Software Requirements Specification

For

BusinessStats

Version 1.0

Ashwin Pilgaonkar, Akash Agarwal, Ayush Singhal, Jai Agrawal, Amit Jhajharia

Group

19.09.2016

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Group	16-09-16	Initial Version	1.0

1. Introduction

1.1 Purpose

The product created is an Android app that provides statistics for a tours and travel company for the owner and allows him to easily manage his business at his fingertips.

1.2 Document Conventions

TBD – To be determined.
Section/Subsection titles are in **bold.**

1.3 Intended Audience and Reading Suggestions

The document is intended for the Developers, Project managers, Testers and Documentation writers. Other involvement required to check the document as such:

Overall Description – In our case, we do not require a marketing team as such because the product is being custom made for a single person, exactly according to his needs.
System features – The system features should be understood by the testers (group) so that they can test thoroughly and give proper feedback to the developers (group).

External Interface Requirements – Since it is an app so the software developers (group) should know the requirements of the client to build upon. The user interface of the app should be user friendly. The group should understand external interface requirements to explain the client about the features of the app and how his requirement has been a priority for the group.

Nonfunctional and Functional Requirements – Software development.

1.4 Product Scope

This product will have features which are exclusively required by the client. The purpose of this app is for the client to have access to the statistics of his business on the go. As it is a smartphone app, this data will be accessible to him, whenever and wherever he requires it. The auto email feature also saves his time as he doesn’t have to check manually check for that particular agent and doesn’t have to manually type the email. He will also have access to all the flight and train schedules at his fingertips. Refer to the project scope paragraph within Purpose sub section for further information.

1.5 References

http://stackoverflow.com/
https://en.wikipedia.org/
https://developer.android.com/
https://material.google.com/

2. Overall Description

2.1 Product Perspective

This app is a new product which will act as a secondary system for the client. The client can access all the data this app gives him on his PC as well, but using the app is a lot faster and much easier. As it will be installed on his smartphone, he will be able to access this data whenever and wherever he requires. The app will also give him some new small, but very useful features. For more information, refer to **2.2 Product Features**.

2.2 Product Functions

The key features of the product are-

1. User will be able to select a year range and a category to sort by (age, country, gender, etc.)
2. The user will then be shown the number of bookings made in each year within the selected range (according to the category selected).
3. User will be able to visualize all the statistics in form of Graphs and Charts for easy and quick understanding.
4. User can also import flight/train schedules of multiple airlines/trains and the app will standardize them into a single spreadsheet file.
5. At the end of every year, an email will be sent automatically (user will be asked for confirmation) to the agent under whom, the most number of bookings have been made.

2.3 User Classes and Characteristics

In this case, there will be only one user for this product i.e. the owner of the firm (Odyssey Tours and Travels). It is being custom made for a single person.

2.4 Operating Environment

This application will run on any device that runs Android 5.0 (Lollipop) and above. There is no specific hardware requirement.

2.5 Design and Implementation Constraints

1. Android APIs do not natively support connecting to MySQL Databases. To do this, we would have to create a PHP script that connects to the database and returns data in JSON format. The app will then connect to the domain that hosts the PHP script, and process the returned JSON data.
2. Android APIs cannot parse spreadsheet files either. To do this, we would have to use the Apache POI library- which is a Java library that provides an API to manipulate different Microsoft Office file formats (doc, xls, ppt, etc). This library works only on Android SDK 21 and above so we had to set the minimum Android version to 5.0 Lollipop.
3. We would also need to use the JavaMail API for automatically sending emails in the background.

4. The file picker that can be implemented using the default Android APIs doesn't allow the users to select multiple files at a time so we have to implement a custom file picker for our app.

2.6 User Documentation

As the product is being custom made according to the requirement of only a single person, there is no need for a separate manual. There will be a help section implemented within the app which should be sufficient.
Documentation will be provided with the source code.

2.7 Assumptions and Dependencies

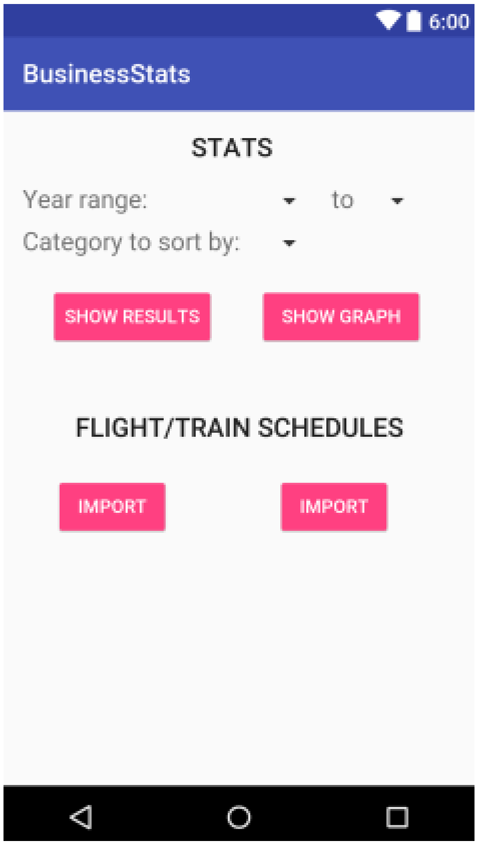
Assumption – The user has an Android device running Android 5.0 (Lollipop) or better.
Dependencies are all listed in **2.5 Design and Implementation constraints**.

