

# **Smart Parking**

## **Software Requirements Specification 0.2**

**April 10, 2016**

**Prepared for  
Sharda University: Hospital Parking**

# Table of Contents

<b>REVISION HISTORY .....</b>	<b>5</b>
<b>DOCUMENT APPROVAL .....</b>	<b>5</b>
<b>1. INTRODUCTION.....</b>	<b>6</b>
1.1 PURPOSE .....	6
1.2 SCOPE .....	6
1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS .....	6
1.4 REFERENCES .....	7
1.5 OVERVIEW .....	7
<b>2. GENERAL DESCRIPTION .....</b>	<b>7</b>
2.1 PRODUCT PERSPECTIVE .....	8
2.2 PRODUCT FUNCTIONS .....	8
2.3 USER CHARACTERISTICS .....	8
2.4 GENERAL CONSTRAINTS .....	9
2.5 ASSUMPTIONS AND DEPENDENCIES .....	9
<b>3. SPECIFIC REQUIREMENTS .....</b>	<b>9</b>
3.1 EXTERNAL INTERFACE REQUIREMENTS .....	9
3.1.1 <i>User Interfaces</i> .....	11
3.1.2 <i>Hardware Interfaces</i> .....	11
3.1.3 <i>Software Interfaces</i> .....	11
3.1.4 <i>Communications Interfaces</i> .....	12
3.2 FUNCTIONAL REQUIREMENTS .....	12
3.3 USE CASES .....	14
3.4 CLASSES / OBJECTS .....	21
3.5 NON-FUNCTIONAL REQUIREMENTS .....	22
3.6 INVERSE REQUIREMENTS .....	
3.7 DESIGN CONSTRAINTS .....	
3.8 LOGICAL DATABASE REQUIREMENTS .....	23
3.9 OTHER REQUIREMENTS .....	

<b>4. ANALYSIS MODELS .....</b>	<b>18</b>
4.1 SEQUENCE DIAGRAMS.....	18
4.3 DATA FLOW DIAGRAMS (DFD) .....	18
4.2 STATE-TRANSITION DIAGRAMS (STD) .....	18
<b>5. CHANGE MANAGEMENT PROCESS.....</b>	
<b>A. APPENDICES.....</b>	
A.1 APPENDIX 1 .....	
A.2 APPENDIX 2 .....	

## Revision History

Date	Description	Author	Comments
January 28, 2016	Version 0.1	Anjali Ayush Yashasvi	First draft
April 10,2016	Version 0.2	Anjali Ayush Yashasvi	Updated use cases

## Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
	Taranpreet Singh Grover	Project Architect	

## 1. Introduction

A Web Based application for an efficient parking experience with additional features of e- payment and navigation. To solve major parking problems in the university, a map of particular area with turn by turn directions is provided .To save customer's time and stress related to parking issues, so that they can focus on the actual reason for visit to the campus.

### 1.1 Purpose

Parking management influences drivers search time and cost for parking spaces, parking revenue, and traffic congestion. The parking authority needs to monitor the state of each parking space in real time and optimize the parking management. The proposed pre-booking systems of parking slots has the potential to simplify the operations of parking systems, as well as alleviate traffic congestion caused by searching for parking and on the spot payment.

### 1.2 Scope

1. The basic scope is that the application reaches to the very base consumer that is, the person who is not very technology friendly.
2. GPS, Maps, BLE, QR code third party application for safe transactions is used.
3. To provide complete satisfaction and to ease the trouble that is caused now a days because of heavy congestion .To provide a long term parking solution for user and operators.

### 1.3 Definitions, Acronyms, and Abbreviations

Administrator	The person governing all bookings and other related activities.
Regular user	Everyday customer who visits the campus almost daily.
One time user/New user	A guest to the campus who visits the campus only once or occasionally.
Parking Administrator	The person at the entry/exit gate who keeps a track of the cars parked and available slots.
GPS(Global Positioning System)	It is a space base navigation system that provides location and time where there is a un obstructed line of sight to four or more GPS satellites.

Maps	It is a desktop web mapping service ,it offers satellite imagery, street maps,360 panoramic views, real time traffic condition, and route travelling by foot, car, bicycle or public transportations.
Information Security Analyst	A person who monitors the authentication procedure and looks after the security of user and parking data.
Database Manager	The database manager looks after all the databases and various operations like add, delete, update, rollback, commit.
Unregistered User	A user without an account in the database.
Third Party Application	A third party transaction is a business deal involving a buyer, seller and a third party. The third parties involvement varies with the type of business transaction.
BLE	Bluetooth low energy (BLE) is a wireless technology standard for personal area networks. BLE is targeted for very low power devices. They are used for sensing the vehicles after they enter the campus.
QR code	A barcode is a machine-readable optical label that contains information about the item to which it is attached. It is used to scan the booking ID of the customer by the parking administrator.

