

# Prototypage

Interface Homme Machine  
Nathanaël Martel

# Pourquoi faire des Prototypes ?



# Une maquette fonctionnelle

- **Un système fonctionnel du produit final**
- **Une maquette interactive**
- **Une maquette suffisante pour évaluer l'interface, les performances et le potentiel**

# Pour qui ?

- **Pour sois-même**
- **Pour les autres personnes de la même équipe**
- **Pour les graphistes**
- **Pour les commanditaires (sponsor, client...)**
- **Pour les utilisateurs**

# Pour quoi ?

- Communiquer plus facilement entre les membres d'une même équipe
- Plus simple à comprendre qu'un cahier des charges ou une maquette
- Cela permet d'évaluer et d'avoir un retour sur l'interface
- Vous pouvez tester vos idées sur vous même
- Cela encourage la réflexion, permet d'essayer plusieurs pistes
- Cela permet de répondre à des questions et donc de faire un choix entre plusieurs possibilités

# Qu'est-ce qu'il faut prototyper ?

- **Les problèmes techniques**
- **Le scénario**
- **La mise en page et l'affichage de l'information**
- **Ce qui pose problème...**

# Choix de prototypage

- **fidélité (réalisme)**
- **horizontal / Vertical**
- **évolutif / Jetable**

# Possibilités

- **croquis, dessins (wireframe)**
- **maquette à base de papier**
- **série de croquis de scenes (storyboard)**
- **Une présentation linéaire (powerpoint)**
- **Une vidéo simulant l'utilisation du système**
- **Une maquette physique (bois, carton)**
- **Un morceau de logiciel (avec des fonctionnalités limitées)**



# Prototype basse fidélité

- **Papier, crayon, gomme**
  - Un croquis de l'aspect extérieur du système prévus
  - Facile, évolutive et peu coûteuse
- **Dans quel cas ?**
  - « brainstorming », tester différentes idées
  - Susciter des réactions utilisateurs
  - Susciter des modifications / suggestions

# Prototype basse fidélité : Croquis

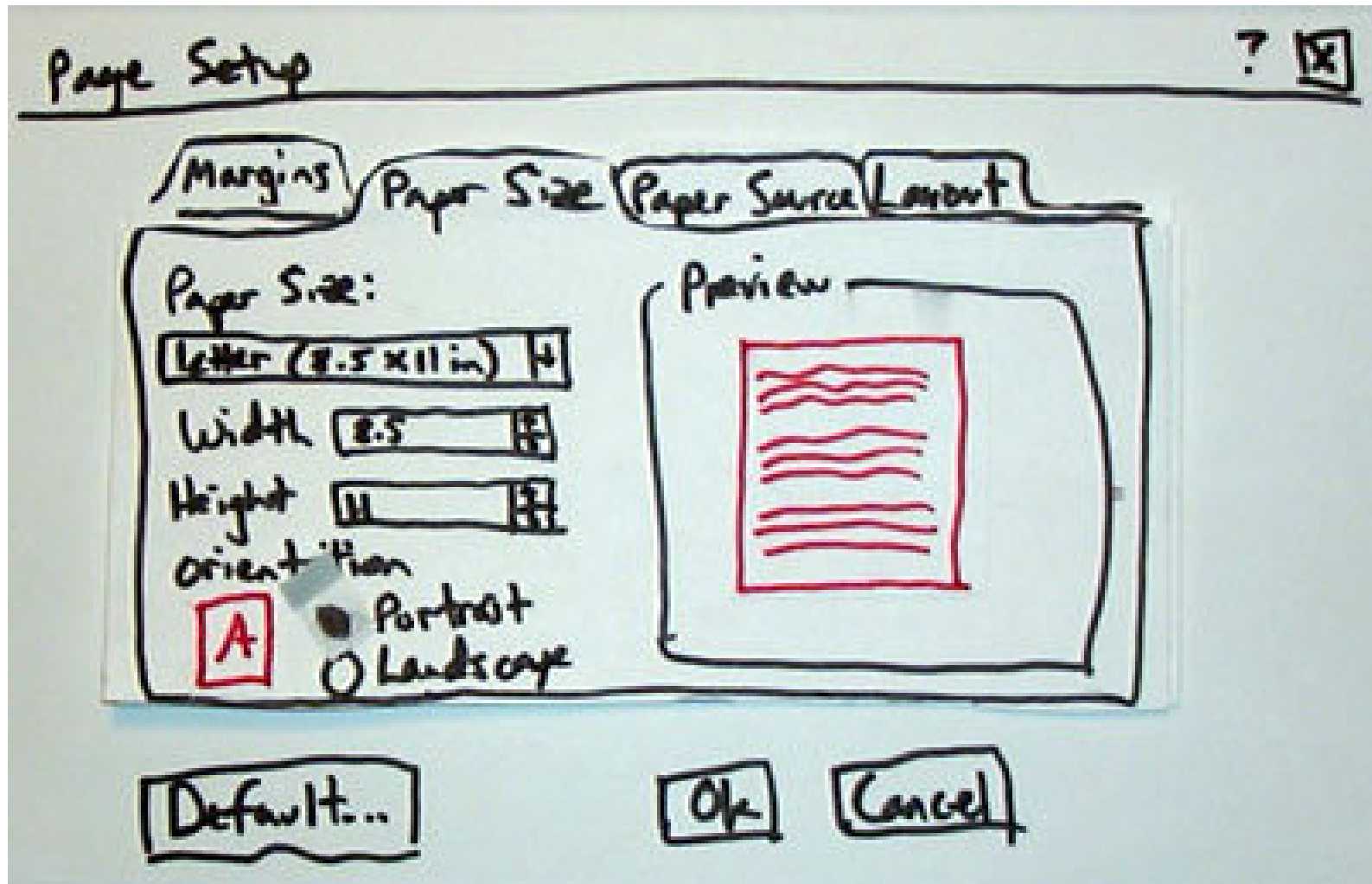
- **Avantages :**

- Comme c'est crû, les gens se concentrent sur un haut niveau d'abstraction
- C'est très rapide et très facile à faire

