

Démonstration

6 avril 2023 – IFT2905

*Diapos par Emma Gallifet
(très légèrement modifiées par Laura)*

Rappel:

☆☆ Examen final - Le 20 avril ☆☆

Tu vois ton final
de 2905
approcher et tu
paniques



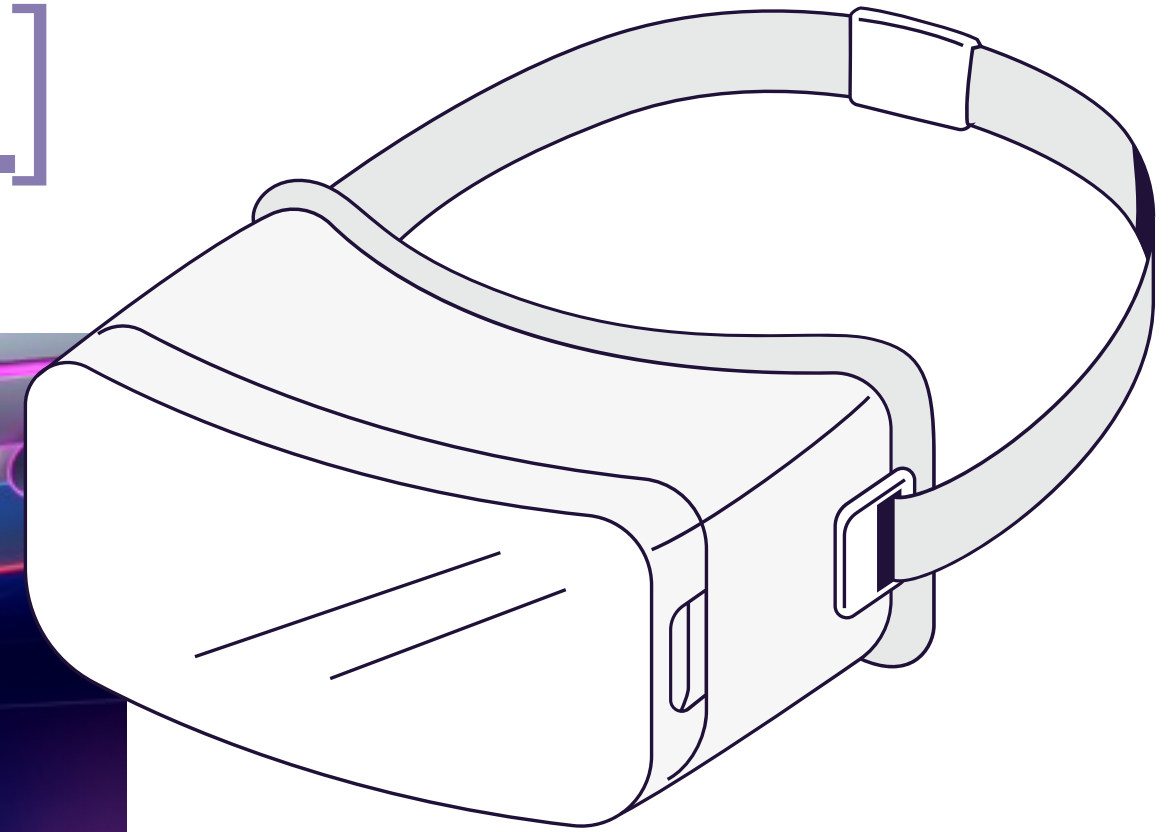
Tu te rappelles
que le prof est
vraiment cool et
met des slides de
révision dans son
site web