3. External Interface Requirements

3.1 User Interface

1. The user will be able to input the year range and the category to sort records by.
2. Clicking the Show button will present the user with all the required data/statistics.
3. User will get an option to convert the given data to a bar graph/pie chart.
4. On the main screen there will be an option to import train/flight schedules of multiple airline companies/trains
5. Clicking the standardize button will standardize them into a single excel file and store it on the device's internal memory
6. There will be a help menu that will instruct the user on how to operate the app
7. A settings menu will be implemented for the user to input his Email address and password to enable the auto email generation feature. (Other settings options are TBD)

Initial UI Mockup



3.2 Hardware Interfaces

1. Internet capable Android device
2. Touchscreen/other input device
3. Display

3.3 Software Interfaces

The app will run only on Android 5.0 (Lollipop) and above. The app connects to a remote database and temporarily stores the information requested by the user, in a local SQLite database.
For more information, refer to **2.5 Design and Implementation Constraints**

3.4 Communications Interfaces

1. Internet connection will be used to connect to the remote database
2. The connection used will be secure
3. All network communication will be done by HTTPS

4. System Features

The system comprises of a client/server architecture. For the major services provided by this product, refer to **2.2 Product Functions**. The following subsections contain the functional requirements for the system feature.

4.1 Information regarding flights

4.1.1 Description and Priority

The user imports the flight schedules of different airlines which is in excel format. For each airline, the data is sorted in a different manner. The app will standardize all this data into a single excel file. It has medium priority.

4.1.2 Stimulus/Response Sequences

* User uploads the flight data of different airlines
* All the data is arranged according to predefined format of the table.
* The app standardizes the data of all flights into one file
* User get the full table in excel format.

4.1.3 Functional Requirements

* Standardization of flight/train schedules

4.2 Statistics of the business

4.2.1 Description and Priority

The system allows the user to select the year range and a category to sort by age, country, gender, etc. The number of bookings made in each year within the selected range are presented to the user. It can either be shown in the form of raw data, or a graph for better visual representation. It has high priority.

4.2.2 Stimulus/Response Sequences

* Select the year range and category to sort by age, gender and country, etc.
* System helps the user to analyze the statistics in form of raw data, graph and charts.
* At the end of every year, the system will send an email automatically (user will be asked for confirmation) to the agent under whom, the most number of bookings have been made.

