Campus Safety

Mobile Application

Software Requirements Specification

Revision 1.1

January 26, 2015

Prepared by: Joseph Callahan **Project Manager**
Raymundo Mejia **Asst. Project Manager**
Sean Butler
Ryan Stegmann
Qianqian Wang
Raul Diaz **QA**
Jorge Gamez **Server**
Victor Enriquez

Advisors: Dr. Arturo Concepcion
Cory Brown

UI/UX Designer: Heather Malone

Table of Contents 4

List of Figures

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the "Campus Safety" mobile application. It will illustrate the purpose and complete declaration for the development of application. It will also explain the features, what this application will and will not do, and the interfaces of this mobile application. This document is intended for the development team of the "Campus Safety" application and will be proposed to the client, Scott Kovach, Support Services Supervisor of the CSUSB Police Department. This project will be done as part of the requirement of CSE455 (Software Engineering) under Dr. Concepcion.

1.2 Scope of the Project

This software will be designed as an Android-only based mobile application. The application will be designed specifically to empower students to collaborate with police. It provides students with links of important phone numbers, such as, emergency contacts. It also has an anonymous messaging feature for reporting to the police suspicious behavior, unsafe situations, or any other safety concerns. In this iteration, we will not do the crime graphics, student location page, and 9-11 page. The application should be free to download from either a mobile phone application store or similar services.

1.3 Definitions, Acronyms, and Abbreviations

Android – A Linux-based operating system designed primarily for touchscreen mobile devices, such as, smart phones and tablet computers.

Android Studio – The IDE of choice for our development team.

Apache Web Server - The Apache HTTP Server, colloquially called Apache, is the world's most widely used web server software. Originally based on the NCSA HTTPd server, development of Apache began in early 1995 after work on the NCSA code stalled.

API - Stands for Application Programming Interface, is a source code based specification with intended use as an interface by software components.

App – Short for applet refers to any software or program that runs on mobile devices such as cell phones, tablets, and Smartphone technology. The majority of the code resides on the device as Java Runtime Environment and the smaller portion of code is on a server that can be downloaded to a client quickly.

CSUSB - California State University, San Bernardino. This is the university at which the application is being developed.

FTPS - (also known as FTP-ES, FTP-SSL and FTP Secure) is an extension to the commonly used File Transfer Protocol (FTP) that adds support for the Transport Layer Security (TLS) and the Secure Sockets Layer (SSL) cryptographic protocols.

IDE - Is an Acronym for Integrated Development Environment. An IDE assists the programmer writing and debugging software.

IEEE - The Institute of Electrical and Electronics Engineers is a professional association with its corporate office in New York City and its operations center in Piscataway, New Jersey. The organization helps make standards for all industries affected by what their title implies.

iOS - is an acronym for the iPad, iPod, iPhone operation system that is used on Apple devices.

Java – Object Oriented Programming Language used in many mobile applications.

Java Development Kit (JDK) – Tools needed for programming in Java.

Javascript - is the programming language of the Web.

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. It is based on a subset of the JavaScript Programming Language, Standard ECMA-262 3rd Edition - December 1999.

MySQL – is the program we will use for database design.

PHP - is a server-side scripting language designed for web development but also used as a general-purpose programming language.

SDK - Stands for Software Development Kit. This tool allows for the creation of applications for certain software platforms.

SRS - Software Requirements Specification. A description of the requirements specifications needed for the Campus Safety application.

Talk Back - Talk Back is an Accessibility Service that helps blind and vision-impaired users interact with their devices. Talkback adds spoken, audible, and vibration feedback to your device. It is a system application that was pre-installed on most devices and is updated when the accessibility service is improved.

XML - Stands for Extensible Markup Language. XML is a markup language that defines a set of rules for encoding documents that are both machine and human read-able.