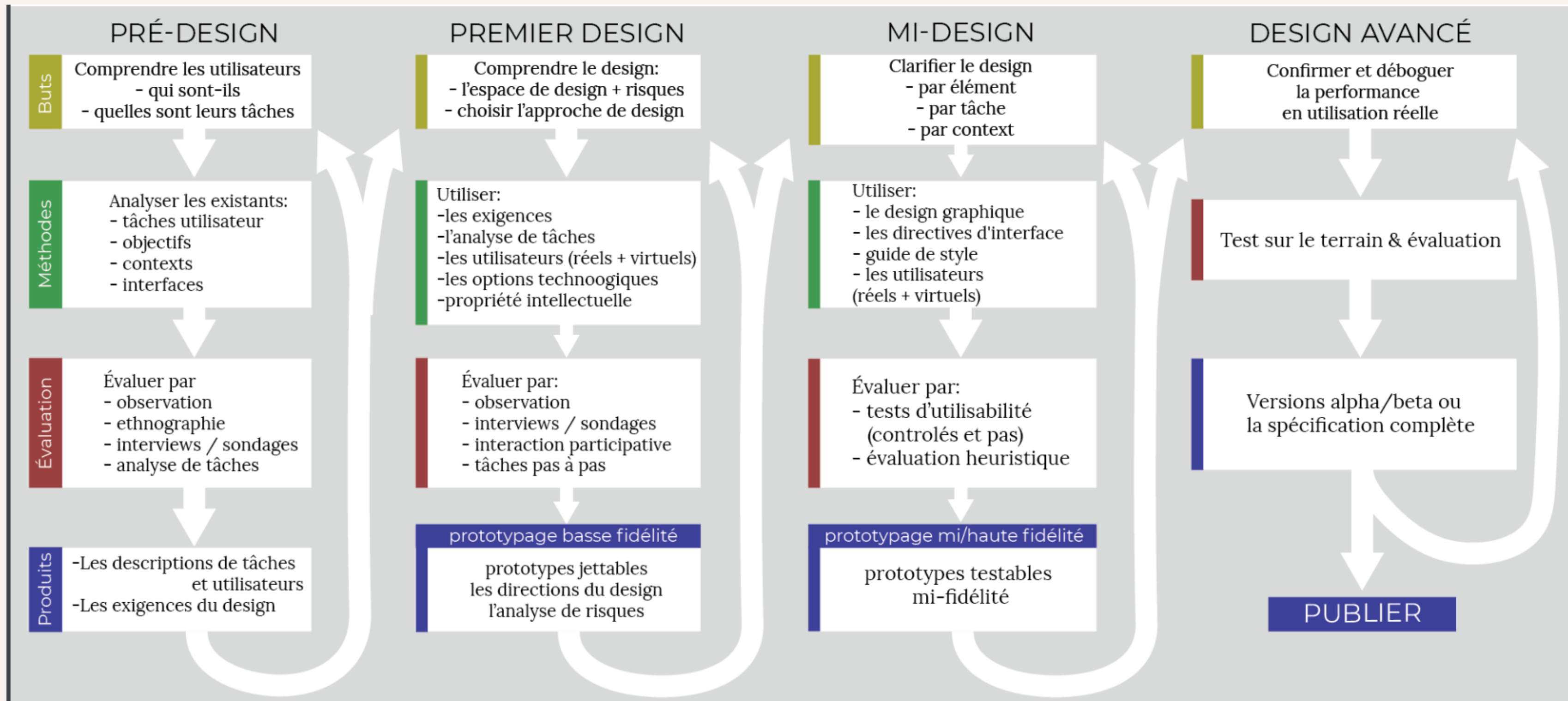
Tes notes au final après avoir
étudié avec les diapos du prof,
revisité tes minidevoirs, et relu le
contenu du site web de tes tpistes



[peut-être: démo vr?]



first steps - oculus quest 2



Pré-Design

Comprendre les utilisateurs

Méthodes

On veut analyser ce qui existe déjà :

- tâches utilisateur
- objectifs
- contexts
- interfaces

Évaluations

On veut évaluer en utilisant :

- les observations
- ethnographie
- interviews/sondages
- analyse de tâches

Produits

On aura comme produit :

- Les descriptions des tâches et utilisateurs
- exigences du design

Sondages

Objectifs : Déterminer qui sont les utilisateurs et que vont-ils faire avec l'application

Questions :

- Quel âge avez-vous?
- Êtes-vous à l'aise avec la technologie?
- Aimeriez-vous utiliser cette application avec vos amis?
- Aimeriez-vous utiliser cette application ?
- Avez-vous des suggestions?
- Voudriez-vous pouvoir utiliser une messagerie?

