

Séance 6

Prototypage

INF 753

Prototypage

– Prototypage

- Prototypage *lo-fi*
- Prototypage *hi-fi*

– Évaluation

- Tests d'utilisabilité

Pourquoi prototyper?

- Communiquer des idées et obtenir rapidement une rétroaction sur notre conception
- Expérimenter des conceptions alternatives
- Régler les problèmes avant le codage
- Centrer la conception sur l'utilisateur

La fidélité dans le prototypage

- La fidélité réfère au niveau de détail
- Haute fidélité (*hi-fi*)
 - Les prototypes paraissent comme le produit final
- Basse fidélité (*lo-fi*)
 - Ébauches papier (croquis)
 - Détails absents (liés au style, à l'esthétique)



Pourquoi utiliser les prototypes *lo-fi*?

- Les méthodes traditionnelles de maquettage *hi-fi* prennent beaucoup de temps
 - Surtout si elles s'insèrent dans un cycle centré utilisateur : *analyse / conception / évaluation* → itération
- les prototypes *lo-fi* permettent :
 - de produire plusieurs itérations rapidement
 - de simuler un déroulement
 - aux non-programmeurs de participer aux efforts de conception

Si le prototype est trop détaillé (*hi-fi*)...

- La perception du testeur/examineur sera biaisée :
 - Les commentaires se feront sur les choix de couleurs, de polices, l'alignement, etc.
 - Il perdra de vue l'ensemble (*the big picture*)

The better it looks, the more narrow the feedback



Looks Done

Mocked up in Photoshop, a multimedia program (Director, Flash, etc.), or a GUI builder (NetBeans, Visual Studio, etc.)

"Can you change the font on that 'T'?"

Not sure I like the bevel line weight..."

Feedback: detailed tweaks to specific features. Very focused and incremental.



Visio, Powerpoint, etc.

Illustrated using a professional drawing or presentation tool.

"I don't like the two-column layout for tools. Can we have them go across the top?"

Feedback: tweaks to the 'screen' or page as a whole. Incremental improvements.



Rough Sketch

Scanned from a hand-drawing, made with a drawing app and a tablet, or using the Napkin Look and Feel skin.

"Maybe the tools should be context-specific...
Let's kill the toolbar and bring up only the tools that make sense at that moment..."

Feedback: higher-level features are questioned, bigger changes possible.



Storyboard or Use Case

The "story" of how the user might need or want to interact with the interface (app, book, product, etc.)

"We should NOT try to put a drawing feature in here... it's featuritis without a key benefit to most users."

Feedback: big-picture ideas, possibility for revolutionary changes.

Construction du modèle

Fixer une date limite s'inscrivant dans l'échéancier

– pas trop long!

- Dessiner un cadre de fenêtre sur un papier large
- Mettre les différentes régions d'écran sur des cartes
 - tout ce qui se déplace, change, apparaît/disparaît
- Préparer la réponse à chaque action de l'utilisateur
 - Ex. : actions associées aux menus
- Faire plusieurs versions à l'aide de photocopies

