

Projet BDR - L'écrou

Introduction

Conceptualisation

Modèle
Relationnel

Application

Conclusion





Problématique



Fonctionnalités



Introduction

Gestion d'un service de réparation d'objets

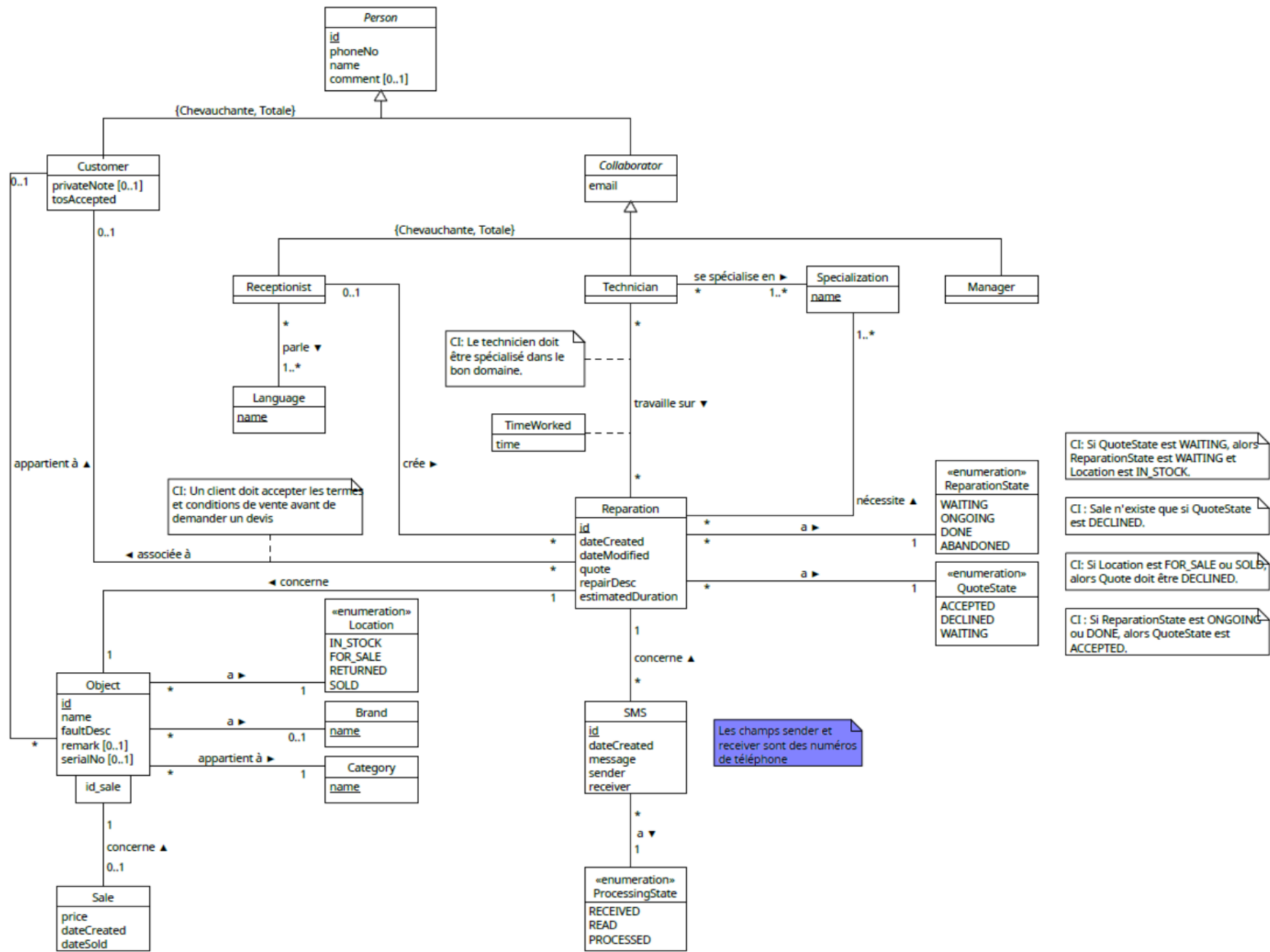
Problématique

- L'association "*L'écrou*" est un atelier de réparation d'objets.
- Les réceptionnistes accueillent les clients.
- Les clients reçoivent un devis pour la réparation.
- Si le devis est accepté, un technicien répare l'objet.
- Si le devis est refusé, le client peut céder l'objet à l'association, qui va le mettre en vente dans sa boutique.
- L'atelier communique avec le client au sujet de sa réparation par SMS

Fonctionnalités

- Statistiques sur les données disponibles sur le dashboard du site web.
- Formulaire en ligne pour créer et modifier des réparations.
- Formulaire en ligne pour créer et modifier des clients.
- Données des collaborateurs, des clients et des réparations affichées sur leurs pages respectives.
- Une section pour chaque type de collaborateur.

UML



Modèle Relationnel

- **Tables de jointure:** relations N-M
- **Vues:** représentation des sous-classes
- **Triggers:** mise à jour des données à travers des vues, assurer la consistance des états, etc.
- **Procédures:** insertion facile des données dans les sous-classes, création d'un objet à travers une réparation

Architecture

The diagram features a large teal circle on the right containing a dark green rectangle with the text 'Application de base de données'. To the left of this circle are two smaller, overlapping grey circles, one labeled 'Architecture' and the other 'Technologies'. A faint green line connects the top of the 'Architecture' circle to the top of the large teal circle.

Technologies

Application
de base de données

Architecture

L'application contient trois composants principaux

- Une base de données
- Une API
- Un site web

Le frontend communique avec l'API qui, à son tour, interroge la base de données afin de récupérer les informations.

L'API traite la réponse de la base de données au frontend, qui l'affiche.

Technologies

- **Base de données:** Construite et interrogée en utilisant la langage *PostgreSQL*.
- **API:** Codée en *Java* avec le web framework *Javalin*.
- **Connecteur:** *JDBC PostgreSQL* est utilisé pour se connecter à l'instance de la base de donnée.
- **Frontend:** Développé avec *React*. Les requêtes sont effectuées avec la librairie *Axios*.
- **Environnement:** *Docker compose* est utilisé pour définir l'environnement nécessaire au fonctionnement de l'application.

Conclusion

Axes de développement

- Inclure des formulaires pour la gestion des réceptionnistes et les managers.
- Inclure une section pour la gestion et la consultation des objets en vente.
- Création d'une page pour la gestion et le suivi du temps de travail des techniciens.