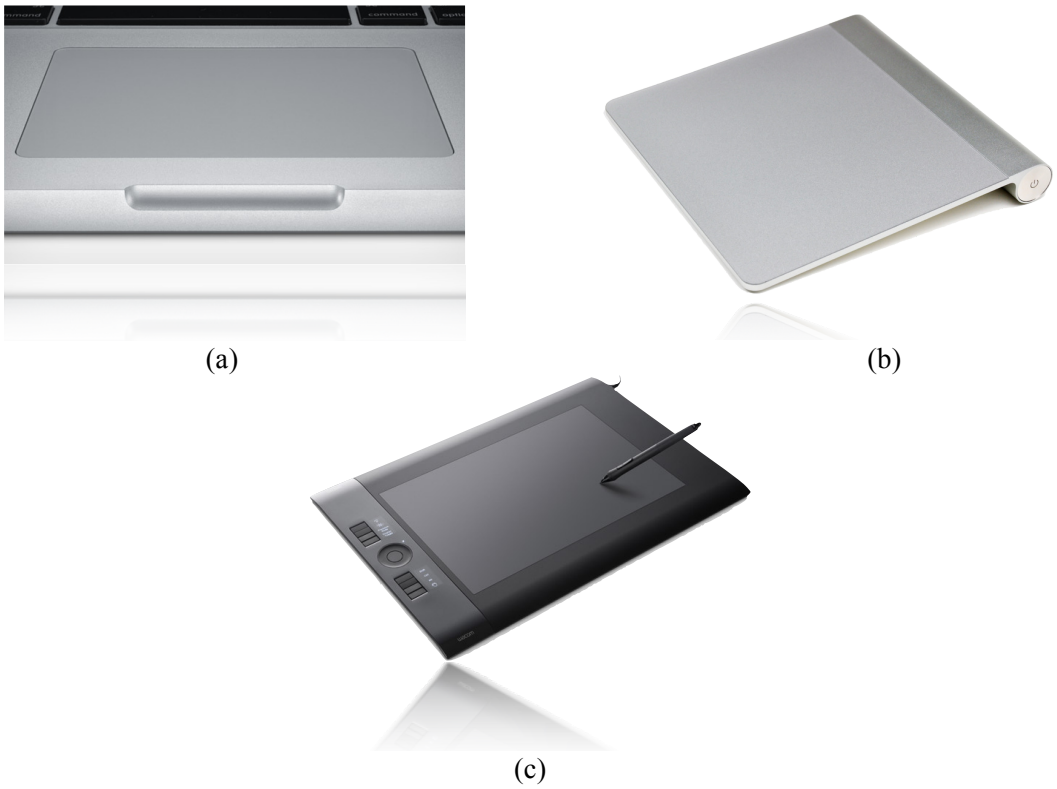


Figure 2-2 : Pavé tactile d'un MacBook (a), Magic TrackPad (b) et tablette graphique Wacom Intuos 4 (c)

En ce qui concerne les pavés tactiles de MacBook ou le Magic TrackPad, le fait que la surface d'exploration soit très petite diminue conséquemment la précision de l'exploration. En effet, un déplacement de quelques millimètres sur la surface provoque un déplacement de plusieurs centimètres sur l'écran. Au cours des quelques tests que nous avons effectués avec un pavé tactile, il était quasiment impossible de l'utiliser correctement et cela prenait un temps bien trop important pour effectuer les tâches que nous avions définies. Nous nous sommes rapidement rendu compte qu'il était préférable d'utiliser le pavé tactile seulement pour les commandes gestuelles multipoints.

Nous pensions pouvoir résoudre, ou au moins diminuer, le problème de la taille de la surface d'exploration en utilisant une tablette graphique. Pour ce faire, nous avons utilisé la tablette graphique Intuos 4 de Wacom qui possède une surface d'exploration de 20,3× 32,5 cm. Pour déplacer le curseur, on peut utiliser une souris ou un stylet. L'utilisation de la souris ressemble plus ou moins à l'expérience avec une souris classique, à l'exception de l'exploration qui est cette fois absolue. L'utilisation du stylet est difficile parce que celui-ci est surtout utilisé pour dessiner,

et non pas pour contrôler une souris. Pour déplacer correctement la souris, il faut que le stylet ne soit pas totalement en contact avec la tablette, et ceci de manière permanente. Cela rend son utilisation vraiment inconfortable au bout de quelques minutes. Ce manque de confort annule l'avantage d'avoir une surface d'exploration plus grande. De plus, il n'est pas possible d'effectuer les commandes gestuelles reconnues par VoiceOver sur ce genre de tablette graphique.

L'exploration spatiale semple donc une bonne idée mais manque de précision avec les pavés tactiles et de confort avec les tablettes graphiques. Le manque de précision pourrait être atténué par l'ajout d'un retour tactile indiquant par exemple les bordures des éléments Web, ou l'ajout de guides sous forme de grille (Ramstein et al., 1996). L'utilisateur ne serait donc plus obligé de se baser seulement sur le retour sonore pour explorer une page Web.

2.3.2 VoiceOver pour appareils mobiles

Contrairement à la version pour Mac OS, VoiceOver pour iOS est compatible avec toutes les applications car tous les développeurs doivent utiliser l'API Cocoa Touch d'Apple pour développer leur application. Nous avons néanmoins préféré tester VoiceOver sur le fureteur Safari qui est fourni par défaut avec iOS. Nous avons aussi testé ce lecteur d'écran sur iPhone et iPad pour voir si la taille de l'écran avait une influence sur l'exploration spatiale.

La grosse différence entre l'exploration spatiale sur iOS et sur Mac OS est qu'elle se fait directement sur l'écran de l'appareil mobile. En effet, un déplacement du doigt sur la surface de l'appareil correspond exactement à un déplacement du curseur sur l'écran. Ce rapport 1:1 entre l'action et le retour rend VoiceOver beaucoup plus précis sur iOS. La navigation au doigt permet donc de déplacer le curseur virtuel de VoiceOver à l'aide d'un simple toucher. Pour activer un élément sélectionné, il suffit de toucher rapidement deux fois n'importe où dans l'écran. C'est à ce moment que l'interaction peut devenir plus compliquée. En effet, même si ce double toucher ne pose pas de problème au cours de l'exploration d'une page, pour activer les hyperliens par exemple, l'utilisation du clavier virtuel de l'appareil mobile devient difficile, surtout sur des petits écrans comme ceux des téléphones intelligents. Il faut toucher une lettre pour la sélectionner puis toucher deux fois l'écran pour l'activer. Il faut donc toucher trois fois l'écran pour entrer une lettre. Ceci n'est pas pratique lorsqu'il faut remplir les champs de texte des formulaires. L'expérience se retrouve ainsi ralentie.

En plus de l'exploration spatiale, VoiceOver pour iOS permet aussi l'exploration linéaire en prenant en compte les commandes gestuelles multipoints. Les gestes sous-jacents à ces commandes sont exactement les mêmes que ceux que l'on peut trouver sous Mac OS, y compris le rotor (Figure 2-3). Le seul ajout au rotor concerne la sélection de la langue de la synthèse vocale. Cet ajout est grandement apprécié car le changement de langue n'est pas aisé dans la version pour Mac OS.

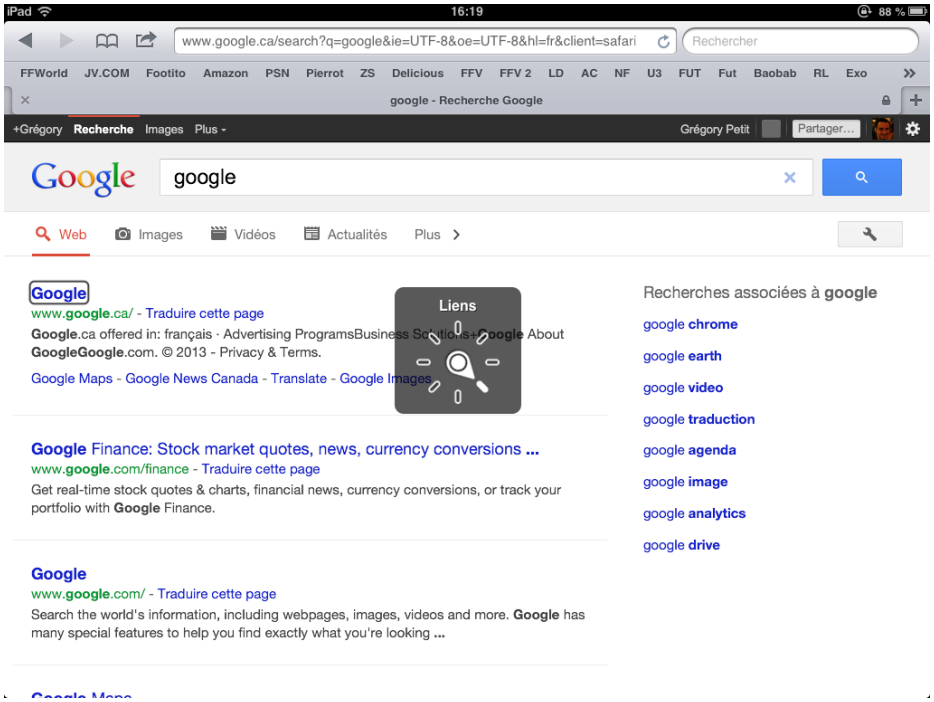


Figure 2-3 : Utilisation du rotor sous iOS

L'exploration linéaire est donc toujours possible à l'aide des commandes gestuelles. Après consultation de plusieurs personnes non-voyantes utilisant VoiceOver sur iOS (presque exclusivement sur iPhone), l'exploration linéaire via commandes gestuelles semble préférée à l'exploration spatiale. Ces personnes nous ont indiqué qu'elles n'utilisaient quasiment jamais l'exploration spatiale sur leur appareil mobile. De plus, l'utilisation du rotor accélère grandement leur vitesse d'exploration. Néanmoins, nous avons observé plusieurs bogues au cours de l'utilisation des commandes gestuelles quand celles-ci sont utilisées rapidement. Le curseur virtuel redémarre aléatoirement au tout début de la page alors qu'il n'est pas du tout rendu à la fin. Il serait intéressant de savoir si l'exploration spatiale serait plus adoptée par les personnes non-voyantes sur des iPad. Malheureusement, nous n'avons pas rencontré de personnes non-voyantes utilisant ces tablettes.

L'exploration linéaire reste aussi possible à l'aide d'un clavier Bluetooth. Dans ce cas, la majorité des raccourcis clavier disponibles sous Mac OS sont utilisables.

2.4 La souris iFeel

Au début de cette thèse, nous souhaitions aussi évaluer l'utilisation de la souris iFeel (Logitech, 2013) que nous avons abordée précédemment dans la section 1.3.2.1. Cette souris ressemble à une souris ordinaire mais peut générer des sensations tactiles quand on passe au-dessus de certains éléments Web, comme par exemple des liens. Cette souris n'est utilisable que sous Windows et Logitech a arrêté son développement. Le problème est que ce genre de technologie ne donne aucun retour sonore.

Nous avions pensé jumeler cette souris à l'utilisation d'un logiciel de lecture d'écran mais aucun logiciel de lecture d'écran n'utilise la souris comme moyen d'interaction. Cette souris reste donc plus un gadget et n'aide pas à améliorer l'accessibilité des pages Web aux personnes non-voyantes. L'utilisation de la souris iFeel avec le lecteur d'écran VoiceOver, particulièrement son exploration spatiale, serait beaucoup plus pertinente mais les deux outils ne fonctionnent pas sur le même système d'exploitation.

2.5 Synthèse

Dans ce chapitre, nous avons utilisé deux logiciels de lecture d'écran, JAWS et VoiceOver, pour identifier leurs points positifs et leurs points négatifs. Cette évaluation ne concernait que leur utilisation avec des pages Web. L'exploration linaire possible avec ces lecteurs permet de naviguer rapidement dans des sites Internet, encore plus si ce sont des sites où l'utilisateur a l'habitude d'aller. Cette rapidité est due principalement au fait de pouvoir filtrer les pages en fonction du type d'élément Web souhaité. Mais lorsqu'il s'agit d'explorer une page Web, l'exploration linéaire devient plus lente. Ce type d'exploration peut aussi provoquer une perte de contexte si l'ordre logique de lecture n'est pas respecté dans le code HTML de la page Web.

De plus, les lecteurs d'écran restent encore tributaires du respect des lignes de conduite d'accessibilité. Même si concevoir un site accessible ne demande pas beaucoup de travail en plus, le respect des lignes de conduite d'accessibilité n'est pas assez ancré dans les mœurs des

entreprises pour garantir leur bonne application partout. Le non-respect de ces lignes de conduite peut rapidement devenir problématique pour les utilisateurs non-voyants.

Le lecteur d'écran VoiceOver a la particularité de proposer une alternative à l'exploration linéaire : l'exploration spatiale des pages Web. Ce type d'exploration est une idée très intéressante mais qui n'est pas totalement exploitée. Son manque de précision pour la version Mac OS et de rapidité pour la version iOS ne joue pas en sa faveur par rapport à l'exploration linéaire. D'ailleurs l'exploration spatiale n'est pas vraiment adoptée par les utilisateurs non-voyants qui lui préfèrent l'exploration linéaire, aidée par des commandes gestuelles multipoints. Nous nous sommes demandé si le fait que le retour soit seulement sonore était un désavantage pour l'exploration spatiale. Le retour sonore ne donne pas vraiment d'encadrement à l'utilisateur au cours de l'exploration spatiale. C'est pour cela que nous nous sommes penchés sur l'exploration spatiale à l'aide de retour sonore et tactile. L'ajout de cette modalité tactile pourrait améliorer la précision de l'exploration spatiale ainsi qu'éventuellement sa rapidité.

CHAPITRE 3 PROBLÉMATIQUE, OBJECTIF ET HYPOTHÈSES

Dans le chapitre précédent, nous avons pu mettre en évidence que les logiciels de lecture d'écran ont des limites lorsqu'il s'agit de donner accès au Web aux utilisateurs non-voyants. Ces limites viennent en partie du fait qu'un lecteur d'écran ne permet qu'une navigation linéaire dans la page Web. Cela gêne la navigation dans la page Web car la structure globale de celle-ci est inexistante pour l'utilisateur non-voyant. De plus, cette navigation linéaire provoque des frustrations dans la compréhension des formulaires, des hyperliens et boutons, et de tout ce qui touche à l'association d'informations entre elles. Enfin, la frustration engendrée par l'utilisation des lecteurs d'écran provoque une perte de temps conséquente. Il est donc pertinent de proposer une alternative à la navigation linéaire : une navigation spatiale. En effet, l'utilisateur ne devrait pas devoir suivre un chemin prédéfini car il doit se faire son propre modèle mental de la page Web et non pas avoir un modèle mental déterminé par le système. C'est encore plus pertinent dans le cas des personnes non-voyantes car elles ne se font pas la même image mentale des choses, qu'elles soient non-voyantes de naissance ou non (voir 1.2.1.3).

Actuellement, l'exploration spatiale est possible pour un utilisateur non-voyant qui va sur le Web, grâce à VoiceOver sous Mac OS. Néanmoins, même si cette exploration spatiale est possible via une entrée tactile, aucun retour tactile n'est disponible ce qui rend l'exploration très difficile. Nous avons d'ailleurs remarqué que les personnes non-voyantes préféraient utiliser des gestes tactiles plutôt que de naviguer spatialement avec VoiceOver.

Aucune étude n'a été faite pour comparer une exploration linéaire avec une exploration spatiale sur le Web. On ne sait donc pas si la non-linéarité améliorera de manière significative l'exploration du Web pour les utilisateurs non-voyants même si cela nous nous parait logique. Néanmoins, une étude a démontré que la combinaison des modalités sonores et tactiles était plus efficace que la seule utilisation de la modalité sonore (Dufresne et al., 1995) dans les interfaces graphiques pour les utilisateurs non-voyants. Cela correspond exactement à notre problème car la linéarité de l'exploration d'une page Web est associée à la seule utilisation de retours sonores et la non-linéarité est associée à l'utilisation combinée des modalités sonores et tactiles.

Pour permettre cette exploration spatiale, et donc pour redonner une notion de structure globale de la page Web aux utilisateurs non-voyants, l'ajout d'une seconde modalité de sortie à la modalité sonore nous semble primordiale. Pour redonner cette notion de structure, la modalité

tactile est la plus appropriée. Celle-ci est beaucoup utilisée pour donner accès aux images aux utilisateurs non-voyants. Ces images tactiles leur permettent de sentir les différents contours de l'image et d'identifier les différentes parties composant l'image. La modalité tactile leur permet donc d'accéder à la structure de l'image et c'est exactement ce qu'on veut faire avec les pages Web.

C'est pour cette raison que nous pensons qu'une exploration spatiale est étroitement liée à une exploration avec des retours tactiles et sonores. Ainsi, dans la suite de cette thèse, nous n'allons pas toujours préciser que l'exploration est audio-tactile car cette notion va de pair avec la notion d'exploration spatiale dans notre recherche.

3.1 Objectif de la recherche

L'objectif de cette recherche est d'améliorer l'accès au Web pour les utilisateurs non-voyants, en leur permettant d'explorer des pages Web de manière spatiale, c.-à-d. à l'aide de retours tactiles et sonores. Cette exploration devra se faire avec le plus de liberté possible pour que l'utilisateur ne soit pas obligé de suivre une méthode d'exploration préétablie, comme c'est le cas avec les lecteurs d'écran linéaires. Il pourra ainsi accéder au document de la manière qu'il le souhaite sans forcément devoir accéder à toutes les informations et il devra pouvoir spécifier ses préférences. C'est cette liberté via la modalité tactile que nous voulons ajouter à l'expérience utilisateur au cours de la navigation Web. Considérant le niveau de frustration des utilisateurs des logiciels de lecture d'écran (Lazar et al., 2007; Web AIM, 2013), il est très pertinent de donner une alternative d'exploration plus libre et plus appropriée au monde du Web. La non-linéarité permettra de diminuer le nombre d'éléments à mémoriser pour l'utilisateur car, grâce à l'ajout de la notion de structure dans la page Web, nous prévoyons qu'il aura une meilleure idée globale de la page Web et pourra donc mieux associer les informations de cette page entre elles.

Pour atteindre notre objectif, il faut concevoir, prototyper puis évaluer un outil audio-tactile permettant l'exploration spatiale. L'évaluation nous permettra de comparer notre outil aux lecteurs d'écran utilisés. Ces derniers sont la référence actuelle pour les personnes non-voyantes afin d'accéder au Web. Il est donc normal de comparer notre outil à cette référence. Nous pourrions aussi comparer notre outil audio-tactile à un outil audio permettant aussi l'exploration spatiale (par exemple, VoiceOver). Néanmoins l'étude menée par Dufresne et al. a mis en relief

le fait qu'une exploration spatiale et multimodale d'interface graphiques, basée sur des retours tactiles et sonores, était plus efficace qu'une exploration spatiale, basée uniquement sur des retours sonores (Dufresne, Martial, & Ramstein, 1995). De plus, nous avons vu dans notre étude empirique (voir 2.3.1.2) que l'exploration spatiale via VoiceOver ne semblait pas adoptée par les personnes non-voyantes car les modalités y sont trop restreintes,

Comme outil audio-tactile, nous avons choisi le Tactograph en considérant les points forts énoncés dans la section 1.3.2.4. Ces points forts sont la finesse et la diversité des rendus tactiles, sa compatibilité avec du contenu dynamique et l'utilisation de fichiers XML dont la syntaxe est très proche du HTML. Ces points forts font en sorte que le Tactograph est l'outil le plus adapté à une exploration spatiale précise et complète. Mais le Tactograph ne peut pas être utilisé tel quel car il n'est pour l'instant adapté qu'à l'exploration audio-tactile d'images. Il doit donc être adapté. De plus, le codage HTML des pages Web doit aussi être adapté pour permettre une exploration tactile. Il faudra donc réaliser cette double adaptation pour permettre la communication entre le Tactograph et le Web. Nous nous sommes focalisés sur des sites de services en ligne ou informationnels. Ayant un partenariat avec les Services gouvernementaux du Québec, nous avons travaillé sur des sites de services en ligne qui s'adressent à de nombreux citoyens québécois.

3.2 Problèmes d'accessibilité pouvant être réglés par une exploration spatiale

Nous prévoyons que l'exploration spatiale pourra régler certains problèmes d'accessibilité aux pages Web.

Le premier problème d'accessibilité concerne l'ordre logique de lecture qui va de gauche à droite et de haut en bas. L'exploration spatiale d'une page écran ne sera pas affectée si cet ordre logique n'est pas respecté dans le code HTML. Par exemple, si le menu principal se situe en haut à gauche de la page, un utilisateur voyant devrait le remarquer rapidement. Mais si ce menu est codé à la fin de la page HTML et inséré visuellement en haut à gauche de la page avec une feuille de style de CSS, alors il ne sera lu qu'à la fin par les lecteurs d'écran. L'exploration spatiale gardera l'ordre logique de lecture en affichant spatialement le menu à gauche.

Le second problème concerne l'accessibilité des formulaires. L'exploration spatiale ne sera pas affectée si l'association entre les étiquettes et les champs du formulaire a été omise dans le code HTML. Avec un lecteur d'écran linéaire, l'étiquette ne sera pas lue lorsque ce dernier arrivera sur le champ de formulaire. De plus, il peut arriver que toutes les étiquettes soient lues l'une à la suite de l'autre, suivies de la lecture des champs. Avec une exploration spatiale, il sera toujours possible d'associer spatialement un champ avec son étiquette, comme le font les utilisateurs voyants. L'étiquette est toujours à gauche ou au-dessus du champ qui lui est associée.

La relation entre les différents éléments Web pourrait aussi être améliorée par l'exploration spatiale. Certains éléments sont parfois reliés entre eux visuellement sans que cela soit considéré par les lecteurs d'écran à cause de la linéarisation de l'information. On peut penser par exemple à un texte situé à côté d'une image mais qui ne sera pas lu juste avant ou juste après l'image au cours de l'exploration linéaire. Ce genre de problème serait inexistant avec l'exploration spatiale.

Enfin, une exploration spatiale pourra aussi faciliter les échanges entre utilisateurs voyants et non-voyants. En effet, il n'est pas rare qu'un utilisateur non-voyant reçoive des indications spatiales de la part d'un utilisateur voyant, qu'il s'agisse d'un ami assis à côté de lui ou d'une personne d'un service technique au téléphone. Au cours d'entretiens avec des personnes non-voyantes, on nous a indiqué que des phrases comme « Cliquez sur le lien en haut à gauche » ou « Vous trouverez cet élément dans le menu à gauche » étaient fréquentes au cours de leurs échanges avec des personnes voyantes. Encore une fois, l'exploration spatiale pourrait s'avérer utile dans ce genre de situation où l'utilisateur doit traiter une information spatiale qui lui est indiquée par une personne voyante.

3.3 Hypothèses

Dans les hypothèses, l'efficacité d'un outil se mesure par le taux de succès de l'utilisateur à effectuer les tâches avec celui-ci. L'efficience se mesure par le niveau de difficulté rencontrée par l'utilisateur lorsqu'il effectue ces tâches et par le temps d'exécution de celles-ci. Chaque hypothèse est divisée en deux hypothèses partielles en fonction de ces deux facteurs.

L'hypothèse principale de notre recherche et les deux hypothèses partielles qui en découlent sont les suivantes :

Hypothèse principale

L'exploration spatiale d'un site Web est plus efficace et plus efficiente que l'exploration linéaire pour des utilisateurs non-voyants.

Hypothèse principale A

L'exploration spatiale d'un site Web est plus efficace que l'exploration linéaire pour des utilisateurs non-voyants.

Hypothèse principale B

L'exploration spatiale d'un site Web est plus efficiente que l'exploration linéaire pour des utilisateurs non-voyants.

Au cours de l'expérience que nous allons réaliser dans le cadre de cette thèse, il faudra tenir compte du fait que les utilisateurs non-voyants sont habitués à l'exploration linéaire depuis longtemps et qu'ils n'ont sans doute jamais utilisé l'exploration spatiale dans un site Web. Considérant la grande différence d'expérience des utilisateurs entre les deux types d'exploration, notre hypothèse principale peut sembler très forte. Cette différence devra être prise en compte lors du test de notre hypothèse et de l'interprétation des résultats. Il faut être conscient du fait qu'un biais d'expérience est presque toujours présent lorsqu'on souhaite comparer un nouveau outil ou système avec son équivalant qui existe depuis longtemps. Il est donc important de comparer une nouvelle forme d'exploration avec l'exploration qui fait présentement référence. Il en est de même concernant le biais généré par la différence de maturité entre les outils permettant les deux types d'exploration. Cette différence de maturité peut clairement influer sur les résultats de notre évaluation en faveur de l'exploration linéaire. En effet, les outils permettant l'exploration linaire existent depuis deux décennies et ont eu le temps de se perfectionner par rapport aux outils permettant l'exploration spatiale. Malgré le biais inévitable causé par la différence d'expérience des utilisateurs et de maturité des outils d'exploration, nous avons de grandes attentes envers l'exploration spatiale du Web pour les personnes non-voyantes. D'où notre hypothèse principale. Il est néanmoins possible que l'exploration linéaire soit plus rapide que l'exploration spatiale durant les tests, parce que nous sommes sur du court terme. Nous espérons tout de même que l'exploration spatiale soit aussi efficace que l'exploration linéaire.

De plus, nous nous attendons à ce que les scores d'efficacité et d'efficience soient différents selon que le site Web est parfaitement accessible ou non avec les lecteurs d'écran. Nous sommes d'avis que l'exploration spatiale sera moins affectée que l'exploration linéaire par le non respect

des lignes de conduite d'accessibilité. Comme nous l'avons vu dans le Chapitre 2, l'exploration linéaire peut, elle, être affectée négativement. Nous émettons donc la sous-hypothèse suivante, ainsi que les deux sous-hypothèses partielles qui en résultent :

Sous-hypothèse 1

L'exploration spatiale est plus efficace et plus efficiente que l'exploration linéaire dans les sites Web non-accessibles utilisés par des utilisateurs non-voyants.

Sous-hypothèse 1A

L'exploration spatiale est plus efficace que l'exploration linéaire dans les sites Web non-accessibles utilisés par des utilisateurs non-voyants.

Sous-hypothèse 1B

L'exploration spatiale est plus efficiente que l'exploration linéaire dans les sites Web non-accessibles utilisés par des utilisateurs non-voyants.

Nous nous attendons aussi à des différences de niveaux d'efficacité et d'efficience des outils selon le type de tâche que l'utilisateur aura à effectuer. Nous sommes d'avis que l'exploration spatiale avantagera l'utilisateur différemment selon qu'il s'agisse d'une tâche de recherche d'information spécifique ou de remplissage de formulaire, parce que les tâches de recherche d'information demandent beaucoup de navigation dans un site Web. Nous avons vu dans le Chapitre 2 que l'exploration linaire permettait de filtrer une page Web en fonction d'un type d'élément Web souhaité, ce qui pouvait grandement accélérer la navigation entre les pages d'un même site. Les tâches de remplissage de formulaire demandent, elles, plus d'exploration au sein d'une même page. Il est possible que l'exploration spatiale soit avantagée dans ce type de tâche. Nous émettons donc les deux sous-hypothèses suivantes, ainsi que les quatre hypothèses partielles qui en résultent :

Sous-hypothèse 2

L'exploration linéaire est plus efficace et plus efficiente que l'exploration spatiale pour la tâche de recherche d'information dans les sites Web par des utilisateurs non-voyants.

Sous-hypothèse 2A

L'exploration linéaire est plus efficace que l'exploration spatiale pour la tâche de recherche d'information dans les sites Web par des utilisateurs non-voyants.

Sous-hypothèse 2B

L'exploration linéaire est plus efficiente que l'exploration spatiale pour la tâche de recherche d'information dans les sites Web par des utilisateurs non-voyants.

Sous-hypothèse 3

L'exploration spatiale est plus efficace et plus efficiente que l'exploration linéaire pour la tâche de remplissage de formulaire dans des sites web pour des utilisateurs non-voyants.

Sous-hypothèse 3A

L'exploration spatiale est plus efficace que l'exploration linéaire pour la tâche de remplissage de formulaire dans des sites web pour des utilisateurs non-voyants.

Sous-hypothèse 3B

L'exploration spatiale est plus efficiente que l'exploration linéaire pour la tâche de remplissage de formulaire dans des sites web pour des utilisateurs non-voyants.

Dans cette recherche, la période d'apparition de la cécité (de naissance ou tardive) chez une personne est une variable contrôlée. Elle ne fait donc pas l'objet d'hypothèse ; nous allons tout de même tester son effet sur l'efficacité et l'efficience des participants dans leur utilisation avec les deux outils.

CHAPITRE 4 TACTOWEB : UN FURETEUR AUDIO-TACTILE À L'USAGE DES PERSONNES NON-VOYANTES

Dans le Chapitre 2, nous avons observé que les logiciels de lecture d'écran donnaient un accès rapide au contenu des pages Web mais qu'ils n'étaient pas optimaux lors de l'exploration en détails d'une page Web. Si le site Web respecte les lignes directrices d'accessibilité, l'utilisation du lecteur d'écran se passe globalement sans problème. Néanmoins, la loi québécoise oblige seulement les sites publics, c.-à-d. rattachés au gouvernement à appliquer ces lignes directrices d'accessibilité. Le fait de ne pas appliquer certaines lignes directrices pose évidemment des problèmes d'accessibilité. Parmi ceux-ci, certains sont directement liés au fait que les lecteurs d'écran ne permettent qu'une exploration linéaire (ou séquentielle) de la page Web. Nous avons donc voulu développer un outil permettant aux personnes non-voyantes d'explorer spatialement une page Web : TactoWeb. Cette exploration en deux dimensions se fait à l'aide de retours tactiles et sonores.

TactoWeb est un logiciel récupérant les codes HTML et CSS d'une page Web à partir d'une adresse URL (Uniform Resource Locator). Il traite ensuite ces codes pour générer une page Web audio-tactile où les éléments qui la composent ont la même taille et localisation que pour un affichage visuel dans un fureteur Web classique. Nous pouvons donc considérer TactoWeb comme un fureteur Web audio-tactile. L'exploration d'une page affichée par TactoWeb se fait par l'intermédiaire d'une version améliorée du Tactograph générant des vibrations et des ondulations en fonction de la position du module tactile sur une surface d'exploration en deux dimensions (voir 1.3.2.4 pour la version originale du Tactograph).

Au cours de ce chapitre, nous nous penchons dans la section 4.1 sur le matériel que nous avons utilisé pour développer ce fureteur audio-tactile. Ensuite, nous décrivons dans la section 4.2 les différentes fonctionnalités que nous avons développées dans TactoWeb. Enfin, la section 4.3 présente le tutoriel que nous avons développé pour expliquer la navigation spatiale et les différentes fonctionnalités de TactoWeb aux personnes non-voyantes.

4.1 Matériel utilisé

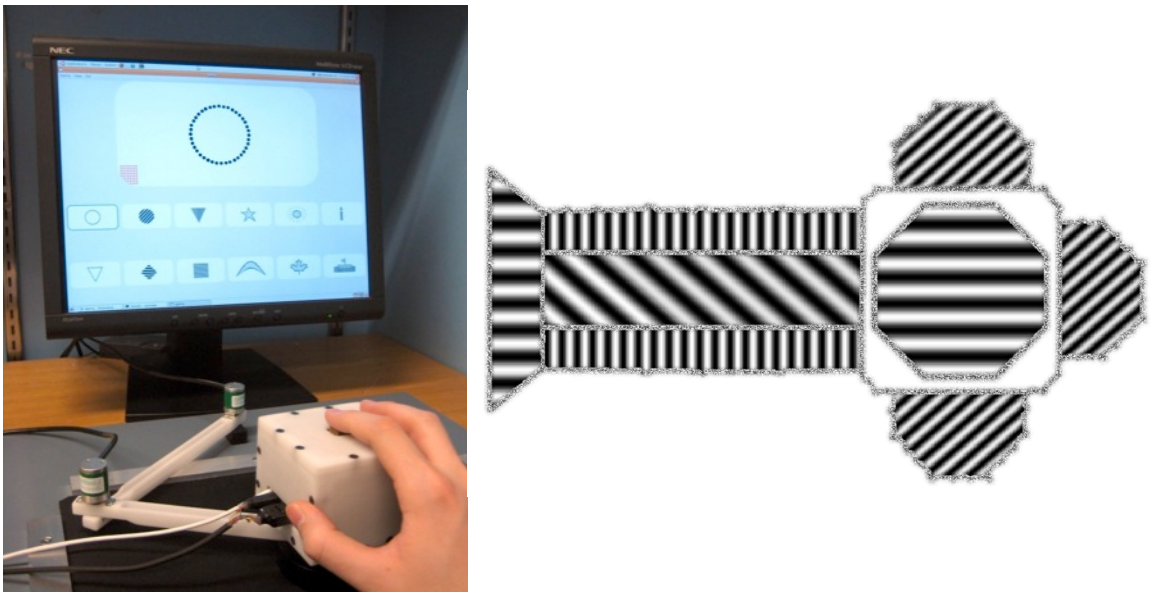
4.1.1 Amélioration du Tactograph

Le Tactograph est un périphérique tactile d'entrée et de sortie en même temps. Il génère des sensations tactiles en fonction de l'endroit où le module tactile se trouve sur un référentiel absolu. Le module tactile fait donc office de souris se déplaçant sur une surface d'exploration fixe en deux dimensions.

4.1.1.1 Le Tactograph au début de la thèse

La version du Tactograph que nous avons utilisée au cours de notre précédente recherche (Petit, Dufresne, Lévesque, Hayward, et al., 2008) fonctionnait correctement mais pouvait aussi être améliorée. Tout d'abord, la surface d'exploration était un peu trop glissante ce qui pouvait rendre son contrôle difficile. En effet, rester en position statique pouvait devenir un peu ardu car il n'y avait aucune résistance qui empêchait le module tactile de bouger sur la surface. De plus, la surface d'exploration était relativement petite, ce qui a été souligné par les participants durant l'expérimentation sur les graphismes tactiles de manuels scolaires. En effet, les images tactiles (ex. : Figure 4-1a) pouvaient être particulièrement grande par rapport à la taille de la surface d'exploration ce qui réduisait la précision de l'exploration spatiale. Enfin, la prise en main du module tactile n'était pas confortable car on ne pouvait pas le prendre pleinement dans la main et le bras de l'utilisateur devait toujours être suspendu pour manipuler correctement l'appareil.

Au début de cette thèse, le laboratoire Haptics de l'Université McGill nous a prêté une nouvelle version du Tactograph. Le module tactile comprenait une nouvelle version du STReSS² avec une cellule tactile de 9 x 11 mm. Cette cellule était composée de 64 actionneurs (8 x 8). La distance entre chaque actionneur était cette fois de 1,2×1,4 mm avec toujours une courbure de 0,1 mm vers la gauche ou vers la droite. Le STReSS² était ensuite intégré dans un cube blanc, puis monté sur un nouveau charriot permettant d'explorer une surface plus grande : 24 x 17 cm (voir Figure 4-1b). La position de la cellule tactile sur la surface d'exploration est calculée à l'aide des angles du bras reliant le charriot à la surface.



(a) (b)

Figure 4-1 : Seconde version du Tactograph (a) et image tactile adaptée d'un manuel scolaire (b)

Cette seconde version du Tactograph corrigeait le problème de la taille de la surface d'exploration mais les deux autres problèmes étaient toujours présents. Le module tactile était toujours aussi glissant sur la surface. Le cube blanc améliorait un peu la prise en main même s'il n'était pas forcément plus confortable à cause des arrêtes très prononcées. De plus, même si le bras ne devait plus être suspendu pour tenir le module tactile, l'utilisateur devait « casser » son poignet pour le saisir.

4.1.1.2 Modification de la surface d'exploration

Nous avons donc décidé de travailler avec TactileLabs, composé d'anciens membres du laboratoire Haptics continuant de développer le module tactile, maintenant appelé Latero (TactileLabs, 2013), pour améliorer l'ergonomie physique du Tactograph. TactileLabs nous a fourni par la suite une version améliorée du Tactograph en suivant nos recommandations.

Nous avons choisi de garder un référentiel fixe, soit une surface parcourue par un module tactile fixé à un bras, pour deux raisons. La première est d'ordre technique pour calculer de la manière la plus précise possible la position du module tactile sur la surface. La seconde raison est qu'un référentiel fixe semble nécessaire en absence de retour visuel. Il se rapporte plus à l'usage fait par les personnes non-voyantes pour lire du Braille ou explorer un graphisme tactile. En effet, la majorité des lecteurs Braille le font en utilisant les deux mains car c'est généralement la manière

dont il est enseigné et c'est aussi plus rapide que la lecture Braille à une seule main (Mousty & Bertelson, 1985). La lecture Braille à deux mains se fait en alternant entre deux méthodes de lecture : l'exploration conjointe et l'exploration disjointe. L'exploration conjointe consiste à garder ses deux mains l'une à côté de l'autre au cours de la lecture. Mais lorsque le lecteur passe en exploration disjointe, la main gauche se place à la ligne suivante alors que le lecteur continue d'utiliser sa main droite pour lire la ligne courante. Certains lecteurs gardent même leur main gauche au début de chaque ligne et laisse leur main droite lire la majorité de la ligne courante (Bertelson, Mousty, & D'Alimonte, 1985). Le fait de garder sa main gauche sur le bord gauche de la page permet de garder un référentiel au cours de la lecture et de ne pas sauter des lignes. Le lecteur sait donc où il est rendu dans la page à l'aide de ce référentiel fixe. C'est ce même genre de référentiel fixe qui pourra aider les utilisateurs à savoir où ils sont spatialement dans la page Web avec le Tactograph, en gardant une main sur le module tactile et l'autre sur le bord de la surface comme point de repère.

En ce qui concerne les matériaux utilisés, la surface est maintenant composée d'aluminium anodisé afin d'augmenter le frottement entre elle et le module tactile. Le fait d'anodiser la surface permet une meilleure résistance à l'abrasion. Une fine couche de polytetrafluoroethylene (PTFE) a ensuite été appliquée sur la surface pour obtenir une résistance au déplacement comparable à celle d'une souris d'ordinateur sur un bureau. La nouvelle version du Tactograph permet d'explorer une surface de 27.9 x 21.6 cm, soit la taille d'une feuille de papier au format Lettre (Figure 4-2).

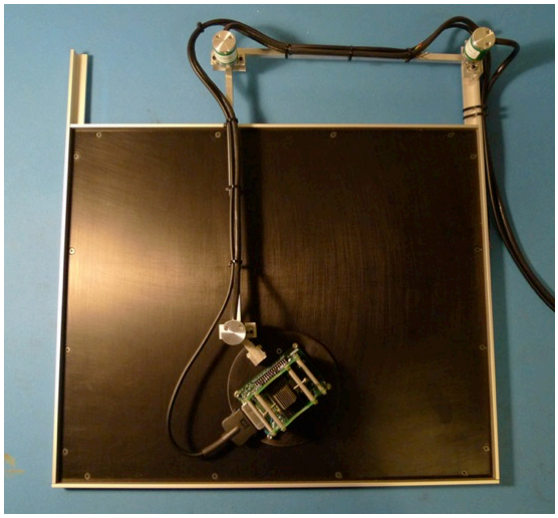


Figure 4-2 : Nouvelle version du Tactograph sans la coquille du module tactile

4.1.1.3 La coquille du module tactile

Le Tactograph nous a d'abord été livré sans coquille pour le module tactile. TactileLabs nous a demandé de leur fournir une maquette de la coquille intégrant les principales caractéristiques que nous souhaitions pour qu'ils puissent ensuite faire un modèle 3D. Nous avons produit cette maquette à l'aide d'une pâte à modeler qui se solidifie en une journée une fois le façonnage terminé (Crayola Model Magic). Nous nous sommes basés sur le design d'une souris d'ordinateur classique pour produire la coquille du module tactile.

La principale contrainte que nous avions pour produire la coquille était la taille du Latero (8 x 6 x 10 cm) ainsi que ses deux points de fixation, à savoir l'attache physique avec le bras reliant le module tactile et la surface, et la connexion électrique avec le câble reliant l'appareil à sa boite de contrôle. La grande taille du Latero nous a forcés à concevoir une coquille presque deux fois plus grosse qu'une souris classique. Nous avons gardé le même profil courbé d'une souris pour permettre une bonne prise en main de l'appareil. Nous avons aussi arrondi chaque arrête pour que le module tactile soit plus proche d'un appareil qu'on pourrait trouver dans le commerce. Nous voulions aussi ajouter des boutons sur le module tactile pour activer certaines fonctionnalités de TactoWeb. Nous avons d'ailleurs ajouté l'emplacement de ces boutons sur la maquette. La Figure 4-3 représente la version finale de la maquette.

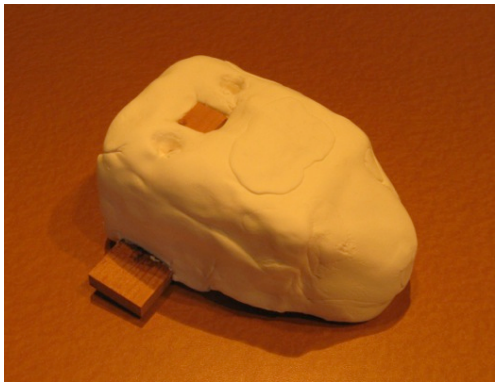
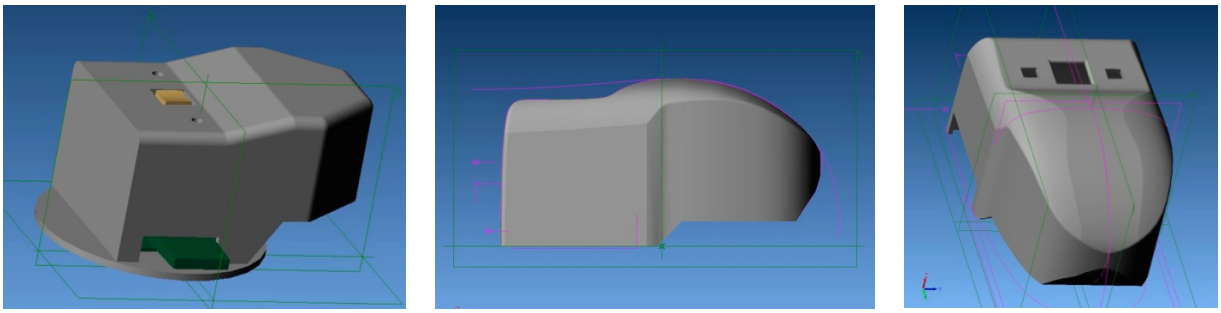


Figure 4-3 : Maquette en pâte à modeler de la coquille du Latero

Une fois la maquette envoyée à TactileLabs, un premier rendu 3D nous a été envoyé sous forme de fichier Blender (Figure 4-4a). Après plusieurs échanges avec TactileLabs concernant la longueur et certaines courbes de la coquille, nous sommes arrivés au design final de la coquille du Tactograph. (Figure 4-4b et c).



(a) (b) (c)

Figure 4-4 : Premier modèle 3D (a) et modèle final 3D (b et c) de la coquille du module tactile du Tactograph

TactileLabs a ensuite procédé à la stéreolithographie (impression 3D), de la coquille en résine via un procesus de photopolymérisation puis est venu l'installer sur le Tactograph. La Figure 4-5 présente la version actuelle du Tactograph qui possède donc une surface d'exploration plus grande et une meilleure prise en main du module tactile à l'aide d'une coquille de protection semblable à une souris.

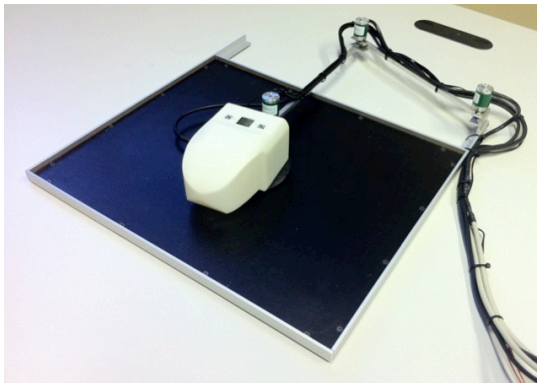


Figure 4-5 : Version actuelle du Tactograph

4.1.2 Plateforme de développement

La contrainte principale que nous avions pour créer TactoWeb est que nous nous ne pouvions le programmer qu'en C++. En effet, les librairies permettant de contrôler le Tactograph (STRESSD) et de générer des sensations tactiles (STRESSR) ne sont disponibles que dans ce langage de programmation. Une autre contrainte était que le fureteur fonctionne sous une plateforme UNIX, soit obligatoirement LINUX et éventuellement MAC OS. Au début du projet, STRESSD et STRESSR ne fonctionnaient que sous LINUX. La dernière contrainte était que l'ordinateur

contrôlant TactoWeb devait être très puissant. En effet, TactoWeb doit tourner à une fréquence très proche du temps réel dans le but d'avoir un retour tactile instantané par rapport aux mouvements du module tactile sur la surface d'exploration.

Pour créer TactoWeb, nous pouvions soit faire une extension d'un fureteur existant, soit créer un nouveau fureteur strictement audio-tactile. Nous nous sommes tout d'abord penchés sur l'option de l'extension pour un fureteur existant. Le problème fut qu'aucun fureteur ne supportait le C++ comme langage de programmation d'une extension, même si le fureteur était entièrement codé en C++. Néanmoins, certains fureteurs développés en C++ sont libres de droit et donnent accès gratuitement à leur code source. Nous avons donc essayé de développer TactoWeb au sein même d'un fureteur libre de droit en regardant les codes sources de Chromium (Google, 2013), un projet parallèle au fureteur Google Chrome, et Midori (Twotoasts, 2013). Ces deux fureteurs sont libres de droits et développés en C++.

