

Charte de projet

***Projet : Développement d'une plateforme e-commerce SaaS pour CYNA-IT
Bachelor***

Équipe de projet :

**DA SILVA William
RASOULI Ali
WAZIRI Dylan
BALMOUSSIÈRE Mattéo**

Année scolaire : 2024–2025

Table des matières

1. Présentation du projet

1.1 Contexte	4
1.2 Objectifs	5
1.3 Alignement stratégique	6
1.4 Description générale du besoin	7
1.5 Documents de référence	7
1.6 Cibles et Utilisateurs.....	8

2. Cadrage et périmètre

2.1 Périmètre fonctionnel	9
2.1.1 Partie Front	9
2.1.2 Partie Back	10
2.2 Contraintes	11
2.3 Hypothèses	12
2.4 Facteurs clés de succès	13

3. Cas d’Affaire

3.1 Justification du projet.....	14
3.2 Bénéfices attendues	15
3.3 Coûts prévisionnels	16

4. Architecture technique

4.1 Architecture générale	17
4.2 Schéma d'architecture	18
4.3 Modèle de données	19
4.4 Sécurité et authentification	21
4.5 Technologies choisies et justifications	22
4.6 Diagrammes techniques	24

5. Gestion de projet

5.1 Méthodologie de gestion.....	25
5.2 Planning prévisionnel	26
5.3 Gantt Project	27
5.4 Livrables attendus	28
5.5 Suivi des risques	29

6. Organisation du projet

6.1 Équipe projet	30
6.2 Répartition des rôles	31
6.3 Instances de gouvernance	32

7. Annexes

7.1 Maquettes ou Schéma	33
7.2 Lexique	34

1.1 Contexte du projet

Cyna-IT est une entreprise spécialisée dans les solutions de sécurité SaaS, qui souhaite élargir son mode de distribution.

Jusqu'à présent, les services de Cyna-IT étaient commercialisés exclusivement via un réseau de partenaires et de vente physique.

Afin de s'adapter aux nouveaux utilisateurs et de toucher une clientèle plus large, l'entreprise a décidé de développer une plateforme e-commerce moderne et sécurisée à l'international pour commercialiser directement ses services en ligne.

Ce projet s'inscrit dans une démarche de transformation digitale, visant à rendre l'offre de Cyna-IT plus accessible, évolutive et facile à administrer. Il vise notamment à :

- *L'accessibilité (site web et application mobile).*
- *Une expérience utilisateur fluide (UX/UI moderne).*
- *L'automatisation de la gestion commerciale (commandes, abonnements et facturation).*
- *La sécurisation des données sensibles (authentification forte et paiement sécurisé).*

1.2 Objectifs du projet

L'objectif principal du projet est de développer une plateforme e-commerce destinée à permettre à Cyna-IT de vendre ses services SaaS en ligne, tout en garantissant

une expérience utilisateur fluide, sécurisée et évolutive.

Ce projet vise à répondre à plusieurs besoins clés de l'entreprise :

- ***Passer à la vente en ligne de ses services SaaS afin de simplifier l'accès pour les clients, avec un fonctionnement plus rapide et plus autonome.***
- ***Développer une interface mobile-first pour une utilisation optimale sur smartphone, tablette et ordinateur.***
- ***Créer une application mobile qui reprend les principales fonctionnalités du site web, afin de proposer une expérience fluide et accessible depuis un smartphone.***
- ***Mettre en place un outil d'administration complet (backoffice) pour que l'équipe d'administration puisse facilement gérer les produits, les commandes, les utilisateurs et les contenus du site.***
- ***Renforcer la sécurité des paiements et des données, grâce à des solutions comme l'authentification à deux facteurs, le chiffrement des informations sensibles, et l'intégration d'un service de paiement fiable (Stripe).***
- ***Prévoir l'évolution de la plateforme dans le temps, en gardant une structure technique souple, capable d'intégrer de nouvelles fonctionnalités si besoin.***

1.3 Alignement stratégique et enjeux

Le projet a pour volonté globale de transformation numérique de Cyna-IT. L'objectif est de faire évoluer l'entreprise vers un modèle plus moderne, plus accessible, et plus agile avec les nouvelles attentes du marché B2B.

Ce projet e-commerce permettra de :

- **Digitaliser la vente** en complément du réseau physique existant.
- **Toucher de nouveaux clients** notamment à l'international, en rendant l'offre accessible 24h/24 et sans intermédiaire.
- **Optimiser les processus internes** (commandes, paiements, gestion client), grâce à l'automatisation.
- **Renforcer l'image technologique** de l'entreprise, en proposant une expérience moderne.

En lien avec les objectifs stratégiques de l'entreprise, ce projet répond à plusieurs enjeux clés :

- **Commercial** : augmenter le chiffre d'affaires via la vente directe en ligne.
- **Technique** : mettre en place une plateforme évolutive, facile à maintenir, et compatible avec d'autres systèmes
- **Sécuritaire** : garantir la protection des données sensibles des clients.
- **Opérationnel** : réduire la charge manuelle des équipes commerciales via un backoffice.

Ce projet est donc pour améliorer à la fois la performance commerciale et l'efficacité opérationnelle de Cyna-IT, tout en s'alignant avec une stratégie de croissance digitale.

1.4 Description générale du besoin

Cyna-IT souhaite disposer d'une plateforme e-commerce complète et sécurisée, permettant aux entreprises clientes d'acheter, gérer et renouveler des solutions de cybersécurité SaaS en ligne.

Le besoin porte sur une solution accessible via web et mobile, combinant une interface client fluide (site et application mobile), un back-office pour l'administration de l'activité commerciale (gestion des produits, utilisateurs, commandes), ainsi qu'une intégration avec un système de paiement fiable et sécurisé.

Le système doit offrir une expérience utilisateur optimisée, une sécurité renforcée (authentification MFA, chiffrement des données) et une architecture évolutive capable de s'adapter à la croissance future de l'entreprise.

