Software Requirements Specification

Team 2

Drew Caldwell, Brady Covyeou,
Sukhkaran Gill, and Dustin Groh

C308 - Systems Analysis and Design

October 2024

Table of Contents

1.0 Introduction

1.1 Goals and Objectives

The main goal of this project is to develop a software that will assist with parking space management in parking lots/garages. The software will allow customers to view how many spaces are currently open and reserve a space ahead of time. Parking lot staff will utilize the software to confirm or cancel customer reservations, as well as adjust the number of open versus occupied spaces as cars leave or enter the lot.

The end goal is a java-based software that will run on most common operating systems.

1.2 Scope

There are 3 major functions of the software:

- Parking Lot Space Management:
    - Individual parking lots will be able to utilize the software in order to keep a running count of how many spaces in the parking lot are vacant versus occupied. This information can be accessed by all users of the software for any parking lot in the system.
- Parking Lot Space Reservation:
    - Individual parking lots will utilize the software to allow customers to make reservations for parking spaces in the lot without actually physically being present at the location of the lot. If there are vacant spaces for a lot, customers can choose to make a reservation. If customers already have a reservation, they can choose to cancel that reservation.
- Parking Lot and Account Database Management:
    - Admin-level users will be able to utilize the software to add, delete, or modify parking lots in the system. Attributes such as total number of spaces or parking lot name can be changed. Admin-level users will also be able to modify or delete user accounts. The software will give admins the ability to promote or demote accounts between account types.

2.0 Functional Description

2.1 User Profiles

The users are going to be primarily Customers who will be purchasing, canceling, and reserving parking lot spaces. Staff who will be responsible for canceling and changing reservations as well as managing parking lot space. There will also be Admins responsible for managing parking lot

databases, managing Staff and Customer information access, as well as managing staff accounts. Super admin users can also create admins.

Customers will be able to automatically register accounts, login to their accounts, and check the parking lot database for parking space availability.

Staff and admins will be able to directly manage parking lot databases, with Staff being able to manually create or delete reservations in a parking lot. Staff will also be able to manage parking space rates.

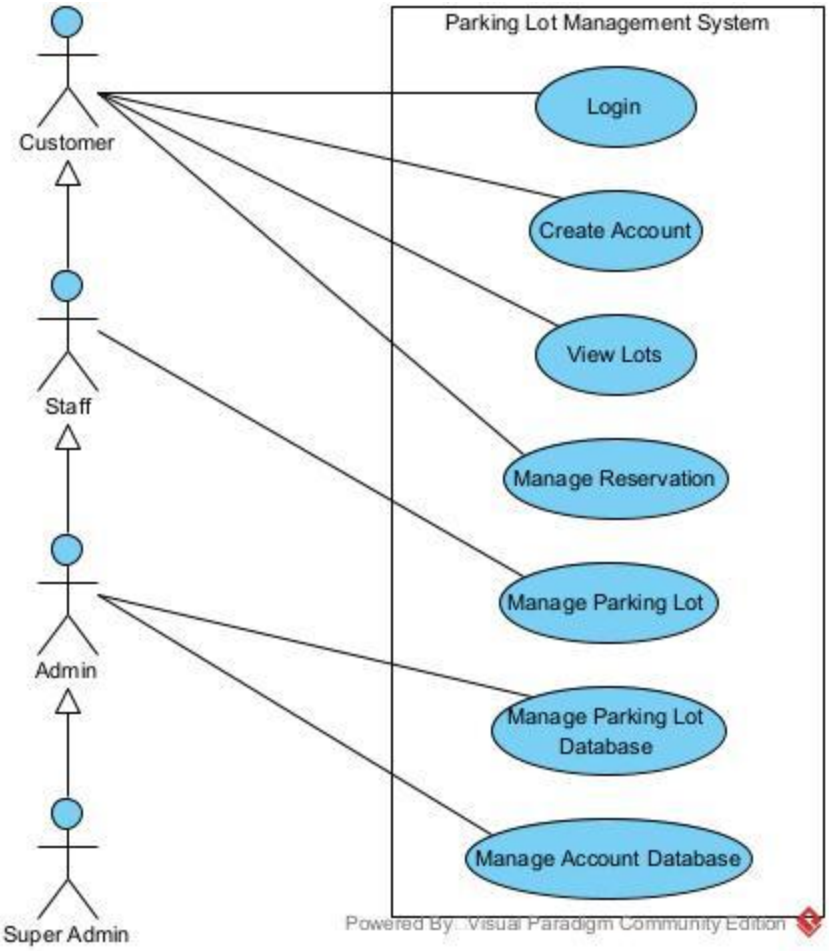
Admin will be able to manage accounts, create parking lot databases on the behalf of Staff and other clients, as well as having the capabilities of both Customers and Staff

Super admin users have all the capabilities of admins as well as creates admins.

2.2 Use Cases

Customers buy and reserve parking lots
Staff manages customers and overviews parking lot databases
Administrators create parking lot databases

Parking Lot App Use Cases			
Customer	Staff	Admin	Super Admin
• Login System • Manage Reservations (Self) • Browse Parking Lots	• Login System • Manage Reservations (Specific Parking Lot) • Manage specific Parking Lot rates	• Login System • Manage Reservations (All Parking Lots) • Manage all Parking Lot rates • Create Parking Lot Database entries • Manage Staff/Custome-r accounts	• Login System • Manage Reservations (All Parking Lots) • Manage all Parking Lot rates • Create Parking Lot Database entries • Manage all user accounts



2.3 Use Case Scenarios

2.3.1 Scenario 1 - Login

This scenario is the step all use cases branch from. A user
Will be prompted upon opening the app to provide a username and
Password on the login page. If the username and password match an existing account they are allowed to proceed with the software and they will have access to a variety of functions as determined by their accounts user level variable (i.e. Customer, Staff, etc.).

