

Propositions d'amélioration de la navigation par
l'ajout de fonctions aux navigateurs Web

André Kurz & Antoine Hayoz

21 juin 2005

Table des matières

1	Introduction	6
2	Etat de l'art de l'hypertexte	7
2.1	Qu'est-ce que l'hypertexte ?	7
2.1.1	Les liens	9
2.1.2	Les noeuds	11
2.1.3	Le contenu multimédia	11
2.1.4	La navigation	12
2.2	Principaux éléments formant l'hypertexte	13
2.2.1	Contenu et présentation	13
2.2.2	Structure	14
2.3	Quels sont les buts de l'hypertexte ?	15
2.3.1	Accéder à l'information	15
2.3.2	Valoriser les informations	15
2.3.3	Augmenter la lisibilité	15
2.3.4	Découvrir...	16
2.4	Avantages et inconvénients de l'hypertexte	16
2.4.1	Avantages	16
2.4.2	Inconvénients	16
2.4.2.1	Désorientation	17
2.4.2.2	Comportement de l'utilisateur	17
2.4.2.3	Unités de restitution	18
2.4.2.4	Standardisation	18
2.4.2.5	Structure et recherche de l'information	19
2.5	Conclusion	19
3	Comparaison des navigateurs Web actuels	21
3.1	Qu'est-ce qu'un navigateur ?	21
3.1.1	Historique des différents navigateurs	23
3.1.2	Les fonctionnalités d'aide à la navigation	23
3.1.2.1	Particularités de certains navigateurs	26
3.1.3	Recherches effectuées	28
3.2	Constat du manque	29

4	Solution	31
4.1	Introduction	31
4.2	<i>Thumbnail</i>	31
4.3	<i>Scanning</i>	32
4.4	Arbre de navigation	34
4.5	<i>Breadcrumbs</i>	35
4.6	<i>Sitemap</i>	36
4.7	Conclusion	37
5	Evaluation	38
5.1	Introduction	38
5.2	Pré évaluation	38
5.2.1	Scénario	38
5.2.2	Remarques	39
5.3	Prototype	39
5.3.1	Prototypage papier	39
5.3.2	Technologie utilisée	39
5.3.3	Solutions développées	41
5.3.3.1	<i>Thumbnail</i>	41
5.3.3.2	<i>Scanning</i>	42
5.3.3.3	Arbre de navigation	42
5.3.3.4	<i>Breadcrumbs</i>	43
5.3.3.5	<i>Sitemap</i>	44
5.3.3.6	Vue d'ensemble	44
5.3.4	Type d'actions	45
5.4	Participants	45
5.5	Méthodologie	46
5.5.1	Familiarisation	46
5.5.2	Description d'une tâche	46
5.5.3	Tâche 1 - <i>Scanning</i>	46
5.5.4	Tâche 2 - Arbre de navigation	46
5.5.5	Tâche 3 - Historique visuel	46
5.5.6	Tâche 4 - <i>Breadcrumbs</i>	47
5.5.7	Tâche 5 - Représentation graphique du chemin de navigation	47
5.5.8	Tâche 6 - <i>Sitemap</i>	47
5.6	Matériel	47
5.7	Hypothèses	47
5.8	Résultats	48
5.8.1	Tâche 1	48
5.8.2	Tâche 2	49
5.8.3	Tâche 3	51
5.8.4	Tâche 4	52
5.8.5	Tâche 5	52
5.8.6	Tâche 6	54
5.8.7	Discussion	55
5.8.8	Améliorations possibles	56

5.9 Conclusion	58
6 Conclusion générale	59

Table des figures

2.1	Concept noeuds - liens	8
2.2	Les " <i>Breadcrumbs</i> "	9
2.3	Liens au contenu similaire	10
2.4	Liens au contenu différent	11
2.5	La navigation	12
2.6	Contenu - Présentation - Structure	13
3.1	Comparaison de navigateurs Web	22
3.2	"Précédente" et "Suivante"	24
3.3	" <i>Hub and spoke</i> "	25
3.4	Le navigateur OmniWeb	27
3.5	<i>Plugin</i> intégrant les trois fonctions de base	28
3.6	Navigateur hyperbolique	29
4.1	Un <i>thumbnail</i> de l'historique visuel	32
4.2	De la page HTML...	33
4.3	... à la mini page	34
4.4	Arbre de navigation	35
4.5	<i>Breadcrumbs</i>	36
4.6	<i>Sitemap</i>	37
5.1	Différentes parties du prototype	41
5.2	<i>Thumbnail</i>	42
5.3	<i>Scanning</i>	42
5.4	Arbre de navigation	43
5.5	<i>Breadcrumbs</i>	44
5.6	<i>Sitemap</i>	44
5.7	Prototype	45
5.8	Tâche 1 - Description de la structure d'une page	49
5.9	Tâche 2 - Durée de navigation	50
5.10	Tâche 2 - Nombre de pages chargées	51
5.11	Tâche 3 - Temps nécessaire pour retourner sur une page visitée	52
5.12	Tâche 5 - Nombre de pages citées sur l'ensemble du chemin de navigation	53
5.13	Tâche 5 - Séquence des pages citées	54

5.14	Tâche 6 - Description de la structure du site	55
6.1	Historique visuel	65
6.2	Arbre de navigation	66
6.3	<i>Breadcrumbs / Sitemap</i>	67
6.4	Page 1 - Informations sur le testeur	69
6.5	Page 2 - Tâche 1	70
6.6	Page 3 - Tâche 2	71
6.7	Page 4 - Tâche 3	72
6.8	Page 5 - Tâche 4	73
6.9	Page 6 - Tâche 5	74
6.10	Page 7 - Tâche 6	75
6.11	Page 8 - Questions destinées aux testeurs du prototype	76

Chapitre 1

Introduction

Qui n'a jamais été perdu lors de sa navigation sur Internet ? Qui n'a jamais souhaité retrouver une page visitée mais devenue introuvable ?

L'objet de notre travail de licence, porte sur la navigation Web et les problèmes qui y sont liés. Plus précisément le sentiment de désorientation qui peut survenir.

Dans le but de cibler et de mieux comprendre les points qui pourraient être améliorés dans ce domaine, nous étudierons des articles abordant ce sujet ainsi que les différentes fonctions des navigateurs Web actuels. Nous ferons également le point sur les recherches en cours et achevées. Ceci nous permettra de distinguer les problèmes liés à l'hypertexte et à la navigation ainsi que les solutions étudiées ou développées, qui pourraient y remédier.

Dans un deuxième temps, nous proposerons un outil permettant de pallier aux problèmes identifiés et non résolus. Cet outil sera développé sous la forme d'un prototype et testé empiriquement.

Ce projet est la continuation d'une étude effectuée pour un cours sur les interfaces des systèmes d'information. Il nous était demandé, comme travail de recherche, de proposer "un vrai navigateur" avec de nouvelles fonctions de navigation. Interpellés par la problématique, nous avons souhaité l'approfondir par le présent travail. Nous nous arrêterons cependant au prototype, mais la suite logique serait le développement d'un *plugin* basé sur nos recherches.

Chapitre 2

Etat de l'art de l'hypertexte

2.1 Qu'est-ce que l'hypertexte ?

En 1945, le scientifique Vannevar Bush, qui est un des pères fondateurs de l'hypertexte, présente sa vision de l'avenir technologique : une machine permettant de stocker une quantité immense de documents scientifiques, avec pour but que ceux-ci puissent être complétés, enrichis et surtout consultés. Vingt ans plus tard en 1965, Theodor Holm Nelson invente le terme "hypertexte", que lui-même définit comme "une écriture non linéaire". D'une manière générale, l'hypertexte est un moyen de navigation dans un ensemble d'informations. Chaque document faisant partie de cette information se nomme **noeud** et est relié à d'autres noeuds par des **liens**, comme nous le montre la figure 2.1.



FIG. 2.1 – Concept noeuds - liens

Les noeuds sont représentés par des documents textes reliés à d'autres documents textes par des liens. Nous remarquerons la non linéarité de ce concept.

Ce concept d'hypertexte voit sa notoriété exploser lorsqu'il est utilisé par le logiciel HyperCard, développé par Bill Atkinson et fourni dès 1987 sur tous les ordinateurs Macintosh. Dans ce programme, les informations sont divisées en unités et l'utilisateur peut passer des unes aux autres en utilisant des liens.

Ce réseau d'informations, qu'est l'hypertexte, est souvent associé à l'image d'une toile. Le Web, qui est un des thèmes de notre travail, en est un bon exemple. Il est bon de rappeler que le premier navigateur Web est sorti en 1993 sous Unix, il se nomme Mosaic et a été développé au *National Center for Superconducting Application* (NCSA). Nous pouvons donc dire que la véritable nouveauté de l'hypertexte appartient à son aspect technologique, l'aspect conceptuel existant déjà depuis fort longtemps à travers l'idée de bibliothèques, d'encyclopédies ou encore de dictionnaires. Ces compilations d'informations sont écrites et organisées d'une manière non linéaire donnant accès à une quantité importante d'informations.

Les définitions du terme hypertexte sont nombreuses, en voici quelques unes. Selon Jakob Nielsen : "L'hypertexte est une écriture non séquentielle, où chaque noeud contient une quantité de textes ou d'autres informations. Les noeuds sont connectés par des liens directs. Un noeud peut avoir plusieurs liens sortants." [2].

Pour le Petit Larousse : "L'hypertexte est une technique ou un système qui permet, lors de la consultation d'une base documentaire de textes, de sauter d'un

document à un autre selon des chemins préétablis ou élaborés à cette occasion." [32].

2.1.1 Les liens

Un lien est une relation entre deux unités d'informations. Ils sont ajoutés par l'utilisateur ou le concepteur mais ne sont pas intrinsèques à l'information. Ils peuvent également être générés dynamiquement [8]. Plus concrètement, un lien est un extrait de texte dans un document faisant référence à un autre document. Ce texte est alors interprété par le langage de programmation *HyperText Markup Language* (HTML) comme étant un lien, celui-ci faisant référence à un autre document.

Concernant leurs utilisations, Jakob Nielsen nous conseille de réduire le nombre de liens aux fonctionnalités les plus basiques, par exemple fournir les liens des niveaux hiérarchiques supérieurs à celui où l'on se trouve [5]. Les "*Breadcrumbs*" sont de bons exemples, ils nous donnent une idée de la localisation de la page courante dans l'arborescence hiérarchique d'un site. La figure 2.2 permet de mieux comprendre cette sorte de chemin thématique.

[Accueil](#) > [Matériel](#) > Moniteurs

FIG. 2.2 – Les "*Breadcrumbs*"

Exemple de "*Breadcrumbs*" fournissant à l'utilisateur le principal chemin d'accès de la page d'accueil à la page courante d'un site. Dans cet exemple, l'utilisateur se trouve dans la partie "Moniteurs" d'un site de vente de matériel informatique. Cette partie est une sous catégorie de la section "Matériel", accessible depuis la page d'accueil.

Les liens locaux sont des liens pointant sur des pages d'un même site. Nous pouvons les classer en deux catégories : les liens qui pointent sur des pages au contenu similaire et ceux qui pointent sur des pages au contenu très différent. Tous deux sont pratiques lorsque leurs contenus ont une certaine complémentarité entre eux.

Pour illustrer ces propos, nous pouvons prendre l'exemple d'un site marchand comme Amazon¹.

La rubrique "*Newspapers*" propose une liste de liens au contenu similaire. En utilisant ce terme, nous faisons référence à un même type de produit, dans ce cas des magazines. La figure 2.3 illustre cette première catégorie de liens.

¹<http://www.amazon.com/>

Magazine & Newspaper Subscriptions

Over 60,000 titles, great prices, easy renewals

A Bonus Just for Subscribing



For a limited time, get a promotional certificate worth \$5 off a future Amazon.com order when you subscribe to selected magazines. Choose among 30 titles, including [Newsweek](#), [Golf Magazine](#), [Sports Illustrated](#), [People](#), and [Reader's Digest](#). See the [complete list](#).

New & Notable Magazines of 2004



Continuing a trend, two new "living" magazines debuted in 2004. [Cottage Living](#) and [Low Carb Living](#) (or, as we like to call it, [Cottage Cheese Living](#)) joined the ranks of [Martha Stewart Living](#), [Country Living](#), [Budget Living](#), and the expansively titled [Living Etc.](#) Check out our picks for the [top 10 new magazines of 2004](#).

Customers' Favorite Magazines of 2004




A [cool tech magazine](#) at a low price with a [special offer](#): three reasons [Wired](#) was our most-ordered title in 2004. Hot on its heels was the celebrity-obsessed [People](#) magazine and its athletic cousin [Sports Illustrated](#). What else beat out the competition? Check out [our customers' top 10 favorite magazines of the year](#).

FIG. 2.3 – Liens au contenu similaire

Pour illustrer la seconde catégorie, nous observons les produits proposés dans la partie "*Were also interested in these items...*" de la figure 2.4. En plus du détail du produit sélectionné, l'utilisateur obtient une liste de liens pointant sur des produits en relation avec lui.

Customers interested in this item...



Kodak MAX Water & Sport Single Use Camera
[Other products by Kodak](#)
Average Customer Review: ★★★★★
Usually ships in 24 hours

Eligible for **FREE Super Saver Shipping** on orders over \$25. [See details.](#)

From the Manufacturer
Survives where your regular camera won't!

- **Rugged, durable, shock-proof rubber shell**
Great for bumpy rides and rocky trails. Roller coaster anyone?
- **Waterproof up to 50 feet**
Divers and snorklers, capture the vibrancy of life 50 feet underwater!
- **Sunscreen- and scratch-resistant lens**
No blurry

[... Read more](#)

Were also interested in these items...


<p>Show items from:</p> <p>All Products</p> <p>▶ Books (1)</p> <p>Camera & Photo (7)</p> <p>DVD (1)</p> <p>Electronics (3)</p> <p>Kitchen & Housewares (1)</p> <p>Toys & Games (1)</p>	 Diving The Seamount by Tom Rapko More like this	
---	--	--

FIG. 2.4 – Liens au contenu différent

Cette dernière liaison met en relation des pages aux contenus très différents mais complémentaires. Il faut toutefois éviter la navigation "sur-faite²", ce qui réduit l'utilisabilité d'un site [5] et crée une surcharge visuelle.

2.1.2 Les noeuds

Un noeud est une unité d'information, aussi appelé document hypertexte, relié à d'autres noeuds par des liens. Chaque noeud peut contenir un ou plusieurs liens. Cette unité d'information peut-être composée de texte ou de contenu multimédia.

2.1.3 Le contenu multimédia

Le contenu multimédia peut-être traduit dans le domaine de l'hypertexte par le terme hypermédia. Celui-ci comprend différents types de médias : images, sons, vidéos, etc. Alan Dix parle du *World Wide Web* comme étant un système hypermédia global, où les animations et les vidéos permettent de visualiser des informations qui sont difficiles à transmettre statiquement [13].

²Ce terme signifie une quantité exagérée de liens, permettant depuis une page d'accéder à presque toutes les autres pages disponibles sur un même site.

2.1.4 La navigation

Contrairement aux textes imprimés dont la pagination est linéaire et la lecture faite pour suivre ce même ordre, l'hypertexte permet une navigation non linéaire entre les documents. La navigation est donc une suite de noeuds visités par un utilisateur en vue d'atteindre un objectif (figure 2.5). Elle est composée des trois points décrits ci-dessus. Le départ se fait à partir d'un noeud pour arriver à un autre noeud en passant par un lien les associant.

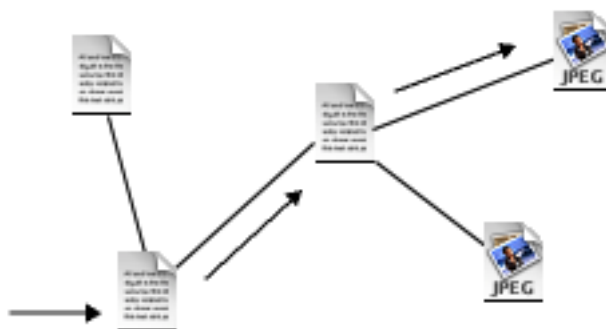


FIG. 2.5 – La navigation

Le point de départ de cette navigation est le document hypertexte se trouvant en bas à gauche de ce schéma. L'utilisateur y est arrivé directement, donc sans venir d'un autre document hypertexte. En utilisant un lien présent sur ce document, il va se déplacer sur un autre document, depuis lequel il accèdera à une image liée à ce second document. Cette image est le point d'arrivée de sa navigation.

Cette navigation inter-noeud est pour Alan Dix le plus important challenge d'interaction de la plupart des sites Web. Comparer la navigation à une interface, c'est poser les trois questions suivantes :

- Où êtes-vous ?
- Que pouvez-vous faire ?
- Où allez-vous ? [3]

Il est donc important de garder en mémoire que la navigation dépend étroitement de la structure de l'information choisie [6].

2.2 Principaux éléments formant l’hypertexte

Un hypertexte peut-être appréhendé selon différents aspects : son contenu, sa présentation et sa structure.



FIG. 2.6 – Contenu - Présentation - Structure

1 : L’aspect graphique de la page est composé notamment du logo, des couleurs de l’institution, des polices utilisées, etc.

2 : Le menu du site peut-être représentatif de la structure des informations.

3 : Le contenu à proprement parler est souvent mis en évidence.

