Urban Shared Parking Management System

Software Requirements Specification

Version 1.0

Dec. 5 , 2018

Qing Xu

Student Number:2016211949

Requirements Analysis Engineer

Introduction to Software Engineering
Fall 2018

Revision History

Date	Description	Author	Comments
5/12/2018	Version 1.0	Qing Xu	First Revision

Document Approval

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

Signature	Printed Name	Title	Date

Table of Contents

目录

1. Introduction

This section gives a brief introduction to the USPMS and the SRS itself, including the purpose of this document, scope of the product, a table of all the definitions, acronyms and abbreviations used in this SRS, references to all the documents that are referenced in this SRS, and a general overview of the overall organization of this document.

1.1 Purpose

This document is intended for analyzing and describing the requirements of the Urban Shared Parking Management System. It is written for all the stakeholders of the system to both design and implement the USPMS and also make use of the system.

1.2 Scope

The name of the software product is Urban Shared Parking Management System.The USPMS is a system designed to solve problems of parking difficultly and uneven parking spaces.The system can help drivers finding parking spaces quickly and help administrators to collect parking spaces data real-timely.More specifically,the USPMS provides different functions and features for different types of users.Users of driver type can log in to the USPMS to inquire empty parking spaces in residential area,in university campus,in the institution,in pubic parking,in access road parking around the destination real-timely and book empty parking space ahead of time.Users of administrator type can log in to the USPMS to collect information of empty parking space occupancy real-timely.The system will charge driver who confirms his booking of parking space according to his parking time.

1.3 Definitions, Acronyms, and Abbreviations

This subsection shows all definitions,acronyms and abbreviations from the SRS.

Table 1: Definitions, Acronyms, and Abbreviations

Term	**Definition**
USPMS	Urban Shared Parking Management System
PA	Parking lot’s Administrator
SA	System Administrator
PMS	Parking Management System
SRS	Software Requirements Specification

Drivers	Users who have registered on the USPMS to book empty parking spaces and are ordinary users.
PS	Parking Spaces
SCR	Specification Change Request. A request to make changes to the SRS to correct errors or change requirements.
RCR	Requirement Change Request. A request to change, add or remove one or more requirements.

1.4 References

[1] IEEE Software Engineering Standards Committee. IEEE SA 830-1998, IEEE Recommended Practice for Software Requirements Specifications. October 20, 1998.

[2] 王安生. 软件工程化. 北京: 清华大学出版社, 2014.

1.5 Overview

The first section of this SRS gives a brief introduction to the USPMS and the SRS itself, including the purpose of this document, scope of the product, a table of all the definitions, acronyms and abbreviations used in this SRS, references to all the documents that are referenced in this SRS, and a general overview of the overall organization of this document.

The next section covers the general description of the USPMS, with a product perspective, functions of the product, analysis on the characteristics of the users, constraints, assumptions and dependencies of the USPMS product.

After that third and the fourth sections describe the specific and modeling requirements of the USPMS , both functional and non-functional. These two sections both analyze the requirements from multiple aspects.

The fifth section talks about the process of updating the system when the scope or requirements change. And the last section covers more additional and helpful information for the SRS that can be referred to when needed.

2. General Description

This section gives an overview of the USPMS. The system will be explained in its context to display how the USPMS interacts with the other systems. Types of users for the EGAS and the different functions for each type of users will be described. At last, the constraints and assumptions for the USPMS will be presented.

2.1 Product Perspective

The system aims to manage scattered parking spaces in public parking and access road parking and promote open empty parking spaces in residential area,in university campus and in the institution along with function of managing information at the same time. The system can help drivers find empty parking space quickly when they can't find place to park. The USPMS work along with WeChat, Alipay,QQ,Baidu map or Gaode map and all parking lots' management systems. What's more,the product provides different functions and permissions for different types of users. A PA can collect information and data of their own parking lot real-timely; a driver can inquire empty parking space and book the parking space ahead of no more than 30 minutes; a SA can check whether a user who apply for permission of a PA is a real administrator.

2.2 Product Functions

The following are main functions of the system:
First, the system can help drivers inquire status of parking spaces around the drivers' destination real-timely;
Second,the product can guide drivers to the closest empty parking space quickly once drivers book a parking space;
Third,the system will charge drivers who use the system to find parking space once they confirm their book;
Fourth,the product also can provide functions of logging in and logging out and manage users' information including their user-type and password,and users can log in with their WeChat account , QQ account or Alipay account;
Fifth,users can pay with Alipay or WeChat;
Last,users of different types have different permissions.For example,an ordinary user can't decide the standard of price.

2.3 User Characteristics

The software has three types of users which are drivers,PA and SA.
i)Only SA can give users PA's permission;
ii) Users' information including account name and telephone number can't be seen by other users except a SA;
iii) A PA can collect data about the parking lot they manage including how much the system earns through their parking lot a day and when the number of parking lot parked is largest in a day.Moreover,a PA can modify basic information of their

parking lot.For example,if a parking lot is rebuilt,the PA can modify current state of the parking lot.
iv) Drivers can book empty parking space no more than 30 minutes in advance and should pay after acknowledgement of his book.
v) SA should record the number of a driver escaping the order and cancel the order.
vi) Software operation and maintenance persons should be proficient in computer and know plenty of solutions to most computer problems.

2.4 General Constraints

i)Legal and policy constraints: When developing this software,it must be implemented in strict accordance with relevant laws and policies.
ii) Interfaces:The system will have interfaces to QQ,WeChat to manage users' account,to WeChat,Alipay to charge drivers,to Baidu Map or Gaode Map to guide drivers to empty parking space which they have booked in advance.
iii) The system is designed for mobile phone.In addition,the software should be able to be applicable on both Apple and Android systems.
iv) The system should also interact with all parking lots' management system which have registered on the system.
v) If there two or more drivers want to book the same parking space at the same time,the driver who has lowest escaping orders times or lowest canceling orders times will book successfully.
vi) A PA can only inquire his parking lot's data; that is to say,he can't get other parking lots' data.
vii) Once a driver book a parking space successfully,the parking space is occupied for other drivers whatever the parking space is occupied really.
viii) The system's UI should be suitable for all different layouts in different kinds of phones.

