



Cahier des charges

Refonte de l'application

1. Contexte

Besoins du client	p.2
Objectifs du projet.....	p.2
Mesure de la réussite du projet.....	p.3
Contraintes réglementaires	p.3
Éléments hors périmètre	p.4

2. Description fonctionnelle

Gestion des comptes utilisateurs.....	p.4
Gestion des livraisons.....	p.5
Gestion de la facturation	p.5
Connexion et déconnexion.....	p.5
Diagramme de cas, diagramme fonctionnel	p.6
Impact social et environnemental	p.7

3. Description technique

Technologies frontend.....	p.8
Technologies backend	p.8
Base de données	p.8
Infrastructure et déploiement.....	p.9
Outils de développement et qualité	p.9
Diagramme de l'architecture de l'application	p.10

1. Contexte du projet

a. Besoins du client

Présentation de l'activité de l'entreprise et de la solution existante :

LiVrai est une société de 150 employés spécialisée dans la livraison de marchandises en grande quantité pour les professionnels. L'entreprise utilise actuellement une application CRM développée il y a plusieurs années pour gérer ses clients et les livraisons. Cette application nécessite l'intervention du service commercial pour établir un contrat de collaboration et créer un compte client, qui permet ensuite aux clients de réserver des livraisons.

Limites de la solution existante :

Avec la forte croissance de l'activité de *LiVrai*, le processus actuel, qui nécessite systématiquement l'intervention du service commercial pour la création de comptes clients, ralentit le développement de l'entreprise. Cette dépendance allonge les délais d'inscription des nouveaux clients et complique leur intégration. Par ailleurs, l'application existante ne leur offre pas une autonomie suffisante pour gérer leurs commandes de livraison.

b. Objectifs du projet

Description des objectifs du projet :

1. Développer une nouvelle application permettant aux clients de créer leur compte de manière autonome et de gérer l'ensemble du cycle de livraison sans intervention obligatoire du service commercial.
2. Maintenir la possibilité pour le service commercial de gérer certains clients spécifiques nécessitant un accompagnement personnalisé.
3. Améliorer l'expérience utilisateur en proposant une interface moderne et interactive, sous forme Single Page Application.
4. Optimiser les processus de commande, suivi et facturation des livraisons.
5. Faciliter la gestion interne des livraisons par le service dédié.
6. Créer une solution évolutive capable d'accompagner la croissance continue de *LiVrai*.
7. Mettre en place une application ergonomique, performante et sécurisée.



c. Mesure de la réussite du projet

Une analyse de différents indicateurs après 6 mois d'utilisation de l'application permettra de mesurer efficacement la réussite du projet.

Indicateurs quantitatifs :

- Utilisation de la plateforme par les clients sans assistance après leur inscription. (>70%)
- Réduction du temps de création de compte client par rapport au processus actuel. (-50%)
- Réduction significative du temps de traitement des commandes. (-50%)
- Augmentation du nombre de commandes traitées sans intervention du commercial. (+50%)
- Réduction du temps consacré par le commercial à la gestion administrative des comptes. (-20%)
- Taux de satisfaction client augmenté (>80%).

Indicateurs qualitatifs :

- Retour positif des utilisateurs sur l'utilisation de la plate-forme.
- Feedback positif des équipes commerciales concernant la gestion des clients.
- Retours positifs du service Livraisons sur l'organisation et le suivi des livraisons.

d. Contraintes réglementaires

RGPD (Règlement Général sur la Protection des Données) :

- Mise en place d'une politique de confidentialité claire et accessible.
- Recueil du consentement explicite des utilisateurs pour la collecte et le traitement de leurs données.
- Implémentation de fonctionnalités permettant aux utilisateurs d'accéder, de modifier ou de supprimer leurs données personnelles.
- Sécurisation des données via des protocoles de chiffrement standards (HTTPS, SSL/TLS).
- Mise en place d'une procédure de surveillance des intrusions.
- Gestion et stockage des données personnelles selon les normes européennes.