1.5 Document de référence

SUP-DE-VINCI-Projet-fil-rouge-CYNA-DEV

1.6 Cibles et utilisateurs

La plateforme e-commerce de Cyna-IT s'adresse à deux grandes catégories d'utilisateurs, chacune avec des besoins spécifiques :

Les clients :

Il s'agit d'entreprises ou de professionnels souhaitant souscrire à une ou plusieurs solutions de sécurité SaaS.

Ils accèdent à la plateforme via le site web ou l'application mobile pour :

- *Consulter les produits et leurs caractéristiques,*
- *Rechercher, filtrer et comparer les offres,*
- *Gérer leur panier et effectuer un paiement sécurisé,*
- *Accéder à un espace personnel (commandes, abonnements, factures...),*
- *Contacter le support via formulaire ou chatbot.*

Les administrateurs

Ils sont responsables de la gestion de la plateforme via une interface dédiée (backoffice) :

- *Gestion des produits, catégories et promotions,*
- *Suivi des commandes et des paiements,*
- *Gestion des utilisateurs et des comptes clients,*
- *Administration du contenu dynamique*
- *Accès à des tableaux de bord et rapports de ventes,*
- *Sécurité renforcée avec authentification à deux facteurs (MFA).*

2.1 Périmètre fonctionnel

2.1.1 Partie Front

Le projet prévoit un ensemble de fonctionnalités destinées à couvrir l'ensemble du parcours utilisateur, aussi bien côté client que côté administrateur. Il repose sur deux interfaces principales :

- *Une plateforme web et mobile accessible aux clients,*
- *Un backoffice d'administration réservé à l'équipe interne de Cyna-IT.*

La partie front-office correspond à l'interface que les clients utiliseront pour naviguer sur la plateforme, consulter les offres et effectuer leurs achats en ligne. L'expérience utilisateur est pensée pour être simple, rapide et responsive, quel que soit l'appareil utilisé.

Les principales fonctionnalités prévues côté client sont :

- *Navigation et recherche dans le catalogue de services (tri, filtres, catégories),*
- *Consultation détaillée des fiches produits (caractéristiques, prix, durée),*
- *Gestion du panier et validation de commande,*
- *Paiement en ligne sécurisé (intégration Stripe),*
- *Création et gestion de compte utilisateur, avec authentification (email + mot de passe),*
- *Espace personnel : consultation des commandes, abonnements, factures,*
- *Formulaire de contact ou chatbot pour l'assistance,*
- *Responsive design : affichage optimisé sur smartphone, tablette et desktop.*

2.1 Périmètre fonctionnel

2.1.2 Partie Back

Le backoffice est l'espace réservé à l'équipe de Cyna-IT, destiné à gérer la plateforme. Il permet d'administrer les contenus, les utilisateurs, les produits et le suivi des ventes.

Les fonctionnalités clés du backoffice sont :

- **Gestion des produits : ajout, modification, suppression,**
- **Gestion des catégories et promotions,**
- **Suivi des commandes et des paiements,**
- **Gestion des utilisateurs (*clients*) et de leurs accès,**
- **Mise à jour du contenu dynamique (*carrousel, bannières...*),**
- **Statistiques de ventes et tableaux de bord,**
- **Authentification renforcée avec double facteur (MFA).**

2.2 Contraintes

Le projet est soumis à plusieurs contraintes qui influencent sa réalisation :

Contraintes techniques :

On s'appuie sur des technologies précises (React, Node.js, SQL...) pour garantir la cohérence du code. Pas question de partir sur un autre framework en cours de route !

Les services externes comme Stripe (paiements) et Google Authenticator (sécurité) sont incontournables – leur intégration devra être nickel.

La plateforme doit fonctionner aussi bien sur mobile que desktop, sans sacrifier l'expérience utilisateur.

Contraintes temporelles :

Le calendrier académique est strict : pas de retard possible sur les livrables.

On aura besoin de temps pour tester et peaufiner – il faudra être efficaces.

Contraintes légales :

Protection des données (RGPD) et normes bancaires (PCI DSS) : on n'a pas le droit à l'erreur.

Contraintes budgétaires :

On privilégie les outils gratuits ou open-source pour éviter les surprises. Les frais Stripe seront à anticiper dans les coûts.

Contraintes humaines :

Chacun a son rôle (front, back, base de données...) – la communication sera clé pour éviter les blocages.

2.3 Hypothèses

Les hypothèses suivantes ont été formulées pour guider le projet :

Techniques :

Les utilisateurs auront un accès Internet stable pour utiliser la plateforme.

Les clients utiliseront indifféremment un appareil mobile ou desktop pour accéder à la plateforme.

Les services externes (Stripe, Google Authenticator) seront disponibles et fonctionnels pendant toute la durée du projet.

Opérationnelles :

L'équipe de développement aura accès à toutes les ressources nécessaires (documentations, outils, etc.) sans restriction.

Les membres de l'équipe pourront se coordonner efficacement malgré les éventuelles contraintes de temps ou de distance.

Commerciales :

Les clients cibles (entreprises et professionnels) seront enclins à utiliser une plateforme en ligne pour souscrire aux services de Cyna-IT.

La plateforme répondra aux attentes des utilisateurs en termes d'UX/UI et de fonctionnalités.

2.4 Facteurs clés de succès

Les éléments suivants détermineront le succès du projet :

Expérience utilisateur (UX) :

Une interface intuitive, fluide et responsive sur tous les appareils.

Des parcours utilisateurs optimisés (navigation, paiement, gestion de compte).

Sécurité :

Protection renforcée des données utilisateurs et des transactions financières.

Mise en place de l'authentification à deux facteurs (MFA) pour les comptes sensibles.

Respect des délais :

Livraison des fonctionnalités principales dans les temps impartis.

Respect du planning prévisionnel pour chaque phase du projet.

