Software Requirements Specification

for

Parking Pals

Version 2.0

Prepared by

Name	Student ID	E-mail
Yuanzhi Ren	**1970673**	**ianren@uw.edu**
Zeng Fu	**1970723**	**zengfu@uw.edu**
Grant Schorbach	**1421941**	**gcs96@uw.edu**
Group Name	**Group 7**	

Prepared for

Course	TCSS559 Services Computing
Instructor	Universal RFID Inc. & Eyhab Al-Masri (PhD)
Date	**December 10, 2019**

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Phase I	Yuanzhi Ren Zeng Fu Grant Schorbach	The Phase I of SRS	11/01/19
Phase II	Yuanzhi Ren Zeng Fu Grant Schorbach	The Final Version of SRS	12/10/2019

1 Introduction

Nowadays, with the number of cars increasing, the parking services are more needed. Too many cars in parking structures may cause mess without a good management system. A well-designed parking system can also bring the owner more popularity. This project is aimed to provide an excellent parking management system that will cover all basic requirements for users and owners. It also provides some web services to make users convenient and make owner easier to update the parking plan. Readers will find the general introduction of this parking management system and background information they need to know.

1.1 Document Purpose

○ The purpose of this document is to introduce our parking management systems and give readers a general overview of functionalities that this project has. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external operations. This document is mainly a reference for developing the first version of the project for the development team.

1.2 Product Scope

"Car Parking Management System" is an information-based system for customers and parking structure owners. Beside the basic payment services online, it can help customers to check the related parking information they want to know and reserve some services in the parking structures. Customers with different statuses (temporary customers, premium customers and so on) can lock different functionalities and gain different privileges. Basically,this system automatically records locations of customers' cars and shows customers real time parking fee and parking time. Obviously, this system can help customers to find their cars when customers want to leave. Furthermore, this system can help customers far away from the parking structures to learn the empty spots information, traffic information and weather near the parking structure, so that they can decide whether it is a good idea to come to this parking structure.

For parking structure owners, this system can collect historical records to help managers to make better management in the future. Using many aspects of data, they can design new or update the current plans for members.

The goal of this system is to remove all concerns from customers about parking. Customers can know all the information they need to know.

1.3 Intended Audience and Document Overview

The remainder of this document contains three chapters. The next chapter, on which "Client" type readers should focus, talks about the details about the functionalities this system provides and how to use this system. It also clarifies the interactions between different types of stakeholders and this systems. The third chapter, on which Professor as the reader should focus, is "Specific Requirements", because it mainly talks about the technical methods for developers to establish this system and describes in technical terms the details of the functionality of the product.

The fourth chapter talks about some non-functional requirements, which “Client” type readers should read. This chapter tell them some matters need attention before and after they use this system.

1.4 Definitions, Acronyms and Abbreviations

Glossary

Term	Definition
User	Customers using this system with cars
Owner	This System’s operator
RFID	Radio-frequency identification
UWT	University of Washington, Tacoma
SRS	Software Requirements Specification, a document that introduces all of the functionalities of a proposed system and the constraints under which it operates. For example, this document.

1.5 Document Conventions

1.6 References and Acknowledgments

[1] R.FI D. Journal. The History of RFID Technology. Retrieved December 8, 2019 from https://www.rfidjournal.com/articles/view?1338

[2] Anon. RFID Parking - Computer Based RFID Hands Free Parking Solution. Retrieved December 9, 2019 from https://www.nephsystem.com/rfid-computer-based-parking-solution-app-9.html

[3] Anon. Parking Control RFID System: RFID Parking Control. Retrieved December 9, 2019 from https://gaorfid.com/gao-parking-control-system/

[4] SRS Template, Retrieved October 10, 2019 from https://canvas.uw.edu/files/58757696/download?download_frd=1

2 Overall Description

2.1 Product Perspective

In order to satisfy the increasing demands of customers on convenient and secure parking, we develop our car parking management system: Parking Pal, which is a self-contained software product. With the help of RFID, our system will let our customers easily park their cars with the helpful guidance. The major components are user information management, public information retrieve, individual functions, just as the diagram showing.

