**Project Name :-Car Online Parking System**

**Group Members:-**

**BISWAPATI MAHATO (220343120025)**

**PRATIK GHOGARE (220343120083)**

**ANSHUMAN KUMAR (220343120014)**

**TUSHAR PATIL (220343120110)**

**Project Guide:- Mrs. Harshita Maheshwari**

**ABSTRACT**

In cities where the population is at its peak and the roads are all messed with vehicle's and long traffics. In such over increasing population it becomes a difficult job to find a parking spot for our vehicle. We came up with an idea where users can logon to our Web Application and find the ideal parking spot. Our project Online Parking Booking system is developed in Java, MYSQL ReactJs. With this the users save's both time as well as fuel. The user can easily view the parking availability on the web application and drive straight to the spot without wasting any time. Users can view the pricing details for parking their vehicles. In this project the user can able to park the vehicles according to their time slot and also the admin who manages all these parking will be very easy to manage all these. With such a system the parking authorities can easily manage their parking spaces efficiently.

**INTRODUCTION**

In the 21st century finding a free car parking slot has become a mind-numbing process, especially for people who travel in the morning to work or are following their daily routine, they find it highly difficult and challenging to get a parking slot for their cars. Moreover, the parking slots are never user-friendly and provide no logical data about the availability of the spot unless the user visits it manually. These kind of problems are faced regularly by every individual because the factor of uncertainty is very high and there are not many possible solutions in existence for solving the issue that may benefit the users by saving their time or keeping their mental state happy and carefree. In our ever populating cities and districts to find parking space is becoming increasingly difficult as traffic increases. Drivers have to go back and forth desperately looking for parking spaces wasting their valuable time, fuel consumption with increased likelihood of causing accidents. In the existing system we can see that some supervision is required for the parking system and it not fully automated. The driver has to make sure that the car is parked in a spot without disturbing the convenience of others. In most cases the main problem is finding the spot and trying to secure the spot for parking which in turn leads to increased stress level for the person driving the car. Moreover, the relative analysis of the data is structural to the implementation of the parking procedure. Nowadays, in this busy world it’s really hard for a person to find a spot for parking. The current parking system doesn’t give the user a specified parking slot inside the area. Parking in general in a long and time consuming process and we hope to provide a solution to alleviate this problem.

**Implementation Technology:-**

**Spring Framework:-**

The Spring Framework provides a comprehensive programming and configuration model for modern Java-based enterprise applications - on any kind of deployment platform.

A key element of Spring is infrastructural support at the application level: Spring focuses on the "plumbing" of enterprise applications so that teams can focus on application-level business logic, without unnecessary ties to specific deployment environments.

## Features

* [Core technologies](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/core.html): dependency injection, events, resources, i18n, validation, data binding, type conversion, SpEL, AOP.

* [Testing](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/testing.html): mock objects, TestContext framework, Spring MVC Test, WebTestClient.
* [Data Access](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/data-access.html): transactions, DAO support, JDBC, ORM, Marshalling XML.
* [Spring MVC](https://docs.spring.io/spring/docs/current/spring-framework-reference/web.html) and [Spring WebFlux](https://docs.spring.io/spring/docs/current/spring-framework-reference/web-reactive.html) web frameworks.
* [Integration](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/integration.html): remoting, JMS, JCA, JMX, email, tasks, scheduling, cache.
* [Languages](https://docs.spring.io/spring-framework/docs/current/spring-framework-reference/languages.html): Kotlin, Groovy, dynamic languages.

Spring Boot:

Spring Boot makes it easy to create stand-alone, production-grade Spring based Applications that you can "just run".

We take an opinionated view of the Spring platform and third-party libraries so you can get started with minimum fuss. Most Spring Boot applications need minimal Spring configuration.

Features

* Create stand-alone Spring applications
* Embed Tomcat, Jetty or Undertow directly (no need to deploy WAR files)
* Provide opinionated 'starter' dependencies to simplify your build configuration
* Automatically configure Spring and 3rd party libraries whenever possible
* Provide production-ready features such as metrics, health checks, and externalized configuration
* Absolutely no code generation and no requirement for XML configuration

**The JDBC Template**

The central class of the Spring JDBC abstraction framework is the **JdbcTemplate** class that includes the most common logic in using the JDBC API to access data, such as handling the creation of connection, statement creation, statement execution, and release of resource. The**Jdbc-Template**class can be found in the **org.springframework.jdbc.core**package.

The **JdbcTemplate** class instances are thread-safe once configured. A single **JdbcTemplate** can be configured and injected into multiple DAOs.

We can use the **JdbcTemplate** to execute the different types of SQL statements. **Data Manipulation Language** (**DML**) is used for inserting, retrieving, updating, and deleting the data in the database such as **SELECT**, **INSERT**, or **UPDATE** statements

**MySQL**

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation.

**Features of MySQL:**

* **MySQL is a database management system.**

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

* **MySQL databases are relational.**

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment.

* **MySQL software is Open Source.**

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything.