2.2.1 Contenu et présentation

Le contenu, soit le message que l’on souhaite transmettre, ou encore l’information mise à disposition du lecteur, peut-être de différentes natures : textes, films, sons et animations sont les médias les plus fréquemment rencontrés.

Le contenu est prépondérant face au graphisme et à sa présentation. C’est ce qu’ont observé Rashmi Sinha, Marti Hearst, Melody Ivory et Maya Draisin lors de leurs recherches [1]. Cependant ils nuancent leurs propos suivant le thème du contenu présenté. Par exemple, si ce dernier est une galerie d’art (p. ex. www.nga.gov), la présentation et les aspects graphiques seront aussi importants que le contenu. Cela ne sera pas le cas pour un site consacré à l’actualité (p. ex. www.cnn.com) ou militant pour une cause (p. ex. www.gssa.ch).

La présentation du contenu sert à rendre plus agréable sa lecture ou sa visualisation. Ces aspects graphiques apportent une crédibilité accrue par rapport aux autres médias qui n’ont pas un graphisme très recherché [4]. C’est pourquoi ils doivent être présents et bien visibles. S’ils sont trop subtils, ils risquent de

passer inaperçus au yeux de l'utilisateur [2]. Ainsi, une interface dont la couleur de fond change subtilement en fonction de certains paramètres risque de ne pas avoir l'effet souhaité. L'utilisateur n'arrive pas à interpréter la relation entre ses actions et la couleur changeante.

Les graphismes imposent des contraintes à l'usage par rapport à d'autres médias, notamment par leurs tailles (en KoKilo-octet) qui limitent, lors de leurs téléchargements, les possibilités d'actions du visiteur. En effet, un utilisateur dont le moyen de surfer sur le Web serait un simple modem analogique traditionnel, sera bien déçu de visiter des sites au graphisme si prometteur, car lassé d'attendre, il finira par visiter des sites plus légers.

Il y a bien eu des tentatives de fournir du contenu en fonction des utilisateurs, de leurs goûts, de leurs caractéristiques de connexions [9], etc. Mais ces recherches n'ont pas encore réussi à populariser une technologie permettant de fournir, à la demande, du contenu personnalisé.

Le principal aspect auquel un utilisateur fait attention lorsqu'il visite un site pour la première fois est sa présentation. Elle est généralement plus perceptible que le contenu lors de la vision globale d'une page. Il se peut que certains utilisateurs soient réticents à visiter une page avec une mauvaise présentation et décident de continuer leur navigation ailleurs. Mais lorsqu'un utilisateur est vraiment intéressé par le contenu d'un site, il prendra le temps de le lire en faisant abstraction de la qualité de la présentation. Nous pouvons donc remarquer que plus le contenu est intéressant, moins la présentation du contenu est importante. Mais il ne faut pas pour autant négliger la structure de l'information, nous rappelle Alan Dix [3].

2.2.2 Structure

La structure apporte une organisation des thèmes présentés à travers des chapitres, titres, sous-titres, paragraphes, etc. La meilleure représentation de la structure d'un contenu serait certainement sa table des matières.

La possibilité d'ajouter des liens vers des noeuds ou pages hors de notre contenu, crée une situation où les hypertextes n'ont pas de début ni de fin. Un concepteur d'hypertexte ne peut pas prévoir précisément que ses visiteurs vont arriver par une page plutôt que par une autre [3].

Le fait que nous puissions arriver par n'importe quelle page sur un site, nous oblige à prendre en considération la structure informationnelle de celui-ci. Cette structure doit être visible en tout temps et portée à la connaissance du visiteur qui parcourt un site. Nous remarquerons que la navigation dépend de la structure intrinsèque de l'information [8]. Cette dernière doit permettre de se situer en fonction du contenu disponible [6].

La lecture d'un texte dans un livre ou une revue est allégée par sa diversité de présentation. En effet, les illustrations et les pieds de pages que rencontre le lecteur lui permettront de mieux se situer dans sa lecture, de mieux suivre son texte. Le contenu *on-line*, s'il est constitué de texte, peut se trouver très homogène et par conséquent difficile à consulter. Nous perdons nos repères et nous nous retrouvons désorientés au milieu du texte. Nielsen [2] a fait cette

constatation et il propose de partager les textes en parties plus petites liées entre elles, chacune est appelée noeud dans la littérature. Concrètement, l'idée est de prendre une page au contenu dense et de la diviser en plusieurs pages liées entre elles. Ces unités de restitution, qui représentent peut-être un paragraphe du texte original, rendent le contenu plus accessible.

Par contre, il serait bon de garder une manière homogène de présenter un même type d'informations. C'est ce que nous disent Melody Y. Ivory et Marti A. Hearst dans leur article "*Improving Web Site Design*". Ils soulignent que les vues d'ensemble et la compréhension de la structure navigationnelle sont nécessaires pour éviter la confusion des utilisateurs [10].

2.3 Quels sont les buts de l'hypertexte ?

L'hypertexte a toujours eu un objectif fondamental : permettre la transmission d'un message à des utilisateurs. Il se décline en plusieurs aspects.

2.3.1 Accéder à l'information

L'utilisation d'un hypertexte étant relativement simple (c.f. chapitre suivant), les pages consultables étant accessibles depuis n'importe quel ordinateur connecté à Internet, nous pouvons considérer que l'accès à un contenu est facilité. Ce n'est pas forcément le fait de trouver l'information qui est favorisé car l'immensité des contenus disponibles rend cette tâche compliquée. Mais il existe des outils, comme des moteurs de recherches (p. ex. www.google.ch, www.yahoo.com), qui aident à combler ce problème. Certains hypertextes favorisent directement la recherche d'informations en fournissant une structure au contenu.

2.3.2 Valoriser les informations

Une autre idée est de valoriser les informations trouvées en favorisant l'émergence de structures identiques dans le mental du lecteur [8]. Ainsi, il sera plus facile pour l'utilisateur de l'hypertexte de comprendre le message qu'il a lu, du fait qu'il a activement participé à sa compréhension. Cette lecture que l'on peut qualifier d'active est le résultat de la navigation à travers le contenu et du choix des rubriques que le lecteur va consulter. Un autre moyen de valoriser les informations serait d'utiliser des fonctions telles que les "*Bookmarks*" ou signets des navigateurs pour classer les pages visitées.

2.3.3 Augmenter la lisibilité

Parmi les sous objectifs, nous pouvons noter l'idée d'améliorer la lisibilité d'un texte [2], [8]. Cette idée a promu le développement des noeuds et des liens pour faciliter l'accès au message en divisant le contenu mis à disposition du lecteur suivant des thèmes principaux. Lier les éléments du contenu de manière

non linéaire est un thème que nous avons déjà abordé [12] dans les sections sur les liens et les noeuds .

2.3.4 Découvrir...

Mark Bernstein nous invite, à travers un hypertexte, à découvrir des nouveaux sujets, à se laisser surprendre. Il observe que les sites qui sont trop rigides ne permettent pas d'arriver par hasard sur une page au contenu non directement lié au but de la visite. Pourtant cette page peut être digne d'intérêt. Il préconise la découverte en utilisant des sentiers plutôt que l'accès à l'information en passant par la grande route. Il propose même de déranger le visiteur pour lui créer une attitude d'ouverture et de curiosité [25].

2.4 Avantages et inconvénients de l'hypertexte

2.4.1 Avantages

En plus de la grande facilité de consultation des pages Web, nous pouvons remarquer la diversité des documents tant sur le plan thématique que culturel. Grâce à la simplicité de mise en place du contenu, chacun peut contribuer à fournir une grande diversité culturelle sur la toile. Nous trouvons des sites abondant tous les sujets dans toutes les langues.

Cette diversité qui touche le contenu est aussi remarquable dans les services fournis. Nous pouvons noter les plus développés tel que le e-commerce, le *e-banking* ou dans des domaines plus pointus comme l'analyse de consommation d'énergie.

Un des aspects qui a retenu notre attention est la standardisation des sites Web. Selon Jakob Nielsen, elle apporte à un utilisateur le sens de la maîtrise d'un site, l'habileté de faire certaines actions comme aller consulter ses courriers électroniques, ou encore un sentiment de satisfaction croissant avec l'expérience [11]. Il existe une loi écrite par Jakob Nielsen qui stipule qu'un utilisateur passe la plus grande partie de son temps sur d'autres sites Web. En prenant en compte les points positifs cités ci-dessus ainsi que la loi de Nielsen, nous pouvons penser que l'utilisation de standards ou de conventions est nécessaire pour garder les visiteurs sur un site, ceux-ci étant habitués à certains standards à force de surfer sur le Web [11]. Ainsi, Jakob Nielsen propose que tous les sites utilisent les mêmes principes. Cette standardisation est d'autant plus importante que la plupart des utilisateurs ne prennent pas vraiment le temps de comprendre la structure d'un site.

2.4.2 Inconvénients

Le nombre des inconvénients liés à l'hypertexte étant plutôt important, nous n'avons retenu que les plus pertinents dans le cadre de notre travail.

2.4.2.1 Désorientation

Le problème de la désorientation de l'utilisateur, plus communément appelé "*Lost in hyperspace*³" est, d'après un constat général, le plus rencontré lors de la navigation à travers des documents hypertextes. Il survient lorsqu'un utilisateur a la sensation d'être perdu dans sa navigation Web, de ne plus savoir où il se trouve après de nombreux déplacements entre différents documents hypertextes. L'utilisateur doit pouvoir comprendre où il a été et où il peut aller. L'amélioration de la compréhension par l'utilisateur de sa localisation dans l'espace informationnel, doit permettre de prévenir cette sensation de désorientation selon Jakob Nielsen [7].

D'un autre point de vue, cette perte de repère est pour Alan Dix, "une conséquence naturelle du don à l'utilisateur du contrôle de sa progression à travers les documents hypertextes" [13]; soit sa liberté de choisir. Une page hypertexte pouvant être accessible depuis plusieurs endroits différents, aucun contexte d'information n'est donc fourni à propos de sa localisation. La solution qu'il propose et que l'on retrouve dans beaucoup de documents consiste à fournir une carte du document hypertexte permettant à l'utilisateur de pouvoir identifier sa position courante. Les liens de départ ou d'arrivée de sa navigation pouvant par la suite être identifiés, il aura alors la sensation d'être moins perdu. Un problème est toutefois lié à cette solution. De très grands documents hypertexte auront des cartes très complexes, ou alors, la carte étant tellement grande que seule une infime partie de celle-ci pourra être visualisable, ce qui pourrait ne pas être suffisant pour qu'un utilisateur se sente confiant et sache où il se trouve [13]. Le défi est donc d'aider l'utilisateur à se situer. L'idée du *Timestamp* d'Alan Dix permettrait de le faire en lui donnant une notion de temps. Quand un utilisateur retourne à un endroit dans l'hypertexte, le système affiche le temps écoulé depuis sa dernière visite. Lorsqu'un utilisateur visite un document pour la première fois, le système le lui signale au lieu d'afficher le *Timestamp*. Cette information temporelle peut aider les utilisateurs à se retrouver dans le cas où ils la mettent en relation avec la navigation qu'ils ont effectué [2].

2.4.2.2 Comportement de l'utilisateur

Quels sont les points caractérisant l'utilisateur lors d'une navigation hypertexte ? Avant toute chose, il est bon de se rappeler que la capacité de traitement cognitif de l'être humain est basée sur sa mémoire à court terme [3]. Concernant les contenus du Web, seulement 16% des utilisateurs lisent les pages mot à mot, alors que 79% les scannent⁴.

Il est donc recommandé d'utiliser du texte scannable : mots-clefs, titres, sous-titres, listes numérotées et pas plus d'une idée par paragraphe. De plus, les utilisateurs détestent les textes à caractère promotionnel ainsi que la publicité [4]. Selon Jakob Nielsen, en scannant une nouvelle page, l'utilisateur regarde

³Traduction littéraire : "Perdu dans l'hyperespace".

⁴Ce terme est utilisé dans l'idée de parcourir une page et de retenir l'essentiel du contenu.

directement le contenu et ignore l'espace de navigation contenant : logo, slogan, barre de navigation, éléments graphiques [5].

L'utilisateur a peu de tolérance envers :

- Ce qui ne fonctionne pas.
- Ce qui est trop compliqué.
- Ce qu'il n'aime pas.

Il est aussi impatient et n'aime pas attendre pour récupérer de l'information sur le Web [10]. Il n'est donc pas évident de concevoir un site Web objectivement optimal.

Bien que Mark Bernstein prétende que seuls les sites mal conçus provoquent le problème de "*Lost in hyperspace*", après une courte visite de son site nous pouvons affirmer qu'il faut tempérer ses propos concernant la structure de l'information. Il nous a fallu en effet plusieurs visites sur son site "*Parks and Gardens*⁵ " pour en comprendre la structure et en faire le tour. Même sur ce site qui ne comporte que quelques pages, nous avons eu l'impression d'être perdus et le contenu ne nous est pas apparu comme digne d'intérêt au premier abord.

2.4.2.3 Unités de restitution

Lorsqu'un utilisateur consulte un site et recherche des informations précises, il a parfois besoin d'avoir une sorte d'aide-mémoire de ce qu'il a lu et un résumé de ce qu'il pourra lire. D'autre fois, cet aide mémoire dérange plus qu'il n'aide car nous avons besoin de nous plonger dans l'information sans être dérangé par ce qui est en dessus ou en dessous de notre lecture. Ces contextes de lecture sont appelés "*larges*" ou "*small*" suivant que nous avons besoin d'informations génériques ou précises [2]. Ce besoin de contexte peut donc varier. Le Web n'apporte pas de solution à ce problème car la majorité des sites ne permettent pas de paramétrer l'affichage des informations au niveau de l'utilisateur.

Un autre aspect contextuel est que ce dernier peut changer d'un utilisateur à un autre. Un exemple de ce problème a été identifié sur le site Web de la conférence IMAGINA99. Lorsque les visiteurs regardaient les horaires des conférences qui se déroulaient soit à Paris soit à Monaco, ils pouvaient croire qu'une conférence se déroulait dans une ville alors qu'elle se déroulait dans l'autre [8]. Les unités de lecture n'étaient pas assez autonomes et ne comportaient pas toutes les informations dont avait besoin le visiteur pour comprendre le message. Ceci nous ramène à l'aspect "*Lost in hyperspace*" dont nous avons parlé à la section 2.4.2.1.

2.4.2.4 Standardisation

Malgré les avantages de la standardisation cités précédemment, nous pouvons regretter le faible nombre de sites basés sur des modèles. Le contenu du Web est créé et maintenu par de nombreux *Webmasters*⁶ peu enclins à utiliser

⁵<http://www.eastgate.com/garden/>

⁶Personne responsable de la création ou de la maintenance d'un site Web.

des standards. Ces derniers, qu'ils soient visuels ou de structure de l'information⁷ sont perçus comme des contraintes par les *Webmasters*. Peut-être que cela est dû à la diversité du contenu des différents sites, peut-être qu'il n'est pas évident pour certains *Webmasters* de comprendre que les standards facilitent l'utilisation du contenu. Mais lorsqu'un utilisateur arrive sur un site qui a une structure informationnelle et visuelle vraiment inhabituelle, il cherche à comprendre comment y naviguer pendant quelques instants. S'il n'y arrive pas assez vite, il va quitter le site. C'est la première possibilité de "défense" de l'utilisateur [28].

Pour les utilisateurs, les sites les plus standardisés sont les plus simples et la confusion de la présentation d'un site est un facteur favorisant son abandon [11].

2.4.2.5 Structure et recherche de l'information

Dire que le Web est composé d'une immense masse d'informations est un pléonasme qu'il est tout de même important de rappeler lorsqu'on aborde la structure et la recherche de l'information. Un petit tour sur le site des *plugins* de Mozilla nous démontre bien que ce problème n'est pas encore résolu et que les tentatives pour le résoudre sont nombreuses. Mais il est profondément ancré dans la structure même du Web et de l'HTML. L'impossibilité de visualiser la structure de l'information pénalise les recherches. Les relations N-N⁸ existantes entre les pages Web rendent la structure informationnelle du Web très complexe à cause de leurs nombres de liens [26].

L'utilisation d'un langage de *metadata*⁹ - *metatags*¹⁰ comme le XML pour qualifier le contenu permettrait de mieux organiser la recherche, soit par navigation soit plus traditionnellement par mots-clefs [12].

2.5 Conclusion

Nous avons vu dans ce chapitre les bases de l'hypertexte, ses composants et ses objectifs. Nous avons discuté de ses avantages ainsi que de ses nombreux problèmes. Nous nous sommes attardés sur les problèmes liés au "*Lost in hyperspace*", aux unités de restitution mal créées, au manque de standards dans la structure du contenu, etc. Nous avons donc, comme l'ont fait les chercheurs présents en 1991 à la troisième conférence de l'*Association for Computing Machinery* (ACM) à propos de l'hypertexte [31], constaté les défauts du Web. Nous nous poserons la question, dans le prochain chapitre, des aspects à améliorer et si cela

⁷Nous pouvons citer comme standards : la position du logo de l'institution en haut à gauche, ou encore la barre de navigation sur la gauche sous le logo.

⁸Nous entendons par relation N-N qu'une page est reliée à un nombre indéterminé de pages et qu'un nombre indéterminé de liens pointent aussi sur cette même page.

