2016

2016

MARZOUQ 1380949 | AMELIA 1388244 | NAWAF 1377387 | OSHADA 1434048

Final Documentation (DRAFT)

Contents

[1 Introduction 5](#_Toc465779339)

[1.1 Background 6](#_Toc465779340)

[1.2 Objectives 6](#_Toc465779341)

[1.3 Report Structure 7](#_Toc465779342)

[2 Technologies and Tools 7](#_Toc465779343)

[2.1 Client Side 7](#_Toc465779344)

[2.1.1 Android Studio 7](#_Toc465779345)

[2.2 Server Side 8](#_Toc465779346)

[2.2.1 NetBeans 8](#_Toc465779347)

[2.2.2 Glass Fish 8](#_Toc465779348)

[2.2.3 Web Services 9](#_Toc465779349)

[2.2.4 Web Server 9](#_Toc465779350)

[2.2.5 XML Soap 9](#_Toc465779351)

[2.3 Database 9](#_Toc465779352)

[2.3.1 MySQL 9](#_Toc465779353)

[2.3.2 GitHub Repository 10](#_Toc465779354)

[2.4 Security 11](#_Toc465779355)

[2.4.1 Hash Password 11](#_Toc465779356)

[2.4.2 SSL (Secure Sockets Layer) 11](#_Toc465779357)

[3 Data Model 12](#_Toc465779358)

[3.1 Data Dictionary 12](#_Toc465779359)

[3.1.1 Appointment 12](#_Toc465779360)

[3.1.2 Lecturer 12](#_Toc465779361)

[3.1.3 Student 12](#_Toc465779362)

[3.1.4 Studentlectureassignment 13](#_Toc465779363)

[3.1.5 Department 13](#_Toc465779364)

[3.1.6 Subject 13](#_Toc465779365)

[3.2 Universe of Discourse 13](#_Toc465779366)

[3.3 Model Entities and Relations 13](#_Toc465779367)

[3.4 Database Relational Schema 13](#_Toc465779368)

[4 Client Side 14](#_Toc465779369)

[4.1 Design 14](#_Toc465779370)

[4.1.1 Basic Interface Design 16](#_Toc465779371)

[4.2 Implementation 21](#_Toc465779372)

[4.3 Testing 21](#_Toc465779373)

[4.3.1 Testing and Performance 21](#_Toc465779374)

[4.3.2 User acceptance testing 23](#_Toc465779375)

[5 Server Side 25](#_Toc465779376)

[5.1 Design 25](#_Toc465779377)

[5.2 Implementation 25](#_Toc465779378)

[5.3 Testing 25](#_Toc465779379)

[6 User Documentation 26](#_Toc465779380)

[6.1 Technical 26](#_Toc465779381)

[6.2 User Functionality 26](#_Toc465779382)

[6.2.1 Lecture 26](#_Toc465779383)

[6.2.2 Student 26](#_Toc465779384)

[6.3 Usage 26](#_Toc465779385)

[7 Conclusion 27](#_Toc465779386)

[7.1 Findings 27](#_Toc465779387)

[7.1.1 iOS Applications 27](#_Toc465779388)

[7.2 Recommendations 27](#_Toc465779389)

[7.3 Summary 27](#_Toc465779390)

[8 Appendix A 28](#_Toc465779391)

[8.1 Project Planning 28](#_Toc465779392)

[8.2 Meeting Logs 28](#_Toc465779393)

[8.2.1 Supervisor Meetings 28](#_Toc465779394)

[8.2.2 Bi-weekly Meetings 28](#_Toc465779395)

[8.2.3 Group Meetings 29](#_Toc465779396)

[8.3 Supervisor Email records 31](#_Toc465779397)

[8.4 Personal Reflection 34](#_Toc465779398)

[8.4.1 Amelia 34](#_Toc465779399)

[8.4.2 Marzouq 34](#_Toc465779400)

[8.4.3 Nawaf 34](#_Toc465779401)

[8.4.4 Oshada 34](#_Toc465779402)

[9 Appendix B 35](#_Toc465779403)

[9.1 Client Code 35](#_Toc465779404)

[9.2 Server Code 35](#_Toc465779405)

[9.3 Database SQL Dump 35](#_Toc465779406)

[10 Bibliography 0](#_Toc465779407)

[11 Project proposal 1](#_Toc465779408)

[1. Executive Summary 2](#_Toc465779409)

[2. Goal 3](#_Toc465779410)

[3. Background 3](#_Toc465779411)

[4. Objectives 3](#_Toc465779412)

[5. Justification 4](#_Toc465779413)

[6. Scope 5](#_Toc465779414)

[6.1 Functional Requirements 5](#_Toc465779415)

[6.2 Exclusions 6](#_Toc465779416)

[7. Project Team 6](#_Toc465779417)

[8. Methodology 7](#_Toc465779418)

[9. Deliverables 8](#_Toc465779419)

[10. Milestones and Tasks 9](#_Toc465779420)

[11. Resource Requirements and Costs 10](#_Toc465779421)

[12. Risk Analysis Checklist 11](#_Toc465779422)

[12.1 Generic Risk Checklist 11](#_Toc465779423)

[12.2 Specific Project Risks 13](#_Toc465779424)

[13. Quality Assurance Process 14](#_Toc465779425)

[14. Project Work Plan 15](#_Toc465779426)

[15. Intellectual Property 15](#_Toc465779427)

[16. Confidentiality 15](#_Toc465779428)

[17. Declaration 16](#_Toc465779429)

[18. Approvals 17](#_Toc465779430)

# Introduction

The Appointment Management System Project was created to develop a multi-tier system to facilitate making out of class appointments between students and lecturers. The multi-tier system comprises of client and server applications supported by relational data base management system. Students and lecturers login via an android mobile application using their respective usernames and passwords to arrange appointments with one another. The choice to create a smart phone client application was to provide students and lecturers with greater accessibility.

The original proposal was to create a web- based application however a couple weeks into the project we decided with the workload and amount of time we had for this project we wanted to put all our focus in creating one application either a mobile application or a web based application. Considering the skills and experience our team had we decided to create a decent mobile application where we could put all our effort into making it the best possible application that we could.

The main reason behind this idea was based on the current practice where students seek a particular lecturer by visiting the staff room or office. Because Unitec has no set system in place, students didn’t have required office hour access with lecturers to address and resolve any learning related issues and difficulties or to make arrangements to address the matter at a more suitable time. More often this practice results in a continuous flow of students desiring to have out of class access to their lecturers.

Consequently, this causes frustrations for both students and lecturers, since lecturers are interrupted, and students experience delays. To minimize the frustration’s lecturers have adopted their own policies to regulate and manage the flow of students requiring further learning assistance. As these policies vary between lecturers, the system continues to be problematic and confusing.

With a mobile application in place, assisting students and lecturers setting up appointments would save time and effectively illuminate time wastage and interruptions of students showing up unexpectedly with no appointment. We created an Appointment Management System with a common clear standard to reduce inefficiencies arising from inconsistent policies. More so the aim was to provide a useable solution to lessen the burden and frustrations experienced by both students and lecturers. The system is deployed via an android mobile-based application allowing lecturers to advertise their availability to students requiring additional office hour access. The system will also offer both students and lecturers the ability to manage their office hour appointments.

The system also is synced with the lecturers Outlook which is the current way Lecturers organize their appointments.

## Background

Unitec is the biggest Institute in New Zealand with over 16,000 attending students and over 1000 teaching staff. Students and teaching staff are located across three campuses. All campuses are located in Auckland, with the main campus situated in Mt. Albert, the other two situated in Henderson and Albany.

Given the large size of Unitec it is imperative to efficiently manage the logistics for office hour appointments between students and lecturers. Currently there is no set standard policy that defines a best practice, and no system in place to facilitate the process. The department we will be focusing on in this project will be the computer department.

## Objectives

1. Building a Mobile android application for making office hour appointments between students and lecturers.
2. Provide students with priori information as to when lecturers are available for an appointment to be made.
3. Minimize the time taken to set up an appointment.
4. Mobile application should also provide some integration with lecturer’s Outlook based calendar.
5. Enable appointments to be amended after these have been made.
6. Provide secure communication.

## Report Structure

# Technologies and Tools

## Client Side

### Android Studio

Two applications were considered for the Android Development of our project; these were Android Studio and Eclipse. As a group, we had to discuss the pros and cons between the two applications and what would benefit our project best. The main advantage of using Android Studio is that the programmer in our group had previous experience in both applications but found to use Android Studio over Eclipse.

In addition to this the programmers in our group already have experience in using this application. User Interface wise Android Studio was built purposely for Android (Rajput, 2015), While other applications are built to all-purpose IDE that can be used with any language and platform.

Android Studio is a tool for the development of Android platform.

The feature Instant Run can facilitate the pushing of code and resource changes of the running apps. Thus, the developers can access the changes without necessarily restarting the app or rebuilding the APK.

The emulators can install and execute apps in a more rapid manner as compared to the actual devices. It also allows them to test their apps on various Android device settings. The various devices include tablets, phone, and Android TV. Besides, the emulator offers simulation option for different hardware features including network latency, GPS functions, and motion sensors among others.

Developers can write better codes in a faster way thus increasing productivity because of the advanced completion of codes, analysis, and refactoring. This is achieved by the suggestions for the user types in the code editor (Android Studio, 2016)

## Server Side

### NetBeans

NetBeans is an integrated development environment that is free and can be used to develop applications for various platforms such as personal computers, smartphones, and tablets. It also helps in making HTML5 applications. The NetBeans Editor helps the developer to create smart codes because it indents the lines of the codes and matches the various words and brackets. It also highlights the source code in a syntactic and semantic manner. Furthermore, it is rich with code templates and tips, as well as code generators.

The NetBeans IDE enhance the management of the projects by offering a variety of ways to view the projects. For instance, one can view multiple project windows and few helpful tools thus allowing developers to concentrate on their data faster. Developers can quickly detect errors in their codes using the various tools such as FindBugs tool offered by NetBeans IDE. Besides, they can set breakpoints in their source codes and monitor the executions of the program. (NetBeans, 2016)

### Glass Fish

GlassFish is a project that was developed by Sun Microsystems to support Java EE platform. It can enhance the development of scalable and portable applications.

The GlassFish Server reduces the footprint’s size by loading it when necessary only. Thus, the load time is reduced significantly saving time. Besides, it reduces the amount of resources used by the application because of the limited footprint sizes.

The developers can choose the tools they are conformable with to use on GlassFish. For instance, they can adopt either NetBeans or Eclipse to develop their applications with GlassFish. Developers are also able to save time by using GlassFish along other tools such as Eclipse and NetBeans. This is because the combination reduces the typical six steps in development to only three steps. Besides, it uses sessions to store data that eliminate the need to repopulate the session during the development process.

An extra benefit of GlassFish Server is that it is available in many languages. As such, it can be used by developers from around the world comfortably because they can choose the languages they are comfortable with (Oracle, n.d.)

### Web Services

A web service is a software that can be found on the internet and send messages through the standardized XML messaging system. It uses XML to transmit messages. It also supports the movement of information from the client to server and back. It also supports the exchange of documents (Saab, et al., 2009)

### Web Server

A web server is a system based on the computer and works as a processor of the request through HTTP to access information from the World Wide Web. It allows users to create websites and configure their security. It also lets users transfer files through the File Transfer Protocol. It also permits the creation of virtual directories. (Web Server (IIS) Role Overview, 2016)

### XML Soap

XML SOAP is a protocol is lightweight in nature and can be used to support the exchange of information that is structured in a decentralized environment. XML Soap has the extensibility features. The extremes of the application can negotiate the behaviours of the various commands that are supported. Besides, the header element in the XML Soap gives developers the opportunity to include additional information such as credential items that enhance controlled access. XML Soap can also support the exchange of information over various protocols. The independence of the messaging framework from SOAP gives users the freedom to choose from the variety of communications protocols (Server, n.d.)