Satisfaction client :

Réponse aux besoins exprimés par Cyna-IT (digitalisation de la vente, accessibilité, sécurité).

Feedback positif des utilisateurs tests sur la facilité d'utilisation et les fonctionnalités.

Performance technique :

Une plateforme stable, rapide et scalable.

Intégration réussie des services externes (Stripe, Google Authenticator).

Collaboration d'équipe :

Bonne coordination et communication entre les membres.

Répartition efficace des tâches selon les compétences de chacun.

3.1 Justification du projet

Le projet consiste à concevoir une plateforme e-commerce pour la vente de solutions de cybersécurité SaaS (SOC, EDR, XDR) proposées par la société Cyna. Historiquement orientée vers un modèle commercial traditionnel (réseau de vente et prospection directe), Cyna souhaite aujourd'hui se positionner sur le canal digital, devenu incontournable.

La création de cette plateforme répond à plusieurs enjeux :

- Modernisation de l'expérience client avec une interface intuitive, mobile-first.*
- Automatisation du processus de vente via des abonnements en ligne.*
- Accessibilité internationale sans contrainte physique.*
- Réduction des coûts d'intermédiation, via la suppression d'étapes commerciales manuelles.*

Ce projet est également une opportunité pour les 4 étudiants impliqués de mener un projet technique complet et professionnalisant : développement fullstack, gestion de projet Agile, sécurité, UX/UI, API, architecture logicielle et documentation technique.

3.2 Bénéfices attendues

Pour l'entreprise :

- - Augmentation des ventes grâce à un canal e-commerce disponible 24h/24.
- - Ouverture à un public international sans besoin de démarchage physique.
- - Automatisation des abonnements et de la gestion des services SaaS.
- - Suivi analytique des ventes via un backoffice et des tableaux de bord.

Pour les utilisateurs :

- - Accès simplifié aux produits Cyna et souscription directe en ligne.
- - Expérience fluide, pensée mobile-first.
- - Paiement sécurisé et gestion autonome de leur compte.
- - Assistance rapide grâce à un chatbot intégré.

Pour l'équipe projet :

- - Application concrète des compétences en développement web et mobile.
- - Travail collaboratif en environnement Agile.
- - Réalisation d'une documentation technique et d'une architecture complète.
- - Entraînement à la livraison d'un produit fonctionnel dans un cadre semi-professionnel.

3.3 Coûts prévisionnel

Étant un projet académique réalisé par une équipe de 4 étudiants en 3 mois, les coûts ne sont pas financiers mais en temps et en ressources techniques.

Poste	Détail	Estimation
Temps de travail	4 personnes × ~300h = 1 200 heures de développement	≈ 0 €
Hébergement / Déploiement	Services gratuits ou éducation (Vercel, Firebase, GitHub, etc.)	0 à 100 €
Outils de design	Figma, ressources libres	0 €
Systèmes de paiement (test)	Intégration via API test Stripe/PayPal	0 €
Documentation & tests	Environ 50 h de rédaction technique & tests	Temps inclus

4.1 L'architecture technique

L'architecture technique du projet repose sur une structure web moderne et modulaire, conçue pour assurer la fiabilité, la sécurité et la scalabilité du système. Elle s'appuie sur une séparation claire entre le frontend, le backend et la base de données, avec une communication centralisée via une API REST

Architecture globale

La plateforme est constituée des composants suivants :

Client (utilisateur ou administrateur)

-Accès via navigateur web ou application mobile

Frontend

-Développé en React pour le site web, React Native pour l'application mobile

-Communique avec l'API via des requêtes HTTP sécurisées

Backend (API REST)

-Développé avec Node.js

-Gère la logique métier, l'authentification, les paiements et la communication avec la base de données

Base de données relationnelle (SQL)

-Stockage des utilisateurs, commandes, produits, paiements, etc.

