**OOP Project**

## Group Number: 26

# Anurag Saksena - 2020A7PS1290H

# Nikhil Madhav Sharma – 2020A7PS1303H

# Lingutla Sai Charan – 2020A7PS1296H

# Kaustav Chatterji – 2020A7PS1112H

# Implementation Details:-

# Frontend:-

Frameworks Used: Vue.js, Vuetify, Axios

Programming Languages Used: Javascript, HTML, CSS

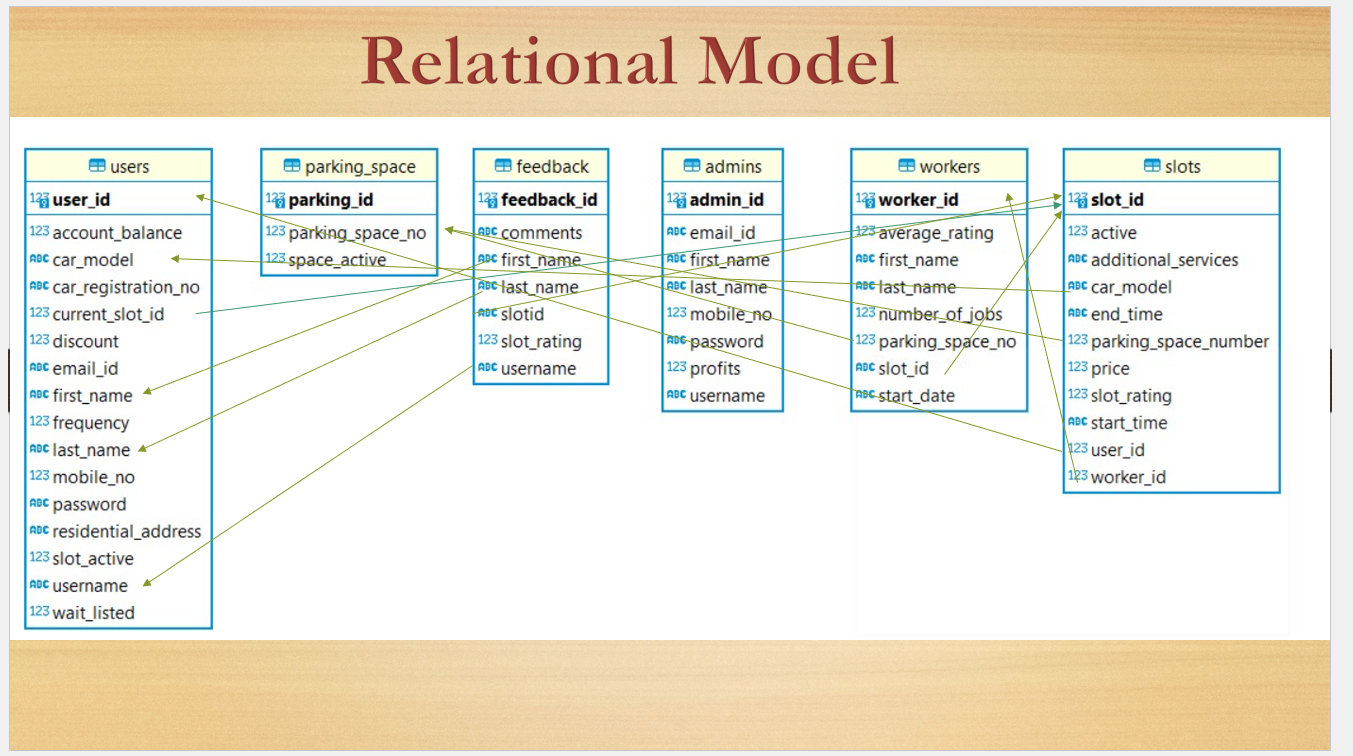
# Backend:-

Frameworks Used: Spring Boot, JPA

Tools Used: MySQL Server, MySQL Command Line Client, Curl, Maven

Programming Languages Used: Java

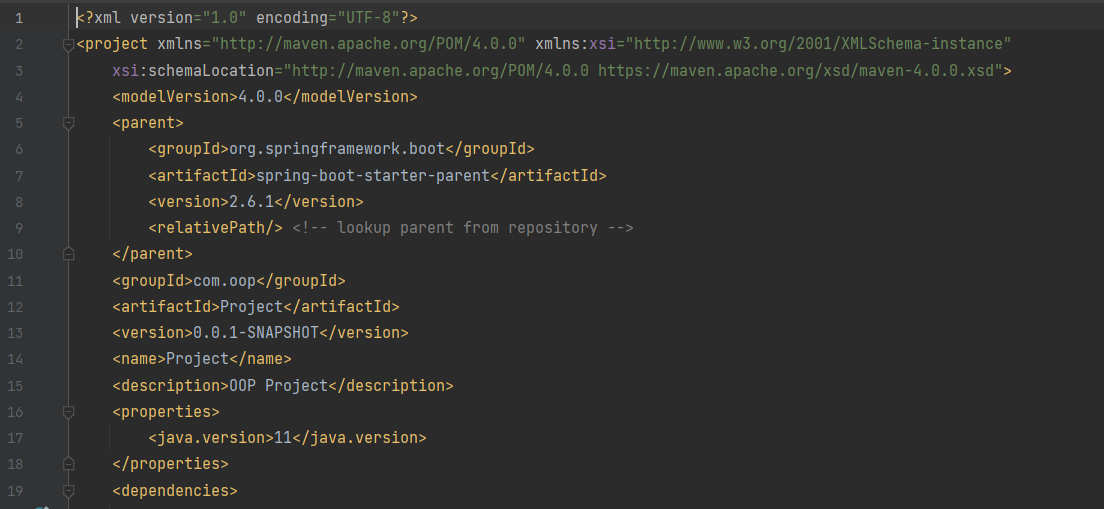
# Class Diagram:



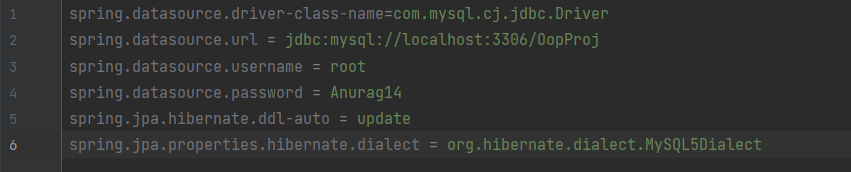
# Code:

# Backend:-

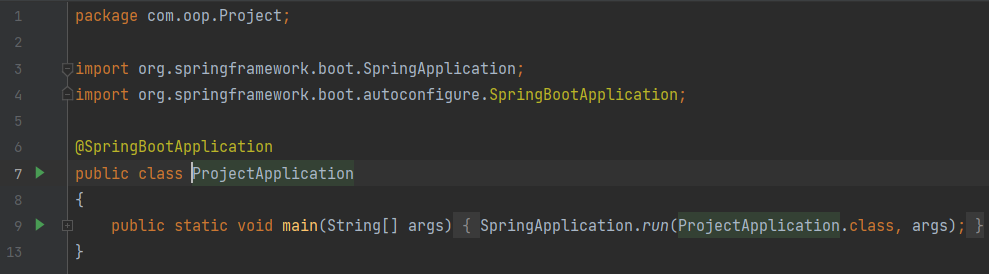
Maven “pom.xml” Project Configuration file:



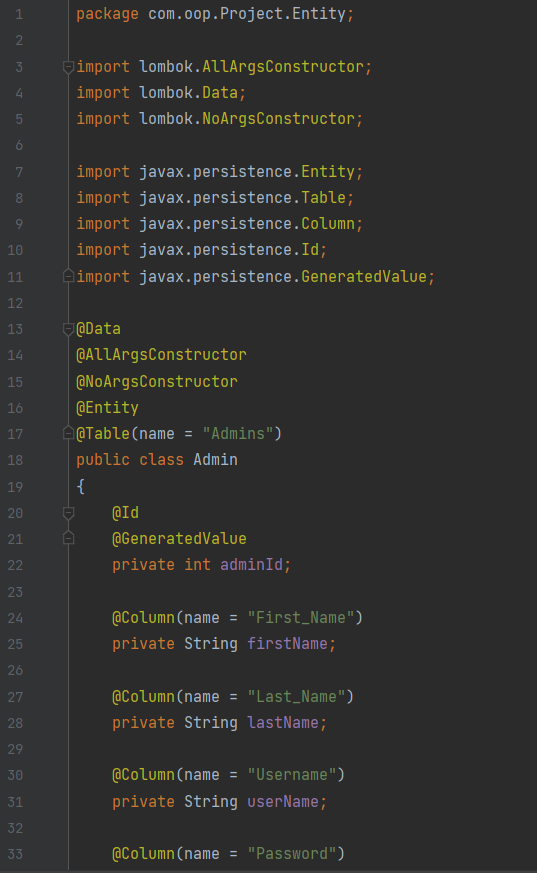
Spring Boot “application.properties” file:

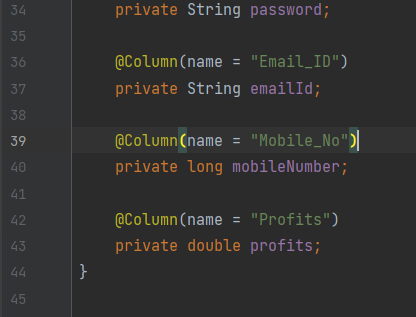


Spring Boot Main Application Java File “ProjectApplication.java”:

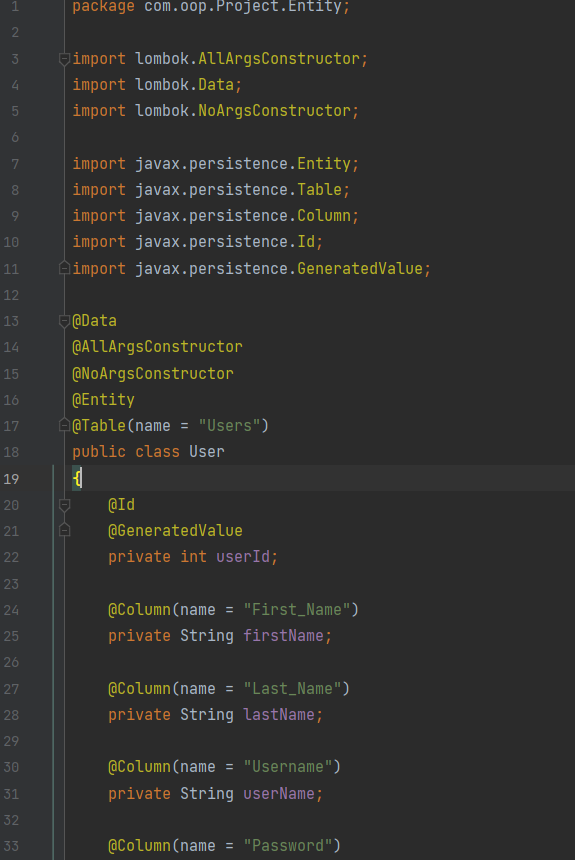


“Admin.java” Entity File:





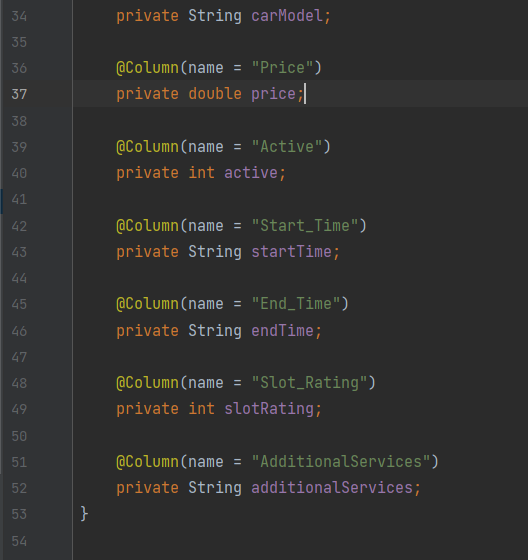
“User.java” Entity File:





“Slot.java” Entity File:

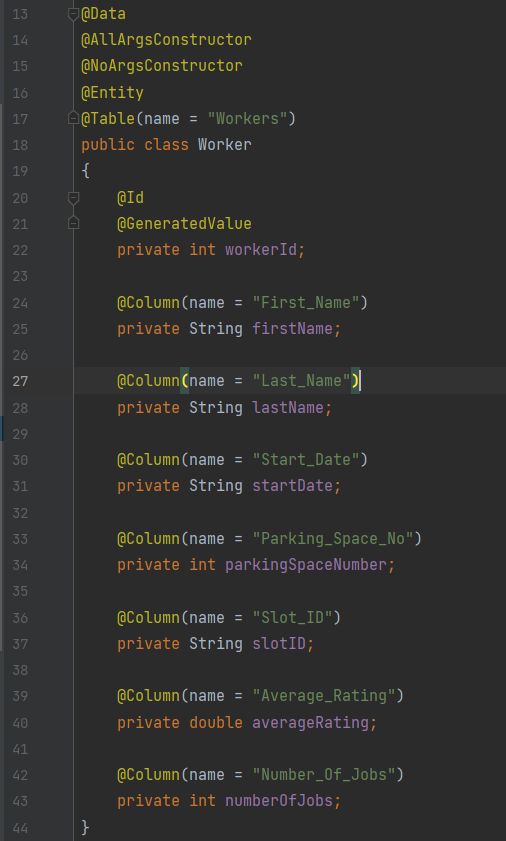




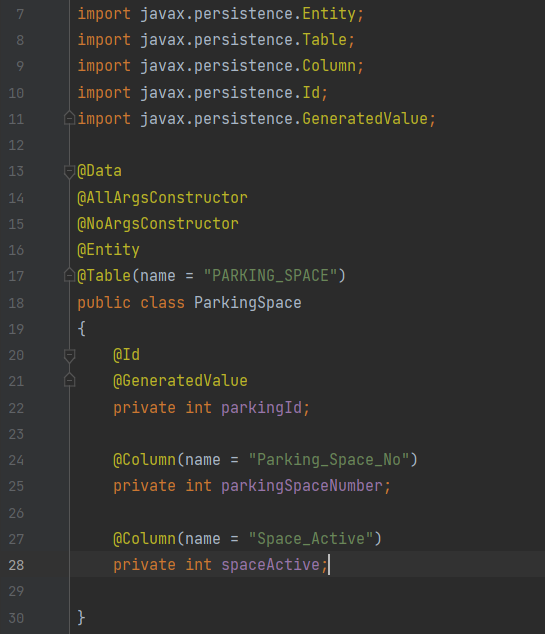
“Feedback.java” Entity File:



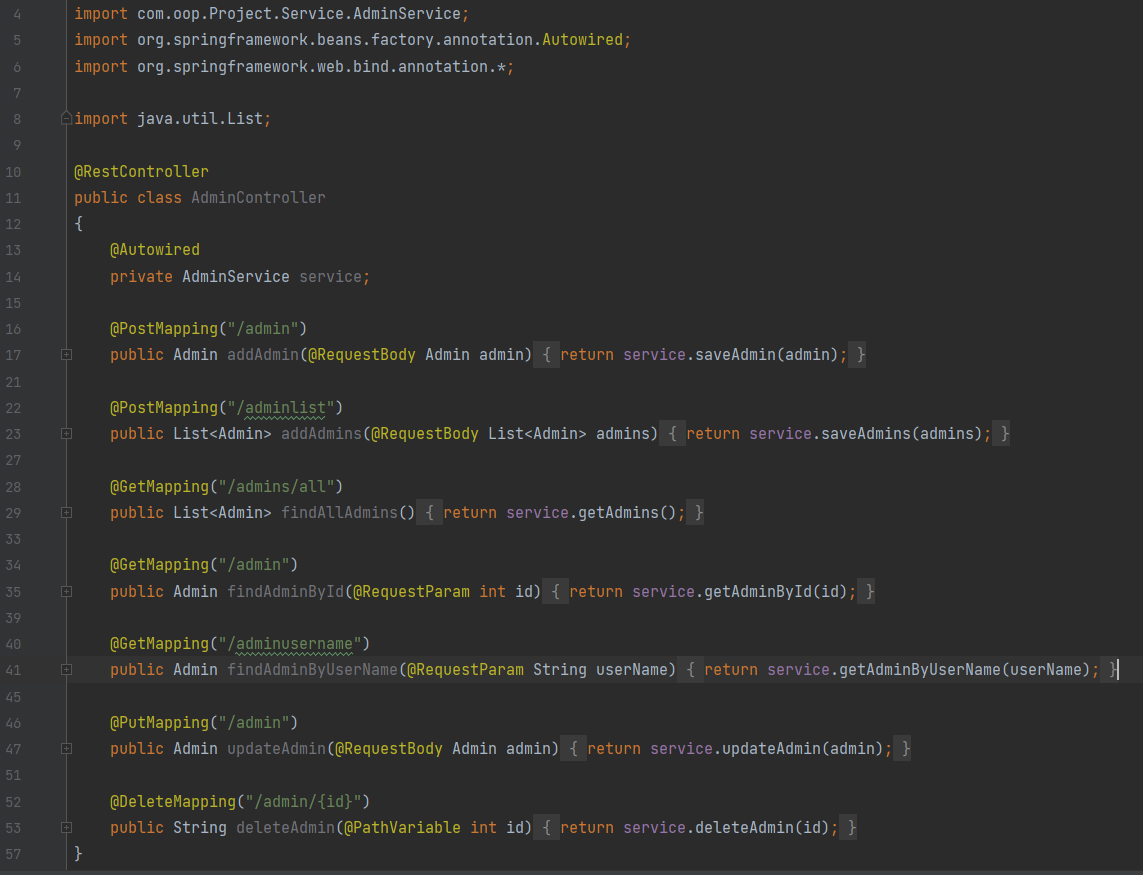
““Worker.java” Entity File:



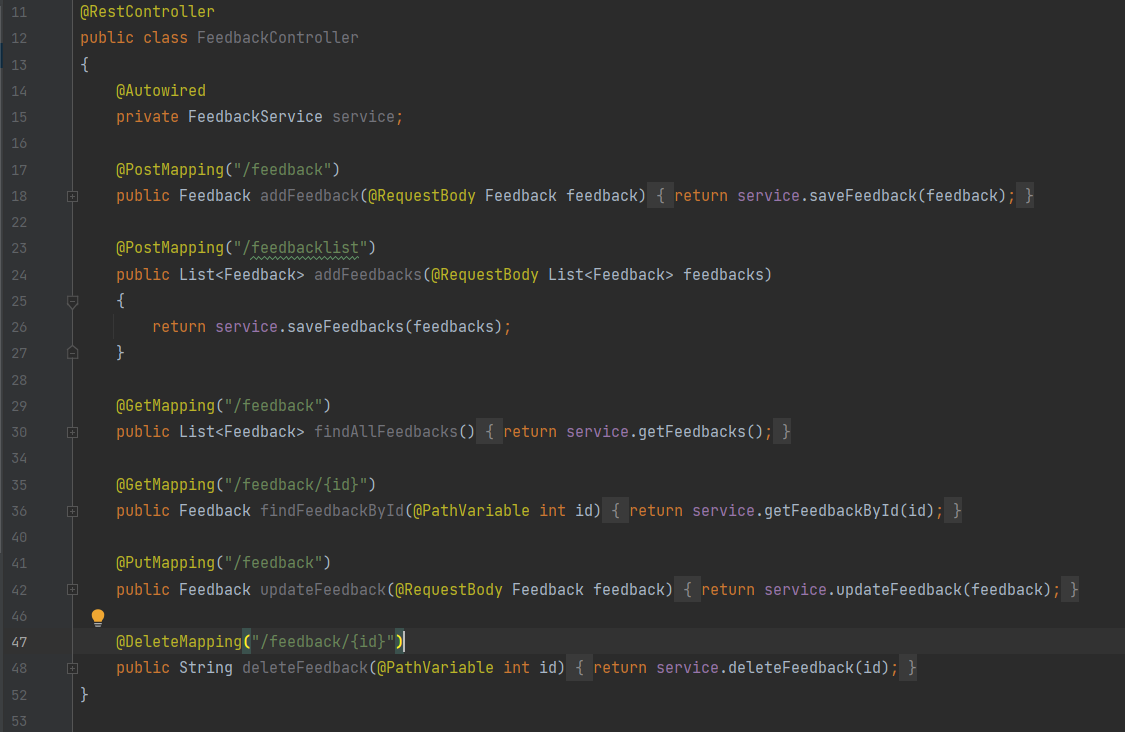
“ParkingSpace.java” Entity File:



