

**Bachelor of Information Technology**

Semester One, 2019

Project Product Document

|  |  |
| --- | --- |
| ****Project Name:**** | SGRANCh Compendium App |
| ****Project Code:**** | 1-1-SGCRANCh-2019 |
| ****Module Code:**** | IT7x01 |

**Written by:**

|  |  |  |
| --- | --- | --- |
| Student names and contact details | | |
| Name | Phone | Email |
| Chris Simmons | 0211769080 | Christopher.Simmons@protonmail.ch |
| Taylor Everett | 0226779068 | tayloreverett@gmail.com |

**Completed Date:**

**Contents**

[System Design 3](#_Toc10205073)

[Introduction 3](#_Toc10205074)

[Interface design 3](#_Toc10205075)

[Wireframe 3](#_Toc10205076)

[Site map 4](#_Toc10205077)

[Database design 4](#_Toc10205078)

[ERD Diagram 5](#_Toc10205079)

[Application Use logic 6](#_Toc10205080)

[Flow Chart 6](#_Toc10205081)

[Test Design 7](#_Toc10205082)

[Interface Test Plan 7](#_Toc10205083)

[Purpose 7](#_Toc10205084)

[Background 7](#_Toc10205085)

[Criteria 8](#_Toc10205086)

[Methodology 9](#_Toc10205087)

[Black Box 9](#_Toc10205088)

[White Box 10](#_Toc10205089)

[Conclusion 11](#_Toc10205090)

[Backend Test Plan 12](#_Toc10205091)

[Purpose 12](#_Toc10205092)

[Background 12](#_Toc10205093)

[Criteria 12](#_Toc10205094)

[Methodology 12](#_Toc10205095)

[Use Case 1 13](#_Toc10205096)

[Use Case 2 13](#_Toc10205097)

[Use Case 3 14](#_Toc10205098)

[Use Case 4 14](#_Toc10205099)

[Conclusion 15](#_Toc10205100)

[Recommendations 15](#_Toc10205101)

[Appendix B: Student made form for client. 16](#_Toc10205102)

