Software Requirements Specification

for

<Park King>

Version 1.1 approved

Prepared by <Ahmed Elgendy, Travis Maupin>

<Ahmed Elgendy, Dillion Driskell, Travis Maupin, Zacc Weldon>

<10/15/2017>

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Project Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Features 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

3. System Features 3

3.1 System Feature 1 3

3.2 System Feature 2 (and so on) 4

4. External Interface Requirements 4

4.1 User Interfaces 4

4.2 Hardware Interfaces 4

4.3 Software Interfaces 4

4.4 Communications Interfaces 4

5. Other Nonfunctional Requirements 5

5.1 Performance Requirements 5

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 6

Appendix C: Issues List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Revision 1 | 11/2/2017 | Implementing a decentralized database instead of a centralized database | 1.1 |
|  |  |  |  |

# Introduction

## Purpose

* Product: Parking meter application (Park King)
* Version: 1.0
* Scope: The whole application

## Document Conventions

* Each requirement will have its own priority.
* Requirements are placed under their related heading with all requirements in the same section having the same format.
* Each heading in the document is numbered (1-6)
* Each subheading takes the number of the heading in addition to a section number. For example section 1 in heading number 1 has the following numbering convention: 1.1

## Intended Audience and Reading Suggestions

* This document is intended for developers, projects managers, testers, and documentation writers.
* The document will have information about the functional and nonfunctional requirements, designs, and security issues of the project.
* Suggested reading sequence:
  + Overall description
  + System features
  + Nonfunctional requirements
  + External interface requirements
  + Other requirements

## Project Scope

* The idea of this project is to facilitate finding and paying for parking meters for the users.
* The goal is to allow users to search through the different parking meters close to them, figure out which ones are occupied and which are not, and then being able to pay for the parking meter they choose provided that it is not occupied. It also allows the user to extend the time period required if the user needs more time to use the parking meter.
* Benefits of this project include:
  + Fast access to available parking meters.
  + Searching parking meters in different locations.
  + Making payments for meters on the fly instead of having to go to the parking meter and make that payment.

## References

* Use case document:
  + Title: Parking Meter Use Case
  + Author: Travis Maupin
  + Version: 1.0
  + Date: 09/05/2017
  + Location: On the CD provided
* Class Diagram:
  + Title: Parking Meter Class Diagram
  + Author: Dilliion Driskell
  + Version: 1.0
  + Data: 09/10/2017
  + Location: On the CD provided

# Overall Description

## Product Perspective

* This product is a new one and does not have any predecessor.
* It was inspired by the fact the many people have many problems with the current parking meters found on the streets.
* The main idea of the product is to allow users to access parking meters from any place and pay for the parking meter using the application.

## Product Features

* The user is required to create an account before using the application
* After creating the account, the user is asked to provide payment information before being able to access the maps.
* After providing the payment information, users can then search for parking meters at any location.
* Once the user decides on the parking meter he/she wants, the user can pay for that meter for a specific amount of time. User can choose to pay for a parking meter for 15, 30, or 60 minutes.
* If the user wants to extend the time, he/she is allowed to do so by selecting the extra amount of time they need and paying for that additional time.

## User Classes and Characteristics

* Any person with a cell phone and internet access can use this application. However, some of the most important features of the users includes:
  + Some technical expertise.
  + Some experience with using maps.
  + Tech savvy: the user must understand that he/she has to create a very strong account to prevent identity theft.

## Operating Environment

* Operating system: Android Lollipop 5.0 or any next version of Android

## Design and Implementation Constraints

* The application only supports the English language.
* User must maintain a strong username and password for his/her account.

## User Documentation

* Not applicable

## Assumptions and Dependencies

* Not applicable

# System Features

## Checking available parking meters

3.1.1 Description and Priority

The user is able to check different parking meters available on the map and see whether they are occupied or free to use. The user can check for parking meters in any location. This feature is of high priority.

3.1.2 Stimulus/Response Sequences

The user has to create an account and then store payment information in order to be able to access this feature.

