CMPE 172 Enterprise Software Platforms

Final Project Report

Car Rental System

Team 5

Dylan Zhang

En-Ping Shih

Tsz Hin Chan

Professor: Babu Thomas

San Jose State University

Spring 2020

May 11th, 2020

Table of Contents

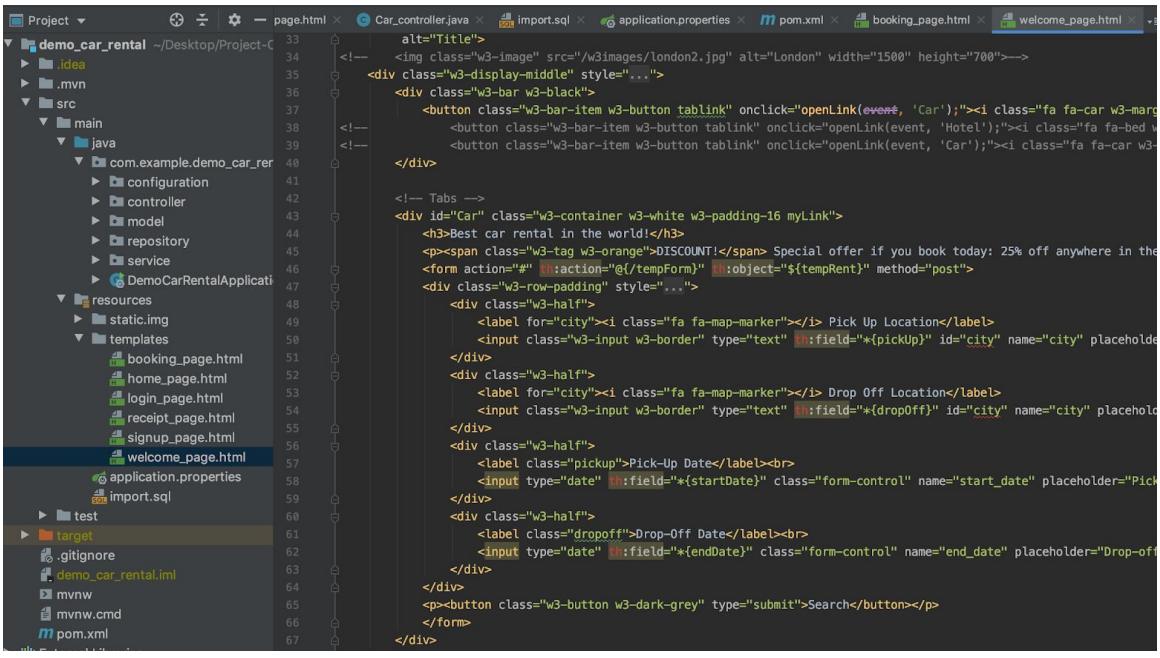
Project Overview:

The car rental platform has become an exciting startup option and has a place in the world. Users who want to rent a car can search for the car according to their needs and rent the car after providing the necessary details and payment. The goal of our project is that the process of registering a "car rental" takes only a few seconds. The user only needs to fill in personal information and select a car(s) to fulfill the entire process.

Project Description:

