Software Requirements Specification

for

SDMS Application

Prepared by Sanjeevani Reddy Vishaka

Software Engineering Project

November 17th, 2014

Table of Contents

Table of Contents 2

Revision History 2

1. Introduction 3

1.1 Purpose 3

1.2 Document Conventions 3

1.3 Intended Audience and Reading Suggestions 3

1.4 Project Scope 3

1.5 References 3

2. Overall Description 3

2.1 Product Perspective 3

2.2 Product Features 3

2.3 User Classes and Characteristics 3

2.4 Operating Environment 4

2.5 Design and Implementation Constraints 4

2.6 User Documentation 4

2.7 Assumptions and Dependencies 4

3. System Features 4

3.1 System Feature 1 4

3.2 System Feature 2 5

3.3 System Feature 3 5

3.4 System Feature 4 5

3.5 System Feature 5 6

4. External Interface Requirements 6

4.1 User Interfaces 6

4.2 Hardware Interfaces 6

4.3 Software Interfaces 6

4.4 Communications Interfaces 6

5. Other Nonfunctional Requirements 7

5.1 Performance Requirements 7

5.2 Safety Requirements 7

5.3 Security Requirements 7

5.4 Software Quality Attributes 7

6. Other Requirements 7

Appendix A: Glossary 7

Appendix B: Analysis Models 8

Appendix C: Issues List 9

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| SDMS V-1 | 10-3-14 | Initial Document | 1 |
| SDMS V-2 | 11-17-14 | Updated Document | 2 |

# Introduction

## Purpose

The purpose of this SDMS Application project is in the name. SDMS stands for Student Database Management System. The group’s goal for this Java/Android application is to provide a system that is usable for students to access information such as grades, courses, catalogs, forms, the academic calendar, and transportation information.

## Document Conventions

As of version 1, no conventions have been adopted.

## Intended Audience and Reading Suggestions

The intended audience for this document includes the developers and the class instructor.

## Project Scope

The software we are developing is intended for student and faculty use at some university. Troy University is being used as an example, but the project itself will reflect a general setup that could be implemented at any university. It will be a simple application that provides access to course information, shuttle schedules, parking maps, and an academic calendar.

## References

**No references yet.**

# Overall Description

## Product Perspective

This particular project was designed with the current Troy University student iOS and Android app in mind. We intend to make our application function in similar ways to the existing application, but we are also going to add more features that we thing students would find useful.

## Product Features

Overall, this application will feature five distinct functions: a student search portal, shuttle schedules, parking maps, an academic calendar, and a course catalog.

## User Classes and Characteristics

The intended user classes for this project are students and faculty. Students are the priority because the functions are aimed toward students’ needs, but faculty will find some areas helpful as well.

## Operating Environment

Our end goal is for this project to run with Java and operate on Android devices. We have yet to decide if we want it to be phone- and tablet- compatible or just simply phone-compatible.

## Design and Implementation Constraints

Because of the limited experience the group has working with security, the students using the application will be able to do things like view fees owed on their accounts, but they will not be able to pay those fees from the application. An interface similar to BlackBoard is ideal for the courses, calendar, and grades, but will be limited to how well the page works with the testing student database in the end.

## User Documentation

Documentation from the Google Code page and Google Drive files will be submitted along with meeting minutes, class diagram, and use case diagram.

## Assumptions and Dependencies

The only thing that could possibly affect the project is programming across different platforms. We are working with both Windows and Mac operating systems, so it is possible that things can get mistranslated in the process of the project.

# System Features

The list of basic features that will be used in this project and how they will connect to the smaller features within. It should be noted that while the application will have five pages at the beginning, each page is going to have its own extra functions within once the user decides what they want to do on that page.

## Student Search Portal

3.1.1 Description and Priority

This Student Search Portal is of the highest priority because it is the basis of the entire project. We cannot have a student management database without some sort of student database. This will provide the application user with a search box. From there, students can be selected and their current grades and courses.

3.1.2 Stimulus/Response Sequences

First, the user will select Student Search from the main menu. Upon pressing the button, they will be presented with a search box where they will be prompted to enter a name. The user will be redirected to the student’s page (given the student exists in the database) where courses and grades can each be views from their own separate page.