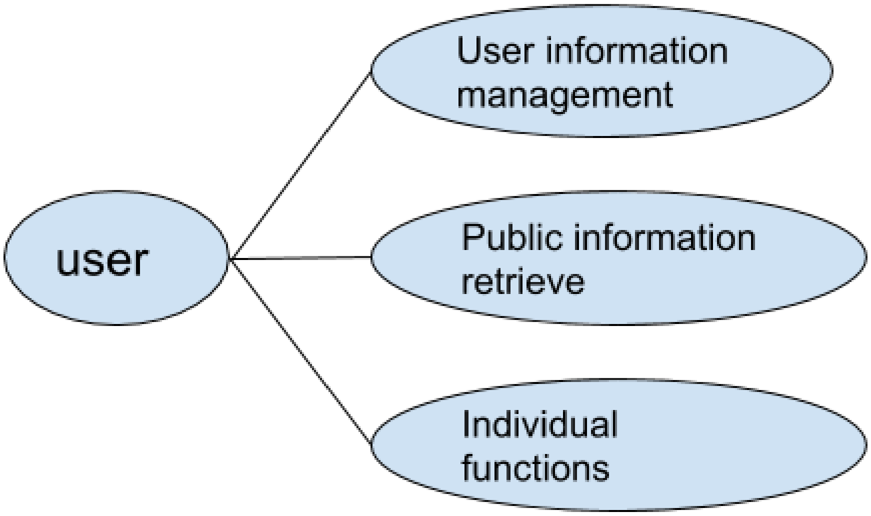


Fig2.1. product functionality components

2.2 Product Functionality

- The system should allow users to use their RFID tag to swipe the RFID reader.
- The system must be able to issue a ticket to regular users.
- The system should allow users to register accounts, log into the system, log out from the system, update user information, check user validation status and upgrade account type.
- The system should enable users to reserve parking spots, if no spots are available, the users should be notified. If other users park on a reserved spot, that user should receive a notification and will be redirected to other empty spots.
- The system should allow users to check available spots at any time.
- The system should provide users with notification about the lowest gas price around, if there is no nearby station, then send a “sorry” message.
- The system should send the users weather reminders and traffic condition messages when they are leaving the parking lot.
- The system should detect the license plate number of the users’ cars when they came in.
- the system should provide current price of the users’ cars when they would like to know.

2.3 Users and Target Markets

Our target markets are all the customers who work, live or go for shopping near UWT area. Our target users consists of all the customers who want to park their cars on UWT parking lot and system administrators. For general customers, there are several categories. In detail, RFID ticket users are customers who do not have a RFID card and they still want to park their cars. In this case, our system will assign them a temporary RFID ticket so that they can get in. RFID card users are customers who want to enjoy our service by buying a period plan with lower average parking cost, like a month plan. Except for the service we provide to RFID users, premium RFID card users are allowed to reserve the parking spots in advance. For administrators, they are users who have root access to our system, and they are allowed to check user information, parking information and reservation information.

In addition, RFID ticket users and RFID card users take most of the space of our target market, and middle sized spaces are taken by premium RFID card users. Also, administrators are individual users.

2.4 Stakeholders

The stakeholders of our system consist of managing level, persons who make decisions and come up with strategies based on marketing information; marketing level, persons who bring new idea on the product updating; developing level, developing team and testing team who build the product first hand ; user level, customers who give feedback on the product.

2.5 Operating Environment

Our system will run on at least one remote Linux 18.04 server, which provides back-end services for users with PC or mobile. Also, the system depends on Paypal API for payment process, local weather information provided by Google weather API.

For users, they are supposed to use product through web browsers, at least individual end-devices are required.

Required environment	A computer that can support five servers running together, Visual Studio, IntelliJ (Community) and a web browser (recommend Google Chrome).
Hardware devices	PC, Mobile, remote server
Hardware tools	RFID Sensor, scanner
Operating system	Windows, Mac

Third party web services	Google weather, Zipcode API, Mygasfeed API, Paypal, IP-API, ipify API, Openalpr API, Tomtom API, Openweathermap API
Others	Glassfish, JDK, MYSQL, Azure, ASP.NET, HTML, CSS, JavaScript, Java, C#