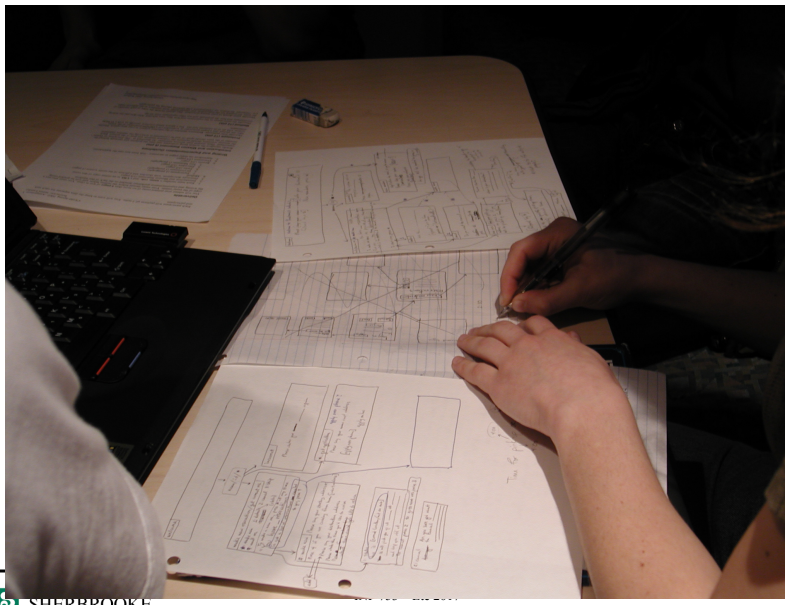
Construction du modèle

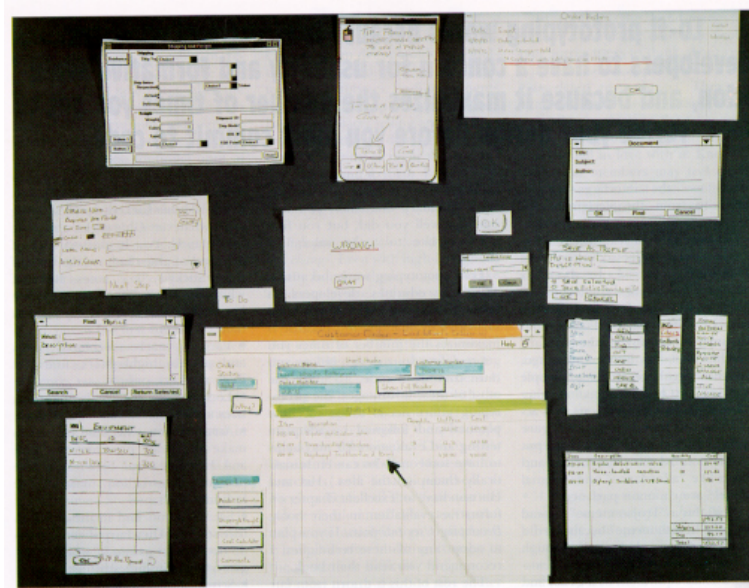


Construction du modèle



Construction du modèle





Avantages des prototypes *lo-fi*

- Interfaces informelles offrant les possibilités :
 - De *brainstorming* (remue-méninges)
 - mettre les conceptions sous une forme tangible
 - considérer rapidement différentes idées (itération rapide – coût de modification faible)
 - donner plus de place à la créativité à cette étape initiale
 - De conception incomplète
 - pas besoin de couvrir tous les cas / exemples importants
 - ne nécessite pas la spécification des détails (polices)
 - De présenter plusieurs alternatives au client
 - Permet aux non-programmeurs de participer à la conception

Limites des prototypes *lo-fi*

- Prototype papier
 - n'évolue pas facilement
 - difficile de conserver la trace historique des itérations
 - nécessite une transformation manuelle dans un format électronique
 - beaucoup de papier à gérer...
 - ne permet pas les interactions de l'utilisateur final
 - ne permet pas de mesurer des Entrées/Sorties réalistes (souris/clavier)
 - changement de l'apparence des boutons, liens visités
 - objets sélectionnables et non sélectionnables

Limites des prototypes *lo-fi* (suite)

- Défi de bien dessiner (reconnaissance des éléments)
- Limites à la critique (sur le plan esthétique)
- Ne permet pas d'évaluer le temps précis de réalisation d'une tâche (plus lent à utiliser qu'un prototype électronique)

Alternative : le prototype *lo-fi* électronique

- Avantages : Permet de visualiser la séquence d'interfaces à différents niveaux (plan du site, arborescence ou mise en page d'une page spécifique)
- Inconvénients : Nécessite malgré tout la spécification de beaucoup de détails
 - on doit préciser une instance particulière d'une idée générale
 - Ex. : choisir les bons *widgets*, les bonnes polices, les bons alignements, les bonnes couleurs...
 - les concepteurs tendent alors à focaliser sur des détails non importants à cette étape
 - Prennent plus de temps à développer (dans certains cas)