1.4 References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

M. Flowler, "UML Distilled: A Brief Guide to the Standard Object Modeling Language," 3rd Ed, Addison- Wesley, 2004.

Cory Brown, *Software Requirements Specification for "Campus Safety Mobile App".*

1.5 Overview

The remainder of this document includes two sections. The second section provides an overall description of the "Campus Safety" mobile application. It gives an overview of the functionality of the product. It includes interfaces, accessibility and usability requirements, user characteristics, constraints, and dependencies.

The third section provides the requirement specifications in detailed terms and a description of the different user interfaces. Different specification techniques are used in order to specify the requirements more precisely for different audiences.

2. Overall Description

This iteration of Campus Safety will build upon the last iteration that was not completed and had many bugs needed to be worked out. The way things work are subject to change but we intend to follow the previous specification and complete the features one step at a time.

2.1 Product Perspectives

We as a team only have the prospect of working with Android based phones for the time being as time is limited and we would like to have a completed project that the client can agree with. An iPhone (iOS) based phone will be looked at in future development. This will provide a great way for students to be able to get important information at any time of the day and be aware of what may be going on campus and plan according for the benefit of their safety. This also provides good feedback for the campus police as well as the benefit of the police working more efficiently in terms of notifying and easily getting the information into the hands of the students at any time because of mobile phones that are able to access the Internet and have applications for specific uses.

2.1.1 System Interfaces

The application will be using a mobile device. These mobile devices are smart phones which are able to access the Internet and make phone calls if necessary. There are two popular mobile devices such as the iPhone made by Apple and Android based phones which are created by a multitude of different businesses like Samsung for example. Our first attempt will only focus on the Android based smart phones.

The smart phone will need to interface with a server and database to store image and text data. The interface is meant to be stored on our client's server system that they will provide. The server will contain software such as a web-server and email client. There may be a need for a web interface or at least a simple way besides the emails to access the images though for the time being will be a simple listing that can only be accessed locally from our client's, not to be confused with the users, end.

The client has control over the server and data that is being received as well. There is not specific chain of command as of yet of who will be accessing the data but will be figured out when product is complete so in general we leave it as the Campus Police will be able to access the data and only the Campus Police.

