

# IFT 2905: INTERFACES PERSONNE-MACHINE

Design Centré sur l'Utilisateur

<http://tiny.cc/ift2905>



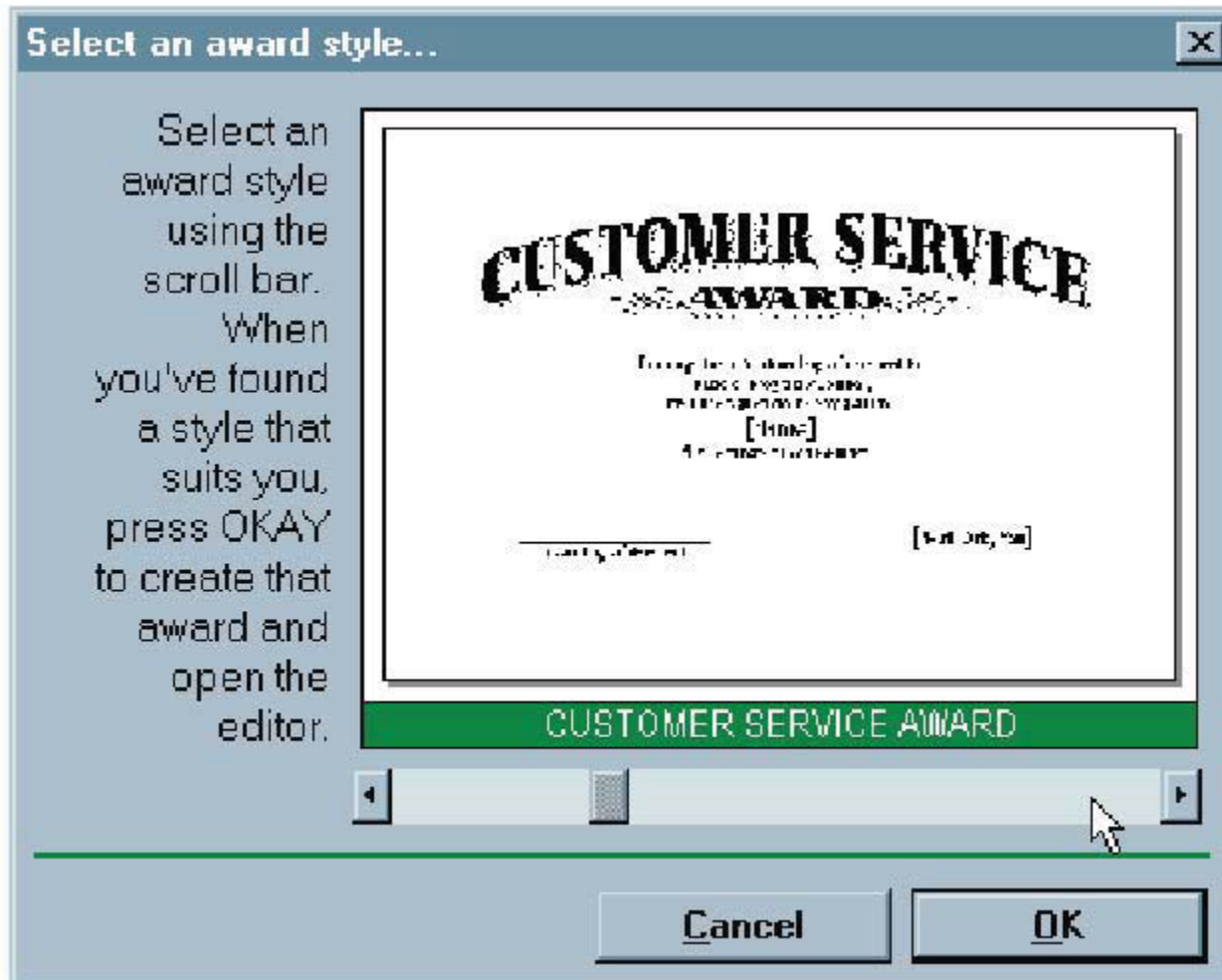
Image © Katerina Kamprani, [www.theuncomfortable.com](http://www.theuncomfortable.com)

