CSUSB Student Advising
Mobile Application

Software Requirements Specification

Revision 2.2

March 8, 2013

Prepared by:
Dylan Allbee
Michael Schenk

Advisor:
Dr. Arturo I. Concepcion

Contents

1 Introduction

1.1 Purpose

This software requirements specification is intended to provide a complete and working description of the CSUSB Student Advising mobile application. It contains both an outlining of the software to be made as well as detailed, specific requirements. This document's expected audience is the faculty of CSUSB's Student Advising department; the software engineering team responsible for the implementation of the mobile application, as a reference; and Dr. Concepcion, for the purpose of review and approval.

1.2 Scope of the Project

The software to be designed will be a mobile application for the use by the students of CSUSB. The application will be designed with the primary goal of assisting students.

Specifically, the application will include a number of tools to assist in assessing progress at CSUSB, information that is currently only available at advising.csusb.edu, course requirements for specific majors from the yearly catalog, and the ability to contact not only student advisors but various faculty for assistance.

The mobile CSUSB Student Advising application is also to assist students in receiving tutoring information, class schedules, and pertinent forms.

The application will attempt to reflect the tools and information currently listed on the CSUSB Academic Advising web site.

1.3 Definitions, Acronyms, and Abbreviations

AES-192 Advanced Encryption Standard
: An encryption algorithm based on the Rijndael block cipher. Approved and recommended for high security usage by the NSA.

Android
: A Linux-based operating system designed primarily for touchscreen mobile devices such as smartphones and tablet computers.

Client
: CSUSB Student Advising

CSS Cascading Style Sheet
: A language designed for creating the graphical style of a web page.

DB Database
: Implies MySQL as the database engine & language

HMAC Hash Based Method Authentation Code
: A method of determining whether or not data has been altered.

HTML Hyper Text Markup Language
: The dominant description language for the frontend display of web pages.

iOS
: A mobile operating system developed and distributed by Apple Inc. for use on iPhone and iPod devices.

JavaScript
: A programming language that allows execution of application logic by the web browser.

MB Megabyte
A unit of memory, equivalent to roughly 1 million bytes.

MySQL DB My Server Query Language
A database engine and language developed by Oracle Corporation for storing various types of data.

PhoneGap
A framework for building cross platform mobile applications.

PHP PHP: Hyptertext Preprocessor
A programming language purposed for developing application logic for web sites.

Python
A programming language that excels in quick development of applications.

QA Quality Insurance
The team responsible for verifying that the application produced by the software team meets its requirements.

RAM Random Access Memory
A type of computer memory used for fast access during program usage.

SDK Software Development Kit
An environment designed for the purpose of developing a specific type of application, containing tools and code libraries that facilitate development.

TCP/IP Internet Protocol Suite
A standard networking protocol for end-to-end connectivitity. Requires some overhead.

Unix Socket
A data communications endpoint for exchanging data between processes executing within the same system.

Varnish, Cache
Data that is stored either locally on the mobile device or in the server's memory for the purpose of fast access.

Webkit
A software layout engine designed to allow web browsers to render pages in accordance with the HTML5 specification.

1.4 References

IEEE Std 830-1998 Software Requirements Specification
Student Advising SRS 1.0 (iteration #1)

1.5 Overview

The rest of the document is split into two sections, the first provides an overall description of the mobile application. This includes its interfaces, accessibility and usability requirements, user characteristics, constraints, and dependencies. The final section contains detailed requirements regarding the actual implementation of the mobile application.

2 Overall Description

2.1 Product Perspectives

The following features and characteristics will be added to the application:

- Advisor information
- A forms page for access to applicable CSUSB forms
- School calendar with important dates
- Major, minor, and general education course descriptions
- Tutoring information
- Campus computer lab information
- Access to CSUSB Course Catalog
- A full, searchable class schedule

The final release will also include these features from SRS 1.0

- Android compatibility
- General education information
- A GPA Calculator
- A contact form
- Campus tutoring locations
- Campus computer lab information
- Access to CSUSB Course Catalog
- SOAR Information
- Study tips
- Time management information
- Internship information
- A full, searchable class schedule

The client has agreed that the Course Catalog may not be able to be implemented given our time constraint.

