**Cahier des Charges : Rafistoleur**

**Présentation du Projet**

Rafistoleur est une application mobile et web visant à faciliter la tâche des couturiers ambulants (Toclo) ainsi que des clients en Côte d'Ivoire. L'application permet aux clients de localiser les couturiers proches, de prendre des rendez-vous, de suivre leurs commandes en temps réel et de payer en ligne. Les couturiers peuvent gérer leurs rendez-vous, suivre leurs clients et recevoir des paiements.

**Objectifs du Projet**

1. Faciliter l'accès aux services de couture pour les clients.
2. Optimiser la gestion des rendez-vous et des déplacements pour les couturiers.
3. Offrir une plateforme sécurisée pour les transactions financières.
4. Améliorer la satisfaction des clients grâce à des évaluations et des avis.

**Fonctionnalités Principales**

**1. Inscription et Authentification**

* **Couturiers et Clients :** Formulaire d'inscription avec vérification par email ou SMS. Authentification par email, numéro de téléphone ou réseaux sociaux.

**2. Profil Utilisateur**

* **Couturiers :** Profil détaillé avec photo, expérience, spécialités, et tarifs.
* **Clients :** Profil avec photo, historique des commandes, et informations de contact.

**3. Géolocalisation**

* **Localisation en Temps Réel :** Affichage des couturiers proches des clients sur une carte.
* **Itinéraire :** Indications pour rejoindre le lieu de rendez-vous.

**4. Prise de Rendez-vous**

* **Calendrier :** Affichage des disponibilités des couturiers.
* **Réservation :** Sélection de la date, de l'heure et du lieu de rendez-vous.
* **Confirmation et Rappel :** Notifications de confirmation et rappels de rendez-vous.

**5. Suivi des Commandes**

* **Suivi en Temps Réel :** Suivi de l'arrivée du couturier et de l'état de la commande.
* **Historique :** Accès à l'historique des commandes passées.

**6. Estimation des Coûts**

* **Calcul Automatique :** Estimation du coût des services en fonction des besoins du client (type de réparation, matériaux nécessaires, etc.).

**7. Paiement en Ligne**

* **Options de Paiement :** Paiement par carte bancaire, mobile money, ou en espèces.
* **Sécurité :** Transactions sécurisées avec cryptage des données.

**8. Évaluation et Avis**

* **Notation :** Les clients peuvent évaluer les couturiers après chaque service.
* **Commentaires :** Les clients peuvent laisser des avis détaillés.

**9. Notifications**

* **Mises à Jour :** Notifications pour les confirmations de rendez-vous, rappels, et mises à jour de l'état de la commande.
* **Promotions :** Notifications pour les offres spéciales et promotions.

**Contraintes**

* **Budget :** Limité, à définir.
* **Délais :** 6 mois pour la première version.
* **Accessibilité :** Application mobile pour Android et iOS, et une version web.
* **Sécurité :** Protéger les données personnelles et financières des utilisateurs.

**Public Cible**

* **Couturiers Ambulants :** En Côte d'Ivoire, cherchant à augmenter leur visibilité et à optimiser leur emploi du temps.
* **Clients :** Cherchant des services de couture rapides, pratiques et fiables.

**Analyse des Besoins**

**Analyse Fonctionnelle :**

* **Utilisateurs :** Différents rôles (couturiers, clients, administrateurs).
* **Scénarios d'Utilisation :** Description des interactions entre les utilisateurs et le système (ex : prise de rendez-vous, paiement, évaluation).

**Analyse Technique :**

* **Technologies Front-end :** Flet.
* **Technologies Back-end :** Django
* **Base de Données :** MySQL
* **API :** RESTful API pour la communication entre le front-end et le back-end.
* **Hébergement :** AWS, Heroku, ou DigitalOcean.

**Conception**

**Architecture :**

* **MVC :** Modèle-Vue-Contrôleur pour séparer la logique métier, la présentation et les données.
* **MVT :** Modèle-Vue-Template

**Maquettes (Wireframes) :**

* **Pages Principales :** Page d'accueil, profil utilisateur, carte de géolocalisation, calendrier de rendez-vous, suivi des commandes, page de paiement, page d'évaluation.

**Modèle de Données :**

* **Schéma ERD :** Diagramme Entité-Relation pour la base de données, incluant les tables Utilisateurs, Couturiers, Rendez-vous, Commandes, Paiements, et Évaluations.

**Développement**

**Front-end :**

* **Interface Utilisateur :** Développement des pages et des composants UI.
* **Responsivité :** Assurer que l'application soit accessible sur mobile et desktop.

**Back-end :**

* **API :** Développement des endpoints pour la gestion des utilisateurs, des rendez-vous, des paiements, etc.
* **Authentification :** Implémentation de JWT ou OAuth pour la gestion des sessions.

**Base de Données :**

* **Modélisation :** Création des tables et relations dans la base de données choisie.
* **Optimisation :** Indexation et optimisation des requêtes.

**Tests**

**Tests Unitaires :**

* **Composants Individuels :** Tester chaque composant et fonction de manière isolée.

**Tests d'Intégration :**

* **Flux Complets :** Tester les interactions entre les différents composants et services.

**Tests Utilisateur :**

* **Feedback Réel :** Faire tester l'application par des utilisateurs réels pour recueillir des feedbacks et améliorer l'UX.

**Déploiement**

**Choix de l'Hébergeur :**

* **Options :** AWS, Heroku, DigitalOcean.
* **Configuration :** Mise en place des environnements de production et de développement.

**Déploiement :**

* **Processus CI/CD :** Implémentation de l'intégration continue et du déploiement continu pour automatiser les mises à jour.

**Maintenance et Evolution**

**Maintenance :**

* **Correction des Bugs :** Surveillance et correction des bugs reportés par les utilisateurs.
* **Mises à Jour :** Ajout de nouvelles fonctionnalités et améliorations en fonction des retours des utilisateurs.

**Evolution :**

* **Nouvelles Idées :** Intégration des nouvelles idées et fonctionnalités au fur et à mesure qu'elles émergent.