Prototypes *hi-fi*

- Outils électroniques (*Drag & drop prototyping*)
- Permet de tester et d'observer certaines idées précises
 - Ex.: maquette promotionnelle pour vendre une idée
- Lorsque les prototypes papiers ne vont pas assez loin
 - permet d'illustrer une dynamique et des interactions
- Permet d'impliquer une variété de spécialistes

Prototypes *hi-fi*

- Évite de construire l'application directement
 - des changements dans l'interface peuvent créer de grands changements au niveau du code
 - développement applicatif prend beaucoup de temps
- Défi : trouver l'équilibre entre la nature et la quantité des éléments à inclure
 - Autrement dit : quel niveau de fidélité obtenir par rapport au produit final?
 - ➔ Privilégier la largeur (inclure toutes les sections, fonctionnalités, *features*...) plutôt que la profondeur (contenus et textes réels non essentiels à cette étape)

Outils de prototypage

- Classiques : *Powerpoint*, *Visio*, *Fireworks*
- Axure RP axure.com
- Sketch sketchapp.com
- Zeplin <https://zeplin.io/>
- InVision invisionapp.com
- LucidChart lucidchart.com
- Adobe XD (*Experience Design - Beta*)
Anciennement *Adobe Edge Reflow*, *Project Comet*, *Adobe Muse*...
- Omnigraffle (Mac) omnigroup.com/omnigraffle/
- Pencil Project pencil.evolus.vn
- iPlotz iplotz.com
- UXPin uxpin.com

AXURE RP : Prototypage rapide évolutif

- 1) Le concepteur « maquette » rapidement ses idées en positionnant des objets et en créant des interactions
 - Textes et images
 - *widgets*
 - Interactions entre les pages, *widgets*, etc
- 2) Le concepteur ou l'utilisateur final teste l'interface
- 3) La maquette peut être transformée en une interface fonctionnelle (format Web ou autre)

www.axure.com

Tests et évaluation des prototypes

- Sélectionner les utilisateurs :
 - identifier le profil des utilisateurs potentiels (expérience avec la tâche, motivation à utiliser ce type d'interface, etc.)
 - recruter des testeurs (éviter famille ou amis)
- Préparer les scénarios :
 - Identification des fonctionnalités significatives à tester (scénarios-types, tâches, sections...)
 - Objectifs mesurables (performance [temps, clics], précision [erreurs], émotions [satisfaction, confiance]...)

Rappel

- Le prototype à tester devrait être le reflet de l'étape du développement à laquelle on se trouve réellement :
 - Phase d'idéation : Mesurer « the big picture »
= *lo-fi*
 - Phases de design et de développement (validation de détails d'interface)
= *hi-fi*

Tests et évaluation des prototypes

- Passation des tests
 - filmés ou non, mouchard électronique, enregistreur d'écran, etc.
<http://www.techsmith.fr/camtasia.asp>
- Passation de questionnaires
- Analyse des observations
- Rédaction rapport final (problèmes et solutions)

Protocole de conduite des tests

- Rôles
 - Mettre les utilisateurs à l'aise, dans une ambiance informelle. C'est l'interface qui est testée, pas l'utilisateur qui est évalué ou qui doit performer.
 - « facilitateur » - le seul membre de l'équipe qui parle
 - donne des instructions et encourage les idées, les opinions
 - « ordinateur » - connaît et contrôle la logique de l'application
 - simule toujours la réponse sans explication
 - « observateur » - prend des notes et des recommandations
- Une session typique dure 1 heure
 - la préparation, le test, compte rendu

Protocole de conduite des tests (suite)

1. Accueillir
 - faire remplir les formulaires, assurer la confidentialité, etc.
2. Tester
 - le facilitateur décrit les tâches que l'utilisateur va entreprendre
 - doivent être claires et détaillées
 - « À quoi pensez-vous en ce moment? », « Penser à voix haute »
 - observer -> pas de « ha ha », pas de rire, regarder bouche bée, etc.
3. Récapituler
 - poser des questions, recueillir les impressions, remercier