## Database

### MySQL

For our database, we decided to use MySQL because it can run on many operating systems and is a free and open sourced management system. In addition, it also allows solid data security layers that protect sensitive data from intruders. (MySQL Administration Guide, 2015). It also supports Java language which is the coding language we will be using to develop our application.

MySQL is a database server that is a robust, structured query language, and supports multiple users. It is used to store data, e-commerce, and support applications that require storage of users’ information. As for security, MySQL has a secure and flexible system that supports password and privileges. As such, it can prevent unauthorized access to the data stored. Also, it uses encryption of passwords whenever a user connects to a server.

MySQL offers a rich source of data types that users can use when they store their data. The advanced data types provided by MySQL include BLOB, Enum, and Binary besides the standard data types such as date, varchar, text, and double Float. This is an essential feature that makes developers from around the world have a great experience in the use of MySQL. This is because it can display error messages in many languages. It also supports many character sets from around the world. Furthermore, the time zone of the server can be changed dynamically, and can be adjusted manually as well ("MySQL :: MySQL 5.7 reference manual…", n.d.).

### GitHub Repository

GitHub repository is an online platform that stores tools necessary for software development. It connects over 15 million people around the world to help them share, learn, and work together to build software.

The GitHub repository has several tools that have been tested in the field that can be used for projects. All the tools have these features of being tested before being deployed in the repository. Besides, the tools are free and can be accessed by anyone in the community.

Users have the opportunity to have their problems solved by other members of the repository. As such, GitHub offers a platform that can support the integration of efforts to address problems in the repository.

GitHub workflow allows the review of codes in a cooperative manner. Members can create a branch and make commitment that attracts the Pull requests from people around the world. These people can help to suggest necessary changes for the projects ("Build software better, together", n.d.).

## Security

### Hash Password

An important aspect of a user account system is how user passwords are protected. To protect our passwords, we decided to use Hash Password, this way we are able to protect our client’s private information and prevent corruption. Hash Password generates a long random salt using CSPRNG.

The general workflow for account registration and authentication in a hash-based account system is as follows:

1. The user creates an account.
2. Their password is hashed and stored in the database. At no point is the plain-text (unencrypted) password ever written to the hard drive.
3. When the user attempts to login, the hash of the password they entered is checked against the hash of their real password (retrieved from the database).
4. If the hashes match, the user is granted access. If not, the user is told they entered invalid login credentials.
5. Steps 3 and 4 repeat every time someone tries to login to their account.

(Security, 2016)

### SSL (Secure Sockets Layer)

SSL is a typical security technology for creating an encrypted link between a server and the client. It allows credit card numbers, and login credentials to be transmitted securely. Sensitive information like such normally gets sent between the browsers and web servers in plain text which will leave you vulnerable to eavesdropping. If an attacker is able to intercept all data being sent between a browser and a web server, they can see and use that information. (digicert, 2016)

# Data Model

The database used for this project was MySQL database which was installed onto two team member’s machines who were responsible for the development of the database. There are 4 tables in the complete database.

## Data Dictionary

### Appointment

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Column | Type | Null | Default | Links to | Referenced column |
| date | VARCHAR(50) | NO |  |  |  |
| start | VARCHAR(50) | NO |  |  |  |
| end | VARCHAR(50) | NO |  |  |  |
| lectureusername | VARCHAR(100) | NO |  | lecturer | Username |
| studentusername | VARCHAR(100) | YES |  | student | username |
| isActive | VARCHAR(10) | NO |  |  |  |

### Lecturer

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Column | Type | Null | Default | Links to | Comments |
| username | VARCHAR(100) | NO |  |  |  |
| title | VARCHAR(45) | NO |  |  |  |
| firstname | VARCHAR(100) | NO |  |  |  |
| lastname | VARCHAR(100) | NO |  |  |  |
| password | VARCHAR(24) | NO |  |  |  |
| salt | VARCHAR(24) | NO |  |  |  |
| department | VARCHAR(100) | NO |  |  |  |
| email | VARCHAR(200) | NO |  |  |  |

### Student

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Column | Type | Null | Default | Links to | Comments |
| username | VARCHAR(100) | NO |  |  |  |
| firstname | VARCHAR(100) | NO |  |  |  |
| lastname | VARCHAR(100) | NO |  |  |  |
| password | VARCHAR(24) | NO |  |  |  |
| salt | VARCHAR(24) | NO |  |  |  |
| email | VARCHAR(200) | NO |  |  |  |

### Studentlectureassignment

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Column | Type | Null | Default | Links to | Referenced column |
| lectureusername | VARCHAR(100) | NO |  | Lecturer | Username |
| studentusername | VARCHAR(100) | NO |  | student | username |
| subject | VARCHAR(45) | NO |  |  |  |

### Department

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Column | Type | Null | Default | Links to | Referenced column |
| department | VARCHAR(45) | NO |  |  |  |

### Subject

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Column | Type | Null | Default | Links to | Referenced column |
| subject | VARCHAR (45) | NO |  |  |  |
| department | VARCHAR (45) | NO |  |  |  |

## Universe of Discourse

## Model Entities and Relations

## Database Relational Schema

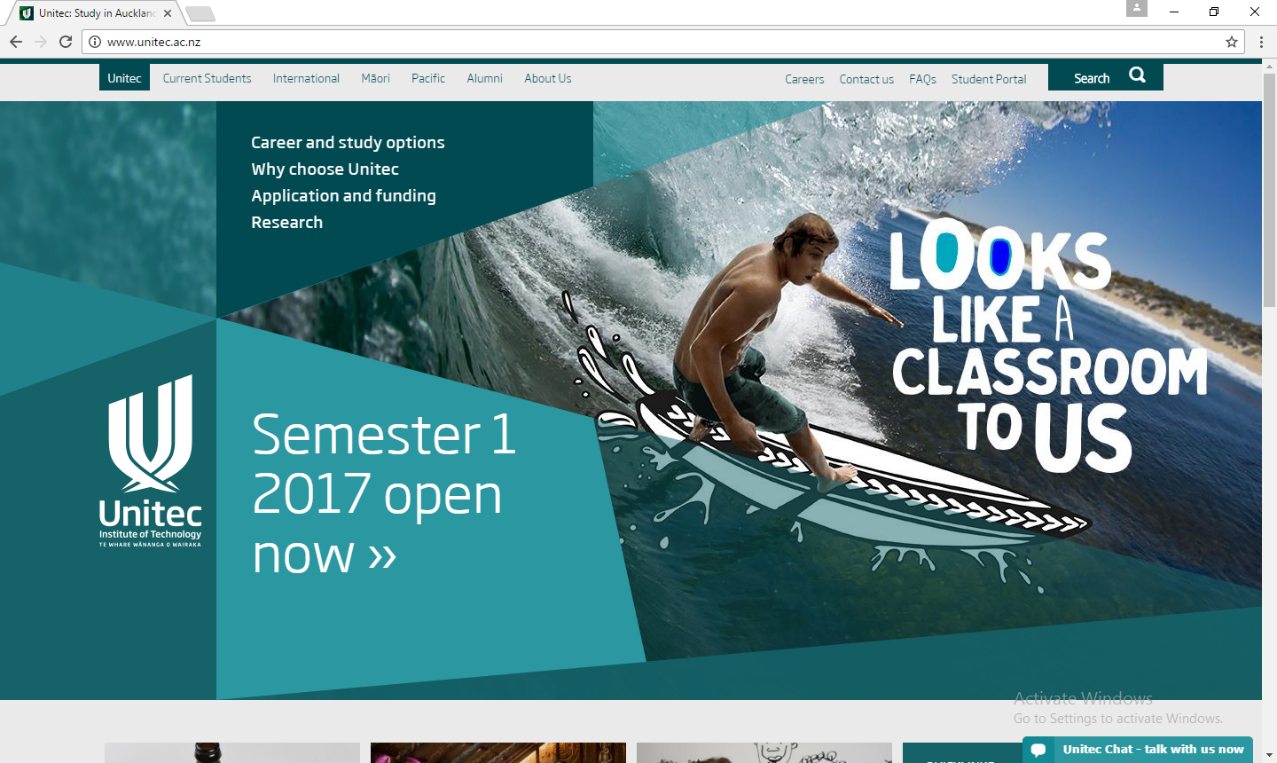
# Client Side

## Design

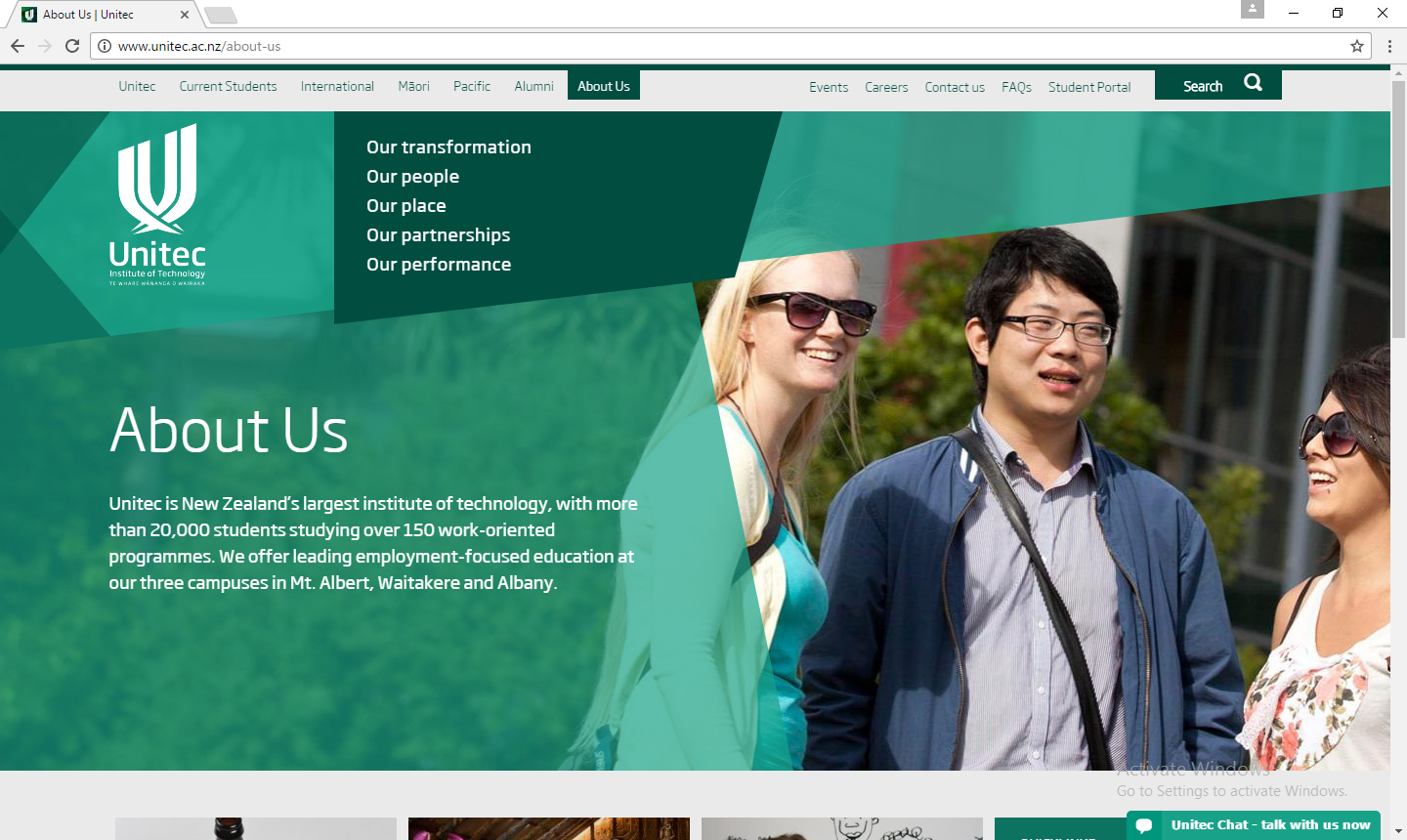
The technique we used for designing the user interface for this application was to make the application predictable and simple for the user (Garrett, n.d.). The main elements incorporated are

* Input Controls
* Navigational Components
* Informational Components
* Containers

We used common UI elements and created consistency throughout the application. We decided to strategically use the colours of Unitec which are green, blue and white. We wanted to incorporate as much of Unitec as we could as we wanted staff and students to be familiar with the new application. As you can see the shades of green will help the user become more familiar with the application they are using. It will also make the application feel authentic and belong to Unitec.