Facturation électronique :

- Conformité avec les obligations comptables et fiscales. Norme EN16931, article L441-9 du CC, article 289 V et 289 bis du CGI, Décret 2022-1299 (7/10/2022).
- Stockage des factures pendant la durée légale requise. (10 ans -L123-22 du CC)

Sécurité des données :

- Utilisation des protocoles HTTPS, sécurisation des connexions avec TLS.
- Chiffrement des données sensibles (mots de passe, informations financières, ...) avec un algorithme robuste (AES-256), hachage des mots de passe (BCrypt, PBKDF2), et stockage des clés de chiffrement sécurisé.
- Authentification robuste, utilisation du MFA pour renforcer la sécurité.
- Implémentation de JWT pour une authentification stateless des requêtes.
- Définition des niveaux d'accès pour les différents rôles (client, service commercial, service livraison, administrateur)
- Audits de sécurité réguliers, journalisation des actions critiques.

Accessibilité numérique (normes WCAG 2.1 AA) :

- Perceptibilité : contraste suffisant (4.5 :1), texte alternatif de description.
- Utilisabilité : navigation clavier fluide, temps d'interaction suffisant.
- Compréhensibilité : libellé clair des liens et boutons, messages d'erreur informatifs, navigation cohérente.
- Robustesse : application utilisable sur toutes tailles d'écran, tout navigateur et appareil.

e. Éléments hors périmètre

Dans cette nouvelle version de l'application, certains points sont à exclure pour conserver une cohérence et une taille raisonnable, mais ces fonctionnalités pourraient faire l'objet d'un développement ultérieur. La liste n'est pas exhaustive.

- Développement d'une application mobile native (une version responsive web sera privilégiée dans cette phase).
- Intégration avec les systèmes de gestion de stock des clients.
- Intégration d'un système de messagerie entre client et commercial.
- Tableau de bord pour l'affichage de statistiques (délais, volumes, évolutions...)
- Système de paiement en ligne intégré (la facturation sera gérée via le système existant, paiement en ligne à envisager pour une future évolution).
- Place de marché pour mise en relation avec d'autres transporteurs.
- Traduction de l'application en d'autres langues (phase initiale uniquement en français, le site pourrait évoluer vers le multilingue).
- Gestion des entrepôts et stocks (non intégré dans cette application).

2. Description fonctionnelle

a. Liste des fonctionnalités

1. Gestion des comptes utilisateurs

- **Création de compte client autonome**
 - Inscription en ligne par les clients avec validation par email.
 - Formulaire de saisie des informations obligatoires (dénomination, coordonnées, adresse facturation et livraison, numéro SIRET et TVA, etc.).
 - Acceptation des conditions générales de service.

Autonomisation des clients et réduction de la charge du service commercial.

- **Création de compte client par le service commercial**
 - Interface dédiée pour la création de comptes par les commerciaux.
 - Possibilité d'appliquer des conditions commerciales spécifiques.

Maintien d'un accompagnement personnalisé pour certains clients.

- **Gestion des profils utilisateurs**
 - Modification des informations du profil par l'utilisateur ou les commerciaux.
 - Gestion des droits d'accès par profil.
 - Différents rôles utilisateurs (client, service commercial, service livraison).

Gestion des informations client mentionné dans la fiche descriptive.

2. Gestion des livraisons

- **Commande de livraison**
 - Formulaire de demande de livraison.
 - Saisie des adresses de collecte et de livraison.
 - Estimation du coût en fonction du volume, poids et distance.

Gestion des commandes de livraison mentionné dans la fiche descriptive.
- **Suivi des livraisons**
 - Tableau de bord présentant l'état des livraisons en cours.
 - Suivi en temps réel du statut des livraisons.
 - Notifications par email ou sms des étapes (confirmation, prise en charge, livraison).
 - Historique détaillé des statuts et des événements liés à chaque livraison.
 - Vue chronologique de toutes les livraisons effectuées.
 - Filtres par date, statut, client, etc.

Suivi des livraisons mentionné dans la fiche descriptive.
- **Gestion des livraisons par le service Livraisons**
 - Interface d'administration pour la planification et l'affectation des livraisons.
 - Outils de gestion des incidents et retards.
 - Vue consolidée des livraisons du jour/de la semaine.

Gestion des opérations par le service Livraisons.

3. Gestion de la facturation

- **Génération automatique des factures**
 - Création des factures après validation de livraison.
 - Application des tarifs et remises selon le profil client.
 - Export au format PDF.