Sondages : Résultats

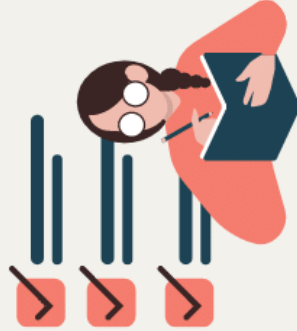
Utilisateurs :

- 15 à 60 ans
- 85% à l'aise avec la technologie et 15% débutant
- 95% des personnes ont répondu qu'ils aimeraient utiliser l'application avec leurs amis
- 22% ont suggéré d'utiliser une carte pour visualiser les données
- 60% ont dit ne pas vouloir de messagerie dans l'application

ÉCRIRE UNE TÂCHE

EN 3 ÉTAPES

Il est très important de comprendre ceci pour ce qui concerne le design basé sur l'utilisateur! Normalement, toutes ces étapes seraient couvertes par une entrevue avec l'utilisateur, mais l'on verra cela plus tard. Notez que ceci est seulement une des plusieurs approches de pre-design!



1. Contexte ♦ Choisir une "persona"

IMPORTANT! bien qu'ils soient sur la même étape, le contexte et la persona ne sont **pas** la même chose. La persona nous aide à créer un utilisateur central pour notre tâche en jeu. Une persona peut contenir un ou plusieurs contextes ou pas de contexte!

Contexte:

Pourquoi?

Le contexte nous permet d'ajouter des détails qui **ne sont pas nécessairement pertinents pour l'application**, mais qui rajoutent des détails à ce qui se passe lors de l'occurrence de la tâche.

Comment?

Tout ce qui est **rajouter "du jus"** à notre description de tâche, qui **n'est pas une tâche concrète** ni des détails concernant la persona.

Astuce pour le contexte:

Si on demandait à l'utilisateur de nous dire pourquoi il aurait besoin d'une telle application, quels détails à propos de son utilisation divulguerait il ou elle?

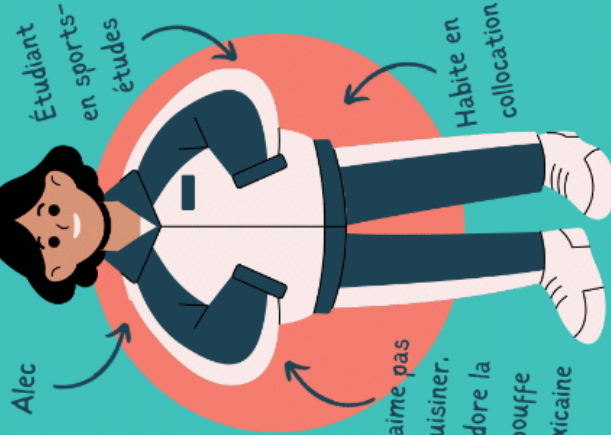
Persona:

Pourquoi?

Ceci nous aide à mieux imaginer comment un utilisateur comme lui ou elle naviguerait l'interface, ce qui aide à l'analyse de celle-ci!

Comment?

Choisir un être humain(!), fictif ou non-fictif, lui **donner un nom**. Si on veut, on lui donne un peu de contexte relié à comment il ou elle utiliserait l'interface.



Alec



Astuce:

Imaginer l'utilisateur en train de nous dire à voix haute **ce qu'il fait**. Si ce dernier utilise son cell, on **n'a pas le droit de regarder** l'écran de ce dernier, pour savoir comment est-ce qu'il le fait.

Pourquoi?

L'on veut savoir qu'est-ce qui se passe **juste avant, pendant et juste après l'utilisation** de l'application. Certaines tâches concrètes pourraient même ne pas l'application du tout, mais nous donner davantage de contexte sur ce qui se passe du côté de l'utilisateur!

Comment?

S'imaginer notre persona en train d'utiliser notre application de manière normale, et **raconter ce qu'il ou elle fait de A à Z** (mieux vaut plus de détails que pas assez!). Attention à:

- **ne pas mentionner des éléments de l'interface elle-même** (boutons, boîtes de textes, menus, etc.).
- Considérer chaque tâche comme représentant une action centrale.

3. Choisir les tâches à soutenir

Exemples de tâches moins pertinentes

Pourquoi?

L'on veut comprendre quels sont les aspects importants qui devraient être présents et faciles à utiliser dans notre application. Aussi, ça nous permettra de savoir (plus tard dans la conception) quels boutons devraient mener à quels écrans, dépendant de l'ordre des actions dans nos tâches!

Comment?

Choisir, parmi les tâches de l'étape précédente (#2), lesquelles concernent l'utilisation de l'application elle-même. L'on ne **rajoute pas plus d'information** que l'étape précédente, mais **on peut enlever certaines parties de phrase** qui seraient moins pertinentes à l'application elle-même.