Campus Safety Component Diagram

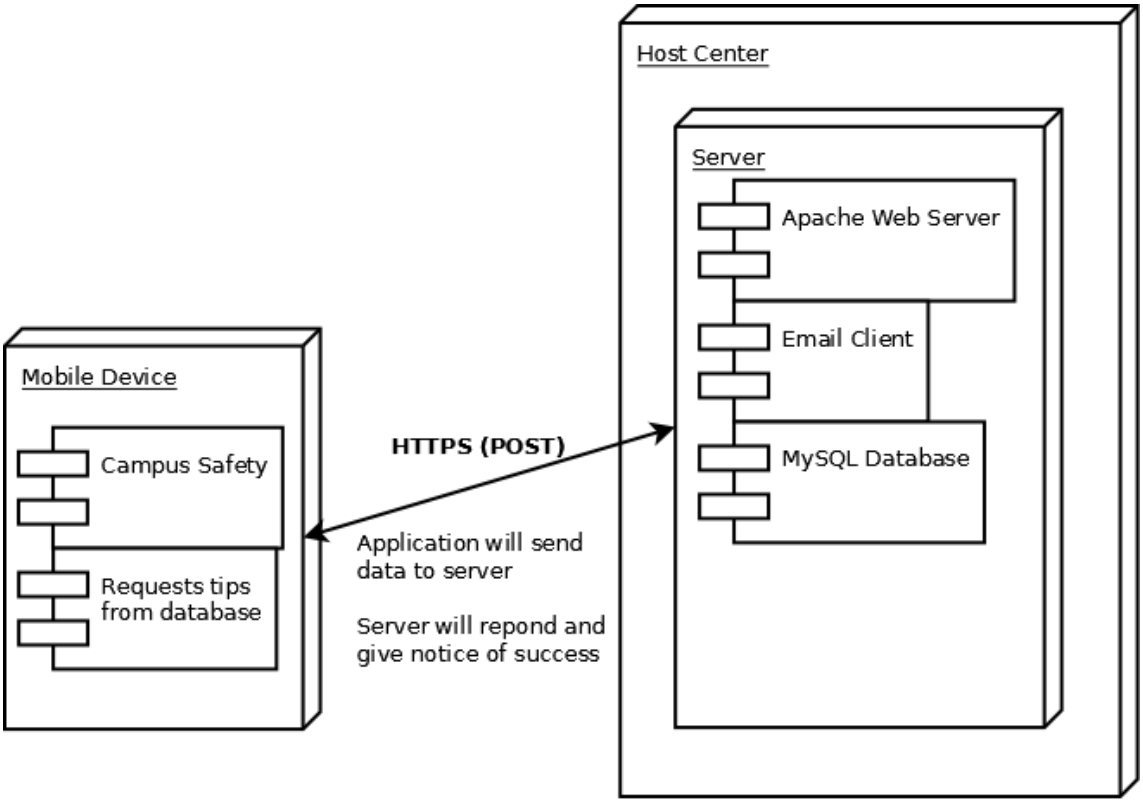


Figure 1: Component Diagram: This diagram displays the software needed on the server to host the app and interface with users.

2.1.2 User Interfaces

The following functional features will exist in the application.*

Splash Screen - Application boot screen to showcase the application. Set for a few seconds.

About Us page - To inform the user how to further contact CSUSB Police regarding non-emergency inquiries.

Resource page - To provide information to other resources regarding safety, law, and health.

9-1-1 page - The purpose of this page is to give a general description of when and when not to use 9-1-1 services.

Services page - To provide the user with additional information on police service's available on campus.

Anonymous Tips * - The user will be able to send anonymous tips to the Police Chief and detective on campus without any information of the user being exchanged.

Reports page - A link to the page of statistical reports at CSUSB.

Drop down menu * - A functional drop down menu with shortcuts to all screens.

Student Location page* - User will be able to request assistance from campus safety anywhere on campus. This will also list other reports and show a map with markers indicating location.

****These features and modifications are subject to change depending on priority and available time.**

2.1.3 Software Interfaces

- Android SDK – (API 16 to API 20)
- Java Development Kit (JDK)
- Android Studio
- Java
- JSON

2.1.4 Communication Interfaces

The anonymous tips, reports, and student locations pages would require 3G, 4G, or Wi-Fi communications. All network communication will use the HTTPS (Hyper Text Transport Protocol Secure) protocol to and from the server with POST data to keep the actual user as anonymous as possible and for basic encryption of data over the air until it has reached its host server.

The estimate amount of memory being sent over the network will be about 10KB of data which means the minimal require amount of bandwidth would be 80Kbps/0.0125Mbps. The average 3/4G bandwidth is about 2Mbps-6Mbps and Wi-Fi being 30Mbps on average. As you can see the app does not require a lot of data to be sent, and a submission can finish in less than a second. Normal mobile data plans should suffice.

2.1.5 Memory

The size of the app may be around 10MB with a RAM usage of about 2MB as well. Mobile devices now and days hold at least 1GB of memory and the app is not memory intensive as it does minimal calculation to be able to send a request over the 3/4G or Wi-Fi network.

2.1.6 Operation

The phone will operate in portrait mode and maintain the ratio of the mobile phone it is on. There are however many different sizes of phones and do we do not have enough time to test for all so we will settle on a 3-4” (inch) screen theoretical base for all other mobile devices to follow. We will also support medium to high definition (HD) resolution images for quality viewing of the application. Although it will work on a tablet there will not be time to support the layout.

There will most likely be need for downtime if new features should be discussed for a later time. There should be no need for downtime as updates to the server or the application itself should be as seamless as updating from the app store. Therefore, the application will run 24 x7 x 365.

