

Project Report Template

The following annotated template shall be used to complete the Software Requirements Specification (SRS) part of your project. The instructor must approve any modifications to the overall structure of this document.

1. Introduction

The introduction to the Software Requirement Specification (SRS) document of your project should provide an overview of the complete SRS document. While writing this document please remember that this document should contain all of the information needed by a software engineer to adequately design and implement the software product described by the requirements listed in this document

1.1 Purpose

What is the purpose of this mobile application.

1.2 Scope

This subsection should:

- (1) Identify the mobile application to be produced by name; for example, Host DBMS, Report Generator, etc*
- (2) Explain what the mobile application will, and, if necessary, will not do*
- (3) Describe the application of the mobile application being specified. As a portion of this, it should:
 - (a) Describe all relevant benefits, objectives, and goals as precisely as possible. For example, to say that one goal is to provide effective reporting capabilities is not as good as saying parameter-driven, user-definable reports with a 2 h turnaround and on-line entry of user parameters.*
 - (b) Be consistent with similar statements in higher-level specifications (for example, the System Requirement Specification), if they exist. What is the scope of this mobile application product.**

2. General Description

This section of the SRS should describe the general factors that affect 'the product and its requirements. It should be made clear that this section does not state specific requirements; it only makes those requirements easier to understand.

2.1 Product Functions

This subsection of the SRS should provide a summary of the functions that the mobile application will perform.

2.2 User Characteristics

This subsection of the SRS should describe those general characteristics of the eventual users of the product that will affect the specific requirements.

3. Specific Requirements

This will be the largest and most important section of the SRS. The customer requirements will be embodied within Section 2, but this section will give the requirements that are used to guide the project's software design, implementation, and testing. Each requirement in this section should be:

- *Correct*
- *Traceable (both forward and backward to prior/future artifacts)*
- *Unambiguous*
- *Verifiable (i.e., testable)*
- *Prioritized (with respect to importance and/or stability)*
- *Complete*
- *Consistent*
- *Uniquely identifiable (usually via numbering like 3.4.5.6)*
- *Follow MOSCOW approach.*

Attention should be paid to the carefully organize the requirements presented in this section so that they may easily accessed and understood. Furthermore, this SRS is not the software design document, therefore one should avoid the tendency to over-constrain (and therefore design) the software project within this SRS.

3.1 Functional Requirements

This section describes specific features of the mobile application project. If desired, some requirements may be specified in the use-case format and listed in the Use Cases Section.

3.2 Non-Functional Requirements

Non-functional requirements may exist for the following attributes. Often these requirements must be achieved at a system-wide level rather than at a unit level. State the requirements in the following sections in measurable terms (e.g., 95% of transaction shall be processed in less than a second, system downtime may not exceed 1 minute per day, > 30 day MTBF value, etc). Non-Functional Requirements are related to:

- *Performance*
- *Reliability*
- *Availability*
- *Security*
- *Maintainability*
- *Portability*

3.3 Other Requirements

Catchall section for any additional requirements.

4. Analysis Models

List all analysis models used in developing specific requirements previously given in this SRS. Each model should include an introduction and a narrative description. Furthermore, each model should be traceable the SRS's requirements.

4.1 Use Case Diagrams

Create a use-case diagram for your mobile application. Your design will be evaluated on completeness as well as level of thought, and attention to principles discussed in class. Your design should also provide good horizontal distribution of your project's functionality and allow for all described features to be developed, as much as possible.

4.2 Class Diagrams

Your class diagram should display all major classes in the system, each class's attributes (fields), relationships between classes, and named directed associational relationships with multiplicity between classes. Your design will be evaluated on completeness as well as level of thought, and attention to principles discussed in class. Your design should also provide good horizontal distribution of your project's functionality and allow for all described features to be developed, as much as possible.

4.3 Database Schema

Your Database Schema should display all major database tables in the system, each table's attributes (fields), relationships between tables, and named primary keys with foreign keys between tables.

5. Screen flows diagram and Screen design

It's fairly common the need to transition between multiple App Screens, we will explore in this section the way to manage those transitions. In this first part of this section, we will show how the screen flows are defined using State diagram to drive each screen transition as shown in figure 1.

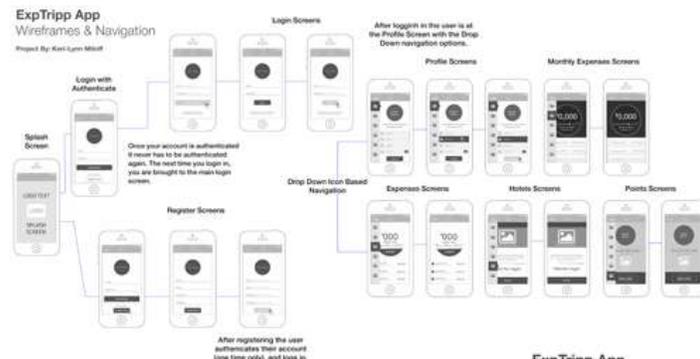


Figure 1. Screen flows diagram

Good navigation, like good design, is invisible. Applications with good navigation just feel intuitive and make it easy to accomplish any task. While there may be many options for navigating content within an app, we want to focus on screen layouts for primary and secondary navigation and interactive design principles. In this second part of this section, we will show how the main 2 screens off your mobile application are designed, as shown in figure 2.

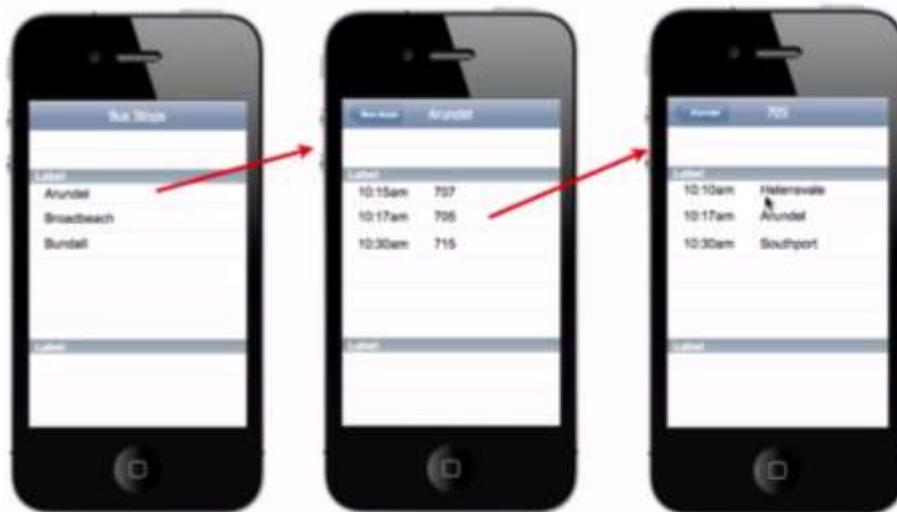


Figure 2. Screen design