2.3.2 Scenario 2 - Create Account

This scenario is an alternative to Login where the user does not have an account. The user creates an account with a Username and Password. The user can also indicate whether they are Staff of an associated company, which they provide the information for which is then later approved by an Admin/Super Admin. A Super Admin has to manually provide Admin access to users.

2.3.3 Scenario 3 - View Parking Lots

This scenario involves a user (of any access level) to search and view a Parking Lot within the database. The Parking Lots will show whether they are empty, full, or have available parking spots. The user can then access the Parking Lot to make a reservation. Staff/Admin/Super Admin also have quick access to their company's associated Parking Lot(s).

2.3.4 Scenario 4 - Manage Parking Lots/Reservations

After accessing a parking lot from scenario 3 a user can manage their parking lot or reservations. Customers will be prompted to reserve the first available parking space if they wish or choose another space, as well as shown the parking rates. Customers can also cancel their reservations. Staff can change Customer reservations as well as edit parking rates. Admin/Super Admins can change Parking Lot information such as location and name.

2.3.5 Scenario 5 - Manage Parking Lot Database

Only users of level Admin or higher will have access to the lot database management menu. This menu will give access to create, delete, or modify parking lots in the system. For existing lots, the parking lot name, location, and total number of spaces can be changed.

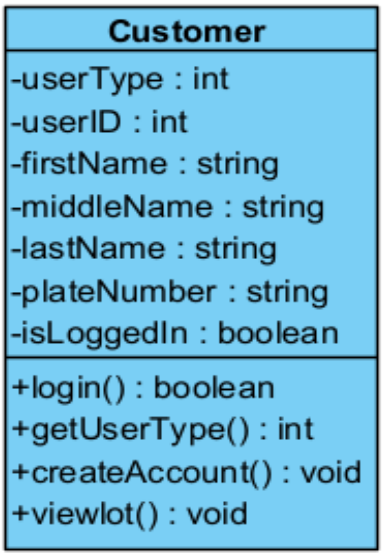
2.3.6 Scenario 6 - Manage Account Database

Only users of level Admin or higher will have access to the account management menu. This menu allows the ability to add, change, or delete any information of accounts of level Staff or lower. It will also allow the ability to promote or demote accounts between different account levels. Accounts cannot be promoted to level Super Admin from within the software.