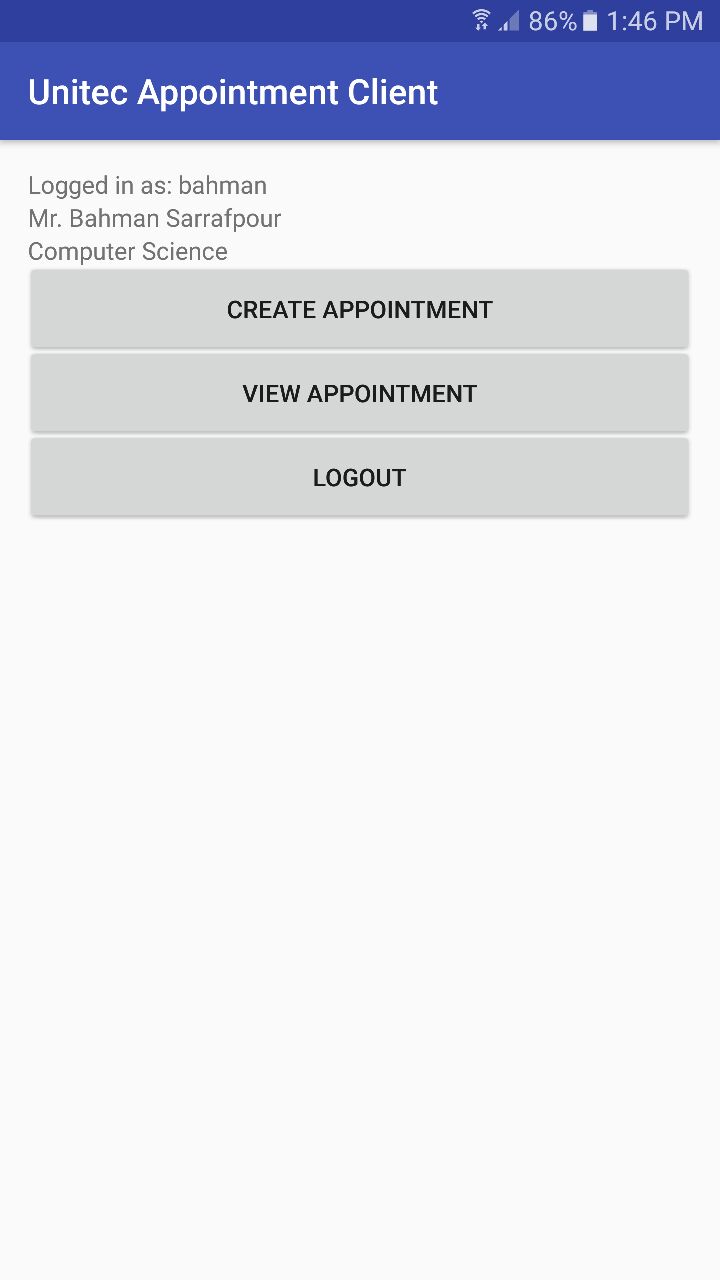
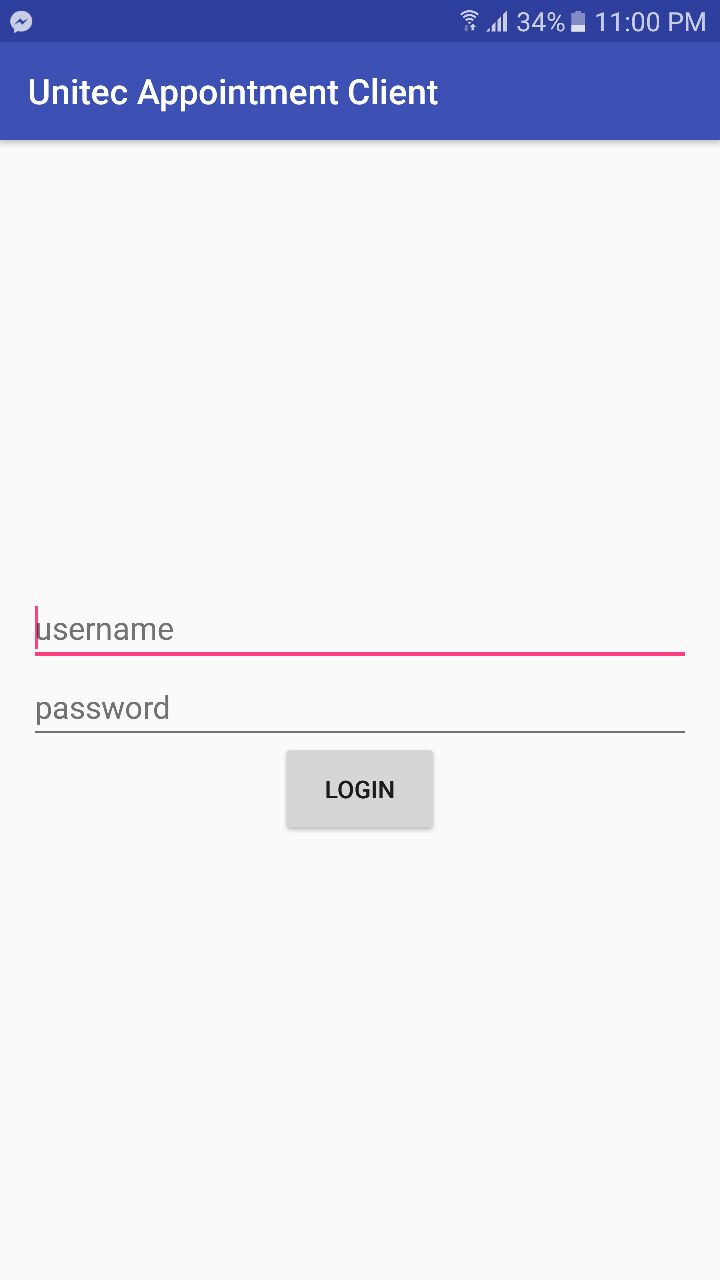
Here we have some examples of the colour scheme we used for the design from the original Unitec website. As you can see the main page is different shades of blue with white writing.

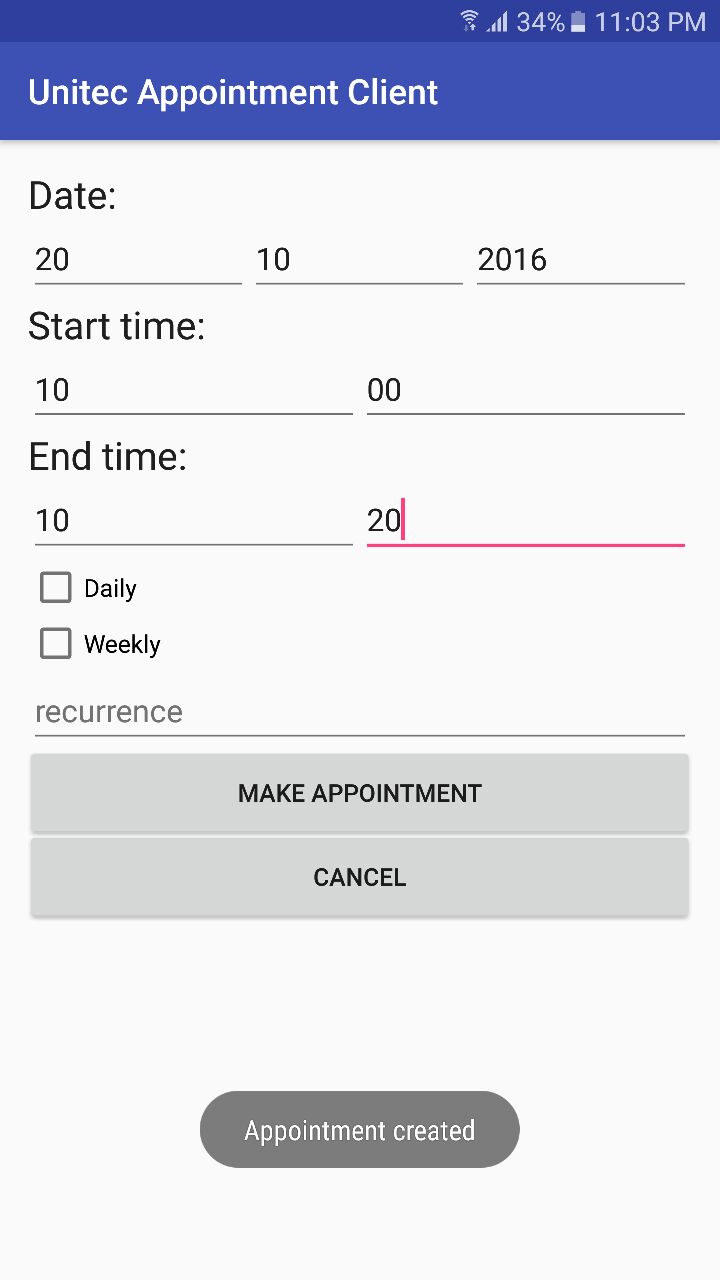
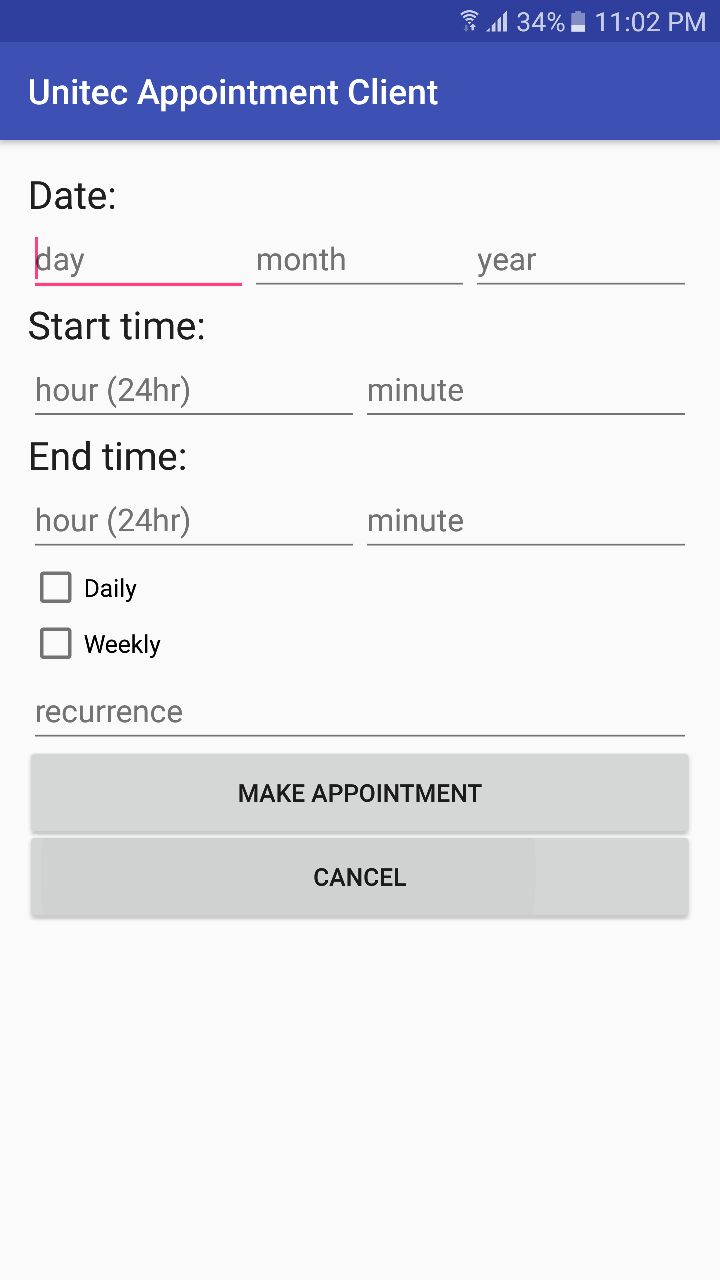
And the About Us page from the Unitec website are different shades of green. These bright colours are



