



Addis Ababa University
Department of Computer Science
Introduction to Software Engineering

Project Title: Parking Lot Management System

Prepared by:

- | | |
|-------------------|-------------|
| 1. Natnael Mesfin | UGR/8654/15 |
| 2. Mikiyas Fasil | UGR/9231/15 |
| 3. Yosef Solomon | UGR/7358/15 |

Submitted to: Ayalew Belay(PhD)

Submission date: December 9, 2024

Table of Content

| | |
|---|-----------|
| 1. Introduction..... | 3 |
| 1.1 Purpose of the document..... | 3 |
| 1.2 Scope of the project..... | 3 |
| 1.3 Document Overview..... | 3 |
| 1.4 Definition, acronym and abbreviations..... | 3 |
| 1.5 References..... | 4 |
| 2. Project Overview..... | 4 |
| 2.1 Overview..... | 4 |
| 2.2 Project Background..... | 4 |
| 2.3 Objectives and Success criteria..... | 5 |
| 2.4 Scope and Boundaries..... | 5 |
| 2.5 Assumptions and Dependencies..... | 6 |
| 3. Stakeholder Analysis..... | 6 |
| 3.1 Stakeholder Identification, Roles and Responsibilities..... | 7 |
| 4. Requirement Elicitation..... | 8 |
| 4.1 Elicitation techniques..... | 8 |
| 4.2 Documented Scenarios and Use Cases..... | 8 |
| Scenario 1: Admin Adds Employee..... | 8 |
| Scenario 2: Admin Removes Employee..... | 9 |
| Scenario 3: Admin to View Employees..... | 9 |
| Scenario 4: Employees and Admin View Status..... | 9 |
| Scenario 5: Admin View Records..... | 9 |
| Scenario 6: Check-In a Car..... | 10 |
| Scenario 7: Check-Out a Car..... | 10 |
| 5. Current System Analysis..... | 11 |
| 5.1 Overview of existing process..... | 11 |
| 5.2 System Architecture and Process flow..... | 11 |
| 5.3 Strength and Limitation of current system..... | 12 |
| 6. Requirement Analysis and Specification..... | 13 |
| 6.1 Functional Requirements..... | 13 |
| 6.2 Non-Functional Requirements..... | 13 |
| 7. System Model..... | 14 |
| 7.1 Use case Model..... | 14 |
| 7. 2 Use case description..... | 14 |
| 7.3 Dynamic Model..... | 28 |
| 7.4 Associations and Relationships..... | 37 |
| 7.5 User Interface and Navigation..... | 38 |
| 8. Glossary..... | 44 |

1. Introduction

1.1 Purpose of the document

The purpose of this requirement analysis document is to systematically understand and record the needs and expectations of stakeholders for the parking lot management system. This process helps identify the specific features, functionalities, and constraints essential for creating a system that supports both admin and employee operations, such as managing parking slots, performing check-ins and check-outs, and generating reports.

1.2 Scope of the project

The scope of the parking lot management project includes supporting both employee and admin profiles to enable seamless management of parking slot reservations. It provides employees with real-time availability of parking spaces to easily reserve a slot, and ensures that payment calculation and display features are integrated. The system operates on a simple architecture that provides optimal performance. Admins can manage employees and slot reservations, while the system is designed to scale as needed to accommodate future growth. The system also incorporates a database to store information on available parking slots and retrieve it when admin and employees request access, which allows a smooth parking experience.

1.3 Document Overview

This document essentially describes the purpose, scope and processes incorporated in the proposed system. It serves as a foundational guide for system users to understand the goals, scope and boundaries, and other core ideas of the project while ensuring alignment between problem-solving needs and technical implementation. It provides a blueprint for the development team with specific requirements to build an effective parking lot management system.

The document outlines both functional and nonfunctional requirements to clearly describe the essential tasks the system must perform, including space allocation, real-time monitoring, and security features. It also defines performance metrics, user accessibility standards, and system integration needs to ensure that the solution is robust, efficient, and user-friendly. Additionally, the document may include use case scenarios, workflows, and diagrams to illustrate the system's operation in real-world settings.

1.4 Definition, acronym and abbreviations

- ✓ **Admin:** refers to the administrator responsible for managing the parking lot, including reservation and employee management.
- ✓ **Employee:** refers to an individual in charge of overseeing parking lot status and managing the physical operations of the parking lot. This includes reserving a parking slot, assisting customers with parking, and ensuring completion of payment.
- ✓ **Customer:** refers to the individual who parks a car in a parking slot.

- ✓ **Parking Slot:** a designated space within a parking lot/area where vehicles are parked.
- ✓ **Reservation:** is the process of booking a specific parking slot for a particular time.
- ✓ **Parking Rates:** are the charges applied based on the time a vehicle occupies a parking slot.

1.5 References

GeeksforGeeks. (n.d.). Use case diagram, from <https://www.geeksforgeeks.org/use-case-diagram/>

Lacher, R. (n.d.). Rapid application development (RAD). Florida State University, from <https://www.cs.fsu.edu/~lacher/courses/COP3331/rad.html>

GeeksforGeeks. (n.d.). Activities involved in software requirement analysis, from <https://www.geeksforgeeks.org/activities-involved-in-software-requirement-analysis/>

ResearchGate. (n.d.). Software requirements engineering: Practical approach, from https://www.researchgate.net/publication/378310722_Software_Requirements_Engineering_Practical_Approach

Maciaszek, L. A. (2007). Requirements Analysis and System Design: Developing Information Systems with UML (3rd ed.). Pearson Education.

2. Project Overview

2.1 Overview

The proposed parking lot management software is designed to streamline the administration and operation of parking spaces. It ensures optimal resource utilization and smooth functionality for both employees and administrators. By keeping track of every vehicle through an integrated file management system, it enables real-time monitoring of available spaces while efficiently recording essential data such as entry time and identification credentials. This robust registration process facilitates accurate payment calculations and seamless checkouts.

For administrators, the system introduces structured features that enhance oversight capabilities. It allows effective monitoring of employees and provides tools to manage key processes such as billing and operational details. These systematic enhancements ensure a cohesive approach to parking lot management in order to address inefficiencies while creating a user-friendly and well-organized environment.

2.2 Project Background

Managing parking lots effectively is becoming a critical challenge, especially in urban areas where space utilization and efficiency are paramount. Traditional parking systems often suffer from inefficiencies such as manual customer registration, lack of real-time updates on space and other factors listed above. These issues include ineffective process, loss of revenue, and operational inefficiencies.

As such, our system seeks to address these challenges by introducing an automated and streamlined solution. By integrating features like real-time vehicle tracking, automated billing, and systematic administrative controls, the system ensures optimal resource utilization, smooth operations, and improved satisfaction. This project is designed to modernize the way parking lots are managed by providing a seamless experience for both customers, employees and administrators while reducing manual errors and inefficiencies.

2.3 Objectives and Success criteria

Objective

The objective of the parking lot management software is to improve the efficiency and effectiveness of managing parking spaces by enabling real-time monitoring of parking slots, accurate payment calculations, and smooth check-in/check-out processes. The system ensures seamless interaction between employees and administrators to streamline parking space allocation, registration, and billing. Additionally, it aims to enhance oversight for administrators by providing features for managing employee activities, ensuring effective monitoring, and simplifying reporting processes.

Another key objective is to provide optimal resource utilization, where employees can efficiently track available parking spaces while ensuring accurate data entry. The system is designed to be accessible, minimizing errors and reducing inefficiencies while offering scalability to accommodate future needs. Ultimately, the goal is to create a smooth, well-organized parking lot management environment that benefits both employees and customers while supporting growth and adaptability.

Success Criteria

The success of the parking lot management system will depend on its ability to provide a smooth user experience and maintain high performance. It should allow users to access critical functionalities like real-time parking slot monitoring and accurate billing without delays, ensuring operational efficiency. Additionally, the system must be flexible, able to evolve with changing needs, such as accommodating additional users or new features. Scalability is also vital, as the system must handle an increase in parking spaces and users without compromising performance. Together, these factors will contribute to the system's overall effectiveness, making it adaptable to growing demands while ensuring a reliable and efficient operation.

2.4 Scope and Boundaries

The parking lot management system aims to enhance operational efficiency by offering distinct functionalities for employees and administrators. Employees will benefit from real-time monitoring of parking slots, facilitating seamless slot reservations, check-in, and check-out processes while ensuring accurate billing. Administrators will have comprehensive control over employee management, slot allocation, and daily operational reporting to enable effective oversight. The system also integrates a robust database for real-time data storage and retrieval to support smooth workflows and reliable performance. Designed with scalability in mind, the system ensures adaptability for future needs while maintaining a convenient interface and streamlined processes for all involved beneficiaries.

Boundaries of the System

Third-Party Payment Integration: while the system facilitates payment calculations and displays billing details, it does not integrate with third-party payment gateways or financial institutions for processing payments.

Advanced Analytics and Reporting: the system will provide basic data retrieval for admins and employees but will not include advanced analytics, predictive modelling, or detailed and dynamic data analysis features.

External Access and Security: the system is designed for use by internal staff, including employees and admins. Customer portals or self-service applications for slot reservations are not part of this project. Additionally, comprehensive cybersecurity measures beyond standard database protections are outside the project scope.

2.5 Assumptions and Dependencies

▪ Assumptions

User Authentication

The system assumes that user profiles for employees and administrators will be securely created and authenticated before use to let them access based on their roles.

Data Accuracy and Integrity

The system assumes that the input data (such as vehicle identification and entry time) will be accurately entered into the system to ensure proper functionality and billing.

Scalability

It is assumed that the system is designed to scale according to future growth, to handle more parking slots, users, and data as needed.

▪ Dependencies

File Management System: the system relies on a database or file management system to store and retrieve real-time data, such as parking slot availability and transaction details.

Hardware Devices: The availability of digital devices for registration of customers' data and real-time tracking of parking slots for entry and exit is required.

3. Stakeholder Analysis

The stakeholders in this project range from the employees who handle daily operations, to the administrators responsible for overseeing the entire system and manage employees, and the customers who use the parking space. It defines employees and administrators as primary stakeholders who handle the management process. Customers are also included in stakeholders as they reserve parking slots and get smooth and efficient operations from the reservation to payment from employees. Additionally, there might be others involved including support team and investors with their own responsibilities to positively influence the effectiveness and success of the system.

3.1 Stakeholder Identification, Roles and Responsibilities

◆ Admin

The administrator is the primary stakeholder who manages employees, and monitors the activities of the parking lot management software. Given access to the records of the system, the admin is enabled to oversee and control the process flow efficiently.

Roles and Responsibilities:

- **Employee Management:** To register new employees, remove employees, and assign appropriate roles to ensure the right resources are available for smooth operations.
- **Supervision of Daily Operations:** To supervise daily records, monitor parking space availability, and make decisions on any issues related to space reservations or occupancy.
- **Billing and Payment Oversight:** To ensure that payment calculations and billing are accurate, that customers are correctly charged based on their duration of stay, and that there are no errors in the system.
- **System Monitoring:** To monitor the seamless operation of the system, including checking for any software malfunctions, overseeing real-time availability of parking spaces, and ensuring the registration process runs efficiently.