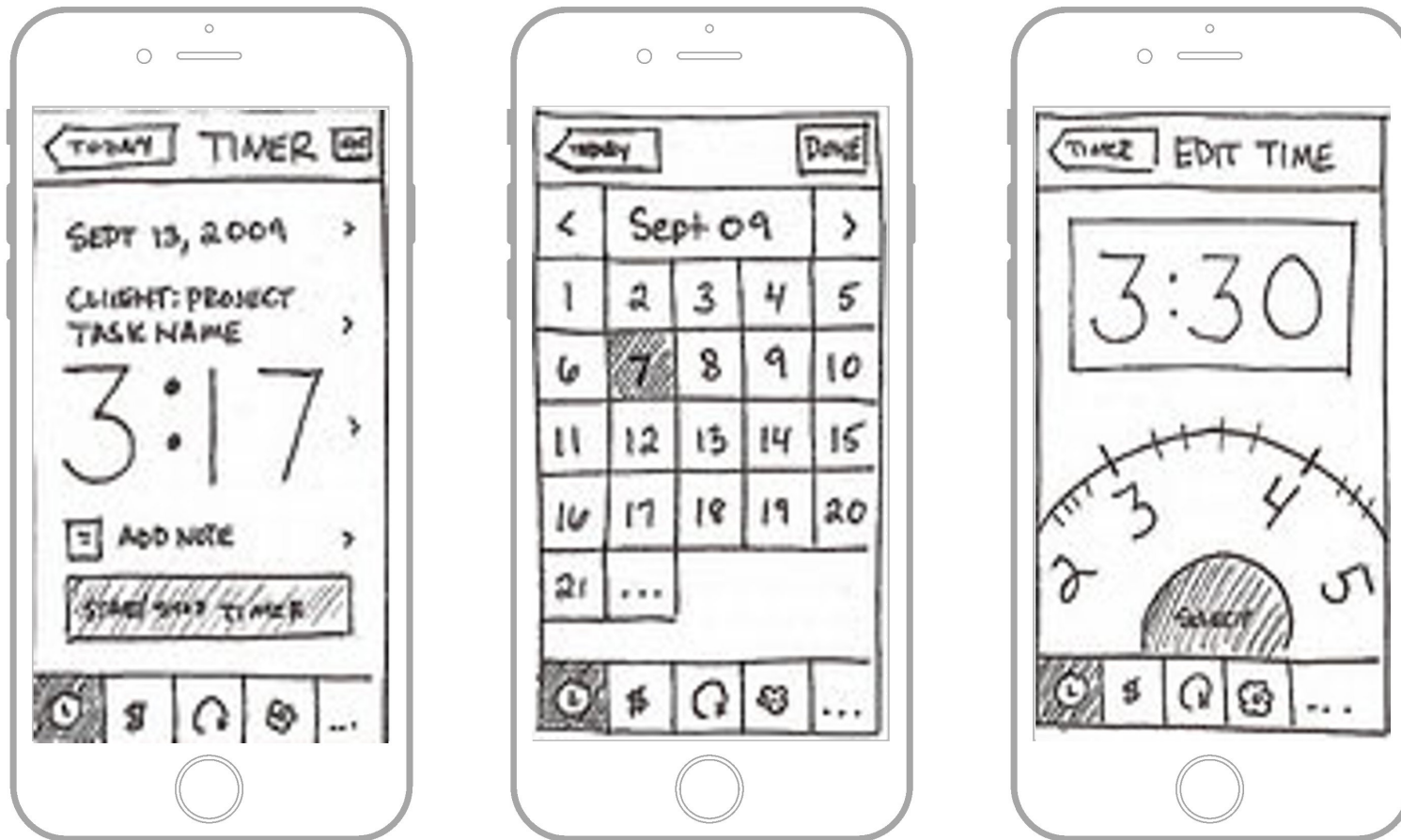
- **Inconvénient :**

- Difficile de montrer un scénario

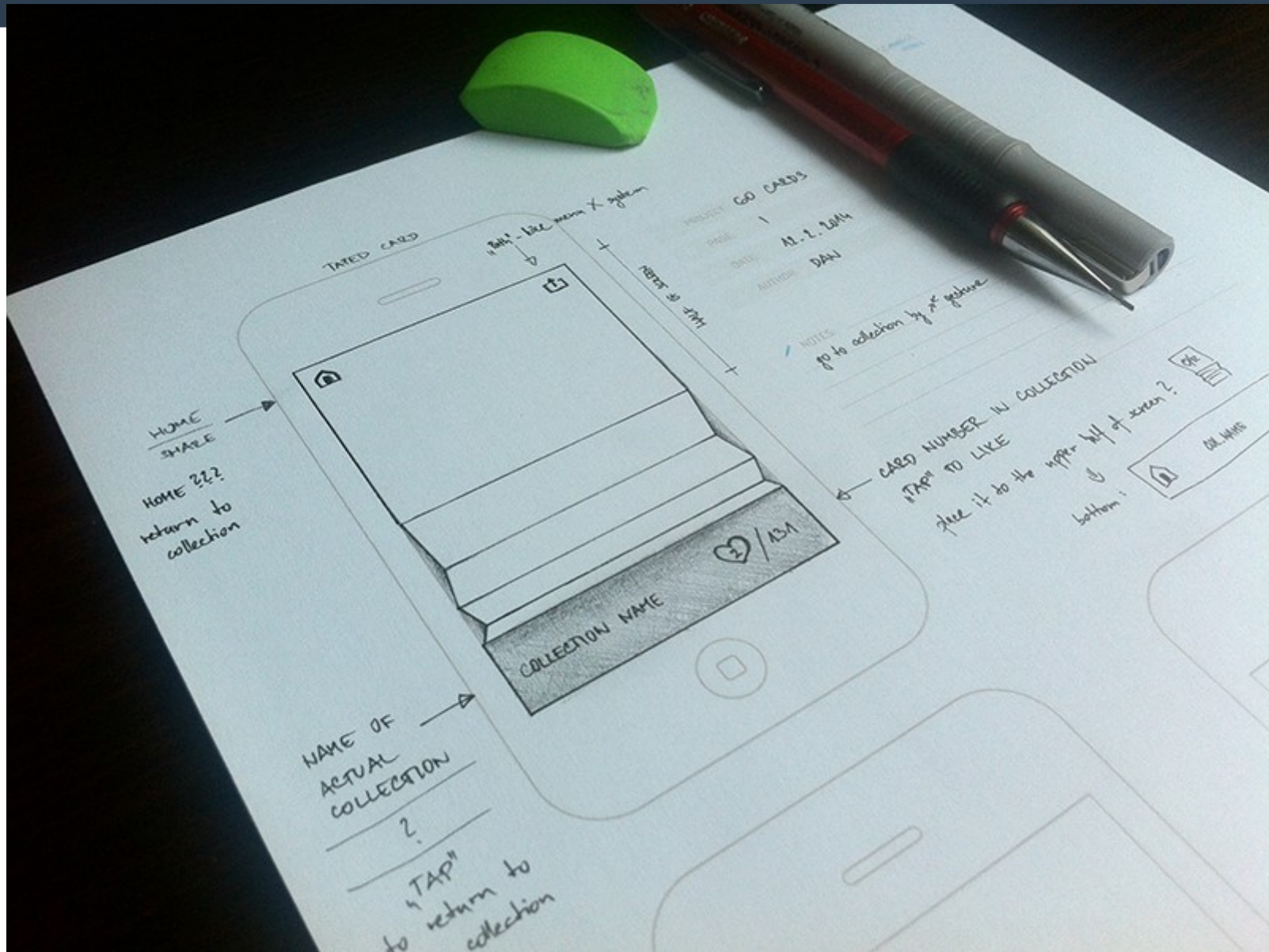
# Prototype basse fidélité : Croquis



# Prototype basse fidélité : Croquis

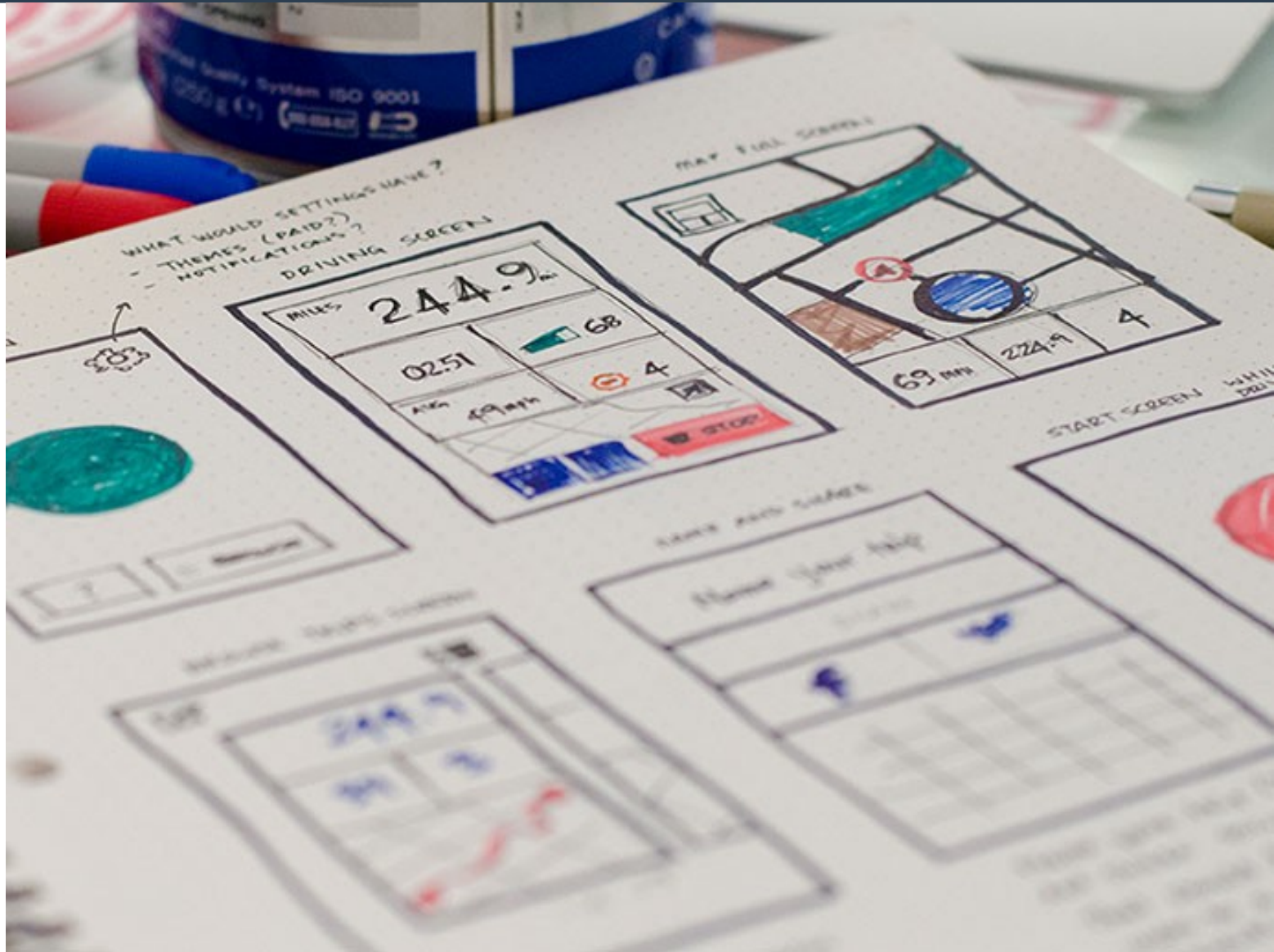


# Prototype basse fidélité : Croquis

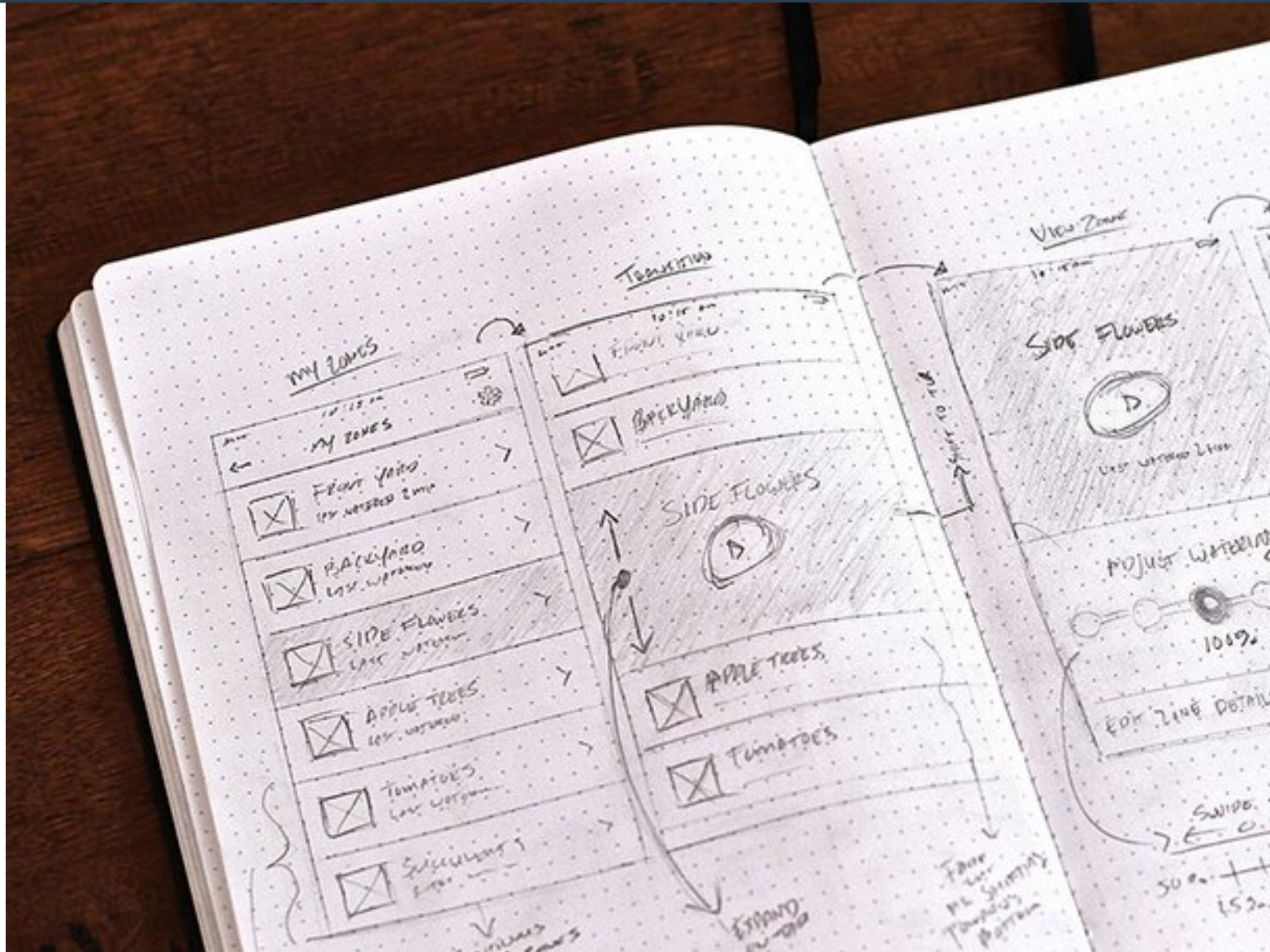




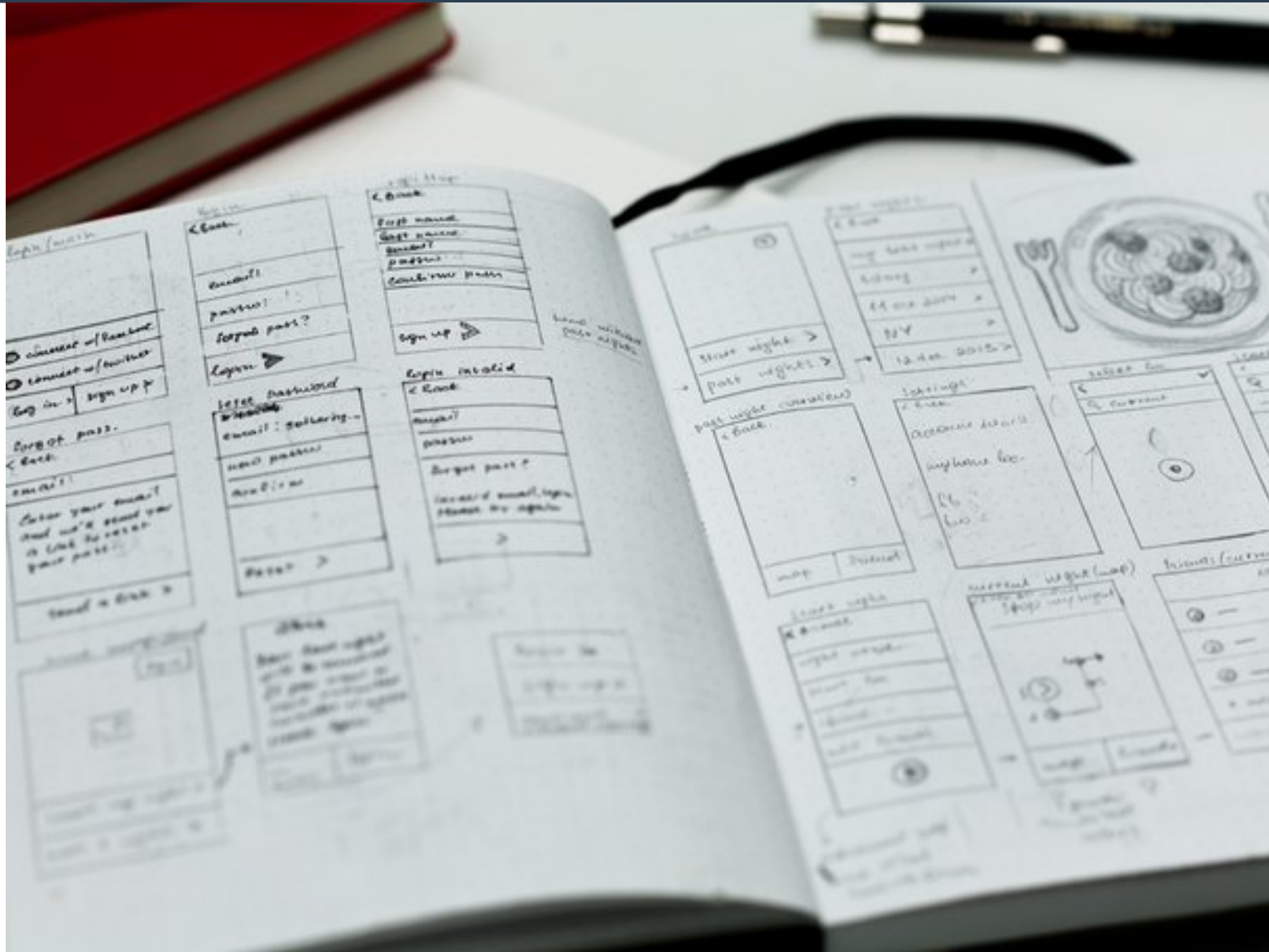
# Prototype basse fidélité : Croquis



# Prototype basse fidélité : Croquis

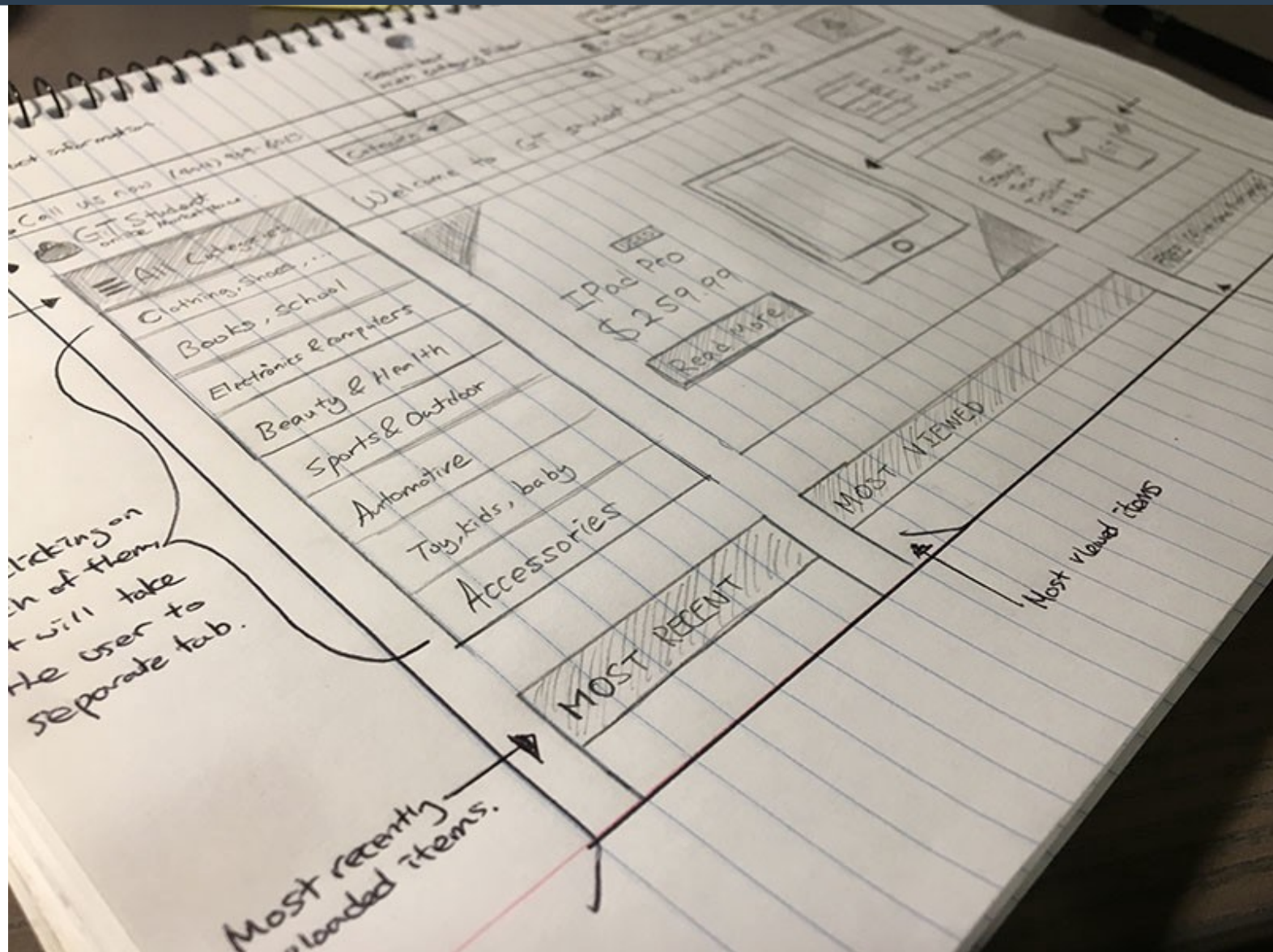