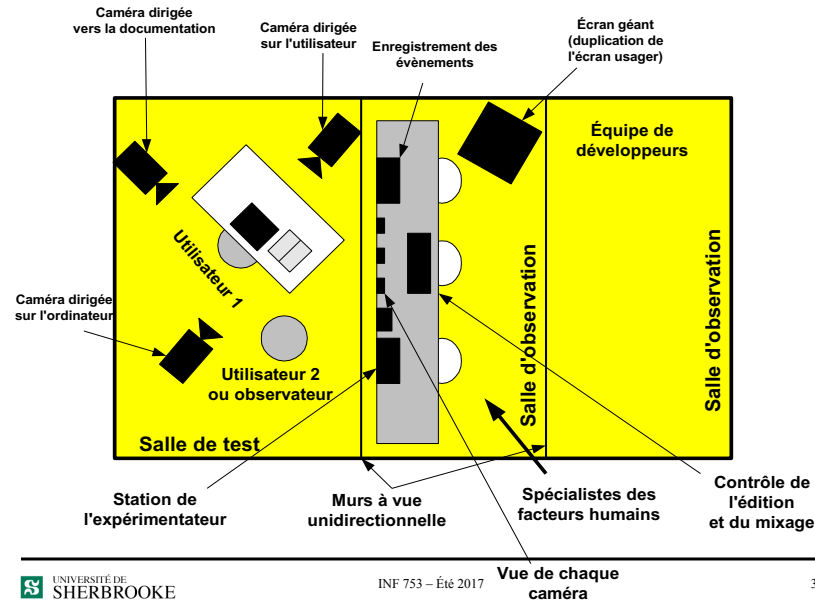
Évaluation des résultats

- Trier les observations et leur associer des priorités
 - Qu'est-ce qui était important?
 - Plusieurs problèmes dans un même domaine?
 - Important : Les suggestions des testeurs doivent être considérées avec recul par les évaluateurs
- Créer un rapport écrit sur les conclusions
 - planifier une réunion pour l'application des changements sur la conception
- Procéder aux changements et itérer

Les tests d'utilisabilité

- Test d'utilisabilité en laboratoire
 - Un lab classique d'utilisabilité comporte deux espaces séparés par un miroir. Un espace est réservé aux utilisateurs en situation d'exécution des tâches et un autre aux observateurs et évaluateurs (concepteurs, gestionnaires ou clients)

Les tests d'utilisabilité



31

Oculométrie (eyetracking)

- Technique de suivi du regard et des actions dans l'interface
 - <http://www.imarklab.com/2015/03/infographie-oculometrie/>
 - <http://www.lapresse.ca/le-soleil/affaires/techno/201405/29/01-4771088-ubisoft-a-la-recherche-du-facteur-fun-.php>
- Constats généraux
 - Évolution du modèle de lecture (en F vs en Z...)
 - Listes à puces
 - Liens de sortie
 - Aveuglement aux bannières

Les tests d'utilisabilité

- Avantages
 - Environ 80 % des problèmes d'utilisabilité d'une version de la maquette ou du produit pourraient être trouvés avec 4-5 sujets
 - Préférez 3 tests (de 3 itérations) avec 5 sujets, qu'un seul test avec 15 sujets
- Limites
 - Se concentre uniquement sur la première utilisation de l'interface vu les temps de performance alloués aux participants (ne tient pas compte de l'apprentissage)
 - Utilisation limitée des fonctionnalités de l'interface

Tests A/B

- Tester simultanément deux versions d'une interface Web et observer les taux de conversion
- <http://www.sparkpage.com/1-billion-worth-of-ab-testing/>
- *Google Analytics Content Experiments*
<https://developers.google.com/analytics/solutions/experiments?hl=fr>
(anciennement *Google Website Optimizer*)
- <http://www.smashingmagazine.com/2010/06/24/the-ultimate-guide-to-a-b-testing/>
- whichtestwon.com
- <http://www.clickz.com/clickz/column/2346615/6-tools-for-cost-effective-website-usability-testing>

Monitoring

- Monitoring (mouchard électronique) :

- recueil automatique des données

- des stratégies utilisées / du séquençement réel des tâches
 - des performances obtenues / du taux de conversion
 - identification des erreurs d'utilisation, etc.

- www.google.ca/analytics/

- www.crazyegg.com

- www.clicktale.com

- <https://lookback.io/home>