Software Requirements Specification

for

Parking Assignment Problem

Version 1.0 approved

Prepared by:	Gurupungav Narayanan	16CO114
	Nihal Haneef	16CO128

Organisation: National Institute of Technology Surathkal

Date: 26th February 2018

Table of Contents

1. Introduction

The scope, description and the basic overview of the entire document is given in this section. A list of definitions, acronyms, abbreviations as well as references is also given in this section.

1.1 Purpose

The purpose of the document is to give to a detailed description of the working of the "Parking Assignment Software"(PAS). It tells the reader the functionalities, the domain of use and the design specifications of the system. It helps users decide whether the software meets their needs, helps managers decide contract details, helps designers and developers with development and helps maintainers with verification of the software.

1.2 Scope

The PAS assigns driver the closest parking slots available based on the destination they have entered into the system. It requires the client to feed the parking layout of the area into the software database beforehand. It gives the client feedback in terms of logistics and various other statistical information.

PAS requires a console for inputting information from the driver as well as OCR readers so as to verify whether a car has parked at the right designated slot. An alarm facility also has to be provided to alert security in case of problems and errors.

1.3 Definitions, Acronyms and Abbreviations

Term	Definition
Console	A tablet like device present at the entry point of the parking layout.
PAS	Stands for Parking Assignment Software. The main software program responsible for calculating optimal parking slot for driver.
Identifier	A sub module responsible for extracting number plate information from an image of the number plate taken by SOU.

SOU	Stands for Slot OCR Unit. Responsible for capturing image of the number plate of the car.
Driver	The owner of the car. The user of the console. The user who needs a slot to be assigned to him/her at the parking layout.
Client	The user who bought the software for his/her parking layout. The main administrator of the system.
LAN	Stands for Local Area Network. Used for connecting two devices/subsystems in a system.
Button	Area on screen which when clicked/touched resulted in shifting of screens. Used for interacting with the subsystems.
Transition Screen	Pop-up like interface present on screens.
Layout	The area where the parking slots are present. Owned by the client.
Offense	Illegal parking ,i.e , parking at an unassigned slot is called so.

1.4 References

[1] IEEE Software Requirements Specification Template (SRS) - MSU CSE, http://www.cse.msu.edu/~cse870/IEEEXplore-SRS-template.pdf , Feb 23rd 2010.

[2] D Berchman, Optical character recognition: An overview and an insight, www.http://ieeexplore.ieee.org/document/6993174/ , Dec 22nd 2014.

[3] N Taheri, A fair assignment of drivers to parking lots - IEEE Conference Publication, www.http://ieeexplore.ieee.org/document/8090788/ , Nov 2nd 2017.

1.5 Overview

The remainder of the document has 3 chapters and few appendices. The second chapters tells the various functionalities of the software. It tells the assumptions taken during development, various constraints and stakeholders.

The third chapters tells in detail to the reader the various interfaces that being utilized in the development of the software. It also highlights the functional and performance requirements of the software. It also brings into lights the design constraints and various software system attributes.

Appendix A will show in detail the various UML diagrams used during the Requirement Engineering stage of the development of the software. The system can be understood accurately after a careful analysis of these diagrams.

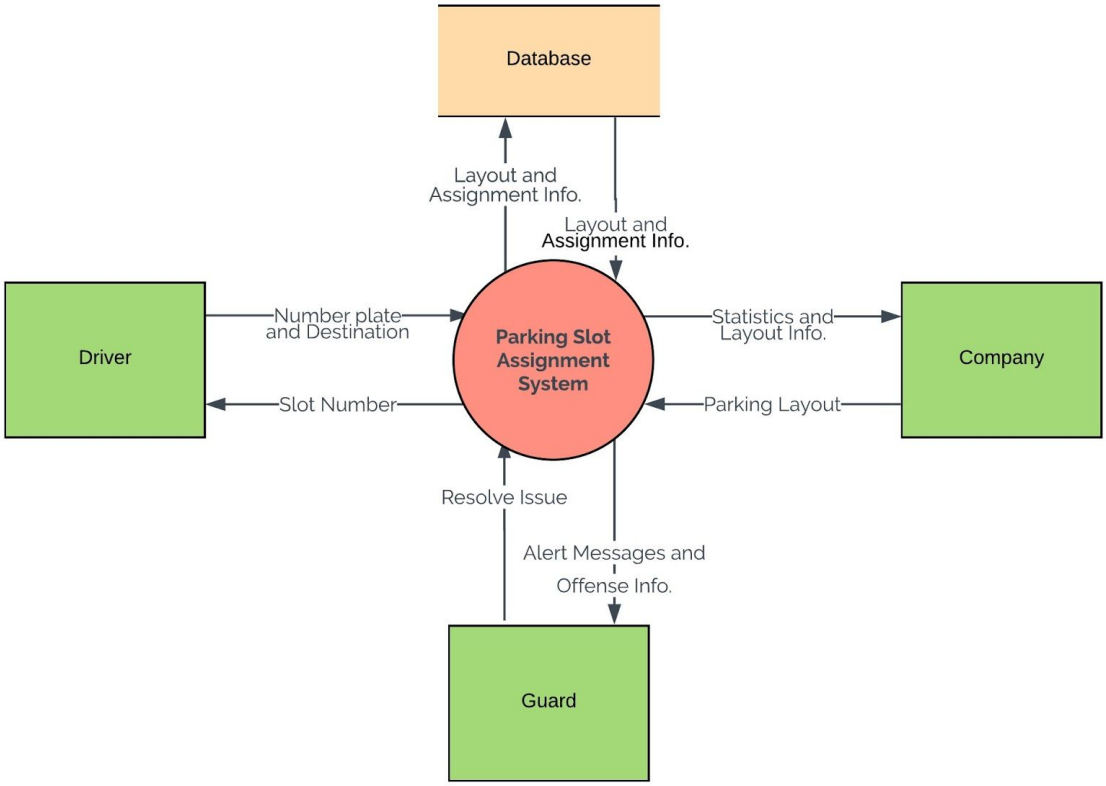
2. Overall Description

This section gives the reader a detailed description on how the software will interact with the various users and how it will operate in its domain. At last, the constraints and assumptions for the system will be presented.