## Mikhail Bessmeltsev

Certaines diapos de Bernhard Thomaszewski

# POURQUOI EST-CE TERRIBLE?

« Sélectionnez un style de certificat »



# UTILISABILITÉ

Rappel: les bonnes interfaces utilisateurs ont une bonne utilisabilité

Effacité  
Efficience  
Satisfaction  
Simplicité d'apprentissage  
Mémorabilité  
Visibilité (*feedback*)  
Erreur (*tolérance et prévention*)

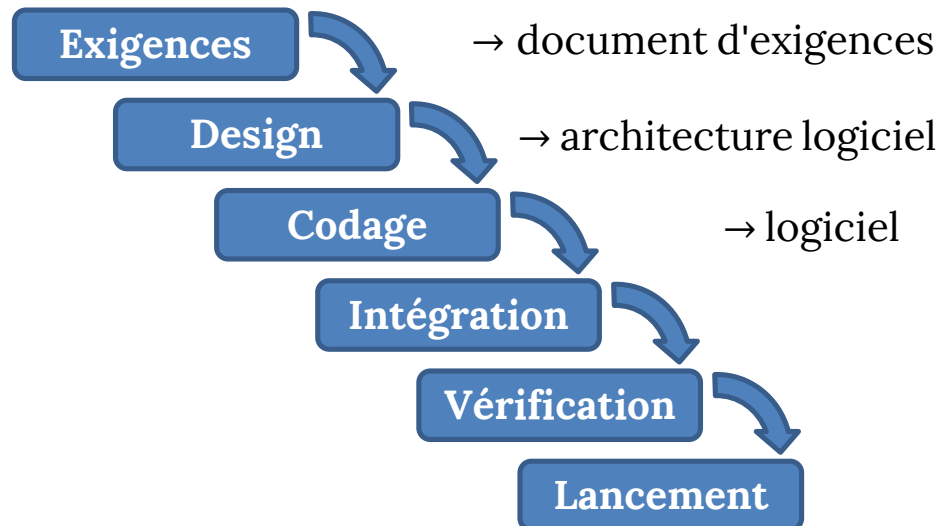
} Difficile à prédire  
→ *faut tester!*

*Quel type de processus de développement est le meilleur pour l'utilisabilité?*

# PROCESSUS TRADITIONNEL DE GÉNIE LOGICIEL

## Le modèle cascade

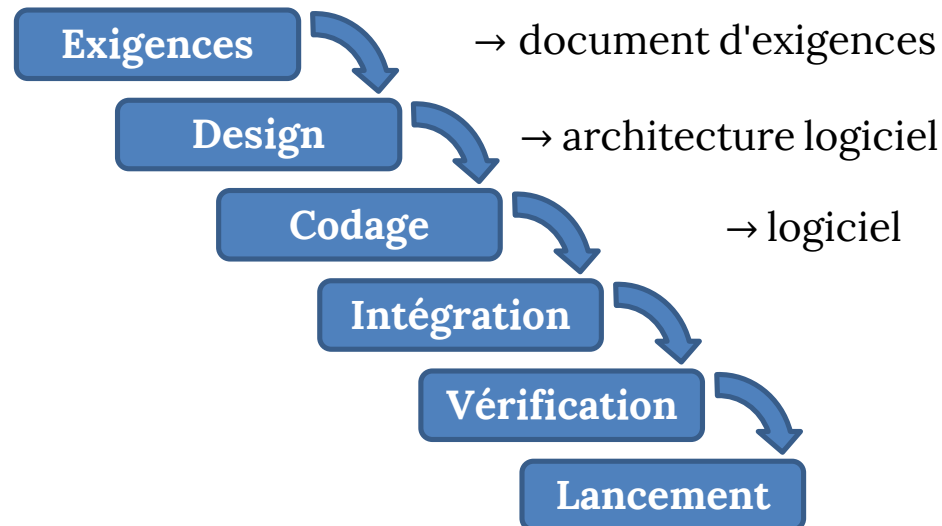
- processus de conception est une séquence d'étapes
- chaque étape conduit à un produit concret
- chaque étape a sa propre validation
  - exigences (fonctionnalité)
  - tests unitaires (vérification de code)
- principe: d'abord penser, puis coder (*think first, code second*)



