Design Document

Veloway
Smart Bicycle Rental System

Prepared by:

Emna Belguith

Rim Barnat

Molka Sahli

Nour Krichen

Supervised by:

Dr. Eng. Mohamed-Bécha Kaaniche

Academic Year: 2024-2025

Contents

1. Introduction

1.1. Problem Statement

The main challenge addressed by this project is the lack of efficient, real-time systems for public bicycle rental management in tourist areas. Current solutions often fail to ensure a seamless user experience that integrates IoT data, mobile applications, and computer vision for damage verification. Users face difficulties locating available bikes, managing their trips, or proving the condition of the bike before and after usage.

1.2. Context of the Project

This project targets urban mobility services and tourists seeking a fast, eco-friendly, and automated way to rent bicycles. It leverages IoT and AI technologies to offer a smart and secure rental experience, integrating mobile interaction, automated payment management based on duration and distance, and real-time monitoring of station availability.

1.3. Global Architecture

The proposed architecture integrates four major components to ensure scalability and real-time responsiveness:

- **Backend (Jakarta EE):** The core system managing authentication, business logic, and APIs. It utilizes **Jakarta NoSQL** for flexible data management with MongoDB.
- **Mobile Application:** A client-side application allowing users to scan QR codes, track their route via GPS, and handle payments.
- **IoT Infrastructure:**
  - **Raspberry Pi Gateway:** Located at stations, it manages local sensors and locks using Node-RED.

  - **MQTT Broker (HiveMQ):** Facilitates real-time communication between the IoT stations and the backend.

- **AI Module:** A computer vision service that analyzes photos taken by users to detect potential damages (scratches, dents) automatically.

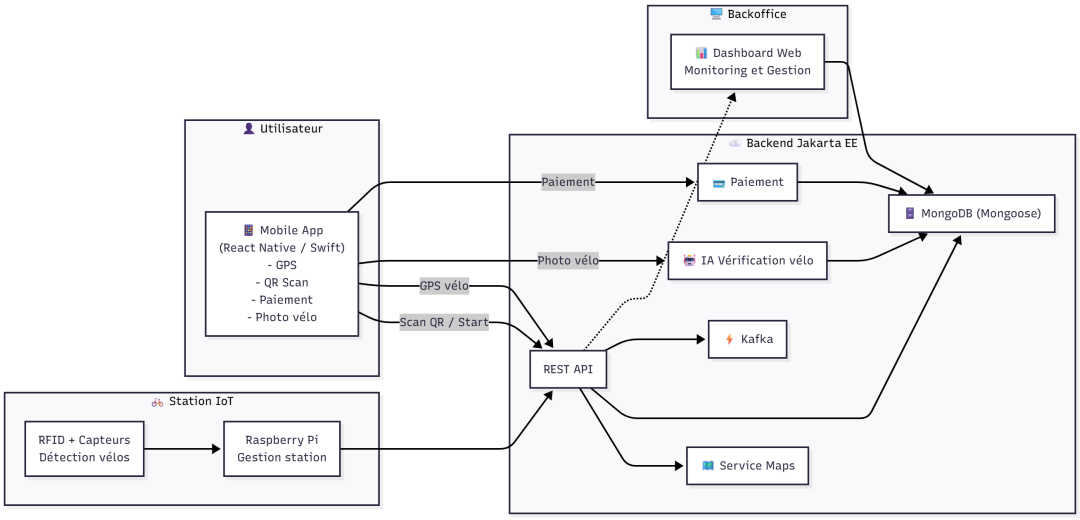

Figure 1.1: Global Architecture Overview

2. UML Diagrams

2.1. Use Case Diagram

The Use Case diagram illustrates the main interactions between the actors and the system functionalities. The system has two primary actors:

- **Users:** Can authenticate, scan QR codes to unlock bikes, track their trips via GPS, report issues, and pay automatically.
- **Administrators:** Manage station data, oversee IoT device status, and validate damage reports.