Dans le cas de Chromium, le code était trop complexe. Nous ne disposions pas des ressources nécessaires en programmation pour comprendre et exploiter correctement l'immensité de son code source, même en nous focalisant seulement sur WebKit, le moteur de rendu graphique. Nous pensions que Midori serait plus simple à comprendre car il possède beaucoup moins de fonctionnalités que Chromium. Mais Midori utilise aussi WebKit comme moteur de rendu graphique, d'où une complexité toujours importante. De plus, nous n'avions pas besoin de la puissance et de la complexité de WebKit car seules la taille et la localisation des éléments Web à rendre nous étaient utiles pour un rendu audio-tactile.

Nous nous sommes donc orientés vers le développement de notre propre fureteur. Le développement C++ a commencé sous Linux via l'environnement de développement KDevelop. Mais pour faciliter le développement, les librairies STRESSD et STRESSR furent adaptées par Vincent Lévesque pour fonctionner aussi sous MAC OS. Le développement sous ce système d'exploitation a d'abord commencé sous Eclipse, puis a basculé sur Xcode. Nous sommes passés sous Xcode car cet environnement de développement est plus approprié au langage de programmation C++ qu'Eclipse. Le développement sous Xcode nous a donc permis de développer plus rapidement puis de déboguer plus facilement TactoWeb.

Enfin, au cours du développement de TactoWeb, nous avons aussi essayé d'adapter TactoWeb pour qu'il puisse fonctionner sous Windows. Néanmoins, même si le code de TactoWeb et des

différentes librairies tactiles et sonores qui y sont attachées compile correctement, nous n'avons pas réussi à correctement lier ces librairies à TactoWeb sous Windows.

4.2 Architecture et fonctionnalités

4.2.1 Architecture

Le logiciel TactoWeb est lié à plusieurs autres librairies qui lui permettent de récupérer le code source des pages Web, de l'analyser, puis de les transformer en pages Web audio-tactiles. La Figure 4-6 présente l'architecture globale de TactoWeb.

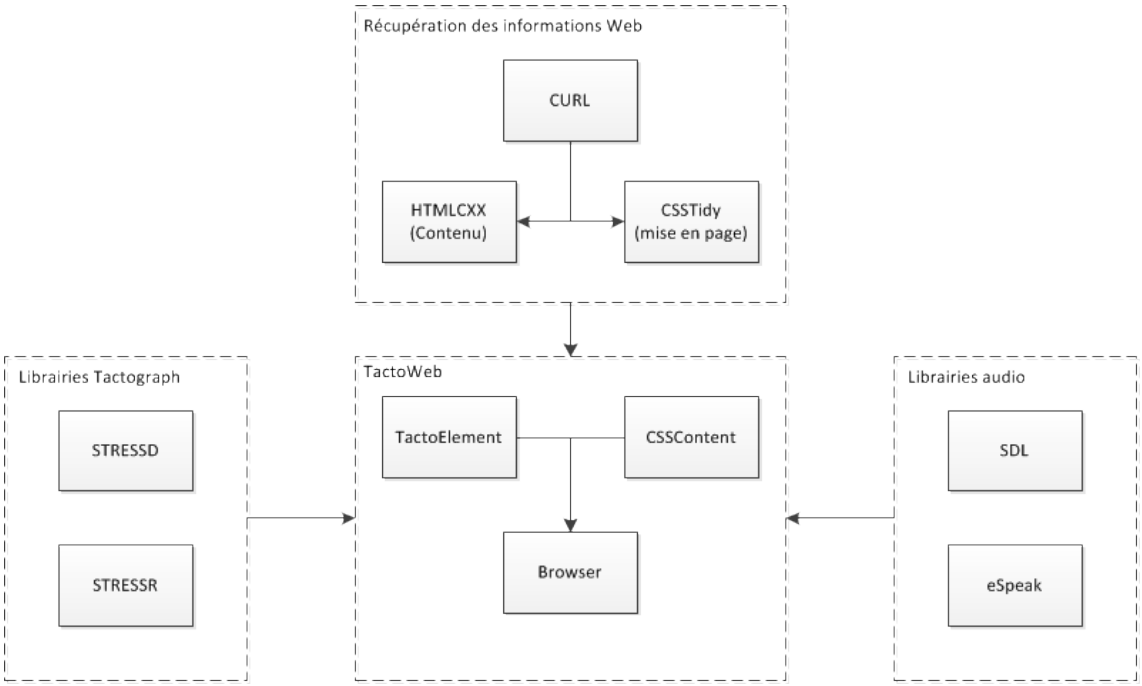


Figure 4-6 : Architecture de TactoWeb

En premier lieu, les codes sources HTML et CSS d'une page Web sont récupérés via la librairie cURL (Haxx, 2013). Le code HTML est traité par la librairie d'analyse syntaxique HTMLCXX (de Castro Reis & Braga Araújo, 2013). Le code CSS est traité par la librairie d'analyse syntaxique CSSTidy (Schmitz, 2013) que nous avons modifiée pour qu'elle s'intègre mieux à TactoWeb. Au cours des deux processus d'analyse syntaxique, des objets TactoElement et CSSContent sont créés. Les objets TactoElement correspondent aux différents types d'éléments Web composant la page. Ils décrivent donc des textes, des titres, des images, etc. On peut donc

considérer un TactoElement comme une représentation audio-tactile d'un élément Web dans TactoWeb. Chaque objet TactoElement est relié à un ou plusieurs objets CSSContent qui vont gérer son formatage et sa mise en page. Pour résumer, l'objet TactoElement se rapporte à une balise HTML et l'objet CSSContent se rapporte à une règle CSS.

Dans l'objet Browser, TactoWeb calcule ensuite la localisation et la taille de chaque TactoElement en s'aidant du ou des objets CSSContent qui lui sont associés. On attribue ensuite un type de rendu tactile à un type de TactoElement à l'aide des différents rendus tactiles définis dans la librairie STRESSR. Des éventuelles audicônes et textes à lire en synthèse vocale sont ajoutée aux TactoElement. On peut donc considérer l'objet Browser comme le rendu d'une page Web.

L'exploration de pages se fait ensuite grâce au Tactograph qui est relié à TactoWeb grâce à la librairie STRESSD. Les audicônes sont lues via la librairie SDL (QNX software developpers, 2013) et le contenu d'un objet TactoElement est lu par l'intermédiaire de la synthèse vocale eSpeak (Source Forge, 2013).

4.2.2 Récupération et traitement des données HTML et CSS

4.2.2.1 Récupération du code source

Pour créer le fureteur TactoWeb, la première chose est de récupérer les informations pertinentes d'une page Web. Pour un affichage audio-tactile, les informations pertinentes sont le contenu, que l'on retrouve aussi dans les lecteurs d'écran, mais aussi le formatage et la mise en page. Il faut donc récupérer les codes HTML et CSS d'une page Web à partir de son URL. Cette récupération s'effectue grâce à la librairie cURL.

Une fonction de la librairie cURL permet de stocker le texte pointé par une URL dans une variable de texte : HTMLCode_. Pour une URL pointant vers une page HTML, cURL permet de récupérer non pas le code source mais le code généré. En effet, si une page Web contient du code PHP, la fonction cURL va retourner le code HTML créé par le code PHP ce qui est un avantage. Par contre, si le code HTML contient une redirection automatique vers une page, cURL ne retourne pas la page vers laquelle pointe la redirection. On obtient un message d'erreur à la place. Une fois le code HTML récupéré, celui-ci est transformé en objet DOM grâce la librairie d'analyse syntaxique HTMLCXX. Cet objet DOM est un arbre contenant le contenu et les

paramètres de chaque balise HTML du code récupéré, tout en conservant la relation parent-enfant de ces balises. La librairie HTMLCXX permet normalement aussi de parcourir et d'analyser du code CSS. Mais cette méthode était loin d'être optimale et semblait oublier certaines informations cruciales du code CSS. Nous avons donc utilisé la librairie CSSTidy pour analyser syntaxiquement le code CSS.

Une première passe est faite dans l'objet DOM pour récupérer les URL des fichiers CSS. Ces URL se trouvent généralement dans les balises <link> de l'en-tête du fichier HTML. Chaque URL pointant vers un fichier CSS est ensuite traitée avec la librairie cURL pour stocker tout le code CSS dans une même variable de texte : CSSCode_. La première passe dans le DOM permet aussi de récupérer le code CSS brut qui pourrait être directement inséré dans le code HTML via la balise <style> dans l'en-tête (<head>) du fichier HTML. Ce code brut est ajouté à la variable CSSCode_. Néanmoins, d'autres fichiers CSS peuvent être liés à un fichier CSS avec le sélecteur CSS @import. Nous procédons à une première passe dans le code CSS stocké dans la variable CSSCode_ pour récupérer les URL de fichiers CSS indiquées par le sélecteur @import. Cette passe se fait grâce à une fonction que nous avons ajoutée dans la librairie d'analyseur syntaxique CSSTidy. En effet, cette librairie ne permettait pas de récupérer le contenu associé au sélecteur @import. Nous avons ajouté cette méthode directement dans CSSTidy plutôt que dans TactoWeb afin d'exploiter la structure des données et les fonctions principales de CSSTidy. Les URL pointant vers les nouveaux fichiers CSS sont traitées avec cURL afin d'ajouter le code supplémentaire CSS dans la variable CSSCode_ contenant le code précédemment traité.

4.2.2.2 Analyseur syntaxique CSSTidy et CSSContent

Une fois tout le code CSS stocké dans CSSCode_, il est analysé syntaxiquement à l'aide de la librairie CSSTidy. Nous avons ensuite développé et ajouté notre propre fonction dans cette librairie pour nous permettre d'exploiter les données analysées syntaxiquement par CSSTidy. Cette fonction nous permet de stocker toutes les données CSS dans un vecteur contenant des objets CSSContent.

Un objet de la classe CSSContent correspond à une règle CSS. Il contient donc un sélecteur qui permet d'identifier le ou les éléments HTML auxquels va s'appliquer la règle. Il contient aussi un ensemble de propriétés CSS qui nous servent à placer correctement et à calculer la taille des

éléments Web. Les propriétés CSS directement stockées dans un objet CSSContent ainsi que leurs valeurs supportées sont les suivantes :

- display : inline, block et none.
- position : toutes les valeurs sont supportées.
- float : toutes les valeurs sont supportées.
- clear : toutes les valeurs sont supportées.
- text-align : left et right.
- visibility : visible et hidden.
- font-size : toutes les valeurs sont supportées.
- background-image : toutes les valeurs sont supportées.

De plus, l'objet CSSContent contient aussi un objet Position où sont stockées toutes les propriétés CSS numériques permettant de calculer la taille de l'élément HTML auquel s'applique la règle CSS. Nous avons choisi de stocker ces propriétés CSS spécifiques dans un objet par souci de clarté lors de l'analyse et du débogage du code. Les propriétés CSS stockées dans l'objet Position sont les suivantes :

- width et height.
- min-width et min-height.
- margin : peu importe son nombre de valeurs, et toutes ses dérivées (ex. : margin-left).
- padding : peu importe son nombre de valeurs, et toutes ses dérivées (ex. : padding-left).
- top, left, bottom et right.

Pour toutes les propriétés de Position et pour font-size, les unités de valeurs supportées sont le point (pt), le pixel (px), le cadratin (em), le pourcentage (%) et, à l'exception de font-size qui ne le supporte pas, le centimètre (cm). La majorité de ces propriétés peuvent aussi avoir comme valeur « inherit », ce qui permet à un élément Web de prendre la valeur de son parent pour cette propriété. Même si TactoWeb récupère cette valeur, celle-ci n'est pas traitée au cours de l'interprétation des TactoElement.

4.2.2.3 Analyseur syntaxique HTMLCXX et TactoElement

Une fois le vecteur de CSSContent créé, TactoWeb traite les éléments Web contenus dans le code HTML récupéré. Ce traitement se fait à l'aide de la fonction DOMParser utilisant les objets

définis dans la librairie HTMLCXX. Cette fonction permet surtout de créer un vecteur d'objets TactoElement qui représentent des éléments Web. Chaque objet TactoElement pointe vers son parent TactoElement et ses enfants TactoElement pour conserver l'architecture originale du DOM dans TactoWeb. Ce respect de l'architecture permet de garder la cohérence d'affichage lors des calculs de la localisation et de la dimension de chaque TactoElement.

Certaines informations communes à tous les types de TactoElement sont récupérées directement dans cet objet. Les deux premières sont les attributs id et class de l'élément Web. Ces deux attributs sont essentiels pour faire l'association entre un objet TactoElement et les différentes règles CSS qui s'y rapportent. Une autre information commune, dépendant du type d'élément Web est stockée dans TactoElement dans la variable « flow_ » Il s'agit du type d'affichage « en ligne » (inline) ou « en bloc » (block) par défaut du type d'élément Web. Si un type d'élément Web doit s'afficher « en ligne », il va s'afficher à droite de l'élément précédent, si celui-ci est aussi un élément Web « en ligne ». Si un type d'élément Web doit s'afficher « en bloc», il va s'afficher en bas de l'élément qui le précède, le plus à gauche possible de la bordure de son élément parent.

En fait, l'objet TactoElement n'est jamais utilisé pour stocker un élément Web. D'autres types d'objets, héritant de TactoElement, ont été créés pour représenter les différents types d'éléments Web qui sont supportés par TactoWeb. Chacun de ses objets hérite des attributs communs décrits dans le paragraphe précédent. Ces objets ainsi que leurs attributs spécifiques sont définis et décrits dans l'Annexe 1.

Enfin, la fonction DOMParser récupère aussi le titre de la page (balise <title>) de la section <head> de la page Web mais ne le stocke pas dans un TactoElement. Cette information est directement stockée dans l'objet Browser pour être lue lors de l'ouverture de la page Web.

4.2.2.4 Association entre TactoElement et CSSContent

Lors de la création d'un TactoElement, un ou plusieurs CSSContent lui sont associés. Cette association se fait exactement de la même façon que l'association entre un élément Web et une règle CSS. Le vecteur d'objets CSSContent est parcouru entièrement. Le sélecteur de chaque CSSContent est comparé aux attributs « id » (texte précédé du caractère « # » dans un bloc du sélecteur), « class » (texte précédé du caractère « . » dans un bloc du sélecteur), au type de balise

(texte sans caractère spécial dans un bloc du sélecteur) et aux ancêtres du TactoElement (nombre de blocs du sélecteur). Un sélecteur n'ayant aucun ancêtre n'est constitué que d'un seul bloc.

Dans le cas où plusieurs CSSContent correspondent à un même TactoElement, ceux-ci sont fusionnés en suivant les règles de priorité des sélecteurs CSS. En effet, si deux règles CSS définissent le même attribut pour un élément Web, la priorité doit être donnée à l'une des deux règles.

Tout d'abord, si le sélecteur d'une des deux règles CSS contient plus de blocs que le sélecteur de l'autre règle, alors c'est ce premier qui l'emporte. Si les sélecteurs des deux règles contiennent le même nombre de bloc, on regarde le dernier bloc de chaque sélecteur et celui qui est le plus haut dans la liste de priorité suivante l'emporte :

1. balise#id.class
2. #id.class
3. balise #id
4. #id
5. balise.class
6. .class
7. tag

Par exemple, si on a un hyperlien avec une classe « menu » (<a class="menu">), les attributs de la règle CSS ayant comme sélecteur « a.menu » seront prioritaires sur ceux qui ont comme sélecteur « a » ou « .menu ».

Lorsqu'un objet TactoElement possède une valeur dans son attribut « style », cela veut dire que des attributs CSS sont directement définis dans l'élément Web (ex. : <a style=" background-color:yellow;">) . Dans ce cas, ces attributs ont la priorité absolue sur les mêmes attributs définis dans le(s) objets CSSContent associés au TactoElement.

Si aucun objet CSSContent créé à partir de la feuille de style ou code CSS imbriqué dans la page HTML ne peut être relié à un objet TactoElement, on attribue un CSSContent vide au TactoElement. Cet objet vide contient la valeur par défaut de chaque paramètre CSS stocké dans un CSSContent.

4.2.3 Affichage des TactoElement : dimensions et localisation

Une fois les objets TactoElement et CSSContent créés et l'association entre eux effectuée, le vecteur de TactoElement est parcouru pour les afficher dans TactoWeb. Ceux-ci sont rendus sur une surface de 1000 × 794 pixels dans TactoWeb. Tous les TactoElement sont traités à l'exception des éléments suivants :

- TactoIgnore car ce n'est pas du contenu (scripts et styles).
- TactoElement ayant l'attribut CSS « display » égal à « none » et n'ayant pas de « frère et sœur ». Si un élément Web avec une règle CSS « display : none; » possède des « frères et sœurs » c'est qu'il fait généralement partie d'un menu utilisant des images de fond comme éléments cliquables. Dans ce cas, le texte est caché avec une règle « display : none » par souci de design et d'accessibilité.
- TactoInput dont le paramètre « type » est égal à « hidden ». Il s'agit d'un champ supplémentaire dans le formulaire mais n'étant pas affiché.
- TactoDiv n'ayant aucun enfant et qui n'est associé à aucun CSSContent. Il s'agit généralement de divisions vides oubliées par le développeur.

L'algorithme permettant le calcul de la localisation et de la dimension des TactoElement est basé sur le calcul de la localisation et de la dimension des éléments Web dans un fureteur classique. Néanmoins, TactoWeb n'est pas aussi complet que ces navigateurs au niveau des attributs CSS et HTML jouant sur la localisation des éléments Web. Ce calcul est directement lié à la dimension de son TactoElement parent, ainsi qu'à la localisation et la dimension des TactoElement qui le précèdent et qui ont le même parent direct.

Même si la dimension et la localisation de tous les TactoElement sont calculées, seuls les TactoImage, TactoText et les TactoElement qui sont des composants de formulaires, sont affichés. En effet, ce sont seulement ces éléments qui correspondent au contenu d'une page Web. Si cela paraît logique pour les TactoImage et les composants de formulaire, cela reste moins évident pour les TactoText. En effet, pourquoi ne pas afficher aussi les TactoLink, TactoHeader et TactoParagraph par exemple? En fait, ses derniers contiennent tous un TactoText et servent de contexte au texte qui est affiché dans TactoWeb. Par exemple, un TactoText qui a comme parent un TactoLink puis un TactoHeader sera interprété comme un hyperlien à l'intérieur d'un titre. C'est à partir des ancêtres du TactoText, donc de son contexte, que TactoWeb décidera des

rendus tactiles et sonores à appliquer. L'élément TactoText ne représente que le dernier élément d'une branche du DOM et ne sert donc qu'à l'affichage dans TactoWeb. Prenons l'exemple suivant :

```
<li>
    <a href="index.html>
        Accueil
    </a>
</li>
```

Le texte « Accueil » correspond au TactoText et les balises qui l'entourent en donnent le contexte. Le TactoText est entouré d'un TactoLink, lui-même entouré d'un TactoListElement. Le texte « Accueil » est donc un hyperlien à l'intérieur d'une liste et seul le texte est pertinent à afficher dans TactoWeb, comme c'est le cas dans les autres fureteurs.

Pour faciliter la compréhension dans la suite de cette thèse, un TactoText ayant un contexte particulier dû à se(s) parents(s) sera indiqué par le nom de se(s) parent(s). Par exemple, un TactoText ayant comme parent un TactoLink sera appelé un TactoLink.

La dimension du TactoText est calculée en fonction de la taille de police qui lui est appliquée (ou celle de son parent s'il n'en a pas) et du nombre de caractères le composant. La largeur d'un caractère correspond environ aux deux tiers de sa hauteur. Cette hauteur équivaut à la taille de police quand celle-ci est affichée en point (pt). Sachant que toutes les informations de taille sont en pixel dans TactoWeb, il faut d'abord convertir la taille de police en point, c.-à-d. multiplier la valeur en pixels par trois quarts. Il faut ensuite multiplier par deux tiers pour obtenir la largeur d'un caractère et enfin multiplier par le nombre de caractères pour obtenir la largeur du texte en pixel :

$$\text{Largeur} = \left(\text{taille_police} \times \frac{3}{4} \times \frac{2}{3}\right) \times \text{nb_car}$$

$$\rightarrow \text{Largeur} = \frac{\text{taille_police} \times \text{nb_car}}{2}$$

Si la largeur du texte est supérieure à la largeur de son parent, le texte doit être placé sur plusieurs lignes. La hauteur totale du TactoText est égale au nombre de lignes nécessaires multiplié par la taille de la police en pixel, plus trois qui correspond à l'espace entre les lignes.

$$\text{Hauteur} = \text{nb_lignes} \times (\text{taille_police} + 3)$$

4.2.3.1 Relation par rapport au TactoElement parent

Lorsqu'on veut placer un TactoElement dans la page Web, le premier facteur à prendre en compte est qu'il doit être placé dans son TactoElement parent. Ce placement prend en compte les marges externes (marge) du TactoElement et les marges internes (padding) du parent qui sont spécifiées dans le ou les CSSContent qui lui sont associées. La Figure 4-7 montre le placement d'un TactoElement par rapport à son parent en prenant en compte les marges. Par souci de simplicité, la figure montre le cas où le parent n'a qu'un enfant direct.

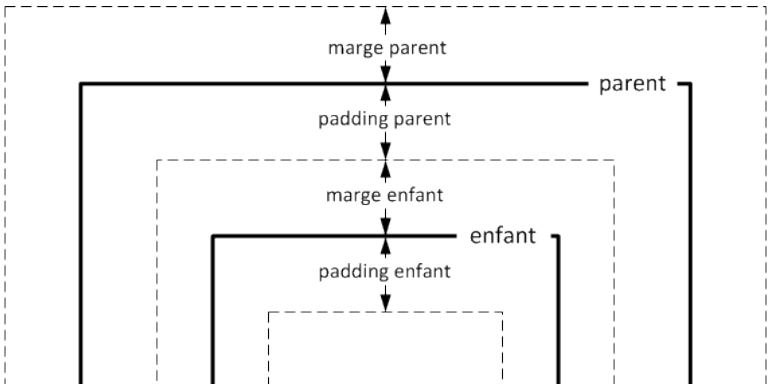


Figure 4-7 : Localisation d'un TactoElement par rapport à son parent dans le cas où le parent n'a qu'un enfant direct

Dans le cas où le parent a plusieurs enfants directs, ceux-ci se partagent l'espace indiqué par le rectangle « enfant » en prenant en compte le second facteur suivant : un TactoElement sera toujours placé après le TactoElement qui le précède dans le DOM et ayant le même parent direct. Dans ce cas, on regarde l'affichage par défaut du type de TactoElement, à savoir s'il est « en ligne » ou « en bloc » (voir 4.2.2.3).

Si un TactoElement a une largeur fixe spécifiée via un CSSContent, ou l'attribut HTML « width » dans le cas d'un tableau, le placement des enfants TactoElement directs doit s'adapter à cette largeur.

Comme on peut le voir dans la Figure 4-8, la largeur du parent ne s'est pas adaptée à celle des enfants. C'est l'inverse qui s'est passé. En effet, l'enfant 3 se retrouve en dessous des enfants 1 et 2 alors que son type d'affichage est « en ligne ». L'enfant 3 est trop large pour rentrer dans l'espace horizontal restant dans le parent, sur la même ligne que les enfants 1 et 2. L'enfant 4 se place correctement en dessous de l'enfant 3 même s'il y restait de la place à droite car son type d'affichage est « en bloc ».

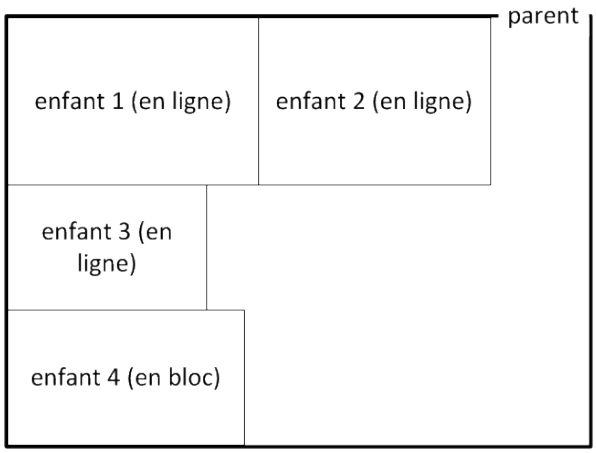


Figure 4-8 : Localisation des enfants directs d'un TactoElement ayant une largeur fixe

Si un TactoElement n'a pas de largeur fixe, sa largeur s'adapte en fonction de la largeur de ses enfants directs. Comme on peut le voir dans la Figure 4-9, l'enfant 3 s'est placé à droite de l'enfant 2 car c'est le comportement par défaut du type d'affichage « en ligne » et son parent n'a pas de restriction au niveau de sa largeur. C'est le parent qui a adapté sa largeur en fonction de la largeur maximale de ses enfants (dans ce cas, l'addition de la largeur des enfants 1, 2 et 3).

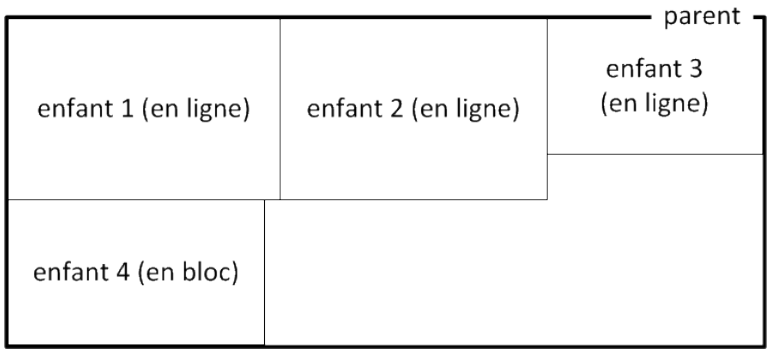


Figure 4-9 : Localisation des enfants directs d'un TactoElement n'ayant pas une largeur fixe

4.2.3.2 Adaptation de la dimension pour TactoWeb

Au cours du processus d'affichage, tous les paramètres et les attributs CSS de dimension et de localisation qui ont une unité de valeur (px, pt, em, cm, etc.) sont convertis en pixels pour s'adapter à l'affichage via le Tactograph. C'est exactement ce que font les fureteurs classiques pour adapter les sites Web à un affichage sur écran. Néanmoins la taille de certains TactoElement est augmentée pour TactoWeb. En effet, si un TactoElement est trop petit, il sera très difficile à trouver au cours de l'exploration spatiale. Deux fonctions d'agrandissement ont donc été implémentées dans TactoWeb.

Le premier agrandissement effectué concerne la largeur des TactoText dont un de ses parents est un TactoHeader, TactoListElement ou TactoLabel. Normalement, la largeur d'un TactoText est calculé en fonction du nombre de caractère qui le compose (voir 4.2.3). Dans le cas d'un texte étant en fait un titre, un élément de liste ou une étiquette de champ de formulaire, ce texte peut être très court. Ces trois éléments Web sont très importants dans l'exploration d'une page. Cela pourrait être moins important pour les listes mais celles-ci sont souvent utilisées pour mettre en page des menus. Les rendre difficiles d'accès à cause de leur petite largeur aurait pu nuire à l'utilisation de TactoWeb.

Comme on peut l'observer dans la Figure 4-10, si un utilisateur explore la liste verticalement en se plaçant sur la droite de cette liste, il pourrait manquer le TactoText « Culture » qui est à l'intérieur d'un TactoLink, lui-même à l'intérieur d'un TactoListElement (b). On évite ce genre d'erreur en maximisant la largeur du TactoText, c.-à-d. en lui faisant prendre tout l'espace horizontal que prend son parent, au lieu de le limiter à la taille correspondant à son nombre de caractères. Ce même traitement est appliqué aux titres et aux étiquettes de champs de formulaire.

(a) (b) (c)

Figure 4-10 : Menu présenté sous forme de liste (a), son adaptation sans adaptation de largeur dans TactoWeb (b) et avec adaptation de largeur dans TactoWeb (c)

Un autre agrandissement est effectué sur certains TactoInput : ceux correspondant aux cases à cocher, aux boutons radio et aux boutons de soumission dans les formulaires. Ceux-ci sont visuellement petits. Les cases à cocher et les boutons radio ont une taille de 12 × 12 pixels et les boutons de soumission une hauteur de 18 pixels par défaut. Ces champs de formulaires sont faciles à identifier visuellement mais bien plus difficiles tactilement à cause de leur petite taille. Pour palier à ce problème, TactoWeb fixe à 30 × 30 pixels la taille des boutons radio et des boîtes à cocher. TactoWeb ajoute aussi 25 pixels à la largeur et 10 pixels à la hauteur par défaut des boutons de soumission et d'annulation dans les formulaires.

4.2.3.3 Propriétés CSS spécifiques à la localisation des éléments Web

Certaines propriétés CSS annulent le placement naturel des éléments Web. La majorité d'entre elles ont été retranscrites dans TactoWeb au niveau du calcul de la localisation d'un TactoElement si ces propriétés sont dans un CSSContent qui lui est associé.

La première de ces propriétés est « display » qui peut prendre comme valeur « block », « inline » ou « none » dans TactoWeb. Le traitement de la valeur « none » a déjà été expliqué dans la section 4.2.3. La valeur « block » permet de transformer le type d'affichage par défaut d'un TactoElement de « en ligne » à « en bloc ». C'est l'inverse pour la valeur « inline ». La valeur « inline » est souvent utilisée pour présenter des listes de manière horizontale car l'affichage par défaut d'un TactoListElement est « en bloc » pour afficher par exemple un menu horizontal.

La seconde propriété est « float » qui peut prendre comme valeur « left », « right » et « none » dans TactoWeb. La valeur « none » correspond au type d'affichage par défaut du TactoElement, qu'il soit « en bloc » ou « en ligne ». Mais si la valeur « left » est attribuée, le TactoElement va se placer le plus à gauche de son parent, après les « frères et sœurs » qui le précèdent. Les « frères et sœurs » qui le suivent vont « flotter » autour du TactoElement en se plaçant directement à sa droite, que leur type d'affichage par défaut soit « en ligne » ou « en bloc ». C'est le même genre de comportement pour la valeur « right » sauf que cette fois, le TactoElement va se mettre le plus à droite possible par rapport à son parent et ses « frères et sœurs » qui le suivent vont toujours « flotter » autour. La Figure 4-11 illustre le comportement de la propriété « float ». Celle-ci est appliquée à l'image montrant une fraise et on peut voir un paragraphe « flotter » autour d'elle en fonction de la valeur de la propriété.

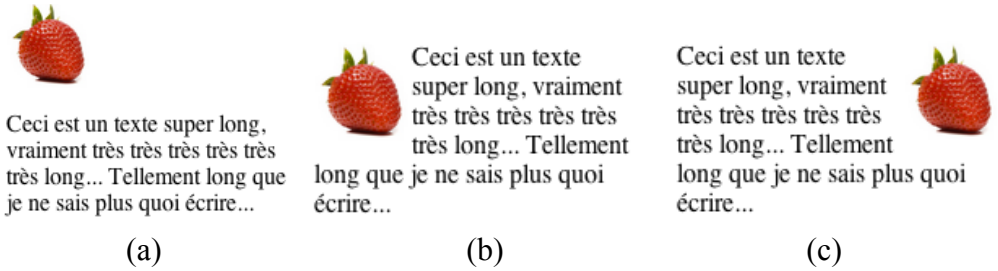


Figure 4-11 : Fonctionnement de la propriété CSS « float » en fonction de sa valeur : (a) « none », (b) « left » et (c) « right ».

Une autre propriété CSS prise en compte par TactoWeb pour calculer la localisation est la propriété « position » qui peut prendre comme valeur « static » (affichage par défaut), « absolute », « fixed » et « relative ». La valeur « static » ne change rien dans le calcul de la localisation du TactoElement. La valeur « relative » ne change pas la localisation du TactoElement mais prend en compte la valeur des attributs CSS « top », « bottom », « left » et « right ». La valeur « fixed » place le TactoElement le plus en haut à gauche de la fenêtre du fureteur puis prend en compte la valeur des attributs CSS « top », « bottom », « left » et « right ». Enfin, la valeur « absolute » place le TactoElement le plus en haut à gauche du premier parent qui n'a pas la valeur « static » pour son attribut « position », puis prend en compte la valeur des attributs CSS « top », « bottom », « left » et « right ».

La dernière propriété CSS qui peut influencer la localisation d'un TactoElement est « text-align ». La valeur « left » est le comportement par défaut et ne change rien dans le calcul de base de la localisation. La valeur « right » aligne un TactoText à droite de la largeur maximale de son parent. La valeur « center » n'est pas gérée dans TactoWeb et est interprétée comme la valeur « left ».

4.2.4 Retour tactile

Le retour tactile tourne à une fréquence de 750 Hz. Cela nous permet d'avoir un retour tactile proche du temps réel qui ne dépasse pas le seuil de perception tactile de nos doigts (1000 Hz).

4.2.4.1 Création des masques graphiques

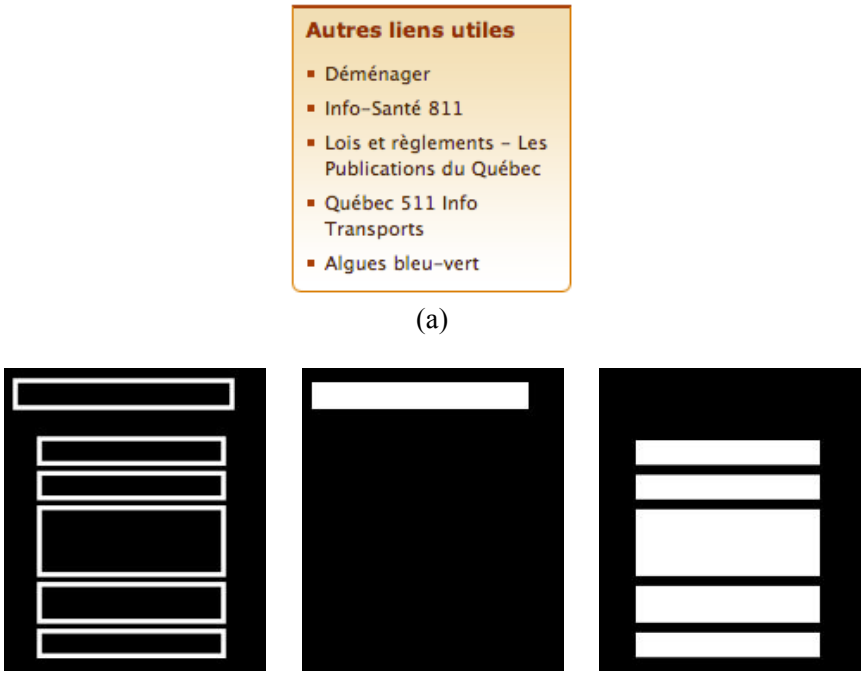
À l'ouverture d'une page Web dans TactoWeb, un masque graphique est créé pour chaque type de TactoElement, ainsi qu'un masque supplémentaire pour les bordures de l'ensemble des TactoElement. Au cours de la création des différents types de TactoElement, ces masques graphiques sont modifiés.

Un masque graphique est composé d'un fond noir et d'une autre couleur correspondant à un niveau de gris. Le fond noir correspond aux endroits du masque où aucun rendu tactile ne sera appliqué. Les endroits où la couleur de niveau de gris est appliquée correspondent aux endroits où est appliqué un rendu tactile. Plus le niveau de gris est proche de la couleur blanche, plus la puissance du rendu tactile est forte. Cette puissance correspond à la fréquence de mouvement des

lamelles de la cellule tactile du Tactograph. Pour TactoWeb, nous souhaitons une puissance maximale pour chaque rendu tactile, donc le blanc est utilisé.

Une fois la localisation et la dimension d'un TactoElement calculées, deux masques graphiques sont modifiés en ajoutant du blanc à des endroits spécifiques. Le premier masque graphique modifié est celui des bordures. Les bordures de chaque TactoElement y sont ajoutées. Ce masque est donc composé de multiples rectangles blancs vides. Le second masque modifié est celui de la texture associé au type de TactoElement. Il correspond en fait à l'intérieur du TactoElement calculé à partir de sa bordure. Ce masque contient donc de multiples rectangles blancs pleins.

La Figure 4-12 présente un extrait de page Web (a) contenant un titre suivi d'une liste d'hyperliens. Trois masques sont modifiés lors de la création des TactoElement correspondant à cet extrait de page Web. Le premier (b) correspond aux bordures de chaque TactoElement, le second (c) correspond à la texture des titres (TactoHeader) et le dernier (d) correspond à la texture des hyperliens (TactoLink).



(a)

(b) (c) (d)

Figure 4-12 : Modification des masques graphiques à partir d'un extrait de page Web.

4.2.4.2 Librairie de rendu tactile

Un type de rendu tactile est ensuite associé à chaque masque graphique. Pour ce faire, nous utilisons la librairie tactile STRESSR développée dans la thèse de doctorat de Vincent Lévesque (Lévesque, 2009) et employée dans les précédentes recherches utilisant le Tactograph ou une de ses précédentes versions (Lévesque & Hayward, 2008; Petit, Dufresne, Lévesque, Hayward, et al., 2008). Cette librairie permet notamment de générer des textures tactiles. Les types de textures tactiles disponibles dans STRESSR sont les suivantes :

- Vibrations : ce rendu tactile correspond à une oscillation rapide et constante d'une ou plusieurs lamelles de la cellule tactile. Les lamelles continuent d'osciller même lorsque le module tactile du Tactograph est immobile sur la surface d'exploration. Les vibrations sont uniformes et ne peuvent pas avoir d'orientation. C'est le type de rendu tactile le plus facilement perceptible et identifiable.
- Ondulations : ce rendu tactile correspond à un mouvement latéral unique d'une ou de plusieurs lamelles de la cellule tactile. La position des lamelles est par contre statique lorsque le module tactile est immobile sur la surface d'exploration. Les ondulations servent surtout à indiquer des textures. Lorsque l'utilisateur passe au-dessus d'une ondulation, il sent une sorte de suite de vagues sous son doigt. Il est possible de changer l'orientation (en degré) d'une ondulation et l'espacement (en millimètre) entre chaque « vague » de l'ondulation. Il est aussi possible de générer des ondulations radiales et des ondulations dynamiques (l'ondulation suit une direction même lorsque le module tactile est immobile).
- Points : ce rendu tactile correspond à la simulation d'une grille de points. On peut préciser l'espacement vertical et horizontal des lignes de points. Il est conseillé d'être en mouvement avec le module tactile pour correctement sentir les points.

4.2.4.3 Grammaire tactile utilisée

Le rendu tactile à appliquer dépend du type de TactoElement. En effet, un TactoLink n'aura pas le même retour tactile qu'un simple TactoText. Néanmoins, il n'est pas forcément pertinent d'utiliser un rendu tactile différent pour chaque type de TactoElement différent. Tout d'abord, l'utilisation d'audicônes permet aussi de distinguer les types de TactoElement, et distinguer ces audicônes reste très facile si elles sont choisies de manière pertinente. De plus, les précédents

travaux avec le Tactograph et ses versions antérieures (Petit, Dufresne, Lévesque, Hayward, et al., 2008) ont démontré qu'il n'était pas nécessaire d'utiliser un grand nombre de textures différentes pour une image tactile, c.-à-d. une page Web dans notre cas.

Ces travaux ont montré que les vibrations étaient les plus pertinentes pour reconnaître une forme en les appliquant seulement à sa bordure car il est plus facile d'en faire le tour (Lévesque & Hayward, 2008). Ces travaux nous ont aussi montré que pour discerner différentes ondulations, il fallait que l'espacement des « vagues » soit d'au moins 0,5 mm et qu'il y ait une différence d'orientation d'au moins 45° (Lévesque & Hayward, 2008). Cela nous donne accès à quatre orientations : 0°, 45°, 90° et 135°. Le type de rendu par point ne s'est pas vraiment montré utile, que cela soit pour identifier un type de texture ou le contour d'une forme. Il en est de même pour les ondulations radiales. Enfin, il est plus facile d'identifier une forme ayant un contour vibrant si on ajoute une texture d'ondulation à l'intérieur de la forme (Petit, 2007).

Nous avons donc utilisé les vibrations pour les bordures de chaque TactoElement. La texture appliquée à l'intérieur de TactoElement dépend ensuite de son type. Tous les textes (TactoHeader, TactoListElement, TactoLabel, etc.) ont une ondulation avec une orientation de 135°, ainsi qu'un espacement de 2 mm entre les « vagues », à l'exception des TactoLink qui une orientation de 90° et un espacement de 2 mm. Même si chaque type de TactoElement se rapportant à du texte est identifié par une audicône différente (voir 4.2.5.1), nous avons souhaité distinguer la texture des hyperliens des autres types de texte, à cause de leur importance au cours de la navigation dans un site Web. De plus, nous avons utilisé un espace relativement étroit pour les « vagues » dans les textes car leur taille dans les pages Web est très souvent petite. Une ondulation de 45° avec un espacement de 6 mm est appliquée à l'intérieur des TactoImage. Nous avons utilisé un espacement plus large que celui des textes car les images sont souvent assez larges pour qu'on puisse distinguer un plus grand espacement dans l'ondulation. Enfin, aucune texture n'a été utilisée à l'intérieur des différents types de champs de formulaire. Le fait qu'il y ait un nombre élevé de différents types de champs de formulaire nous a dissuadés d'y ajouter une texture. La Figure 4-13 résume la création des rendus tactiles (d) à partir des masques modifiés (a, b et c) dans l'exemple déjà montré dans la Figure 4-12. Chacun de ces masques est associé à un rendu tactile différent.

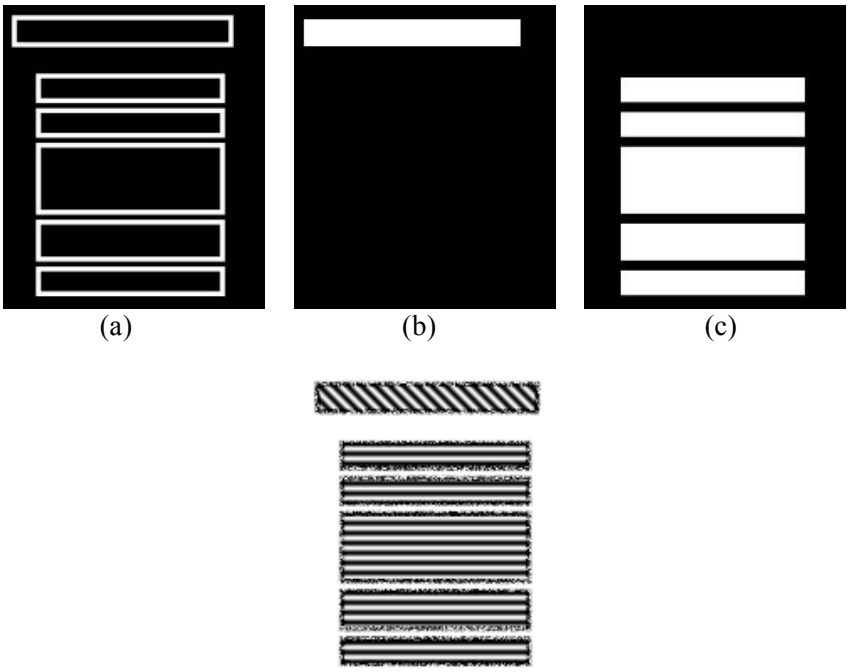


Figure 4-13 : Création des rendus tactiles (d) à partir des masques graphiques (a, b et c).

Une fois tous les TactoElement traités, on obtient le rendu tactile de la page Web entière. Comme il a déjà été montré dans certaines figures précédentes, un rendu graphique est aussi généré pour contrôler la bonne interprétation de la page Web par TactoWeb. Cela permet aussi d'avoir un retour visuel lorsqu'on souhaite observer un utilisateur non-voyant utiliser TactoWeb. Dans ce rendu graphique, les vibrations (pour les contours) sont représentées par un bruit noir et blanc. Les ondulations (texture à l'intérieur des TactoElement) sont représentées par un ensemble de bâtons noirs sur un fond blanc. L'espacement et l'orientation de ces bâtons représentent l'espacement et l'orientation de l'ondulation. La Figure 4-14 représente la page Web originale du programme d'étude en science de la nature du Collège André-Grasset de Montréal[1] et la Figure 4-15 sa transcription tactile complète.