⁹Une méta donnée est littéralement une donnée sur une donnée. Plus précisément, c'est un ensemble structuré d'informations décrivant une ressource.

¹⁰Les méta balises sont des balises servant à définir des méta données.

peut-être fait au niveau d'un navigateur et de ses fonctionnalités, plutôt qu'au niveau des sites et de leurs structures.

Chapitre 3

Comparaison des navigateurs Web actuels

Dans le but de répondre aux questions précédemment posées, nous allons parler de différents sujets touchant aux navigateurs et espérons ainsi améliorer la perception des problèmes liés à la navigation.

3.1 Qu'est-ce qu'un navigateur ?

Un navigateur est un logiciel permettant à un utilisateur de se connecter à un serveur via le protocole *H*yper*T*ext *T*ransfer *P*rotocol (HTTP), dans la majorité des cas, et de permettre ainsi le transfert et l'affichage de pages Web au format HTML. Elles sont composées en partie d'images, de textes, de liens, etc. La plupart des navigateurs gèrent aussi d'autres protocoles, comme le *H*yper*T*ext *T*ransfer *P*rotocol *S*ecure (HTTPS, site Web sécurisé) et le *F*ile *T*ransfer *P*rotocol (FTP, transfert de fichiers), ou encore les transactions de paiement sécurisées en *S*ecure *S*ocket *L*ayer (SSL) [38]. Les navigateurs les plus populaires en parts de marché sont, en décembre 2004 [34] :

- Internet Explorer avec 71.4%
- Mozilla Firefox avec 21.2%
- Opera 7 avec 2.1%

Considérant l'importante part de marché du navigateur de Microsoft, Internet Explorer, nous le prendrons comme référence pour nos comparaisons. Il est bon de savoir que cette grande part de marché n'est pas liée à la qualité du produit, mais à une stratégie qui le rend disponible par défaut sur la majorité des systèmes d'exploitations. De ce fait, la plupart des utilisateurs s'en servent par paresse plutôt que de changer de navigateur.

WWW Feature support

	CSS	Frames	Java	JavaScript	Tabbed browsing	Download manager	Toolbar searching	Pop-up blocking
Amaya	CSS2	No	No	No	No	—	—	No
Camino	CSS2	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Dillo	—	No	No	Yes	No	—	—	No
Epiphany	CSS2	Yes	Yes	Yes	Yes	—	—	Yes
Galeon	CSS2	Yes	Yes	Yes	Yes	—	—	Yes
Internet Explorer	Partial	Yes	Yes	Yes	No	Yes	No	Yes
K-Meleon	CSS2	Yes	Yes	Yes	Yes	—	—	Yes
Konqueror	CSS2	Yes	Yes	Yes	Yes	No	Yes	Yes
Links (Non-graphical)	No	Yes	No	Partial	No	—	—	No
Lynx (Non-graphical)	No	No	No	No	No	—	—	No
Mosaic	No	No	No	No	No	—	—	No
Mozilla	CSS2	Yes	Yes	Yes	Yes	—	—	Yes
Mozilla Firefox	CSS2	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Netscape	CSS2	Yes	Yes	Yes	Yes	—	—	Yes
OmniWeb	CSS2	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Opera	CSS2	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Safari	CSS2	Yes	Yes	Yes	Yes	Yes	Yes	Yes
WorldWideWeb (Non-graphical)	No	No	No	No	No	—	—	No
	CSS	Frames	Java	JavaScript	Tabbed browsing	Download manager	Toolbar searching	Pop-up blocking

FIG. 3.1 – Comparaison de navigateurs Web

Cette liste de fonctions proposées par les différents navigateurs est fournie à titre informatif. Elle provient du site www.wikipedia.org et a été modifiée par nos soins pour des raisons de mise en page. Nous regrettons qu'elle n'aborde pas les aspects de navigation des navigateurs Web.

3.1.1 Historique des différents navigateurs

Comme cité dans le chapitre précédent, en 1993, Mosaic a été le premier navigateur graphique à avoir été développé. Il est suivi en octobre 1994 par Netscape Navigator qui rapidement s'est octroyé la quasi totalité des parts de marché. Après avoir remarqué ce succès, Microsoft se met dans la course aux navigateurs et sort en août 1995 une première version d'Internet Explorer. Microsoft commence alors à grignoter des parts de marché et en 1997, lorsqu'ils sortent la quatrième version de leur navigateur, ils en détiennent 18% alors que Netscape Navigator en détient 72%. C'est alors que commence une véritable guerre des navigateurs axée sur la recherche de nouvelles fonctions, plutôt que sur la stabilisation des développements effectués. Il en résultera des produits de mauvaise qualité, tant chez Microsoft que chez Netscape. De plus, le non-respect par ces deux entreprises des standards présents et validés par le *World Wide Web Consortium*¹ (W3C), engendrera de multiples problèmes. Nous pouvons nous rappeler des problèmes d'affichage avec les tags HTML `<div>` et ``.

Les chiffres cités précédemment concernant les parts de marché des principaux navigateurs nous montrent qu'aujourd'hui Microsoft a réussi à s'imposer dans ce domaine. Nous remarquerons quand même le retour de Netscape, à travers Mozilla son successeur, qui se trouve à nouveau dans une phase de gain de popularité.

D'autres navigateurs actuels sont dignes d'intérêts quant à certaines de leurs fonctionnalités, mais leur utilisation étant trop marginale, nous en tiendrons moins compte. Pour citer quelques exemples :

- Opera
- Safari
- OmniWeb
- Konqueror

3.1.2 Les fonctionnalités d'aide à la navigation

Les différentes fonctionnalités de base que l'on retrouve dans tous les navigateurs sont les suivantes :

Les boutons "Précédente" et "Suivante", ou "*Back*" et "*Forward*", sont les fonctions les plus employées par les utilisateurs. Elles sont simples et leur utilisation consiste en un simple clic.

¹Consortium fondé en octobre 1994 chargé de définir les normes (standards) du *World Wide Web* telles que HTML, XHTML, XML, CSS, etc. [37]



FIG. 3.2 – "Précédente" et "Suivante"

Ces boutons se situent dans la barre de navigation supérieure du logiciel Internet Explorer.

Cette fonction offre la possibilité de sortir des "voies sans issue", ainsi que de corriger les erreurs de navigation en revenant sur nos pas. Il faut préciser que 2/3 des pages visitées sont quittées de cette manière [3] et que 30% des actions effectuées lors de la navigation consistent à utiliser le bouton "*Back*" [14]. Par exemple, lorsqu'un utilisateur se rend compte que la page visitée ne correspond pas à ses attentes, il pourra rebrousser chemin sur une page plus connue à l'aide de cette fonction. Ces boutons permettent aussi de se déplacer dans une suite de pages visitées lors d'une navigation. Les utilisateurs ont souvent une représentation mentale naïve des fonctions "*Back*" et "*Forward*", ils sont parfois surpris de leurs comportements. Etant donné que ces fonctions sont basées sur des piles², lorsqu'un utilisateur revient en arrière dans sa navigation à l'aide de la fonction "*Back*" et qu'il repart sur de nouvelles pages, une partie de son ancien chemin de navigation ne sera plus disponible.

²Dans ce cas, une pile est une liste d'adresses Internet dont les propriétés sont identiques à une pile quelconque : les éléments ajoutés viennent sur le dessus et les éléments supprimés sont également pris au dessus de la pile. Le terme anglais de ce concept est *First In Last Out* (FILO). Une représentation parlante est l'utilisation d'une pile d'assiettes.

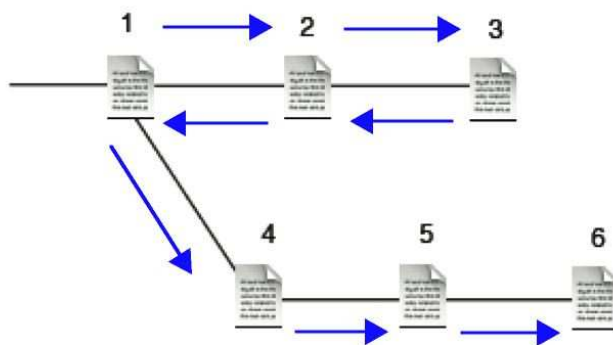


FIG. 3.3 – “Hub and spoke”

Dans cet exemple, l'utilisateur commence sa navigation par le document 1 et va jusqu'au 3 (pile de l'historique : 1,2,3). Il décide de revenir au document 1 à l'aide du bouton “Back” (pile de l'historique : 1,2,3). C'est alors qu'il change d'orientation et part pour le document 4 et va jusqu'au 6 (pile de l'historique : 1,4,5,6). Si l'utilisateur veut revenir au document 2 ou 3, il sera face à un problème. Le fait de revenir en arrière dans la navigation, donc de redescendre dans la pile jusqu'au document 1, et de continuer dans une autre voie, va remplacer les anciennes adresses qui avaient été ajoutées par les nouvelles. Dans ce cas, le document 1 est un *hub* auquel sont reliées des branches ou *spoke*.

Un autre problème est, par exemple, l'utilisation de *Common Gateway Interface* (CGI) ou de *Javascript* qui permettent de prendre en compte des variables et de faire des transitions automatiques entre des pages. Comme la fonction “Back” n'informe pas le serveur Web des déplacements de l'utilisateur, ce dernier peut se retrouver à recharger des pages avec des valeurs de variables différentes [3]. Par contre, le titre de la page ne change pas. Cela a pour effet d'afficher plusieurs occurrences du nom de la page en question dans l'historique et l'utilisateur ne sait plus forcément quelles étaient les informations affichées dans chaque page. Comme le font remarquer Saul Greenberg et Andy Cockburn dans leur article “*Getting Back to Back : Alternate Behaviors for a Web Browser's Back Button*”, les pages visitées qui ne sont plus accessibles à l'aide des ces boutons ne sont pas forcément dignes d'intérêt [26]. Nous pensons, tout comme eux, qu'il est risqué d'avancer un argument pareil car il n'est pas toujours vrai que l'utilisateur ne veuille pas revisiter une de ces pages “perdues”.

Les “favoris” sont des raccourcis qui permettent d'accéder directement à certaines pages. Ils représentent 3% des actions effectuées dans les navigateurs [14]. Il existe plusieurs types de “favoris” : certains sont visibles en premier plan dans

un navigateur, alors que d'autres sont accessibles depuis un menu déroulant. Pour vouloir ajouter une page dans les "favoris", il faut être conscient de notre intérêt pour celle-ci. Dans le cas contraire, il sera beaucoup plus difficile pour un utilisateur de retrouver cette page. Un autre aspect négatif de cette fonction est la maintenance qu'elle nécessite. Par peur de ne pas retrouver une page, nous avons tendance à toujours en ajouter dans les favoris sans pour autant effacer celles qui ne nous sont plus utiles.

L' "Historique" recense toutes les pages visitées au cours d'une navigation. Celles-ci deviennent accessibles via un menu déroulant et peuvent être triées de différentes manières : dates, fréquences, regroupement par sites visités, etc. Cette fonction est simple d'utilisation et représente 1% des actions effectuées dans les navigateurs [14]. Un des problèmes lié à cette fonction est le nom donné à certains documents hypertextes. Celui-ci est souvent le même pour la totalité des pages d'un site. De plus, il n'est pas toujours parlant par rapport au contenu du document. Il n'est donc pas toujours évident de retrouver la page qui nous intéresse dans une grande liste. L' "Historique" est pratique pour revisiter les pages loin en arrière dans une session, au contraire du bouton "Back" qui lui nécessite beaucoup de clics pour effectuer la même tâche.

Ces trois fonctions de base ont une certaine importance, dans le sens où leur but commun est de pouvoir aider les utilisateurs à retourner sur d'anciennes pages visitées. D'après une étude menée par L. Tauscher et S. Greenberg, dont ils parlent dans l'article "*Revisitation patterns in World Wide Web Navigation*", 60% des pages Web sont revisitées et un nombre restreint d'entre elles est fréquemment visité par les utilisateurs [17], d'où l'importance de l'aide fournie par ces fonctions de base.

3.1.2.1 Particularités de certains navigateurs

Un des aspects de navigation qui est toujours problématique est la recherche d'informations sur le Web. Nous remarquons un nombre important de *plugins* qui font appel aux moteurs de recherche et qui essaient de combler ce problème. Mozilla et son nouveau navigateur Firefox a directement intégré cette fonction de recherche. De plus, des fonctions de "recherche dans la page" plus accessibles que dans Internet Explorer permettent de trouver plus facilement le contenu souhaité s'il est dans une longue page Web.

Parmi les nouvelles fonctions qui aident à la navigation nous pouvons mentionner Apple avec son navigateur Safari et son "*SnapBack*". Cette fonction permet de revenir à un point crucial de la navigation. Ces points sont notamment :

- Des résultats de moteurs de recherche
- La première page d'un serveur (p. ex. www.cnn.com)

Opera, dans sa version 7, a implémenté des fonctions appelées "*Fast-Forward*" et "*Rewind*". La première analyse le texte dans une page pour logiquement proposer la prochaine page à visiter. Elle utilise des mots-clés tel que "*next*" pour suivre le lien qui lui est attaché. La seconde fonction est similaire au "*SnapBack*" d'Apple.

Malheureusement, ces fonctions ne sont pas vraiment intuitives pour les utilisateurs lambda, sans avoir lu leur descriptif il n'est pas facile de comprendre où elles nous emmènent et par conséquent semblent relativement peu utilisées.

Une autre amélioration apportée par Opera est la création à la volée d'une liste des liens disponibles dans une page.

Firefox quant à lui propose la possibilité de naviguer sur plusieurs pages à la fois grâce à des onglets. Cette option, très appréciée des utilisateurs, comporte un problème majeur : plusieurs sessions de navigation se passent en parallèle. Cela rend l'utilisation de l'historique et de la fonction "Back" non-intuitive.

La société "The Omni Group" propose, à travers la cinquième version de leur navigateur OmniWeb, une fonction d'onglets identique à celle de Firefox mais avec un *thumbnail*³ représentatif de la page en cours de visite. Chaque nouvel onglet est donc affiché sous la forme d'une image, celle-ci se mettant à jour automatiquement tout au long d'une session de navigation. Cette fonction fournit une aide visuelle à l'utilisateur en lui apportant plus de clarté pour se retrouver dans ses différentes sessions.



FIG. 3.4 – Le navigateur OmniWeb

Les onglets, représentés par des *thumbnails* et placés en colonne, sont dans la partie gauche du navigateur.

³Un *thumbnail* est une miniature souvent utilisée dans le *World Wide Web* pour représenter, soit une plus grande image, soit une page dans le cas de ce navigateur. Sa taille est, comme son nom l'indique, environ de celle de l'ongle du pouce.

3.1.3 Recherches effectuées

Certaines voies dans l'amélioration de la navigation ont déjà été explorées. C'est le cas pour l'accès aux moteurs de recherche. Même si ce domaine a beaucoup été traité, il reste d'actualité avec le développement de "LongHorn", la nouvelle version du système d'exploitation de Microsoft, ainsi qu'avec la guerre que mène Google face au géant de Redmond.

Un autre aspect qui a été traité est l'intégration des trois fonctions de base que sont l'historique, les favoris et le "Back" d'un navigateur. Cette recherche met en valeur la pertinence d'une liaison entre le bouton "Back" et un "historique modifié". Pour l'illustrer, le laboratoire GroupLab de l'Université de Calgary [14] a développé un *plugin* pour Internet Explorer. Il propose ces modifications et d'autres comme la création de *thumbnails* représentant les pages visitées. Pour l'instant, ces propositions n'ont pas encore été reprises par les développeurs d'Internet Explorer. Nous verrons si elles apparaissent dans une prochaine version.



FIG. 3.5 – *Plugin* intégrant les trois fonctions de base

Nous remarquons que des *thumbnails* sont disponibles pour chaque page se trouvant dans l'historique. De plus il est possible de définir l'importance de chaque page, afin de pouvoir les trier selon ce critère (c.f. la barre verticale disposée à côté du *thumbnail* lui donne son degré d'importance).

Une autre voie explorée est celle des navigateurs hyperboliques [35]. C'est une approche que nous trouvons intéressante mais elle souffre d'être bimodale.

Premièrement nous cherchons la page que nous souhaitons visiter et ensuite nous l'affichons. Ces navigateurs permettent de bien visualiser les liens entre les pages, ainsi que le chemin que l'on peut parcourir depuis notre position, mais ils ne disposent pas d'un historique. C'est aussi intéressant d'avoir adopté une approche égocentrée qui nous semble assez représentative de nos comportements sociologiques. Ce navigateur est expérimental et n'est pas encore prêt à être utilisé par le grand public.

