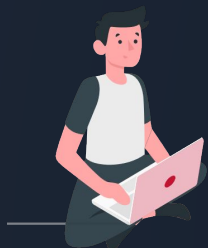


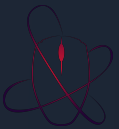


Développement Web / Mobile



Objectifs de la formation

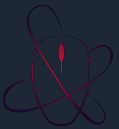
- ✓ Les étapes d'une étude en dev. web / mobile
- ✓ Identifier les fonctionnalités et leur difficulté
- ✓ Découvrir les architectures types
- ✓ Culture : connaître au moins de nom différentes technologies





Les grandes étapes

1. Identifier le besoin
2. Mockup / Wireframing
3. Design
4. Frontend
5. Backend
6. Déploiement



1 - Identifier le besoin

- ✓ Site web, application mobile
- ✓ Lister les fonctionnalités
- ✓ Identifier celles qui sont importantes et celles qui sont “bonus”
- ✓ Identifier ce qui prendra du temps à développer
- ✓ Ne rien oublier



1 - Identifier le besoin : Exemples de fonctionnalités

- ✓ Authentification
- ✓ Plusieurs type de comptes
 - ↔ *Administrateur, marchand, utilisateur*
- ✓ Back Office : Gestion de ressources
 - ↔ *Gestion de stock*
- ✓ Création de posts
- ✓ Possibilité d'upload des fichier ★
 - ↔ *Leboncoin : upload de photos de l'article*
- ✓ Streaming de vidéo ★
- ✓ “ Se connecter avec Google ” ★
 - *OAuth*
- ✓ Utilisation d'API externe ★
 - ↔ *Récupérer la météo*
 - ↔ *Géolocalisation*
 - ↔ *Google Maps*
- ✓ Animations ★
- ✓ Paiement en ligne ★★★
 - *Intégrer Stripe, Paypal*
- ✓ Temps réel ★★★
 - *Web Sockets, Socket.io*
 - ↔ *Multijoueur*
- ✓ Envoi de mail, SMS, Notification Push ★★★



2 - Mockup / Wireframing

- ✓ Lister chaque page
- ✓ Positionner les éléments
- ✓ Établir les relations
- ✓ Penser l'expérience utilisateur (UX)
- ✓ Allers / Retours avec le Client pour valider

→ Balsamiq



3 - Design

- ✓ Appliquer la charte graphique
- ✓ Faire le design du site (UI)

→ Figma

↔ [awwards.com](https://www.awwards.com)

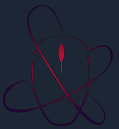
↔ [landingfolio.com](https://www.landingfolio.com)

↔ [pageflows.com](https://www.pageflows.com)



4 - Frontend : Web

- ✓ Statique ou dynamique
- ✓ Interactions avec le Backend ?
- ✓ Framework
 - CSS : Bootstrap, Tailwind.css
 - JS : React.js, Vue.js, Angular
- ✓ Adapté ordinateur, mobile, responsive
- ✓ Single Page App (SPA) ★
 - Pas de chargement entre les pages
- ✓ Progressive Web App (PWA) ★★★
 - S'installe sur les appareils
 - Permet d'utiliser des fonctionnalités avancées



4 - Frontend : Mobile

- ✓ Utilisation d'API Native ?
- ✓ Framework
 - React Native, Flutter
 - Limité par rapport à une app native
 - Plus facile à appréhender
 - Même code pour toutes les plateformes
- ✓ Application native ★★
 - Code spécifique à chaque plateforme



5 - Backend

- ✓ Technologie
 - PHP, Ruby, Node.js
- ✓ Server Side Rendering (SSR)
 - PHP, Next.js, Nuxt.js
- ✓ Base de donnée
 - MySQL, PostgreSQL, MongoDB
- ✓ BaaS: Backend as a Service
 - Firebase, Amplify, Supabase



6 - Déploiement et hébergement

- ✓ Données statique, BDD, logique
 - ✓ Gestion de la charge : nombre d'utilisateurs
 - ✓ Gestion des certificats SSL (https)
 - ✓ Gestion de la stabilité
-
- ✓ PaaS ★
 - Heroku, Google App Engine
 - ✓ IaaS : Infrastructure as a Service ★★
 - OVH, Google Cloud, AWS, Azure