### 1.4 References:

Deloitte standards of SRS

### 1.5 Overview

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

### 2. General Description

The overall perspective of the software is given in the following sections. The basic functionality of the application is specified.

### 2.1 Product Perspective

This application will be used on web portals (Web-Based). The main users are the campus staff and students who require parking solutions on a daily basis and also the visitors who visit the campus either once or occasionally. A monitoring view will be given to the parking guard for managing full and vacant slots. The other software used along with this application are GPS, Maps, QR code, BLE Integrations, Third party for transactions,

The web application will need to communicate to a GPS application within the mobile phone, which in turn communicates with a physical GPS device to find the location of the user. The GPS will provide the application with locations of the user and the booked parking slot and the distance between them, but it will also provide maps and the functionality to display the application's data on the map. The functionality provided by the GPS will be embedded into the application. The map of the parking area is provided which indicates vacant and taken slots. The QR code is generated for each booking which is afterwards scanned by the parking administrator. This is used to identify the customers and their respective slots and also to map the booking space with the authenticated customer to avoid any kind of inefficiency. After the QR is successfully scanned, the BLE guides the customer with turn by turn directions and taking over the GPS, now the sensor systems come into action. The BLEs also sense the free and parked areas and update the database respectively.

Since this is a data-centric product it will need somewhere to store the data. For that, a database will be used. The application will communicate with the database. The data in the database can be added and modified by authenticated access.

### 2.2 Product Functions

With the application, the user will be able to book a parking slot in Sharda University.

After the log in procedure, the user will select the arrival time and book a slot by viewing the available parking slots in the form of a map.

After booking the slot, he pays for the booking through a credit based system.

The user has an option to get a turn by turn directed map from his start point to the assigned parking slot through a GPS system.

After reaching the campus, the Parking Administrator scans the QR code and verifies the possession and BLE guides the user thereafter and updates the slot map.

While the user is leaving the slot, the BLE senses the exit and updates the particular slot status as free.

### 2.3 User Characteristics

The application has 3 types of users: The regular user, the one-time user and the Parking Administrator and Customer support manager.

The Regular and one-time users can register, log-in to the application, book a slot at a particular time, pay through credits, navigate to the parking slot.

Regular customer gets additional benefits of 2 free bookings per month.

A one-time user gets map of the campus after the parking slot is reached to guide him to the desired location in the campus.

The customer support manager handles all the customer feedback and complaints via calls or e-mail.

### 2.4 General Constraints

The application is constrained by the system interface to the GPS navigation system within the mobile phone. Also, there may be a difference between what navigation features each of them provide.

The Internet connection is also a constraint for the application. Since the application fetches data from the database over the Internet, it is crucial that there is an Internet connection for the application to function. The scanning of QR code must be efficient enough to avoid any kind of problem when the customer reached the spot. The BLEs must be active and reliable all the time for the database to be updated on time.

The application will be constrained by the capacity of the database. Since the database is shared between User and PG, it may be forced to queue incoming requests and therefor increase the time it takes to fetch data.

### 2.5 Assumptions and Dependencies

Every user has a smart phone through which he can access the web application. All users will use the application to book parking slots.

The app will be dependent on the facilities provided by GPS, google maps and internet facilities.

The smart phone is compatible enough to be sensed by the BLE. The user's phone must be able to use these facilities.

The user is familiar with the basic functionality like using maps.

## 3. Specific Requirements

This section contains all of the functional and quality requirements of the system. It gives a detailed description of the system and all its features.

### 3.1 External Interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.

#### 3.1.1 User Interfaces

##### 3.1.1.1 Welcome page:

Register: Used to register new users

Log in: Used for log in for existing customers Click on:

Register -> **Register page:**



## Smart Parking

Text boxes where the user enters Name, Mobile number, a maximum of 3 cars per account, captcha-mail id.

Save: The details are stored to the database by clicking on this button.

If no similar record is found in the database,

Click on save-> Log in page Click on: log in -> **Log in page**

**Else** display “Already Registered” and go to the log in page.

The user enter his mobile number and registered car number.