2.1.1 System interfaces

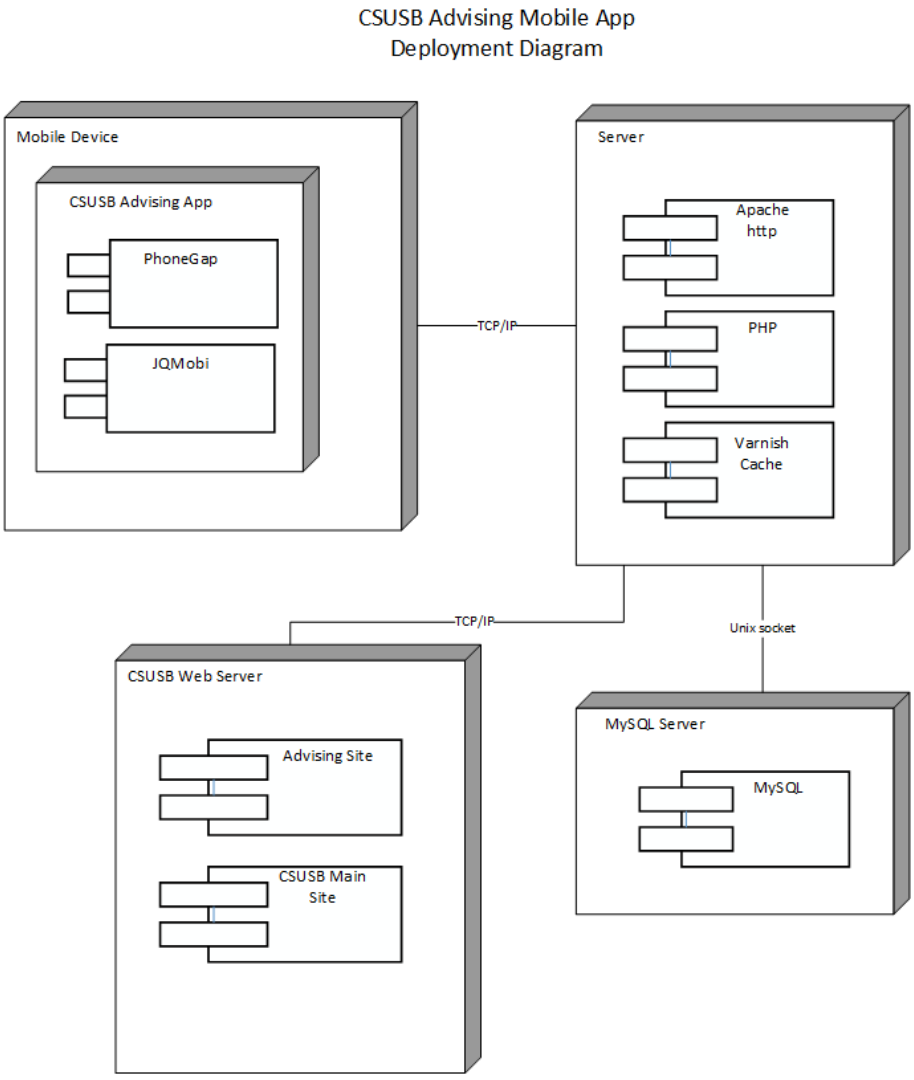

Figure 1: CSUSB Advising App Deployment Diagram

We will add a Varnish cache server that will speed up page loads and reduce the memory foot print.

The mobile application software will have most of its logic run on a remote server. The remote server will fetch new data from the CSUSB Advising website on a daily basis. This data will then be cached and sent to the user's mobile device upon request. The remote server will be handling program logic with PHP, serving cache with Varnish, and storing data in MySQL.

The data sent to the mobile device will be in the form of HTML & JavaScript. The mobile device will render this data using PhoneGap, a rendering engine based on WebKit. Images will be stored and cached on the device for maximum performance.