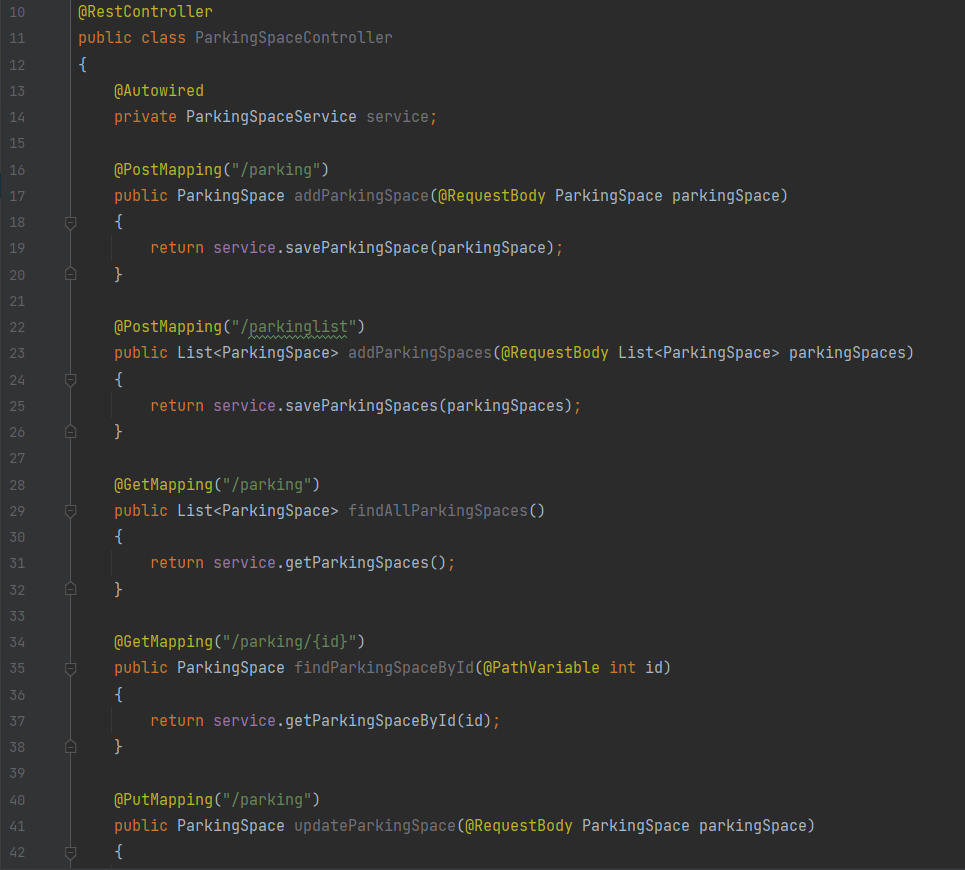
“AdminController.java” Controller File:



“FeedbackController.java” Controller File:



“ParkingSPaceController.java” Controller File:





Slot Controller

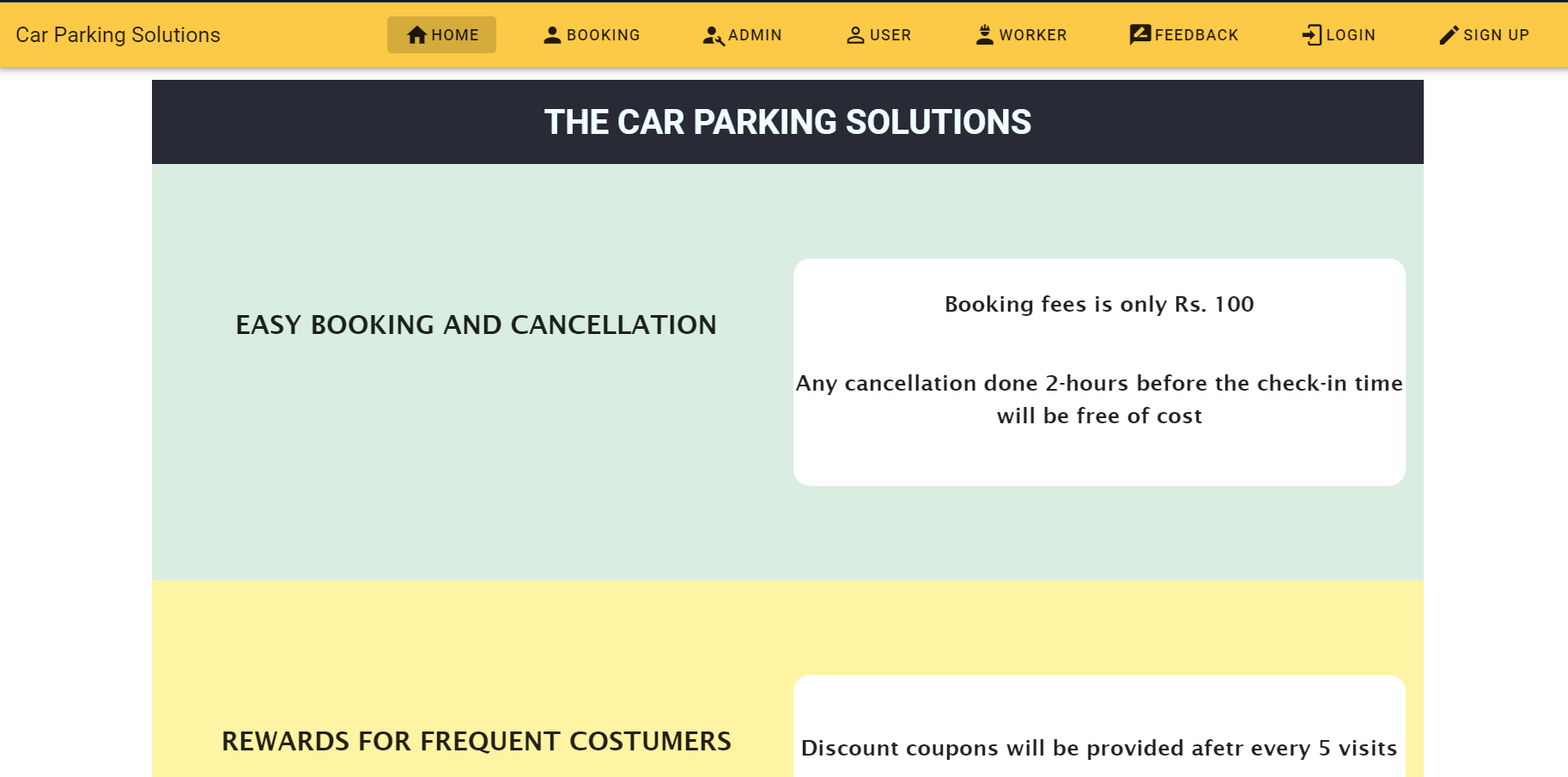
@Data  
@AllArgsConstructor  
@NoArgsConstructor  
@Entity  
@Table(name = "Slots")  
public class Slot  
{  
 @Id  
 @GeneratedValue  
 private int slotId;  
  
 @Column(name = "User\_ID")  
 private int userId;  
  