Enter button: Confirms log in details.

After successful log in, **Account page** is displayed

Available credits: Credits available

Recharge: Clicking on this directs to the recharge page

Booking: The active and past bookings are displayed, in case of active booking, the time slot is displayed, with an option to cancel the booking.

The user can also create a new booking on clicking the “New booking” option. This takes him to the slot selection page.

### 3.1.1.2 Slot selection page:

#### **If slots are free:**

A map of the parking area is shown, where the vacant spots are shown in a different color which can be selected.

The user selects one slot and clicks on confirm button.

Confirm button: Confirms the booked slot and the user is directed to the next page: Estimated arrival time.

The user enters the arrival time and clicks the “Confirm” button.

#### **Else:**

The message “Parking Full, try again later!” is displayed.

### 3.1.1.3 Payment page:

Available credits

**If it is  $\geq 20$** , then it (20 credits) is deducted from the available credits

#### **Else**

1. The message” No credits, please recharge” is displayed.
2. The “recharge page” is opened
3. The user enters the amount to be recharged in amount text box.
4. The user is directed to a third party application for payment through e-wallet/credit/debit/net banking.
5. The user enters the credit card details: Card number, name, CVV, Expiry date.

6. The entered details are verified with the database values.
7. The user is then redirected to the payment page after successful transaction and credits are deducted for the booking.
8. After successful payment, the booking ID is generated

### **3.1.1.4 Map:**

Turn by turn directions are provided through google maps until the user reaches the slot.

On reaching the gate, booking id is shown to parking guard who confirms the possession of the booked spot.

### **On the reaching the spot:**

The message “You have reached your destination” is displayed.

(Optional)View campus map button: This opens a campus map, where the user can select where he wants to go within the campus. A path with turn by turn directions are given.

### **3.1.1.5 Account display page**

1. The entire account information of the user is given
2. The user can also edit the information
3. The database is updated with the edited information.

### **3.1.1.6 Booking history page**

1. The user can view his current and previous booking in this page.
2. Booking will be shown in a queue format and the latest one will be shown at the top.
3. The user can cancel the current bookings by visiting this page.

## **3.1.2 Hardware Interfaces**

Since the application does not have any designated hardware, it does not have any direct hardware interfaces. The physical GPS is managed by the GPS application in the mobile phone and the hardware connection to the database server is managed by the underlying operating system on the mobile phone and the web server. The BLE is used to sense the vehicles in the campus which have a pre booking. The parking slot status is managed by the BLEs.

## **3.1.3 Software Interfaces**

The mobile application communicates with the GPS application in order to get geographical information about where the user is located and the visual representation of it, and with the database in order to get the information about the available parking slots. The QR code authenticates the user at the time of arrival which is managed by the Parking Administrator. The communication between the database and the web portal consists of operation concerning both reading and modifying the data. The application also communicates with the third party application for e- payment which in turn communicates with the user's bank for transactions.

### **3.1.4 Communications Interfaces**

The communication between the different parts of the system is important since they depend on each other. However, in what way the communication is achieved is not important for the system and is therefore handled by the underlying operating systems for the Application.

## **3.2 Functional Requirements**

The user functions are described in this section.

### **3.2.1 User registration**

The user needs to make an account by providing his details. The account can be edited and viewed by the user.

The user enters his name, Phone number and a maximum of 3 numbers.

After the user provides his detail, on confirmation, they are stored in the database as a new record.

An account is created of the user with all the details specified.

The database limit must be checked. Repetitive data must be avoided.

### **3.2.2 User Log-in:**

The user needs to log in to his account to book a parking slot.

The mobile number and the car used is entered to log in.

The entered credentials are compared with the database records. If the match occurs then the user is considered as signed in.

The user then gets access to his account. He has read, write privileges on his own data. In case of wrong credentials, the user is asked to enter his details again.

### **3.2.3 Account information**

The user views his account and can edit his details.

The user has the option to recharge, edit details, make and view bookings.

After selection of the required option, the user moves to the next (Recharge, booking, edit account) page as per selection.

### **3.2.4 Recharge**

The user can recharge his account.

The user enters the desired amount to be payed.

The third party application opens up for transaction through the bank. The amount recharged is converted into credits and stored.

In case of wrong transaction, roll back option must be available.

### **3.2.5 Time selection**

The user sets the time in which he needs the booking.

## Smart Parking

The time is entered by the user.

