ParkFinder
Software Requirements Document
SE 3A04

Abdul Ahad	Salma Belal	Josh Chatten	Nathanael Jordan	Robert Stuart
akhteraa	belalsm	chattejj	jordanen	stuarr2

March 4, 2016

Contents

1 Introduction

1.1 Purpose

The purpose of this software requirement specification document is to provide a description of the requirements needed to design the software for controlling the ParkFinder app. This app helps users by providing a more efficient method for looking up parks and acquiring park information.

The intended readers of this document include all of the project's stakeholders. This includes the end-user, the software engineers, and the park authorities.

1.2 Scope

The software product being described in this document is called the ParkFinder app. This product will have datasets of information about parks all over the world and will allow the client to use search methods in order to find parks based on their desired attributes. The app is meant to be used anywhere in the world, provided an Android or iOS device with the app installed. This provides clients with an easier, faster, and more efficient way to look up parks and acquire information such as the location, facilities, activities, and rentals that the parks provide.

1.3 Definitions, Acronyms, and Abbreviations

Experts A search criteria used for the identification of a particular park, or group of parks. ex. An expert could be for the rentals available at a park

SRS Software requirements specification.

API Application programming interface.

1.4 References

N/A

1.5 Overview

The following section of this document, Overall Description, provides the reader with an overview of many important aspects of ParkFinder. This includes information about the functionality of the app, characteristics of the intended users, the constraints that will limit developer options, and any assumptions or dependencies that can potentially change the requirements. The last two sections of the SRS deal with both the Functional and Non-Functional Requirements. Included with the Functional Requirements will be all the business events the system will need to handle and the viewpoints for each event. The Non-Functional Requirements are divided into several categories such as requirements for the look and feel of the system or the performance requirements.

2 Overall Description

2.1 Product Perspective

Several web-services, such as Google or `www.ontarioparks.ca`, are available to assist with locating parks. However these services are not available to an off-line user. The ParkFinder app will allow its users to find parks that match their search criteria. More importantly, the app will be able to perform the majority of its search functionalities without an Internet connection, thus allowing the user the freedom to use the app whenever and wherever they want. As the ParkFinder app only requires external resources for a subset of its functionality and no external applications depend upon the ParkFinder app, it cannot be considered to be a component of a larger system.

2.2 Product Functions

a) The product shall maintain a database of parks and associated attributes

b) The app will provide a minimum of four methods for querying the park database, these methods will be referred to as "experts"

c) One expert must be able to locate parks based on location, to do so an on-line mapping service will need to be utilized (when an Internet connection is available)

d) Results from a users query shall be displayed via a centralized "forum"

e) Information sent to and from the "forum" shall be encrypted to ensure privacy

f) The user can select a park from the query results and the ParkFinder app shall display more detailed information about that park, including a weather report (when an Internet connection is available)

2.3 User Characteristics

The intended users of the ParkFinder app are people wishing to discover new parks that they have not been to before as well as people completely new to visiting parks. It is expected that the primary users will be adults however, visiting parks can be a family affair and thus children can also be expected to use the ParkFinder. Thus we expect the ParkFinder app to be used by children no younger than that of a third grader (8-9 years old), adults, and the elderly. It is expected that users will understand English, up to a grade three level. Being an app solely available on mobile devices, it can be expected that users will possess the bare minimum skill required to operate a mobile device. Such skills would include being able to select buttons, use a keyboard, and navigate menu screens.

2.4 Constraints

The primary constraint limiting the development teams options is time. Due to chaotic scheduling between this project and others, some minor functionality may not be implemented.

2.5 Assumptions and Dependencies

For the use external services, such as weather updates or the use of a mapping service, it is assumed that the user has an Internet connection. These services are also assumed to provide accurate information to the user. Also assumed that the user will run the application on supported android device. All information acquired about each park is assumed to be correct and up to date.

