SRS-005

SOFTWARE REQUIREMENT SPECIFICATION

Department Store Parking Control System

Client:

Transmart Buah Batu

Prepared by:

Tukang Parkir

Faculty of Informatics

Telkom University

2023

Revision List

Revision	Description
A	
B	
C	
D	
E	
F	
G	

INDEX	-	A	B	C	D	E	F	G
Date								
Created by								
Reviewed by								
Approved by								

List of Revision Pages

Page	Revision	Page	Revision

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Purpose of Writing the Document

The objective of the department store parking control system application is to efficiently manage and optimize parking operations within our establishment using advanced AI. By minimizing congestion, reducing waiting times, and ensuring a seamless parking experience, we seek to improve overall customer convenience while maximizing space utilization. This system aims to enhance customer satisfaction by providing real-time parking availability.

1.2 Scope of Problem

By integrating secure payment gateways and loyalty programs, it will incentivize cashless transactions and reward loyal customers with parking discounts. Security measures will ensure vehicle and customer safety and eco-friendly initiatives such as electric vehicle charging stations will align with environmental sustainability goals, along with premium valet services which would give opportunities for unemployed people. The scope includes the implementation of cutting-edge technologies, seamless customer interactions, and data-driven decision-making using a wide range of AI technologies to ensure the smooth running of the system.

1.3 Definitions and Terms

This section provides a comprehensive list of key terms and their definitions used throughout the Department Store Parking Control System Software Requirements Specification (SRS) document.

1.3.1 Acronyms and Abbreviations

- SRS: Software Requirements Specifications
- AI: Artificial Intelligence
- SMS: Short Message Service
- e.g: For Example

1.3.2 Glossary

-

1.3.3 Comments and Notes

The provided acronyms and terms are crucial for understanding the various components and functionalities described in this SRS document.

This section aims to ensure clarity and consistency in communication by defining the key terms and acronyms used throughout the document. Refer to this section whenever clarification on specific terms is required.

1.4 References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

2. General Description of the Software

2.1 Statement of Software Objectives

The department store parking control system application software is a new product that is being developed to address the growing need for efficient parking management in department stores. The software will provide a convenient and easy-to-use online parking reservation system for customers, reduce parking congestion and improve customer satisfaction, and increase revenue for the department store by maximizing parking utilization.

2.2 Software Benefits / Functions

The Department Store Parking Control System provides a user-friendly platform for customers to book parking slots in advance through a powerful AI network. This advanced technology ensures a safe and secure experience for both reserving and utilizing parking spaces across all participating department stores. In essence, this application eliminates parking struggles and anxieties, transforming the parking experience into a seamless and convenient journey.

2.3 User Characteristics

1. Customers: use the application to find and reserve parking spaces, pay for parking, and access real-time parking.
2. Security Personnel: utilize the system for monitoring the parking area, reviewing surveillance footage, and ensuring the safety and security of both vehicles and users.
3. Maintenance and Operations Staff: responsible for the maintenance and operation of the parking facility use the application to manage equipment, address technical issues, and keep the parking infrastructure in optimal condition.

2.4 Software / System Limitations

1. Hardware Limitations: The system's functionality may be limited by the hardware infrastructure available. For example, memory and storage capacity.
2. Specific Technologies, Tools, and Databases: The department store may have predefined technologies, tools, and databases that developers are required to use.

Related to the selection of programming languages, developments framework, or database systems.

3. Parallel Operation: The system may need to handle multiple parallel operations, such as managing entry and exit simultaneously or handling multiple transactions concurrently.
4. Security Considerations: Access control, data encryption, and protection against unauthorized access.
5. Maintenance Responsibilities: The system's architecture and design should facilitate ease of maintenance, and developers may need to provide sufficient documentation for user-led maintenance tasks.
6. Design Conventions and Programming Standards: The applications may have specific design conventions or programming standards that developers must adhere to. This includes coding styles, documentation requirements, and testing procedures, which may impact the development process.

3. Detailed Description of Software

3.1 Requirement Descriptions

3.1.1 *Functional Requirements*

No.	Requirement ID	Feature	Description	Use case
1.	FR-01	User login	The system shall allow users to login into their respective accounts by providing a valid username and password	Login
2	FR-02	User authentication	The system shall authenticate user based on their username and password and grant access to their accounts if they are valid.	Login
3	FR-03	Unauthorized access prevention	This function shall prevent unauthorized access to user accounts by denying access to users who provide invalid credentials.	Login
4	FR-04	User logs out	This function shall allow users to log out of their accounts at any time	Login

5	FR-05	Account sign up	This function shall allow users to create new accounts by providing the required information such as their name/username, email, and password (SIGN UP USE CASE)	Sign up
6	FR-06	User information validation	The system shall validate the user's information and ensure that it is complete and accurate.(i.e email validation by sending codes to email)	Sign up
7	FR-07	Creating user account database	The system shall create a new user account in the database if the user's information is valid.	Sign up
8	FR-08	Update profile information	The system shall allow users to view and update their profile information, such as their name, email address, and profile picture. (UPDATE PROFILE USE CASE)	Update profile
9	FR-09	Validating user's profile update	The system shall validate the user's changes and ensure that they are complete and accurate.	Update profile
10	FR-10	Saving user's profile update to the database	The system shall save the user's updated profile information to the database.	Update profile
11	FR-11	Successful profile update feedback	The system shall provide users with feedback on their profile updates, such as whether their changes were successful or not.	Update profile