2.1 Product Perspective

The system consists of multiple parts: a mobile based software for the console, a desktop application (PAS) , a desktop based Identifier submodule, camera based hardware for every Slot OCR Unit , a mobile based software for the security personnel and an IR sensor for every slot (to detect whether a car has is present at the slot or not).

The mobile application for the console is responsible for obtaining information from the driver and displaying the necessary assigned slot. It interacts with appropriate mobile hardware (it could be run on a tablet) which will provide networking capabilities using which it can interact with the PAS (via WiFi/ Physical LAN). It also interacts with a printer hardware to print a slip showing the slot number.



CONTEXT DIAGRAM

The PAS will be run on a desktop environment and will have networking capabilities to interact with the Slot OCR Units, the identifier and the console. It is responsible for calculating the optimal slot and providing logistics. The Slot OCR Units are responsible for capturing number plate images. The SOU interact with the PAS through a network. The mobile software for the security personnel has integrated wifi modules to interact with the PAS.

As the program is quite data centric , it requires a database. The PAS will communicate with the database via a server-client implementation. The Identifier sub module is solely responsible for returning number plate information to the PAS upon receiving an image query from the PAS.

2.2 Product Functions

With the mobile application for the console, the driver will be able to input their desired destination and receive necessary slip showing the slot number via the appropriate printer hardware which is in interaction with the console. The camera at the console captures the car's number plate as an image and sends it to the PAS.

The PAS is responsible for calculating the optimal parking slot for the driver based on given constraints and send back query to the console. It is responsible for calculating and sending logistics of the parking layout to the client. It send images captured by the cameras as queries to the identifier. It is responsible for sending alert messages to the security personnel,s mobile software. The client has the power to disable slots to restore order at the layout via the PAS.

The identifier is a sub module which is a machine learning model responsible for accepting image queries from the PAS and then returns the predicted number plate value back to the PAS. It is necessary for verification purposes, i.e , whether a car has parked at the appropriate parking slot.

The mobile software for the security personnel will provide alert messages in case problems arise at the parking layout. Alert messages will be sent by the PAS in case of illegal parking or errors in the software and they can view the slot number, number plate and other details of the case.

2.3 User Characteristics

There are three types of users that interact with the system ; the users of the mobile software of the console a.k.a the drivers , the users of the mobile software at the security end a.k.a the security personnel and the client who owns the PSA.

The drivers will only be able to select a destination from a given list of destinations present on the console. They will be provided with a slip indicating their allotted parking slot. They have to make sure the car they own has a proper number plate, whose picture will be taken by the cameras present at the layout for verification and logistical purposes.

The security personnel will only be able to receive alert messages from the mobile software which is linked with the PAS in cases of emergences, illegal parking or errors. They can view the number plate, slot number and the time of parking of the particular cart. They have to be physically present at the parking layout to make sure everything runs smoothly at the layout.

The client will be in charge of the PAS. They can view the logistics and statistical information of their parking layout which will be provided by the PAS. Management of layout can also be done from the client's side to meet new emerging needs.

2.4 Constraints

Network connectivity via Wifi or LAN is a constraint for the entire system. An active active connection should be present at all times for the proper interaction of the various sub systems with each other, without which the system will fail.

The system is also constrained by the size of the database. If it doesn't have enough space, it might lead to queueing of a lot of incoming requests resulting in a lot of time to fetch data, which is not appreciated.

The parking layout information will have to be inputted into the system in a specific format. This is required to standardize the various layouts present in the world.

2.5 Assumptions and Dependencies

This version of the software assumes that a linear/flat parking layout will be given as input to the PAS and not multi-level parking layouts.

The mobile softwares used in the system assumes that the mobile hardware interfaces have enough performance. They should be able to provide networking capabilities like WiFi as and when it is required.

3. Specific Requirements

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

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 Console Application

The console application should show a button ‘Start’ for a new driver arriving at the start point. (see Fig. 1.1).
When this button is pressed, the driver should be taken to a different activity screen showing a scrollable list of destinations which are present in the locality of the layout (the list is provided by the client). The driver should be able to choose a destination from this scrollable list. An option for “No preference” which can be chosen by the driver if they have no particular destination on their mind. A timer clocked to 30 seconds is present on the top of the screen. Failure to choose a destination before the timer elapses should lead to allocation of a random free slot to the driver (see Fig 1.2).

The driver should be taken to a loading screen, during which the Slot OCR Unit should capture a picture of the number plate and should send a query to the identifier to receive the value of the number plate. (see Fig. 1.3).

A screen should then appear which should show the driver's assigned slot number. The printer should print a slip showing the time of entry, slot number and the number plate of the car, which should be retained by the driver (see Fig. 1.4).

The driver should be shown a "Proceed" message after which the driver can proceed to park their car in the assigned slot (see Fig. 1.5).

3.1.1.2 Parking Assignment Software

The PAS should be a desktop application. It should begin with a login page where the client enters their username and password (see Fig. 2.1).

After a successful login attempt, the PAS should shift to a screen where three options should be available for the client ; One to see 'Statistics' of the parking layout, second to 'Manage the Layout' and third to 'List of Broken Slot OCR Units' (see Fig 2.2).