2.5 Assumptions and Dependencies

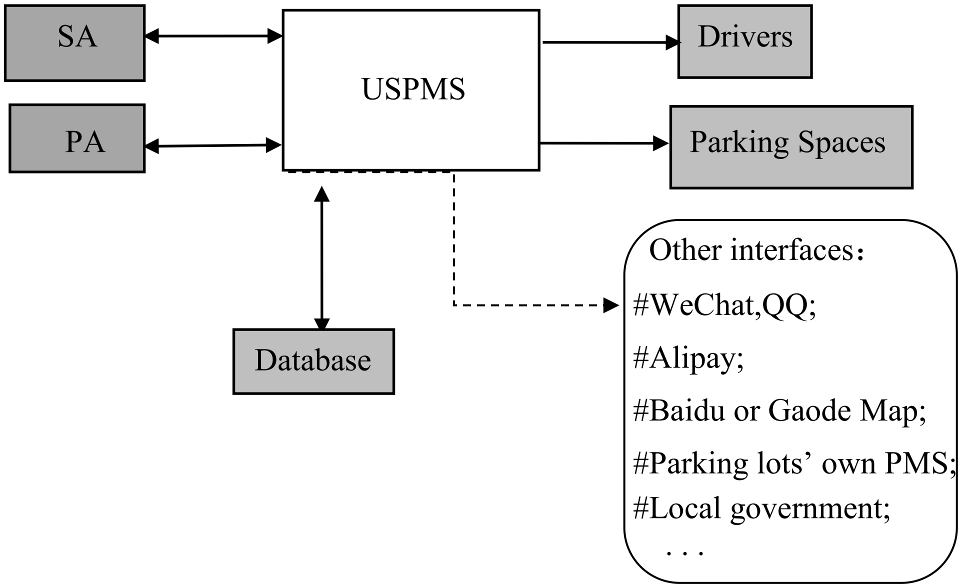
This subsection shows factors that affect the requirements.
i)It it assumed that the scope of the parking spaces on the system is in our city.
ii) It is assumed that the most concurrency is no more than 10000;that is to say,no more than 10000 drivers can book parking space simultaneously.

3. Functional Requirements

3.1 Context Analysis

This section contains the functional and quality requirements of the USPMS, with a more accurate, clearer, more complete and unambiguous description of the features provided by the system.

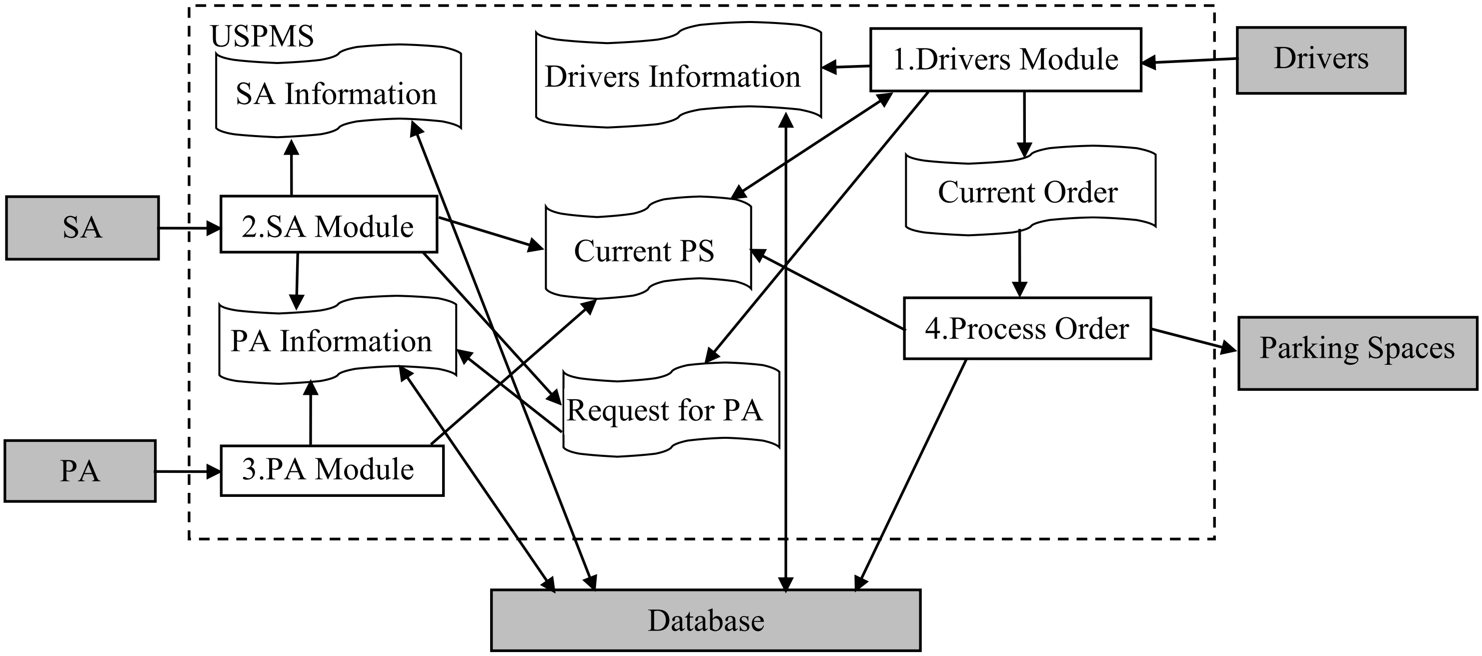
3.1.1 Context Diagram



Graph3-1 Context Diagram of the USPMS

Before describe the functional requirements of USPMS,it is necessary to define the external border of the USPMS. Graph 3-1 displays the interaction between USPMS and external environment and interfaces. SA,PA and Drivers are three kinds of users with different permission. Other interfaces mean that USPMS need to work with other software together or USPMS can't achieve its functions.