3.1.3 Functional Requirements

REQ-1: Search box

REQ-3: List view where items on the list, when selected, will redirect to a new page

## Shuttle Services

3.2.1 Description and Priority

The Shuttle Services section will ideally provide a detailed list of the shuttle run times and pick-up locations. This is another higher priority section because it is something that students need, but do not have access to in the Troy University application.

3.2.2 Stimulus/Response Sequences

First, the user will select Shuttle from the main menu. Upon pressing the button, they will be brought to a list view page that displays the school’s shuttle schedule as well as a button that, when pressed, will display a map of shuttle pick-up locations.

3.2.3 Functional Requirements

REQ-1: List view

REQ-2: Map of shuttle pickup location

## 3.3 Academic Calendar

3.3.1 Description and Priority

The Academic Calendar section will give the user a list of events scheduled for the semester. This list will include events such as official start days for classes, payment days, registration days, drop days, and due dates for any additional important information like intent to graduate forms. This is a high priority section.

3.3.2 Stimulus/Response Sequences

First, the user will select Academic Calendar from the main menu. Upon pressing the button, they will be brought to a list view page that displays all of the important dates for the semester.

3.3.3 Functional Requirements

REQ-1: List view

REQ-2: Calendar information

## 3.4 Course Catalog

3.4.1 Description and Priority

The Course Catalog section is intended to provide the user with a list of courses offered at the Troy campus of Troy university. For this project, we will only display Computer Science courses to demonstrate the application. This is a medium priority page.

3.4.2 Stimulus/Response Sequences

First, the user will select Course Catalog from the main menu. Upon pressing the button, they will be brought to a page that simply displays all of the Computer Science courses offered at the Troy campus.

3.4.3 Functional Requirements

REQ-1: List view

REQ-2: Catalog information

## 3.5 Parking Maps

3.5.1 Description and Priority

The Parking Maps section will simply have buttons to display parking maps. Each button will link to an image view page with the map and legend. This is a medium priority page.

3.5.2 Stimulus/Response Sequences

First, the user will select Parking Maps from the main menu. Upon pressing the button, they will be brought to a page with one or two buttons. The buttons, when pressed, will bring the user to the corresponding page for the specific maps.

3.5.3 Functional Requirements

REQ-1: Buttons

REQ-2: Maps of Troy University campus and Parking

# External Interface Requirements

## User Interfaces

The project will incorporate multiple user interfaces, all coming from the main app page. The user interfaces will consist of buttons, list views, search boxes, and image views.

## Hardware Interfaces

The application will be developed for Android phones in the beginning. An android phone running Froyo or lower will suffice to support the application.

## Software Interfaces

This application is intended to work with Android OS Froyo or anything lower than it. If we get to it, the databases can be imported and implemented through MySQL, though that is not the highest priority.

## Communications Interfaces

The use of electronic forms will be implemented, as well as a network connection through which to access all of the information. BlackBoard integration is on the drawing board if we get far enough to implement it.

# Other Nonfunctional Requirements

## Performance Requirements

The application will run as it is intended. Other than the standard requirements needed to run the application, no further performance requirements are intended to be added. This is for a basic user experience.

## Safety Requirements

There are no real safety risks with this application. No sensitive information is provided.

## Security Requirements

Student fee information and student courses and grades will be available on the app. No other information (that would be more sensitive) will be here.

## Software Quality Attributes

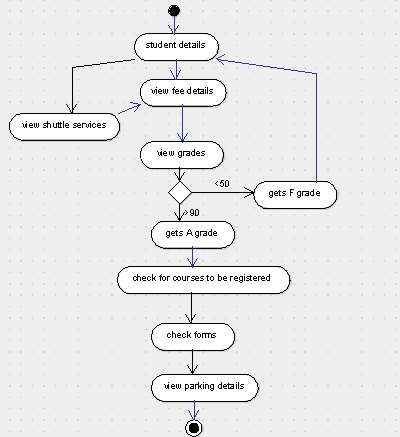
We aim to provide an application that is easy to use, simple to navigate. The application, if it were a full-blown official application for the university, would provide up to date calendars, course catalogs, and grades of the students.