[1] http://www.grasset.qc.ca/Programmes-d-etudes/sciences-de-la-nature-276.php

Figure 4-14 : Page Web originale du programme d'étude en science de la nature du Collège André-Grasset de Montréal

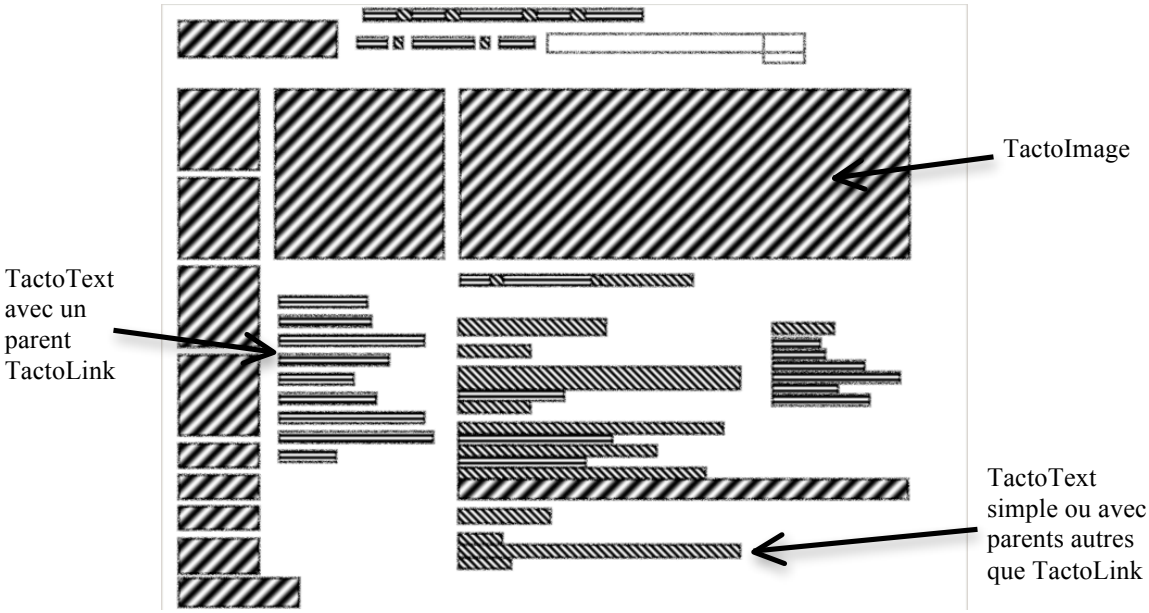


Figure 4-15 : Représentation de la page Web du programme d'étude en science de la nature du Collège André-Grasset de Montréal dans TactoWeb

4.2.5 Retour sonore

L'association d'un retour sonore à un TactoElement se fait juste après l'association avec le rendu tactile. Le retour sonore peut contenir des audicônes et du texte qui sera lu en synthèse vocale. Ce retour sonore se déclenche lorsque le centre de la cellule tactile de la souris rentre dans un TactoElement. La façon de déclencher des retours sonores avec le Tactograph fonctionne de la même façon que déclencher un retour tactile, c.-à-d. avec des masques graphiques. Le fond noir correspond à une absence de retour sonore. Le retour sonore se déclenche donc quand l'utilisateur entre dans la partie du masque qui n'est pas noire.

En plus de présenter le contenu textuel d'un TactoElement, le retour sonore donne toujours la fonction du TactoElement, que cela soit par l'intermédiaire d'une audicône ou d'un texte supplémentaire lu avec la synthèse vocale. Les seuls TactoElement qui n'ont pas de retour sonore supplémentaire pour en indiquer la fonction sont les TactoText et les TactoParagraph car ils n'ont pas de fonction particulière à part présenter du contenu textuel, ainsi que les TactoImage car ils ont une texture tactile qui leur est propre.

4.2.5.1 Audicônes et SDL

Les audicônes sont utilisées pour indiquer les types de TactoElement. Le seul TactoElement qui n'est pas un type de champ de formulaire et qui n'a pas d'audicône est le simple TactoText, c.-à-d. celui qui n'a pas comme parent un TactoLink, un TactoHeader, un TactoLabel ou un TactoListElement. Ce TactoText n'a pas de contexte, donc seul son texte est lu en synthèse vocale. Les TactoElement se rapportant à des types de champs de formulaire n'ont pas d'audicône non plus car ils sont trop nombreux. Nous avons limité le nombre d'audicônes différentes pour réduire la charge cognitive au cours de l'utilisation de TactoWeb. En effet, notre mémoire à court terme a une capacité de sept, plus ou moins deux éléments (Miller, 1956). Cette limite ne serait pas présente si l'utilisateur avait accès à la correspondance des audicônes quand il le souhaite mais ce n'est pas le cas. De plus, l'utilisateur doit traiter les informations tactiles et les informations sonores données par la synthèse vocale en plus des audicônes, ce qui diminue l'attention auditive (Occelli, Spence, & Zampini, 2013). Et même si les personnes non-voyantes de naissance ont une meilleure mémoire auditive que les personnes voyantes, ce n'est pas le cas pour les personnes non-voyantes tardives (Röder & Rösler, 2003).

Dufresne & al. ont mis en relief plusieurs principes permettant de choisir une audicône pertinente par rapport à l'objet qu'elle doit symboliser (Dufresne, Martial, Ramstein, & Mabilleau, 1996). Tout comme une icône graphique, une audicône doit être une sorte de métaphore de l'objet qu'elle symbolise. Elle ne doit pas non plus créer d'interférences avec les autres modalités du système. Nous avons essayé de suivre ces principes pour choisir nos audicônes. Néanmoins, il est complexe de trouver des audicônes vraiment significatives à associer à des éléments Web comme des éléments de liste ou des étiquettes par exemple. Mais il est possible d'en trouver des assez significatifs pour les hyperliens ou les titres.

L'audicône associée à un TactoLink est un son représentant un clic léger car les TactoLink sont les seuls éléments cliquables dans TactoWeb. Lorsqu'un utilisateur appuie sur le bouton gauche lorsqu'il est au-dessus d'un TactoLink (voir 4.2.6.1), une autre audicône représentant le son d'un véritable clic (plus lourd que le premier) se déclenche pour indiquer que le système a correctement interprété l'action de l'utilisateur comme un clic. Les autres audicônes qui peuvent paraître significatives sont celles associées aux TactoHeader, soit les titres. Ces audicônes sont des notes musicales et plus le niveau du titre descend, plus la note musicale est basse. Les utilisateurs font donc l'association entre les notes musicales et les titres, et la hauteur de la note lui donne une indication sur le niveau du titre qu'il est en train d'explorer.

Pour les TactoLabel et les TactoListElement, nous n'avons pas trouvé d'audicônes significatives. L'audicône associée au TactoLabel ressemble au bruit d'une balle de pingpong rebondissant une fois sur une table. L'audicône associée au TactoListElement est un bruit métallique. Même si les audicônes ne sont pas toutes significatives, il était important qu'elles soient toutes facilement distinctives, donc courtes et différentes les unes des autres.

Les audicônes sont intégrées via la librairie SDL dans TactoWeb. Cette librairie, libre de droit et multiplateforme, permet de lire des fichiers sonores au format WAVE. Nous avons implémenté cette librairie directement à l'intérieur de la librairie STRESSR car, comme nous l'avons précisé précédemment, le retour sonore fonctionne avec le même genre de masque graphique qui est utilisé pour le retour tactile.

4.2.5.2 Synthèse vocale et eSpeak

Après que l'éventuelle audicône d'un TactoElement ait été jouée, le texte qui lui est associé est lu via une synthèse vocale, que cela soit un titre, un hyperlien, une étiquette, un paragraphe, le texte

de description d'une image, etc. Des informations supplémentaires peuvent aussi être ajoutées dans le texte qui sera lu. En effet, on ajoute à tous les TactoElement se rapportant à des champs de formulaire, un texte décrivant l'action à faire pour remplir le champ. Par exemple, pour un champ de saisie de texte, le texte « Entrez du texte » suivi de l'éventuelle étiquette du champ sera lu. On indique donc l'action à faire avant de donner l'objet de cette action.

La synthèse vocale est aussi utilisée pour indiquer d'autres actions comme le démarrage de TactoWeb et l'ouverture d'une page en indiquant son titre, que cela soit via la barre d'adresse de TactoWeb ou en cliquant sur un hyperlien.

La synthèse vocale que nous utilisons dans TactoWeb est eSpeak. Son principal avantage est qu'elle est libre de droit et qu'elle supporte plusieurs langues. Nous avons d'ailleurs implémenté la synthèse vocale en français et en anglais dans TactoWeb. Néanmoins, la qualité de eSpeak reste inférieure à des synthèses vocales payantes comme celles utilisées dans la majorité des lecteurs d'écran.

4.2.6 Autres fonctionnalités

4.2.6.1 Boutons

TactoWeb reconnaît les signaux envoyés par le Tactograph lorsque les boutons situés à la droite et à la gauche de la cellule tactile sont pressés. Ces deux boutons sont considérés comme des clics de souris dans TactoWeb. Les clics courts, les clics longs et les doubles clics sont gérés dans TactoWeb, pour les deux boutons. La frontière entre un clic court et un clic long est d'une seconde. Il doit y avoir un maximum d'une demi-seconde entre les deux clics pour que cela soit considéré comme un double clic.

Même si TactoWeb reconnaît les trois types de clic pour les deux boutons, seuls les simples clics courts génèrent une action dans TactoWeb pour le moment. Ceci fut décidé dans le but de ne pas compliquer l'interaction avec TactoWeb par une mémorisation de trois actions différentes par bouton.

Le simple clic gauche correspond au clic principal de la souris lors de l'exploration d'une page. Il ne fonctionne donc que sur les TactoLink. Lorsqu'une pression sur le bouton gauche est reconnue comme un simple clic, une audicône ressemblant au son d'un clic se déclenche. Si l'audicône ne

se déclenche pas, cela veut dire que TactoWeb n'a pas reconnu la pression comme un simple clic court. Le texte « Ouverture de la page », suivi du nom de la page est ensuite lu par le synthétiseur vocal. Si ce texte n'est pas lu et que l'utilisateur entend tout de même l'audicône, cela veut dire qu'il n'a pas cliqué sur un TactoLink. Comme pour le déclenchement du retour sonore, c'est le centre de la cellule tactile qui fait référence pour savoir si l'utilisateur est dans un TactoLink ou non.

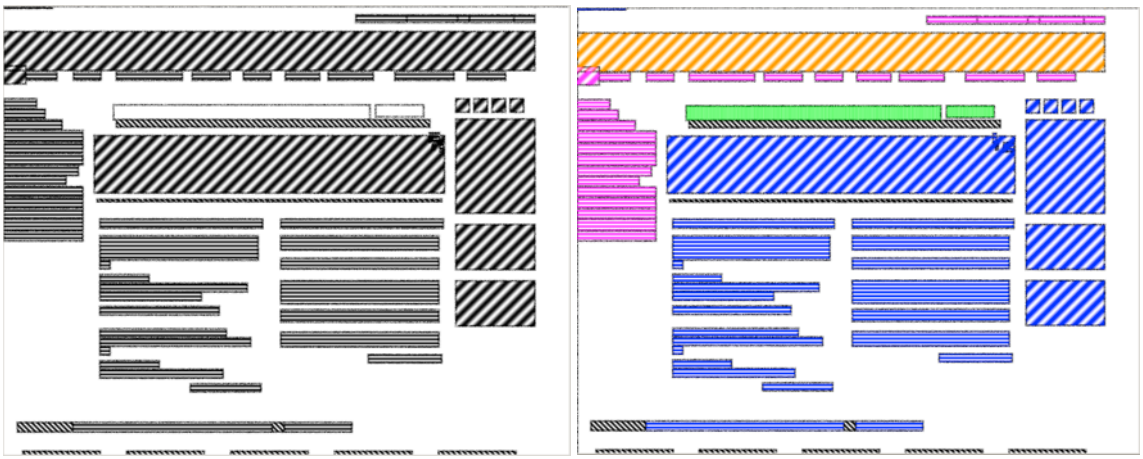
Le simple clic droit fonctionne exactement de la même façon que le clic gauche sur un hyperlien (audicône et texte lu), sauf qu'il fait office de bouton « précédent ». L'utilisateur peut donc retourner à la page précédente de son historique en pressant sur le bouton droit, peu importe où il se trouve dans la page.

4.2.6.2 TactoColor

Une version alternative fut développée dans le cadre de la Maîtrise en communication d'Aurélie Jonquet. Cette version s'adresse aux personnes malvoyantes. Comparées aux personnes fonctionnellement non-voyantes, les personnes malvoyantes ont un résidu visuel mais ont tout de même besoin d'outils d'adaptation pour naviguer sur Internet. TactoColor produit exactement les mêmes rendus tactiles et sonores que la version « classique » de TactoWeb. La différence avec TactoWeb est que les textures appliquées aux TactoElement sont colorées dans le rendu graphique. Ainsi, les hyperliens et images-liens sont en bleu, les images sont en orange, les champs de formulaires sont en vert et les menus sont en rose. Les autres types d'élément Web restent noirs ou vides. Un hyperlien est identifié comme faisant partie d'un menu si un de ses parents contient le mot « menu » dans ses propriétés « id » et/ou « class ». Cette façon d'identifier les menus a été effectuée pour mieux regrouper visuellement les hyperliens appartenant à un menu. La Figure 4-16 illustre les différences entre TactoColor et TactoWeb pour le site de la Bibliothèque et Archives Nationales du Québec[2].

---

[2] http://www.banq.qc.ca/accueil/



(a) (b)

Figure 4-16 : Différences entre TactoWeb (a) et TactoColor (b)

4.3 Tutoriel

Sachant que la grande majorité des utilisateurs de TactoWeb n'ont jamais exploré spatialement un site Web, il est impossible de leur faire utiliser cet outil sans aucun entrainement. Nous avons donc créé un tutoriel permettant de se familiariser avec l'exploration spatiale et les différentes fonctionnalités de TactoWeb. Le principal défi de ce tutoriel était qu'il devait être strictement textuel. En effet, la majorité des tutoriels existants sont très visuels, soit par l'intermédiaire de vidéos, soit par l'intermédiaire de succession d'images avec des explications textuelles.

Il existe trois types d'interactions pour transmettre un savoir à une personne : l'interaction tuteur(s)-apprenant(s), l'interaction apprenant(s)-apprenant(s), et l'interaction apprenant(s)-contenu (Moore, 1989). L'interaction tuteur(s)-apprenant(s) est perçue comme la plus appréciée et la plus malléable par les éducateurs selon Moore. En effet, même si le contenu à apprendre reste le même, le tuteur peut l'adapter en fonction de l'apprenant en insistant sur certains points ou en passant rapidement sur d'autres. De plus, ce type d'interaction est plus motivant pour l'apprenant et donne une meilleure expérience d'apprentissage (Garrison, 1990; Sabry & Baldwin, 2003). Cette méthode nous paraîssait aussi la plus adaptée à des explications strictement textuelles. Nous nous sommes donc basés sur cette approche pour créer notre tutoriel.

4.3.1 Objectifs d'apprentissage

Pour créer un tutoriel, il faut tout d'abord définir les objectifs d'apprentissage, puis créer les différentes activités qui permettront d'atteindre ces objectifs (Fernandez & Rubens, 2005). Les objectifs d'apprentissage correspondent à ce que l'apprenant doit être capable de faire une fois le tutoriel parcouru. Ces objectifs doivent être très spécifiques (Fernandez & Rubens, 2005) car définir des objectifs trop vagues complique la création des activités permettant de les atteindre. Afin de définir nos objectifs d'apprentissage, nous avons suivi les lignes directrices proposées par l'Université Park, située dans le Missouri aux États-Unis (Park University, 2013). Parmi ces lignes, on retrouve le fait que les objectifs doivent être centrés sur l'apprenant et qu'ils doivent être facilement observables.

Nous avons défini six objectifs d'apprentissage ainsi que les différents points et observations à effectuer pour remplir ces objectifs :

- L'apprenant, ici l'utilisateur de TactoWeb, doit se familiariser avec le design de l'appareil.
    - Le tuteur laisse l'apprenant toucher librement la coquille du module tactile.
    - Le tuteur laisse l'apprenant parcourir librement la surface d'exploration pour se faire une idée de sa taille.
- L'apprenant doit acquérir la stratégie de navigation spatiale.
    - Le tuteur vérifie si l'apprenant se déplace correctement de manière verticale et horizontale.
    - Le tuteur explique les sens logiques d'exploration d'une page Web : de haut en bas ou gauche à droite, jamais en diagonale.
- L'apprenant doit comprendre la fonctionnalité des boutons du Tactograph.
- L'apprenant doit identifier les différents types d'éléments Web.
    - La page de tutoriel doit contenir les différents éléments Web qu'on retrouve généralement dans une page (menu, contenu central, hyperliens, listes, titres, images, etc.).
    - Le tuteur s'assure que l'apprenant comprend la signification de chaque type d'élément Web.

  - Le tuteur explique à l'apprenant quels retours tactiles et sonores sont utilisés pour chaque élément.
- L'apprenant doit interpréter les différents types de champs d'un formulaire.
  - La page de tutoriel doit contenir tous les types de champs de formulaire gérés par TactoWeb.
  - Le tuteur doit s'assurer que l'apprenant comprend la signification de chaque type de champ de formulaire.
- L'apprenant doit comprendre comment explorer un formulaire.
  - Le tuteur explique l'organisation spatiale d'un formulaire.
  - Le tuteur explique clairement la relation entre une étiquette et un champ de formulaire.

4.3.2 Activités

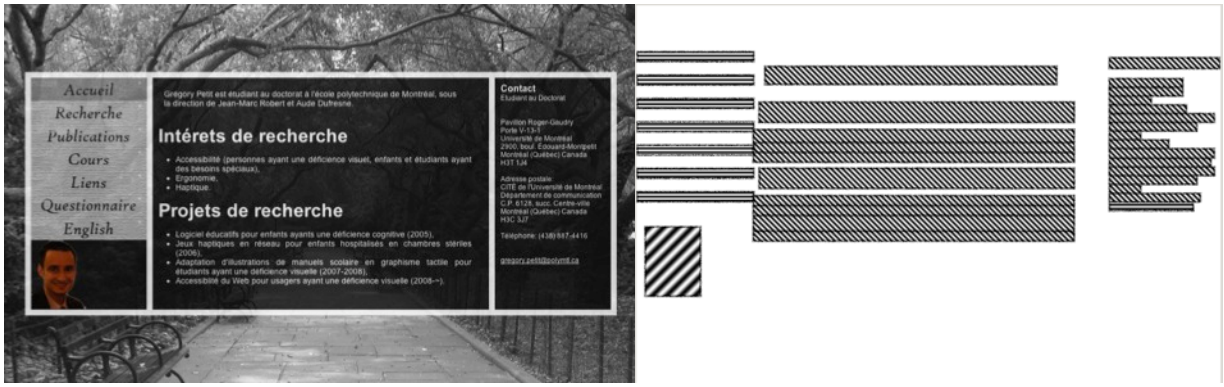
Quatre activités principales ont été créées à partir des objectifs d'apprentissage. Tout au long des activités, le tuteur précise à l'apprenant qu'il ne faut pas aller trop vite et qu'il doit strictement suivre les étapes qui lui sont présentées. La « docilité » de l'apprenant est essentielle pour que le tutoriel se déroule correctement car celui-ci pourrait être amené à découvrir TactoWeb sans suivre les instructions du tutoriel et manquer, ou ne pas comprendre correctement, des concepts importants de l'outil.

La première activité correspond à l'apprentissage de la manipulation du Tactograph et à l'introduction de ce qu'est TactoWeb. L'apprenant prend connaissance de l'outil qu'il devra manipuler pour utiliser TactoWeb. Il se familiarise avec la taille de l'appareil, la localisation de la cellule tactile et des boutons, ainsi qu'avec la taille de la surface d'exploration. Le tuteur présente aussi rapidement le concept de TactoWeb et ses différences majeures avec les lecteurs d'écran.

Le principe d'exploration spatiale est aussi expliqué au cours de cette activité. Le tuteur montre différentes techniques d'exploration à l'apprenant comme la navigation de haut en bas et de gauche à droite, le fait de suivre les bordures vibrantes pour suivre un élément Web et se déplacer tout droit, et le fait de sortir puis d'entrer à nouveau dans un élément Web pour réécouter le retour sonore. Ces techniques d'exploration ne sont que des suggestions et l'apprenant est libre d'en utiliser d'autres qu'il aurait trouvées par lui-même.

La seconde activité se rapporte à l'apprentissage des éléments Web et des stratégies d'exploration dans TactoWeb. Pour cela, une page Web de tutoriel est présentée à l'apprenant (Figure 4-17). Cette page correspond à la page d'accueil d'un site personnel[3], légèrement adaptée pour le tutoriel, afin de présenter les différents types d'éléments Web gérés par TactoWeb. Cette page contient des hyperliens, des titres, des listes, une image et du texte classique. Il est d'abord demandé à l'apprenant de descendre tranquillement depuis le coin en haut à gauche de la page jusqu'à ce qu'il rencontre la première sensation tactile et le premier retour sonore.

Les concepts de retour tactile et de retour sonore lui sont expliqués puis le tuteur le laisse explorer librement le reste de la page. Quand il rencontre un nouveau type d'élément Web, celui-ci lui est expliqué, ainsi que l'éventuelle audicône, qui est répété si besoin est. Le fonctionnement des boutons est aussi expliqué au cours de cette activité.



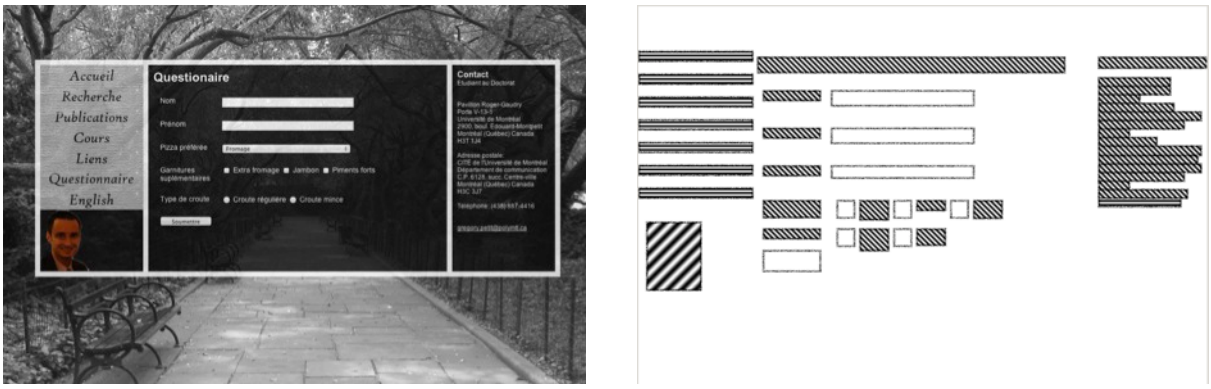
(a) (b)

Figure 4-17 : Page Web du tutoriel (a) et son affichage dans TactoWeb (b)

La troisième activité concerne l'apprentissage des formulaires dans TactoWeb (Figure 4-18). L'apprenant y accède de lui-même en cliquant sur l'hyperlien « Questionnaire » de la page du tutoriel. Cette seconde page Web du tutoriel comprend les différents champs de formulaires compatibles avec TactoWeb. Lorsque l'apprenant rencontre le premier champ ou la première étiquette, l'association entre ces deux éléments et la mise en page des formulaires lui est expliquée. Quand il rencontre un nouveau type de champ de formulaire, celui-ci lui est expliqué, ainsi que la façon lui permettant de saisir l'information requise par le champ. Le tuteur lui

[3] http://lrcm.com.umontreal.ca/greg/tactoweb/tuto.html

explique aussi que si l'association entre le champ et l'étiquette n'a pas été faite dans le code HTML, il peut toujours retrouver spatialement l'étiquette d'un champ et vice-versa. En effet, l'étiquette d'un champ se trouve très souvent juste à sa gauche, voire juste au-dessus.



(a) (b)

Figure 4-18 : Page Web du tutoriel pour les formulaires (a) et son affichage dans TactoWeb (b)

La dernière activité est en fait un contrôle permettant de vérifier si l'apprenant est à l'aise avec les différentes choses qu'il a apprises au cours du tutoriel. Deux lots de trois tâches ont été préparés. Si l'apprenant n'éprouve pas de difficulté à effectuer les tâches du premier lot, les tâches du second lot ne sont pas effectuées. Dans le cas contraire, le tuteur identifie les différentes choses qui n'ont pas été comprises par l'apprenant et lui fournit des explications supplémentaires. Les tâches du second lot sont ensuite effectuées par l'apprenant pour vérifier si les explications supplémentaires l'ont aidé à mieux comprendre le fonctionnement de TactoWeb.

La totalité du tutoriel est disponible à l'Annexe 2.

4.4 Synthèse et contribution

TactoWeb est une des contributions principales de cette thèse. Il s'agit d'un fureteur audio-tactile à l'usage des personnes non-voyantes leur permettant d'explorer spatialement une page Web à l'aide du Tactograph. Le Tactograph est un appareil comprenant un module tactile qui peut être déplacé sur une surface à deux dimensions. Le module tactile contient une cellule générant des sensations tactiles en fonction de l'endroit où le module se trouve sur la surface d'exploration. Nous avons d'ailleurs participé à la modification du Tactograph pour améliorer sa prise en main et permettre une meilleure exploration spatiale de pages Web.

TactoWeb fut développé en C++ et permet d'interpréter une page Web à partir de son URL en transformant les éléments Web qui la composent en TactoElement. Contrairement aux lecteurs d'écran, TactoWeb donne un accès au formatage et la mise en page de la page, et non pas seulement à son contenu, en calculant la taille et la localisation des TactoElement. Des retours tactiles et sonores (audicônes et synthèse vocale) sont ensuite attribués aux TactoElement afin de générer une adaptation audio-tactile complète de la page Web.

Étant donné que les personnes non-voyantes ne sont pas forcément familières avec l'exploration spatiale, un tutoriel a été développé afin de les former pour utiliser TactoWeb. Ce tutoriel était essentiel pour assurer le bon fonctionnement de l'expérimentation qui est décrite dans le chapitre suivant.

CHAPITRE 5 ÉTUDE COMPARATIVE DE LA PERFORMANCE HUMAINE BASÉE SUR L'EXPLORATION SPATIALE VS L'EXPLORATION LINÉAIRE DE SITES WEB

Dans le but de comparer l'exploration spatiale avec l'exploration linéaire et de tester nos hypothèses de recherche, nous avons réalisé une expérimentation faisant appel à plusieurs sujets humains. L'exploration spatiale a été effectuée avec TactoWeb, le fureteur audio-tactile que nous avons développé, tandis que l'exploration linéaire a été effectuée avec JAWS 12, le lecteur d'écran actuellement le plus utilisé par les personnes non-voyantes.

Ce chapitre présente dans la section 5.1 la méthodologie que nous avons mise en place pour mener notre étude comparative. Nous décrivons ensuite dans la section 5.2 comment nous avons recueilli et préparé les données issues de notre étude. Nous présentons enfin l'analyse de ces données dans la section 5.3.

5.1 Méthodologie

5.1.1 Participants

Quatorze personnes non-voyantes ont participé à l'expérimentation. Nous prévoyions à l'origine la participation de 20 participants mais nous n'avons pas réussi à atteindre ce nombre malgré les différentes annonces postées à l'INLB et au bureau de soutien aux étudiants en situation de handicap à l'Université de Montréal (Annexe 3). Dans tous les cas, nous souhaitions avoir un nombre pair de participants afin que chaque tâche soit effectuée par un même nombre de participants avec TactoWeb et avec JAWS. Afin d'avoir plus de participants, nous aurions pu recruter des personnes voyantes dont nous aurions bandé les yeux pour l'expérimentation. Néanmoins, les caractéristiques des personnes voyantes sont très différentes des personnes non-voyantes. Leur représentation mentale des pages Web et leur acuité tactile sont très différentes. Il aurait aussi fallu former les personnes voyantes à JAWS et ceci demande bien plus de temps d'apprentissage que l'utilisation d'un tutoriel de 45 minutes. De plus, nous avons trouvé préférable d'effectuer l'expérimentation avec les utilisateurs finaux de notre système.

Nous n'avions que deux exigences à satisfaire lors du recrutement de nos participants non-voyants. La première était qu'ils sachent utiliser JAWS pour naviguer sur le Web. Une

connaissance basique de JAWS était nécessaire pour effectuer les huit tâches de l'expérimentation. Faire appel à des novices de JAWS était exclu car nous ne pouvions pas les former à l'utilisation de ce lecteur d'écran dans un temps assez court. Il n'était pas nécessaire de connaître toutes les fonctionnalités et tous les raccourcis clavier disponibles dans JAWS. Nous leur avons néanmoins demandé leur degré de connaissance de JAWS afin d'en tenir compte dans l'analyse des résultats. Nous avions aussi pensé évaluer les habiletés des participants à utiliser JAWS en mesurant leur performance. Néanmoins, les questions posées lors du recrutement ont permis de lever tout doute sur la maîtrise de JAWS de la part des participants.

La seconde exigence était que les participants aient une bonne acuité tactile. Nous sommes partis du principe qu'une personne non-voyante possède une acuité tactile suffisante pour utiliser TactoWeb sauf si cette personne est atteinte de diabète, une des trois causes principales de la perte de la vision. En cas de diabète, l'acuité tactile peut aussi se détériorer. Dans le cas d'une personne diabétique, nous avons vérifié si elle pouvait lire le Braille avant de l'intégrer dans l'étude car une bonne acuité tactile est nécessaire pour effectuer ce genre de lecture.

Nous souhaitions aussi faire deux groupes de participants en fonction du degré d'expertise de l'utilisation de JAWS. En effet, cet outil exige beaucoup de temps d'apprentissage et certains utilisateurs non-voyants tardifs ne souhaitent pas l'utiliser pour cette raison. Nous espérions une différence dans l'utilisation de TactoWeb entre les experts et les non-experts de JAWS. Nous avions fixé la frontière entre non-experts et experts à cinq heures d'utilisation de JAWS par semaine. Nous nous sommes rapidement rendus compte que toutes les personnes répondant à notre annonce utilisaient JAWS plusieurs heures par jour. Le participant ayant le moins d'expertise avec JAWS passait tout de même 12 heures par semaine à l'utiliser.

Nous avons plutôt fait deux groupes de participants sur la base de la durée de leur cécité : depuis la naissance ou non. Nous avons eu sept participants dans chacun de ces deux groupes, ce qui fut idéal pour l'analyse des résultats. Comme nous l'avons vu dans la section 1.2.1.3, l'image mentale que se font les personnes non-voyantes de naissance est différente de celle que se font les personnes non-voyantes tardives. Nous étions donc curieux de voir si leurs résultats avec TactoWeb seraient différents.

Comme notre étude portait sur des sujets humains non-voyants, elle dut être approuvée par un comité d'éthique de la recherche (avec des sujets humains) autre que celui de l'École

Polytechnique de Montréal. En effet, chaque étude portant sur des sujets ayant un handicap dans la région de Montréal doit être approuvée par le CRIR (Centre de Recherche Interdisciplinaire en Réadaptation). Une fois le certificat d'éthique émis par le CRIR (voir Annexe 4), celui-ci fut validé par l'École Polytechnique de Montréal.

Le recrutement des participants s'est fait par l'intermédiaire de l'INLB et de l'Université de Montréal. Une fois en contact avec un éventuel participant, l'auteur de l'étude lui posait quelques questions (voir Annexe 5) sur son profil (tranche d'âge, nature du handicap visuel, niveau d'éducation) et ses habitudes d'utilisation d'outils adaptés (Braille, lecteur d'écran).

Le Tableau 5-1 présente les différentes caractéristiques de nos participants.

Tableau 5-1 : Caractéristiques des participants

Participant	Sexe	Âge (années)	Cécité Naiss.	Nature cécité	Jaws (années)	Racc. clavier	Utilisation JAWS (h/sem)	Occupation	Niveau académ.
1	M	41 – 50	Oui	Décollement rétine	> 6	Oui	100	Musicien	Sec 5
2	M	41 – 50	Non	Trauma crânien	> 6	Non	35	Maison	BAC
3	M	26 – 30	Oui	Amaurose de Leber	> 6	Oui	70	Téléphoniste	Sec 5
4	F	26 – 30	Oui	Rétinite pigmentaire	> 6	Oui	50	Étudiante	BAC
5	M	> 60	Non	Rétinite pigmentaire	> 6	Oui	25	Retraité	Maîtrise
6	F	51 – 60	Non	Dégénérative	> 6	Oui	65	Agente développement	BAC
7	M	41 – 50	Non	Décollement rétine	> 6	Oui	70	Informaticien	Collégial
8	F	41 – 50	Oui	Rétinite pigmentaire	> 6	Oui	60	BANQ	Collégial
9	F	41 – 50	Non	Rétinite pigmentaire	> 6	Oui	30	Maison	DESS
10	M	31 – 40	Oui	Cataracte	> 6	Oui	40	Maison	BAC
11	M	18 – 25	Oui	Mal formation	> 6	Oui	70	Étudiant	Certificat
12	F	> 60	Non	Rétinite pigmentaire	> 6	Non	20	Retraité	Sec 5
13	M	41 – 50	Oui	Amaurose de Leber	> 6	Oui	50	Enseignement Braille	DESS
14	M	41 – 50	Non	Glaucome - Cataracte	1 - 3	Oui	12	Étudiant	DESS

Nous avons eu neuf hommes et cinq femmes. Un participant se trouvait dans une tranche d'âge de 18 à 25 ans, deux dans une tranche de 26 à 30 ans, une dans la tranche de 31 à 40 ans, sept

dans la tranche de 41 à 50 ans, une dans la tranche de 51 à 60 ans, et deux participants avaient plus de 60 ans. Le nombre de personnes non-voyantes de naissance était égal au nombre de personnes non-voyantes tardives. Tous les participants sauf un utilisaient JAWS depuis plus de six ans, et seulement deux n'étaient pas familiers avec les raccourcis clavier avancés disponibles dans JAWS. Enfin, le niveau académique des participants variait grandement : trois avaient complété leurs études secondaires, deux avaient complété des études collégiales, quatre possédaient un baccalauréat, une possédait un certificat, trois détenaient un DESS, et une détenait une maîtrise.

5.1.2 Tâches

L'expérimentation fut divisée en deux parties selon l'outil utilisé. Dans la première partie, le participant effectuait des tâches de recherche d'information, de navigation, d'exploration et de remplissage de formulaire dans divers sites Web avec le logiciel de lecture d'écran JAWS. Dans la seconde partie, il effectuait des tâches similaires, quoique différentes, avec l'outil TactoWeb. L'ordre d'utilisation des deux outils fut contrebalancé pour avoir un même nombre de participants ayant commencé par chacun des deux outils.

Pour chaque outil, le participant devait effectuer quatre tâches correspondant à deux types. Le premier type consiste à trouver une certaine information dans un site Web en navigant librement dans les pages et en explorant librement chacune d'elles. Le second type de tâche consiste à explorer, remplir et soumettre un formulaire. Ainsi, le participant devait trouver une information et remplir un formulaire avec chaque outil dans deux sites Web différents : un site accessible, c.-à-d. respectant les lignes directrices pour personne handicapées et un site inaccessible, c.à-d. posant des problèmes lors de l'utilisation d'un lecteur d'écran. Nous souhaitions ainsi démontrer que contrairement à JAWS, TactoWeb permet d'accéder aussi bien à des sites accessibles qu'à des sites non accessibles.

Le plan expérimental de l'étude fait varier l'ordre des quatre variables prises en compte (trois variables indépendantes et une contrôlée) dans le but de contrebalancer l'effet d'apprentissage sur la performance et la satisfaction des participant. Ces quatre variables sont les suivantes :

- L'outil utilisé pour effectuer la tâche : JAWS et Tactoweb.
- Le type de tâche effectué : recherche d'information ou formulaire;

- L'accessibilité site Web : accessible vs non accessible;
- La période d'apparition de la cécité (contrôlée) : depuis la naissance ou plus récemment

Le Tableau 5-2 décrit comment nous avons monté notre plan expérimental en fonction de nos trois variables indépendantes. Nous n'avons pas contrebalancé la variable « période d'apparition de la cécité » parce que nous n'étions pas certains de pouvoir recruter autant de participants non-voyants de naissance que de sujets tardifs. Nous ne voulions pas que ce facteur soit déterminant dans la sélection de nos sujets à cause de la difficulté à recruter des participants non-voyants.

Tableau 5-2 : Répartition des tâches en fonctions des variables indépendantes

Outil	Accessibilité	Type de tâche	Nom tâche	NB Participant
JAWS	Site Accessible	Recherche Information	S1	7
			S2	7
		Formulaire	Q1	7
			Q2	7
	Site Non Accessible	Recherche Information	S3	7
			S4	7
		Formulaire	Q3	7
			Q4	7
TactoWeb	Site Accessible	Recherche Information	S1	7
			S2	7
		Formulaire	Q1	7
			Q2	7
	Site Non Accessible	Recherche Information	S3	7
			S4	7
		Formulaire	Q3	7
			Q4	7

Les deux prochaines sous-sections décrivent les huit tâches. Les consignes et conditions de succès sont présentées ainsi que les différents problèmes d'accessibilité pouvant être rencontrés avec JAWS pour les tâches effectuées dans des sites non-accessibles. Les impressions d'écran des tâches ainsi que leur rendu dans TactoWeb sont disponibles à l'Annexe 6.

5.1.2.1 Tâches de recherche d'information

S1 : Gouvernement du Québec – Déménagement (accessible)

Consigne : vous déménagez prochainement, vous voulez savoir comment aviser les différents services gouvernementaux de votre changement d'adresse. Vous débutez sur la page d'accueil du gouvernement du Québec[4].

Critère de succès : accéder à la page Déménager du site du gouvernement du Québec[5]. Cette page peut être atteinte depuis un lien dans le menu de droite de la page d'accueil ou en passant par la section « Services Québec Citoyens » du menu principal à gauche de la page d'accueil.

Description : le but de cette tâche était de vérifier s'il est possible de faire facilement avec TactoWeb une tâche qui est relativement basique et très rapide à faire avec JAWS. Ce site a une organisation classique : haut de page, menu principal vertical à gauche, contenu principal au centre.

S2 : Bibliothèque et archives nationales du Québec (BANQ) – Services adaptés (accessible)

Consigne : on vous a parlé du Service adapté de la Grande bibliothèque de Montréal. Vous désirez obtenir plus d'information sur ce service. Vous débutez sur la page d'accueil de la BANQ[6].

Critère de succès : accéder à la page des services adaptés[7]. Cette page peut être atteinte directement depuis un lien situé dans le menu horizontal de la page d'accueil ou en passant par la section « Services ». Le participant peut aussi passer par les sections « Services de A à Z » ou « Services spécialisés ».

Description : le but de cette tâche est pratiquement identique à celui de la tâche S1 même si elle peut prendre un peu plus de temps. De plus, le fait que le menu soit orienté horizontalement sur la page d'accueil peut augmenter la difficulté dans TactoWeb.

---

[4] http://www.gouv.qc.ca/portail/quebec/pgs/commun/

[5] http://www4.gouv.qc.ca/FR/Portail/Citoyens/Evenements/changement-adresse/Pages/accueil.aspx?liens_fr

[6] http://www.banq.qc.ca/accueil/

[7] http://www.banq.qc.ca/services/services_specialises/services_adaptes/index.html

S3 : Auberge du Jardin au Petit Saguenay – Tarifs d'une chambre (non-accessible)

Consigne : vous partez bientôt en voyage dans le Saguenay. On vous a parlé de l'auberge du Jardin, dans le Petit Saguenay. Vous souhaitez consulter le tarif de sa chambre confort pour 2 personnes, incluant le petit déjeuner. Vous débutez sur la page d'accueil de l'auberge[8].

Critère de succès : se rendre sur la page des tarifs[9] puis donner le prix exact d'une chambre confort pour 2 personnes, incluant le petit déjeuner, soit 64 $ par personne. La page des tarifs ne peut être atteinte qu'en passant par le menu principal du site qui se situe sur la gauche.

Description : cette tâche est la seule tâche de recherche d'information où le participant doit trouver une information précise. En plus de naviguer dans le site, il doit explorer attentivement la page des tarifs dans le but de trouver le tableau qui contient, entre autres, le prix demandé.

Problèmes d'accessibilité : le principal problème d'accessibilité concerne le tableau des tarifs pour une chambre. Celui-ci n'est pas accessible car ses lignes et colonnes d'en-tête n'ont pas été codées pour cela. Un autre problème est que des titres sont utilisés pour faire la mise en page de beaucoup d'éléments, même des paragraphes. Donc le lecteur d'écran interprète des paragraphes comme des titres alors qu'ils ne le sont pas. De plus, plusieurs images n'ont pas de texte de description.

S4 : Collège André-Grasset – Condition d'admission (non accessible)

Consigne : vous venez vous renseigner sur le programme « Science de la nature » du collège André-Grasset pour votre enfant. Vous souhaitez connaître les conditions d'admission pour ce collège. Vous débutez sur la page du programme Science de la nature du collège[10].

Critère de succès : se rendre à la page des conditions d'admission du collège[11]. Pour ce faire, le participant doit passer par l'image-lien « Admission » de menu principal à gauche pour accéder à la page « Admission ». Il peut ensuite accéder aux conditions d'admission grâce à

---

[8] http://www.aubergedujardin.com/

[9] http://www.aubergedujardin.com/tarifs.html

[10] http://www.grasset.qc.ca/Programmes-d-etudes/sciences-de-la-nature-276.php

[11] http://www.grasset.qc.ca/Admission/admission-327.php

un sous-menu de la page d'admission. Il est possible aussi que le participant passe par la page d'accueil pour accéder à la page « Admission ».

Description : c'est la tâche qui demande le plus d'effort en termes de navigation. En effet, le participant doit passer par au moins une page intermédiaire avant de trouver la bonne page.

Problèmes d'accessibilité : le principal problème d'accessibilité est que l'ordre logique de lecture de la page n'est pas respecté dans le code. Le menu principal de droite n'est codé, donc lu avec JAWS, qu'à la fin de la page. De plus, ce menu est constitué d'image-liens qui n'ont aucun texte de description ce qui le rend très difficile à explorer. Le plus étrange est que les images liens du menu principal de la page d'accueil ont, elles, des textes de description.

5.1.2.2 Tâches de remplissage de formulaire

Par souci d'équité entre l'exploration spatiale et l'exploration linéaire, le remplissage d'un champ de formulaire se fait oralement dans les deux cas car il est pour l'instant impossible d'entrer des données avec TactoWeb. Nous ne voulions pas que les participants passent du temps à remplir des champs au clavier dans JAWS alors qu'ils devaient de le faire oralement dans TactoWeb.

Q1 : Gouvernement du Québec – Nous joindre par courriel (accessible)

Consigne : vous souhaitez contacter le gouvernement du Québec pour lui indiquer que son service de dépôt direct pour le crédit d'impôt ne vous est pas accessible. Vous débutez sur la page de contact par courriel du gouvernement du Québec[12].

Critère de succès : remplir le formulaire et le soumettre.

Description : ce formulaire est le plus simple des quatre. Il ne contient que trois champs de saisie de texte et un bouton de soumission. C'est un des plus simples qu'on puisse trouver sur le Web et il est parfaitement accessible.

Q2 : IGA – Inscription à l'infobulletin (accessible)

Consigne : vous souhaitez vous inscrire à l'infobulletin d'IGA. Vous débutez sur la page d'inscription à l'infobulletin d'IGA[13].

---

[12] http://www.gouv.qc.ca/portail/quebec/pgs/commun/nousjoindre/courriel?lang=fr&s=0

Critère de succès : remplir le formulaire et le soumettre. Pour que le succès soit complet, la première boîte à cocher doit être cochée car c'est elle qui active l'inscription à l'infobulletin.

Description : ce formulaire est un exemple classique d'inscription à un infobulletin. La plupart des champs sont pour la saisie de texte mais il y a tout de même un groupe de boutons radio ainsi que deux cases à cocher.

Problème d'accessibilité : même si ce formulaire est accessible, deux problèmes mineurs sont à noter. Le premier est que les champs obligatoires ne sont pas indiqués. C'est surtout un problème d'utilisabilité car même les personnes sans handicap ne peuvent pas identifier correctement les champs obligatoires. Le second est que JAWS lit la page avec une synthèse vocale en anglais alors que le texte de la page est en français. Cela vient d'une erreur au début du code de la page Web qui indique que la langue utilisée pour cette page est l'anglais.

Q3 : Le point – Inscription à la newsletter (non-accessible)

Consigne : vous souhaitez vous inscrire à l'infolettre du journal « Le Point ». Vous débutez sur la page d'inscription à l'infolettre du journal « Le Point »[14].

Critère de succès : remplir le formulaire et le soumettre.

Description : c'est le même genre de formulaire que celui de la tâche Q2. Il possède en plus plusieurs listes de sélection.

Problèmes d'accessibilité : plusieurs étiquettes ne sont pas codées comme telles, mais plutôt comme du texte simple à l'intérieur d'une cellule de tableau. Cela empêche le lecteur d'écran de faire correctement l'association entre les étiquettes et les champs de formulaire. De plus, du texte est présenté sous forme d'image sans texte de description. Enfin, l'ordre logique de lecture n'est pas respecté pour la majeure partie du formulaire. Seul le champ « email » est

---

[13] http://www.iga.net/ci/index.php/fr/infobulletin/

[14] http://inscription.infos-lepoint.fr/default.asp

correct, avec l'étiquette qui est lue juste avant le champ. Pour le reste du formulaire, toutes les étiquettes sont lues, puis ensuite tous les champs sont lus.

Q4 : Disquaire Martin Lévesque – S'inscrire (non-accessible)

Consigne : vous souhaitez vous inscrire sur le site du disquaire Martin Lévesque pour acheter des disques de musique venant de ce magasin sur Internet. Vous débutez sur la page d'inscription du disquaire Martin Lévesque[15].

Critère de succès : remplir le formulaire et le soumettre.

Description : il s'agit d'un formulaire classique d'inscription à un site Web. Il contient des champs de saisie de texte, un groupe de boutons radio et une liste de sélection. C'est la seule page Web qui est fictive. Nous l'avons créée pour l'expérimentation car nous voulions recréer un certain type de problème d'accessibilité des formulaires qui nous avait été décrit.

Problèmes d'accessibilité : ce formulaire est ce qu'on peut trouver de pire en terme d'accessibilité. Il n'y a aucune relation entre les étiquettes et les champs de formulaires. De plus, l'ordre logique de lecture n'est pas respecté car toutes les étiquettes sont lues les unes à la suite des autres, suivies par tous les champs, eux aussi lus les uns à la suite des autres.