3.1.2 Data flow level 1

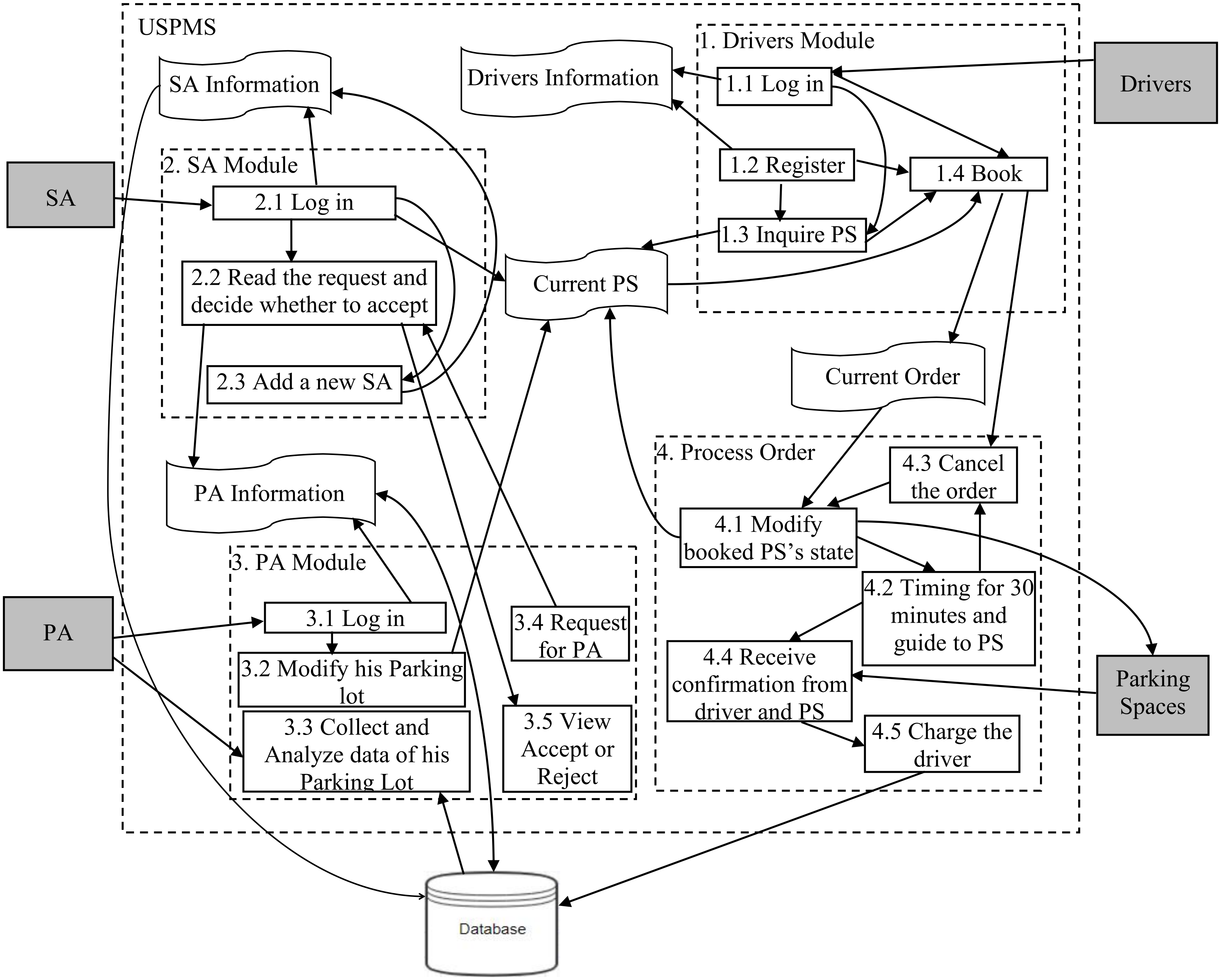


Graph 3-2 Level 1 Data Flow Diagram for the USPMS

Graph 3-2 displays the basic internal functions and the flow of data of the USPMS. The external entities are the same as the context diagram of the USPMS.

The internal functions of the USPMS are concluded to three modules and one process. Three modules include Drivers Module,SA Module and PA Module. Driver module includes what a driver can do on the USPMS. SA module consists of a SA's functions and his permission. PA module is made up of a PA's actions and his permission. The Process Order process describes what the system will do after drivers have booked a PS.

3.1.3 Data flow level 2



Graph 3-3 Level 2 Data Flow Diagram for the USPMS

Graph 3-3 shows the detailed internal functions and the flow of data of the USPMS. The internal functions of level 1 are elaborated in the level 2, with more detailed sub-functions added to each of the functions and modules in level 1.

In Drivers Module, drivers that is also called ordinary users should log in or register first before they begin to enter the USPMS. They can use WeChat or QQ account to log in because the system also has interfaces to WeChat and QQ. They can inquire current state of parking spaces around their destination.After that, drivers can book the empty parking spaces they have found. Then, switch to Process Order.

In Process Order, first of all, the state of the booked parking space should be modified from empty to occupied for other drivers. Next, the system will start a timer for 30 minutes to judge whether the driver gets to the booked parking space in no more than 30 minutes. The Baidu Map or Gaode Map will guide the driver to the parking space. If the driver arrives within the specified time, the system will receive confirmations from the driver and the perception of the parking space. After that, the system will charge the driver for providing information of empty parking spaces. The driver can pay with Alipay. If time out, the order will be canceled and the state of the parking space will be back to empty. Whether the order is completed or not, it will be written to database including the driver’s id and the parking lot the booked parking space belongs to and the time when the order is created. If the driver cancels the order within 30 minutes, the order will also be recorded with note ‘canceled’. If the driver don’t pay, the order will recorded with note ‘escape order’. If time out, the order will be written with note ‘time out’.

In PA Module, logging in is required at first. When a PA have logged in, the system will check if he has PAs’ permission in PAs’ table in database. If yes, the PA has all functions drivers have. Apart from this, the PA also can collect data and analyze data of his parking lot. For instance, a PA can know the traffic flow in the parking lot that he can discuss with a SA or the boss of the system how much they can get as dividend. Moreover, if a user wants to become a PA, he can send a request to SA. SA will check whether the request is reasonable and real and send the result back to the applicant and the reason if the request is refused. That is to say, the applicant can view the result and the reason of his request after some time.

If the request is accepted, the information including which parking lot the applicant manages submitted by the applicant when he sends a request will be saved in PAs' table in database. The PA can modify his parking lot's state according to actual situation. For example, the PA can stop the parking lot if the parking lot is being rebuilt.