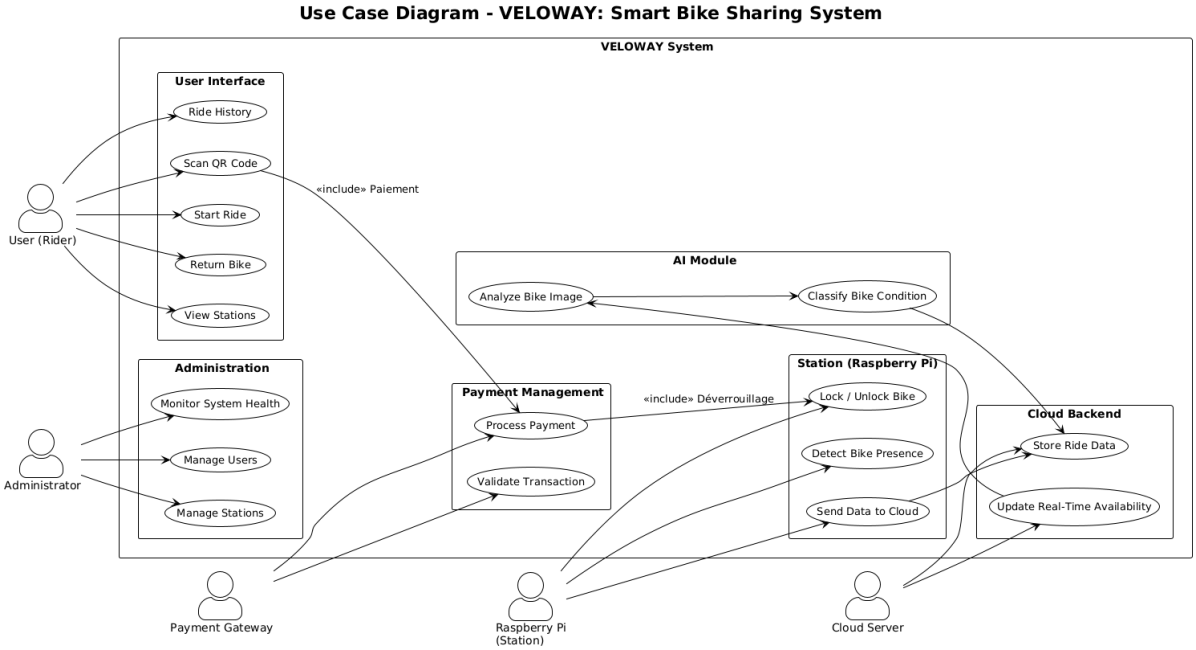

Figure 2.1: Use Case Diagram

2.2. Deployment Diagram

The deployment diagram illustrates the physical distribution of software components. The architecture is designed for IoT efficiency:

- **Mobile Device:** Runs the client application connecting via HTTPS/REST.
- **IoT Layer:** Consists of a **Raspberry Pi** running Node-RED, communicating via **MQTT** through a Cloud Broker (HiveMQ) to ensure low-latency control of bike locks.
- **Application Server:** Hosts the Wildfly/Jakarta EE backend, the AI Service, and connects to the **MongoDB** database.