Gestion de la facturation mentionné dans la fiche descriptive.
- **Administration de la facturation**
 - Interface dédiée pour le service commercial et le service Livraisons.
 - Possibilité d'ajustement manuel des factures.

Accès à la facturation par les services internes.

4. Connexion et déconnexion

- **Authentification, persistance de session, déconnexion**
 - Connexion des utilisateurs (email, mot-de-passe, MFA) et génération de JWT.
 - Stockage sécurisé du token JWT, rafraîchissement, vérifications.
 - Gestion des permissions et redirections pour les pages à accès restreint.
 - Déconnexion avec suppression du token JWT.

b. Fonctionnement de l'application

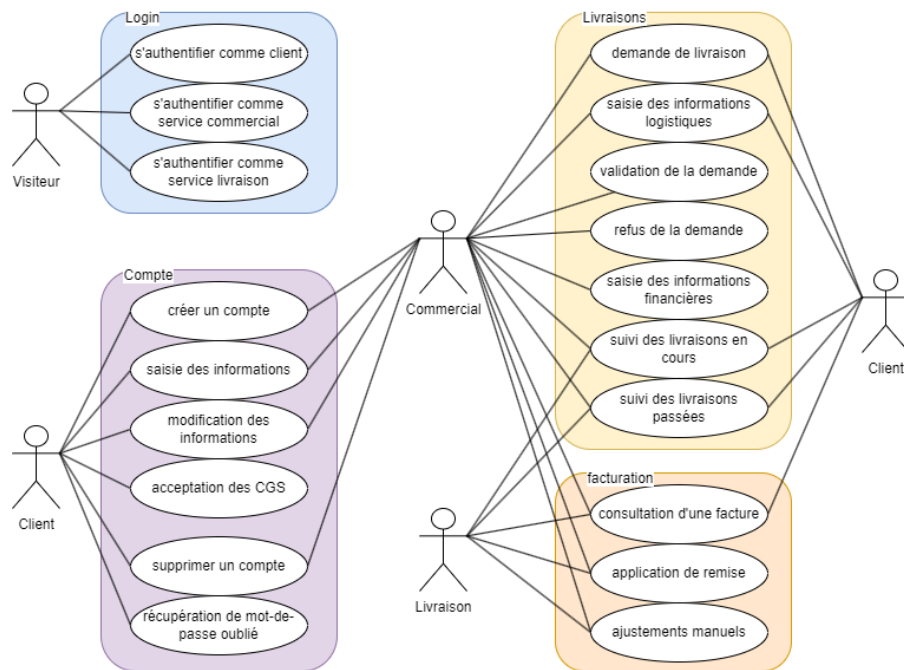


Diagramme de cas d'utilisation

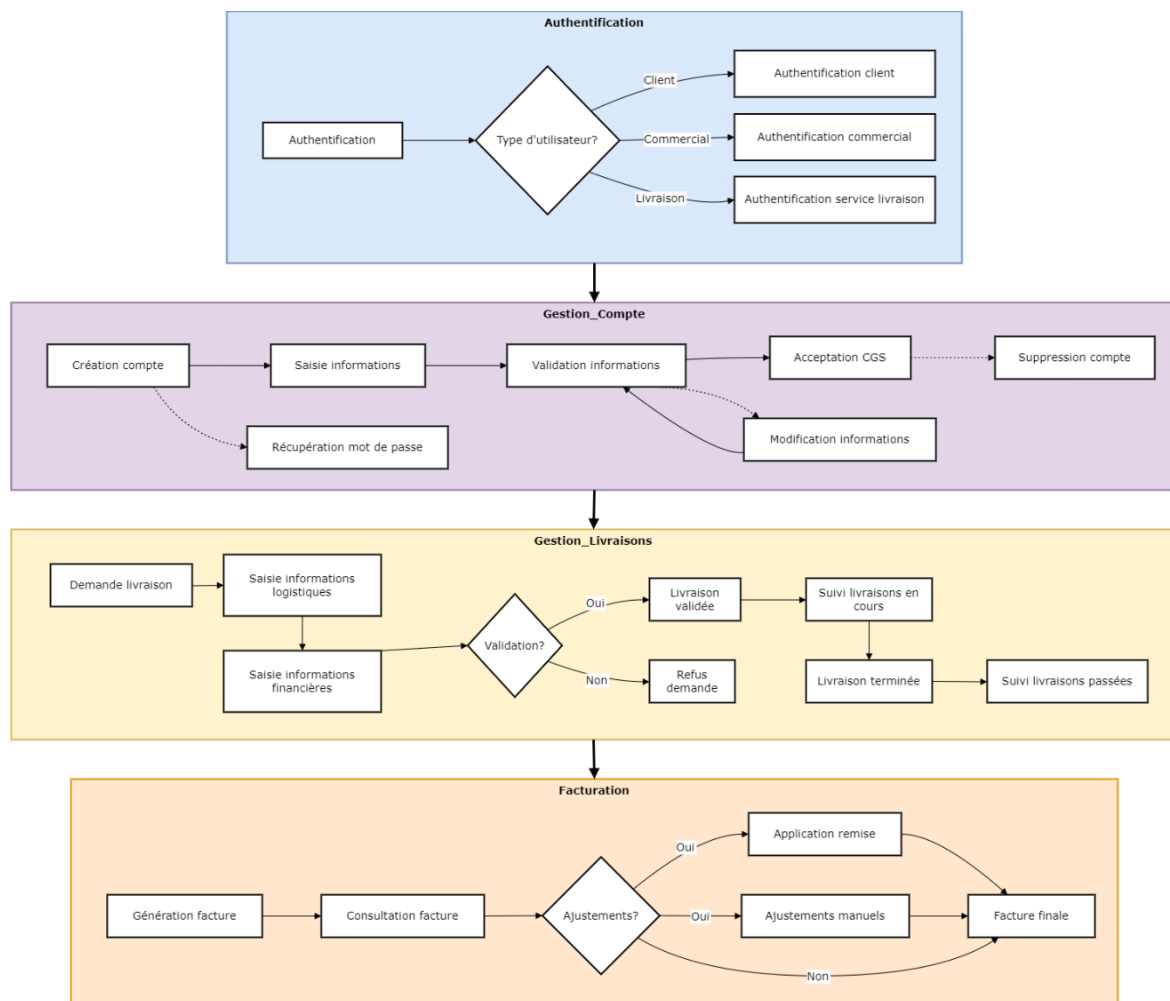


Diagramme fonctionnel

c. Impact Social

Impact environnemental

- **Réduction de la consommation de papier** : La dématérialisation des documents (factures, bons de livraison) permettra de réduire significativement l'utilisation de papier.
- **Optimisation des trajets de livraison** : En facilitant l'accès aux demandes par le service livraison, l'application contribuera à l'optimisation des itinéraires et à la réduction des émissions de CO₂.
- **Réduction des déplacements des commerciaux** : en centralisant et automatisant les tâches, la nécessité de déplacement physique pour la gestion clients sera limitée.

Accessibilité et impact social

- **Conformité WCAG 2.1 niveau AA** : L'application respectera les standards d'accessibilité web pour garantir son utilisation par tous, y compris les personnes en situation de handicap.
- **Interface adaptative** : Conception responsive permettant l'utilisation sur différents types d'appareils et tailles d'écran.
- **Compatibilité avec les technologies d'assistance** : Support des lecteurs d'écran, navigation au clavier, et autres outils d'assistance pour utilisateurs mal- ou non-voyants ou avec difficultés motrices.
- **Contrastes et typographie adaptés** : Utilisation de contrastes suffisants et de polices lisibles pour les personnes malvoyantes ou dyslexiques, amélioration du confort de lecture.
- **Messages d'erreur clairs** : Indication précise des erreurs et suggestions de correction, destinées aux utilisateurs rencontrant des difficultés cognitives ou maîtrisant peu le numérique, pour faciliter leur navigation et interaction avec l'application.
- **Réduction de la charge de travail du service commercial** : Permettant de se concentrer sur les clients à plus forte valeur et nécessitant un accompagnement spécifique.
- **Optimisation des livraisons** : Contribuant à une meilleure gestion des ressources logistiques et à un service client amélioré.