When the statistics button is clicked, the PAS should shift to another screen where the parking layout will be displayed with each slot acting as an interactive button. Below this layout usage statistics should be displayed (see Fig 2.3).

When a particular slot is clicked, a transition screen should open up which should show slot log (which should include at what specific times the slot has been occupied and the cars that were there at that slot) , Slot OCR Unit Error rate (which is the number of times a personnel has marked the Slot OCR Unit producing false outputs by the number of total outputs) and the current slot state (see Fig 2.4).

When the 'Manage the Layout' button is clicked on, a screen showing a list of three options should be shown ; One saying 'Add layout', second saying 'Modify Layout and third saying 'Delete Layout' (see Fig. 2.5).

Upon clicking add layout button the PAS should shift to a screen with a button saying "Add Standardized Blueprint for Layout' , which when clicked allows client to choose blueprint and upload onto the system as an image. When the modify button is clicked the parking layout is shown and the client should be able to click on a slot an enable or disable the slot. When the delete button is clicked the client should be able to delete existing layout after further confirmation (see Fig. 2.6).

When list of broken OCR units button is clicked the PAS should shift to a screen showing a list of Slot OCR units which are broken and are not functioning properly (see Fig.2.7).

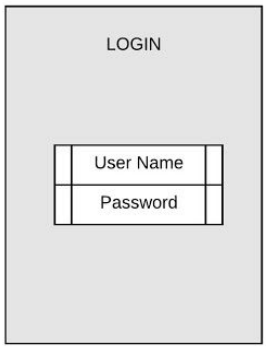


Fig. 2.1

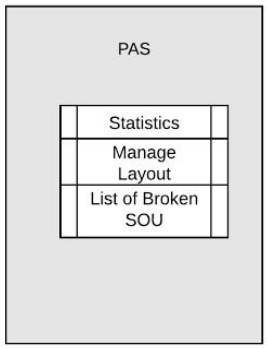


Fig. 2.2

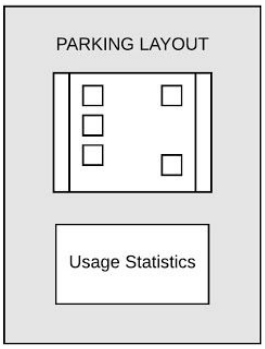


Fig. 2.3

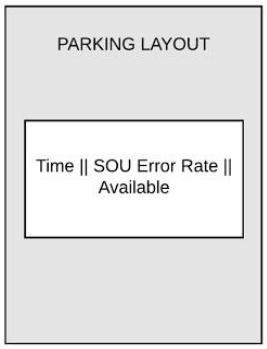


Fig. 2.4

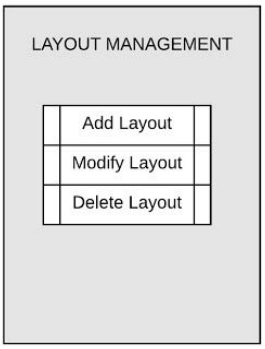
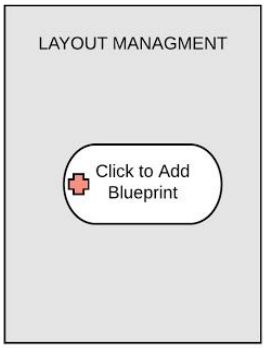
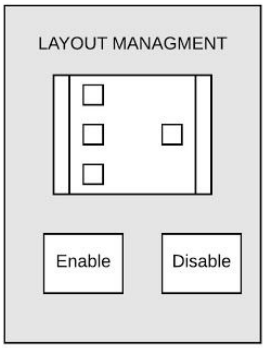


Fig. 2.5





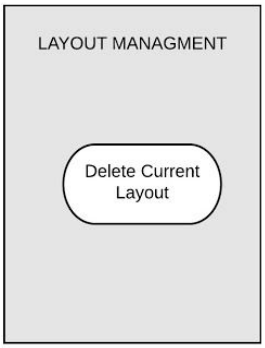


Fig. 2.6

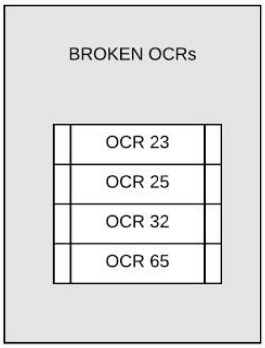


Fig. 2.7

3.1.1.3 Guard Application

The guard application should show a screen showing two options ; One to show the ‘Real Time Parking Layout’ and second to show ‘List of offenses’ (see Fig. 3.1).

Upon clicking real time parking layout, the layout should be shown in accordance with color scheme and properties as mentioned in ‘Section 3.2.2.1’. When a slot is clicked, a transition screen showing slot number, slot state and offenses list should be shown with a resolve icon (see Fig. 3.2).

When list of offenses button is clicked the application should shift to a screen showing a list of offenses that are active at the moment. When an offense is clicked on the application should show a transition screen showing slot number, slot state and offenses list should be shown with a resolve icon (see Fig 3.3).

