

RAPPORT DE PROJET DE FIN D'ÉTUDES

Ingénieur d'État

Filière : Génie informatique

Année universitaire : 2024-2025

Développement d'un Système Intelligent de Rétention Client & Recommandation de Services (Big Data /IA)

Élaboré et soutenu le 15/07/2025 par :
ZAMOURI Nouhaila

Devant le Jury :

Nom et Prénom	Établissement	Qualité
M. EL BOUHDIDI Jaber	ENSI de Tanger	Encadrant pédagogique
M. GHAILANI Mohamed		Examineur

Encadrant externe :
M. FADIL Badr

PFE effectué au :

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Dédicaces

Je dédie ce rapport de stage de fin d'études, réalisé dans le cadre de l'obtention de mon diplôme, à mes parents, véritables piliers de ma vie. Leur amour inconditionnel, leur patience, leurs sacrifices quotidiens et leur soutien indéfectible ont été les fondements de mon parcours académique et personnel. Grâce à leur présence constante, à leurs encouragements dans les moments de doute, et à la force qu'ils m'ont transmise, j'ai pu avancer avec confiance et sérénité. Ils ont toujours cru en moi, même quand je doutais de mes propres capacités, et pour cela, je leur serai éternellement reconnaissant.

À ma famille proche, frères, sœurs et tous ceux qui, de près ou de loin, m'ont accompagné dans cette aventure, je tiens à adresser mes plus sincères remerciements. Vos mots, votre patience et votre bienveillance m'ont été d'un grand soutien tout au long de cette étape cruciale.

Je souhaite également remercier mes amis les plus chers, pour leur présence fidèle, leurs conseils avisés, et leur capacité à m'écouter, à m'encourager et à croire en moi. Vous avez su rendre cette période plus légère, plus humaine et plus riche.

Enfin, je n'oublie pas toutes les personnes rencontrées dans le cadre de mon stage, ainsi que les enseignants et encadrants qui m'ont guidé tout au long de ma formation. Chacun, à sa manière, a contribué à ma progression et à l'aboutissement de ce projet.

Ce rapport est plus qu'un travail académique, il est le reflet de l'effort collectif, de la confiance reçue, et du chemin parcouru grâce à vous tous.

Merci du fond du cœur.

Remerciement

Je remercie Allah, Le Miséricordieux, Le Tout-Puissant, de m'avoir accordé la force, la patience et la persévérance nécessaires pour mener à bien ce projet de fin d'études, réalisé dans le cadre de l'obtention de mon diplôme.

Je tiens à exprimer ma profonde gratitude à **M. CHAKIR Jihad**, mon encadrant académique, pour son suivi rigoureux, ses conseils avisés et sa disponibilité tout au long de cette aventure.

Mes remerciements sincères vont également à **M. FADIL Badr**, mon encadrant professionnel au sein de l'entreprise **HARDTEC**, pour son accompagnement, sa confiance et son soutien tout au long de la réalisation de ce projet.

Je tiens aussi à remercier **M. EL BOUHDIDI Jaber**, encadrant pédagogique, pour son implication précieuse, sa bienveillance et l'attention particulière qu'elle a portée à mon évolution.

Je remercie chaleureusement l'ensemble de **l'équipe de HARDTEC**, pour leur accueil, leur disponibilité et l'environnement professionnel enrichissant qu'ils m'ont offert.

Ma reconnaissance s'étend également à tous mes enseignants de **l'École des Nouvelles Sciences et de l'Ingénierie de Tanger**, pour la qualité de leur enseignement et leur accompagnement durant tout mon parcours académique.

Enfin, je remercie sincèrement **les membres du jury** pour avoir accepté d'évaluer mon travail, ainsi que pour leurs remarques et recommandations constructives.

Résumé

Ce projet de fin d'études, réalisé au sein de l'entreprise **HardTec**, porte sur la conception et le développement d'un système intelligent de rétention client et de recommandation de services, en s'appuyant sur les technologies du Big Data et de l'intelligence artificielle.

L'objectif principal était d'améliorer la fidélisation client et de personnaliser les services proposés grâce à des outils analytiques avancés. Un pipeline Big Data a été mis en place avec Apache Spark pour le traitement et l'enrichissement des données issues du **CRM**, de la facturation et du support.

Un modèle prédictif de churn a été implémenté avec les algorithmes **XGBoost** et **Random Forest**, tandis qu'un moteur de recommandation hybride a été développé à l'aide de Python, **Pandas**, et **Scikit-learn**, en combinant filtrage collaboratif et approche basée sur le contenu.

L'intégration technique repose sur une **API REST** développée avec **FastAPI**, connectée au CRM existant (Laravel), et les résultats sont visualisés via un dashboard interactif dans **Power BI**. Une attention particulière a été accordée au respect des normes **RGPD** et à l'explicabilité des modèles à l'aide des outils SHAP et LIME.

Ce projet m'a permis de mobiliser concrètement mes compétences en data science, en développement web et en intégration de systèmes, tout en contribuant à la transformation numérique d'une entreprise marocaine innovante.

Mots-clés :

Projet de Fin d'Études, Big Data, Intelligence Artificielle, Apache Spark, Hadoop, XGBoost, Recommandation, CRM, Power BI, RGPD, Data Science, Rétention Client.

Abstract

This final year project, carried out within the company HardTec, focuses on the design and development of an intelligent customer retention and service recommendation system, leveraging Big Data and Artificial Intelligence technologies.

The main objective was to improve customer retention and personalize the services offered through advanced analytical tools. A Big Data pipeline was implemented using Apache Spark to process and enrich data from the CRM, billing, and support systems.

A churn prediction model was built using XGBoost and Random Forest algorithms, while a hybrid recommendation engine was developed with Python, Pandas, and Scikit-learn, combining collaborative filtering and content-based approaches.

The technical integration was achieved through a REST API developed with FastAPI, connected to the existing Laravel-based CRM, and the results were visualized using an interactive dashboard in Power BI. Special attention was given to GDPR compliance and model explainability using tools such as SHAP and LIME.

This project allowed me to concretely apply my skills in data science, web development, and system integration, while contributing to the digital transformation of an innovative Moroccan company.

Keywords:

Final Year Project, Big Data, Artificial Intelligence, Apache Spark, Hadoop, XGBoost, Recommendation, CRM, Power BI, GDPR, Data Science, Customer Retention.

Avant-propos

Élève ingénieur :

ZAMOURI Nouhaila

Intitulé du sujet :

Conception et développement d'un système intelligent de rétention client et de recommandation de services basé sur les technologies Big Data et Intelligence Artificielle.

Établissement d'accueil :

HARDTEC

Encadrant Pédagogique :

M. EL BOUHDIDI Jaber

Encadrant Technique :

M. FADIL Badr

Période de projet :

Du 03 Mars au 30 juin 2025

Cadre du Stage :

Projet de Fin d'Études présenté en vue de l'obtention de diplôme d'ingénieur d'état en génie informatique.

Listes des sigles et acronyme

<u>Acronyme</u>	<u>Désignation</u>
API	Application Programming Interface
XGBoost	Extreme Gradient Boosting
Power BI	Business Analytics Service from Microsoft
CRM	Customer Relationship Management
AI	Artificial Intelligence
KPI	Key Performance Indicator
RGPD	Extreme Gradient Boosting
BD	Base Données

Tableau 1 : Listes des sigles et acronyme

Liste des figures

Figure 1 : Logo HARDTEC	16
Figure 2 : Cycle Agile.....	24
Figure 3 : Diagramme de GANTT.....	27
Figure 4 : Spécification des besoins fonctionnels liés à la solution BI.....	38
Figure 5 Architecture fonctionnelle globale du système prédictif de churn et de recommandation.....	39
Figure 6 : Schéma conceptuel des entités principales	41
Figure 7 : Résultat de l'évaluation du modèle.....	42
Figure 8 : Tableau de bord Power BI Visualisation test des résultats.....	43
Figure 9 : Design du dashboard développé avec Next.js (avant intégration réelle des données)	45
Figure 10 : Architecture MVVM du projet	47
Figure 11 : Structure du backend	48
Figure 12 : Structure du frontend développé avec Next.js	49
Figure 13 : Terminal du modèle Python	51
Figure 14 : Visualisation Power BI des résultats de churn.....	52
Figure 15 : Terminal de démarrage du frontend Next.js	52
Figure 16 : Interface principale du dashboard – prédictions de churn clients	53
Figure 17 : Résumé statistique des résultats de churn	54
Figure 18 : Visualisation graphique des résultats – scores de churn	54
Figure 19 : Customer Churn Data – Résultats détaillés par client	54
Figure 20 : Personnalisation des colonnes (customize columns)	55
Figure 21 : Zone d'alerte – Suivi des clients à risque.....	55
Figure 22 : Email de recommandation – Offre ciblée envoyée au client	56
Figure 23 : Conteneurs Docker – Backend et Frontend exécutés via Docker Desktop	57

Liste des tableaux

Tableau 1 : Sigles et abréviations	8
Tableau 2 : Contraintes identifiées lors du projet	21
Tableau 3 : Technologies mobilisées et rôle dans le système prédictif	36
Tableau 4 : Tableau récapitulatif des étapes fonctionnelles principales	40
Tableau 5 : Étapes techniques du pipeline de modélisation prédictive	42

Table des matières

Dédicaces	3
Remerciement.....	4
Résumé	4
Mots-clés :.....	5
Abstract.....	6
Avant-propos.....	7
Sigles et abréviations	8
Liste des figures	9
Liste des Tableaux	10
Table des Matières	11
Introduction Générale	14
Chapitre 1 : Présentation de l'entreprise d'accueil et Contexte Général du Projet.....	16
Introduction.....	17
1. Présentation de l'organisme d'accueil	17
1.1. Présentation de HardTec.....	17
1.2. Mission de l'entreprise	17
1.3. Activités et domaines d'expertise	17
1.4. Services et valeurs.....	17
2. Cadre général du projet	18
2.1. Présentation générale du projet	18
2.2. Situation actuelle	18
2.3. Problématiques identifiées	18
2.4. Besoins identifiés pour la solution.....	19
2.5. Objectifs du projet	20
2.6. Contraintes du projet.....	21
2.7. Solution envisagée	22
Conclusion.....	22
Chapitre 2 : Gestion de projet et technologies.....	23
Introduction.....	24
1. Conduite du projet	16
1.1. Méthodologie Agile adaptée	18
1.1.1 Méthode Agile simplifiée.....	24
1.1.2 Avantage de la Méthode Agile	25

1.1.3 Diagramme de GANTT	25
2. Outils et Technologies utilisés	27
2.1 Outils de développement et de collaboration	28
2.2 Technologies Big Data et Intelligence Artificielle	30
2.3 Backend et Intégration	31
2.4 Frontend et Visualisation	32
2.5 Source des données	32
3. Les langages utilisés	33
Conclusion	34
Chapitre 3 : Architecture et conception technique du système prédictif	35
Introduction.....	36
1. Analyse des besoins	37
1.1 Objectifs fonctionnels.....	37
1.2 Objectifs techniques	37
1.3 Besoins liés à la visualisation et à la BI.....	37
2. Architecture fonctionnelle du système	38
3. Analyse fonctionnelle	39
3.1 Flux principal.....	39
4. Schéma conceptuel des entités principales.....	40
5. Pipeline technique Big Data	41
5.1 Étapes techniques (PySpark & Scikit-learn).....	41
6. Visualisation analytique avec Power BI.....	43
7. Composants intelligents intégrés : recommandations & alertes	44
8. Prototype de l'interface – Design initial du Dashboard	44
Conclusion	45
Chapitre 4 : Mise en œuvre et Réalisation.....	45
Introduction	46

Conclusion	57
Conclusion Générale.....	58
Références	59

Introduction Générale

Ce rapport présente les travaux réalisés dans le cadre de mon Projet de Fin d'Études au sein de l'entreprise HardTec, spécialisée dans les Technologies de l'Information et de la Communication. Le projet vise à développer un système intelligent de rétention client et de recommandation de services, en exploitant des technologies Big Data et d'intelligence artificielle (IA).