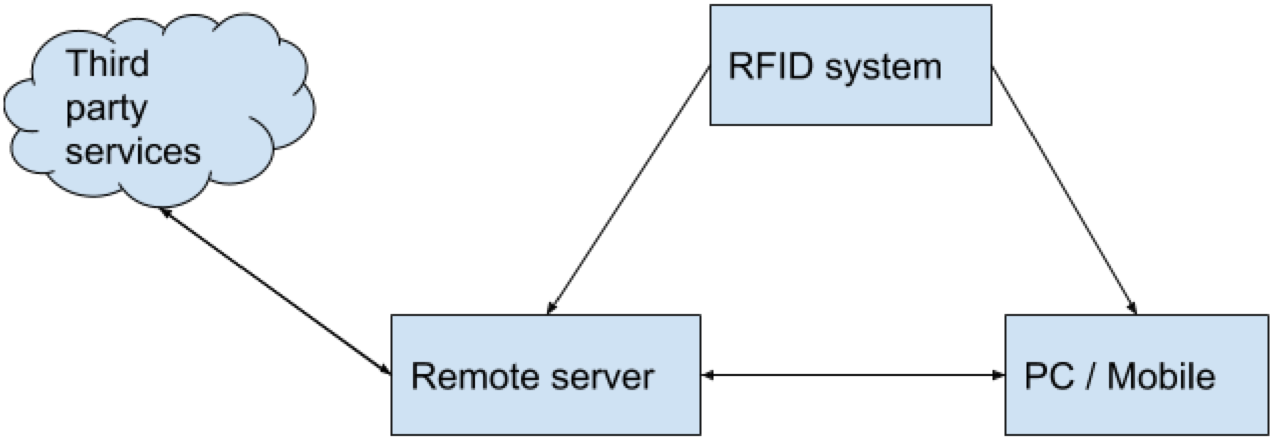


Fig2.2 System component diagram

2.6 Design and Implementation Constraints

Implementation constraints	
System memory requirement	8GB
System response time	2s
Database	MYSQL1.0
Internet connection protocol	HTTP2.0
Front end language	PHP
Back end language	Java

2.7 User Documentation

The first user instruction is the account management instruction, which guides the users to login to the system and log out of the system, check users' validation status, premium status. The second one is the car parking status management instruction, which shows users how to make reservations of parking, check cars' security and available spots. The third one is the payment instruction, which allows users to pay in cash following the instructions, or pay online.

2.8 Business Requirements

1. the product should be delivered in two months after the project begins.

2. the product should be established as a stable and viable technology.

3. user information should be stored properly and should not be used for any purpose other than for providing better services.

2.9 Assumptions and Dependencies

1. users are able to use a web browser to perform basic operations.
2. every component is implemented and tested according to the timeline.
3. every RFID hardware works well with software, from sensor to GPS.

Dependencies	
Server	Amazon Cloud
Internet access	RFID. LTD
RFID sensor & GPS	RFID. LTD
online payment support	PayPal