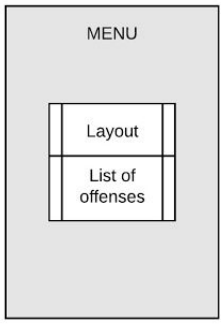
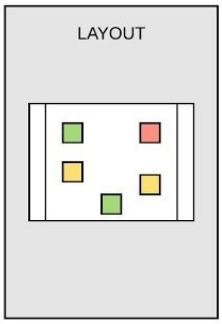


Fig. 3.1



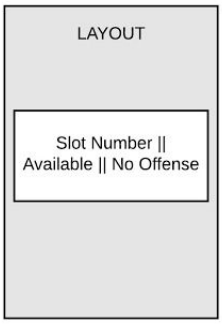
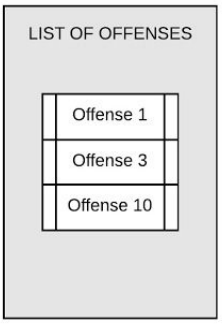


Fig. 3.2



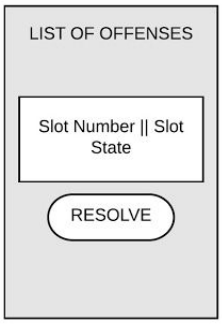


Fig. 3.3

3.1.2 Hardware Interfaces

The system requires a few hardware interfaces for the proper functioning of the system. The Slot OCR units contains cameras which are responsible for capturing number plate images of the cars. They have network adapters via which they can send these images to the PAS.

IR sensors are present at each parking slot which detects if a car is present at the parking slot or not. It is connected to the PAS with the help of cables and wires.

Console should be implemented with a tablet like device which has networking capabilities. The security personnel should have mobile devices with similar capabilities.

3.1.3 Software Interfaces

Each of the sub-systems should have software interfaces that support networking so as to provide interactions between all the components within the system which is vital for the proper functioning of the system.

The camera in the slot OCR units require a camera API so as to capture and store images. This has to then interact with the network adapter and networking interface to send the image to the PAS.

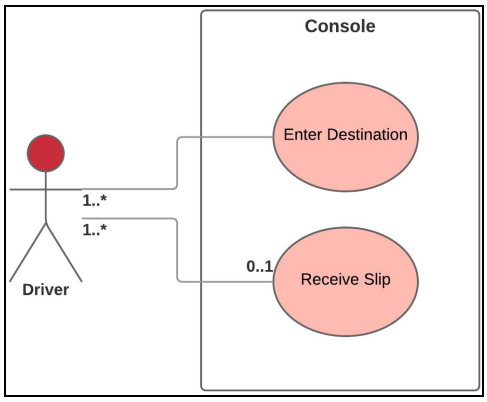
3.1.4 Communications Interfaces

Operating systems will take care of the underlying communication in each sub system. Different subsystems will communicate with each other via networking ,i.e, via Wifi or physical LAN.

3.2 Functional Requirements

This section includes the requirements that specify all fundamental actions of the software system.

3.2.1 User Class 1 - The Driver



3.2.1.1 *Functional Requirement 1.1*

ID: FR1
TITLE: Console - Destinations View
DESC: The console( ) should display the list of destinations to the user. The console should also display the number of parking lots available near the destination.
RAT: In order to give enough information to a driver to make a choice.
DEP: None

3.2.1.2 *Functional Requirement 1.2*

ID: FR2
TITLE: Console - Select Destination
DESC: A driver should be able to select a destination by interacting with the console.
RAT: In order for a driver to select his destination
DEP: FR1

3.2.1.3 *Functional Requirement 1.3*

ID: PR1
TITLE: Identifier - Input Number Plate Picture
DESC: The Identifier subsystem should accept a picture of a number plate as input.
RAT: In order to scan through the image and find the number plate.
DEP: none

3.2.1.3 *Functional Requirement 1.4*

ID: PR2
TITLE: Identifier - Return Number Plate Value
DESC: The Identifier subsystem should be able to recognize the number plate and return the value to the main system
RAT: In order to store the number plate value

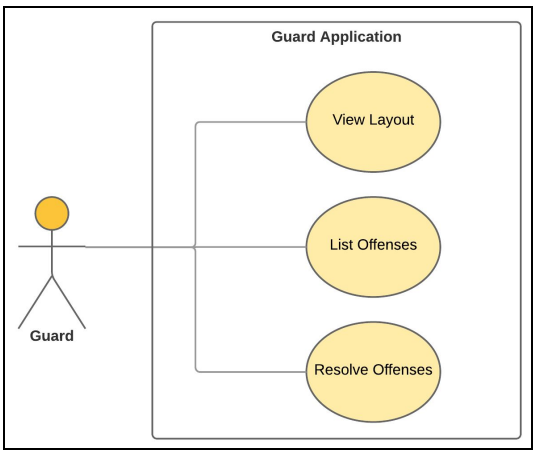
3.2.1.3 *Functional Requirement 1.5*

ID: PR3

TITLE: Calculate Ideal Slot
DESC: The system should be able to calculate an ideal slot based on console input with driver preferences.
RAT: In order to inform the driver which slot to park in

3.2.1.4 *Functional Requirement 1.6*

ID: FR3
TITLE: Console - Return Assigned Slot
DESC: The console should display the assigned slot on its screen and print a slip that clearly indicates the assigned slot number.
RAT: In order for the driver to know which slot to park in. The slip would later be used for error correction.,
DEP: FR1, FR2, PR3



3.2.2 User Class 2 - Security Personnel

3.2.2.1 *Functional Requirement 2.1*

ID: FR4
TITLE: Guard App - Display Layout
DESC: The personnel should be able to see a clean layout of the parking spaces which indicates the current status for every parking lot. The parking lots should be colored to indicate their status:

1. Green = Vacant and Unassigned Space
2. Yellow = Vacant Space but Assigned to a Car
3. Blue = Occupied and no offense
4. Red = Occupied and possible offense
5. Black = Disabled Parking Space

RAT: Would allow the personnel to easily pinpoint problems and ensure the system's functioning
DEP: PR1, PR2

3.2.2.2 *Functional Requirement 2.2*

ID: FR5
TITLE: Guard App - Display list of offenses
DESC: The personnel should be able to view a list of all possible offences in the form of a table.
RAT: In order for the personnel to take necessary action by physically visiting the slot.
DEP: PR1, PR2

