ONLINE ACADEMIC APPOINTMENT SCHEDULING SYSTEM

TEH BIN SHUN

A project report submitted in partial fulfilment of the requirements for the award of Bachelor of Science (Hons.) Software Engineering

Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

Jan 2017
DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature : ________________________________

Name : ________ Teh Bin Shun______________

ID No. : ___________ 1300515_____________

Date : _________________________________
I certify that this project report entitled “ONLINE ACADEMIC APPOINTMENT SCHEDULING SYSTEM” was prepared by TEH BIN SHUN has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science (Hons.) Software Engineering at Universiti Tunku Abdul Rahman.

Approved by,

Signature : _________________________

Supervisor : Dr. Victor Tan Hock Kim

Date : ___________________________
The copyright of this report belongs to the author under the terms of the copyright Act 1987 as qualified by Intellectual Property Policy of University Tunku Abdul Rahman. Due acknowledgement shall always be made of the use of any material contained in, or derived from, this report.

© 2017, Teh Bin Shun. All right reserved.
ABSTRACT

Academic staff such as professors, lecturers and researchers hold high responsibilities to improve learning environment by fulfilling their duties expectations at all times. Setting up and attending appointments is one of their duties. In UTAR, majority of academic staff use manual technique, the pen-and-paper technique to manage their appointment schedule. Multiple problems emerged due to lack of dedicated system for appointment scheduling, along with the problem where no fixed platform for requestor to make appointment. Therefore, an automated online academic appointment scheduling system is proposed as the solution for the current manual scheduling technique.

The proposed solution helps a lot in reducing human involvement for appointment scheduling which reducing the work load of academic staff. The human involvement will be reduced in approval of any making, bring forward/postpone or cancellation of appointment. It helps by obtaining the availabilities of every academic staff and compute the free time slots for requester to complete their appointment management by themselves without the consent of academic staff. Moreover, the system will be providing the feature of storing all the appointment information which enable easy tracking for every academic staff and requesters. And, along with several settings such as rules and google calendar synchronization, the system will indeed be the efficient solution for resolving the problems in current workflow of appointment scheduling.
CHAPTER

1 INTRODUCTION

1.1 Background 1
1.2 Problem Statement 1
1.3 Proposed solution 2
1.4 Project Objectives 3
1.5 Project scope 3
  1.5.1 The users of the system 3
  1.5.2 Scope coverage 4
  1.5.3 Out of Scope 8
  1.5.4 Conclusion 8

2 LITERATURE REVIEW AND FACT FINDING 11

2.1 Existing Online Appointment Scheduling System 11
  2.1.1 Acuity Scheduling 11
  2.1.2 FlexBooker 15
  2.1.3 Square 18
  2.1.4 Comparison between the scheduling system 22
2.1.5 Conclusion 22

2.2 Research Methodologies 24

2.2.1 Quantitative Research Approach 24

2.2.2 Qualitative Research Approach 28

2.2.3 Comparison between quantitative and qualitative research 33

2.2.4 Conclusion 34

2.2.5 Data Collection and Analysis 36

3 METHODOLOGY AND DEVELOPMENT TOOLS 48

3.1 System Development Methodology 48

3.1.1 Waterfall Model 48

3.1.2 Spiral Model 51

3.1.3 Incremental Model 53

3.1.4 Prototyping model 55

3.1.5 Rapid Application Development 57

3.1.6 Agile Model 58

3.1.7 Comparison between system methodologies 60

3.1.8 Conclusion 61

3.2 Development Tools 64

3.2.1 JQuery 64

3.2.2 Bootstrap 64

3.2.3 Laravel 65

3.2.4 PHP, HTML5, JavaScript, CSS3 and SQL 65

3.2.5 MySQL 66

3.2.6 Axure RP Pro 66

3.3 Project Plan 66

3.3.1 Work Breakdown Structure 66

3.3.2 Gantt Chart 68

4 PROJECT SPECIFICATION & DESIGN 73

4.1 Functional Requirements 73

4.2 Use Case 75
4.2.1 Use Case Diagram 76
4.2.2 Use Case Description 77
4.3 Sequence Diagram 89
4.3.1 Sequence Diagrams for Academic Staff 90
4.3.2 Sequence Diagrams for Requester 104
4.3.3 Sequence Diagram for Administrator 111
4.4 Entity Relationship Diagram 114
4.4.1 Logical Entity Relationship Diagram 114
4.4.2 Physical Entity Relationship Diagram 115
4.4.3 Data Dictionary 117
4.5 User Interface Design 126
4.5.1 Login Page UI 126
4.5.2 User Interfaces for Academic Staff 126
4.5.3 User Interfaces Design for Requester 133
4.5.4 User Interfaces for Administrator 136
4.6 Architecture Pattern 136

