Phase 3 – NUST Connect

Muhammad Saifullah Khan

Software Design and Architecture

1/15/17

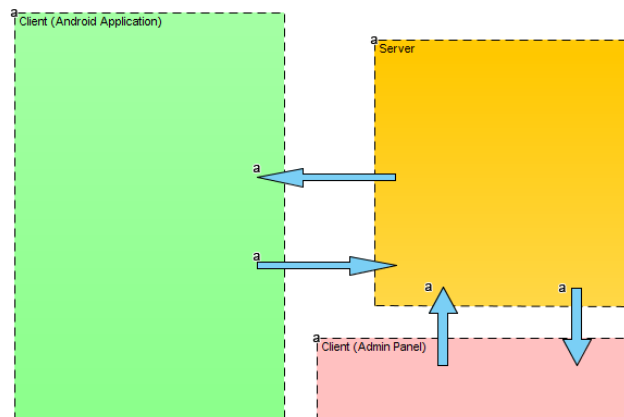# 1 CONTENTS

2 SYSTEM ARCHITECTURE

The system, which (for the end-user) is an Android application which is communicates with other users of the application through a shared server. Which makes the choice of client-server architecture obvious.

The system administrators use a web-based Admin Panel to perform administrative tasks.
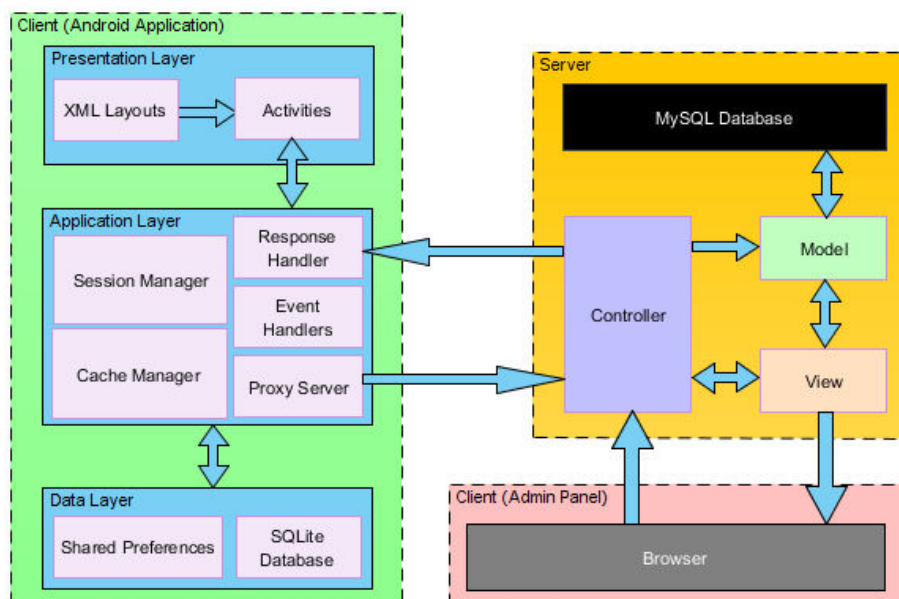
This gives us two clients, the Android application and the Admin Panel, and a server to handle queries from the application and the Admin Panel's requests.