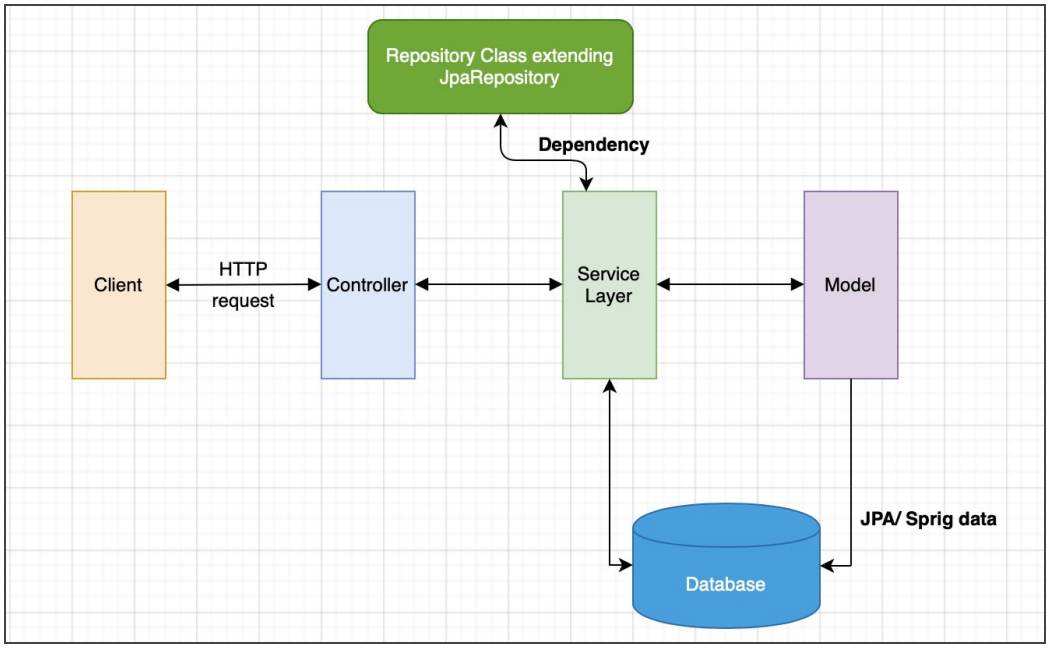
Our project is a Three-tier architecture that we need a client, server, and database. For the frontend part, we used **HTML** and CSS. The reason why we use HTML is that based on our research, HTML is the most compatible with Spring Boot and Thymeleaf. Plus, all of our team members have some experiences in HTML.

➔ HTML/CSS



For the backend part, we used **Spring Boot MVC** to serve web content, and use Maven to run the spring boot application. We used **AWS RDS** for the remote database and **MySQL** workbench for the local database. **JPA** was used to connect the database. We used **Docker** for dockerizing the application so that the web application can run on any OS system. Finally, we used the **AWS EC2** for the server which allows our application to run on a public URL.

➔ Spring boot Flow Diagram (Three-Tier)



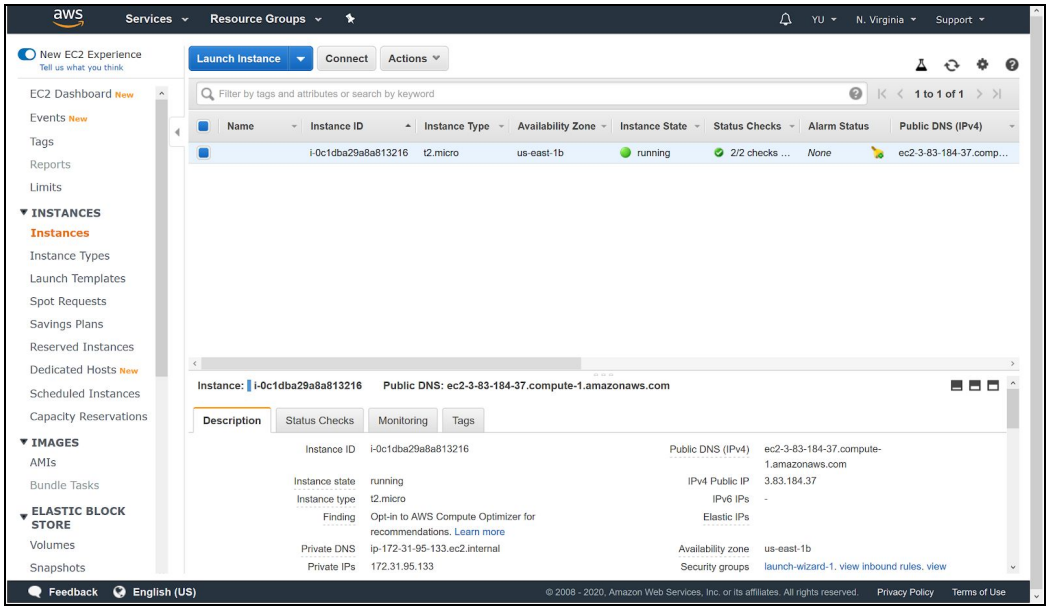
➔ AWS RDS(MySQL)

application.properties | Car_controller.java | demo_car_rental/pom.... | WebSecurityConfigura... | RentDAO.java