The availability of the slots as per time is checked. The availability of the slots as per time is displayed. In case no slots are available, a message is displayed.

### 3.2.6 Slot Selection

The user selects the parking slot for booking.

The user views the parking slot map and selects the desired spot.

The particular slot is considered as booked which is notified to the parking guard. The slot is booked if available after confirmation from the user.

Absence of free slots adds the user to the waiting queue.

### 3.2.7 Payment

The user pays for the booking through the credits available in the account. After selection of slot, the user confirms reduction of credits.

The balance is updated in the account.

The Booking ID is generated and the booking is confirmed.

The updating of credit balance must be synchronized.

### 3.2.8 Navigation

The user is guided to the parking slot from his current position. The GPS tracks the user's location using the GPS feature in his mobile.

The shortest route is selected with minimum traffic for the user.

The route is displayed providing the user with turn by turn directions.

In case of loss of signal, the path is recalculated by viewing the user's current position.

### 3.2.9 Cancellation

The user can cancel his booking within 30 minutes of the booking. The user selects the cancel booking option.

The cancelled booking for the slot is updated in the servers and set as free. Refund of credits is initiated.

A full refund of the credits is given to the user and the booking is cancelled.

The synchronization between the cancellation and server information must be checked.

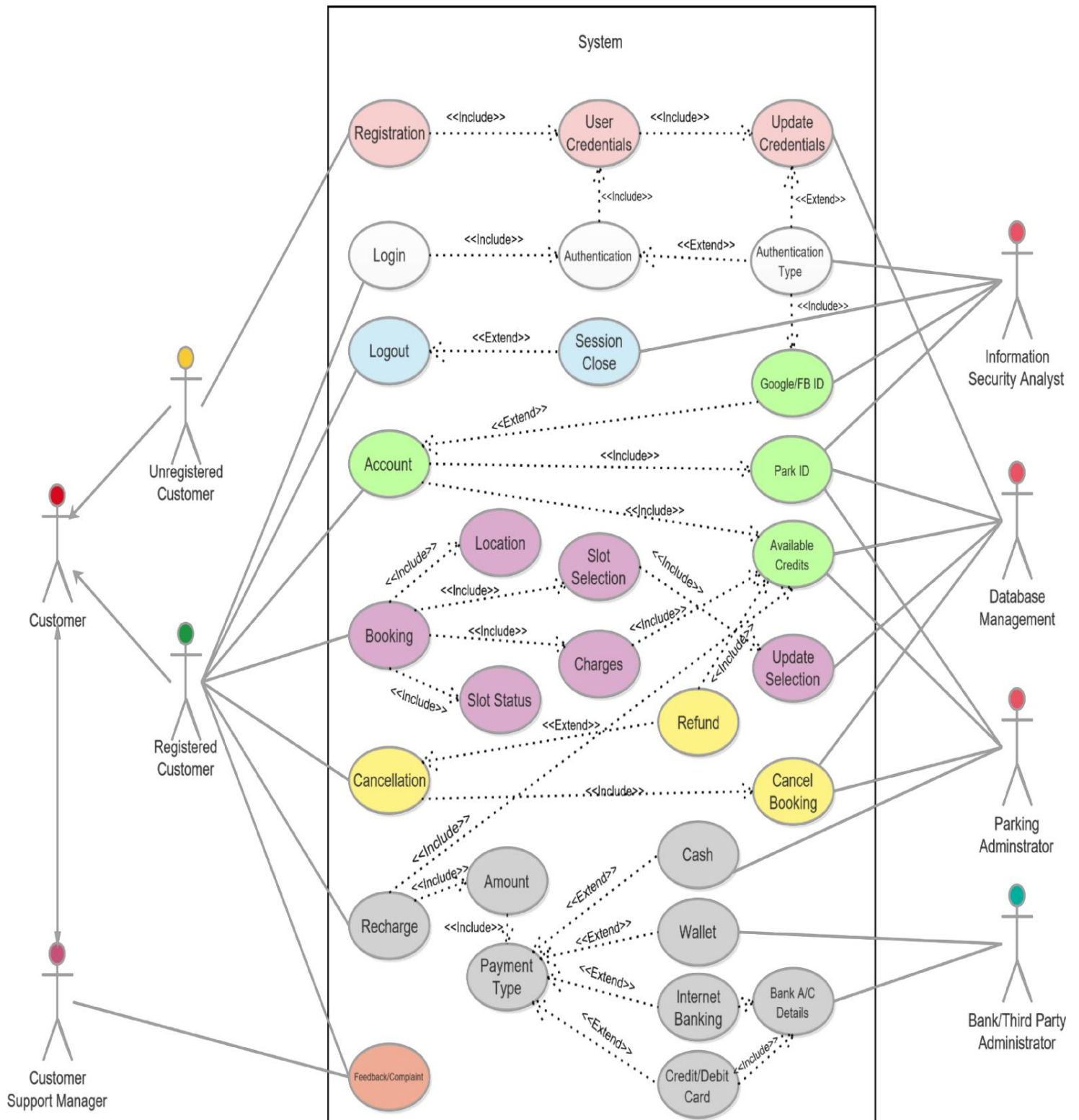
### 3.2.10 Feedback/Complaint

The user can submit a feedback or complaint.

The user can either call the customer support manager or send also file the complaint via e-mail.

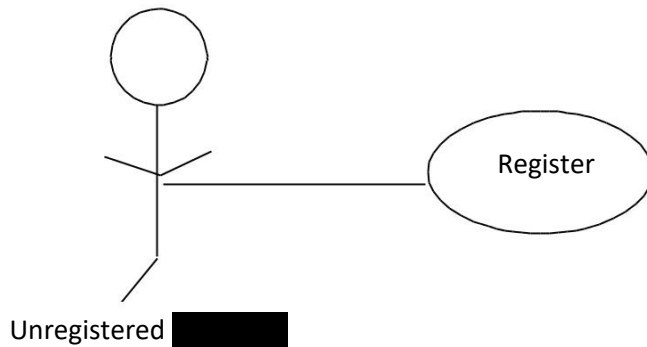
The reply is sent from the customer support manager to the e-mail address of the customer if the mail option is used.

### 3.3 Use case:



**Customer:**

**Use case 3.3.1: Register**



**Brief Description**

A new user registers to create a new account into the application

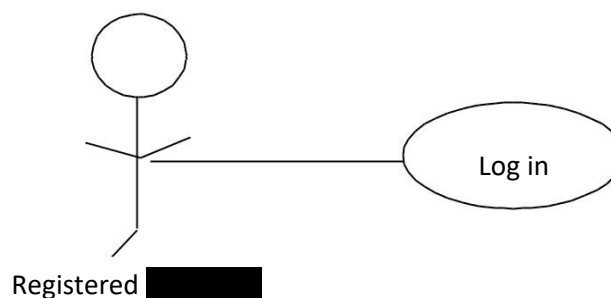
**Initial Step-By-Step Description**

The user creates a new account in this use case

Before this use case can be initiated, the Users is already provided with the environment.

1. User opens the application and clicks on the register button to register himself
2. After this the details for registration is displayed and user enters mobile number name, his car no., e-mail address, name and clicks on submit.
3. The user credentials are updated in the database which can be later updated.
4. After this registration is completed.

**Use case 3.3.2: Log in:**



**Brief description**

In order to use the application user must enter the login details.

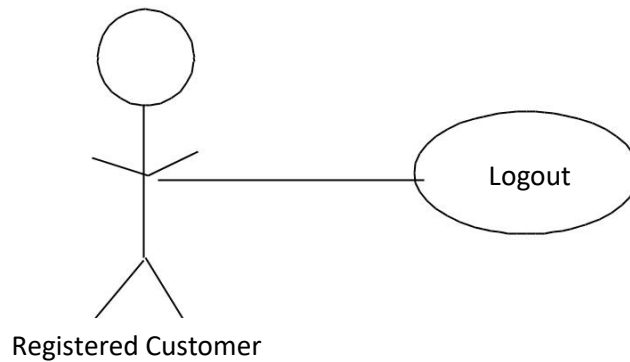
**Initial Step-By-Step Description**

Before this use case can be initiated, the Users is already provided with the environment.

1. Users on opening the application enters the login credentials.
2. If the user is authenticated, he will be granted to use the application.

3. No access is provided to the user without credentials.
4. The information security analysts monitor the security of the process.

