**Cahier des Charges Fonctionnel (CDCF)**

**1. Introduction et Contexte**

**a. Buts de l’application**

L’application vise à fournir un outil d’analyse et de gestion des données clients pour :

* Optimiser les stratégies de fidélisation et d’acquisition.
* Prédire la valeur à vie des clients (Customer Lifetime Value - CLV).
* Identifier les anomalies dans les transactions et comportements clients.
* Analyser les tendances saisonnières et préférences d’achat.
* Améliorer la segmentation des clients pour des campagnes ciblées.

**b. Public visé**

* **Équipes Marketing** : Pour l’optimisation des campagnes.
* **Gestionnaires de Projets** : Pour le suivi et la planification des actions.
* **Analystes de Données** : Pour l’exploitation et l’interprétation des données.
* **Direction Générale** : Pour les prises de décisions stratégiques.

**c. Portée (périodes de planning…)**

* **Phase 1** : Développement du prototype (3 mois).
* **Phase 2** : Tests et validation (1 mois).
* **Phase 3** : Déploiement et formation des utilisateurs (1 mois).
* **Phase 4** : Suivi et maintenance continue.

**d. Références (site avec maquettes statiques)**

* **Site principal** : URL des maquettes statiques (à définir).
* **Références techniques** : Documentation des API et bases de données utilisées.
* **Références légales** : Normes RGPD pour la gestion des données sensibles.

**e. Liste de diffusion (qui est concerné ?)**

* **Chef de Projet** : Responsable de la coordination globale.
* **Développeurs** : Implémentation des fonctionnalités.
* **Testeurs QA** : Validation des fonctionnalités et tests de performance.
* **Clients Pilotes** : Utilisateurs pour les tests bêta.

**f. Planning (cahiers des charges et livraisons)**

* **Livraison 1** : Prototype fonctionnel - Fin du mois 3.
* **Livraison 2** : Version testée et validée - Fin du mois 4.
* **Livraison 3** : Version finale en production - Fin du mois 5.

**g. Contraintes (Performances, Implémentations, Lois, Audits…)**

* **Performances** :
  + Temps de réponse < 2 secondes pour les requêtes standards.
  + Capacité à traiter jusqu’à 10 000 transactions simultanées.
* **Implémentations** :
  + Backend en Python avec framework Flask/Django.
  + Frontend en React avec Tailwind CSS.
* **Lois et Audits** :
  + Conformité RGPD sur la gestion et la sauvegarde des données.
  + Audit de sécurité avant mise en production.
  + Stockage sécurisé des données sensibles avec chiffrement AES-256.

**2. Description Générale**

**a. Perspectives**

L’application est conçue pour offrir un cadre d’analyse de données fiable et évolutif afin d’aider les entreprises à comprendre et prédire le comportement des clients. Elle vise à fournir des rapports dynamiques et des visualisations interactives pour simplifier la prise de décisions stratégiques.

**b. Principales Fonctionnalités**

* **Analyse prédictive** : Prévision des comportements et valeurs des clients.
* **Segmentation client** : Regroupement des clients selon leurs caractéristiques.
* **Détection d’anomalies** : Identification des comportements inhabituels.
* **Analyse de tendances** : Décomposition saisonnière pour identifier les périodes clés.
* **Tableaux de bord interactifs** : Visualisations graphiques des données en temps réel.
* **Exportation des rapports** : Génération de rapports en formats CSV, PDF et Excel.

**c. Caractéristiques de l’utilisateur**

* **Niveau technique** :
  + Utilisateurs métier avec peu de compétences techniques.
  + Analystes de données ayant besoin de fonctionnalités avancées.
* **Rôles** :
  + Administrateurs (gestion des utilisateurs et des accès).
  + Utilisateurs standards (lecture et analyse des rapports).

**d. Environnement Technique (quels OS)**

* **Systèmes d’exploitation supportés** :
  + Windows 10 et versions ultérieures.
  + macOS (Mojave et versions plus récentes).
  + Linux (Ubuntu 20.04 LTS et versions plus récentes).
* **Navigateurs supportés** :
  + Google Chrome (version récente).
  + Mozilla Firefox.
  + Microsoft Edge.