# FEEDBACK DANS LE MODÈLE CASCADE

## Le modèle cascade

- Risque: des problèmes ne sont pas trouvés avant les étapes ultérieures
  - p.e., l'implémentation peut révéler des problèmes non anticipés
- Revenir en arrière peut être très coûteux



# LE MODÈLE CASCADE

*Le modèle cascade est mauvais pour le design d'IU*

- La conception de l'interface utilisateur est intrinsèquement risquée
  - Vous n'êtes pas l'utilisateur
  - L'utilisateur a toujours raison
  - Les utilisateurs ne sont pas des concepteurs
  - Nous risquons de nous tromper
- L'utilisateur ne participe pas à la validation avant les tests d'acceptation
  - On trouve des problèmes qu'à la fin du processus
- Les défauts de l'interface utilisateur (*fin de processus*) entraînent souvent des changements dans les exigences et la conception (*début de processus*)
  - Corrections souvent dramatiques
  - Faut jeter du code soigneusement écrit et testé

# DESIGN ITÉRATIF

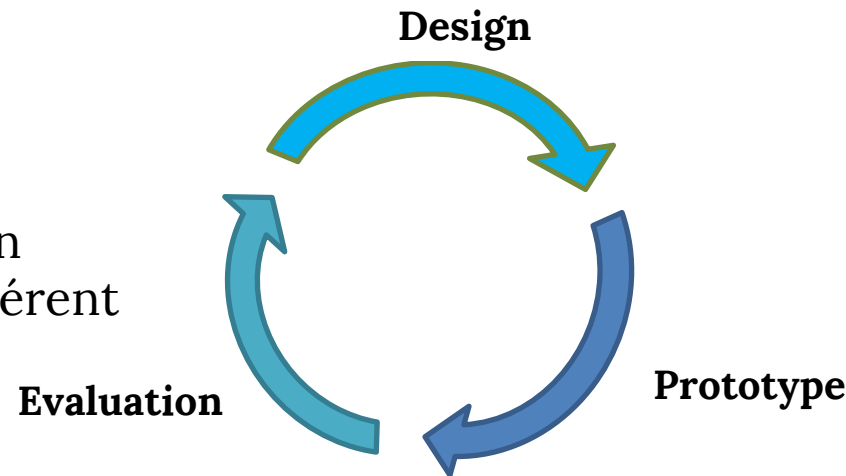
*L'ingénierie de l'utilisabilité est un processus itératif.*

*On y arrivera pas au premier essaie.*

## Design itératif

- au lieu d'une longue séquence, faire beaucoup de petites itérations de conception, prototypage et évaluation
- offre un moyen de gérer le risque inhérent dans la conception de l'IU

## Comment ça marche?



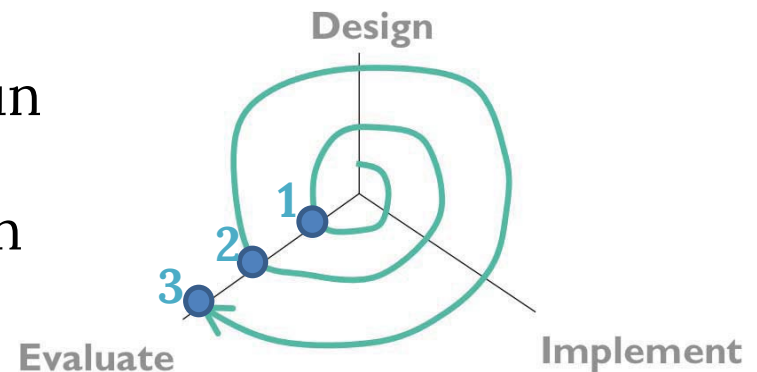
# LE MODÈLE EN SPIRALE

*S'il y a beaucoup d'itérations, comment gérer les coûts?*

Le modèle en spirale prévoit plusieurs itérations dans le processus de conception

- Principe
  - réduire autant que possible les coûts des itérations initiales
  - commencer le processus avec un prototype en papier
  - lorsque convergé, procéder à un prototype programmé