2 Specific Requirements

2.1 External Interface Requirements

2.1.1 User Interfaces

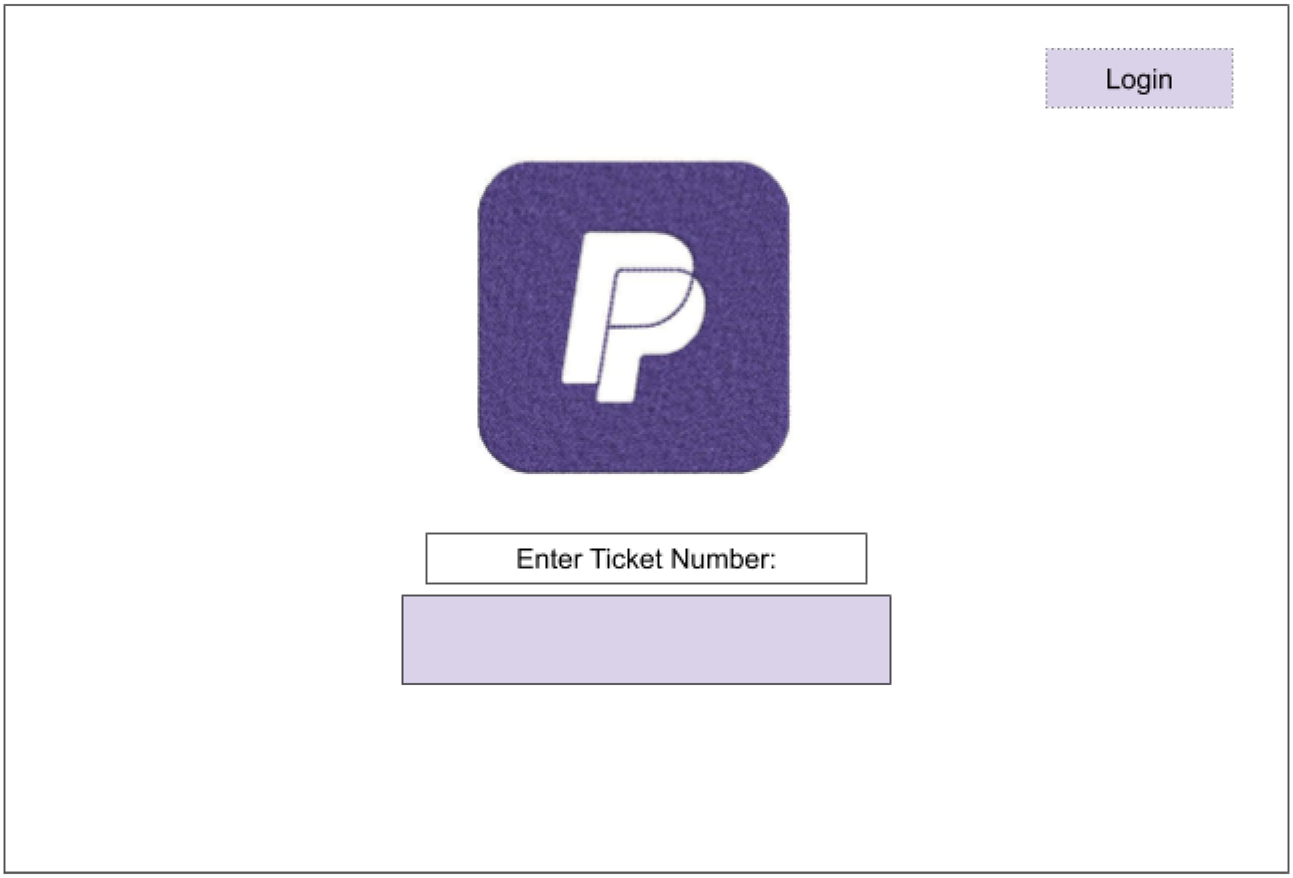


Fig 1.1: Splash page.

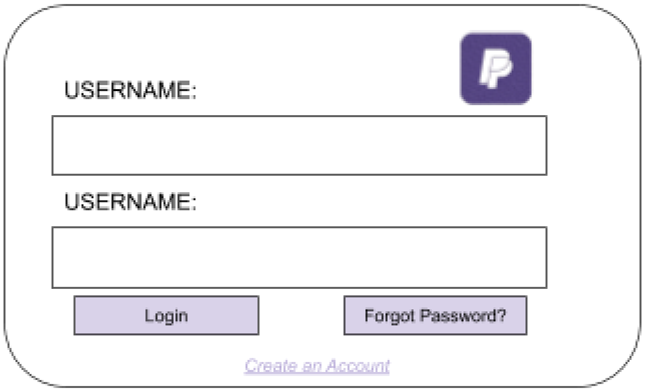


Fig 1.2: Login page

This is a splash page the user will be first prompted with once they get to the client web service. This will ask them to enter the ticket number if they are not a member or they will be able to log into the system if they are an existing user. Following this they will get to the *home* page of the system that give the users all the information that they will need about the system.

Above in Fig 3.3 there is a basic diagram of what the user interface will look like. This will give the user a lot of information. Starting from the top the the interface will display the ticket number that the user has entered. Since they do not need to have an account with our system in order to access this information there will be some information that is applicable the non-member users, also known as free user, and there will be other information that is not available.