L'objectif principal est de prédire les risques de désengagement des clients grâce à des modèles de machine learning (XGBoost, Random Forest) et de proposer des recommandations personnalisées via un moteur hybride conçu en Python. Les données sont traitées avec Apache Spark, les résultats intégrés au CRM Laravel via FastAPI, et visualisés dans Power BI.

Ce rapport est structuré en quatre chapitres :

Le premier chapitre présente le contexte général du projet, l'entreprise d'accueil, la problématique et les objectifs poursuivis.

Le deuxième chapitre détaille la gestion du projet : la méthodologie suivie, la planification et les outils mobilisés.

Le troisième chapitre aborde l'analyse fonctionnelle et la conception du système, en mettant en lumière les choix techniques réalisés.

Le quatrième chapitre décrit la mise en œuvre technique, les développements réalisés, les résultats obtenus et les visualisations produites.

Ce projet m'a permis de consolider mes compétences en data science, IA, intégration web et visualisation de données, tout en contribuant à un enjeu stratégique pour l'entreprise.

Chapitre 1 :

Présentation de l'entreprise d'accueil et Contexte Général du Projet

➤ Introduction

Ce chapitre présente le cadre du projet de fin d'études réalisé chez HardTec, entreprise marocaine spécialisée dans les TIC. Le projet vise à développer un système intelligent de rétention client et de recommandation de services, en exploitant des technologies telles que le Big Data et l'IA. Il répond aux limites des approches classiques de fidélisation en proposant une solution automatisée et personnalisée. Le chapitre expose également la problématique, les objectifs, les contraintes et le cahier des charges du projet.

1. Présentation de l'organisme d'accueil :

1.1. Présentation de HardTec :

Fondée en 2013 et basée à Casablanca, HardTec est une entreprise marocaine spécialisée dans les Technologies de l'Information et de la Communication (TIC). Elle se distingue par une offre diversifiée de services technologiques allant du développement d'applications web et mobiles, à l'intégration de systèmes, en passant par la cybersécurité, la conception de sites web, et l'analyse de données.



Figure 1 : Logo HARDTEC

1.2. Mission de l'entreprise :

Créer de la valeur et stimuler la croissance en proposant des solutions et services innovants dans le domaine des nouvelles technologies de l'information et de la communication (NTIC). HardTec s'engage à accompagner l'évolution, l'efficacité et la performance de ses clients, en cultivant une culture d'entreprise fondée sur le talent, l'intégrité et l'esprit d'équipe.

1.3. Activités et domaines d'expertise :

L'entreprise HardTec propose un large éventail de prestations techniques et de solutions digitales personnalisées. Ses domaines d'activité couvrent :

- **Le développement logiciel (applications web, mobiles, solutions métiers)**
- **L'intégration de systèmes d'information**
- **La cybersécurité (sécurisation des données et des infrastructures)**
- **Le design UI/UX**
- **Community management (dans une optique de fidélisation)**
- **Le déploiement de solutions IoT sur mesure**

Ces services sont adaptés aux besoins d'un large éventail de secteurs : agriculture, industrie, santé, logistique, administration publique, etc.

1.4. Services et valeurs :

HardTec fonde son développement sur des valeurs humaines et professionnelles fortes, qui guident ses équipes au quotidien et renforcent sa culture d'innovation :

- **Innovation continue** : intégration permanente de nouvelles technologies pour anticiper les besoins des clients et créer des solutions sur mesure.
- **Excellence technique** : exigence de qualité dans chaque service ou solution délivrée, grâce à une équipe hautement qualifiée.
- **Intégrité** : transparence, éthique et responsabilité dans les relations internes comme externes.
- **Esprit d'équipe** : collaboration, partage des connaissances et intelligence collective pour atteindre les objectifs communs.
- **Flexibilité** : capacité à s'adapter rapidement aux évolutions du marché et aux spécificités des clients.

2. Cadre général du projet :

2.1. Présentation générale du projet :

Ce projet s'inscrit dans une volonté stratégique de renforcement de la fidélisation client et de valorisation des données commerciales. Il vise à anticiper les risques de départ des clients (churn), à proposer des services personnalisés, et à outiller l'équipe commerciale d'alertes et recommandations proactives, afin de dynamiser les ventes et limiter les pertes de revenus récurrents.

2.2. Situation actuelle :

Avant le démarrage du projet, aucun système prédictif ou automatisé n'était mis en place chez HardTec pour évaluer le comportement des clients ou leur engagement à long terme. Les méthodes utilisées étaient essentiellement manuelles et réactives, ce qui engendrait plusieurs limitations :

- **Attrition client non maîtrisée** : Aucun modèle ne permettait de détecter les clients à risque de résiliation.
- **Réactions tardives aux signaux de mécontentement** : Les baisses d'activité ou retards de paiement n'étaient pas systématiquement traités de manière préventive.
- **Opportunités de ventes croisées manquées** : Les services complémentaires (maintenance, SEO, design, etc.) n'étaient pas automatiquement proposés aux clients.
- **Segmentation rudimentaire** : Les actions commerciales reposaient sur des critères de base (ex. secteur d'activité), sans analyse comportementale.
- **Décision commerciale non pilotée par les données** : Absence d'indicateurs prédictifs ou d'outils d'aide à la décision.
- **Manque de centralisation des données** : Les données clients étaient éparpillées entre le CRM, les factures et les échanges support.

2.3. Problématiques identifiées :

L'analyse du fonctionnement actuel chez **HardTec** a mis en lumière plusieurs faiblesses dans la gestion de la relation client et l'exploitation des données commerciales :

- **Attrition client non maîtrisée :** Aucun dispositif analytique ne permet d'identifier les clients susceptibles de se désengager. Les signaux de mécontentement (comme les retards de paiement ou la diminution des demandes) sont traités trop tardivement.
- **Perte de revenus récurrents :** En l'absence de stratégie de rétention anticipée, l'entreprise subit des pertes liées au départ de clients existants, tout en supportant des coûts d'acquisition élevés pour en attirer de nouveaux.
- **Manque d'exploitation des ventes croisées :** Les clients se voient proposer un service isolé (par exemple, développement web), sans recommandations de compléments pertinents (comme la maintenance ou le design 3D), ce qui freine le potentiel de croissance du chiffre d'affaires.
- **Gestion manuelle et peu optimisée :** La segmentation client repose sur des critères basiques (ex. : secteur d'activité), sans réelle prise en compte du comportement utilisateur. Les décisions commerciales sont majoritairement réactives, non guidées par la donnée.

2.4. Besoins identifiés pour la solution :

L'analyse de la situation actuelle chez HardTec a permis de faire émerger des besoins clairs pour concevoir une solution efficace et durable. Ces besoins concernent à la fois l'automatisation des processus, la valorisation des données, l'amélioration du pilotage commercial et la conformité aux normes en vigueur.

- **Automatiser la détection et la recommandation :**
 - Détecter automatiquement les clients présentant un risque de désengagement, à partir de leurs comportements récents.
 - Proposer des services complémentaires pertinents, adaptés à chaque profil client.
- **Centraliser et exploiter les données :**
 - Regrouper l'ensemble des données clients dans une base unifiée pour faciliter leur exploitation.
 - Offrir une vision globale et structurée des interactions clients, accessible via une interface claire.

- Renforcer la fiabilité des analyses par la suppression des doublons et des données incohérentes.
- **Faciliter le suivi et la prise de décision :**
- Mettre en place des tableaux de bord dynamiques permettant une visualisation instantanée des risques, des opportunités et des indicateurs clés.
- Actualiser automatiquement les prédictions et les recommandations selon les nouvelles données.
- **Garantir la sécurité et la conformité :**
- Assurer la protection des données personnelles selon les exigences du RGPD.
- Mettre en œuvre des contrôles d'accès rigoureux et des sauvegardes automatiques.

2.5. Objectifs du projet :

Le projet a été défini autour de plusieurs objectifs stratégiques et techniques, visant à améliorer l'efficacité commerciale et à optimiser la gestion de la relation client :

- ✓ **Prédire le départ des clients** à l'aide d'un modèle d'intelligence artificielle fiable et interprétable.
- ✓ **Générer des recommandations personnalisées** afin d'augmenter la satisfaction et la valeur client.
- ✓ **Automatiser certaines prises de décision**, en intégrant les résultats dans le CRM existant.
- ✓ **Appuyer les actions commerciales par des données exploitables**, via une interface visuelle simple et efficace.
- ✓ **S'assurer du respect des normes de protection des données**, sans compromettre la performance du système.

2.6. Contraintes du projet :

La mise en œuvre du projet s'est accompagnée de plusieurs contraintes qu'il a fallu prendre en compte tout au long de son développement :

Type de contrainte	Description
- Contraintes techniques :	Intégration de données hétérogènes issues de systèmes existants (CRM, facturation), volume important de données à traiter, nécessité d'assurer la compatibilité avec l'architecture Laravel existante.
- Contraintes temporelles :	Le projet devait être conçu, développé et testé dans un laps de temps limité à la durée du stage (quatre mois), ce qui a imposé une planification rigoureuse et la priorisation des tâches essentielles.
- Contraintes organisationnelles :	Accès restreint à certaines données sensibles pour des raisons de confidentialité, disponibilité limitée de certains collaborateurs clés pour
- Contraintes réglementaires :	Respect des exigences du RGPD en matière de traitement de données personnelles, nécessité d'assurer l'anonymisation et la traçabilité des données utilisées.

Tableau 2 : Contraintes identifiées lors du projet

2.7. Solution envisagée :

- **Un modèle prédictif de churn**, entraîné à l'aide des algorithmes XGBoost et Random Forest, sur un jeu de données (Telco Customer Churn – Kaggle) prétraité avec PySpark. Ce modèle a permis d'atteindre un score AUC d'environ 0.83, indiquant une performance satisfaisante.
- **Un moteur de recommandation hybride**, basé à la fois sur le filtrage collaboratif et l'analyse de contenu, implémenté en Python avec les bibliothèques Scikit-learn et Pandas, pour proposer des services complémentaires adaptés au profil de chaque client.
- **Un pipeline de traitement Big Data**, développé avec Apache Spark, pour le nettoyage, l'enrichissement et l'agrégation des données provenant du CRM, de la facturation et du support client.
- **Une API REST, développée avec FastAPI**, assurant l'intégration fluide des résultats issus des modules IA au sein du CRM utilisé par l'entreprise.
- **Un dashboard interactif**, réalisé avec Power BI, permettant à l'équipe commerciale de visualiser en temps réel les profils à risque, les recommandations, ainsi que divers indicateurs (taux de churn, segmentation, services utilisés, etc.).
- **Un dashboard web frontend** développé avec Next.js et Tailwind CSS, interfacé via l'API REST, permettant de consulter dynamiquement les KPIs, les alertes commerciales et les recommandations sous forme de tableaux interactifs avec filtres avancés.

➤ Conclusion

Ce chapitre a permis de poser les fondations du projet en présentant le contexte stratégique de HardTec, les limites du système actuel, ainsi que les besoins métiers. Il a défini les objectifs prioritaires, les solutions techniques retenues, et les fonctionnalités développées ces éléments constituent un cadre clair qui guidera les étapes méthodologiques détaillées dans les chapitres suivants.

