Software Requirements Specification

Geo Sensing Parking System

Augustine Thorbjornsen

Wujun Song (Carl)

Xuemin Guo (Emma)

Qijing Huang (Joshua)

# **1 Introduction – Augustine**

This section gives a scope, description and overview of everything included in this SRS document. Also, the purpose for this document is described and a list of abbreviations and definitions is provided.

## **1.1 Purpose – Augustine**

The purpose of this document is to give a detailed description of the requirements for the “Geo Sensing Parking System” software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval

## **1.2 Scope – Augustine**

The “Geo Sensing Parking System” is a GPS-based mobile web application which helps people to find the closest parking based on the user’s current position and automatically book their position once parked. The application should be free to access on all devices.

Furthermore, the software needs both Internet and GPS connection to fetch and display results and a live feed from sensors in the car parks. All system information is maintained in a database, which is located on a web-server. The software also interacts with the Google Maps API so users can view available parking spots on a map and be navigated to them. The application also has the capability of manually pre-booking a parking spot.

### 

## **1.3 Definitions, acronyms, and abbreviations**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| User | Someone who interacts with the web application |
| Admin/Administrator | System administrator who is given specific permission for managing and controlling the system |
| GPS | Global Positioning System |
| GPS-Navigator | An installed software on mobile phone which could provide GPS connection and data, show locations on map and find paths from current position to defined destination |
| Stakeholder | Any person who has interaction with the system who is not a developer. |
| PLAN | The level at which good success can be claimed contained in a language statement |
| DESC | Description |
| RAT | Rational |
| DEP | Dependency |

## **1.4 Overview**

At the end of this document there are three additional chapters and an appendix. The second chapter provides an insight into how the system works and how it interacts with other systems. This chapter presents different types and system interactions of the stakeholders. The chapter also refers to the restrictions and assumptions of the product system. The third chapter provides detailed requirements and a description of the various interfaces of the system. In order to specify the requirements more precisely for different audiences, different specification techniques are used. In the fourth chapter the requirements are prioritized. It provides a basis for the priority methods chosen and discusses why other alternatives have not been selected. At the end of the document, the Appendices include all the prioritization results of the requirement and a release plan based on them.

## **1.5 Feasibility Study – Xuemin Guo**

### **1.5.1 Market feasibility:**

1. The population grow rate in Darwin is consistent and strong.

2. About 70% of people go to work by driving in Darwin.

As more and more people move to Darwin every year and most of the people choosing to drive to work. Darwin needs to find a better way to arrange the public car park. A Geo-sensing parking system will be useful in the future.

### **1.5.2 Technical Feasibility:**

The whole architecture is based on the J2EE component framework. The following programming techniques will meet the technical demand of this system:

1. The website for users will be developed through JSP, BOOSTRAP, HTML, CSS and JS

2. The back end will be using JavaBean, Spring (part of JavaEE framework), XML, DAO, Java Servlet, JavaScript

3. The system will use SQL and ORACLE database.

## **1.6 Milestones – Augustine/ Xuemin Guo**

|  |  |  |
| --- | --- | --- |
| **Project Milestone** | **Initial Estimated Completion Date** | **Actual Completion Date** |
| Meet with Client | 18/3/2019 | 18/3/2019 |
| Paper Prototype | 18/3/2019 | 18/3/2019 |
| Finalizing software requirements | 25/3/2019 | 25/3/2019 |
| Finalizing the software specifications | 19/3/2019 | 19/3/2019 |
| Demo of the designed system | 25/03/2019 |  |
| Usability Testing | 25/03/2019 |  |
| Report of development | 26/03/2019 |  |
| Report of Testing(function & security) | 26/03/2019 |  |
| Delivery result Report | 10/04/2019 |  |

# **2 Overall description – Augustine**

An overview of the whole system is provided in this section. In its context, the system will demonstrate how the system interacts with other systems and introduce its fundamental functionality. It also describes the type of stakeholders who will use the system and the features available for each type. The system will at last present its constraints and assumptions.

## **2.1 Product Perspectives – Xuemin Guo**

# 

## **2.2 Product Functions – Xuemin Guo**

The users will be able to register and login in the web first. After the users search for the parking space, the results will be shown in a map, the users can choose the parking space by click on the map. A map for the parking space will show all the available parking slot. The users can choose one of the parking slot and book for it.

When the users parking their car into the slot, the sensor signal turn red into yellow, in the meanwhile, the GPS location will be sent to the system, after the system matches the location of the GPS and slot, it will create a order. When the parking is completed, the signal will turn to red and the payment will be automatically done, the users will get a receipt sending from the system.