12	FR-12	Reservation history	System shall allow users to view their parking lot reservation history (Reserve number, date, time, status (pending, confirmed, canceled),etc) (RESERVATION HISTORY USE CASE)	Reservation history
13	FR-13	Reservation history search by keyword	System shall allow users to search their reservation history by keyword such as reserve number, date, time, and status	Reservation history
14	FR-14	Reservation history search by date range	System shall allow users to filter out reservation by date range	Reservation history
15	FR-15	Reservation Cancel	The system shall allow users to cancel their reservation.	Reserve Parking Slot
16	FR-16	Parking Slot Display	The system shall display the slot availability in real-time	Reserve Parking Slot
17	FR-17	Parking slot(s) Reservation	The user shall be able to view available parking slots in the particular parking location.	Reserve Parking Slot
18	FR-18	Reservation Total	The system shall display the total amount and the total number of user reservation(s).	Reserve Parking Slot
19	FR-19	Reservation Payment	The shall click the “payment” button in order to complete the reservation.	Reserve Parking Slot

20	FR-20	Customer dashboard interface	The system shall provide a customer dashboard's interface to update profile, looking for reservation history and also reserving parking slot	Customer dashboard
21	FR-21	Customer dashboard access	The customer dashboard shall be accessible to all registered users	Customer dashboard
22	FR-22	Staff dashboard interface	The system shall provide a staff dashboard's interface that is gonna be used by staff to accept offers, manage users, and also update parking availability	Staff dashboard
23	FR-23	Staff dashboard access	The system shall provide access for staff dashboard to verified staff	Staff dashboard
24	FR-24	Make offer	The application shall allow users to make offer to make an offer for the available parking slot (Make offer use case)	Make payment
25	FR-25	Offer management	The application shall allow users to view, edit, or cancel their pending offers before they are accepted	Make payment
26	FR-26	Offer status tracking	System shall be able to send customers real-time notifications about the status of their offers whether they have been accepted, rejected, or expired.	Make payment

27	FR-27	Offer payment integration	System shall be able to seamlessly integrate payment information when making offers to ensure secure and efficient payment processing	Make payment
28	FR-28	Offer management	The System shall be able to allow operation staff to accept or reject offer that have been made by the customer	Accept offer
29	FR-29	Payment checking	The System shall be able to send payment information from the customer to operations staff to secure payment process	Accept offer
30	FR-30	Software update interface	Application shall provide a dedicated interface for software updating that will be managed by maintenance staff(Software update use case)	Software update
31	FR-31	Payment arrangement integration	System shall be able to integrate payment from customers offer and sending it to operation staff when they are accepting the offer(Make payment use case)	Make payment
32	FR-32	Payment confirmation	System shall be able to give customers confirmation of successful payment processing, including a detailed transaction receipt	Make payment
33	FR-33	Payment method selection	System shall give customers the option to choose for their preferred payment method	Make payment

34	FR-34	Payment History	Customer shall be able to see payment history including viewing past transactions, reviewing payment methods	Make payment
35	FR-35	Update parking availability	System shall be able to display real time parking availability that is updated by the security personnel (Update parking availability use case)	Update parking availability
36	FR-36	Security and availability integration	System shall allow the security to update the parking lot availability in the application whether it is available or not	Update parking availability
37	FR-37	Checking parking lot	The system shall be able to allow security personnel to make a report about the parking lot whether it is available or not (Monitor parking status use case)	Monitor parking status
38	FR-38	Real time parking status	The system shallregularly retrieve and display real-time parking status information.	Monitor parking status
39	FR-39	Alerts and Notifications	The system shall trigger alerts and notifications when the vehicle leave first before or after the reservation time runs out.	Monitoring parking status
40	FR-40	Location parking lot	The system shall allow security personnel to filter the parking space view by location.	Monitoring parking status

41	FR-41	Patrolling parking lot	The system shall be able to receive report from security personnel's patrol (Patrolling the parking lot use case)	Patrolling/ report the parking lot
42	FR-42	Patrolling time stamp	System shall provide the security personnel a time stamp on when is the patrol occurs	Patrolling/ report the parking lot
43	FR-43	Security dashboard interface	The system shall provide security dashboard's interface report about parking lot patrol and also to monitor parking lot status	Security dashboard
44	FR-44	Security dashboard access	The system shall provide access to all verified security	Security dashboard
45	FR-45	User role management	The system shall be able to define between customer, staff, and security	User management
46	FR-46	User management functionality	The system shall be able to generate reports on user activity, such as user logins, account creations, and account edits.	User management
47	FR-47	User management access	The user management shall be only accessible to authorized staff members only	User management

48	FR-48	Login option	The system shall let staff manage whether users will go to customer, staff or security dashboard	Login

3.1.2 *Non-Functional Requirements*

No.	Quality	Requirement ID	Description	Use case
1.	Login performance	NFR-01	The login process shall be completed within 3 seconds (LOGIN USE CASE)	Login
2	Login availability	NFR-02	The login process shall be available at all times	Login
3	Login security	NFR-03	The login process shall be secure and protect user credentials from unauthorized access	Login
4	Login scalability	NFR-04	The login process shall be scalable to support a large number of concurrent users	Login
5	Login reliability	NFR-05	Login process shall be visually appealing	Login
6	Sign up performance	NFR-06	Sign up process shall be completed within 3 seconds (SIGN UP USE CASE)	Sign up
7	Sign up availability	NFR-07	Sign up process shall be available at all time	Sign up
8	Sign up security	NFR-08	Sign up process shall be secure and protect user data from unauthorized access	Sign up
9	Sign up reliability	NFR-09	Sign up process shall be compliant with all applicable laws and regulations	Sign up
10	Update profile performance	NFR-10	Update profile process shall be completed within 3 seconds.	Update profile
11	Update profile availability	NFR-11	Update profile process shall be available at all time	Update profile