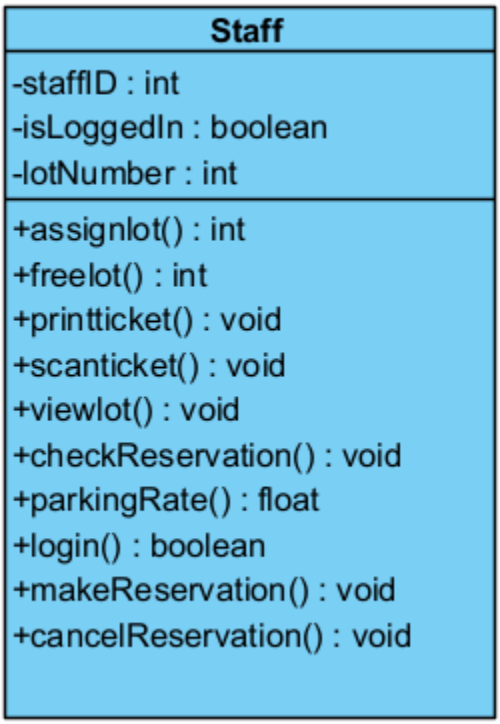
3.0 Data Description

3.1 Data Objects

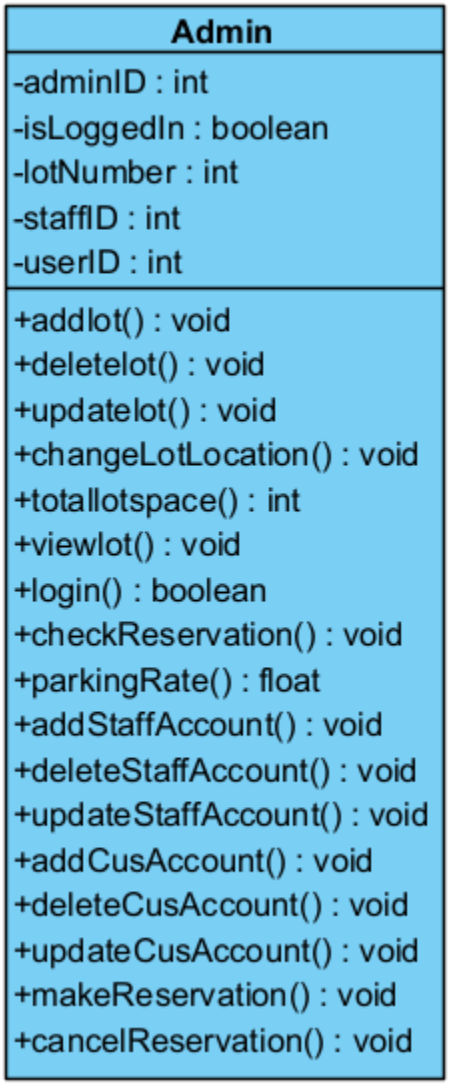
- Customer Class



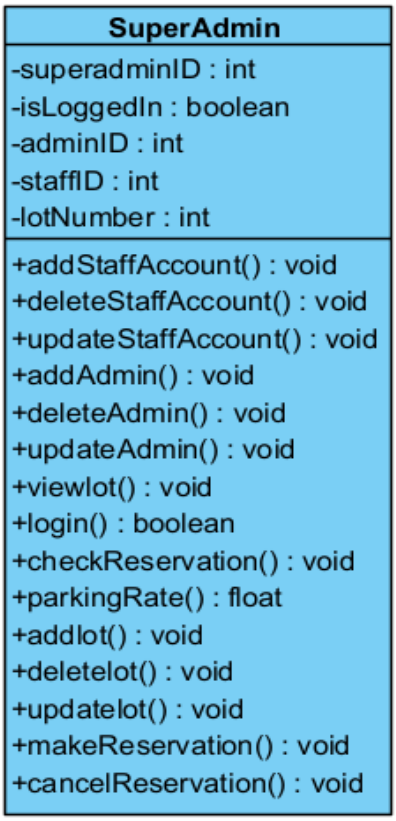
- Staff Class


Staff
-staffID : int
-isLoggedIn : boolean
-lotNumber : int
+assignlot() : int
+freelot() : int
+printticket() : void
+scanticket() : void
+viewlot() : void
+checkReservation() : void
+parkingRate() : float
+login() : boolean
+makeReservation() : void
+cancelReservation() : void

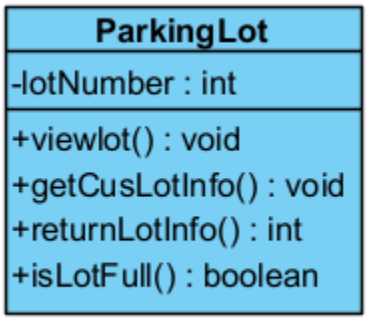
- Admin


Admin
-adminID : int
-isLoggedIn : boolean
-lotNumber : int
-staffID : int
-userID : int
+addlot() : void
+deletelot() : void
+updatelot() : void
+changeLotLocation() : void
+totallotspace() : int
+viewlot() : void
+login() : boolean
+checkReservation() : void
+parkingRate() : float
+addStaffAccount() : void
+deleteStaffAccount() : void
+updateStaffAccount() : void
+addCusAccount() : void
+deleteCusAccount() : void
+updateCusAccount() : void
+makeReservation() : void
+cancelReservation() : void

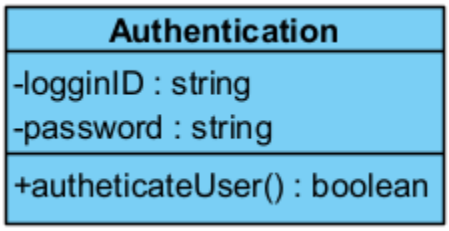
- SuperAdmin



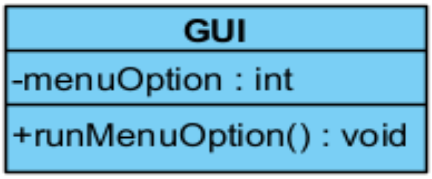
- ParkingLot



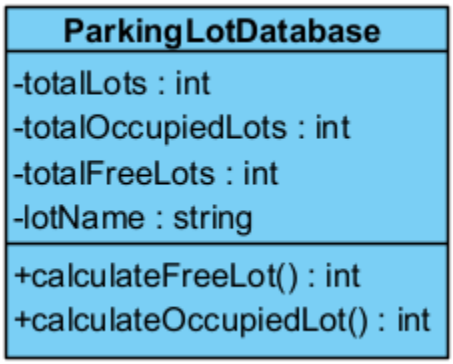
- Authentication



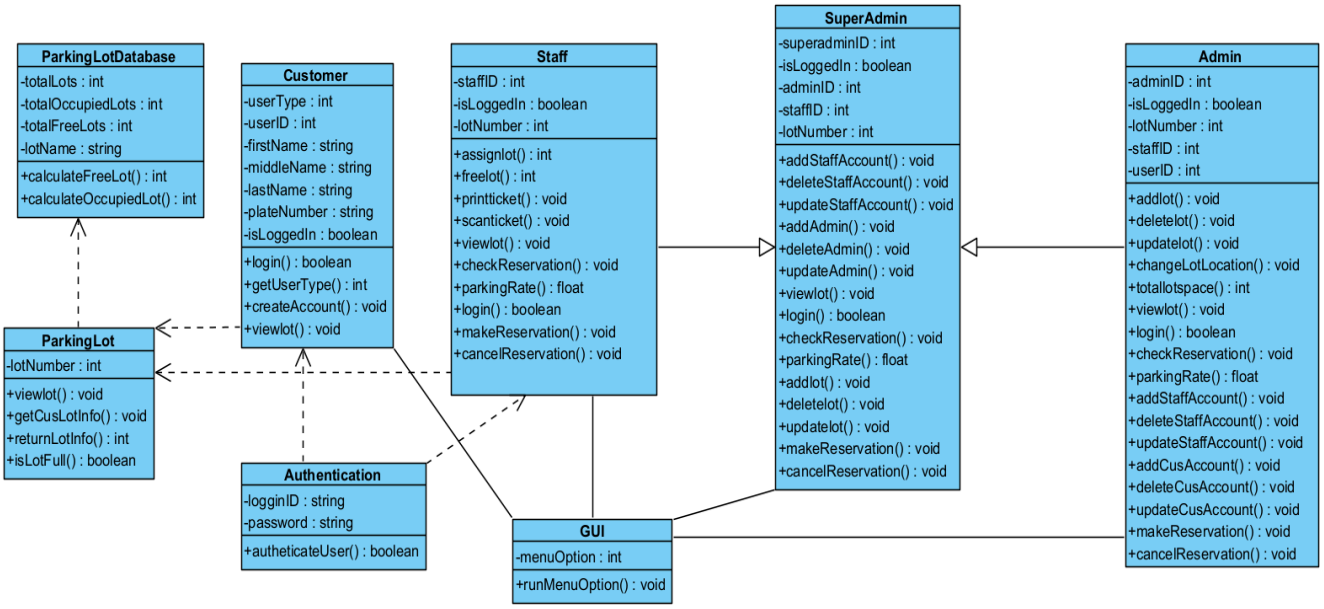
- GUI



- ParkingLotDatabase



3.2 Data Relationships



4.0 Software Interface Description

4.1 External Machine Interfaces

This software may require direct interaction with a barcode printer/scanner. An interface may need to be developed in order for the system to be integrated with a barcode system. Additional interfaces will also need to be developed for any sensors the software may work with, such as weight or motion sensors.