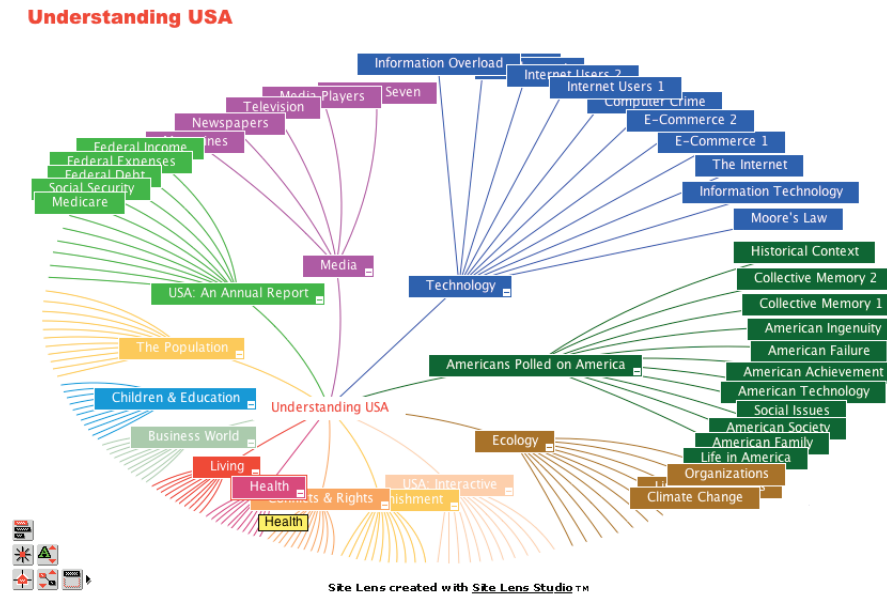


FIG. 3.6 – Navigateur hyperbolique

Le navigateur est centré sur la page courante soit “*Technology*” dans ce cas. Cette dernière est reliée à la page d'accueil ou à une page précédente, ainsi qu'aux pages disponibles depuis celle-ci.

3.2 Constat du manque

Nous avons constaté que chaque navigateur a ses forces et ses faiblesses. Si nous prenons le meilleur des navigateurs, il serait déjà possible d'améliorer considérablement la navigation au niveau :

- De la consistance de l'historique.
- De l'aspect visuel de l'historique.
- De la consultation d'une page déjà visitée.
- De la recherche dans une page.

Des outils pour pallier à ces problèmes existent, mais ne sont pas ou plus implémentés dans les navigateurs populaires actuels [36].

Un autre aspect digne d'intérêt est la nécessité de comprendre la structure d'un site, ce que beaucoup d'utilisateurs ne cherchent pas à faire, soit par paresse, soit par manque d'intérêt. Par conséquent, des outils aidant à se situer dans la navigation, à un niveau visuel, seraient certainement un des aspects les plus importants à étudier. Cela permettrait à l'utilisateur d'aborder plus facilement tous les sites d'une manière similaire au niveau du navigateur.

La gestion des sessions est relativement bien implémentée dans les navigateurs les supportant, mais nous pouvons remarquer l'absence de classement par sessions dans l'historique général. Cela permettrait à un utilisateur de retourner, selon sa logique personnelle, dans les pages qu'il a visitées au cours d'une session.

Malgré tout les aspects qui ont été étudiés, nous pouvons toujours faire le constat des manques des navigateurs. Comme nous n'avons pas trouvé de solution qui couvre les aspects les plus importants de la navigation, nous avons commencé à développer quelques idées de fonctions. Elles sont détaillées dans le chapitre suivant.

Chapitre 4

Solution

4.1 Introduction

Nous allons essayer d'apporter une réponse effective à plusieurs problèmes ; le plus important étant sans aucun doute : “*Lost in hyperspace*”. Cet aspect est très vaste et très complexe, il comprend de nombreux sous-problèmes. C'est pourquoi, dans le but d'aider l'utilisateur à mieux se situer, nous essayons de lui fournir des informations sur sa provenance, sa localisation actuelle ainsi que sur les différentes possibilités de déplacement dans sa navigation. Ces éléments répondent aux questions de situation posées par Alan Dix (c.f. section 2.1.4). A travers les points suivants, nous présentons quelques solutions dans l'optique de remédier à cette sensation de perte des utilisateurs.

4.2 *Thumbnail*

Le *thumbnail* est une solution que nous avons retenue pour améliorer l'historique. Ce dernier est composé d'une liste de liens visités, chacun d'eux y est représenté par le titre de la page chargée. Malgré l'utilité d'un tel recensement, il n'est pas toujours évident de retrouver une page par son titre, certains n'étant pas vraiment parlants et d'autres étant identiques pour toutes les pages d'un même site. C'est pourquoi, nous proposons cette amélioration. Ces *thumbnails* sont propres à chaque pages, ce qui permet de bien distinguer ces dernières entre elles. Ils permettent aussi une meilleure compréhension visuelle de la navigation effectuée par un utilisateur, à savoir sa provenance. D'autant que, plus le graphisme d'un site est reconnaissable et particulier pour un utilisateur, plus il sera facile de reconnaître le *thumbnail* le représentant. Ces images permettent donc de retrouver plus facilement un site visité.

Lorsque l'utilisateur fait appel à l'historique, un cadre s'ouvre dans la partie gauche de l'espace de navigation. Il contient la liste des liens visités sous la forme d'un *thumbnail* avec le titre de la page en dessous (c.f. figure 4.1). Dès

que l'utilisateur positionne le curseur de la souris sur un *thumbnail*, ce dernier fait appel à la fonction *Scanning* (c.f. section 4.3).



FIG. 4.1 – Un *thumbnail* de l'historique visuel

L'historique visuel que nous proposons est composé d'une liste de *thumbnails*. Chacun représentant une page visitée.

4.3 *Scanning*

Le *Scanning* que nous proposons est une fonction qui parcourt le contenu d'une page, dans le but d'en retenir sa structure principale. Elle se résume à travers quelques balises HTML, tels que : `<title>` (titre), `<h1>` (titre de niveau 1), `<h2>` (titre de niveau 2), etc.

Ces informations sont regroupées dans une mini page, laquelle est affichée à la place du *thumbnail* lorsque l'utilisateur passe son curseur dessus. De plus, une liste des liens disponibles depuis la page sélectionnée apparaît sur les côtés de cette mini page. Cette liste est générée lors du chargement d'une page et est stockée pour pouvoir être réutilisée par la suite. Cette mini page est d'une taille légèrement supérieure à celle d'un *thumbnail* pour favoriser sa lisibilité.

Outre la meilleure compréhension de la structure d'un site par l'utilisateur. Cette fonction lui permet aussi de se faire une idée du contenu d'une page, sans pour autant devoir l'afficher dans le navigateur.

Lorsqu'un utilisateur fait appel à la fonction *Scanning*, en survolant avec le curseur un *thumbnail* présent dans l'historique visuel ou dans l'arbre de navigation (c.f. section 4.4), la page hypertexte est analysée et la structure de l'information est extraite de celle-ci (c.f. figure 4.2). Ces informations sont alors regroupées dans une mini page affichée à l'écran (c.f. figure 4.3).

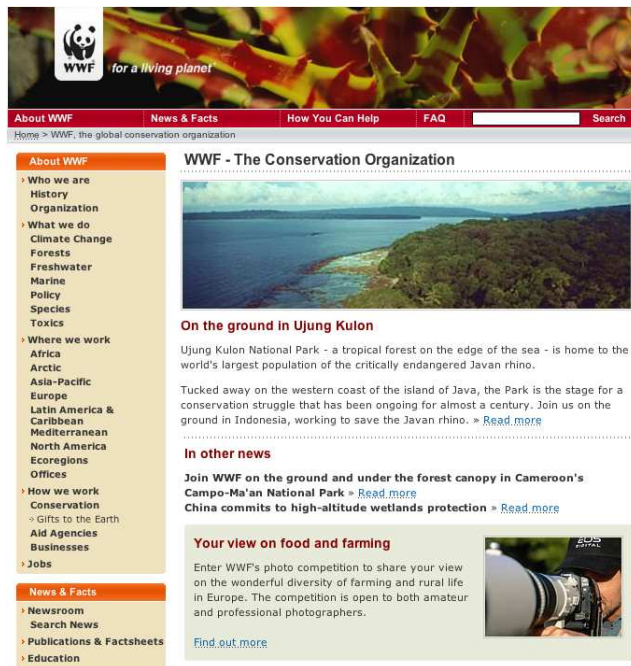


FIG. 4.2 – De la page HTML...

Pour la création d'une mini page, la fonction *Scanning* extrait les balises HTML (<title>, <h1>, <h2>, etc.) représentant la structure d'une page hypertexte ainsi que les différents liens présents sur celle-ci.

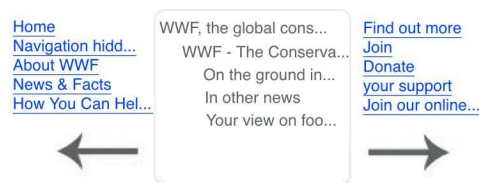


FIG. 4.3 – ... à la mini page

Voici un modèle de mini page. Nous remarquons dans la partie centrale, un résumé de la structure de la page hypertexte. Les sous niveaux sont décalés par rapport au titre et le nombre de caractères affichés est limité. Les parties latérales sont composées des liens présents sur la page hypertexte, les flèches permettent un déplacement vers la gauche ou la droite pour accéder aux autres liens disponibles. Elles ne sont visibles que dans le cas où elles peuvent être utilisées.

4.4 Arbre de navigation

L'arbre de navigation est une fonction permettant d'afficher, sous la forme d'une arborescence, les pages accessibles depuis la page courante. Cette fonction est accessible lors du survol, dans l'historique, du *thumbnail* de la page en cours de visite. Les feuilles de cet arbre sont des *thumbnails* représentant des pages (c.f. figure 4.1). Le survol par le curseur d'un de ces *thumbnails* fait appel à la fonction *Scanning* décrite ci-dessus.

Cet arbre de navigation répond au problème de localisation de l'utilisateur en l'aidant à mieux se situer dans un site. De plus, il favorise la compréhension des possibilités de navigation offertes depuis la page courante. L'aspect visuel fourni par l'utilisation de *thumbnails* permet de guider un utilisateur dans le cas où il revisiterait des pages connues, ou lorsqu'il s'attend à trouver des informations. Par exemple, le logo d'une marque reconnaissable dans le *thumbnail*.

L'utilisation de la fonction *Scanning* à travers l'arbre de navigation permet un accès à deux niveaux : le premier niveau représente les liens présents sur la page courante, alors que le second niveau se compose des liens disponibles sur les pages du premier niveau.

Prenons l'exemple d'une personne qui recherche différentes peintures d'un artiste sur le Web. En arrivant sur un site parlant du peintre concerné, l'utilisation de l'arbre de navigation peut faciliter les recherches ainsi que la navigation. Cette personne n'aura pas besoin de parcourir et afficher toutes les pages du site pour localiser les tableaux. Ils seront directement visibles sur les *thumbnails*, comme nous l'illustre la figure 4.4.

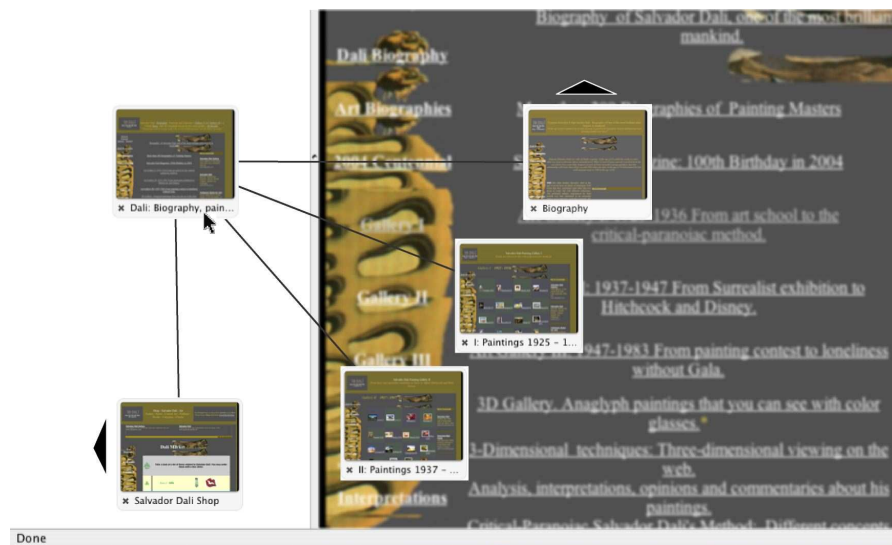


FIG. 4.4 – Arbre de navigation

Les “feuilles” de cet arbre (affichées en arc de cercle) sont des *thumbnails* représentant les pages disponibles depuis la page courante (représentée par le *thumbnail* en haut à gauche). La présence de flèches, aux extrémités de l’arc de cercle, est due au nombre supérieur de pages disponibles en comparaison au nombre de *thumbnails* affichés.

4.5 *Breadcrumbs*

La fonction *Breadcrumbs* se situe entre la barre d’outils du navigateur Web et la page en cours de visite (c.f. figure 4.5). Sa petite taille nous permet de la laisser affichée en permanence. Les *breadcrumbs* permettent de répondre au besoin de se situer dans la hiérarchie d’un site. De plus, ils empêchent le visiteur de se retrouver dans une situation d’impasse, en permettant de “remonter” au niveau supérieur d’un site. Ils permettent à l’utilisateur de gagner du temps en lui donnant un accès direct à plusieurs niveaux [39]. Ces niveaux sont basés, soit sur le nom des répertoires dans lesquels les pages sont enregistrées, soit sur le système des liens qui pointent sur cette page.

La figure 4.5 nous montre la localisation courante d’un utilisateur de la page d’accueil à la page courante. Les différents niveaux de cet exemple sont basés sur les noms des répertoires du site utilisé. La racine d’un site est dans ce cas renommée “*Home*”, pour que l’utilisateur améliore sa perception et sa compréhension de la structure du site qu’il visite.

Nous pouvons dire qu’à l’intérieur de la hiérarchie de ce site, l’utilisateur se

trouve à 3 niveaux de la racine. A partir de la page d'accueil du site ("Home") que l'on peut désigner comme le niveau 0, l'utilisateur s'est déplacé d'un niveau pour se retrouver dans la partie "About WWF". Depuis ce premier niveau, il s'est déplacé au second niveau "Where we work", puis a terminer sa navigation au niveau 3 "WWF working in Africa".



FIG. 4.5 – *Breadcrumbs*

Les *breadcrumbs*, à l'intérieur de la zone en rouge, permettent à un utilisateur de savoir en temps réel où il se trouve dans la hiérarchie d'un site.

4.6 *Sitemap*

La fonction *Sitemap* recherche un lien faisant référence au "plan du site"¹, soit par son nom, soit par l'*Uniform Resource Locator* (URL) qui pointe vers celui-ci. Si cette fonction ne trouve pas ce plan sur la page courante, elle charge la page racine du site et effectue à nouveau cette recherche. Dans le cas où ce plan ne serait toujours pas disponible, la fonction apparaît grisée et non utilisable.

La fonction *Sitemap* est accessible sous la forme d'un bouton qui se situe à gauche de la fonction *Breadcrumbs*. Elle révèle son utilité lorsqu'elle est utilisée avec l'arbre de Navigation.

Tant que cette fonction est inutilisable, le bouton *Sitemap* s'affiche grisé. Mais lorsqu'elle devient disponible, le bouton *Sitemap* est mis en évidence, comme nous le montre la figure 4.6. L'utilisateur n'a plus qu'à cliquer dessus pour que la page "plan du site" soit chargée comme page courante. Il peut alors naviguer clairement à travers la hiérarchie du site

¹Page représentant la structure d'un site.

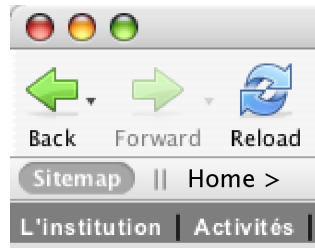


FIG. 4.6 – *Sitemap*

Le bouton faisant appel à cette fonction est mis en évidence lorsque celle-ci est utilisable.

4.7 Conclusion

Dans ce chapitre, nous avons établi quelques propositions de fonctions qui, selon nos idées, pourraient palier à certains manques encore présents dans l'utilisation des navigateurs Web actuels. Dans le chapitre suivant, nous proposons une évaluation de ces différentes fonctions dans le but d'étudier leur efficacité et de pouvoir affirmer ou infirmer concrètement leurs apports à la navigation.

Chapitre 5

Evaluation

5.1 Introduction

Pour pouvoir évaluer au mieux l'utilisabilité de notre prototype ainsi que son réel apport à la navigation, nous avons opté pour la méthode nommée "*Cognitive Walkthrough*¹" [41]. Cette méthode d'évaluation est basée sur le modèle cognitif humain de "l'apprentissage par l'exploration"².

Ce dernier est défini par Jean-François Richard comme un apprentissage effectué au cours de la réalisation d'actions qui ne sont pas de simples exécutions, mais qui comportent une part de résolution du problème auquel le sujet doit faire face [42]. L'ensemble des observations que nous allons effectuer lors des tests va nous permettre de mieux comprendre la perception par l'utilisateur des fonctions que nous lui proposons.

La navigation "normale" étant profondément ancrée dans les habitudes des utilisateurs, il est difficile de proposer une nouvelle manière de naviguer qui soit immédiatement adoptée. Nous nous attendons donc à obtenir des résultats biaisés, parce que les utilisateurs testant notre prototype risquent de ne pas toujours utiliser nos fonctions.

5.2 Pré évaluation

5.2.1 Scénario

Avant de se lancer dans la réalisation d'un prototype, nous avons pensé qu'il serait judicieux de faire une première évaluation de notre scénario par quelques utilisateurs à l'aide d'un navigateur standard. Cette évaluation va nous permettre d'avoir une première idée du comportement des utilisateurs et ainsi pouvoir mieux cibler les différents cas à traiter dans notre prototype.