#### Use case 3.3.3 Log out



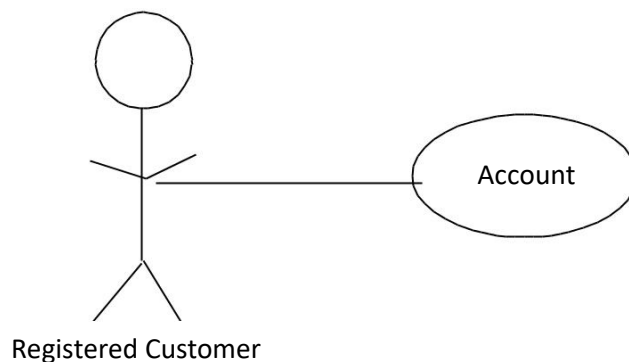
##### **Brief description.**

The user logs out and closes the session currently in progress

##### **Initial Step-By-Step Description**

1. The user clicks on the log out option.
2. The confirmation is received from the user.
3. The session is then closed and the home page is displayed.

#### Use case 3.3.4: Account



##### **Brief Description**

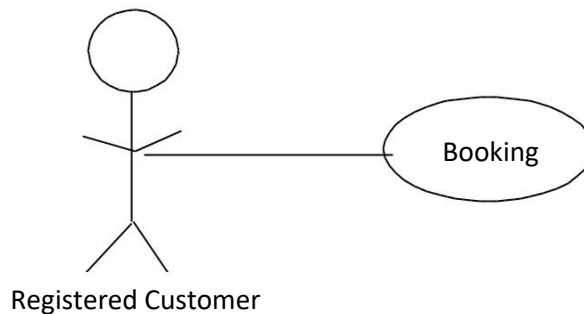
The user can view his account information and edit

##### **Initial Step-By-Step Description**

1. The user enters into his account.
2. The user can also access an account through google/FB ID.
3. The user can edit by clicking on edit option.
4. The database is updated with new information.
5. The available credits are also displayed on this page.
6. The unique park ID of the customer is also displayed.

7. The user can also move to other pages like booking, recharge booking history, through this page.

### Use case 3.3.5: Booking



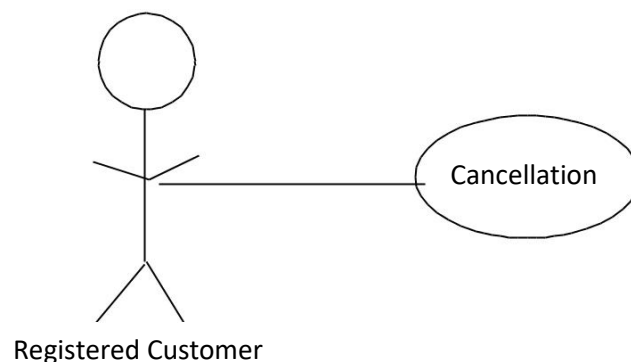
#### Brief Description

In this user books the parking space.

#### Initial Step-By-Step Description

1. The parking map is displayed with the status of each slot which is continuously updated by the BLEs.
2. The parking slot is selected by the user in this step from the available spots.
3. Here the user selects the parking slot, then selects the estimated arrival time.
4. After the parking slot is selected and payment charges are received from the credits of the user, the booking ID is generated and the booking is confirmed.
5. The user can update the selection before payment.
6. The location of the user is tracked through the GPS for navigation.

### Use case 3.3.6: Cancellation



#### Brief Description

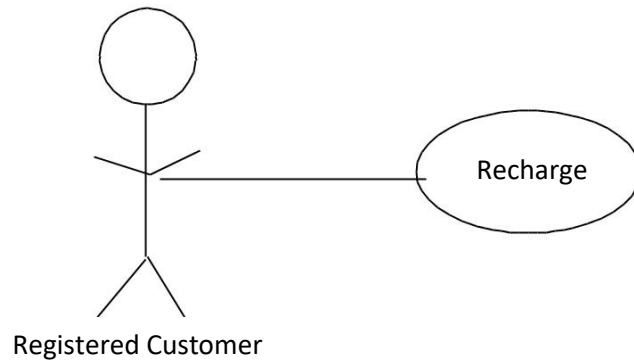
The user is allowed to cancel a booking within 30 minutes of the transaction.



### Initial Step-By-Step Description

1. The user selects on cancel booking on the screen.
2. After confirmation, the booking is cancelled and the booked slot is set free.
3. A refund is given if the cancellation is done within this time else no refund is obtained.