12	Update profile security	NFR-12	Update profile process shall be secure and protect user data from unauthorized access	Update profile
13	Update profile reliability	NFR-13	Update profile process shall be compliant with all applicable laws and regulations	Update profile
14	Reserve history availability	NFR-14	Reserve history shall be available at all time.	Reserve history
15	Reserve history scalability	NFR-15	Reserve history shall be able to handle large number of concurrent users	Reserve history
16	Reserve history security	NFR-16	Reserve history shall be secure, this means that only the user shall be able to look for the history	Reserve history
17	Reliability	NFR-17	The system shall allow users to reserve when the department store is open.	Reserve Parking Slot
18	Performance	NFR-18	The response time for every operation shall be less than 5 seconds.	Reserve Parking Slot
19	Scalability	NFR-19	The system shall be able to support 100 concurrent users who want to reserve.	Reserve Parking Slot
20	Usability	NFR-20	The system shall provide clear instructions during the reservation process.	Reserve Parking Slot
21	Compatibility	NFR-21	The system shall be compatible with an iOS device as it is on an Android device.	Reserve Parking Slot
22	Make offer performance	NFR-22	The response time for every operation shall be less than 3 seconds	Make payment
23	Make offer availability	NFR-23	It shall be available at all times with a minimum uptime of 99.9%	Make payment
24	Make offer security	NFR-24	The application shall protect user data from unauthorized access, modification, or disclosure	Make payment
25	Make offer reliability	NFR-25	The application shall be reliable and should not crash or fail unexpectedly	Make payment

26	Make offer maintainability	NFR-26	The application shall be well designed and easy to maintain	Make payment
27	Make payment arrangement scalability	NFR-27	This shall be able to handle large number of concurrent payment transactions without any performance degradation	Make payment
28	Make payment arrangement availability	NFR-28	The payment shall be up and available at all time	Make payment
29	Update parking availability performance	NFR-29	The system shall be able to update parking availability on a real time with less than 3 second refresh time	Update parking availability
30	Update parking availability availability	NFR-30	The parking update shall be always up and running at all time	Update parking availability
31	Update parking availability maintainability	NFR-31	The system shall be able to fix an error if there is, within 5 minutes	Update parking availability
32	Monitor parking status performance	NFR-32	The system shall continuously monitor and evaluate parking lot performance metrics	Monitor parking status
33	Monitor parking status availability	NFR-33	the system shall promptly address any errors in parking lot availability detection	Monitor parking status
34	Monitor parking status maintainability	NFR-34	The system shall be able to detect and correct errors in parking availability information	Monitor parking status
35	Monitor parking status security	NFR-35	The system shall be able to ensure the safety and security of vehicle and pedestrian in the parking lot	Monitor parking status
36	Patrolling the parking lot availability	NFR-36	The system shall enable remote monitoring of parking lot availability through real-time video surveillance	Patrolling the parking lot