3.2.2.3 *Functional Requirement 2.3*

ID: FR6
FEATURE: Resolve Offenses
In order to resolve offenses
A security personnel
Should have access to the list of possible offenses (FR4)

Scenario: Number Plate Mismatch and Car is in a wrong slot
Given a Car was assigned a slot.
But the Car was parked at a slot that is different than the one assigned
Then such a slot would be marked Red in the layout (FR3)
And would be listed as a possible offense (FR4)
When the personnel confirms that the car is in a wrong slot
Then the personnel has resolved the offense

Scenario: Number Plate Mismatch but Car is in the right slot
Given a Car was assigned a slot.
And the car was parked correctly in the assigned slot
But the Slot OCR Unit returned a number different to that of the system
Then such a slot would be marked Red in the layout (FR3)
And would be listed as a possible offense (FR4)
When the personnel confirms that the car is in a right slot
Then the personnel has resolved the offense

3.2.3 User Class 3 - Company Administrator

3.2.3.1 *Functional Requirement 3.1*

ID: FR10
FEATURE: Administrator log in
In order to administer the system
An administrator
Should be logged in to the PAS

Scenario: Successful log-in
Given the administrator wants to log in
When the administrator logs in with an administrator account
Then the administrator should be logged in as an administrator

3.2.3.2 *Functional Requirement 3.2*

ID: FR7
FEATURE: Manage Layout
In order to monitor and manage the system
An administrator

Should be able to manage the Parking Slot Layout

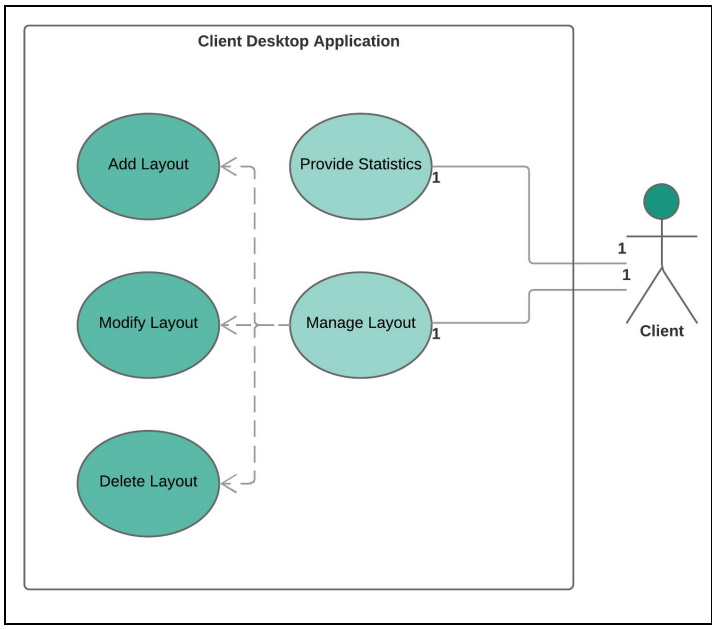
Scenario: Add a Layout
Given a layout is not already in use
When the administrator provides an image with the layout blueprint
Then the layout should be adopted to be used by the system

Scenario: Modify a Layout
Given a layout is already in use
When the administrator enables a new slot
Or When the administrator disables a slot
Then the updated layout should be used by the system

Scenario: Delete a Layout
Given a layout is already in use
When the administrator deletes the layout
Then the layout should no longer be used by the system.

3.2.2.3 *Functional Requirement 3.3*

ID: FR8
TITLE: List erratic/broken Slot OCR Units
DESC: The administrator should be able to view a list of Slot OCR Units detected by the system that are giving erratic outputs or no outputs.
RAT: In order for the administrator to take appropriate action for the units
DEP: PR1, PR2



3.2.3.4 *Functional Requirement 3.4*

ID: FR9
TITLE: Display Statistics
DESC: The administrator should be able to view detailed statistics of slots including:

1. Slot Logs - which should include at what specific times the slot has been occupied
2. Slot OCR Unit Error rate - The number of times a personnel has marked the Slot OCR Unit producing false outputs (FR6) / Number of total outputs
3. Current Slot State - Whether the slot is enabled/disabled (FR7)

RAT: To provide the administrator with useful usage statistics and system functioning
DEP: FR6, FR7, FR10

3.3 Non-Functional Requirements

The requirements in this section provide a detailed specification of the user interaction with the software and measurements placed on the system performance, reliability and security.

3.3.1 *Response Time*

ID: QR1
GIST: The fastness of the slot assignment
SCALE: The Response TIme of a slot assignment
METER: Measurements obtained from 1000 assignments during testing
MUST: No more than 4 seconds 100% of the time
WISH: No more than 1 seconds 100% of the time

3.3.2 *Usage of Console*

ID: QR2
TITLE: Usage of Console
DESC: The destination list should be clearly displayed and should be clearly grasped by the driver.
RAT: In order for a driver to choose a destination easily
DEP: None

3.3.3 *Usage of Guard App - Display Layout*

ID: QR3
TITLE: Usage of Guard App - Display Layout
DESC: The layout should be akin to the actual physical layout. The colors should be distinct and differentiable.
RAT: In order for a security personnel to easily monitor the parking lots
DEP: None

3.3.4 *Usage of Guard App - List Offenses*

ID: QR4
TITLE: Usage of Guard App - List Offenses
DESC: The Offenses list should be displayed in a clear and concise manner. The list should be scrollable and the text easily legible.
RAT: In order for a security personnel to easily identify offenses