Chapitre 2 :

Gestion de projet Et technologies

➤ Introduction

Ce chapitre présente la gestion du projet de développement d'un système intelligent de rétention client et de recommandation de services réalisé chez **HardTec**. Il décrit la méthodologie suivie, le déroulement des étapes clés, la planification, les outils utilisés, ainsi que les technologies choisies pour assurer la réussite du projet dans les délais impartis.

1. Conduite du projet :

1.1 Méthodologie Agile adaptée :

La conduite du projet chez HardTec s'est appuyée sur une méthode de travail agile adaptée au contexte d'un projet individuel. Étant donné la nature exploratoire et technique du système développé (modélisation IA, traitement de données, intégration web), l'approche Agile a été privilégiée pour sa souplesse, son adaptabilité et sa capacité à livrer des résultats fonctionnels de manière itérative.

1.1.1 Méthode Agile simplifiée :

Plutôt que de suivre un modèle linéaire, le projet a été divisé en itérations courtes (sprints), chacune focalisée sur un composant technique (par exemple : modèle prédictif, API, visualisation). À la fin de chaque sprint, un livrable était produit, évalué, et ajusté si nécessaire. Ce mode de travail a permis une montée en complexité progressive tout en assurant une amélioration continue des résultats.

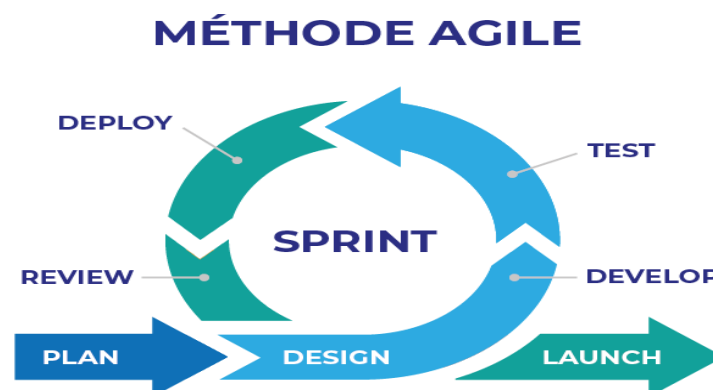


Figure 2 : Cycle agile

Le processus Agile s'est articulé autour des étapes suivantes :

- **Planification initiale** : Définition des besoins, identification des composants clés (prédiction, recommandation, visualisation, intégration).
- **Découpage du projet en sprints** : Chaque sprint (1 à 2 semaines) était dédié à une fonctionnalité livrable.
- **Développement incrémental** : Conception, codage et test de chaque module avant de passer au suivant.
- **Revues régulières** : Évaluation des résultats à chaque étape pour améliorer ou corriger si nécessaire.
- **Documentation et intégration** : À chaque livrable stable, la documentation technique était mise à jour, et le module intégré au système global.

1.1.2 Avantage de la Méthode Agile :

- **Souplesse** : Les modèles IA a été ajustés en fonction des résultats intermédiaires (churn score AUC, recommandations).
- **Vision claire** : Un diagramme de Gantt a été utilisé pour planifier, structurer et suivre l'avancement des tâches tout au long du projet, facilitant la gestion du temps et des priorités.
- **Livrables concrets à chaque étape** : modèles entraînés, fichiers de résultats (CSV), API opérationnelle, Dashboard Power BI, puis Next.js.
- **Amélioration continue** : Les tests ont permis d'améliorer progressivement les performances, l'ergonomie et la sécurité du système.

1.1.3 Diagramme de GANTT :

La planification du projet a été établie selon une progression logique en six grandes phases, dont chacune correspond à une étape clé dans le développement du système intelligent de rétention client. Voici le détail :

✓ Phase 1 : Analyse & Planification (03 mars - 10 mars 2025) :

- Étude du contexte fonctionnel et technique de l'entreprise
- Recueil des besoins et exploration des sources de données internes (CRM, support, facturation)

- Définition des livrables techniques
- Rédaction du rapport d'analyse initial
 - ✓ **Phase 2 : Ingestion & Préparation des Données (11 mars - 28 mars 2025) :**
- Ingestion des données via PySpark
- Nettoyage et traitement des données brutes
- Feature engineering (transformation des données, création de variables explicatives)
- Préparation du jeu d'entraînement pour les modèles IA
 - ✓ **Phase 3 : Modélisation IA & Recommandation (29 mars - 19 avril 2025) :**
- Entraînement des modèles de prédiction du churn (XGBoost, Random Forest)
- Évaluation et ajustement des modèles (score AUC ≈ 0.83)
- Développement du moteur de recommandation hybride (filtrage collaboratif + content-based)
- Tests internes sur un dataset simulé (Kaggle - Telco Churn)
 - ✓ **Phase 4 : Industrialisation & Intégration (20 avril - 06 mai 2025) :**
- Mise en place du pipeline IA connecté à une API REST (FastAPI)
- Développement du dashboard frontend avec Next.js et Tailwind CSS
- Appels API via Axios pour afficher les résultats dynamiquement
 - ✓ **Phase 5 : Éthique, Validation & Livraison (07 mai - 22 mai 2025) :**
- Mise en conformité RGPD (anonymisation des données, contrôle d'accès)
- Intégration des outils d'explicabilité (SHAP, LIME)
- Préparation du livrable final : code source, documentations, rapport technique
- Présentation interne et validation par l'équipe encadrante
 - ✓ **Phase 6 : Finalisation & Préparation de la Soutenance (24 mai - 30 juin 2025) :**
- Finalisation du frontend (filtrage, dashboard interactif)
- Résolution des bugs restants et ajustements de performance
- Ajout des filtres et des pages avancées dans Next.js
- Mise à jour finale des visualisations Power BI

- Rédaction du rapport PFE complet
- Préparation à la soutenance (présentation + démonstration technique)

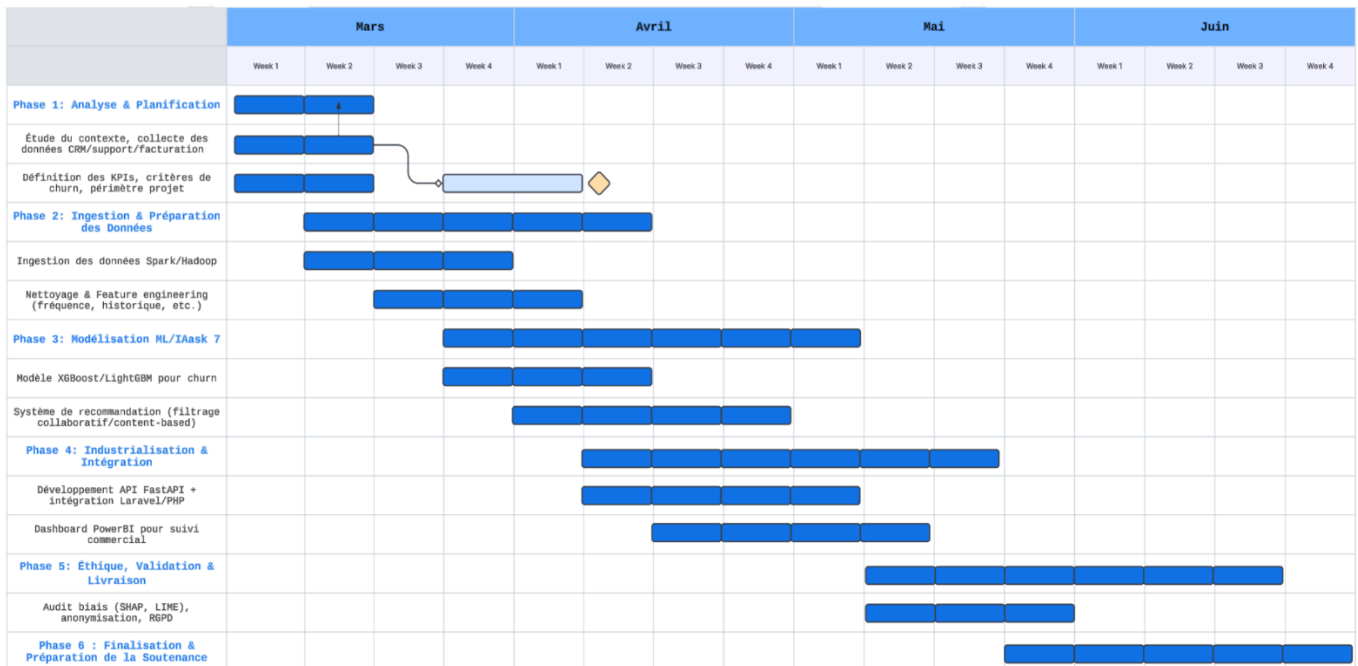


Figure 3 : Diagramme de GANTT

2. Outils et Technologies utilisés :

Le choix des technologies a été une étape déterminante dans la réussite du projet. Chaque outil a été sélectionné selon des critères précis, garantissant performance, cohérence et adaptabilité à un système intelligent de rétention client.

Les principaux critères ayant guidé cette sélection sont les suivants :

- **Pertinence fonctionnelle** : Les technologies ont été choisies en fonction de leur capacité à répondre précisément aux objectifs du projet, notamment la prédiction du churn, la génération de recommandations et l'intégration avec un CRM existant.
- **Performance et scalabilité** : L'ensemble de l'architecture devait pouvoir traiter un volume important de données tout en garantissant rapidité et évolutivité, notamment dans les phases de modélisation et d'inférence.
- **Compatibilité et intégration** : L'intégration fluide entre les différentes couches (modèle IA, API, CRM, visualisation) a nécessité l'usage de framework.

- **Maturité et communauté** : Des technologies éprouvées et bien documentées ont été privilégiées pour bénéficier d'un support actif et d'une base solide d'exemples et de bonnes pratiques.
- **Sécurité et conformité** : Enfin, la conformité au RGPD et la sécurisation des données ont guidé le choix de frameworks capables d'offrir un contrôle d'accès fin, un suivi des requêtes et un traitement sécurisé des données sensibles.

2.1. Outils de développement et de collaboration :

- **GitHub Desktop** :



GitHub Utilisé pour la gestion du code source en local. Il a permis un suivi clair des versions, une visualisation des modifications apportées, et une organisation rigoureuse du projet (modèles IA, API, visualisation). Il a facilité un travail autonome tout en assurant la traçabilité.

- **Visual Studio Code** :



Visual Studio Code (VS Code) Environnement principal de développement utilisé pour le scripting Python (modélisation IA, nettoyage de données), la création de l'API REST (FastAPI), ainsi que le développement frontend en Next.js. Des extensions comme Python et REST Client ont amélioré la productivité.

- **Anaconda** :



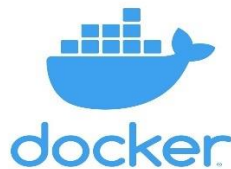
Anaconda a été utilisé comme gestionnaire d'environnements virtuels pour Python. Il a permis d'installer et d'organiser les bibliothèques nécessaires à la modélisation IA (Scikit - learn, Pandas, XGBoost, etc.), en garantissant un environnement stable et reproductible.

- **Jupyter Notebook :**



Jupyter a été utilisé pour le prototypage, l'expérimentation et la visualisation des résultats des modèles prédictifs. Il a servi de support pour les tests de modèles, l'analyse exploratoire des données, et la documentation du pipeline IA.

- **Docker :**



Bien que Docker n'ait pas été déployé en production, une image Docker a été préparée pour simuler un environnement reproductible et isolé à des fins de test. Cela illustre la capacité future du système à être conteneurisé pour des déploiements industriels.

- **Lucidchart :**



Utilisé pour la conception du diagramme de Gantt et des schémas d'architecture du système, **Lucidchart** a permis une représentation visuelle claire et professionnelle du cycle de vie du projet.