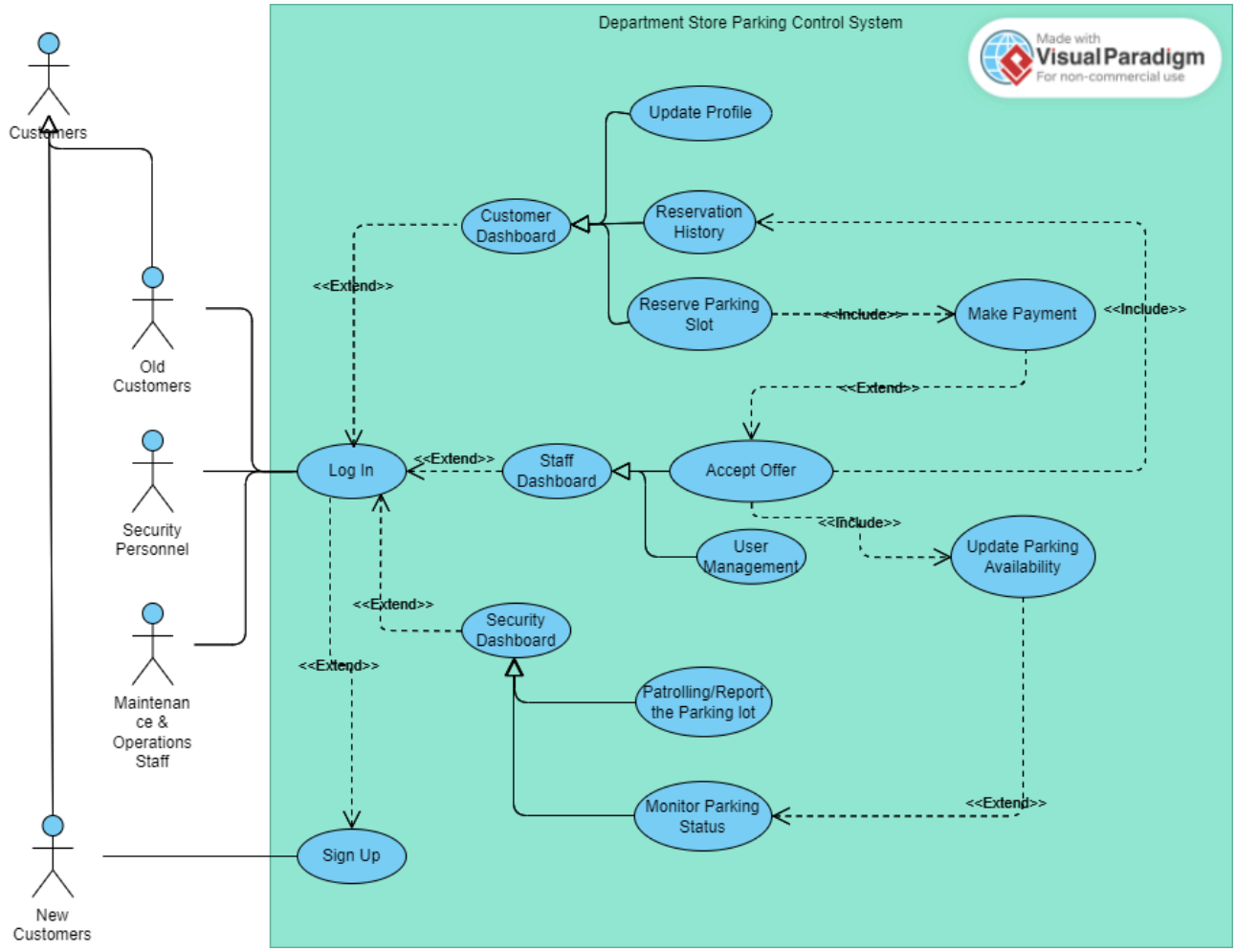
37	Patrolling the parking lot scalability	NFR-37	The system shall be scalable to accommodate an increasing number of parking areas and parking space	Patrolling the parking lot
38	Patrolling the parking lot maintainability	NFR-38	the system shall be able to monitor and maintain the integrity of parking lot infrastructure and equipment	Patrolling the parking lot
39	User management performance	NFR-39	The user management system shall be able to handle a large number of concurrent users without experiencing performance degradation.	User management
40	User management scalability	NFR-40	The user management system shall be scalable to accommodate an increasing number of users and data.	User management
41	User management usability	NFR-41	The user management system shall be easy to use for staff members.	User management
42	Customer dashboard reliability	NFR-42	System shall be able to provide an easy to use and navigate interface for customers	Customer dashboard
43	Customer dashboard security	NFR-43	System shall ensure that the customer dashboard is safe from unauthorized access	Customer dashboard
44	Staff dashboard reliability	NFR-44	System shall provide an easy to use and navigate interface for staff	Staff dashboard
45	Staff dashboard security	NFR-45	System shall be able to define an access only for staff	Staff dashboard
46	Security personnel dashboard reliability	NFR-46	System shall provide an easy to use dashboard interface for security personnel	Security dashboard
47	Security personnel dashboard security	NFR-47	system shall only provide access for security personnel and also authorized staff manager	Security dashboard

48	Customer dashboard perfomance	NFR-48	The customer dashboard shall load within 3 second under normal operating conditions and also should handle concurrent request from at least 300 users without significant performance degradation.	Customer dashboard
49	Customer dashboard scalability	NFR-49	The infrastructure shall be capable of handling a peak load during store's events, ensuring uninterrupted service.	Customer dashboard
50	Customer dashboard availability	NFR-50	The customer dashboard shall have an uptime of 99,9% and if there's any maintenance system, should be communicated to users at least 24 hours in advance.	Customer dashboard
51	Customer dashboard reliability	NFR-51	The system shall ensure data integrity (max loss 0,01%) regular backups of customer data should be performed and data restoration should be tested periodically.	Customer dashboard
50	Customer dashboard security	NFR-50	User authentication shall follow standard security practices (e.g., strong password), personal and financial information must be encrypted during transmission and storage.	Customer dashboard
51	Accept offer perfomance	NFR-51	The system shall process and reflect the acceptance of an offer within 90 seconds.	Accept offer
52	Accept offer scalability	NFR-52	The system shall be capable of handling a growing number of offer acceptances without significant performance degradation.	Accept offer
53	Accept offer security	NFR-53	Only authorized staff members shall have the ability to accept offers and update parking availability, and any sensitive	Accept offer

			data related to the offer acceptance process should be encrypted to ensure data security.	
54	Accept offer reliability	NFR-54	The system shall be available to acceptance and parking availability updates 24/7 with maximum allowable downtime of 12 hours per month.	Accept offer

3.2 Analysis Modeling

3.2.1 *Usecase Diagram*



3.2.1.1 Usecase Scenario #1 Log In

i. Pre-Condition: The user has an account on the application.

ii. Use Case Description

a. Primary Flow:

- The user accesses the system's login page.
- The system prompts the user to enter their credentials, including a username and password.
- The user enters the correct username and password.
- The system validates the entered credentials against the stored user information.
- If the credentials are valid, the system grants access to the user and proceeds to the main dashboard according to their roles.
- The main dashboard displays relevant information and options based on the user's role and permissions.

b. Alternative Flows:

1) Invalid Credentials:
   - If the entered credentials are invalid, the system displays the following error message: “The username or your password is incorrect.”
   - The user is prompted to re-enter the correct username and password.
2) Forgot Password:
   - If the user forgets their password, they can click on the "Forgot Password" link.
   - The system provides a password recovery option, such as sending a reset link to the user's registered email or SMS/WhatsApp.
   - The user follows the provided instructions to reset their password.
3) Account Lock-Out
   - After a specified number of consecutive failed login attempts, the system temporarily locks the user's account for 3 minutes
   - The user may try to login again once 3 minutes have passed.

iii. Post-Condition:

- Upon successful login, the system displays the order display or the relevant landing page based on the user's role.
- The user has access to the features and functionalities associated with their role.
- The system maintains the user's session until signout.