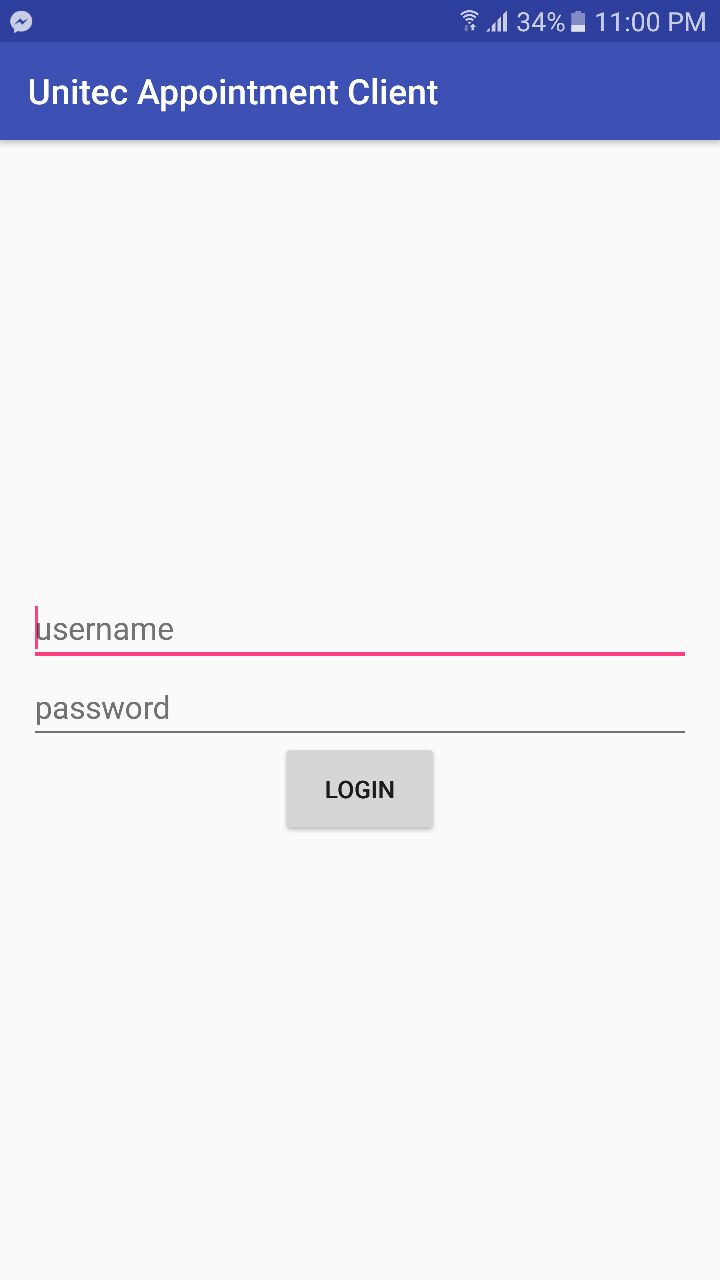
### Basic Interface Design

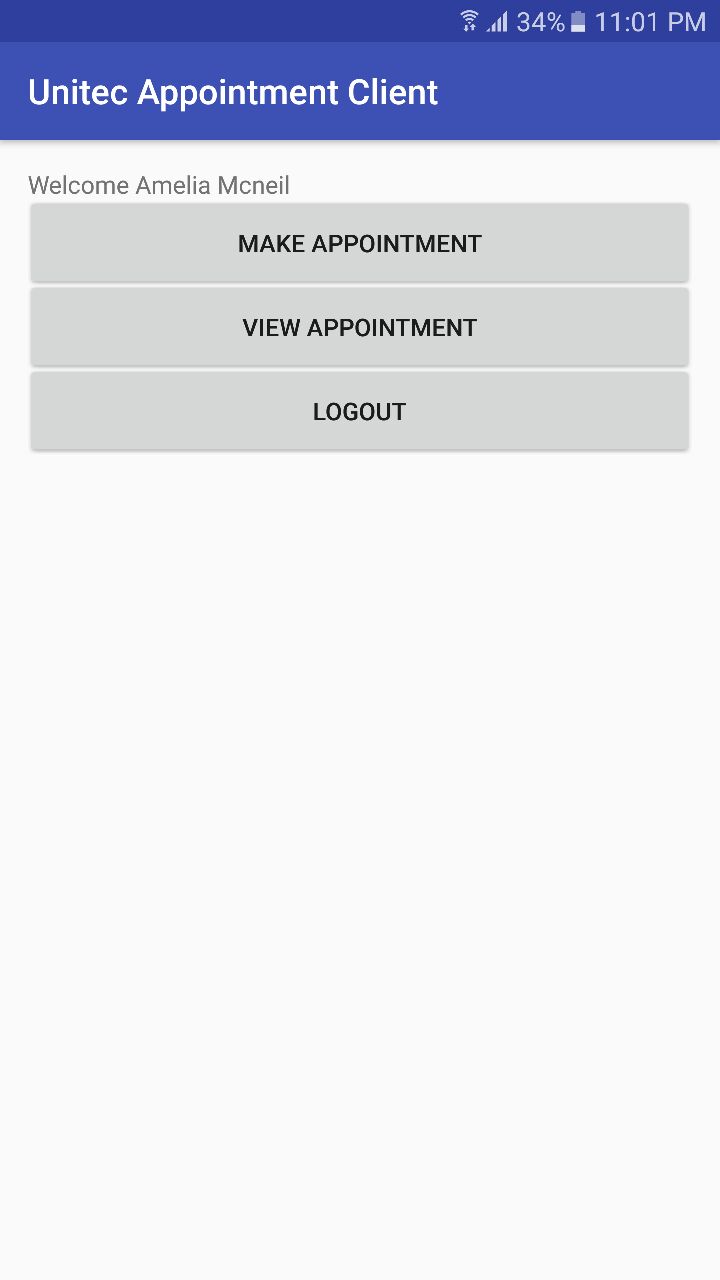
#### Lecturer:

Log in page Home page

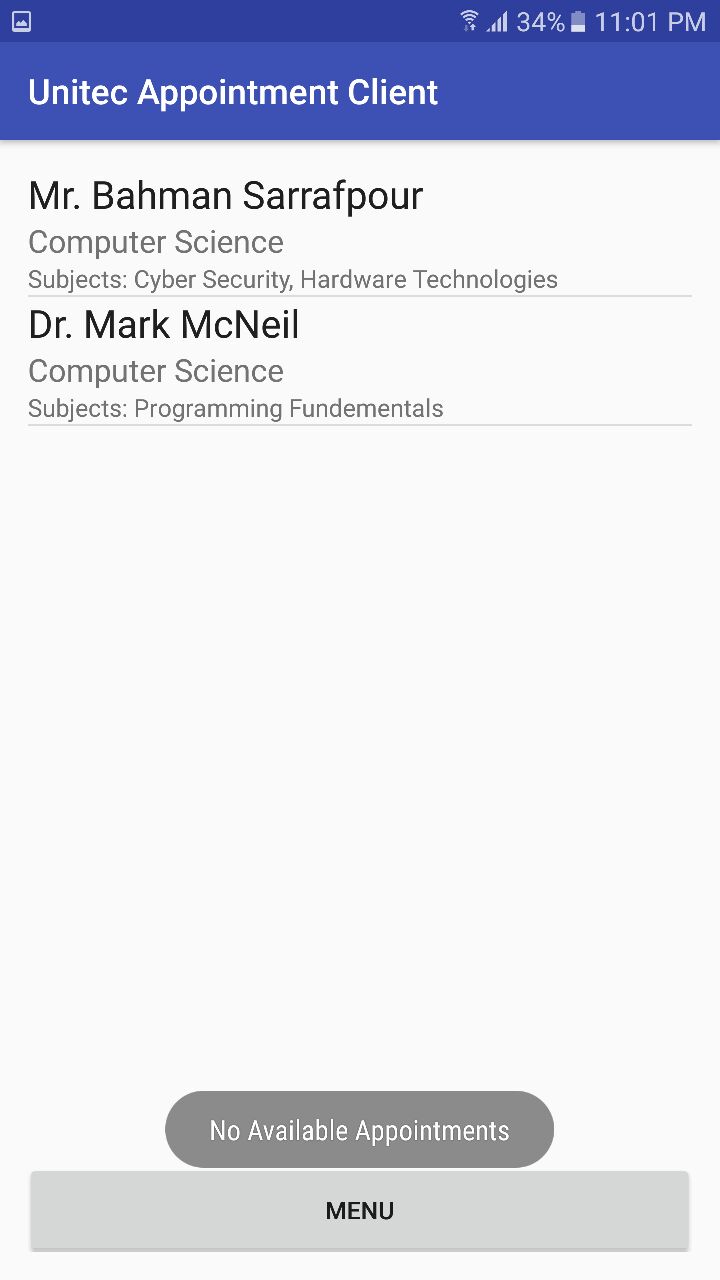
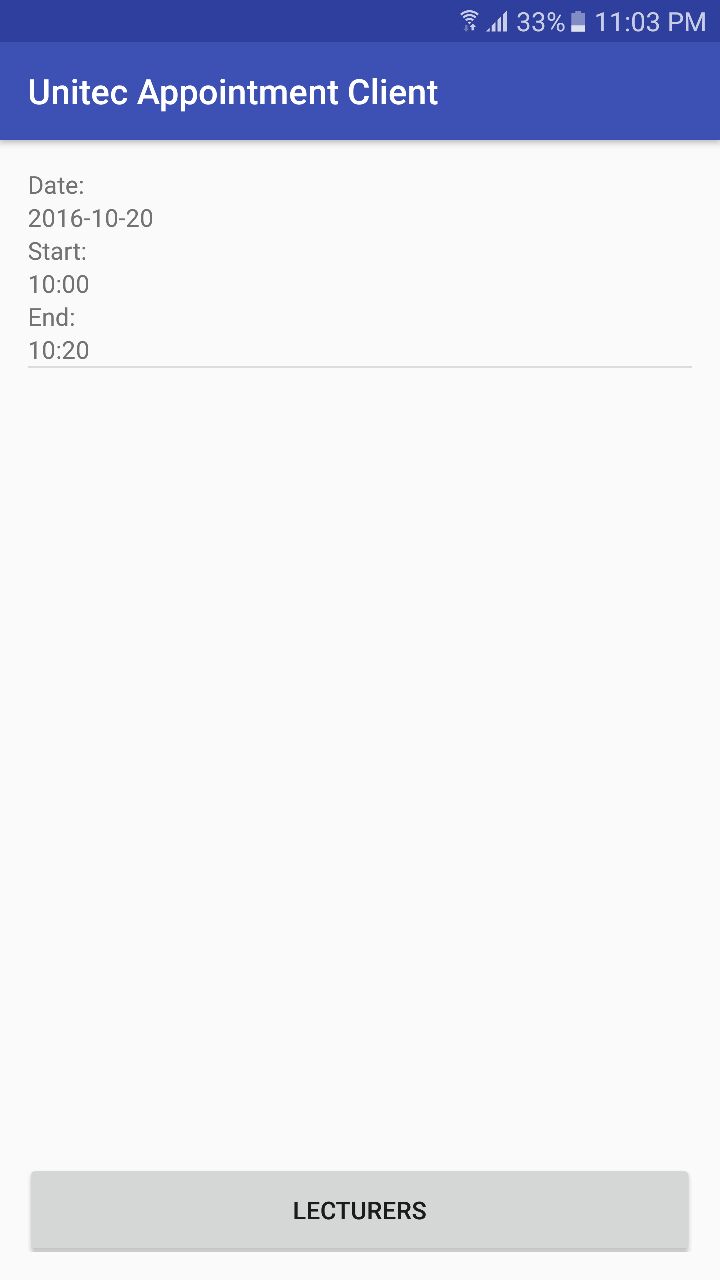
Create appointment Create Appointment (Input Data)