5.1.3 Variables étudiées

Pour cette expérimentation, nous avons trois variables indépendantes (les trois premières ci-dessous) et une variable contrôlée (la dernière); nous nous sommes assurés de les contrebalancer dans le plan expérimental. Ces variables sont les suivantes :

- Type d'exploration : aussi appelée « Outil utilisé ». L'exploration peut être spatiale via TactoWeb, ou linéaire via JAWS. Il s'agit de la principale variable indépendante, c'est elle qui nous intéresse en premier lieu pour ce qui est de l'impact sur les variables dépendantes et qui est essentielle pour confirmer notre hypothèse.
- Types de tâches : il s'agit soit de rechercher une information, soit de remplir un formulaire. Nous voulions tester ces deux types de tâches car les problèmes d'accessibilité ne sont pas les mêmes dans ces deux cas.

---

[15] http://lrcm.com.umontreal.ca/greg/tactoweb/form.html

- Accessibilité du site Web avec JAWS : certains sites explorés au cours de la tâche respectent les lignes de conduite d'accessibilité tandis que d'autres sites ne le font pas. Comme indiqué précédemment, nous voulions savoir si TactoWeb permet d'éviter certains problèmes d'accessibilité. Il était donc pertinent de faire des tâches dans des sites accessibles et dans des sites non-accessibles.
- Période d'apparition de la cécité (variable contrôlée) : un participant peut être aveugle de naissance ou non. Nous nous attendons à certaines différences de performance entre ces deux types de participants car les non-voyants peuvent avoir des notions de navigation spatiale datant du temps où ils voyaient encore.

Les variables dépendantes sont typiques des variables mesurées lors de tests d'utilisabilité des interfaces utilisateurs. Nous n'avons pas demandé aux participants de penser tout haut pendant les tâches car nous mesurions leur temps d'exécution. Voici la liste des variables dépendantes :

- Taux de succès dans la tâche : cela correspond à l'efficacité. Le calcul de ce taux dépend du type de tâche. Pour les tâches de recherche d'information, le taux est égal à 100% si le participant se rend sur la bonne page à l'exception de S3. Pour cette tâche, le taux est de 100% si le participant trouve le bon prix pour la chambre. Le taux diminue de 25% si le participant se trompe d'une ligne dans le tableau indiquant la liste des prix de la chambre. Pour les formulaires, le taux de succès correspond au pourcentage de champs obligatoires correctement remplis. Le fait qu'un champ ait été laissé vide alors qu'il était obligatoire compte aussi comme un champ erroné. Ici, le meilleur taux de succès qu'un participant puisse obtenir est 100%, qui est équivalent à 1 dans nos analyses statistiques.
- Temps d'exécution de la tâche : cela correspond à une partie de l'efficience. Le temps démarre lorsque le participant appuie pour la première fois sur une touche du clavier pour JAWS, et lorsque qu'il commence à bouger le module tactile pour TactoWeb. Le temps s'arrête dans tous les cas lorsque le participant accède à la bonne information ou lorsqu'il soumet le formulaire. Concernant la tâche S3, on peut aussi regarder plus spécifiquement le temps d'exploration du tableau des tarifs de la chambre confort. Ici le meilleur score qu'un participant puisse obtenir est le temps d'exécution le plus court.
- Niveau de difficulté rencontrée au cours de la tâche : cela correspond à l'autre partie de l'efficience. Le calcul du niveau de difficulté rencontrée au cours de la tâche correspond à

une valeur pointant vers l'infini. Une valeur de 0 correspond à une difficulté nulle, et plus cette valeur augmente, plus la difficulté rencontrée augmente. Nous avons établi cette valeur en fonction de notre observation des difficultés rencontrées par le participant. Ajouter un point à la difficulté dépend du type de tâche. Pour les tâches de recherche d'information, on ajoute un point de difficulté lorsque le participant accède à une page Web qui fait partie du chemin possible pour accéder à l'information demandée. On ajoute deux points de difficulté si le participant accède à une page Web qui ne fait pas partie du chemin possible pour accéder à l'information, s'il appuie sur le bouton « Précédent », ou s'il accède à l'éventuelle page contenant le plan du site. Pour les tâches de formulaire, on ajoute un point de difficulté lorsque le participant corrige un champ qu'il avait déjà rempli. Ici le meilleur score qu'un participant puisse obtenir est 0 et le pire ne peut pas être défini car il dépend de la tâche.

- Satisfaction de l'utilisateur : après chaque ensemble de quatre tâches, on a posé plusieurs questions sur l'expérience et la satisfaction de l'utilisateur à l'égard de l'outil utilisé. Certaines questions portaient sur les deux outils tandis que d'autres étaient spécifiques à un outil. C'était en majorité des questions fermées où le participant devait choisir une valeur entre un et cinq (meilleur score). La signification des extrêmes de l'échelle était spécifiée à chaque question. Nous avons choisi un nombre impair de valeurs pour que les participants aient la possibilité de choisir une valeur médiane. Trois questions ouvertes furent aussi posées pour chaque outil. Trois questions ouvertes de satisfaction furent posées à la fin de l'expérimentation afin de comparer l'expérience des participants avec les deux outils. Le questionnaire est disponible à l'Annexe 7.
- Image globale de la page Web : une question fermée portant sur l'image globale de la page Web fut posée dans le questionnaire.

5.1.4 Procédure expérimentale

Deux exemplaires du formulaire de consentement furent envoyés aux participants quelques jours avant l'expérimentation. Le but visé était de faire en sorte que les participants soient plus à l'aise avec l'expérimentation et puissent poser toutes les questions qu'ils souhaitent avant de prendre part à l'expérimentation. Nous avions prévu un maximum de trois heures pour la durée totale de l'expérimentation : nous avons opté pour des sessions d'une demi-journée pour chaque test.

Le jour de l'expérimentation, le modérateur attendait le participant à l'entrée du pavillon Marie-Victorin de l'Université de Montréal ou allait le chercher à une station de métro. Une fois dans la salle d'expérimentation, le participant remettait les deux copies du formulaire de consentement au modérateur qui lui en rendait une signée. L'expérimentation pouvait alors commencer et durait entre deux et trois heures. Il n'a pas été demandé aux participants de penser à voix haute durant leur travail afin de ne pas créer de surcharge cognitive qui aurait augmenté les temps d'exécution.

Les points suivants furent ensuite expliqués au participant :

- Vous allez utiliser deux outils: JAWS que vous connaissez déjà, et TactoWeb qui permet d'explorer tactilement des pages Web.
- Pour chacun des outils, vous aurez à effectuer deux types de tâches : trouver une information dans un site Web et remplir un formulaire. Vous ferez chaque type de tâche dans deux sites Web, ce qui fait un total de quatre tâches. Pour les formulaires, je ne vous demanderai pas de remplir les champs mais plutôt de me dire à l'oral ce que vous mettriez dans le champ quand vous êtes dessus avec JAWS ou le module tactile de TactoWeb.
- C'est l'outil que nous évaluons et non pas votre performance. Si vous voulez abandonner une tâche, il n'y a aucun problème, vous pouvez le faire quand vous le voulez.
- Sachant que vous n'avez jamais utilisé TactoWeb, nous procèderons à une phase d'entraînement vous permettant de vous familiariser avec les retours tactiles et sonores, ainsi qu'avec l'exploration spatiale.
- Après avoir utilisé chaque outil, une série de questions vous sera posée. Les questions portent sur votre expérience d'aujourd'hui avec les outils, et non pas sur l'ensemble de votre expérience avec JAWS ou n'importe quel appareil tactile.

Ensuite, le participant commençait avec le premier outil. L'ordre d'utilisation des outils JAWS et TactoWeb alternait d'un participant à l'autre. Le tutoriel de TactoWeb fut suivi par le participant avant d'effectuer les quatre tâches avec cet outil. Pour chaque tâche, le modérateur indiquait la consigne et la page de départ au participant (voir 5.1.2.1 et 5.1.2.2). Il intervenait que lorsque le participant pensait avoir trouvé l'information ou s'il décidait d'abandonner la tâche. Une fois les quatre tâches effectuées avec le premier outil, le participant répondait à une série de questions portant sur son expérience avec l'outil et son appréciation. Il passait ensuite aux quatre tâches à

effectuer avec le second outil puis répondait aux questions sur celui-ci. Trois questions de comparaison des deux outils lui étaient ensuite posées.

Enfin, le modérateur demandait si le participant avait des questions ou des remarques générales sur l'expérimentation. Il donnait la compensation financière au participant et lui faisait signer un reçu. Il raccompagnait le participant à l'entrée du pavillon ou sur le quai d'une station de métro.

5.2 Recueil des données

Chaque séance de test fut enregistrée de façon à avoir une trace des retours visuels donnés par TactoWeb ainsi que des retours sonores de JAWS et TactoWeb. Les commentaires des participants à la fin de chaque séance furent aussi enregistrés. La vidéo de chaque séance fut ensuite marquée au moyen du logiciel de balisage Observer XT (Noldus, 2013). Ce logiciel permet d'insérer des marqueurs dans une vidéo afin de noter manuellement l'occurrence d'événements ponctuels, ainsi que le début et la fin d'événements, ce qui permet alors de connaître la durée des tâches et la durée d'exploration du tableau de S3. Nous avons donc pu mesurer précisément la durée de chaque tâche ainsi que les événements ponctuels suivants, afin de calculer le taux de succès, le temps d'exécution et le niveau de difficulté des tâches :

- Accès à une page incluse dans le chemin possible pour accéder à l'information demandée.
- Accès à une page hors du chemin possible pour accéder à l'information demandée.
- Retour à la page précédente.
- Temps au premier lien cliqué.
- Champ de formulaire mal rempli.
- Champ de formulaire oublié.
- Champ de formulaire corrigé.
- Temps au premier champ rempli.

Une fois toutes les vidéos codées avec Observer XT, nous avons exporté tout l'encodage dans un fichier Excel. Nous avons traité l'encodage brut pour calculer le nombre d'occurrences des événements ponctuels et la durée exacte des tâches. Ce traitement nous a ensuite permis de calculer les taux de succès de chaque tâche et les niveaux de difficulté rencontrés dans chacune, pour ensuite préparer les données de l'analyse. Les données recueillies au moyen des questions

fermées du questionnaire ont elles aussi été traitées via Excel et préparées pour l'analyse. Les réponses aux questions ouvertes ont été compilées séparément dans un fichier Word.

5.3 Analyse des résultats

Toutes les données traitées via Excel ont été importées puis analysées avec le logiciel d'analyse SPSS Statistics 20 (IBM, 2013). Le seul travail préliminaire à faire avant d'importer les données d'Excel fut de créer dans SPSS les variables correspondantes aux colonnes des fichiers Excel.

5.3.1 Analyses des tâches

Comme nous l'avons vu dans la section 5.1.3, nous avons testé trois variables indépendantes et manipulé une variable contrôlée, cette dernière étant traitée comme une variable indépendante dans les analyses statistiques. Pour simplifier l'écriture, nous parlons de quatre variables indépendantes dans la suite du texte. Chaque variable indépendante avait deux valeurs possibles : JAWS vs TactoWeb; tâche de recherche d'information vs remplissage d'un formulaire; site Web accessible vs non-accessible; cécité de naissance vs tardive.

Considérant que l'ordre des tâches était contrôlé, puisque nous avons contrebalancé toutes les variables indépendantes à l'exception de la variable contrôlée, nous souhaitions faire plusieurs tests de Student (test-t) pour échantillons appariés ou échantillons indépendants. Néanmoins, nous nous sommes rapidement rendu compte que nos données ne suivaient pas une distribution normale, ce qui est une hypothèse de base lors de l'utilisation des test-t. Nous avons donc transformé nos données afin d'obtenir une distribution normale. La transformation logarithmique nous a permis d'obtenir une distribution normale pour les temps d'exécution car leur distribution était à l'origine exponentielle. Pour nos deux autres variables dépendantes, le taux de succès et la difficulté rencontrée, la transformation logarithmique n 'était pas suffisante. Nous avons dû utiliser la transformation de Box-Cox qui est beaucoup plus puissante pour linéariser la distribution de données (Osborne, 2010).

Nous avons effectué des test-t pour échantillons appariés sur un ensemble de tâche dépendant de la valeur de la variable indépendante du type de tâche, soit sur l'ensemble des tâches et deux sous-ensembles : les tâches de recherche d'information et les tâches de remplissage de formulaire. Il est plus pertinent d'utiliser le type de tâche comme filtre de sous-ensembles plutôt

que comme variable indépendante d'analyse car le calcul des taux de succès et des niveaux de difficulté dépend du type de tâche. Nous avons aussi formé des sous-ensembles en prenant en compte les tâches effectuées dans des sites accessibles ou non-accessibles. Nous avons aussi étudié l'ensemble des tâches effectuées dans des sites accessibles, puis dans des sites non-accessibles, afin de valider ou non notre sous-hypothèse 1.

Nous avons du travailler sur des échantillons appariés car nos participants se retrouvaient dans les deux groupes pour chaque sous-ensemble. Même si nos participants n'effectuaient pas les mêmes tâches avec les deux outils, ils effectuaient quand même les mêmes types, ou sous-types, de tâches avec les deux outils. C'est ainsi que nous pouvons justifier l'appariement de nos échantillons dans le cas d'étude d'un ensemble ou sous-ensemble de tâches.

Lorsque nous voulions étudier seulement une seule tâche, nous avons fait un test-t pour échantillons indépendants car, cette fois-ci, aucun participant n'était dans les deux groupes. Nous avons regardé chaque tâche séparément pour identifier si certains traitements d'éléments Web fonctionnaient mieux avec l'exploration linéaire ou avec l'exploration spatiale.

Pour résumer, nous avons étudié la variance de chacune des variables dépendantes citées précédemment en fonction de l'outil utilisé dans nos trois ensembles (ensemble global et deux sous-ensembles en fonction du type de tâche). Nous avons aussi effectué des test-t pour échantillons indépendants en fonction de la période d'apparition de la cécité pour observer les différences de performances entre les participants non-voyants de naissance et tardifs.

La liste suivante résume les différentes analyses statistiques que nous avons effectuées en fonction des ensembles que nous avons définis. Pour chaque analyse statistique, les variables dépendantes sont le taux de succès dans la tâche, le niveau de difficulté rencontrée au cours de la tâche, ainsi que le temps d'exécution de la tâche.

Ensemble des tâches (hypothèse principale)

- Test-t pour échantillons appariés en fonction de l'outil
- Test-t pour échantillons appariées en fonction de l'outil dans les tâches portant sur des sites accessibles (sous-hypothèse 1)
- Test-t pour échantillons appariés en fonction de l'outil dans les tâches portant sur des sites non-accessibles (sous-hypothèse 1)

- Test-t pour échantillons appariés en fonction l'apparition de la cécité chez les participants utilisant JAWS
- Test-t pour échantillons appariés en fonction l'apparition de la cécité chez les participants utilisant TACTOWEB

Sous-ensemble 1 : tâches de recherche d'information (sous-hypothèse 2)

- Test-t pour échantillons appariés en fonction de l'outil
    - Test-t pour échantillons appariés en fonction de l'outil dans les tâches portant sur des sites accessibles (sous-hypothèse 1)
        - Test-t pour échantillons indépendants en fonction de l'outil dans S1
        - Test-t pour échantillons indépendants en fonction de l'outil dans S2
    - Test-t pour échantillons appariés en fonction de l'outil dans les tâches portant sur des sites non-accessibles (sous-hypothèse 1)
        - Test-t pour échantillons indépendants en fonction de l'outil dans S3
        - Test-t pour échantillons indépendants en fonction de l'outil dans S4

Sous-ensemble 2 : tâches de remplissage de formulaire (sous-hypothèse 3)

- Test-t pour échantillons appariés en fonction de l'outil et de l'accessibilité des sites
    - Test-t pour échantillons appariés en fonction de l'outil dans les tâches portant sur des sites accessibles (sous-hypothèse 1)
        - Test-t pour échantillons indépendants en fonction de l'outil dans Q1
        - Test-t pour échantillons indépendants en fonction de l'outil dans Q2
    - Test-t pour échantillons appariés en fonction de l'outil dans les tâches portant sur des sites non-accessibles (sous-hypothèse 1)
        - Test-t pour échantillons indépendants en fonction de l'outil dans Q3
        - Test-t pour échantillons indépendants en fonction de l'outil dans Q4
- Test-t pour échantillons appariés en fonction l'apparition de la cécité chez les participants utilisant JAWS

5.3.1.1 Analyses globales

Les analyses globales sont des test-t pour échantillons appariés portant sur l'ensemble des tâches, sans prendre en compte leur type, en fonction de l'outil utilisé.

La Figure 5-1 nous montre les différences entre les moyennes. On peut remarquer que les moyennes de taux de succès dans la tâche (a) sont très similaires en fonction de l'outil utilisé avec un très léger avantage pour TactoWeb. Pour ce qui est du niveau de difficulté rencontrée au cours de la tâche (b), on remarque un léger avantage pour JAWS dans les sites accessibles alors que c'est le contraire pour les sites non-accessibles, avec un avantage plus important pour TactoWeb. Par contre, l'avantage pour JAWS est très net pour ce qui est du temps d'exécution des tâches, surtout celles qui ont été faites dans des sites accessibles.

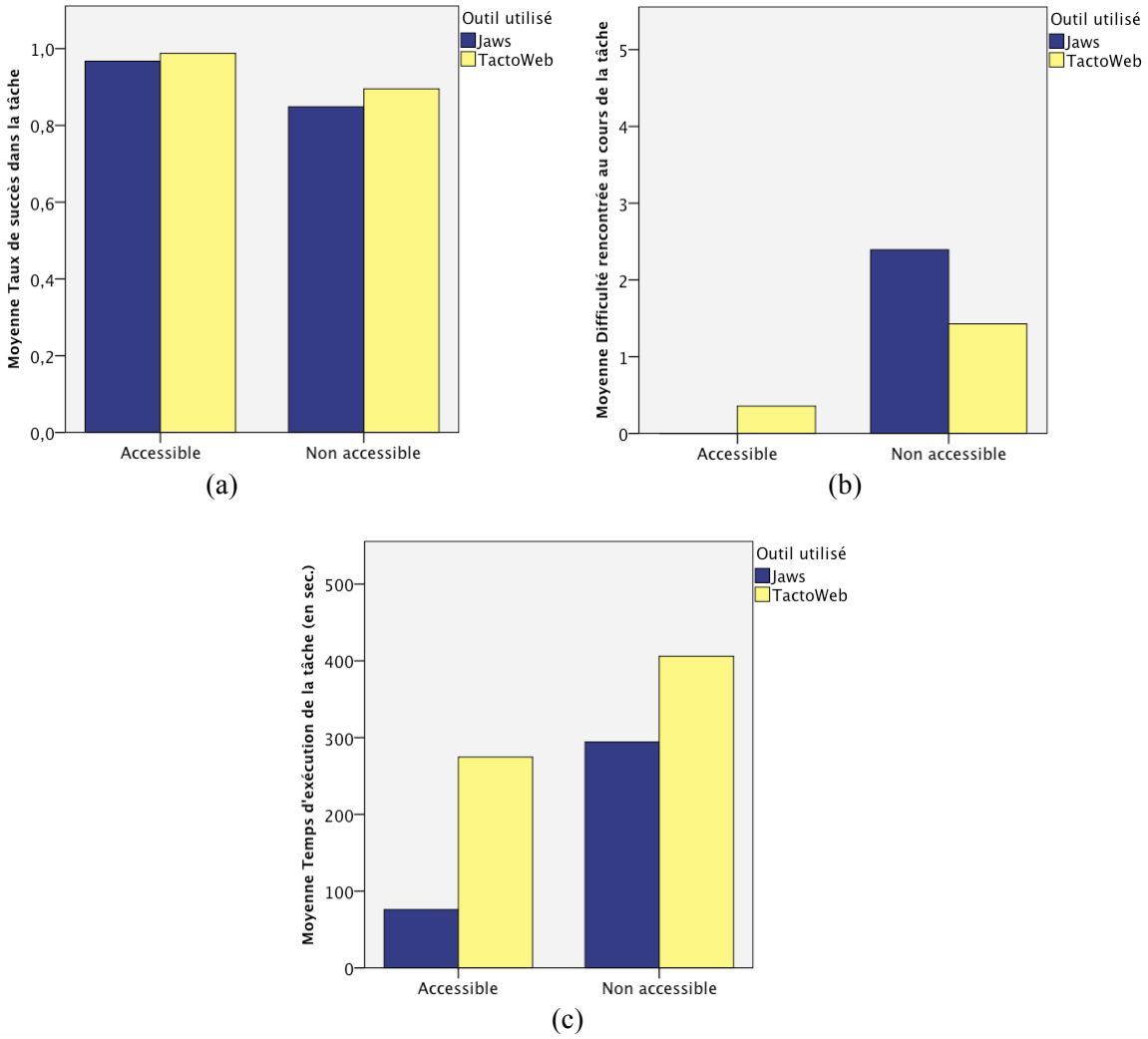


Figure 5-1 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b), et des temps d'exécution (c) en fonction de l'outil utilisé et de l'accessibilité des sites Web pour l'ensemble des tâches

Néanmoins, l'analyse statistique (voir Tableau 5-3) révèle que seulement la différence de moyennes des temps d'exécution de la tâche est significative (p = 0,000). Les différences de

moyennes des taux de succès et des niveaux de difficulté sont minimes : 0,034 point pour le taux de succès et 0,309 point pour le niveau de difficulté, tous les deux à l'avantage de TactoWeb mais avec un taux de signification le plus élevé possible (p = 1.000).

Tableau 5-3 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution de l'ensemble des tâches en fonction de l'outil

Toutes les tâches	Moyenne			Test-t sur échantillons appariés		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	0,907	**0,940**	-0,034	0,000	55,000	1,000
Difficulté rencontrée	1,218	**0,909**	0,309	0,000	55,000	1,000
Temps d'exécution	**187,125**	341,549	-154,424	-6,774	53,000	**0,000**

Grâce à cette première analyse, on peut en déduire que JAWS est globalement plus rapide que TactoWeb. Une explication possible tient à la très grande différence d'expérience des participants avec les deux outils et à la différence de maturité entre ces derniers. En effet, TactoWeb n'a été utilisé qu'entre 30 et 50 minutes au cours du tutoriel (moyenne de 38 minutes) avant la première utilisation alors que les participants utilisaient JAWS depuis des années, plusieurs heures par jour. De plus, il ne s'agit que de la première version de TactoWeb alors que JAWS existe depuis presque 20 ans. Il serait intéressant de voir si les temps d'exécution de TactoWeb s'amélioreront avec la maturation technologique de TactoWeb, mais surtout avec plus de temps utilisation. Par contre, ces différences d'expérience et de maturité ne jouent pas sur le taux de succès et le niveau de difficulté. Ce point est très positif car nous arrivons globalement avec TactoWeb à une efficacité équivalente à celle de JAWS avec une durée d'apprentissage beaucoup plus courte.

Notre seconde analyse statistique globale porte sur l'ensemble des tâches effectuées dans des sites accessibles (Tableau 5-4). Les résultats sont assez similaires à la première analyse avec seulement les différences de moyennes des temps d'exécution étant significatives (p = 0,000).

Tableau 5-4 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches dans les sites accessibles en fonction de l'outil

Toutes les tâches (accessible)	Moyenne			Test-t sur échantillons appariés		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	0,967	**0,988**	-0,021	-1,609	27,000	0,119
Difficulté rencontrée	**0,000**	0,370	-0,370	-0,219	27,000	0,829
Temps d'exécution	**76,022**	274,615	-198,593	-10,103	25,000	**0,000**

L'interprétation de cette analyse est aussi similaire à la précédente. Nous obtenons une efficacité similaire avec les deux outils alors que les participants avaient moins d'expérience avec TactoWeb, qui est un outil beaucoup plus récent. Ces différences d'expérience et de maturité ont pu jouer énormément sur le temps d'exécution de la tâche dans les sites accessibles.

Si on effectue maintenant le même analyse sur les tâches portant sur des sites non-accessibles (Tableau 5-5), la seule différence significative concerne encore seulement le temps d'exécution (p = 0,035). Néanmoins, la différence entre les moyennes est de 112 secondes alors qu'il était de 199 secondes pour les tâches portant sur des sites accessibles.

Tableau 5-5 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches dans les sites non-accessibles en fonction de l'outil

Toutes les tâches (non accessible)	Moyenne			Test-t sur échantillons appariés		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	0,848	**0,894**	-0,046	0,525	27,000	0,604
Difficulté rencontrée	2,393	**1,429**	0,964	0,063	27,000	0,950
Temps d'exécution	**294,260**	406,092	-111,832	-2,224	27,000	**0,035**

Le fait que la différence entre les moyennes des temps d'exécution ait diminuée de presque une minute et demi dans les sites non-accessibles montre que TactoWeb performe encore mieux dans ce cas. Il reste toujours moins efficient que JAWS, fort vraisemblablement à cause des différences d'expérience et de maturité entre les deux outils. L'efficacité dans les sites accessibles est similaire pour les deux outils, donc à l'avantage de TactoWeb qui bénéficie d'un temps de pratique moindre.

La dernière analyse globale est composée de deux test-t pour échantillons indépendants que nous avons effectués sur les mêmes variables dépendantes, mais en fonction la période d'apparition de la cécité. Nous avons observé les différences entre les participants non-voyants de naissance et tardifs dans JAWS (Tableau 5-6) et TactoWeb (Tableau 5-7). Ces analyses n'ont fait ressortir aucune différence de moyennes en fonction de la période d'apparition de la cécité. Néanmoins, les personnes non-voyantes de naissance semblent plus rapides que les personnes non-voyantes tardives avec les deux outils malgré des écarts non significatifs. Nous avons un écart de 66 secondes dans JAWS (p = 0,703) et un écart de 98 secondes dans TactoWeb (p = 0,193).

Tableau 5-6 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution des tâches en fonction de l'apparition de la cécité chez les participants utilisant JAWS

Tâches effectuées avec JAWS	Moyenne			Test-t sur échantillons indépendants		
	Naissance	Tardif	Ecart (N-T)	t	df	valeur-p
Taux de succès	**0,909**	0,904	0,005	0,082	54,000	0,935
Difficulté rencontrée	1,590	**0,670**	0,920	1,265	39,471	0,213
Temps d'exécution	**158,542**	224,045	-65,503	-0,383	53,000	0,703

Tableau 5-7 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution des tâches en fonction de l'apparition de la cécité chez les participants utilisant TactoWeb

Tâches effectuées avec TACTOWEB	Moyenne			Test-t sur échantillons indépendants		
	Naissance	Tardif	Ecart (N-T)	t	df	valeur-p
Taux de succès	**0,960**	0,914	0,047	1,670	37,804	0,103
Difficulté rencontrée	1,060	**0,670**	0,390	1,021	54,000	0,312
Temps d'exécution	**300,754**	398,307	-97,554	-1,318	53,000	0,193

Nous nous attendions à observer une plus grande aisance pour les participants non-voyants de naissance avec JAWS car ceux-ci ont logiquement plus d'expérience avec les technologies d'adaptation. Mais cette tendance est plus étrange avec TactoWeb et nous pensions que les personnes non-voyantes tardives auraient plus de facilité avec l'exploration spatiale. Concernant les analyses effectuées dans chaque type de tâche, aucun résultat significatif n'est apparu, à part celui discuté dans la section 5.3.1.3.

5.3.1.2 Analyse des tâches de recherche d'information

Nous avons effectué un test-t pour échantillons appariés sur toutes les tâches de recherche d'information (S1, S2, S3 et S4) en fonction de l'outil utilisé. La Figure 5-2 nous montre les différences entre les moyennes des taux de succès dans la tâche, des niveaux de difficulté rencontrée et des temps d'exécution des tâches en fonction de cette variable indépendante (outil).

Les moyennes des taux de succès (a) sont maximales et identiques pour les tâches effectuées dans des sites accessibles et il y a un léger avantage pour JAWS dans les sites non accessibles. En ce qui concerne le niveau de difficulté, on observe qu'il est légèrement plus élevé avec TactoWeb

pour les sites accessibles, et beaucoup plus élevé avec JAWS pour les sites non-accessibles. Nous verrons ci-dessous si ces différences sont significatives. En ce qui concerne le temps d'exécution des tâches, JAWS est clairement plus rapide. L'écart entre les moyennes des temps d'exécution est aux alentours des 200 secondes pour les sites accessibles et pour les sites non-accessibles.

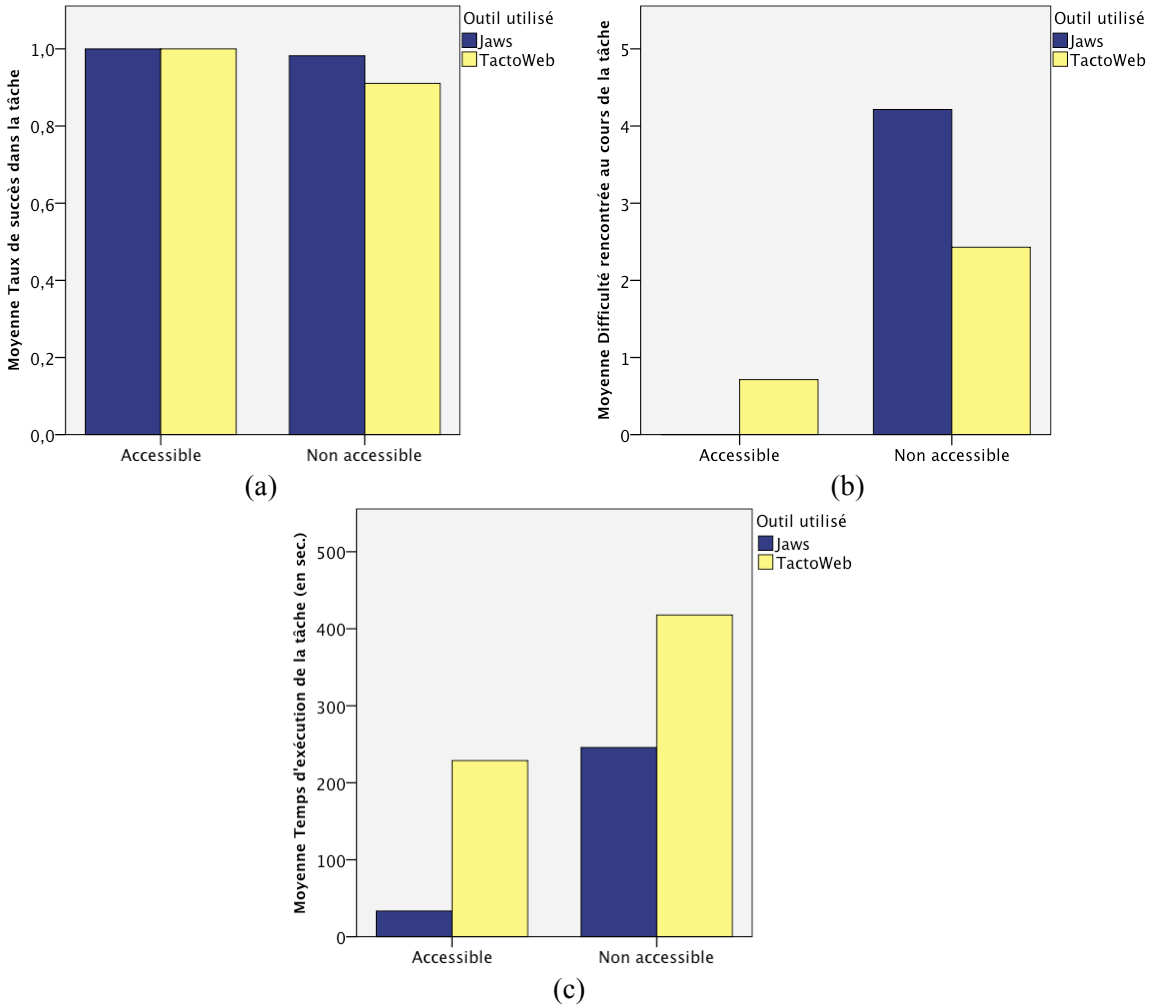


Figure 5-2 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b), et des temps d'exécution (c) pour les tâches de recherche d'information en fonction de l'outil et de l'accessibilité

L'analyse statistique (Tableau 5-8) révèle que la différence entre les temps d'exécution est significative (p = 0,000) lorsqu'on considère la variable indépendante « outil». Il n'y a pas de différence significative pour les taux de succès et les difficultés rencontrées. Une analyse plus spécifique sur les sous-ensembles de tâches en fonction de l'accessibilité nous en dira plus.

Tableau 5-8 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution dans les tâches de recherche d'information en fonction de l'outil

Recherche information	Moyenne			Test-t sur échantillons appariés		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	**0,991**	0,955	0,036	1,342	27,000	0,191
Difficulté rencontrée	2,107	**1,571**	0,536	-0,617	27,000	0,542
Temps d'exécution	**139,628**	323,350	-183,722	-5,872	27,000	**0,000**

À la lumière à cette analyse, on peut affirmer que JAWS est plus rapide que TactoWeb pour les tâches de recherche d'information. Ces tâches contiennent bien sûr de l'exploration de pages Web, mais aussi de la navigation dans le site Web. Or JAWS dispose de nombreux raccourcis clavier pour accélérer la navigation. Par exemple, des raccourcis comme « Insert + F7 » permettent d'afficher la liste des liens de la page Web et de naviguer rapidement à l'intérieur de celle-ci. C'est une observation que nous avons pu faire tout au long de l'expérimentation.

Néanmoins, le fait qu'il n'y ait pas de différence significative entre les taux de succès et entre les niveaux de difficulté rencontrée renforce le constat fait dans la section 5.3.1.1, à savoir que TactoWeb a la même efficacité que JAWS, avec un temps d'apprentissage beaucoup plus court.

Tâches de recherche d'information dans les sites accessibles

Si on se concentre sur les tâches de recherche d'information dans les sites accessibles (Figure 5-3), on se rend compte que les moyennes des taux de succès (a) sont égales et optimales pour les deux outils. Ce n'est pas le cas pour l'efficience où le niveau de difficulté rencontrée (b) est légèrement plus haut pour TactoWeb, bien que très faible, et le temps d'exécution (c) beaucoup faible pour JAWS.

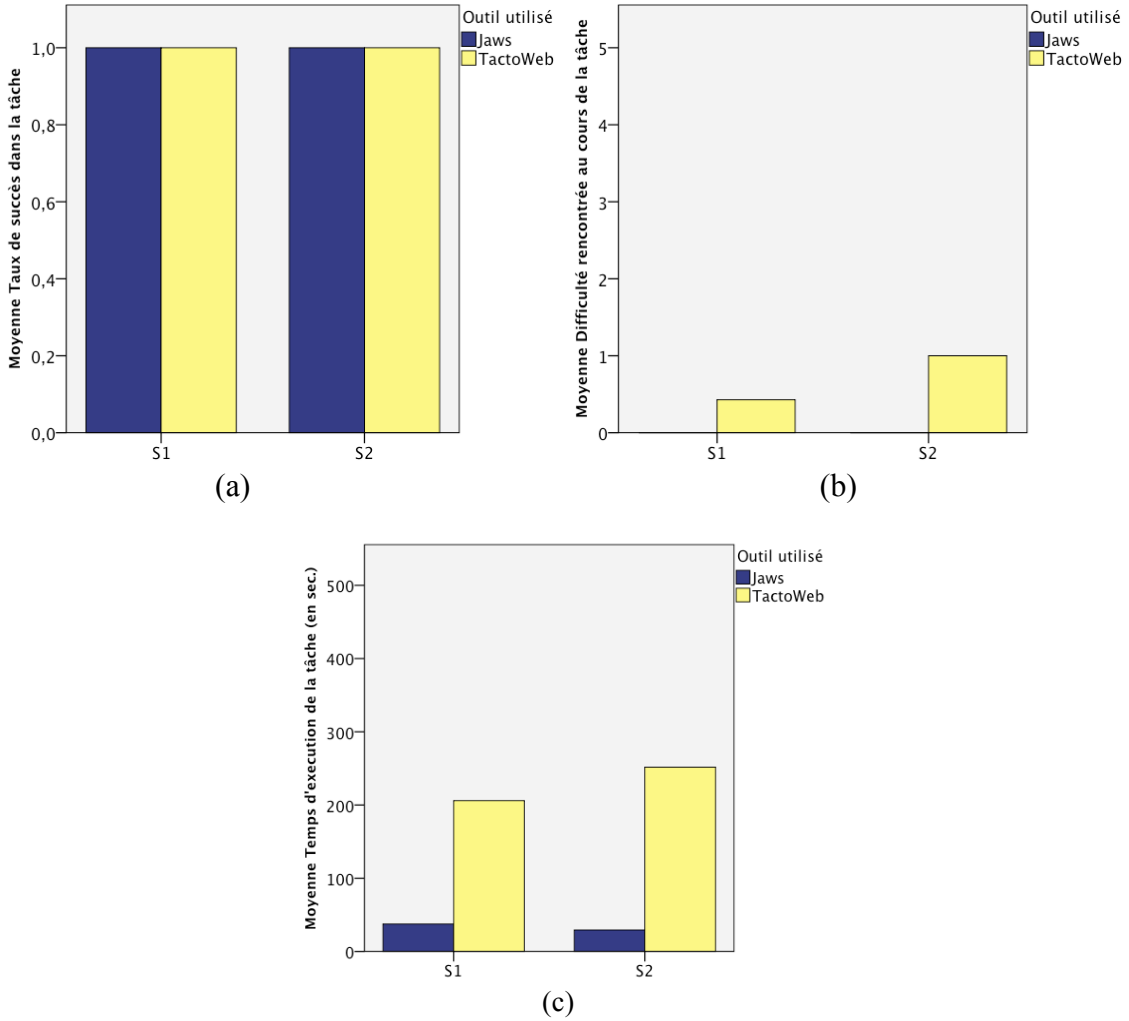


Figure 5-3 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b) et des temps d'exécution (c) pour les tâches de recherche d'information dans des sites accessibles en fonction de l'outil

Si on regarde sous l'angle statistique ce sous-ensemble de tâches en effectuant un test-t pour échantillons appariés (Tableau 5-9), on trouve une seule différence significative pour le temps d'exécution (p = 0,000) avec un avantage pour JAWS. Nous avons presque une différence significative pour le niveau de difficulté (p = 0.070). Néanmoins, l'écart entre les moyennes des niveaux de difficulté n'est que de 0,71 point à l'avantage de JAWS, soit l'équivalent de moins d'une page visitée qui en plus faisait partie du chemin pour trouver l'information, par rapport à JAWS. De plus, la moyenne des taux de succès est de 100% pour les deux outils, ce qui montre que tous les participants ont aussi trouvé l'information demandée avec TactoWeb. JAWS et TactoWeb sont donc aussi efficaces l'un que l'autre pour les tâches de recherche d'information

dans les sites accessibles. En ce qui concerne les temps d'exécution, la différence moyenne est de 195 secondes de moins pour JAWS. Si on ajoute à cela la différence presque significative entre les niveaux de difficulté, il semble que JAWS soit plus efficient que TactoWeb dans la recherche d'information dans des sites accessibles.

Tableau 5-9 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches de recherche d'information dans les sites accessibles en fonction de l'outil utilisé

Recherche information (accessible)	Moyenne			Test-t sur échantillons appariés		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	**1,000**	**1,000**	0,000	-	-	-
Difficulté rencontrée	**0,000**	0,714	-0,714	-1,972	13,000	0,070
Temps d'exécution	**33,440**	228,800	-195,360	-11,169	13,000	**0,000**

Mais si on regarde les statistiques de chacune des deux tâches de recherche d'information S1 et S2 dans des sites accessibles, on se rend compte qu'il n'y a pas de différence significative (p = 0,078) pour le niveau de difficulté de la tâche S1 (Tableau 5-10) et qu'il y a une différence significative (p = 0,018) pour le niveau de difficulté de la tâche S2 (Tableau 5-11). C'est aussi pour la tâche S2 que l'écart entre les temps d'exécution de la tâche est le plus important. Néanmoins, les moyennes des niveaux de difficulté et de temps d'exécution sont plus grandes pour TactoWeb dans les deux cas. Cela confirme le fait que JAWS semble plus efficient pour naviguer dans les sites accessibles que TactoWeb.

Tableau 5-10 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour la tâche de recherche d'information S1 en fonction de l'outil utilisé

Tâche S1	Moyenne			Test-t sur échantillons indépendants		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	**1,000**	**1,000**	0,000	-	-	-
Difficulté rencontrée	**0,000**	0,429	-0,429	-2,121	6,000	0,078
Temps d'exécution	**37,489**	206,011	-168,522	-6,067	8,116	**0,000**

Tableau 5-11 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour la tâche de recherche d'information S2 en fonction de l'outil utilisé

Tâche S2	Moyenne			Test-t sur échantillons indépendants		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	**1,000**	**1,000**	0,000	-	-	-
Difficulté rencontrée	**0,000**	1,000	-1,000	-3,240	6,000	**0,018**
Temps d'exécution	**29,391**	251,589	-222,199	-6,095	12,000	**0,000**

La différence significative entre les niveaux de difficulté rencontrée pour S2 peut s'expliquer par le fait que les pages Web du site de la BANQ possèdent deux menus principaux dont un horizontal. En effet, les participants ont eu quelques difficultés à naviguer horizontalement dans ce menu. Il est possible que ne pas avoir eu d'exemple de menu horizontal dans le tutoriel ait aussi joué, même s'il avait été indiqué qu'on pouvait rencontrer de tels menus. La moyenne des niveaux de difficulté pour S2 avec TactoWeb reste tout de même faible avec une valeur de 1,00 point. Les participants n'ont pas eu ce problème avec JAWS grâce à la linéarisation de l'information. On peut présumer qu'avec plus de pratique, l'utilisateur non-voyant sera plus efficace pour explorer des menus horizontaux.

L'absence de différence significative entre les niveaux de difficulté rencontrés pour S1 est aussi intéressante. Le site du gouvernement du Québec a une organisation classique avec un seul menu principal vertical. Il est donc bon signe que TactoWeb soit aussi facile pour ce genre de site.

Tâches de recherche d'information dans les sites non-accessibles

Si on regarde maintenant les tâches de recherche d'information dans les sites Web non-accessibles (Figure 5-4), on se rend compte qu'encore une fois, l'efficacité des deux outils (a) (taux de succès) semble quasiment équivalente, avec un léger avantage pour JAWS dans la tâche S3. Il plus compliqué de juger l'efficience car le niveau de difficulté rencontré (b) semble égal pour la tâche S3 mais bien moindre pour TactoWeb dans la tâche S4, alors que le temps d'exécution (c) semble encore pour court pour JAWS, avec un écart un peu moins important dans la tâche S4.

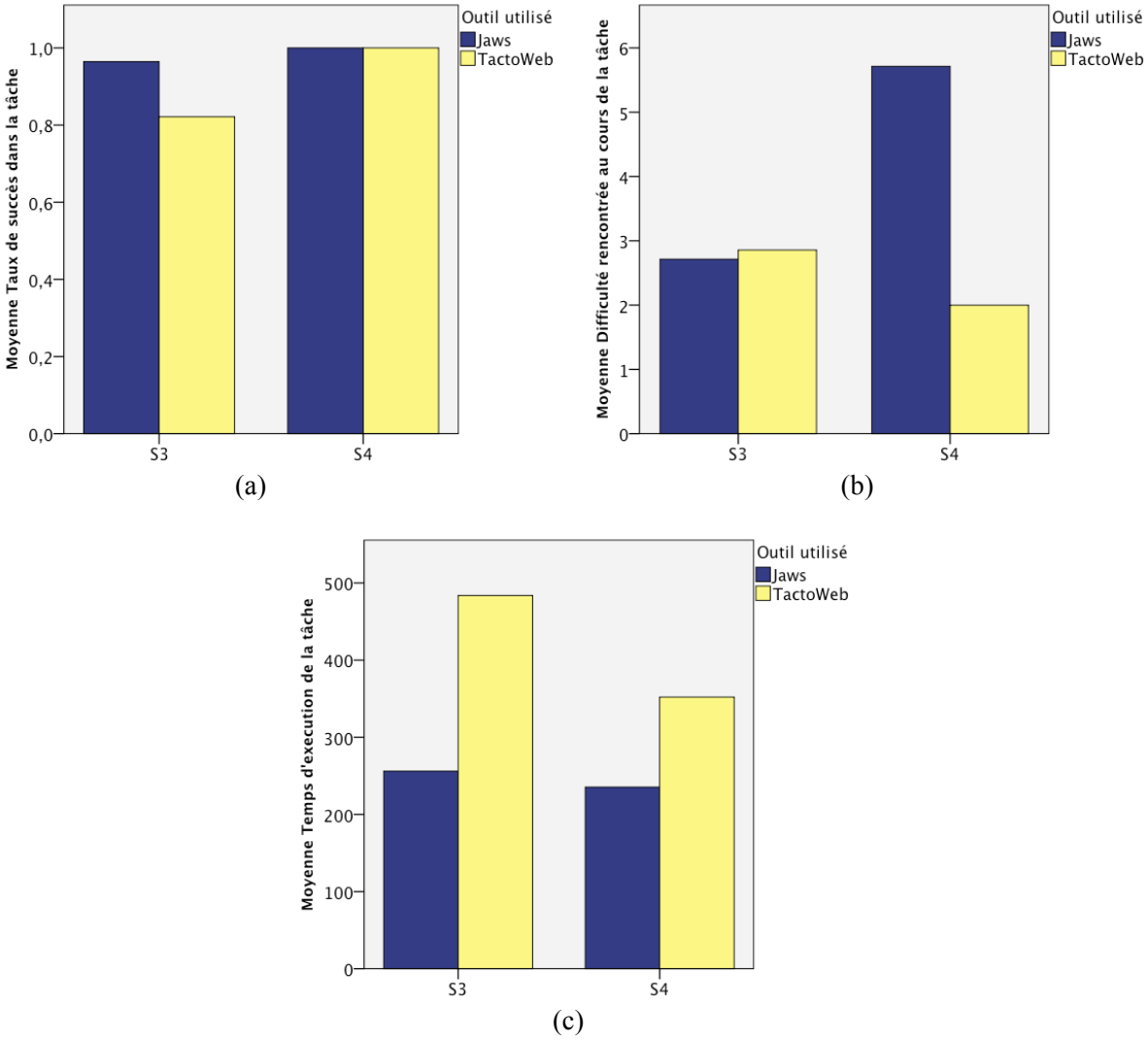


Figure 5-4 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b) et des temps d'exécution (c) pour les tâches de recherche d'information dans des sites non-accessibles en fonction de l'outil

En regardant statistiquement ce sous-ensemble de tâches (Tableau 5-12), il n'y a aucune différence significative pour nos trois variables dépendantes. Nous avons néanmoins deux différences proches du seuil de signification. La première (p = 0,085) porte sur les temps d'exécution de la tâche avec une différence moyenne de 172 secondes en moins pour JAWS. La seconde (p = 0,073) porte sur les taux de succès dans la tâche qui restent très similaires avec une différence de moyennes de seulement 0,071. Mais contrairement aux tâches de recherche d'information dans des sites accessibles, le niveau de difficulté de ces tâches effectuées dans des sites non-accessibles semble inférieur pour TactoWeb avec une différence moyenne de 1,79

points, cependant la différence entre les moyennes est loin d'être significative (p = 0,884). Nous aurions donc une performance équivalente entre les deux outils pour les tâches de recherche d'information effectuées dans les sites non-accessibles.