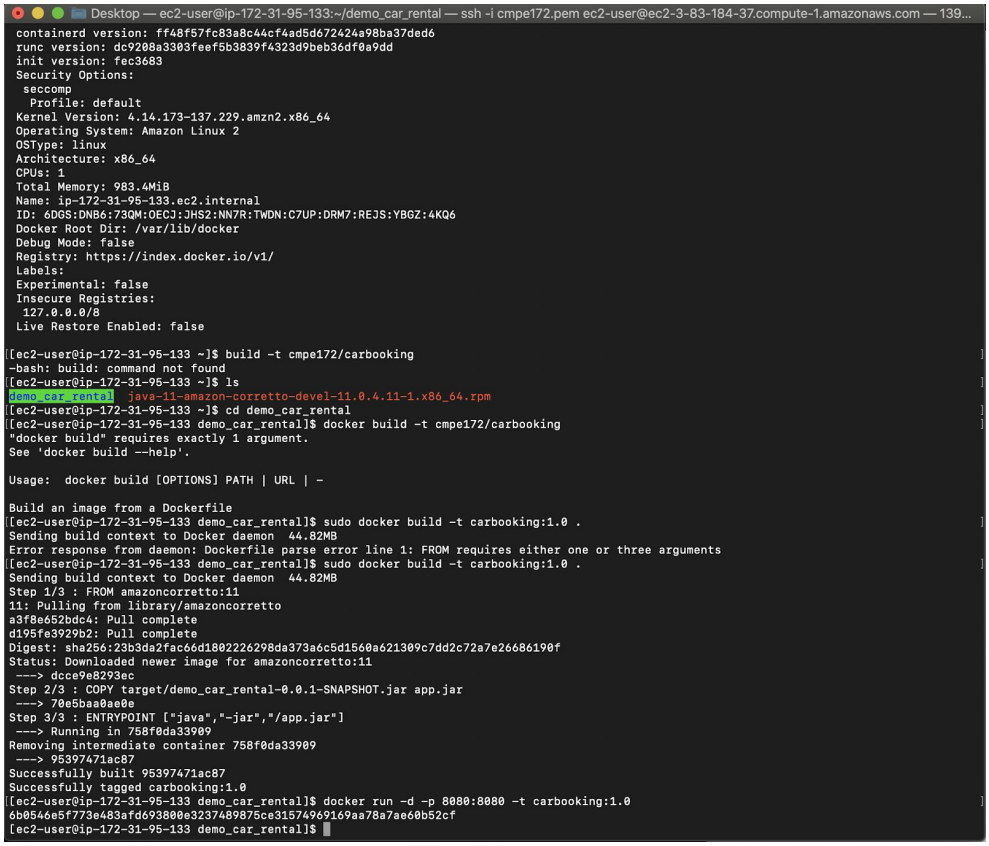
```
#debug=true
#spring.http.log-request-details=true
spring.jpa.hibernate.ddl-auto = create
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://database-1.cp0nqf3tr05o.us-east-2.rds.amazonaws.com:3306/car
spring.datasource.username=admin
spring.datasource.password=test1234
server.port=5000

## Hibernate Properties
# The SQL dialect makes Hibernate generate better SQL for the chosen database
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL5Dialect
```

