
Software Requirements Specification

For

BusinessStats

Version 1.0

Ashwin Pilgaonkar (U101114FCS052)

Group

15.09.2016

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 Product Scope	1
1.5 References	1
2. Overall Description	1
2.1 Product Perspective	2
2.2 Product Functions	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. External Interface Requirements	3
3.1 User Interfaces	3
3.2 Hardware Interfaces	4
3.3 Software Interfaces	4
3.4 Communications Interfaces	4
4. System Features	3
4.1 System Feature 1	Error! Bookmark not defined.
4.2 System Feature 2 (and so on)	Error! Bookmark not defined.
5. Other Nonfunctional Requirements	6
5.1 Performance Requirements	6
5.2 Safety Requirements	6
5.3 Security Requirements	6
5.4 Software Quality Attributes	6
5.5 Business Rules	Error! Bookmark not defined.
6. Other Requirements	6
Appendix A: Glossary	7
Appendix B: Analysis Models	7
Appendix C: To Be Determined List	8

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.

All titles are in **bold**.

Important terms have been highlighted/italicized.

1.3 Intended Audience and Reading Suggestions

This document is intended for the Developers, Project managers, Testers and Documentation writers.

The system features should be understood by the testers so that they can give adequate feedback to the developers.

Software developers should know the requirements of the client to build upon. The interface should be user friendly. The group should understand external interface requirements to explain the client about the features of the app.

Nonfunctional and Functional Requirements – Software development.

1.4 Product Scope

The purpose of this app is for the client to have access to the statistics of the client's business, on the go. 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.

1.5 References

Mentioned in Final SRS.

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.

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

There will be only one user- The owner of Odyssey Tours and Travels.

2.4 Operating Environment

This application will run on any smartphone/tablet that supports Android 5.0 (Lollipop) and above.

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

1. A help section will be implemented within the app.
2. Documentation will be provided with the source code.

2.7 Assumptions and Dependencies

It is assumed that the user has an Android device with version 5.0 Lollipop or higher. Dependencies are listed under **2.5 Design and Implementation constraints**.

3. External Interface Requirements

3.1 User Interfaces

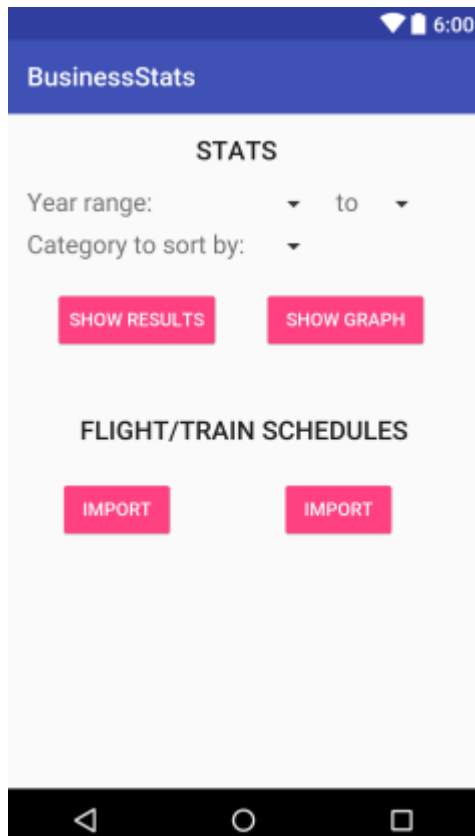
On opening the app, the user will be presented with a screen where he will be able to input the year range and the category to sort records by. On clicking the show button, the user will be presented with all the requested data. It can also be shown in the form of a bar graph/pie chart, based on the user's choice.

On the main screen, there will also be an option to upload train/flight schedules of multiple airline companies/trains and standardize them into a single spreadsheet file.

There will also be a help section that will act as a guide to the user whenever required.

A settings menu will also be implemented where the user can enter his email address and password to enable the auto email feature. Other settings options are yet to be decided.

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 the Android operating system v5.0 (Lollipop) and above. The app connects to a remote MySQL database and temporarily stores the information requested by the user, in a local SQLite database.

3.4 Communications Interfaces

Most of the features of the app will work offline so no communication interfaces are required. The only area where it is required is when connect to the database. For this, the HTTPS protocol will be used.

4. System Features

The system comprises of a client/server architecture.

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 order. The app will standardize all this data into one excel file.

4.1.2 Stimulus/Response Sequences

User uploads the flight data of different airlines
The app standardizes the data of all flights into one file
The standardized file is saved to the user's device in excel format

4.1.3 Functional Requirements

Standardization

4.2 Statistics of the business

4.2.1 Description and Priority

The user selects 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 displayed to the user.

4.2.2 Stimulus/Response Sequences

Select the year range and category to sort by age, gender and country, etc.

Processed data is displayed to the user.

At the end of every year, an email will automatically be sent to the agent under whom, the most number of bookings have been made.