5 IMPLEMENTATION 138
5.1 Implementation of Modules 138
5.1.1 Modules for Academic Staff 138
5.1.2 Modules for Requester 164
5.1.3 Modules for Administrator 173
5.1.4 System Modules 175

6 TESTING 181
6.1 Unit Testing 181
6.1.1 Unit Test Cases 181
6.1.2 Unit Test Scripts 202
6.2 Integration Testing 203
6.2.1 Integration Test Cases 203
6.3 System Testing 207
6.3.1 System Test Cases 207
6.4 User Acceptance Test 210
6.4.1 User Acceptance Test Cases (for Academic Staff) 210
6.4.2 User Acceptance Test Cases (for Requester) 213
6.4.3 User Acceptance Test Cases (for Administrator) 214
6.5 Software Testing Tools 215
6.5.1 Selenium IDE 215

7 CONCLUSIONS AND RECOMMENDATIONS 216
7.1 Contribution of the Application 216
7.2 Limitation and Future Enhancements 216
7.3 Conclusion 218

7 REFERENCES 220

7 APPENDICES 223
LIST OF TABLES

Table 2.1: Comparison between online scheduling systems 22
Table 2.2: Strengths and limitations of quantitative research 28
Table 2.3: Strengths and limitations of qualitative research 32
Table 2.4: Comparison between quantitative and qualitative research (Castellan 2010) 33
Table 2.5: Common problems faced during appointment process 40
Table 2.6: Recommended features for appointment scheduling project 41
Table 2.7: Common problems encountered during appointment process 42
Table 2.8: Recommended features for appointment scheduling project 43
Table 3.1: Advantages of Waterfall model 49
Table 3.2: Disadvantages of waterfall model 49
Table 3.3: Appropriate situations for waterfall model 50
Table 3.4: The details of phases of spiral model 51
Table 3.5: Advantages of spiral model 52
Table 3.6: Disadvantages of spiral model 52
Table 3.7: Appropriate situations for spiral model 53
Table 3.8: Advantages of incremental model 54
Table 3.9: Disadvantages of incremental model 54
Table 3.10: Appropriate situations for incremental model 55
Table 3.11: Advantages of prototyping model 56
Table 3.12: Disadvantages of prototyping model 56
Table 3.13: Appropriate situations for prototyping model 56
Table 3.14: Advantages of RAD model 57
Table 3.15: Disadvantages of RAD model 58
Table 3.16: Agile Manifesto Principles 59
Table 3.17: Advantages of agile model 59
Table 3.18: Disadvantages of agile model 60
Table 3.19: Appropriate situations for agile model 60
Table 3.20: Comparison between system methodologies 60
Table 3.21: Project schedule table 68
Table 4.1 Description for Physical ERD 116
Table 4.2: Class-Responsibility-Collaborator Card for MVC architecture pattern 137
LIST OF FIGURES

Figure 2.1: Sample of Confirmations message 11
Figure 2.2: Settings for synchronization in Acuity Scheduling 12
Figure 2.3: Settings of availability in Acuity Scheduling 13
Figure 2.4: Booking process view 15
Figure 2.5: Clipboard view of Daily Recap 16
Figure 2.6: Sample of notifications about availability of time slot 16
Figure 2.7: Book multiple services view 20
Figure 2.8: Preview of Square Application for IOS and Android 20
Figure 2.9: Book multiple services view 21
Figure 2.10: Appointment scheduling technique used by academic staff 39
Figure 2.11: Problems encountered using pen-and-paper technique 39
Figure 2.12: Platforms used for appointment booking 42
Figure 3.1: The phases of Waterfall model 48
Figure 3.2: The phases of spiral model 51
Figure 3.3: Incremental model 53
Figure 3.4: Prototyping model process 55
Figure 3.5: Rapid Application Development Phases 57
Figure 3.6: Agile model 58
Figure 3.7: Rapid Application Development Phases 61
Figure 3.8: Work Breakdown Structure diagram 67
Figure 3.9: Project 1 Gantt chart 70
Figure 3.10: Project 2 Gantt chart (part 1) 71
Figure 3.11: Project 2 Gantt chart (part 2) 72
Figure 4.1: Use Case Diagram