2.2. Technologies Big Data et Intelligence Artificielle :

- **Apache Spark :**



Employé pour l'ingestion, le nettoyage et la transformation des données massives. Spark a servi de base pour construire un pipeline robuste capable de traiter des données issues de différents systèmes métiers (CRM, facturation...).

- **Hadoop (HDFS local) :**



Utilisé comme couche de stockage simulée, Hadoop HDFS a permis de tester un fonctionnement distribué pour l'ingestion de données dans un environnement local.

- **Pandas & Scikit-learn- & matplotlib :**



Librairies centrales du projet : **Pandas** a été utilisé pour la manipulation des données tabulaires et la préparation des features ; **Scikit-learn** a servi à l'entraînement et à l'évaluation des modèles de machine learning (classification et recommandation) ; enfin, **Matplotlib** a permis la visualisation graphique des résultats, notamment pour analyser la distribution des variables et illustrer les performances des modèles.

- **XGBoost :**



Deux algorithmes supervisés utilisés pour la prédiction du churn. XGBoost a été privilégié pour sa performance ($AUC \approx 0.83$), tandis que Random Forest a renforcé les tests comparatifs.

2.3. Backend et Intégration :

- **FastAPI :**



Framework léger et performant utilisé pour le développement de l'API REST. Il permet l'exposition des scores de churn, recommandations, et alertes à travers des endpoints intégrables au frontend.

2.4. Frontend et Visualisation :

- **Next.js :**



Next.js a été utilisé pour développer une interface web moderne permettant de visualiser les prédictions de manière dynamique. Tailwind CSS a permis une personnalisation rapide et élégante du design.

- **Power BI :**



Solution de visualisation interactive utilisée pour générer des graphiques et indicateurs stratégiques (segmentation, risque client, services utilisés...). Les données étaient importées depuis les résultats de l'API ou des fichiers CSV exportés.

2.5. Source des données :

- **Kaggle :**



Pour la phase de prototypage et de validation des modèles d'apprentissage automatique, un jeu de données public a été utilisé, issu de la plateforme Kaggle. Plus précisément, le dataset **"Telco Customer Churn"** a servi de base pour entraîner et tester les modèles prédictifs de churn et les algorithmes de recommandation. Kaggle est une plateforme de référence dans le domaine de la data science, proposant des datasets variés, des compétitions, et une large communauté d'experts.

3. Les langages utilisés :

- **JavaScript :**



JavaScript est un langage de programmation utilisé pour rendre les pages web interactives, permettant des fonctionnalités dynamiques comme des animations et des formulaires interactifs. Il complète HTML pour la structure et CSS pour la présentation, formant un trio essentiel pour développer des sites web réactifs.

- **HTML :**



HTML est un langage de balisage qui structure le contenu des sites web en utilisant des balises pour définir titres, paragraphes, images et liens. Il forme la base des pages web, permettant l'intégration de styles CSS et de fonctionnalités JavaScript.

- **CSS :**



CSS, acronyme de Cascading Style Sheets, est un langage informatique utilisé pour la mise en forme et la présentation des pages web HTML. Il permet de contrôler l'apparence des éléments, tels que les couleurs, les polices et les mises en page, en séparant le style du contenu.

- **Python :**



Python est un langage principal du projet, utilisé pour le traitement des données, la modélisation IA et le développement de l'API REST. Ses bibliothèques riches ont facilité le développement rapide de prototypes robustes.

➤ Conclusion

Ce chapitre a présenté en détail la gestion du projet, les outils mobilisés et les choix technologiques qui ont structuré le développement du système prédictif. La méthode agile a permis une exécution progressive, tout en assurant une qualité technique à chaque étape.

L'ensemble des technologies utilisées a été sélectionné pour leur complémentarité, leur efficacité dans le contexte Big Data, et leur adaptabilité à un projet de modélisation intelligente. Le prochain chapitre portera sur la conception fonctionnelle et technique du système, notamment les flux de traitement, l'architecture et la modélisation conceptuelle

Chapitre 3 :

Architecture et conception technique du système prédictif

➤ Introduction

La phase d'analyse et de conception constitue une étape déterminante dans le développement d'un système intelligent. Elle permet de transformer les besoins exprimés en une architecture logique et technique, facilitant ainsi le développement et garantissant la cohérence globale du projet. Ce chapitre décrit l'étude fonctionnelle, l'analyse conceptuelle, ainsi que les choix d'architecture adoptés pour le système de prédiction de churn et de recommandation client développé chez HardTec.

À titre de rappel, les technologies suivantes ont été mobilisées, chacune ayant un rôle précis dans la chaîne de traitement et de valorisation des données :

Technologie	Rôle principal dans le système
- Apache Spark	Traitement distribué des données (ingestion, nettoyage, transformation).
- PySpark	Intégration de Spark dans un environnement Python.
- XGBoost / Random Forest	Modélisation de la prédiction de churn.
- FastAPI	Développement de l'API REST pour exposer les résultats.
- Power BI	Visualisation interactive et analytique des résultats prédictifs.
- Next.js	Développement du dashboard web frontend.
- Docker	Conteneurisation pour portabilité et déploiement reproductible.

Tableau 3 : Technologies mobilisées et rôle dans le système prédictif

1. Analyse des Besoins :

1.1 Objectifs fonctionnels :

- Anticiper la perte de clients via des modèles prédictifs robustes.
- Générer automatiquement des recommandations de services personnalisés.
- Alerter l'équipe commerciale via une interface dédiée en cas de risque élevé de départ.
- Visualiser dynamiquement les résultats dans un dashboard interactif (Power BI, Next.js).

1.2 Objectifs techniques :

- Traitement massif de données multi-sources (CRM, facturation, support client).
- Construction d'un pipeline scalable de Machine Learning (PySpark + XGBoost).
- Intégration via une API REST pour exposition des résultats.
- Conteneurisation et reproductibilité via Docker.

1.3 Besoins liés à la visualisation et à la BI :

Dans le cadre du projet développé chez hardtec, la mise en place d'un système de visualisation interactif est essentielle pour permettre à l'équipe métier d'exploiter efficacement les résultats du moteur de prédiction et de recommandation.

Le schéma suivant présente les principaux besoins fonctionnels liés à la solution BI, notamment la centralisation des données, l'automatisation des tableaux de bord, la mise à jour continue et la prise de décision rapide.

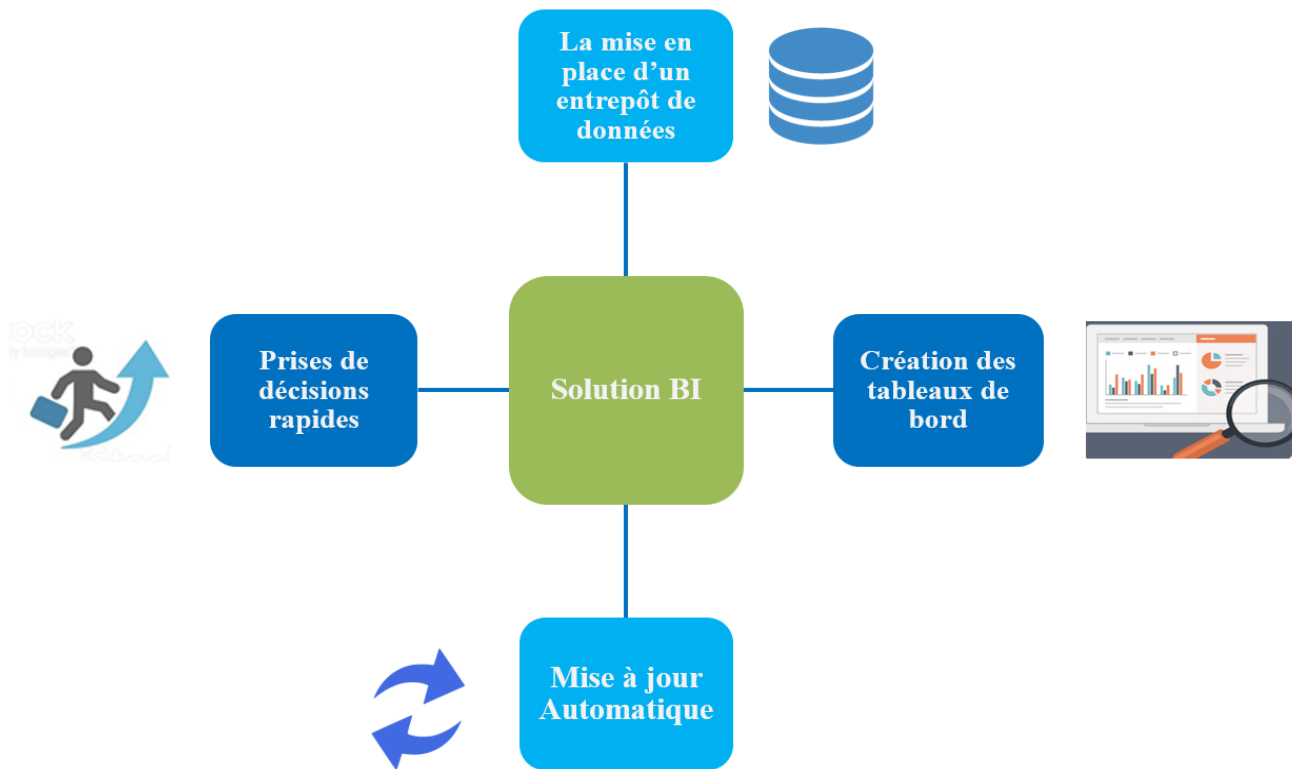


Figure 4 : Spécification des besoins fonctionnels liés à la solution BI

2. Architecture fonctionnelle du système :

L'architecture fonctionnelle représente la structure globale du système développé pour la prédiction du churn client et la recommandation de services personnalisés. Elle décrit les différentes couches logicielles, leur rôle respectif, ainsi que les interactions entre elles, de l'acquisition des données jusqu'à la visualisation finale.

Le système repose sur un pipeline complet, depuis les sources de données internes et externes (CRM, logs, jeux de données publics Kaggle), en passant par un traitement distribué avec Apache Spark, jusqu'à la modélisation prédictive avec des algorithmes de machine learning (XGBoost, Random Forest).

Les résultats sont exposés via une API REST développée avec FastAPI, et intégrés à une interface frontend dynamique construite avec Next.js. Une visualisation complémentaire des

données et des scores prédictifs est également assurée avec Power BI pour permettre à l'équipe commerciale d'exploiter les résultats et de réagir aux alertes de churn.

