

1. Front-end de l'interface utilisateur

- **Étude des besoins** : Recueillir les besoins des utilisateurs pour s'assurer que l'interface couvre tous les aspects de la gestion de factures.
- **Conception de l'interface** : Créer des maquettes pour le design des pages principales (page d'accueil, tableau de bord, gestion des clients, gestion des factures, et rapport financier).
- **Développement de l'interface avec JavaFX** : Utiliser JavaFX pour construire les pages, en se concentrant sur l'ergonomie. Les éléments à développer incluent :
 - Page d'accueil avec vue d'ensemble des activités
 - Formulaires pour l'ajout de clients et de factures
 - Tableau de bord affichant les factures impayées et les échéances
- **Navigation et fluidité** : S'assurer que la navigation entre les pages est intuitive et rapide.
- **Test utilisateur** : Effectuer des tests pour recueillir des retours sur l'expérience utilisateur, puis apporter des ajustements si nécessaire.

2. Back-end et logique de facturation

- **Définition des fonctionnalités clés** : Élaborer les cas d'utilisation des fonctionnalités comme la création et l'édition de factures, la gestion des statuts (payé, impayé, en attente), et les notifications.
- **Création de classes et méthodes** :
 - Créer des classes pour la gestion des clients et des factures (par exemple : `Client`, `Facture`, `Paiement`).
 - Développer des méthodes pour créer, mettre à jour, et supprimer des factures.
 - Implémenter une logique de calcul pour le total des factures, y compris la TVA et les éventuels rabais.
- **Intégration des notifications** : Développer un système qui envoie des rappels automatiques pour les factures impayées, en utilisant un service de messagerie (email ou SMS).
- **Validation des données** : Mettre en place des validations pour s'assurer que les informations saisies (dates, montants, etc.) respectent les normes de facturation.
- **Tests unitaires et intégration** : Créer des tests pour vérifier la fiabilité de chaque fonctionnalité et éviter les erreurs dans la gestion des données.

3. Base de données et génération de rapports

- **Conception de la base de données** : Créer le schéma de base de données qui inclut des tables pour :
 - Clients : informations de contact, identifiants uniques
 - Factures : montant, date de création, statut, client associé
 - Paiements : date de paiement, mode de paiement, montant
- **Implémentation de la base de données (MySQL)** : Développer le modèle de base de données en utilisant MySQL ou une autre base de données SQL.
- **Accès et manipulation des données** :

- Créer des requêtes SQL pour insérer, mettre à jour, et récupérer des données en fonction des besoins de l'application.
 - Mettre en place des fonctions de recherche et de filtrage pour accéder aux factures spécifiques.
- **Génération de rapports :**
 - Concevoir des rapports financiers, comme le total des factures payées et impayées par mois ou par client.
 - Intégrer une fonctionnalité d'exportation pour télécharger les rapports en formats comme PDF ou Excel.
- **Optimisation des requêtes et sécurité :** Veiller à la sécurité des données avec des permissions et des contrôles d'accès. Optimiser les requêtes pour de meilleures performances.
- **Tests de la base de données :** Vérifier la cohérence et l'intégrité des données avec des tests pour éviter toute perte ou corruption de données.