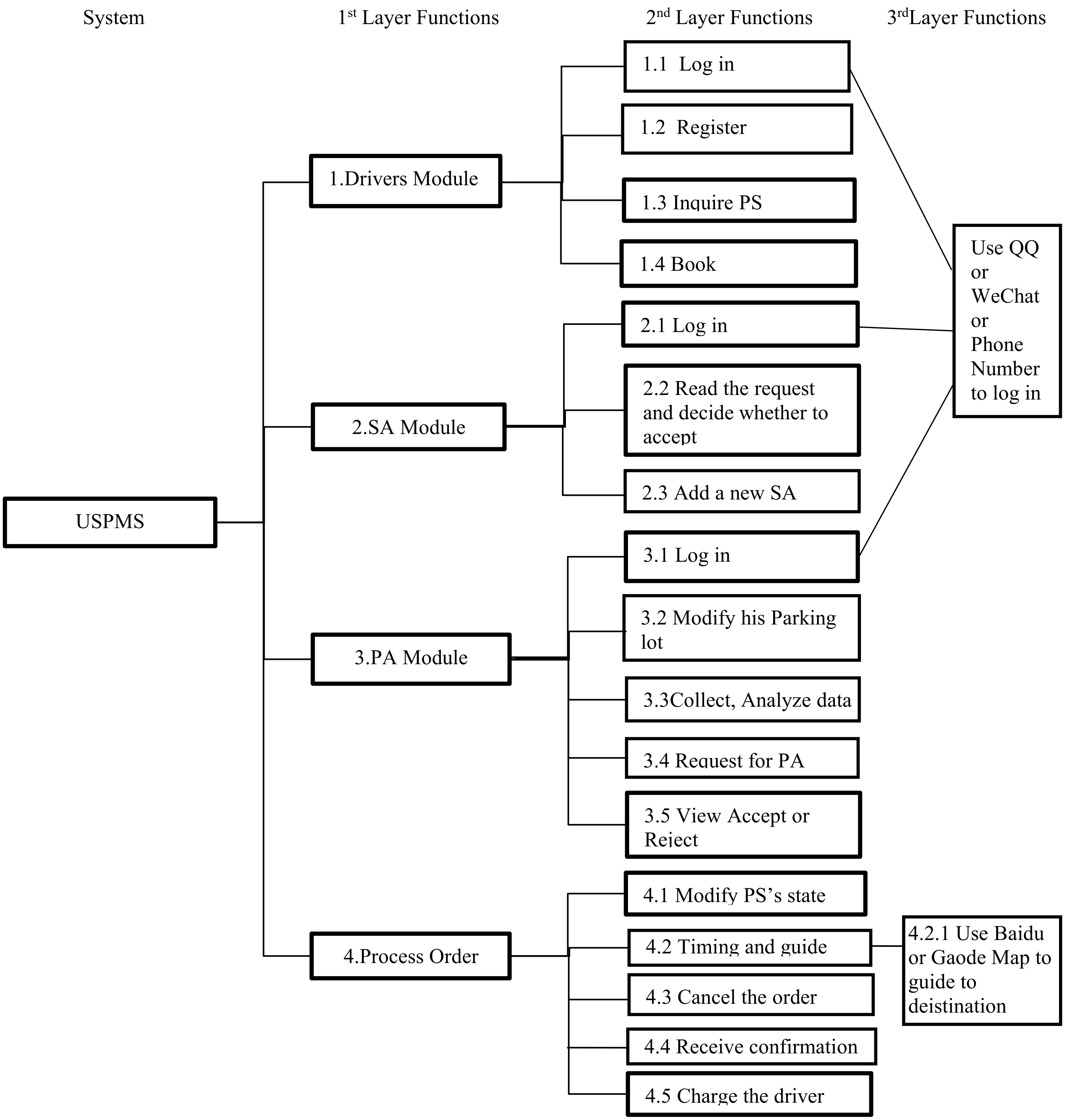
In SA Module, logging in is also the first step. If logging in successfully, the SA also has all functions drivers have like a PA. In addition, a SA can read requests for PA's permission and decide to accept or not after checking whether the applicant is the real administrator of the parking lot. What's more, the SA can add a new SA and then the information of the new SA will be written to SAs' table in database.

3.2 Functional Requirements

The section will describe the detail of all functional requirements.

3.2.1 Functional Structure

The functional structure diagram will display the functional scale of the whole system clearly.Combining this functional structure diagram with the data flow diagram above can provide an intuitive and clear understanding of the functional requirements of the system.



Graph 3-4 Functional Structure of the USPMS

3.2.2 Functional Requirement or Feature #1:Register to the USPMS

3.2.1.1 Introduction

Before a new user wants to use the USPMS, he must register to the system. When the user choose to register, the register page will display two types of users that the user can choose to register----Driver and PA. If the user registers as a driver, he should submit some necessary and basic information; and then the system will deal with the register-request. If the result is correct, the user registers successfully and the information about the user will be saved in Users' table in database. If the user wants to register as a PA, he will submit his phone number, his parking lot's number, his identity number and his password. Then the system will deal with the request for a PA from the user. If the request is approved, the user has become a PA and his information will be written to PAs' table in database.

3.2.1.2 Sub-Functional requirement#1:Register as Drivers

3.2.1.2.1 Inputs

The driver's license number and basic information of the new user should be stored inside the browser's cookies for the USPMS to read as inputs.

3.2.1.2.2 Processing

The system should check if it is a real and legal driver's license number. If yes, the USPMS should check whether the driver's license number has registered; if not, all information of the new user should be saved in Drivers' table in database.Otherwise, the system should return the failed result and the reason.

3.2.1.2.3 Outputs

The page with message of successfully registered will be displayed and the website homepage should be shown if the result is right.

3.2.1.2.4 Error Handling

If the driver's license number is nonexistent, the system should prompt the user that the driver's license number submitted is nonexistent or out-of-date and redirect the browser to the register page .

If the driver's license has registered, the system should prompt the user that your license number has registered and you can log in directly and redirect to the login page.

3.2.1.3 Sub-Functional requirement#1:Register as PAs

3.2.1.3.1 Inputs
The parking lot's number and basic information of the new user should be stored inside the browser's cookies for the USPMS to read as inputs.

3.2.1.2.2 Processing
The system should send the request to a SA, and the SA should check if the parking lot's number is real and if real supply a validation between the administrator's identity and the identity submitted by the user. If true, all information of the new user should be saved in PAs' table in database. The SA should return the validation result to the applicant.

3.2.1.2.3 Outputs
The new PA's parking lot's information page should be shown if the validation is true.

3.2.1.2.4 Error Handling
If the validation is wrong, the USPMS should alert the applicant the wrong information provided and ask the user whether to refill the application form or not.