4.2 External System Interfaces

It is expected that other systems may regularly interact with this software in order to retrieve information such as current parking space availability. This functionality is not currently available without directly using the software, but in the future an API may be developed which allows other systems effortless access to parking lot information.

4.3 Human Interface

The overall design of this software is meant to be very simple and easy to understand. Minimum graphics should be used to avoid distraction. User Interface should be intuitive, with elements such as buttons being immediately recognizable. For simplicity, users of all levels will share the

same default views. Staff will be shown additional options on their lot pages. Admins will have access to two additional menus. The GUI will be implemented using FlatLaf (Flat Look and Feel) Java Swing.

5.0 Description of Software Behavior

5.1 Overall Behavior

The Software will be able to create smooth interactions between customer, staff, and the administrators. The application will have a flow that allows users to easily navigate through different functionalities based on their access level. After launching the application, users will encounter a login screen where they are able to get access to their accounts.

After logging in, customers will be able to view the parking lots availability, make reservations, and manage their parking lot bookings.Staff members will be able to see parking lots, handle current customer reservations, and update occupancy data in real time. Administrators will have extra capabilities that allow them to manage parking lots and user accounts.

5.2 Login Behavior

The flow when a user attempts to log in is as follows,

1. The user is prompted to enter their credentials
2. The system checks if the credentials are valid against the database
3. If correct, the user is granted access and redirected to their appropriate dashboard based on their access level.
4. If the credentials are incorrect, an error message is displayed, and we go back to step 1.
5. There will be a “Forgot Password?” link to initiate a password recovery process

5.3 Session Management

- After a successful login, the user's session will begin and will be maintained until they either log out or time out.

- Users may log out at any time which will stop their session and send them back to the login page.

- Sessions will expire after a set duration if inactivity to keep the application running efficiently and increase security.

5.4 Reservation Management Behavior

When a customer makes a new reservation,

1. They select a parking lot that they would like to park in, and view the number of spaces available.
2. The system will prompt them to choose a specific spot or move them to the next available spot.
3. After confirming the reservation, the system updates the database to decrement the count of available parking spaces and add the reservation to the user's account.
4. Customers will receive a confirmation notification either through app, email, or text, confirming their reservation.
5. Customers can view, modify, or cancel their reservations through their dashboard.

On the staff dashboard,

- Staff can view the current reservations and occupancy levels for their assigned parking lots.
- They will have the authority to either confirm or cancel customer reservations, which will update the system accordingly.

On the administrator dashboard,

- Admins will be able to oversee all of the reservations and have the ability to manage any customer's reservations as necessary.

5.5 Error Handling and Notifications

- The system will provide notifications for actions such as reserving or canceling reservations. There will also be detailed error messages for things such as a system time out, unable to load/reserver, and other errors that would keep a user from being able to reserve a parking spot.

- Users will be prompted to correct any errors, such as an incorrect password, invalid parking lot, or others to ensure a smooth and correct operation.

