



SOMMAIRE

Objet du document	1
Objectifs du projet	1
Principes d'architecture	1
Architecture existante	1
Architectures	2
Métier	3
De données	4
Technologiques	5
Justification de l'approche architecturale	5
Architectures de transition	5

Objet du document

Ce document permet de modéliser l'architecture qui sera réalisée par les équipes de développement. Au-delà de la modélisation graphique, il s'agit également d'énoncer les principes sur lesquels l'architecture s'appuie, de justifier cette approche architecturale, mais également d'indiquer des transitions si nécessaire.

Objectifs du projet

Concevoir une nouvelle application qui regroupe tous les clients à travers le monde, elle sera déployée à l'international pour s'adapter à la croissance de la clientèle.

Principes d'architecture

- **Simplicité** : Il s'agit d'un des principes les plus importants de l'architecture informatique. Avec le temps et les besoins qui évoluent, la simplicité n'est pas toujours possible, c'est la raison pour laquelle cet objectif doit être visé. Cela permettra de fortifier le système à mesure que les besoins s'accroissent.
- **Flexibilité** : Un système flexible peut être facilement modifié ou adapté pour répondre à de nouveaux besoins. L'importance de ce principe est telle qu'il permet à l'architecture d'évoluer au fur et à mesure que l'entreprise se développe et évolue. Négliger la flexibilité pourrait entraîner plusieurs problèmes : des modifications difficiles et coûteuses, un système obsolète et incapable de répondre aux besoins.
- **Modularité** : Un système modulaire comprend de petits modules indépendants qui peuvent être combinés pour étendre un système existant. On peut citer deux avantages à la modularité : une compréhension du système et sa maintenance car chaque module est autonome, le deuxième avantage est que ce principe peut rendre le système plus flexible, c'est à dire qu'on peut ajouter ou supprimer des modules selon les besoins.
- **Redondance** : Ce principe nous assure qu'il existe plusieurs copies de données afin qu'en cas de défaillances, une sauvegarde puisse être utilisée. Une redondance correctement

configurée permet aux applications et aux données de continuer à fonctionner peu importe sa disponibilité, de ses modifications et de ses maintenances.