Looking at each of the subsystems individually, the apparent choice for the server was the popularly-used MVC architecture being easily implementable in both PHP and ASP .NET and availability of wide support.
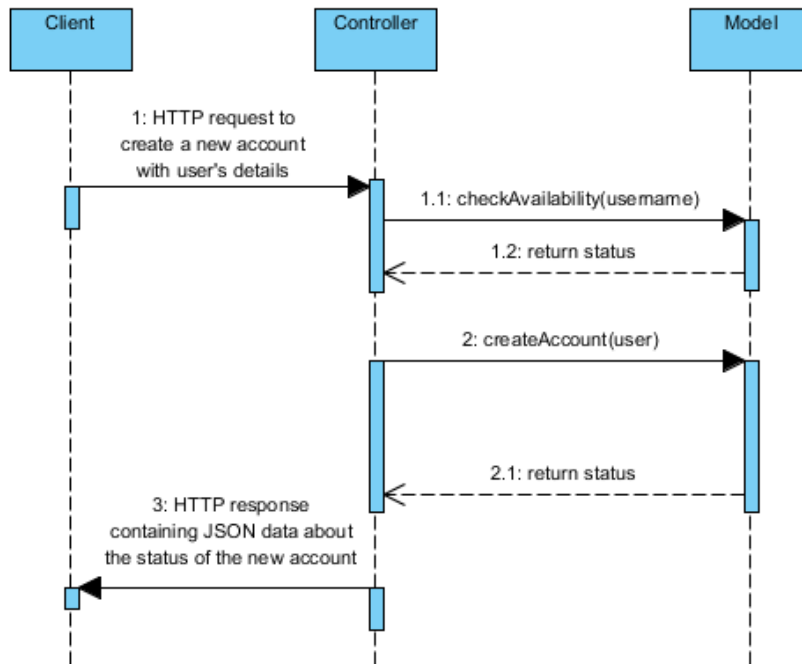
The design of Android's architecture and the Java Framework's APIs it provides do not really make MVC the easiest thing to implement, so for the Android application, it was better to arrange the components in logically-grouped layers.
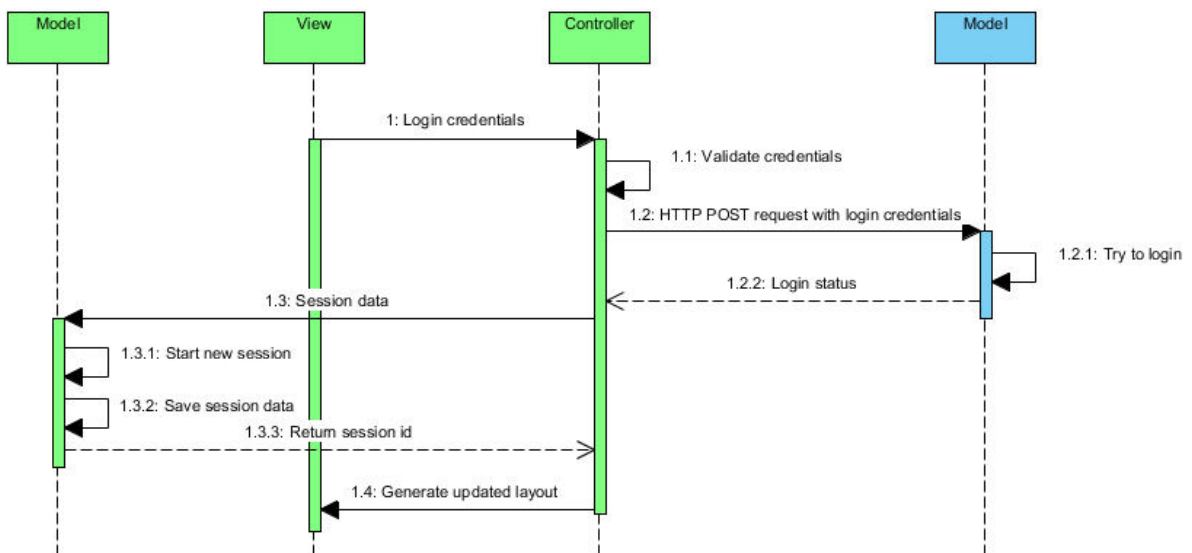
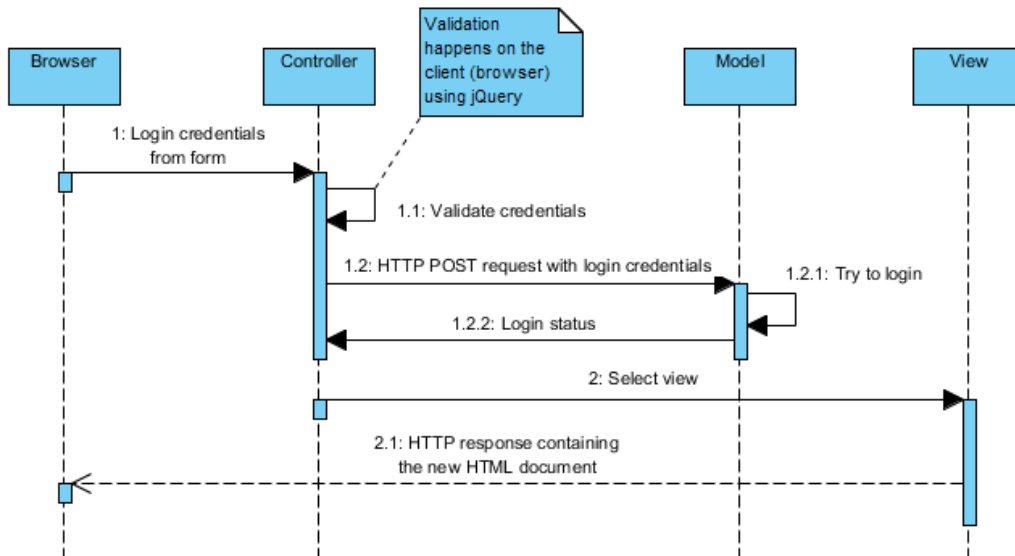The final architecture looks as below:

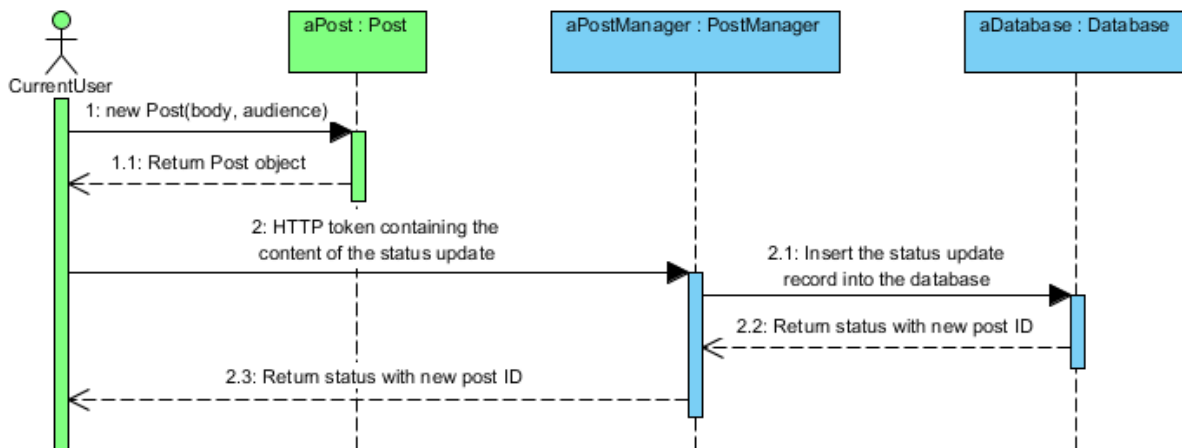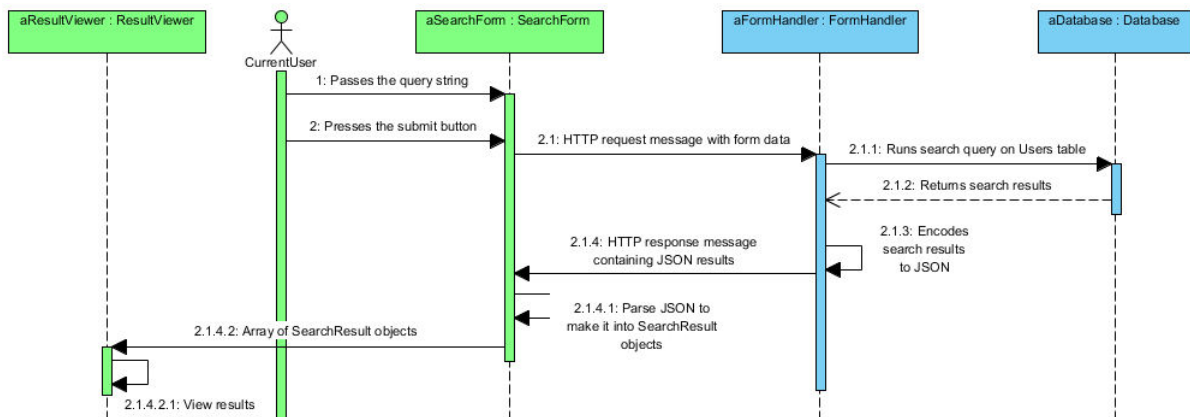# 3 USE CASES