* **The MySQL Database Server is very fast, reliable, scalable, and easy to use.**

MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

* **MySQL Server works in client/server or embedded systems.**

The MySQL Database Software is a client/server system that consists of a multithreaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

1. **Hardware and Software Requirements (Minimum):**

**Hardware:**

1. Intel i3 processor 3rd generation or later / AMD Ryzen 200 2nd generation or later

2. 4 GB ddr3 ram.

3. Windows 10 Home edition or later.

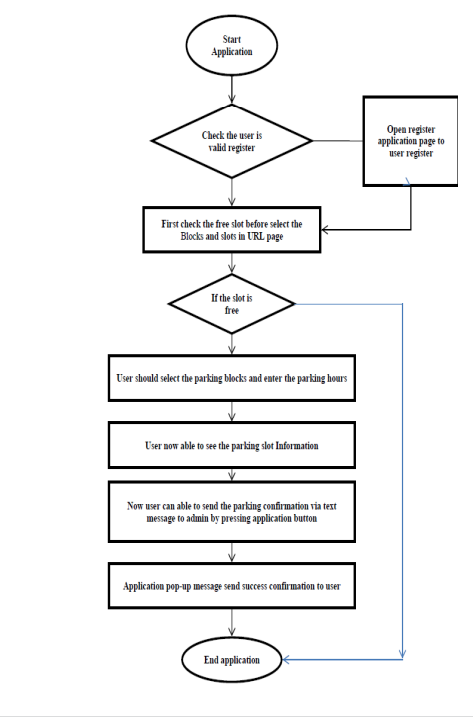
4. 200 GB Sata HDD Space

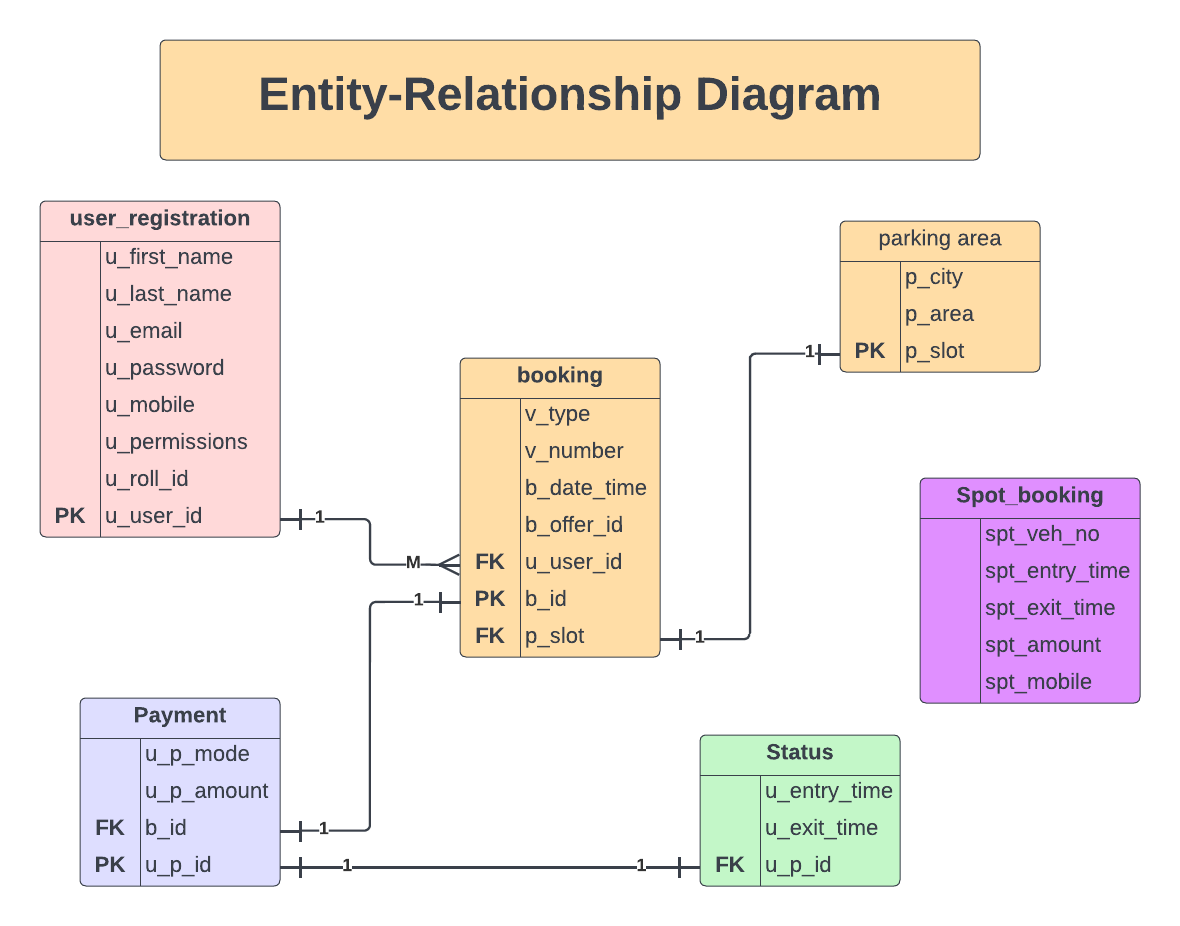
5. Data Connection 200 kbps

**Software:**

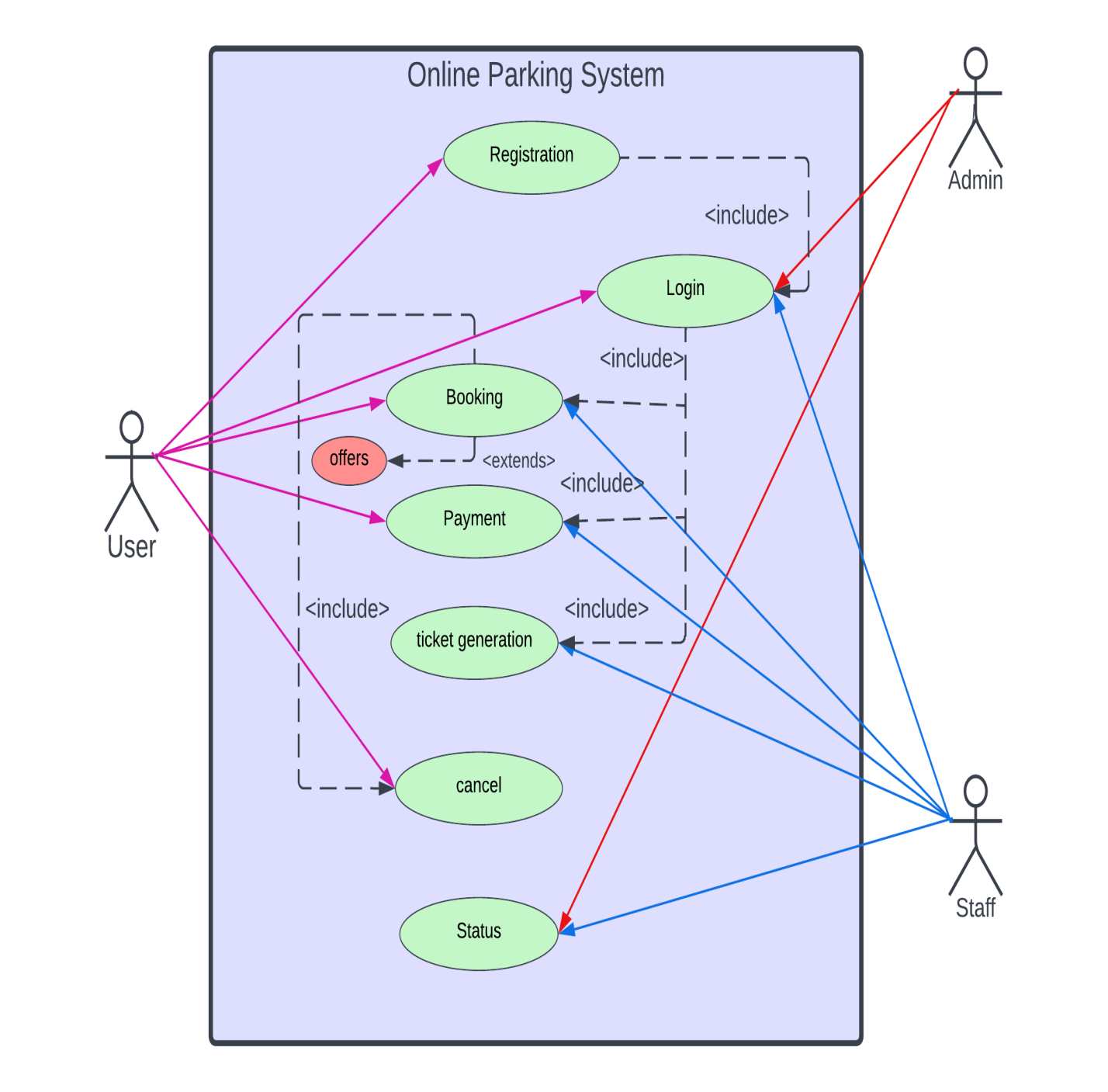
1. STS 3.9.18
2. MySQL 5.7 with Workbench 8.0
3. Google Chrome version
4. Apache Tomcat Server 9.0
5. Maven Dependencies

**Work Flow Of Website**





**Use Case Diagram:-**

****

1. **End to End Flow of Application:**

**User:**

* 1. User will login to the portal or will have to register if he is not a registered user.
  2. After registration User will login and Dashboard page will be displayed to him which will display the parking bookings if any.
  3. From that page can User can click on the ‘**book slot’** button and able to see parking slots and cost.
  4. A ‘**booking status’** will be displayed on the Website showing all the details of the bookings.
  5. User will only be able to see his bookings after the spot booking.

**Admin:**

1. Admin will login as Admin from the ‘**Admin login**’ page and will be able to see all registered users .
2. Admin can manage rate of the parking and parking area.
3. Admin can abele to see the collections of amount.
4. Admin can add new parking area and remove also.