¹Marche à suivre cognitive.

²Plutôt que de lire un manuel, l'utilisateur préfère souvent apprendre un nouveau système par "essai et erreur".

Lorsque nous avons commencé la rédaction du scénario, nous avons rapidement remarqué que l'utilisation et la recherche du site "parfait", à savoir basé sur certaines balises (h1, h2) et incluant un *sitemap* ne serait pas chose facile. En effet, un site comportant un nombre élevé de pages a de préférence un *sitemap*, mais n'utilise pas forcément des balises de base, mais plutôt des techniques d'affichage plus évoluées comme le *Cascading Style Sheets* (CSS). Au contraire, un site simple avec peu de pages peut plus facilement utiliser les balises concernées, mais n'a sûrement pas de *sitemap*. De ce fait, nous avons utilisé un seul site auquel nous avons ajouté un *sitemap* pour regrouper toutes nos idées dans le même prototype.

5.2.2 Remarques

Pour ce faire, nous avons choisi trois personnes et leur avons demandé d'effectuer notre scénario avec le navigateur Firefox. Ceci de la même manière que nous le ferons pour le test final. Les utilisateurs présents à cette pré évaluation ne participeront pas à l'évaluation finale.

Nous avons alors remarqué que pour avoir un prototype efficace, il faut diriger l'utilisateur dans sa navigation. Le problème est que cela ne nous permettra pas d'évaluer nos fonctions de manière objective. C'est pourquoi, nous devons trouver le juste milieu pour orienter l'utilisateur sans pour autant le guider. Le but étant d'obtenir de bons résultats tout en laissant l'utilisateur libre dans ses choix.

Nous avons aussi remarqué que certains termes utilisés dans notre questionnaire étaient trop spécifiques à nos recherches et de ce fait, peu explicites et mal interprétés par nos testeurs. De plus, nous avons constaté que les tâches proposées étaient plutôt destinées à des personnes ayant un minimum de connaissances sur Internet et sur la navigation Web. Cette dernière n'étant pas forcément évidente pour tous.

5.3 Prototype

5.3.1 Prototypage papier

Pour favoriser notre créativité et ne pas perdre du temps dans le développement de notre prototype, nous avons opté pour une première version papier. Elle nous a permis de bien mettre nos idées au clair et d'avoir la possibilité de créer plusieurs versions. La dernière version de notre prototype papier se trouve en annexe.

5.3.2 Technologie utilisée

Pour la création de notre navigateur factice, nous avons utilisé le programme Macromedia Flash MX [43]. Ce dernier nous a permis de développer un prototype associant des images et des animations. A l'aide de ces dernières, une

navigation limitée a pu être développée grâce aux actions³ que nous avons pu leur affecter suivant les cas de figures. Nous avons ainsi pu concevoir l'interface que représenterait notre navigateur en produit fini.

Ce prototype est fondé sur un fichier Flash qui contient la base de la structure de données. Ce fichier fait appel, charge et affiche les *thumbnails* ainsi que les pages en fonction de la navigation de l'utilisateur. Ces deux derniers éléments sont chacun des animations Flash à part entière. Le fichier de base est toutefois divisé en trois parties décrites ci-dessous et représentées visuellement dans la figure 5.1 :

- Une **barre de navigation** est située sur toute la largeur dans la partie supérieure du navigateur. Cette dernière provient du navigateur Firefox, avec lequel nous avons effectué nos tests. Un *screenshot*⁴ de cette barre de navigation a été effectué, puis importé comme image dans notre prototype.
- Notre **historique visuel** est situé dans la partie gauche. L'espace qui lui est consacré est légèrement inférieur à un tiers de la largeur globale de la fenêtre du prototype. L'importance de cet espace est due au fait que l'historique contient des *thumbnails* qui se transforment en mini page. Un minimum d'espace est nécessaire pour permettre une meilleure perception et compréhension des informations reçues par l'utilisateur. Cet historique est la représentation graphique d'une liste. Chaque nouveau *thumbnail* chargé est ajouté au début de cette liste, les autres *thumbnails* sont alors décalés d'un cran en arrière. De cette manière, nous pouvons afficher le dernier *thumbnail* chargé en bas de cet historique, les autres *thumbnails* étant décalés vers le haut. Cinq *thumbnails* sont affichés à la fois et l'utilisation d'une liste comme structure de données permet un déplacement dans cette dernière, de la même manière que dans l'historique d'un navigateur existant.
- Une **fenêtre de défilement** est située dans la partie droite et occupe plus des deux tiers de l'espace. Nous pouvons la considérer comme un cadre dans lequel les pages appelées sont chargées. Cette fenêtre permet un défilement vertical pour les grandes pages.

³A l'aide du langage *ActionScript*.

⁴Une capture d'écran.

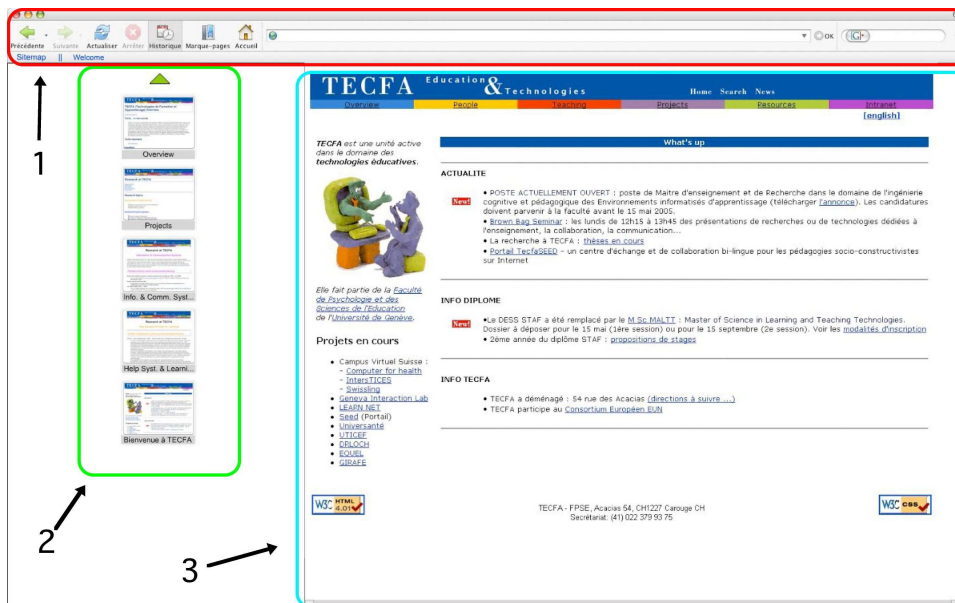


FIG. 5.1 – Différentes parties du prototype

- 1) La barre de navigation ; 2) L'historique visuel ; 3) La fenêtre de défilement

5.3.3 Solutions développées

Entre le moment où nous avons présenté les concepts de nos idées et le développement du prototype, nous avons remarqué que certaines améliorations de nos fonctions étaient envisageables. Nous présentons ci-dessous les fonctions développées et utilisées dans notre prototype, ainsi que les changements effectués. Pour nous aider dans cette démarche, nous avons précédemment défini les cas d'utilisation de nos fonctions. Ils sont disponibles en annexe.

5.3.3.1 *Thumbnail*

Nous avons remplacé le titre de la page concernée, situé en dessous de l'illustration, par le lien sur lequel l'utilisateur a/doit cliquer pour accéder à cette page.



FIG. 5.2 – *Thumbnail*

Dans ce cas, l'utilisateur doit cliquer sur le lien “*Projects*”, présent sur la page en cours, pour charger la page représentée par ce *thumbnail*.

5.3.3.2 *Scanning*

Pour une meilleure visibilité de l'ensemble, nous avons déplacé et diminué la taille des flèches. Ceci nous a permis d'agrandir la police des liens, tout en maintenant à cinq leur nombre de chaque côté. En cliquant sur une des flèches latérales, les dix liens affichés seront remplacés par les dix suivants.

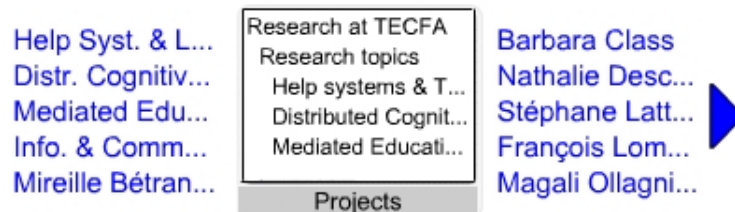


FIG. 5.3 – *Scanning*

Lorsque l'utilisateur se déplace avec le curseur sur le *thumbnail*, ce dernier se transforme en mini page. Il peut soit cliquer sur un des liens présents sur le côté pour charger la page désirée, soit cliquer sur le *scanning* pour charger la page représentée par le *thumbnail* et le *scanning* affichés.

5.3.3.3 *Arbre de navigation*

Nous avons agrandi en partie l'arbre de navigation pour pouvoir afficher cinq *thumbnails*. Ces derniers sont disposés de telle manière, que la transformation de l'un d'eux en mini page n'entrave pas la visibilité des autres. Nous avons aussi ajouté une bordure qui délimite la zone d'action au-delà de laquelle l'arbre de navigation se referme. De plus, lorsque ce dernier est affiché, la page en cours de

visite reste affichée en arrière plan mais devient transparente pour ne pas gêner la lisibilité des mini pages au premier plan.

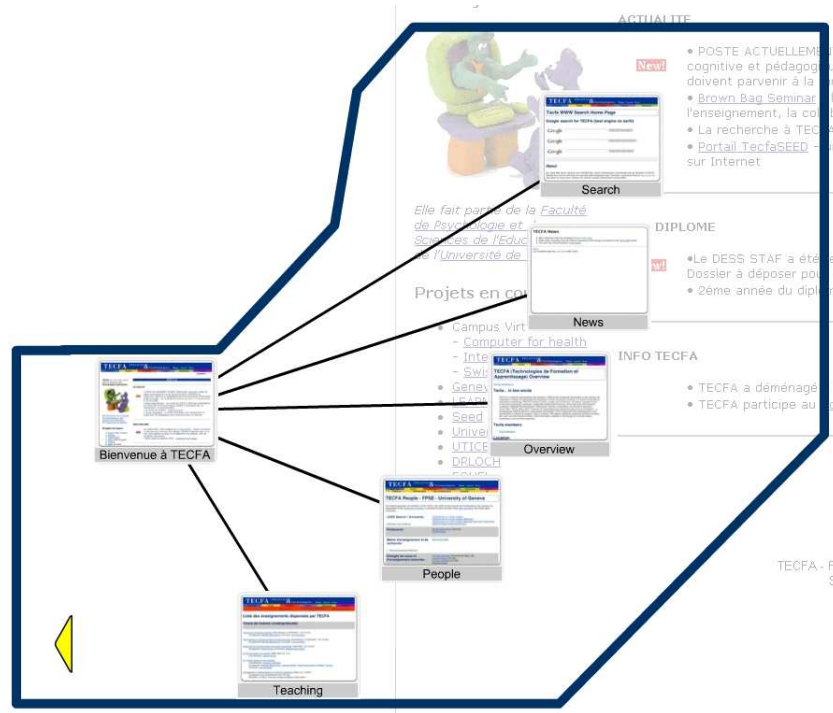


FIG. 5.4 – Arbre de navigation

Dans cet exemple, l'utilisateur peut cliquer sur la flèche de la zone inférieure de l'arbre de navigation pour afficher, un par un, les autres *thumbnails* disponibles.

5.3.3.4 Breadcrumbs

Les *Breadcrumbs* restent placés en dessous de la barre de navigation et sont visibles durant toute la navigation.



FIG. 5.5 – *Breadcrumbs*

Dans ce cas, l'utilisateur est dans le niveau "*Help Syst. & Learning*". Il peut à tout moment cliquer sur "*Projects*" ou "*Welcome*" pour remonter dans la hiérarchie du site.

5.3.3.5 Sitemap

Le lien "*Sitemap*" est affiché seulement si un "plan de site" est disponible pour le site en cours de visite. Il reste placé à la gauche des *Breadcrumbs*.



FIG. 5.6 – *Sitemap*

Le lien est visible, l'utilisateur peut donc cliquer dessus pour charger le *Sitemap* du site comme page courante. Il pourra ainsi utiliser l'arbre de navigation pour mieux comprendre la hiérarchie du site.

5.3.3.6 Vue d'ensemble

La figure suivante est un *screenshot* du prototype avec l'arbre de navigation ouvert et la transformation d'un *thumbnail* de l'arbre en mini page.

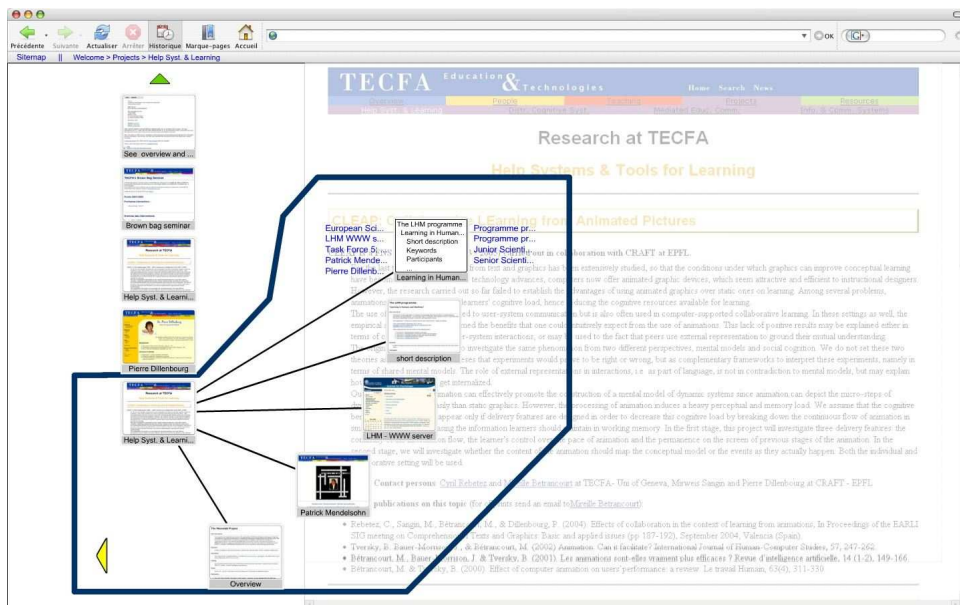


FIG. 5.7 – Prototype

5.3.4 Type d'actions

Du fait que la navigation offre une infinité de cas, être exhaustif dans notre prototype relève d'une durée de conception conséquente. C'est pourquoi, vu que les utilisateurs sont orientés sur les tâches à effectuer, nous traitons les trois cas d'actions suivantes :

1. L'action qui mène directement au succès.
2. L'action qui peut mener au succès en passant par un autre chemin.
3. L'action qui mène à l'erreur, en utilisant un chemin qui ne peut pas mener au succès, en empruntant une voie sans issue ou encore en étant hors sujet.

5.4 Participants

Le nombre de participants requis pour cette évaluation était de dix. Cinq personnes ont effectué le scénario proposé sur notre prototype et les cinq autres sur le navigateur Firefox.

Ce groupe était composé de personnes âgées de 23 à 36 ans dont la majorité a suivi des études universitaires. La moitié de ces personnes travaillent ou étudient dans une branche informatique. L'ensemble des personnes utilise quotidiennement un ordinateur et leur fréquence moyenne d'utilisation d'Internet se monte à environ 1 heure 30 par jour. De plus, elles ont des connaissances de base en langue anglaise, certaines pages du site utilisé n'étant pas en français.

5.5 Méthodologie

5.5.1 Familiarisation

Avant de commencer le scénario d'évaluation de notre prototype, les participants ont reçu une introduction pendant quelques minutes (de 5 à 10 minutes) sur les nouvelles fonctions que nous proposons. Ils ont eu le choix d'utiliser nos fonctions ou de naviguer librement. Toutefois, la fonction “*Back*” ainsi que les pages du site qui ne nous étaient d'aucune utilité n'ont pas été implémentées dans le prototype.

5.5.2 Description d'une tâche

Il est important de prendre en compte les points suivants pour chacune des tâches qui composent notre scénario [41] :

1. Définir des conditions initiales.
2. Définir le critère de succès.
3. Spécifier une durée maximum de réussite, après laquelle une aide peut-être donnée.
4. Définir ce qui constitue une erreur.

5.5.3 Tâche 1 - *Scanning*

A l'aide de la fonction “arbre de navigation”, l'utilisateur doit pouvoir nous fournir des informations sur le contenu d'une page disponible depuis la page la page d'accueil du site. Pour ce faire, il doit/peut utiliser la fonction “*Scanning*” sur un des *thumbnails* qui s'affichent dans l'arbre de navigation.

5.5.4 Tâche 2 - Arbre de navigation

A l'aide de la fonction “arbre de navigation”, l'utilisateur devrait pouvoir s'orienter correctement à travers le site pour charger une page prédéfinie par nos soins. Cette page se trouve au niveau 3 de la hiérarchie du site. Lorsque l'utilisateur se trouve au niveau 0 du site (page d'accueil), elle n'est pas directement accessible avec l'utilisation de “l'arbre de navigation”. De plus, l'utilisateur est amené à charger d'autres pages avant de pouvoir rechercher la page concernée. L'utilisateur doit donc naviguer à travers le site avant d'accéder à cette page.