# Prototype basse fidélité : Croquis

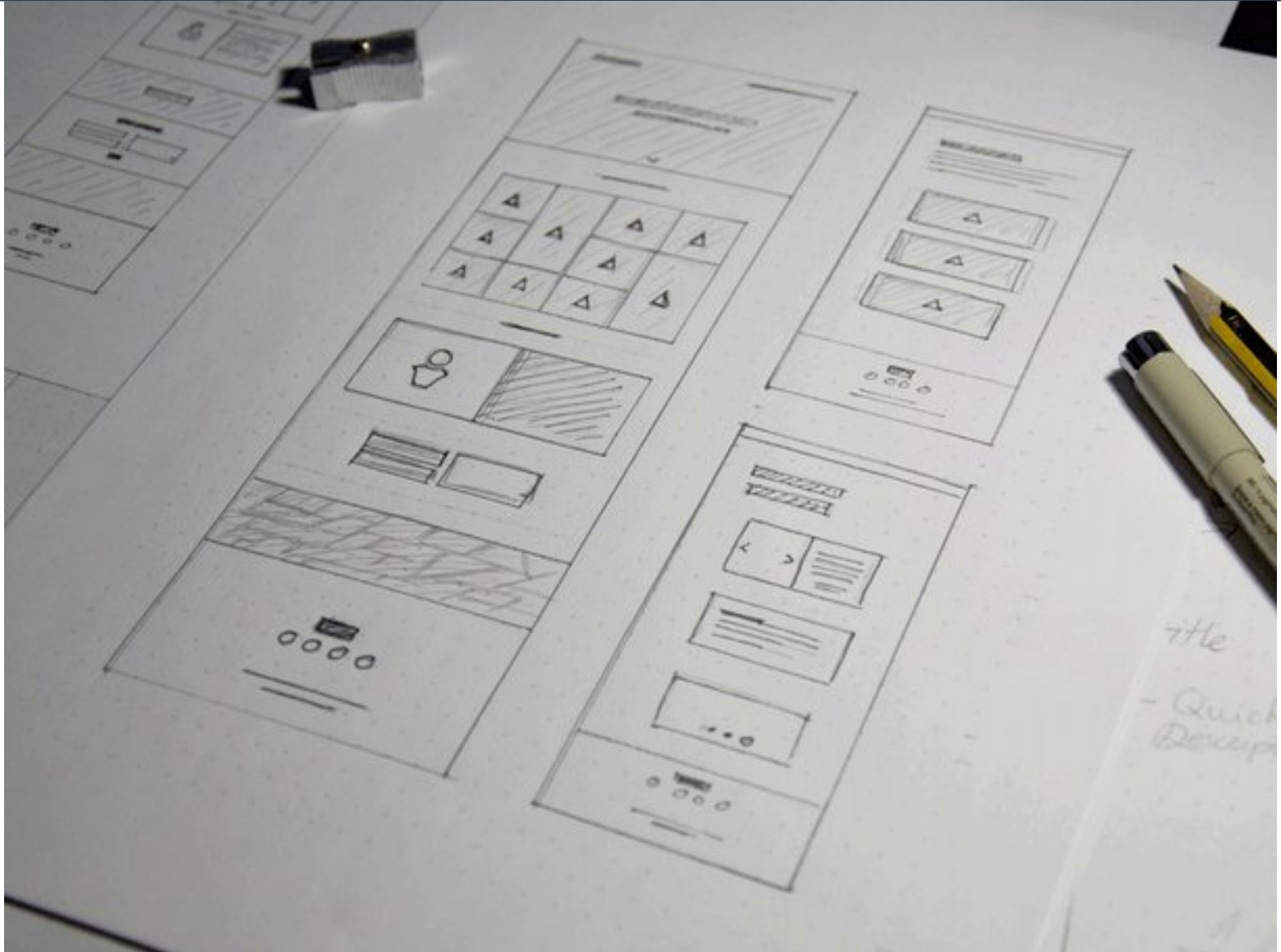




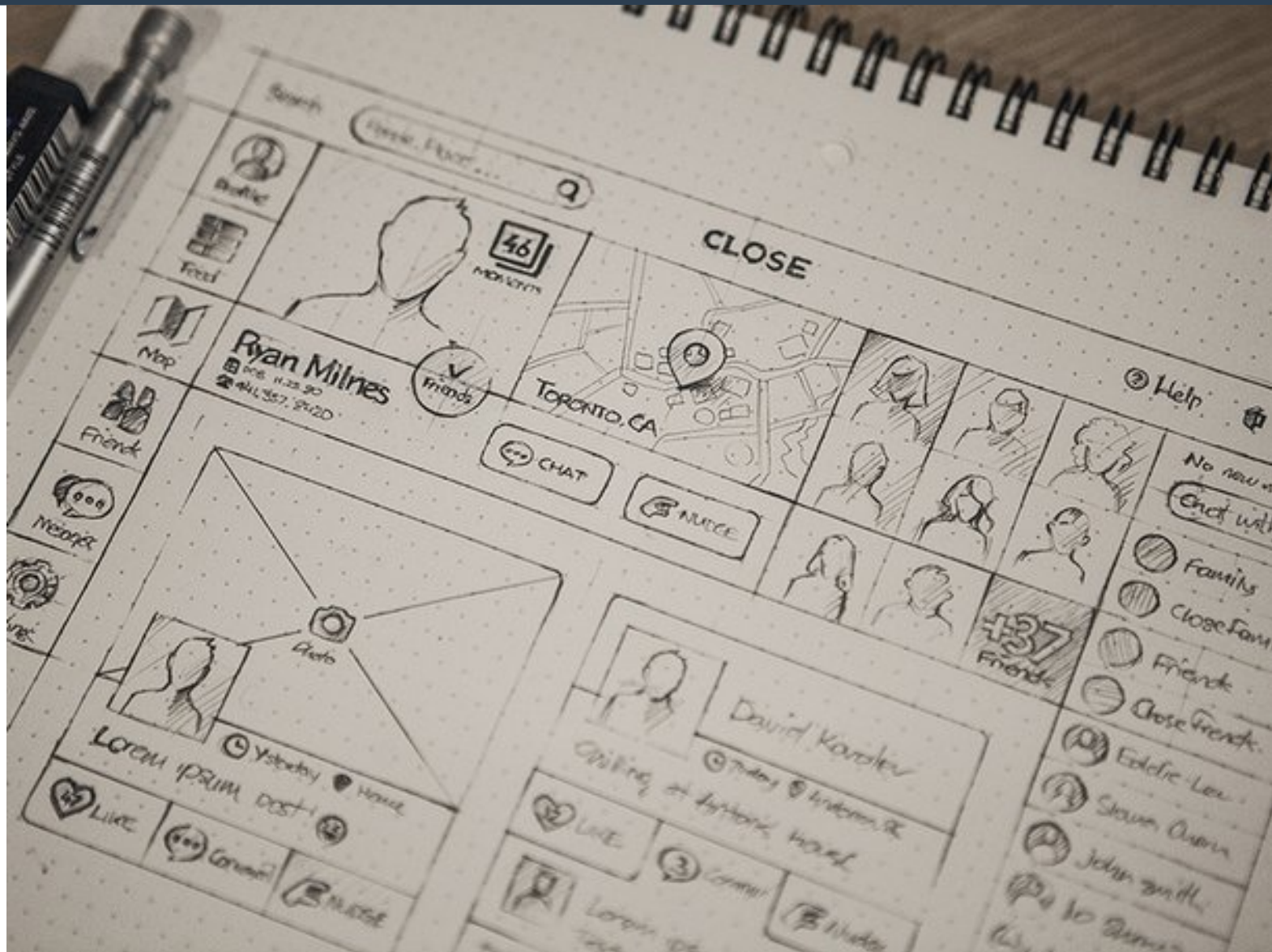
# Prototype basse fidélité : Croquis



# Prototype basse fidélité : Croquis

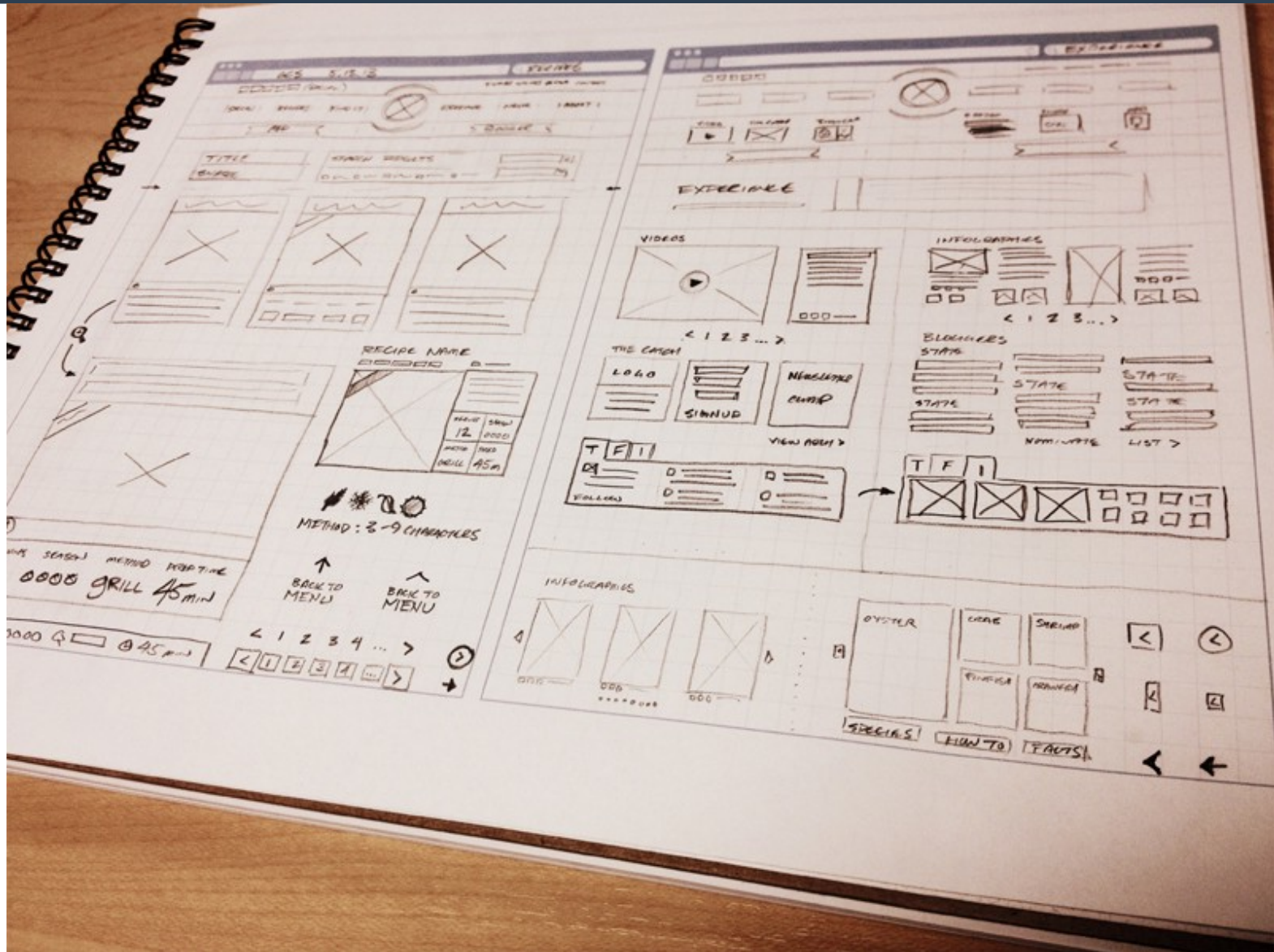


# Prototype basse fidélité : Croquis





# Prototype basse fidélité : Croquis

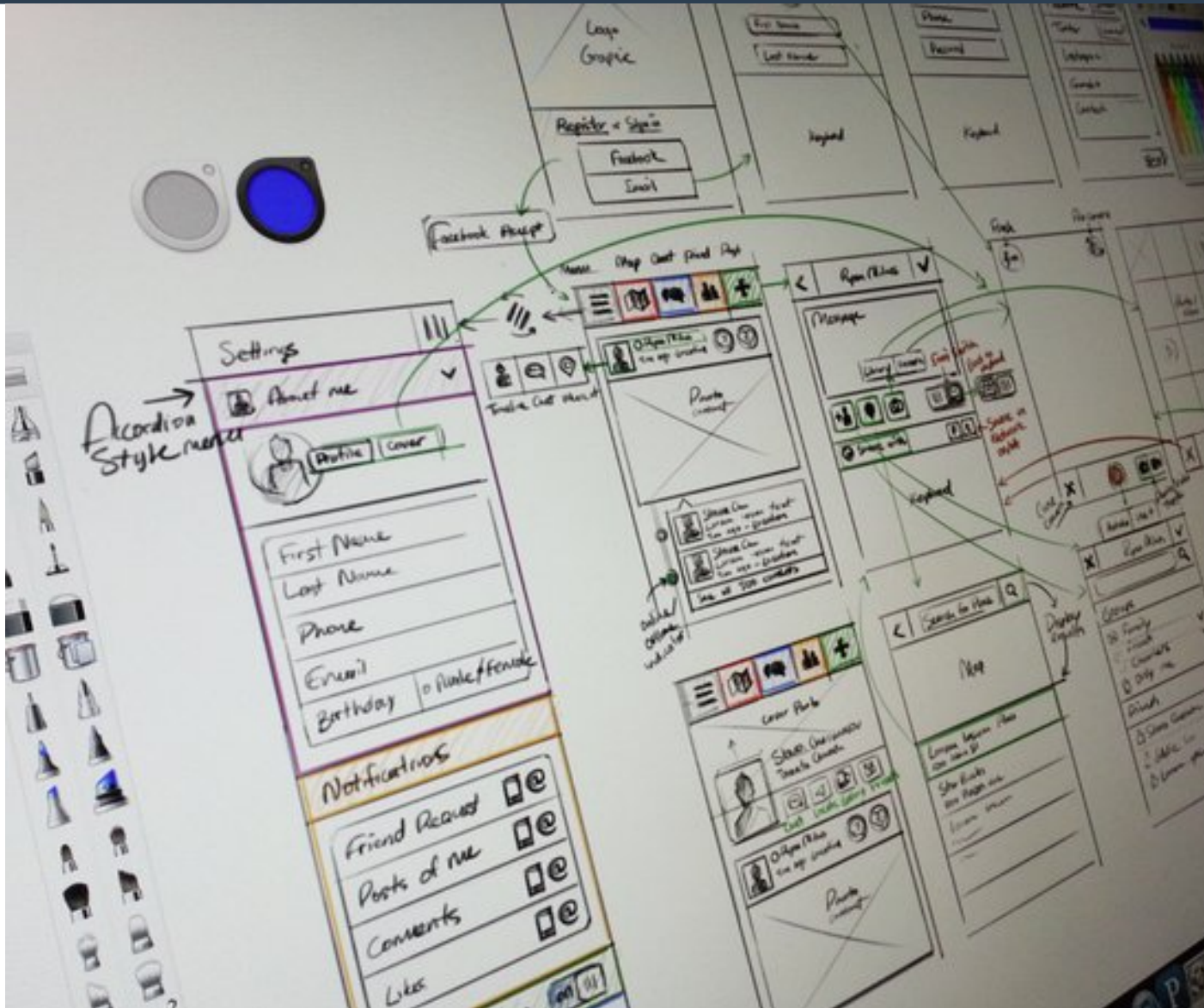


# Prototype basse fidélité : Croquis





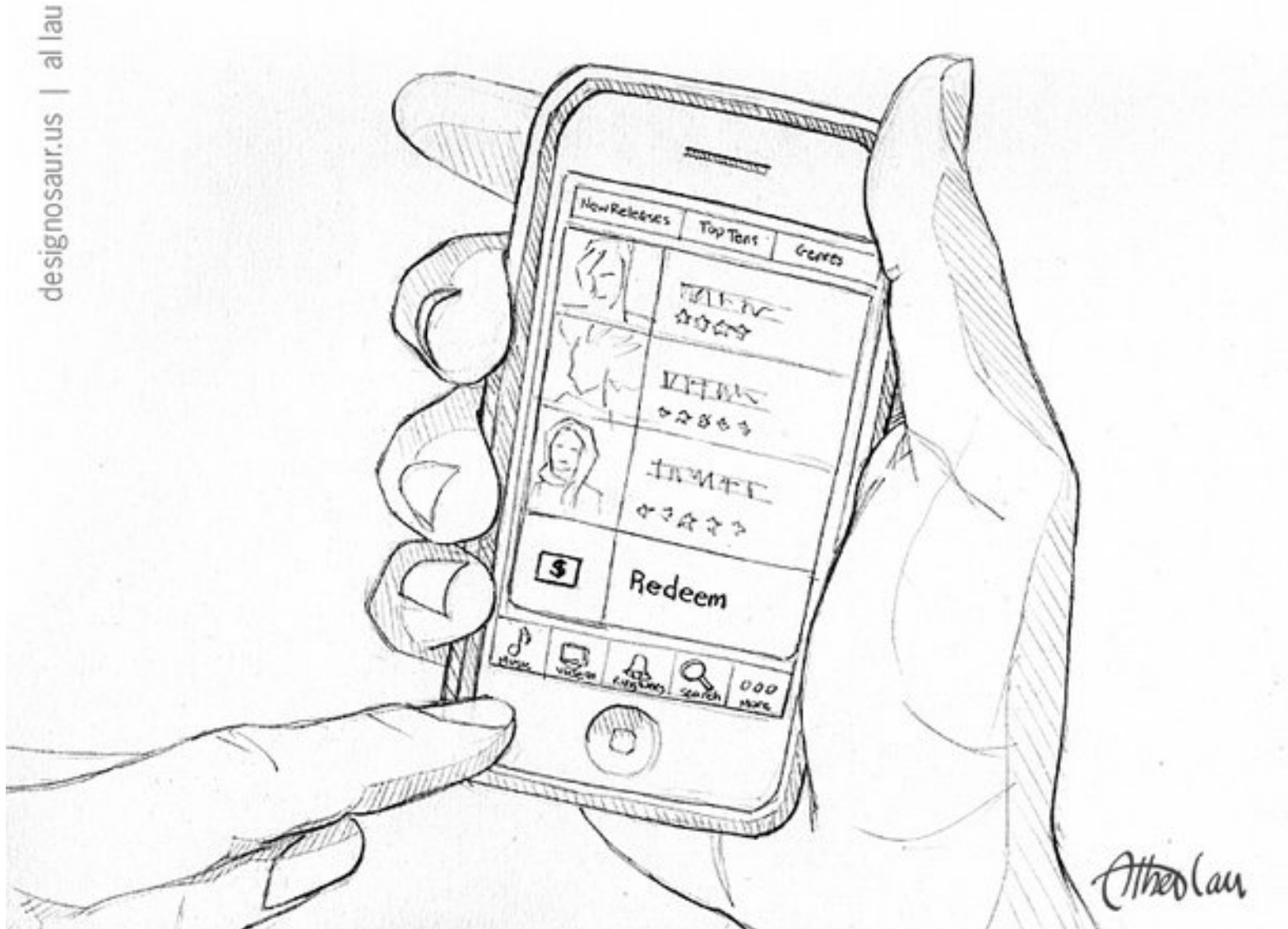