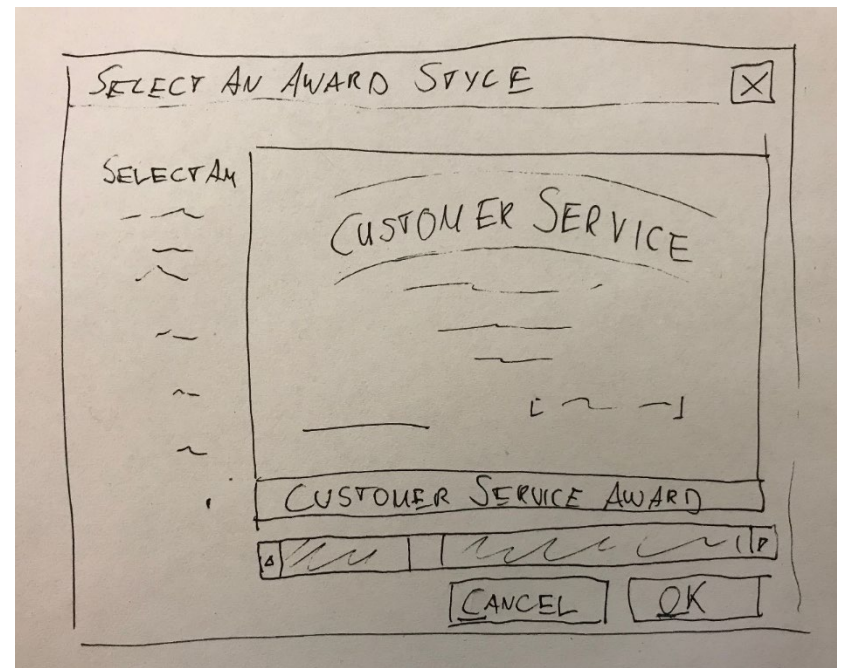
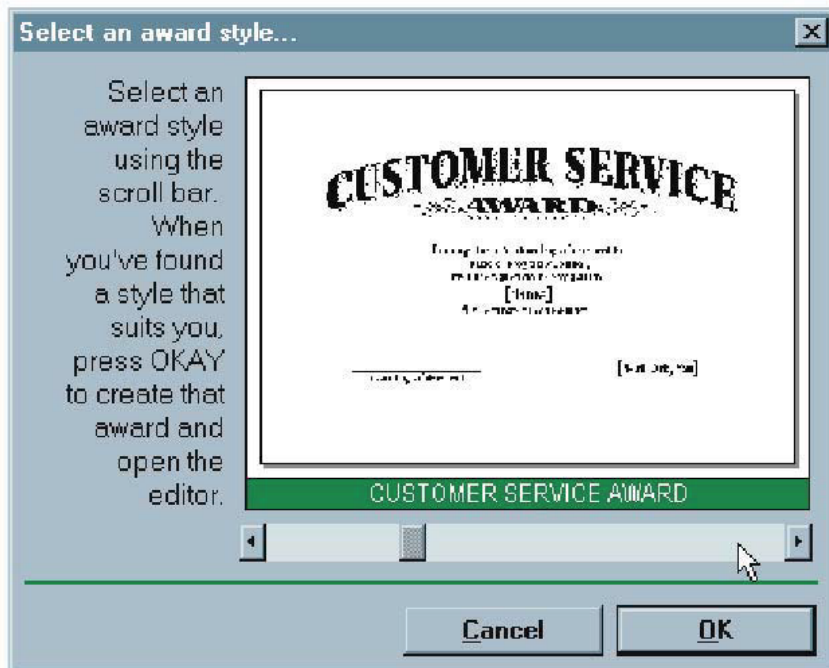
**Dimension radiale:** coût, fidélité, précision





# DÉTECTER DES PROBLÈMES D'UTILISABILITÉ

Des prototypes simples peuvent détecter les problèmes d'utilisabilité



# DESIGN ITÉRATIVE

- Idée centrale du modèle en spirale:
  - le risque est le plus élevé dans les premières étapes
  - rendre les premières étapes les moins chers
- Les itérations ultérieures utilisent des implémentations plus riches après que le risque de l'interface utilisateur a été atténué
- Plus d'itérations conduit généralement à une meilleure interface
  - On s'arrête quand il n'y a plus de temps/argent
- Seules les itérations matures sont présenter au monde (aux clients)