On peut par la suite faire une **liste** à part avec ces tâches que l'on a choisi, en réécrivant les verbes à l'infinitif.

Alec demande à sa colloque si elle veut aussi commander quelque chose à manger

Il passe son cell à sa colloque

Tâches gardées/à soutenir

Elle ajoute son choix de nourriture

~~Après avoir repris son cell~~
Alec place la commande

ici, l'on a enlevé une partie de la phrase moins importante

Exemple: Alec commande sur Uber Eats!

Légende



Contexte + persona



Tâches concrètes



Tâches concrètes à soutenir

Alec est un étudiant en sports-études. Il adore la bouffe Mexicaine, car ça lui rappelle à ce que sa mère lui cuisinait. Il n'aime toutefois pas cuisiner, donc il commande très souvent de la bouffe avec l'application Uber Eats. Il habite avec sa colloque, Maria. Quand il commande de la bouffe, il ouvre Uber Eats et regarde brièvement la sélection de restaurants qui s'ouvrent à lui. Souvent, il ne sait pas quoi choisir, donc il choisit de regarder sa liste de restaurants préférés pour diminuer le nombre de choix possibles de places. Il choisit presque toujours son restaurant mexicain préféré. Il regarde les plats disponibles, et finit par choisir un de ses plats recommandés. Il ne fait pas d'ajustements aux ingrédients, et ajoute sa commande à son chariot. Alec demande à sa colloque si elle veut aussi commander quelque chose à manger, et lui passe son cell. Elle choisit un repas, et ajuste un des ingrédients. Alec reprend son cell, et regarde dans son chariot que tout est correct. Il place la commande et attend impatiemment son arrivée!

Premier Design

Comprendre le design

Méthodes

- les exigences
- l'analyse des tâches
- les utilisateurs
- les options technologiques
- propriété intellectuelle

Évaluation

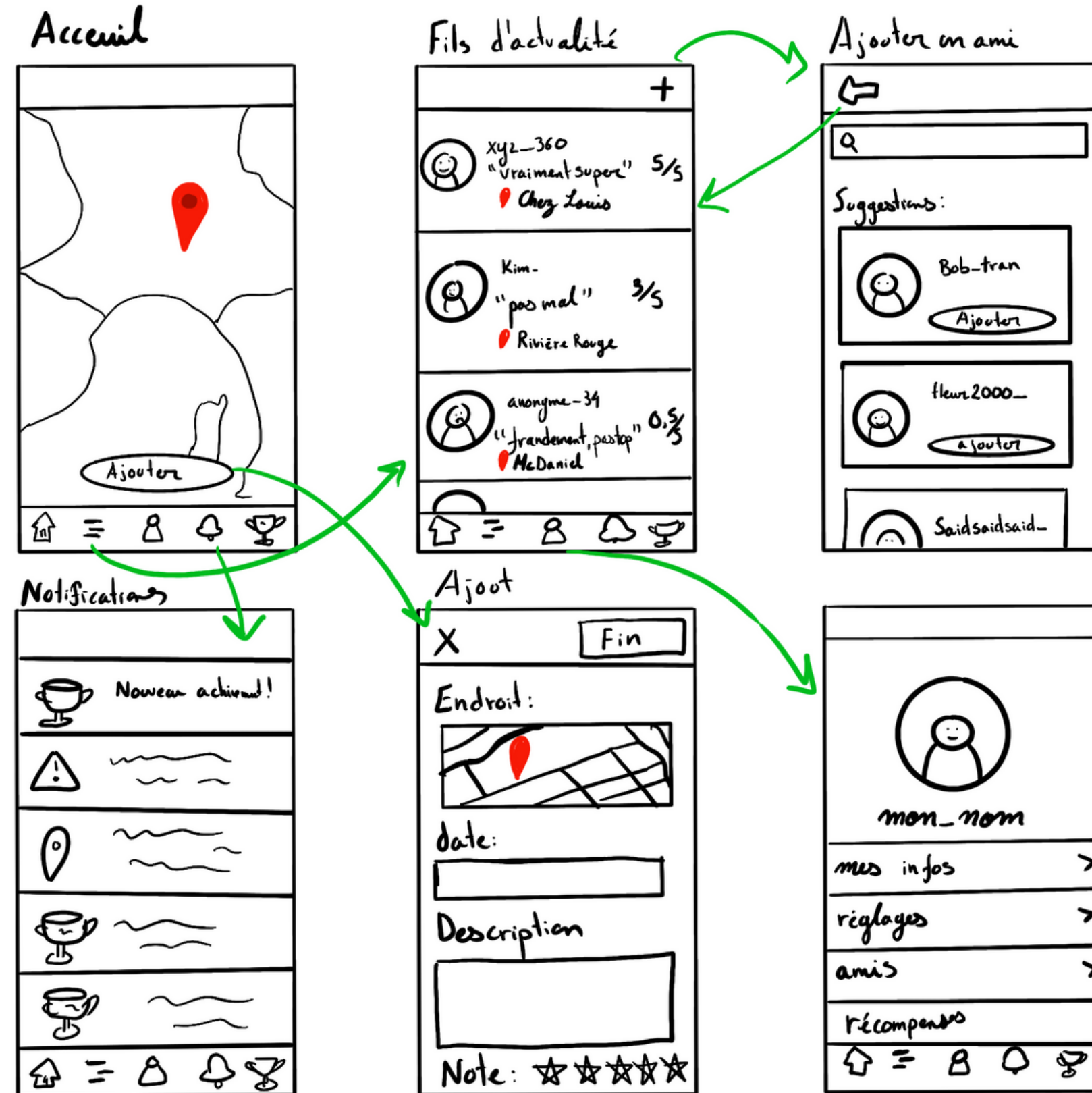
- les observations
- interviews/sondages
- interaction participative
- tâches pas à pas