2.1.2 User interfaces

We will add the ability to swipe the screen to control the side slide-out menu. We will add "about us" option to the main menu. Additionally, we will alphabetize the side menu content.

For a unified identity with other CSUSB applications, the user interface will display the same CSUSB logo at the top as the CSUSB Mobile android application, as well as the same blue bottom menu.

The interface will have a toggleable sidebar that contains additional links for the application. The sidebar will have the same look and feel as the current CSUSB Mobile application.

Buttons in the application are to be developed in close accordance with the client. Buttons are to be clean, sharp, and to quote the client, 'Pretty'.

2.1.3 Communication interfaces

There are three primary communications that are established.

Server to CSUSB Website
TCP/IP Connection over port 80

Server to MySQL database
Unix socket

Mobile application to Server
TCP/IP Connection over port 80

2.1.4 Memory

We are adding the following memory constraints to the application for ease of download and performance.

- The application will not exceed 50MB of RAM.
- The application will use no more than 5MB of disk space.

The actual values will vary depending on the operating system and phone.

2.2 Use Cases

The following diagram is a complete description of the application use cases.

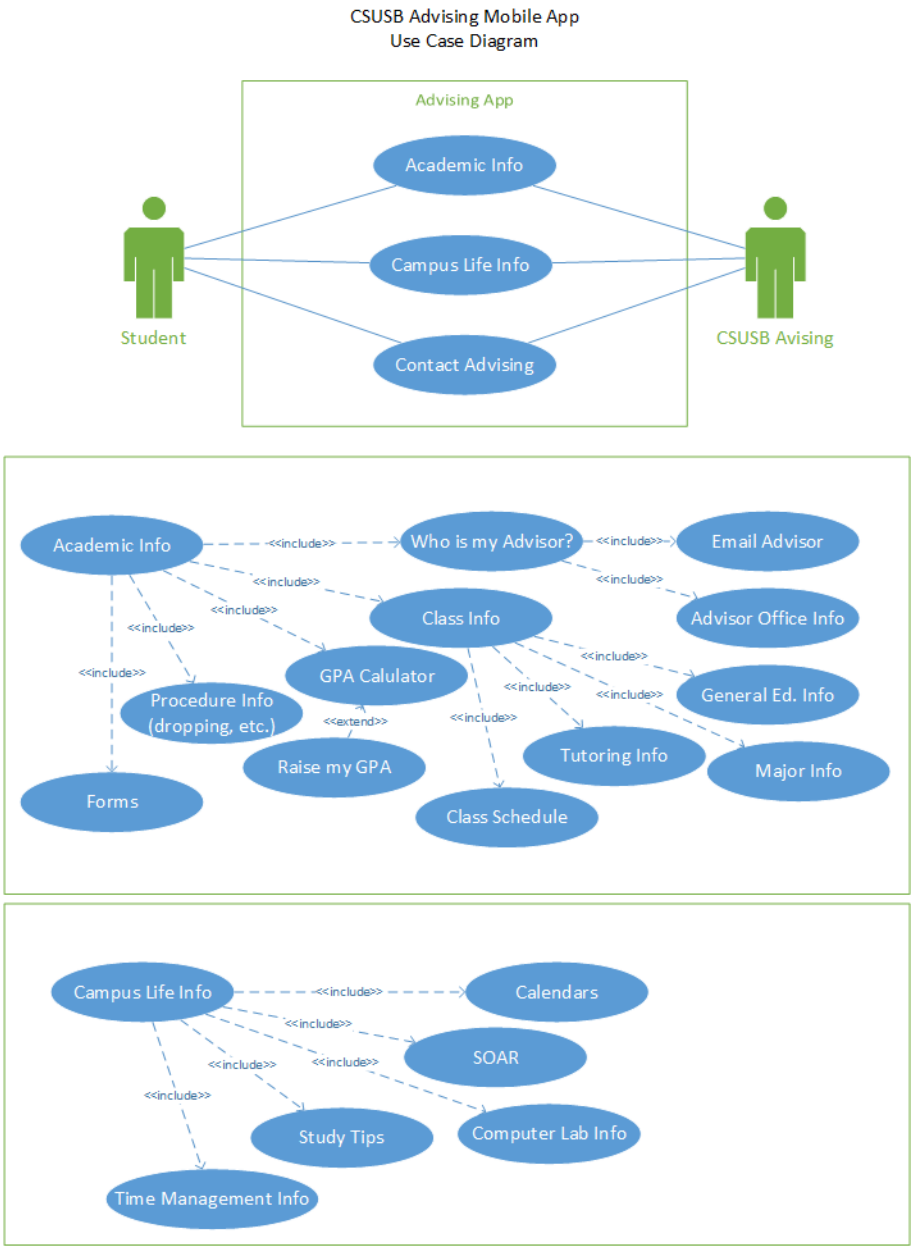

Figure 2: CSUSB Advising App Use Case Diagram

2.3 User Characteristics

A typical user is expected to be able to:

- Read the content on the page, either visually or with the assistance of a dictation device.