3.2.3 Functional Requirement or Feature #2: Log In to the USPMS

3.2.3.1 Introduction
A user should be able to log in to the USPMS through a log-in page before entering the system. The USPMS shall store the information of the logged-in user into the cookies of his/her browser. The USPMS should be able to check whether a user has logged in and read his/her information out of the cookies. The USPMS must validate whether the information is provided by the database. Especially, the user can log in using QQ or WeChat account. In that case, the USPMS should make use of the interface to QQ or WeChat to process the problem of authorizing.

3.2.3.2 Inputs
The account number and basic information of the logged-in user should be stored inside the browser's cookies for the EGAS to read as inputs.

3.2.3.3 Processing
The USPMS should check whether there is a user logged in by a validation value in the cookies, get his/her user number, and validate the login state of the user by comparing the login token values from both the cookies and the server database. If

the validation is ensured, the USPMS should continue to read the user's basic information and use it to fetch data from the server.

3.2.3.4 Outputs
The homepage should be displayed if the login state has been validated.

3.2.3.5 Error Handling
If there is no login information in the cookies, the USPMS should redirect the browser to the login page.
If the login information is fake or out-of-date, the USPMS should redirect the browser to the login page.

3.2.4 Functional Requirement or Feature #3: Inquire State of PS

3.2.4.1 Introduction
The USPMS allow users to view the current state of all parking spaces registered on the system ------ empty or occupied everywhere and all the time. The user can choose a destination, and the USPMS shall return the state of parking spaces around the destination.

3.2.4.2 Inputs
The users should click on the "Overview" link on the home page of the USPMS. The destination users type in and users' number and the login token value shall be sent to the server as the inputs.

3.2.4.3 Processing
The USPMS should check the validation of the login state of the user using the user number and login token from the inputs. If the login state is valid, the USPMS should search for the destination sent and find all parking spaces within 5 kilometers distance. Then the USPMS should display the state of all parking spaces picked up, and occupied state is represented by red while empty state is represented by blue. The server shall send the information including parking spaces' state and location back to the browser.

3.2.4.4 Outputs
The parking spaces' icon with different color should be display on the page, which imply the state of parking spaces.

3.2.4.5 Error Handling

If the login state is invalid, the USPMS should alert the driver of the exceptional login state and guide the driver to the login portal page.
If the destination submitted is out of range, the information of out of range shall be informed to the user and remind the user to choose a new destination again.

3.2.5 Functional Requirement or Feature #4: Book a PS

3.2.5.1 Introduction
The users of the USPMS can book a PS ahead of time but no more than 30 minutes. Inquiring states of parking spaces is a previous and necessary step to book a PS. After viewing PSs around the destination, users can choose an empty parking space which is the most suitable one to book. Users should choose to book and choose a wanted one. Then if the book is successful, the interface between the USPMS and Baidu or Gaode Map will guide users to the booked parking spaces.

3.2.5.2 Inputs
After viewing all PSs around, users should click on “Book” link on the State Of PS page and choose an empty parking space to book. Then the user’s number and the parking space’s information including its location and the parking lot’s number of which it belongs to and the login token value should be sent to server as the inputs.

3.2.5.3 Processing
The USPMS should check the validation of the login state of the user using the user number and login token from the inputs. If the login state is valid, the server shall modify the state of the booked one to all other users. That is to say, for other users, the booked parking space is occupied. And the USPMS shall go to the interface that connects Gaode or Baidu Map to present the best route from the current location to the booked parking space for the user. In addition, the user’s number and parking lot’s number and time to make the book and some other necessary message should be written to Books’ table in database. What’s more, the USPMS should start a timer for 30 minutes to check whether the user is late or not.

3.2.5.4 Outputs
The page with the best route displayed shall be shown to guide users to the booked parking space. And the page also contains how long the user remain to get to the parking space.

3.2.5.5 Abnormal Situation Handling

If the login state is invalid, the USPMS should alert the driver of the exceptional login state and guide the driver to the login portal page.
If the user cancels the order halfway, the request for ending the current order should be sent to server and the times to cancel orders should be added 1.
If the time estimated is more than 30 minutes, the longest time can be extended to 45 minutes most. If the time estimated is more than 45 minutes, the longest time can't be extended more.

3.2.6 Functional Requirement or Feature #5: Time Out of a Book

3.2.6.1 Introduction
When the user has booked an empty PS, if the user don't get to the booked PS within the given time, the timer set up for the user shall time out. Then the order should be canceled automatically and inform the user his order has been canceled due to timing out. Then the order should be saved in Books' table in database. Moreover, the booked parking space should be modified to empty and all changes should be written back to database.

3.2.6.2 Inputs
The remind for timing out and current book's number and user's number and the login token value shall be sent to server.

3.2.6.3 Processing
The USPMS should check the validation of the login state of the user using the user number and login token from the inputs. The server should be informed of timing out of the current order. After that, the order should be modified and written back to the database. Then the USPMS should inform the user he don't get to the destination within the limited time. In addition, server should modify the state of the booked one from occupied to empty and return states of all parking spaces to browser.

3.2.6.4 Outputs
The prompt about the timing out order should be shown on the time-out-order page to the user. As for others, they can view one more empty parking space.

3.2.6.5 Error Handling
If the user close the USPMS abnormally, the order should wait until timing out.

3.2.7 Functional Requirement or Feature #6: Confirm a Book

3.2.7.1 Introduction

After the user booking an empty parking space, if the user arrives at the booked parking space and click “Confirm Order” , the USPMS should inform server that the order is finished and send all related data to server. Then, the USPMS should charge the user for information fee. And then the USPMS should switch to the PMS according to the parking lot number using interface to every independent parking lot management system.

3.2.7.2 Inputs

The user should click “Confirm The Book” and the confirmation of the order along with the user’s number and book’s number and the login token value should be sent to the server as inputs.

3.2.7.3 Processing

The USPMS should check the validation of the login state of the user using the user number and login token from the inputs. If the user confirms his book, that means the order is successful. Then the USPMS should charge the user for information fee. The user can pay for the fee with WeChat or Alipay. The USPMS should go to the interface to WeChat wallet or Alipay to charge the user. If the user pays for the information fee, the interface should send message to the USPMS to tell the user has paid for the fee. Remaining parking fee is dependent on corresponding PMS. The USPMS should acquire the message about reminding the user his order finishes successfully on the page.

3.2.7.4 Outputs

Information of the user’s result should be displayed in the result page.

3.2.7.5 Error Handling

If the login state is invalid, the USPMS should alert the driver of the exceptional login state and guide the driver to the login portal page.