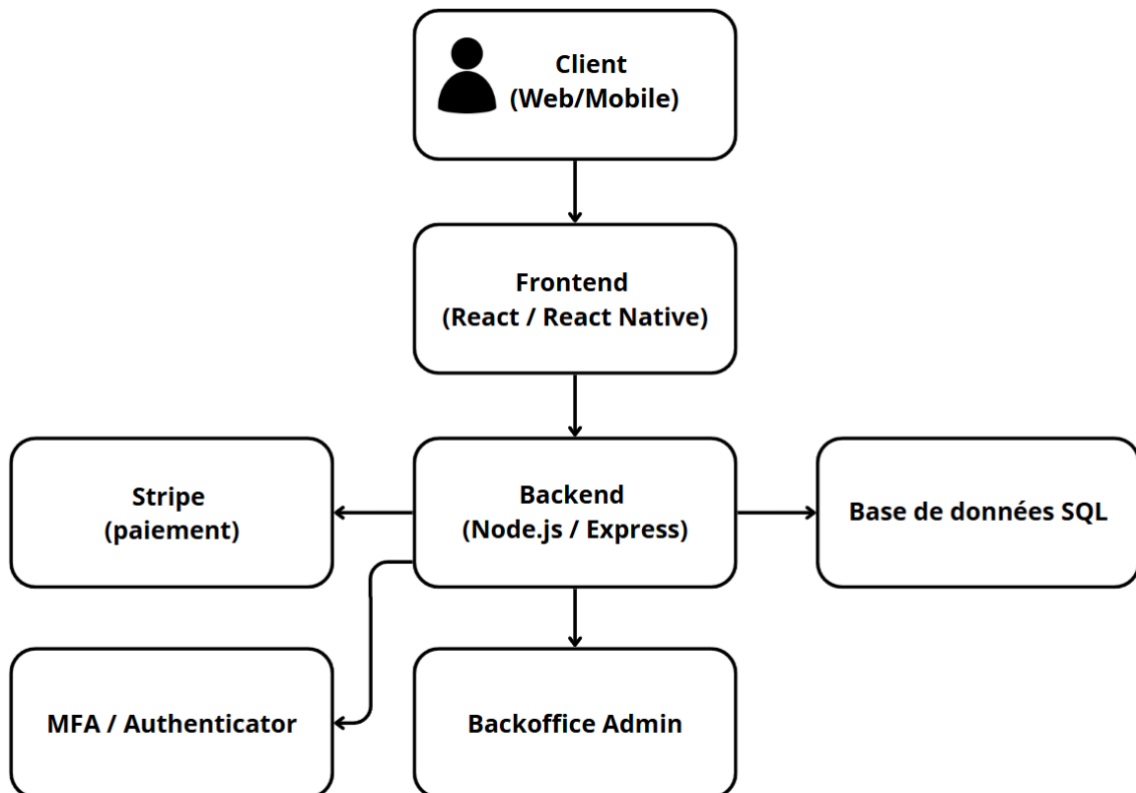
-Structure normalisée et sécurisée

Services externes intégrés

-Stripe : pour la gestion des paiements

-Google Authenticator : pour l'authentification à deux facteurs (MFA)

4.2 Schéma d'architecture



4.3 Modèle de données

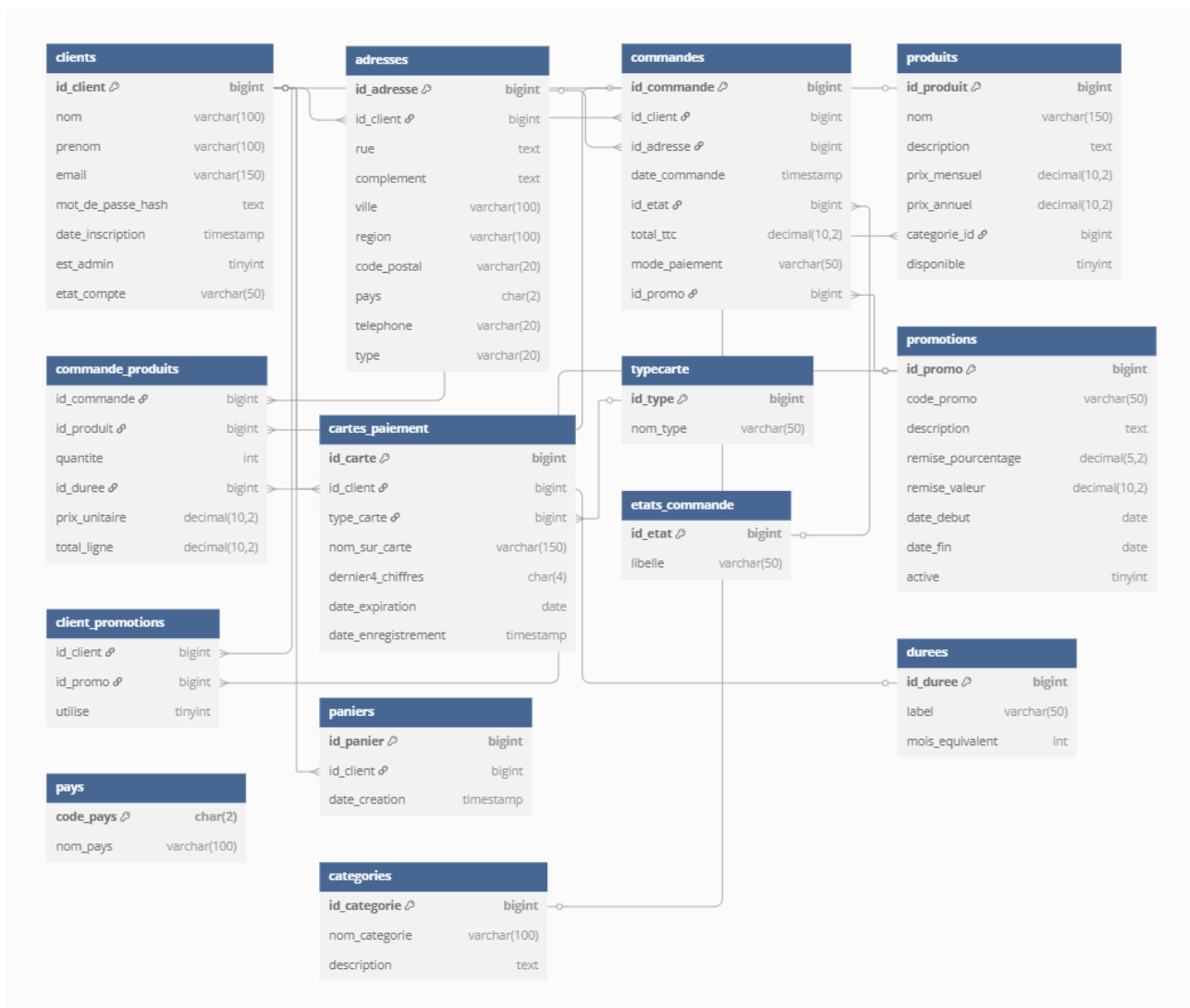
La base de données du projet est conçue pour structurer efficacement l'ensemble des informations manipulées par la plateforme. Elle repose sur un modèle relationnel clair, orienté vers la performance, la cohérence et la sécurité des données.

Objectifs de la base de données

Centraliser les informations essentielles (utilisateurs, produits, commandes, paiements, etc.)

Garantir l'intégrité des données via des relations bien définies

Optimiser les échanges entre le backend et la base pour assurer réactivité et fiabilité



Principales entités

<u>Entité</u>	<u>Rôle dans le système</u>
Utilisateur	Stocke les infos client ou admin : nom, email, mot de passe, rôle
Adresse	Adresse(s) liée(s) à un utilisateur (facturation/livraison)
Commande	Représente un achat réalisé (statut, date, total)
Produit	Service SaaS proposé par Cyna-IT (prix, durée, description)
Paielement	Paielement lié à une commande (montant, statut, méthode)
Promotion	Réduction appliquée sur une commande
Catégorie	Catégorie de produit (EDR, SOC, etc.)
EtatCommande	Statut de traitement (en cours, validée, annulée...)
Durée	Permet de gérer la période associée à un produit (mensuel, annuel, etc.).
TypeCarte	Type de carte utilisé pour le paiement (Visa, Mastercard, etc.)