# Prototype basse fidélité : Croquis



# Prototype basse fidélité : Croquis



# Prototype basse fidélité : Croquis





# Prototype basse fidélité : Croquis



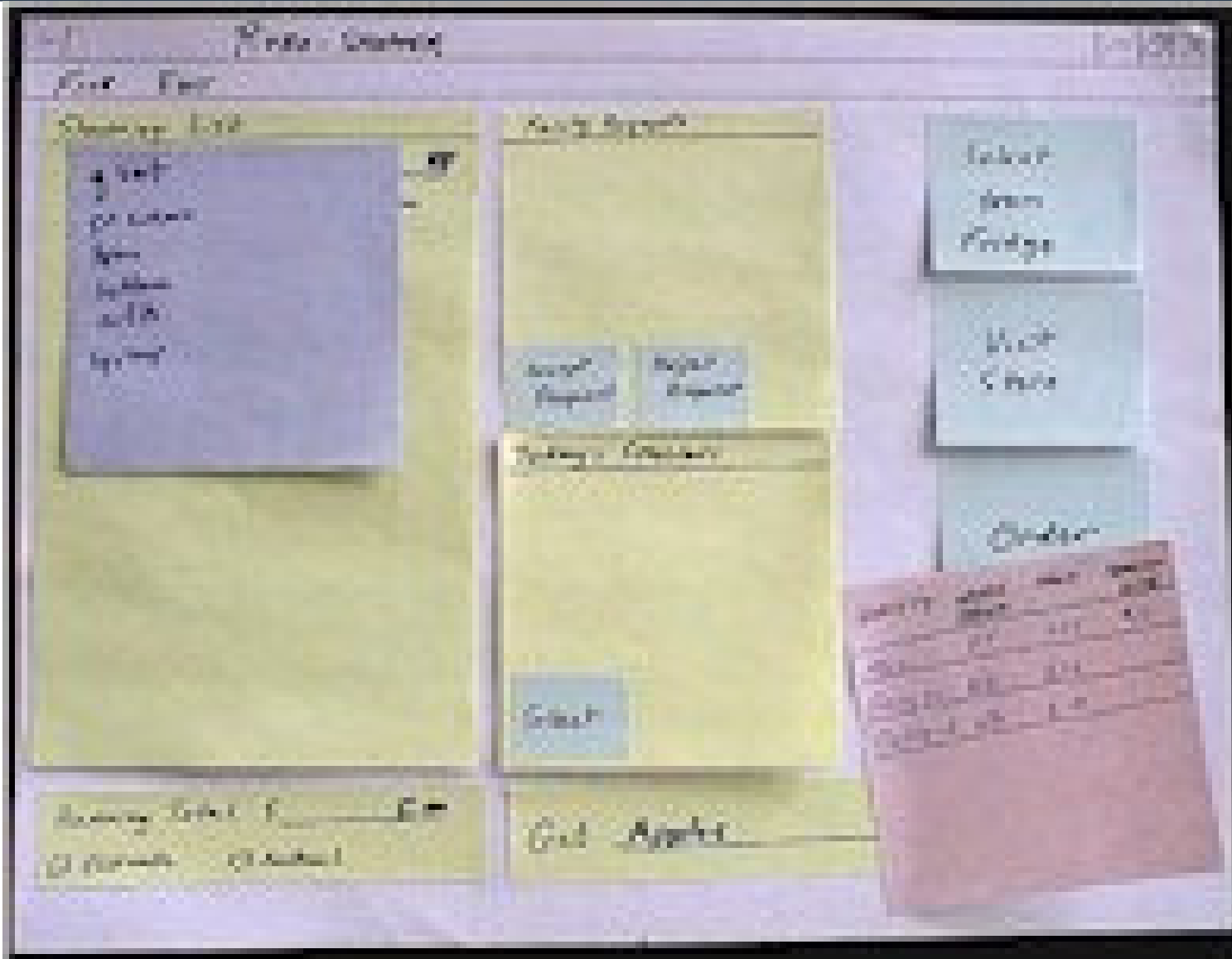
# Prototype basse fidélité : Maquette

- **+ post-it, imprimante, ciseaux, colle...**
- **Impression de pictogrammes, d'éléments d'interfaces, découpage de ces éléments pour pouvoir les manipuler ensuite**
  - Une maquette papier de l'interface (apparence, fonctionnalité)
- **Inconvénient :**
  - Difficile de montrer un scénario