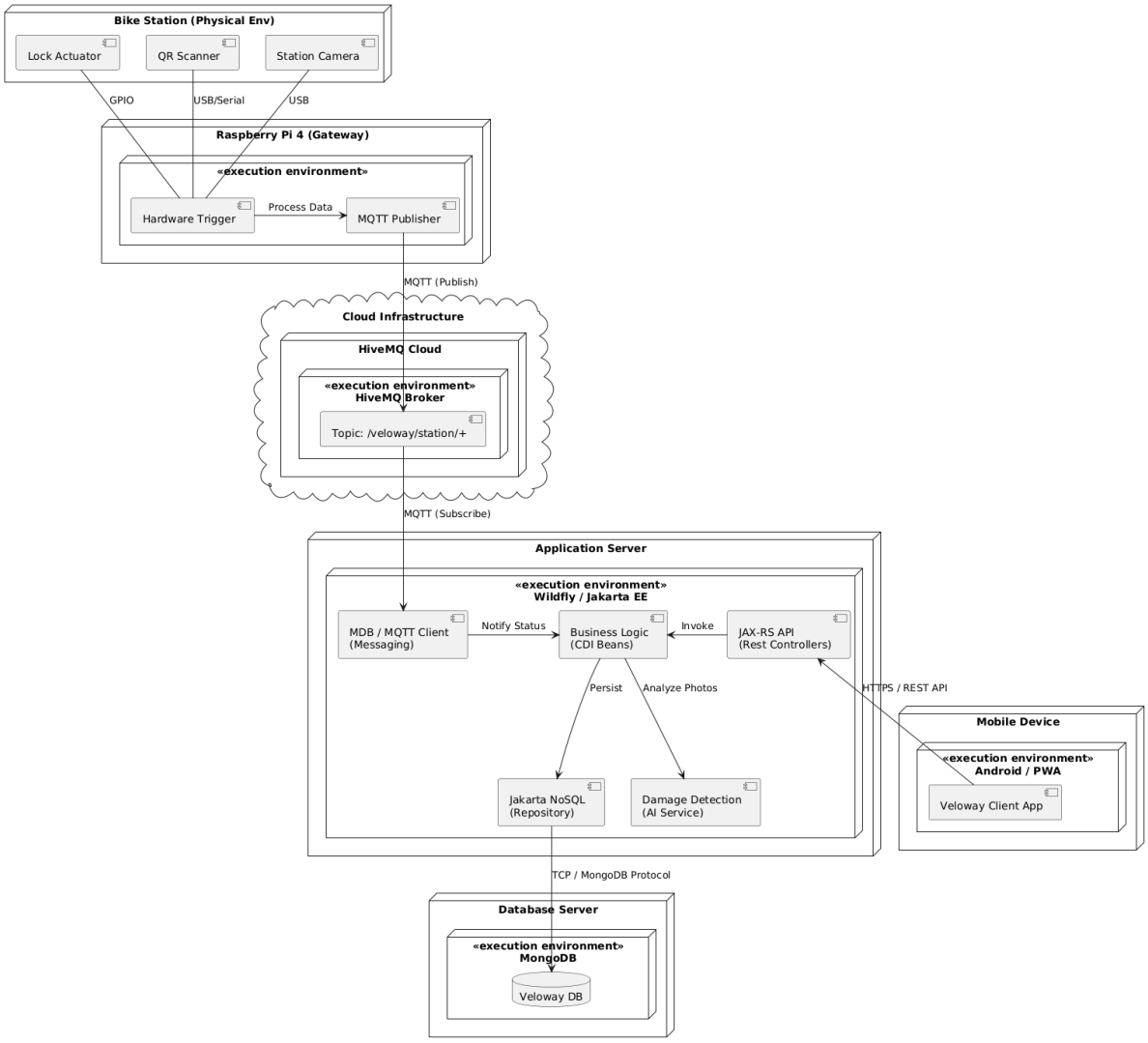

Figure 2.2: Deployment Diagram

2.3. Class Diagram

The Class diagram represents the data structure of the system, designed to support a NoSQL approach (ID-based relationships):

- **Identity (User):** Manages security credentials and profiles.
- **Bike & Station:** Represents physical assets. Bikes are linked to stations via IDs.
- **Rental:** The central entity linking a User, a Bike, and Stations (Start/End). It tracks duration and cost.
- **Damage:** Stores AI analysis results, linked to a specific Rental and Bike.