# DEUX APPROCHES

Design centré  
sur la technologie

Design centré  
sur l'utilisateur

# DESIGN CENTRÉ SUR LA TECHNOLOGIE

- Naturel
- Construire ce qu'on peut
- Les décisions de design sont guidées pas la technologie

C'est l'attitude la plus courante dans le monde, parce que:

- La technologie est amusante
- Faire de nouvelles choses est intéressant
- Il semble que cela fonctionnerait bien
  - Les gens l'achètent aussi

# DESIGN CENTRÉ SUR LA TECHNOLOGIE

- Peut créer les choses plus innovatives
- N'est pas nécessairement mauvais!

Mais il peut oublier les problèmes de vraies personnes

« Segway prévoit de mettre fin à la production de son transporteur personnel »

## *Segway to End Production of Its Original Personal Transporter*

Segway Inc. said it would stop making the two-wheeled Segway PT, once hailed as a revolutionary device that would change the way people get around.



Initially called the Segway Human Transporter, the vehicle was the subject of hype and fevered speculation before it was unveiled in 2001. Andrew Harnik/Associated Press

<https://www.nytimes.com/2020/06/24/business/segway-pt-discontinued.html>

# UN SPECTRE D'ATTITUDES ENVERS LE DESIGN

technologie

## **Design centré sur la technologie**

- Le progrès est fait par la technologie qui avance
- Le but est inventer les nouveaux systèmes

## **Design centré sur un designer**

- Le progrès est fait pas l'intuition de designer
- Imagine ce que l'utilisateur va faire

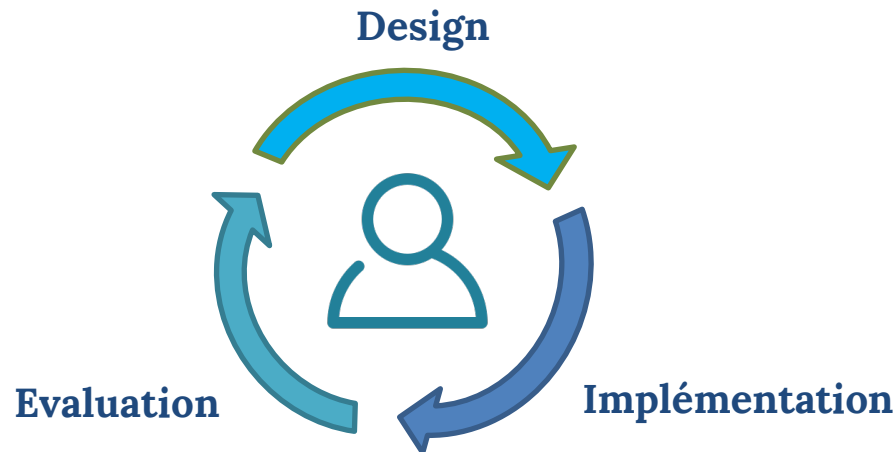
## **Design centré sur l'utilisateur**

- Les utilisateurs font partie de processus de design
- Les études empiriques commencent tôt dans le processus

utilisateur

# DESIGN CENTRÉE SUR L'UTILISATEUR

Le *design centré sur l'utilisateur* (DCU) vise à optimiser le produit en fonction de la façon dont les utilisateurs peuvent, veulent ou doivent l'utiliser le produit, plutôt que de forcer les utilisateurs à modifier leur comportement pour s'adapter au produit.