5.6 Data Consistency

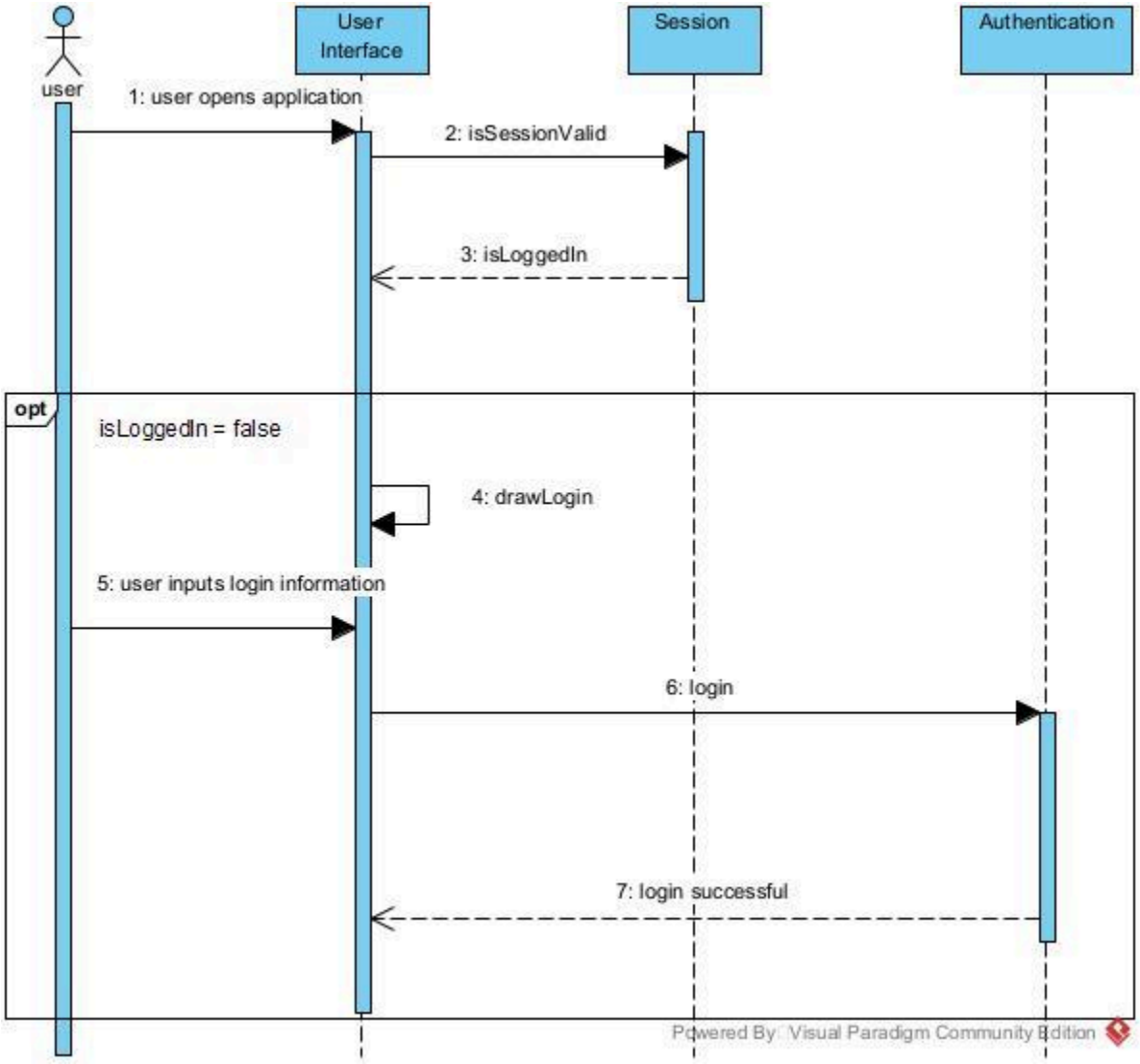
To ensure data integrity,

- All operations will involve database transactions to make sure that changes are either complete or rolled back to the previous state in the case of an error.

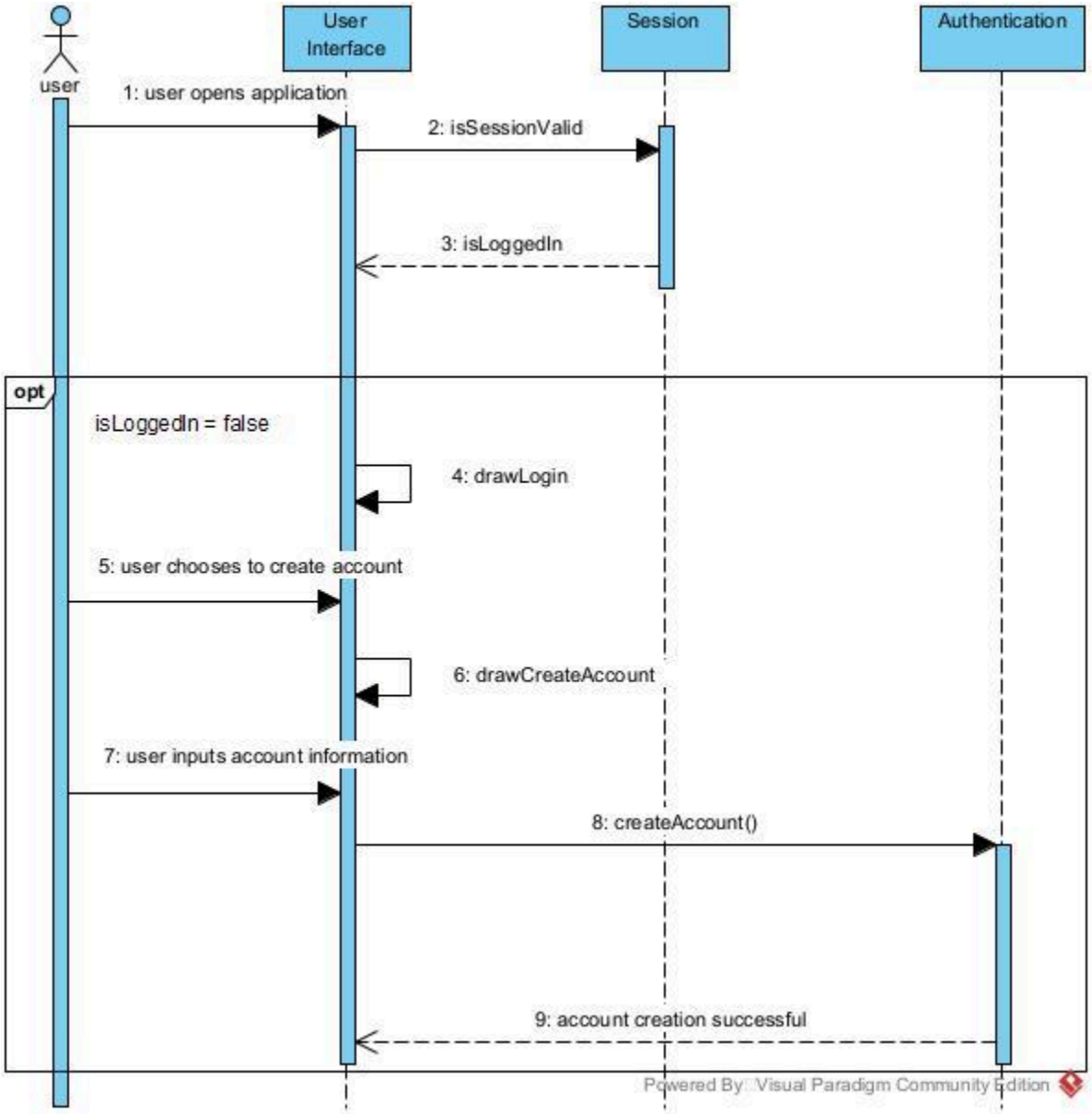
- The software will implement locking mechanisms to prevent conflicting changes when multiple users try to modify the same data simultaneously.