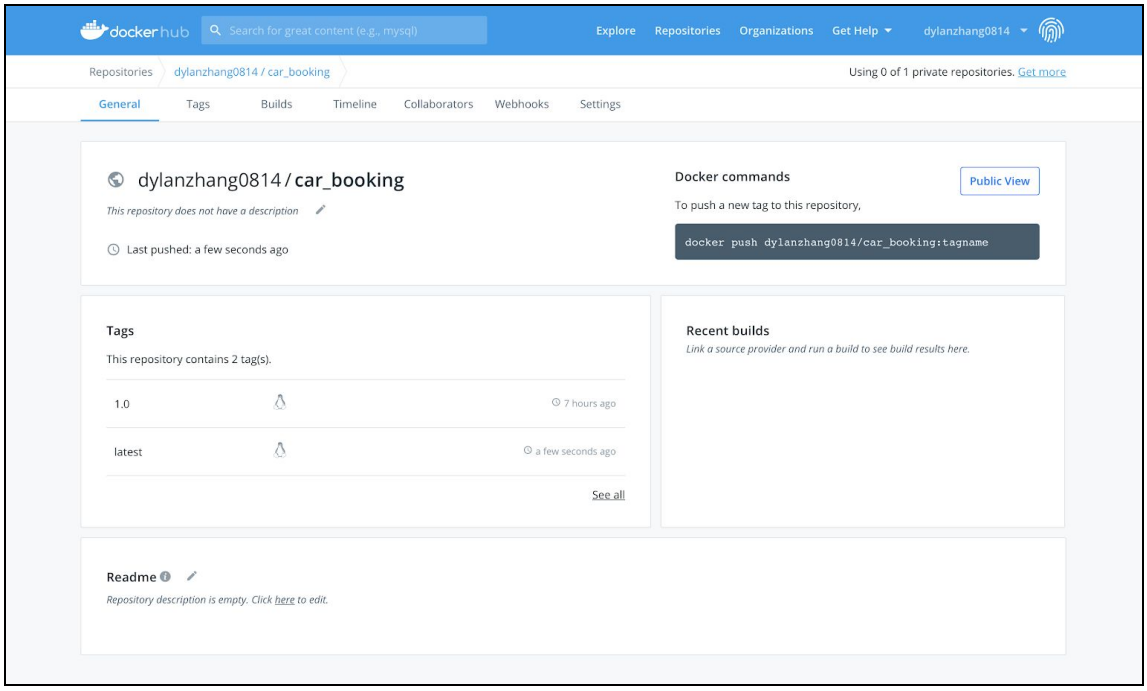
➔ AWS EC2



➔ Docker installed with AWS EC2

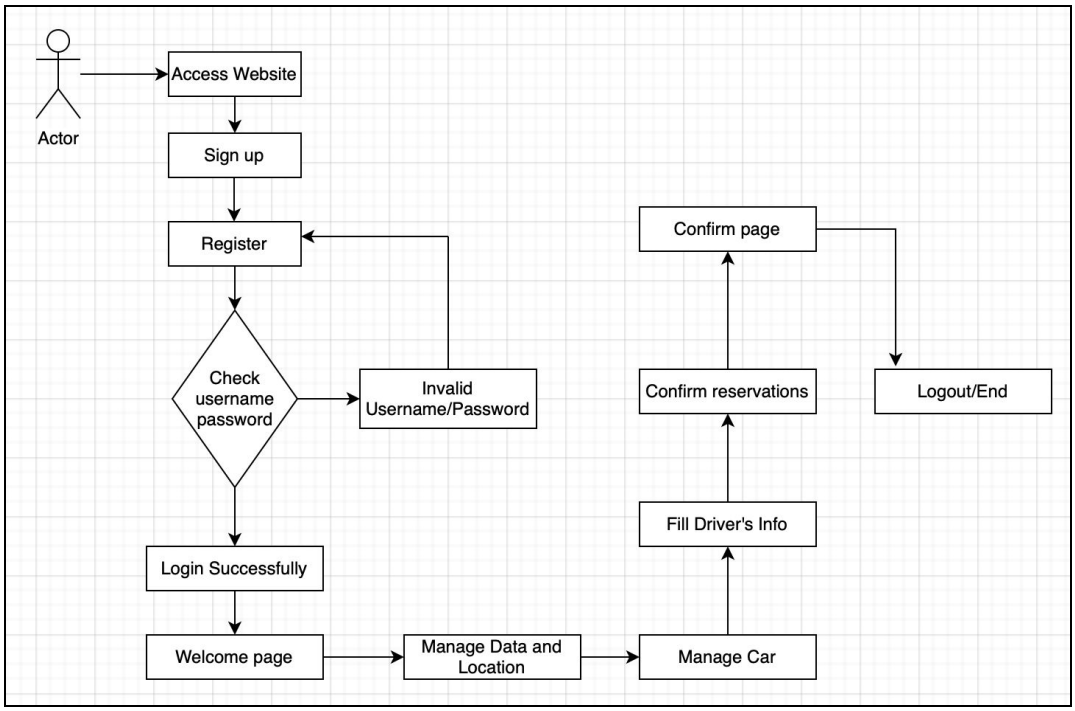


➔ Docker Hub

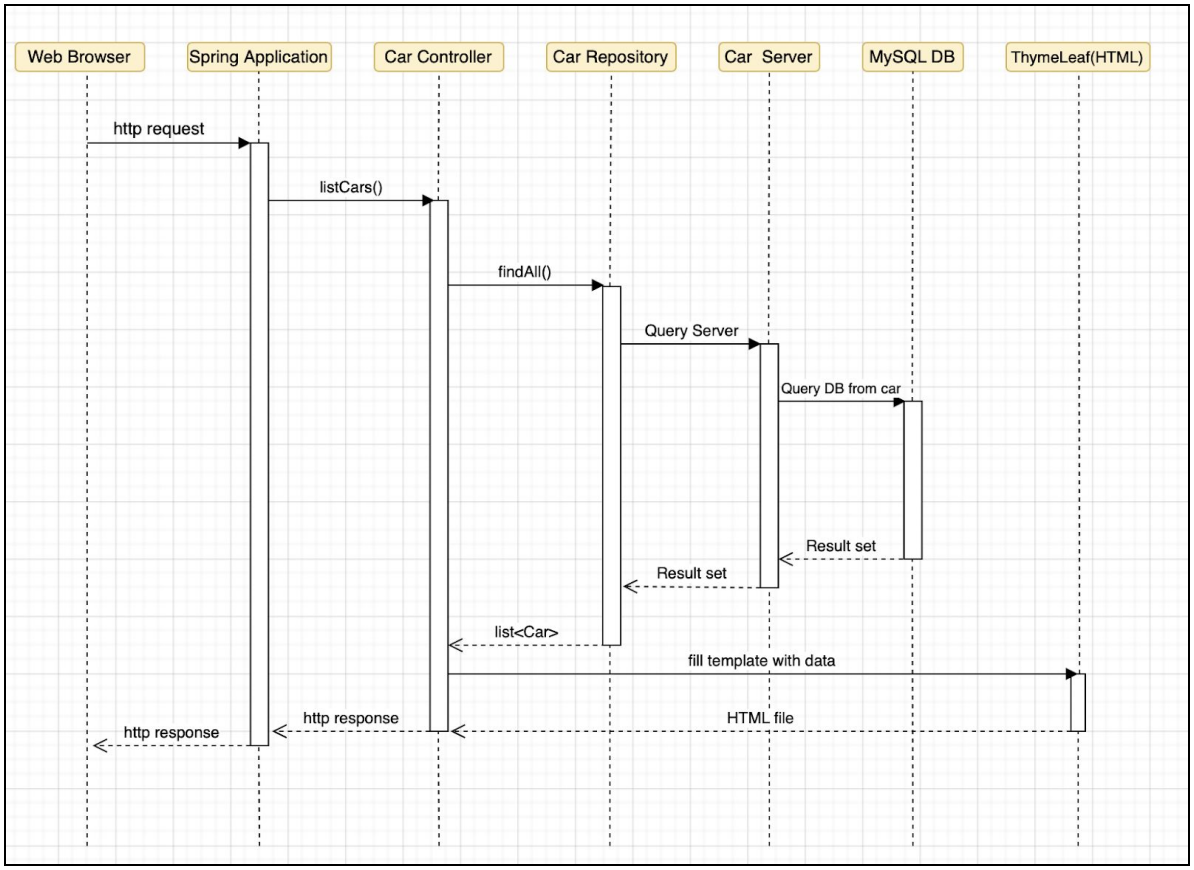


System Diagram:

- Car Rental Sequence Diagram



- System Class Template (UML)



Team Member Contributions:

Components	Team member
Spring boot, RDS, Docker, PPT	Dylan Zhang
HTML/CSS, Project Report, Demo Video, README.md	En-Ping Shih
Spring boot, EC2, HTML(ThymeLeaf), Docker	Tsz Hin Chan

Next Steps/Lessons Learned:

- Next Step

  Features we would like to implement in the future:

  - Enable registration via social media accounts for quick registration.
  - Forget/Reset password
  - Profile page
  - Add more car types
  - Search by title Function
  - Add customer service message box

- Lesson learned

  - Spring boot
  - AWS RDS
  - AWS EC2
  - Amazon Simple Storage Service (S3)
  - RESTful Web Services
  - Docker

Relevant Links:

- Links to the GitHub upload:
  https://github.com/SharonShih/Project-CarRental

- Public URL to the application:
  http://ec2-3-91-214-15.compute-1.amazonaws.com:8080/welcome
  (*NOTE: The Link is now **disabled** since we want to avoid any extra charges by AWS. But it was running when we did the video & live demo in class on 5/4/2020.)

- Test account credentials

  There's no test account in the database since the app is only runnable on the localhost and local database. We recommend creating a new account and use that account to sign in.

  The data we already have will be under the SQL file in our repo.

  File Path: *"Project-CarRental/demo_car_rental/src/main/resources/import.sql"*

References:

- All CMPE 172 Slides
- Spring boot guide: https://spring.io/guides/gs/serving-web-content/
- AWS EC2: https://aws.amazon.com/ec2/
- AWS RDS: https://aws.amazon.com/rds/