# QUATRE ÉTAPES

**Pré-design:** comprendre le problème

**Premier design:** explore l'espace de design

**Mi-design:** développer l'approche choisie

**Design avancé:** intégrer et commencer à déployer

# QUATRE ÉTAPES

**Pré-design:** comprendre le problème

**Premier design:** explore l'espace de design

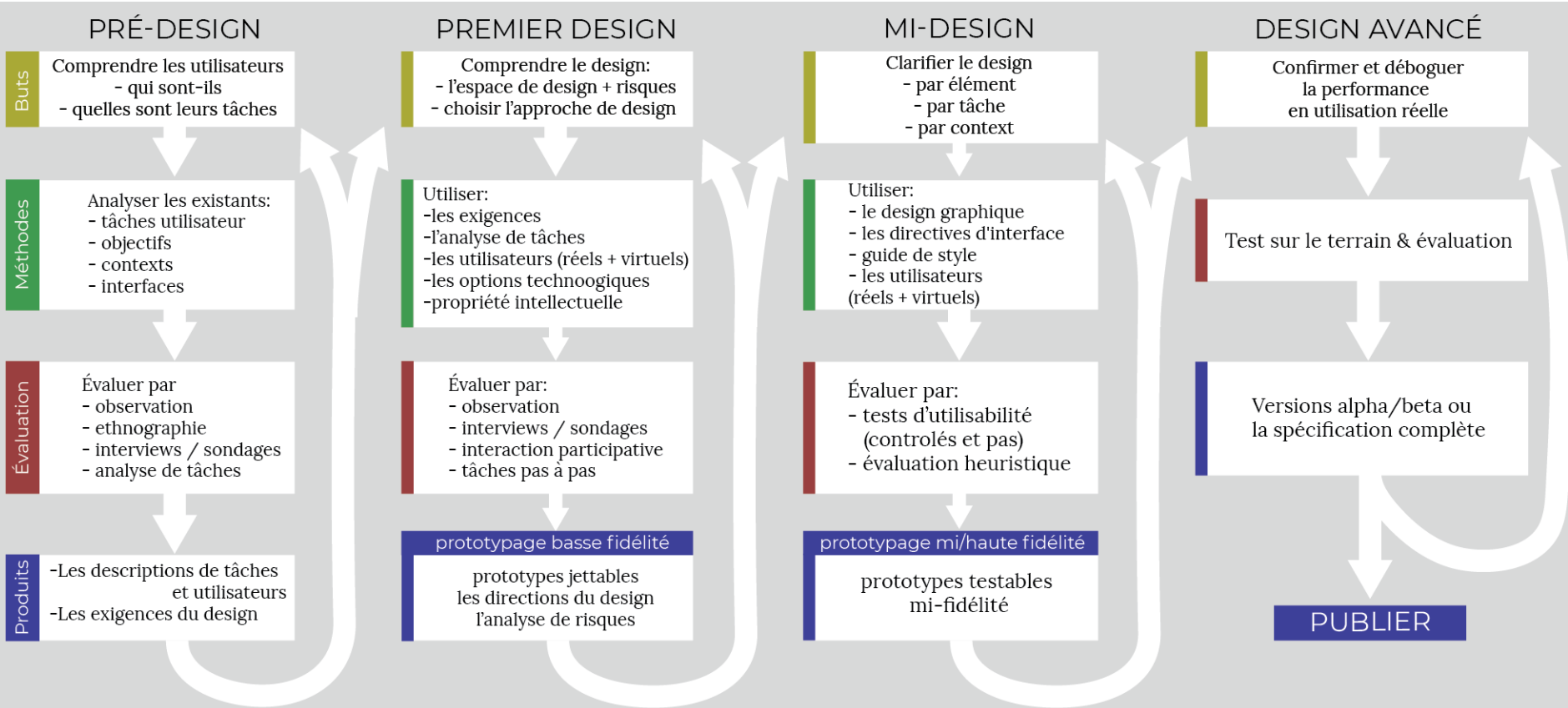
**Mi-design:** développer l'approche choisie

**Design avancé:** intégrer et commencer à déployer

À chaque étape, **ÉVALUER**



# LE PROCESSUS



# PRÉ-DESIGN

Buts

Comprendre les utilisateurs  
- qui sont-ils  
- quelles sont leurs tâches

## Comprendre le problème

- Les utilisateurs ont-ils vraiment le problème que vous pensez qu'ils rencontrent?
- Est-ce que c'est important pour eux?
- Qui sont les utilisateurs? Quelle est leur tâche?
  - à quel point sont-ils variés?
  - expertise, capacités, priorités, besoins particuliers, contraintes
  - Quels non-utilisateurs sont impliqués dans le problème et sa solution potentielle?
  - Qu'est-ce qu'ils essaient vraiment de faire? Pourquoi ça ne marche pas?
- Quelles propriétés une solution doit-elle avoir?