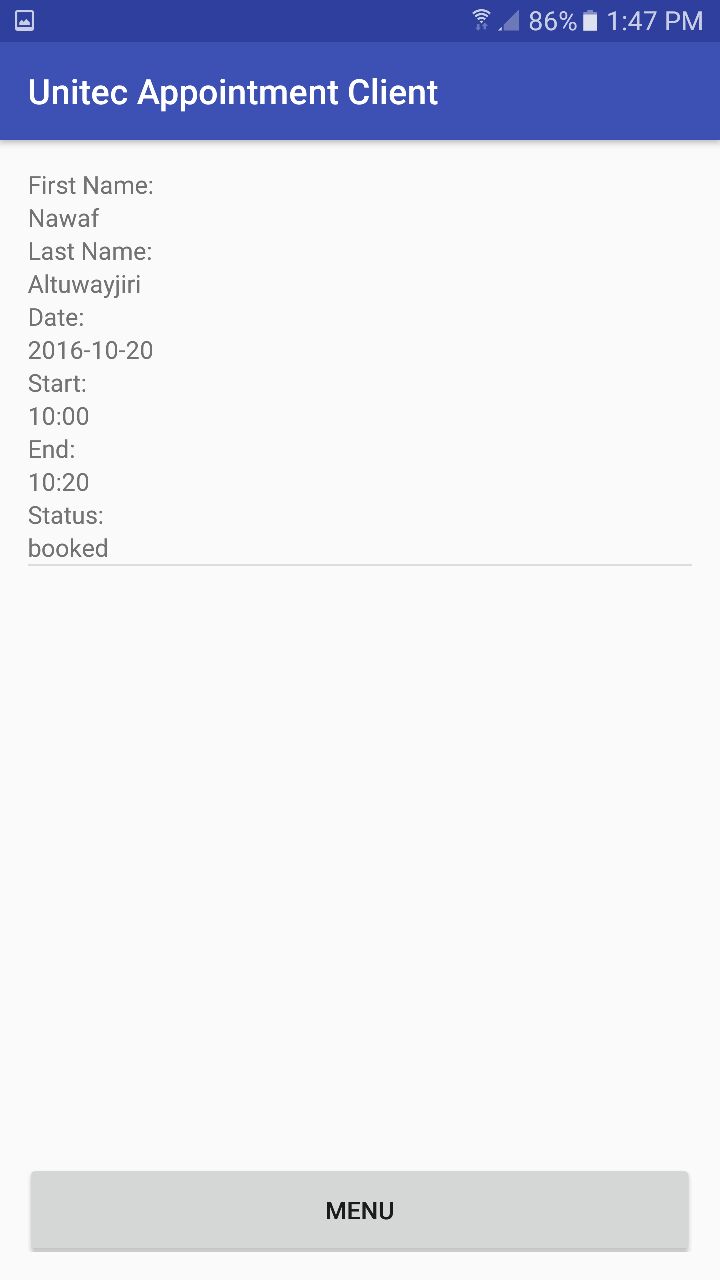
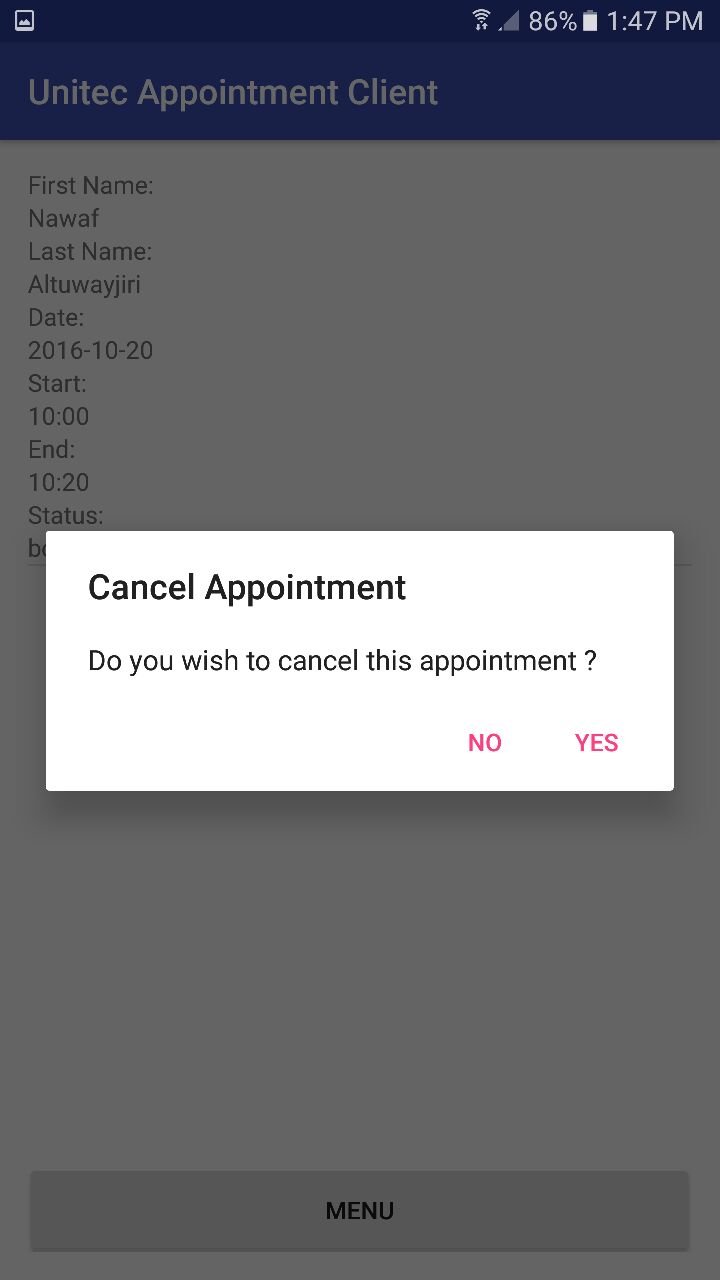
#### Student

Login Homepage



Make appointment(lecturers) Select appointment time slot



Appointment created Cancel Appointment

## Implementation

## Testing

### Testing and Performance

User:

Student

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement to test | Test data input | Expected Outcome | Actual Outcome |
| Student login | Enter username and password. | Student is logged in successfully. | Success |
| Make an appointment | Select make an appointment from home page.  Select what lecturer whom you want to make an appointment with.  Select available time slot available.  Appointment has been created. | Student has an appointment.  Student has successfully created an appointment. | Success |
| View an appointment | Select view an appointment from home page. | All available appointments are listed.  Student can successfully view appointment. | Success |
| Delete appointment | Select view appointment tab.  Select which appointment you want to delete.  A box will appear to confirm deletion.  Select yes. | Appointment has been deleted. | Success |

User:

Lecturer

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement to test | Test data input | Expected Outcome | Actual Outcome |
| Lecturer login | Enter username and password | Lecturer is logged in successfully | Success |
| Lecturer add appointment time slot | Enter date.  Enter start time and end time hours and minutes (24hr).  Select create appointment. | Appointment time has been created.  Student can now view lecturer’s appointments. | Success |
| Lecturer view appointment | Lecturer needs to be logged in.  Select view appointment. | Appointments that are booked will be displayed. | Success |
| Notification sent by email when appointment is booked by student | Student makes an appointment. | Email notification will be sent to the lecturer’s email. | Success |
| Delete appointment |  |  |  |

### User acceptance testing

**Name:** **Email:**

**Date:** **Comment:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tasks* | *Poor* | *Fair* | *Good* | *Excellent* |
| *Navigation bar* |  |  |  |  |
| *Mobile design* |  |  |  |  |
| *Functionality* |  |  |  |  |
| *Performance* |  |  |  |  |
| *User Interface Design* |  |  |  |  |

**Name:** **Email:**

**Date:** **Comment:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tasks* | *Poor* | *Fair* | *Good* | *Excellent* |
| *Navigation bar* |  |  |  |  |
| *Mobile design* |  |  |  |  |
| *Functionality* |  |  |  |  |
| *Performance* |  |  |  |  |
| *User Interface Design* |  |  |  |  |

**Name:** **Email:**

**Date:** **Comment:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tasks* | *Poor* | *Fair* | *Good* | *Excellent* |
| *Navigation bar* |  |  |  |  |
| *Mobile design* |  |  |  |  |
| *Functionality* |  |  |  |  |
| *Performance* |  |  |  |  |
| *User Interface Design* |  |  |  |  |

**Name:** **Email:**

**Date:** **Comment:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tasks* | *Poor* | *Fair* | *Good* | *Excellent* |
| *Navigation bar* |  |  |  |  |
| *Mobile design* |  |  |  |  |
| *Functionality* |  |  |  |  |
| *Performance* |  |  |  |  |
| *User Interface Design* |  |  |  |  |

# Server Side

## Design

## Implementation

## Testing

# User Documentation

## Technical

## User Functionality

### Lecture

Can do the following:

* They must be a staff member at Unitec and registered in the database to log into the android application.
* Can create appointments through the application.
* Can cancel appointments through the application.
* Can view, cancel and update appointments using their Outlook calendars.

### Student

Can do the following:

* Must be enrolled in Unitec and registered in the database to log into the android application.
* Students can view appointments if an appointment has been made.
* Students can make an appointment with a lecturer.
* Students can cancel appointments.