3.1 USER REQUESTS ACCOUNT
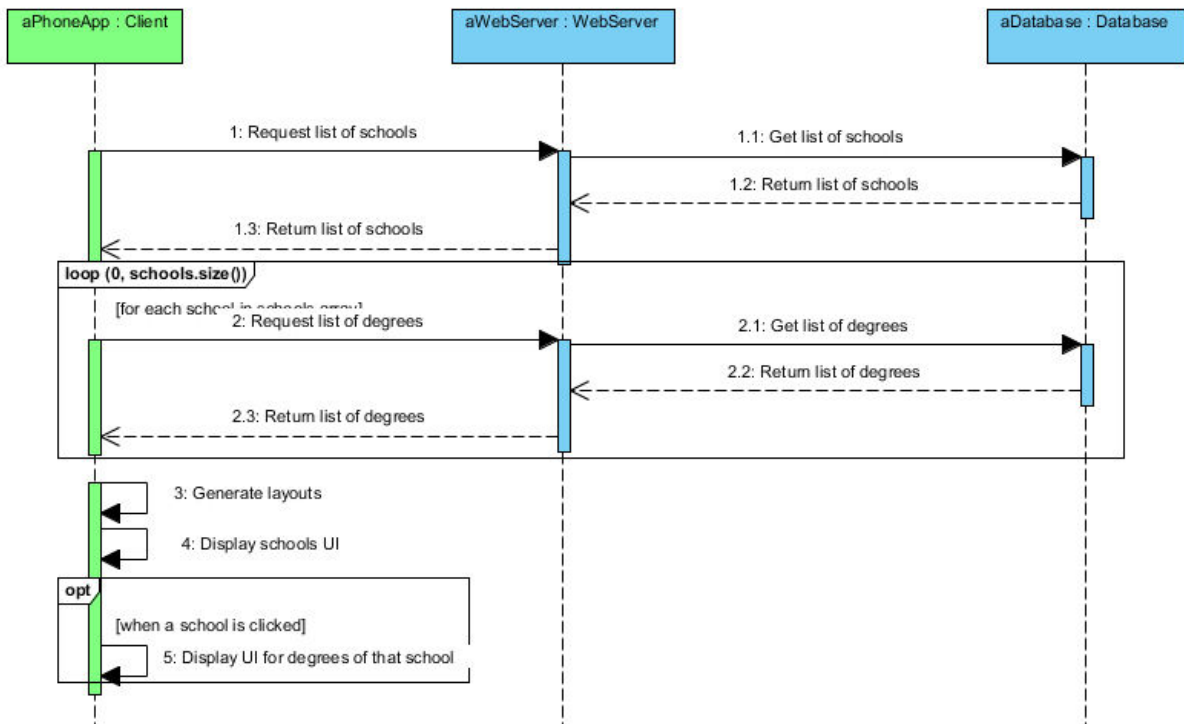

3.2 LOGIN (USER)

3.3 LOGIN (SYSTEM ADMIN)

Browser | Controller | Validation happens on the client (browser) using jQuery | Model | View

1: Login credentials from form
1.1: Validate credentials
1.2: HTTP POST request with login credentials
1.2.1: Try to login
1.2.2: Login status
2: Select view
2.1: HTTP response containing the new HTML document