4.4 Technologie choisies et justifications

<u>Technologie</u>	<u>Usage</u>	<u>Justification</u>
React	Frontend de l'application web	Framework moderne, rapide et maintenable. Composants réutilisables, UX fluide.
React Native	Développement de l'application mobile	Permet un code partagé entre Android et iOS. Intégration native simple.
Node.js et Express	Backend / API REST	Léger, rapide, performant pour les I/O. Large écosystème.
SQL	Base de données relationnelle	Parfaitement adapté aux données structurées
Sequelize	Communication avec la base depuis Node.js	Facilite les requêtes, sécurise les échanges, évite les injections SQL.
Stripe	Paiement en ligne sécurisé	Référence mondiale en paiement SaaS. Intégration simple et fiable.
Google authentification	Authentification à deux facteurs	Renforce la sécurité des comptes.
Json Web Token	Gestion des sessions / authentification	Authentification stateless, sécurisée, facilement intégrée dans une API REST.
GitHub	Versioning et collaboration	Suivi de l'évolution du projet, travail en équipe facilité.

4.5 Sécurité et authentification

La sécurité est un enjeu central du projet, en particulier pour une plateforme SaaS manipulant des données sensibles (informations personnelles, moyens de paiement, historiques de commandes, etc.). Plusieurs dispositifs techniques sont mis en œuvre pour garantir la confidentialité, l'intégrité et la sécurité des données.

Authentification

L'accès à la plateforme est sécurisé par une authentification par email et mot de passe, avec les pratiques suivantes :

- *Hashage des mots de passe (via bcrypt), pour éviter toute conservation en clair.*
- *Vérification des identifiants via l'API backend, sans exposition des données sensibles.*
- *Gestion des sessions avec JWT (JSON Web Tokens), permettant une authentification sécurisée, stateless et compatible avec les APIs REST.*

Pour les administrateurs et les accès sensibles, une authentification à deux facteurs (MFA) est mise en place à l'aide de Google Authenticator. Ce second facteur ajoute une couche de sécurité supplémentaire pour prévenir les accès non autorisés, même en cas de vol de mot de passe.

Paielements sécurisés

Les paiements en ligne sont traités par l'intermédiaire de Stripe, une solution de paiement externe reconnue pour sa fiabilité. Les données bancaires ne transitent jamais par nos serveurs, limitant ainsi les risques de compromission.

Stripe prend en charge :

- *Le chiffrement des données de paiement,*
- *La conformité PCI DSS,*
- *Les vérifications anti-fraude automatiques.*

Sécurisation des échanges

Tous les échanges entre le client, l'API et la base de données sont sécurisés par HTTPS (TLS 1.2+), assurant le chiffrement des données en transit.

Des vérifications supplémentaires sont mises en place :

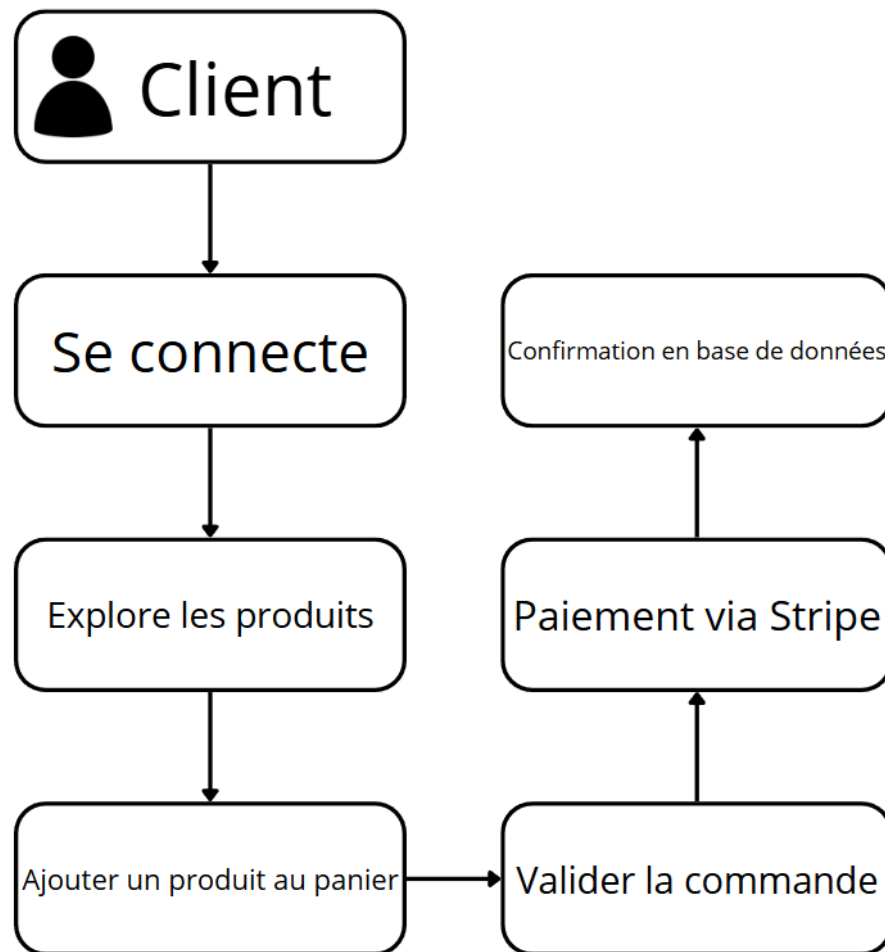
- Validation côté client et serveur *des champs de formulaire,*
- Filtrage des entrées *pour prévenir les injections SQL et attaques XSS,*
- Restrictions d'accès aux routes sensibles *côté backend (middleware d'autorisation).*

Surveillance et résilience

Des outils de log et de monitoring (type Winston, ou services cloud) pourront être intégrés pour :

- *Suivre les accès,*
- *Identifier les comportements suspects,*
- *Anticiper d'éventuelles failles ou anomalies.*

4.6 Diagrammes techniques



5.1 Méthodologie

Le projet a été conduit en suivant une **méthodologie Agile**, plus précisément le **cadre Scrum**, afin de garantir une organisation souple, une bonne répartition des tâches et une capacité d'adaptation rapide aux imprévus.