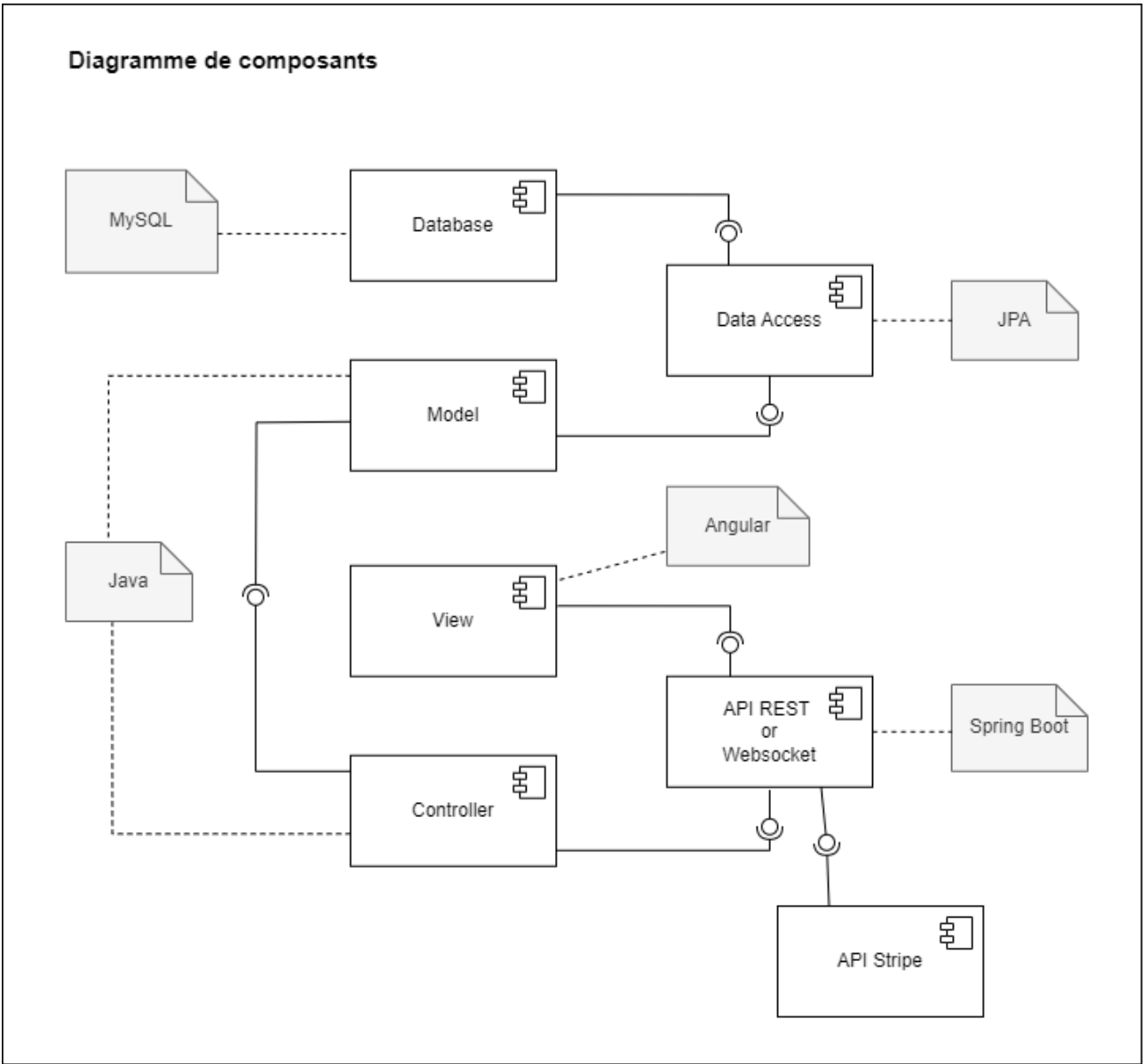
- **Évolutivité** : Un principe utilisé pour garantir la croissance de données d'un système sans perdre en performances. L'évolutivité va permettre d'accueillir plus d'utilisateurs et de données.