3.4 POSTING A NEW STATUS UPDATE

CurrentUser | aPost : Post | aPostManager : PostManager | aDatabase : Database

1: new Post(body, audience)
1.1: Return Post object
2: HTTP token containing the content of the status update
2.1: Insert the status update record into the database
2.2: Return status with new post ID
2.3: Return status with new post ID

3.5 SEARCHING FOR A USER

aResultViewer : ResultViewer | CurrentUser | aSearchForm : SearchForm | aFormHandler : FormHandler | aDatabase : Database

1: Passes the query string
2: Presses the submit button
2.1: HTTP request message with form data
2.1.1: Runs search query on Users table
2.1.2: Returns search results
2.1.3: Encodes search results to JSON
2.1.4: HTTP response message containing JSON results
2.1.4.1: Parse JSON to make it into SearchResult objects
2.1.4.2: Array of SearchResult objects
2.1.4.2.1: View results

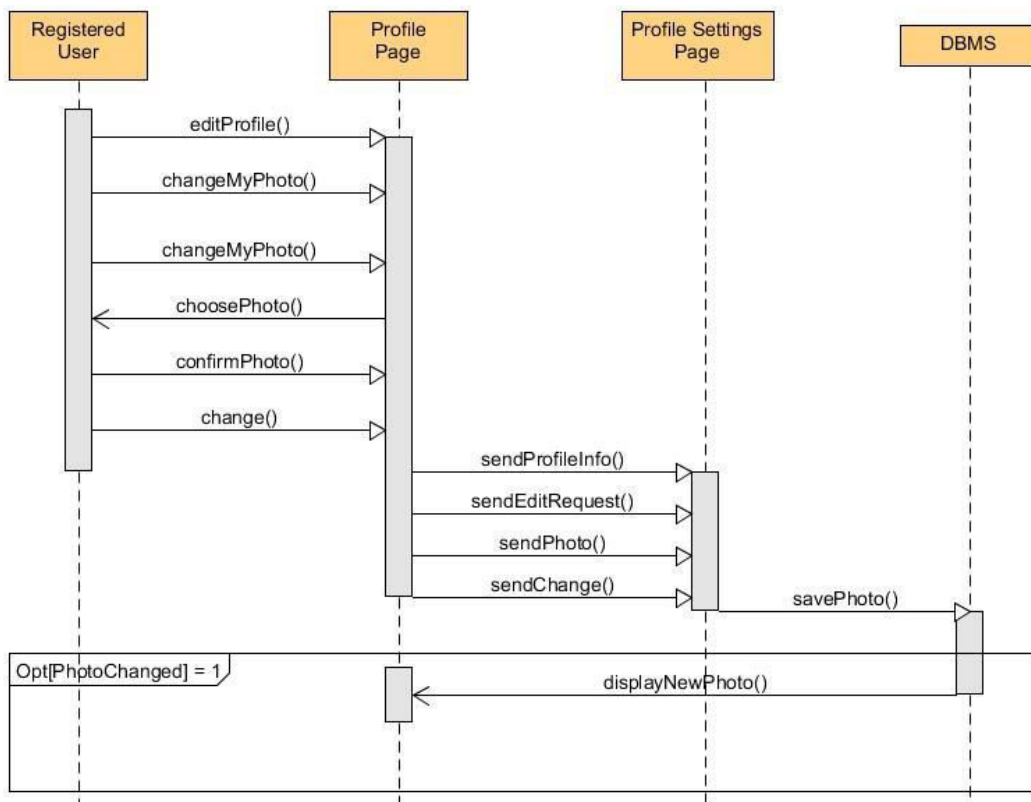## 3.6   BROWSING USER ACCOUNTS

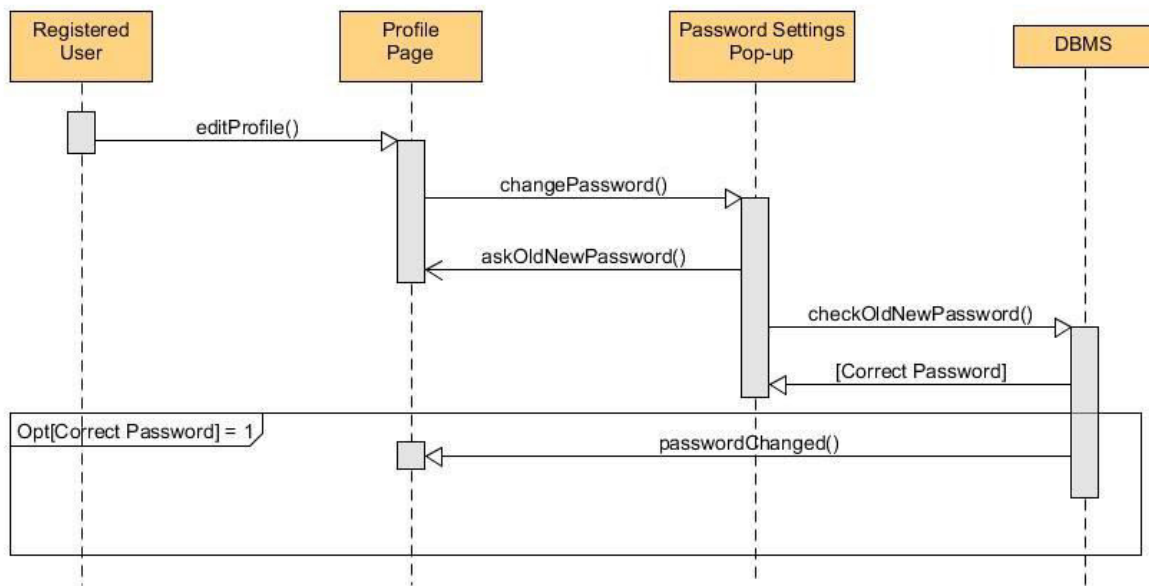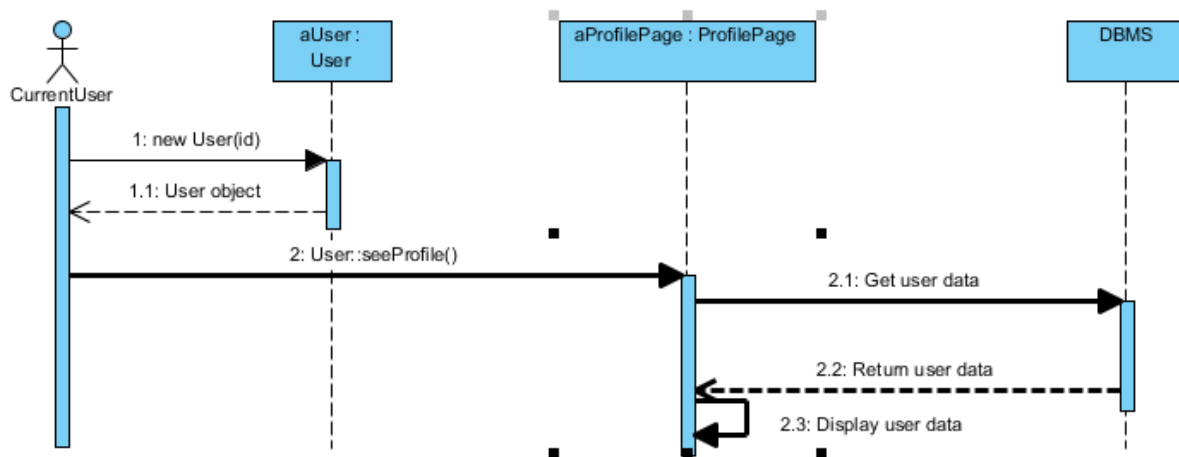## 3.7   CHANGING PROFILE PICTURE

3.8 CHANGING PASSWORD

3.9 DISPLAYING A USER'S PROFILE

3.10 RECEIVING A NOTIFICATION