3.3.5 *Usage of List erratic/broken SOU*

ID: QR5
TITLE: Usage of List erratic/broken SOU

DESC: The erratic/broken SOU list should be displayed in a clear and concise manner. The list should be scrollable and the text easily legible.
RAT: In order for an administrator to easily identify erratic/broken SOUs

3.3.6 *Usage of Guard App - Resolve Offenses*

ID: QR6
TITLE: Usage of Guard App - Resolve Offenses Feature
DESC: The resolve offenses feature should be prominent and intuitively apparent to the security personnel.
RAT: In order for a security personnel to easily resolve offenses

3.3.7 *Usage of Manage Layout Feature*

ID: QR7
TITLE: Usage of Manage Layout Feature
DESC: The different managing options should be evident, simple and easy to understand
RAT: In order for an administrator to manage the layout easily
DEP: None

3.3.8 *Usage of Display Statistics*

ID: QR8
TITLE: Usage of Display Statistics
DESC: The various usage and system statistics should be displayed in an organized and concise manner.
RAT: In order for an administrator to easily grasp the statistics
DEP: None

3.3.7 *System Reliability*

ID: QR9
GIST: The reliability of the system
SCALE: The reliability that the system always assigns the best possible slot.
METER: Measurements obtained from 1000 assignments during testing
MUST: More than 95% of the time.
PLAN: More than 99% of the time.
WISH: 100% of all assignments

3.3.7 *Identifier Reliability*

ID: QR10
GIST: The fault tolerance of the Identifier
SCALE: The error rate of the Identifier. The ratio of number of wrong number plates recognized to the total number of plates recognized.
METER: Measurements obtained from 10000 recognitions during testing

MUST: No more than a 1% error rate.
WISH: No more than a 0.01% error rate.

3.3.8 *System Availability*

ID: QR11
GIST: The availability of the system when it is used
SCALE: The average system availability (not considering network failure)
METER: Measurements obtained from 100 hours of usage during testing
MUST: More than 98% of the time.
PLAN: More than 99% of the time.
WISH: 100% of the time.

3.3.8 *Identifier Availability*

ID: QR12
GIST: The availability of the Identifier when it is used
SCALE: The average Identifier availability (not considering network failure)
METER: Measurements obtained from 100 hours of usage during testing
MUST: More than 98% of the time.
PLAN: More than 99% of the time.
WISH: 100% of the time.

3.3.8 *Network Connection*

ID: QR13
TITLE: Network Connection
DESC: The Identifier, PAS, Consoles, Guard Application and SOUs should be connected to the same network.
RAT: In order for different subsystems to communicate with each other.
DEP: None

3.3.8 *Administrator Log-in Security*

ID: QR14
TITLE: Administrator Log-in Security
GIST: Security of accounts.
SCALE: If an administrator tries to log in to the web portal with a non-existing account then the admin should not be logged in. The admin should be notified about log-in failure.
METER: 1000 attempts to log-in with a non-existing user account during testing.
MUST: 100% of the time.

3.3.9 Communication Security

ID: QR15
GIST: Security of the communication between the system and server.

SCALE: The messages should be encrypted for log-in communications, so others cannot get user-name and password from those messages.
METER: Attempts to get user-name and password through obtained messages on 1000 log-in session during testing.
MUST: 100% of the Communication Messages in the communication of a log-in session should be encrypted.

3.4 Design Constraints

This section includes the design constraints on the software caused by the hardware.

3.4.1 Identifier Hard Drive Space

ID: QR16
GIST: Identifier Hard Drive Space
SCALE: The Identifier's need of Hard drive space
METER: Megabyte (MB)
MUST: No more than 200 MB.
PLAN: No more than 100 MB.
WISH: No more than 50 MB.

3.4.2 PAS Hard Drive Space

ID: QR17
GIST: PAS Hard Drive Space
SCALE: The PAS's need of Hard drive space
METER: Megabyte (MB)
MUST: No more than 150 MB.
PLAN: No more than 50 MB.
WISH: No more than 25 MB.

3.4.3 Guard App Hard Drive Space

ID: QR18
GIST: Guard Application Hard Drive Space
SCALE: The Guard App's need of Hard drive space on the mobile device
METER: Megabyte (MB)
MUST: No more than 50 MB.
PLAN: No more than 25 MB.
WISH: No more than 12 MB.

3.4.4 Identifier memory usage

ID: QR19
GIST: The amount of Operating System memory used by the Identifier.
SCALE: Megabyte (MB).
METER: Observations done from the performance log during testing
MUST: No more than 100 MB.
PLAN: No more than 50 MB
WISH: No more than 25 MB

3.4.5 PAS memory usage

ID: QR20

GIST: The amount of Operating System memory used by the PAS.
SCALE: Megabyte (MB).
METER: Observations done from the performance log during testing
MUST: No more than 50 MB.
PLAN: No more than 25 MB
WISH: No more than 12 MB