3.2.1.2 Usecase Scenario #2 Sign Up

i. Pre-Condition:

- The system is installed and operational.
- The user has access to the sign-up page.

ii. Use Case Description

a. Primary Flow:

- The user accesses the system's sign-up page.
- The system prompts the user to provide necessary information, including a unique username, email address, and password.
- The user enters the required information.
- The system validates the entered data, ensuring that the username is unique, and the email address is properly formatted.
- If the entered information is valid, the system creates a new user account and stores the provided details.
- The system sends a confirmation email to the user's provided email address with a verification link.
- The user clicks on the verification link to confirm their email address and activate their account.
- After successful verification, the system displays a confirmation message, and the user can now log in using their credentials.

b. Alternative Flow:

1) Username or email already in use:

- If the chosen username or email address is already associated with an existing account, the system prompts the user to choose a different username or use a different email address.

iii. Post-Condition:

- Upon successful sign-up and email verification, the user can log in using their newly created credentials.
- The system will redirect the user to the login page after successful verification.

3.2.1.3 Usecase Scenario #3 Reserve Parking Slot

i. Pre-Condition:

- The user is logged into the system.
- The user has a valid and confirmed payment method associated with their account.
- The system is operational and has up-to-date information on available parking slots.

ii. Use Case Description

a. Primary Flow:

- The user navigates to the "Reserve Parking" section within the system.
- The system displays a list of available parking slots, organized by department stores.
- The user selects their desired department store from the list.

- The system presents a visual representation of available parking slots for the selected department store.
- The user chooses a specific parking slot from the available options.
- The system calculates the parking fee based on the selected slot, displaying the cost to the user.
- The user confirms the reservation and initiates the payment process.
- The system redirects the user to a secure payment gateway where they provide payment details.
- The payment gateway processes the transaction, and upon successful payment confirmation, notifies the system.
- The system updates the reservation status to "confirmed" and assigns the chosen parking slot to the user.

b. Alternative Flow:

1) Payment Failure:
   - If the payment transaction fails, the system informs the user about the issue.
   - The user is prompted to review and correct payment details or choose an alternative payment method.
2) Parking Slot Unavailability:
   - If the selected parking slot becomes unavailable during the payment process (e.g., due to simultaneous reservations), the system informs the user.
   - The user can choose an alternative slot or proceed with a different reservation.
3) Reservation Cancellation:
   - The user has the option to cancel the reservation before making the payment.
   - If the user chooses to cancel, the system releases the selected parking slot back to the available pool.

iii. Post-Condition:

- The user receives a confirmation message with details about the reserved parking slot and the corresponding parking fee.
- The system updates the parking slot status to "reserved" and associates it with the user's account.
- The user can view their parking reservation details in the system as a part of their reservation history, including the reserved slot, department store, and payment information.

3.2.1.4 Usecase Scenario #4 Monitor Parking Status

i. Pre-Condition:

- The user has completed their parking session and left the parking lot.

- The security personnel responsible for updating parking lot availability has access to the system.
- The system is operational and accessible to security personnel.

ii. Use Case Description

a. Primary Flow:

- The security personnel log into the system using their authorized credentials.
- The system displays a dashboard with options, including "Monitor Parking Status"
- The security personnel selects the "Monitor Parking Status" option.
- The system presents a list of parking lots organized by department stores.
- The security personnel selects a specific department store to check the parking lot availability.
- The system provides a real-time visual representation of the current occupancy status for each parking slot in the selected department store.
- The security personnel physically inspects the parking lot, updating the system with the current availability status.
- If there are changes in the availability (e.g., a parking spot becomes vacant), the security personnel updates the system accordingly.
- The system reflects the updated parking lot availability information in real-time.

b. Alternative Flow:

1) System Unavailability
   - If the system is temporarily unavailable during the update process, the security personnel may note the changes manually and update the system once it becomes accessible.
2) Emergency Updates
   - In case of emergencies or sudden changes in parking lot availability, the security personnel has the option to perform an immediate update to reflect the current status.

iii. Post-Condition:

- The system displays the updated parking lot availability for the selected department store.
- Users accessing the system for parking reservations can view real-time information about available parking slots based on the recent update.

3.2.1.5 Usecase Scenario #5 Accept Offer

i. Pre-Condition:

- The system is operational, and the maintenance and operations staff or administration staff have access to the system.
- There are pending parking reservation offers that require verification.

ii. Use Case Description

a. Primary Flow:

- The maintenance and operations staff or administration staff log into the system using their authorized credentials.
- The system displays a dashboard with options, including "Accept Offer."
- The staff selects the "Accept Offer" option to review pending parking reservation offers.
- The system presents a list of pending offers with relevant details, including user information, reserved parking slot, payment status, and any associated issues or emergencies.
- The staff systematically reviews each offer to ensure accuracy and compliance with system policies.
- For each offer, the staff verifies the following:
    a. Payment Status: Verify that the payment associated with the offer is successful.
    b. Emergency Situations: Assess if there are any emergency situations that require special attention.
- Based on the verification, the staff can take one of the following actions:
    a. Accept the offer if everything is in order.
    b. Deny the offer if there are issues with payment or emergencies. Provide a reason for denial.
- The system updates the status of the parking reservation offer accordingly.

b. Alternative Flow:

1) Parking Lot Full

    If there is an emergency, the staff may prioritize the verification process for offers associated with emergency situations.

2) Payment Issues

    If there are payment-related problems, the staff may deny the offer and provide a specific reason, such as payment failure or discrepancies.