## Usage

(screenshots of how to use the application)

# Conclusion

## Findings

### iOS Applications

## Recommendations

## Summary

# Appendix A

## Project Planning

Gantt Chart

## Meeting Logs

### Supervisor Meetings

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Topic | Time | Attendees |
| 01/08/2016 | Project proposal resubmission | 3pm | Amelia  Marzouq  Nawaf |
| 12/08/2016 | New group member | 2pm | Amelia  Marzouq  Nawaf  Oshada |
| 01/09/2016 | Catch up meeting -Progress check | 12pm | Amelia  Marzouq  Nawaf |
| 09/09/2016 | Catch up meeting -Progress check | 1pm | Amelia  Marzouq  Nawaf  Oshada |
| 19/09/2016 | Catch up meeting -Progress check | 4pm | Amelia  Marzouq  Nawaf |
| 10/10/2016 | Catch up meeting – Progress check | 2:30pm | Amelia  Marzouq  Nawaf |

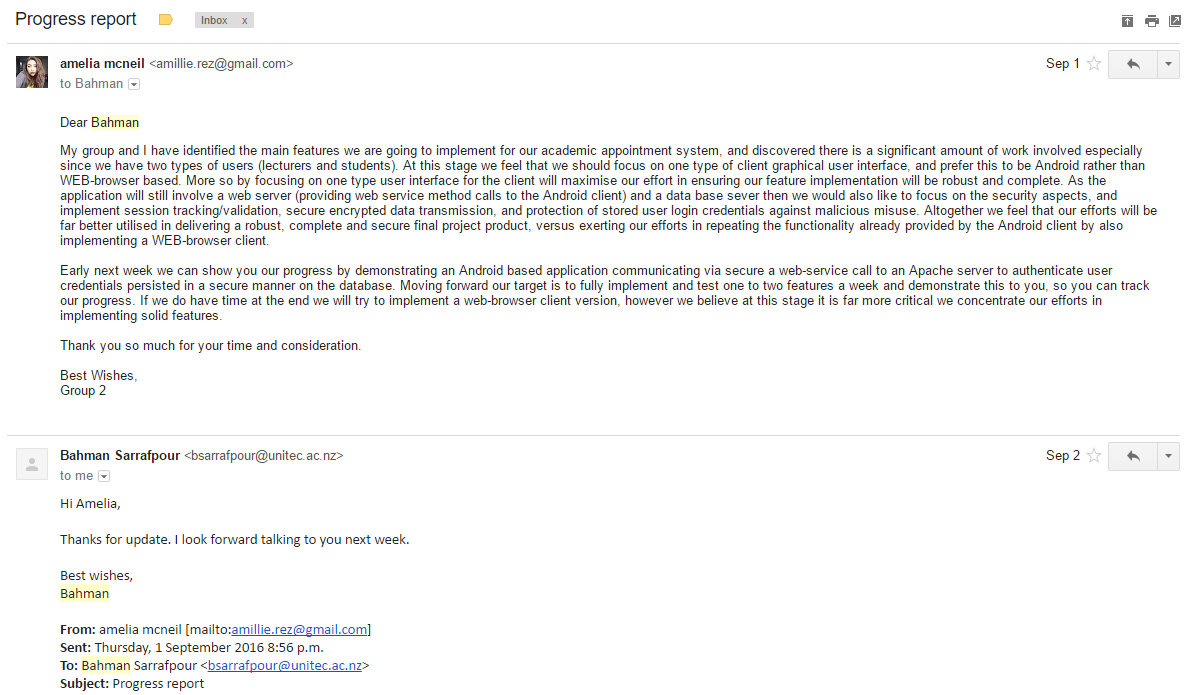
### Bi-weekly Meetings

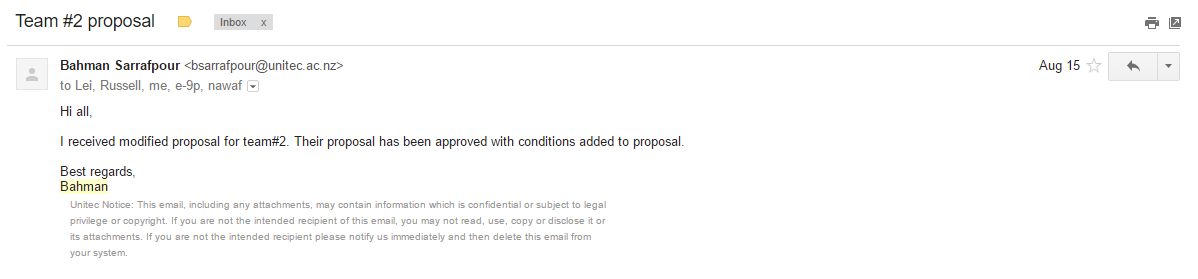
|  |  |  |  |
| --- | --- | --- | --- |
| Date | Topic | Time | Attendees |
| 22/07/2016 | Information Session | 11am |  |
| 11/08/2016 | Proposal Defence | 9:45am – 10:45am | Amelia  Marzouq  Nawaf |
| 26/08/2016 | Meeting 1  Presentation - Progress report 1 | 10am – 11am | Amelia  Marzouq  Nawaf  Oshada |
| 09/09/2016 | Meeting 2  Presentation - Progress report 2 | 10am – 11am | Amelia  Marzouq  Nawaf |
| 21/10/2016 | Meeting 4  Presentation – Progress report 4 | 10am – 11am | Amelia  Marzouq  Oshada |

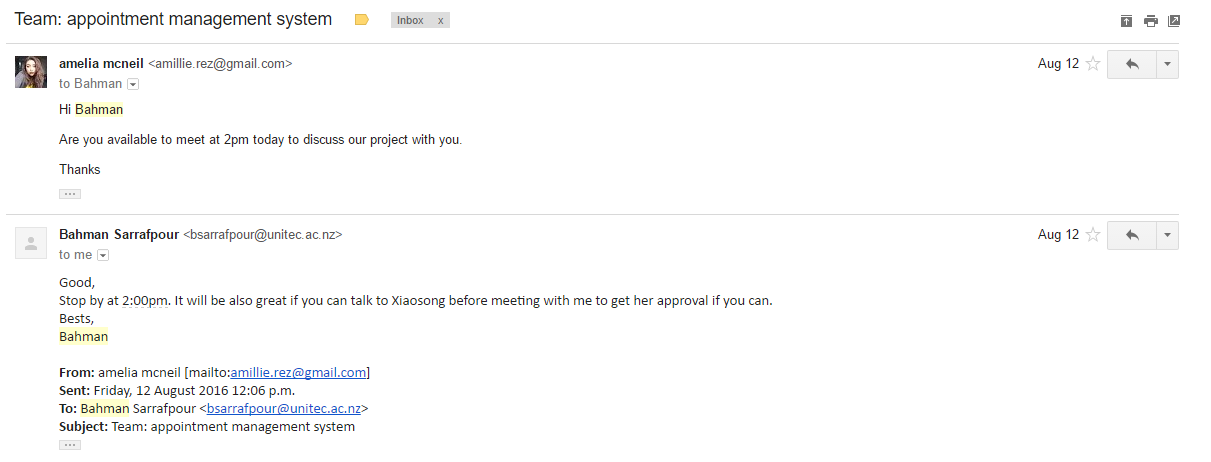
### Group Meetings

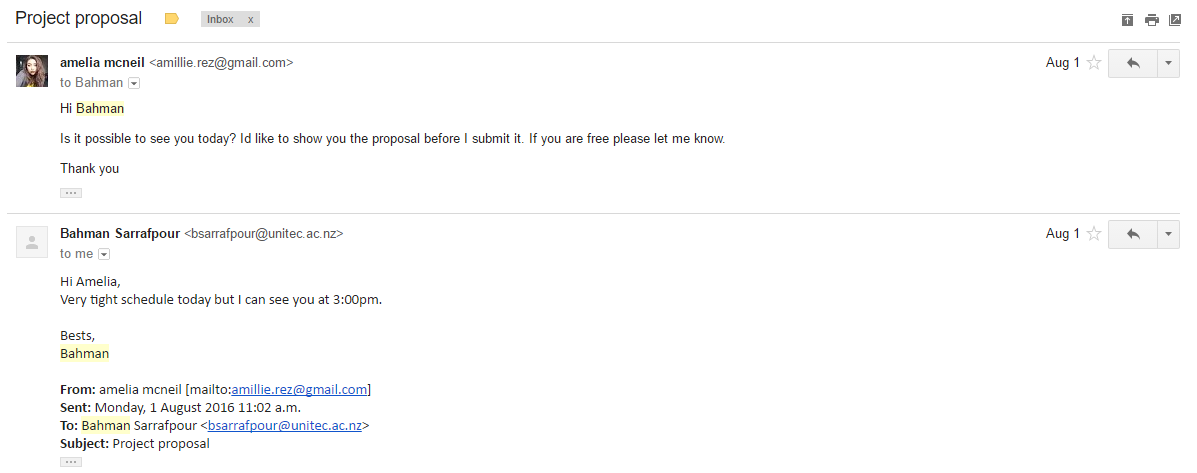
|  |  |  |  |
| --- | --- | --- | --- |
| Date | Topics | Time | Attendees |
| 04/08/2016 | Project proposal write up | 5:30pm – 6:30pm | Amelia  Marzouq  Nawaf |
| 09/08/2016 | Re submission of proposal with new scope |  |  |
| 11/08/2016 | Team catch up/ Progress check up | 3 – 3:30pm | Amelia  Marzouq  Nawaf  Oshada |
| 15/08/2016 | Proposal resubmission  (accepted) | 2:00pm – 3:00pm | Amelia  Marzouq  Nawaf  Oshada |
| 04/09/2016 | Project re evaluation  -Decide to focus on one application | 12pm – 2pm | Amelia  Marzouq  Nawaf  Oshada |
| 07/09/2016 | Team catch up/ Progress check | 11am – 1pm | Amelia  Marzouq  Nawaf  Oshada |
| 08/09/2016 | Team catch up/ Progress check | 2pm – 3pm | Amelia  Marzouq  Nawaf |
| 14/09/2016 | Team catch up/ Progress check | 12pm – 1pm | Amelia  Marzouq  Nawaf  Oshada |
| 25/09/2016 | Team catch up/ Progress check | 5:30pm – 9pm | Amelia  Marzouq  Nawaf  Oshada |
| 26/09/2016 | Team catch up/ Progress check | 5pm – 7pm | Amelia  Marzouq  Nawaf  Oshada |
|  |  |  |  |