**e. Dépendances**

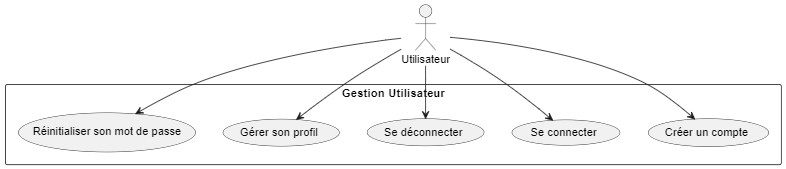
* **Langages et frameworks** :
  + Python (3.8+).
  + Flask ou Django pour le backend.
  + React.js avec Tailwind CSS pour le frontend.
* **Base de données** :
  + PostgreSQL ou MySQL.
* **Bibliothèques analytiques** :
  + Scikit-learn, Pandas, NumPy, Matplotlib.
  + Lifelines et Prophet pour l’analyse de séries temporelles.

**f. Design**

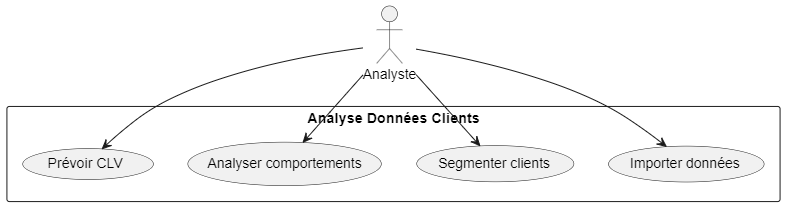
* **Interface utilisateur** :
  + Conception centrée sur l’utilisateur avec une navigation intuitive.
  + Tableaux de bord interactifs et filtres dynamiques.
* **Charte graphique** :
  + Couleurs sobres et modernes alignées avec l’identité visuelle de l’entreprise.
  + Icônes et graphiques clairs pour faciliter l’interprétation des données.
* **Accessibilité** :
  + Prise en charge des normes WCAG pour l’accessibilité des utilisateurs.
  + Compatibilité avec les lecteurs d’écran.

**3. Description Détaillée des Fonctionnalités**

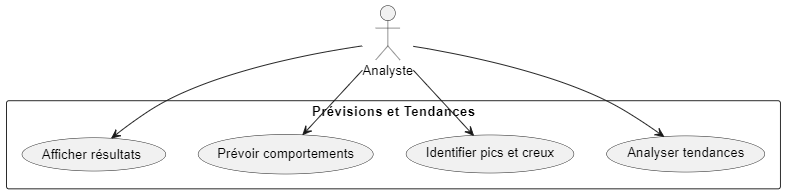
**a. Gestion des Utilisateurs**



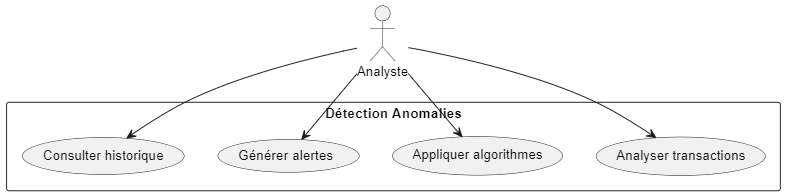
**b. Analyse des Données Clients**



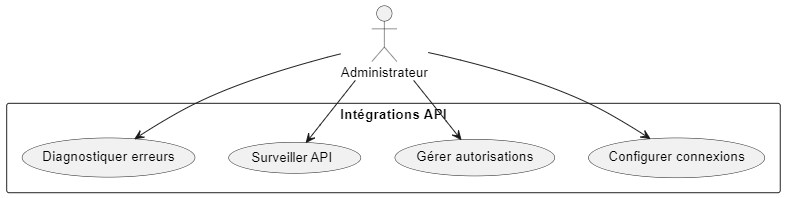
**c. Gestion des Rapports et Visualisations**



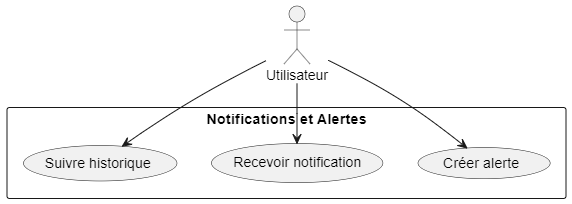
**d. Détection d’Anomalies**



**e. Gestion des Intégrations API**

****

**f. Notifications et Alertes**



**4. Maquettes**

**5. Users Stories**

****

**a. Connexion utilisateur**

* **En tant qu’utilisateur**, je veux me connecter avec mon email et mot de passe afin d’accéder à mon espace personnel.