2.6 Apportioning of Requirements

The addition of parks from all over the world may be delayed and just parks from Ontario will be implemented in the initial version of the system.

3 Functional Requirements

This section of the SRS contains all of the functional software requirements for the ParkFinder app. This section enables designers to design a system satisfying those requirements, and the testers to test that the system satisfies those requirements.
Note: The functional requirements are organized by business events (BE), then by viewpoints (VP).

BE1. The developer wants to change or remove an expert

VP1.1 Developer

i. The system will allow the developer to swap or remove the expert

VP1.2 Security

i. The system will check if the swap is being made by an authorized party

VP1.3 User

i. N/A

VP1.4 Manager

i. The manager will be asked to give permission for the swap

BE2. The user wants to search for parks that fit specified criteria

VP2.1 Developer

i. N/A

VP2.2 Security

i. The system will encrypt and then decrypt the user's input

ii. The system will encrypt and then decrypt the system's output

VP2.3 User

i. The system shall enable the user to choose different types of search criteria, such as: amenities, activities, rentals, park size, and seasonal dates

ii. Search results will be displayed, showing all parks that match the chosen criteria

VP2.4 Manager

i. N/A

BE3. The user wants to view more information about a specific park

VP3.1 Developer

i. N/A

VP3.2 Security

i. The system will encrypt and then decrypt the user's input

ii. The system will encrypt and then decrypt the system's output

VP3.3 User

i. The system shall give the user an overview of the park. Thus will include the highlights and popular attributes of the park, the address, phone number, size, website, and operational dates

ii. The system shall give the user information about all of the available amenities, activities, and rentals at the park

iii. The system will show the current weather conditions at the park

VP3.4 Manager

i. N/A

BE4. The user requests to view the location or locations of a selected park or several parks

VP4.1 Developer

i. N/A

VP4.2 Security

i. The system will encrypt and then decrypt the user's input

ii. The system will encrypt and then decrypt the system's output

VP4.3 User

i. The system shall display the location of the park or parks on a map

VP4.4 Manager

i. N/A

BE5. The user wants to find the nearest 5 parks to their current location

VP5.1 Developer

i. N/A

VP5.2 Security

i. The system will encrypt and then decrypt the user's input

ii. The system will encrypt and then decrypt the system's output

VP5.3 User

i. The system will display the closest 5 parks to the user's location

VP5.4 Manager

i. N/A

4 Non-Functional Requirements

4.1 Look and Feel Requirements

4.1.1 Appearance Requirements

LF1. The application shall use the team logo displayed on team developed pages of the application.

LF2. The application shall use interactive displays that look like they can be interacted with.

4.1.2 Style Requirements

LF3. After reading a brief description of the application, 90 percent of potential users shall feel that the application is trustworthy.

4.2 Usability and Humanity Requirements

4.2.1 Ease of Use Requirements

UH1. 90 percent of a test panel with a third grade education shall successfully complete the use of three application experts within 20 minutes of using the application.

UH2. After a two-week period with the application, users shall exhibit an error rate of less than two percent.

UH3. A feedback survey shall show that 85 percent of users with little English background shall successfully use one expert's functionality within 20 minutes.

UH4. 90 percent of a sample of people with a third grade education shall understand 99 percent of the language used in the application.

4.2.2 Personalization and Internationalization Requirements

UH5. The application shall allow the user to choose their preferred background colour scheme for the application.

4.2.3 Learning Requirements

UH6. It shall take 20 minutes to learn how to use 90 percent of the application's functionality based on a test pool of people with a third grade education.

UH7. All application functionality shall be easy to use without a tutorial or training for the application.

4.2.4 Understandability and Politeness Requirements

UH8. Use of the application experts shall be intuitive as to what inputs these experts expect.

UH9. The application shall hide its internal details from the user.

4.2.5 Accessibility Requirements

N/A

4.3 Performance Requirements

4.3.1 Speed and Latency Requirements

PR1. The application shall return information to the user within 2 seconds.