 @Column(name = "Parking\_Space\_Number")  
 private int parkingSpaceNumber;  
  
 @Column(name = "Worker\_ID")  
 private int workerId;  
  
 @Column(name = "Car\_Model")  
 private String carModel;  
  
 @Column(name = "Price")  
 private double price;  
  
 @Column(name = "Active")  
 private int active;  
  
 @Column(name = "Start\_Time")  
 private String startTime;  
  
 @Column(name = "End\_Time")  
 private String endTime;  
  
 @Column(name = "Slot\_Rating")  
 private int slotRating;  
  
 @Column(name = "AdditionalServices")  
 private String additionalServices;  
}

User Controller

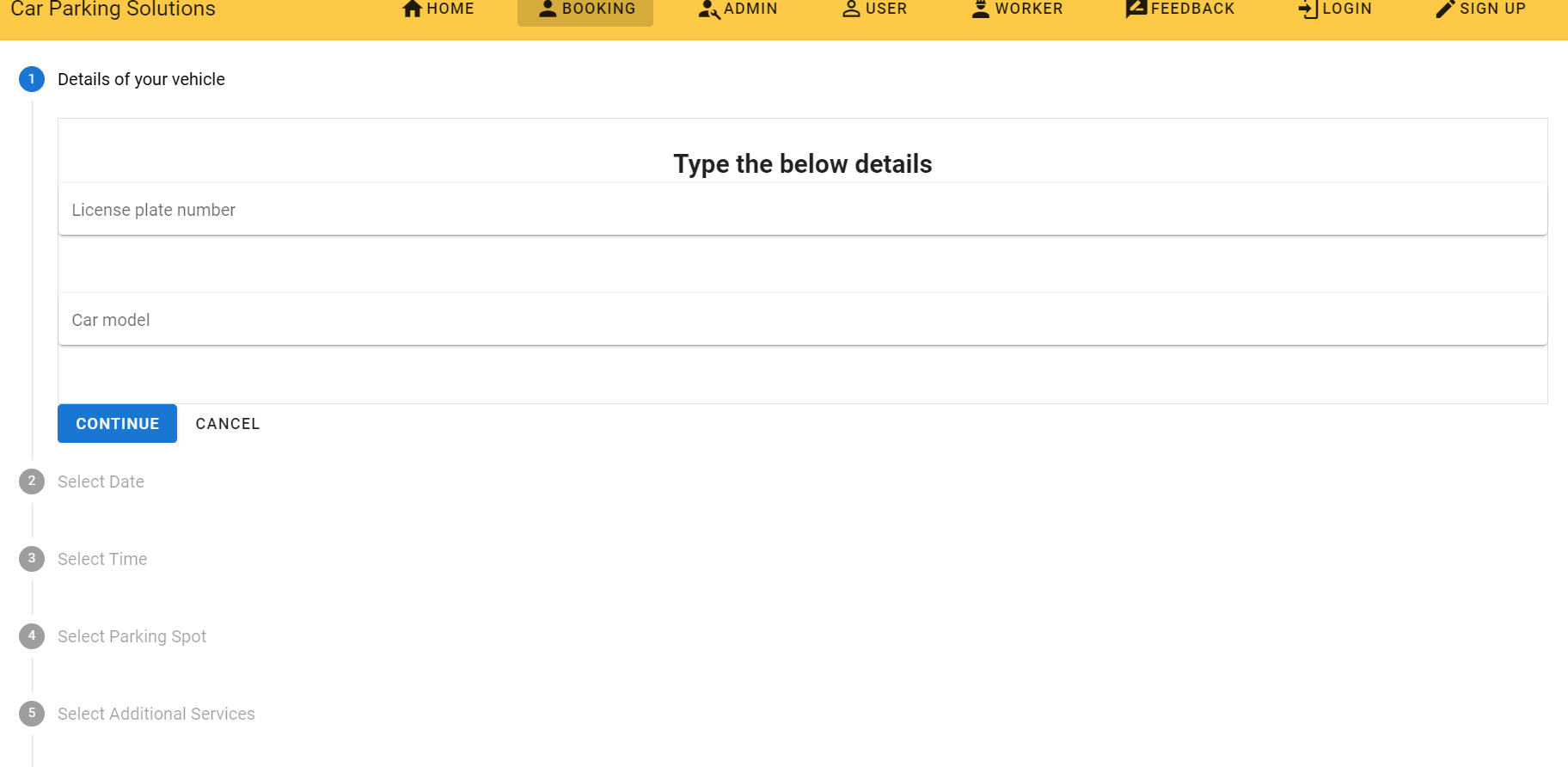
@RestController  
public class UserController  
{  
 @Autowired  
 private UserService service;  
  
 @PostMapping("/user")  
 public User addUser(@RequestBody User user)  
 {  
 return service.saveUser(user);  
 }  
  
 @PostMapping("/userlist")  
 public List<User> addUsers(@RequestBody List<User> users)  
 {  
 return service.saveUsers(users);  
 }  
  
 @GetMapping("/users/all")  
 public List<User> findAllUsers()  
 {  
 return service.getUsers();  
 }  
  
 @GetMapping("/user")  
 public User findUserById(@RequestParam int id)  
 {  
 return service.getUserById(id);  
 }  
  
 @GetMapping("/username")  
 public User findUserByUserName(@RequestParam String userName)  
 {  
 return service.getUserByUserName(userName);  
 }  
  
 @PutMapping("/user")  
 public User updateUser(@RequestBody User user)  
 {  
 return service.updateUser(user);  
 }  
  
 @DeleteMapping("/user/{id}")  
 public String deleteUser(@PathVariable int id)  
 {  
 return service.deleteUser(id);  
 }  
}