If the user don’t confirm the book after 10 minutes of his arrival, the USPMS should regard the user’s behavior as escaping the order. Then the USPMS should save the user’s escaping behavior to the database. Whether the user arrives the booked parking space or not can be detected by means of GPS(Global Positioning System).

3.2.8 Functional Requirement or Feature #7: Collect and Analyze data of Parking Lot

3.2.8.1 Introduction

A PA can collect and analyze all the data of his managing parking lot from the database of the USPMS. If a PA choose to overview his data, the USPMS should check if the request is allowed first; then if allowed the USPMS should select corresponding data from database and display them on the page in any way the PA want.What's more, the form of the desired data display can be changed according to the PA's choice, such as pie chart, histogram, line cart and so on. The USPMS should analyze the data with the help of the corresponding PMS if the PA asks.

3.2.8.2 Inputs

The PA logged in can click "Overview Data" on the page. The PA can add one or more filters to the data using the option area on the web page, including last day, lase three days and last week and then can choose the time including at noon, at night and all day. The selected filters, the PA's number and the login token value will be sent to the server as inputs.

3.2.8.3 Processing

The USPMS should check the validation of the login state of the PA using the PA's number and login token from the inputs. If the login state is valid, the USPMS should convert the filters to corresponding query conditions used to fetch data from the database. The USPMS should acquire the desired data and display on the page. Apart from this, the PA can click the form wanted of the data display. The default form is line chart.

3.2.8.4 Outputs

Information including the number of cars parked in the PA's parking lot at different time period and the amount of money the parking lot can earn by the USPMS should be shown on the result page.

If there's no record of the parking lot, the USPMS should inform the PA of that there's no data to display because there is no order for the parking lot.

3.2.8.5 Error Handling

If the login state is invalid, the USPMS should alert the PA of the exceptional login state and guide the PA to the login portal page.

If the user is not a PA, the USPMS should be able to alert the user to the insufficient authority and reject the user's inputs.

3.2.9 Functional Requirement or Feature #8: Modify State of Parking Lot

3.2.9.1 Introduction
A PA can modify the current state of his parking lot according to actual situation. For example, if a parking space is being rebuilt, the corresponding PA should modify the state of the parking space from available to disabled. A disabled parking space can't be booked by users. Moreover, the PA can also modify the whole parking lot's state to closed.

3.2.9.2 Inputs
The PA can choose "Modify Parking Lot" on the home page and click the disabled one on the structural diagram of the parking lot. Then the PA should type in the reason for which the PS is disabled or for which the parking lot is closed and click on the "Confirm" button. The user's number, the information about the disabled one and login token value will be sent to the server as the inputs.

3.2.9.3 Processing
The USPMS should check the validation of the login state of the PA using the PA's number and login token from the inputs. If the login state is valid, the information should be saved in database. The state should last until the PA modify next time.

3.2.9.4 Error Handling
If the login state is invalid, the USPMS should alert the user of the exceptional login state and guide the user to the login portal page.
If the user in not a PA, the USPMS should be able to alert the user to the insufficient authority and reject the user's inputs.

3.2.10 Functional Requirement or Feature #9: Add a new SA

3.2.10.1 Introduction
A SA is the manager of the whole system. A SA can add a new SA to the USPMS. The SA should choose "Add New" to send a request for adding new SA. The request should be sent to other SAs. Only if all other SAs agree the request, the new SA can be added.

3.2.10.2 Inputs
A SA can choose "Add New" on the personal center page. After the click, the SA should type in the account of the new SA and click on the "Confirm" button. The

user's number, the information about the new SA and login token value will be sent to the server as the inputs.

3.2.10.3 Processing

The USPMS should check the validation of the login state of the PA using the PA's number and login token from the inputs. If the login state is valid, the request with all information about the new SA should be sent to all other SAs in the USPMS. The new SA should be accept until all other SAs agree the request. If there is a disagreement, the new SA can't be added and the refusal should be sent to the SA.

3.2.10.4 Results

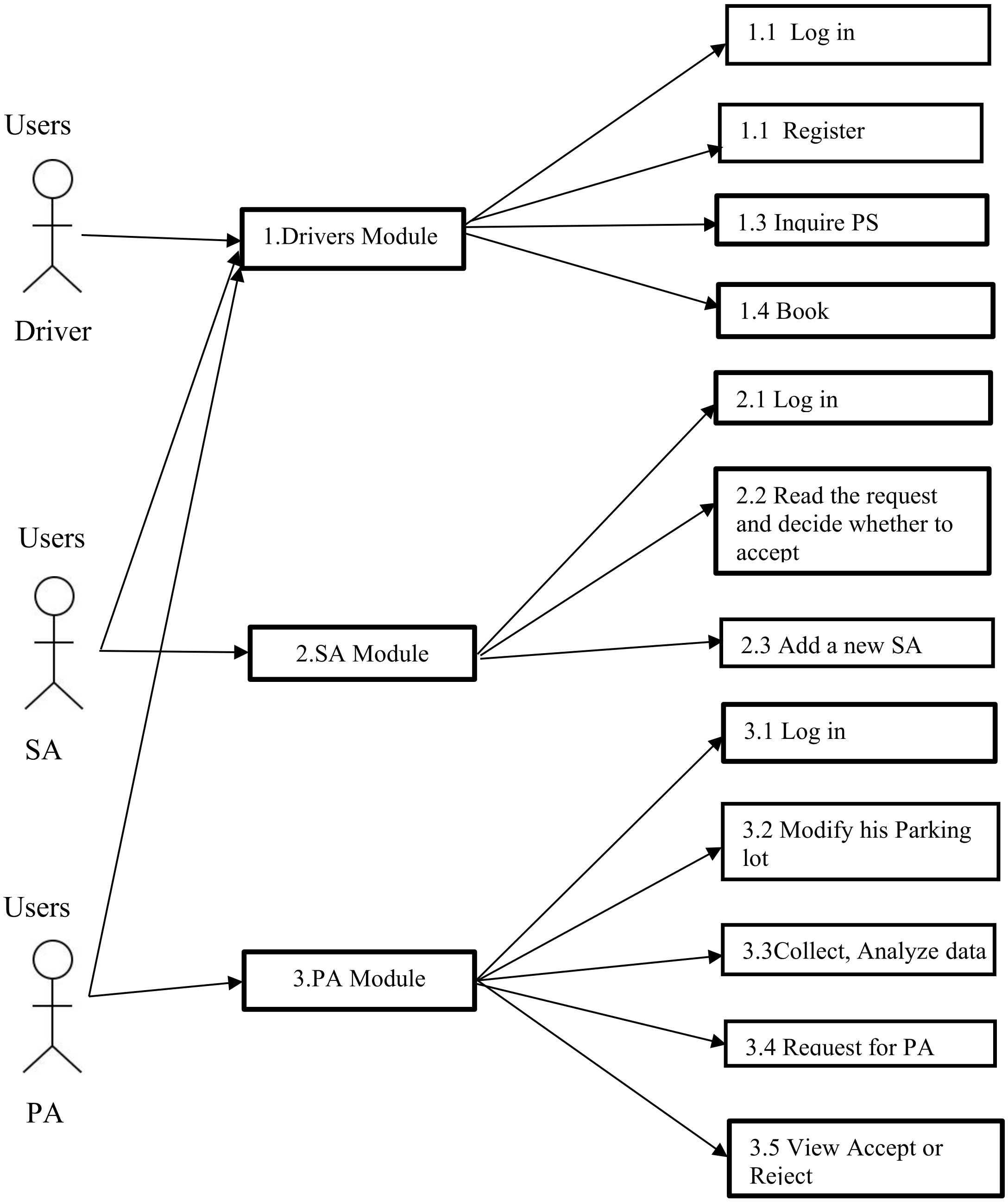
If the request is agreed, the new SA's information should be written to database;
If the request is refused, the information about the refusal should be sent to the SA.