3. Description technique

Technologies Frontend

- **Angular 18** : Conformément aux exigences, l'application sera développée sous forme de SPA (Single Page Application) avec Angular qui offre :
 - Une structure modulaire facilitant la maintenance
 - Un système de routing performant pour une navigation fluide
 - Une gestion efficace des états avec les services et RxJS
 - Des performances optimisées grâce au mécanisme de détection des changements
- **TypeScript** : Langage de programmation fortement typé améliorant la qualité du code et facilitant la maintenance.
- **Angular Material** : Bibliothèque de composants UI basée sur Material Design, offrant :
 - Des composants accessibles et responsives
 - Une cohérence visuelle sur l'ensemble de l'application
 - Une implémentation facilitée des fonctionnalités courantes (tableaux, formulaires, etc..)
- **SCSS** : Préprocesseur CSS permettant une meilleure organisation et réutilisation du code de style.
- **NgRx** : Bibliothèque de gestion d'état pour les applications Angular complexes, inspirée de Redux.

Technologies Backend

- **Java 21 (LTS)** : Conformément aux exigences, le backend sera développé avec la dernière version LTS de Java, offrant :
 - Une compatibilité à long terme avec support et mises à jour de sécurité
 - Des performances optimisées
 - Une utilisation efficace des ressources
- **Spring Boot 3.2+** : Framework Java facilitant le développement d'applications robustes :
 - Configuration simplifiée
 - Intégration aisée des services et composants
 - Support intégré pour les API REST
 - Gestion des dépendances avec Maven
- **Spring Security** : Module de sécurité offrant :
 - Authentification et autorisation robustes (JWT ou OAuth2)
 - Protection contre les vulnérabilités web courantes
 - Gestion des JWT (JSON Web Tokens) pour l'authentification stateless
- **Hibernate/JPA** : ORM (Object-Relational Mapping) facilitant l'interaction avec la base de données.

Base de données

- **Base de données SQL** : Système de gestion de base de données relationnelle compatible avec les outils internes. (la base MySQL actuelle est cohérente avec l'application, migration possible vers PostgreSQL si besoin lors de montée en charge ou en complexité)

Infrastructure et déploiement

- **Docker** : L'application sera déployée dans des conteneurs Docker :
 - Isolation des composants
 - Reproductibilité des environnements
 - Facilité de déploiement et de mise à l'échelle
- **Hébergement** : Infrastructure compatible avec les plateformes Cloud courantes, le choix définitif n'étant pas encore arrêté.

Outils de développement et qualité

- **Git** : Système de contrôle de version pour le suivi des modifications du code.
- **Github Actions** : Intégration et déploiement continu (CI/CD).
- **SonarQube** : Analyse statique du code pour maintenir la qualité et détecter les vulnérabilités.
- **JUnit & Mockito** : Tests unitaires et d'intégration côté backend.
- **Jest** : Tests unitaires et d'intégration côté frontend.
- **Swagger/OpenAPI** : Documentation automatique des API REST.

Diagramme de l'architecture de l'application

Cette architecture modulaire assure :

- Une séparation claire des responsabilités
- Une scalabilité indépendante de chaque composant
- Une maintenance facilitée
- Une haute disponibilité
- Une sécurisation des échanges à tous les niveaux

Les Routes API REST pourraient être les suivantes : (Rôles : **Utilisateur/Commercial/Logistique**)

Methode	Route	Description	Auth	Rôle
POST	auth/login	Vérifie l'utilisateur et retourne un JWT	-	
GET	auth/me	Retourne les infos de l'utilisateur connecté	+	U
POST	account/register	Création d'un compte client	-	U/C
PUT	account/{id}	Mise à jour des informations du compte	+	U/C
GET	account/{id}	Récupère les informations d'un compte	+	U/C
DELETE	account/{id}	Supprime le compte	+	C
POST	delivery	Demande une nouvelle livraison	+	U/C
GET	delivery/{id}	Consulte une livraison	+	U/C/L
PUT	delivery/{id}/status	Met à jour le statut d'une livraison	+	C/L
GET	delivery	Liste des livraisons pour un client	+	U/C/L
POST	invoice	Création d'une nouvelle facture	+	C
PUT	invoice/{id}	Modification d'une facture existante	+	C/L
PUT	invoice/{id}/status	Modification du statut d'une facture	+	C/L
GET	invoice/{id}	Consultation d'une facture	+	U/C/L
GET	invoice	Liste des factures pour un client	+	U/C/L