4.2.3 Functional Requirements

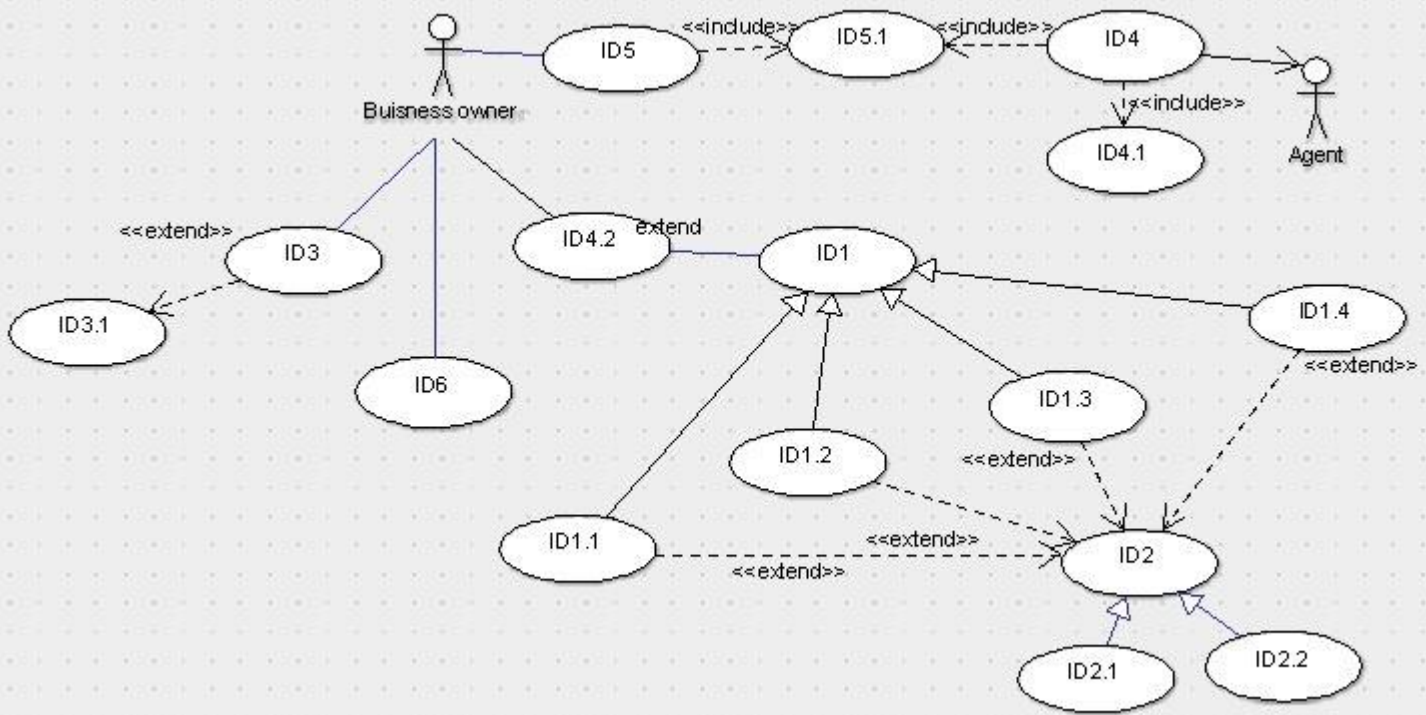
Search Category

Result

Auto generated email

Functional Requirement table:

ID	REQUIREMENT	DESCRIPTION
ID1	Search Category	Allows the user to select a year range and a category to sort by.
ID1.1	Year	Select year range
ID1.2	Agent	Sort by Agent
ID1.3	Age	Sort by Age
ID1.4	Gender	Sort by Gender
ID1.5	Country	Sort by Country
ID2	Result	System displays the result in the form of raw data, graphs or charts for easy and quick understanding.
ID2.1	Raw data for result	Result information
ID2.2	Graph	Visual information
ID3	Import	Import flight/train schedules of multiple airlines/trains.
ID3.1	Standardize	System will standardize the imported files into a single spreadsheet file.
ID4	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.
ID4.1	Email IDs	Retrieval of Email ID
ID4.2	Database	Main tour database that contains all the business related information
ID5	Settings Menu	User can provide his email address and password to enable the auto email feature.
ID5.1	Email Address and password	User can enter his Email address and password.
ID6	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 app will depend on the hardware of the user's Android smartphone/tablet. To query the remote database, the app will require an internet connection. The time it takes to perform this task will be dependent on the internet connection strength and bandwidth.

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

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 for easy maintenance.

Portability

The app can be ported to other systems provided the libraries used (or something similar) are available on the target platform.

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:

<u>ID</u>	<u>REQUIREMENTS</u>	<u>DESCRIPTION</u>
1	Software Quality Attributes	Availability, Usability, Maintainability , Portability , Reliability
2	Performance	Dependent on the user's device hardware
3	Security	TBD
4	Safety	TBD

5.5 Business Rules

TBD

6. Other Requirements

TBD

Appendix A: Glossary

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.

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.

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.

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

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

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. Business Rules
5. References
6. Safety/Security/Other Requirements