Produits

- On aura comme produit :
- Prototypes jetables

- Analyse de risques
- Directions du design

Prototype jetable



Évaluation : Cognitive Walkthrough

Objectifs : Vérifier le prototype jetable

Étapes :

1. Extraire les éléments concrets de la tâche
2. Créer le scénario + actions
3. Noter les erreurs et se demander : l'utilisateur saura-t-il quoi faire? comment le faire et comprendra-t-il la réponse du système?

Mi-Design

Clarifier le design

Méthodes

- le design graphique
- les directives d'interface
- guide de style
- les utilisateurs (réels + virtuels)

Évaluation

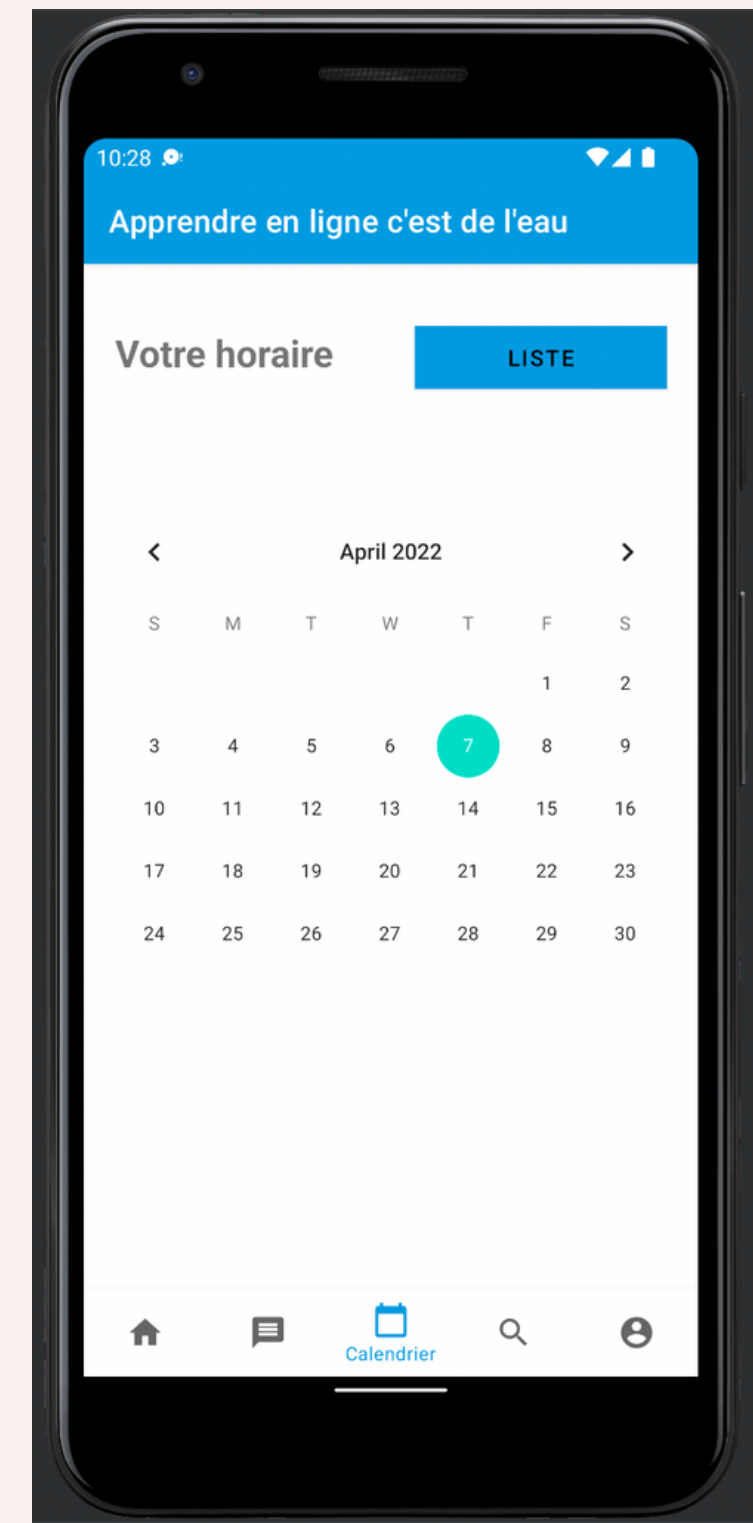
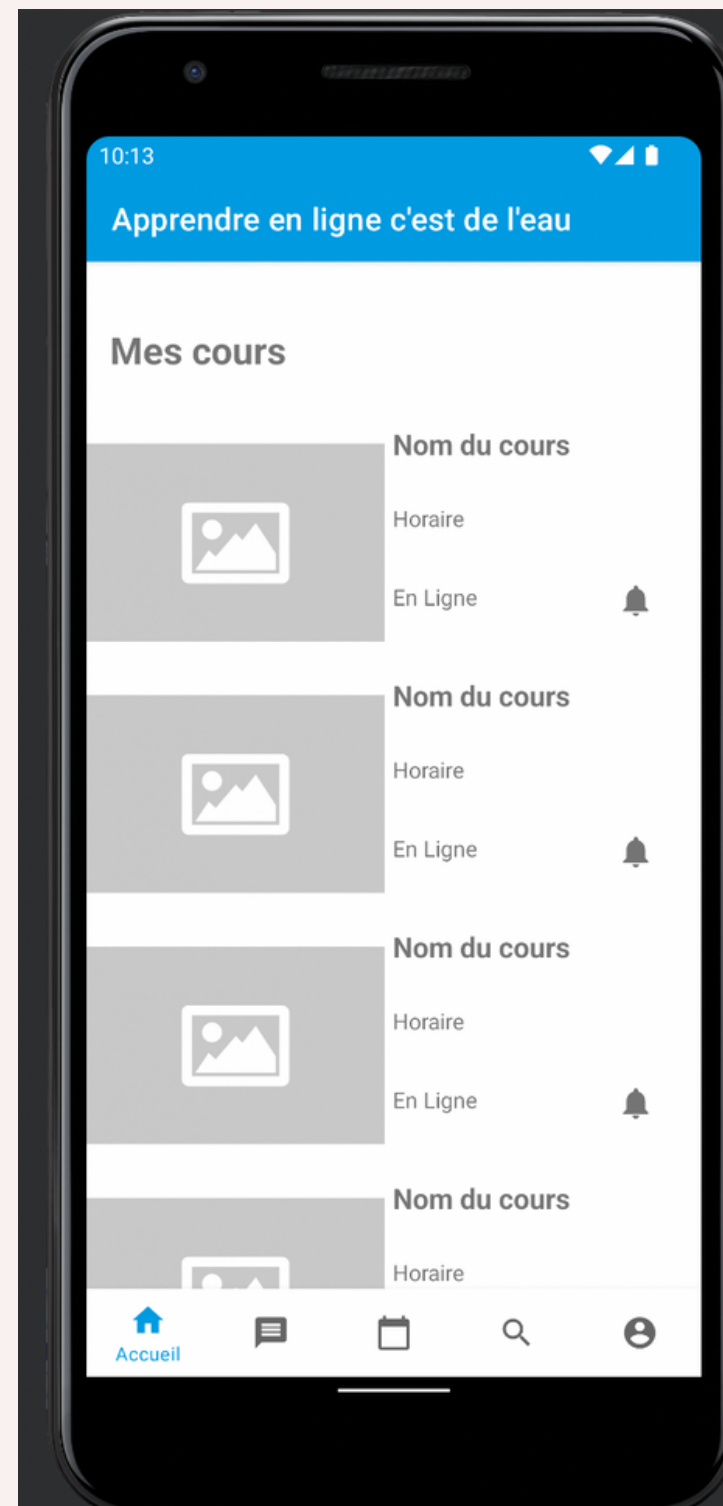
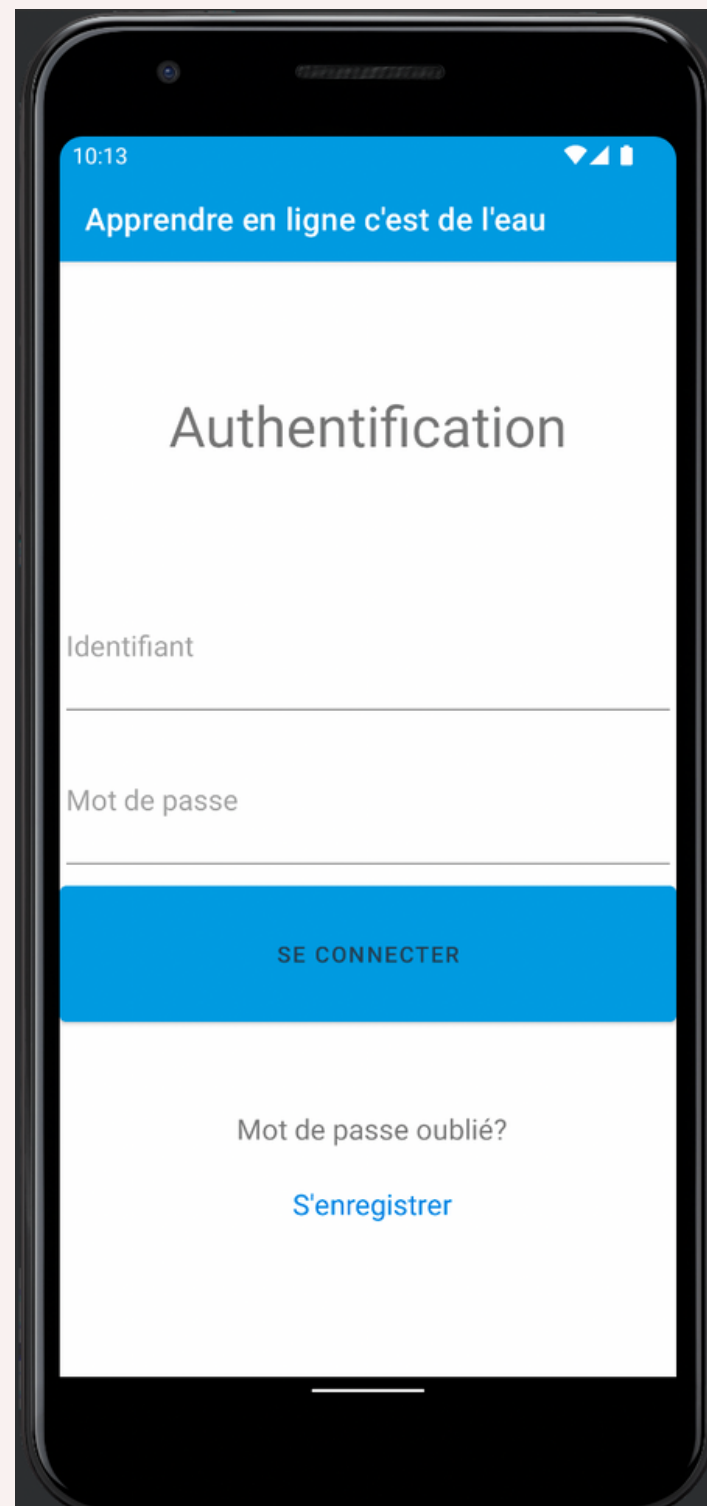
- tests d'utilisabilité
- évaluation heuristique