◆ Employee

Employees are also important stakeholders in the parking lot management system. They are responsible for implementing the day-to-day operations of the system. They interact directly with the customers and manage the proper allocation of parking slots, and control the check-in and check-out processes. Their role is vital to maintaining the smooth flow of operations and delivering an efficient parking experience to users.

Roles and Responsibilities:

- **Managing Parking Space Allocation:** Employees check the availability of parking slots in real-time and assign spaces efficiently to customers.
- **Customer Check-in and Registration:** During check-in, employees collect customer data including vehicle details and license plate information to enable the recording of necessary information for billing and parking management.
- **Real-time System Updates:** Employees update the parking management system in real-time, and accurately track slot usage and customer data to efficient parking operations.
- **Smooth Check-out Process:** Employees verify customer details and calculate the parking fee based on the duration during check-out to generate an accurate bill and release the occupied parking slot.

❖ Customers

While customers do not interact with the system directly, they are positively affected by the streamlined process of the parking lot management. They benefit from smooth as well as fast check-in and check-out processes, accurate billing calculation without confusion and delays and overall convenient parking experience.

4. Requirement Elicitation

4.1 Elicitation techniques

Observation

The primary elicitation technique used for this project was observation. During the observation, the focus was on how administrators managed parking lots and employees handled key tasks such as vehicle registration, monitoring available parking spaces, managing payment processes, and interacting with customers. This technique revealed the challenges in tracking parking space availability, manual entry errors, and the slow-paced nature of the check-in and check-out procedures. It also implied the lack of an effective administration and monitoring system. Observing these processes provided deep insights into system inefficiencies, and helped identify the pain points in the manual processes that hindered smooth operations, which formed the foundation for designing the new system.

Interviews

Although observation was the primary technique, interviews were also conducted with employees and administrators to gather additional perspectives on the current system's challenges. Interviewees discussed the need for real-time updates on available parking spaces, more accurate billing processes, and simplified employee management tools. The response gained helped to complement the observational data and enhance understanding of the system's shortcomings and the expectations for the new software solution.

4.2 Documented Scenarios and Use Cases

The following scenarios illustrate the key functionalities and interactions within the parking lot management system. They describe the actions taken by various actors, including administrators and employees, in different situations. Each scenario outlines step-by-step procedures to ensure seamless operations.

Scenario 1: Admin Adds Employee

Actor: Admin

Flow of events:

1. The admin logs in to the system using valid credentials.
2. The system authenticates the admin and displays the admin dashboard.
3. The admin selects the "Add Employee" option from the menu.
4. The system displays an employee registration form.
5. The admin fills out the form with the required details and submits it.

6. The system validates the input and stores the new employee record in the database.

Scenario 2: Admin Removes Employee

Actor: Admin

Flow of events:

1. The admin logs in to the system.
2. The system authenticates the admin and displays the dashboard.
3. The admin selects the "Remove Employee" option.
4. The system displays a list of employees.
5. The admin selects an employee to remove and confirms the action.
6. The system deletes the selected employee's record from the database.

Scenario 3: Admin to View Employees

Actor: Admin

Flow of events:

1. The admin logs in to the system.
2. The system authenticates the admin and displays the dashboard.
3. The admin selects the "Show Employee" option.
4. The system retrieves and displays a list of all registered employees along with their details.
5. The admin views the details.

Scenario 4: Employees and Admin View Status

Actor: Admin and Employees

Flow of events:

1. The admin or the employee logs in to the system.
2. The system authenticates the admin or the employee and displays the dashboard.
3. The admin or the employee selects the "View Status" option.
4. The system retrieves the real-time parking lot status, which includes available and occupied slots.
5. The admin or the employee reviews the information.

Scenario 5: Admin View Records

Actor: Admin

Flow of events:

1. The admin logs in to the system.
2. The system authenticates the admin and displays the dashboard.
3. The admin selects the "View Records" option.
4. The system retrieves and displays detailed historical data, including parking lot usage, duration and bill.
5. The admin reviews the records done.

Scenario 6: Check-In a Car

Actor: Employee

Flow of events:

1. The Employee logs in to the system.
2. The system authenticates the Employee and displays the dashboard.
3. The Employee selects the "Check-In" option.
4. The system prompts the Employee to enter customer details (e.g., name, license plate).
5. The Employee submits the details.
6. The system validates the data, assigns an available parking slot, and updates the database.
7. The Employee confirms the slot assignment to the customer.

Scenario 7: Check-Out a Car

Actor: Employee

Flow of events:

1. The Employee logs in to the system.
 2. The system authenticates the Employee and displays the dashboard.
 3. The Employee selects the "Check-Out" option.
 4. The system prompts the Employee to enter the license plate of the vehicle.
 5. The Employee submits the details.
 6. The system retrieves the check-in data, calculates the parking duration and bill, and updates the database to release the slot.
 7. The Employee collects payment and confirms the transaction to the customer.
-

5. Current System Analysis

As-Is Scenario Documentation

5.1 Overview of existing process

The current parking lot management system has several shortcomings that hinder efficiency and operational effectiveness. It is difficult for employees to monitor and manage space utilization effectively as they often lack access to real-time updates on available parking slots. The manual handling of the process along with inaccuracy of payment calculations and bill generation makes the process ineffective and prone to delays.

The system also struggles to calculate the exact duration of a vehicle's stay in real-time, leading to potential errors in billing. Additionally, customer registration is predominantly manual, which is time-consuming and increases the likelihood of human error. Finally, the system provides limited control and oversight for administrators, and it is challenging to enforce effective management practices.

5.2 System Architecture and Process flow

□ System Architecture

Digital Support:

- **Employee Interface:** Employees manually handle customer registrations, check-ins, and check-outs without systematic support for slot availability.
- **Admin Interface:** Admins lack efficient tools for monitoring parking lot activities and employee performance. There is no integrated system for tracking or reporting.

Database:

- There is manual storage of parking slot availability, customer details, and transaction records, and real-time updates are not available.
- There is no automated integration for payment calculation, so employees must calculate fees by hand.

Business Logic Layer:

- The process of checking available parking slots is manual and requires employees to physically check availability and assign slots.
- Payment calculations are also manual, leading to possible errors in billing due to human oversight.
- Employee performance tracking and parking lot space allocation are not integrated and are handled separately.

Payment and Billing:

- Payment calculations and bill generation are done manually by employees, increasing the chance of errors, especially when calculating time spent in the parking lot.

Admin Control Layer:

- Admins have limited visibility of the system's operational data and need to manually review reports to assess employee performance and parking usage.

□ Process flow

In the current parking lot management system, the process flow is largely manual and disconnected. Employees start by manually registering customers and checking parking slot availability. Once a slot is assigned, they track the vehicle's entry and exit times without real-time updates on availability. To calculate parking fees, employees manually determine the duration of stay, which can lead to errors in billing. Afterward, employees generate bills manually and process payments. Administrators, with limited system visibility, review manual reports for employee performance and overall operations, which hampers effective management and oversight. The entire process is slow and prone to human error.

5.3 Strength and Limitation of current system

❖ Strengths

Simplicity: The current system, being manual, is relatively easy to set up and doesn't require complex technology or infrastructure. This simplicity could be an advantage in a smaller environment.

Low Initial Cost: Since the system is largely paper-based or involves basic manual tools, it doesn't involve high upfront costs associated with software or hardware systems.

No Dependency on Technology: In areas with unreliable technical infrastructure, the current system doesn't rely on digital tools, which ensures operation everywhere.

❖ Limitations

Inefficiency: The major limitation of the current system is inefficiency. The lack of real-time updates for available parking spaces makes it difficult for employees to manage parking slots.

Error-Prone: The manual registration of customer details, parking slots, and payment calculations increases the likelihood of human errors, which could lead to incorrect billing or registration.

Lack of Real-Time Data: The absence of automated systems to track parking slot availability in real-time leads to delays in processing and decision-making. Employees and admins are unable to quickly assess the status of the parking lot.

Limited Control and Monitoring for Administrators: Admins have little oversight over the system due to the manual nature of record-keeping. This limits their ability to monitor employees, track performance, or generate real-time reports to help with decision-making.

Slow Payment Processing: Payment calculations and bill generation are mostly manual, which can be slow and prone to errors. This leads to inefficiencies during checkout and customer dissatisfaction.

Scalability Issues: As parking lot demand grows, the system becomes harder to manage without the ability to scale. More employees and manual records would be required and the organization may face potential challenges.

6. Requirement Analysis and Specification

6.1 Functional Requirements

Parking Slot Management: The system must track and update the status of parking slots in real-time, including their availability (available or reserved). This ensures employees and users are aware of current parking slot status at any given time.

Payment and Billing: The system should calculate charges based on the parking duration and generate corresponding bills for users.

Reports: The system must generate detailed, accurate reports on parking lot usage and the revenue generated over specified periods. These reports help administrators analyse performance and make data-driven decisions.

Data Handling: The system must securely store essential data in a database, including vehicle information, reservation details, and payment history. Administrators should have the ability to manage employee profiles, parking slots, and payment settings.

Real-Time Monitoring: Real-time tracking of parking slot occupancy status is required. The system must display this status to both users and administrators, ensuring that users can easily find available slots and administrators can monitor the usage efficiently.

6.2 Non-Functional Requirements

6.2.1 Ease of Use: The system should have a convenient interface for both employees and administrators. This interface should provide simple navigation and clear instructions for tasks such as payment, slot reservation, and report generation.

6.2.2 Reliability: The system must ensure consistent and accurate tracking of parking slot availability. It should operate without errors and minimize the likelihood of issues with billing or data entry.

6.2.3 Performance: Real-time updates of parking slot availability should occur quickly to ensure employees get accurate data of the system. The payment processing system must be capable of calculating duration and generating bills within seconds.