Tableau 5-12 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches de recherche d'information dans des sites non-accessibles en fonction de l'outil utilisé

Recherche information (non accessible)	Moyenne			Test-t sur échantillons appariés		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	**0,982**	0,911	0,071	1,840	13,000	0,089
Difficulté rencontrée	4,214	**2,429**	1,786	-0,149	13,000	0,884
Temps d'exécution	**245,816**	417,899	-172,083	-1,902	13,000	0,080

Pour la tâche de recherche d'information S3 (Tableau 5-13), il y a une différence significative (p = 0,031) entre les moyennes des taux de succès dans la tâche avec un avantage pour JAWS. Il y a une différence significative entre les moyennes des temps d'exécution (p = 0,037) et une différence non significative entre les moyennes des niveaux de difficulté (p = 0,925).

Tableau 5-13 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour la tâche de recherche d'information S3 en fonction de l'outil utilisé

Tâche S3	Moyenne			Test-t sur échantillons indépendants		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	**0,964**	0,821	0,143	2,449	12,000	**0,031**
Difficulté rencontrée	**2,714**	2,857	-0,143	-0,096	12,000	0,925
Temps d'exécution	**256,151**	483,762	-227,611	-2,519	7,806	**0,037**

Le taux de succès dans cette tâche est calculé en fonction du prix annoncé par le participant, avec une perte de 25% par décalage d'une ligne dans le tableau de prix entre le prix correct et celui annoncé. Les participants n'ayant pas indiqué le bon prix (64$) ont tous indiqué 116$, soit le prix de la ligne du dessus dans la même colonne. Cinq participants sur sept ont donné le mauvais prix avec TactoWeb contre un participant avec JAWS. Cela démontre qu'il y a des améliorations à apporter dans l'exploration des tableaux dans TactoWeb. Tout comme pour la tâche de recherche d'information S2, ce sont les mouvements horizontaux qui semblent poser problème avec TactoWeb car c'est au niveau des lignes du tableau que les participants ont fait l'erreur.

Cela démontre aussi que même si un tableau simple n'est pas accessible avec JAWS, les utilisateurs non-voyants arrivent tout de même à se débrouiller parce qu'ils ont de l'expérience. Néanmoins, les participants ont passé beaucoup du temps à essayer de comprendre un tableau aussi simple. Il n'y a d'ailleurs pas de différence significative (p = 0,522) entre les moyennes du temps passé à explorer le tableau avec JAWS (123 secondes) et du temps passé avec TactoWeb (164 secondes). Passer plus de deux minutes à explorer un tableau aussi simple est très problématique car on aurait pu obtenir la même information beaucoup plus rapidement avec un court paragraphe. De plus, seulement un participant sur sept a utilisé les raccourcis clavier de navigation en deux dimensions avec JAWS (voir 2.2.1), ce qui montre qu'il semble difficile de se rappeler des nombreux raccourcis clavier.

Pour la tâche de recherche d'information S4 (Tableau 5-14), il n'y a aucune différence significative entre les deux outils. Il n'y a pas de données statistiques pour le taux de succès car les moyennes sont de 100% pour les deux outils. La différence entre les moyennes des niveaux de difficulté est de 3,71 à l'avantage de TactoWeb mais elle n'est pas significative (p = 0,159). De plus, la tâche S4 est la seule tâche de recherche d'information où il n'y pas de différence significative entre les temps d'exécution et où l'écart entre les moyennes de temps est inférieur à deux minutes.

Tableau 5-14 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour la tâche de recherche d'information S4 en fonction de l'outil utilisé

Tâche S4	Moyenne			Test-t sur échantillons indépendants		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	**1,000**	**1,000**	0,000	-	-	-
Difficulté rencontrée	5,714	**2,000**	3,714	1,500	12,000	0,159
Temps d'exécution	**235,481**	352,035	-116,555	-0,617	12,000	0,549

La tâche de recherche d'information S4 est celle qui demandait le plus de navigation dans le site et qui avait le plus de problème d'accessibilité. Nous avons observé que la navigation à l'intérieur d'un site était plus rapide avec JAWS mais ce n'est pas aussi évident quand un site a autant de problème d'accessibilité que le site du Collège André-Grasset. Le fait que l'ordre logique de lecture ne soit pas respecté dans le code source et que les image-liens du menu principal soient dépourvues de texte de description, réduit l'efficacité de JAWS et avantage TactoWeb car

l'exploration spatiale n'est pas affectée par ce problème d'accessibilité. Les participants ont été obligés d'explorer entièrement les pages Web car les raccourcis clavier de filtrage ne leur étaient d'aucune aide. Cela se confirme par une différence entre les moyennes des niveaux de difficulté de 3,71 points en faveur de TactoWeb, soit deux à trois pages explorées en moins par rapport à JAWS, dans le but de trouver l'information demandée. Cette observation reste à prendre avec précaution car elle n'est pas significative ($p = 0,159$). JAWS resterait plus efficient pour la navigation mais cela ne serait pas aussi clair dans les sites non-accessibles qui demandent plus d'exploration.

5.3.1.3 Analyse des tâches de remplissage de formulaire

Nous avons ensuite effectué un test-t pour échantillons appariés sur toutes les tâches de remplissage de formulaire (Q1, Q2, Q3 et Q4) avec les variables indépendantes suivantes : outil utilisé et accessibilité de la tâche avec JAWS. La Figure 5-5 montre les moyennes des taux de succès (a), des niveaux de difficultés (b) et des temps d'exécution (c) pour ces des tâches en fonction des deux variables indépendantes mentionnées ci-dessus.

Une observation rapide nous indique que, contrairement aux tâches de recherche d'information, certains résultats sont à l'avantage de TactoWeb. Les taux de succès pour TactoWeb sont légèrement supérieurs à ceux obtenus avec JAWS, avec un écart encore plus important pour les tâches effectuées dans des sites non-accessibles. Il est en de même avec les niveaux de difficulté rencontrée dans la tâche. De plus, il est important de noter que quatre participants qui utilisaient JAWS ont abandonné une tâche dans les sites non-accessible. Ces abandons sont pris en compte dans les résultats. Il n'y a eu aucun abandon avec TactoWeb. Néanmoins, JAWS semble toujours plus rapide que TactoWeb, tout particulièrement pour les tâches effectuées dans les sites accessibles.

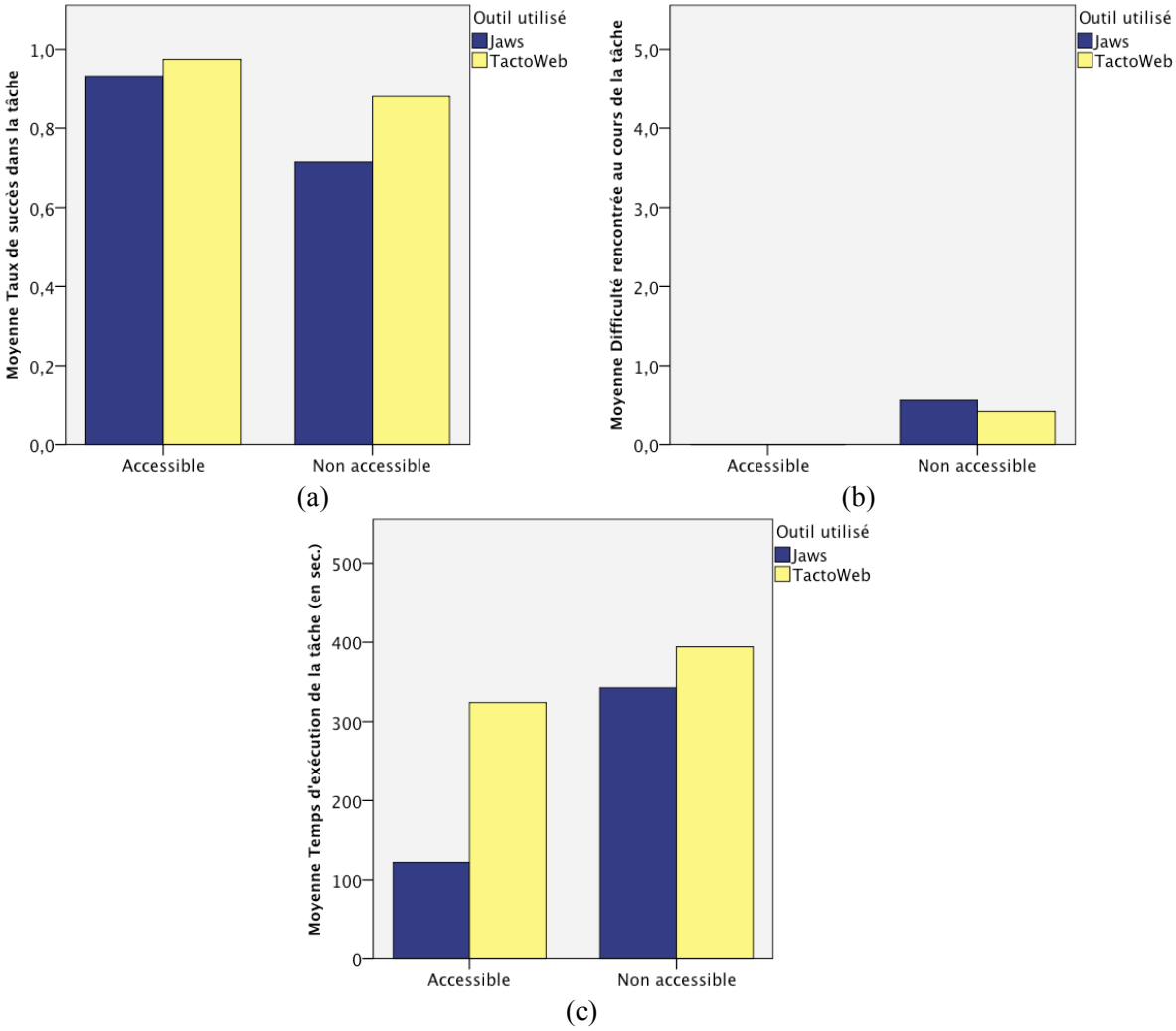


Figure 5-5 : Histogrammes présentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b) et des temps d'exécution (c) pour les tâches de remplissage de formulaire en fonction de l'outil et de l'accessibilité

Sur le plan statistique (Tableau 5-15), la différence entre les moyennes des niveaux de difficulté est significative (p = 0,012). L'écart est de 0,074 en faveur de TactoWeb donc très minime. L'autre différence significative (p = 0,001) concerne encore le temps d'exécution avec un écart de 124 secondes à l'avantage de JAWS. Nous avons aussi une différence entre les moyennes des taux de succès de 0,106 à l'avantage de TactoWeb mais celle-ci n'est pas significative (p = 0,367). L'analyse plus spécifique qui portera sur les sous-ensembles de tâches en fonction de l'accessibilité nous en dira plus.

Tableau 5-15 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches de remplissage de formulaire en fonction de l'outil et de l'accessibilité

Remplissage formulaire	Moyenne			Test-t sur échantillons appariés		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	0,819	**0,925**	-0,106	-0,917	27,000	0,367
Difficulté rencontrée	0,296	**0,222**	0,074	2,679	27,000	**0,012**
Temps d'exécution	**236,381**	360,422	-124,042	-3,787	25,000	**0,001**

Malgré une différence non-significative, il semblerait que TactoWeb ait globalement une meilleure efficacité que JAWS pour l'exploration des formulaires. L'écart de 0,106 (10,6%) entre les moyennes des taux de succès correspond à peu près à un champ de formulaire mal rempli ou oublié en plus dans JAWS. Cet écart peut s'expliquer par le fait que les tâches sur les formulaires n'impliquent que de l'exploration, et non pas de l'exploration et de la navigation comme c'est le cas pour les tâches de recherche d'information. Néanmoins, nous ne pouvons affirmer assurément que TactoWeb ait une meilleure efficacité que JAWS à cause de la différence non-significative.

En ce qui concerne l'efficience, même si le niveau de difficulté semble légèrement plus élevé dans JAWS, le temps d'exécution est toujours plus long avec TactoWeb. Mais l'écart entre les moyennes des temps d'exécution avec les deux outils est de 124 secondes alors qu'il était de 184 secondes pour les tâches de recherche d'information, soit une différence d'une minute. L'avantage de JAWS pour effectuer rapidement les tâches semble donc moins important pour les tâches de remplissage de formulaires.

Les tâches de remplissage de formulaire ont aussi mis en relief la seule différence significative en fonction de la période d'apparition de la cécité chez les participants. En effet, si on se concentre sur ce type de tâche effectué avec TactoWeb, il y a une différence significatives ($p = 0,022$) entre les moyennes des taux de succès, à l'avantage des participants non-voyants de naissance. Cette différence est de plus assez importante (0,103) et correspond à un champs de formulaire oublié ou mal remplis.

Tableau 5-16 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution des tâches de remplissage de formulaire en fonction de l'apparition de la cécité chez les participants utilisant TactoWeb

Formulaires remplis avec TACTOWEB	Moyenne			Test-t sur échantillons indépendants		
	Naissance	Tardif	Ecart (N-T)	t	df	valeur-p
Taux de succès	**0,968**	0,865	0,103	2,591	13,742	**0,022**
Difficulté rencontrée	0,250	**0,170**	0,080	0,431	26,000	0,670
Temps d'exécution	**302,397**	444,824	-142,427	-1,847	25,000	0,077

Cette différence entre les taux de succès est inattendue. Le remplissage d'un formulaire avec TactoWeb demande une bonne maîtrise de l'exploration spatiale. Les personnes non-voyantes de naissance auraient moins de problème que les personnes non-voyantes tardives pour explorer spatialement alors que ces derniers ont pourtant déjà vu dans leur passé. On peut expliquer cette observation par le fait que les personnes non-voyantes de naissances sont plus familières avec les technologies d'adaptation et se sont donc peut-être plus rapidement habituées à l'exploration spatiale avec le Tactograph.

Tâches de remplissage de formulaire dans les sites accessibles avec JAWS

Si on se focalise sur les tâches de remplissage de formulaire effectuées dans des sites accessibles (Figure 5-6), l'efficacité mesurée par l'intermédiaire des moyennes des taux de succès (a) semble encore similaire pour les deux outils, avec un léger avantage pour TactoWeb dans la tâche Q2. En ce qui concerne l'efficience, le niveau de difficulté rencontré (b) est nul pour le deux outils et JAWS semble avoir toujours un temps d'exécution (c) plus rapide que TactoWeb.

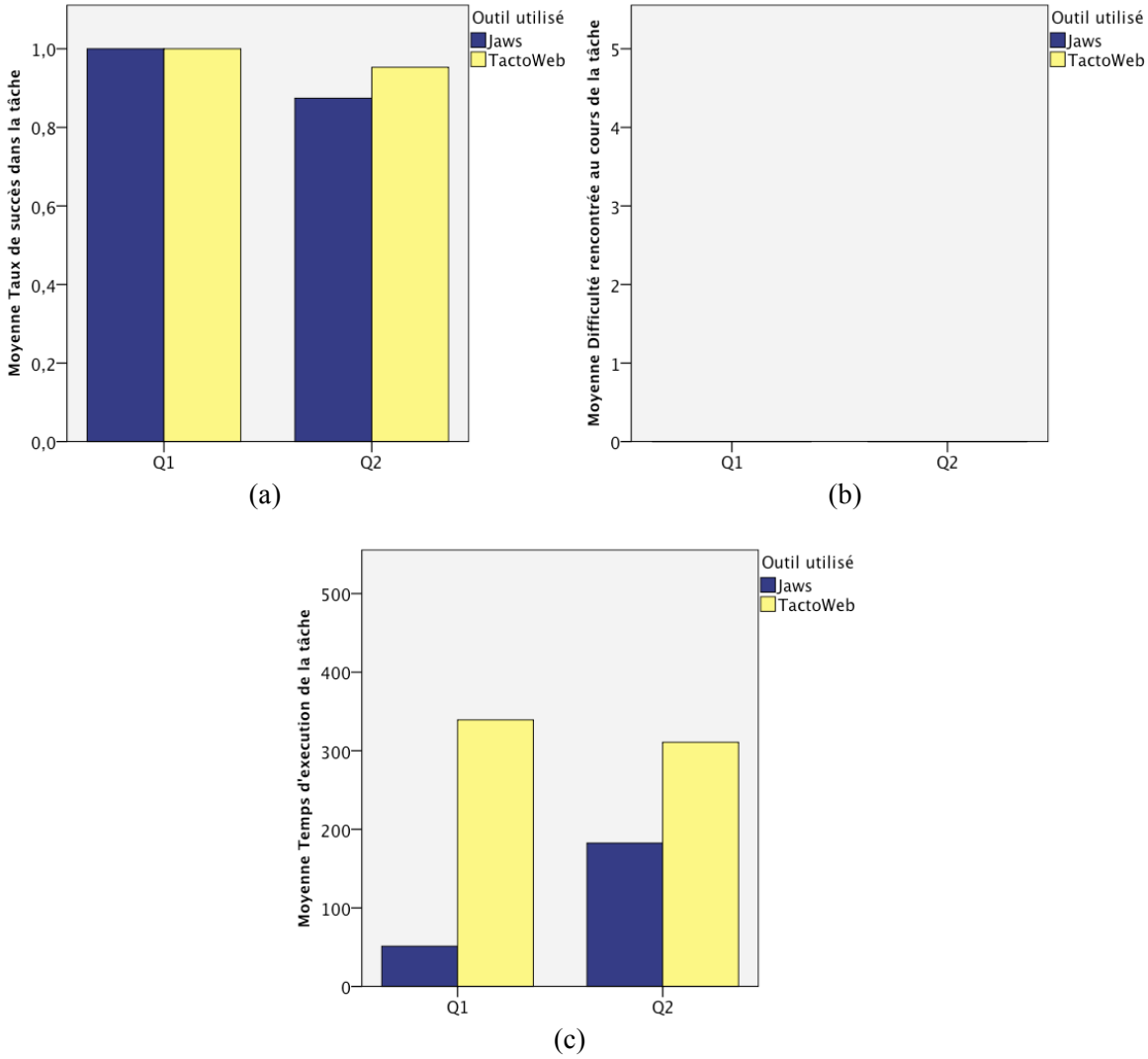


Figure 5-6 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b) et des temps d'exécution (c) pour les tâches de remplissage de formulaire dans des sites accessibles en fonction de l'outil

Si on regarde maintenant ce sous-ensemble de tâche de manière statistique (Tableau 5-17), la seule différence significative (p = 0,000) concerne les temps d'exécution de la tâche avec un écart entre les différences moyennes de 202 secondes en faveur de JAWS. L'écart entre les moyennes des taux de succès est minime (0,043 point) avec un léger avantage pour TactoWeb sans être significatif (p = 0,374). Il n'y a pas de données statistiques sur les niveaux de difficulté car les moyennes sont de 0 pour les deux outils dans les sites accessibles.

Tableau 5-17 : Test-t pour échantillons appariés sur les moyennes des taux de succès et des temps d'exécution pour des tâches de remplissage de formulaires dans des sites accessibles en fonction de l'outil utilisé

Remplissage formulaire (accessible)	Moyenne			Test-t sur échantillons appariés		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	0,932	**0,974**	-0,043	-0,920	13,000	0,374
Difficulté rencontrée	**0,000**	**0,000**	0,000	-	-	-
Temps d'exécution	**121,880**	323,954	-202,075	-5,097	11,000	**0,000**

On peut affirmer que l'efficacité de JAWS et TactoWeb est similaire dans les sites accessibles. JAWS garde tout de même son avantage pour ce qui est de la rapidité. Cela est confirmé si on regarde plus précisément les tâches Q1 (Tableau 5-18) et Q2 (Tableau 5-19). Les différences entre les moyennes des temps d'exécution des deux tâches sont significatives.

Tableau 5-18 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution de la tâche de remplissage de formulaire Q1 en fonction de l'outil utilisé

Tâche Q1	Moyenne			Test-t sur échantillons indépendants		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	**1,000**	**1,000**	0,000	-	-	-
Difficulté rencontrée	**0,000**	**0,000**	0,000	-	-	-
Temps d'exécution	**51,245**	339,249	-288,004	-5,991	10,000	**0,000**

Tableau 5-19 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution de la tâche de remplissage de formulaire Q2 en fonction de l'outil utilisé

Tâche Q2	Moyenne			Test-t sur échantillons indépendants		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	0,873	**0,952**	-0,079	-1,213	12,000	0,249
Difficulté rencontrée	**0,000**	0,000	0,000	-	-	-
Temps d'exécution	**182,424**	310,845	-128,421	-2,691	12,000	**0,020**

Il y a une forte différence entre les temps d'exécution de ces deux tâches : 288 secondes pour Q1 et 128 secondes pour Q2, soit moins de la moitié. Cela peut s'expliquer en partie par le fait que la

tâche Q1 comporte moins de champs que la tâche Q2 et qu'il est possible de trouver les champs d'un formulaire avec une simple touche de clavier dans JAWS.

Si on regarde de plus près, on voit que l'écart entre les moyennes porte surtout sur le temps passé à chercher le premier champ dans les deux tâches. Ce temps est indépendant de la complexité du formulaire. L'écart entre les moyennes de temps à rechercher le premier champ avec les deux outils est de 190 secondes pour Q1 et la différence est significative ($p = 0,015$) alors qu'il est de 116 secondes pour Q2 et la différence est non significative mais très proche du seuil ($p = 0,051$). Cet écart de 116 secondes pour Q2 correspond presque entièrement à l'écart pour le temps d'exécution total (116 sur 128 secondes) alors que cela ne correspond qu'aux deux tiers du temps d'exécution pour Q1 (190 sur 288 secondes).

Cela signifie qu'il y a un écart de 98 secondes entre les moyennes des temps d'exécution pour remplir le formulaire de Q1. Cet écart entre les moyennes vient du dernier champ du formulaire. En effet, le dernier champ correspond à une saisie de texte sur plusieurs lignes. Cela ne fait aucune différence dans JAWS mais dans TactoWeb, le fait que le champ soit sur plusieurs lignes a bien sûr augmenté la hauteur de sa taille. Les participants étaient habitués à explorer des champs n'ayant qu'une ligne de hauteur et cela les a ralentis même si, à la fin de la tâche, ils se sont rendus compte que le champ prenait plusieurs lignes. Le manque d'expérience avec TactoWeb a donc été plus handicapant pour la tâche Q1. Enfin, il faut noter qu'un participant a abandonné la tâche Q2 avec JAWS à cause du problème d'accessibilité de la langue de la synthèse vocale par défaut.

Tâches de remplissage de formulaire dans les sites non-accessibles avec JAWS

Concernant les tâches de remplissage de formulaire dans les sites non-accessibles (Figure 5-7), TactoWeb semble avoir une plus grande efficacité (taux de succès) (a) que JAWS, tout particulièrement dans la tâche Q4. Pour ce même sous-ensemble de tâches, l'efficience semble égale mais avec des résultats complètement différents en fonction de la tâche observée. Même si les moyennes des niveaux de difficulté rencontrés (b) sont faibles, le niveau de difficulté est plus faible pour JAWS dans la tâche Q3 alors que le niveau de difficulté avec TactoWeb est plus faible pour la tâche Q4. Il est en de même pour le temps d'exécution (c) qui est bien plus court pour JAWS dans la tâche Q3, alors qu'il est bien plus court pour TactoWeb dans la tâche Q4.

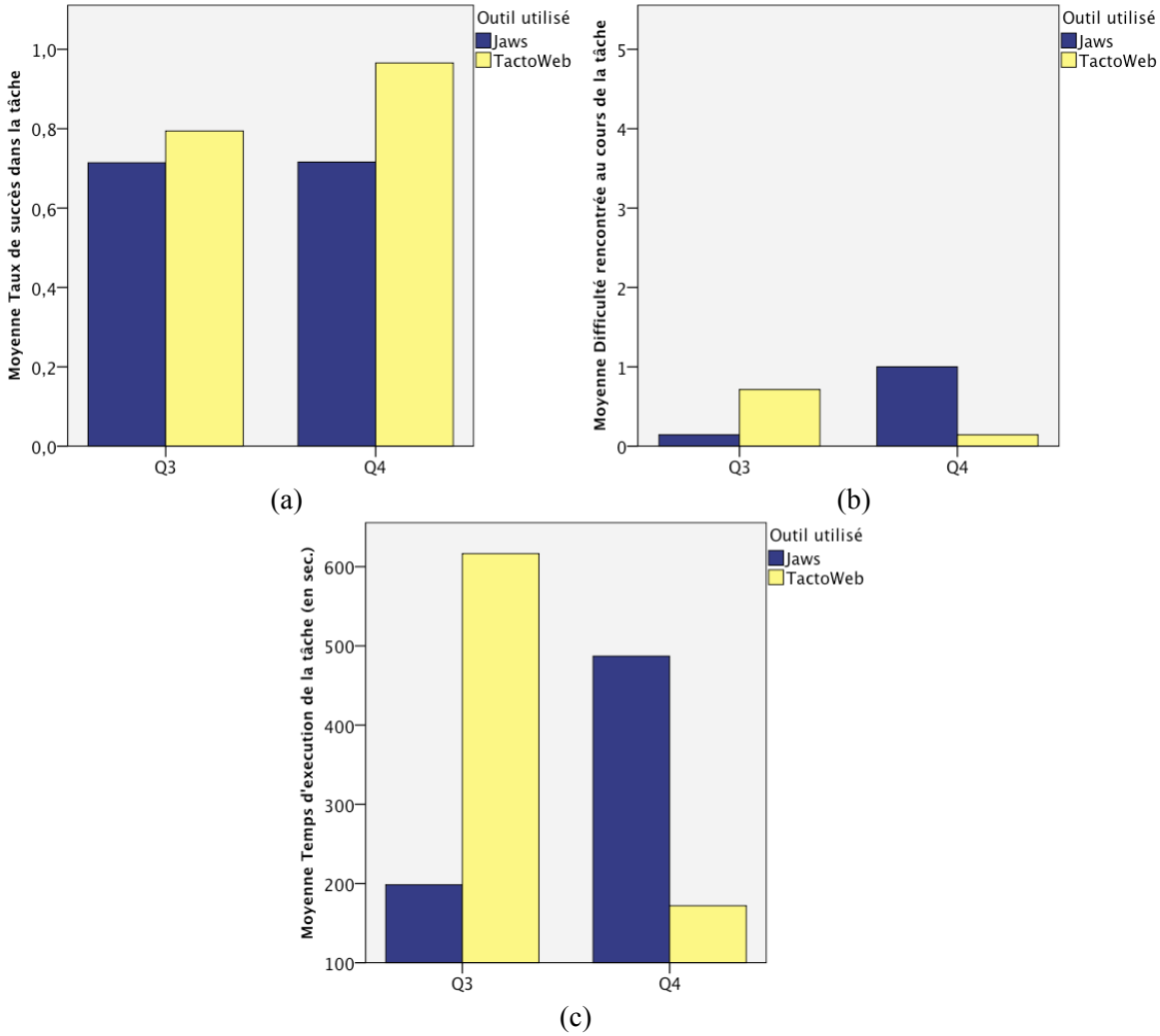


Figure 5-7 : Histogrammes représentant les moyennes des taux de succès dans la tâche (a), des niveaux de difficulté rencontrés (b) et des temps d'exécution (c) pour les tâches de remplissage de formulaire dans des sites non-accessibles en fonction de l'outil

Dans les tâches de ce sous-ensemble (Tableau 5-20), il n'y a aucune différence significative entre les moyennes des variables dépendantes. Cependant certaines différences entre moyennes méritent notre attention. La différence entre les moyennes des taux de succès dans les tâches effectuées dans les sites non accessibles est presque quatre fois plus grande que dans les tâches effectuées dans les sites accessibles, toujours à l'avantage de TactoWeb, sans être néanmoins proche du seuil de signification (0,562). Par contre, la différence entre les moyennes de temps d'exécution est seulement de 52 secondes et n'est pas non plus significative (p = 0,270).

Malgré une différence importante entre les deux outils pour le remplissage de formulaire dans des sites non accessibles, on ne peut pas déduire que TactoWeb est significativement plus efficace

que JAWS pour remplir des formulaires dans de tels sites (p = 0, 562). De plus, remplir des formulaires dans des sites non-accessibles correspond au seul ensemble de sous-tâches où la différence entre les moyennes de temps d'exécution est inférieure à une minute (52 secondes), tout en étant non-significative. La grande différence de temps d'apprentissage entre les deux outils ne joue donc pas en faveur de JAWS dans ce cas précis, ce qui est un avantage conséquent pour TactoWeb.

Tableau 5-20 : Test-t pour échantillons appariés sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour les tâches de remplissage de formulaires dans des sites non-accessibles en fonction de l'outil

Remplissage formulaire (non accessible)	Moyenne			Test-t sur échantillons appariés		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	0,714	**0,878**	-0,164	-0,596	13,000	0,562
Difficulté rencontrée	0,571	**0,429**	0,143	0,907	13,000	0,381
Temps d'exécution	**342,704**	394,286	-51,582	-1,151	13,000	0,270

Les tâches de remplissage de formulaire Q3 (Tableau 5-21) et Q4 (Tableau 5-22) ont donné des résultats quasiment opposés. Le seul point commun entre les deux tâches est que TactoWeb obtient un meilleur taux de succès, même si c'est beaucoup plus prononcé dans Q4, avec un écart assez important de 0,25 qui est presque significatif (p = 0,067), que dans Q3 avec un écart de seulement 0,08 qui n'est pas du tout significatif (p = 0,598). Là où les deux tâches s'opposent, c'est pour le niveau de difficulté et les temps d'exécution des tâches. Pour ces deux variables dépendantes, l'avantage est pour JAWS dans Q3, surtout pour les temps d'exécution avec un écart de 418 secondes qui est significatif (p = 0,000). C'est le contraire pour TactoWeb dans Q4, avec un écart de 0,86 point entre les niveaux de difficulté et surtout de 315 secondes entre les temps d'exécution. Mais ces deux écarts ne sont pas significatifs.

Tableau 5-21 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution pour la tâche de remplissage de formulaire Q3 en fonction de l'outil

Tâche Q3	Moyenne			Test-t sur échantillons indépendants		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	0,714	**0,792**	-0,078	-0,552	6,874	0,598
Difficulté rencontrée	**0,143**	0,714	-0,571	-1,789	12,000	0,099
Temps d'exécution	**198,327**	616,594	-418,267	-4,881	12,000	**0,000**

Tableau 5-22 : Test-t pour échantillons indépendants sur les moyennes des taux de succès, des niveaux de difficulté et des temps d'exécution de la tâche de remplissage de formulaire Q4 en fonction de l'outil

Tâche Q4	Moyenne			Test-t sur échantillons indépendants		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Taux de succès	0,714	**0,964**	-0,250	-2,202	6,473	0,067
Difficulté rencontrée	1,000	**0,143**	0,857	1,867	7,271	0,103
Temps d'exécution	487,080	**171,977**	315,103	1,456	12,000	0,171

La tâche Q3 a posé les problèmes d'accessibilité attendus avec JAWS. D'ailleurs, deux participants ont abandonné la tâche avec cet outil. Mais la tâche a aussi posé des problèmes avec TactoWeb alors que nous nous y attendions moins. Tout d'abord, le manque d'espace vertical entre les champs rendait plus difficile l'association spatiale entre les champs et leur étiquette respective, à cause de la difficulté à se déplacer tout droit de manière horizontale avec TactoWeb. Il est donc arrivé qu'un participant explore horizontalement une étiquette puis se retrouve sur le champ de la ligne du dessus ou du dessous car il ne se déplaçait pas assez en ligne droite De plus, certaines étiquettes n'étaient pas codées comme telles mais seulement comme de simples textes. La maximisation de la largeur du TactoElement ne se faisait donc pas dans ces cas là. Certains participants sont donc passés à coté de ces étiquettes car ils naviguaient entre elles sur le côté droit de l'ensemble des étiquettes, alors que leur largeur n'était pas égale (voir un exemple dans la Figure 4-10). Il apparaît donc pertinent de garder un espacement vertical minimum entre les champs, ainsi que d'essayer de reconnaître automatiquement dans TactoWeb les étiquettes qui ne sont pas codées comme telles.

Pour la tâche Q4, nous avons obtenu les résultats que nous avions prévus, même ci-ceux ne sont pas statistiquement significatifs. Remplir le formulaire dans un site particulièrement inaccessible a posé de gros problème dans JAWS. D'ailleurs, un participant a abandonné cette tâche lorsqu'il utilisait cet outil. Mais avec TactoWeb, le participant a parfaitement fait son travail. L'association spatiale entre les étiquettes et les champs a pallié aux problèmes de leur non-association dans le code et du non-respect de l'ordre logique de lecture. De plus, c'est la seule tâche parmi les huit où le temps d'exécution moyen fut plus court dans TactoWeb que dans JAWS.

5.3.2 Analyse des questions fermées du questionnaire

Dans le but d'évaluer la satisfaction des participants à l'égard des deux outils, nous avons d'abord analysé la moyenne des scores des participants sur une échelle de Likert de 1 à 5 aux questions fermées de notre questionnaire. Nous nous sommes focalisés en premier lieu sur les 10 questions fermées qui étaient communes aux deux outils, afin de comparer les niveaux de satisfaction. Pour ces 10 questions, plus le score sur l'échelle de un à cinq était élevé, plus la satisfaction envers la caractéristique de l'outil traitée dans la question était élevée, à l'exception de la question 5 où c'était l'inverse. Par souci d'homogénéisation des moyennes, nous avons donc inversé les scores de la question 5 pour que la satisfaction varie dans le même sens que les neuf autres questions. De plus, nous avons calculé la moyenne des moyennes de ces 10 questions afin d'obtenir une satisfaction globale pour les deux outils.

La Figure 5-8 montre les moyennes des réponses aux 10 questions communes à JAWS et TactoWeb. On voit rapidement que JAWS procure une plus grande satisfaction sur tous les aspects, sauf pour donner une idée globale de la page. Néanmoins, les différences de moyennes entre les deux outils sont rarement supérieures à un point. Les différences les plus importantes portent sur le retour sonore où il y a un écart de 1,00 point pour la facilité à comprendre les explications vocales, et un écart de 1,93 pour la qualité de la synthèse vocale. L'autre différence importante à l'avantage de JAWS porte sur la concentration qui est exigée par l'outil avec un écart de 0,93 point. La seule différence supérieure ou proche d'un point à l'avantage de TactoWeb porte sur le fait d'avoir une meilleure idée globale de la page avec un écart de 1,14 point. Enfin, l'écart de satisfaction globale n'est que d'un peu plus d'un demi-point (0,57) en faveur de JAWS.

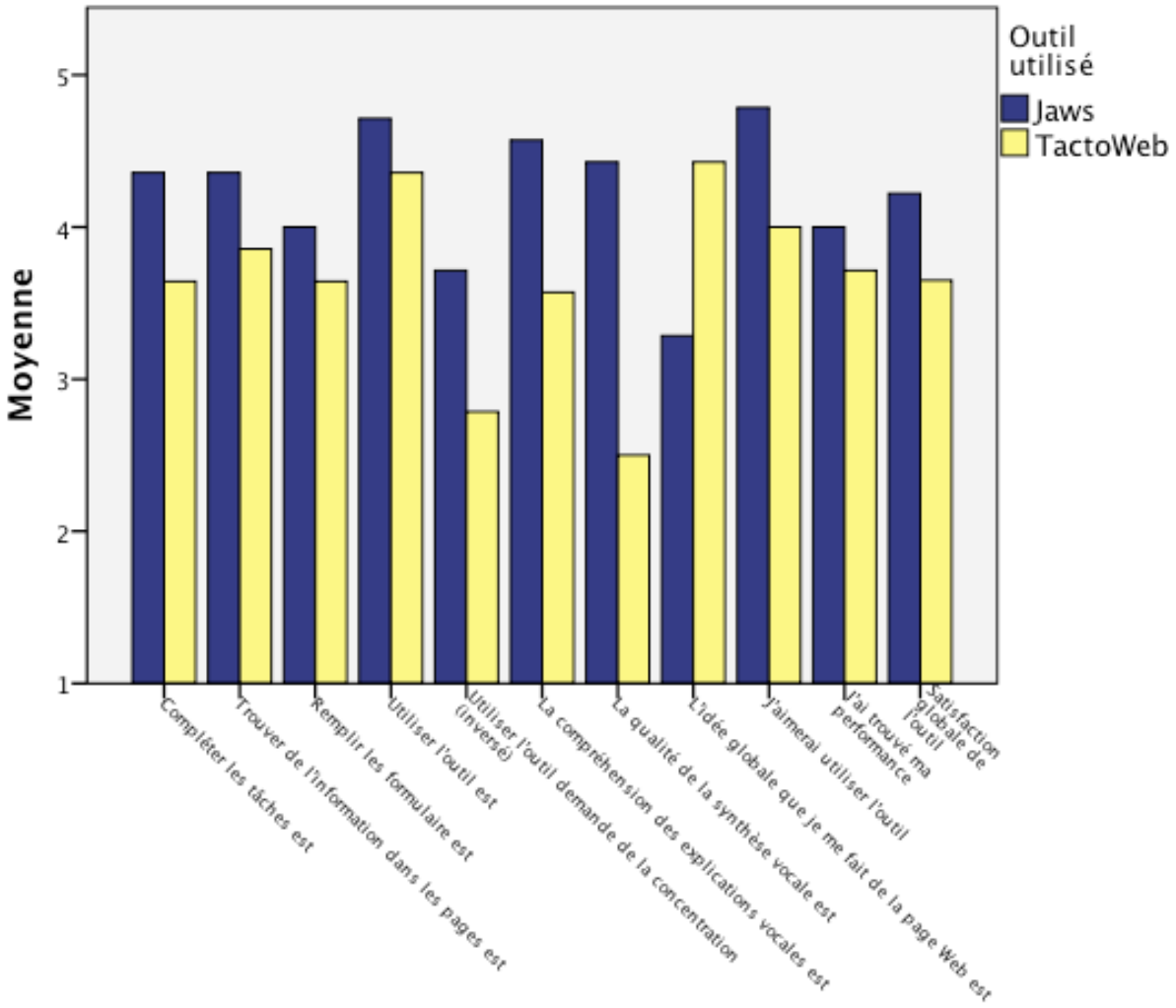


Figure 5-8 : Histogramme présentant les moyennes des réponses aux questions fermées du questionnaire sur la satisfaction à l'égard des deux outils

L'analyse statistique des résultats (Tableau 5-23) révèle que les cinq principales différences que nous avons décrites ci-dessus sont toutes significatives. En effet, les valeur-p sont de 0,016 et 0,000 pour les deux questions traitant le retour sonore; la valeur p est de 0,013 pour la question sur la concentration demandée par l'outil; elle est de 0,005 pour la satisfaction globale; elle est de 0,012 pour l'idée globale de la page Web, la seule question à l'avantage de TactoWeb.

Trois autres différences entre les moyennes sont significatives. La première concerne la facilité de compléter les tâches qui est à l'avantage de JAWS avec un écart entre les moyennes de 0,72 point (p = 0,027). La seconde se rapporte à la facilité à trouver de l'information dans les pages Web avec un écart entre les moyennes de 0,5 point (p = 0,013). La dernière est le souhait

d'utiliser l'outil fréquemment avec un avantage pour JAWS et un écart entre les moyennes de 0,79 point (p = 0,016).

Tableau 5-23 : Test-t (échantillons appariés) sur les moyennes des réponses aux questions fermées du questionnaire portant sur le taux de satisfaction à l'égard de l'outil

Questionnaire	Moyenne			Test-t sur échantillons indépendants		
	JAWS	TactoWeb	Ecart (J-T)	t	df	valeur-p
Complêter les tâches est	**4,360**	3,640	0,720	2,500	13,000	**0,027**
Trouver de l'information dans les pages est	**4,360**	3,860	0,500	2,876	13,000	**0,013**
Remplir les formulaire est	**4,000**	3,640	0,360	1,161	13,000	0,266
Utiliser l'outil est	**4,710**	4,360	0,350	1,439	13,000	0,174
Utiliser l'outil demande de la concentration	**3,710**	2,790	0,920	2,879	13,000	**0,013**
La compréhension des explications vocales est	**4,570**	3,570	1,000	2,754	13,000	**0,016**
La qualité de la synthèse vocale est	**4,430**	2,500	1,930	6,324	13,000	**0,000**
L'idée globale que je me fait de la page Web est	3,290	**4,430**	-1,140	-2,929	13,000	**0,012**
J'aimerai utiliser l'outil	**4,790**	4,000	0,790	2,242	13,000	**0,043**
J'ai trouvé ma performance	**4,000**	3,710	0,290	1,000	13,000	0,336
Satisfaction globale de l'outil	**4,221**	3,650	0,571	3,346	13,000	**0,005**

Les résultats des réponses aux questions fermées du questionnaire indiquent que les participants ont été plus satisfaits par JAWS que par TactoWeb. Mais comme pour les résultats des tâches, on peut expliquer cette différence de niveau de satisfaction par la différence d'expérience avec les deux outils. Même si nous avons demandé aux participants de ne prendre en compte que l'expérience du jour et non pas leur expérience globale avec JAWS, les participants nous ont indiqué qu'il était beaucoup facile d'utiliser JAWS étant donné leurs années d'expérience.

Les participants nous l'ont dit en répondant aux questions fermées, comme par exemple « C'est sûr que moi, je suis habitué à JAWS là » (P04) et « C'est sûr que c'est très facile [avec JAWS] mais si je n'avais pas l'expérience que j'ai avec… » (P14). Il en est de même lorsqu'ils abordent la facilité pour faire les tâches avec TactoWeb : « C'est vrai que c'est la première fois que je l'utilise. Sûrement qu'après une couple de jours, je m'habituerais. Mais là je dirais un bon 4 sur 5 … je suis pas encore habitué complètement mais ça va quand même assez bien » (P01). Un

participant a aussi indiqué qu'il avait plus pris son temps avec TactoWeb car il ne l'avait jamais utilisé : « Comme c'est une interface [TactoWeb] que je connais moins, j'ai pris mon temps pour faire le tour » (P13). Le fait que l'habitude d'utiliser JAWS a plus influencé les réponses est encore plus flagrant avec la question sur le souhait d'utiliser fréquemment l'outil. Les participants ont tous indiqué 5 (sur 5) comme réponse mais plus par obligation : « J'ai pas ben ben l'choix » (P05), « J'ai pas le choix c'est ça… » (P06), et « Ça me prend ça dans la vie » (P14).

JAWS demande aussi moins de concentration que TactoWeb. Encore une fois, cela peut s'expliquer par l'expérience que les participants avaient avec TactoWeb : « C'est sûr qu'il faut être attentif mais moi vu que j'ai l'habitude [avec JAWS], je n'ai pas forcément besoin de me concentrer » (P01). Néanmoins, cet écart de concentration peut aussi venir du fait que JAWS n'utilise qu'une modalité de sortie alors qu'il y en a deux pour TactoWeb. En effet, les participants devaient faire attention aux retours tactiles et sonores en même temps dans le but de correctement utiliser TactoWeb.

Pour résumer, JAWS reste plus facile à utiliser selon les participants dans tous les cas et c'est logique. Tester TactoWeb à plus long terme nous indiquera s'il pourrait mieux performer; certains participants en étaient convaincus : « Puis à l'avenir, je pense que cela serait un bon complément à JAWS » (P01) et « Il faut se faire à l'idée. Mais à force d'explorer l'écran, on s'y fait » (P11).