### Use case 3.3.7: Recharge



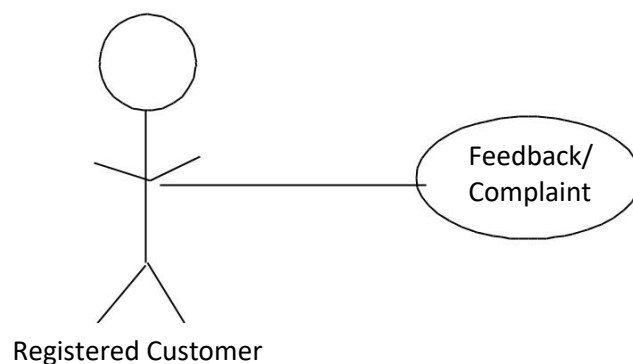
### Brief Description

This is introduced for quick payment at the time of booking.

### Initial Step-By-Step Description

1. The user must enter the amount to be recharged
2. User is directed to the third party site to complete the pay.
3. The user selects the payment type: credit/debit/net banking/e-wallet.
4. The payment credentials are submitted and verified.
5. The money then is transferred to the user's account as credit which can be used later for faster transaction.

### Use case 3.3.8. Feedback/complaint



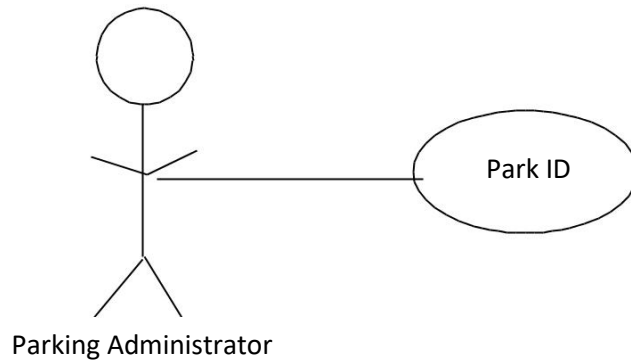
### Brief description:

The user can submit feedback/query through the application **Initial step by step description**

1. The user clicks on the feedback/complaint option.
2. The user can either talk to a customer support manager or send an e-mail.
3. The response to the e-mail is replied back to the given e-mail address of the user.
4. This section is handled by the customer support manager.

### 3.4. Use case for Parking Administrator:

#### Use case 3.4.1: Park ID



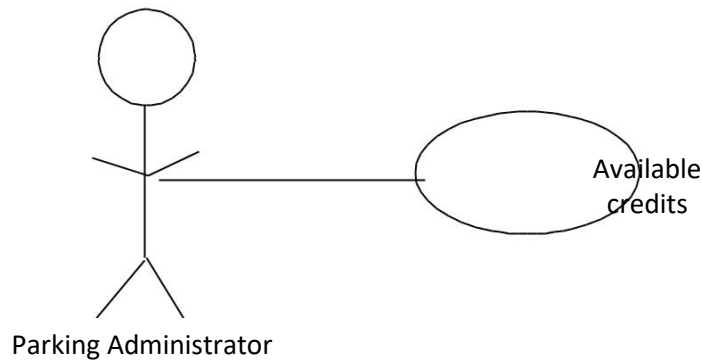
#### **Brief description.**

The Parking Administrator has access to the user's park ID for authentication while he enters the campus.

#### **Initial Step-By-Step Description**

1. Each user has a unique park ID which is viewed by the Parking Administrator.
2. The QR code of the user is scanned and the slot booked by the park ID is matched.
3. The BLE takes over after scanning the QR code.

### Use case 3.4.2: Available credits



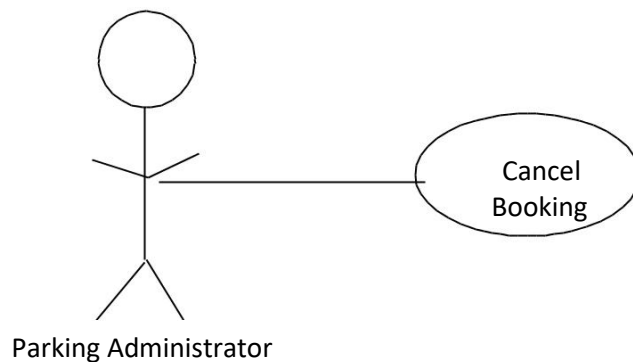
#### Brief description

The parking Administrator can

#### Initial step by step description

1. If a registered user comes on the spot without a booking ,then the parking administrator views the park ID of the user.
2. He can deduct the available credits from the user's account to allot a slot for the registered user.