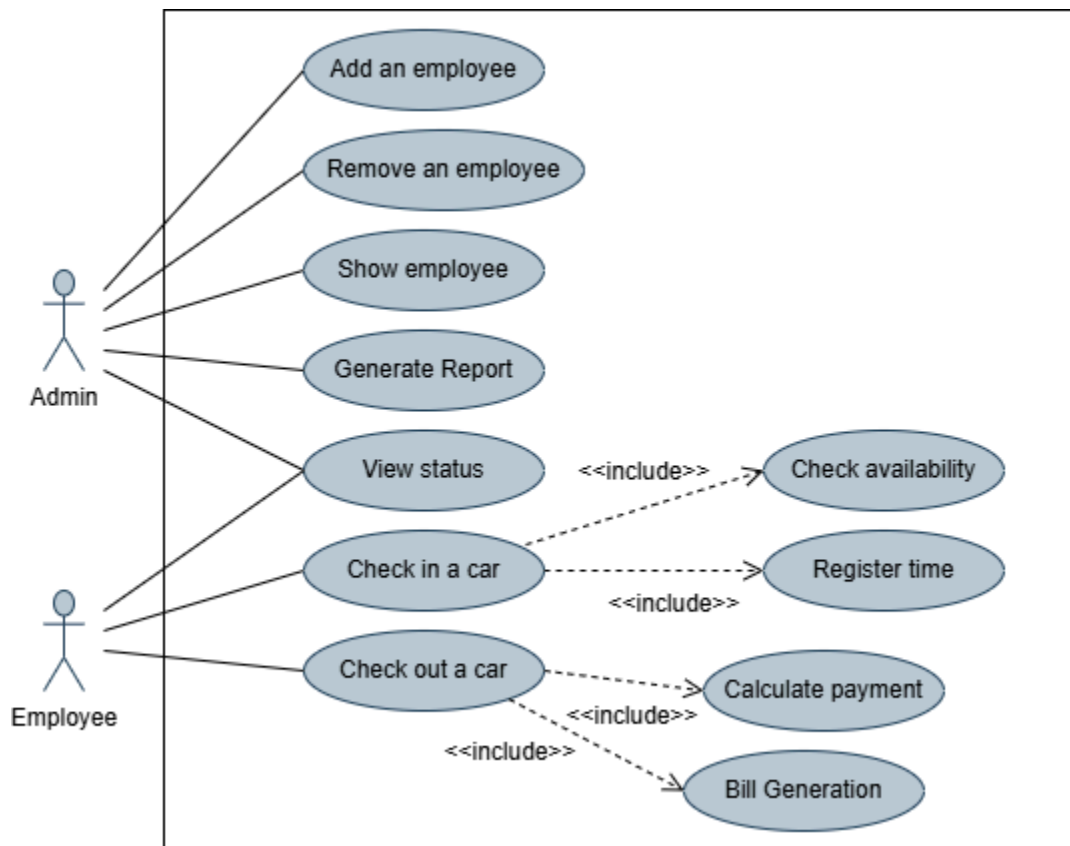
6.2.4 Scalability: The system should be designed to scale easily. As the demand for parking spaces increases, the system should be capable of supporting additional parking slots and possibly new locations without negatively affecting its performance.

6.2.5 User-Friendly Interface: The system should offer an easy-to-navigate interface with minimal steps required for employees to perform check-in and check-out operations, and for administrators to easily manage employees. It should be designed to facilitate a smooth user experience with clear options for completing tasks quickly.

7. System Model

7.1 Use case Model

Use case diagram



7.2 Use case description

Actor and Use Case Identification

Actors:

- **Admin:** The primary actor responsible for system administration

tasks, such as managing employees, generating reports, and changing system settings.

- **Employee:** Users who can check in vehicles, manage parking statuses, generate reports, and perform daily tasks related to parking management.
- **System:** The software that supports operations such as storing vehicle details, managing employees, and processing reports.

Use Cases:

- **Login to System**

Admin and **Employee** can log into the system to access their respective functionalities.

- **Check-In a Vehicle**

Employee performs the action of registering a vehicle, including capturing details like plate number, customer info, and parking spot allocation.

- **Check-Out a Vehicle**

Employee checks out vehicles, calculates the bill, generates a report, and updates parking information.

- **View Parking Status**

Admin or **Employee** can view current parking availability, checking in/out statuses, and manage parking spots.

- **Generate Daily Reports**

Admin can generate reports for daily activities, including check-ins, check-outs, and revenue calculations.

- **Change Password**

Employees can change their password for secure access to the system.

- **Add Employee**

Admin can add new employees to the system.

- **Remove Employee**

Admin can remove employees from the system when needed.

The following pages provide a detailed overview of the system's use cases, outlining the specific actions and interactions that users can perform within the system. Each use case is explained thoroughly, offering a clear understanding of the system's functionality and how it supports various user role

1. Use Case Description: Login to System (Admin)

Use Case Name: Admin Login to System

Actor: Administrator

Preconditions: 1. The system is operational and the admin has a valid username and Password.
2. The admin's credentials are already registered in the system.

Main Sequence

1. **Admin initiates login:**
 - The admin opens the application and selects the login option.
2. **System displays login form:**
 - A login interface with fields for username and password is displayed.
3. **Admin enters credentials:**
 - The admin provides the username and password and clicks the "Login" button.
4. **System validates credentials:**
 - The system verifies the username and password against stored admin records.
5. **Successful login:**
 - If credentials are valid, the admin is redirected to the admin portal/dashboard.
6. **Unsuccessful login:**
 - If credentials are invalid, the system displays an error message ("Invalid username or password") and allows the admin to retry.

Post conditions

1. If login is successful, the admin gains access to the admin portal with role-specific features.
2. If login fails after multiple attempts, the account may be locked (optional security feature).

Alternative Flow (Invalid Login)

Step 4 - Validation fails:

- If the username or password is incorrect, the system highlights the error.
- Optionally, an account lock mechanism is triggered after three failed attempts.

2. Use Case Description: Login to System (Employee)

Use Case Name: Employee Login to System

Actor: Employee

Preconditions: 1. The system is operational, and the employee has valid login credentials (username and password).
2. The employee's credentials are registered in the system database.

Main Sequence

1. **Employee initiates login:**
 - The employee opens the application and selects the login option.
2. **System displays login form:**
 - A login interface with fields for username and password is presented.
3. **Employee enters credentials:**
 - The employee enters their username and password, then clicks the "Login" button.
4. **System validates credentials:**
 - The system checks the username and password against the employee records in the database.
5. **Successful login:**
 - If the credentials are valid, the system grants the employee access to the employee portal/dashboard.
6. **Unsuccessful login:**
 - If the credentials are invalid, the system displays an error message ("Invalid username or password") and prompts the employee to retry.

Postconditions

1. If login is successful, the employee gains access to the employee portal with their respective functionalities.
2. If login fails after multiple attempts, the system will be locked.

Alternative Flow (Invalid

Login) 1. **Step 4 - Validation**

fails:

- The system detects incorrect credentials and informs the employee.
- After three failed attempts, the system might lock.

3. Use Case Description: Check-In a Vehicle

Use Case Name: Check-In Vehicle

Actor: Employee

Preconditions

1. The employee is logged into the system.
2. Parking slots are available.
3. The vehicle's details (e.g., plate number) and customer information (e.g., name and contact) are provided by the customer.

Main Sequence

1. **Employee selects Check-In:**
 - The employee navigates to the "Check-In" section in the employee portal.
2. **System displays the Check-In form:**
 - Fields for customer name, contact number, plate number, and available parking slots are shown.
3. **Employee inputs customer and vehicle details:**
 - The employee enters the customer's name, phone number, vehicle plate number, and selects an available parking slot.
4. **System validates data:**
 - The system verifies that the parking slot is free and that all required fields are correctly filled.
5. **System assigns a parking slot:**
 - Upon successful validation, the system records the parking slot, check-in time, and customer details into the parkingInfo file.
6. **Confirmation:**
 - The system confirms the successful check-in by displaying a message ("Check-In Successful") with assigned parking details.

Postconditions

1. The customer's vehicle is successfully checked into an available parking spot.
2. The parkingInfo file is updated with the new entry, including customer details, plate number, check-in time, and assigned slot.

Alternative Flow (No Parking Slot Available)

Step 4 - Validation fails:

- If no parking slots are available, the system displays an error message ("No parking slots available") and halts the check-in process.

Step 5 - Resolution:

- The employee may advise the customer to wait until a slot becomes available or try again later.

Alternative Flow (Duplicate Plate Number) **Step 4 - Validation fails:**

- If a vehicle with the same plate number is already checked in, the system alerts the employee with a message ("Vehicle already checked in").

Step 5 - Resolution:

- The employee reviews the details or resolves the conflict with the customer.

4. Use Case Description: Check-Out Vehicle

Use Case Name: Check-Out Vehicle

Actor: Employee

Preconditions

1. The employee is logged into the system.
2. The vehicle's check-in details exist in the parkingInfo file.
3. The system is configured to calculate billing amounts.

Main Sequence

1. **Employee selects "Check-Out" option:**
 - The employee navigates to the "Check-Out" page from the portal.
2. **System displays check-out form:**
 - The system provides a form to input the vehicle's license plate number.
3. **Employee enters license plate number:**
 - The employee enters the plate number of the vehicle being checked out and submits the form.
4. **System validates the plate number:**
 - The system searches for the plate number in the parkingInfo file.
 - If found, the system retrieves the associated check-in details.
5. **System calculates billing amount:**
 - Using the check-in time and the current time, the system calculates the total bill.
6. **System updates parking records:**
 - The system removes the vehicle's record from parkingInfo.
 - A daily report entry is generated with details:
 - Customer Name
 - Phone Number
 - Plate Number
 - Check-In Time

- Check-Out Time
- Total Bill

7. System displays confirmation:

- The system shows the bill amount and confirms the successful check-out. 8.

Employee informs the customer:

- The employee communicates the bill amount and completes the transaction with the customer.

Postconditions

1. The vehicle is marked as checked out.
 2. The parking space is made available for other vehicles. 3.
- A record of the transaction is added to the daily report.

Alternative Flow (Invalid Plate Number)

Step 4 - Validation fails:

- If the plate number is not found in the parkingInfo file, the system displays an error message ("Plate number not found").
- The employee can re-enter the plate number or confirm with the customer.

Alternative Flow (Billing Error)

Step 5 - Billing fails:

- If the system encounters an error while calculating the billing amount, an error message is displayed.

Alternative Flow (File Write Error)

Step 6 - Parking record update fails:

- If the system fails to update the parkingInfo file or save the daily report, the system alerts the employee.

Step 7 - Retry or manual log:

- The employee can retry or log the details manually for later system updates.

5. Use Case Description: View Parking Status

Use Case Name: View Parking Status

Actor: Admin, Employee

Preconditions

1. The user (Admin or Employee) is logged into the system.
2. Parking status information (e.g., occupied and available slots) is stored in the system (parkingInfo file).

Main Sequence

1. **User selects "View Parking Status":**
 - o The user navigates to the "View Parking Status" option in the portal.
2. **System retrieves parking data:**
 - o The system reads the parkingInfo file to gather information about occupied and available slots.
3. **System displays the parking status:**
 - o Occupied slots, customer details (e.g., name, phone number, plate number), and the check-in time are shown.
 - o Available slots are also listed for reference.
4. **User reviews the parking status:**
 - o The user can scroll through specific slots or vehicles based on the displayed information.

Postconditions

1. The user successfully views the current parking status.
2. The parkingInfo file remains unchanged unless modified by another operation.

Alternative Flow (No Vehicles Parked)

Step 2 - Retrieval returns empty data:

- o If no vehicles are parked, the system displays a message ("No vehicles currently parked") instead of a status list.

Alternative Flow (File Access Error)

Step 2 - System fails to retrieve data:

- o If the parkingInfo file cannot be read due to an error, the system displays an error message ("Unable to fetch parking status. Please try again later.").

Step 3 - Resolution:

- o The user contacts system support or checks for issues with the file system.

6. Use Case Description: Generate Daily Reports

Use Case Name: Generate Daily Reports

Actor :Admin

Preconditions

1. The Admin is logged into the system.
2. Check-in and check-out data for vehicles are stored in the parkingInfo file.
3. The system has functionality to create and write files.