5.5.5 Tâche 3 - Historique visuel

Demander à l'utilisateur de recharger une page anciennement visitée, si possible à l'aide de l'historique visuel.

5.5.6 Tâche 4 - *Breadcrumbs*

Demander à l'utilisateur de nous fournir le niveau de la page qu'il a du recharger à la tâche 3.

5.5.7 Tâche 5 - Représentation graphique du chemin de navigation

Demander à l'utilisateur de nous fournir une représentation graphique du chemin de navigation qu'il a parcouru.

5.5.8 Tâche 6 - *Sitemap*

Demander à l'utilisateur de nous donner les grandes lignes de la structure du site. Il peut s'aider de la fonction *Sitemap* qui est disponible dans notre prototype ainsi que dans le navigateur utilisé.

5.6 Matériel

Pour effectuer tous nos tests, nous avons utilisé un ordinateur portable avec écran panoramique de 15,4 pouces. Nous avons créé une fiche d'observation qui nous a permis de relever :

- Le temps de chaque action effectuée par un participant lors de sa navigation.
- Diverses observations.
- Les remarques des utilisateurs.

De plus, nous avons fourni aux utilisateurs un questionnaire, disponible en annexe, comportant :

- Les tâches à effectués.
- Quelques questions post-expérimentales.

5.7 Hypothèses

Pour pouvoir valider nos tests, nous nous sommes basés sur une hypothèse que nous allons soit infirmer soit confirmer suivant les résultats obtenus. Malheureusement, il n'existe pas de moyen pour mesurer l'utilisabilité d'un système de manière absolue et c'est pourquoi nous allons utiliser plusieurs critères.

Nous sommes partis de l'hypothèse que notre projet apporte une certaine structure dans la navigation de l'utilisateur, celle-ci lui permettant une meilleure compréhension et orientation dans ses choix de navigation. Cette hypothèse se décline suivant les fonctions que nous utilisons :

1. Les *thumbnails* devraient permettre de retrouver une page suivant son apparence graphique sans avoir besoin de lire son titre où de la consulter. Ils devraient faire appel à la mémoire visuelle de l'utilisateur et permettre ainsi d'augmenter sa vitesse de navigation.

2. Le *Scanning* devrait nous permettre d'avoir une idée de la structure du contenu d'une page sans avoir à l'ouvrir.
3. L'arbre de navigation devrait faciliter la localisation d'un utilisateur dans un site, en lui permettant de mieux visualiser les liens hiérarchiques entre les pages.
4. Les *Breadcrumbs* permettraient de se situer instantanément dans la hiérarchie du site. Ils fourniraient également une aide à la navigation en offrant un moyen pour changer de niveau.
5. La fonction *Sitemap* devrait favoriser la perception de la structure globale d'un site Web.

En proposant une telle démarche, nous espérons répondre en partie aux problèmes multiples du "*Lost in hyperspace*".

5.8 Résultats

Toutes les tâches ont été effectuées sur le site de TECFA⁵. Ce site a été choisi pour sa complexité et sa navigation peu intuitive.

5.8.1 Tâche 1

Pour rappel, la tâche une consistait à donner la structure informationnelle d'une page. Les résultats que nous avons considérés comme justes étaient les plus précis, nous leurs avons attribué un total de 100%. Une réponse fautive équivalait à 0%. Entre ces extrêmes, nous avons attribué des valeurs intermédiaires suivant la qualité de la réponse.

Pour les personnes qui ont utilisé le prototype, nous avons obtenu un taux de réussite moyen de 68.75%. Ce dernier est de 73,33% pour les personnes qui ont utilisé le navigateur. Avec des résultats aussi proches et un nombre de testeurs si restreint, il est difficile de se prononcer sur un quelconque résultat. Nous observons toutefois une tendance défavorable pour notre prototype. Cette dernière peut très facilement varier dans un sens comme dans l'autre.

Notons qu'une personne ayant effectué notre scénario sur le prototype n'a pas fait usage de nos fonctions. Par conséquent, nous avons pris en compte son résultat dans l'autre catégorie.

⁵Technologies de Formation et Apprentissage - <http://tecfa.unige.ch>

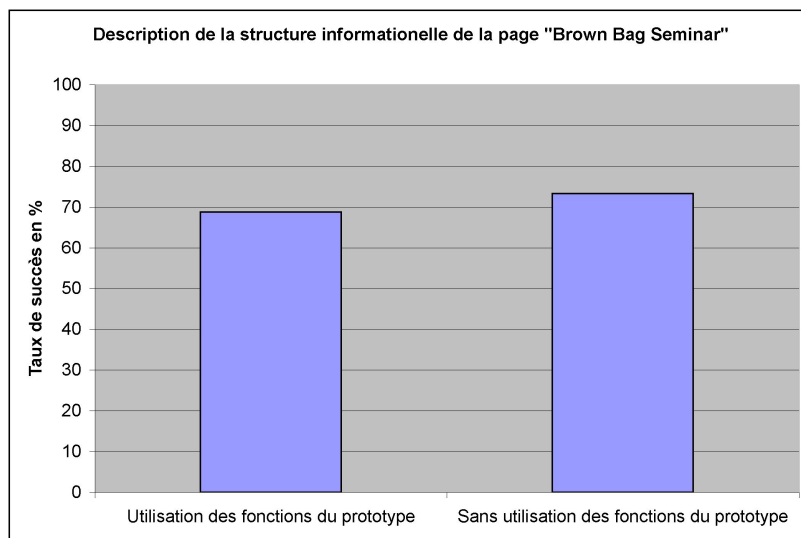


FIG. 5.8 – Tâche 1 - Description de la structure d'une page

5.8.2 Tâche 2

La seconde tâche consistait à charger une page située assez bas dans la hiérarchie du site. Nous avons chronométré les utilisateurs lors de leurs parcours. Comme l'ensemble des participants ont réussi cette tâche, les critères que nous avons pris en compte sont la durée totale et le nombre de pages visitées pour accéder à la page demandée. Il semble logique que les personnes ayant mis le moins de temps et chargé le moins de pages aient obtenu les meilleurs résultats.

Dans l'ensemble, les utilisateurs du prototype ont mis en moyenne 353,75 secondes soit 5'54", avec un écart type de 1'48". Alors que les utilisateurs du navigateur ont mis en moyenne 251 secondes soit 4'11", avec un écart type de 2'26".

Les résultats bruts que nous avons obtenu ne sont pas très représentatifs. En les étudiant de plus près, nous observons la présence de sous groupes tant pour le prototype que pour le navigateur (c.f. figure 5.9).

En se concentrant sur la catégorie "prototype", nous constatons que des personnes ont mis 262,5 secondes soit 4'22" ($\pm 31,8$ ") alors que d'autres ont mis 445 secondes soit 7'25" ($\pm 35,3$ "). Nous observons le même genre de répartition dans la catégorie "navigateur". Un sous groupe a mis 162,7 secondes soit 2'42" ($\pm 52,9$ ") et l'autre 427,5 secondes soit 7'07" ($\pm 74,2$ ").

Comme à la tâche précédente, un utilisateur n'a pas fait usage de nos fonctions. Nous l'avons donc intégré à la catégorie "navigateur".

La différence entre les deux sous groupes que nous avons observé, pourrait être due à une mauvaise compréhension de la tâche par une partie des testeurs.

D'après ces résultats, nous constatons toutefois que la vitesse de navigation est plus lente avec notre prototype.

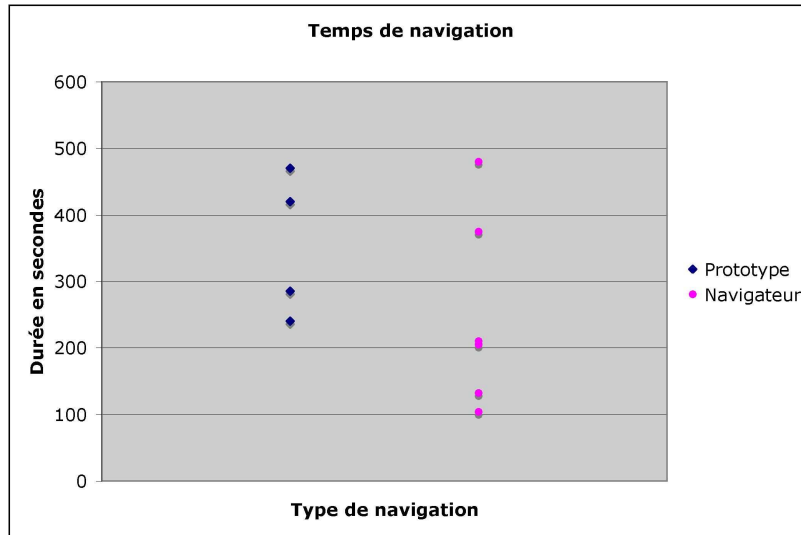


FIG. 5.9 – Tâche 2 - Durée de navigation

En ce qui concerne le nombre de pages visitées, la moyenne est de 6 pages ($\pm 1,8$ pages) pour les utilisateurs du prototype contre 5,1 pages ($\pm 1,5$ pages) pour le navigateur. Nous constatons que les personnes ayant utilisé le prototype visitent en moyenne 1 page de plus.

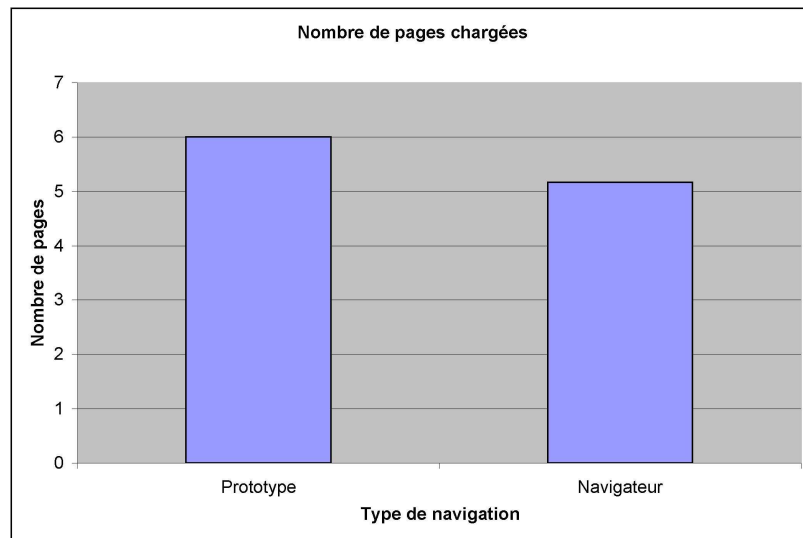


FIG. 5.10 – Tâche 2 - Nombre de pages chargées

5.8.3 Tâche 3

La troisième tâche consistait à retourner sur une page précédemment visitée. Comme critère d'évaluation nous nous sommes basés sur la durée qu'on mis les utilisateurs pour revenir à une page sélectionnée depuis la dernière page visitée lors d'une session de navigation.

Avec le prototype, la durée moyenne est de 112" soit 1'52" (± 89 secondes) alors qu'avec le navigateur, elle est de 32" (± 11 secondes). En observant les résultats de plus près, nous pouvons observé deux sous groupes dans les utilisateurs du prototype (c.f. figure 5.11) :

- Ceux qui ont effectués un temps que nous qualifierons de "record", avec une durée moyenne de 16.5 secondes ($\pm 4,9$ "). Nous remarquons que l'ensemble de leurs temps sont inférieurs au meilleur temps de l'autre catégorie.
- Ceux dont la durée moyenne est de 176 secondes ($\pm 26,2$ "). Ce temps est largement plus long que le plus mauvais temps obtenu avec le navigateur.

Nous supposons que ces résultats sont dûs à une bonne compréhension par certains utilisateurs de nos fonctions et inversement pour les autres. Ces derniers nécessiteraient encore une période de familiarisation.

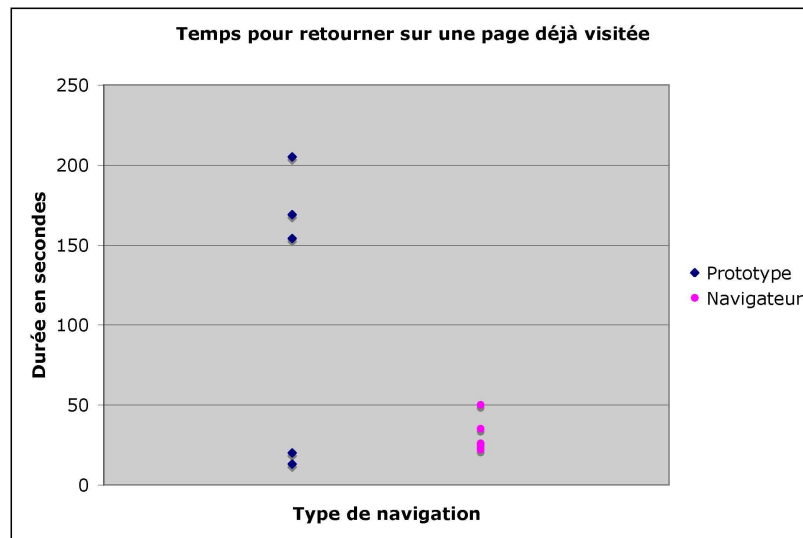


FIG. 5.11 – Tâche 3 - Temps nécessaire pour retourner sur une page visitée

5.8.4 Tâche 4

La quatrième tâche consistait à demander à l'utilisateur de nous fournir sa localisation dans la hiérarchie du site. Le résultat est considéré comme juste ou faux, il n'y a donc pas de valeurs intermédiaires.

Les 5 personnes ayant utilisé le prototype nous ont donné une réponse correcte, alors que 3 personnes sur 5 ayant utilisé le navigateur ont répondu de manière correcte. 4 utilisateurs du prototype se sont aidés des *breadcrumbs* et 1 utilisateur de "l'arbre de navigation". Pour pouvoir nous fournir une réponse, les utilisateurs du navigateur qui n'ont pas eu d'autres moyens que de naviguer dans le site. Nous supposons que les *breadcrumbs* ont bien été compris par les utilisateurs et que l'ajout de ces derniers dans un produit fini serait un avantage.

5.8.5 Tâche 5

L'avant dernière tâche faisait appel à la mémoire des utilisateurs. Nous leur avons demandé de nous décrire leur chemin de navigation. Les critères que nous avons évalué sont le nombre de pages dont ils se sont souvenu et leur séquence. Cette séquence a été évaluée de 0% à 100% selon la précision de la réponse. Nous avons observé que les utilisateurs du prototype se rappelaient mieux des pages qu'ils ont visitées.

En moyenne, les utilisateurs du prototype se souviennent de 4,6 pages ($\pm 2,07$ pages) tandis que les utilisateurs du navigateur se souviennent de 3,8 pages ($\pm 2,17$ pages).

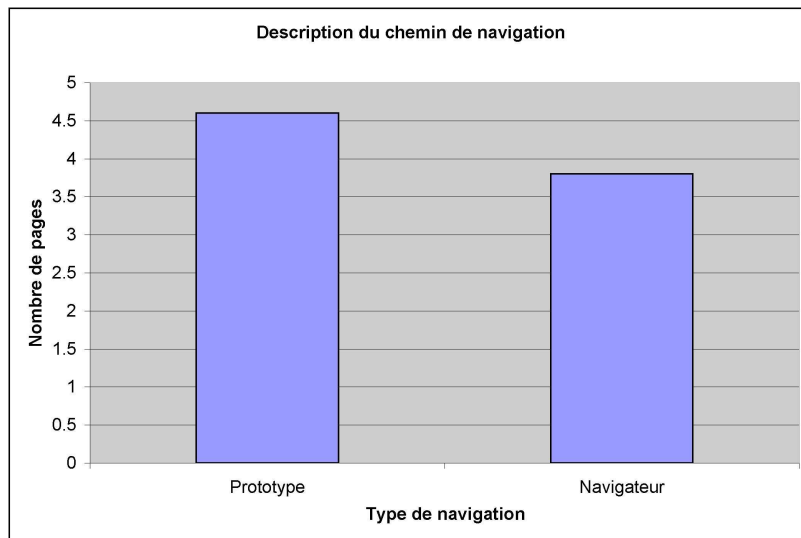


FIG. 5.12 – Tâche 5 - Nombre de pages citées sur l'ensemble du chemin de navigation

Les résultats concernant la séquence vont dans le même sens. Les premiers ont obtenu un taux de réussite de 50%, alors que les seconds ont obtenu un taux de réussite de 45%. Nous pouvons remarquer que ces résultats sont très proches, il est donc difficile d'en tirer avec précision une bonne interprétation.