2.1.7 Site Adaptation Requirements

The site will need to be able to host the server on campus. Computers may need to be setup, or a possible shortcut to the information made available only to the campus police. The server information should not be accessible outside of the campus and should for the most part only receive data from the mobile application.

2.2 Product Functions (Use-Case Diagram)

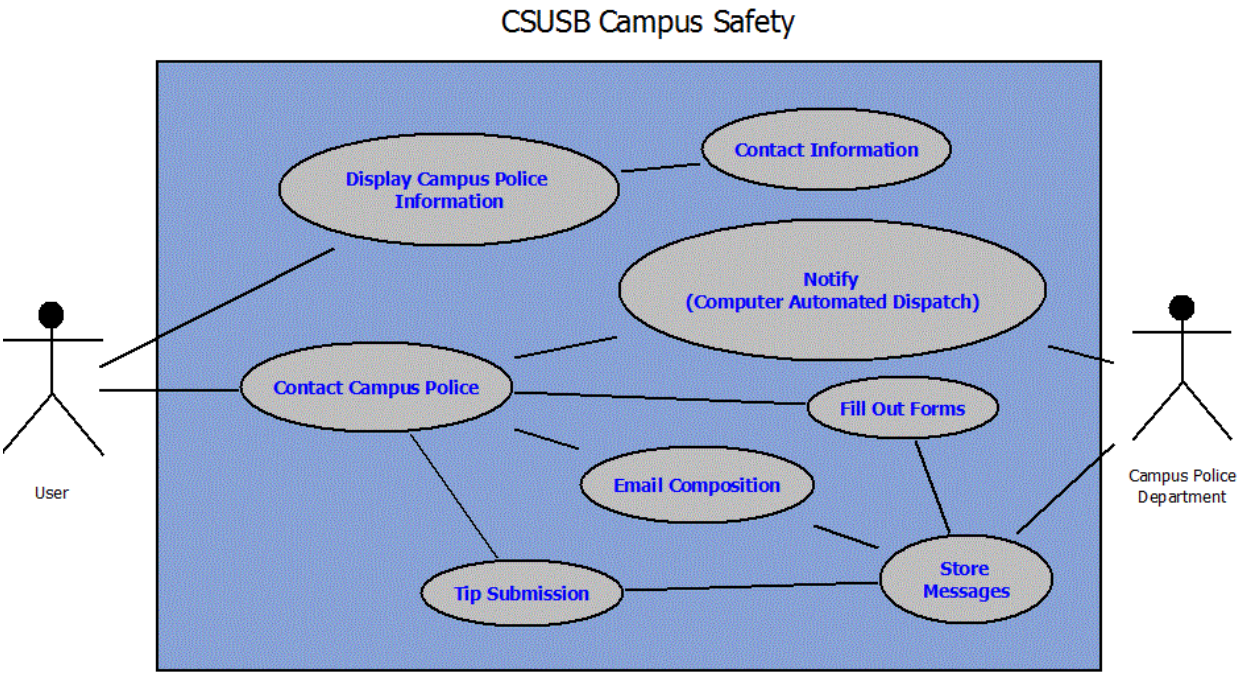


Figure 2: Use-Case Diagram for the Campus Safety App.

2.3 User Characteristics

- A typical contemporary student user is technically savvy in the use of mobile devices.
- A typical police officer is academy trained including various modes of telecommunications.

2.4 Constraints

2.4.1 Operating Systems

Usage of the device would require a mobile application running on either a smartphone or tablet. The planned supported operating systems include Android 4.0+. Older versions of this operating system will not be supported. There will not be preliminary support for any version of iOS or Windows Phone.

2.4.2 Connections

The application will be limited to a minimum requirement of a 2G connection. Usage of a 3G connection or better will be supported for peak performance. Wifi will be fully supported.