Main Sequence

1. **Admin selects "Generate Daily Reports":**
 - The Admin navigates to the "Generate Reports" option in the portal.
2. **System retrieves parking data:**
 - The system reads the parkingInfo file and compiles records of vehicles that checked in or out during the current day.
3. **System processes the report:**
 - Data is formatted to include details such as:
 - Customer Name
 - Phone Number
 - Plate Number
 - Check-In Time
 - Check-Out Time
 - Bill Amount
 - The system calculates bill amounts for completed transactions.
4. **System saves the report:**
 - The report is saved as a file in a designated location.
5. **System displays a confirmation:**
 - The Admin is informed that the report was successfully generated and its location.
6. **Admin reviews the report:**
 - The Admin can choose to view the report.

Postconditions

1. A daily report is successfully created, containing all necessary details.
2. The report is saved for future reference.

Alternative Flow (No Transactions for the Day)

Step 2 - Retrieval returns no data:

- If no check-ins or check-outs occurred on the current day, the system generates an empty report.

Alternative Flow (File Save Error)

Step 4 - File save fails:

- If the system is unable to save the report (e.g., due to file system restrictions), an error message is displayed.

Step 5 - Resolution:

- The Admin is prompted to retry.

7. Use Case Description: Change Password (Employee)

Use Case Name: Change Password

Actor: Employee

Preconditions

1. The employee is logged into the system.
 2. The employee has access to the "Change Password" functionality.
 3. The employee knows their current password.
-

Main Sequence

1. **Employee selects "Change Password" option:**
 - The employee navigates to the "Account Settings" or "Profile" section in the system and selects the "Change Password" option.
2. **System displays change password form:**
 - The system prompts the employee to enter their current password, new password, and confirm the new password.
3. **Employee enters current and new password:**
 - The employee enters their current password, a new password, and then confirms the new password in the provided fields.
4. **System validates the current password:**
 - The system checks if the entered current password matches the one stored in the system.
 - If the current password is incorrect, the system displays an error message ("Current password is incorrect.") and prompts the employee to try again.
5. **System checks new password criteria:**
 - The system checks whether the new password meets the required criteria.
 - If the new password does not meet the criteria, the system displays an error message and prompts the employee to enter a valid new password.
6. **Employee confirms the new password:**
 - If the system validates the new password successfully, the employee confirms the new password by entering it in the "Confirm New Password" field.

7. System updates password:

- Once the new password is successfully confirmed, the system updates the password in the employee's profile.

8. System displays success message:

- The system displays a message confirming that the password has been changed successfully.
- The employee stays logged in with the new password.

9. Employee completes process:

- The employee returns to the main dashboard or profile page, now using the updated password.

Postconditions

1. The employee's password is successfully updated in the system.
2. The employee can use the new password for subsequent logins.
3. The old password is no longer valid.

Alternative Flow (Invalid Current Password) Step 4 - Current password is incorrect:

- If the employee enters an incorrect current password, the system displays an error message ("Current password is incorrect.") and allows the employee to retry entering the correct current password.

Alternative Flow (Password Doesn't Meet Criteria)

Step 5 - New password doesn't meet criteria:

- If the new password doesn't meet the required criteria, the system displays an error message ("Password does not meet criteria.") and prompts the employee to enter a password that satisfies the requirements.

Alternative Flow (Password Confirmation Mismatch) Step 6 - Passwords do not match:

- If the new password and the "Confirm New Password" fields do not match, the system displays an error message ("Passwords do not match.") and prompts the employee to re-enter the password correctly.

Alternative Flow (System Error)

Step 7 - System error during password change:

- If there is an error in updating the password in the system (e.g., database error), the system displays an error message ("An error occurred while updating your password.") and suggests retrying the action.

8. Use Case Description: Add Employee (Admin)

Use Case Name: Add Employee

Actor: Admin

Preconditions

1. The admin is logged into the system with the appropriate admin privileges.
 2. The admin is on the "Employee Management" or "Admin Dashboard" page, with access to the "Add Employee" option.
 3. The system is connected to a database or storage where employee information is kept.
-

Main Sequence

1. **Admin selects "Add Employee" option:**
 - The admin clicks the "Add Employee" button or link in the system, navigating to the employee creation form.
2. **System displays employee creation form:**
 - The system shows a form with required fields for entering employee details, such as:
 - Full Name
 - User name
 - Password (temporary or set by the admin)
3. **Admin enters employee details:**
 - The admin fills in all required fields on the form with the new employee's information.
4. **System validates input data:**
 - The system checks the validity of the entered information:
 - Verifies that mandatory fields are not left blank.
 - Ensures the password meets security criteria.
5. **Admin submits the form:**
 - After reviewing the details, the admin clicks "Submit" to add the new employee.
6. **System creates employee record:**
 - The system stores the new employee's information in the database or storage.
 - A unique employee ID is assigned to the new employee.
7. **System sends confirmation:**
 - The system displays a success message ("Employee successfully added.") confirming that the new employee has been added.

Postconditions

1. The new employee's information is stored in the system and accessible via the employee management system.
2. The new employee can log in with their credentials if applicable
3. The new employee's role, position, and contact information are correctly displayed in the system.

Alternative Flow (Invalid Input)

Step 4 - Invalid input data:

- o If the admin enters invalid data the system displays error messages indicating which fields need correction.

Alternative Flow (System Error)

Step 6 - System error during employee creation:

- o If an error occurs when saving the employee's information in the database, the system displays an error message ("An error occurred while adding the employee.") and suggests retrying the operation.

Alternative Flow (Admin Cancels Action)

Step 1 - Admin cancels the process:

- o If the admin decides not to proceed with adding the employee, they can cancel the action. The system returns the admin to the "Admin portal page" page without making any changes.

9. Use Case Description: Remove Employee (Admin)

Use Case Name: Remove Employee

Actor: Admin

Preconditions

1. The admin is logged into the system with the appropriate admin privileges.
2. The admin is on the Admin portal page, with access to the "Remove Employee" option.
3. The system contains a list of existing employees.

Main Sequence

1. **Admin selects an employee to remove:**

- The admin navigates to the Admin portal page
- The admin selects the employee to be removed by searching for the employee using their username.
- 2. System asks for confirmation:**
 - The system displays a confirmation prompt, asking the admin to confirm the removal of the employee. The prompt will display the employee's name.
- 3. Admin confirms removal:**
 - The admin clicks "Confirm" or "Yes" to proceed with removing the employee. Alternatively, the admin can cancel the operation.
- 4. System removes employee from the system:**
 - Upon confirmation, the system deletes the employee's record from the database, including their personal details, role, and login credentials.
 - The employee is no longer listed in the employee management system.
- 5. System displays success message:**
 - The system displays a success message (e.g., "Employee removed successfully.") confirming that the employee has been deleted from the system.
- 6. Admin is returned to the admin portal page:**
 - After the employee is removed, the system returns the admin to the admin portal page.

Postconditions

1. The employee's record is removed from the system, and they can no longer log into the system.
2. The system no longer displays the employee in the employee list or any related systems.
3. Any roles, permissions, or access previously assigned to the employee are removed from the system.

Alternative Flow (Employee Not Found)

Step 1 - Employee not found:

- If the employee is not found in the system (e.g., the employee was already removed or the ID is incorrect), the system displays an error message ("Employee not found.") and prompts the admin to search again or verify the employee's details.

Alternative Flow (Admin Cancels Action)

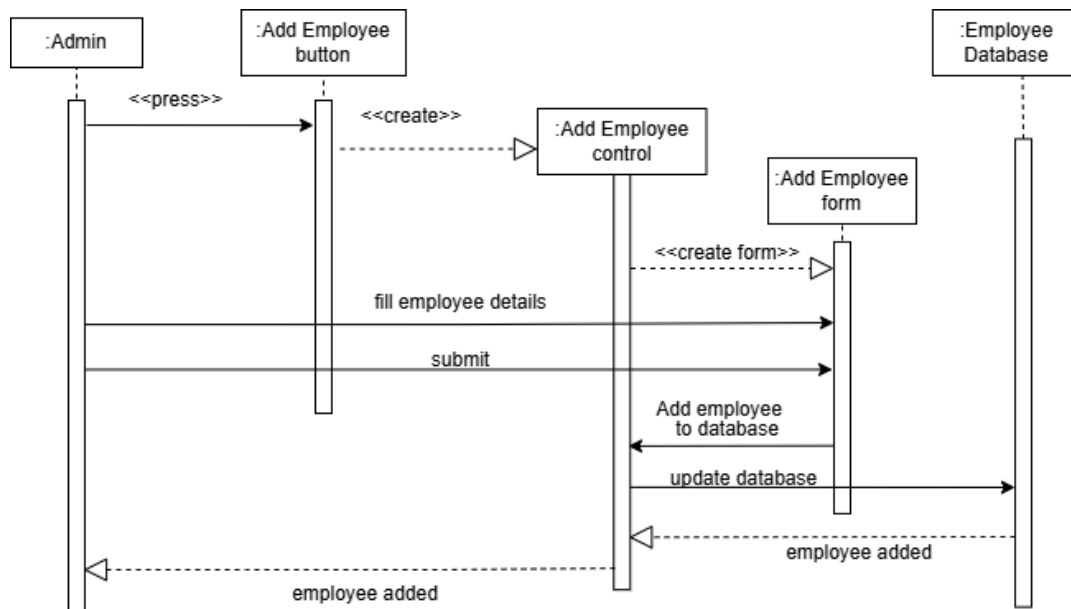
Step 3 - Admin cancels removal:

- If the admin decides not to proceed with the employee removal, they can cancel the action. The system will return them to the previous page without removing the employee.