Certains aspects de JAWS sont préférés à ceux de TactoWeb, peu importe la différence du temps d'apprentissage. Le point le plus critique est la synthèse vocale. C'est là où l'écart entre les moyennes obtenues par les deux outils est le plus important, soit presque deux points. Même s'ils n'étaient pas habitués à la synthèse vocale utilisée dans TactoWeb, les participants avaient l'habitude d'utiliser différentes synthèses et celle-ci fut clairement évaluée comme de faible qualité : « Cette synthèse là, je l'aime pas beaucoup car elle est plus difficile à comprendre » (P01) ou « Ça serait bien si c'était la même synthèse que JAWS » (P03). Il sera essentiel de trouver une meilleure synthèse vocale pour améliorer TactoWeb. Les autres aspects à l'avantage de JAWS seront abordés dans la section 5.3.3.1.

Là où TactoWeb a bien répondu à nos attentes, c'est dans l'idée globale des pages Web que se sont faits les participants. L'outil que nous avons créé permet clairement de donner une image

globale des pages Web et les participants l'ont clairement exprimé : « On voit l'image » (P01), « On va juste naviguer d'un lien à l'autre [avec JAWS] mais on sait pas pantoute comment la page est faite. Alors qu'avec TactoWeb…» (P09), ou « Pour le contenu ça va, mais pour l'image, JAWS c'est pas ça » (P11). Malgré un temps d'apprentissage court, TactoWeb permet d'accéder à la même information que les utilisateurs voyants (voir section 5.3.3.2 pour plus de détails). C'est une grande satisfaction que nous tirons de cette expérimentation.

Si on regarde maintenant les questions spécifiques qui étaient posées à propos de TactoWeb (Figure 5-9), on se rend compte que même si la synthèse vocale n'était pas de bonne qualité, le retour sonore était très utile (moyenne de 4,86), surtout grâce aux différentes audicônes qui furent faciles à distinguer (moyenne de 4,29). De plus, ces audicônes (indices sonores dans la question) n'ont pas étés jugées peu ou trop nombreuses (moyenne de 2,79). Les audicônes sont donc une autre bonne source de satisfaction dans TactoWeb. Elles ont permis de bien discerner les différents types d'élément Web, sans ajouter du texte supplémentaire à lire avec la synthèse vocale.

Les boutons placés sur le module tactile du Tactograph ont été jugés très faciles à utiliser (moyenne de 4,50 sur 5) sans doute à cause du fait qu'il n'y avait qu'une action possible pour chacun d'entre eux. Si dans le futur, nous souhaitons ajouter plusieurs actions par boutons (double-clic ou clic long), il faudra faire attention à ne pas rendre leur utilisation trop complexe. En effet, ajouter plusieurs actions par boutons pourrait les rendre difficiles à apprendre.

La charge mentale demandée par TactoWeb qui est discutée au début de cette section (moyenne de 2,79 sur 5 à la question sur la concentration demandée par l'outil) est confirmée par la question sur la quantité d'information à gérer en même temps avec une moyenne quasiment équivalente (2,71). Cette moyenne reste assez basse malgré le fait que l'utilisateur ait à gérer différents types de retours sonores, le retour tactile et le déplacement du module tactile en même temps.

Enfin, malgré une expérimentation d'au moins une heure avec TactoWeb, la fatigue tactile ne s'est pas trop fait sentir parmi les participants (moyenne de 1,86). Le fait qu'il y des pauses entre les tâches et les différentes étapes du tutoriel a sans doute aidé. Néanmoins, le fait que TactoWeb provoque une trop grande fatigue tactile était une de nos inquiétudes. Cette inquiétude s'est dissipée avec l'expérimentation.

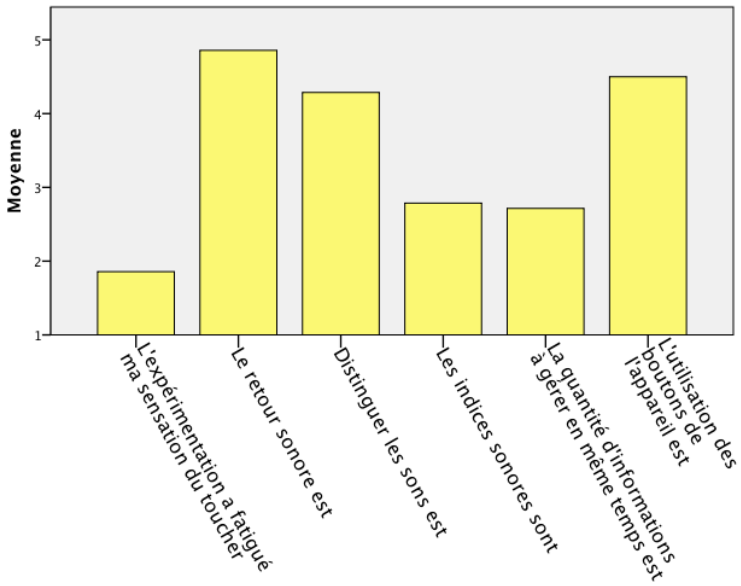


Figure 5-9 : Histogramme présentant les moyennes des réponses aux questions fermées spécifiques à TactoWeb

5.3.3 Analyse des questions ouvertes du questionnaire

Dans les questions ouvertes du questionnaire, nous demandions aux participants les points positifs et négatifs de chaque outil, ainsi que ce qu'ils souhaiteraient ajouter, supprimer ou modifier dans chacun d'eux, s'il y a lieu. Nous avons ensuite demandé aux participants de comparer les deux outils en fonction de trois critères.

5.3.3.1 Évaluation subjective de JAWS

L'évaluation subjective de JAWS a mis en relief les points suivants à propos de cet outil :

- Points positifs
  - Raccourcis clavier permettant de filtrer la page en fonction du type d'élément Web souhaité
  - Configuration poussée de l'outil
  - Qualité de la synthèse vocale
- Points négatifs
  - Aucune image globale des pages Web
  - Difficulté à naviguer dans les sites Web non-accessibles
  - Difficulté à explorer les formulaires non-accessibles

    - Temps d'apprentissage de l'outil particulièrement long
- Amélioration proposée
    - Outil combinant JAWS et TactoWeb

Les trois principaux points positifs de JAWS sont les nombreux raccourcis clavier permettant de filtrer la page (cité six fois), la possibilité de configurer JAWS en profondeur (cité six fois) et la qualité de la synthèse vocale (cité quatre fois).

Comme nous l'avons indiqué dans la section 2.2.1, les raccourcis clavier pour naviguer entre les liens, les titres, etc. permettent de naviguer très rapidement dans un site Web sans que l'on doive prêter attention au contenu. C'est très utile lorsque l'utilisateur sait qu'il n'est pas encore dans la page où se trouve l'information qu'il cherche. Une fois la bonne page trouvée, il n'utilise que très rarement les raccourcis de filtre et préfère faire une exploration complète de la page.

La possibilité de changer en profondeur les paramètres de JAWS fut aussi très appréciée par certains participants (cité six fois). En plus de pouvoir configurer les différents aspects de la synthèse vocale, ils ont apprécié pouvoir changer le mode d'exploration des formulaires et la reconnaissance automatique de la langue d'une page Web. La qualité de la synthèse vocale de JAWS fut aussi un atout selon certains participants. Même s'il est possible de changer la synthèse par défaut, celle-ci restait d'une bonne qualité selon eux.

Quant aux points négatifs de JAWS, les participants ont principalement mentionné le fait de ne pas avoir d'image globale de la page (cité six fois), les difficultés rencontrées pour naviguer dans les sites non-accessibles (cité six fois) tout particulièrement pour les formulaires (cité cinq fois), ainsi que le long temps d'apprentissage exigé par l'outil (cité quatre fois).

Certains participants ont souligné que le fait de ne pas avoir d'image globale de la page Web les empêche de se repérer correctement dans la page. JAWS reformate la page et peut donc changer l'interprétation que l'utilisateur se fait. De plus, le fait de ne pas avoir de repère spatial complique la communication avec les utilisateurs voyants, que cela soit lorsque l'utilisateur communique avec quelqu'un à côté de lui ou lorsqu'il appelle à un centre d'appel pour obtenir de l'aide. Dans ce dernier cas, il est effectivement fréquent qu'un agent de centre d'appel donne des instructions en rapport avec la localisation des éléments dans une page Web.

Les participants ne se sont pas trop plaints de JAWS lorsqu'ils rencontraient des difficultés dans les sites non-accessibles. D'après eux, JAWS fait ce qu'il peut et c'est aux concepteurs de rendre leurs sites accessibles. Les concepteurs ne se rendent pas compte que les utilisateurs non-voyants sont limités à la voix pour explorer et naviguer dans un site Web. Les participants ont aussi mentionné que les formulaires non-accessibles pouvaient devenir un véritable casse-tête, surtout que de plus en plus de services sont maintenant disponibles seulement via Internet.

Le temps d'apprentissage a aussi été mentionné comme point problématique. Certains participants ont précisé qu'il y avait beaucoup de choses à retenir dès le début et qu'apprendre tous les raccourcis clavier demande du temps. Le temps d'apprentissage particulièrement long peut être démotivant au début de l'apprentissage, surtout pour les personnes moins à l'aise avec les nouvelles technologies.

Parmi les choses pouvant être ajoutées à JAWS, un participant a souligné qu'il serait intéressant d'avoir TactoWeb comme complément à JAWS pour les situations où ce dernier est moins performant. Cet aspect de complémentarité a été mentionné quelques fois au cours de l'évaluation subjective de TactoWeb.

5.3.3.2 Évaluation subjective de TactoWeb

L'évaluation subjective de TactoWeb a mis en relief les points suivants à propos de cet outil :

- Points positifs
    - Accès à une image globale des sites Web
    - Retour tactile très précis
    - Utilisation des audicônes
    - Exploration des formulaires
    - Utilisation d'une souris comme les personnes voyantes
    - Temps d'apprentissage court
    - Temps d'exécution diminuant avec le temps
- Points négatifs
    - Mauvaise qualité de la synthèse vocale
    - Grosseur de la souris
    - Fatigue tactile au bout d'un certain temps d'utilisation

    - Demande beaucoup de précision
    - Navigation trop lente
- Amélioration proposée
    - Outil combinant JAWS et TactoWeb
    - Ajout d'un clavier
    - Ajout d'un centre des paramètres

Les quatre principaux points positifs de TactoWeb sont le fait qu'on puisse se faire une image globale du site Web (cité 11 fois), l'utilisation d'un retour tactile très précis (cité cinq fois), l'utilisation des audicônes (cité cinq fois) et un meilleur accès aux formulaires (cité cinq fois).

Le principal point positif mentionné par presque tous les participants est le fait que TactoWeb donne une image globale du site. Ces derniers ont indiqué qu'ils pouvaient savoir où les différents éléments se trouvaient et que cela les aidait à comprendre la page. Ils ont souligné que l'expérience avec TactoWeb se rapproche plus de l'expérience qu'ont les utilisateurs voyants et qu'il devrait favoriser l'échange avec eux. Avoir les retours tactiles et sonores en même temps était aussi positif. Les participants ont eu l'impression d'avoir accès à des choses qui ne leur sont pas accessibles avec JAWS.

Le retour tactile précis est aussi un point positif. Certains participants nous ont indiqué que cela leur permettait de reconnaître facilement la taille et la forme des éléments composant une page Web. Cela les a aussi guidés pour identifier la direction dans laquelle il fallait aller afin d'entendre le retour sonore de l'élément qu'ils frôlaient tactilement.

Certains participants ont aussi apprécié les audicônes car elles étaient diversifiées et peu nombreuses. Elles leur permettaient de rapidement identifier les différents types d'éléments Web. De plus, ils ont souligné que les formulaires étaient simples à visualiser avec TactoWeb. Ils les ont trouvé bien faits, peu importe qu'ils soient accessibles ou non avec JAWS.

Parmi les autres points positifs, certains participants ont aussi mentionné qu'ils pouvaient utiliser une souris comme les utilisateurs voyants (cité trois fois), que le temps d'apprentissage de TactoWeb serait sans doute court (cité trois fois) et que ce dernier pouvait devenir plus rapide à utiliser dans les sites non-accessibles (cité deux fois).

Pour ce qui est des points négatifs de TactoWeb, le plus important concerne la qualité de la synthèse vocale (cité huit fois). Même si les participants étaient familiers avec plusieurs synthèses vocales, ils nous ont dit que celle utilisée dans TactoWeb était plus difficile à comprendre, même s'ils s'y sont rapidement habitués. Certains participants ont critiqué la grosseur de la coquille entourant le module tactile (cité trois fois). En effet, la souris peut être plus difficile à prendre en main pour les personnes ayant de plus petites mains.

Le fait que la vibration sur le doigt puisse devenir fatigante, voire énervante, a aussi été indiqué par certains participants (cité trois fois). Ceux-ci ont mentionné qu'il serait sans doute difficile de passer des heures à utiliser TactoWeb sans que la sensation tactile diminue. Le fait que TactoWeb demande beaucoup de précision (cité deux fois) et que la navigation ne soit pas très rapide (cité deux fois) fut aussi souligné par des participants.

Parmi les choses pouvant être ajoutées à TactoWeb, la combinaison avec JAWS fut mentionnée cinq fois par les participants. Ceux-ci ont proposé d'associer les qualités de JAWS (synthèse vocale et raccourcis clavier pour filtrer l'information) à la qualité principale de TactoWeb qui est l'accès à l'image globale du site. Pour eux, cette association passe aussi par l'ajout d'un clavier (cité deux fois) et de paramètres configurables (cité trois fois).

5.3.3.3 Comparaison subjective des outils

À la toute fin de l'expérimentation, les participants étaient invités à dire quel outil ils préféraient, lequel était le plus facile à utiliser et lequel était le plus plaisant. Ils n'étaient pas obligés de faire un choix et pouvaient répondre les deux en même temps en justifiant leur choix.

En ce qui concerne l'outil préféré, nous obtenons un score quasi nul avec un avantage de seulement un point pour TactoWeb. Les participants ont aimé JAWS car ils le connaissent et savent où ils s'en vont quand ils l'utilisent. Les raccourcis clavier permettant de filtrer l'information sont toujours appréciés. Les participants ont aimé TactoWeb même s'il est jugé moins performant. Ils ont trouvé que c'était un bon outil de réadaptation qui les rendrait plus fonctionnels. Ils ont considéré que c'était un outil plus concret que JAWS car il permet de se faire une image du site Web, semblable à celle qu'ont les utilisateurs voyants. Les participants ont réitéré dans leur réponse à cette question le fait que les outils sont complémentaires et que ça serait une bonne idée d'offrir un outil combinant JAWS et TactoWeb.

Lorsque nous leur avons demandé quel outil ils trouvaient le plus facile à utiliser, JAWS l'a emporté largement avec sept voix contre zéro pour TactoWeb (les sept autres participants ont indiqué que les deux outils étaient faciles à utiliser). Tous les participants ayant choisi JAWS ont indiqué qu'il était plus facile pour eux car cela faisait longtemps qu'ils l'utilisaient. Ils ont indiqué qu'ils étaient plus habitués à utiliser un clavier qu'une souris et que les raccourcis clavier rendaient plus facile la navigation quand ils sont connus. Néanmoins, certains participants ont indiqué que TactoWeb leur permettait de se faire une meilleure image mentale d'une page Web et que cela rendait plus facile le processus de recherche d'une information. Ces derniers souhaitent d'ailleurs utiliser TactoWeb à plus long terme pour savoir s'il pourrait devenir plus facile à utiliser que JAWS avec le temps.

Enfin, les participants ont jugé que les deux outils étaient aussi plaisants à utiliser. JAWS est portable et ancré dans la vie des participants. TactoWeb a eu un effet de nouveauté et de plaisir car il leur a permis de palper un site Web à l'écran. La majorité des participants étaient d'accord sur le fait que TactoWeb possède un gros potentiel.

5.4 Synthèse et contributions

La comparaison entre JAWS et TactoWeb nous a permis d'obtenir des résultats intéressants et de faire plusieurs observations. Vérifions d'abord nos hypothèses de départ. Le principal résultat est que les deux types d'exploration sont globalement aussi efficaces l'un que l'autre avec un temps d'apprentissage beaucoup plus court pour l'exploration spatiale. Même si cela infirme notre hypothèse principale A, il s'agit d'une des contributions principales de cette thèse. Pour ce qui est de l'efficience, le niveau de difficulté rencontré est semblable pour les deux types d'exploration mais l'exploration linéaire demeure plus rapide à cause vraisemblablement de l'expérience des participants avec celle-ci et d'une plus grande maturité de JAWS, quelque soit le type de tâche. Ceci infirme clairement notre hypothèse principale B. Le fait de pouvoir filtrer les éléments Web puis de les parcourir à l'aide de raccourcis clavier influe beaucoup sur cette rapidité. Il est raisonnable de penser que la rapidité de l'exploration spatiale augmenterait avec la pratique. Néanmoins, l'exploration linéaire reste pour l'instant drastiquement plus rapide. En tenant compte de tous ces points, notre hypothèse principale est donc infirmée.

Pour les tâches de recherche d'information dans un site Web, on obtient des résultats similaires à l'expérience globale avec un avantage net pour l'exploration linéaire. Ceci confirme notre sous-hypothèse 2B mais infirme notre sous-hypothèse 2A. La difficulté de l'exploration linéaire est inférieure à celle de l'exploration spatiale pour les sites accessibles. Cela peut s'expliquer par le fait que l'exploration linéaire n'est pas affectée par les contraintes de formatage et que les raccourcis clavier de filtrage aident grandement la navigation dans un site Web. Par exemple, un menu sera exploré comme une liste de liens, peu importe qu'il soit orienté verticalement ou horizontalement. L'exploration spatiale a d'ailleurs posé des problèmes lorsqu'il fallait explorer horizontalement une partie de l'écran, et cela de manière assez précise. Ce problème d'exploration horizontale s'est confirmé dans les sites non-accessibles, ce qui infirme notre sous-hypothèse 1. Ce problème est aussi apparu lors de l'exploration du tableau, même si celui-ci présentait des problèmes d'accessibilité. Le traitement des tableaux sera à améliorer avec TactoWeb. Néanmoins, le taux de succès dans la tâche et le niveau de difficulté rencontré ne sont pas significativement différents sur l'ensemble des tâches de recherche d'information. Nous avons par ailleurs observé que l'exploration spatiale semblait plus adaptée lorsque l'utilisateur devait explorer l'ensemble d'une page Web et que les raccourcis clavier de l'exploration linéaire ne lui étaient d'aucune utilité (comme dans la tâche S4).

Pour les tâches de remplissage de formulaire, l'exploration spatiale semble plus efficace (en terme de taux de succès dans la tâche) que l'exploration linéaire mais cela n'est pas confirmé de manière significative. Nous ne pouvons donc pas confirmer catégoriquement notre sous-hypothèse 3A. L'exploration linéaire est toujours significativement plus rapide que l'exploration spatiale pour ce type de tâche, ce qui infirme notre sous-hypothèse 3B malgré un difficulté moindre pour l'explorations spatiale. Le fait qu'il soit très facile de trouver le premier champ d'un formulaire à l'aide d'un raccourci clavier peut d'ailleurs jouer sur cette rapidité. Néanmoins, l'association spatiale entre le champ de formulaire et l'étiquette qui lui est associée fonctionne correctement, même si cette association n'est pas présente dans le code source HTML, comme nous avons pu l'observer dans les tâches effectuées dans des sites non-accessibles. Nous obtenons un meilleur taux de succès et un niveau de difficulté moins élevé pour l'exploration spatiale, même s'ils ne sont pas significatifs. Ce même sous-ensemble de tâches a fait ressortir certains problèmes dans l'exploration spatiale. L'espacement vertical entre les différents champs

devrait être optimisé si celui-ci est trop petit. Il en est de même pour l'espacement horizontal entre un champ et l'étiquette qui lui est associée.

Pour revenir sur notre sous-hypothèse 1, elle est clairement infirmée pour les tâches de recherche d'information. Cela est cependant moins évident pour les tâches de remplissage de formulaire. De plus, les écarts d'efficacité et d'efficience sont moins importants dans les sites non-accessibles que dans les sites accessibles, tout particulièrement pour les tâches de remplissage de formulaire.

Il faut aussi noter que les tâches de l'expérimentation étaient orientées vers un but précis, à savoir trouver une information ou remplir un formulaire. Il s'agit d'un cas d'utilisation fréquent de l'Internet mais il arrive souvent qu'on explore le Web sans but précis, dans des sites qu'on ne connaît pas. Dans ce dernier cas, il n'est pas aussi évident que l'exploration linéaire soit aussi rapide et que l'écart de vitesse d'exploration entre les deux types d'exploration soit aussi important.

En ce qui concerne la satisfaction, les participants ont globalement préféré leur expérience avec l'exploration linéaire. Il faut néanmoins nuancer ce résultat en soulignant qu'ils utilisent ce type d'exploration (à l'aide de lecteurs d'écran) depuis des années. Il n'empêche que les participants ont trouvé l'exploration linéaire globalement plus facile. Les participants ont aussi mentionné le fait que la synthèse vocale utilisée pour l'exploration linéaire était de bien meilleure qualité que celle utilisée pour l'exploration spatiale. Ils ont aussi souligné que le fait de pouvoir filtrer l'information des pages Web avec des raccourcis clavier était un gros avantage pour l'exploration linéaire. Par contre, ils ont trouvé que l'exploration spatiale leur donnait un meilleur accès au formulaire, particulièrement lorsqu'ils ne suivent pas correctement les lignes de conduite d'accessibilité. De plus, le retour tactile a été apprécié pour obtenir une idée précise de la taille et de la forme des éléments Web. Les participants ont aussi très apprécié l'emploi d'audicônes afin d'identifier les différents types d'élément Web composant une page. Le fait qu'on puisse alléger le texte énoncé par la synthèse vocale à l'aide d'audicônes est donc un bon point dans notre outil d'exploration spatiale.

Toujours à propos de la satisfaction, le fait d'avoir accès à l'image globale des pages Web a été fortement apprécié. Il s'agit d'une autre contribution importante de cette thèse. Les participants ont indiqué que l'exploration spatiale leur donnait une meilleure idée globale du site Web, chose à laquelle nous pouvions raisonnablement nous attendre. Mais ils ont surtout indiqué que c'était

un point positif pour eux car cela leur donnait accès aux mêmes informations que les utilisateurs voyants. D'après eux, l'exploration spatiale leur permettrait de mieux communiquer avec les personnes voyantes lorsqu'ils sont sur un site Web. Le fait d'avoir accès à l'image du site n'est donc pas qu'un gadget, c'est quelque chose qui est jugé utile.

Le tutoriel que nous avons développé est aussi une contribution majeure de cette thèse car nous sommes parvenus à obtenir des taux de succès équivalents avec les deux outils malgré le fait que les participants avaient une expérience de plusieurs années avec l'exploration linaire et étaient relativement novices avec l'exploration spatiale. De plus, les tâches de contrôle de l'apprentissage du tutoriel n'ont posé aucun problème. Cela montre que le tutoriel a bien rempli son rôle et que les participants se sont montrés rapidement à l'aise avec l'exploration spatiale. Cela montre aussi que l'exploration spatiale demande un temps d'apprentissage relativement court. Si on prend en compte le fait que l'apprentissage de l'exploration linaire peut être particulièrement long (entre six mois et un an) et que les utilisateurs non-voyants peuvent trouver cela bloquant, TactoWeb et son exploration spatiale possèdent un avantage non négligeable en terme de temps d'apprentissage.

Par ailleurs, nous n'avons trouvé aucune différence significative entre les performances globales des personnes non-voyantes de naissance et tardives dans l'utilisation des outils. Nous nous attendions à ce que les personnes non-voyantes tardives performent mieux avec l'exploration spatiale car elles développeraient un modèle mental de leur environnement beaucoup plus proche de celui des personnes voyantes. Le fait qu'il n'y ait pas de différence est donc une surprise. Cette surprise est encore plus importante quand on regarde les résultats obtenus dans les tâches de remplissage de formulaire avec TactoWeb, où les personnes non-voyantes de naissances ont été plus efficaces. Ce type de tâches demande plus de repères au niveau de l'exploration spatiale et nous aurions plus pensé que les personnes non-voyantes tardives auraient été plus performantes dans ce cas. Les personnes non-voyantes de naissance semblent avoir surmonté ce désavantage à l'aide de leur plus grande expérience avec les technologies d'adaptation.

Enfin, à la lumière des commentaires des participants au sujet de JAWS et de TactoWeb, et de nos propres observations, il s'avère que l'exploration linéaire semble plus adaptée aux tâches de navigation et l'exploration spatiale plus adaptée aux tâches d'exploration. L'exploration linéaire est plus rapide pour naviguer grâce à la fonctionnalité de JAWS permettant de filtrer

l'information et de sauter rapidement d'un élément filtré à un autre. Elle permet ainsi de trouver plus facilement la bonne page où se trouve l'information souhaitée. De son côté, l'exploration spatiale est plus pertinente lorsqu'il s'agit d'explorer la page dans laquelle on sait où va se trouver l'information cherchée. L'exploration spatiale semble aussi plus adéquate pour explorer un formulaire car elle n'est pas affectée par le manque d'accessibilité du site aux personnes non-voyantes. Une combinaison des deux types d'exploration est donc à envisager pour améliorer l'accès des sites Web aux personnes non-voyantes.

CONCLUSION

Cette thèse s'est intéressée au problème d'accès des personnes non-voyantes aux sites Web. Des efforts ont été faits par des gouvernements, des entreprises de technologie d'adaptation et par le W3C pour améliorer l'accès des personnes ayant une déficience visuelle aux sites Web. Les personnes non-voyantes ont présentement accès aux sites Web à l'aide de logiciels de lecture d'écran. Ces logiciels reformatent l'information présentée sur un page Web en la linéarisant, puis en générant un retour sonore lu par une synthèse vocale. Ce retour peut aussi être généré sur une page Braille. Pour que les logiciels de lecture d'écran fonctionnent de manière optimale lors de l'exploration d'une page Web, celle-ci doit suivre des lignes de conduite d'accessibilité. Il en existe des globales, rédigées par le W3C, ou des plus spécifiques à certaines nations, comme les États-Unis ou le Québec. Ces mêmes nations rendent obligatoire l'application de ces lignes de conduite au moins pour les sites gouvernementaux. Néanmoins, elles sont loin d'être entrées dans les mœurs des entreprises privées, ce qui provoque des problèmes d'accessibilité dans de nombreux sites Web existants.

Au début de cette thèse, nous avons mentionné que l'utilisation des lecteurs d'écran créait de la frustration chez les utilisateurs à cause du non-respect des lignes de conduite d'accessibilité. Nous avons remarqué que plusieurs problèmes étaient directement liés au type d'exploration linéaire propre aux logiciels de lecture d'écran. Cette linéarisation de l'information peut entraîner la perte du contexte de l'information ou ralentir l'exploration d'une page.

Nous avons d'abord évalué deux lecteurs d'écran : JAWS, le lecteur d'écran le plus utilisé par la population non-voyante, et VoiceOver proposant aussi une exploration spatiale. Les résultats ont révélé que le temps d'apprentissage d'un lecteur d'écran est particulièrement long. Il se compte en mois car il faut d'abord apprendre à se servir tactilement d'un clavier puis apprendre les nombreux raccourcis clavier. Néanmoins, une fois maîtrisés, les lecteurs d'écran peuvent être assez rapides à utiliser, notamment au cours de la navigation entre les pages d'un même site Web, surtout s'il s'agit d'un site familier aux utilisateurs. Cette rapidité d'utilisation peut s'expliquer par le fait de pouvoir filtrer les pages en fonction d'un type d'élément souhaité. Néanmoins, dans un site inconnu, l'exploration d'une page est plus lente si la page ne suit pas correctement les lignes de conduite d'accessibilité. L'utilisation efficace des lecteurs d'écran dépend donc fortement de la bonne application de ces lignes de conduite.

Contributions

Une alternative à l'exploration linéaire est présentement possible : l'exploration spatiale à l'aide du logiciel de lecture d'écran VoiceOver, disponible sur ordinateurs et appareils mobiles de la gamme Apple. Cette exploration spatiale ne semble malheureusement pas adoptée par les personnes non-voyantes à cause de son manque de précision et de rapidité. Les utilisateurs non-voyants lui préfèrent l'exploration linéaire via des commandes gestuelle, disponibles sur ce même lecteur d'écran. Ce manque de précision et de rapidité de l'exploration spatiale est principalement causé par le fait que seul un retour sonore est proposé pour guider l'utilisateur.

La première et principale contribution de cette thèse a été de créer le logiciel TactoWeb qui vise à remédier à ce problème. Il s'agit d'un logiciel d'exploration spatiale de l'affichage à l'écran offrant des retours sonore **et tactile** afin de créer une meilleure expérience d'exploration spatiale. La souris y est remplacée par un appareil tactile : le Tactograph. En plus de prendre en compte la mise en page et le formatage des éléments Web, TactoWeb génère des retours tactiles et sonores en fonction de leur type. Les types d'éléments Web sont indiqués grâce à des textures et/ou des audicônes. Le texte de la page Web est lu par l'intermédiaire d'une synthèse vocale.

Le temps d'apprentissage des lecteurs d'écran étant particulièrement long, nous ne voulions pas que cela soit un problème avec TactoWeb. L'exploration spatiale n'est pas un processus évident pour les personnes non-voyantes, particulièrement les personnes non-voyantes de naissance. Nous avons donc créé un tutoriel d'aide à l'apprentissage de TactoWeb basé sur l'interaction tuteur-apprenant. Ce tutoriel est la seconde contribution majeure de cette thèse. Contrairement à la grande majorité des tutoriels existants, celui-ci est strictement textuel à cause de la nature du handicap des apprenants. C'était d'ailleurs un défi à relever que de devoir se priver d'une des modalités d'interaction principales utilisées lors de la conception d'un tutoriel, à savoir le retour visuel.

Nous avons mené une étude pour comparer la performance et la satisfaction de sujets non voyants à qui il était demandé de réaliser deux types de tâches sur des sites Web en faisant une exploration linéaire avec JAWS ou une exploration spatiale avec TactoWeb. Les principaux résultats montrent que l'exploration linéaire et l'exploration spatiale sont aussi efficaces l'une que l'autre, malgré le fait que les participants utilisaient l'exploration linéaire depuis des années et que l'exploration spatiale se fasse sur un outil ayant très peu de maturité technologique. Même

si le tutoriel de TactoWeb a permis d'atténuer un peu ces deux biais, le fait que leur efficacité soit similaire montre que l'exploration spatiale semble plus rapide à assimiler et prouve le succès de notre tutoriel. Dans nos conditions d'expérimentation, l'exploration linéaire reste néanmoins plus rapide à utiliser, surtout lorsqu'il s'agit de naviguer entre les pages d'un site Web. C'est la possibilité de filtrer l'information en fonction du type d'élément qui donne cet avantage à l'exploration linéaire. Néanmoins, la plus grande rapidité de l'exploration linéaire est à nuancer, justement à cause de nos conditions d'expérimentation, parce qu'il y a une grande différence d'expérience des participants avec les deux types d'exploration et une différence de maturité entre les deux outils que nous avons évalués. Il est raisonnable de penser qu'il pourrait en être autrement lorsque ces deux conditions seront levées. Le biais apporté par ces deux conditions s'est moins fait ressentir dans les sites non-accessibles. Ceci montre que l'exploration spatiale semble être moins dépendante du non-respect des lignes de conduite d'accessibilité.

Un autre résultat important de notre étude est que l'exploration spatiale semble plus efficace que l'exploration linéaire lorsqu'il s'agit de remplir des formulaires, même s'il faudrait faire une expérimentation à plus long terme pour le prouver de manière significative. L'exploration spatiale est beaucoup moins affectée par le non respect des lignes de conduite d'accessibilité que l'exploration linéaire dans ce type de tâches. En effet, elle n'est pas désavantagée par le fait que les étiquettes soient codées comme telles et que l'association entre les étiquettes et les champs soit faite dans le code source de la page HTML. Néanmoins, si les formulaires sont sur des sites accessibles, l'exploration linéaire demeure plus rapide. Les formulaires sont basés sur l'exploration d'une page, et non pas sur la navigation entre les pages d'un même site Web. Dans ce cas, l'exploration spatiale semble avoir le dessus, malgré un temps d'apprentissage très court. Si on ajoute cette observation à celle rapportée dans le paragraphe précédent, l'exploration linéaire est plus appropriée pour naviguer entre les pages d'un site Web, alors que l'exploration spatiale semble plus appropriée pour explorer une page Web. Ce mélange entre les types d'exploration a d'ailleurs été mentionné par les participants lorsqu'ils répondaient au questionnaire de satisfaction à la fin de l'expérimentation.

Les principaux résultats sur le niveau de satisfaction des utilisateurs à l'endroit des deux modes d'exploration montrent premièrement que l'exploration linéaire génère un niveau de satisfaction plus élevé que l'exploration spatiale. Néanmoins, les participants ont nuancé leurs propos en indiquant que c'était normal qu'ils préfèrent l'exploration linaire car ils l'utilisent depuis des

années. Ils ont d'ailleurs émis le souhait d'utiliser l'exploration spatiale avec TactoWeb sur une plus longue durée afin de se faire un meilleur jugement. Deuxièmement, l'exploration spatiale donne clairement une meilleure idée globale d'une page Web que l'exploration linéaire. Produire un outil d'exploration spatiale donnant une meilleure idée globale des pages Web était un gros défi au début de la thèse. Enfin, l'exploration spatiale améliore l'interaction entre les personnes non-voyantes et voyantes. C'est un résultat très intéressant. En effet, les personnes non-voyantes doivent souvent interagir avec des personnes voyantes. Les échanges entre elles peuvent souvent inclure des références spatiales, car une personne voyante ne fait pas forcément attention au fait que la personne à laquelle elle s'adresse est non-voyante.

Un élément que nous n'avons pas pris en compte dans nos hypothèses de départ est la durée d'apprentissage. Celle-ci est particulièrement longue pour l'exploration linéaire, même si son apprentissage inclut des fonctionnalités autres que l'exploration du Web. Au contraire, le temps d'apprentissage semble très court pour l'exploration spatiale. À l'aide d'un tutoriel d'une trentaine de minutes, nous obtenons une efficacité équivalente pour l'exploration spatiale. Cette observation relativise le fait que la majorité de nos hypothèses de départ ont été infirmées. En effet, elles ont été infirmées car nous avons obtenu une performance équivalente avec les deux types d'exploration (sauf pour le temps d'exécution). Mais obtenir une performance équivalente avec un temps d'apprentissage beaucoup plus court pour l'exploration spatiale est un gros point positif et une de nos contributions majeures.

En fait, les deux types d'explorations seraient complémentaires selon le contexte d'utilisation. L'exploration spatiale est un complément à l'exploration linéaire. Elle permet de donner une bonne idée globale d'une page qu'on voudrait explorer plus en détails ou de donner une alternative lorsque le site ne suit pas les lignes de conduite d'accessibilité. En effet, l'exploration spatiale n'est pas affectée par les problèmes des sites non-accessibles, en particulier le non-respect de l'ordre logique de lecture d'une page et les formulaires non-accessibles.

Perspectives

L'avenue principale de recherche concerne la coopération entre les explorations linéaire et spatiale. Celle-ci semble très prometteuse et demanderait beaucoup d'effort. Les participants ont plusieurs fois émis le souhait d'avoir accès à une combinaison de JAWS et de TactoWeb, pour bénéficier des avantages des deux types d'exploration. Pour ce faire, il faudra associer le curseur

de l'exploration linéaire au curseur du module tactile de l'exploration spatiale. Il paraît difficile de faire cette association en utilisant deux logiciels en même temps car il faudrait que ceux-ci se « parlent » constamment. Intégrer l'exploration linéaire à l'intérieur de TactoWeb semble la solution générant le moins de contrainte. En effet, le DOM utilisé par les logiciels de lecture d'écran est déjà récupéré par TactoWeb, alors que ce dernier n'est pas un fureteur classique compatible avec JAWS. Sachant qu'il est possible d'avoir accès à l'information linéarisée grâce au DOM, il semble tout à fait réalisable d'intégrer les fonctionnalités principales de l'exploration linaire dans TactoWeb. Ces fonctionnalités incluraient bien sûr l'exploration pas à pas à l'aide des flèches directionnelles du clavier, mais surtout la possibilité de filtrer l'information en fonction des types d'éléments Web. De plus, pour éviter l'apprentissage du clavier, on pourrait penser à une méthode plus simple pour entrer des commandes de l'exploration linéaire, telle que la commande vocale ou un contrôleur de jeux vidéo, possédant déjà des flèches directionnelles et un assez grand nombre de boutons pour activer les fonctionnalités principales de la synthèse vocale. Le dernier problème concerne la synchronisation des curseurs de l'exploration linéaire et de l'exploration tactile. Si le déplacement du curseur de l'exploration linéaire en fonction de celui de l'exploration tactile ne semble pas problématique, la difficulté est toute autre pour le processus inverse. Une solution serait l'utilisation d'un retour haptique qui permettrait de bouger automatiquement le module tactile en fonction du déplacement du curseur de l'exploration linéaire.

En ce qui concerne l'exploration spatiale en tant que telle, elle semble prometteuse mais a encore plusieurs limitations. Plusieurs fonctionnalités restent à améliorer. D'abord il faudrait tester l'exploration spatiale à plus long terme. Cela permettrait de savoir si les temps d'exécution et la satisfaction des utilisateurs s'amélioreront avec l'expérience. Puis il faudrait améliorer la synthèse vocale utilisée. Nous avions choisi une seule synthèse vocale multiplateforme, multilingue et gratuite mais celle-ci n'est pas d'assez bonne qualité. Nous avons préféré la flexibilité à la qualité, ce qui ne s'est pas avéré la bonne façon de faire. Il vaudrait mieux implémenter plusieurs systèmes de synthèse vocale de bonne qualité pour chaque langue et/ou système d'exploitation. La synthèse vocale souhaitée pourrait d'ailleurs être choisie par l'utilisateur au sein d'un centre des paramètres du même genre que celui disponible dans JAWS. TactoWeb est pour l'instant très succinct comme fureteur Web. Il n'y a pas de paramètres et les

favoris doivent être saisis manuellement. La page d'accueil, les favoris et les paramètres de la synthèse vocale pourraient donc être gérés via un centre des paramètres.

Le centre des paramètres permettrait aussi de gérer les types d'éléments Web. On pourrait laisser à l'utilisateur le choix de créer une combinaison d'audicônes, de texte à lire en synthèse vocale et/ou de textures tactiles afin de les identifier. Le centre des paramètres pourrait aussi permettre la gestion des deux boutons du Tactograph. La gestion des clics longs et des double-clics serait ajoutée et pourrait permettre par exemple de répéter le retour sonore, d'ajouter la page courante aux favoris, ou d'interagir avec les champs de formulaires.

L'interaction avec les champs de formulaire serait aussi une fonctionnalité à ajouter dans TactoWeb. Nous souhaitons toujours éviter l'utilisation d'un clavier, vu qu'il faut beaucoup de temps pour apprendre à l'utiliser tactilement. L'alternative la plus pertinente semble être la reconnaissance vocale, même si elle semble bien plus compliquée à implémenter. De nombreux progrès ont tout de même été faits dans cette technologie. Elle est d'ailleurs maintenant disponible auprès du grand public avec la majorité des téléphones intelligents.

L'exploration en deux dimensions avec le Tactograph pourrait aussi être améliorée. Les sites Web sont dans la grande majorité conçus sous la forme d'une grille. Si on veut se déplacer spatialement, il est nécessaire de se déplacer correctement de manières horizontale et verticale. Une des limitations de TactoWeb est qu'il n'est pas encore très précis pour l'exploration horizontale. Des guides de déplacement sur la surface d'exploration pourraient être utiles afin d'améliorer la précision de l'exploration horizontale. Ces guides pourraient aussi être sous la forme de retours haptiques qui limiteraient les déplacements diagonaux du module tactile. L'exploration en diagonale serait toujours possible mais une résistance serait appliquée lorsque l'utilisateur souhaiterait effectuer des mouvements en diagonale. Ces guides, haptiques ou non, aideraient fortement l'exploration des formulaires et des tableaux. On peut aussi imaginer le cas où un retour haptique permettrait de filtrer l'information d'une page Web en fonction d'un type d'élément Web souhaité. Le retour haptique pourrait guider le module tactile vers les éléments d'un type choisi en créant des sortes d'encoches pour chacun de ces éléments.

Un autre ajout qui semble pertinent concerne le défilement de la page Web. En effet, TactoWeb ne permet pour l'instant que d'explorer le premier écran de la page. Le problème de défilement de la page avec l'exploration spatiale se situe au niveau de la référence du curseur d'exploration.

Comment garder le même point de référence du curseur alors qu'on fait défiler sur une page? Cette question ne se pose pas pour les utilisateurs voyants qui ont un repère visuel lorsqu'ils font défiler une page Web. Mais dans le cas de l'exploration spatiale qui est absolue, le bout de page qui est associé à la surface d'exploration « glisserait » vers le haut ou le bas, sans que le curseur représentant la cellule tactile ne bouge de la surface. On peut donc imaginer que l'utilisateur sentirait et entendrait des éléments Web sous son doigt en faisant défiler la page, mais sans bouger le module tactile. Le problème est que l'utilisateur aura perdu le point de référence où il était avant de faire défiler la page. Un retour haptique pourrait encore une fois être une solution. Celui-ci hameçonnerait le point de référence et bougerait automatiquement le module tactile pour que le point de référence et le module restent associés. Ainsi, l'utilisateur serait moins perdu au cours du défilement de la page.

Afin de mettre en place tous ces changements, un plan de développement devra être défini. TactoWeb est à la base un moyen de tester l'exploration spatiale avec des retours tactiles et sonores. Il n'a été développé que par un seul développeur. Si nous voulons créer un véritable fureteur, compatible avec beaucoup plus de sites Web, il faudra sans doute l'aide de l'industrie des technologies adaptées.

BIBLIOGRAPHIE

AI Squared. (2013). Making accessibility simple - ZoomText. Consulté le 17 avril 2013 à http://www.aisquared.com/zoomtext/

American Foundation for the Blind. (2013). Key Definitions of Statistical Terms. Consulté le 17 avril 2013 à http://www.afb.org/section.aspx?SectionID=15&DocumentID=1280

Apple. (2013). Accessibilité - VoiceOver. Consulté le 17 avril 2013 à http://www.apple.com/fr/accessibility/voiceover/

Association des médecins ophtalmologistes du Québec. (2013a). Section informations sur les maladies - La cataracte. Consulté le 17 avril 2013 à http://www.amoq.org/InfoSurMaladies/InfoSurMala-cataracte.htm

Association des médecins ophtalmologistes du Québec. (2013b). Section informations sur les maladies - La dégénérescence maculaire. Consulté le 17 avril 2013 à http://www.amoq.org/InfoSurMaladies/InfoSurMala-Maculaire.htm

Association des médecins ophtalmologistes du Québec. (2013c). Section informations sur les maladies - La rétinopathie diabétique. Consulté le 17 avril 2013 à http://www.amoq.org/InfoSurMaladies/InfoSurMala-diab.htm

Association des médecins ophtalmologistes du Québec. (2013d). Section informations sur les maladies - Le glaucome. Consulté le 17 avril 2013 à http://www.amoq.org/InfoSurMaladies/InfoSurMala-Glucome.htm

Baccino, T., Bellino, C., & Colombi, T. (2005). *Mesure de l'utilisabilité des interfaces*. Hermes Science.

Barreto, A. (2008). Visual Impairments. Dans Harper, S., & Yesilada, Y. (dir.), *Web accessibility: a foundation for research* (p. 3-13). Springer.

Bastian, C. H. (1887). The "muscular sense". *Brain, 10*(1), 1-119.

Bertelson, P., Mousty, P., & D'Alimonte, G. (1985). A study of braille reading: 2. Patterns of hand activity in one-handed and two-handed reading. *The Quarterly Journal of Experimental Psychology Section A, 37*(2), 235-256.

Boff, K. R., Kaufman, L., & Thomas, J. P. (1988). *Handbook of Perception and Human Performance: Sensory processes and perception, 1*. Wiley.

Bolt, R. A. (1980). *"Put-that-there": Voice and gesture at the graphics interface*. *Proceedings of the 7th annual conference on Computer graphics and interactive techniques*, New York, NY, USA, 262-270.

Burdea, G. A. (1996). *Force and touch feedback for virtual reality*. Wilter-Interscience.

Card, S. K., Newell, A., & Moran, T. P. (1983). *The Psychology of Human-Computer Interaction*. L. Erlbaum Associates Inc.

Colas, S., Monmarché, N., & Slimane, M. (2009). Assistive Tools to Reach European Accessibility Web Standards. Dans Stephanidis, C. (dir.), *Universal Access in Human-Computer Interaction. Applications and Services* (Vol. 5616, p. 654-663). Springer Berlin Heidelberg.

Crichton, P., & Crichton, J. (1994). Penfield's homunculus. *Journal of neurology, neurosurgery, and psychiatry, 57*(4), 525-533.

de Castro Reis, D., & Braga Araújo, R. (2013). HTMLCXX - HTML and CSS APIs for C++. Consulté le 17 avril 2013 à http://htmlcxx.sourceforge.net/

Dolphin. (2013). SuperNova Screen Reader – Screen reader with Speech and Braille support. Consulté le 17 avril 2013 à http://www.yourdolphin.com/productdetail.asp?id=5

Dufresne, A., Martial, O., & Ramstein, C. (1995). *Multimodal user interface system for blind and "visually occupied" users: Ergonomic evaluation of the haptic and auditive dimensions.* Nordby, K. *Proceedings of the 5th IFIP Conference on Human-Computer Interaction (INTERACT'95)*, Lillehammer, Norway, 163-168.