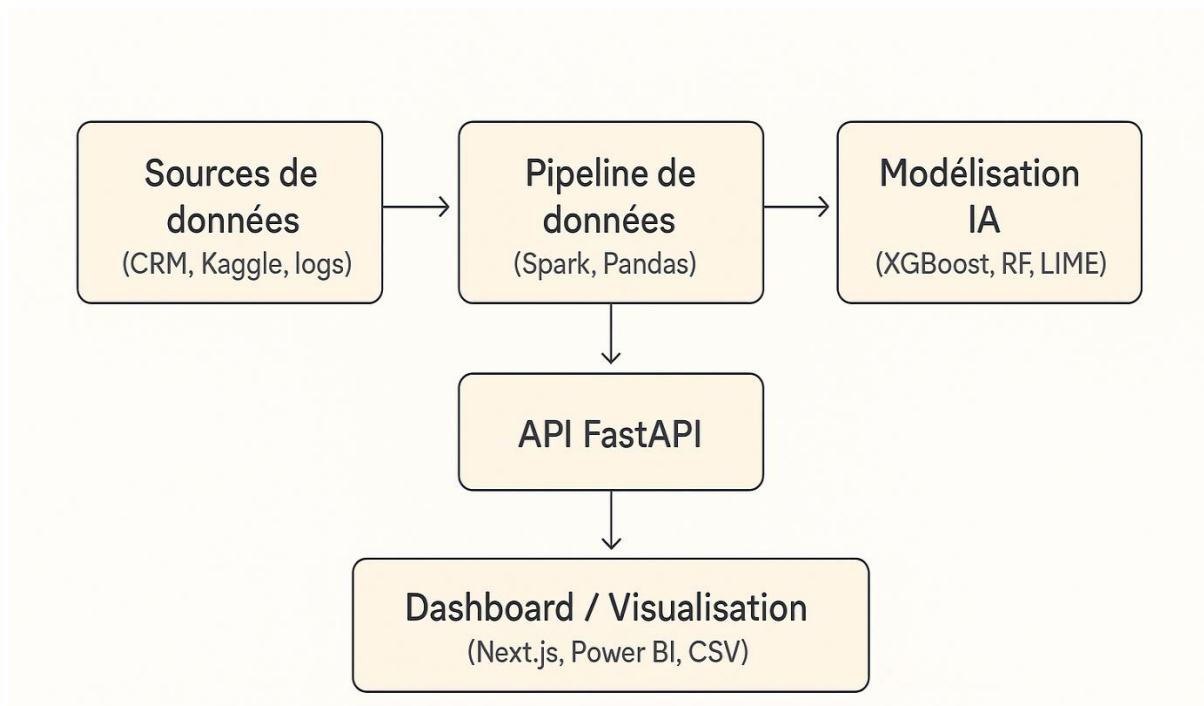


Figure 5 : Architecture fonctionnelle globale du système prédictif de churn et de recommandation

3. Analyse fonctionnelle :

3.1 Flux principal :

Avant de passer à la modélisation technique, il est essentiel de comprendre le flux global du système développé. Ce flux fonctionnel décrit les grandes étapes de traitement des données, depuis leur collecte jusqu'à leur exploitation visuelle par l'équipe métier. Il permet de structurer l'architecture de travail du projet de manière progressive et logique.

Étape	Description
- Collecte	Import des données (CRM, logs internes, dataset Kaggle) via Spark.
- Préparation	Feature engineering, transformation, encodage.
- Modélisation	Entraînement de modèles de classification (churn) et de recommandation.
- Inférence	Génération de scores de churn + services recommandés par client.
- Exposition	Mise à disposition des résultats via API.
- Visualisation	Dashboard interactif (Next.js) + rapports analytiques via Power BI.

Tableau 4 : Tableau récapitulatif des étapes fonctionnelles principales

4. Schéma conceptuel des entités principales :

Afin de garantir une organisation claire et cohérente des données utilisées dans le système de prédiction de churn et de recommandation, un schéma de structuration des entités a été défini. Ce schéma, sans se baser formellement sur la méthode Merise (MCD), permet de visualiser les entités principales impliquées dans le pipeline et leurs relations fonctionnelles. Chaque entité représente un composant stratégique du système :

- **Client** : stocke les informations générales liées aux utilisateurs.
- **Prédiction** : contient les scores générés par les modèles IA (churn, recommandations).
- **Recommandation** : regroupe les suggestions personnalisées de services à proposer.
- **Interaction** : historise les échanges et actions (ex. : clics, paiements, feedback).

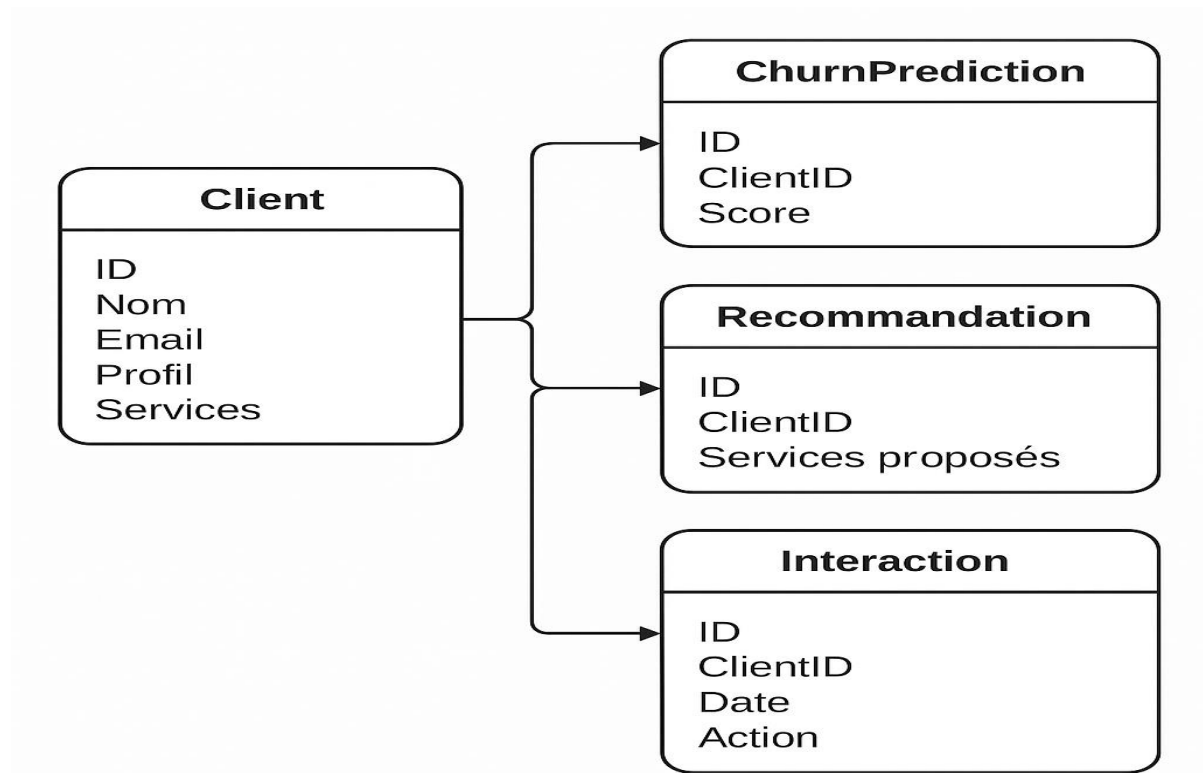


Figure 6 : Schéma conceptuel des entités principales

5. Pipeline technique Big Data :

5.1 Étapes techniques (PySpark & Scikit-learn) :

Le pipeline de traitement des données mis en place suit une démarche rigoureuse et structurée, allant de l'ingestion brute jusqu'à l'évaluation des modèles prédictifs. Chaque étape du pipeline joue un rôle fondamental dans la qualité des résultats obtenus, en garantissant la fiabilité des données traitées et la performance des algorithmes d'apprentissage supervisé. Le tableau ci-dessous résume les principales étapes ainsi que les outils et fonctions associés :

Étape	Outils / Fonction
- Ingestion	Spark (lecture CSV/Parquet).
- Nettoyage	Traitement des nulls, doublons, outliers.
- Feature Engineering	Encodage, normalisation, agrégation.
- Modélisation	Entraînement des modèles XGBoost et Random Forest
- Évaluation	AUC, précision, F1-Score.

Tableau 5 : Étapes techniques du pipeline de modélisation prédictive

Cette figure illustre le score AUC obtenu par le modèle XGBoost après évaluation : un résultat de 0.83 qui reflète une bonne performance prédictive.

```

Set SPARK_HOME to: D:\Documents D\ENSI\G-INFO-4\Hardtec\spark-3.5.5-bin-hadoop3
Added Spark bin to PATH: D:\Documents D\ENSI\G-INFO-4\Hardtec\spark-3.5.5-bin-hadoop3\bin
Added PySpark to Python path: D:\Documents D\ENSI\G-INFO-4\Hardtec\spark-3.5.5-bin-hadoop3\python
Added PySpark lib to Python path: D:\Documents D\ENSI\G-INFO-4\Hardtec\spark-3.5.5-bin-hadoop3\python\lib
Added virtual environment site-packages to Python path: D:\Documents D\ENSI\G-INFO-4\Hardtec\venv-py311\Lib\site-packages
Environment setup complete!
Set PYSARK_PYTHON to: D:\Documents D\ENSI\G-INFO-4\Hardtec\venv-py311\Scripts\python.exe
The system cannot find the path specified.
25/04/02 13:32:23 WARN Shell: Did not find winutils.exe: java.io.FileNotFoundException: java.io.FileNotFoundException: HADOOP_HOME and hadoop.home.dir are unset. -see https://wiki.ap
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
25/04/02 13:32:24 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
AUC Score: 0.8344648251810344

Feature Importances:
tenure: 0.17021656000745924
MonthlyCharges: 0.022737769685828926
TotalCharges: 0.08680268174164392
gender_encoded: 0.00038238451087192255
Partner_encoded: 0.0002598601994083805
Dependents_encoded: 0.0018008516979360985
PhoneService_encoded: 0.0006011748500869575
InternetService_encoded: 0.0008967975672890559
OnlineSecurity_encoded: 0.0008224225205376638
OnlineBackup_encoded: 0.0004109441997962411
DeviceProtection_encoded: 0.0011696461143546938
TechSupport_encoded: 0.06873628727827559
StreamingTV_encoded: 0.018577517063508426
StreamingMovies_encoded: 0.010736374377696261
Contract_encoded: 0.08896745900006685
PaperlessBilling_encoded: 0.005896952142047422
PaymentMethod_encoded: 0.0174196964194884
SUCCESS: The process with PID 19320 (child process of PID 16184) has been terminated.
SUCCESS: The process with PID 16184 (child process of PID 224) has been terminated.
SUCCESS: The process with PID 224 (child process of PID 16804) has been terminated.

```

Figure 7 : Résultat de l'évaluation du modèle

6. Visualisation analytique avec Power BI :

Afin de valoriser les résultats produits par les modèles de prédiction de churn, une interface de visualisation a été développée avec Power BI, outil puissant d'aide à la décision. Cette interface repose sur un jeu de données de test issu de la plateforme Kaggle (Telco Customer Churn) et permet une lecture claire et interactive des résultats générés par le modèle XGBoost. Les indicateurs clés affichés dans le dashboard sont les suivants :

- Jauge dynamique indiquant le nombre total de clients prédits comme susceptibles de churn.
- Diagramme circulaire (Pie Chart) comparant le volume de churn prédit et réel.
- Histogramme horizontal représentant la répartition des scores de churn selon les probabilités générées.
- Bar chart agrégé affichant le nombre total de churns identifiés par le système.