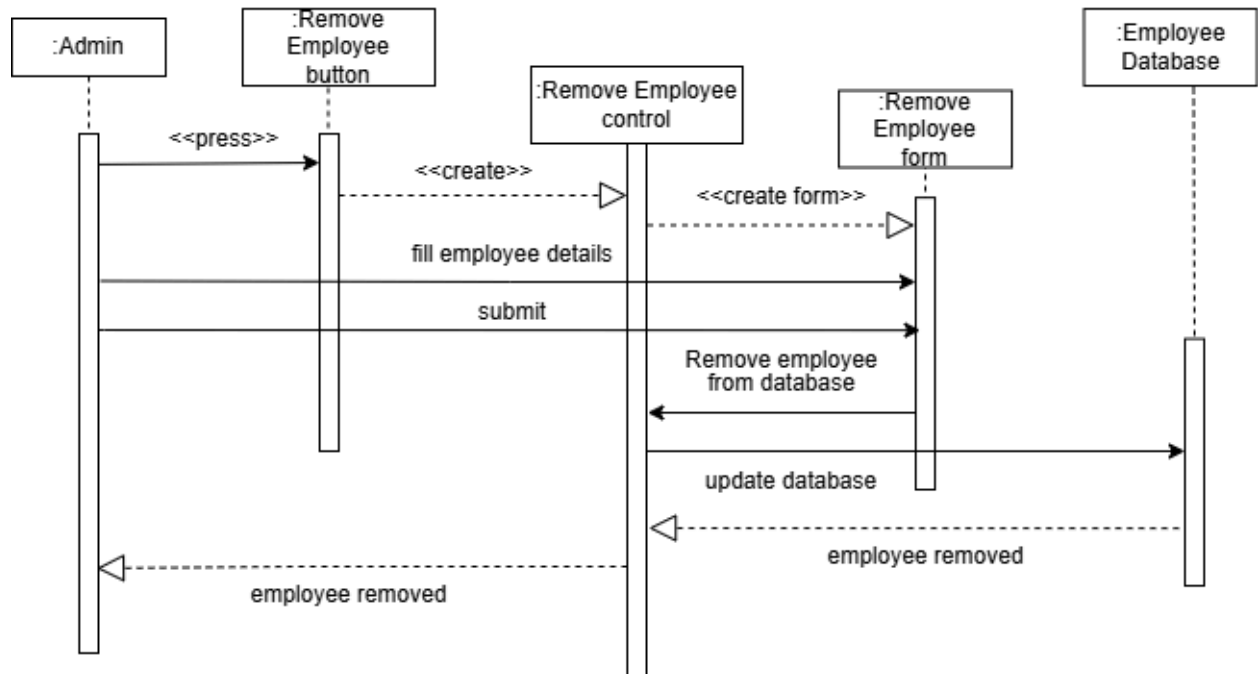
7.3 Dynamic Model

Sequence Diagrams

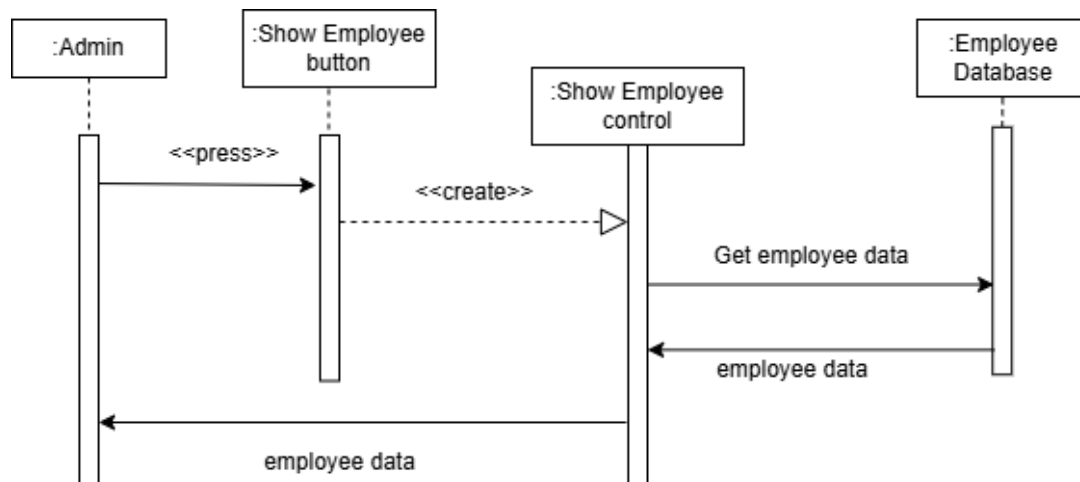
Add Employee Sequence diagram:



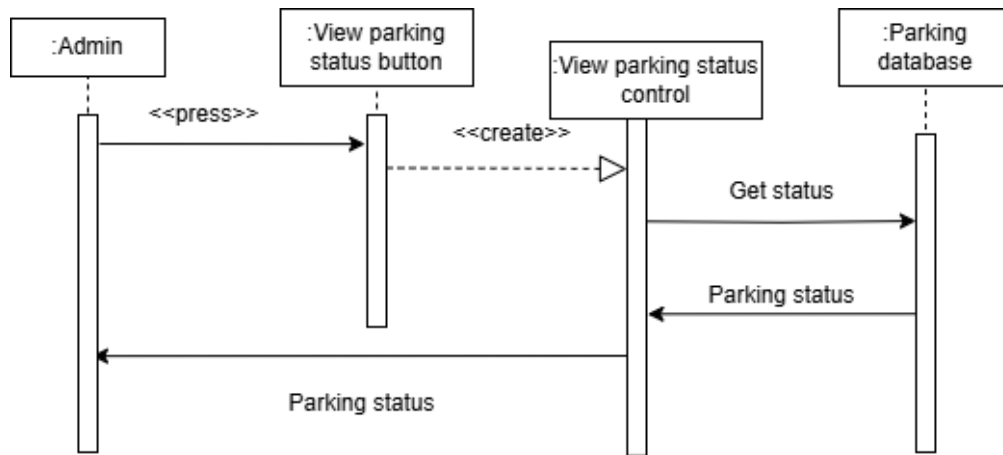
Remove Employee Sequence diagram:



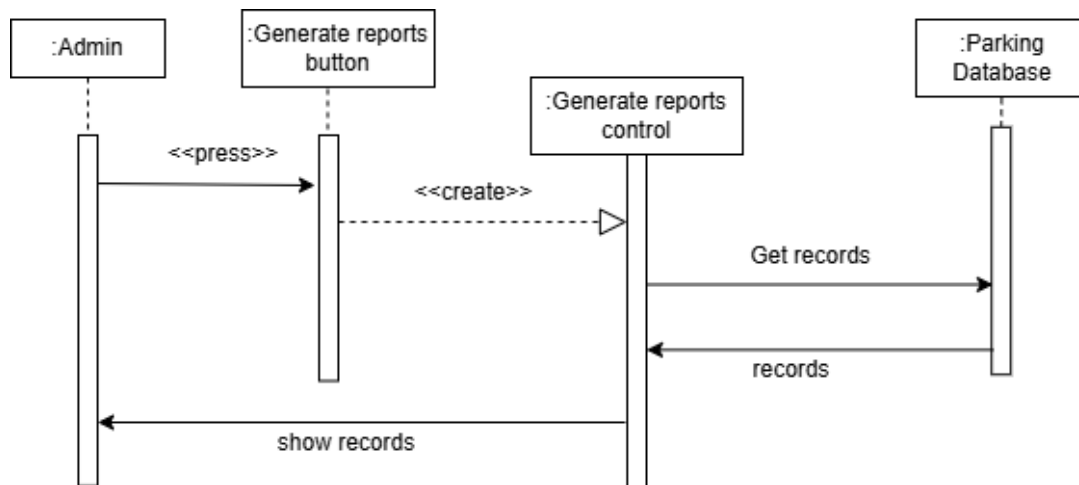
Show Employee Sequence diagram:



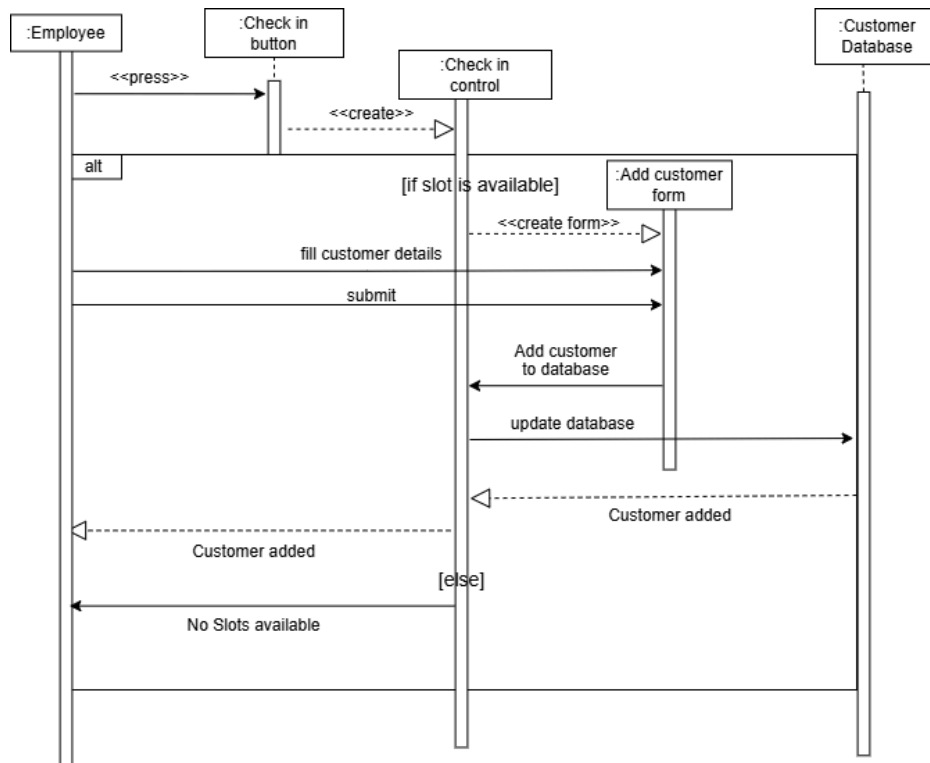
View Parking Status Sequence diagram:



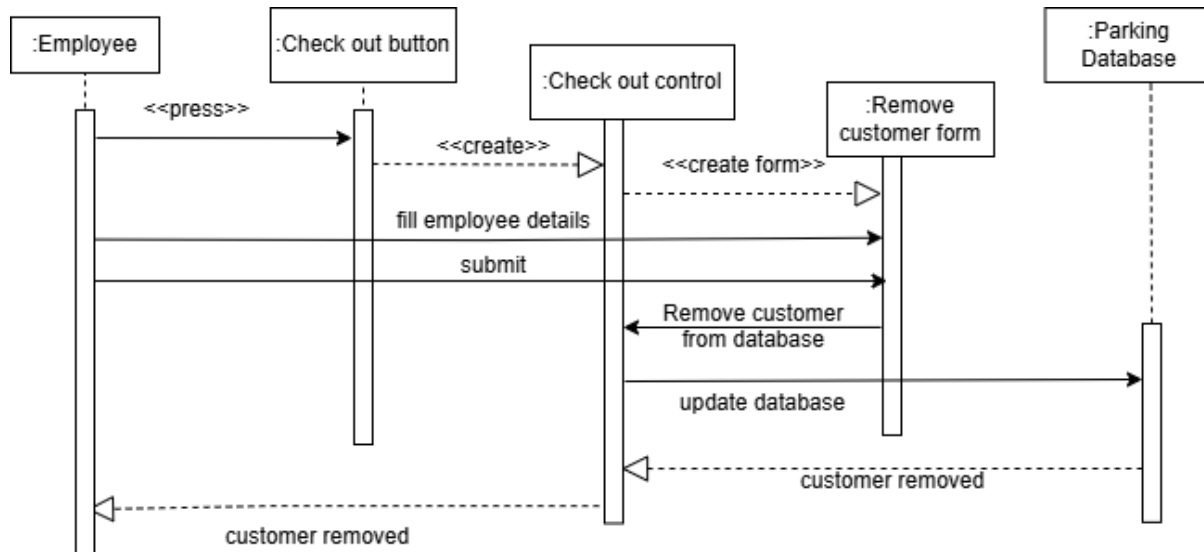
Generate Reports Sequence diagram:



Check in a car Sequence diagram:

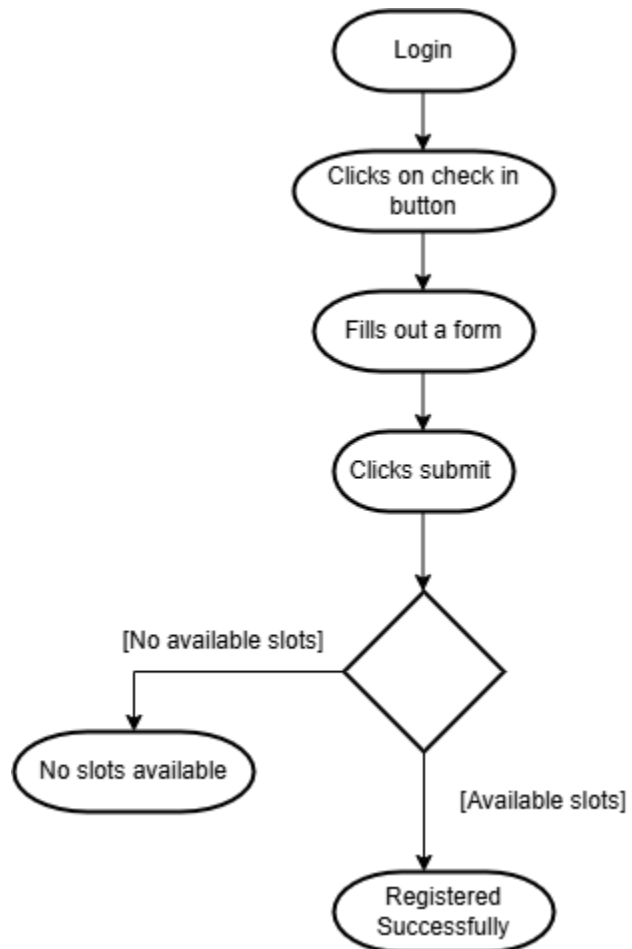


Check out a car Sequence diagram:

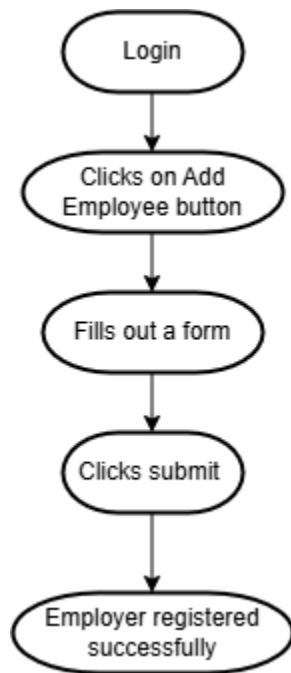


Activity Diagrams

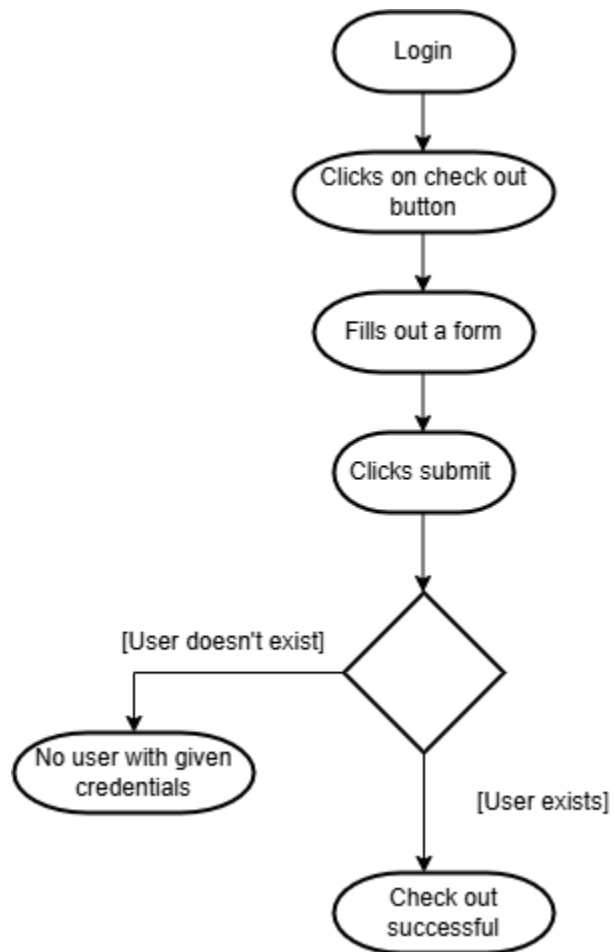
Check in:



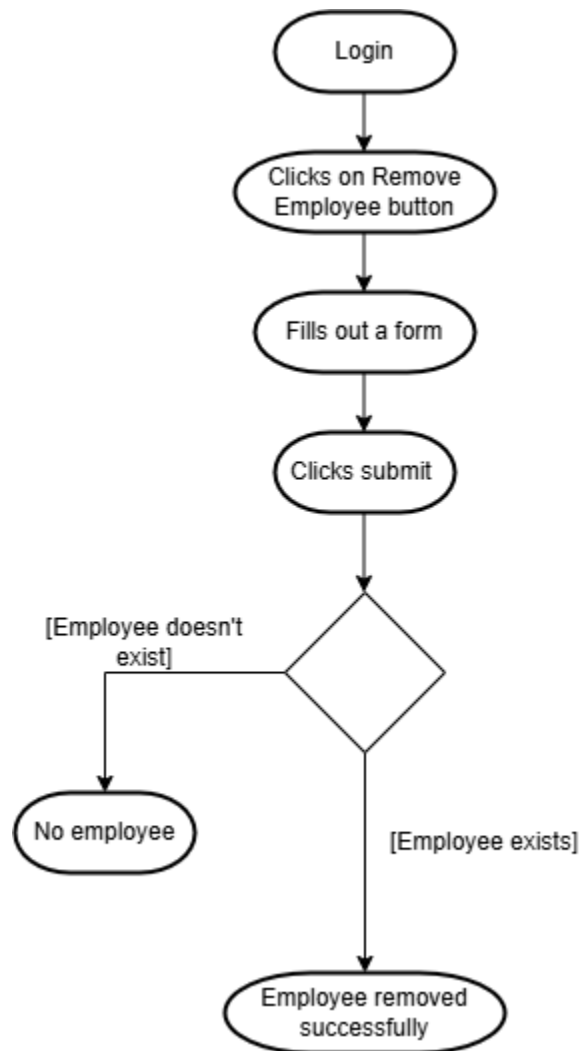
Add Employee:



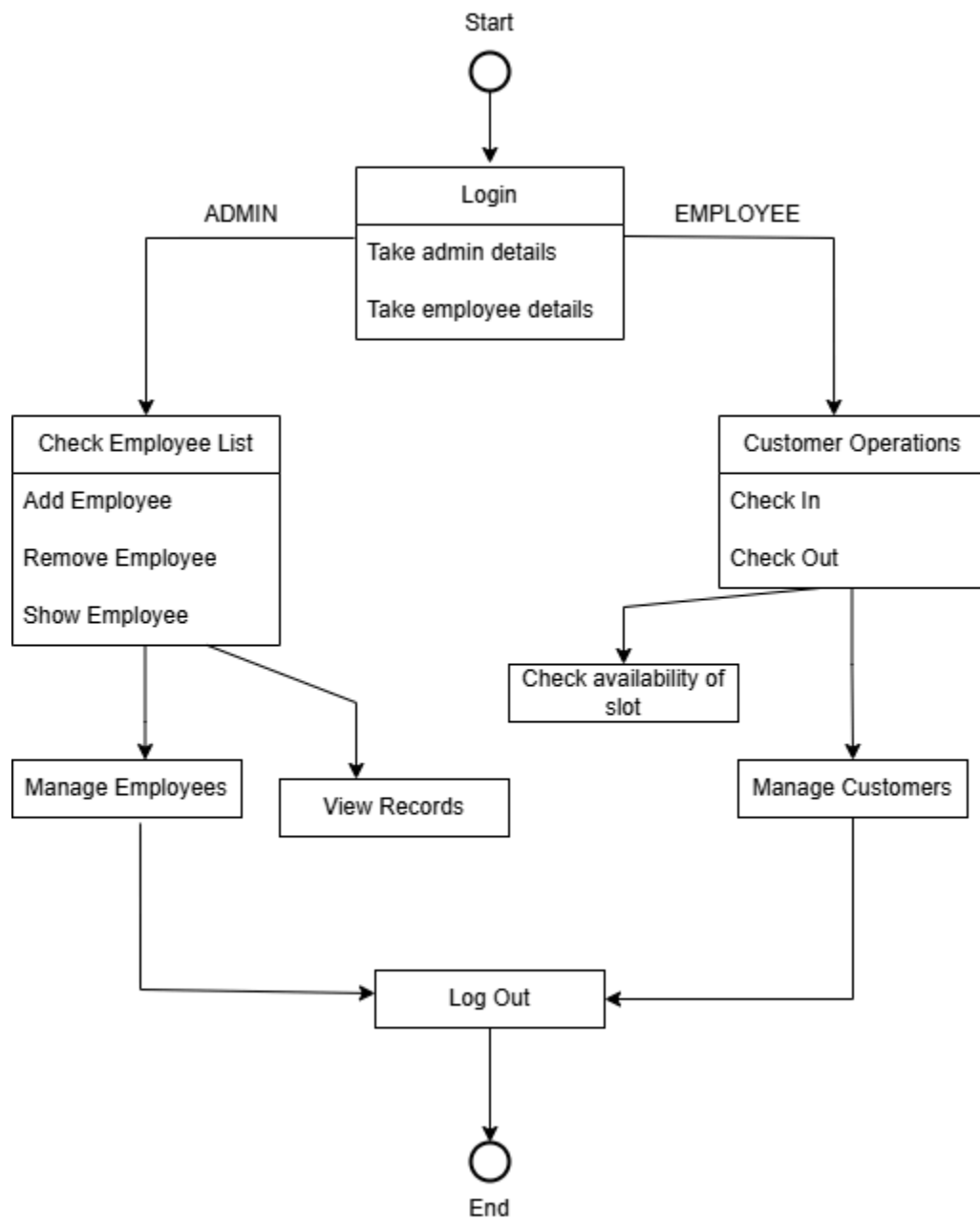
Check Out:



Remove Employee:

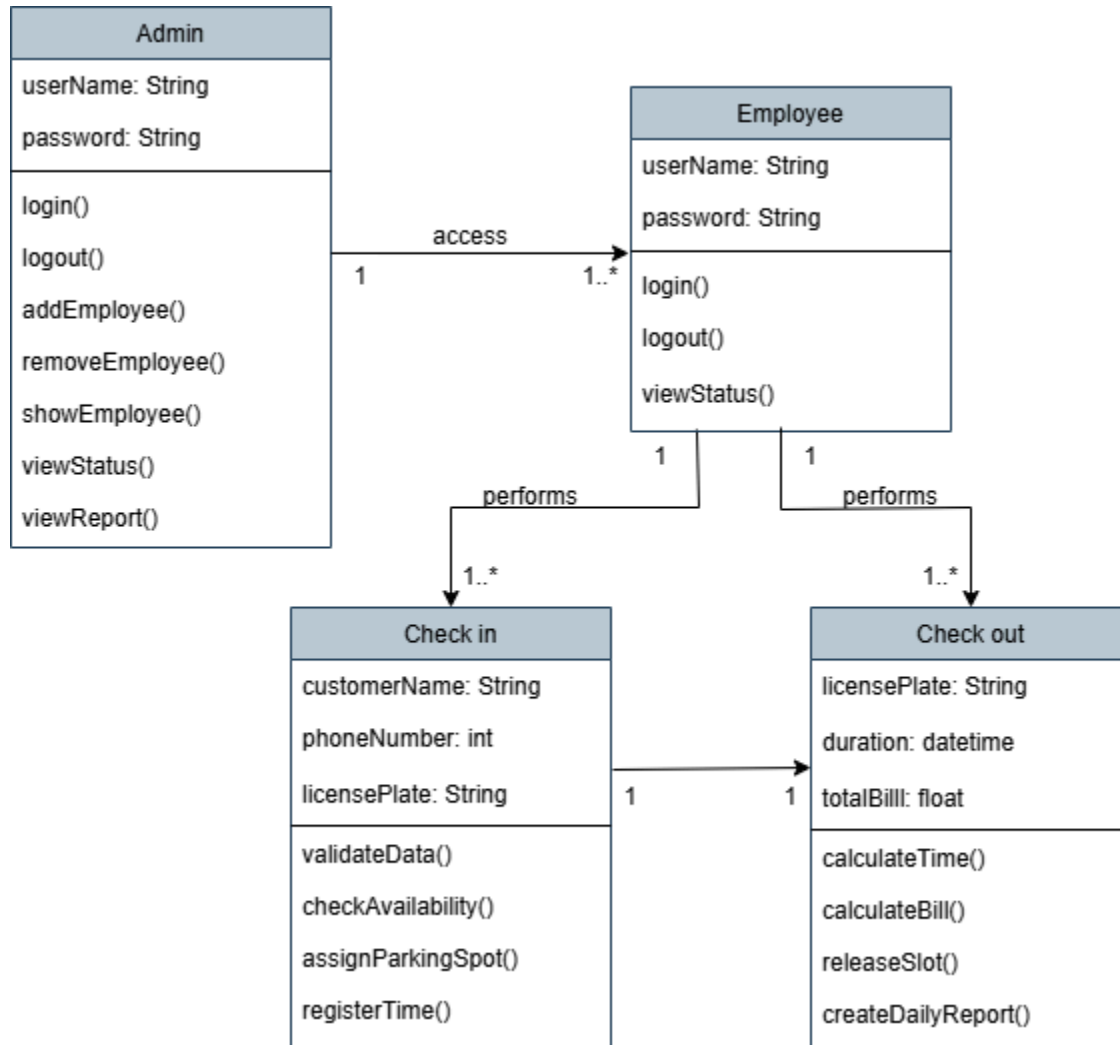


State-chart diagram:



7.4 Object Model

Class Diagram:



7.4 Associations and Relationships

Based on the provided class diagram, here is an analysis of the associations and relationships between the classes:

1. **Admin ↔ Employee:**

- **Relationship: One-to-Many (1..*)**
- **Description:** An Admin has the capability to manage multiple Employees. This includes adding, removing, or viewing employee details. The association is labeled as "access" in the diagram, meaning the admin oversees and manages employees.

2. **Employee ↔ Check-In / Check-Out:**

- **Relationship: One-to-Many (1..*)**
- **Description:** Each Employee can handle multiple check-in or check-out processes during their operation. This is depicted as "performs" in the diagram, meaning employees are responsible for managing these tasks.

3. Check-In ↔ Check-Out:

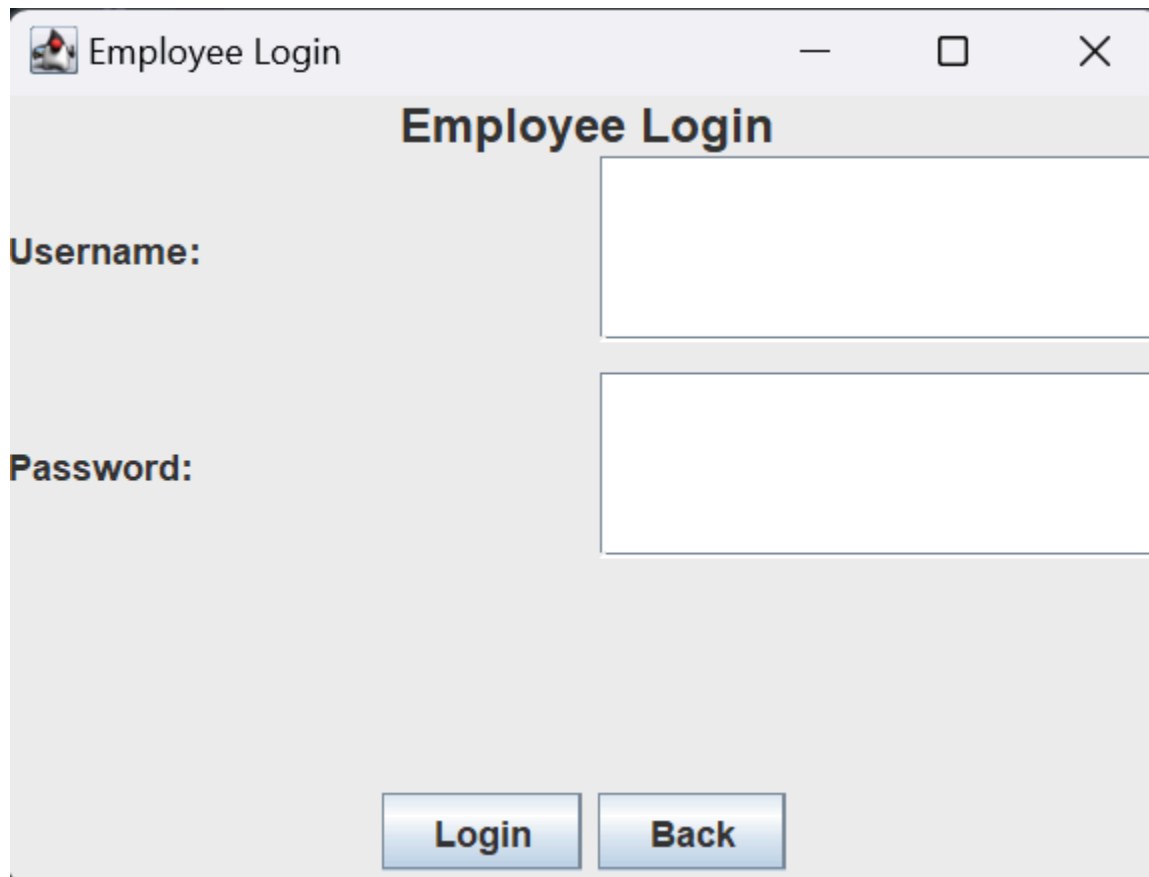
- **Relationship: One-to-One (1..1)**
- **Description:** Each Check-In process is directly tied to a single Check-Out process for a particular vehicle. This ensures consistency in tracking the same customer's parking details.

7.5 User Interface and Navigation

1. Home Page:



2. Employee login:



The image shows a graphical user interface for an "Employee Login" window. The window has a title bar with the text "Employee Login" and standard minimize, maximize, and close buttons. The main content area has a light gray background. At the top right of the content area, the title "Employee Login" is displayed in a large, bold, black font. Below this, there are two input fields. The first field is preceded by the label "Username:" in a bold, black font. The second field is preceded by the label "Password:" in a bold, black font. At the bottom of the window, there are two buttons: "Login" and "Back". Both buttons have a blue gradient and a 3D effect.

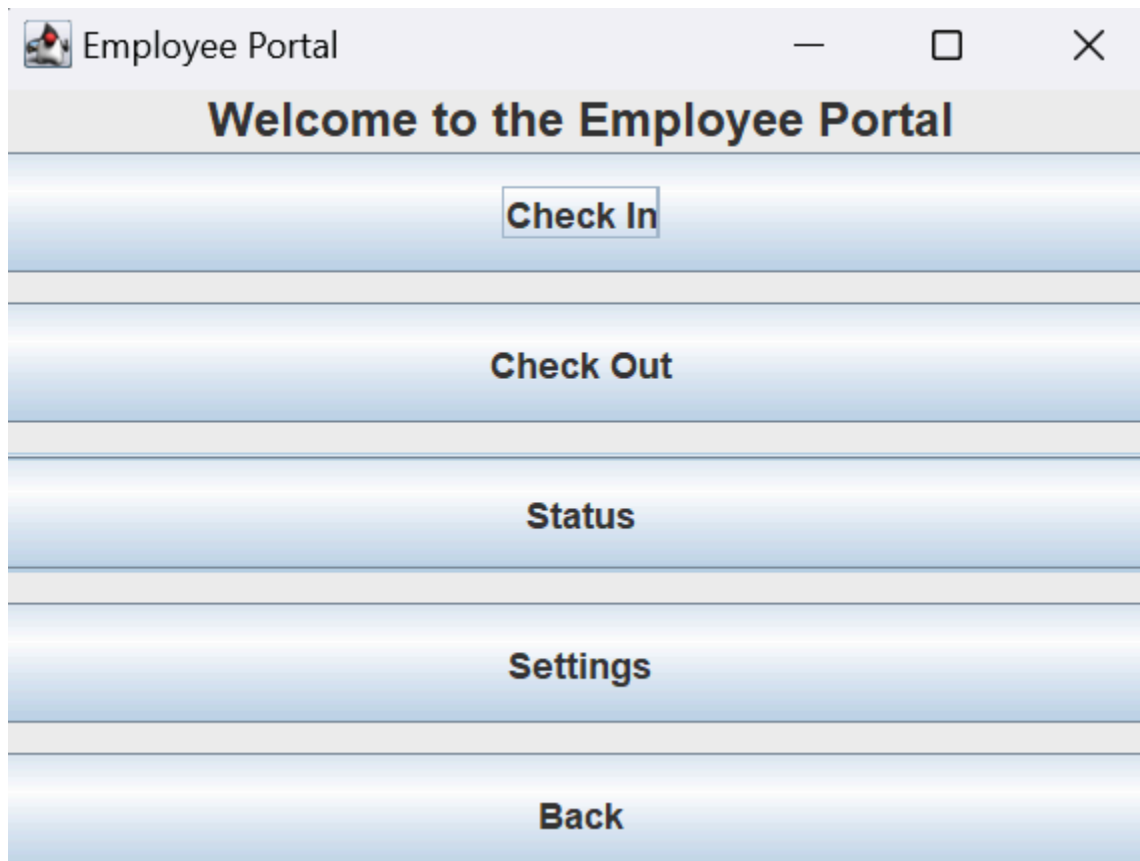
Employee Login

Username:

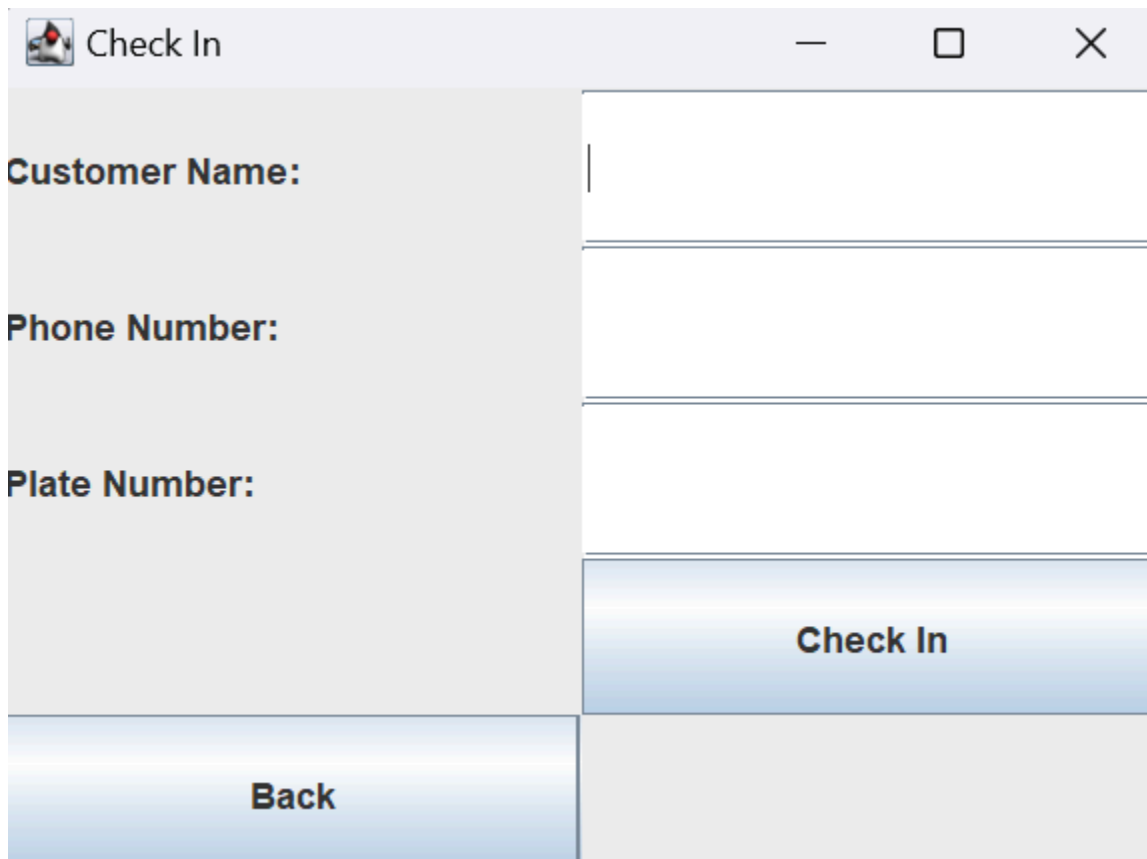
Password:

Login **Back**

3. Employee Portal:

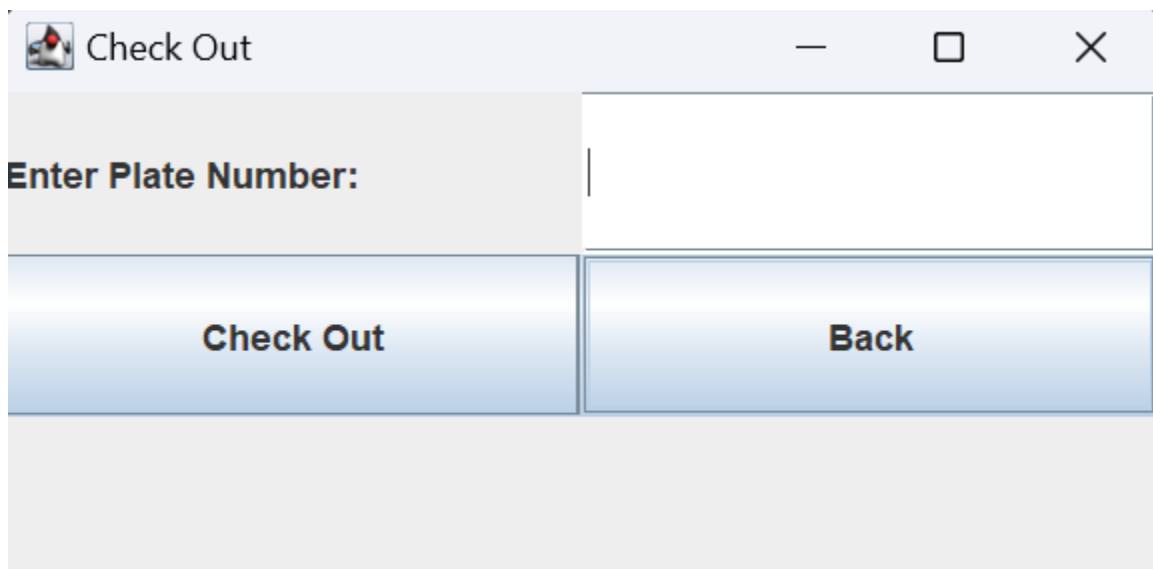


4. Check in:



A dialog box titled "Check In" with a standard Windows window header (minimize, maximize, close buttons). The dialog is divided into two main sections. The left section is a light gray area containing three labels: "Customer Name:", "Phone Number:", and "Plate Number:". The right section is a white area with three corresponding text input fields. Below the input fields is a blue button labeled "Check In". At the bottom left of the dialog is another blue button labeled "Back".

5. Check out:

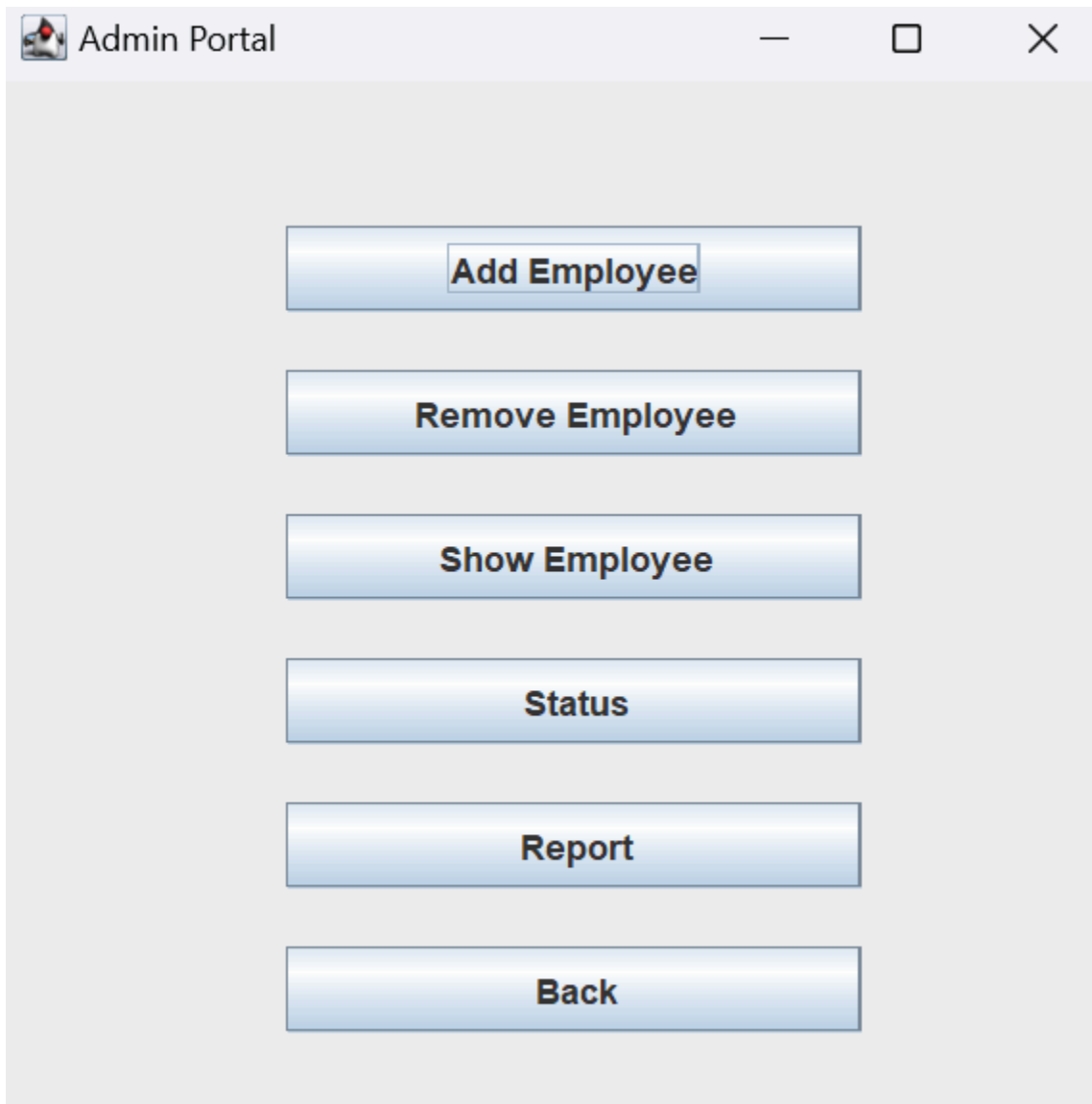


A dialog box titled "Check Out" with a standard Windows window header (minimize, maximize, close buttons). The dialog is divided into two main sections. The left section is a light gray area containing one label: "Enter Plate Number:". The right section is a white area with one corresponding text input field. Below the input field are two blue buttons: "Check Out" on the left and "Back" on the right.

6. Parking status Page

| Parking Status | | | | |
|---|--------------|--------------|--------------|---------------------|
| Back | | | | |
| Customer Name | Phone Number | Plate Number | Parking Spot | Check In Time |
| 123 | 123 | 123 | P1 | 2024-12-07 13:47:37 |
| Ab | 1234 | 1234 | P2 | 2024-12-07 13:53:05 |
| Kiki | 08324986 | 11111 | P3 | 2024-12-07 13:58:47 |
| yoyo | 08346543 | 2222 | P4 | 2024-12-07 14:01:10 |
| Man | 098367334 | 232 | P5 | 2024-12-07 14:12:35 |
| AAA | 0911223344 | 83464 | P6 | 2024-12-07 14:32:24 |
| Total Free Spots: 44 Total Taken Spots: 6 | | | | |

7. Admin Portal



8. Glossary

Parking Slot

Individual spaces within the parking area allocated for vehicles.

Parking lot

An area/space where cars are parked.

Ticketing System

A process of issuing parking tickets to record entry time and other relevant details.

Fee Structure

The pricing model used to calculate charges for parking based on factors like duration or type of vehicle.

Availability Tracking

Real-time monitoring of vacant and occupied parking slots.

Customer Registration

A feature allowing users to register their vehicles and details for parking, especially for repeat or subscription-based customers.

Payment System

A mechanism for collecting parking fees, supporting cash or digital transactions.

Access Control

The system managing vehicle entry and exit, often integrated with barriers or gates.

Reports and Analytics

Data summaries or dashboards offering insights into parking usage patterns, revenue, and overall efficiency.

Security Features

Measures, such as CCTV cameras or barriers, to ensure safety within the parking lot.

Real-Time Slot Updates

Instant updates on slot occupancy to ensure effective space utilization.

Exit Validation

Verification or payment confirmation required for vehicle departure.