

PEC - B3 Alt

Matières et intervenants en lien avec le PEC

Matières	Coachs
Backend en PHP (POO)	Edenn Touitou
Création de framework SASS (HTML & CSS)	Edouard Sombié
Management de projets	Karl MARQUES BERNARDO

Présentation du projet

Description fonctionnelle

Doc 2 Wheels est une société spécialisée dans le service de réparation de deux-roues (motos, scooters) à domicile, au bureau ou dans la rue. L'objectif du projet est de développer une application qui facilitera la vie des motards et des scootéristes dans les mégapoles. Le service répondra à des problématiques telles que la pénurie de disponibilité des ateliers officiels, des coûts de main-d'œuvre élevés et la nécessité d'une prise en charge rapide des urgences.

Fonctionnalités principales /15

En tant que client

- Inscription et Connexion :
 - Création de compte avec vérification de l'identité.
 - Profil utilisateur
 - Authentification via e-mail avec validation du mail.
- Demande de Réparation :
 - Interface pour sélectionner le type de service (réparation, entretien, dépannage d'urgence).
 - Choix du lieu d'intervention (domicile, bureau, adresse personnalisée).
 - Sélection de la plage horaire disponible
- Estimation des Coûts et Devis :
 - Calcul automatique d'un devis préliminaire en fonction de la nature de la panne, du modèle du véhicule, et de la localisation.
- L'application conserve un historique des demandes de réparations
- L'utilisateur est notifié (mail) lorsqu'une demande de réparations est acceptée.
- Avis et Retour d'Expérience :

- Système d'évaluation pour noter le service du technicien.
- Possibilité de laisser un commentaire pour chaque intervention.

En tant qu'administrateur

- Suivi et Analyse des Performances (Dashboard) :
 - nombre d'interventions, satisfaction client, taux de retour, etc.
 - Suivi des revenus générés par période et par technicien.
- Interface pour créer et gérer l'ensemble des entités :
 - utilisateurs
 - demande de réparations
 - techniciens
 - services
- Gestion des Commandes :
 - Validation d'une demande de réparation (assignation d'un technicien à une demande)
 - Attribution automatique ou manuelle des techniciens en fonction de leur disponibilité et de leur localisation.
 - Visualisation des interventions en cours et des disponibilités.
- Support et Gestion des Réclamations/Avis client :
 - Interface pour répondre aux plaintes ou demandes des clients.
 - Modération et suivi des avis laissés par les clients.
 - Système d'alerte en cas de retour négatif pour une intervention.

Fonctionnalités supplémentaires /5

- **Notifications** : Sur l'ensemble du workflow (email, sms, whatsapp, ...)
 - **Suivi en temps réel du camion-atelier** : géolocalisation en direct du camion pour l'arrivée du mécanicien.
 - **Programme de fidélité** : système de points ou de récompenses pour les clients réguliers.
 - **Paiement en ligne** : intégrer un système de paiement en ligne type Stripe
 - **Recommandations de tutoriels personnalisés** : suggestions basées sur l'historique d'interventions.
- Optimisation des tournées des camions : calcul intelligent pour réduire le temps d'attente des clients.
- **Accès technicien**
 - **Support multi-langues** : possibilité d'utiliser l'application dans différentes langues.
 - **Architecture logicielle** : Utilisation de design patterns avancés tels que Query Builder, ActiveRecord, Model, Factory, Event Listener, Observer, ...
 - **Etc** (sous validation des intervenants ou coach)

Contraintes par matière

- Backend en PHP
 -
 - Création framework SASS
 -
-

Rendus attendus

- Repository (github)

Vous devez TOUS commit sur votre repository (*commits obligatoirement auto-signés*)

Un [README.md](#)

- noms de chacun et vos comptes github associés (pseudo)
 - qui a fait quoi (listing des fonctionnalités du projet)
 - etc...
 - Procédure de lancement du projet en local
- Mise en production

Votre projet doit être obligatoirement mis en production pour le valider, le jour de la soutenance et ce durant 2 semaines au moins. Cette mise en production peut être sur n'importe quel type de serveur (VPS / dédié / heroku / raspberry / etc).

- Backend en PHP
 - Repository Git
 - Création framework SASS
 - Repository Git
 - Sécurité
 - Aucune faille de sécurité autorisée (malus)
-

Notation Matière vs PEC

Matières

- **Contrôle.s continu.s** : Réalisé.s pendant le cours
- **Partiel** : Analyse technique sur le rendu

PEC (Projet d'Etude de cas)

- **Contrôle continue** : Suivi de projet sur les différentes dates de suivi (avec votre responsable pédagogique)
- **Partiel** : Soutenance du PEC (rendu global)

Barème sur la note de suivi du PEC (Projet d'Etude de cas)

Pour chaque session de suivi, le barème ci-dessous est à remplir pour chaque équipe par le responsable pédagogique. La note de suivi est une note de groupe, mais peut-être individualisée.

- Barème des suivis par groupe /20

Compétence à évaluer	Echec (0 pt)	Médiocre (1 pt)	Passable (2 pts)	Bien (3 pts)	Excellent (4 pt)
Présentation du projet (pitch rapide à l'oral du projet)					
Qualité du travail produit					
Respect du niveau de l'avancement du projet demandé					
Organisation du projet (attribution des tâches / etc)					
Cohésion du groupe					

Organisation des soutenances

Pour soutenir votre projet, un ordre de passage sera réalisé, vous aurez 30 minutes par groupe découpées comme suit :

- 10 minutes de présentation,
- 20 minutes de questions/réponses

Template de présentation de votre projet (première partie) :

- Présentation rapide de votre concept (à l'oral, 1 minutes)
- Présentation des grandes fonctionnalités du projet (2 minutes max)
- Démonstration de votre projet à travers les différents parcours utilisateurs (7 minutes)

Attention : Aucune démonstration de code n'est demandée. Tenue professionnelle obligatoire. Projet lancé et prêt à être présenté (code source et maquette ouvert en fond pour les questions/réponses).