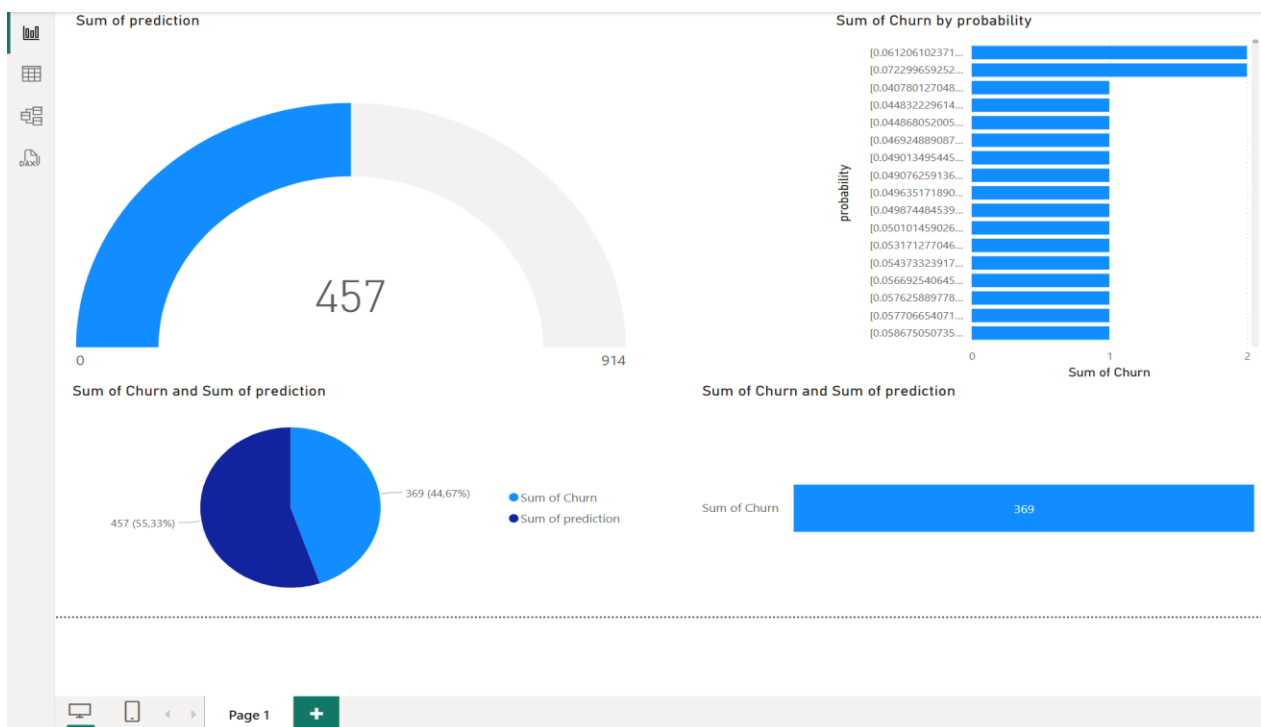


Figure 8 : Tableau de bord Power BI – Visualisation test des résultats prédictifs à partir du dataset Kaggle

7. Composants intelligents intégrés : recommandations & alertes :

Le système développé va au-delà de la simple prédiction de churn, en intégrant des mécanismes intelligents visant à automatiser les actions de rétention et à renforcer la réactivité de l'équipe commerciale.

- Recommandations personnalisées par email :

Pour chaque client identifié comme étant à risque, une recommandation de service complémentaire est générée automatiquement par le système, en fonction de son profil et de son historique. Ces suggestions sont ensuite envoyées par email au client (ex. : proposition de contrat de maintenance, accompagnement SEO ou mise à niveau technique). Cette stratégie permet de cibler les besoins réels des clients et de créer des opportunités de fidélisation ou de ventes croisées.

- Alertes visuelles intégrées dans le dashboard :

Dès qu'un score de churn dépasse un seuil critique prédéfini (par exemple ≥ 0.8), une alerte visuelle s'affiche automatiquement dans le tableau de bord Next.js. Cette alerte, sous forme de notification

8. Prototype de l'interface – Design initial du Dashboard :

L'interface du dashboard a été développée en Next.js, avec une mise en forme assurée par Tailwind CSS. Ce prototype visuel a été conçu sans données réelles (ni Kaggle, ni données internes), dans le but de valider uniquement l'ergonomie, le design général, ainsi que l'architecture des composants. À ce stade, aucune donnée n'était intégrée. Le but était de :

- Structurer le code frontend et préparer l'affichage des sections clés (churn, recommandations, alertes).
- Assurer la responsivité et la cohérence UX/UI du tableau de bord.
- Préparer la communication future avec l'API FastAPI.

Ce design a ensuite servi de base pour la version finale connectée aux données réelles, présentée au **chapitre 4**.

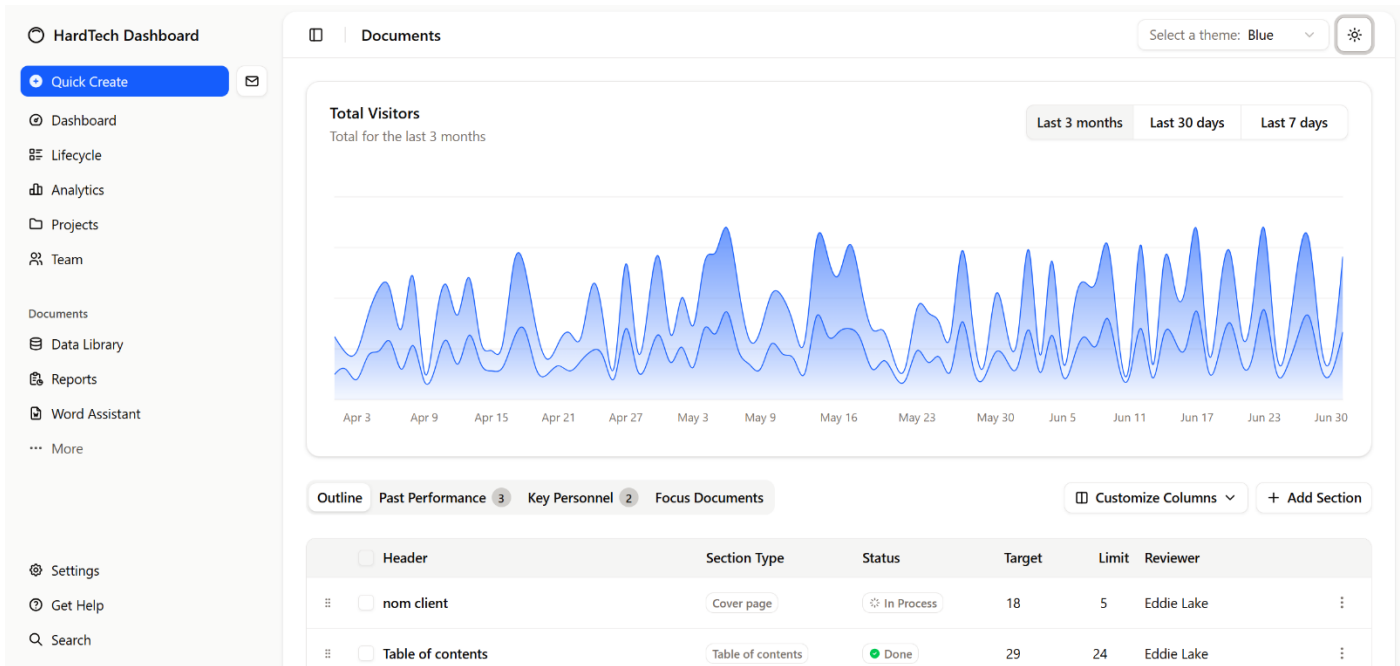


Figure 9 : Design du dashboard développé avec Next.js (avant intégration réelle des données)

➤ Conclusion

Ce chapitre a permis de détailler la conception technique du système prédictif de churn et de recommandation mis en place chez HardTec. L'architecture adoptée s'appuie sur des technologies éprouvées (Apache Spark, FastAPI, XGBoost, Power BI, Next.js), assurant un traitement performant et scalable des données, ainsi qu'une visualisation claire pour l'équipe commerciale. L'ensemble des choix techniques a été guidé par la recherche d'efficacité, de robustesse et d'interprétabilité, dans le respect des bonnes pratiques industrielles. Cette base fonctionnelle servira de socle à la prochaine phase, consacrée à la mise en œuvre opérationnelle et à l'évaluation des résultats.

Chapitre 4 :

Mise en œuvre et

Réalisation

➤ Introduction

Ce chapitre présente la concrétisation technique du projet à travers l'implémentation des composants développés, l'intégration des différentes couches (modèle prédictif, API, frontend), ainsi que la mise en place des interfaces utilisateur. Il s'appuie sur des captures d'écrans, des démonstrations fonctionnelles et des résultats réels obtenus durant le développement du système chez HardTec.

1. Structure de projet :

1.1 Architecture de projet :

L'architecture adoptée dans le cadre du projet HardTec repose sur le modèle **MVVM** (**Modèle – Vue – VueModèle**), qui favorise une séparation claire entre la logique métier, la logique de présentation et l'interface utilisateur. Cette structure est particulièrement adaptée aux applications web modernes, en garantissant une bonne maintenabilité et une réutilisabilité du code.

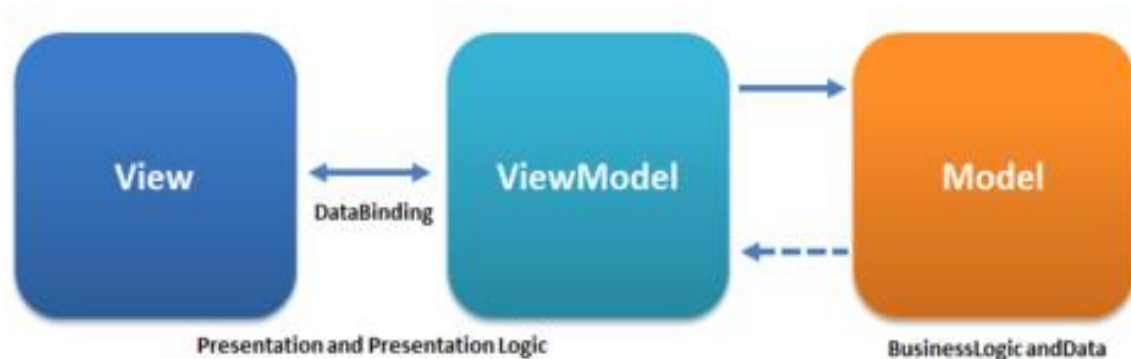


Figure 10 : Architecture MVVM du projet

- **Modèle (Model)** : Backend en Python (Spark, FastAPI, Scikit-learn) pour le traitement des données et les prédictions de churn.
- **Vue (View)** : Frontend en Next.js + Tailwind CSS pour l'affichage des scores, alertes et recommandations.

- **VueModèle (ViewModel)** : Couche d'interaction (hooks, composants) assurant les appels API et la gestion des états dans Next.js.

2. Structure de projet côté Backend :

Le backend, développé en Python, utilise PySpark pour le traitement de données, des scripts de modélisation IA pour la prédiction, et FastAPI pour exposer une API REST. Le tout est conteneurisé avec Docker pour un déploiement simple et portable.

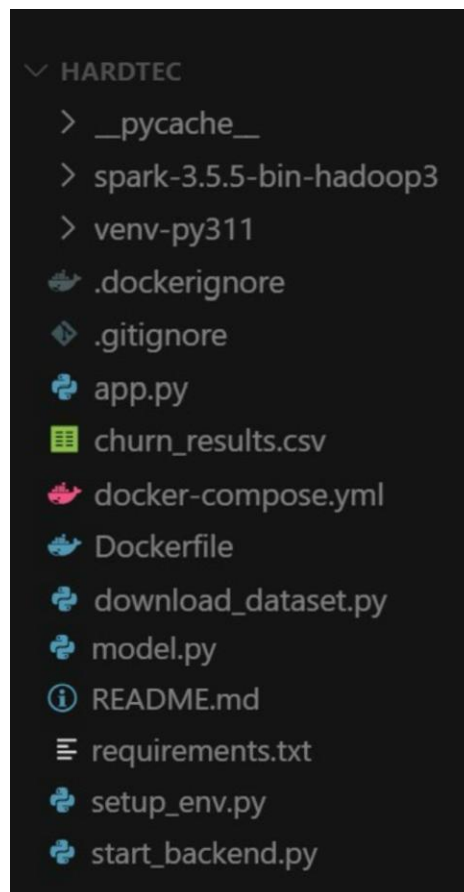


Figure 11 : Structure du backend – traitement des données, modélisation IA et API FastAPI conteneurisée avec Docker.