# PRÉ-DESIGN

Buts

- Comprendre les utilisateurs
  - qui sont-ils
  - quelles sont leurs tâches

## Comprendre le problème

- Les utilisateurs ont-ils vraiment le problème que vous pensez qu'ils rencontrent?
- Est-ce que c'est important pour eux?

• Qui sont-ils?

- à quel point ils sont experts?
- Quels sont leurs besoins?
- Quelles solutions ont-ils déjà essayées?
- Qu'est-ce qui ne marche pas?

• Quelles sont leurs contraintes?



Quelles tâches?

Quelles contraintes  
le problème et sa

Qu'est-ce qui ne marche pas?

Quelles solutions ont-ils déjà essayées?  
Qu'est-ce qui ne marche pas?  
Quelles sont leurs contraintes?  
Qu'est-ce qui ne marche pas?  
Quelles solutions ont-ils déjà essayées?  
Qu'est-ce qui ne marche pas?

# DESIGN CENTRÉE SUR L'UTILISATEUR

Le DCU est une forme particulière de design itératif

- Cycle: design, implémentation (prototype), évaluation
- Priorité aux utilisateurs et aux besoins dès le début
  - analyse de l'utilisateur: qui sont les utilisateurs?
  - analyse des tâches: que doivent-ils faire?
- Impliquer des utilisateurs en tant qu'évaluateurs et consultants
- Évaluation continue
  - les utilisateurs sont impliqués dans chaque itération
  - chaque prototype est évalué

# ITÉRATION: LA CARACTÉRISTIQUE CLÉ

Pourquoi devons-nous tant itérer dans le processus de design?

Des tâches, des capacités, des besoins sont complexes, puis il est difficile de prévoir ou modéliser

- Les contextes d'utilisation
- Comment les gens veulent faire leur tâche
- Comment ils vont faire les erreurs



# DCU: DESIGN

## Les différentes composants de la partie *design*

- Analyse des utilisateurs, des tâches, et du domaine
- Principes de conception (pour l'utilisabilité)
- Prévention des erreurs et gestion des erreurs
- Contrôle par l'utilisateur et liberté

# DCU: **IMPLÉMENTATION**

- Prototypage
  - Implémentation *jetables*
  - Basse fidélité: papier, maquette (*Wizard of Oz*)
  - Moyenne fidélité: HTML, Flash
- Techniques d'implémentation de l'interface graphique
  - Framework et plateformes (Android, Qt, ...)
  - Génie logiciel (design pattern)
  - Contraintes et mise en page
  - Modèles de sortie et d'entrée

# PROTOTYPES EN PAPIER



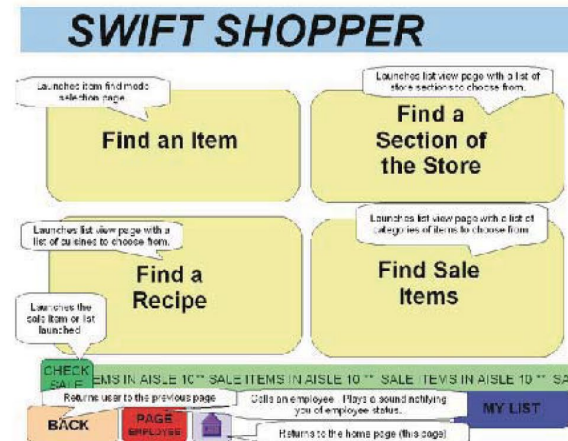
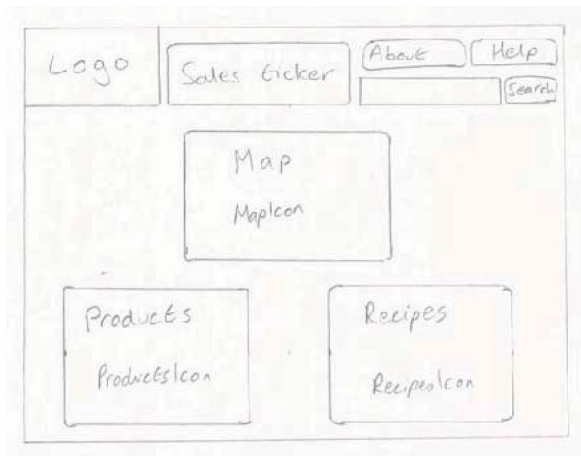
Nina Wei

<https://zweiprototype.wordpress.com/category/paper-prototype/>

# PROTOTYPES EN PAPIER

## Pourquoi papier?

- c'est très rapide à faire
- c'est simple: on ne distrait pas de l'essentiel
- c'est abstrait: on n'impose aucun style visuel



© MIT Students. All rights reserved.