## **2.3 User characteristics – Qijing Huang**

The system has two types of users: web application users and administrators. Each of these two types of users uses the system differently so that each user has its own needs.

In this parking system, the users will be able to search for the parking space, choose one of them in the map and then book the available parking slot. The users will get the receipt of the payment from the system after finished the parking, it will show the start time, end time and total payment.

The administrators also interact only with the web application. They manage the whole system so that there is no wrong information being stored. The administrators may manage information for the web application.

## **2.4 Product Constraints – Wujun Song**

The web application has a few constraints as the Google Maps API may not be turned on for each user. The Internet connection for the application is also a constraint. Since the application retrieves data from the database, it is essential that the application has an internet connection.

The capabilities of the database will constrain the web application. As the database handles incoming requests and it may have to force a queue and as a result the time takes to fetch data increases.

## **2.5 Assumptions and Dependencies – Augustine**

Every user will have a different device. This application assumes that the user is using a mobile device which has GPS and internet connection. This mobile device is assumed to have enough performance to retrieve real time data from the database.

Another assumption is the GPS which is assumed to have +-1m accuracy which is needed for correctly booking a parking spot. If the GPS is not accurate it will cause false data to be uploaded to database and incorrectly book a parking spot.

# **3 Specific requirements**

This section contains all the system requirements for functional and quality. The system and all its features are described in detail.

### **3.1.1 User interfaces -Xuemin Guo**

When the user opens our website, the login in page will be shown, if it is the first for the user, he or she can do register first by clicking the “sign up”. Then the user will be able to do the register first. (figure)

### **3.1.2 Hardware interfaces – Augustine**

Since the mobile application has no designated hardware, it has no direct hardware interfaces. The physical GPS is managed by the user’s device and the hardware linked to the database server is managed by the underlying mobile and web server operating system.

### **3.1.3 Software interfaces – Augustine**

In order to get geographical information about the location of the user and the view and the information on the parking spots available, the web application communicates with Google Maps API. The communication between the database and the web application consists of read and modify data.

### **3.1.4 Communications interfaces *– Yet to be Completed***

### 

## **3.2 Functional requirement:**

## 

### **3.2.1 User Class 1 – The User**

#### **3.2.1.1 Functional requirement 1.1**

**ID: FR1**

TITLE: User access – Web application

DESC: The user has to have access to the internet, then the user can access the web application.

RAT: In order for a user to access the web application.

DEP: None

#### **3.2.1.2 Functional requirement 1.2**

**ID: FR2**

TITLE: User registration – Web application

DESC: Given that a user has accessed the web application via a browser, then the user should be able to register. The user must provide email address, password and credit card details.

RAT: In order for a user to register on the mobile application.

DEP: FR1

#### **3.2.1.3 Functional requirement 1.3 -Wujun Song/Qijing Huang**

**ID: FR3**

TITLE: User login – Web application

DESC: Given that a user has registered, then the user should be able to log in to the web application. The log-in information will be stored in the database.

RAT: In order for a user to login into the web application.

DEP: FR2

1. As a end user(driver), I want the system could show available slots in each area so that slots could be easily found.

3. As a client, I want a system integrated with Google Map so that the system could provide Google service as well.

4. As a end user(driver), I want an web-based mobile friendly application so that I can visit it also from mobile phone.

5. As a end user(driver), I want 2 hours free period. Parking time starts from 8 am - 4:30 pm.

6. As a end user(driver), I want the system could show available slots in each area so that slots could be easily found.

### **3.2.2 Non-functional requirement: Augustine/Xuemin Jun**

1. As a client, I want multilayered system so that the system could have a multi-layered defence.

2. As a client, I want security test to ensure that System could prevent Cross site scripting attacks or forgery.

3. As a client, I want security test to ensure that System could prevent SQL Injection.

# **4 References**

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

[2]Davis M A, “Just Enough Requirements Management: Where Software Development Meets Marketing”, New York, Dorset House Publishing, 2005.

[3]Karlsson J, “A Cost-Value Approach for Prioritizing Requirements”, Norges Teknisk- Naturvitenskapelige Uni. 1997

[4]<http://www.population.net.au/darwin-population/>

[5]<http://stat.abs.gov.au/itt/r.jsp?RegionSummary&region=70101&dataset=ABS_REGIONAL_ASGS2016&geoconcept=ASGS_2016&measure=MEASURE&datasetASGS=ABS_REGIONAL_ASGS2016&datasetLGA=ABS_REGIONAL_LGA2017&regionLGA=LGA_2017&regionASGS=ASGS_2016>

# **5 Appendix**

# 