2.4.3 Platforms

The application is to be written in Java and is designed to optimally run on smartphone devices with Android 4.0+ installed. Tablet devices will also be supported.

2.4.4 Content Creation

User input shall be required for certain aspects of the application which include:

- Filling out request forms
- Tip submission
- Email Composition

2.4.5 Accessibility

The application is planned to be developed with the following:

- Talk Back
- Large Text
- Text-to-Speech Output
- Magnification Gestures

However, the former is not guaranteed to appear in any prototypes.

2.4.6 Limitations

Items that will limit the user's options:

- Incompatibility with devices running older Android versions
- No incorporation of Android Material Design language
- Updates to the Mobile operating system

2.5 Assumptions and Dependencies

The application will require:

- Android 4.0+
- One way connection to the CSUSB Computer-Aided Dispatch (CAD)
- Maintainability by the CSE 455: Software Engineering Server Team

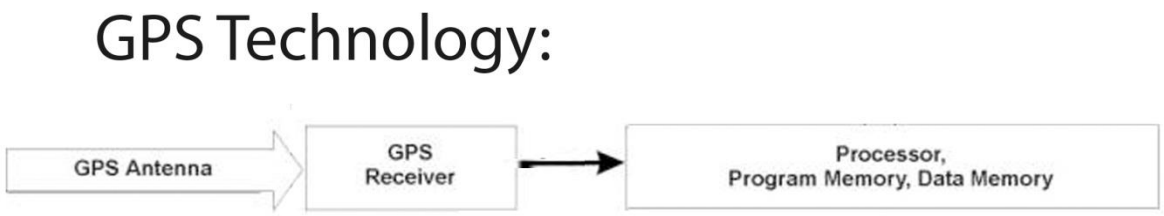
3. Specific Requirements

3.1.1 External Interface Requirements

The mobile device will connect via https through TCP ports decided by the server team and will automatically query the database on the client's end.

3.1.2 Hardware Interfaces

We are aiming for Android ADK 2.3 but targeting up to Android 4.4



3.1.3 Software Interfaces

Technologies/Languages required:

- Server Side:
  MySQL – Database
- PHP/Javascript – connection to database
  (http get/post requests from application)

Application:

- Java (Android SDK + APIs provided by google)

3.1.4 Connection Interfaces

This application is browser independent.