Statut d'une livraison : En attente / Acceptée / Refusée / Terminée / Retardée

Statut d'une facture : Initiale / Active / Archivée

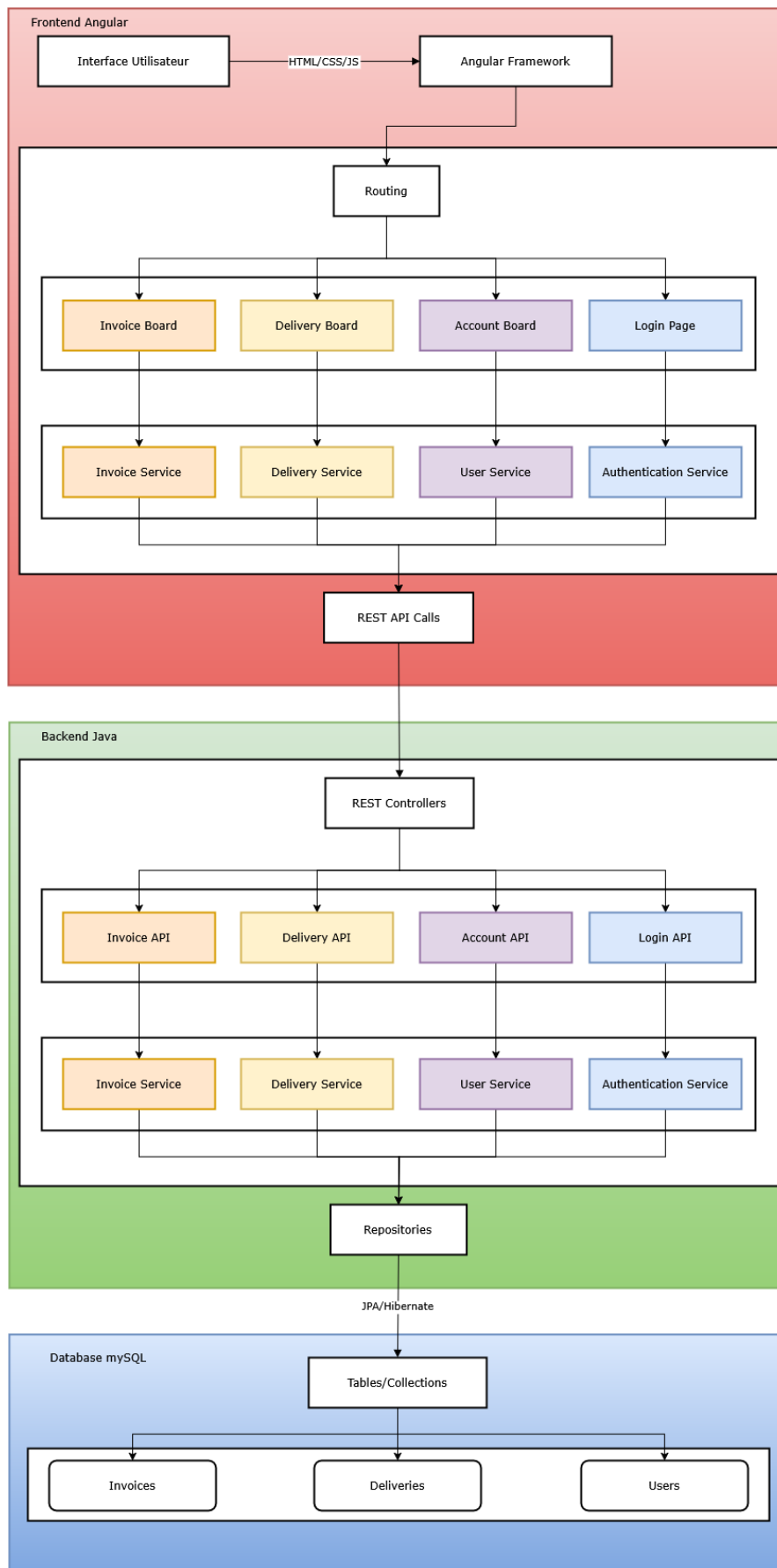


Diagramme de l'architecture modulaire de l'application LiVrai. Pour simplifier le diagramme, les controllers entre views et services dans le frontend ne sont pas présentés, de même que les DTO Mappers entre APIs et Services du backend.