# Prototype basse fidélité : Maquette



# Prototype basse fidélité : Maquette



# Prototype basse fidélité : Maquette





# Prototype basse fidélité : Maquette

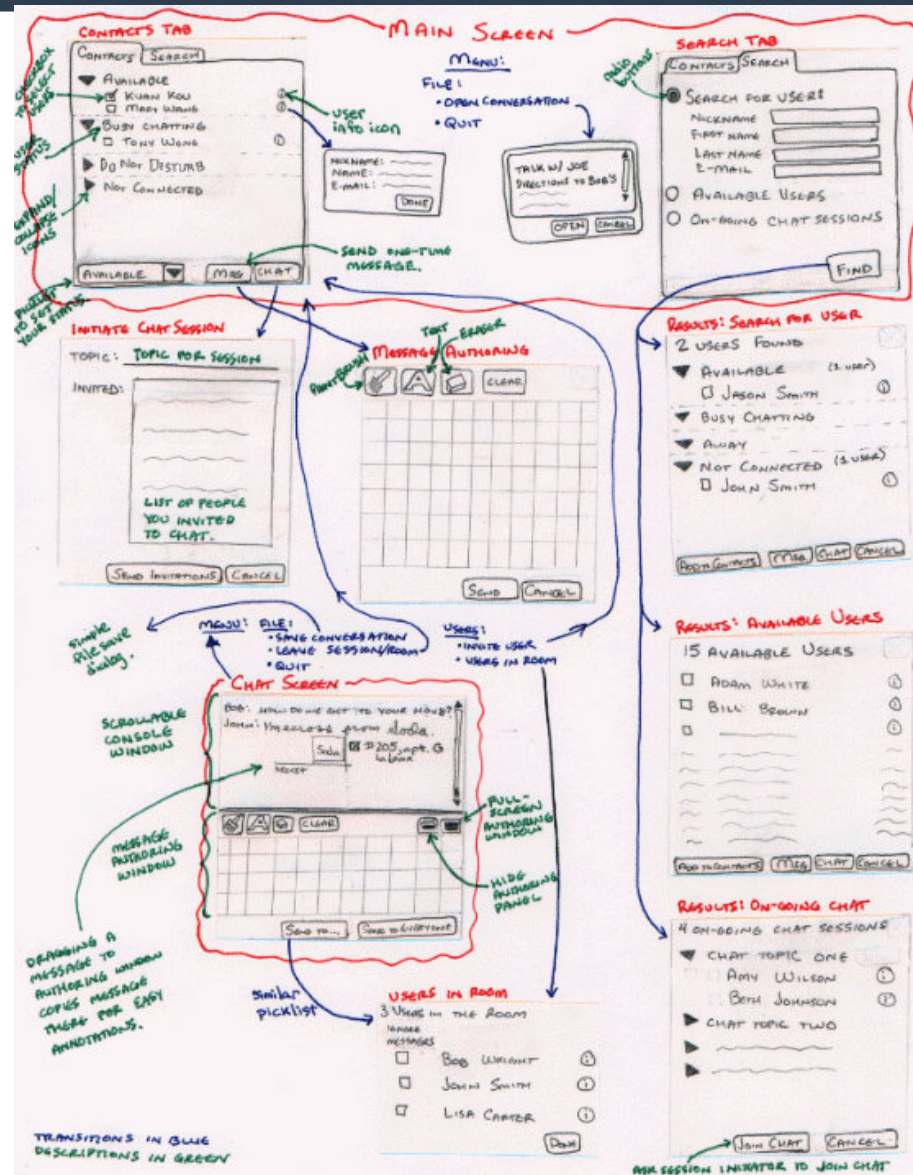


# Prototype : Storyboarding

- **Une série de croquis ou de maquettes**
  - Pour dérouler un scénario
  - montrer quelques points d'interactions particulier
- **Avantage :**
  - Comprendre rapidement la cohérence de l'interface.



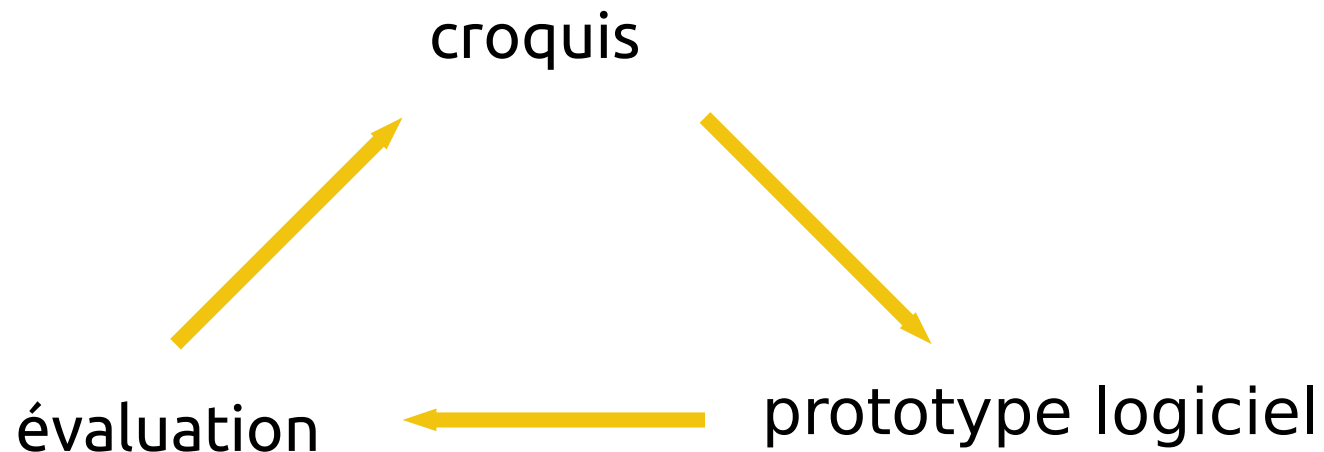
# Prototype : Storyboarding





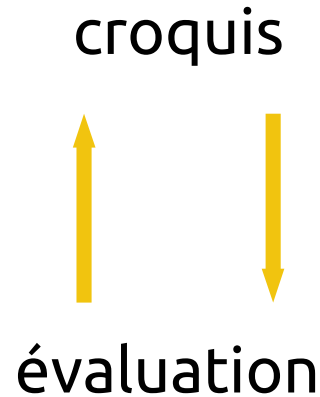
# Prototype basse fidélité

- Pourquoi faire des prototype basse fidélité ?



# Prototype basse fidélité

- Si on peut faire des tests sur le croquis, ça ira plus vite !



- le croquis est un prototype

# Prototype basse fidélité

- **Pourquoi faire des prototype basse fidélité ?**
  - Pas cher !
  - Facile à réaliser
  - Facile à changer
  - Peut être très ambiguë
  - Peut être modifier par le testeur
  - Peut convenir à n'importe quel plateforme

# Prototype moyenne fidélité

## Différentes approches pour limiter les fonctionnalités :

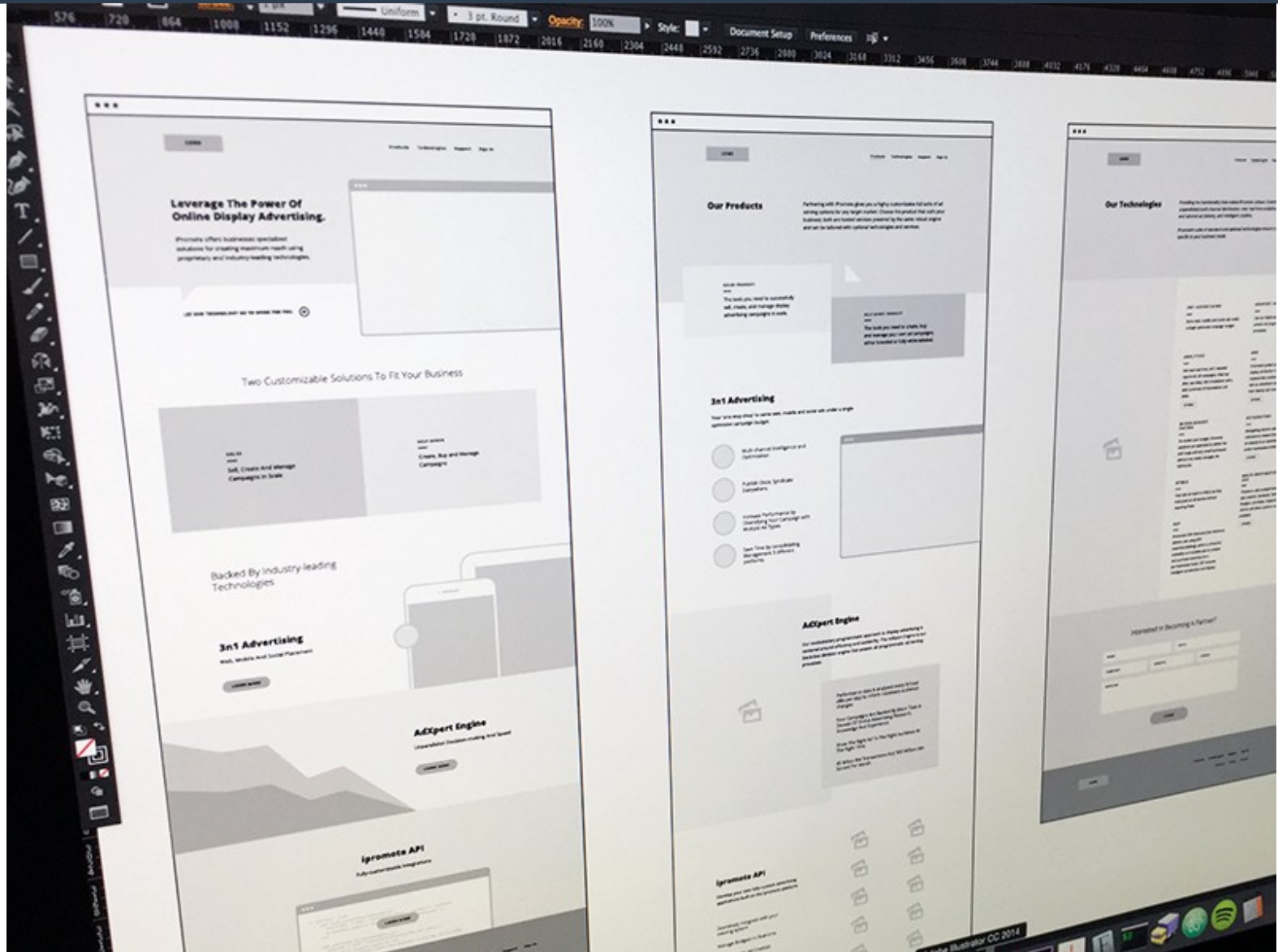
- **Vertical :**
  - réaliser une fonctionnalité complètement (ou quelques unes)
  - le scénario peut être testé en profondeur
- **Horizontal :**
  - réaliser toute l'interface sans les fonctionnalités
  - une simulation, rien ne fonctionne vraiment
- **Idéal pour tester un scénario sans déviations possible**

# Prototype moyenne fidélité

## Dessins / Maquette

- Dessiner chaque scène de storyboard à l'ordinateur
- Un prototype horizontal
- Pas d'interaction