# PROTOTYPES WIZARD OF OZ



# DCU: ÉVALUATION

*(On en parlera en détail dans un cours suivant)*

- L'évaluation met les prototypes à l'épreuve
- Évaluation d'experts
  - Heuristiques et procédures pas-à-pas
- Évaluation prédictive
  - Tests contre un modèle formel (Model Human Processor)
- Évaluation empirique
  - Tests d'interface avec des vrais utilisateurs

# POURQUOI ÉVALUER

Pourquoi chaque étape a-t-elle une composante d'évaluation?

Parce que nous devons relier la progression de notre conception aux besoins, aux tâches et aux contextes de l'utilisateur

Connaître vos outils et choisir efficacement

- Méthodes d'évaluation: outils dans une boîte à outils
- Chaque outil a des avantages/désavantages et un coût d'utilisation

# ÉVALUATION

## Premier design

## Mi-design / avancé

### Pré-design

ethnographie  
observation,  
des entretiens  
groupes de discussion  
questionnaires  
enquêtes

des entretiens,  
groupes de discussion,  
observation  
questionnaires,  
enquêtes

enquête contextuelle &  
modélisation de travail

analyse des tâches,  
tâche / cognitif  
procédures pas à pas

conception participative  
évaluation heuristique

observation,  
des entretiens,  
questionnaires,  
enquêtes  
*en utilisant avancé  
prototypes*

évaluation heuristique

formel  
performance /  
tests d'utilisation



# ÉVALUATION

Certaines techniques sont partout

entretiens, observations, questionnaires

- Toujours utiles

Ils peuvent être exécutés différemment:

- Tôt: pour mieux comprendre
- Plus tard: pour entrer sur votre approche de design et les détails

# EN RÉALITÉ

Il est difficile de suivre le processus idéal

- Les dates limites
- Le budget
- Il est difficile d'impliquer les bons utilisateurs au bon moment
- Les utilisateurs directs peuvent ne pas être les seuls concernés par le système
  - **Les parties intéressées**

# QUI SONT LES PARTIES INTÉRESSÉES?

Une partie intéressée = toute personne qui a des raisons de se soucier de l'interface

- Il est possible qu'il y en ait beaucoup
- Les besoins peuvent être en conflit

*Les utilisateurs directs:* fonctionnalité, facilité d'utilisation

*Les directeurs des utilisateurs:* efficience

*Les développeurs:* la facilité de développement

*Un constructeur:* les coûts de production

*Publicitaire:* visibilité

# QUI SONT LES PARTIES INTÉRESSÉES?

- qui le demandera?
- qui l'utilisera?
- qui décidera de l'utiliser?
  - (ou si quelqu'un d'autre va l'utiliser?)
- qui va payer pour cela?
- qui doit le faire (concevoir / construire)?
- qui doit en tirer profit?
- qui autrement rendra votre vie misérable s'ils ne l'aiment pas?

# QUI SONT LES PARTIES INTÉRESSÉES?

- machine à ultrasons médicale
- jouet pour enfant (2 ans contre 12 ans)
- produit de bureau dans une grande entreprise / université
- département où la compatibilité est importante (par ex. logiciel de courriel)
- article familial coûteux (par exemple une nouvelle fourgonnette)
- StudiUM?

# RÉSUMÉ

## Modèles pour le développement de logiciels

- Le modèle en cascade a du sens pour les projets à faible risque
- Des modèles itératifs ou en spirale sont nécessaires lorsque les exigences sont inconnus (ou mal connus)
- Le développement de l'interface utilisateur est souvent risqué

## Design centré sur l'utilisateur

- Itératif, piloté par prototypes
- Priorité aux utilisateurs et aux besoins dès le début
- Évaluation continue