6.0 Use Case Realizations

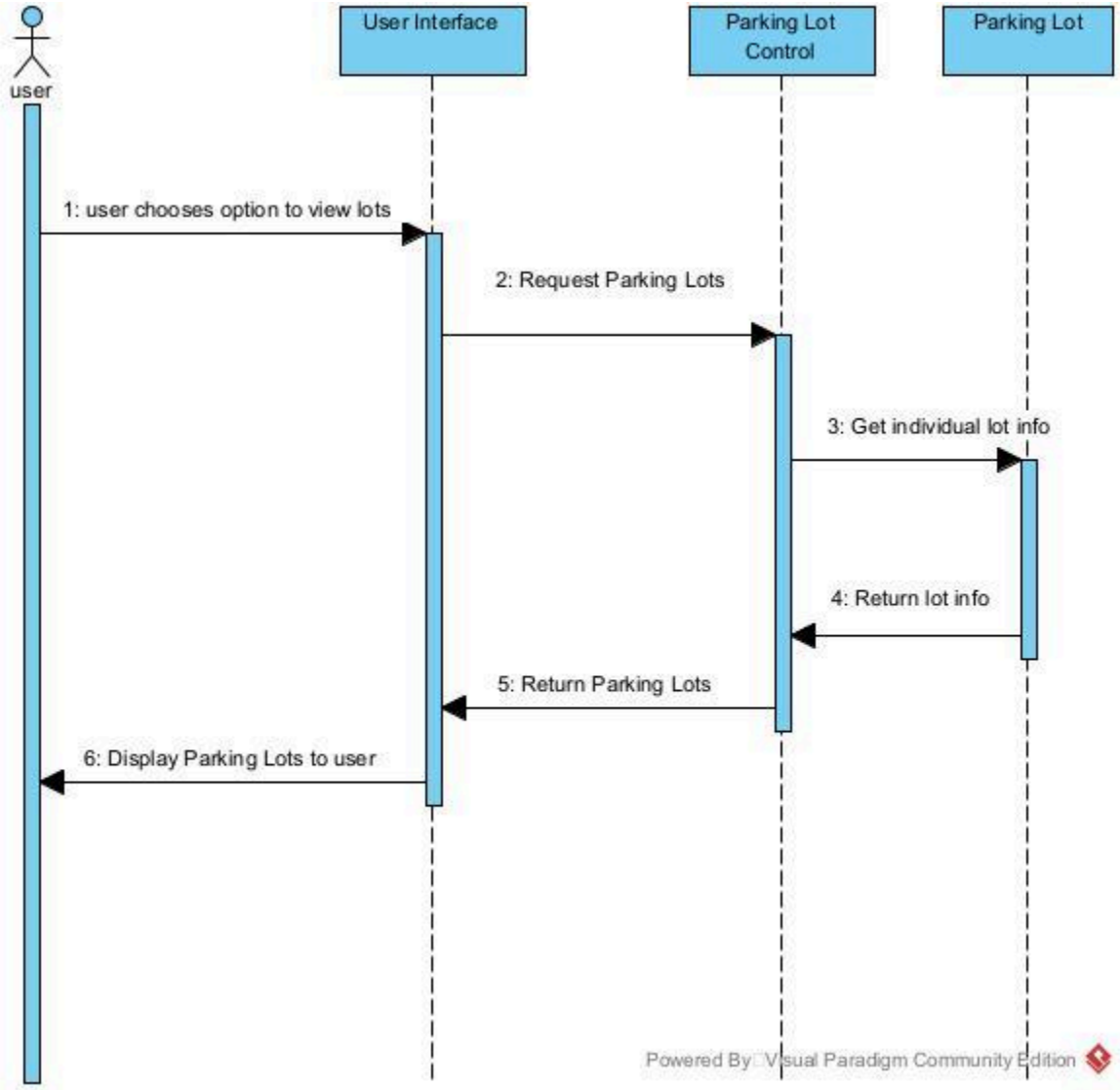
6.1 Realization 1 - Login



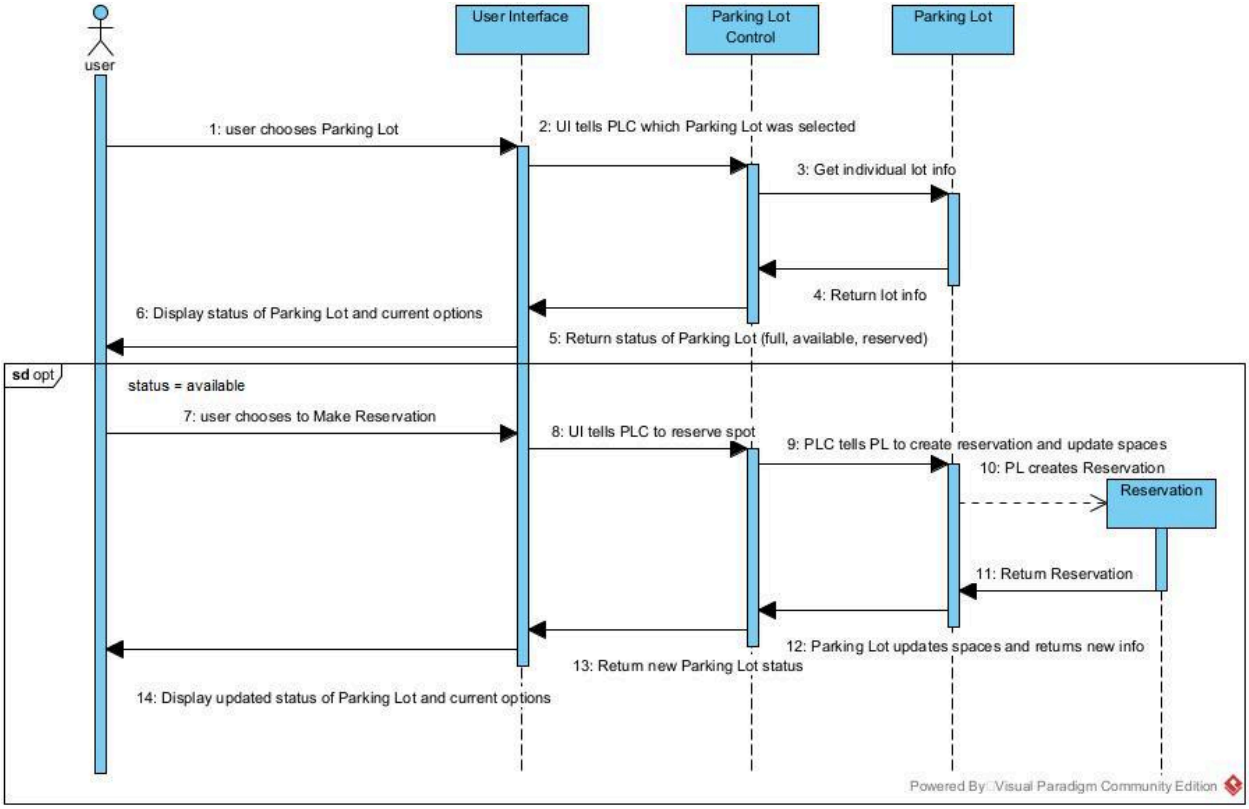
6.2 Realization 2 - Create Account



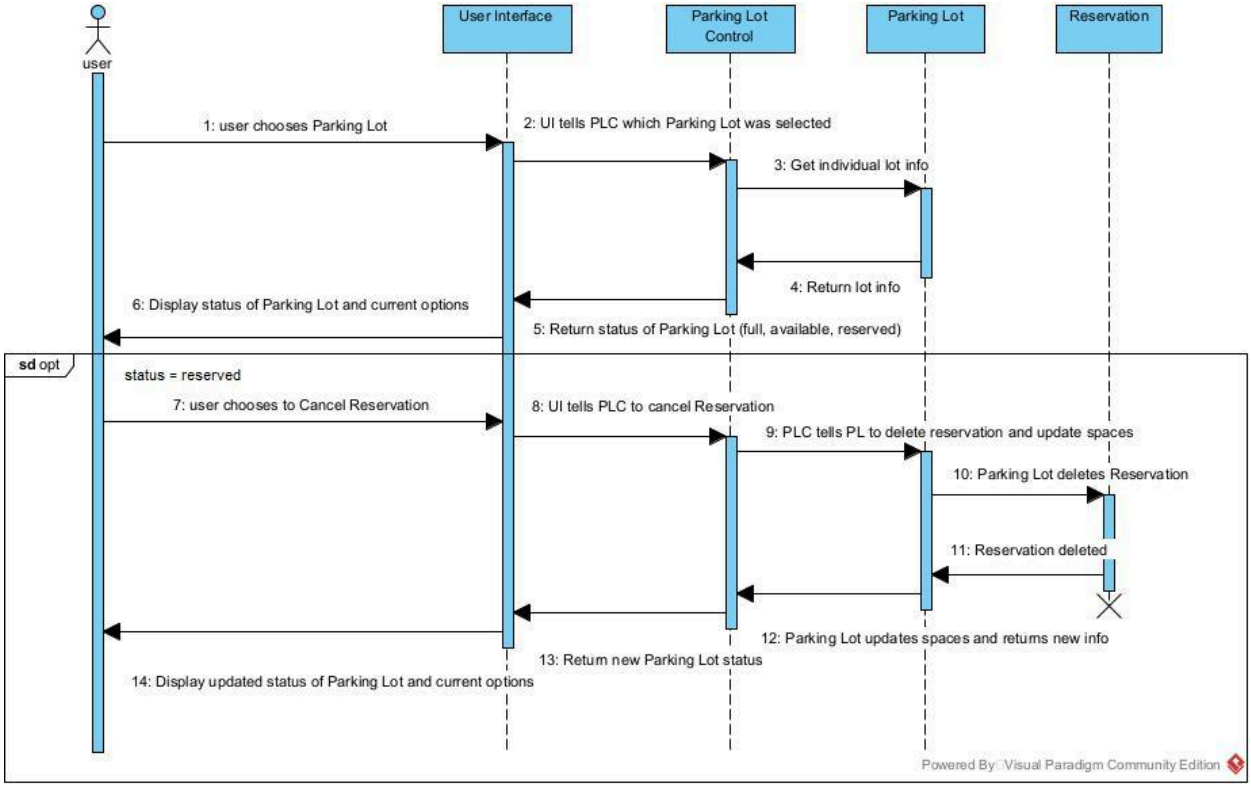
6.3 Realization 3 - View Parking Lots



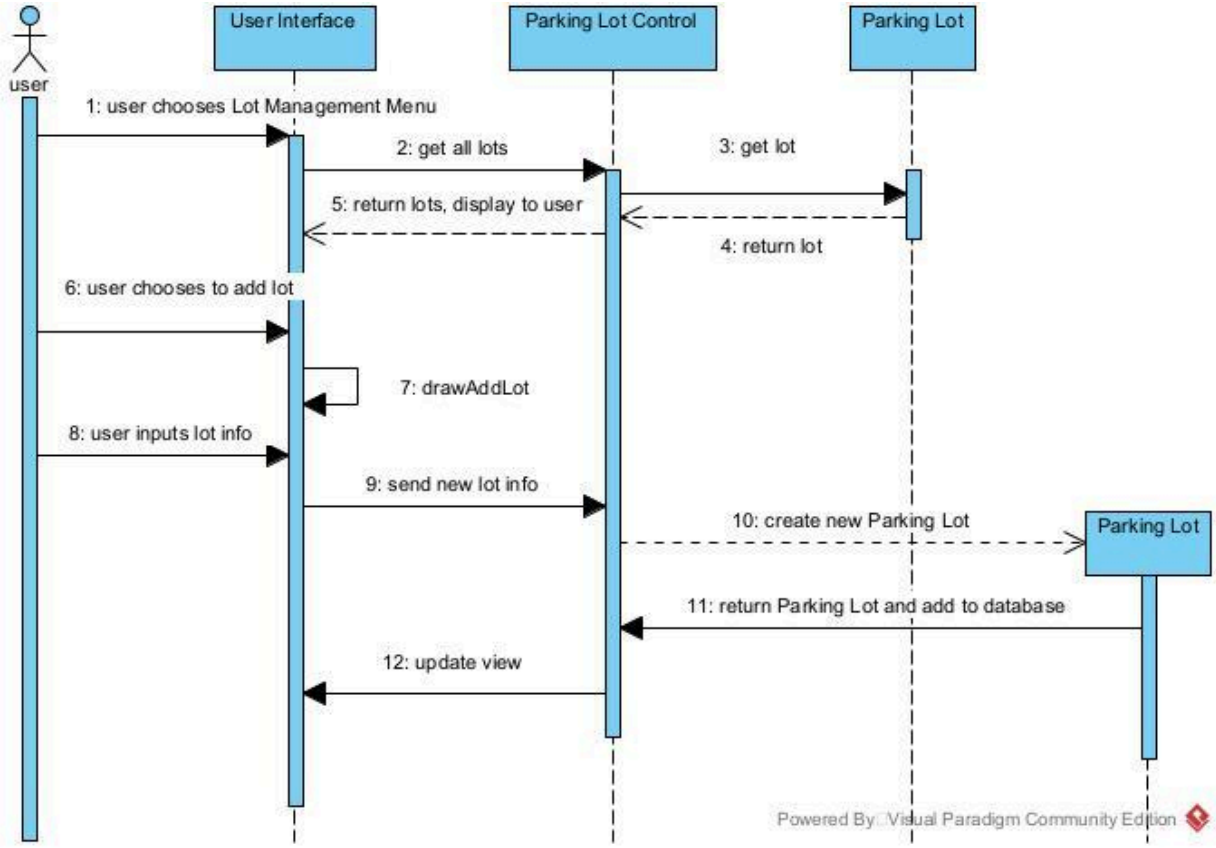
6.4 Realization 4.1 - Make Reservation



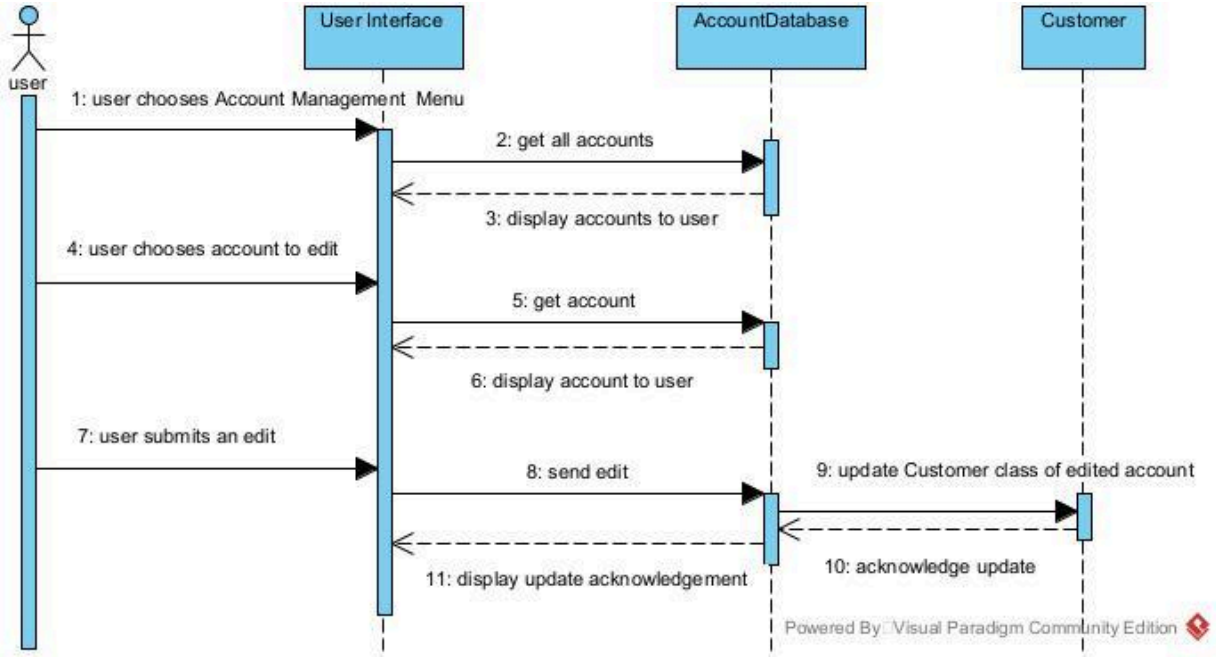
6.5 Realization 4.2 - Cancel Reservation



6.6 Realization 5 - Manage Parking Lot Database



6.7 Realization 6 - Manage Account Database



7.0 Restrictions, Limitations, and Other Issues

7.1 Assumptions

The software is designed for vehicles that can fit into standard sized parking spaces. The software assumes the user is parking a vehicle of regular dimensions.

7.2 Limitations

When a parking lot or user account is deleted from the database, there is no way to retrieve it. Back-up functionality may be added in the future.
Parking spaces are not tracked by position, making parking lot visualization and reservation of specific spaces not possible.

7.3 Restrictions

A personal computer is required to run this application. Mobile devices are not currently supported.

7.4 Other Issues

No other known issues at this time.