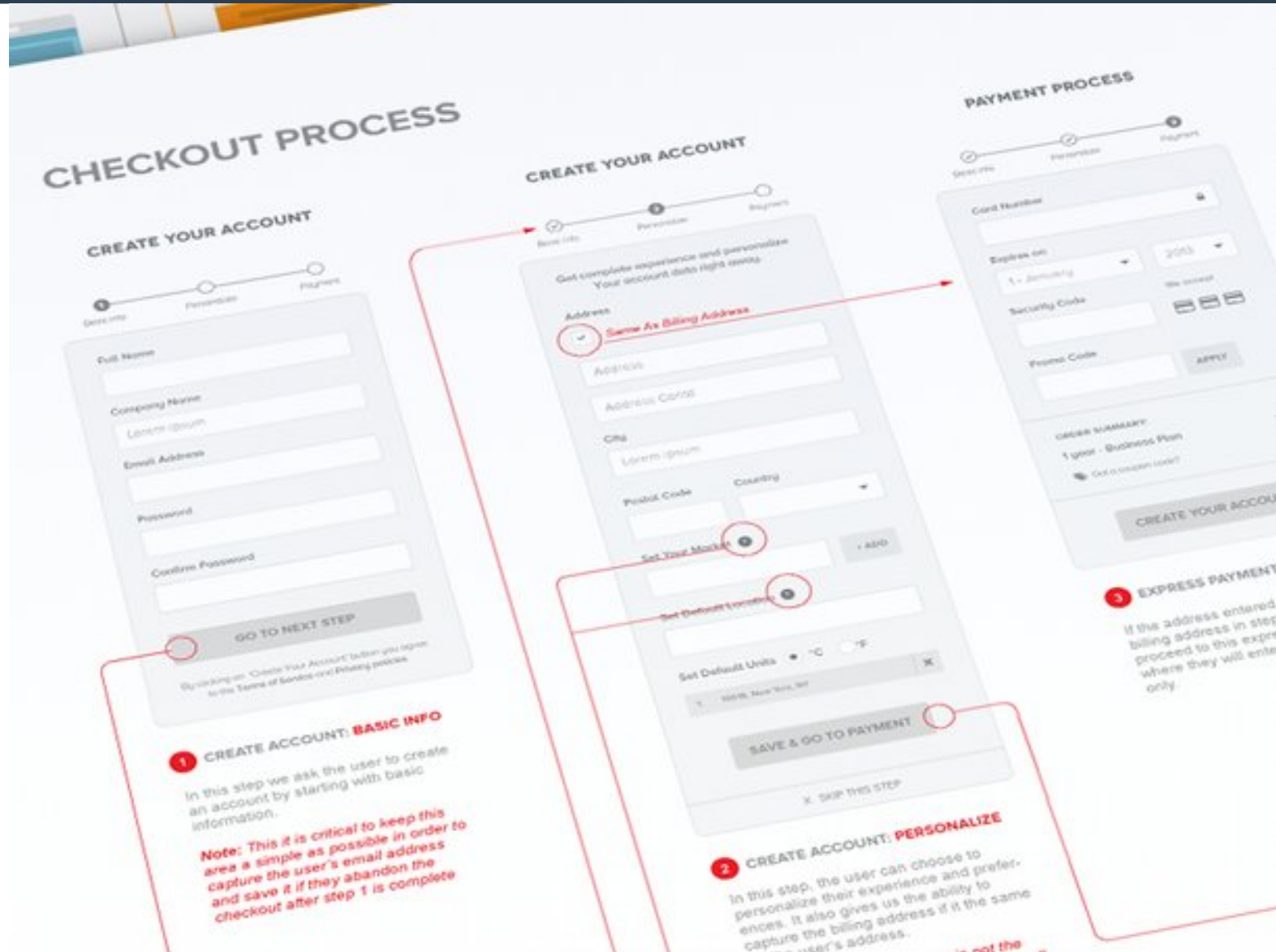
# Prototype moyenne fidélité



# Prototype moyenne fidélité



# Prototype moyenne fidélité

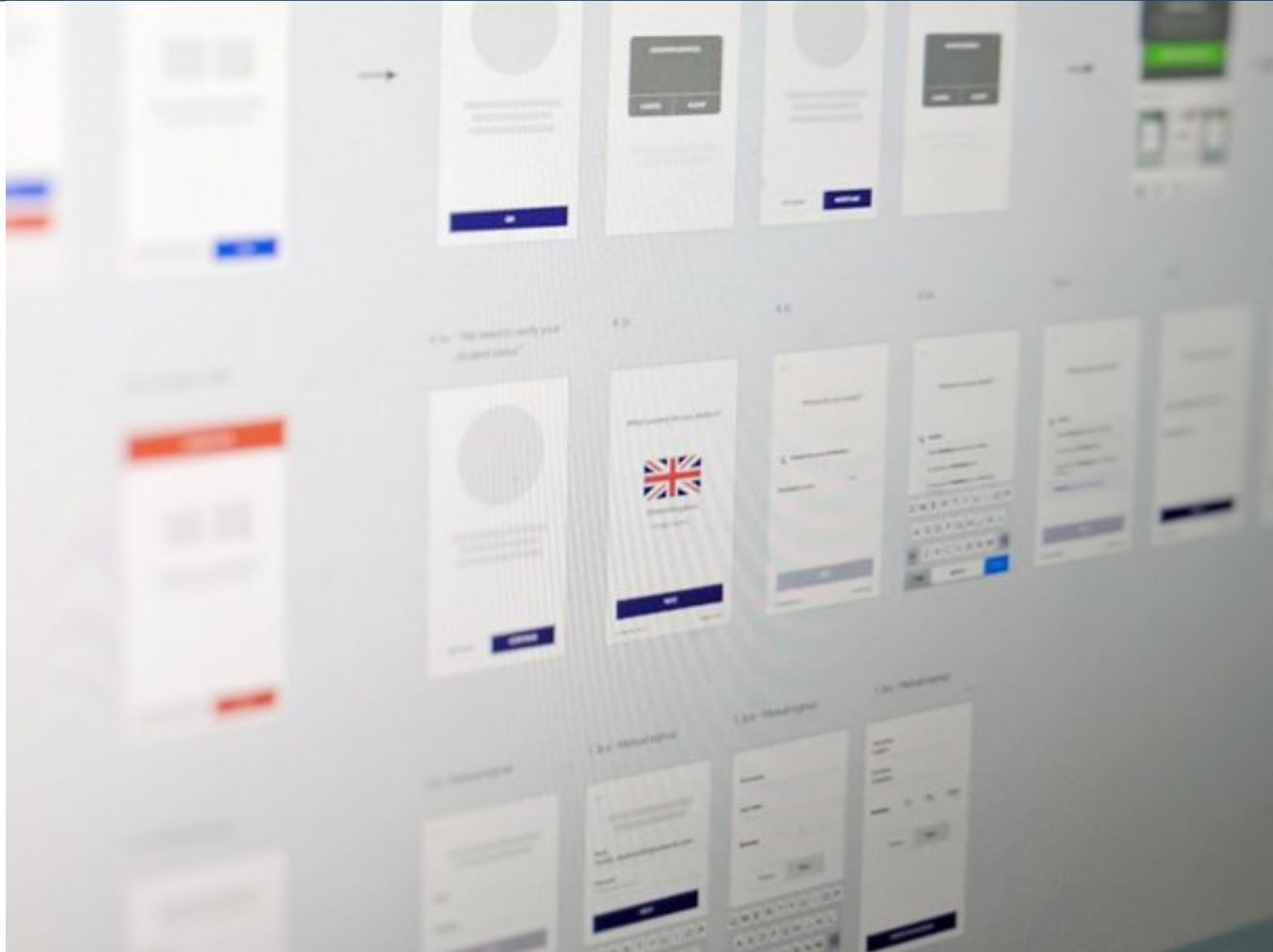




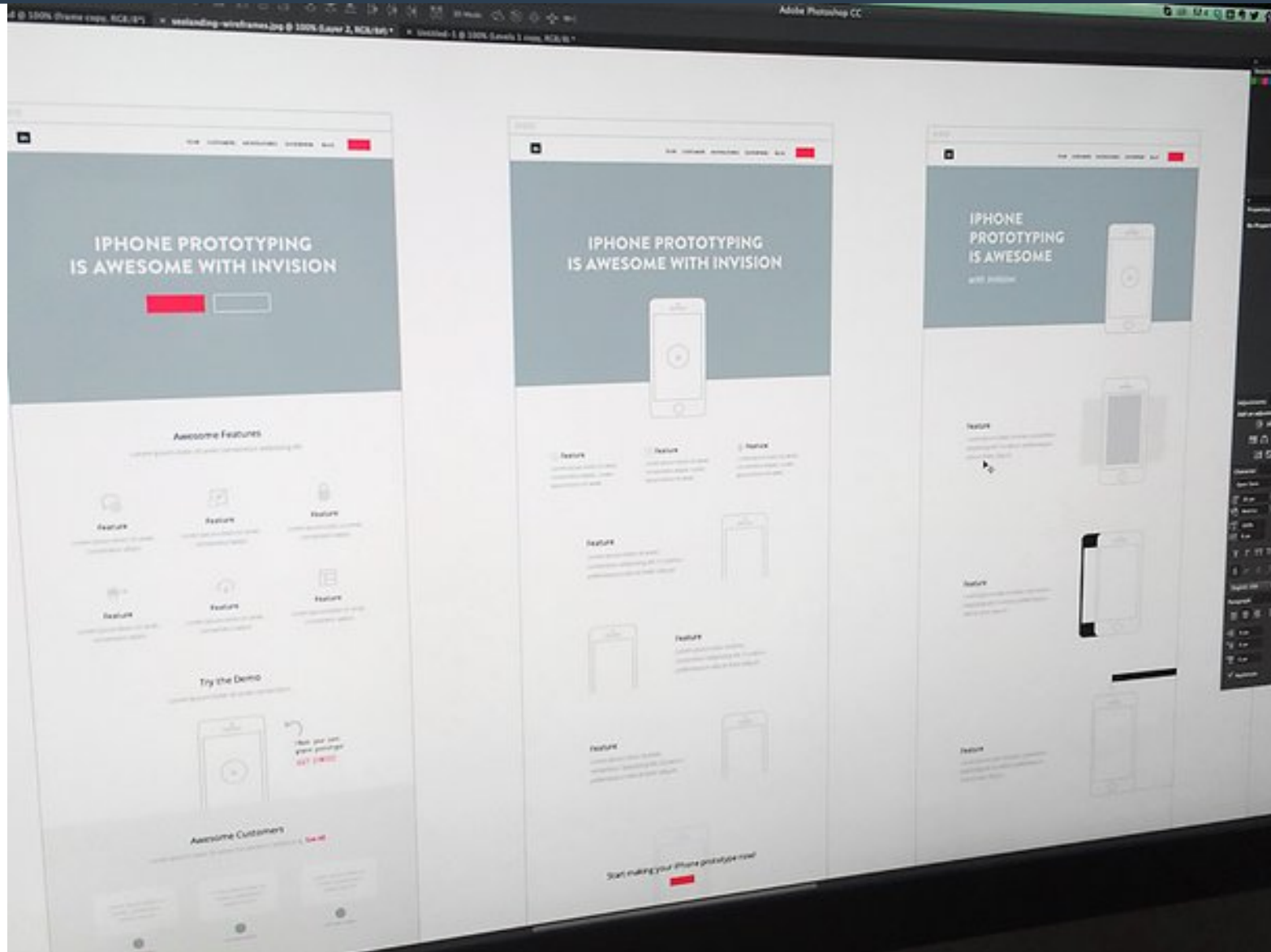
# Prototype moyenne fidélité



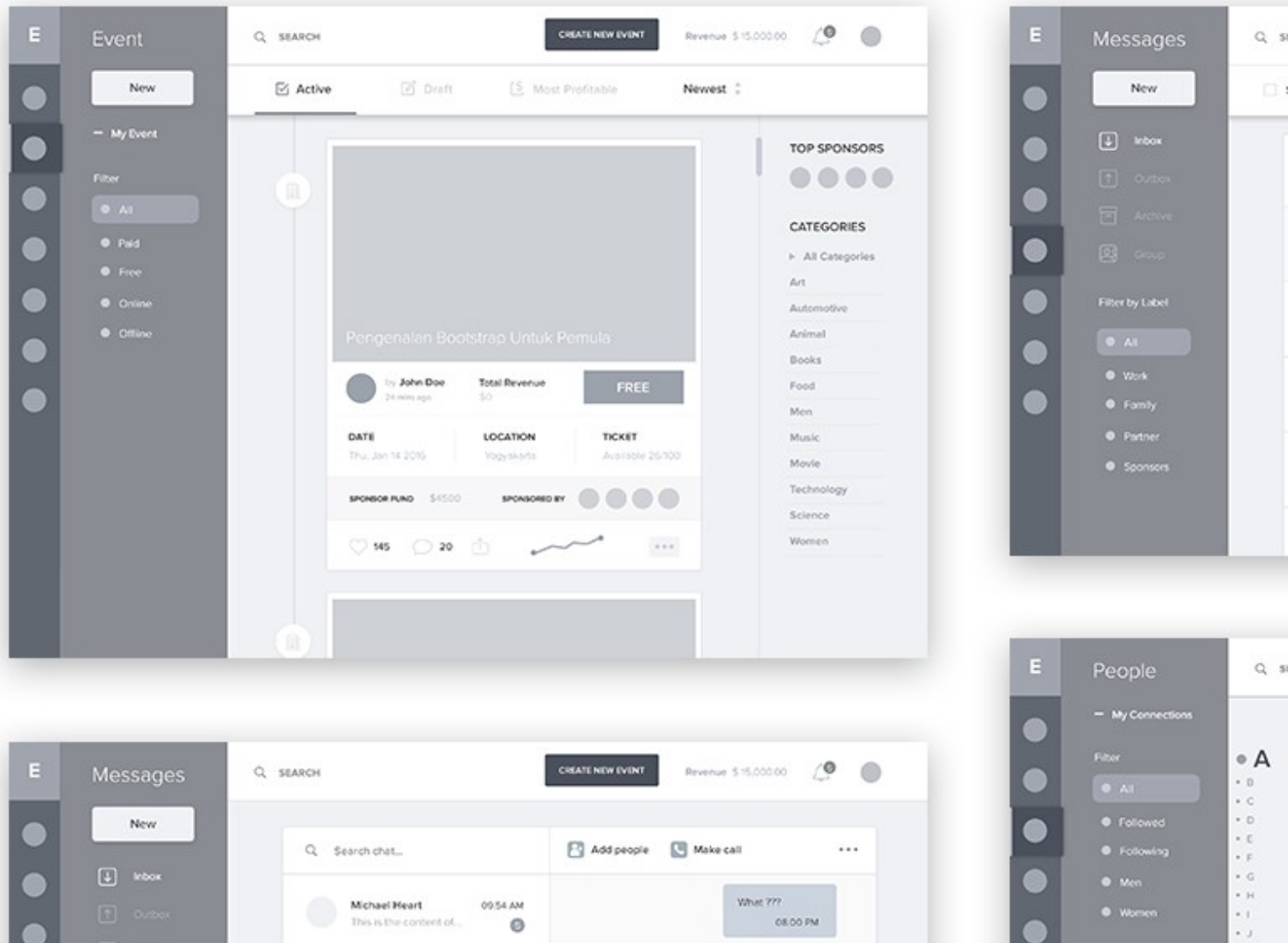
# Prototype moyenne fidélité



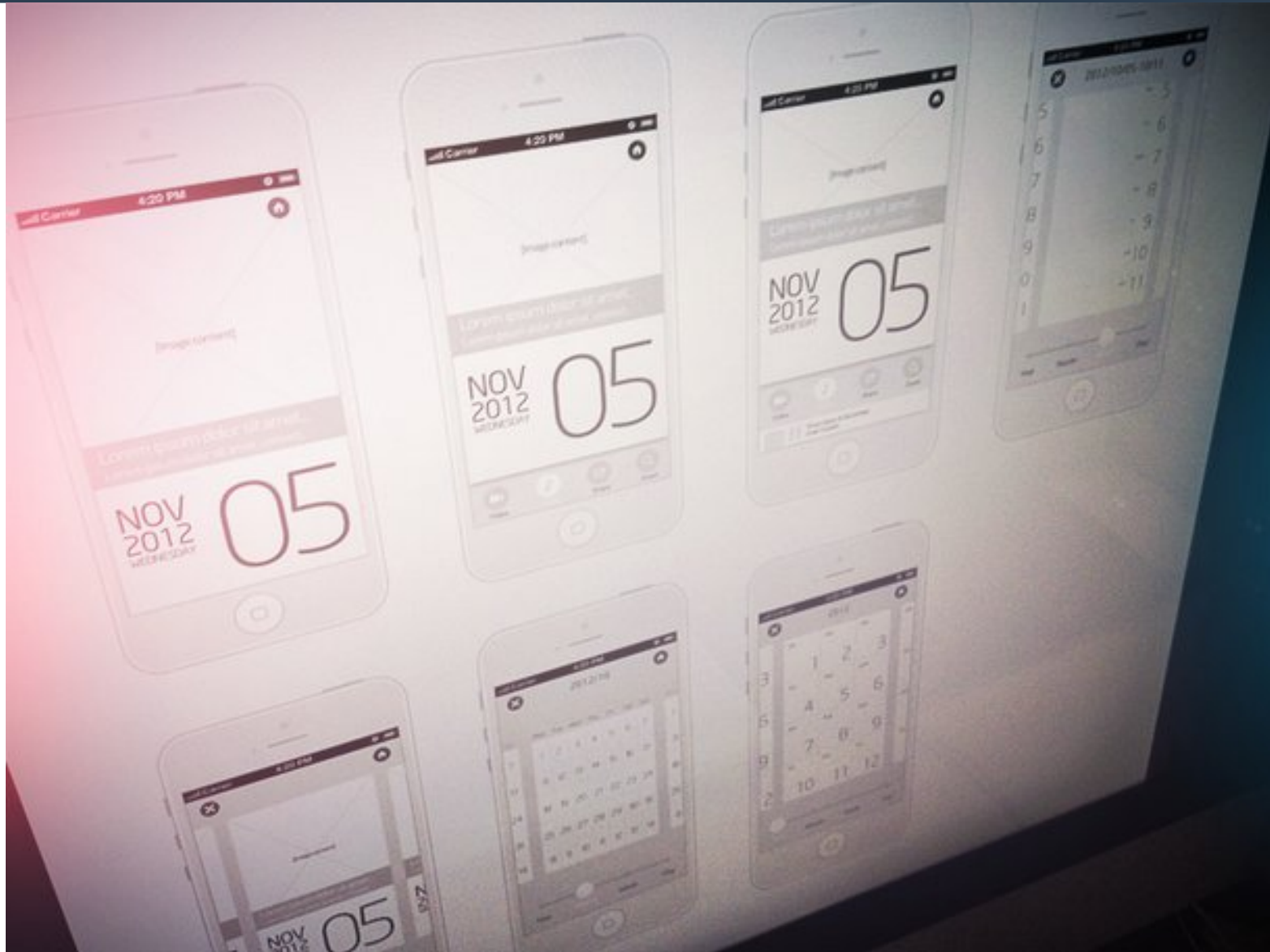
# Prototype moyenne fidélité



# Prototype moyenne fidélité



# Prototype moyenne fidélité

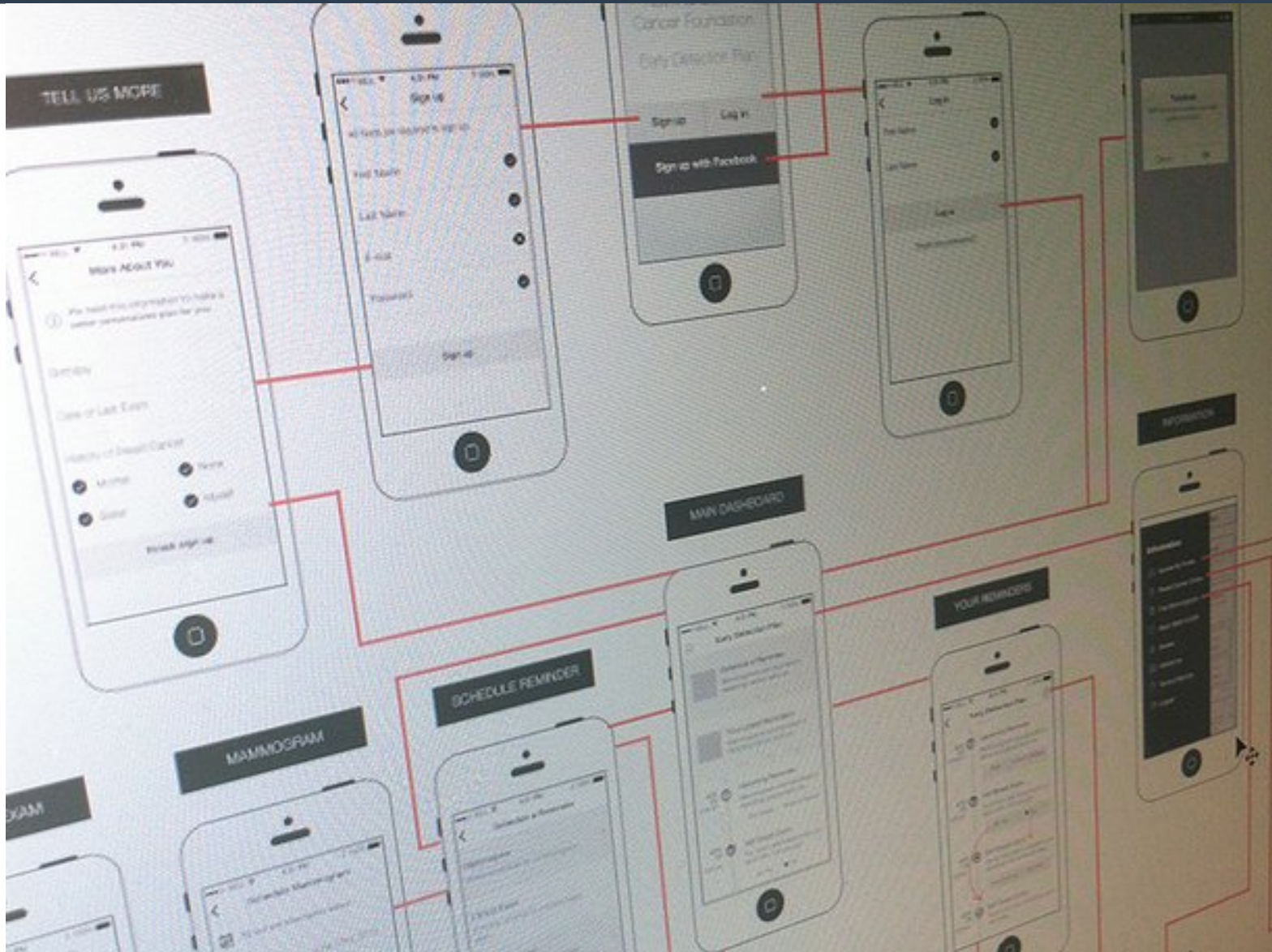