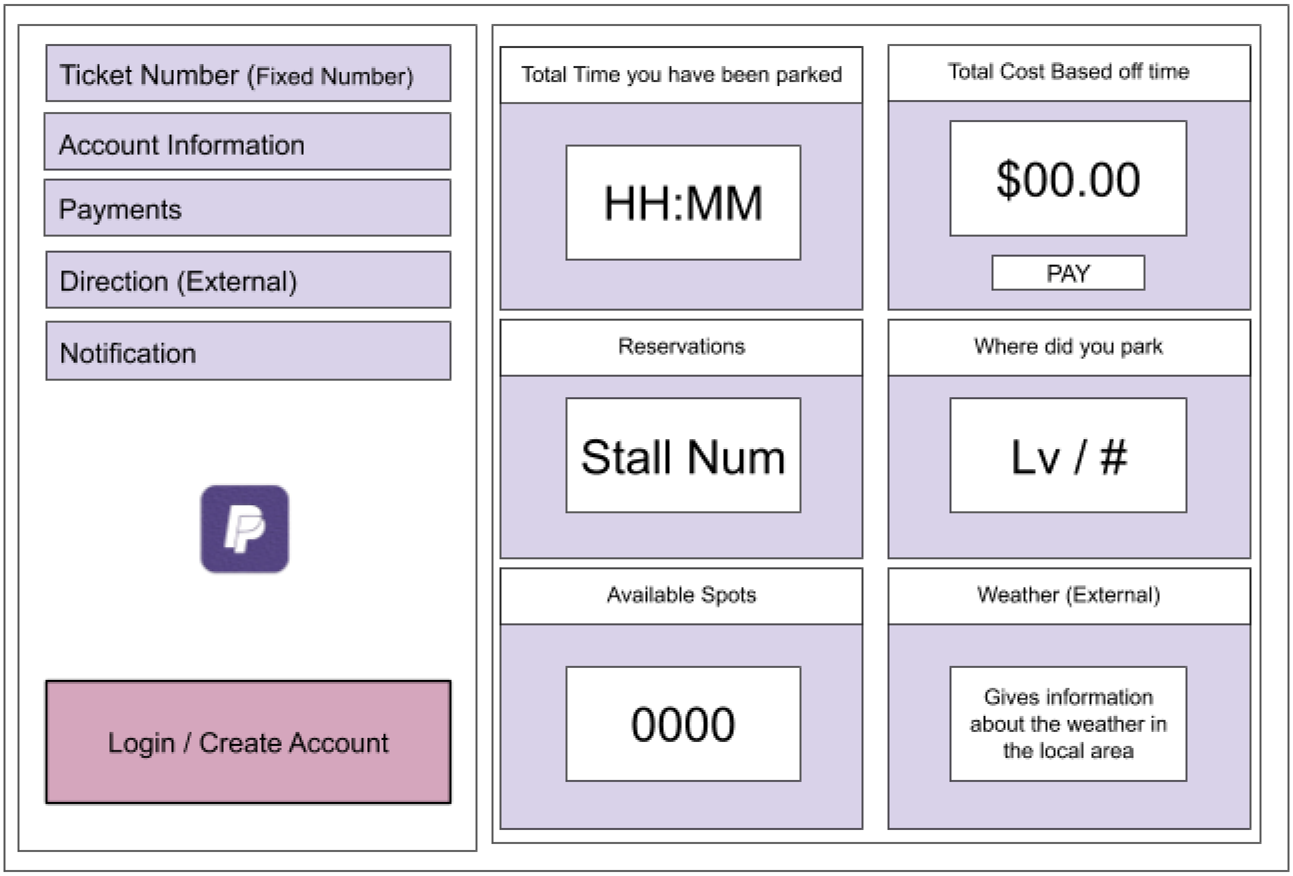


Fig. 1.3: Parking Pal home page.

Free users will have the basic information which would be the total time they have been parked in the stall, how much their bill will cost them, access to using the 3rd party plug-ins, and the location of where they parked. All the other information will be *unclickable* until they either make an account and/or become a standard or premium user. Another type of free user will be someone who has an account but is not paying for the monthly service. This individual will have access to all the features above as well as being able to manage account settings and receive notifications via email, text or both.

