# Software Requirements Specifications

Team 1

March 2, 2016

## 1 Introduction

This section of the SRS gives an overview of everything included in the SRS.

### 1.1 Purpose

The purpose of this document is to give a detailed description of the software. It will illustrate the features of the software, the purpose, what it will do, the requirements, any constraints, and interactions with other systems or with its environment. This document will mainly be utilized by the developers of the system, but will also be reviewed the TAs.

### 1.2 Scope

The "Movie Matcher" is mobile application that will help the user find a movie given some information the user knows about the movie. The application should also display information about any movies matching the user's criteria. If the user finds the movie they were looking for, they should be directed to nearby theaters that are playing the movie if it is a new movie. The user should also be given links to purchase the movie if they want. The goal of this software is to help people identify movies based on characteristics they know about the movie they are looking for.

This software will not guarantee that it will always find the correct result, or any result at all. The software will try its best to identify nearby theaters based on its interaction with the GPS in the phone. The software will also interact over an Internet connection to perform the search.

This application is intended to be used in Canada, so it will be use Canadian English text. The input to the software is expected to Canadian English as well. Also, since the application is intended for use in Canada only, the theater location service and movie purchasing features only concern Canadian theaters and retailers.

### 1.3 Definitions, Acronyms, and Abbreviations

a) Developer: the people creating the software

b) User: the person using the software

### 1.4 References

None

### 1.5 Overview

The next section, Overall Description, gives an broad view of factors that affect the system-to-be. It does not detail specific requirements, it gives background information which is relevant to the requirements. The section after, Functional Requirements, details each of the requirements for the system. The requirements are elaborated in enough detail to allow the design of the system and allow tests to be constructed. The next section, Non-Functional Requirements, contains qualities that the overall system should have. The last section, Division of Labour, states who did a certain piece of work.

# 2 Overall Description

Movie Matcher is an application which provides a list of possible movies matching criteria that the user gives it. To this effect the application must communicate with the user, execute internal computations and communicate with external experts. Given this environment, certain requirements must be defined in order for a clear product to be designed and developed.

## 2.1 Product Perspective

The Movie Matcher application is an aggregate searching tool that returns possible movies matching the query made by the client (user) modified by parameters such as cast and crew, notable quotes, as well as genre and plot description. Movie Matcher will achieve this by accessing data from external and disparate experts that will be referenced against the parameters provided by the client.

## 2.2 Product Functions

Movie Matcher is intended to handle the following 3 business events:

1. User enters a query
2. User sorts and selects a result

The Movie Matcher system will uniquely respond to all of these business events as follows:

1. Query its experts based on the user input parameters and process the results to return to the user a list of possible matching movies
2. The query is returned as per usual and are ordered depending on the sorting parameter

## 2.3 User Characteristics

The following assumptions pertain to at least 80% of our expected users:

a) The user is able to comprehend enough of the English language to operate Movie Matcher

b) The user is able to operate an Android OS mobile application

c) The user has access to an operational Internet connection

Assumptions of the users is based on aspects required of the user to operate Movie Matcher in some fashion.

## 2.4 Constraints

a) Application must use external connections and performance is reliant on the network that is being used

b) The quality and format of the data of the experts is dependant on the choices of the experts

c) The nature of the data provided by the experts may limit the possible search parameters

## 2.5 Assumptions and Dependencies

a) The application will be distributed as an Android OS mobile application

b) The application will externally communicate with the chosen experts

## 2.6 Apportioning of Requirements

The social aspect of Movie Matcher has yet to be defined at this time and as such will be considered later.

Thus when the social aspect of Movie Matcher is considered its requirements must be evaluated again.

# 3 Functional Requirements

This section of the SRS should contain all of the software requirements to a level of detail sufficient to enable designers to design a system to satisfy those requirements, and testers to test that the system satisfies those requirements. Throughout this section, every stated requirement should be externally perceivable by users, operators, or other external systems. These requirements should include at a minimum a description of every input (stimulus) into the system, every output (response) from the system, and all functions performed by the system in response to an input or in support of an output.

BE1. User enters a query

VP1.1 User

i. The product allows the user to search for movies based on the search criteria entered by the user.
ii. The product allows the user to enter a quote as search criteria for searching for movies.
iii. The product allows the user to choose genre/type of movies for search criteria.
iv. The product allows the user to enter the names of the cast for search criteria.
v. The product allows the user to choose whether it wants to search box office movies only for search criteria.

VP1.2 Developer

i. Developer has access to network and security.
ii. Developer has access to a debug mode where implicit computations are visible on the console.

BE2. User sorts and selects a result

VP2.1 User

i. The product displays the search results with each of the three experts results.
ii. The product displays the best answer to the search result as well.
iii. The product allows user to sort the result of the search based on search criteria including but not limited to rating, release date.
iv. The product allows the user to select which movie to proceed with which further directs the user into a Google maps view of the location and timings of the show.