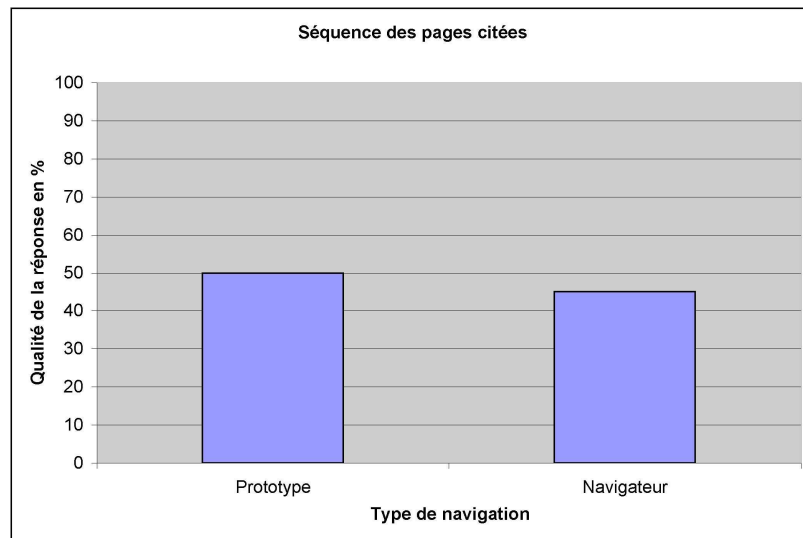


FIG. 5.13 – Tâche 5 - Séquence des pages citées

5.8.6 Tâche 6

La dernière tâche consistait à nous fournir la structure hiérarchique du site. Aucun *Sitemap* n'étant disponible pour le site utilisé, nous en avons créé un et l'avons rendu disponible, tant dans le prototype que dans le navigateur. Son introduction a été faite de la même manière pour tout le monde. Les utilisateurs ayant répondu correctement ont obtenu un total de 100%. Les utilisateurs ayant répondu faux ont obtenu un total de 0%. Entre ces extrêmes, nous avons attribué des valeurs intermédiaires suivant la qualité des réponses.

La moitié des utilisateurs ont utilisé la fonction *Sitemap* ont obtenu un total moyen maximum de 100% . Alors que les utilisateurs n'ayant pas utilisé le *Sitemap* ont obtenu un total moyen de 88%. Nous pensons qu'une fonction *Sitemap* disponible au niveau d'un navigateur favoriserait l'utilisation des plans de site ainsi qu'une meilleure compréhension de la structure d'un site.

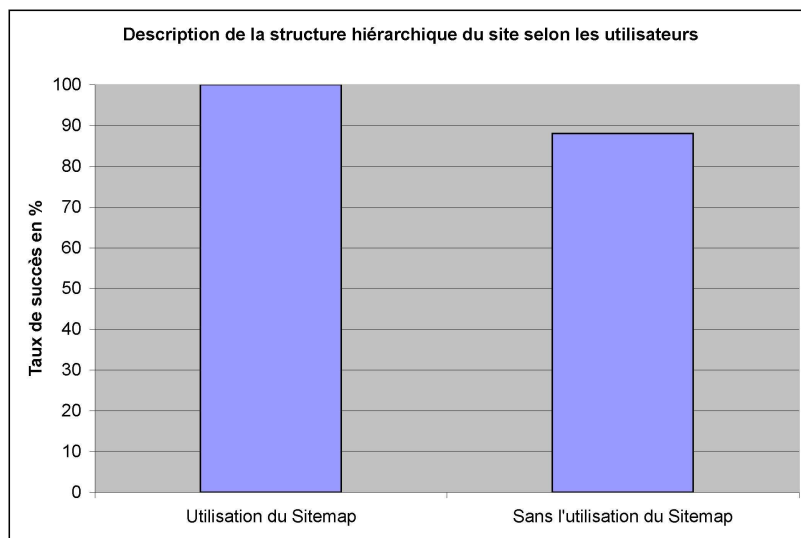


FIG. 5.14 – Tâche 6 - Description de la structure du site

5.8.7 Discussion

Avant de commencer nos tests, nous nous attendions à obtenir des résultats nous permettant de vérifier si notre prototype était meilleur ou moins bon que les autres navigateurs. Nous avons observé que les résultats n'étaient pas aussi déterminés que nous le souhaitions. Ces derniers n'étaient pas seulement quantitatifs mais aussi qualitatifs de par les commentaires des utilisateurs. Ces commentaires ont souvent été formulés à la fin des tests, lors du remplissage du questionnaire post-expérimental.

Nous avons également été confrontés aux habitudes de navigation des utilisateurs. Il est difficile de demander à un utilisateur d'oublier ses réflexes de navigation. Surfer sur le Web est une activité qui prend en moyenne 1 heure 30 par jour aux participants de nos tests. La découverte de notre prototype bouleverse leurs habitudes. Nous avons toutefois réussi à vérifier que l'utilisateur avait une meilleure image et compréhension de sa navigation malgré qu'elle soit plus lente. Ce point négatif aurait pu être diminué si nous avions consacré plus de temps à familiariser les utilisateurs à nos fonctions. Plusieurs d'entre eux nous ont fait part d'un besoin d'apprentissage pour l'utilisation de notre outil, par exemple en les laissant naviguer par eux-même un long moment sur un prototype similaire, en utilisant des exemples, une vidéo de démonstration, etc. La durée du test ne permet pas de dire que les fonctions ont réellement été utilisées, mais qu'elles ont plutôt été découvertes. Plusieurs personnes nous ont rapporté avoir saisi leurs fonctionnements à la fin du test.

À la question de savoir si nos fonctions étaient intuitives, nous avons reçu des avis positifs. Cependant, les personnes nous ayant donné un avis favorable n'ont

pas forcément mieux répondu aux questions que les autres. Nous avons constaté que l'historique visuel était plus intuitif que l'arbre de navigation. Ceci est peut-être dû au nombre important d'informations fournies par l'arbre de navigation et à la difficulté de se situer dans ces informations. Pour illustrer ce problème, un utilisateur nous a rapporté avoir eut peur de passer à côté d'informations importantes lorsqu'il utilisait l'arbre de navigation. La grande disponibilité d'informations submerge peut-être l'utilisateur novice et expliquerait le manque de succès rencontré par la fonction *Scanning*.

Les utilisateurs ont été confronté à différentes, voire trop nombreuses, possibilités de naviguer. Ils ont eu le choix entre l'utilisation de :

- L'arbre de navigation.
- L'historique visuel.
- La navigation standard.

Cela nous a fait remarquer le manque de repères de certains utilisateurs.

Selon nos hypothèses, nous avons pu prouver l'utilité de nos fonctions *Breadcrumbs* et *Sitemap*. Les *breadcrumbs* ne prennent pas beaucoup de place et offrent un moyen simple de se repérer à tout moment, c'est pourquoi ils ont été bien compris par les testeurs du prototype. Le *Sitemap* est rarement utilisé par nos testeurs sur des site "normaux", mais il a été chargé naturellement par la moitié des personnes lors de nos tests. Si ces deux fonctions étaient disponibles en standard sur les navigateurs, nous pourrions supposer qu'elles seraient encore plus utilisées et qu'elles favoriseraient la compréhension de la structure des sites.

A la question de savoir si les personnes ayant testé notre prototype, utiliseraient nos fonctions, la tendance générale était positive mais pas excessivement enthousiaste.

En plus de ces remarques, nous avons obtenu quelques commentaires qui méritent d'être cités. Par exemple, un utilisateur a remarqué que le nombre de clics pouvait être diminué grâce aux 2 niveaux disponibles depuis l'arbre de navigation. Plusieurs personnes nous ont dit que l'historique visuel était un plus. De même pour l'arbre de navigation qui permet de remplacer un menu "*drop down*" dans un site Web. Nous avons aussi eu comme écho que la navigation était agréable et que nos fonctions clarifiaient la structure du site.

5.8.8 Améliorations possibles

En dehors des nombreux aspects traités ci-dessus, de nombreuses améliorations seraient possibles et souhaitables dans une seconde version de notre prototype.

- Les problèmes de consistance des liens. Nous avons observé plusieurs testeurs qui recherchaient le titre de la page dans son *thumbnail*. Le titre en est absent car il a été remplacé par le lien permettant d'accéder au *thumbnail*. Ceci pour mettre en évidence la relation entre une page et son *thumbnail*. Nous pourrions envisager d'inclure le titre de la page dans le *Scanning* d'une page.
- La petite taille de la police utilisée lors de la transformation en mini page d'un *thumbnail* a été mentionnée plusieurs fois. La rendre paramétrable

permettrait de s'adapter aux besoins de chacun. Dans la même optique, nous pourrions paramétrer d'autres fonctions :

- Le nombre de *thumbnails* qui s'affichent dans l'arbre.
 - L'affichage optionnel des liens dans les *thumbnails*.
 - Le niveau de précision du *Scanning*.
 - Etc.
- Lorsque nous cherchions à afficher les liens disponibles sur une page, les liens trop longs étaient tronqués. Ceci posait le problème de liens similaires lorsque ceux-ci commençaient par la même chaîne de caractères. Nous pourrions résoudre ce problème en affichant l'intégralité du lien lorsque la souris se trouve au-dessus de ce dernier.
 - Concernant la page courante dans l'historique, nous n'avons pas utilisé la fonction *Scanning* car nous pensions que l'affichage de la page en grand serait suffisant. Ceci n'a pas été aussi clair pour tous, certains utilisateurs ont cherché à afficher une mini page en survolant, avec la souris, le *thumbnail* de la page courante. Comme solution, nous pourrions envisager de concilier la fonction *Scanning* avec l'arbre de navigation.
 - Dans l'arbre de navigation, les *thumbnails* étaient organisés selon leur ordre d'apparition dans une page. Certains liens peu visibles sur la page ont apporté une confusion lors de l'utilisation de l'arbre car les testeurs ne s'attendaient pas à les trouver à cet emplacement. La logique de la page n'est pas forcément celle que nous percevons. Pour résoudre ce problème, nous envisageons de donner une pondération entre les liens, suivant un critère d'importance. Par exemple, les liens dans les titres seraient privilégiés par rapport aux autres liens.
 - Lorsque les testeurs se déplaçaient dans l'historique visuel et qu'ils chargeaient une page, l'historique n'affichait pas immédiatement la page courante, soit la dernière page à avoir été chargée. Cela a créé de la confusion chez les testeurs et nécessiterait d'être corrigé. Nous avons aussi constaté que la chronologie de l'historique n'était pas intuitive pour tous. Il faudrait rajouter un outil visuel qui permette de se situer dans le temps.
 - Un problème moins dérangeant pour la compréhension, mais toutefois agaçant pour la navigation, était lorsqu'un utilisateur sortait de l'arbre de navigation. Ce dernier se réinitialisait au lieu de garder son état précédent. Si l'utilisateur sortait par mégarde de l'arbre de navigation, il devait le parcourir à nouveau pour retrouver les *thumbnails* précédemment affichés. Nous pourrions éviter ce problème en gardant la position de l'arbre de navigation en mémoire.
 - Un utilisateur nous a suggéré de regrouper les flèches de déplacement de l'arbre de navigation dans un module et d'y ajouter un bouton permettant de retourner au premier *thumbnail*. Ceci faciliterait la navigation et pourrait être appliqué par analogie aux mini pages et à l'historique visuel.

5.9 Conclusion

Avant d'entamer la partie de mise en pratique de nos idées, nous pensions que cette phase serait rapide et simple. Nous avons sous-estimé le temps nécessaire au développement d'un prototype ainsi qu'à son évaluation. Nous avons essayé d'appliquer une approche scientifique en suivant notre ligne conductrice à savoir l'analyse de la compréhension de la navigation par les utilisateurs. Malheureusement, plusieurs résultats que nous avons présenté sont partiels. Les résultats obtenus ne permettant pas d'être évalué d'une manière stricte, ils ont été évalués subjectivement suivant nos appréciations.

Selon cette approche, nous pouvons supposer que les utilisateurs qui ont testé notre prototype ont pu se créer un meilleur modèle mental de leur navigation, à l'inverse des personnes qui ont utilisé le navigateur. Toutefois, ces dernières étaient plus performantes dans leur vitesse de navigation. Nous n'avons pas déterminé la cause de cette rapidité, mais supposons que les personnes qui ont utilisé notre prototype étaient dans une phase de découverte de nos fonctions. Nous pensons qu'avec une familiarisation plus importante, nos fonctions pourraient être utilisées à meilleur escient. Ceci nous aurait apporté des résultats plus significatifs, de même que si nous avions forcé les testeurs à utiliser nos fonctions. Leurs habitudes de navigation n'auraient ainsi pas pu reprendre le dessus.

Chapitre 6

Conclusion générale

Le but initial de notre projet était de fournir une aide à la navigation, en imaginant de nouvelles fonctionnalités à intégrer dans un navigateur.

Nous avons débuté ce travail en introduisant l'hypertexte et la navigation Web pour les non-initiés. L'importance du contenu, de la présentation et de la structure de tout document (livre, magazine, page Web, etc.) nous est apparue très rapidement. Nous avons parlé des buts de l'hypertexte, de ses avantages et de ses inconvénients. Cela nous a permis de situer ses forces et ses faiblesses, dont le fameux problème "*Lost in hyperspace*" et de ses multiples dérivés. Nous nous sommes intéressés à comparer quelques navigateurs Web pour chercher les fonctions qui pourraient résoudre ces problèmes ou du moins les limiter. Nous avons trouvé plusieurs fonctions sensées corriger ces problèmes, mais elles ne nous ont pas empêché de constater un manque évident.

Nous pensions être sur un chemin en partie inexploré, mais au cours de nos différentes lectures et recherches, nous avons constaté qu'un nombre important d'études avaient déjà été réalisées dans ce domaine.

Malgré cela, nous avons créé un jeu de fonctions qui pourraient répondre partiellement à ce manque : *thumbnails*, *Scanning*, historique visuel, arbre de navigation, *Breadcrumbs*, *Sitemap*. Nous les avons imaginées sur du papier avant de les développer avec le logiciel Flash et de les faire tester par des utilisateurs.

Suite à nos évaluations, nous avons obtenu des résultats qui nous ont permis de supposer que nos fonctions ont apporté une aide au niveau de la localisation de l'utilisateur dans une navigation. Il faudrait toutefois encore améliorer le prototype en Flash suivant les commentaires des utilisateurs et refaire une série de tests. Cela devrait confirmer ou infirmer les tendances observées et ainsi nous dire si le développement d'un *plugin* comprenant nos fonctions pourrait être envisagé. A l'aide de ce dernier, il serait possible de vérifier, par une série de tests quantitatifs sur une longue période, si les fonctions répondent à un réel besoin.

Même si les ressources informatiques (taille des écrans, connexions Internet à large bande) tendent à rendre possible l'usage d'un outil comme celui issu de nos recherches, nous voyons un problème en ce qui concerne son adoption

par la communauté des utilisateurs du Web. Il nécessite que les documents qu'il affiche soit structurés, ce qui n'est pas vraiment le cas des documents actuellement disponibles sur Internet. Pour donner une structure à tous les documents disponibles sur le Web, il faudrait qu'une institution ait une influence comparable à celle de Microsoft pour demander aux développeurs de pages Web de suivre des directives de développement.

Nous ne pouvons pas nous empêcher de penser à Mark Bernstein qui disait n'avoir jamais rencontré de solution techniques au problème du "*Lost in hyperspace*". Il concluait que des documents compliqués et mal structurés ne seraient jamais simples, quel que soit l'outil utilisé pour les consulter [25]. Ces documents ne seront probablement jamais simples mais peut-être qu'ils seront moins compliqués à consulter. C'est en tout cas dans ce but que nous avons mené nos recherches.

Remerciements

Nous souhaitons sincèrement remercier toutes les personnes qui nous ont permis de réaliser ce mémoire. Nous remercions particulièrement Patrick Roth, notre “gourou”, pour son enthousiasme, sa disponibilité et son soutien, ainsi que Gilles Falquet pour son encadrement et ses judicieux conseils de lecture.

N’oublions pas les testeurs pour le temps qu’ils nous ont consacré et nos familles pour nous avoir supporté durant ces longs mois.