Produit

On aura comme produit :

- prototypes testables de mi-fidélité

Prototype testables mi-fidélité



Observation

Objectifs : Déterminer si l'application permet de compléter les tâches et si elle est efficiente.

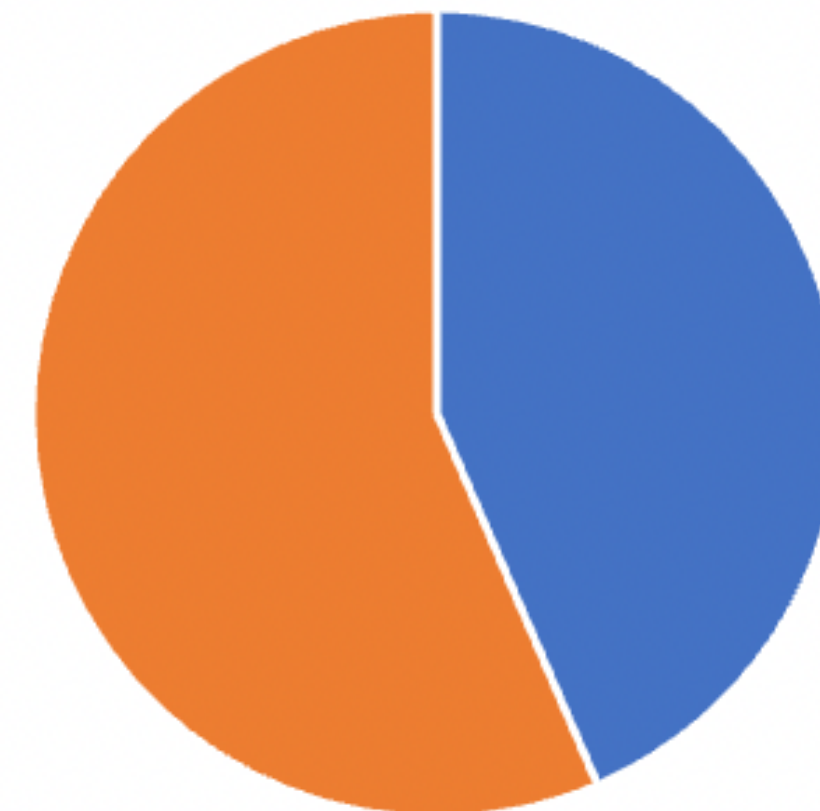
Feuille de codage :

- La tâche a-t-elle été accomplie?
- Le nombre d'erreurs
- La nature des erreurs
- En combien de temps la tâche a-t-elle été réalisée?
- Suggestions de l'utilisateur
- Notes supplémentaires

Observation : résultats

Tableau : Temps en seconde pour chaque tâche et la moyenne				
Utilisateur	Tâche 1	Tâche 2	Tâche 3	Tâche 4
A001	90	280	40	350
A002	120	310	37	96
A003	89	289	39	104
A004	100	305	43	99
A005	400	500	41	300
Moyenne	159,8	336,8	40	189,8

Les utilisateurs ont réussi à accomplir la tâche?



■ Oui ■ Non

Évaluation heuristique



Visibility of
System Status

1



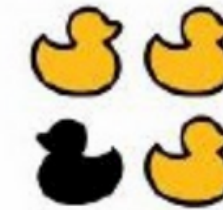
Match Between
System & Real World

2



User Control
And Freedom

3



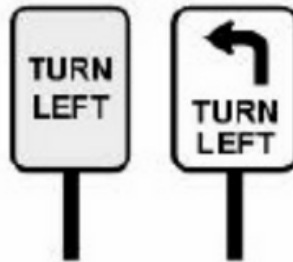
Consistency
And Standards

4



Error
Prevention

5



Recognition
Rather Than Recall

6



Flexibility And
Efficiency of Use

7



Aesthetic And
Minimalistic Design

8



Help Users
With Errors

9



Help And
Documentation

10

Design avancé

Confirmer et déboguer

Évaluation

- Test sur le terrain & évaluation

Produit

- On aura comme produit :
- Version alpha/beta ou la spécification complète



Publier

discours de fin de session



- Considérer l'utilisateur est **très** important si vous voulez qu'il ou elle aime utiliser votre logiciel/machine/etc et que son expérience soit smooth et efficace
- Ceci s'applique **même si vous travaillez sur des choses sont destinés aux ""nerds""** comme vous et moi (ex: une CLI).
- Maintenant que vous savez ce que vous savez, **apprenez aux autres l'importance du ☆☆UX/UI☆☆** dans l'informatique et au delà