# System Design

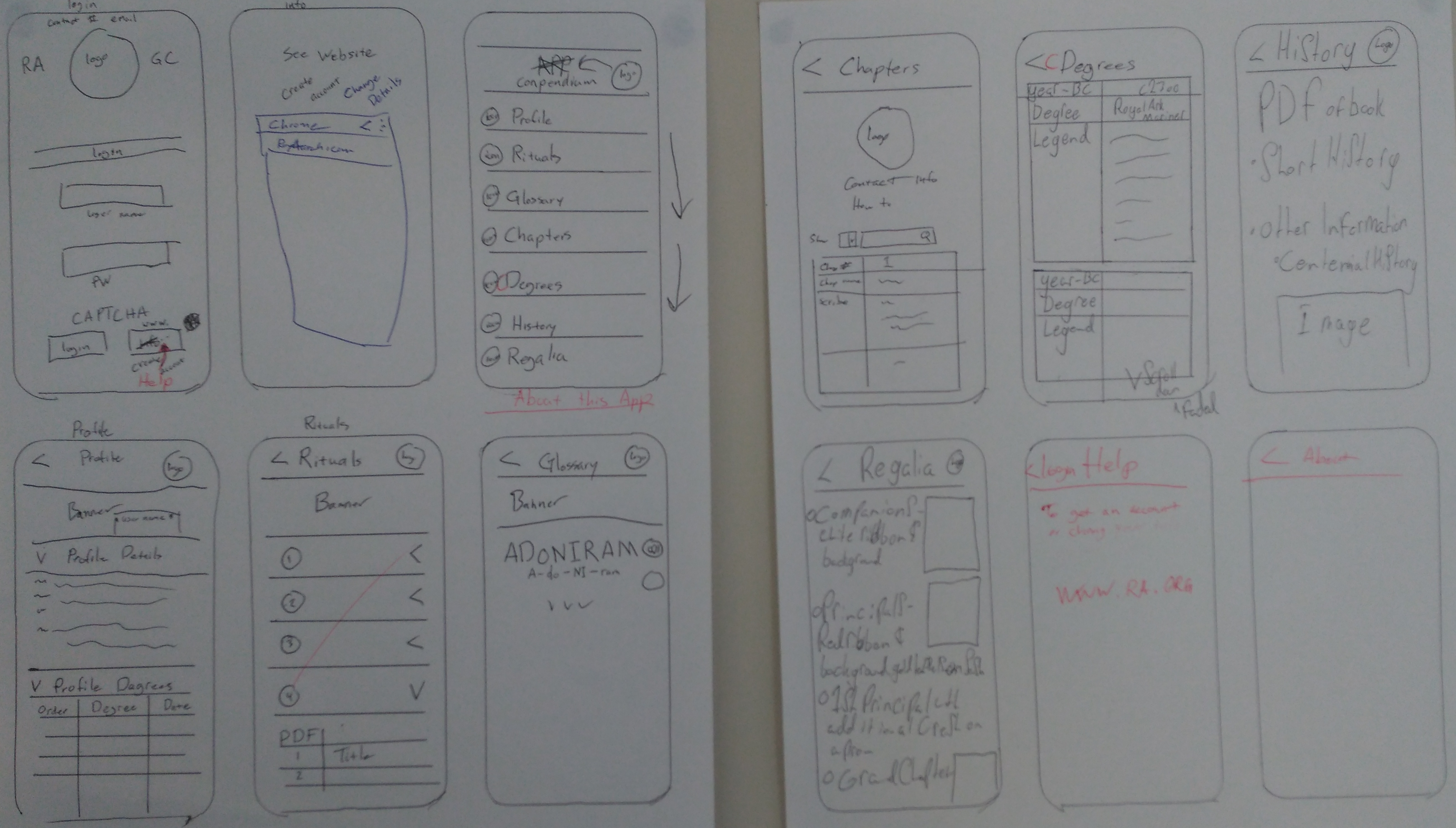
## Introduction

This documentation will describe the design of the system and the reasons behind the design decisions. The initial design was based on what we could understand from interviews and documentation given to us by the client which resulted in the creation of our wireframe. The next step was to create a paper prototype to help show the client what we were going to build, this became a lengthy process since the idea was very rough. After we gave the client more time to adjust and tune our design, we started creating our application in Unity.