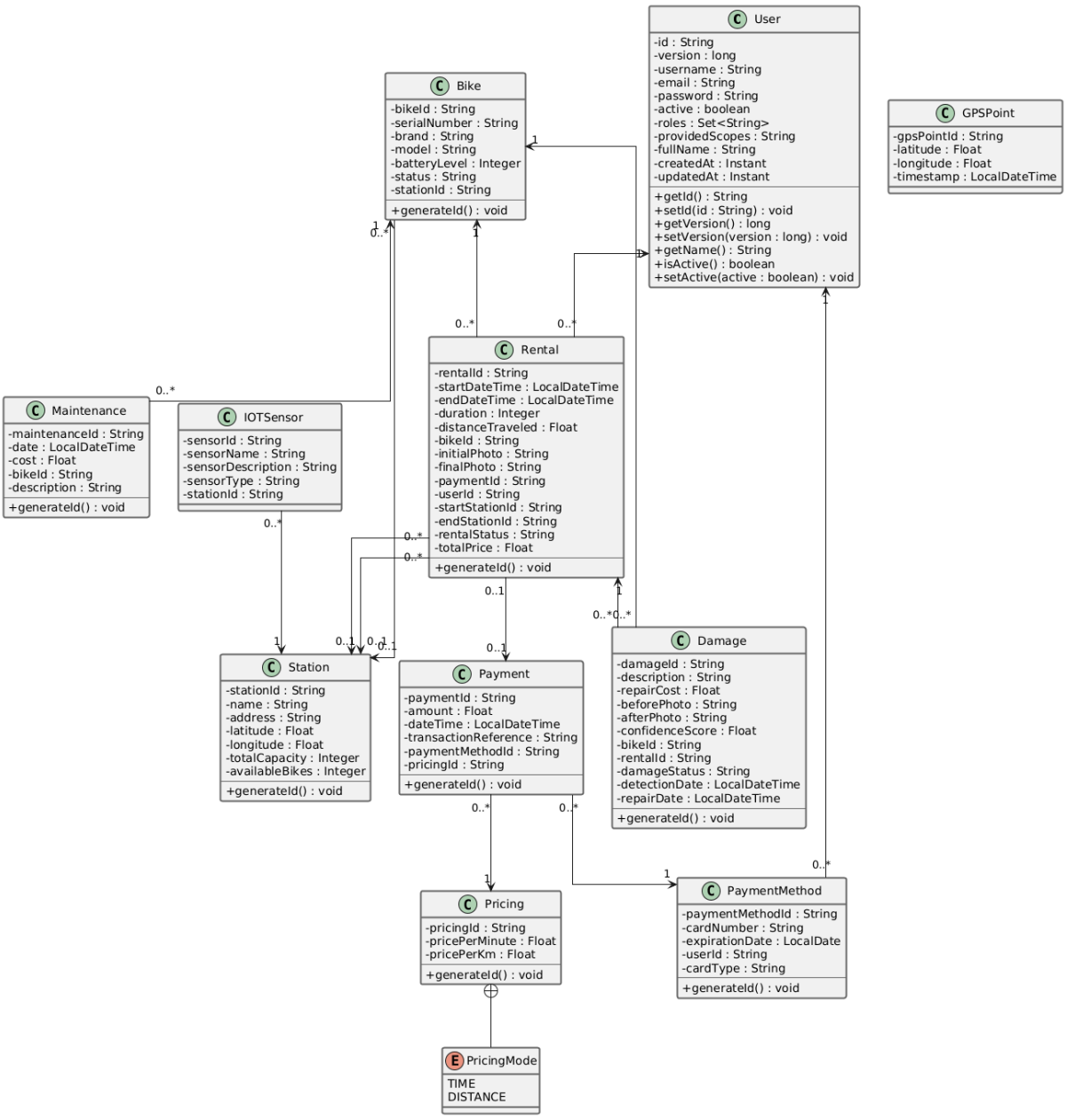

Figure 2.3: Class Diagram

2.4. Sequence Diagram

The sequence diagram details the rental lifecycle flow:

1. **Unlock:** The user scans a QR code. The backend sends an **MQTT command** to the station to unlock the specific bike.
2. **Ride:** The app periodically sends GPS coordinates to the backend to track the route.
3. **Lock & Verification:** Upon locking the bike, the user takes a photo. The backend triggers the **AI Service** to analyze the image for damages.
4. **Payment:** Once the bike status is updated (Available or Damaged), the system calculates the fare and processes the payment.