3.4.6 Guard Application memory usage

ID: QR21

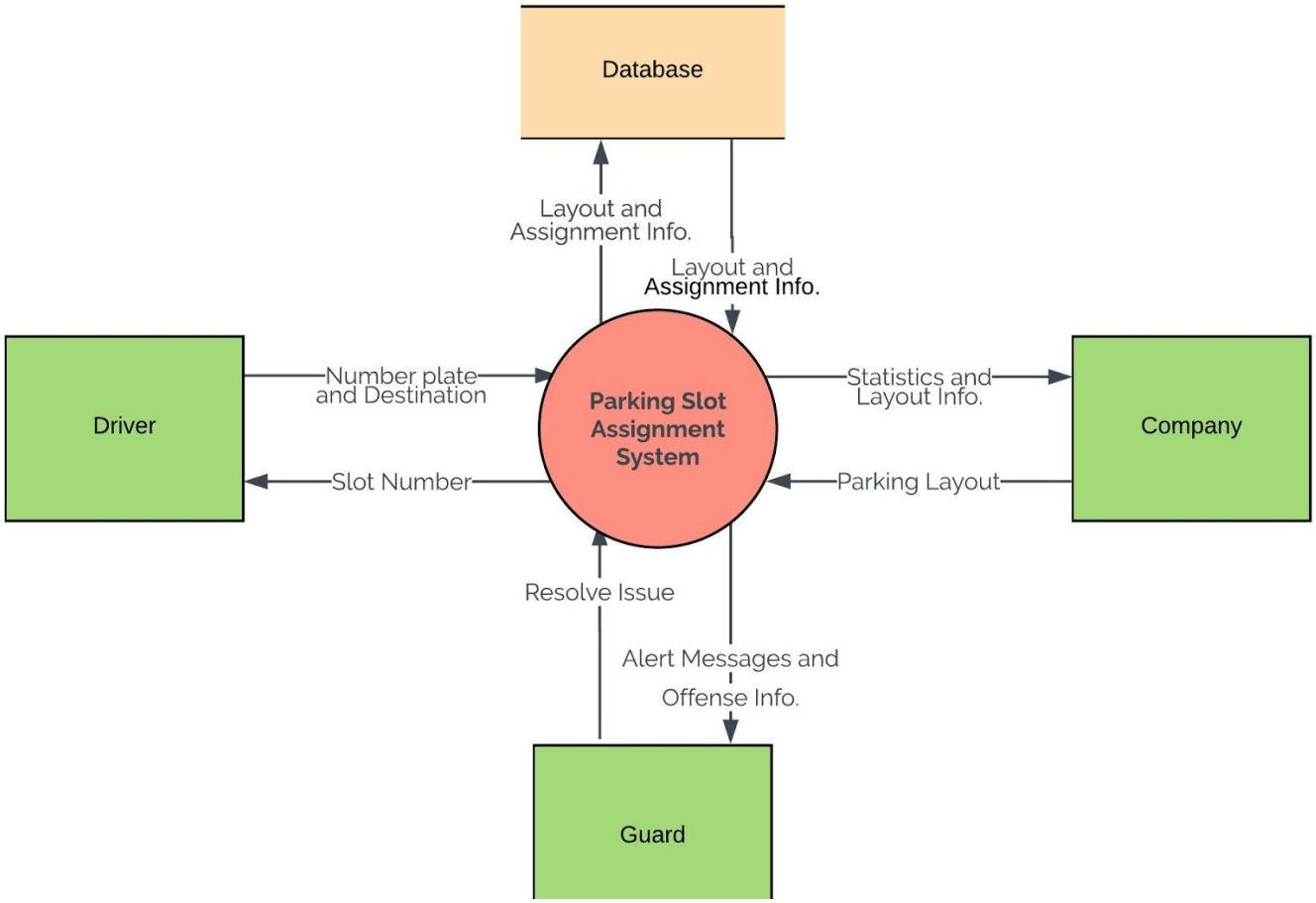
GIST: The amount of Mobile Operating System memory used by the guard application.
SCALE: MB.
METER: Observations done from the performance log during testing
MUST: No more than 20 MB.
PLAN: No more than 15 MB
WISH: No more than 10 MB

Appendix A: Analysis of Model Diagrams

In this section the various Model diagrams designed and used to give a detailed insight and explanation for the system. These models were designed during the Requirement Engineering phase of the development process of the software.