- **venv-py311/** : Environnement virtuel contenant les dépendances Python.
- **spark-3.5.5-bin-hadoop3/** : Binaire Apache Spark local pour traitement distribué.
- **model.py** : Script de modélisation IA (XGBoost, Random Forest).
- **download_dataset.py** : Téléchargement et préparation du dataset.
- **churn_results.csv** : Fichier CSV contenant les prédictions de churn générées par les modèles.

- **start_backend.py** : Script de lancement simplifié du backend, combinant l'environnement, FastAPI et Spark.
- **setup_env.py** : Script d'initialisation de l'environnement local (variables, chemins Spark, etc.).
- **requirements.txt** : Liste des bibliothèques Python nécessaires (FastAPI, PySpark, scikit-learn, etc.).
- **Dockerfile** : Définit l'image Docker pour containeriser l'application backend.
- **docker-compose.yml** : Configuration multi-conteneurs (Spark, API, etc.) pour l'orchestration et le déploiement.

3. Structure de projet côté frontend :

Le frontend a été développé avec Next.js, en s'appuyant sur TypeScript pour la robustesse du typage, et Tailwind CSS pour un design moderne, responsive et facilement maintenable. Cette combinaison permet de construire une interface utilisateur performante, dynamique et évolutive.

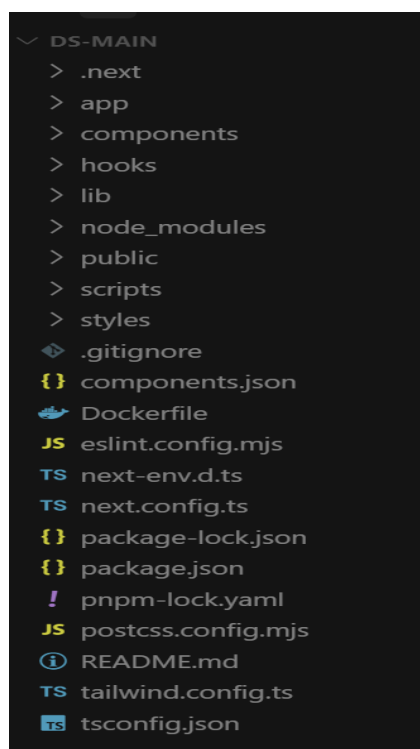


Figure 12 : Structure du frontend développé avec Next.js

- **/app, /components, /hooks** : Répartissent respectivement les pages dynamiques, les composants réutilisables (UI), et la logique d'état via des hooks personnalisés.
- **/styles** : Fichiers CSS personnalisés utilisant Tailwind.
- **/public** : Dossier destiné aux ressources statiques (images, icônes, fichiers).
- **next.config.ts, tailwind.config.ts** : Fichiers de configuration essentiels au bon fonctionnement de Next.js et Tailwind CSS.
- **tsconfig.json** : Définit les options de compilation du projet TypeScript
- **package.json** : Contient la liste des dépendances, scripts et métadonnées du projet.

4. Les interfaces graphiques :

4.1 Les composants de Projet par partie :

Dans cette section, nous allons examiner les principales interfaces graphiques développées dans le cadre du projet de prédiction de churn client chez HardTec. Ces interfaces sont conçues pour répondre aux besoins des différentes parties prenantes : data scientists, développeurs, équipes métiers et commerciaux. Elles offrent une expérience utilisateur claire, interactive et orientée action. Voici un aperçu des différents composants du système ainsi que leur fonctionnement.

4.1.1 Partie Backend – Exécution du modèle :

✓ Terminal du modèle Python :

Cette interface montre le processus de lancement du backend en Python à l'aide de FastAPI et Spark. On y voit l'entraînement du modèle XGBoost, l'inférence, l'AUC obtenu ainsi que l'export des résultats sous forme CSV.

Fonctionnalités visibles :

- Lancement du serveur avec Uvicorn.
- Score AUC : **0.776**.
- Export des résultats vers churn_results.csv.

```
2025-07-01 07:40:52,721 INFO XGBoost-PySpark: _fit Finished xgboost training!
Model training completed!
Making predictions...
Predictions completed!
2025-07-01 07:41:01,165 INFO XGBoost-PySpark: predict_udf Do the inference on the CPUs
AUC Score: 0.7764112903225805

Feature Importances:
tenure: f0
MonthlyCharges: f1
TotalCharges: f2
gender_encoded: f3
Partner_encoded: f4
Dependents_encoded: f5
PhoneService_encoded: f6
InternetService_encoded: f7
OnlineSecurity_encoded: f8
OnlineBackup_encoded: f9
DeviceProtection_encoded: f10
TechSupport_encoded: f11
StreamingTV_encoded: f12
StreamingMovies_encoded: f13
Contract_encoded: f14
PaperlessBilling_encoded: f15
PaymentMethod_encoded: f16
2025-07-01 07:41:09,340 INFO XGBoost-PySpark: predict_udf Do the inference on the CPUs

✓ Résultats exportés vers : D:\Documents D\ENSI\G-INFO-4\HARDTEC-PFE\Hardtec\churn_results.csv
INFO: Started server process [1516]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
```

Figure 13 : Terminal du modèle Python

4.1.2 Partie Analytique –Power BI :

✓ Interface Power BI :

Une visualisation complémentaire a été réalisée dans Power BI, permettant d'exploiter dynamiquement les résultats du modèle. On y retrouve :

- Des jauges et graphiques circulaires sur la proportion de churn.
- Une distribution des probabilités.
- Un comparatif churn réel / prédiction.

Figure 14 : Visualisation Power BI des résultats de churn

4.1.3 Partie Frontend – Interface utilisateur Next.js :

✓ Terminal de lancement du dashboard (Next.js) :

Ce terminal montre l'exécution du frontend avec Next.js 15.3.1, accessible localement à l'adresse <http://localhost:3000>.

```
> dashboard@0.1.0 dev
> next dev --turbo

▲ Next.js 15.3.1 (Turbo)
- Local:      http://localhost:3000
- Network:    http://192.168.137.1:3000

✓ Starting...
Slow filesystem detected. If D:\Documents D\ENSI\G-INFO-4\HARDTEC-PFE\DS-main\next is a network drive, consider moving it to a local folder. If you have an antivirus enabled, consider excluding your project directory.
✓ Ready in 56.8s
█
```

Figure 15 : Terminal de démarrage du frontend Next.js

✓ **Dashboard principal :**

Cette interface centrale permet à l'utilisateur d'accéder à l'ensemble des résultats de prédiction de churn. Affichage central des clients avec leurs scores de churn, statuts et profils. Possibilité de filtrer par churn, colonne personnalisable.

Fonctionnalités visibles :

- Colonne personnalisable.
- Mise en forme conditionnelle selon le risque de churn.
- Navigation fluide et responsive.