3.1.3 Functional Requirements

The device used has to have internet access.

The software of the device should be Android 5.0 or any version after that.

## Changing account information

3.2.1 Description and Priority

The user is able to change any of the account information from the settings after that information has been entered earlier. This feature is of low priority

3.2.2 Stimulus/Response Sequences

The user has to create an account in order to be able to access this feature.

3.2.3 Functional Requirements

The device used has to have internet access.

The software of the device should be Android 5.0 or any version after that.

## Paying for the desired parking meter

3.2.1 Description and Priority

The user is able to pay for the desired parking meter on the condition that it is not occupied by anyone. This feature is of high priority.

3.2.2 Stimulus/Response Sequences

The user has to create an account, store payment information, and select an available parking meter in order to be able to access this feature.

3.2.3 Functional Requirements

The device used has to have internet access.

The software of the device should be Android 5.0 or any version after that.

## Extending the time of the currently used parking meter

3.2.1 Description and Priority

Before the time for the parking meter is over, a user can choose to extend that time by making a new payment if he/she finds out that they need more time for parking the vehicle. This feature is of high priority.

3.2.2 Stimulus/Response Sequences

The user has to create an account, store payment information, and select an available parking meter in order to be able to access this feature.

3.2.3 Functional Requirements

The device used has to have internet access.

The software of the device should be Android 5.0 or any version after that.

## Searching for meters on the map

3.2.1 Description and Priority

The user is able to use the search mechanism implemented with the map to search for parking meters in different locations

3.2.2 Stimulus/Response Sequences

The user has to create an account, store payment information.

3.2.3 Functional Requirements

The device used has to have internet access.

The software of the device should be Android 5.0 or any version after that.

## Storing the information in a decentralized database

3.2.1 Description and Priority

After making any changes to the user information or the payment information, the user is able to save these changes in the centralized database

3.2.2 Stimulus/Response Sequences

The user has to create an account, store payment information

3.2.3 Functional Requirements

The device used has to have internet access.

The software of the device should be Android 5.0 or any version after that.

## Getting notifications once the meter’s time is close to expiration

3.2.1 Description and Priority

After the user chooses the meter and pays for it, the user will get a notification on the device that warns the user that the meter’s timer is closing to expiration

3.2.2 Stimulus/Response Sequences

The user has to create an account, store payment information, select an available parking meter, and pay for the meter chosen in order to be able to access this feature.

3.2.3 Functional Requirements

The device used has to have internet access.

The software of the device should be Android 5.0 or any version after that.

# External Interface Requirements

## User Interfaces

The user interface begins with a standard login/create account screen. There are two edit text fields for username and password with a login button that will compare that to a database to see if it is accurate. There is also a create an account button that will take the user to the create account screen. A toast appears upon successful login

The create account screen contains four edit text fields in which the user is prompted to enter a username, choose a password, verify the password, and enter an email address. A submit button is available that will store all information to a database. After submitting a payment info screen appears. A toast appears upon successful account creation

The payment screen has eight edit text fields labeled appropriately with “Credit Card Info” and “Billing Info” they take the card name, number, expiration date, security code, billing address, zip code, city/state, and country. There is also a submit button that stores all information in a database. After the account creation process is done or if the login is successful, a map screen will appear. A toast appears upon successful registrations of payment

The map screen has a Google API map with nodes representing each parking meter. Upon clicking any node information about availability appears. There are also navigational options which will open up the official google map navigation app and give directions to the specified node. If the availability information is clicked on, a full meter information screen is opened.

The meter information screen contains the address, ID, price, current availability/time until available. It also has radio buttons to choose how long you would like to reserve the meter and a payment button that will render payment for the meter. A toast appears upon payment

## Hardware Interfaces

The application supports any android device running at least android 5.0 lollipop. The application does not support any other type of device at this time.

## Software Interfaces

This application makes use of the android operating system. It requires at least android 5.0 lollipop to function. This system makes use of the xml language in order to display UI elements and handles most other operations with JAVA. The application also makes use of an SQLite database for the storing of customer and meter information. A google API is also used in order to render the map functionality of the application.