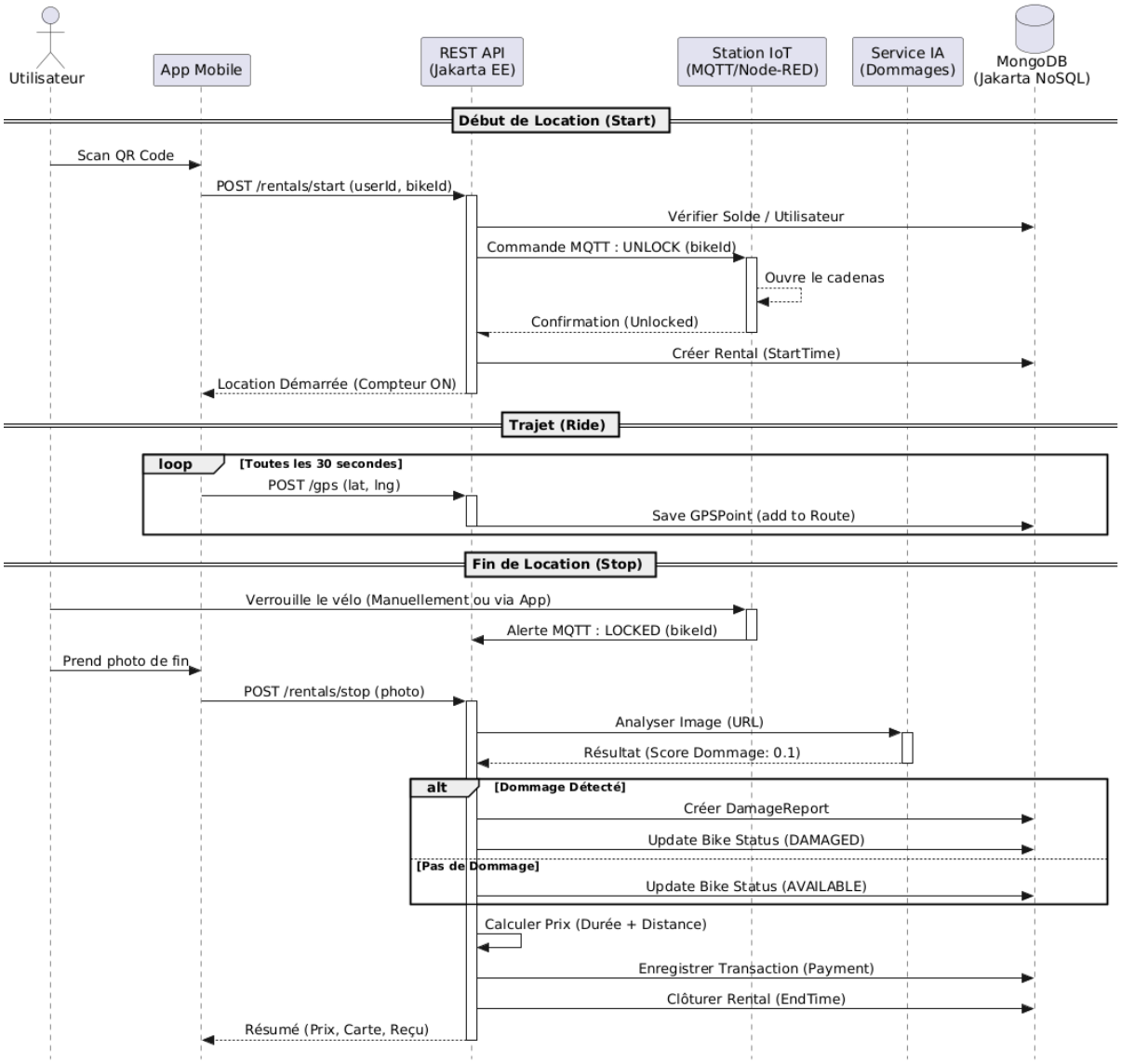

Figure 2.4: Sequence Diagram

Conclusion

In conclusion, the **Veloway** Smart Bicycle Rental System provides an integrated and scalable solution for tourist bike-sharing management. By combining Jakarta EE, MQTT-based IoT, and Artificial Intelligence, the system ensures:

- A seamless "Scan Go" user experience.
- Secure and real-time communication between stations and the cloud.
- Automated verification of bike conditions using computer vision, reducing maintenance overhead.
- Flexible data management tailored for modern cloud deployments.

This project demonstrates the effective application of modern software engineering practices to solve urban mobility challenges.