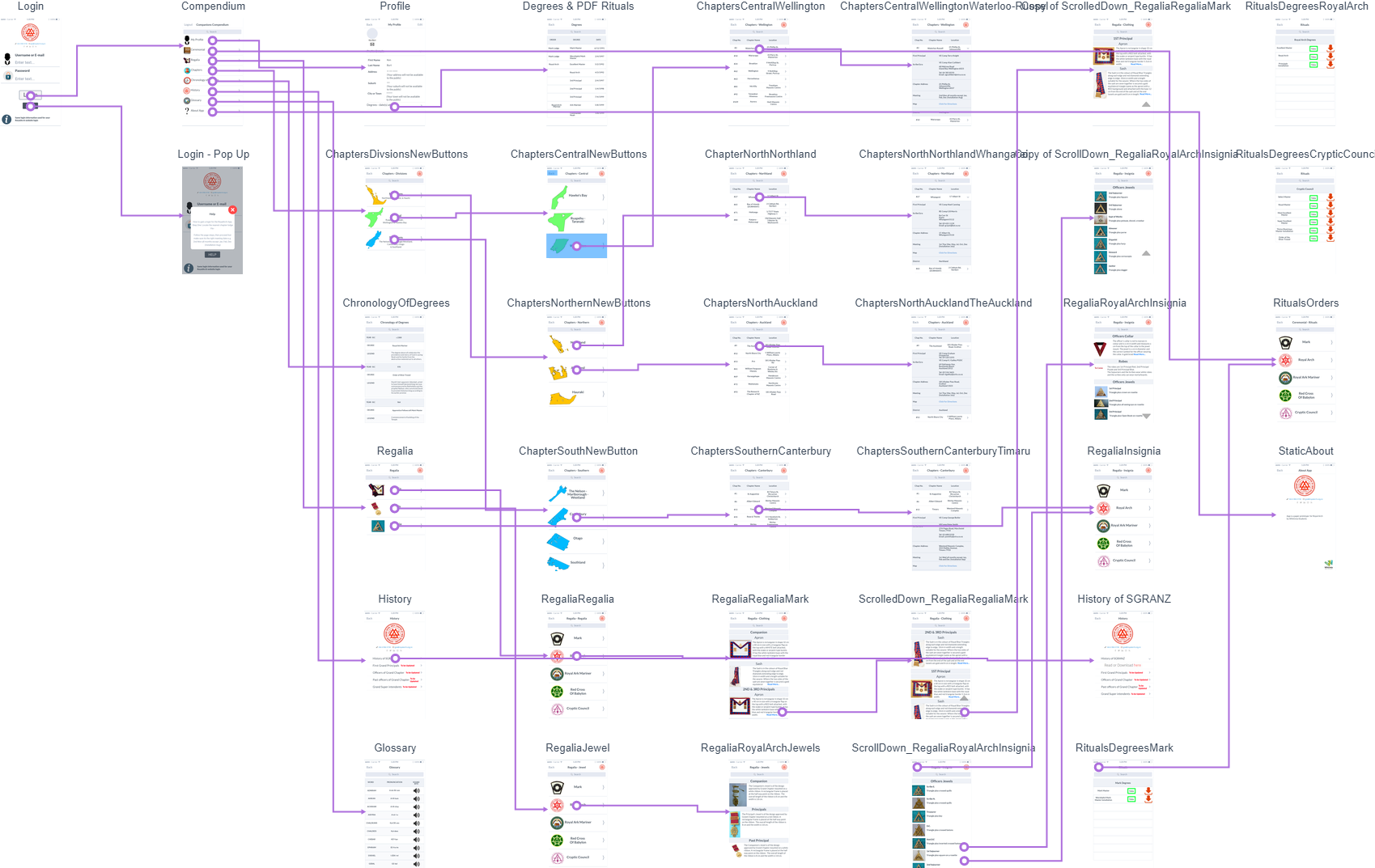
## Interface design

Through our initial consultation, our client was able to tell us what he was looking forward to in an application by comparing his idea to the IOS Tips & Tricks section on his phone. We later looked for screenshots to help fill in any gaps for the display. We made a wireframe on paper to set up the look of the application then followed it with a paper prototype. Below you will see our hand drawn wireframe followed by a site map of our paper prototype.

### Wireframe



### Site map



## Database design

The project had a set back since we couldn’t attain any information about the client’s systems, so we had to create our own database designed on what we knew from a power point provided by the client. This PowerPoint had a screenshot of a user’s profile which was used to determine attributes and data types. Every other table other than ‘Profile’ was fabricated by us (the students) through questioning the client and basic understandings of relational databases. Below is an ERD diagram of the database.

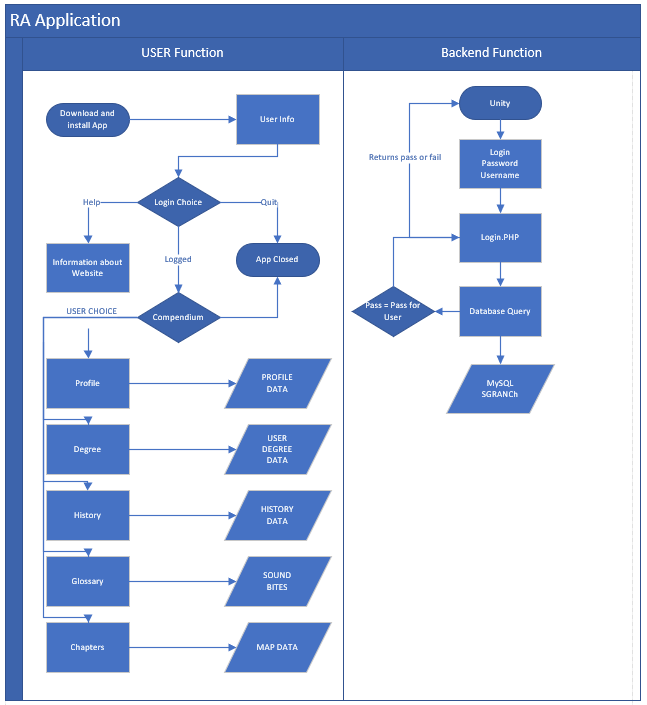
### ERD Diagram



## Application Use logic

Logic for the application was based on what we thought would work with the idea that was given to us. The idea was vague, so for the most part we used the paper prototype with the client to get a deeper understanding. Below is a flow chart showing user and backend functions. User functions are based on navigation and the backend function is focused on how the application’s logic would apply the login functionality.