## Communications Interfaces

This application requires location information and a web connection in order to properly process the map functions and communicate with the API. It also requires an email address upon registration of the user account. The email is used to both provide a point of contact between the user and the application development team, and it is used in order the hinder the creation of mass bot accounts.

# Other Nonfunctional Requirements

## Performance Requirements

* The application should start in a timely manner.
* With a good internet connections, maps should appear quickly and user should be able to go through the maps without any delay.

## Safety Requirements

* The user must use a well-known and trusted network to prevent any possible attacks from unknown sources.
* The user also must make sure to have a strong password to prevent other people from anticipating the user’s password and as a consequence get to know information about the user’s payment methods.
* The user must make sure to logout after using the application to prevent anyone from accessing the user’s information in case the device is left unattended or gets stolen.

## Security Requirements

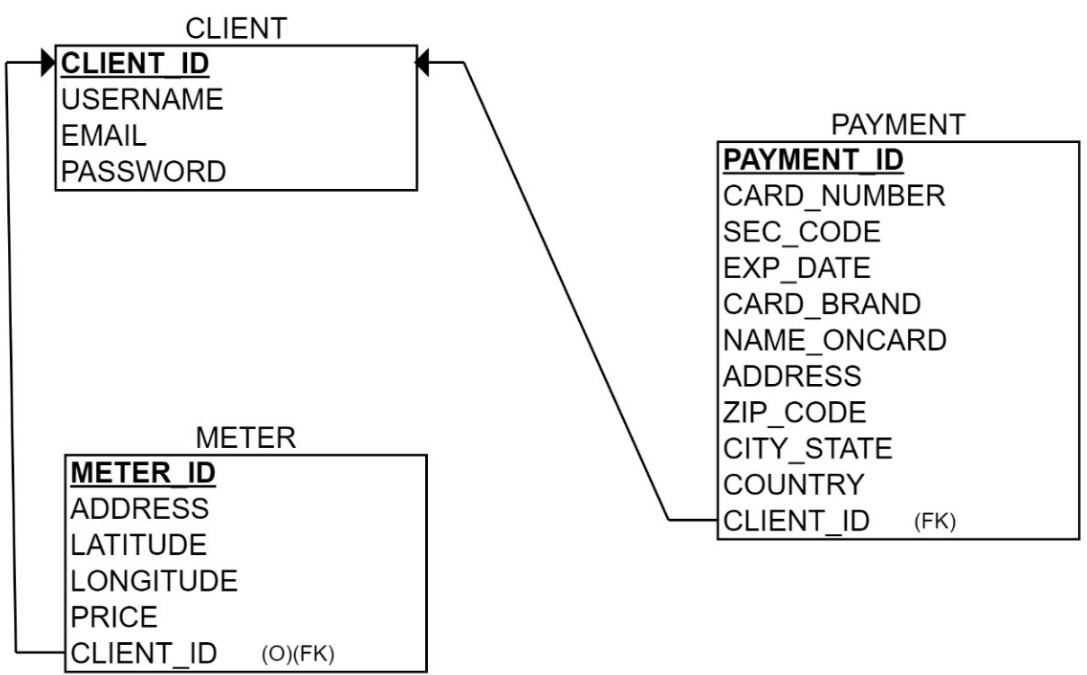
* User is required to create an account before using the app.
* After creating an account and logging in, the user must add a payment method if he/she intends to use a parking meter or add time to their current time.
  + The payment information needed will be:
    - Name on card
    - Card number
    - Security code for the card
    - Expiration date

## Software Quality Attributes

* The product will run on any version of Android starting from Lollipop 5.0.
* The application can be used only on phones and tablets, but can’t be used for TV and wear.
* Test cases can be produced to make sure that the application runs as intended.
* Users can adapt to the application easily with its user friendly UI.