3.2 Functional Requirements

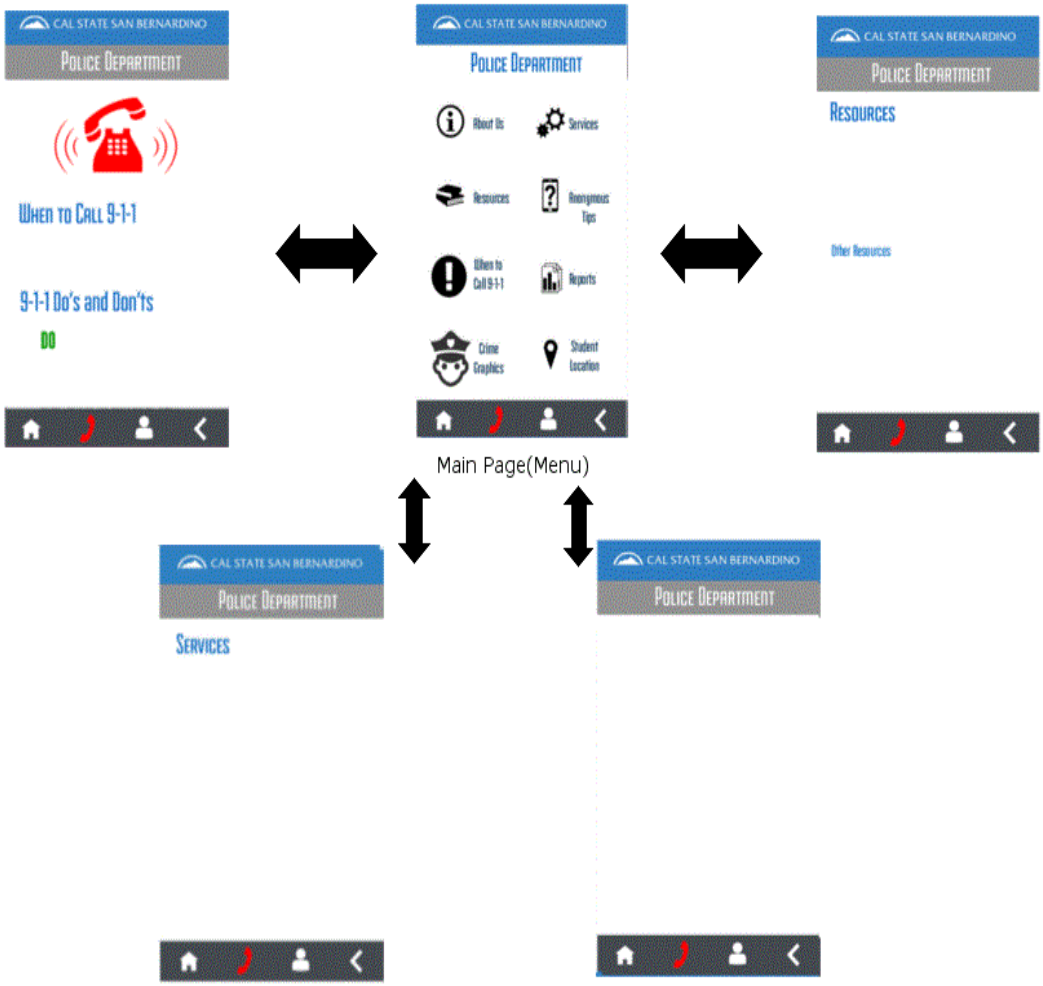


Figure 3: User Interface examples.