4.3.2 Safety-Critical Requirements

PR2. The application shall not consume more than 500mW of power at any given time.

4.3.3 Precision or Accuracy Requirements

PR3. The application shall display park ranges as a integer value.

PR4. The application shall display park size as an integer value.

PR5. The application shall display GPS coordinates up to two decimal places using decimal degree format.

4.3.4 Reliability and Availability Requirements

PR6. The application's weather service and mapping service will only be available when the device has an Internet connection.

4.3.5 Robustness or Fault-Tolerance Requirements

PR7. In the event of losing Internet connectivity, the user will still be able to use the application's experts.

4.3.6 Capacity Requirements

N/A

4.3.7 Scalability or Extensibility Requirements

N/A

4.3.8 Longevity Requirements

PR8. The application is expected to operate and be available for download for at least 5 years without any budget constraint.

4.4 Operational and Environmental Requirements

4.4.1 Expected Physical Environment

OE1. The application can be used anywhere on any supported Android device.

OE2. The application location-based services will only be available where an Internet connection is present.

4.4.2 Requirements for Interfacing with Adjacent Systems

OE3. The application will interface with a mapping service, weather service and the device global positioning system.

4.4.3 Productization Requirements

OE4. The application will be easy to install through the application store service.

4.4.4 Release Requirements

OE5. The application will be available to install through the application store service.

4.5 Maintainability and Support Requirements

4.5.1 Maintenance Requirements

MS1. Application updates will be pushed through the application store services, annually, to all devices with installed instances of the app.

MS2. Any application security vulnerabilities will be patched immediately and pushed to the application store service.

4.5.2 Supportability Requirements

MS3. The application shall only run on supported Android devices.

4.5.3 Adaptability Requirements

MS4. The application shall be modular enough to make it easier for future feature additions and platform migrations.

4.6 Security Requirements

4.6.1 Access Requirements

SR1. Expert information and modification will only be accessible by system administrators, such as the managers and developers.

4.6.2 Integrity Requirements

SR2. The application shall prevent incorrect data from being introduced.

4.6.3 Privacy Requirements

SR3. The application shall encrypt all user data. Any user data including searches will be kept confidential.

4.6.4 Audit Requirements

N/A

4.6.5 Immunity Requirements

SR4. The application shall be immune to any manual attacks such as man-in-the-middle as well as automated attacks from Trojan horses and malicious scripts.

4.7 Cultural and Political Requirements

4.7.1 Cultural Requirements

CP1. The application will remain culturally and religiously sensitive.

4.7.2 Political Requirements

N/A

4.8 Legal Requirements

4.8.1 Compliance Requirements

LR1. The application shall comply with the regional safety and confidentiality requirements.

4.8.2 Standards Requirements

LR2. The application shall comply with Android standards including resolution, screen size, and speed limitations.

A Division of Labour

Contributions	Name	Signature
Sections 4.4-4.7	Abdul Ahad	
Sections 1.1-1.2 & 3	Salma Belal	
Sections 1 & 2.5-2.6	Josh Chatten	
Sections 4.1-4.3	Nathanael Jordan	
Sections 2.1-2.4	Robert Stuart	

B Change Log

B.1 March 4, 2016 - Re-submission with Deliverable 2

The following changes were made due to feedback from the teaching assistants:

- **Product Functions** Experts being separate and easily swappable removed
- **Product Functions** Modified encryption/decryption statement to remove the idea of the user communicating with the experts
- **User Characteristics** Explicitly stated the expected ages of the users as well as the expected target audience language (English)
- **Functional Requirements** Combined previously separated business events for each Expert type into a single business event (see BE2)
- **Functional Requirements** Removed the Geographical viewpoint from all business events
- **Access Requirements** Corrected interpretation of the requirement
- **Integrity Requirements** Corrected interpretation of the requirement
- **Political Requirements** Removed unneeded requirement