Dufresne, A., Martial, O., Ramstein, C., & Mabilleau, P. (1996). *Sound, space, and metaphor: Multimodal access to windows for blind users*. Frysinger, S. P., & Kramer, G. *Proceedings of the 3rd International Conference on Auditory Display (ICAD96)*, Palo Alto, CA, U.S.A, 51-58.

Ebina, T., Igi, S., Miyake, T., & Takahashi, H. (1998). *Graph Access System for the Visually Impaired*. *Proceedings of the 3rd Asian Pacific Computer and Human Interaction*, Shonan Village Center, Japan, 254-259.

Edwards, A. D. N. (2008). Assistive Technologies. Dans Harper, S., & Yesilada, Y. (dir.), *Web accessibility: a foundation for research* (p. 141-162). Springer.

Farhat, S., & Bellik, Y. (1999). *SeeWeb: Dynamic Improvement of the accessibility of HTML documents for blind persons*. Sasse, M. A., & Johnson, C. *Proceedings of the 7th IFIP Conference on Human-Computer Interaction (INTERACT'99)*, Edinburgh, Scotland, 256-262.

Fernandez, E., & Rubens, E. (2005, 19-22 Oct. 2005). *Developing online tutorials: the methodology and the reality*. *Proceedings of the 35th Annual Conference on Frontiers in Education (FIE '05)*, Indianopolis, IN, F2E.14-F12E.18.

Freedom Scientific. (2013a). JAWS for Windows Screen Reading Software. Consulté le 17 avril 2013 à http://www.freedomscientific.com/products/fs/jaws-product-page.asp

Freedom Scientific. (2013b). MAGic Screen Magnification Software. Consulté le 17 avril 2013 à http://www.freedomscientific.com/products/lv/magic-bl-product-page.asp

Ganglbauer, E., Schrammel, J., Deutsch, S., & Tscheligi, M. (2009). *Applying Psychophysiological Methods for Measuring User Experience: Possibilities, Challenges, and Feasibility*. Gross, T., Gulliksen, J., Kotzé, P., Oestreicher, L., Palanque, P., Prates, R., & Winckler, M. *Proceedings of User Experience Evaluation Methods in Product Development (UXEM'09)* Uppsala, Sweden.

Gapenne, O., Rovira, K., Ali Ammar, A., & Leney, C. (2003). *Tactos: Special computer interface for the reading and writing of 2D forms in blind people* Stephanidis, C. *Universal Access in HCI, Inclusive Design in the Information Society*, Heraklion, Greece, 1270-1274.

Garrison, D. R. (1990). An analysis and evaluation of audio teleconferencing to facilitate education at a distance. *American Journal of Distance Education, 4*(3), 13-24.

Gibson, J. J. (1962). Observations on active touch. *Psychological Review, 69*, 477-490.

Ginsburg, A. P., & Hendee, W. R. (1993). Quantification of Visual Capability. Dans Hendee, W. R., & Wells, P. N. T. (dir.), *The perception of visual information* (p. 57-86). Springer-Verlag.

Goldish, L. H., & Taylor, H. E. (1974). The Optacon: A Valuable Device for Blind Persons. *New Outlook for the Blind, 68*, 49-56.

Goldreich, D., & Kanics, I. M. (2003). Tactile Acuity is Enhanced in Blindness. *The Journal of Neuroscience, 23*(8), 3439-3445.

Google. (2013). The Chromium projects. Consulté le 17 avril 2013 à http://www.chromium.org/

Gould, J. D., & Lewis, C. (1985). Designing for usability: key principles and what designers think. *Communications of the ACM, 28*(3), 300-311.

Graupp, H., Gladstone, K., & Thompson, L. (2002). *Haptic Perception of 2D Pictures and 3D Objects: Accurate Mental Representation as a Function of Visual Status*. Miesenberger, K., Klaus, J., & Zagler, W. L. *Proceedings of the 8th International Conference on Computers Helping People with Special Needs (ICCHP '02)*, Linz, Austria, 543-550.

Guyton, A. C., & Hall, J. E. (2006). *Textbook of Medical Physiology*. (11e éd.). Saunders Company.

GW Micro. (2013). GW Micro - Window-Eyes. Consulté le 17 avril 2013 à http://www.gwmicro.com/Window-Eyes/

Haxx. (2013). cURL and libcurl. Consulté le 17 avril 2013 à http://curl.haxx.se/

Hayward, V., & MacLean, K. E. (2007). Do it yourself haptics: part I. *IEEE Robotics & Automation Magazine, 14*(4), 88-104.

Hollins, M. (1989). *Understanding Blindness: An Integrative Approach*. Lawrence Erlbaum Associates.

HumanWare. (2013). Produits Basse Vision pour la Dégénérescence Maculaire. Consulté le 17 avril 2013 à http://www.humanware.com/fr-canada/home

IBM. (2013). SPSS Statistics. Consulté le 23 avril 2013 à http://www-01.ibm.com/software/analytics/spss/products/statistics/

Jones, L. A., & Lederman, S. J. (2006). *Human hand function*. Oxford University Press.

Kandel, E., Schwartz, J. H., & Jessell, T. M. (2000). The Bodily Senses (*Principles of Neural Science* (4e éd., p. 430-450). McGraw-Hill Medical.

Katz, D. (1989). *The world of touch [Der aufbau der tastwelt]*. (Traduit par Krueger, L. E.). Lawrence Erlbaum.

Kirkpatrick, A. (2006). Overview of accessible technologies. Dans Thatcher, J. (dir.), *Web accessibility : web standards and regulatory compliance* (p. 103-124). Friends of ED.

Klatsky, R. L., Lederman, S. J., & Metzger, V. A. (1985). Identifying objects by touch: An "expert system". *Perception & psychophysics, 37*(4), 299-302.

Lazar, J., Allen, A., Kleinman, J., & Malarkey, C. (2007). What Frustrates Screen Reader Users on the Web: A Study of 100 Blind Users. *International Journal of Human-Computer Interaction, 22*(3), 247 - 269.

Lederman, S. J., Klatsky, R. L., Chataway, C., & Summers, C. D. (1990). Visual Mediation and the Haptic Recognition of 2-Dimensional Pictures of Common Objects. *Perception & psychophysics, 47*(1), 54-64.

Leporini, B., & Paternò, F. (2004). Increasing usability when interacting through screen readers. *Universal Access in the Information Society, 3*(1), 57-70.

Lévesque, V. (2009). *Virtual Display of Tactile Graphics and Braille by Lateral Skin Deformation.* (Ph.D., McGill University, Montréal, Québec, Canada).

Lévesque, V., & Hayward, V. (2008). *Tactile Graphics Rendering Using Three Laterotactile Drawing Primitives*. Colgate, J. E., Lederman, S., & Prattichizzo , D. *Proceedings of 16th Symposium on Haptic Interfaces For Virtual Environment And Teleoperator Systems (HAPTICS 2008)*, 429-436.

Logitech. (2013). Logitech's iFeel™ Optical Mice Add a New Dimension to the Human-Computer Interface. Consulté le 17 avril 2013 à http://www.logitech.com/en-roeu/press/press-releases/1183

Maberley, D. A. L., Hollands, H., Chuo, J., Tam, G., Konkal, J., Roesch, M., . . . Basset, K. (2006). The prevalence of low vision and blindness in Canada. *Eye, 20*(3), 341-346.

MacLean, K. E., & Hayward, V. (2008). Do It Yourself Haptics: Part II *IEEE Robotics & Automation Magazine, 15*(1), 104-119.

Mankoff, J., Dey, A., Batra, U., & Moore, M. (2002). *Web accessibility for low bandwidth input*. *Proceedings of the 5th international ACM conference on Assistive technologies*, Edinburgh, Scotland, 17-24.

Martini, F. H., & Nath, J. L. (2008). *Fundamentals of Anatomy & Physiology*. (8e éd.). Benjamin Cummings.

Massaro, D. W. (1998). *Illusions And Issues In Bimodal Speech Perception*. Burnham , D., Robert-Ribes, J., & Vatikiotis-Bateson, E. *Proceedings of Auditory Visual Speech Perception '98 (AVSP'98)*, Terrigal-Sydney, Australie, 21-26.

Maucher, T., Schemmel, J., & Meier, K. (2000). *The Heidelberg Tactile Vision Substitution System*. Vollmar, R. *Proceedings of the 6th International Conference on Computers Helping People with Special Needs*, Karlsruhe, Allemagne, 256-262.

Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review, 63*(2), 81-97.

Moore, M. (1989). Editorial: Three types of interaction. *American Journal of Distance Education, 3*(2), 1-7.

Mousty, P., & Bertelson, P. (1985). A study of braille reading: 1. Reading speed as a function of hand usage and context. *Q J Exp Psychol A, 37*(2), 217-233.

Napier, J. R. (1993). *Hands*. Princeton University Press.

National Center for Accessible Media. (2013). NCAM/Web Accessibility QA Favelet. Consulté le 17 avril 2013 à http://ncam.wgbh.org/webaccess/favelet/

Neto, A. T., Bittar, T. J., Fortes, R. P. M., & Felizardo, K. (2009). *Developing and evaluating web multimodal interfaces - a case study with usability principles*. Shin, S. Y., & Ossowski, S. *Proceedings of the 29th ACM Symposium on Applied Computing*, Honolulu, Hawaii, 116-120.

Nielsen, J. (1993). *Usability engineering*. Academic Press.

Nigay, L. (2004). Design Space for Multimodal Interaction (*Building the Information Society* (p. 403-408). Kluwer Academics.

Nigay, L., & Coutaz, J. (1995). *A generic platform for addressing the multimodal challenge*. Katz, I. R., Mack, R. L., Marks, L., Rosson, M. B., & Nielsen, J. *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI'95)*, Denver, Colorado, United States, 98-105.

Noldus. (2013). Behavioral research software and observation labs | The Observer XT. Consulté le 26 avril 2013 à http://www.noldus.com/human-behavior-research/products/the-observer-xt

Novint. (2013). Novint Falcon. Consulté le 17 avril 2013 à http://home.novint.com/index.php/products/novintfalcon

NV Access. (2013). NVDA. Consulté le 17 avril 2013 à http://community.nvda-project.org/

OATS. (2013). Gnopernicus - OATS. Consulté le 17 avril 2013 à http://www.oatsoft.org/Software/gnopernicus

Occelli, V., Spence, C., & Zampini, M. (2013). Auditory, tactile, and audiotactile information processing following visual deprivation. *Psychol Bull, 139*(1), 189-212.

Osborne, J. W. (2010). Improving Your Data Transformations: Applying the Box-Cox Transformation. *Practical Assessment, Research & Evaluation, 15*(12), 1-9.

Oviatt, S. L. (2003). Multimodal interfaces. Dans Jacko, J. A., & Sears, A. (dir.), *The human-computer interaction handbook : fundamentals, evolving technologies, and emerging applications* (p. 286-304). Lawrence Erlbaum Associates.

Oviatt, S. L., DeAngeli, A., & Kuhn, K. (1997). *Integration and synchronization of input modes during multimodal human-computer interaction*. Pemberton, S. *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI'97)*, Atlanta, Georgia, United States, 415-422.

Oviatt, S. L., & Kuhn, K. (1998). *Referential Features and Linguistic Indirection in Multimodal Language*. H. Mannell, R., & Robert-Ribes, J. *Proceedings of the 5th International Conference on Spoken Language Processing*, Sydney, Australia, 2339-2342.

Park University. (2013). Faculty Resources Quick Tips - Learning Objectives. Consulté le 17 avril 2013 à http://www.park.edu/cetl/quicktips/writinglearningobj.html

Pelli, D. G., Robson, J. G., & Wilkins, A. J. (1988). The design of a new letter chart for measuring contrast sensitivity. *Clinical Vision Science, 2*(3), 187-199.

Petit, G. (2007). *Transformation d'illustrations de manuels scolaires en images tactiles et sonores pour les non-voyants.* (Mémoire de Maîtrise, Université du Maine, Le Mans, France).

Petit, G., Dufresne, A., Lévesque, V., & Hayward, V. (2008). Exploration multimodale d'images pour des utilisateurs ayant une déficience visuelle. *Sciences et Technologies pour le Handicap, 2*, 175-186.

Petit, G., Dufresne, A., Lévesque, V., Hayward, V., & Trudeau, N. (2008). *Refreshable tactile graphics applied to schoolbook illustrations for students with visual impairment. Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*, Halifax, NS, Canada, 89-96.

PolymorpheDesign. (2009). Fiches techniques produits pour musées. Consulté le 17 novembre 2009 à http://www.polymorphe-design.fr/fiches-techniques.php

Prescher, D., Weber, G., & Spindler, M. (2010). *A tactile windowing system for blind users*. *Proceedings of the 12th international ACM SIGACCESS conference on Computers and accessibility*, Orlando, Florida, USA, 91-98.

QNX software developpers. (2013). Simple DirectMedia Layer. Consulté le 17 avril 2013 à http://www.libsdl.org/index.php

Raman, T. V. (2013). Emacspeak --The Complete Audio Desktop. Consulté le 17 avril 2013 à http://emacspeak.sourceforge.net/

RampWeb. (2013). Section 508 Toolbar - Test Your Web Site for Section 508 Compliance. Consulté le 17 avril 2013 à http://www.rampweb.com/Accessibility_Resources/Section508/

Ramstein, C., & Hayward, V. (1994). *The pantograph: a large workspace haptic device for multimodal human computer interaction*. Adelson, B., Dumais, S., & Olson, J. S. *Proceedings of Conference Companion on Human Factors in Computing Systems (CHI'94)*, Boston, Massachusetts, USA, 57-58.

Ramstein, C., Martial, O., Dufresne, A., Carignan, M., Chassé, P., & Mabilleau, P. (1996). *Touching and hearing GUI's: design issues for the PC-Access system*. *Proceedings of the 2nd Annual ACM Conference on Assistive Technologies (ASSETS'96)*, Vancouver, British Columbia, Canada, 2-9.

Révész, G. (1950). *Psychology and art of the blind*. (Traduit par Wolff, H. A.). Longmans, Green.

Röder, B., & Rösler, F. (2003). Memory for environmental sounds in sighted, congenitally blind and late blind adults: evidence for cross-modal compensation. *International Journal of Psychophysiology, 50*(1–2), 27-39.

Rosenblum, L. D. (2011). *See What I'm Saying: The Extraordinary Powers of Our Five Senses*. W. W. Norton.

Sabry, K., & Baldwin, L. (2003). Web-based learning interaction and learning styles. *British Journal of Educational Technology, 34*(4), 443-454.

Scharver, C., Evenhouse, R., Johnson, A., & Leigh, J. (2004). Designing cranial implants in a haptic augmented reality environment. *Commun. ACM, 47*(8), 32-38.

Schmitz, F. (2013). CSSTidy. Consulté le 17 avril 2013 à http://csstidy.sourceforge.net/

Secrétariat du Conseil du Trésor. (2013). Standard sur l'accessibilité du Web. Consulté le 17 avril 2013 à http://www.tresor.gouv.qc.ca/fileadmin/PDF/ressources_informationnelles/AccessibiliteWeb/access_web_ve.pdf

Section 508. (2013). Section 508 Standards Summary. Consulté le 17 avril 2013 à http://www.section508.gov/index.cfm?fuseAction=stdsSum

Seiderman, A., Marcus, S. E., & Hapgood, D. (1989). *20/20 is not enough: the new world of vision*. Knopf.

Sensable. (2013). PHANTOM Omni Sensable. Consulté le 17 avril 2013 à http://www.sensable.com/haptic-phantom-omni.htm

Sherrington, C. S. (1906). *The integrative action of the nervous system*. Constable.

Shinohara, M., Shimizu, Y., & Mochizuki, A. (1998). Three-dimensional tactile display for the blind. *IEEE Transactions on Rehabilitation Engineering, 6*(3), 249-256.

Smith, D. C., Irby, C., Kimball, R., Verplank, B., & Harslem, E. (1989). Designing the Star user interface (1982) (*Perspectives on the computer revolution* (p. 261-283). Ablex Publishing Corp.

Sony. (2013). DUALSHOCK®3 Wireless Controller For The PS3™ System - PlayStation®3 Accessories. Consulté le 17 avril 2013 à http://us.playstation.com/ps3/accessories/dualshock-3-wireless-controller-ps3.html

Source Forge. (2013). eSpeak: Speech Synthesizer. Consulté le 17 avril 2013 à http://espeak.sourceforge.net/

Stanciulescu, A., Limbourg, Q., Vanderdonckt, J., Michotte, B., & Montero, F. (2005). *A Transformational Approach for Multimodal Web User Interfaces Based on UsiXML*. Lazzari, G., Pianesi, F., Crowley, J. L., Mase, K., & Oviatt, S. L. *Proceedings of the 7th International Conference on Multimodal Interfaces*, Trento, Italy.

Standardization, I. O. f. (Writer). (1998). Ergonomic requirements for office work with visual display terminals (VDTs) -- Part 11: Guidance on usability.

Stevens, J. C., Foulke, E., & Patterson, M. Q. (1996). Tactile acuity, aging, and braille reading in long-term blindness. *Journal of Experimental Psychology: Applied, 2*(2), 91-106.

TactileLabs. (2013). Latero. Consulté le 17 avril 2013 à http://www.tactilelabs.com/products/haptics/latero-tactile-display/

Thatcher, J. (2006a). Accessible Content. Dans Thatcher, J. (dir.), *Web accessibility : web standards and regulatory compliance* (p. 103-124). Friends of ED.

Thatcher, J. (2006b). Assistive technology: screen readers and browsers. Dans Thatcher, J. (dir.), *Web accessibility : web standards and regulatory compliance* (p. 103-124). Friends of ED.

Thrustmaster. (2013). Thrustmaster - Racing Wheels. Consulté le 17 avril 2013 à http://www.thrustmaster.com/fr_FR/produits/categories/volants

Twotoasts. (2013). Midori. Consulté le 17 avril 2013 à http://twotoasts.de/index.php/midori/

Vision Australia. (2013). Web Accessibility Toolbar for IE - 2012. Consulté le 17 avril 2013 à http://www.visionaustralia.org/business-and-professionals/digital-access/resources/tools-to-download/web-accessibility-toolbar-for-ie---2012

W3C. (2013a). HTML5 Definition Complete, W3C Moves to Interoperability Testing and Performance. Consulté le 17 avril 2013 à http://www.w3.org/2012/12/html5-cr

W3C. (2013b). HTML Reference - (HTML5 Compliant). Consulté le 17 avril 2013 à http://www.w3schools.com/tags/default.asp

W3C. (2013c). Web Content Accessibility Guidelines 1.0. Consulté le 17 avril 2013 à http://www.w3.org/TR/WAI-WEBCONTENT/

W3C. (2013d). Web Content Accessibility Guidelines (WCAG) 2.0. Consulté le 17 avril 2013 à http://www.w3.org/TR/WCAG20/

Wang, Q., & Hayward, V. (2006). *Compact, Portable, Modular, High-performance, Distributed Tactile Transducer Device Based on Lateral Skin Deformation*. *Proceedings of the Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTICS'06)*, Arlington, VA, USA, 67-72.

Wang, Q., Lévesque, V., Pasquero, J., & Hayward, V. (2006). *A haptic memory game using the STRESS² tactile display*. Olson, G. *Proceedings of CHI EA '06 Extended abstracts on Human factors in computing systems*, Montréal, Québec, Canada, 271-274.

Warren, D. H. (1978). Perception by the Blind. Dans Carterette, E. C., & Friedman, M. P. (dir.), *Handbook of perception* (Vol. 10, p. 65-86). Academic Press.

Web AIM. (2013). Screen Reader User Survey Results. Consulté le 17 avril 2013 à http://webaim.org/projects/screenreadersurvey2/

Weber, E. H. (1978). *The sense of touch [De Tactu]*. (Traduit par Ross, H. E., & Murray, D. J.). London Academic Press for Experimental Psychology Society.

Webster, M. (1985). *Webster's Ninth New Collegiate Dictionary*. (9e éd.). Merriam-Webster Inc.

Wikipedia. (2013). Lynx (web browser). Consulté le 17 avril 2013 à http://en.wikipedia.org/wiki/Lynx_(web_browser)

WingMan Team. (2013). Home of the WingMan Team. Consulté le 17 avril 2013 à http://www.wingmanteam.com/index.htm

Woodworth, R. S. (1899). The accuracy of voluntary movement. *Psychological Review, 26*(12), 743-752.

World Health Organization. (2013). MOS | Maladies oculaires prioritaires. Consulté le 17 avril 2013 à http://www.who.int/blindness/causes/priority/fr/index.html

ANNEXE 1 – LISTE DES TACTOELEMENT

- TactoBody : correspond à la balise <body>. À l'exception de TactoRoot, chaque autre type de TactoElement l'a comme ancêtre.
- TactoRoot : c'est le parent de TactoBody, soit la racine de la page affichée dans TactoWeb. C'est le seul objet qui n'a pas de parent. Il a été créé seulement pour que TactoBody ait un parent car un élément Web doit obligatoirement avoir un parent pour être traité dans TactoWeb.
- TactoDiv : correspond à la balise . Il représente une division de la page Web. Les divisions sont essentielles pour correctement segmenter une page Web.
- TactoParagraph : correspond à la balise <p> qui représente un paragraphe.
- TactoSpan : correspond à la balise . C'est la même chose qu'un paragraphe à l'exception de son affichage par défaut qui est « en ligne » alors que le paragraphe est « en bloc ».
- TactoLink : correspond à la balise <a> qui représente un hyperlien. Il possède aussi un attribut spécifique, « href », qui correspond à l'URL cible de l'hyperlien.
- TactoHeader : correspond aux balises <h1>, <h2> … <h6> qui représentent les six niveaux de titre supportés en HTML. Le niveau du titre est stocké dans un attribut car il peut influer sur la taille de police de caractères.
- TactoImage : correspond à la balise <img> qui représente une image dans une page Web. On récupère en plus le texte alternatif de l'image (attribut alt). Si le texte alternatif est vide et que l'image est aussi un lien, le texte alternatif prend le nom du lien sans son extension pour être lu plus tard par la synthèse vocale. De plus, l'image est stockée en local pour accéder à ses informations de dimensions. Un TactoImage ne peut être le parent d'aucun autre objet TactoElement.
- TactoList : correspond aux balises <ol> et <ul> qui représentent des listes numérotées et des listes à puce.

- TactoListElement : correspond à la balise <li> qui représente un élément de liste appartenant à un objet TactoList.
- TactoBlockquote : correspond à la balise <blockquote> qui représente une citation. La particularité d'une citation par rapport à un paragraphe est qu'elle possède une marge à gauche. Certains développeurs s'en servent d'ailleurs pour faire de la mise en page alors que ce n'est pas sa fonction première.
- TactoTable : correspond à la balise <table> qui représente un tableau. Cet objet possède comme attribut le nombre de lignes du tableau.
- TactoTableRow : correspond à la balise <tr> qui représente une ligne de tableau. Cet objet possède comme attribut le nombre de cellule que contient cette ligne de tableau. La largeur de chacune de ces cellules est aussi stockée dans cet objet sous forme de vecteur dans le but de faciliter le calcul la largeur de chaque colonne du tableau.
- TactoTableData : correspond aux balises <td> et <th> qui représentent des cellules de données et des cellules d'en-tête d'un tableau.
- TactoText : ne correspond à aucune balise. Cet objet est créé quand le contenu d'une balise n'est que du texte, donc que cette balise ne contient pas d'autres balises. Donc un objet TactoText ne peut être le parent d'un autre objet TactoElement. Le contexte du TactoText est indiqué par ses ancêtres. Par exemple, si un des ces ancêtres est un objet TactoLink, l'objet TactoText se rapporte à un hyperlien.
- TactoButton : correspond à la balise <button> qui représente un bouton déclenchant généralement des scripts.
- TactoForm : correspond à la balise <form> qui représente un formulaire. Tout champ de formulaire est sensé avoir un TactoForm parmi ses ancêtres.
- TactoInput : correspond à la balise <input> qui représente les champs d'entrée d'information par l'utilisateur. Le type d'entrée change en fonction de la valeur de son attribut type qui est stockée dans le TactoInput. Les types d'entrée supportés par TactoWeb sont les suivants :
    - text : champs de saisie. L'utilisateur doit saisir du texte au clavier.

- o password : mot de passe. Identique à text sauf que le système remplace visuellement le texte entré par des étoiles.
- o checkbox : case à cocher. L'utilisateur peut choisir plusieurs cases parmi un même groupe de cases à cocher.
- o radio : bouton radio. L'utilisateur ne peut choisir qu'un bouton parmi un même groupe de boutons radio.
- o submit : bouton de soumission. Le formulaire est envoyé quand l'utilisateur clique sur ce bouton.
- o button : bouton ne permettant pas de soumettre le formulaire.
- o image : bouton de soumission de formulaire présenté sous forme d'image.
- o reset : bouton de réinitialisation du formulaire.
- o hidden : champs de formulaire caché. Sa valeur est prédéfinie par le système et l'utilisateur ne peut pas l'éditer.

En plus du type d'entrée, TactoWeb récupère les attributs suivants pour les TactoInput :

- o value : indique la valeur par défaut des champs de saisie, et le texte écrit dans les boutons.
- o size : indique la largeur visible des champs de texte.
- o maxlength : indique le nombre maximum de caractères qu'il est possible d'entrer dans un champs de saisie.
- o checked : indique si une case à cocher ou un bouton radio est sélectionné ou non par défaut.
- o disabled : indique si un champ de formulaire est activé ou non. S'il n'est pas activé, l'utilisateur ne peut pas le remplir.

- TactoTextArea : correspond à la balise <textarea> qui représente un champ de saisie de texte sur plusieurs lignes. Les attributs « cols » et « rows » sont récupérés et représentent la largeur et la hauteur visibles du TactotextArea.

- TactoSelect : correspond à la balise <select> qui représente une liste de sélection parmi plusieurs valeurs. Cette sélection peut être simple ou multiple (attribut « multiple »). Le second attribut récupéré est « size » qui correspond au nombre de valeurs visibles à l'écran. Les valeurs de la liste de sélection sont codées sous la forme de balises <option> à l'intérieur de la balise <select>. Ces valeurs sont stockées dans un vecteur, dans l'objet TactoSelect.
- TactoFieldset : correspond à la balise <fieldset> qui représente un groupe de champs de formulaire. On peut l'interpréter comme l'équivalent de la balise mais pour les formulaires. On peut trouver une balise <legend> dans une balise <fieldset> qui correspond à la description du groupe de champ (ex. : « adresse » regroupant tous les champs relatifs à l'adresse postale d'une personne). Cette information est stockée dans l'objet TactoFieldset.
- TactoLabel : correspond à la balise <label> qui représente l'étiquette d'un champ de formulaire. Si la valeur de l'attribut « for » d'un TactoLabel est la même que la valeur de l'attribut « id » d'un champ de formulaire TactoInput, TactotextArea ou TactoSelect, l'étiquette et le champ de formulaire sont associés.
- TactoIgnore : correspond aux balises <script> et <style>. Même si elles ne sont pas interprétées par TactoWeb, il est nécessaire de les stocker pour ne pas briser le traitement de DOMParser. Elle ne se rapporte pas au contenu mais à la mise en page ou aux fonctions dynamiques.

ANNEXE 2 – TUTORIEL TACTOWEB

Maintenant, nous allons utiliser un autre outil pour naviguer sur le Web. Cet outil s'appelle TactoWeb **[ou]** Nous allons utiliser en premier un outil de navigation Web appelé TactoWeb.

TactoWeb permet de naviguer dans un site Web en passant d'une page à l'autre. Contrairement à JAWS, TactoWeb fonctionne avec une souris un peu spéciale qui produit des vibrations et des textures tactiles. Quand vous passez au dessus d'un élément spécifique de la page Web, vous entendrez un son et/ou un texte associé à cet élément. Vous sentirez aussi généralement quand vous entrerez dans cet élément à l'aide d'une sensation tactile. Par exemple, chaque fois que vous passerez au dessus d'un lien avec la souris, vous sentirez une vibration sous votre doigt, mais vous entendrez aussi ce son **[jouer son]** suivi du texte du lien. Le mieux est qu'on suive un petit tutoriel tous les deux pour que vous vous familiarisiez avec TactoWeb. On va y aller tranquillement ensemble, étape par étape.

1. Au cours de ce tutoriel, vous trouverez les éléments Web suivants :
   a. Texte classique
   b. Liens : un hyperlien permet d'accéder à une autre page Web
   c. Des titres : un titre permet de savoir le contexte des éléments qui sont en dessous de ce titre.
   d. Des listes : des textes qui se rapportent à un même sujet
   e. Des images
2. Première chose. Je vais vous demander de prendre la souris de l'appareil avec votre main droite pour vous familiariser avec sa taille et pour que vous puissiez trouver facilement.
   a. Si vous mettez votre index au centre de la souris, vous sentirez un carré avec des petits points. Il s'agit de la cellule tactile qui va générer les sensations tactiles par la suite.
   b. À gauche de la cellule tactile, vous avez un bouton qui permettra de cliquer sur les liens par la suite et d'accéder aux autres pages Web.
   c. À droite de la cellule tactile, vous avez un autre bouton qui vous permettra de retourner à la page précédente. Ce bouton est inactif sur la première page où vous commencerez chaque tâche.
3. Avec cette souris, vous pouvez vous déplacer sur une surface fixe. C'est comme si l'écran de l'ordinateur était projeté sur cette surface, et vous explorez cet écran avec la souris par dessus. Je vous laisse faire le tour de cette surface avec la souris pour vous familiariser.
4. **[Apprendre à se déplacer droit].**
5. Pour chaque tâche, je vais vous indiquer le but de la tâche et vous indiquer sur quelle page vous allez commencer.
6. Avant de commencer une tâche, je vais vous demander de vous placer en haut à gauche avec la souris. Quand on explore un site avec une souris, on commence généralement à cette position. On explore ensuite de gauche à droite, ou de haut en bas. On ne va jamais en diagonale.

Page de tutoriel

7. Donc pour ce tutoriel, vous êtes sur la page d'accueil de mon site personnel.
8. Dans cette page, vous trouverez des liens, des listes, une image et des titres. Vous allez les rencontrer au fur et à mesure de votre exploration.
9. Dans la majorité des sites, les informations sont organisées de manière verticale, donc depuis là où vous êtes, en haut à gauche, je vais vous demander de descendre tranquillement avec la souris.
10. **[Arrêt lors de la rencontre avec le 1er élément du menu de gauche]** OK. Que venez-vous d'entendre et de sentir?
    a. **[Si pas remarquer l'icône sonore du lien]** Remontez légèrement et redescendez pour rentrer de nouveau dans la même zone et essayer d'écouter s'il n'y a rien d'autre avant le texte « Accueil ».
    b. **[Si pas remarquer le retour tactile]** Remontez légèrement et redescendez pour rentrer de nouveau dans la même zone et observer si vous sentez quelque chose de spécial sous votre doigt.
11. Cette technique de remonter légèrement puis redescendre vous permet de réécouter les sons et les textes que vous n'auriez pas correctement entendus la première fois que vous seriez rentrer dans un élément avec la souris. Cette technique vous sera très utile pour la suite.
12. Le petit son que vous avez entendu avant le texte « Accueil » vous donne la signification du texte dans lequel vous entrez. Dans ce cas ci il s'agit d'un lien. **[jouer et rejouer le son à la demande du participant]** Le texte « Accueil » vous donne la signification du lien. Pour chaque lien, vous pouvez cliquer dessus avec le bouton de gauche pour ouvrir une nouvelle page. Pour ce tutoriel, je vous demanderai de ne cliquer sur aucun lien.
13. Comme pour la zone de texte « Accueil », vous sentirez toujours une vibration quand vous rentrez dans une zone de texte et vous entendrez la même voix vous lire ce texte. Pour pourrez aussi entendre des autres sons avant la voix, comme celui que vous venez d'entendre pour le lien.
14. Je vous laisse continuer à descendre avec la souris pour découvrir les autres zones de texte. Au début, il est conseillé de s'arrêter dès que vous rencontrez un élément pour entendre entièrement le texte et le son associé à l'élément.
15. **[Une fois arrivée tout en bas à l'image]** Au cours de votre descente, qu'avez-vous identifié?
    a. **[Parler de l'image si pas identifiée]** Est-ce que l'élément que vous avez senti en dernier était différent des autres? Il s'agit d'une image. Si vous retournez dedans, vous sentirez une texture un peu spéciale. Le texte que vous entendez quand vous rentrez dans une image est une description de l'image. Parfois, il n'y a pas de description de l'image.
16. Vous avez correctement exploré la partie gauche de la page, je vous laisse maintenant explorer librement le reste de la page en gardant les mêmes techniques d'exploration, de haut en bas, ou de gauche à droite.
    a. **[Quand liste identifiée, demander si le son est correctement identifié et si besoin de le rejouer]** Le son que vous avez entendu avant le texte vous permet de savoir que l'élément fait partie d'une liste. Dans ce cas, le mieux est de trouver le premier élément de la liste. Pour cela, vous devez remonter avec la souris. Remontez jusqu'à ce que vous rencontriez un élément qui ne fait pas partie d'une liste, donc un élément

qui ne commence pas par le même son ou qui n'aura pas de son. [Attendre que cela soit fait] Cela veut dire que vous venez de sortir de la liste et que le premier élément de la liste est celui que vous aviez rencontré juste avant. Redescendez au premier élément de la liste. Vous pouvez maintenant descendre jusqu'en bas de la liste en vous arrêtant à chaque élément pour les identifier. Une fois que vous rencontrez un élément qui n'est pas un élément de liste, donc un élément ne commençant pas par le son d'un élément de liste, cela veut dire que la liste est terminée. Continuez d'explorer.

b. **[Quand titre identifié, demander si le son est correctement identifié et si besoin de le rejouer]** Le son que vous avez entendu avant le texte vous permet de savoir que l'élément est un titre. Un titre est généralement en haut d'une section. Cette section contient généralement d'autres éléments comme des liens, des listes, du texte basique, voire d'autres titres pour des sous-sections. Le titre permet donc de savoir le contexte des éléments qui sont en dessous de ce titre.

c. **[Si quelques problèmes]** Globalement, lorsqu'on trouve un élément, on remonte généralement pour identifier le premier élément d'une même famille, puis on redescend.

17. Vous comprenez que pour identifier des éléments, et voir comment des éléments sont reliés les uns aux autres, il faut se déplacer de haut en bas, et aussi de gauche à droite.
18. Pensez aussi que si vous sentez une zone mais que vous n'entendez rien, c'est que vous êtes en train de la frôler. Vous pouvez aller dans la direction de la sensation tactile pour entendre le son et le texte associé à la zone.
19. Les sensations vous aident aussi à vous déplacer tout droit.
20. **[Quand page explorée totalement]** Passons maintenant à une autre page. Pour ce faire, je vais vous demander de retrouver le lien « Questionnaire» et de cliquer dessus.
21. Avez-vous remarqué le son que cela a fait lorsque vous avez cliqué? **[rejouer le son si besoin]** Si vous n'entendez pas ce son, le clic n'a pas fonctionné.

Tutoriel Formulaire

22. Dans un formulaire, vous allez trouver :
    a. Des zones de texte
    b. Des listes de sélection
    c. Des cases à cocher
    d. Des boutons radio
    e. Des boutons de soumission de formulaire
23. **[Une fois sur la page du formulaire]** Quand vous naviguez sur un site Web, vous allez vous rendre compte que certains éléments restent les mêmes de page en page. C'est souvent le cas pour les menus ou les hauts de page par exemple.
24. Pour ce questionnaire, vous allez retrouver le même menu à gauche, ainsi que mes éléments de contacts que vous aviez identifiés à droite sur la page précédente.
25. Je vous laisse donc voir ce qui a changé sur cette page par rapport à la page précédente.
26. **[Une fois un élément du questionnaire rencontré]** Que venez-vous d'entendre?
    a. **[Si pas correctement entendu ou si c'est faux]** Faites la technique que nous avons apprise tantôt en remontant légèrement puis en redescendant dans le même élément.
    b. **[Si champs du formulaire]** Vous venez de rencontrer un champ de formulaire.

i. **[Si passé par c.]** La dernière chose que vous entendez est le rappel de l'étiquette associée au formulaire. Si vous ne l'entendez pas, c'est juste que l'association entre l'étiquette et le champ n'a pas été faite. Dans ce cas, vous pouvez faire vous même cette association car vous venez d'apprendre que l'étiquette et le champ de formulaire associé étaient toujours l'un à côté de l'autre, ou l'un en dessous de l'autre. **[Aller à iii.]**

ii. **[Si pas passé par c.]** Chaque champ de formulaire est associé à une étiquette qui permet de le désigner. La dernière chose que vous allez entendre quand vous allez rentrer dans le champ de formulaire, c'est son étiquette (c'est-à-dire sa désignation). Ici, c'était (dire l'étiquette entendue) Il arrive parfois qu'aucune étiquette ne soit associée au champ de formulaire. Vous pouvez le trouver avec la souris. Généralement, les étiquettes sont à gauche, ou parfois en haut, du champ de formulaire. Essayez de trouver cette étiquette. **[Une fois trouvé, si pas correctement entendu aller à a, sinon aller à c.]** Revenez maintenant au champ de formulaire.

iii. Vous venez de rencontrer un champ de formulaire. Un formulaire contient plusieurs champs. Ces champs vous permettre de saisir ou de sélectionner des informations (par exemple votre âge). Vous devez d'abord identifier quel est le type de champ de formulaire. Ici vous venez d'entendre…

   1. « Entrez du texte ». Cela veut dire que vous devez normalement saisir une information. Cette information se rapporte à l'étiquette que vous avez entendue après. Pour le besoin de ce test, dites-moi seulement à l'oral ce que vous auriez saisi quand vous êtes sur ce champ de formulaire.
   2. « bouton radio (non) sélectionné ». Un bouton radio veut dire que vous devez sélectionner une valeur parmi plusieurs boutons radio. Chaque bouton radio a une valeur qui est aussi à droite de ce bouton. Les différents boutons radio sont généralement les uns à côté des autres de gauche à droite ou de haut en bas. Il y a une étiquette qui désigne l'ensemble des boutons radio. Comme n'importe quelle étiquette, celle-ci est à gauche ou au dessus de l'ensemble de boutons radio. Pour le besoin de ce test, dites-moi seulement quand vous voulez sélectionner un bouton radio lorsque vous êtes sur le bouton.
   3. « case à cocher (non) sélectionnée ». Une case à cocher veut dire que vous pouvez sélectionner plusieurs cases parmi plusieurs cases à cocher. Chaque case à cocher a une valeur qui est aussi à droite de cette case. Les différentes cases à cocher sont généralement les unes à côté des autres de gauche à droite ou de haut en bas. Il y a une étiquette qui désigne l'ensemble des cases à cocher. Comme n'importe quelle étiquette, celle-ci est à gauche ou au dessus de l'ensemble à cocher. Pour le besoin de ce test, dites-moi seulement quand vous voulez sélectionner une case à cocher lorsque vous êtes sur la case.
   4. « Choisir X éléments parmi les X éléments possibles». Il s'agit d'une liste de choix qui vous sont lus les uns à la suite des autres. Chaque choix est précédé d'un numéro. Pour le besoin de ce test, dites-moi juste votre ou vos choix parmi les éléments de la liste quand vous êtes sur la liste.

5. « soumettre le formulaire ». Il s'agit d'un bouton qui vous permet de soumettre le formulaire. Il suffit de cliquer dessus quand vous pensez que vous avez rempli tous les champs du formulaire.

c. Le son que vous avez entendu avant le texte vous permet d'identifier que c'est une étiquette. **[Rejouer le son si besoin]**
   i. **[Si passé par b.]** Ce que vous venez d'entendre est la désignation du champ où vous étiez avant. Parfois, il n'y a pas d'étiquette lorsque vous rencontrez un champ de formulaire. Vous savez maintenant à quoi correspond ce champ vu que vous avez trouvé l'étiquette qui était à côté de celui-ci. Les étiquettes sont toujours à gauche ou en haut d'un champ. **[retourner à b iii]**
   ii. **[Si pas passé par b.]** Un formulaire contient plusieurs champs. Ces champs vous permettent de saisir ou de sélectionner des informations (par exemple votre âge). Chaque champ de formulaire est associé à une étiquette qui permet de le désigner. L'étiquette vous permet de savoir à quoi se rapporte le champ de formulaire qui est à côté de lui. Généralement, ce champ de formulaire est à sa droite, voire parfois en dessous. Trouvez ce champ de formulaire **[Une fois trouvé, allez à b.]**

d. Remplissez maintenant tout le formulaire. Il y a plusieurs types de champs. **[Refaire à chaque fois qu'on rencontre un autre champ de formulaire mais on peut aller plus rapidement, sauf pour expliquer le type de champ.]**

27. Soumettez le formulaire.
28. Retournez à la page d'accueil.

Tâches de contrôle

29. Je vais vous poser maintenant quelques questions. Même si vous vous rappelez de la réponse, je vous demanderai de retrouver cette réponse en utilisant TactoWeb.
    a. 1er ensemble
       i. Quels sont mes intérêts de recherche?
       ii. À quel numéro peut-on me joindre?
       iii. Parmi combien de pizzas pouvez-vous choisir votre pizza préférée?
    b. 2ème ensemble [à faire seulement si le 1er ensemble ne se passe pas bien et que le participant a eu besoin de mon aide].
       i. Trouvez le lien menant à mes publications (mais ne cliquez pas dessus)
       ii. En quelle année ai-je travaillé sur le projet « Logiciels éducatifs pour enfants ayant une déficience cognitive » ?
       iii. Quelle est la deuxième case à cocher parmi les lecteurs d'écran qui vous sont proposés dans le questionnaire?
30. Pour ce tutoriel, les pages étaient faites d'une certaine façon. Au cours des tâches que vous allez devoir accomplir, la structure des pages ne sera pas la même. Mais vous savez maintenant quelle stratégie d'exploration utiliser pour naviguer dans une page Web avec TactoWeb. Je vous ai expliqué plusieurs techniques pour explorer. Libre à vous de les utiliser ou d'en utiliser d'autres que vous auriez trouvées vous-même et que vous trouveriez meilleures. De plus, vous n'êtes pas obligé d'explorer les pages en entier si vous pensez avoir trouvé l'information que vous souhaitez avant.

Êtes-vous prêt à commencer les tâches avec TactoWeb?

ANNEXE 3 – ANNONCE DE RECRUTEMENT

Participants non-voyants recherchés pour une expérience sur la navigation sur le Web

Nous sommes à la recherche de participants non-voyants pour une expérience sur la navigation sur le Web. Vous serez appelé à utiliser un appareil tactile et sonore ainsi qu'un logiciel de lecture d'écran pour accomplir des tâches de navigation et d'exploration de pages Web. L'expérience est sécuritaire et confortable. Mais comme mesure de précaution, les personnes ayant des problèmes cardiaques ou un stimulateur cardiaque ne sont pas autorisées à participer à l'expérience. Une bonne perception tactile est aussi exigée. L'expérience dure environ 2 heures et prendra place à l'Université de Montréal ou l'Institut Nazareth et Louis Braille. Une montant de 45$ vous sera versé. N'hésitez pas à nous contacter pour avoir plus d'information.

Grégory Petit
gregory.petit@polymtl.ca
(438) 887-4416

Superviseure: Prof. Aude Dufresne
aude.dufresne@umontreal.ca
(514) 343-6111 (poste 55511)

CITÉ de l'Université de Montréal
Département de communication
C.P. 6128, succursale Centre-ville
Montréal, QC H3C 3J7

ANNEXE 4 – CERTIFICAT D'ÉTHIQUE DE RECHERCHE

Comité d'éthique de la recherche
des établissements du CRIR

CRIR

Certificat d'éthique

Par la présente, le comité d'éthique de la recherche des établissements du CRIR (CÉR) atteste qu'il a évalué, par voie accélérée, le projet de recherche **CRIR-690-0212** intitulé :

« Exploration tactile et sonore du Web à l'usage des personnes non-voyantes ».

Présenté par : **Grégory Petit**

Le présent projet répond aux exigences éthiques de notre CÉR. Le Comité autorise donc sa mise en œuvre sur la foi des documents suivants :

- Formulaire A daté du 23 février 2012 ;
- Lettre et grille du Comité d'évaluation scientifique conjoint CRIR/CIIRIS, datée du 21 mars 2012, attestant de la rigueur scientifique du projet ;
- Formulaire d'évaluation de l'Institut Nazareth et Louis-Braille, daté du 9 mai 2012, attestant que l'établissement accueille favorablement le projet sur le plan de la convenance institutionnelle.
- Protocole de recherche (version du 18 mai 2012) ;
- Liste sélective des références relatives au projet ;
- Budget
- Formulaire de consentement (version du 18 mai 2012) ;
- Annonce de recrutement (version du 18 mai 2012) ;
- Document intitulé « Question de recrutement » ;
- Curriculum vitae de Grégory Petit.