# Prototype moyenne fidélité



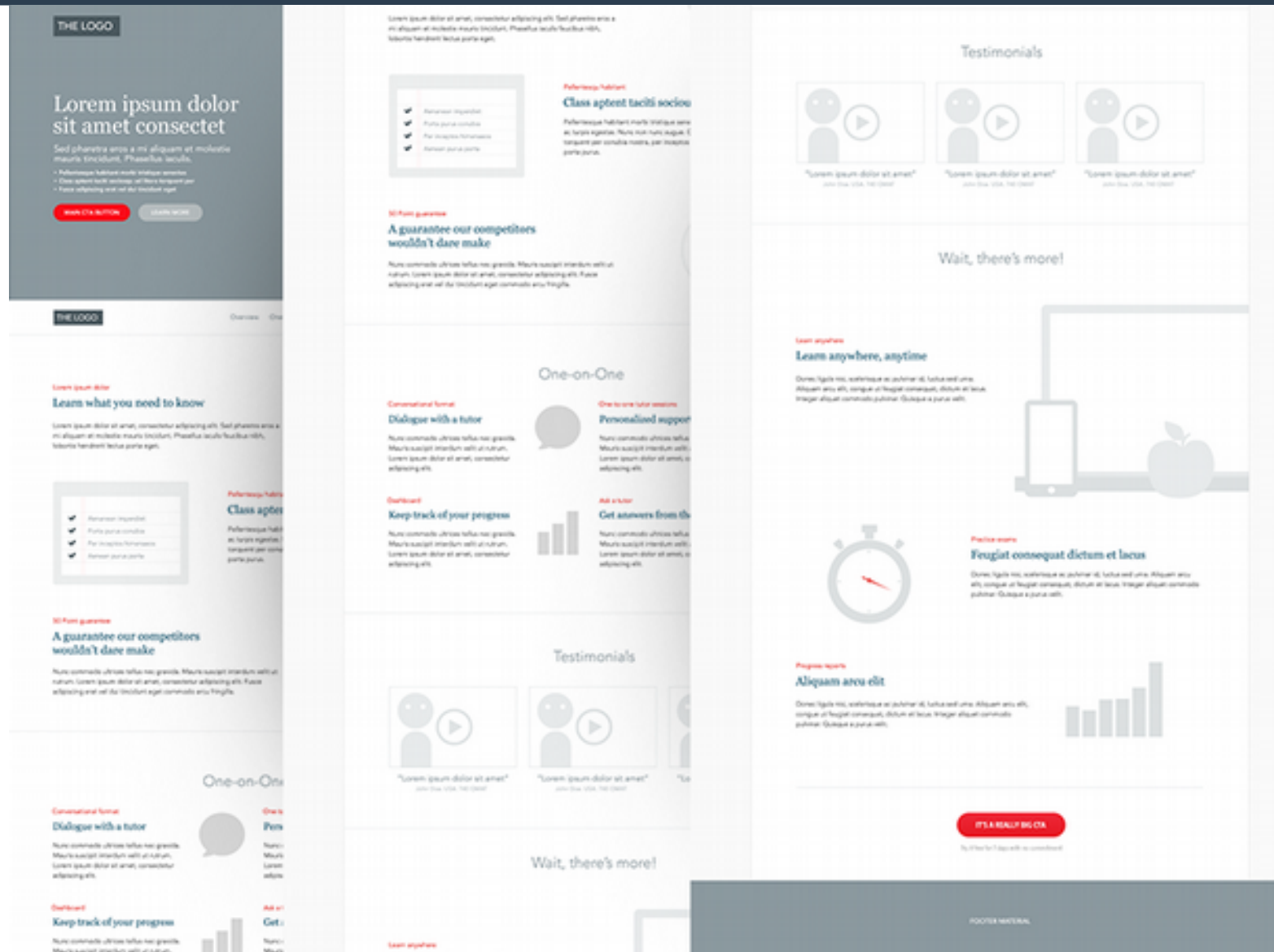
# Prototype moyenne fidélité



# Prototype moyenne fidélité



# Prototype moyenne fidélité





# Prototype moyenne fidélité

## Script / Simulation / Slideshows / Vidéo

- **Réaliser le storyboard**
  - avec n'importe quel outil de média
  - transition avec n'importe quelle action de l'utilisateur
  - vertical (avec un peu d'horizontal)
- **L'utilisateur doit suivre le scénario**
  - ça ressemble au système
  - la moindre déviation annule la simulation