Worker Controller

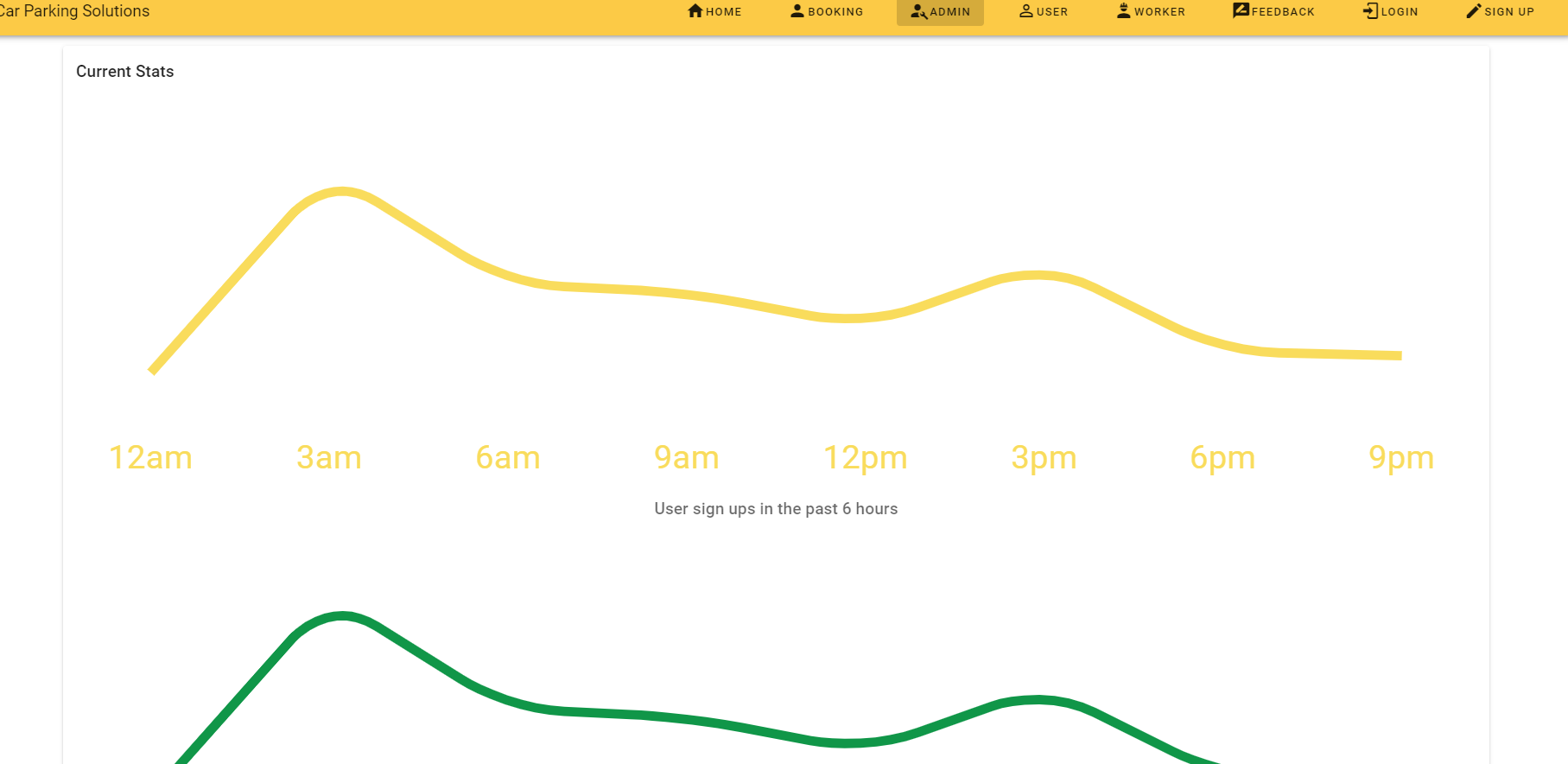
@RestController  
public class WorkerController  
{  
 @Autowired  
 private WorkerService service;  
  
 @PostMapping("/worker")  
 public Worker addWorker(@RequestBody Worker worker)  
 {  
 return service.saveWorker(worker);  
 }  
  
 @PostMapping("/workerlist")  
 public List<Worker> addWorkers(@RequestBody List<Worker> workers)  
 {  
 return service.saveWorkers(workers);  
 }  
  
 @GetMapping("/worker")  
 public List<Worker> findAllWorkers()  
 {  
 return service.getWorkers();  
 }  
  
 @GetMapping("/worker/{id}")  
 public Worker findWorkerById(@PathVariable int id)  
 {  
 return service.getWorkerById(id);  
 }  
  
 @PutMapping("/worker")  
 public Worker updateWorker(@RequestBody Worker worker)  
 {  
 return service.updateWorker(worker);  
 }  
  
 @DeleteMapping("/worker/{id}")  
 public String deleteWorker(@PathVariable int id)  
 {  
 return service.deleteWorker(id);  
 }  
}