### Flow Chart



## Test Design

For our test plans we chose to use methods we have used in the past such as ‘black box’ and ‘white box’ testing for the front end and ‘use case’ for the backend of the application. We didn’t use a more elaborate testing plan such as ‘unit test’ because of the unknown elements of the clients’ systems. We thought it wouldn’t have had any more value than what we used. The current application being build may not be suitable for future use other then being a template for the next iteration to build on. If the next iteration were to use the same methods, unit testing with integrated client systems would be necessary.

# Interface Test Plan

## Purpose

This test plan is designed to check the applications frontend development to ensure we have properly outlined the applications look and feel to test the interface.

## Background

The project started off as just an idea, we have gone through an extended design phase because a proper analysis of the client’s systems hasn’t been made possible yet. Through many meetings and open conversations with the use of ‘basecamp’ with the client, changes to the first wireframe have been made, if the project hadn’t been changed from a finished product to a prototype, the extended design phase wouldn’t had been possible.

## 

## Criteria

|  |  |  |
| --- | --- | --- |
| Number | Item | Implemented |
| 1 | Login Scene; Username input, password input, logo, contact info, and Information about website credentials. | Yes |
| 2 | Compendium Scene, can access all other Scene ‘main navigation through the application’ | Yes |
| 3 | My Profile Scene, have a placement for Profile Details and Degree earned. |  |
| 4 | Ceremonial Scene, have each order listed and open to a new Scene for rituals to be listed. |  |
| 5 | Regalia Scene, have a selection for regalia, jewels, and insignia when selected have a list of each order to select to view |  |
| 6 | Chapter Scene, selection of Divisions, selection of districts, list of chapters. |  |
| 7 | Chronology Scene, list of data. |  |
| 8 | History Scene, drop downs: have link to download for ‘SGRANZ’ have to be Updated for all other history. |  |
| 9 | Glossary Scene, List of words given and Pronunciation, and a placement four sound bite button. |  |
| 10 | About App Scene, talk about how the app has been made by Whitireia students. |  |
| 11 | Title on Compendium Scene reads “Companions Compendium” | Yes |

## 

## Methodology

For this test plan we felt it was necessary to use black and white box testing.

*What is Black Box testing?*

In Black-box testing, a tester doesn't have any information about the internal working of the software system. Black box testing is a high level of testing that focuses on the behaviour of the software. It involves testing from an external or end-user perspective. Black box testing can be applied to virtually every level of software testing: unit, integration, system, and acceptance.

*What is White Box testing?*

White-box testing is a testing technique which checks the internal functioning of the system. In this method, testing is based on coverage of code statements, branches, paths or conditions. White-Box testing is considered as low-level testing. It is also called glass box, transparent box, clear box or code base testing. The white box Testing method assumes that the path of the logic in a unit or program is known.