# Other Requirements

Information entered by the user has to follow the following schema of the database:

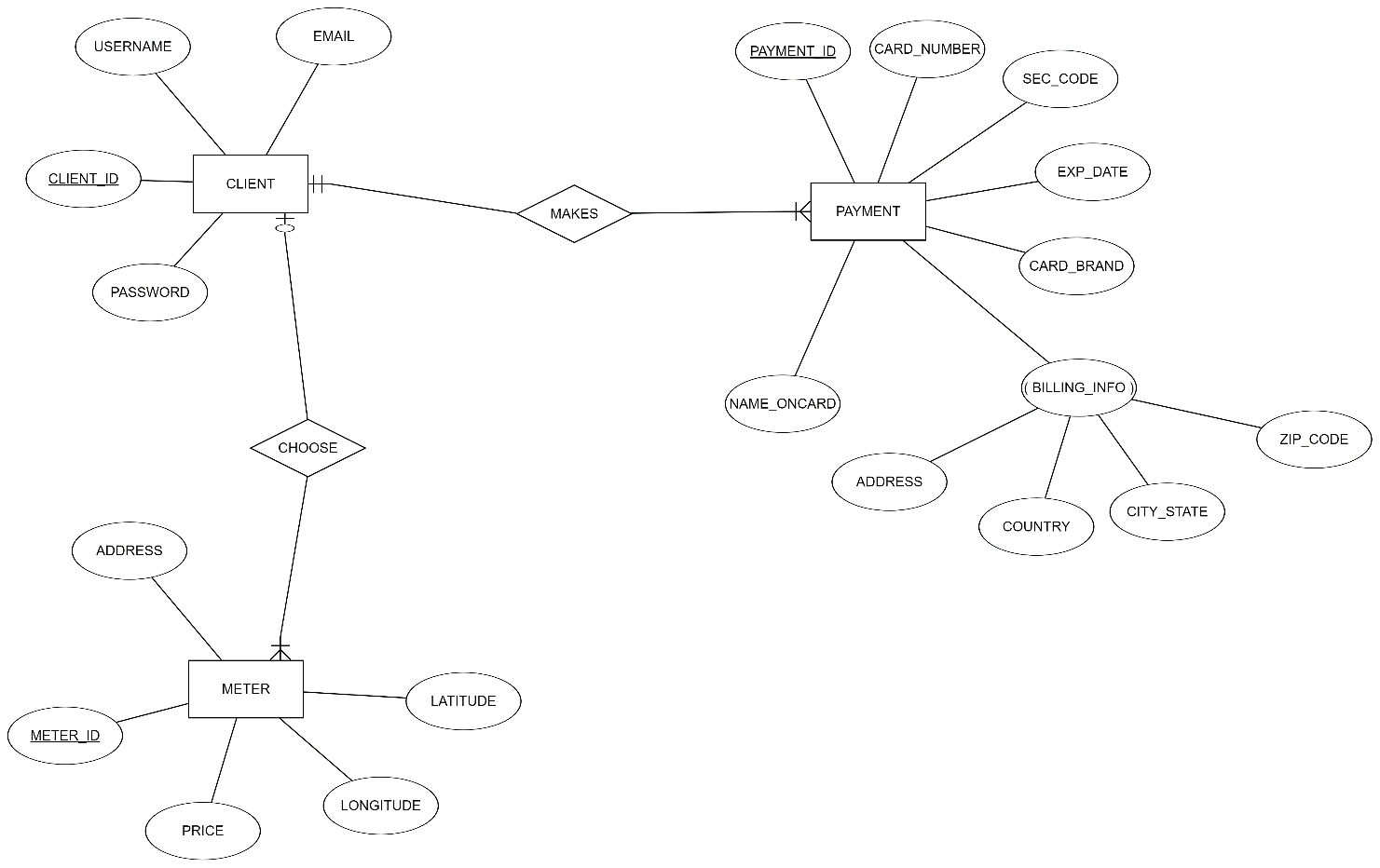


Appendix A: Glossary

* Not applicable

Appendix B: Analysis Models

* Class diagrams and use case diagrams are provided on the CD.
* Entity relationship model for the database if provided as follows:

**

Appendix C: Issues List

Not applicable