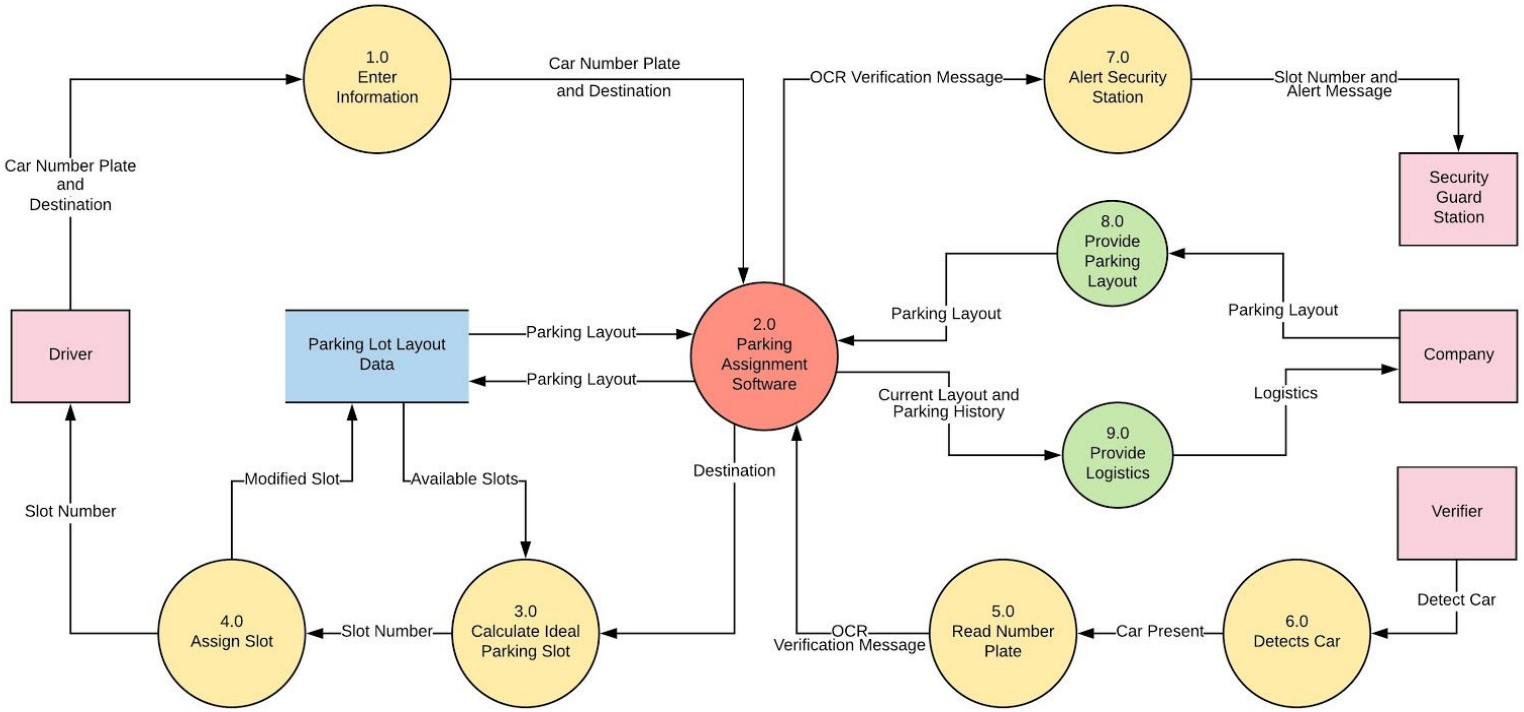
A.1 Context Diagram

Context is a model diagram which shows how the environment, users and different external entities interact with the system under consideration.



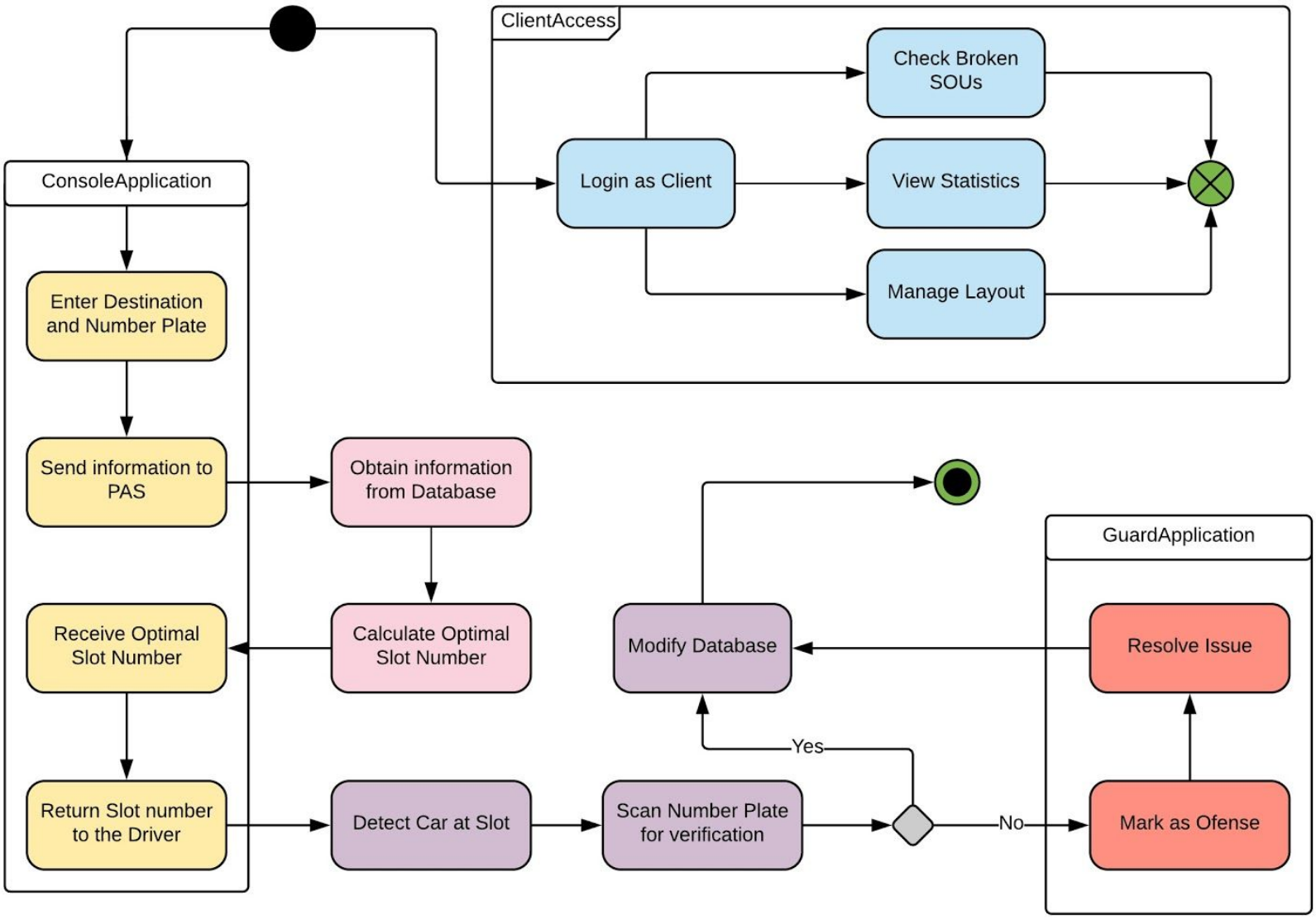
A.2 Data Flow Diagram

A data flow diagram or DFD is a graphical or visual representation of the flow of information in the system under consideration. It is the main model used to get proper insight of the system.



A.3 Activity Diagram

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modeling. They can also describe the steps in a use case diagram.



END OF SRS