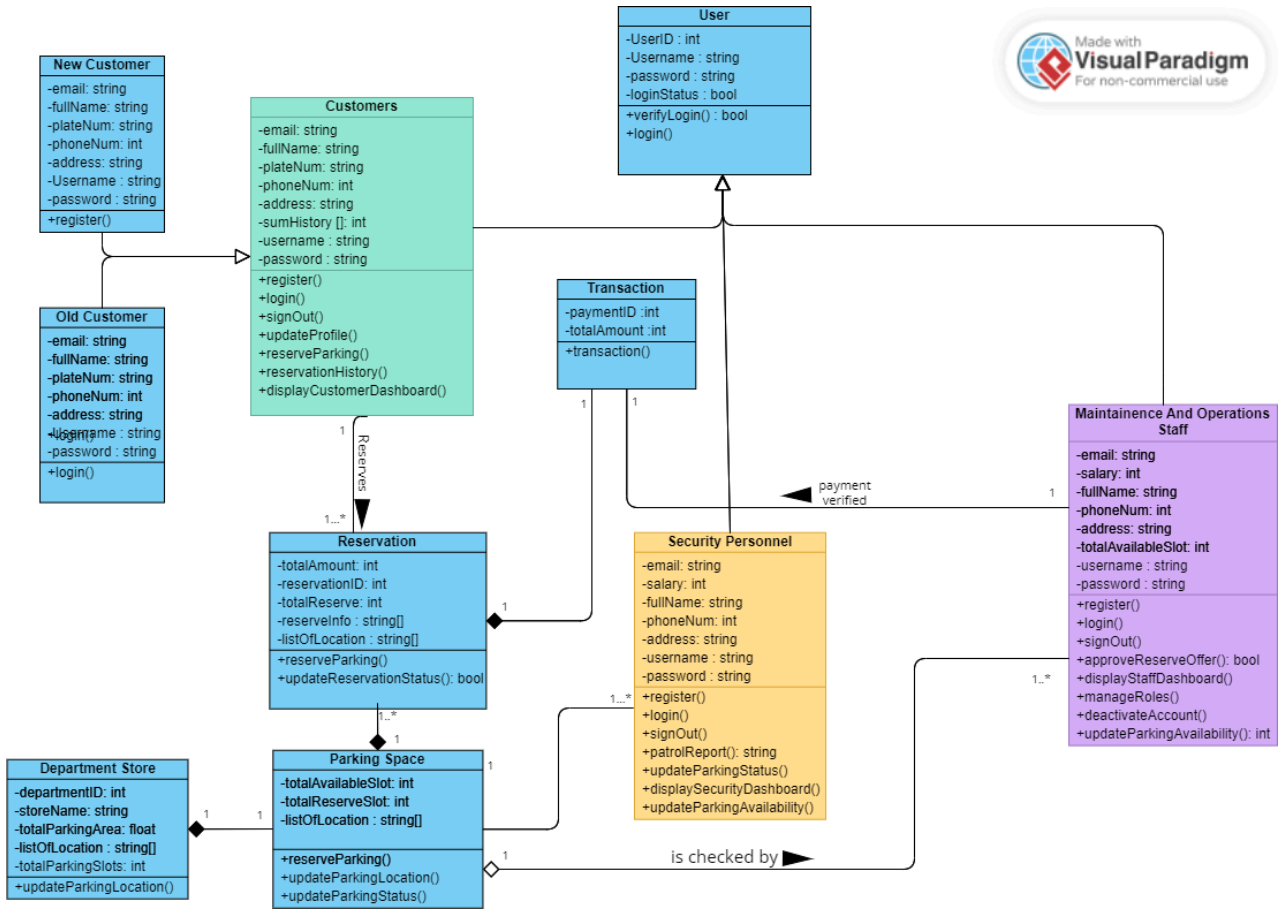
3) Emergency Situations

    If there is an emergency, the staff may prioritize the verification process for offers associated with emergency situations.

iii. Post-Condition:

- The system reflects the updated status of the parking reservation offer based on the staff's verification.
- Users receive notifications regarding the status of their parking reservation, including acceptance or denial in the application and their emails.

3.2.2 *Class Diagram*



3.2.2.1 Class Diagram Description

Class ID	Class Name	Attribute (visibility)	Method / Operation
001	*Customers*	*-email: string* *-fullName: string* *-plateNum: string* *-phoneNum: int* *-address: string* *-sumHistory []: int* *- username: string* *-password: string*	*+register()* *+login()* *+signOut()* *+updateProfile()* *+reserveParking()* *+reservationHistory* *+displayCustomerDashboard( )*
002	*Security Personnel*	*-email: string* *-salary: int* *-fullName: string* *-phoneNum: int*	*+register()* *+login()* *+signOut()* *+patrolReport(): string*

		-address: string *-username: string* *-password: string*	*+updateParkingStatus()* *+displaySecurityDashboard* *+updateParkingAvailability()*
003	*Maintenance and Operations Staff*	*-email: string* *-salary: int* *-fullName: string* *-phoneNum: int* *-address: string* *-totalAvailableSlot: int* *-username: string* *-password: string*	*+register()* *+login()* *+signOut()* *+approveReserveOffer(): bool* *+editAccount()* *+manageRoles()* *+updateParkingAvailability()* *+deactivateAccount()* *+displayStaffDashboard()*
004	*Reservation*	*-totalAmount: int* *-reservationID: int* *-totalReserve: int* *- reserveInfo: string[]* *-listOfLocation: string[]*	*+reserveParking()* *+updateReservationStatus(): bool*
005	*Parking Space*	*-totalAvailableSlot: int* *-totalReserveSlot: int* *-listOfLocation: string[]*	*+reserveParking()* *+updateParkingLocation()* *+updateParkingStatus()*

- *the operation "-" means private, access is allowed for itself only.*
- *the operation "+" means public, access is allowed for itself and all other classifier instances.*

- ***Customers:***
    - Attributes:
        1) -email: string
           Visibility: Private
           Description: Email address of the customer for communication.
        2) -fullName: string
           Visibility: Private
           Description: Full name of the Customer
        3) -PhoneNumber: integer
           Visibility: Private
           Description: Contact number of the customer.
        4) -address: String
           Visibility: Private
           Description: The address of the customer.
        5) -plateNumber: String
           Visibility: Private
           Description: The vehicle plate number associated with the customer's account.
        6) -sumHistory: List<Reservation>

Visibility: Private
Description: Contains the number of reserved slots the user reserved in each reservation session.