Architecture existante

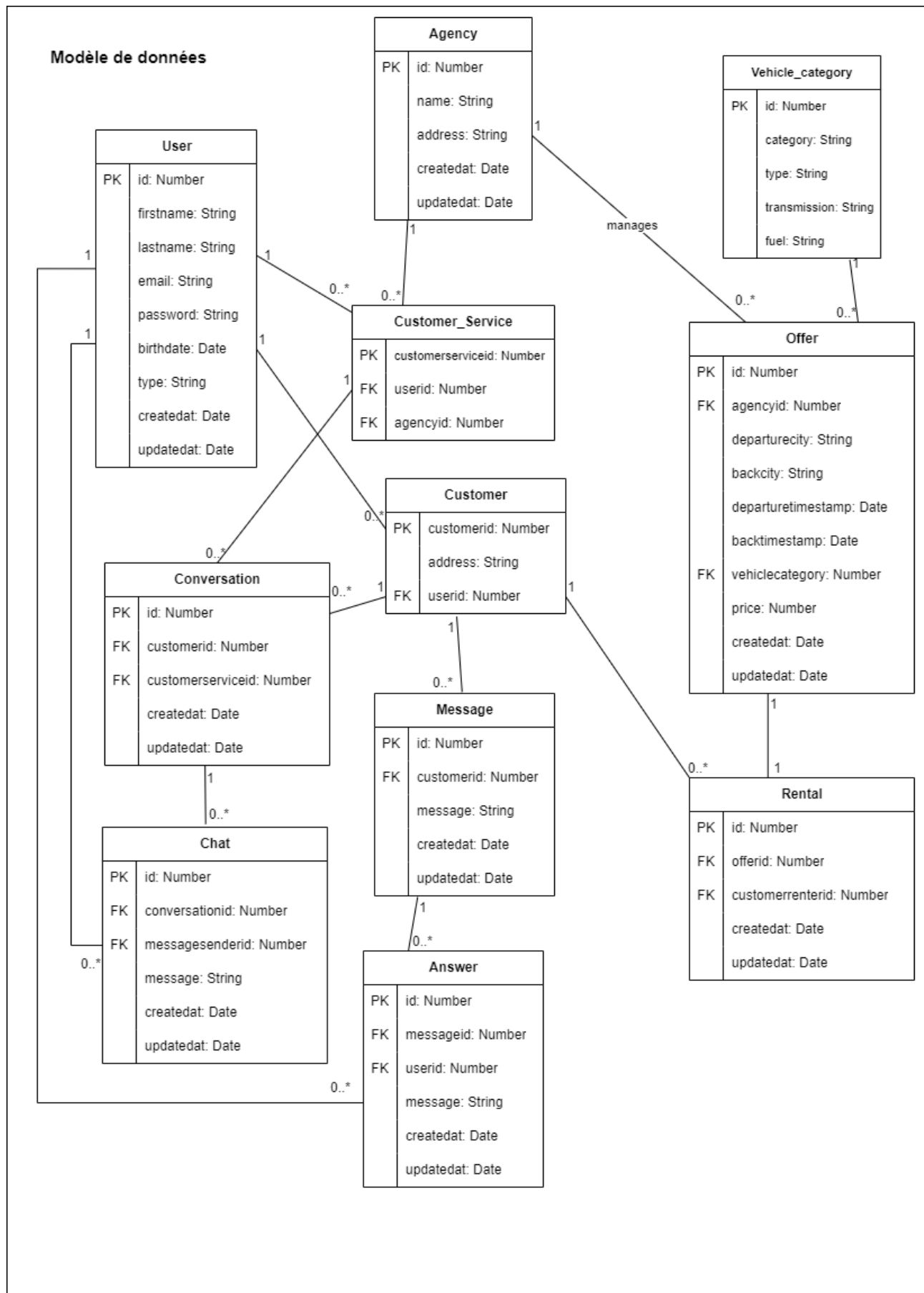
Le projet ne s'appuie pas sur un produit existant qui serait repris puis mis à jour. Il n'y a donc pas d'architecture existante à définir.