Pourquoi Scrum ?

- Adapté aux projets avec des délais courts (ici 3 mois),
- Permet une progression par itérations (sprints),
- Favorise la collaboration et la communication au sein de l'équipe,
- Met l'accent sur la livraison rapide de fonctionnalités opérationnelles.

Mise en œuvre dans le projet :

Le projet a été découpé en **6 sprints de 2 semaines** chacun. À chaque sprint, des fonctionnalités précises ont été développées, testées et validées avant de passer à la suite.

Activités mises en place :

- **Sprint planning** : pour planifier les tâches de chaque sprint.
- **Réunions de suivi** : pour échanger sur l'avancement, les blocages et les priorités.
- **Sprint review** : à la fin de chaque sprint, pour présenter les avancées fonctionnelles.
- **Sprint retrospective** : pour discuter de ce qui a bien fonctionné ou non, et s'améliorer dans le sprint suivant.
- **Backlog** géré collectivement et réajusté à chaque sprint.

Outils utilisés :

- **Trello** pour la gestion des tâches,
- **GitHub** pour le versioning et la collaboration sur le code,
- **Discord** pour la communication et les réunions de suivi

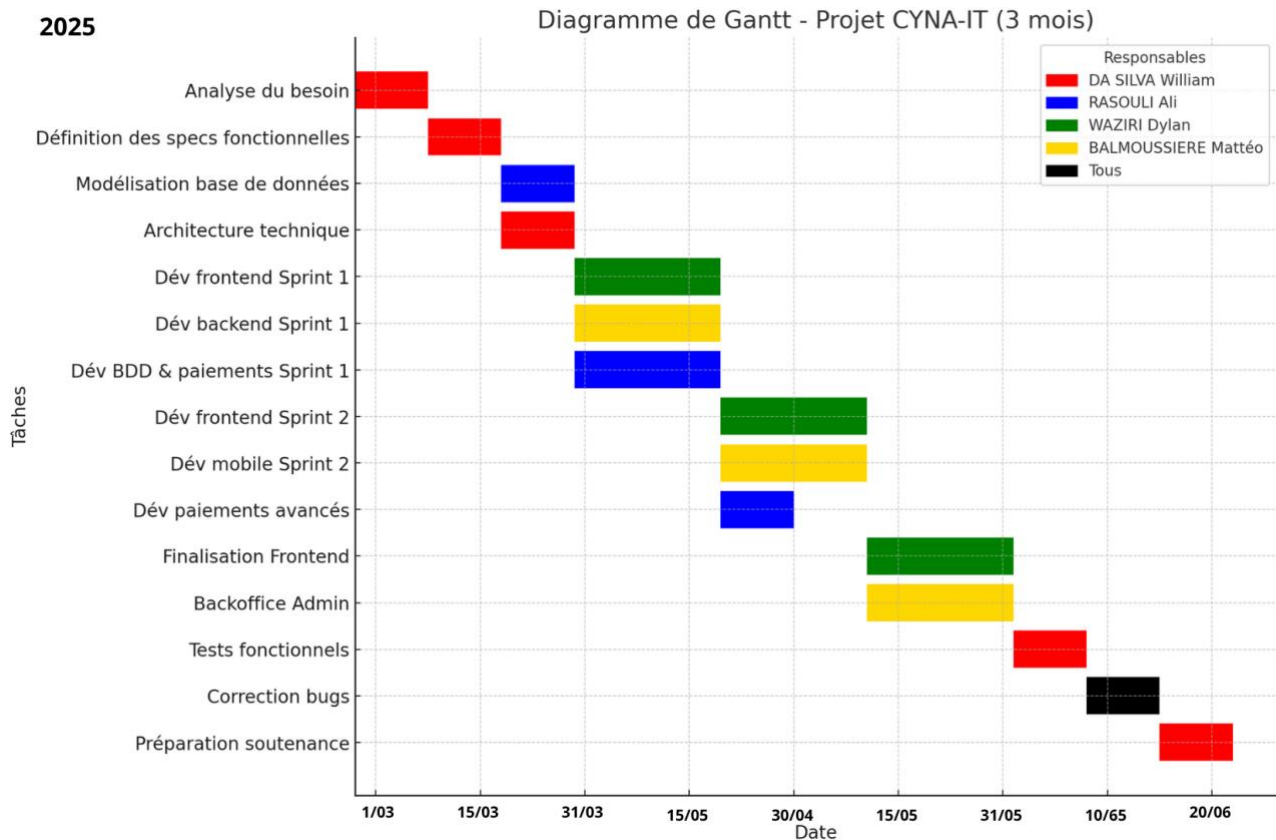
5.2 Planning prévisionnel

Le projet était initialement prévu sur 8 mois, mais dans le cadre du calendrier pédagogique, nous avons eu 3 mois effectifs pour concevoir, développer et livrer un produit fonctionnel. Voici les principales tâches réparties par responsable et période :