Ce projet se déroulera dans le site du CRIR suivant : **Institut Nazareth et Louis-Braille.**

Ce certificat est valable pour un an. En acceptant le présent certificat d'éthique, le chercheur s'engage à :

1. Informer, dès que possible, le CÉR de tout changement qui pourrait être apporté à la présente recherche ou aux documents qui en découlent (Formulaire M) ;
2. Notifier, dès que possible, le CÉR de tout incident ou accident lié à la procédure du projet ;
3. Notifier, dès que possible, le CÉR de tout nouveau renseignement susceptible d'affecter l'intégrité ou l'éthicité du projet de recherche, ou encore, d'influer sur la décision d'un sujet de recherche quant à sa participation au projet ;
4. Notifier, dès que possible, le CÉR de toute suspension ou annulation d'autorisation relative au projet qu'aura formulée un organisme de subvention ou de réglementation ;
5. Notifier, dès que possible, le CÉR de tout problème constaté par un tiers au cours d'une activité de surveillance ou de vérification, interne ou externe, qui est susceptible de remettre en question l'intégrité ou l'éthicité du projet ainsi que la décision du CÉR ;

6. Notifier, dès que possible, le CÉR de l'interruption prématurée, temporaire ou définitive du projet. Cette modification doit être accompagnée d'un rapport faisant état des motifs à la base de cette interruption et des répercussions sur celles-ci sur les sujets de recherche ;
7. Fournir annuellement au CÉR un rapport d'étape l'informant de l'avancement des travaux de recherche (formulaire R) ;
8. Demander le renouvellement annuel de son certificat d'éthique ;
9. Tenir et conserver, selon la procédure prévue dans la *Politique portant sur la conservation d'une liste des sujets de recherche*, incluse dans le cadre réglementaire des établissements du CRIR, une liste des personnes qui ont accepté de prendre part à la présente étude ;
10. Envoyer au CÉR une copie de son rapport de fin de projet / publication.

Me Michel T. Giroux
Président du CÉR

Date d'émission
18 mai 2012

ANNEXE 5 – QUESTIONNAIRE DE RECRUTEMENT

1. Sexe : ☐ Homme ☐ Femme

2. Âge : ☐ < 18 ☐ 18 – 25 ☐ 26 – 30 ☐ 31 – 40 ☐ 41 – 50
   ☐ 51 – 60 ☐ > 60

3. Quelle est la nature de votre handicap visuel ? (naissance ou tardive, cause)

4. Lisez-vous le Braille? Oui Non

5. Utilisez-vous un logiciel de lecture d'écran ? (exemples : JAWS, Window Eyes, NVDA)
   Oui Non

6. Si oui, depuis combien de temps ? ☐ < 1 an ☐1 – 3 ans ☐4 – 6 ans ☐> 6 ans

7. Et combien de temps l'utilisez-vous par semaine?
   ☐ 1j/s ☐ 2-3j/s ☐ 4-5j/s ☐ 6-7j/s

8. Si oui, utilisez-vous les raccourcis clavier pour naviguer entre les différents liens ou en-têtes par exemple?

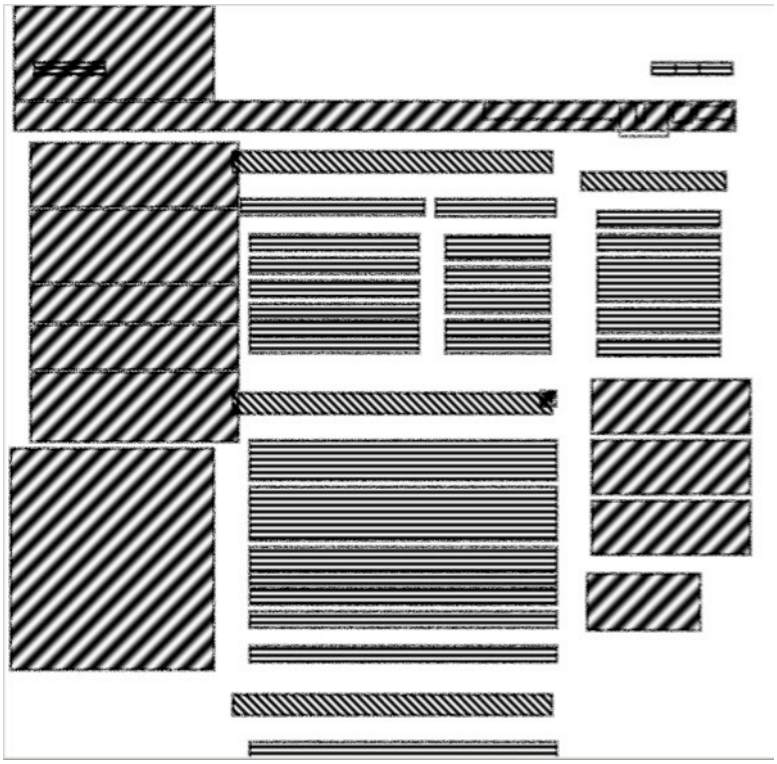
9. Combien de temps par semaine utilisez-vous votre ordinateur? h/semaine

10. Quelle est votre occupation principale?

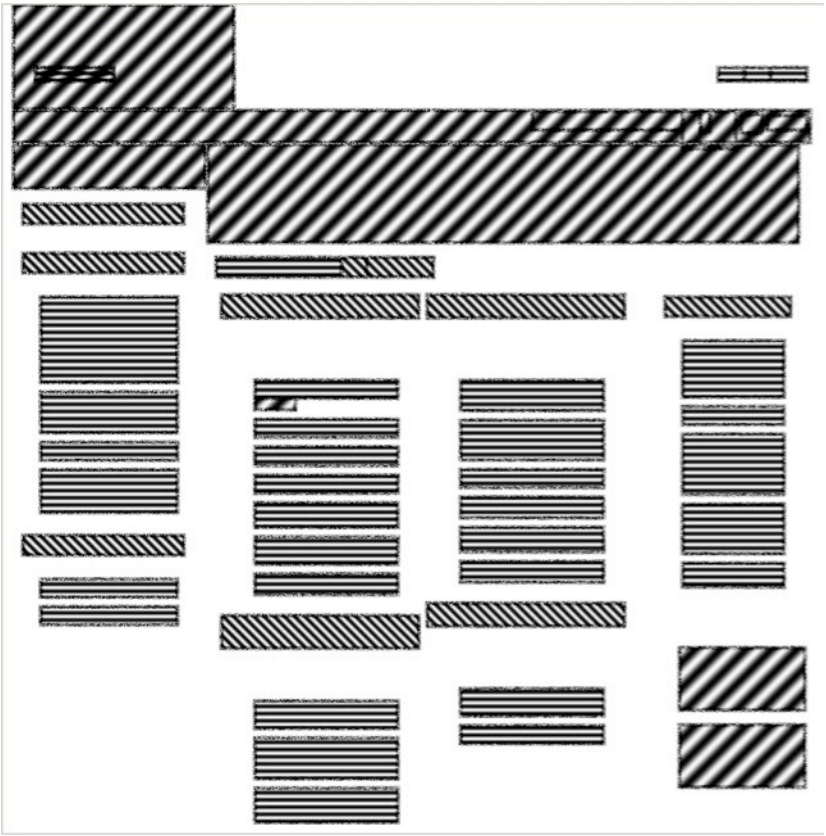
11. Quel est votre niveau de formation académique?

ANNEXE 6 – IMPRESSION D'ÉCRAN ET RENDUS DANS TACTOWEB POUR LES TÂCHES EXPÉRIMENTALES

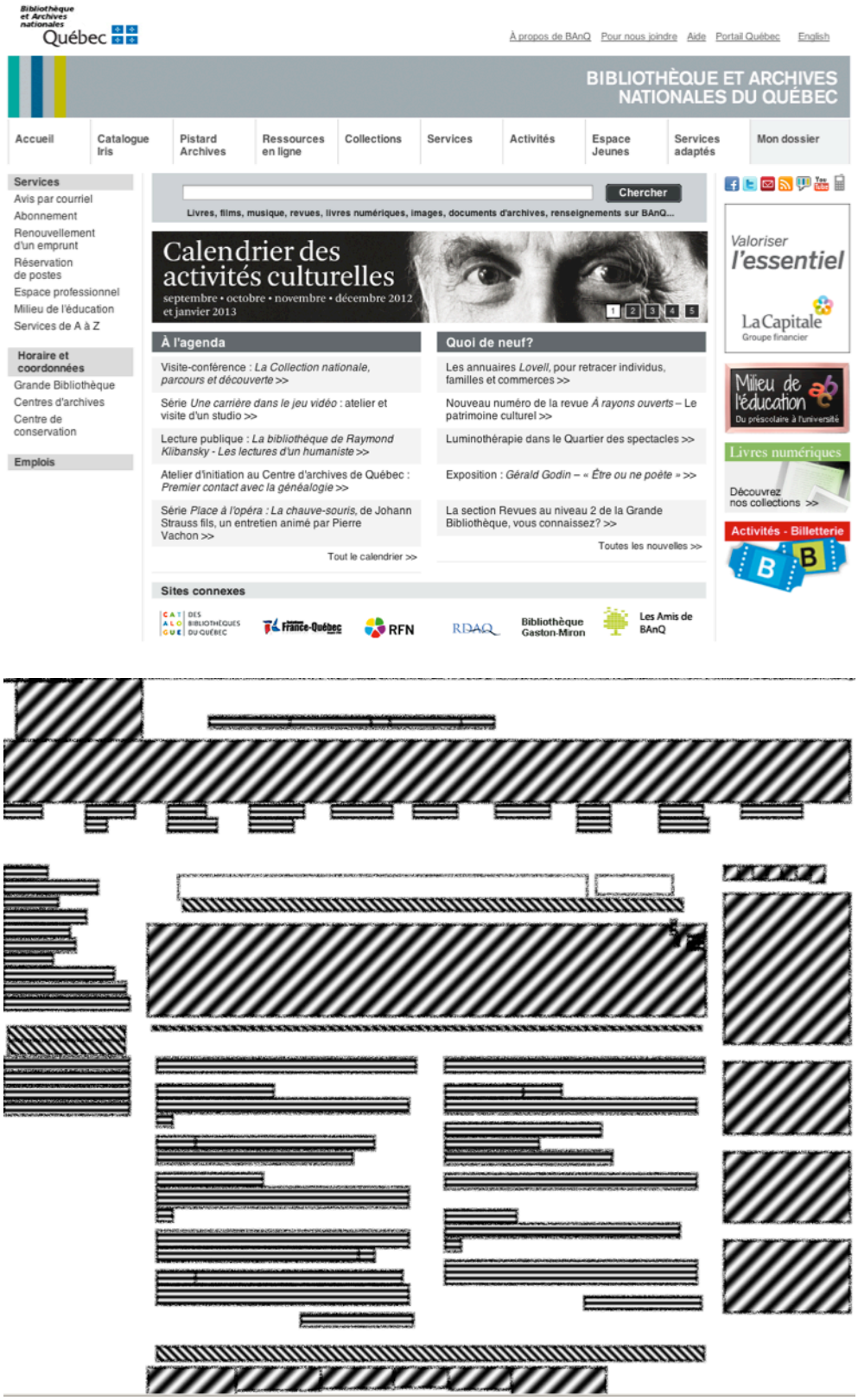
S1 : Gouvernement du Québec – Déménagement – Page d'accueil



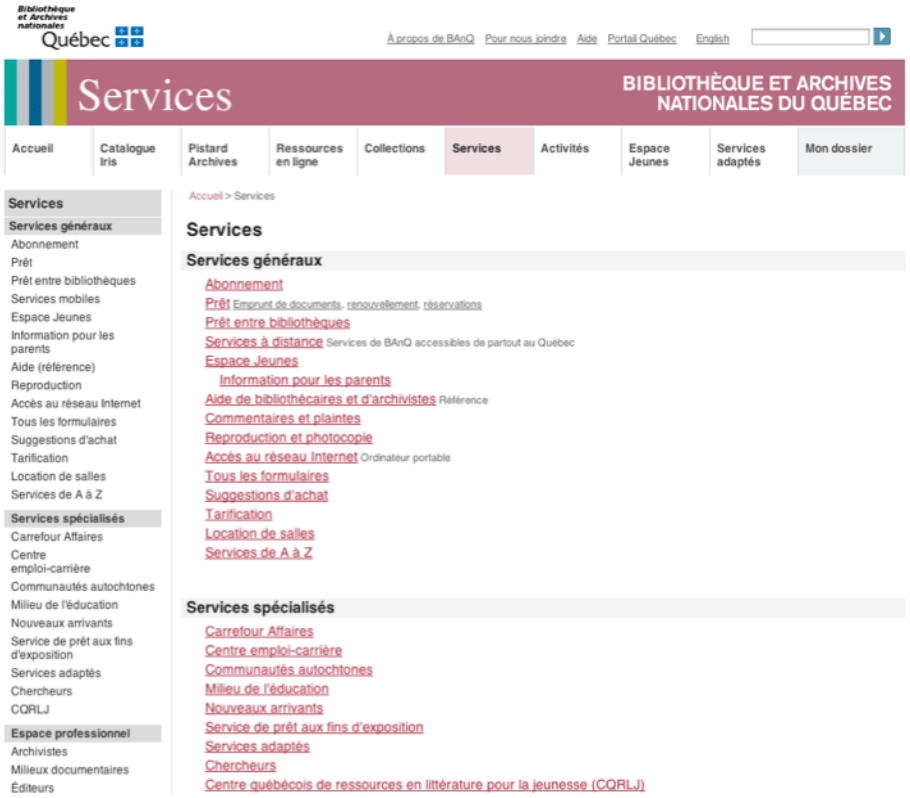
S1 : Gouvernement du Québec – Déménagement – Page « Services Québec Citoyens »

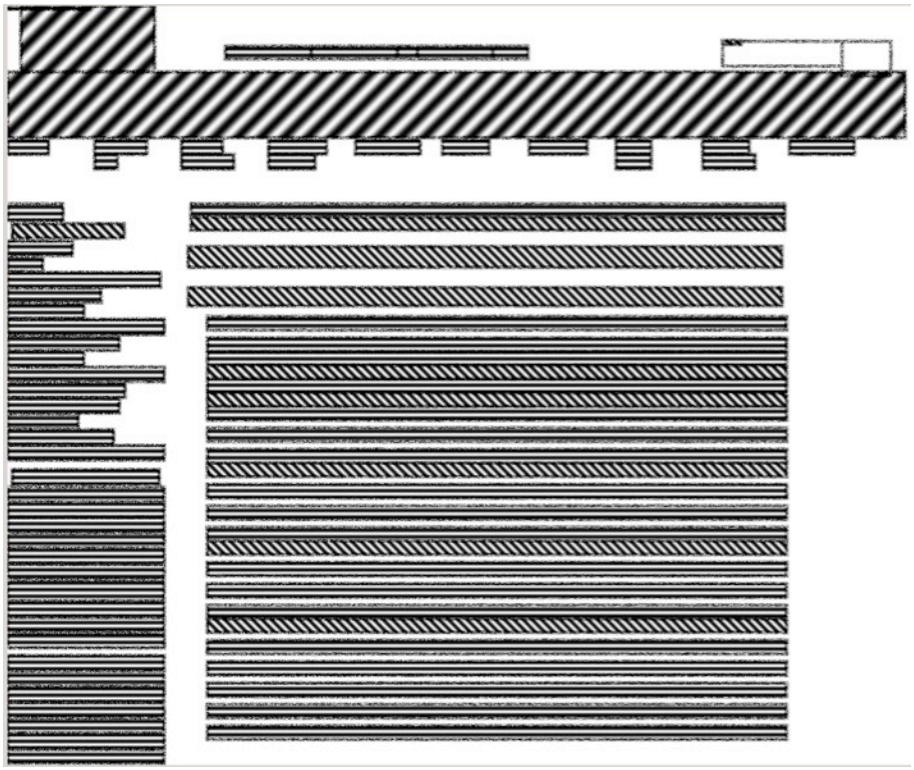


S2 : BANQ – Services adaptés – Page d'accueil

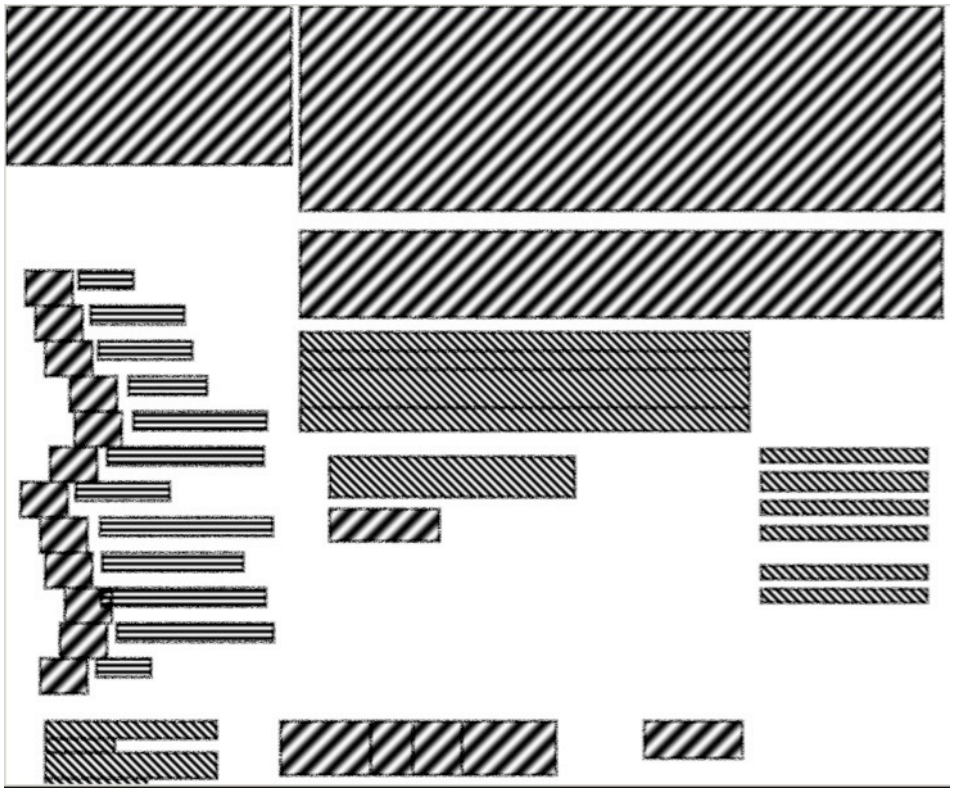


S2 : BANQ – Services adaptés – Page « Services »

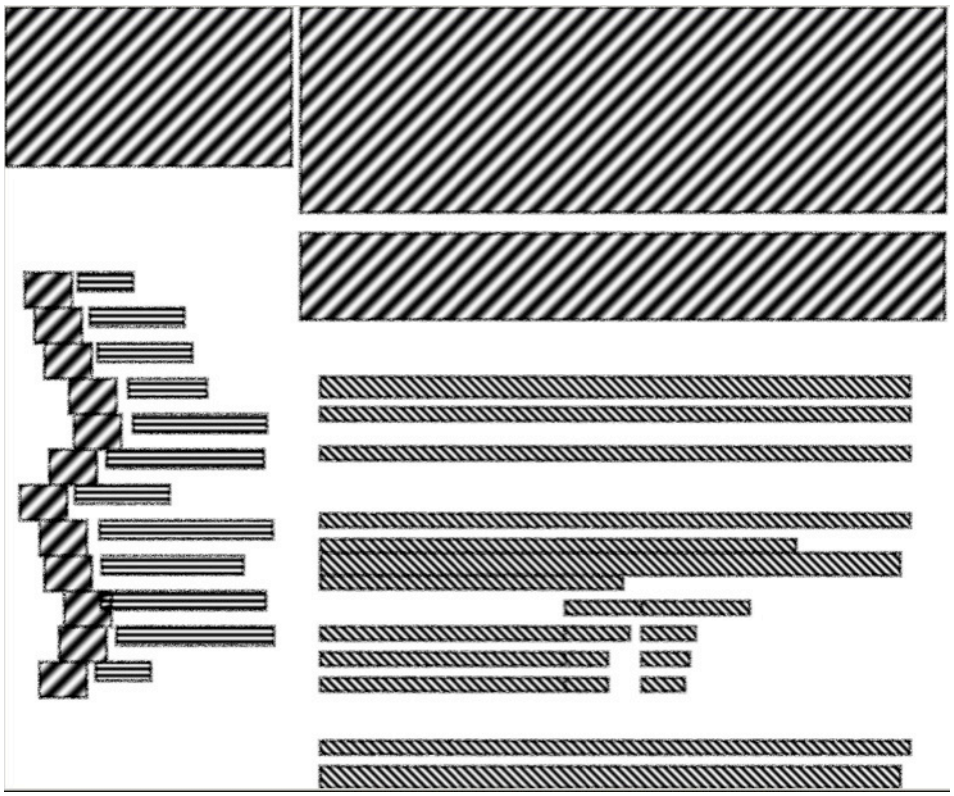




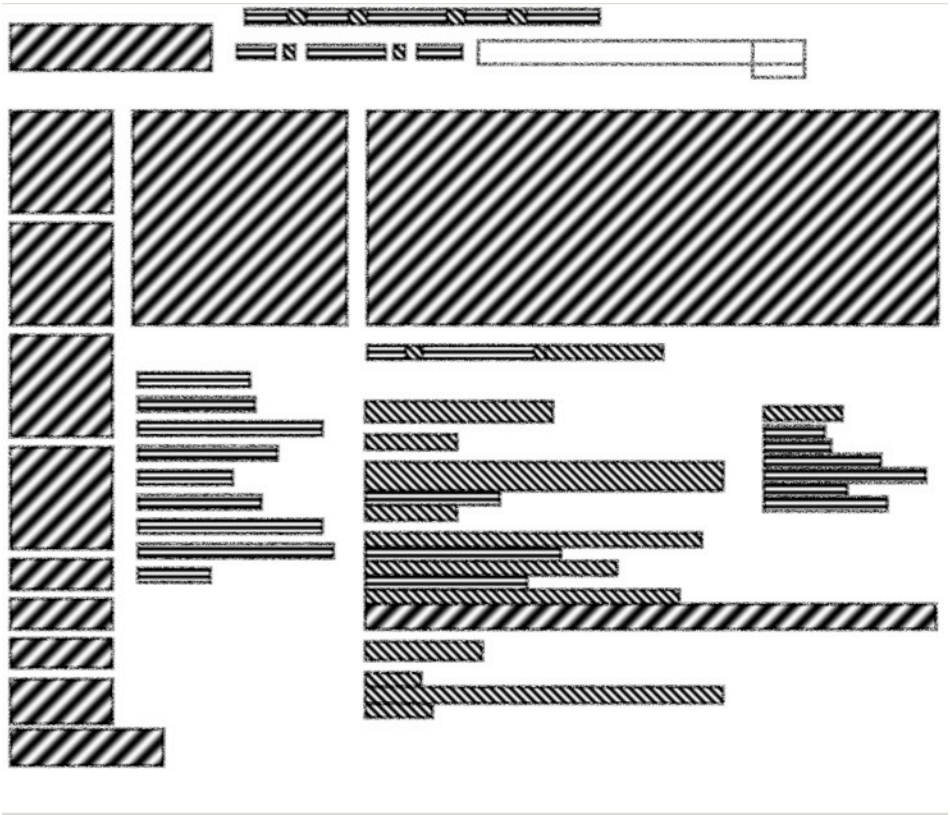
S3 : Auberge de jardin – Tarifs – Page d'accueil



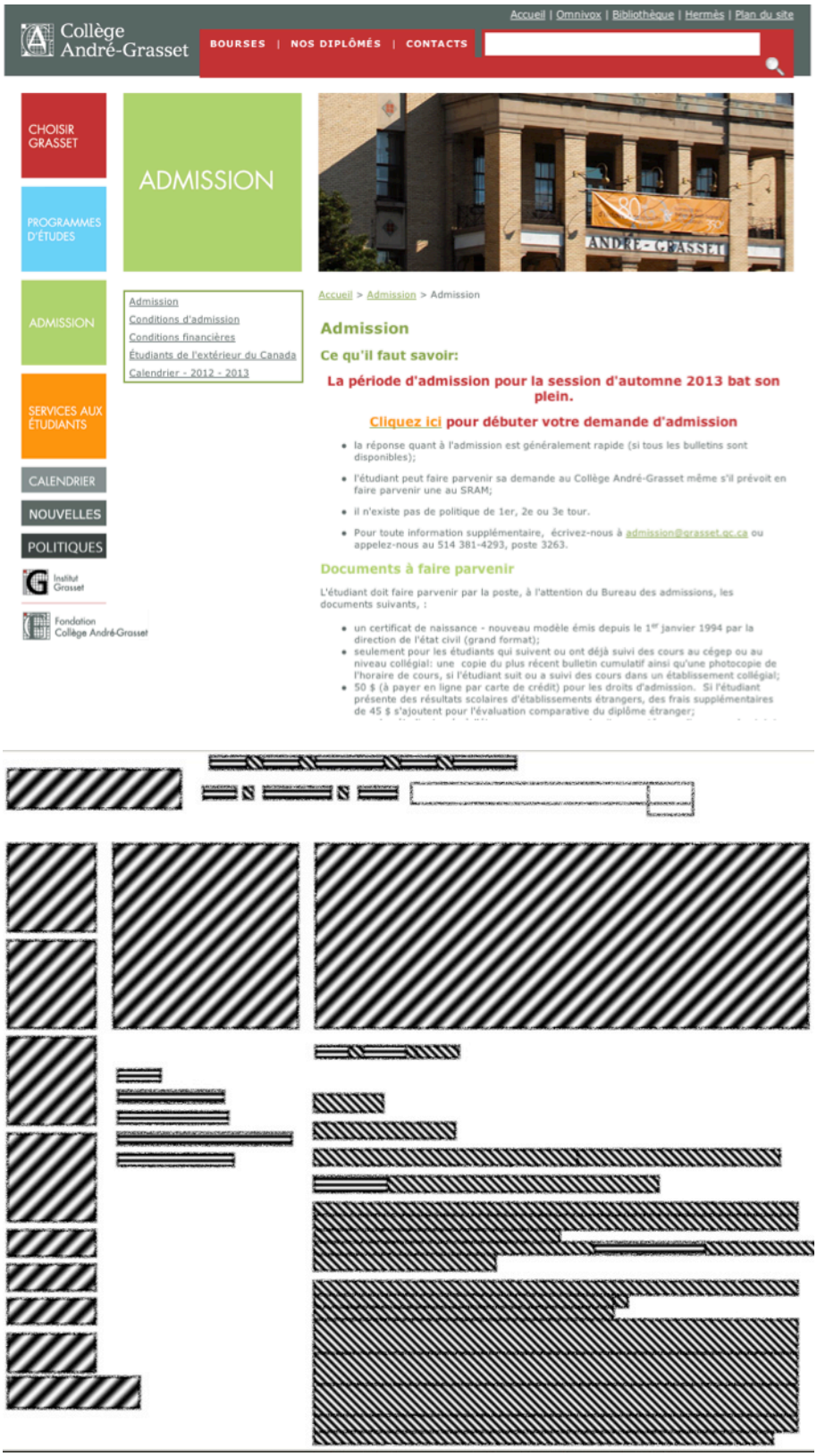
S3 : Auberge de jardin – Tarifs – Page des tarifs



S4 : Collège André-Grasset – Admission – Page « Programme en Sciences de la nature »



S4 : Collège André-Grasset – Admission – Page « Admission »



Accueil | Omnivox | Bibliothèque | Hermès | Plan du site

Collège André-Grasset

BOURSES | NOS DIPLÔMES | CONTACTS

CHOISIR GRASSET

PROGRAMMES D'ÉTUDES

ADMISSION

SERVICES AUX ÉTUDIANTS

CALENDRIER

NOUVELLES

POLITIQUES

Institut Grasset

Fondation Collège André-Grasset

ADMISSION

Admission
Conditions d'admission
Conditions financières
Étudiants de l'extérieur du Canada
Calendrier - 2012 - 2013

Accueil > Admission > Admission

Admission

Ce qu'il faut savoir:

La période d'admission pour la session d'automne 2013 bat son plein.

Cliquez ici pour débuter votre demande d'admission

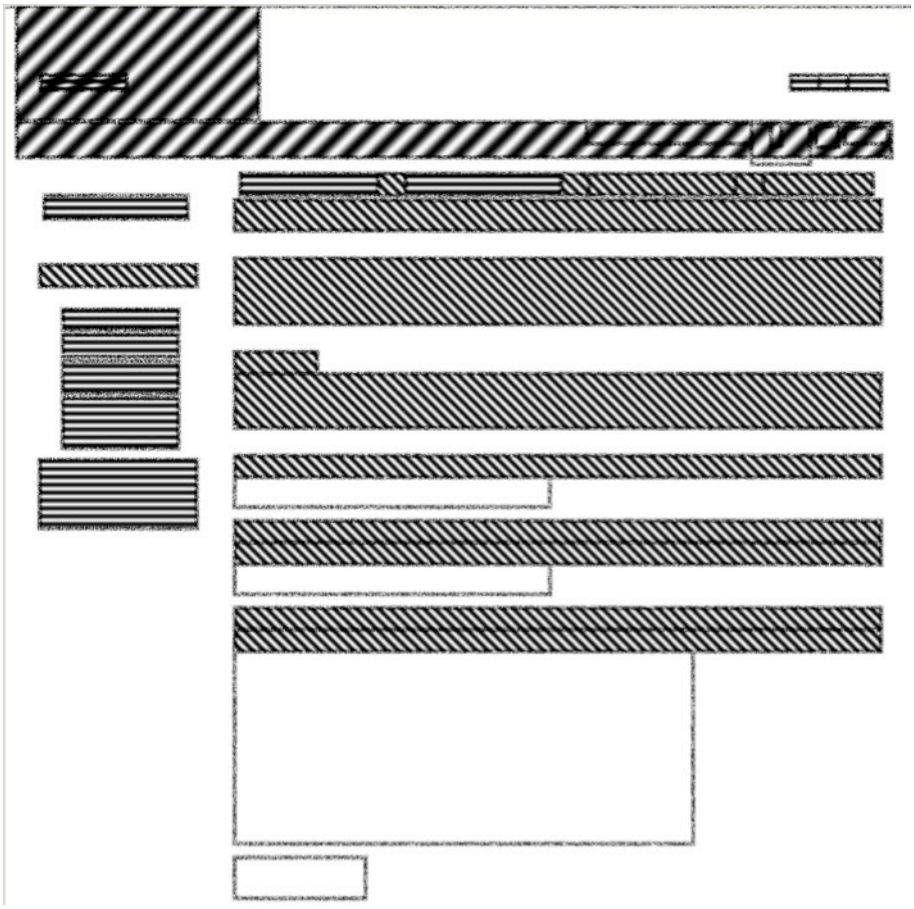
- la réponse quant à l'admission est généralement rapide (si tous les bulletins sont disponibles);
- l'étudiant peut faire parvenir sa demande au Collège André-Grasset même s'il prévoit en faire parvenir une au SRAM;
- il n'existe pas de politique de 1er, 2e ou 3e tour.
- Pour toute information supplémentaire, écrivez-nous à admission@grasset.qc.ca ou appelez-nous au 514 381-4293, poste 3263.

Documents à faire parvenir

L'étudiant doit faire parvenir par la poste, à l'attention du Bureau des admissions, les documents suivants, :

- un certificat de naissance - nouveau modèle émis depuis le 1er janvier 1994 par la direction de l'état civil (grand format);
- seulement pour les étudiants qui suivent ou ont déjà suivi des cours au cégep ou au niveau collégial: une copie du plus récent bulletin cumulatif ainsi qu'une photocopie de l'horaire de cours, si l'étudiant suit ou a suivi des cours dans un établissement collégial;
- 50 $ (à payer en ligne par carte de crédit) pour les droits d'admission. Si l'étudiant présente des résultats scolaires d'établissements étrangers, des frais supplémentaires de 45 $ s'ajoutent pour l'évaluation comparative du diplôme étranger;

Q1 : Gouvernement du Québec – Nous joindre par courriel



Q2 : IGA – Infobulletin

INFO-BULLETIN

L'info-bulletin est un courriel hebdomadaire envoyé à nos abonnés. On y trouve des informations sur la nutrition, des recettes, des trucs et astuces et des offres promotionnelles.

À chaque parution de l'info-bulletin, vous courez la chance de gagner un **certificat-cadeau IGA d'une valeur de 25 $** ou un **lot de 25 milles de récompense AIR MILES**md. Pour participer… c'est simple, vous n'avez qu'à vous abonner (résidents du Québec seulement)!

Prénom:

Nom:

Code postal:

Date de naissance:

Courriel:

Sexe: Homme Femme

Téléphone:

Numéro Air Miles®:

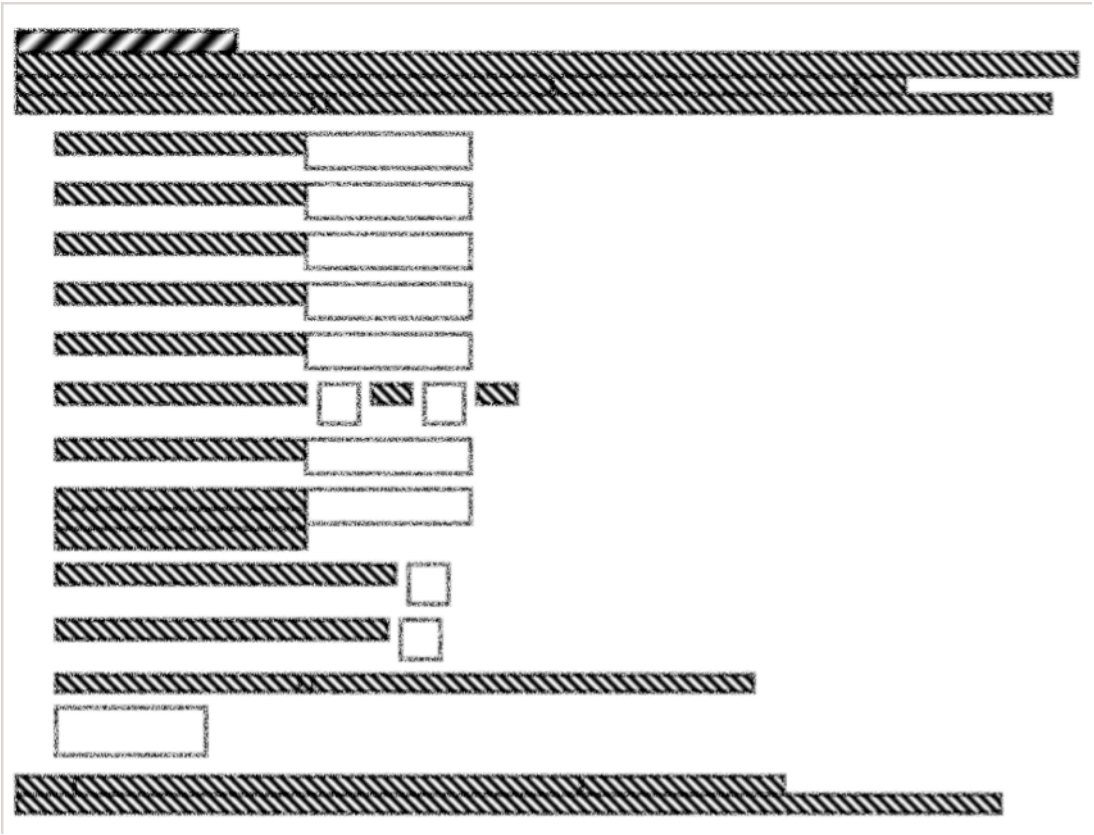
Veuillez m'inscrire à l'info-bulletin IGA.

Veuillez m'inscrire aux offres exclusives.

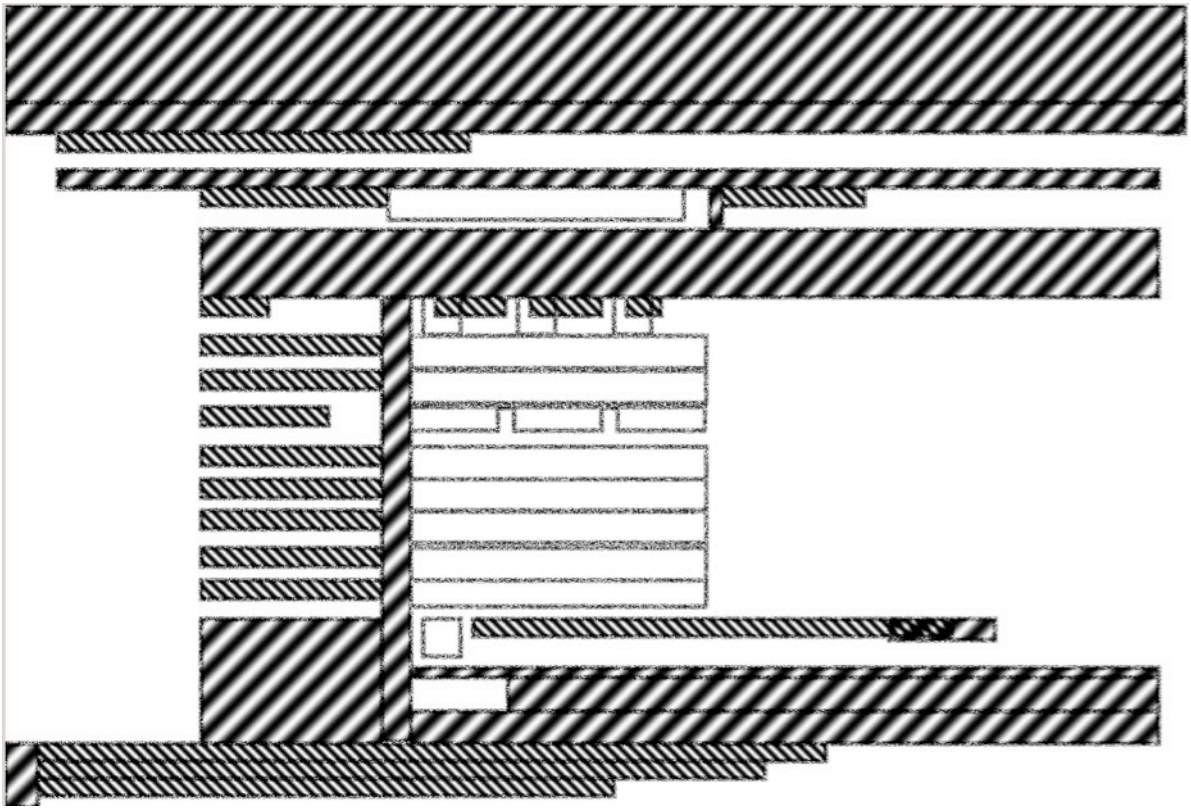
En inscrivant votre numéro AIR MILESmd, vous acceptez de recevoir des offres exclusives de la part d'IGA.

S'ABONNER

Remarque : Si vous utilisez un nettoyeur de pourriels, assurez-vous d'ajouter **noreply@infobulletin-iga.net** à votre liste de contacts. Cela évitera que les courriels provenant d'IGA.net ne soient traités comme des courriels indésirables.



Q3 : Le Point – Newsletter



Q4 : Disquaire Martin Lévesque – Inscription

Discaire Martin Lévesque

Inscrivez-vous

Nom :

Prénom :

Adresse courriel :

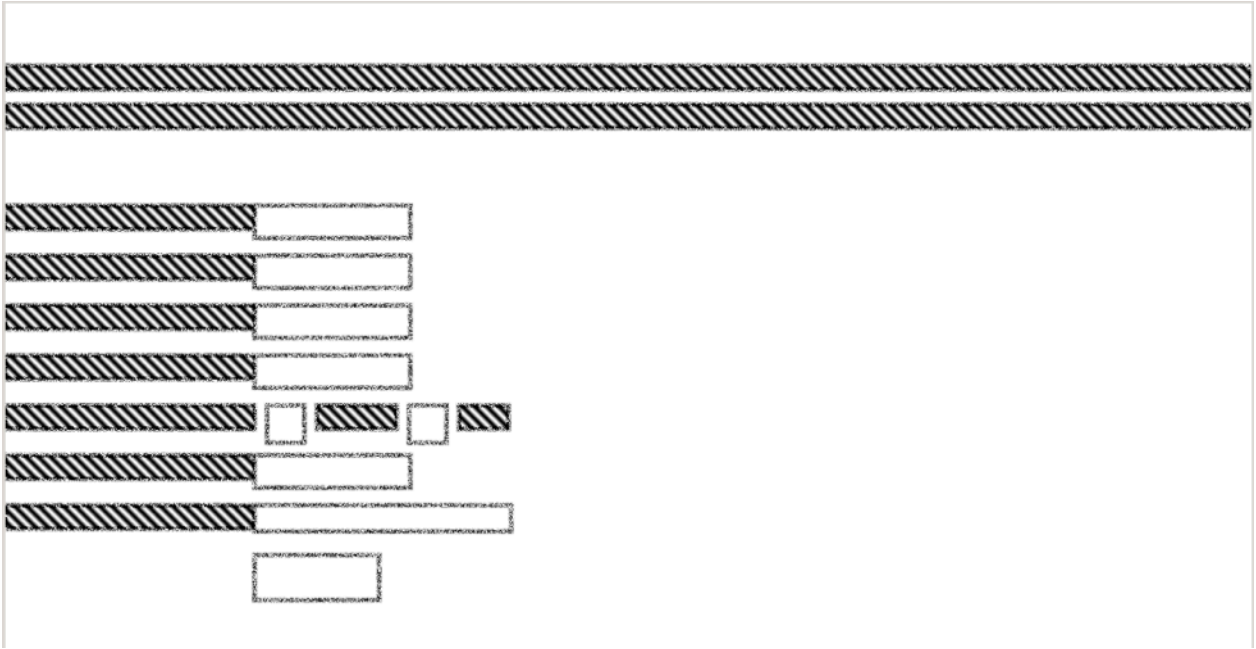
Mot de passe :

Sexe : Homme Femme

Numéro de téléphone :

Province : Sélectionner

Valider



ANNEXE 7 – QUESTIONNAIRE POST-EXPÉRIMENTATION

Les prochaines questions ne concernent que l'outil JAWS. Pour la majorité des questions, vous devrez exprimer votre point de vue sur une échelle de 1 à 5 afin d'exprimer les limites (1) ou les qualités (5) de l'outil selon différents aspects.

JAWS

1. Compléter les tâches est : 1 2 3 4 5
   Très difficile Très facile
2. Trouver de l'information dans les pages Web est : 1 2 3 4 5
   Très difficile Très facile
3. Remplir les formulaires est : 1 2 3 4 5
   Très difficile Très facile
4. Utiliser JAWS est : 1 2 3 4 5
   Très difficile Très facile
5. Interagir avec le clavier dans JAWS est : 1 2 3 4 5
   Très difficile Très facile
6. Utiliser JAWS demande de la concentration : 1 2 3 4 5
   Très peu Beaucoup
7. La compréhension des explications vocales est : 1 2 3 4 5
   Très difficile Très facile
8. La qualité de la synthèse vocale est : 1 2 3 4 5
   Très mauvaise Très bonne
9. L'idée globale que je me fais des pages Web que j'ai explorées est : 1 2 3 4 5
   Très mauvaise Très bonne
10. J'aimerais utiliser JAWS : 1 2 3 4 5
    Très rarement Très souvent
11. J'ai trouvé ma performance : 1 2 3 4 5
    Très insatisfaisante Très satisfaisante

12. Selon vous, y-a-t-il des points positifs dans JAWS? Si oui lesquels?

13. Selon vous, y-a-t-il des points négatifs dans JAWS? Si oui lesquels?

14. Voudriez-vous modifier, ajouter ou supprimer des aspects de JAWS? Si oui, lesquels?

Les prochaines questions ne concernent que l'outil TactoWeb. Pour la majorité des questions, vous devrez exprimer votre point de vue sur une échelle de 1 à 5 afin d'exprimer les limites (1) ou les qualités (5) de l'outil selon différents aspects.

TactoWeb

1.	Compléter les tâches est :	1	2	3	4	5
		Très difficile			Très facile	
2.	Trouver de l'information dans les pages Web est :	1	2	3	4	5
		Très difficile			Très facile	
3.	Remplir les formulaires est :	1	2	3	4	5
		Très difficile			Très facile	
4.	Utiliser TactoWeb est :	1	2	3	4	5
		Très difficile			Très facile	
5.	Interagir avec le clavier dans TactoWeb est :	1	2	3	4	5
		Très difficile			Très facile	
6.	Utiliser TactoWeb demande de la concentration :	1	2	3	4	5
		Très peu			Beaucoup	
7.	L'expérimentation a fatigué votre sensation du toucher.	1	2	3	4	5
		Très peu			Beaucoup	
8.	Le retour sonore est :	1	2	3	4	5
		Inutile				Utile
9.	La compréhension des explications vocales est :	1	2	3	4	5
		Très difficile			Très facile	
10.	La qualité de la synthèse vocale est :	1	2	3	4	5
		Très mauvaise			Très bonne	
11.	Distinguer les sons est :	1	2	3	4	5
		Très difficile			Très facile	
12.	Les indices sonores sont :	1	2	3	4	5
		Peu nombreux			Trop nombreux	

13. La quantité d'informations à gérer en même temps (son, sensation, lecteur d'écran) est

	1	2	3	4	5
	Minimale				Élevée

14. L'utilisation des boutons de l'appareil est :	1	2	3	4	5
	Très difficile			Très facile	
15. L'idée globale que je me fais des pages Web que j'ai explorées est :	1	2	3	4	5
	Très mauvaise			Très bonne	
16. J'aimerais utiliser TactoWeb:	1	2	3	4	5
	Très rarement			Très souvent	
17. J'ai trouvé ma performance :	1	2	3	4	5
	Très insatisfaisante			Très satisfaisante	

18. Selon vous, y-a-t-il des points positifs dans TactoWeb? Si oui lesquels?

19. Selon vous, y-a-t-il des points négatifs dans TactoWeb? Si oui lesquels?

20. Voudriez vous modifier, ajouter ou supprimer des aspects de TactoWeb? Si oui, lesquels?

Comparaison entre TactoWeb et JAWS

1. Est-ce qu'il y a un outil que vous avez préféré?

Si oui, lequel et pourquoi?

2. Est-ce qu'il y a un outil que vous trouvez plus facile à utiliser?

Si oui, lequel et pourquoi?

3. Est-ce qu'il y a un outil que vous trouvez plus plaisant à utiliser?

Si oui, lequel et pourquoi?