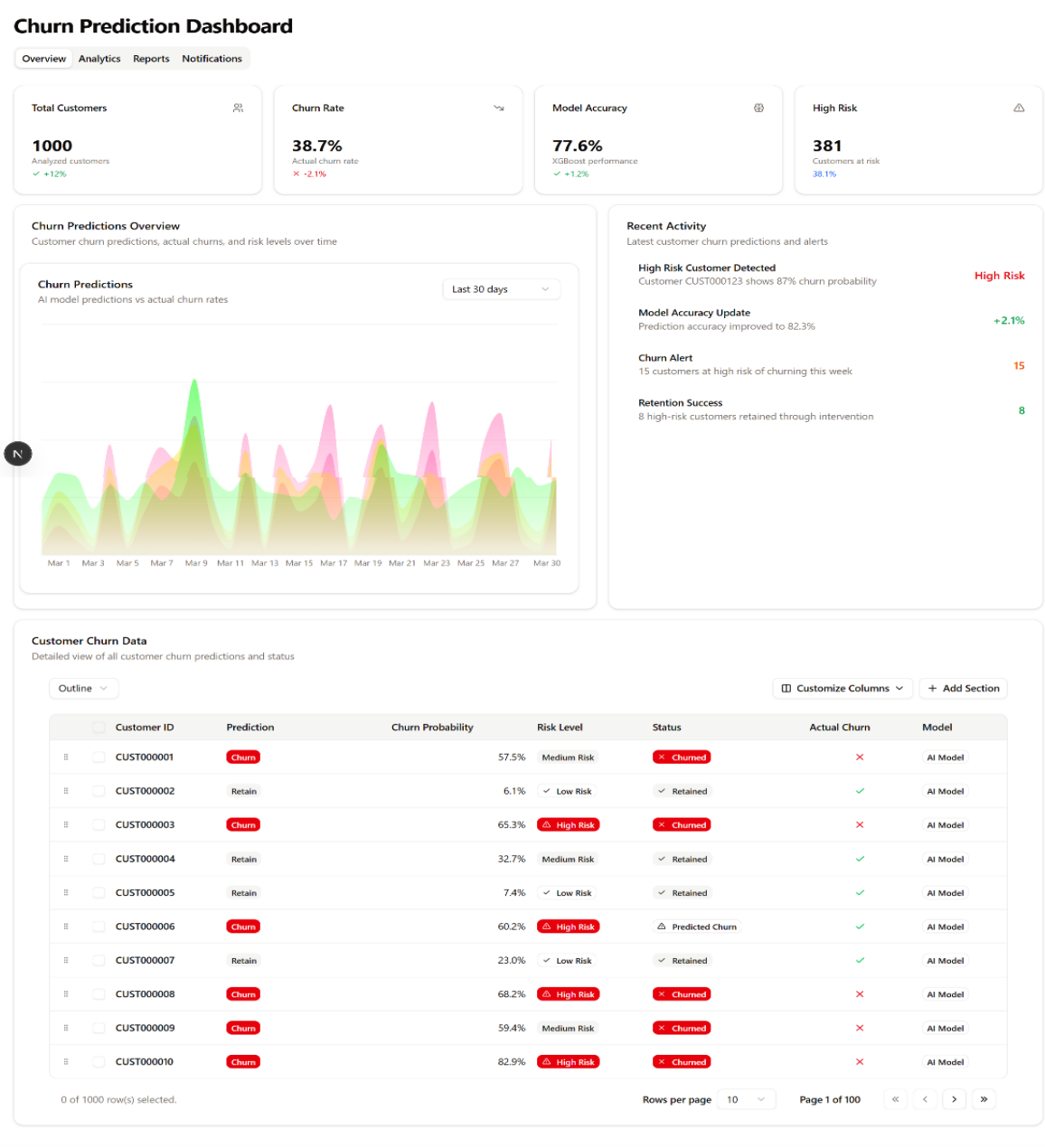


Figure 16 : Interface principale du dashboard – prédictions de churn clients

Churn Prediction Dashboard

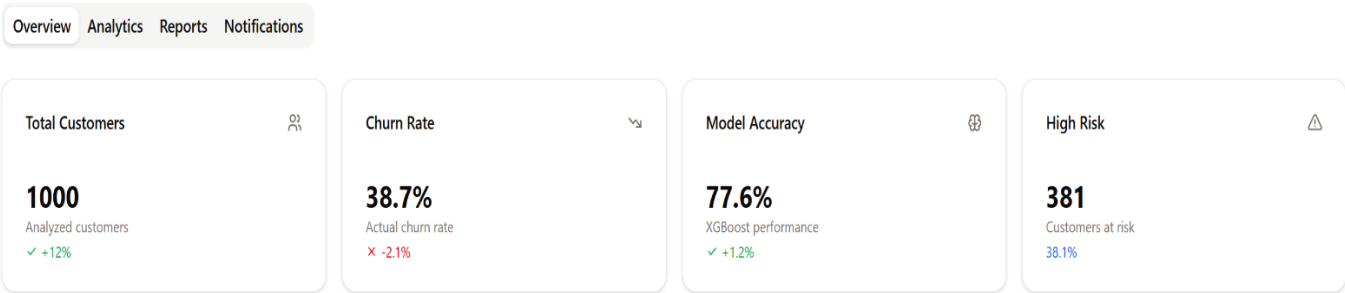


Figure 17 : Résumé statistique des résultats de churn

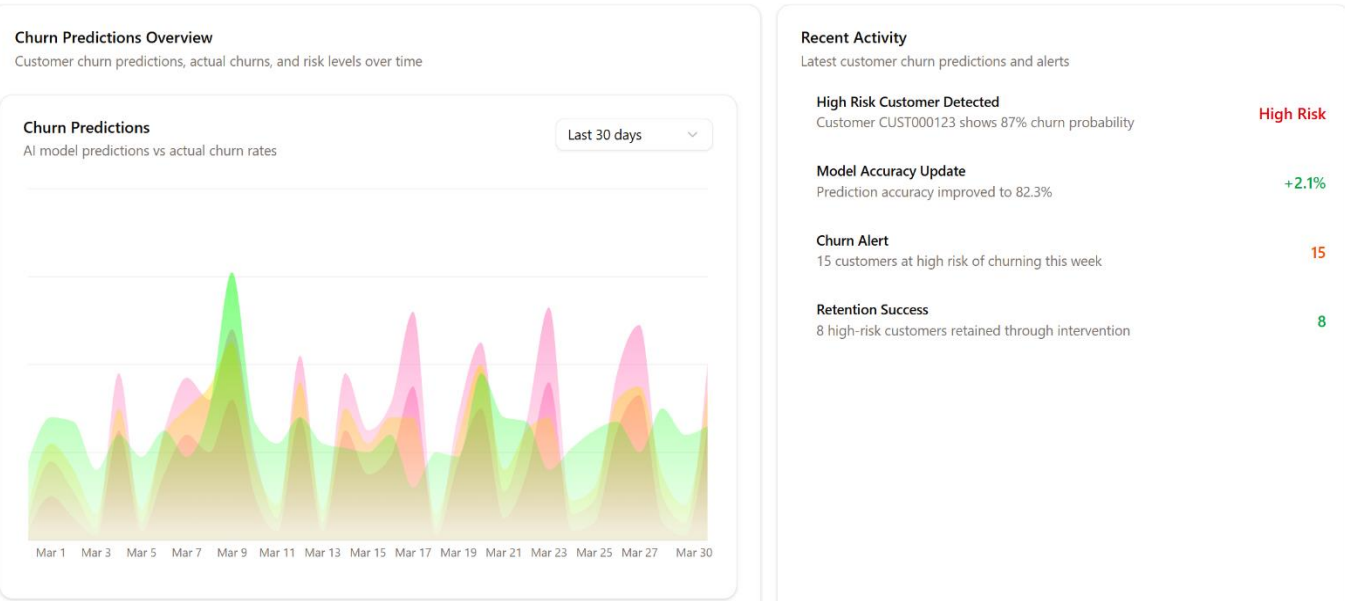


Figure 18 : Visualisation graphique des résultats – scores de churn

Customer Churn Data

Detailed view of all customer churn predictions and status

Outline

Customize Columns

Add Section

Customer ID	Prediction	Churn Probability	Risk Level	Status	Actual Churn	Model
CUST000001	Churn	57.5%	Medium Risk	✗ Churned	✗	AI Model
CUST000002	Retain	6.1%	Low Risk	✓ Retained	✓	AI Model
CUST000003	Churn	65.3%	High Risk	✗ Churned	✗	AI Model
CUST000004	Retain	32.7%	Medium Risk	✓ Retained	✓	AI Model
CUST000005	Retain	7.4%	Low Risk	✓ Retained	✓	AI Model
CUST000006	Churn	60.2%	High Risk	⚠ Predicted Churn	✓	AI Model
CUST000007	Retain	23.0%	Low Risk	✓ Retained	✓	AI Model
CUST000008	Churn	68.2%	High Risk	✗ Churned	✗	AI Model
CUST000009	Churn	59.4%	Medium Risk	✗ Churned	✗	AI Model
CUST000010	Churn	82.9%	High Risk	✗ Churned	✗	AI Model

Figure 19 : Customer Churn Data – Résultats détaillés par client

Customer Churn Data

Detailed view of all customer churn predictions and status

Outline ▾

Customize Columns ▾ + Add Section

<input type="checkbox"/> Customer ID	Prediction	Churn Probability	Risk Level	Status	Model
<input type="checkbox"/> CUST000001	Churn	57.5%	Medium Risk	✗ Churned	AI Model
<input type="checkbox"/> CUST000002	Retain	6.1%	Low Risk	✓ Retained	AI Model
<input type="checkbox"/> CUST000003	Churn	65.3%	High Risk	✗ Churned	AI Model
<input type="checkbox"/> CUST000004	Retain	32.7%	Medium Risk	✓ Retained	AI Model

☒ Prediction
☒ Probability
☒ Risk_level
☒ Status
☒ Churn
☒ Reviewer

Figure 20 : Personnalisation des colonnes (customize columns)

4.1.4 Système d'Alerte – Visualisation des risques en temps réel :

✓ Alertes : affichage intégré dans le dashboard pour les équipes métiers :

Une zone du dashboard est dédiée aux alertes générées automatiquement par le système affiche les clients à risque, les alertes hebdomadaires et l'évolution des performances du modèle (ex. +2.1 %). Elle permet une réaction rapide des équipes métiers.

Recent Activity
Latest customer churn predictions and alerts

High Risk Customer Detected
Customer CUST000123 shows 87% churn probability

High Risk

Model Accuracy Update
Prediction accuracy improved to 82.3%

+2.1%

Churn Alert
15 customers at high risk of churning this week

15

Retention Success
8 high-risk customers retained through intervention

8

Figure 21 : Zone d'alerte – Suivi des clients à risque

4.1.5 Recommandations – Offres personnalisées par e-mail :

✓ Recommandations envoyées automatiquement :

En cas de churn prédit, un e-mail est généré avec une offre ou un pack personnalisé (selon la politique de l'entreprise). Cela permet de fidéliser les clients à risque.

Figure 22 : Email de recommandation – Offre ciblée envoyée au client

4.1.6 Conteneurisation avec Docker :

✓ Déploiement conteneurisé du système avec Docker :

L'ensemble de la solution – backend Python (FastAPI, Spark), modèle IA, et frontend Next.js est conteneurisé à l'aide de Docker. Cela permet une installation rapide, reproductible et indépendante de l'environnement local.

La capture suivante illustre l'exécution des différents conteneurs via Docker Desktop, avec mapping des ports pour l'accès aux services :

- **churn-backend** → API Python sur le port 8000
- **churn-frontend** → Interface utilisateur sur le port 3000

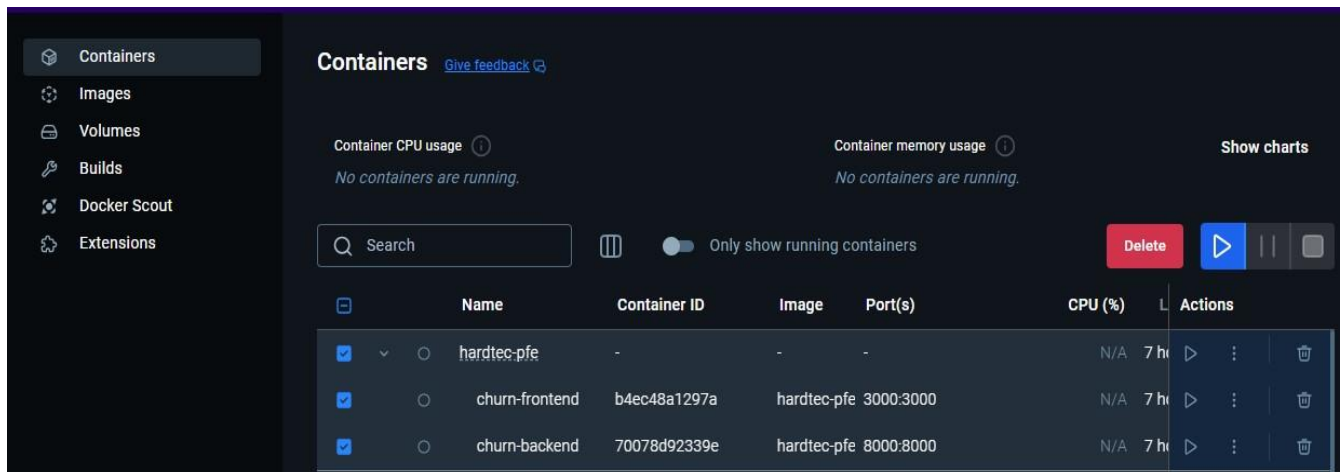


Figure 23 : Conteneurs Docker – Backend et Frontend exécutés via Docker Desktop

➤ Conclusion

Ce chapitre a présenté la réalisation technique du projet de prédiction de churn client chez HardTec. Il détaille l'architecture complète mise en place, combinant un backend Python/FastAPI, un frontend Next.js avec Tailwind CSS, et une orchestration via Docker. À travers les interfaces développées (dashboard interactif, alertes automatisées, recommandations personnalisées), le système démontre sa capacité à répondre aux besoins métiers de manière efficace. L'ensemble valide la cohérence technique du projet et pose les bases d'un déploiement en environnement réel.

Conclusion Générale

Mon stage chez HardTec a été une expérience très enrichissante et formatrice. J'ai eu l'opportunité de travailler sur un projet innovant de rétention client et de recommandation de services, qui m'a permis de renforcer mes compétences techniques en Big Data, intelligence artificielle et développement web. HardTec, en tant qu'entreprise spécialisée dans les solutions data-driven, se distingue par son engagement à valoriser ses données pour améliorer la fidélisation client et optimiser ses stratégies commerciales. La diversité des tâches réalisées et le soutien constant de l'équipe ont créé un cadre favorable à mon apprentissage et à la réalisation de mes objectifs professionnels.

Pour améliorer davantage le système développé, plusieurs fonctionnalités pourraient être ajoutées. Par exemple, l'implémentation d'un module de recommandations personnalisées en temps réel permettrait d'augmenter la pertinence des offres proposées aux clients. De plus, l'intégration d'un tableau de bord interactif pour les équipes commerciales faciliterait la prise de décision et le suivi des indicateurs clés. Enfin, le déploiement d'alertes automatiques basées sur le comportement des clients pourrait anticiper les risques de churn et déclencher des actions préventives.

Je suis convaincu que cette expérience chez HardTec constituera un véritable tremplin pour ma carrière. Les compétences acquises et les défis relevés me préparent efficacement aux exigences du monde professionnel dans les domaines du Big Data, de l'intelligence artificielle et des systèmes décisionnels. Je suis reconnaissant d'avoir contribué à un projet aussi ambitieux et porteur de valeur pour l'entreprise.

Références

- **HardTec. Site officiel de l'entreprise :**
<https://hardtec.net/>
- **Dépôt GitHub du projet HardTec :**
<https://github.com/nouhailaZamouri/hardtec>
- **Apache Spark Documentation :**
<https://spark.apache.org/docs/latest/>
- **XGBoost Documentation :**
<https://xgboost.readthedocs.io/en/stable/>
- **Random Forest Algorithm Overview:**
<https://scikit-learn.org/stable/modules/ensemble.html#forest>
- **FastAPI Documentation :**
<https://fastapi.tiangolo.com/>
- **Next.js Documentation :**
<https://nextjs.org/docs>
- **Power BI Documentation :**
<https://learn.microsoft.com/en-us/power-bi/>