3.2.10.5 Error Handling

If the login state is invalid, the USPMS should alert the user of the exceptional login state and guide the user to the login portal page.
If the user is not a SA, the USPMS should be able to alert the user to the insufficient authority and reject the user's inputs.

3.3 Use Cases

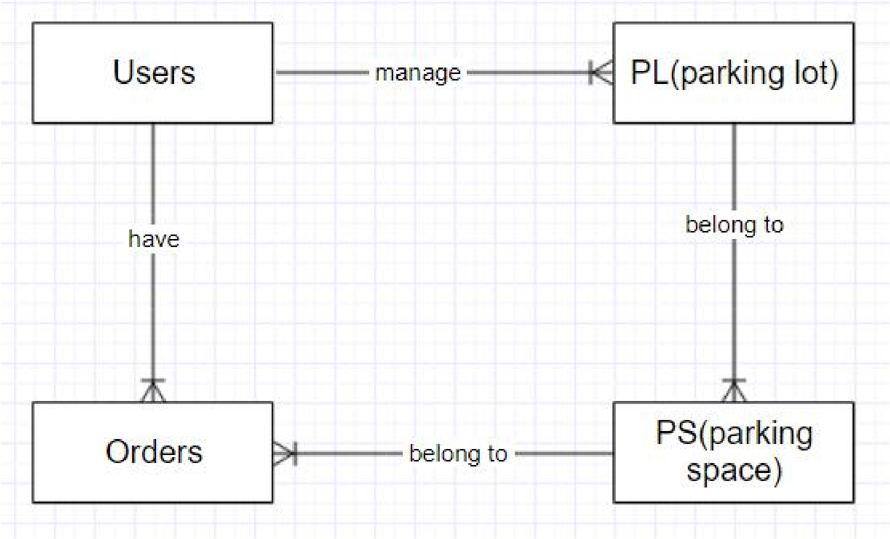
3.3.1 User Role



Graph 3-5 User Cases Diagram between the User Roles and the System Functions of the USPMS

Graph 3-5 displays the User Cases Diagram of the USPMS. The USPMS has three kinds of users ---- SA, PA, Driver. The driver is the main user targeted by the system usually; so driver has all functional features. As for PA, PA should have all driver's functions and his own functions. For SA, SA should have all driver's functions and his own functions.

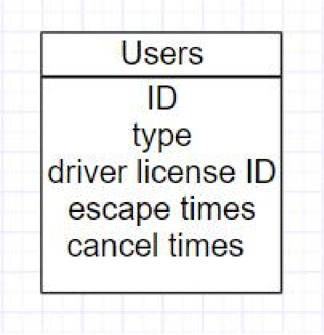
3.4 Classes / Objects



Graph 3-6 Entity-Relationship Diagram

Graph 3-6 is the whole entity-relationship among the USPMS. There are four tables in database and their relationships are displayed. The following are some details.

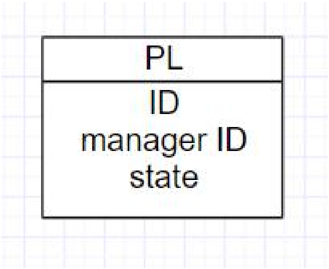
3.4.1 Class / Object #1: Users



There are five attributes in Users Table.
ID represents the identity number of every users;

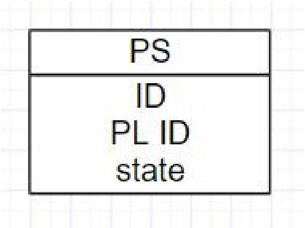
Type represents the type of the user ----Drivers or SA or PA;
Driver license number represents the identity number of every driver, the SA or PA may have this attribute and may not;
Escape times represents the times for the user don't pay for the fee, which should play an import part in making choices when booking PS concurrently;
Cancel times represents the times for the user cancels his order halfway.

3.4.2 Class / Object #2: PL



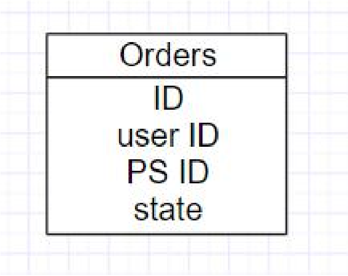
There are three attributes in PL Table. (Parking Lot)
ID represents the identity number of every parking lot;
Manager ID represents the identity number of its manager;
State represents state of current state of the parking lot ----available or closed.

3.4.3 Class / Object #3: PS



There are three attributes in PS Table. (Parking Space)
ID represents the identity number of every parking space;
PL ID represents the identity number of the parking lot which the parking space belongs to;
State represents the current state of the parking space ----occupied or empty or disabled.

3.4.4 Class / Object #4: Orders



There are four attributes in Orders Table.
ID represents the identity number of every order;
User ID represents the identity number of the user who makes the order;
PS ID represents the identity number of the parking space which the user books;
State represents the state of the order ----escaped or canceled or timed out or finished.