**b. Gestion des tâches**

* **En tant qu’utilisateur**, je veux créer, modifier et supprimer des tâches pour organiser mon travail.

**c. Consultation des rapports**

* **En tant qu’analyste**, je veux consulter les rapports interactifs pour visualiser les données et identifier des tendances.

**d. Administration**

* **En tant qu’administrateur**, je veux gérer les rôles et permissions afin de sécuriser l’accès aux données.

**6. Types et Gestion des Données**

**a. Types de Données**

* **Données utilisateurs** : Informations personnelles (nom, email, rôle, permissions).
* **Données transactionnelles** : Historique des achats, montants dépensés, fréquences d’achat.
* **Données comportementales** : Interactions sur l’application, actions et clics.
* **Données analytiques** : Résultats des analyses prédictives, tendances saisonnières et segmentation client.
* **Données de performance** : Temps de réponse des requêtes et rapports d’utilisation.

**b. Gestion des Données**

* **Collecte** : Données importées via fichiers CSV, formulaires utilisateurs ou intégrations API.
* **Stockage** : Bases de données relationnelles (PostgreSQL ou MySQL) avec sauvegardes automatiques.
* **Sécurisation** : Chiffrement des données sensibles (AES-256) et contrôle des accès via des rôles.
* **Traitement** : Prétraitement des données avec Pandas et nettoyage des valeurs manquantes.
* **Analyse** : Utilisation d’algorithmes de machine learning pour la segmentation, la prévision et la détection d’anomalies.
* **Archivage et Suppression** : Archivage des données obsolètes et conformité avec les politiques de suppression RGPD.

**7. Sécurité**

**a. Authentification et Autorisation**

* Authentification multi-facteurs (MFA) pour renforcer la sécurité.
* Gestion des rôles et permissions basée sur des groupes d’utilisateurs.

**b. Chiffrement des Données**

* Chiffrement AES-256 pour les données sensibles en repos.
* Connexions HTTPS/TLS pour sécuriser les communications réseau.

**c. Sauvegarde et Récupération**

* Sauvegardes automatiques journalières avec stockage sur serveur sécurisé.
* Plans de reprise après sinistre (PRA) avec tests réguliers.

**d. Audits et Conformité**

* Journalisation des activités des utilisateurs pour audits de sécurité.
* Conformité aux normes RGPD pour la gestion et suppression des données personnelles.

**8. Fonctionnalités Optionnelles**

**a. Intégration avec des outils tiers**

* **CRM et ERP** : Connexion avec des outils de gestion client et de planification des ressources d’entreprise pour centraliser les données.
* **Marketing Automation** : Intégration avec des plateformes de marketing pour automatiser les campagnes.
* **Systèmes de paiement** : Connexion à des passerelles de paiement sécurisées pour traiter les transactions.

**b. Notifications intelligentes**

* **Emails et SMS** : Envoi d’alertes pour des événements spécifiques comme des promotions ou des rappels.
* **Alertes en temps réel** : Notifications pour les anomalies détectées dans les données.

**c. Tableau de bord personnalisable**

* **Widgets dynamiques** : Personnalisation des tableaux de bord pour afficher uniquement les informations pertinentes pour l’utilisateur.
* **Filtres avancés** : Tri et regroupement des données par critères spécifiques.

**d. Modules d’intelligence artificielle**

* **Prévisions avancées** : Utilisation d’IA pour améliorer la précision des prévisions de ventes et d’anomalies.
* **Chatbot intégré** : Assistance intelligente pour répondre aux questions fréquentes des utilisateurs.

**e. Analyse de sentiment**

* **Traitement du langage naturel (NLP)** : Analyse des avis clients pour comprendre leur satisfaction et détecter des tendances émotionnelles.

**9. Maintenance**

**a. Support technique**

* **Assistance 24/7** : Support disponible en continu par email et chat.
* **Documentation utilisateur** : Guides et tutoriels détaillés pour l’utilisation de l’application.

**b. Mises à jour**

* **Correctifs de sécurité** : Déploiement immédiat en cas de vulnérabilité détectée.
* **Améliorations fonctionnelles** : Mise à jour trimestrielle avec ajout de nouvelles fonctionnalités et optimisations.
* **Compatibilité** : Maintien des compatibilités avec les nouvelles versions des systèmes d’exploitation et navigateurs.