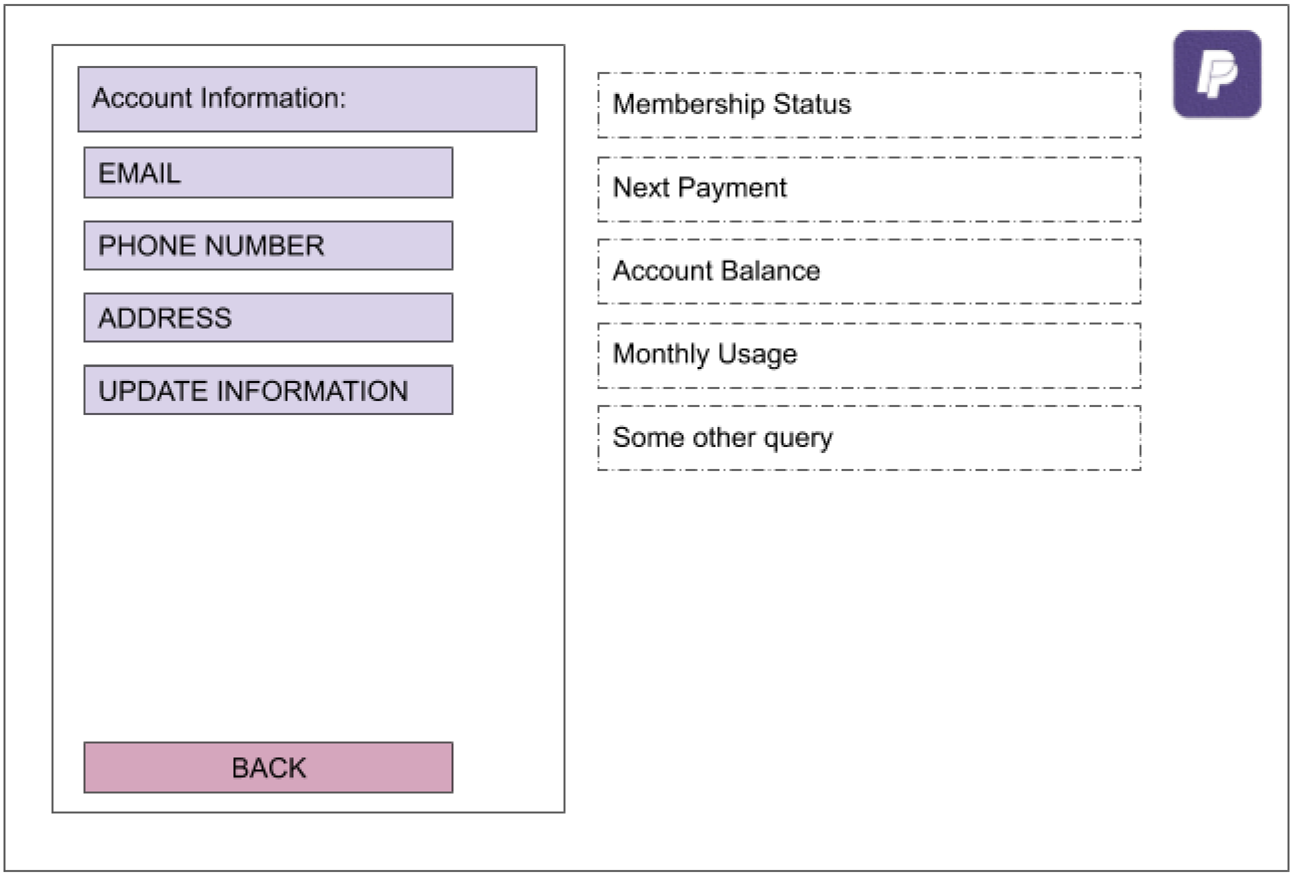


Fig 1.4: Account information page

The next tier of user will be the standard user. This user is someone who is paying for our service. This will allow the user to view how many spots are available, as well as paying a flat rate for using the garage and all the features listed above. They will also receive a key card so that are able to simply *scan* in and out of the garage without dealing with the tickets. The last type of user is a premium user. This will allow the user to reserve spots and get all the features that were available prior.

2.1.2 Hardware Interfaces

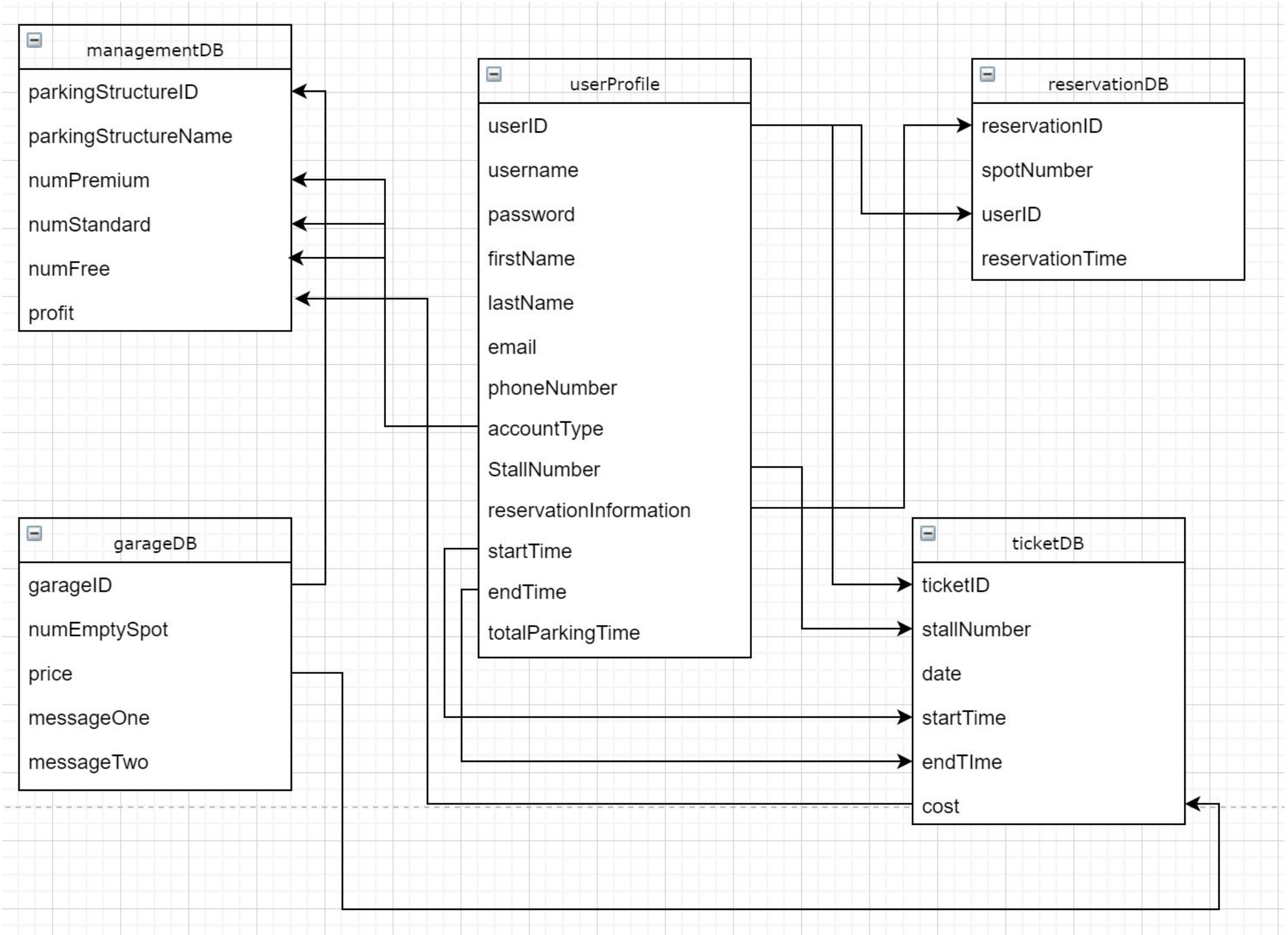
List of hardware devices:

Entrance RFID Scanner
Ticket Printer
Exit RFID Scanner
Ticker Taker
Kiosk
Key Cards
RFID Tickets

At the entrance there will be a scanner and a ticket printer. The scanner will allow for users to scan their key cards. The printer will print an RFID ticket if the user does not have a key card. At the exit there will be another scanner for key cards. As for the tickets there will be a ticket taker box that will validate if the ticket has been paid. The physical key card is an item that will be mailed to the individual when they subscribe for monthly parking. The tickets are printed every time someone enters the garage without a key card. The kiosk is a device that the user can pay for their parking via cash.

2.1.3 Software Interfaces

Since this is a web based application this package will be available via any computer, smartphone or tablet that is connected to the internet with access to the Parking Pals server. When the user enters a ticket number this number will be checked against a database to see if the ticket is in the database. If the ticket is not in the database, then the user will be prompted that the ticket number was not found.



This database illustrated above shows that everything is linked to a ticket number. The system will implement a REST back with the use of MySQL to help with the database management.

There will be external information coming in from out 2 additional web service that will return local weather as well as traffic information. The traffic web plug-in will also default by showing the user how to get from their current location to the location of the garage.

Data entering the server:

- time user enters garage
- time user parks
- time user leaves parking spot
- time user leave garage

Data Leaving the server:

- Elapsed time user has been parked
- Accumulated cost based off elapsed time

Data from outside services:

- Weather app
- Traffic app

List of services:

- Garage Notifications
    - Emergencies
    - Security
    - General Updates
- Total Time
- Total Cost
- Spot Recall (Remembering where you parked)
- Spot Reservations
    - Spot assignment
- Weather Report
- Traffic Report
    - Directions to the garage
- Spot Availability
- Managing Account Settings
    - Subscription manager
    - Online Payment

2.1.4 Communications Interfaces

There will be either text or email notification for the user. The users will be given a notification that will be sent from the server based off certain incidents that have occurred.

Notifications:

- linked to the maps
- linked to the Parking Garage Updates
    - theft
    - emergencies

2.2 Functional Requirements

ID: FR 1:
Feature: RFID Scanners
Description: The user will swipe their key card or get a ticket
Scenarios:

- The user has a key card that they scan at the entrance and they system check for authentication. Once the card has been authenticated the user will be able to enter the garage
  - This would receive user ID form the key card and compare it against what is in the database.
- The user does not have a key card and the system prints out a ticket. This ticket will be added to the system and the time stamp will be taken.
  - Make sure that the ticket number is a valid ticket in the database.