Front Page

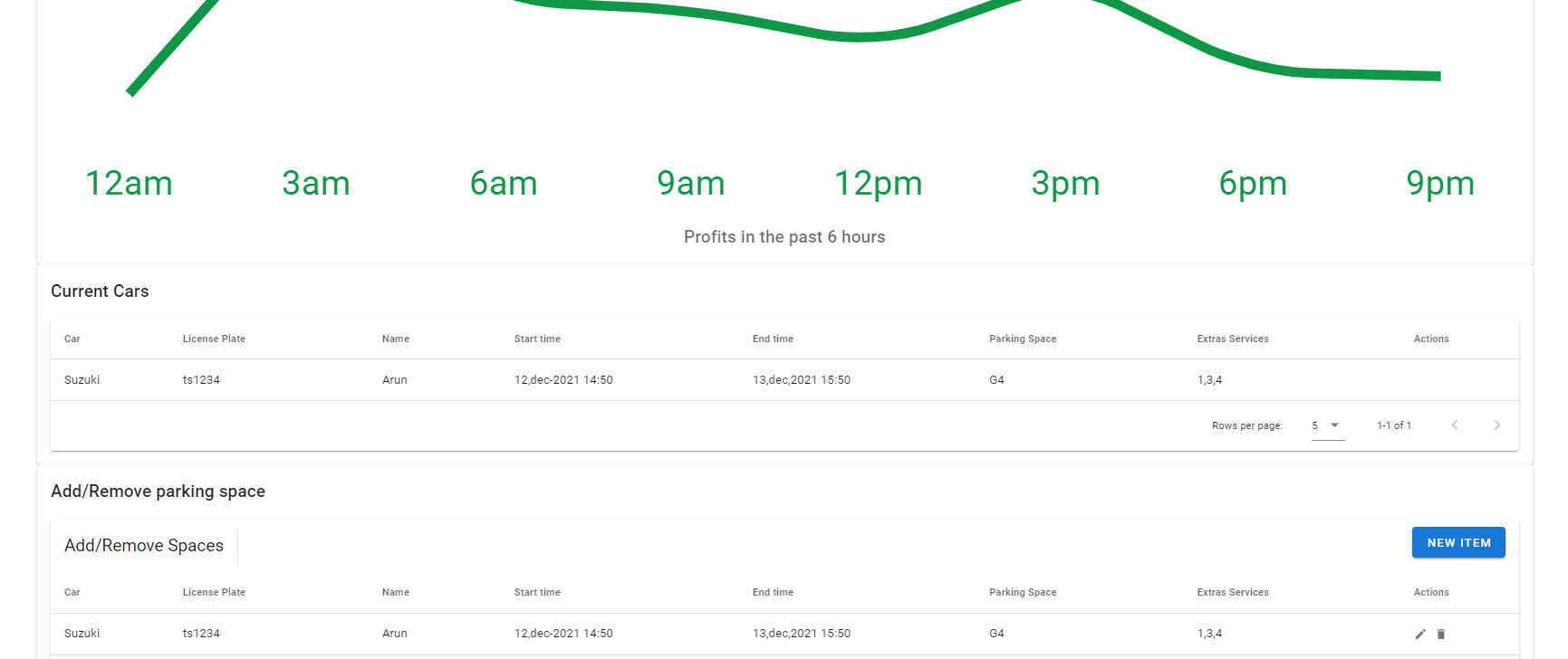
Booking Page



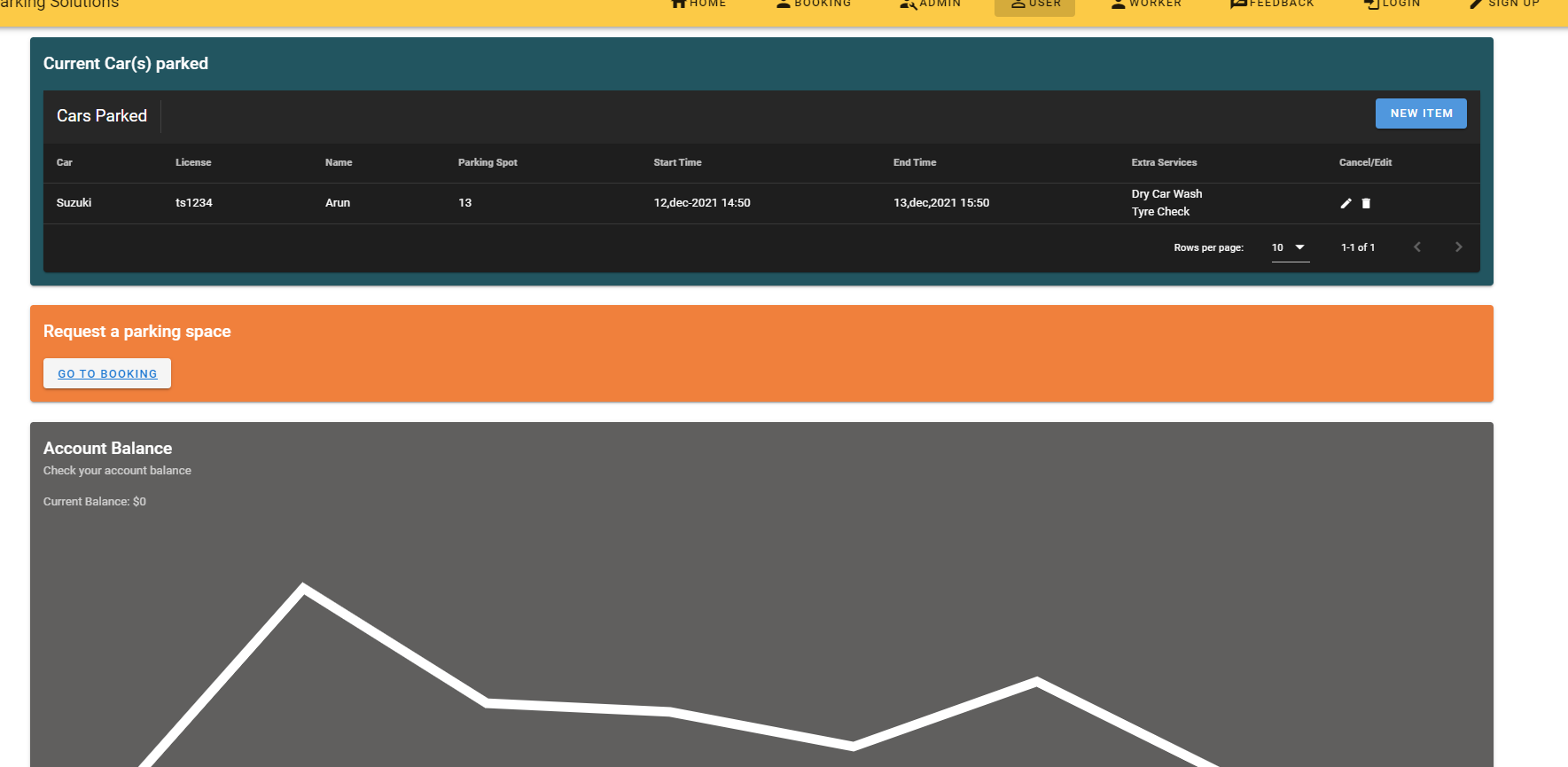
Admin page



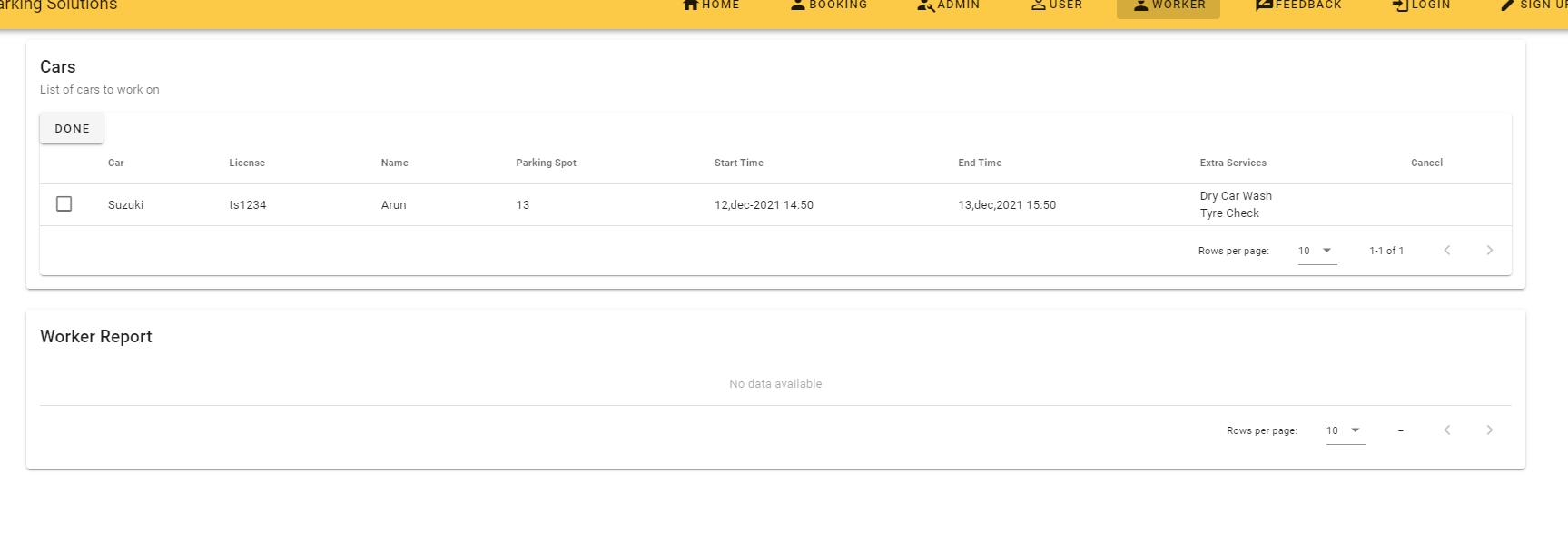