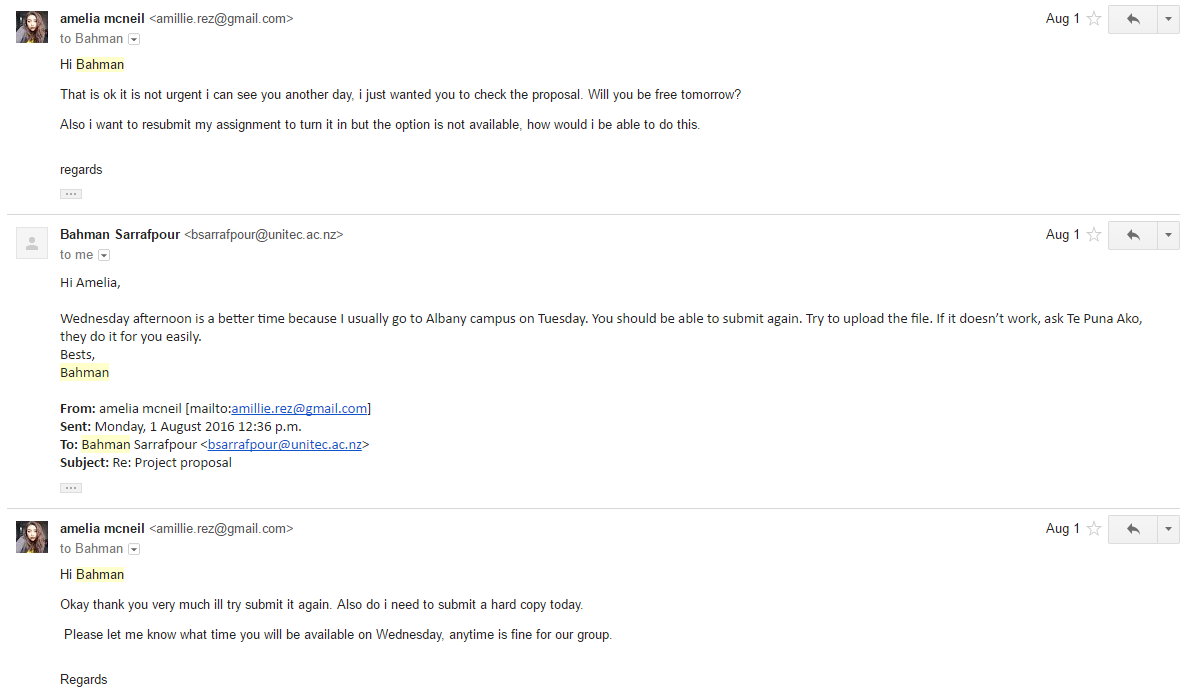
## Supervisor Email records

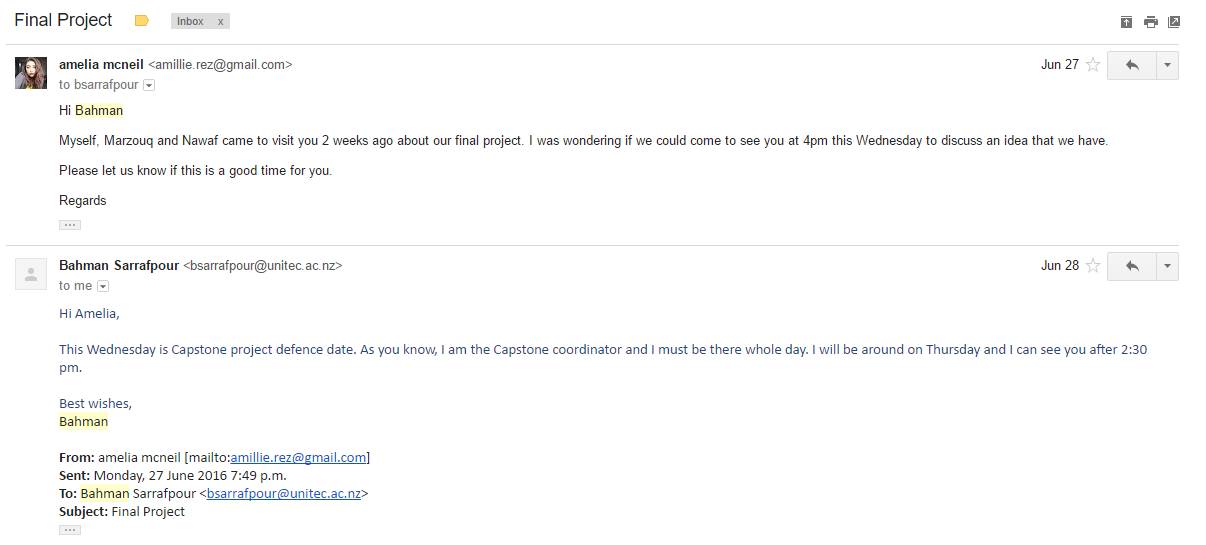


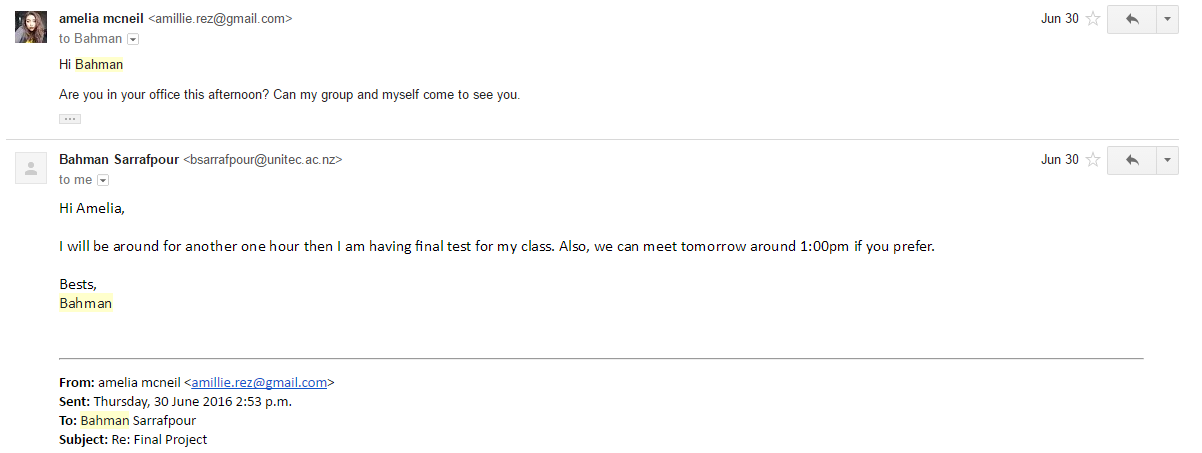












## Personal Reflection

### Amelia

### Marzouq

### Nawaf

### Oshada

# Appendix B

## Client Code

## Server Code

## Database SQL Dump

# Bibliography

*Android Studio*. (2016, October 24). Retrieved from Android Studio Features: https://developer.android.com/studio/features.html

digicert. (2016). *What Is SSL (Secure Sockets Layer) and What Are SSL Certificates?* Retrieved from digicert: https://www.digicert.com/ssl.htm

Garrett, J. J. (n.d.). *User Interface Design Basics*. Retrieved from usability.gov: https://www.usability.gov/what-and-why/user-interface-design.html

*MySQL Administration Guide*. (2015). Retrieved from Novell: http://www.novell.com/documentation/nw65/web\_mysql\_nw/data/aj5bj52.html

NetBeans. (2016). *Netbeans IDE - The smart and faster way to code*. Retrieved from Netbeans: https://netbeans.org/features/

Oracle. (n.d.). *Oracle* . Retrieved from Oracle Glassfish Server: http://www.oracle.com/us/products/middleware/application-server/050870.pdf

Rajput, M. (2015, May 20). *Why Android Studio Is Better For Android Developers Instead Of Eclipse*. Retrieved from DZone/Mobile Zone: https://dzone.com/articles/why-android-studio-better

Saab, C. B., Haddad, S., Coulibaly, D., Melliti, T., Moreaux, P., & Rampacek, S. (2009). An Integrated Framework for Web services orchestration. (L.-J. Zhang, Ed.) *International Journal of Web Services Research, Volume 6, Issue 4*, 3. Retrieved from http://www.irma-international.org/viewtitle/37386/

Security, D. (2016, September 26). *Salted Password Hashing - Doing it Right*. Retrieved from CrackStation: https://crackstation.net/hashing-security.htm#normalhashing

Server, W. (n.d.). *Microsoft*. Retrieved from Web Server (IIS) Role Overview: https://technet.microsoft.com/en-us/library/cc770634(v=ws.11).aspx

*Web Server (IIS) Role Overview*. (2016). Retrieved from Microsoft: https://technet.microsoft.com/en-us/library/cc770634(v=ws.11).aspx

# Project proposal

Appointment Management System

Project Team

2016

Marzouq 1380949 | Amelia 1388244 | Nawaf 1377387 | Oshada 1434048

Table of Contents

[1. Executive Summary 1](#_Toc457819229)

[2. Goal 2](#_Toc457819230)

[3. Background 2](#_Toc457819231)

[4. Objectives 2](#_Toc457819232)

[5. Justification 3](#_Toc457819233)

[6. Scope 3](#_Toc457819234)

[6.1 Functional Requirements 4](#_Toc457819235)

[6.2 Exclusions 4](#_Toc457819236)

[7. Project Team 5](#_Toc457819237)

[8. Methodology 5](#_Toc457819238)

[9. Deliverables 6](#_Toc457819239)

[10. Milestones and Tasks 7](#_Toc457819240)

[11. Resource Requirements and Costs 8](#_Toc457819241)

[12. Risk Analysis Checklist 9](#_Toc457819242)

[12.1 Generic Risk Checklist 9](#_Toc457819243)

[12.2 Specific Project Risks 11](#_Toc457819244)

[13. Quality Assurance Process 12](#_Toc457819245)

[14. Project Work Plan 13](#_Toc457819246)

[15. Intellectual Property 13](#_Toc457819247)

[16. Confidentiality 13](#_Toc457819248)

[17. Declaration 14](#_Toc457819249)

[18. Approvals 15](#_Toc457819250)

### Executive Summary

From time to time students require office hour access with lecturers to address and resolve any learning related issues and difficulties. Currently the practice is for students to usually seek a particular lecturer by visiting the staff room or office to either deal with a learning matter or make arrangements to address the matter at a more suitable time. More often this practice results in a continuous flow of students desiring to have out of class access to their lecturers. Consequently, this causes frustrations for both students and lecturers, since lecturers are interrupted, and students experience delays. To minimize the frustration’s lecturers have adopted their own policies to regulate and manage the flow of students requiring further learning assistance. As these policies vary between lecturers, the system continues to be problematic and confusing.

The proposal is to set up a computerised system with a common clear standard to reduce inefficiencies arising from inconsistent policies. More so the aim is to provide a useable solution to lessen the burden and frustrations experienced by both students and lecturers. The system will be deployed via a web-based application and mobile application allowing lecturers to advertise their availability to students requiring additional office hour access. The system will also offer both students and lecturers the ability to manage their office hour appointments.

### Goal

The target of the project is to deliver an efficient and more effective way of making appointments for both the students and lecturers. The web-based application and mobile application must provide a useable interface that facilitates the appointment process and management for both students and lecturers.

### Background

Unitec is the biggest Institute in New Zealand with over 16,000 attending students and over 1000 teaching staff. Students and teaching staff are located across three campuses. All campuses are located in Auckland, with the main campus situated in Mt. Albert, the other two situated in Henderson and Albany.

Given the large size of Unitec it is imperative to efficiently manage the logistics for office hour appointments between students and lecturers. Currently there is no set standard policy that defines a best practice, and no system in place to facilitate the process.

### Objectives

1. Building a web application and mobile application for making office hour appointments between students and lecturers.
2. Provide students with priori information as to when lecturers are available for an appointment to be made.
3. Minimise the time taken to set up an appointment.
4. Both the web-application and mobile application should also provide some integration with lecturer’s Outlook based calendar.
5. Enable appointments to be amended after these have been made.
6. Provide secure communication.

### Justification

The project was considered as many students were finding it difficult to get an appointment with their lecturers. Based on feedback provided by students at Unitec there is no standard clear way of contacting a lecturer. Many staff members have resorted to defining their own policies for students wishing to meet with them causing inadvertent frustrations, confusion, and complications.

At times a lecturer could be busy and cannot check all their emails to see the student emailed them to make an appointment, consequently students will just show up unexpectedly and interrupt the staff members. Another example is sometimes a student will show up and there are many students lined up waiting to see the same lecturer causing human traffic. On other occasions students may find a sign on the door indicating the member of staff is only vacant on particular times.

Both students and the staff need an organized and consistent way of arranging and managing meetings. We have researched the project in terms of what skills it takes to create a web application as this can provide convenient wide access across all campuses. Using this idea as our final project, we intended to improve the current system for making appointments between lecturer and student by facilitating the interaction and reduce inconsistencies.

This project will provide an opportunity to create a useful web application, and further develop our programming skills across a wide area of ICT technologies span ranging from client-side to server-side as well as database design and implementation. As the system we are developing is distributed, hence having a background in computer networking will be very useful.

### Scope

Our scope is to create an appointment booking and management web application and also a mobile application that will assist students and lecturers in making appointments between each other. The desire of this new process of making an appointment through the web application and mobile application is to provide greater standardisation and reduce current inefficiencies arising from inconsistent practices.

The web application will serve as a service for lecturers and student to interact. The web application will assist students in making an appointment through a structured and uniform calendar based system, which will save resources over the unstructured system currently in use. The creation of this web application and the mobile application will encourage a common standard.