7) -Username: string
Visibility: Private
Description: The customer username that is used to log into their account
8) -Password: string
Visibility: Private
Description: The customer password that is used to log into their account

- Operation
  1) + register(fullName: String, email: String, phoneNumber: Integer, address: String, username: String, password: String): void
  Visibility: Public
  Description: Registers a new user with the provided information, including first name, last name, email, phone number, username, and password.
  2) +login(Username: String, Password: String): Boolean
  Visibility: Public
  Description: Logs in the user with the provided username and password. Returns true if the login is successful, false otherwise.
  3) + signOut(): void
  Visibility: Public
  Description: Logs the user out of their account.
  4) +updateProfile(fullName: String, address: String, phoneNumber: integer, email: String, plateNumber: String): void
  Visibility: Public
  Description: Allows the user to update their full name, address, phone number, and plate number.
  5) +reserveParking(): void
  Visibility: Public
  Description: Allows the user to reserve a parking slot.
  6) +reservationHistory(): List<Reservation>
  Visibility: Public
  Description: Retrieves the reservation history for the customer.
  7) +displayCustomerDashboard():(updateProfile, reservationHistory, reserveParkingSlot)
  Visibility: Public
  Description: allows user to update profile, view reservation history, and also to reserve a parking slot.

- **Security Personnel:**
  - Attributes:
    1) email: string

Visibility: Private
Description: Email address of the security personnel.

2) salary: integer
   Visibility: Private
   Description: Total salary of the security personnel.
3) fullName: string
   Visibility: Private
   Description: The full name of the security personnel.
4) phoneNum: integer
   Visibility: Private
   Description: Security personnel's phone number.
5) address: string
   Visibility: Private
   Description: Home address of the security personnel.
6) username: string
   Visibility: Private
   Description: Username assigned to the security personnel
7) password: string
   Visibility: Private
   Description: Password to secure user's account.

- Operations:
  1) +register(fullName: String, email: String, salary: integer, phoneNum: Integer, address: String, username: String, password: String): void
     Visibility: Public
     Description: Registers a user with the provided information, including full name, email, salary, phone number, username, and password.
  2) +login(Username: String, Password: String): Boolean
     Visibility: Public
     Description: Log in the user with the provided username and password. Returns true if the login is successful, false otherwise.
  3) +signOut(): void
     Visibility: Public
     Description: Logs the user out of their account.
  4) +patrolReport(): string
     Visibility: Public
     Description: The security personnel will give the report about the parking lot's condition.
  5) +updateParkingStatus()
     Visibility: Public
     Description:Updates parking status available or not available slots.
  6) +displaySecurityDashboard(): void
     Visibility: Public

Description: Display security dashboard, providing various option such as, patrolling/report the parking lot and monitoring parking status.

7) +updateParkingAvailability(): void
   Visibility: Public
   Description: The security personnel can update the parking availability manually with this procedure.

- **Maintenance and Operations Staff:**
  - Attributes:
    1) email: string
       Visibility: Private
       Description: Email address of the maintenance & operation staff.
    2) salary: int
       Visibility: Private
       Description: Total salary of the maintenance & operation staff.
    3) fullName: string
       Visibility: Private
       Description: The full name of the maintenance & operation staff.
    4) phoneNum: int
       Visibility: Private
       Description: Maintenance & operation staff's phone number.
    5) address: string
       Visibility: Private
       Description: Home address of the maintenance & operation staff.
    6) totalAvailableSlot: int
       Visibility: Private
       Description: The number of available parking slots so that the maintenance & operations staff can have and change the information about the recent total available parking slots in case there is an emergency.
    7) username: string
       Visibility: Private
       Description: Username assigned to the staff member
    8) password: string
       Visibility: Private
       Description: Password to secure the user's account
  - Operation:
    1) +register(fullName: String, email: String, salary: integer, phoneNum: Integer, address: String, username: String, password: String): void
       Visibility: Public
       Description: Registers a new user with the provided information, including full name, salary, email, phone number, username, and password.
    2) +login(Username: String, Password: String): boolean
       Visibility: Public

Description: Logs in the user with the provided username and password. Returns true if the login is successful, false otherwise.

3) +signOut(): void
   Visibility: Public
   Description: Logs the user out of their account.
4) +approveReserveOffer(): boolean
   Visibility: Public
   Description: Allows the administrator to approve or deny reservation offers. The Customer is denied from reservation if there's a case of emergency or if there are any discrepancies in payment or reserved slots. Returns true if the approval is successful, false otherwise.
5) +displayStaffDashboard(): void
   Visibility: Public
   Description:Displays the staff dashboard, providing various options for managing the system efficiently.
6) +manageRoles(): void
   Visibility: Public
   Description: Allows management of roles and permissions for different staff members and security personnel.
7) +deactivateAccount(): void
   Visibility: public
   Description: Deletes a user's account upon their request or under specific circumstances.
8) +updateParkingAvailability(): void
   Visibility: Public
   Description: Updates parking availability status and the number of available slots upon reservation or when a user leaves the parking space.