N°	Tâche	Responsable	Dates	Objectif / Description
1	Analyse du besoin	DA SILVA William	01/03 → 08/03	Comprendre les attentes du client et établir les grandes lignes du projet
2	Définition des specs fonctionnelles	DA SILVA William	08/03 → 15/03	Rédiger les fonctionnalités principales à implémenter
3	Modélisation base de données	RASOULI Ali	15/03 → 22/03	Créer le MCD/MPD et structurer les entités clés
4	Architecture technique	DA SILVA William	15/03 → 22/03	Définir l'organisation frontend/backend/API/base
5	Dév frontend Sprint 1	WAZIRI Dylan	22/03 → 05/04	Développement de l'interface client : structure et navigation
6	Dév backend Sprint 1	BALMOUSSIERE Mattéo	22/03 → 05/04	Mise en place des routes API REST et de la logique métier
7	Dév BDD & paiements Sprint 1	RASOULI Ali	22/03 → 05/04	Connexion BDD + début intégration Stripe
8	Dév frontend Sprint 2	WAZIRI Dylan	05/04 → 19/04	Ajout de fonctionnalités UX/UI : panier, espace client
9	Dév mobile Sprint 2	BALMOUSSIERE Mattéo	05/04 → 19/04	Développement de l'application mobile (React Native)
10	Dév paiements avancés	RASOULI Ali	05/04 → 12/04	Finalisation du module de paiement + MFA
11	Finalisation Frontend	WAZIRI Dylan	19/04 → 03/05	Derniers écrans + responsive complet
12	Backoffice Admin	BALMOUSSIERE Mattéo	19/04 → 03/05	Interface de gestion pour administrateurs
13	Tests fonctionnels	DA SILVA William	03/05 → 10/05	Vérifications des fonctionnalités, tests croisés
14	Correction bugs	Toute l'équipe	10/05 → 17/05	Correction des bugs identifiés lors des tests
15	Préparation soutenance	DA SILVA William	17/05 → 24/05	Slides, entraînement à la présentation orale

5.3 Gantt Project

2025



5.4 Livrables attendus

Pour ce projet de Bachelor, plusieurs livrables seront remis afin de présenter le travail, aussi bien sur le plan technique que documentaire.

Livrables documentaires :

Document de cadrage/Charte de projet

Slides de soutenance pour la présentation orale.

Livrables techniques :

Code source complet de la plateforme

Base de données SQL

Documentation technique :

Instructions d'installation et de lancement du projet,

Explication de l'architecture et des API principales.

5.5 Suivi des risques

Le suivi des risques s'est fait de manière continue tout au long du projet, notamment lors des réunions de sprint. Voici les principaux risques identifiés :
Risque Impact Mesure mise en place Retard dans le développement Élevé Sprints courts + planification stricte Difficultés techniques (authentification, Stripe) Moyen Documentation, entraide, tutoriels Mauvaise répartition des tâches Moyen Attribution claire par compétences Conflits Git / versioning Faible Respect des branches et des PR Perte de motivation ou indisponibilité Moyen Réunions régulières pour maintenir le rythme

6.1 Équipe projet

L'équipe est composée de quatre membres, chacun ayant contribué au projet selon ses compétences techniques et ses appétences personnelles.

DA SILVA William – Chef de projet

Responsable de la coordination générale, de la gestion du planning, des livrables documentaires, et du suivi des tâches. Il a également assuré un rôle de facilitateur et a proposé des axes d'amélioration continue tout au long du projet.

RASOULI Ali – Développeur base de données & paiements

En charge de la modélisation de la base de données (MCD/MPD, script SQL), il a également intégré le module de paiement Stripe, garantissant un processus de transaction sécurisé.

WAZIRI Dylan – Développeur frontend

Spécialiste de la partie interface utilisateur web, il a conçu un frontend responsive selon les bonnes pratiques UX/UI et a travaillé en lien avec l'API pour connecter les fonctionnalités.

BALMOUSIERE Mattéo – Développeur mobile & backoffice

En charge du développement de l'application mobile (React Native), il a également conçu le backoffice d'administration, permettant à l'entreprise de gérer ses produits, utilisateurs et commandes.