VP2.2 Developer

BE3. User selects location to navigate to

VP3.1 User

i. The product shall display locations where the movie selected is playing.
ii. The application shall open up the phones navigation application.

VP3.2 Developer

# 4 Non-Functional Requirements

## 4.1 Look and Feel Requirements

### 4.1.1 Appearance Requirements

LF1. The application should be visually appealing.

LF2. The overall design should look familiar to other query based applications.

LF3. The interface should be descriptive, telling the user what to do and how to do it.

LF4. The query search and the experts should be easily visible.

LF5. The results displayed should be easily readable.

#### 4.1.2 Style Requirements

N/A

1. hello

## 4.2 Usability and Humanity Requirements

#### 4.2.1 Ease of Use Requirements

UH1. The application should be easy to use just like an ideal query processing software (navigate, surf, search, get results, etc.)

#### 4.2.2 Personalization and Internationalization Requirements

#### 4.2.3 Learning Requirements

UH2. The overall functionality of the application should be basic, a child of 10 years or older should be able to use it.

UH3. The application should be designed in a way that query-processing is not difficult.

#### 4.2.4 Understandability and Politeness Requirements

#### 4.2.5 Accessibility Requirements

## 4.3 Performance Requirements

#### 4.3.1 Speed and Latency Requirements

PR1. The application should be able to present the results of a search requested by the user within five seconds of the request being made.

PR2. The application should be respond within three seconds to a user command other than a search (open, close, etc.).

#### 4.3.2 Safety-Critical Requirements

PR3. The application should project proper lighting on the screen (prevent seizures and blindness).

PR4. The application must abide by all internet/software safety laws.

#### 4.3.3 Precision or Accuracy Requirements

PR5. The software shall provide at most 10 different movies when a query search is made.

PR6. The software shall perform the exact search the user requests. It shall not assume a different spelling (i.e. to correct a spelling mistake, etc.).

#### 4.3.4 Reliability and Availability Requirements

PR7. The product should be usable for 24 hours per day, 365 days per year.

#### 4.3.5 Robustness or Fault-Tolerance Requirements

#### 4.3.6 Capacity Requirements

PR8. The application shall allow at most one user pewr device to operate the software.

PR9. The application shall ideally search for one movie per search.

#### 4.3.7 Scalability or Extensibility Requirements

#### 4.3.8 Longevity Requirements

### 4.4 Operational and Environmental Requirements

#### 4.4.1 Expected Physical Environment

OE1. The product shall run on any android platform device along with computers, tablets, PCs, and cellphones (other devices that support Android).

#### 4.4.2 Expected Operational Behaviour

OE2. The system shall resemble a simple input/output mechanism. The user will enter a search and the software shall process the requested query and present the results.

OE3. The software shall interact with the user to perform transactions.

OE4. The software shall interact with the system that has it installed (memory usage, battery percentage, etc.).

#### 4.4.3 Requirements for Interfacing with Adjacent Systems

#### 4.4.4 Productization Requirements

#### 4.4.5 Release Requirements

### 4.5 Maintainability and Support Requirements

#### 4.5.1 Maintenance Requirements

MS1. The main software holding the components together should easily be updated when it is needed.

MS2. The software should easily be modifiable just in case the user decides to add more functionality to the application.

#### 4.5.2 Supportability Requirements

MS3. The application should be able to be used in various platforms such as Windows, Macintosh, and Android.

MS4. The application should be able to be used in various devices such as a desktop computer, cellphone, laptop, etc.

#### 4.5.3 Adaptability Requirements

## 4.6 Security Requirements

### 4.6.1 Access Requirements

#### 4.6.2 Integrity Requirements

SR1. The application shall use encryption to protect transactions being sent from the user to the client.

#### 4.6.3 Privacy Requirements

SR2. The application should protect the identity of the person using the software. It shall not pass personal information (if any) to any unknown party without the explicit permission of the user.

#### 4.6.4 Audit Requirements

#### 4.6.5 Immunity Requirements

## 4.7 Cultural and Political Requirements

### 4.7.1 Cultural Requirements

CP1. The application should not use symbols, text, or media which debases a race, culture, or a political landscape.

#### 4.7.2 Political Requirements

CP2. The application shall show a disclaimer explaining any similarities that may arise to political symbol or figure is coincidental.

## 4.8 Legal Requirements

### 4.8.1 Compliance Requirements

LR1. The application should abide by any privacy and copyright laws.

LR2. The use of experts shall be done, whilst the owner permits its free usage.

#### 4.8.2 Standards Requirements

# A Division of Labour

| Contributions | | | |
|---|---|---|---|
| **Name** | **Student Number** | **Contribution** | **Signature** |
| Joshua | | | |
| Keyur | | | |
| Justin | | | |
| Bilal | | | |
| Shaad | | | |
| Abdullah | | | |