**c. Surveillance et performance**

* **Monitoring en temps réel** : Suivi des performances des serveurs et alertes en cas d’incidents.
* **Tests de charge réguliers** : Vérification des performances sous haute sollicitation.

**d. Sauvegardes et restauration**

* **Sauvegardes automatiques** : Planifiées quotidiennement avec conservation des versions.
* **Plan de reprise après sinistre (PRA)** : Procédure documentée pour restaurer les données et l’application en cas de défaillance.

**10. Tests**

**a. Tests fonctionnels**

* **Validation des fonctionnalités principales** : Tests des modules d’analyse, de segmentation et de détection d’anomalies.
* **Tests d’intégration** : Vérification des interactions entre les différents modules et composants.
* **Tests utilisateurs** : Simulation des scénarios réels pour vérifier l’ergonomie et la satisfaction des utilisateurs.

**b. Tests de performance**

* **Tests de charge** : Simulation d’un grand nombre d’utilisateurs simultanés.
* **Tests de stress** : Analyse de la capacité de l’application à gérer des pics d’utilisation.

**c. Tests de sécurité**

* **Tests de vulnérabilité** : Analyse des failles potentielles et évaluation des protections.
* **Tests d’intrusion (PenTests)** : Simulation d’attaques pour valider la robustesse des défenses.

**d. Tests de compatibilité**

* **Navigateurs** : Vérification sur Chrome, Firefox et Edge.
* **Systèmes d’exploitation** : Tests sur Windows, macOS et Linux.
* **Appareils mobiles** : Tests d’affichage et de performance sur smartphones et tablettes.

**e. Tests de régression**

* **Automatisation des tests** : Exécution répétée pour garantir l’absence de régressions après chaque mise à jour.

**11. Documentation**

**a. Documentation Utilisateur**

* **Manuel utilisateur** :
  + Instructions détaillées pour la navigation dans l’application.
  + Guides pour la création et la gestion des rapports.
  + Exemples d’utilisation des fonctionnalités d’analyse prédictive et de segmentation.
  + Section FAQ pour les problèmes courants.
* **Tutoriels vidéo** :
  + Présentation pas à pas des principales fonctionnalités.
  + Formation à l’utilisation des tableaux de bord et des filtres.
* **Support en ligne** :
  + Chatbot intégré pour répondre aux questions fréquentes.
  + Accès à une base de connaissances en ligne.

**b. Documentation Technique**

* **Architecture logicielle** :
  + Diagrammes UML décrivant les composants et leurs interactions.
  + Documentation des API avec exemples de requêtes et réponses.
* **Code source** :
  + Documentation intégrée au code avec des commentaires détaillés.
  + Instructions pour configurer l’environnement de développement.
* **Base de données** :
  + Diagrammes des schémas relationnels et descriptions des tables.
  + Scripts SQL pour la création et la gestion des bases.
* **Sécurité** :
  + Guide sur les mécanismes de chiffrement et d’authentification.
  + Procédures de gestion des droits d’accès.

**c. Documentation de Maintenance**

* **Guide de déploiement** :
  + Étapes pour l’installation et la configuration sur serveur.
  + Instructions pour les mises à jour et correctifs.
* **Plan de reprise après sinistre (PRA)** :
  + Procédures pour restaurer les données et la configuration après un incident.
  + Tests périodiques pour valider la restauration.
* **Journal des versions** :
  + Liste des modifications et améliorations par version.
  + Historique des bugs corrigés et fonctionnalités ajoutées.

**d. Documentation de Test**

* **Rapports de tests** :
  + Résultats des tests fonctionnels, de performance et de sécurité.
  + Captures d’écran et descriptions des erreurs rencontrées.
* **Plans de test** :
  + Scénarios détaillés pour chaque fonctionnalité.
  + Critères d’acceptation et seuils de performance.
* **Automatisation des tests** :
  + Scripts pour exécuter des tests automatisés et vérifier les régressions.

**e. Documentation de Formation**

* **Modules de formation** :
  + Matériel pédagogique pour les utilisateurs finaux et administrateurs.
  + Exercices pratiques pour maîtriser l’application.
* **Sessions de formation en direct** :
  + Webinaires ou ateliers pratiques pour accompagner la prise en main.