6.2 Répartition des rôles

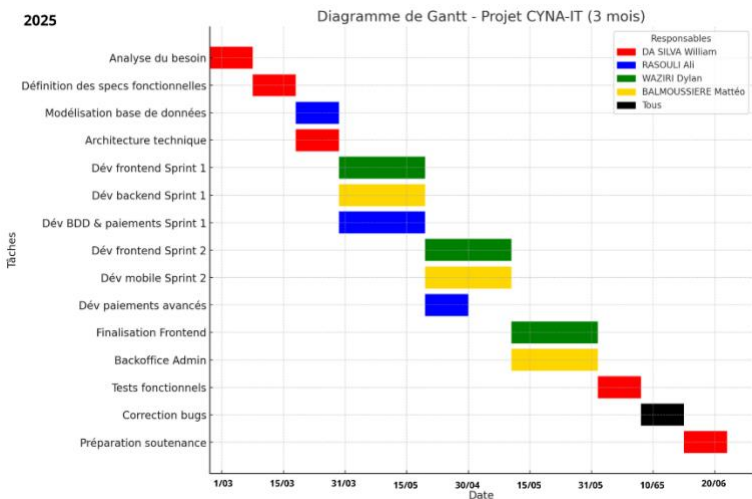
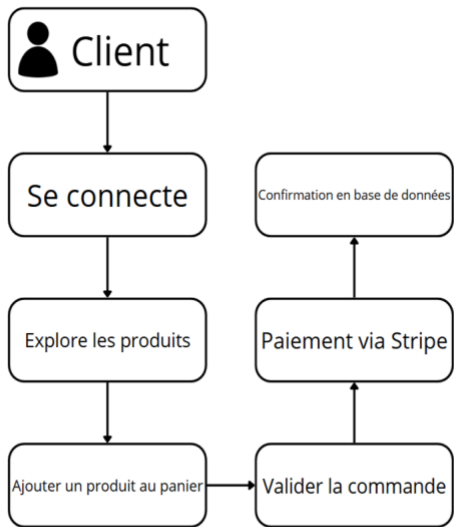
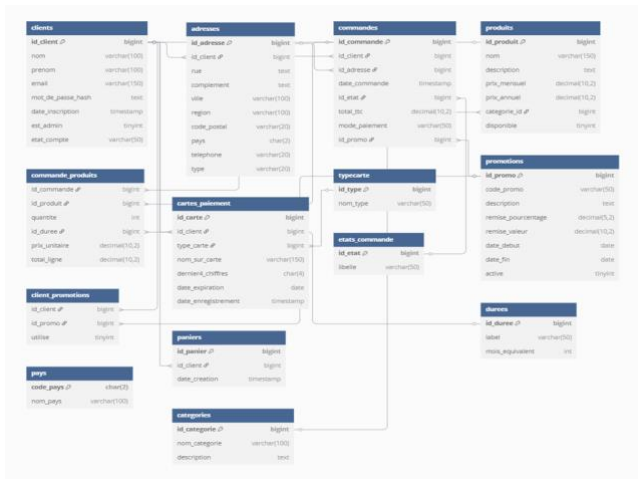
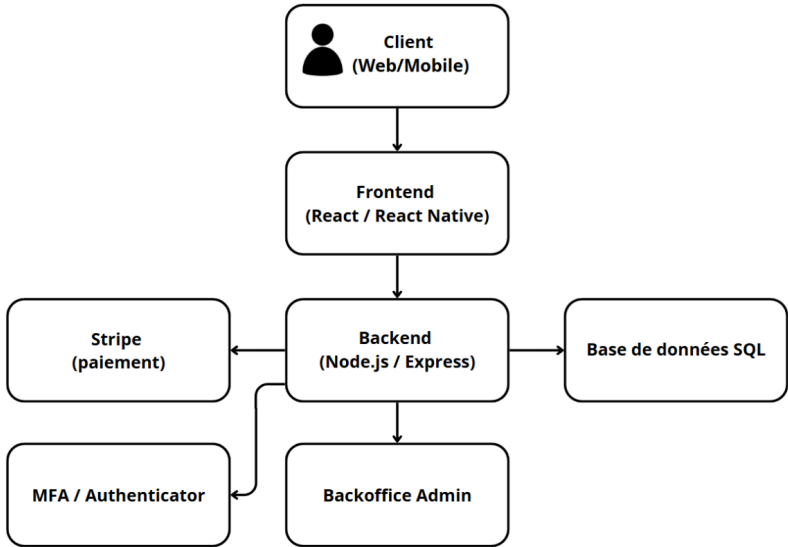
Membre de l'équipe	Rôle principal	Tâches clés
William DA SILVA	Chef de projet	Planification, gestion des livrables, coordination, suivi de l'équipe, documentation
Ali RASOULI	Développeur base de données & paiements	Modélisation MCD/MPD, création de la base de données, intégration Stripe
Dylan WAZIRI	Développeur frontend	Développement de l'interface web utilisateur, intégration des pages, responsive design
Mattéo BALMOUSSIERE	Développeur mobile & backoffice	Développement de l'application mobile, création du backoffice, gestion mobile-first

6.3 Instances de gouvernance

Le projet a été mené selon une méthodologie Agile/Scrum, avec une organisation souple mais structurée. Les instances de gouvernance suivantes ont été mises en place :

- Réunions de lancement : pour définir les objectifs, le périmètre, les priorités, et les outils de collaboration.
- Daily meetings (hebdomadaires dans notre cas) : pour suivre l'avancement, discuter des blocages et réajuster les tâches.
- Réunions de sprint review : à la fin de chaque phase majeure (démonstration, test, validation).
- Pilotage documentaire : sous la responsabilité du chef de projet, avec centralisation des livrables sur GitHub ou un drive commun.
- Communication interne : assurée via Discord et Trello (ou outil équivalent) pour le suivi des tâches et la répartition de la charge de travail.

7.1 Maquettes ou Schémas



7.2 Lexique

Terme / Acronyme	Définition
SaaS	Software as a Service : modèle de distribution de logiciel accessible via Internet, sans installation locale.
B2B	Business to Business : désigne les relations commerciales entre entreprises.
UX / UI	User Experience / User Interface : expérience utilisateur et interface visuelle, centrées sur la facilité d'usage et l'esthétique.
MCD / MPD	Modèle Conceptuel de Données / Modèle Physique de Données : schémas de la base de données définissant les entités et leurs relations.
API REST	Interface de programmation basée sur le protocole HTTP permettant la communication entre le frontend et le backend.
MFA	Multi-Factor Authentication : authentification à plusieurs facteurs, ici via Google Authenticator.
JWT	JSON Web Token : standard sécurisé pour transmettre des informations d'authentification entre client et serveur.
Stripe	Plateforme de paiement en ligne sécurisée utilisée pour gérer les transactions financières.
React / React Native	Framework JavaScript pour le développement d'interfaces web (React) et mobiles (React Native).
Node.js / Express	Environnement et framework JavaScript pour le développement du backend (API).
SQL	Structured Query Language : langage utilisé pour manipuler des bases de données relationnelles.

Sequelize	ORM (Object-Relational Mapping) JavaScript pour interagir avec une base SQL depuis Node.js.
Trello	Outil de gestion de projet en ligne basé sur le système de tableaux et de cartes.
GitHub	Plateforme de versioning et de collaboration autour du code, basée sur Git.
Discord	Application de communication utilisée pour coordonner l'équipe projet.
Sprint	Itération de développement dans Scrum, durant laquelle une partie fonctionnelle du produit est développée et livrée.
Backoffice	Interface d'administration d'un site web, permettant de gérer les contenus, utilisateurs, commandes, etc.
Responsive Design	Conception d'une interface qui s'adapte automatiquement à toutes les tailles d'écran (PC, tablette, mobile).
RGPD	Règlement Général sur la Protection des Données : législation européenne sur la confidentialité et la protection des données personnelles.
PCI DSS	Payment Card Industry Data Security Standard : normes de sécurité pour le traitement des paiements par carte bancaire.