Admin page



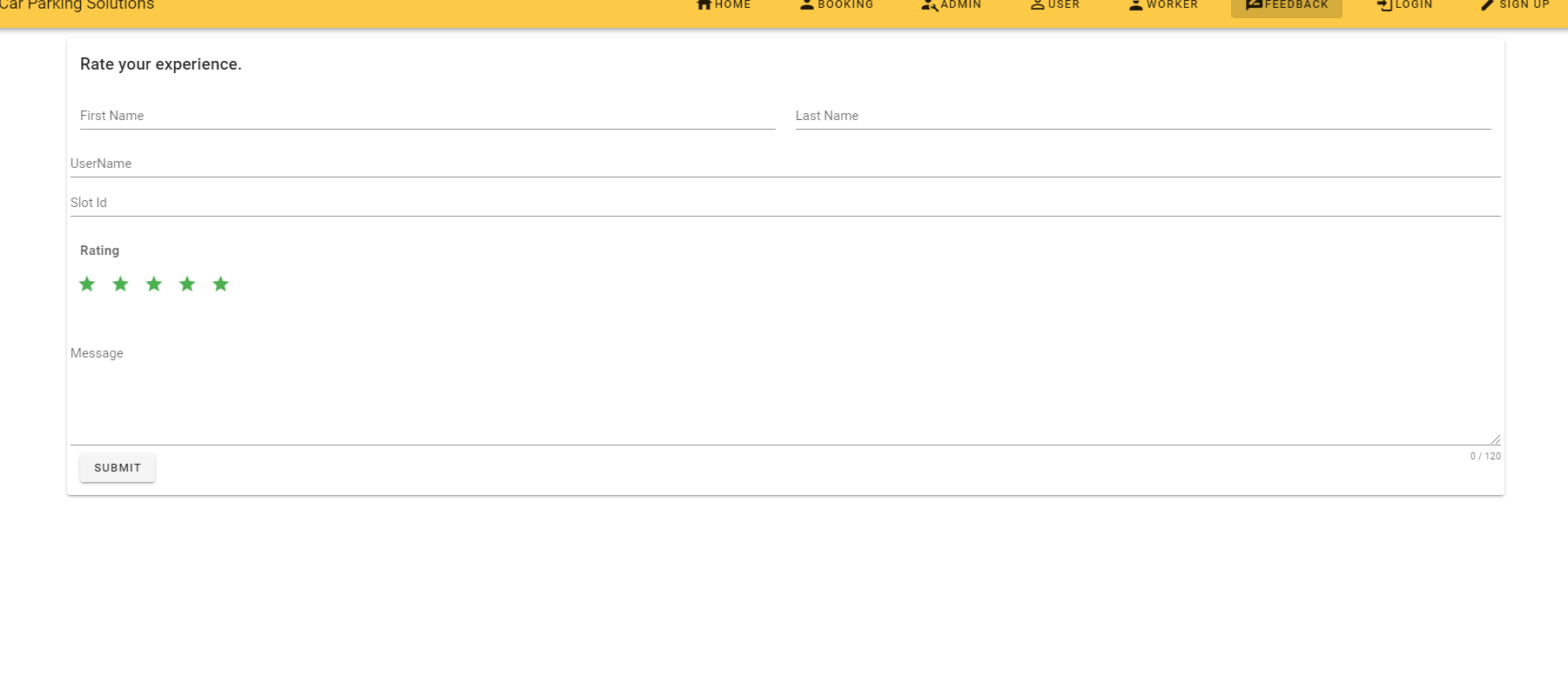
User page



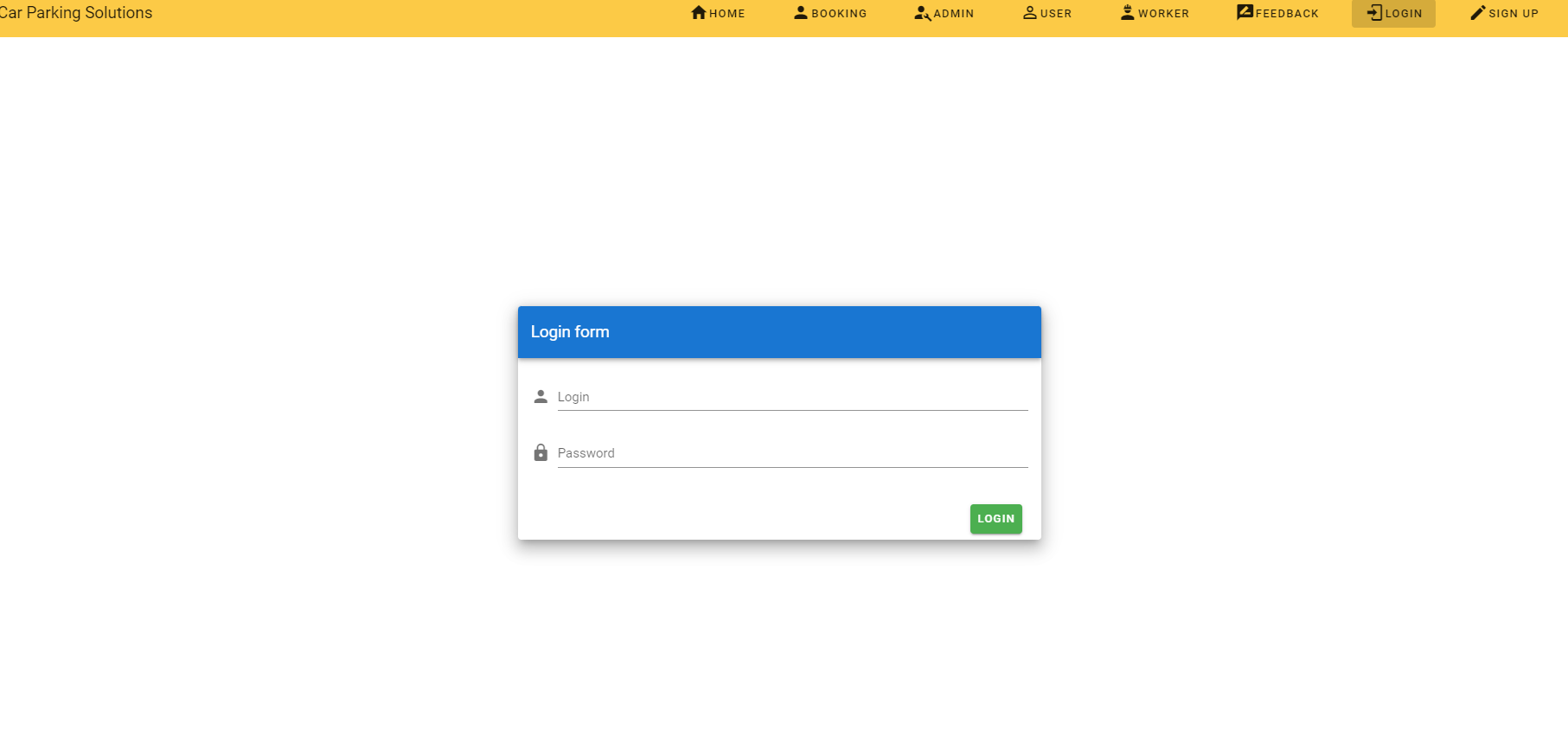
Worker page



Feedback page



Login page



Sign up page