Architectures

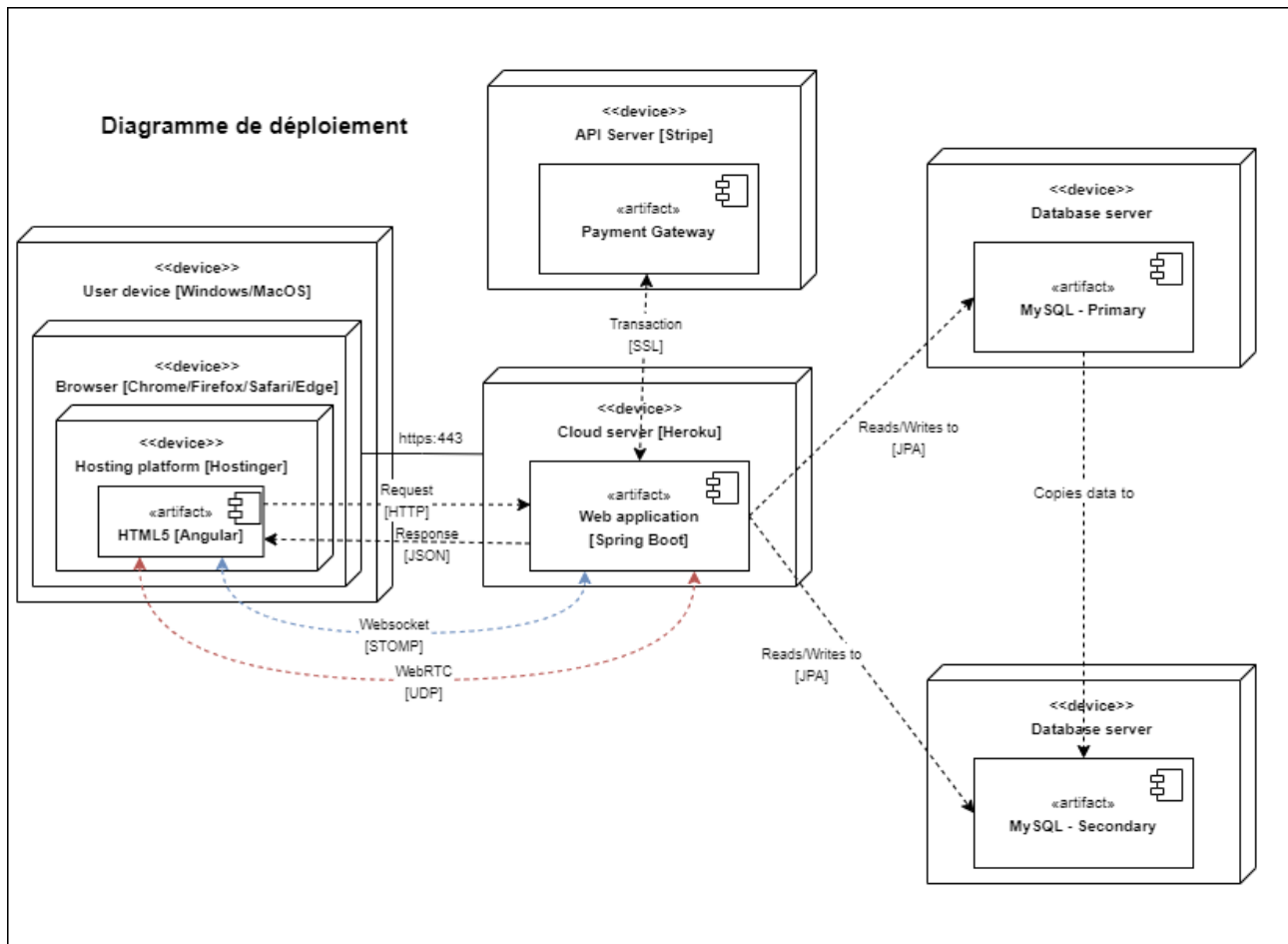
Métier



De données



Technologiques



Justification de l'approche architecturale

L'architecture modélisée répond aux objectifs de manière multiple, pour commencer, elle va permettre de mettre en place une architecture client-serveur avec les technologies suivantes Angular en front et Spring Boot en back. Cette architecture client-serveur se présente sous le modèle MVC (Modèle, Vue, Contrôleur, voir diagramme de composants), le serveur va gérer le Modèle et le Contrôleur tandis que le client va gérer la Vue. Cette disposition respecte les principes d'architecture déjà définies dans la section *Principes d'architecture*, on peut voir que les diagrammes de composants et de déploiement tendent à la **simplicité**. Par exemple, le diagramme de déploiement représente plusieurs nœuds comme la machine de l'utilisateur, la machine exécutant le serveur et les bases de données, tous ces nœuds sont détaillés selon leurs environnement d'exécution. La séparation de ces nœuds favorisent les principes de **modularité** et de **flexibilité**. On peut également noter, toujours dans le diagramme de déploiement, la présence de deux bases de données, une primaire, copiant ses données vers la secondaire, il s'agit ici du principe de **redondance**.

Architectures de transition

[Cette section est optionnelle – si la mise en oeuvre complète de toute l'architecture revêt une forte complexité, la définition d'architectures de transition, pour passer du point de départ à l'architecture finale à travers différentes étapes progressives, peut être envisagée.]