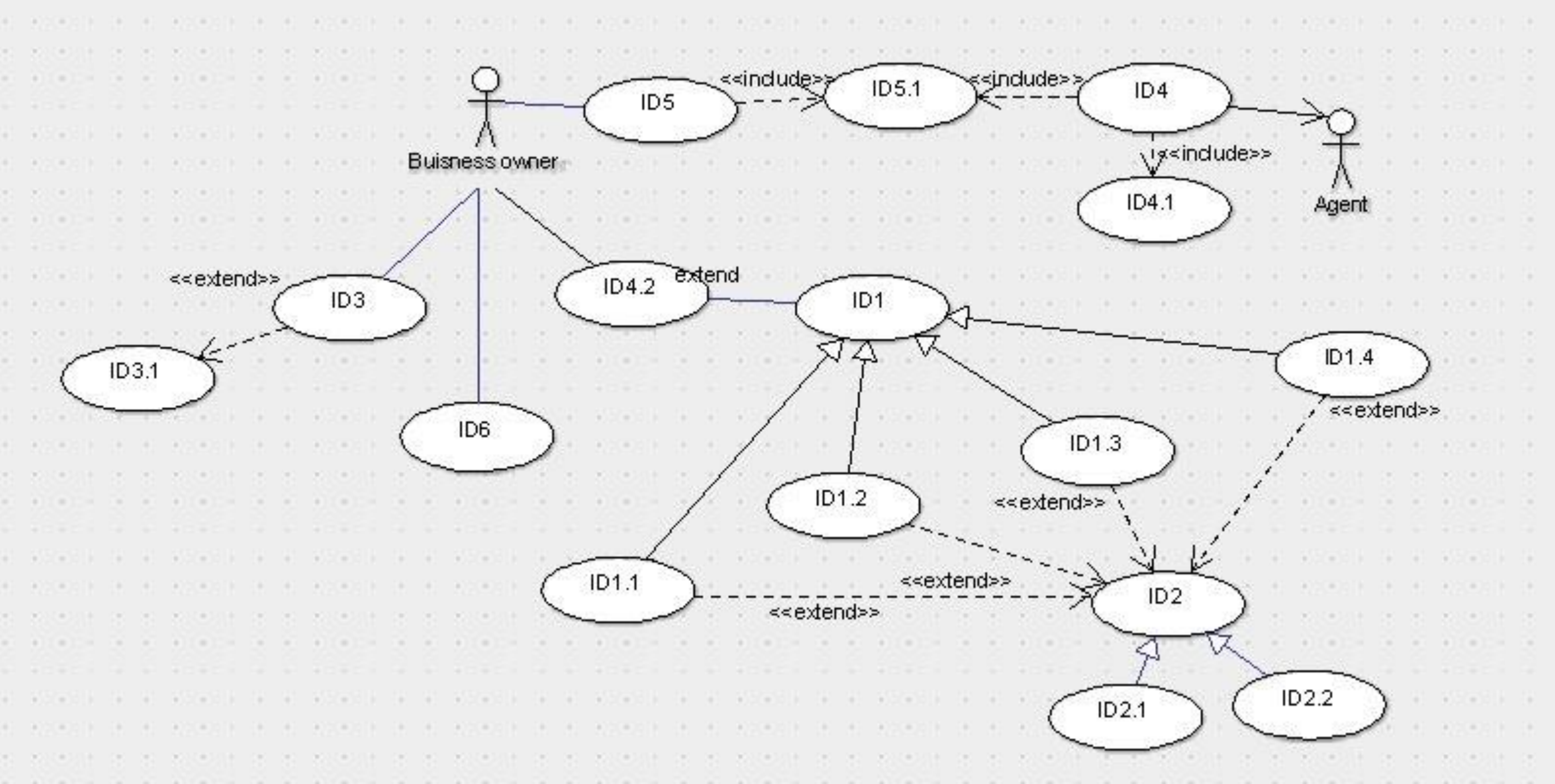
4.2.3 Functional Requirements

* Search Category
* Result
* Generation of email

Functional Requirement table:

ID	REQUIREMENT	DESCRIPTION
1	Search Category	Allows the user to select a year range and a category to sort by.
1.1	Year	Select year range
1.2	Agent	Sort by Agent
1.3	Age	Sort by Age
1.4	Gender	Sort by Gender
1.5	Country	Sort by Country

2	Result	System displays the result in the form of raw data, graphs or charts for easy and quick understanding.
2.1	Raw data for result	Result information
2.2	Graph	Visual information
3	Import	Import flight/train schedules of multiple airlines/trains.
3.1	Standardize	System will standardize the imported files into a single spreadsheet file.
4	Generation of email	At the end of every year, system will send an email automatically (user will be asked for confirmation) to the agent under whom, the most number of bookings have been made.
4.1	Email IDs	Retrieval of Email ID
4.2	Database	Main tour database that contains all the business related information
5	Settings Menu	User can provide his email address and password to enable the auto email feature.
5.1	Email Address and password	User can enter his Email address and password.
6	Help Section	Generic guide within the app for easy use of application



Use Case Diagram

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The performance of the product will primarily depend on the hardware of the user's Android smartphone/tablet.
To fetch certain data, the product will require internet connectivity (Wi-Fi). The time it takes to perform this task will be dependent on the internet connection strength and bandwidth.
The performance of other features (such as the processing of spreadsheet files) depends on how powerful the processor of the smartphone/tablet is.

5.2 Safety Requirements

TBD

5.3 Security Requirements

TBD

5.4 Software Quality Attributes

Availability
Internet access must be available when the remote database needs to be queried. For all other functions, internet access is not required and the app will function normally.

Usability
As the app performs relatively simple functions and is being tailor made to suit the needs of just one person, it will very easy to use. The UI will be made very user friendly and a help section will be provided within the app.

Maintainability
The source code will be documented thoroughly to enable easy maintenance.

Portability
The app is relatively simple and can be ported quite easily to other platforms provided that the external libraries used, are available for the platform being ported to.

Reliability
The system is designed to be reliable. It will fail (partially) only if the server hosting the database fails. The reliability also depends on the changes made in the future versions of Android OS.

Non Functional Requirements Table:

ID	***REQUIREMENTS***	***DESCRIPTION***
ID1	Performance	The performance of the product will primarily depend on the hardware of the user's Android smartphone/tablet.
ID2	Software Quality Attributes	Availability, Usability, Maintainability , Portability , Reliability
ID3	Safety	TBD
ID4	Security	TBD

5.5 Business Rules

This software is Open Source software.

Though this is being tailor made for a client, none of his confidential information (like username/password of his database) will be made public in the open source code.

6. Other Requirements

TBD

Appendix A: Glossary

Android SDK: It is a set of development tools used to develop applications for android platform.

API: (Application Programming Interface) is a set of functions and procedures which allow the creation of applications which access the features or data of an operating system, application, or other service.

Apache POI: It provides pure Java libraries for reading and writing files in Microsoft Office formats, such as Word, PowerPoint and Excel.

JSON: JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

Client/Server system: It is a distributed system in which some sites are client sites and other are server sites, all data resides at the server sites and all applications execute at the client sites.

Distributed System: It is a model in which components located on networked computers communicate and coordinate their actions by passing messages.

Statistics: The practice or science of collecting and analyzing numerical data in large quantities.

Appendix B: Analysis Models

Refer to **4. System Features** for the use case diagram.

Appendix C: To Be Determined List

1. Other Requirements
2. User Interfaces – Settings options
3. Security Requirements
4. Safety Requirements
5. Other Requirements