## Black Box

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test item no.** | **Test Item** | **Test Step** | **Expected Result** | **Actual Result** | **Test Pass/ Fail** | Notes |
| 1.A | Login Scene | User Taps login button | Application changes scene to ‘Compendium’ scene | Application changes scene to ‘Compendium’ scene | Pass |  |
| 1.B | Login Scene | User Taps Help | Application changes scene to ‘Help’ scene | Application changes scene to ‘Help’ scene | Pass |  |
| 2.A | Compendium Scene ‘Navigation’ | User Taps any of the tabs. | Application can navigate to the appropriate option. | Application can navigate to the appropriate option. | Pass |  |
| 2.B | Compendium Scene ‘Navigation’ | User Taps the back button on the top of the application | Application can return to compendium Scene with the use of the back button | Application can return to compendium Scene with the use of the back button | Pass |  |
| 3 | Profile Scene | User Taps Profile Scene | Application displays this scene which will house user data later. | Application displays this scene which will house user data later. | Pass |  |
| 4.A | Ceremonial Scene | User Taps  Ceremonial Scene | Application displays this scene which has options for Order selection. | Application displays this scene which has options for Order selection. | Pass |  |
| 4.B | Ceremonial Scene | User Taps an Order in the  Ceremonial Scene | Application displays rituals in a table. | Application displays rituals in a table. | Pass |  |
| 5.A | Regalia Scene | User Taps Regalia Scene | Application displays 3 options regalia, jewels, and insignia. | Application displays 3 options regalia, jewels, and insignia. | Pass |  |
| 5.B | Regalia Scene | User taps regalia, jewels, or insignia. | Application displays this scene which has options for Order selection. | Application displays this scene which has options for Order selection. | Pass |  |
| 5.C | Regalia Scene | User taps an Order | Application displays images and Text about the regalia, jewels, or insignia. | Application displays images and Text about the regalia, jewels, or insignia. | Pass |  |
| 6A | Chapter Scene | User Taps Chapter Scene | Application displays different division selection | Application displays different division selection | Pass |  |
| 6.B | Chapter Scene | User Taps a Division | Application displays different district selection | Application displays different district selection | Pass |  |
| 6.C | Chapter Scene | User taps a District | Application displays chapters from the district in a list with drop down option | Application displays chapters from the district in a list with drop down option | Pass |  |
| 7 | History Scene | User Taps History Scene | Application displays a history selection with a hyperlink to a PDF | Application displays a history selection with a hyperlink to a PDF | Pass |  |
| 8 | Glossary Scene | User taps Glossary Scene | Application displays a list of pairing words with a sound image. | Application displays a list of pairing words with a sound image. | Pass |  |
| 9 | About App Scene | User taps About App Scene | Application displays a page detailing who made the application. | Application displays a page detailing who made the application. | Pass |  |
| 10 | Chronology Scene | User taps Chronology Scene | Application displays information on the organization. | Application displays information on the organization. | Pass |  |

## White Box

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test item no.** | **Test Item** | **Test Step** | **Expected Result** | **Actual Result** | **Test Pass/Fail** | **Notes** |
| 1.A | Login Scene | Login button remains grey till there is an input | Login button is grey with blank input | Login button is grey with blank input | Pass |  |
| 1.B | Login Scene | Login input changes button colour | Login button is no longer grey after input filled | Login button is no longer grey after input filled | Pass |  |
| 2 | Ceremonial Scene | User navigates passed orders to rituals | Rituals are all greyed out. | Rituals are all greyed out. | Pass | Should show grey till the DB is connected |

## Conclusion

This test has helped us to ensure the interface met the criteria and that the application has the basic design implementation that the client expressed. With this test plan now implemented we can take the next step in backend development.

# Backend Test Plan

## Purpose

This test plan is designed to check the applications backend development to ensure we have properly executed the functions asked to be included in this application.

## Background

As this project progressed, the idea of what the application should look like has changed considerably but the functionality has remained the same. Unfortunately, we could not get any of the information on the clients’ systems and have had to create a lot of our own environment.

## Criteria

|  |  |  |
| --- | --- | --- |
| Number | Item | Implemented |
| 1 | Use of a MySQL | Yes |
| 2 | Use of a PHP server (as an API) | Yes |
| 3 | Connection to the Database | Yes |
| 4 | Log in to application using Database | No |
| 5 | User Profile showing correct information | Yes |
| 6 | User can only view Rituals they have degrees for | Yes |

## Methodology

For this test plan we have chosen to implement a ‘use case’ on four functions required by our project; Login, Profile, Chapters, and Ceremonial. We felt that this form of testing was the best to implement because of the amount of data we lacked about the clients’ systems. With ‘use case’ we can focus on the user actions and response of the system to the user actions.