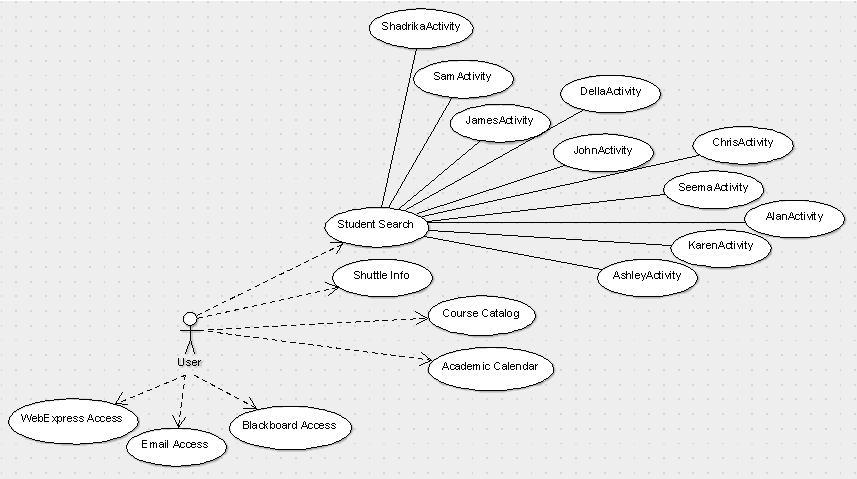
# Other Requirements

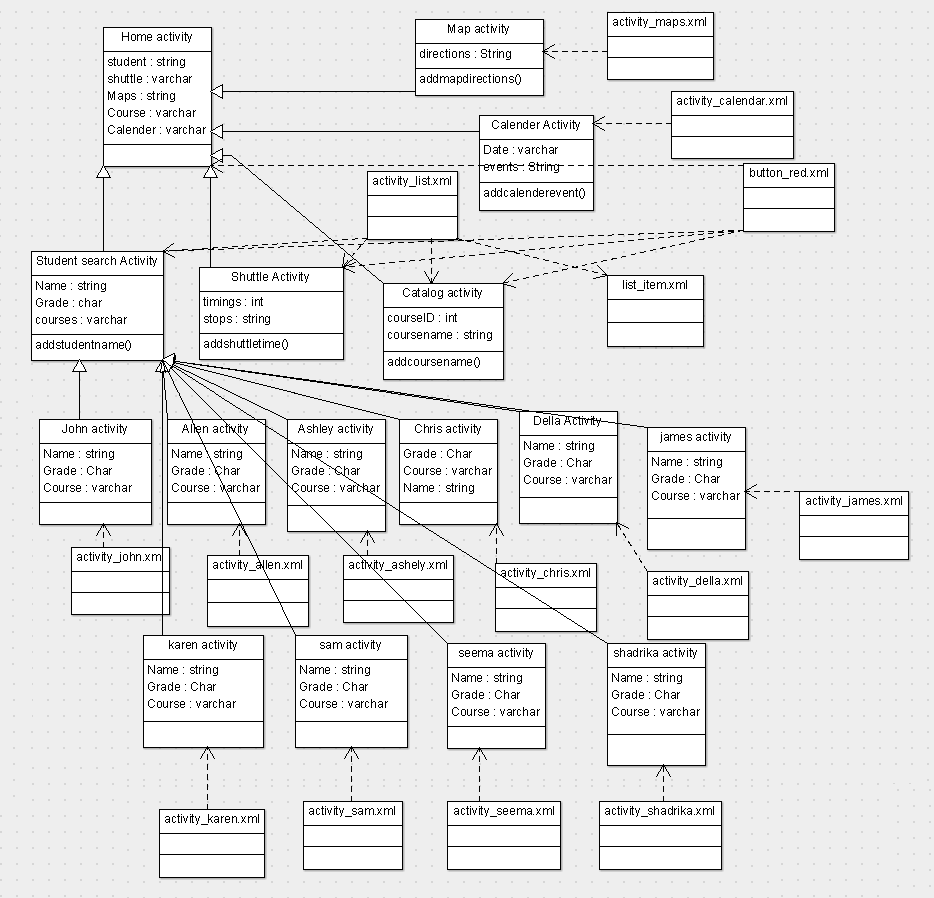
No other requirements as of this moment.

.

Appendix A: Analysis Models

**





Appendix B: Issues List

* Decision to include alerts in the application for dates on the calendar.
* Decision to include alerts for the shuttle schedule.
* Decision to include Blackboard integration.