# Prototype moyenne fidélité

## Interface builder

- **Avantages**

- outils et widgets classique
- mode de test : les objets réagissent comme des vrais
- très efficace pour montrer l'apparence... mais limiter à la bibliothèque
- large prototype horizontal
- besoins de programmation pour un prototypage vertical

## bootstrap

# Prototype moyenne fidélité

## Magicien d'Oz

- **Principe**

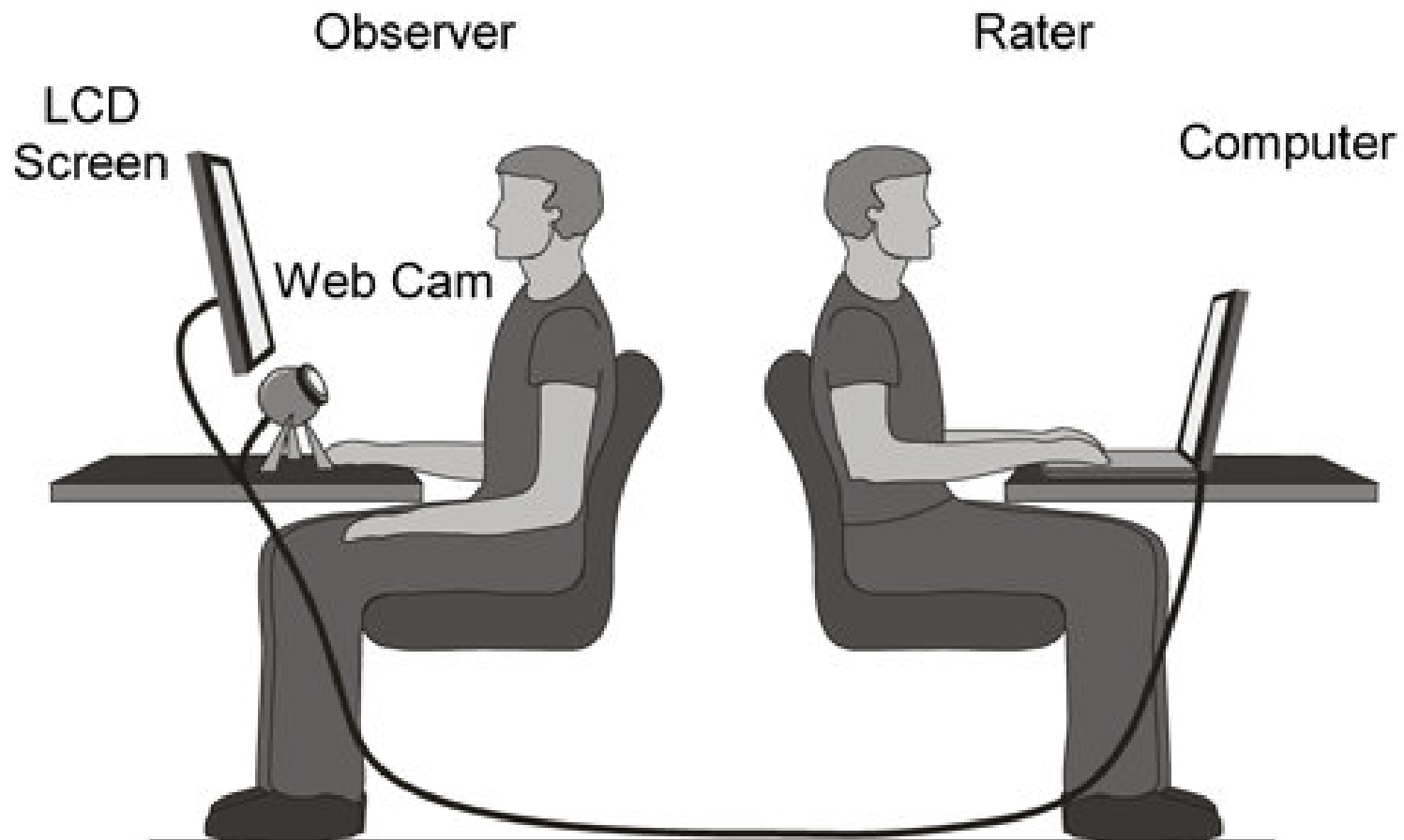
- Une personne simule l'intelligence du système et interagit avec l'utilisateur
- Avec des maquettes ou l'interface réelle
- L'utilisateur se sert de l'ordinateur comme prévu
- Le magicien (de préférence caché) interprète ce que fait l'utilisateur et réagit comme si c'était l'ordinateur qui le faisait

- **Utile pour**

- Tester des fonctionnalités très complexes (difficile à programmer)
- Tester des idées futuristes (comment l'utilisateur va réagir)

# Prototype moyenne fidélité

## Magicien d'Oz



# Prototype moyenne fidélité

## Approches d'intégration du prototype dans le produit

- **Jetable :**
  - le prototype est juste là pour résoudre un problème
  - il faut qu'il soit rapide à faire, sinon c'est trop cher
- **Incrémental :**
  - création de composants séparément
  - les composants sont testé séparément pour ajouté au système final
- **Évolutif (méthode Agile) :**
  - le prototype est modifié au fur et à mesure pour prendre en compte les changement de design
  - éventuellement le prototype deviendra le produit final

# Prototype haute fidélité

- **Utilisez les outils qui seront dans le produit final**
- **Le prototype ressemble fortement au produit final**
- **Les utilisateurs auront tendance à croire que le système est complet**

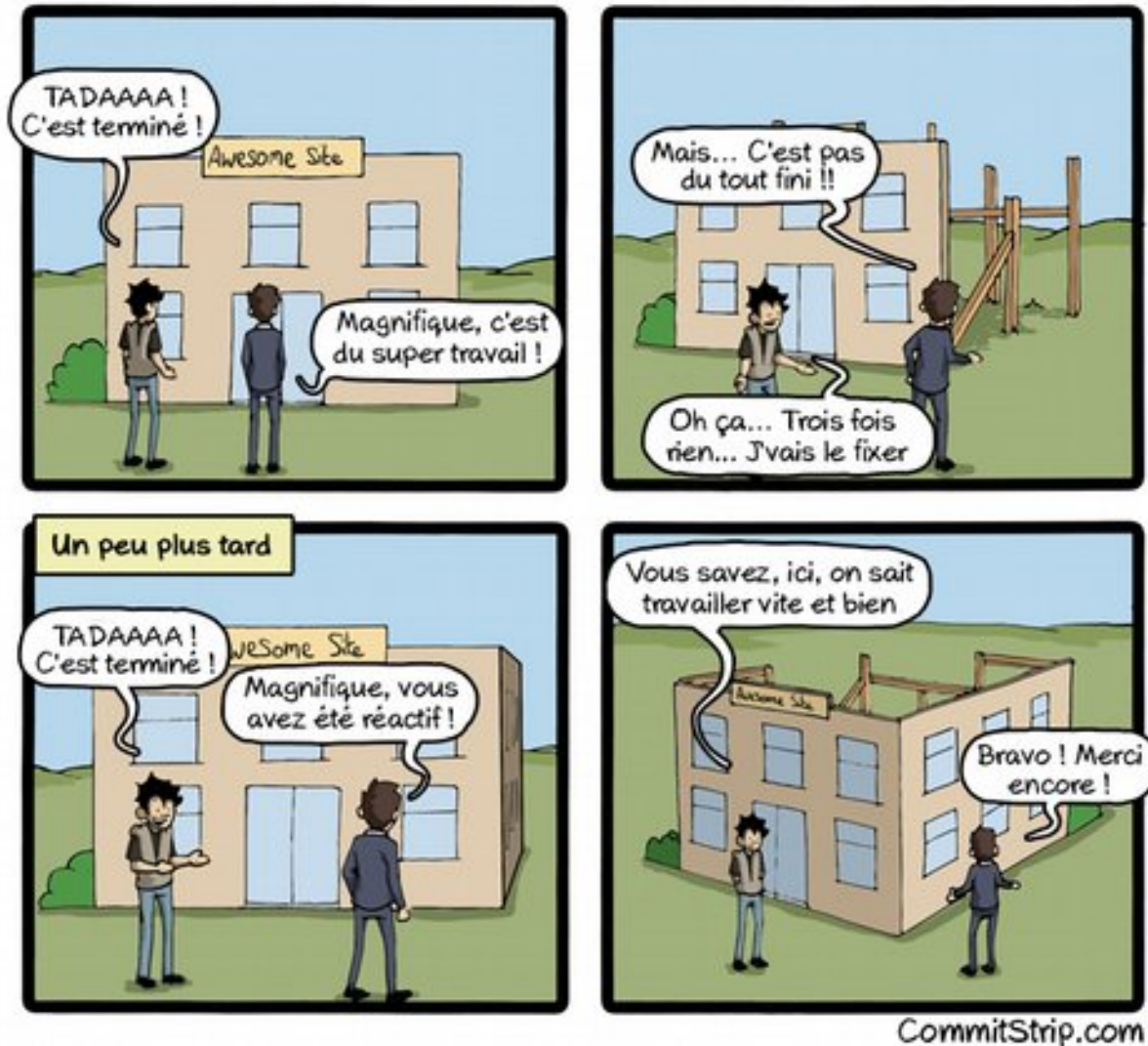


# Prototype haute fidélité

## Pourquoi faire des prototypes haute fidélité ?

- Ça paraît réelle
- Ça permet de reproduire les interactions
- Ça impressionne le client
- On peut tout montrer
- On peut montrer les détails

# Prototype haute fidélité



CommitStrip.com

# Croquis vs. Haute fidélité

- **Les prototypes haute fidélité sont problématique :**
  - les utilisateurs croient que c'est réelle
  - les utilisateurs voient les petits détails plutôt que les grands principes
  - les développeurs vont avoir le blues quand on va tout jeter...

# Croquis vs. Haute fidélité

- **Les prototypes basse fidélité évite cela :**
  - c'est facile à changer
  - les utilisateurs voient les grands principes (structures, fonctionnalités)
  - c'est jetable

**Avant de faire de la haute qualité, assuré vous que la basse est bonne**

# Prototypes

## Premières ébauches

Recherche de plusieurs représentations  
Brainstorming  
Chosir une représentation  
se concentrer sur la tâche  
Affinage de l'interface  
Évaluation heuristique

Tests d'ergonomie

Tests en situation

Tests A/B

Prototype basse fidélité

Prototype moyenne fidélité

Prototype haute fidélité

Système fonctionnel

## Travail avancé



# Prototypes

- **Prototype :**

- ça permet à l'utilisateur de réagir aux propositions d'interfaces et de proposer des changements
- croquis, maquettes papier et storyboard sont plus adaptés pour faire des choix de représentation
- pour affiner l'interface, l'utilisation d'outils avancés est plus adéquate

- **Méthode de prototypage :**

- vertical, horizontal, scénario...
- croquis, maquettes papier et storyboard
- utilisation de bibliothèque d'outils et de widget
- simulations programmées
- magicien d'Oz

# Conclusion

« C'est en forgeant que l'on devient forgeron »

Les prototypes sont facile et rapide à forger