Bibliographie

- [1] Rashmi Sinha, Marti Hearst, Melody Ivory and Maya Draisin, “Content or Graphics ? An Empirical Analysis of Criteria for Award-Winning Websites”, 7th Conference on Human Factors and the Web, June 2001. <http://Web-tango.berkeley.edu/papers/>
- [2] Jakob Nielsen, “The Art of Navigating through Hypertext”, Communications of the ACM, Volume 33, Number 3, March 1990. <http://portal.acm.org/citation.cfm?id=77483>
- [3] Alan Dix, “Design of User Interfaces for the Web”, UIDIS’99 - User Interfaces to Data Intensive Systems, Edinburgh, 1999. <http://www.hiraeth.com/alan/papers/UIDIS99/>
- [4] Jakob Nielsen, “How users read on the Web”, October 1st 1997. <http://www.useit.com/>
- [5] Jakob Nielsen, “Is navigation useful?”, January 9, 2000. <http://www.useit.com/>
- [6] EdubyWeb, “Interface d’un support pédagogique multimédia : les grandes règles à respecter”. <http://www.edubyWeb.com/>
- [7] Jakob Nielsen, “Beyond the Browser”, March 2001. <http://www.eweek.com>
- [8] Jocelyne et Marc Nanard sous la direction de Christophe Kolski, Environnements évolués et évaluation de l’IHM (Chap. 5), “Principes hypermédias et systèmes d’information”, Hermès, Paris, 2001. <http://www.univ-valenciennes.fr/LAMIH/>
- [9] Jakob Nielsen, “Be Succint! (Writing for the Web)”, March 15, 1997. <http://www.useit.com/alertbox/>
- [10] Jakob Nielsen, “Report From a 1994 Web Usability Study”, 1994. <http://www.useit.com/papers/>
- [11] Jakob Nielsen, “The Need for Web Design Standards”, September 13, 2004. <http://www.useit.com/alertbox/>
- [12] Jennifer English, Marti Hearst, Rashmi Sinha, Kirsten Swearington and Ping Yee, “Examining the Usability of Web Site Search”, unpublished, 2001. <http://bailando.sims.berkeley.edu/papers/epicurious-study.pdf>
- [13] Alan Dix, Janet Finlay, Gregory Abowd and Russell Beale, Hypertext, multimedia and the world wide Web (chapter 16),

- “Human-Computer Interaction Second Edition”, Prentice Hall, 1997. <http://www.hcibook.com/hcibook/hci2e.html>
- [14] Shaun Kaasten and Saul Greenberg, “Integrated Back, History and Bookmarks in Web Browsers”, In Extended Abstracts of the ACM Conference of Human Factors in Computing Systems (CHI’01), ACM Press, 2001. <http://grouplab.cpsc.ucalgary.ca/papers/2001.html>
- [15] Gerti Kappel, Werner Retschitzegger, Wieland Schwinger, “Modeling Customizable Web Applications, A requirement’s Perspective” Gerti Kappel 1 Werner Retschitzegger 1 Wieland Schwinger 2 1 Institute of Applied Computer Science, Department of Information Systems (IFS) University of Linz, Altenbergerstraße 69, A-4040 Linz, Austria, {gerti | werner}@ifs.uni-linz.ac.at 2 Software Competence Center Hagenberg (SCCH) wieland.schwinger@scch.at, <http://www.rspa.com/reflib/AnalysisWebApps.html>
- [16] Melody Y. Ivory and Marti A. Hearst, “Improving Web Site Design”, IEEE Internet Computing, April 2002. <http://csdl.computer.org/comp/mags/ic/2002/02/w2toc.htm>
- [17] L. Tauscher and S. Greenberg, “Revisitation patterns in World Wide Web Navigation”. In ACM SIGCHI 97 Proceedings of the Conference on Human Factors in Computing Systems, Atlanta, Georgia, USA, ACM Press, March 1997. <http://www.cpsc.ucalgary.ca/grouplab/papers/1997/97-RevisitationPatterns.CHI/cameraready.pdf>
- [18] Jakob Nielsen, “Saying No : How to Handle Missing Features”, January 23, 2000. <http://www.useit.com/alertbox/>
- [19] Jakob Nielsen, “Make it Usable”, June 2001. <http://www.pcmag.com>
- [20] Wolf-Gideon Bleek, Martti Jeenicke, Ralf Klischewski, “Developing Web-based Applications through e-Prototyping”, 26th Annual International Computer Software and Applications Conference, IEEE Computer Society, August 2002. <http://csdl.computer.org/comp/proceedings/compsac/2002/1727/00/1727toc.htm>
- [21] Reina and J. Torres, “Analysing the navigational aspect”, Technical report by the Institute of Computer Science III, University of Bonn, Germany. February 2002. <http://www.lsi.us.es/~reinaqu/mispapers.html>
- [22] Jacco van Ossenbruggen, Joost Geurts, Frank Cornelissen, Lynda Hardman and Lloyd Rutledge, “Toward Second and Third Generation Web-Based Multimedia”, In Proceedings of the 10th International World Wide Web Conference, Hong Kong, May 2001. <http://www10.org/cdrom/papers/423/index.html>
- [23] Jakob Nielsen, “Response Times : The Three Important Limits”. <http://www.useit.com/papers/>
- [24] Jakob Nielsen, “The Internet Desktop”, March 1996. <http://www.useit.com/alertbox/>
- [25] Mark Bernstein, “Hypertext Gardens”, <http://www.eastgate.com/garden/>

- [26] Greenberg, S. and Cockburn, A. (1999). "Getting Back to Back : Alternate Behaviors for a Web Browser's Back Button". Proceedings of the 5th Annual Human Factors and the Web Conference, Held at NIST, Gaithersburg, Maryland, USA, June 3th.
- [27] Cockburn, A., Greenberg, S., McKenzie, B., Smith, M. and Kaasten, S. (1999) "WebView : A Graphical Aid for Revisiting Web Pages". In 'Proceedings of the OZCHI'99 Australian Conference on Human Computer Interaction', (November 28-30, Wagga Wagga Australia). <http://cite-seer.ist.psu.edu/cockburn99Webview.html>
- [28] EZ Ayers and JT Stasko. "Using graphic history in browsing the world wide Web". In Proceedings of the Fourth International World Wide Web Conference. 11-14 December, Boston, 1995. Also available from : <http://www.w3.org/pub/Conferences/WWW4/Papers2/270/>.
- [29] Jakob Nielsen, "Top Ten Mistakes in Web Design". <http://www.useit.com/alertbox/>
- [30] Jakob Nielsen, "Usability 101 : Introduction to Usability". <http://www.useit.com/alertbox/>
- [31] m.c. schraefel, Leslie Carr, David De Roure and Wendy Hall, "You've Got Hypertexte". <http://jodi.ecs.soton.ac.uk/Articles/v05/i01/schraefel/>
- [32] Le Petit Larousse illustré, Edition Larousse, Paris, 1993
- [33] Wikipedia, http://en.wikipedia.org/wiki/Web_browser
- [34] W3 Schools, http://www.w3schools.com/browsers/browsers_stats.asp
- [35] <http://www.understandingusa.com/understanding.html>
- [36] <http://www.useit.com/alertbox/newfeatures.html>
- [37] <http://fr.wikipedia.org/wiki/W3C>
- [38] http://fr.wikipedia.org/wiki/Navigateur_Web
- [39] <http://psychology.wichita.edu/surl/usabilitynews/61/breadcrumb.htm>
- [40] <http://www.panda.org/>
- [41] Keith Andrews, "Human-Computer Interaction", 2004, <http://courses.iicm.edu/hci/>
- [42] Jean-François Richard, 1990 : *Traité de psychologie cognitive 2*, Bordas, Paris
- [43] <http://macromedia.com/>

Annexes

Prototype papier

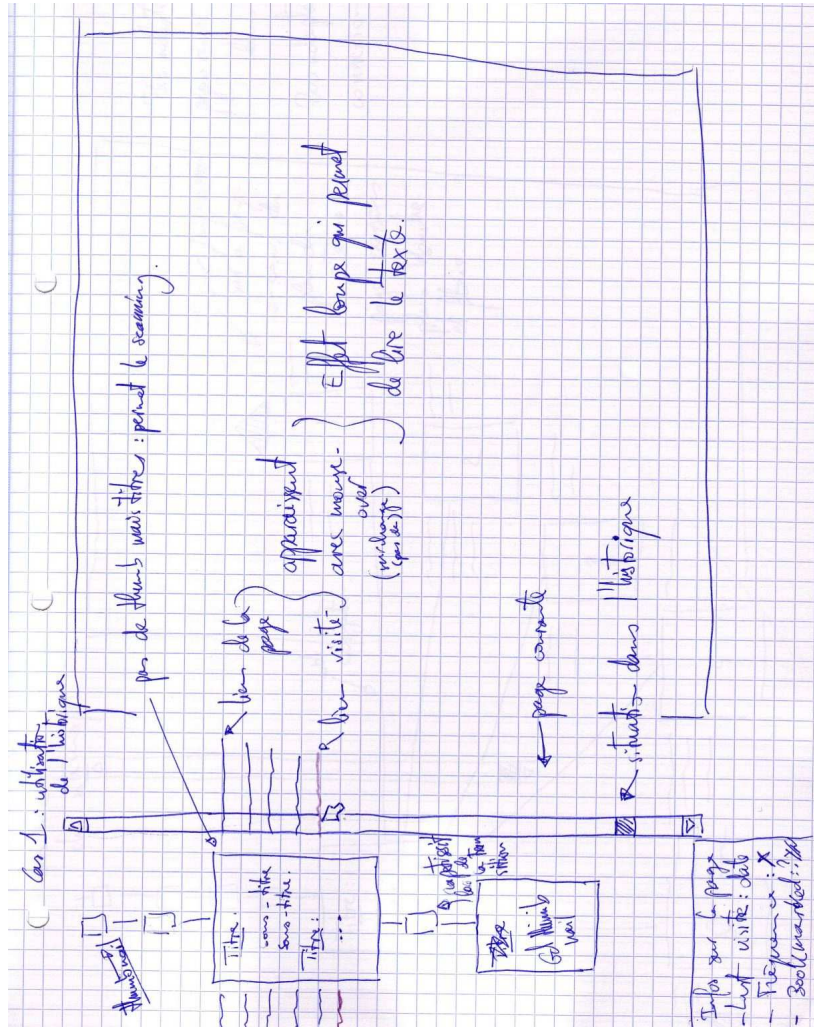


FIG. 6.1 – Historique visuel

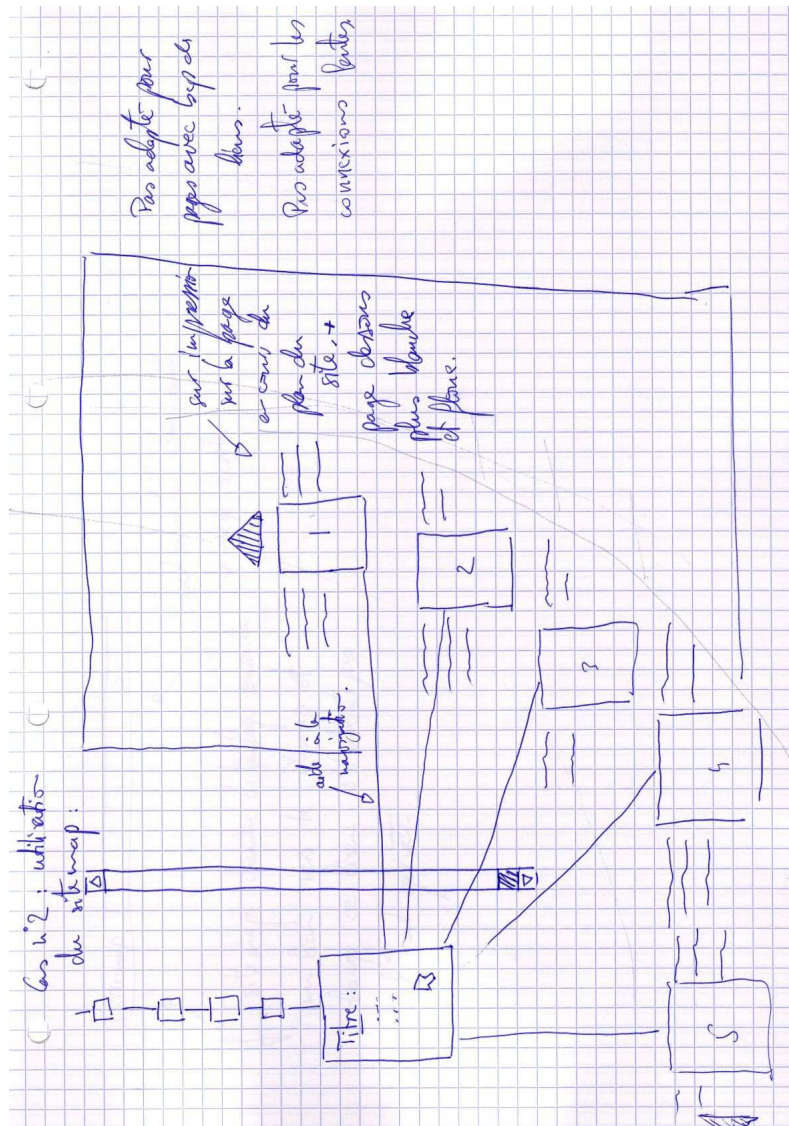


FIG. 6.2 – Arbre de navigation

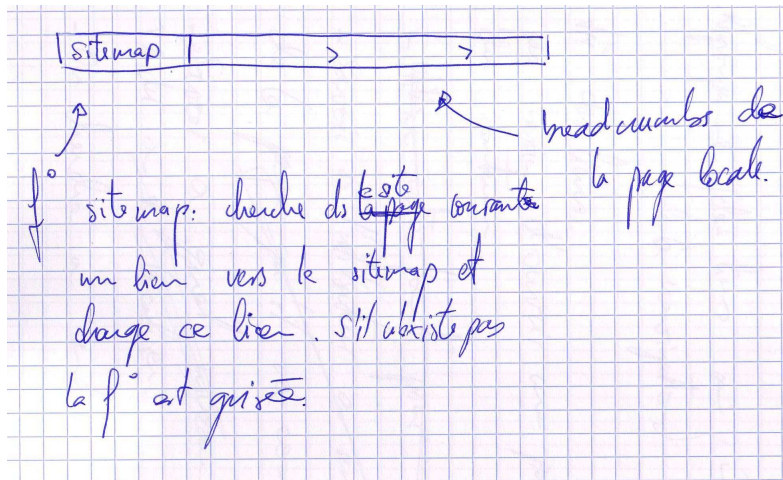


FIG. 6.3 – Breadcrumbs / Sitemap

Cas d'utilisation

Les différents cas d'utilisation que nous pouvons rencontrer dans notre prototype sont les suivants :

Ouverture d'une page

- Création d'un *thumbnail*
- Création d'une mini page
- Ajout du *thumbnail* dans l'historique
- Affichage de la page dans le navigateur

Le curseur se trouve sur le dernier *thumbnail* ajouté à l'historique

- Ouverture de l'arbre de navigation
- La page courante devient transparente
- Création des *thumbnails* des liens disponibles depuis la page courante
- Création des mini pages à l'aide de la fonction *Scanning*
- Affichage de l'arbre de navigation avec les *thumbnails*
- Affichage des flèches de déplacement dans l'arbre de navigation

Le curseur est sur un *thumbnail* (soit dans l'arbre de navigation soit dans l'historique mais pas sur le *thumbnail* de la page courante)

- Affichage du *Scanning* du *thumbnail* survolé par le curseur (transformation du *thumbnail* en mini page)
- Affichage des liens disponibles sur la page représentée par le *thumbnail*

Clic sur une des flèches de déplacement de l'arbre de navigation

- Déplacement des *thumbnails* affichés

Clic sur une des flèches de déplacement d'une mini page

- Déplacement des liens affichés

Le curseur sort de la zone d'action de l'arbre de navigation

- L'arbre de navigation disparaît
- La page courante perd sa transparence

Questionnaire post-expérimental

Evaluation des fonctions

Données personnelles:

Date:

Nom:

Prénom:

Age:

Sexe:

Niveau d'étude: Primaire / Secondaire / Université

Branche d'activité / d'étude:

A quelle fréquence utilisez vous un ordinateur?

Jamais 1 x /mois 1 x /semaine 3 x /semaine Tous les jours

A quelle fréquence utilisez-vous le web?

Jamais 1 x /mois 1 x /semaine 3 x /semaine Tous les jours

Combien de temps y passez-vous à surfer le web? _____

Lire attentivement toute la question avant de commencer à répondre!

1) Depuis la page d'accueil du site TECFA (Technologies de formation et Apprentissage: <http://tecfa.unige.ch>), pouvez-vous nous fournir la structure informationnelle (titres, sous-titres, etc.) de la page « Brown Bag seminar »?

Si oui: quel est-elle? _____

- 2) Depuis la page d'accueil du site <http://tecfa.unige.ch>, veuillez:
 1. Trouver la page qui aborde le projet « Hypermedia et Learning »
 2. Dans le paragraphe « Hypermedia & Learning », charger le lien « See overview and research report ».

FIG. 6.6 – Page 3 - Tâche 2

- 3) Pouvez vous ouvrir la page « Projects », puis « Info. & Comm. Systems », puis le lien « research themes ». Recherchez le paragraphe « Past projects » et dans le sous-paragraphe « Artificial Intelligence in Education » ouvrir le lien « A brief introduction to the theme ». Sous la rubrique « Research », cliquez sur le lien « Past ».

Pouvez-vous recharger la page « Research at Tecfa » que vous avez rencontrée lors de votre navigation?

4) Pouvez-vous nous indiquer à quel niveau hiérarchique appartient la page courante dans l'arborescence du site?

Non Oui _____

Si oui, qu'avez vous utilisé pour vous situer? Autre technique?

FIG. 6.8 – Page 5 - Tâche 4

5) Pouvez-vous nous décrire l'ensemble de votre chemin de navigation?

FIG. 6.9 – Page 6 - Tâche 5

6) Pouvez vous nous fournir d'une manière détaillée la structure hiérarchique du site?

FIG. 6.10 – Page 7 - Tâche 6

Questionnaire post-expérimental

- 1) A votre avis, quels sont les aspects négatifs et positifs de la fonction « Arbre de navigation »?
- 2) La transformation des *thumbnails* en mini-page (utilisation de la fonction *Scanning*), vous ont-ils apporté des informations déterminantes pour votre navigation?
- 3) L'aspect graphique de l'historique vous a-t-il facilité l'accès aux pages visitées?
- 4) La fonction « *Sitemap* » vous a-t-elle été utile? Cherchez-vous habituellement à vous repérer grâce à la fonctionnalité *Sitemap*?
- 5) Ces fonctions sont-elles intuitives?
- 6) Utiliseriez-vous ces fonctions dans votre vie pratique?
- 7) Que doit-on leur ajouter pour pouvoir se familiariser plus facilement avec elles? (p.ex. FAQs / help / tutorial)
- 8) Avez-vous des suggestions à nous proposer?

FIG. 6.11 – Page 8 - Questions destinées aux testeurs du prototype