### Use case 3.4.3: Cancel booking



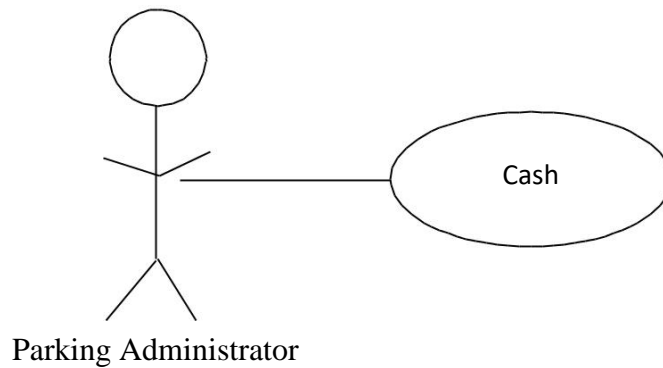
#### Brief description

The parking Administrator also has the authority to cancel a booking (At peak hours).

#### Initial step by step description

1. The parking administrator selects on a slot books but not occupied by the estimated arrival time.
2. He then selects the cancel booking option and confirms the cancellation.
3. The slot can be now occupied by someone else.
4. The credits are now updated accordingly in the user's account.

### 3.4.4 Cash



**Brief description.**

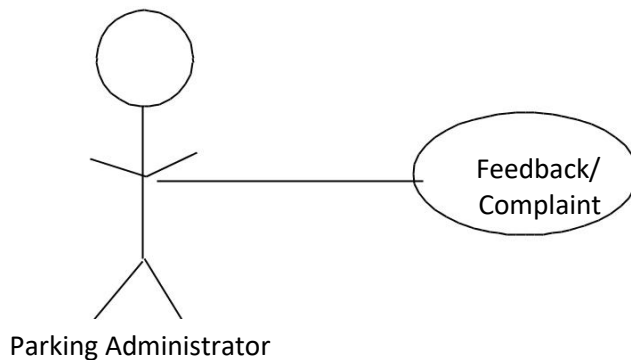
The on the spot parking users who have no bookings are managed through cash payments

**Initial Step-By-Step Description**

1. The users who appear without booking enter the gate.
2. The parking administrator receives cash payment from this user.
3. The parking slot is allotted only if free/ unbooked slots are available.

### 3.5.Customer Support manager

#### Use case 3.5.1: Feedback/Complaint



**Brief description**

The customer support manager receives feedback/complaint from users and responds to them. **Initial step by step description**

1. The customer support manager answers calls from the user where he interacts with them.
2. He also receives mails from the user which are responded back to the user's e-mail ID.

## **3.5 Non-Functional Requirements**

### **3.5.1 Functionality**

Ability to obtain a vacant spot without losing time. Ability to reserve a spot online.

Ability to provide appropriate spots for reserved spots.

Ability to pay for by using the smart phone by Mobile Wallet. Ability to navigate to the parking spot easily.

### **3.5.2 Usability**

Easy to understand parking rules.

The system should be easy to handle. Available service any time.

Easy to use the mobile app without any difficulty.

Short time and steps to make a reservation.

Providing a guideline to the customer.

Short time to get a response for the reservation.

Easy to find reservation icon on the website.

### **3.5.3 Reliability**

The system will be available 24 hours in 7 days and greater than or equal to 99.9%

Software reliability is greater or equal to 99.99%.

Smart Parking system shall have no defect that can interface with normal operation with the mobile application.

The system should be trustworthy.

The system will reduce the customer's money and time to find the spot. The system must have a strong security.

The size for the database will be acceptable.

The database must be synchronized after interaction with user and PG. Transactions should be correct and accurate.

### **3.5.4 Performance**

Software should have instant response to user inputs.

The performance is determined by the average response time for each request given by the user and the transaction done per a second.

The update of the system will be in short time.

The time taken to search for slots should be minimum. Credit transaction should be quick.

### **3.5.6 Supportability**

The system will be fully supported for each customer.

Frequent software updates will be available to the customer. The system will run on different platforms.

The update of the system will be in short time. The system will work automatically.

### **3.7 Design Constraints**

The user's phone must be accessible to the internet at all times.

The user's phone must be compatible to all the requirements.

The update of the crucial data like slot availability must be synchronized.

There should be proper synchronization between third party and the merchant site so that bookings can be confirmed.

The user's phone must be compatible with the required software's.

### **3.8 Logical Database Requirements**

The database must be maintained to manage all the user data including mobile Number, car numbers and other details. The details of various slots in the parking area must also be maintained. The database is updated during each transaction and changes made by the user. It is affected by the data given by user and parking guard. The security of database must be maintained by the administrator.