- **Reservation:**
  - Attributes:
    1) totalAmount: int
       Visibility: Private
       Description: Total cost from the reservation that the user made.
    2) reservationID: int
       Visibility: Private
       Description: Code that the user gets from the reservation.
    3) -totalReserve: int
       Visibility: Private
       Description: The total of the parking slots that the user wants to reserve.
    4) - reserveInfo: List<Reservation>
       Visibility: Private
       Description: Contains the chosen parking location of the slot(s) that the user wants to reserve.

- Operation:
    1) +reserveParking()
       Visibility: Public
       Description: to reserve a parking space in advance to ensure availability. The user can choose a specific location for the parking space.
    2) +updateReservationStatus(): bool
       Visibility: Public
       Description: to update the reserved parking slot once the payment/offer is accepted by the maintenance & operations staff. True means it is already reserved while false means it is still available.

- **Parking Space:**
    - Attributes:
        1) totalAvailableSlot: int
           Visibility: Private
           Description: the total number of slots in the parking space.
        2) TotalReserveSlot : int
           Visibility: Private
           Description: the total number of reserved parking slot in the parking space
        3) listOfLocation : string
           Visibility : Private
           Description: a list of available parking location description. the description provide clear information about the location of a specific parking slot.
    - Operation:
        1) reserveParking()
           Visibility : public
           Description : reserves a parking slot for vehicle. the operation should check if there are any available slot
        2) updateParkingLocation()
           Visibility : public
           Description : the operation should allow users to provide update information about the location of parking slot.
        3) updateParkingStatus()
           Visibility : Public
           Description : to update the status of a specific parking slot from available to occupied.

3.3 System Environment Requirements

3.3.1 Operating Environment

The software operates on mobile devices, especially mobile phones. These devices are smartphones running iOS or Android operating systems. iOS is compatible with iPhones and iPads, and it is designed to run on iOS 14, 15, or later. Android compatible with smartphones and tablets, run on Android 10, 11, or later.

The software components include the application itself, providing features like real-time parking availability, security patrols, and updates. The app will communicate with a central server/database system to retrieve and update parking information. The parking information needs mapping and location services to display the parking lot layout to help users navigate available spaces.

3.3.2 Hardware Requirements

a. Mobile Processor

The software that is compatible with iOS and Android, for iOS it is designed to run on iOS 14, 15, or later. Android runs on Android 10, 11, or later.

b. Mobile Memory

- Logical characteristics: The software may specify a minimum amount of RAM required for efficient operation.
- Physical characteristics: the mobile device should have enough RAM to support the application, considering other concurrent processes.

c. Mobile Storage(Internal/External)

- Logical characteristics: the software may specify a minimum amount of RAM required for efficient operation.
- Physical characteristics: the mobile device should have enough RAM to support the application, considering other concurrent processes.

d. Network Interface(Mobile Data/Wi-Fi)

- Logical characteristic: the software requires network connectivity, supporting mobile data and Wi-Fi.
- Physical Characteristic: the mobile device should have a working mobile data connection and Wi-Fi capabilities.

e. Camera

- Logical characteristics: the software may use the camera for various purposes, such as scanning QR codes for parking access.
- Physical Characteristics: the mobile device should have a camera with sufficient resolution and capabilities.

f. Touchscreen and Input Devices

- Logical Characteristics: the software supports touchscreen input and may utilize other sensors.
- Physical characteristics: the mobile device should have a responsive touchscreen, and the app may use GPS for location-based services.

g. Power Management

- Logical characteristics: the software is designed to optimize power usage, possibly including features like low-power mode, to enhance the device's battery life.
- Physical explanation: the mobile device should have a battery with sufficient capacity, and the app should implement power-efficient practices to minimize battery drain during regular use.

3.3.3 Software Dependencies

1. Software Components:
    a. Mobile Parking Lot Management App
    Purpose: Core applications for users to check parking availability.
    Data in: user inputs (parking requests)
    Data out: request for parking data
    b. Central Server/Database
    Purpose: stores and manages data (parking information, user data, and security)
    Data in update from the mobile app
    Data out: real-time information
    c. Mapping and Location Service
    Purpose: provides mapping and location parking
    Data in parking area
    Data out parking layout and navigation

2. Implementation Constraints:
    a. Security Measures
        - Secure communication protocols for data transmission.
        - user authentication and authorization for app access.
    b. Data Integrity
        - implement checksums or hash functions to ensure data integrity during transmission.
3. Data Sharing Mechanisms:
    a. Real-time Updates:
        - implement a publish-subscribe mechanism for real-time updates between the mobile app and the central server.
    b. Caching:
        - use caching mechanisms to optimize data retrieval and reduce server load.

4. Appendix

<Define other requirements not covered in this SRS. These may include database requirements, internationalization requirements, legal requirements, reuse goals for the project, and so on. Added new sections related to the project.>

Appendix A: List of Difficult Words

<Determine all requirements necessary to interpret this SRS correctly, including acronyms and abbreviations. You may want to create separate lists that cover multiple projects or the entire organization, and only include terms specific to one project in each SRS.>

Appendix B: Analysis Models

*<**Optional**. Enter related analysis models, such as state-transition diagrams, flow-maps, or entity-relationship diagrams (ERD).*
Note: Flow-map and ERD + BD Relationship Scheme (for SI applications) and Flow chart (for non-SI applications, for example games>