## 

## Use Case 1

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | Log in | |
| **Scenario:** | Logging In | |
| **Triggering Event:** | User has input their data | |
| **Brief Description:** | Users are trying to log into the application. | |
| **Actors:** | Users | |
| **Stakeholders:** | Students, Administrators, and Clients | |
| **Preconditions:** | User already has an account and has downloaded the application to their android devise. | |
| **Post-conditions:** | User has logged out of the application. | |
| **Flow of Actions:** | Actor | System |
| User data has been entered the login input boxes for username and password.  User now taps the login button. | Application passes a form through PHP to check if user information matches the DB.  If information is correct form will return a value of ‘0’ and allow the next scene to open. |
| **Exception Conditions:** | The Data the User entered is correct and the User is then able to have access to the application. | |

## Use Case 2

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | Profile | |
| **Scenario:** | Wanting to view personal data | |
| **Triggering Event:** | User Has Logged into their account successfully | |
| **Brief Description:** | Users are logged into the application and wish to view their personal data | |
| **Actors:** | Users | |
| **Stakeholders:** | Students, Administrators, and Clients | |
| **Preconditions:** | User already has an account and has downloaded the application to their android devise. | |
| **Post-conditions:** | User has logged out of the application. | |
| **Flow of Actions:** | Actor | System |
| User data has been entered the login input boxes for username and password. | Database has the User account already.  Application passes a form to check if user |
| **Exception Conditions:** | The Data the User entered is correct and the User is then able to have access to the application. | |

## 

## Use Case 3

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | Ceremonial | |
| **Scenario:** | Looking for Rituals | |
| **Triggering Event:** | User has input their data | |
| **Brief Description:** | Users are trying to log into the application. | |
| **Actors:** | Users | |
| **Stakeholders:** | Students, Administrators, and Clients | |
| **Preconditions:** | User already has an account and has downloaded the application to their android devise. | |
| **Post-conditions:** | User has logged out of the application. | |
| **Flow of Actions:** | Actor | System |
| User data has been entered the login input boxes for username and password. | Database has the User account already.  Application passes a form to check if user |
| **Exception Conditions:** | The Data the User entered is correct and the User is then able to have access to the application. | |

## Use Case 4

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | Chapter | |
| **Scenario:** | Looking for Chapter | |
| **Triggering Event:** | User taps the Chapter tab then taps a Division, District and then a chapter of their choosing. | |
| **Brief Description:** | Users selects a Division, District and then a chapter from Chapter Scene | |
| **Actors:** | Users | |
| **Stakeholders:** | Students, Administrators, and Clients | |
| **Preconditions:** | User already has already successfully logged in and is viewing compendium scene and has tapped Chapter.  User will have full access to all chapters within Unity application | |
| **Post-conditions:** | User has logged out of the application. | |
| **Flow of Actions:** | Actor | System |
| User taps Chapter tab and is met with a selection of Divisions, Districts then chapters to choose from.  User taps the Division they want.  User sees the District that division has then selects their chapter of choosing. | Dynamic DB GET |
| **Exception Conditions:** | User has no access to the chapters tab either because of errors or the chapters haven’t committed to the DB yet.  User is not logged in. | |

## Conclusion

From the testing done we can see that the applications functionality is meant to be working with the intended Users’ actions. At the time of this document students could not get all systems to operate as intended so some elements have been made static for data that would otherwise be fetched from the database. We are currently focusing on the product artefact to see if we can fix our system failures before the deadline.

## Recommendations

Once the client can release the system data, the project code may need to be severely changed, it is highly recommended to do unit testing.

With addition to unit testing, testing plans for both android and IOS should be implemented.

# Appendix B: Student made form for client.

**Introduction**

Purpose of this document is to help the following iteration teams to have critical information ready for the completion of the compendium application. This will allow the next groups to appropriately scope the next steps needed to connect the application to the websites.

**System Parts**

From what we understand, the Supreme Grand Royal Arch web systems are required to be part of the application final product. From what we know, the systems in place are not localized to one system. We would like each system to be defined with name, configuration, location, company and level of access we students may have.

**Website:**

**Degree database:**

**Other 1:**

**Other 2:**

**Database Details**

A database Schema for each Database would be required. For the Profile information to be displayed in the application, students would also need a blank copy of the database holding that data.

**Permissions**

For each System we would like to know if we may add or change attributes, this is in regard to adding new tables, altering existing tables for functionality and security reasons. This will also be needed for adding PHP.

**Website**

An account on the current website for students to use for access.

Website source code would be appreciated for login functionality.

Is there an admin panel and who has access?