ID: FR 2:
Feature: SQL and POS
Description: The user is ready to leave and need to pay.
Scenarios:

- Free user, i.e. someone who has a ticket. They will need to pay at the kiosk, i.e. in cash, or they will be able to pay using the web service. They will type in their ticket number and they system will provide them with the needed information about their parking event. There will be a payment tab that allows them to pay.
  - This would take the ticket number and retrieve from the database how long the user has been parked for and based off how long they have been parked return a value for how much they owe.
- A standard or premium user who has a key card. They would simply go to the exit and scan their key card. Once the systems authenticates they will be let out.
  - This would get the user ID and then check to make sure they user has valid credentials and then would allow them to exit the garage, aka open the gate.

ID: FR 3:
Feature: MySQL writing data to a database
Description: Creating an Account
Scenarios:

- Once the user enters the ticket number, and does not have an account, they will have an option to create an account. If they do not have an account then the system will ask them to fill out a field of information
  - This would check the make sure the ticket number they have entered is a valid ticket number, then would create pull all the information that the user has entered into the user text files, after making sure all the data is in the correct format would then commit the data to the database.
- If a user has an account they will be able to log in and there will be no need for them to make an account.

    - This should not happen since the ticket number is linked to the account, so if someone tried to make a new account with an existing ticket number they would then be told that this ticket already exists in the system.

ID: FR 4:
Feature: SQL query
Description: Reserving a parking spot
Scenarios:

- A premium user reserves a parking spot and they arrive at the garage and are able to park in that parking spot
    - Would check the database and make sure that the user is a premium member with valid ticket number. Once this is done it would assign them a unique parking stall number.
- A premium user reserves a parking spot and they are unable to park in the reserved parking spot because someone else is parked there. They system will automatically reassign them into a new parking spot.
    - This would automatically reassign the premium user a new unique parking stall that is not taken.
    - If there is no stall available then the premium user would be refunded their reservation fee and the user who parked in the wrong stall would be notified and fined.

ID: FR 5:
Feature: SQL query
Description: Checking available spots
Scenarios:

- The user logins into the web service to check how many parking spots are available.
    - This would make a call to the Ticket Database and identify how many parking spots are available.

ID: FR 6:
Feature: User Authentication
Description: Checking the status of the users account
Scenarios:

- The user is a free user so they will enter a ticket number in order to get past the splash page. Once they are at the home page of the service they will then be able to create an account. Once the account has been created then they will be able to check the status.
    - The ticket number will be checked against the Ticket Database. If there is an invalid ticket then the user will be told they have entered a bad ticket number. If it is a valid ticket number then they will be directed to the home page.
- The user is a free user who already has an account, a standard user, or a premium user. The user will login to the account click on the Account Management button. They will be able to view the status of their account.
    - For a user who would like to log into their account they will click on the login button and this will redirect them to the login page. Once here they will be able to log into their account. Refer to FR7 for details about logging in. once valid information if produced then the user will be directed to the Parking Pals home page.

ID: FR 7:

Feature: User Authentication
Description: User logging into Parking Pals
Scenarios:

- A user enters a valid username but an invalid password
    - The system will let them know their password is invalid
- A user enters a invalid username
    - The system will let them know their username is not valid.

ID: FR 8:
Feature: SQL Query
Description: Checking a ticket number that is linked to an account
Scenarios:

- If a user tries to enter a ticket number that is linked to a paid account they will be prompted to log into Parking Pals.
- If the user enters a ticket number that is not is the database they will be informed that they ticket is not valid
- If the user enters a valid ticket number then they will be redirected to the home page.

3 Other Non-functional Requirements

3.1 Performance Requirements

The scanning in and out of the garage is the most time sensitive part in our system. This is because we want to get as many people in or out of the garage as possible. We would like the users to have a response from the system within no more than 0.5 seconds whether or not their key card is valid or not. If the user has to wait more than this there is a problem in our system and will have the user rescan their card and see what is going on.

Other services that are available in our system are based on the quality of user experience. limiting all of these to being no more than a 1 seconds wait time. Id the user is able to visually see that they system has different portions of the service loading at different times then this is something that will decrease the user's experience and limiting this would be the first priority.

3.2 Safety and Security Requirements

Our system is only storing basic user information such as addresses, phone numbers, and emails. To help protect this information we will be able to use an encryption on the users passwords when we store them in our database. The only payment information that we will be storing is that of how much their bill has accrued over time. All financial transactions will be handled by a third party company.

3.3 Software Quality Attributes

This system should be available 24/7. Even if there are the hours when the garage is closed they system should still be available. Any user should be allowed to create, remove, edit, check the status of the garage, or reserve a parking spot at any point in time.

4 Other Requirements

Appendix A – Glossary