Connection of Pages Diagram

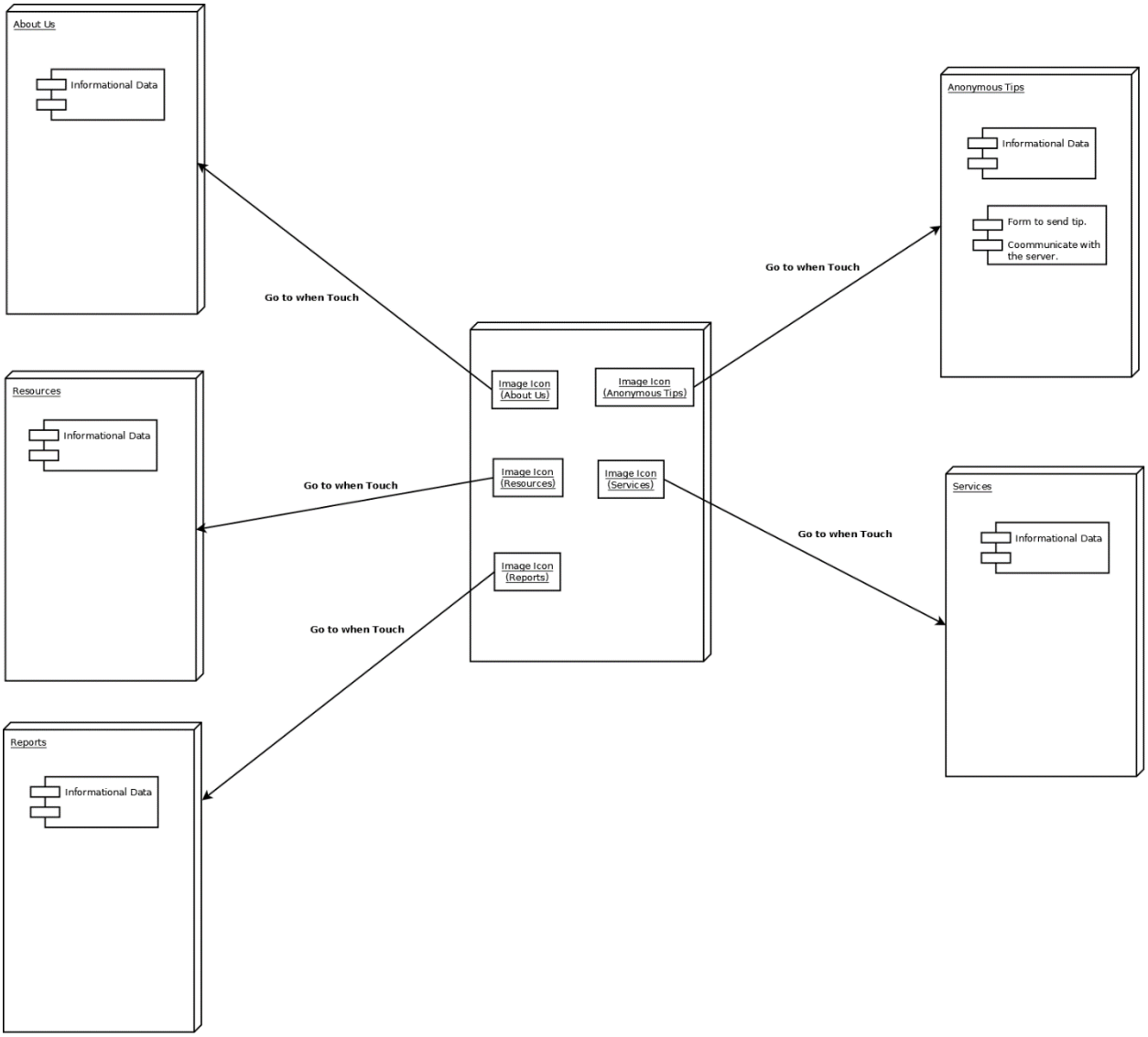


Figure 4: Connection of Pages Diagram:

3.3 Performance Requirements

- User should get a response no longer than 5 seconds with a confirmation
- Maintain a stable platform for easy access of information
- Support the most common and current Android versions as well as some older versions
- Aim for minimal resource requirements so that even the most basic of Android phones can have a positive experience
- Data connection either mobile or Wi-Fi will be necessary to retrieve updated information

3.4 Design constraints (limitations)

Time is a factor in the design of the product. Our development team will work on designated feature of application and will continue on to the next feature only when that feature is publishable. We are a new team with little knowledge of previous created source code as well. We are limited to seven weeks of development in which that time is also spread throughout other difficult classes. This time constraint limits the number of features and we may not be able reach the maximum amount of features requested so our focus will be on the features with the highest priority.

3.5 Software System Attributes

3.5.1 Reliability

The Campus Safety App will provide the user with a tool to access information on the service's that are available to students on campus. Additional information regarding safety, health, law will also be provided.

3.5.2 Availability

This app will run on Android mobile devices and can be used without a network service to access certain informative features. A network service will be required to use additional services such as links, Anonymous Tips page, and Reports page.

3.5.3 Security

Information in the Anonymous Tip page will be one way and be sent through an email client using FTPS. MySQL will be used for assistance request around the parking lot per client's request.

3.5.4 Maintainability

This app is being developed under the Android Studio platform with Java and XML being the languages at work in this working environment.

3.5.5 Portability

This app can be ported to other major platforms such as IOS, Windows, Blackberry but further modifications will be needed to ensure usability is consistent on those platforms.

3.6 Other Requirements

There are not many other requirements needed for this mobile app, but to name a few:

- An important requirement is to have a smartphone, so one can use the mobile app.
- Internet access within the campus, as the information will not be accessible if you are not within the campus
- The ability to use the app and navigate it
- Future versions may give the campus police instant call back capability for better service in an emergency situation.