- Touch through navigation
- Type with his or her mobile device, or use a speech to text device.
- Use a search form

2.4 Constraints

2.4.1 Operating Systems

Normal usage of the device will require a mobile application: either tablet or phone. Supported operating systems will be Android 2.1+ and there will be preliminary support for iOS 4.0+. Older versions of these operating systems will not be supported.

2.4.2 Connections

The application will be limited to 2G or better connections. Maximum performance will require a 3G connection or better. Wifi will be fully supported.

2.4.3 Platforms

The application must be written using multi-platform technologies for ease of upgrading to new platforms, as well as for the purpose of supporting new platforms in the future.

2.4.4 Content Creation

Data input should be fetched automatically, and no manual input shall be required for the operation or updating of the application's content.

2.4.5 Accessibility

The application must conform to accessibility standards as outlined by w3c
http://www.w3.org/standards/webdesign/accessibility

2.5 Assumptions and Dependencies

We now require the additional dependencies.

- Python 2.6
- Varnish Cache 3.0+

The application will also depend on the previously listed technologies from SRS 1.0:

- Android 2.1+ or iOS 4.0+
- PHP 5.4+
- MySQL 5.5
- JavaScript

These technologies are to be installed, configured, and maintained by the CSE 455 Server Team.

3 Specific Requirements

3.1 External Interface Requirements

3.2 User interfaces

The following screen shots show the new design of the user interface:

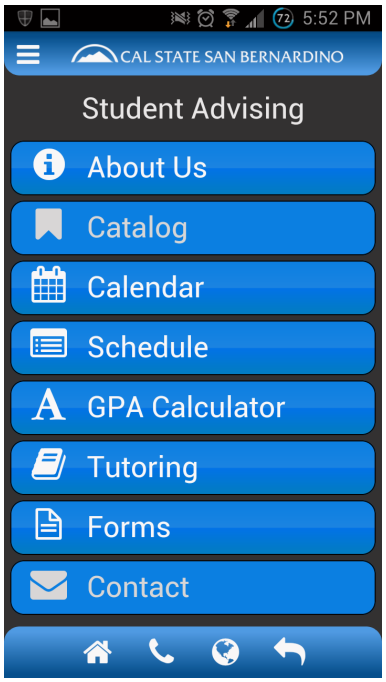

Figure 3: Application Front Panel

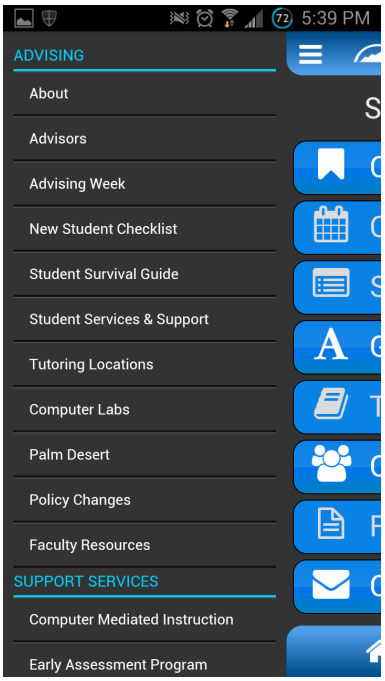

Figure 4: Side Menu

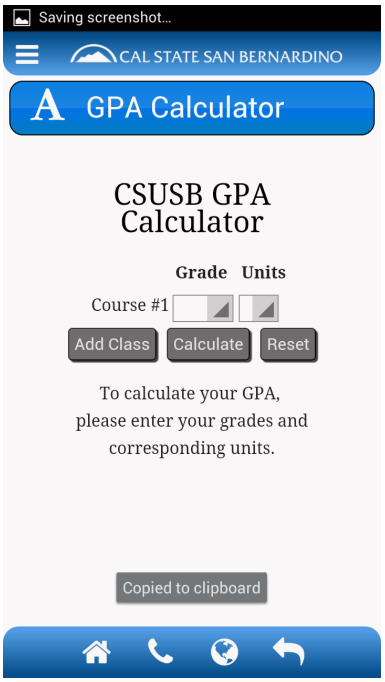

Figure 5: GPA Calculator

3.3 Hardware interfaces

An Android 2.1+ or iOS 4.0+ compatable device will be required to run the application. iOS 4.0+ support will be preliminary.

3.4 Functional Requirements

The following functional requirements shall be added to the application:

1. School calendar with important dates
    - PDF Download from current year's calendar.
    - Selectable years and months.
2. Calculate grades needed in order to raise GPA
3. General education information
4. Major, minor, and general education course descriptions
    - Selectable degree from a list
    - Download's PDF of the catalog for that specific degree
    - General Education included as a 'degree'
    - Full searchable class schedule for recent and future years
    - Mimics the currently existing CSUSB Mobile class schedule function
5. *Class schedule
    - Full searchable class schedule for recent and future years
    - Mimics the currently existing CSUSB Mobile class schedule function

The final release will also include these requirements, as outlined in SRS 1.0

1. Advisor information
2. General education information
3. School calendar with important dates
    - PDF Download from current year's calendar.
    - Selectable years and months.
4. A GPA Calculator
    - Calculate GPA based on courses already taken
5. A contact form
    - Drop down 'I need to...'
    - Drop down for a department to choose from
    - Text box to enter question
    - Text box to enter a reply-to address
    - Submit button
6. Campus tutoring locations
7. Campus computer lab information
8. Information on withdrawls
9. Information on leave of absence

10. Incompletes, Course overloads, and Simultaneous enrollment
11. SOAR Information
12. *Study tips
13. *Time management information
14. *Internship information

Functions marked with * are not guaranteed to be completed by end of quarter.

3.5 Performance Requirements

The application will be up within 5 seconds across all devices.

All application functions will respond within 50 milliseconds. Examples of application functions are pressing buttons, swiping the sidebar, and keyboard input.

Non local functions, such as loading a calendar or searching the class schedule, will be limited by both CSUSB's network and the mobile device's connection. Various caching techniques will be used to mitigate these limitations, but no response time can be guaranteed.

3.6 Design Constraints

Graphics will be small enough to be transferred over a 2G mobile connection in under 10 seconds total time, and over a 3G mobile connection in under 3 seconds total time.

Graphics will be designed to scale for both tablet and phone sized devices for an equal experience.

The application is to look and feel the same as the CSUSB Mobile application for the purpose of unity.

3.7 Security Requirements

Several measures will be taken to prevent user information compromise.

Cookies
: Any cookies sent to the mobile user will be AES-192 encrypted and run through an HMAC mechanism for maximum security. Cookies will only contain data to a random, unique id in the database that corresponds with any stored data.

Cross Site Scripting
: All output data will be run through HTML and JavaScript filters.

Cross Site Request Forgery
: Temporary, unique, and random tokens will be used for all submitted forms.

SQL Injection
: All code that works with the database will be run through prepared statements which validate and filter input before being released to the database.

Code Review
: Application code will be inspected by a QA team to look for any potential security vulnerabilities.

3.8 Document Approval

Document must be approved by:

- Dylan Allbee
  Project Manager
- Dr. Arturo Concepcion
  Project Advisor
- Ebony Staten
  Academic Advising