### Functional Requirements

**Web Contents**

The web application will have several user views. There will be a common navigation interface across all user views to assist with usability of the web-application. The user views will include:

Home Page

* Information about the website
* Demo how to set up appointments

Profile Page

* Student profile view /Lecturer profile view
* Upcoming appointments
* Saved lecturers/students based on classes

Appointment Page

* Calendar view
* Booking appointments

Lecturers Page

* Preview of the lecturer’s name and picture if available
* Description of the lecturer’s details including their position in Unitec and the papers they teach

Contact

* Contact details
* Email for trouble logging in

### Exclusions

1. There will be no desktop application, as the client interaction is web-based.
2. There will be no IOS compatible application. However, the aforementioned devices can access the system via their respective web browsers.

### Project Team

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Unitec Team members** | **Name** | **Student ID** | **Email** | **Phone** |
| Marzouq Almarzooq | 1380949 | [e-9p@hotmail.com](mailto:e-9p@hotmail.com) | 0222584444 |
| Amelia McNeil | 1388244 | [Amillie.rez@gmail.com](mailto:Amillie.rez@gmail.com) | 0221654743 |
| Nawaf Altuwayjiri | 1377387 | [nawaf.s.h@hotmail.com](mailto:nawaf.s.h@hotmail.com) | 0223914544 |
| Oshada | 1434048 | [Oshada78@gmail.com](mailto:Oshada78@gmail.com) | 02040851845 |

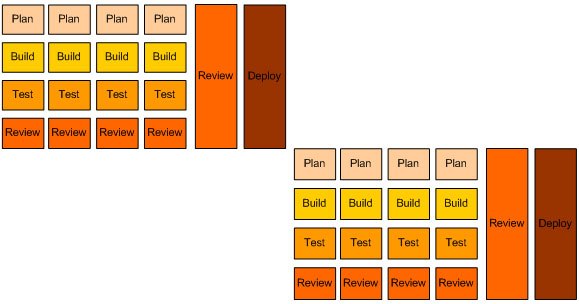
|  |  |  |  |
| --- | --- | --- | --- |
| **Project Supervisor from sponsoring company.** | Bahman  Sarrafpour | Department of computing  Unitec Institute of Technology | 098154321 ext.6043  bsarrafpour@unitec.ac.nz |

### Methodology

**Agile methodology - Scrum Development**

Initially we looked at a waterfall method so that we could complete a phase before moving onto the next phase this would keep the project organized, however this development carries a lot of risk because in this method we cannot revisit a phase, once it is completed you are water falling you cannot go back. We decided that the waterfall method was not suitable due to high risk factor.

A more risk averse approach is to use an Agile methodology such as Scrum. Scrum development is achieved with sprints. Sprints would allow us to focus on delivering independent, tested features within manageable workloads, hence minimising risk as shown in the diagram below. This approach will allow us to plan, build, test and review each feature and then work on the next feature in a more structured and reliable fashion.



### Deliverables

|  |  |  |
| --- | --- | --- |
| **Deliverable** | **Description** | **Contribution to the project** |
| **Project proposal** | First documentation, proposing the idea. | 10% |
| **Project resources** | Gathering resources needed for the project software, hardware etc. | 20% |
| **Project requirements** | Computers, software, hardware | 10% |
| **Client-Side Application Development** | Development of the web application and the mobile application | 10% |
| **Server-Side Web Development** | Development using NetBeans | 10% |
| **Relational Database Management System Development {Multi-Tier System Development}** | Development using MySQL | 10% |
| **Final Web and mobile application** | Final stages on building the web application and mobile application. | 20% |
| **Final Documentation** | Documents ready for submission. | 5% |
| **Project Presentation** | The final presentation. | 5% |

### Milestones and Tasks

|  |  |
| --- | --- |
| Milestone and Tasks | Description |
| Project meeting 1 | First project meeting |
| Hand-in project proposal | Submit final proposal |
| Proposal defence | Presentation and defence in front of panel |
| User Stories | Create User stories |
| Scrum Sprints | Developing the applications in individual sprints |
| Testing final project | Test completed applications |
| Hand-in project documentation | Finalise projects final documentation |
| Project presentation and defence | Final Presentation |
| Close project | Project is finished |

### 

### Resource Requirements and Costs

|  |  |  |
| --- | --- | --- |
| **Resources provided by Unitec** | | |
| **Resource** | **Quantity** | **Status** |
| PC | 3 | Available |
| Swipe Cards | 4 | Available |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Software needed** | | | | |
| **Resource** | **Quantity** | **Status** | **Price** | **Reference** |
| Microsoft Office (Word, PowerPoint, Excel etc.) | 3 | Available | Provided by Unitec | |
| MySQL | 3 | Available | Free | <https://www.mysql.com/downloads/> |
| J Query | 3 | Available | Free | <https://jquery.com/download/> |
| NetBeans | 3 | Available | Free | <https://netbeans.org/> |
| Adobe -Dreamweaver | 3 | Available | Free | Provided by Unitec |
| VisualStudio | 3 | Pending | Free | Provided by Unitec |
| AndroidStudio | 3 | Pending | Free | Provided by Unitec |

### Risk Analysis Checklist

### 12.1 Generic Risk Checklist

The following risk checklist is a generic model and is used to give an overall (non-specific) picture of the project’s risk factors. It can be used to compare the relative risks to the organisation of a number of different projects.

|  | **Low (Yes)** | | | | **RISK** | | | | **(No) High** | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | | 4 | 5 | 6 | | 7 | 8 | 9 |
| **Inherent Risks** |  |  |  |  | |  |  |  | |  |  |  |
| **Project Objectives** |  |  |  |  | |  |  |  | |  |  |  |
| Is the project small? |  |  |  |  | |  |  |  | |  | **√** |  |
| Is the project of minor importance to the business? |  |  |  |  | |  | **√** |  | |  |  |  |
| Is the project functionally straightforward? |  |  |  |  | |  | **√** |  | |  |  |  |
| Are several parties able to define the requirements? |  |  |  |  | |  |  | **√** | |  |  |  |
| Is the subject area well documented? |  |  |  |  | | **√** |  |  | |  |  |  |
| Are preceding projects well documented? |  |  |  |  | |  |  | **√** | |  |  |  |
| **User Organisation** |  |  |  |  | |  |  |  | |  |  |  |
| Does the project maintain existing user procedures? |  |  |  |  | |  |  |  | | **√** |  |  |
| Is other organisational change unlikely during the project? |  |  |  |  | |  |  |  | | **√** |  |  |
| Are the users grouped in one location? |  | **√** |  |  | |  |  |  | |  |  |  |
| **Technology** |  |  |  |  | |  |  |  | |  |  |  |
| Is tried hardware being used? |  |  |  | **√** | |  |  |  | |  |  |  |
| Is tried software being used? | **√** |  |  |  | |  |  |  | |  |  |  |
| Can custom programming be avoided? |  |  |  |  | |  | **√** |  | |  |  |  |
| Is the project technically straightforward? |  |  |  |  | |  |  | **√** | |  |  |  |
| Is the quality of existing data good? |  |  |  |  | | **√** |  |  | |  |  |  |
| **Acquired Risks** |  |  |  |  | |  |  |  | |  |  |  |
| **Scope and Approach** |  |  |  |  | |  |  |  | |  |  |  |
| Is the project scope well defined and agreed? |  |  | **√** |  | |  |  |  | |  |  |  |
| Is the project approach well defined and agreed? |  |  | **√** |  | |  |  |  | |  |  |  |
| **Project Organisation** |  |  |  |  | |  |  |  | |  |  |  |
| Are people’s roles clearly defined? |  |  |  |  | | **√** |  |  | |  |  |  |
| Are users committed to the project? |  |  | **√** |  | |  |  |  | |  |  |  |
| Are staff able to commit sufficient time to the project? | **√** |  |  |  | |  |  |  | |  |  |  |
| Are the required skills available? |  | **√** |  |  | |  |  |  | |  |  |  |
| Does backup exist for all members of the project? |  | **√** |  |  | |  |  |  | |  |  |  |
| Are political and personal relationships good? | **√** |  |  |  | |  |  |  | |  |  |  |
| Is the project independent of third parties? |  | **√** |  |  | |  |  |  | |  |  |  |
| Can a small group achieve the design? |  | **√** |  |  | |  |  |  | |  |  |  |
| Can a “Big Bang” implementation be avoided? |  |  |  |  | | **√** |  |  | |  |  |  |
| **Experience, Training and Support** |  |  |  |  | |  |  |  | |  |  |  |
| Does the IT team know the technology? |  | **√** |  |  | |  |  |  | |  |  |  |
| Do the users know the technology? |  |  |  | **√** | |  |  |  | |  |  |  |
| Is the technology well supported? |  |  | **√** |  | |  |  |  | |  |  |  |

### 

### 12.2 Specific Project Risks

This section identifies risks that are specific to this project and are in addition to the generic risks considered above.

| Issue | Probability | Impact | Schedule | Issue/Action |
| --- | --- | --- | --- | --- |
| Misplaced documentation | Low | Low |  | Make sure to save regularly and make multiple copies on USB’s, emails and computers to be cautious. |
| Project sponsor changes the original scope | Medium | High |  | Project will be delayed. |
| Team member does not contribute or is unable to work due to unexpected circumstances | Medium | Low |  | Some case the member could be slacking and not paying full attention with helping the in the project due to lack of motivation.  Family emergency arises or member falls ill and cannot help the team until said person’s health has improved.  There are no ways to avoid unexpected circumstances. In this situation the other group members will contribute work for that individual. |
| Security breach (computer virus) | Low | Medium |  | Security breaches can occur unexpectedly to prevent this we need to make sure the computers we are working on have anti-virus security installed. If we do get a virus and we are not protected it can affect our whole project. |
| Proposal gets rejected | Low | Medium |  | The project could get rejected due to many reasons including not enough supporting facts.  Before submitting the proposal, a lot of research is required stating the many benefits of the applications and how it could impact the students and lecturers.  We will make sure all the key points are clearly stated in the proposal. |
| PC cannot connect to server | Low | Low |  | We will have trouble shoot programs in backup to prevent such scenarios and to make sure there is always an available server for the PC to connect to. |

### 

### Quality Assurance Process

To assure quality of the project we will have iterative tests, this way we won’t be leaving testing to the last stage, we will be building a feature and testing it to assure quality.

Another technique to maintain quality assurance we will use will be pilot testing by using our fellow students to try the feature and provide us with feedback of how the feature worked for them was it easy to use and why, what problems they experienced etc. Pilot testing.

### Project Work Plan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Task/Milestone** | **July** | **August** | **September** | **October** | **November** |
| Proposal |  |  |  |  |  |
| Proposal Presentation |  |  |  |  |  |
| User stories/ Use Cases |  |  |  |  |  |
| Scrum Sprints |  |  |  |  |  |
| Final Report Write-up |  |  |  |  |  |
| Final Presentation/  Demo |  |  |  |  |  |

### 

### Intellectual Property

The deliverables created in this project will belong to Bahman Sarrafpour the project supervisor and the project team.

### Confidentiality

UNITEC is not permitted to retain a copy of the project documentation for academic purposes including: assessment, accreditation and peer review, without permission.

### Declaration

I am not employed by the sponsoring organization.

The people involved with this project from the sponsoring organization are not related to me or close friends.

I declare that the information in this proposal is, to the best of my knowledge, correct.

Team member name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Team member name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Team member name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Team member name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### Approvals

Date approval in principle received:

I agree that the scope, objectives, resource estimates and plans given in this project proposal describe my general requirements for the project. I confirm that I have the authority to approve the expenditure outlined in this proposal. I understand that this is a student project and that UNITEC and the students will endeavour to provide the services described but for whatever reason may not be able to do so.

Project Sponsor \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date \_\_\_\_/\_\_\_\_/\_\_\_\_\_\_\_\_

I agree that the project described in this project proposal is generally suitable to meet the learning outcomes for the course ISCG 7431 Capstone Project.

UNITEC Project Coordinator \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date \_\_\_\_/\_\_\_\_/\_\_\_\_\_\_\_\_