4. Non-Functional Requirements

4.1 Quality Requirements

4.1.1 Performance

1. With a probability of higher than 97%, the USPMS can support 8000 users on line or more.
2. With a probability of 100%, the system can support more than 1000 users' simultaneous queries. And the delay shall not exceed 0.5 second.
3. With a probability of higher than 99%, the system can support more than 500 users' simultaneous modifications. And the delay shall not exceed 3 seconds.
4. Less than every 10 minutes, the system shall refresh its data once.
5. The system shall have memories which supports ten-year storage of grades and three-year of log.

4.1.2 Reliability

1. At more than 98% of time, the system works regularly.
2. The system can check its work state once per day, and report the consequence to the SA.
3. When faults occur, the system shall recover within 15 minutes.

4.1.3 Availability

1.The USPMS must be highly available, especially when traffic is congested. So the available rate must be higher than 98%.

2.Friendly user interface.
3. Explicit operation navigation when the new user first uses the software.

4.1.4 Security

1. Backup Mechanism shall be supported.
2. Periodic check shall be supported with an interval of no more than one week.

4.1.5 Maintainability

1. Easy to test when errors occur.
2. Loose coupling. This means that a modification in one block will rarely affect another.

4.1.6 Portability

The system must be available on Android and Apple system.

4.2 Engineering Requirements

4.2.1 Inverse Requirements

The software should be easy to maintenance.

4.2.2 Design Constraints

No specific design constraints.

4.2.3 Logical Database Requirements

The USOMS should use a MySQL database. The details of the tables in database are described in 3.4.

5. Change Management Process

When one or more errors are found in this SRS, changes should be submitted in a Specification Change Request to describe the content to be changed and the reasons for doing so. Analysis and discussion should be make over the SCR to determine whether to approve or to reject it.
If the SCR is approved, changes should be made to the requirements, and several Requirement Change Requests should be written by assigned persons to summarize the detailed information on the requirement change. The RCRs should be reviewed by a review committee consisting of members from all types of stakeholders.

If an RCR is approved by the review committee, the RCR will be used in the procedures of the project to make changes to the designing, coding and testing. If an RCR is rejected, it should be edited and sent back to the review committee again.
If all the RCRs are closed, the SCR will be acknowledged and summarized, and here comes the end of the change management process.

A. Appendices

A.1Quality evaluation document

This document describes the evaluation of the above Software Requirements Specification of the Urban Shared Parking Management System. There are 21 quality targets to be evaluated.

A.1.1 Unambiguous

Every requirements clause has the only explanation when the SRS is unambiguous. The 98% of the specific requirements clauses are clearly and unambiguous in the SRS of the USPMS.

A.1.2 Complete

The SRS is completed only when the SRS includes all possible situations and their effect on inputs and all page number, table number, graph number, definitions of terms and all references. According to the model of requirements integrity, the calculation result of integrity is 90%.

A.1.3 Correct

A SRS is correct only when every requirement expresses the expectation of the system; that is to say, every requirement satisfies the actual requirement. Based on the criterion, the correct of the SRS is 90% at least.

A.1.4 Understandable

A SRS should be understood by all kinds of readers including end-users, project managers and software developers easily. According to the formula ---- N_m/N_r, the measure of the Understandable is more than 93%.

A.1.5 Verifiable

If there exists limited and reasonable technologies to test and verify the system, a SRS can be verifiable. But there are always some nonfunctional requirements which are difficult to test and verify. Therefore, the measure of the verifiable is 90%.

A.1.6 Internally Consistent

A SRS is internally consistent only when there is no conflicts among stated requirements clauses. According to the DFD, the internal inconsistency can reach 94%.

A.1.7 External Consistent

A SRS is external consistent when the current stated requirements clauses are in line with baseline specification document. With the formula ---- N_{ec}/N_r, the external inconsistency can reach 95%.

A.1.8 Achievable

When and only when there is at least a design and implement solution to the SRS, a SRS can be achievable. So the Q_8 of the SRS is 1 which means there exists achievable solutions .

A.1.9 Concise

If the SRS is short and brief as much as possible and the quality is not effected, a SRS is concise. So the Q_9 = 1 / Size+1 means the degree of the concise of SRS. The Q_9 of the SRS is 30%.

A.1.10 Design-Independent

A SRS is designed independently when there exists more than one solutions can implements all requirements in a SRS. The Q_{10} is 90% in this SRS.

A.1.11 Traceable

A SRS is traceable when every requirements clause can be referenced independently. Every paragraph is numbered and has only one requirement clause. What's more, every requirement is given the only number and is described with "shall" or "should". Therefore, the Q_{11} is 1.

A.1.12 Modifiable

A SRS is modifiable when the style and structure of the SRS can be modified easily and consistently. The SRS has a catalog and index of tables and diagrams, so the Q_{12} is 1.

A.1.13 Annotated By Relative Importance

A SRS is annotated by relative importance when readers can judge which requirements are most needed for clients easily. So the SRS has been annotated by relative importance already.

A.1.14 Annotated By Relative Stability

A SRS is annotated by relative stability when readers can judge which requirements may be changeable usually easily. So the SRS is annotated by relative stability.

A.1.15 Annotated By Version

A SRS is annotated by version if readers can easily judge whether the requirements satisfy the requirements of product's version.

A.1.16 Not Redundant

A SRS is not redundant if the same requirements are described only once. Therefore, because the SRS has some repeated descriptions, the Q_{16} is 60%.

A.1.17 Precise

A SRS is precise if and only if all expressions are numerical as much as possible especially in nonfunctional requirements. So the SRS's accuracy is 90% at least.

A.1.18 Reusable

A SRS is reusable if and only if its sentences and paragraphs and sections can be adopted and reused easily by other SRSs. The degree of the SRS's reusability is more than 90%.

A.1.19 Traced

A SRS is traced if and only if every requirement's original source is clear. This SRS's information's source are noted clearly.

A.1.20 Organized

A SRS is organized if and only if its contents have been arranged according to different readers' type. The SRS is not arranged by the type of readers.

A.1.21 Cross-Referenced

A SRS is cross referenced if and only if the contents of SRS has cross-referenced relationships with other corresponding sections. The SRS has built good cross-referenced relationships among the whole document.

A.2 Table Catalog

A.3 Diagram Catalog