Figure 4.2: Sequence diagram of Academic Staff (Manage Appointments) Part 1

Figure 4.3: Sequence diagram of Academic Staff (Manage Appointments) Part 2

Figure 4.4: Sequence diagram of Academic Staff (Manage Appointments) Part 3

Figure 4.5: Sequence diagram of Academic Staff (Manage Appointments) Part 4

Figure 4.6: Sequence diagram of Academic Staff (Manage Appointments) Part 5

Figure 4.7: Sequence diagram of Academic Staff (Manage Availabilities)

Figure 4.8: Sequence diagram of Academic Staff (Manage Google Calendar Synchronization)

Figure 4.9: Sequence diagram of Academic Staff (Manage Setting of Reminder Notice)

Figure 4.10: Sequence diagram of Academic Staff (Manage Appointment Types)

Figure 4.11: Sequence diagram of Academic Staff (Manage Appointment Rules)

Figure 4.12: Sequence diagram of Academic Staff (Manage Intake Form)

Figure 4.13: Sequence diagram of Academic Staff (Generate Reports)

Figure 4.14: Sequence diagram of Academic Staff (Manage Profile)

Figure 4.15: Sequence diagram of Academic Staff (Change Account Password)

Figure 4.16: Sequence diagram of Requester (Manage Appointments) Part 1

Figure 4.17: Sequence diagram of Requester (Manage Appointments) Part 2
Figure 4.18: Sequence diagram of Requester (Manage Appointments) Part 3

Figure 4.19: Sequence diagram of Requester (Manage Appointments) Part 4

Figure 4.20: Sequence diagram of Requester (Manage Appointments) Part 5

Figure 4.21: Sequence diagram of Requester (Register Account)

Figure 4.22: Sequence diagram of Requester (Manage Profile)

Figure 4.23: Sequence diagram of Requester (Change Account Password Account)

Figure 4.24: Sequence diagram of Administrator (Manage Accounts) Part 1

Figure 4.25: Sequence diagram of Administrator (Manage Accounts) Part 2

Figure 4.26: Sequence diagram of Administrator (Manage Accounts) Part 3

Figure 4.27: Logical Entity Relationship Diagram

Figure 4.28: Physical Entity Relationship Diagram

Figure 4.29: UI Design for Login Page

Figure 4.30: UI Design for side navigation bar (Academic Staff)

Figure 4.31: UI Design for Appointment Calendar (Academic Staff)

Figure 4.32: UI Design for Appointment List (Academic Staff)

Figure 4.33: UI Design for Validation of Appointment (Academic Staff)

Figure 4.34: UI Design for Availability Management (Academic Staff)

Figure 4.35: UI Design for Appointment Type Listing (Academic Staff)

Figure 4.36: UI Design for Create New Appointment Type (Academic Staff)
Figure 4.37: UI Design for Update Appointment Type (Academic Staff) 129
Figure 4.38: UI Design for Viewing and Update of Appointment Rules (Academic Staff) 129
Figure 4.39: UI Design for Intake Form Question Listing (Academic Staff) 129
Figure 4.40: UI Design for Create/ Update Form of Intake Questions (Academic Staff) 130
Figure 4.41: UI Design for Google Calendar Synchronization (Academic Staff) 130
Figure 4.42: UI Design for Google Calendar Desynchronization (Academic Staff) 130
Figure 4.43: UI Design for Reminder Notice Management (Academic Staff) 131
Figure 4.44: UI Design for Report Generation (Academic Staff) 131
Figure 4.45: UI Design for viewing of user profile (Academic Staff) 132
Figure 4.46: UI Design for editing of user profile (Academic Staff) 132
Figure 4.47: UI Design for Change Password (Academic Staff) 132
Figure 4.48: UI Design for Staff Directory (Requester) 133
Figure 4.49: UI Design for Booking Form (Requester) 133
Figure 4.50: UI Design for Appointment Listing (Requester) 134
Figure 4.51: UI Design for Appointment Details (Requester) 134
Figure 4.52: UI Design for viewing of user profile (Requester) 134
Figure 4.53: UI Design for editing of user profile (Requester) 135
Figure 4.54: UI Design for Change Password (Requester) 135
Figure 4.55: UI Design for Account Registration (Requester) 135
Figure 4.56: UI Design for Account Creation (Administrator) 136
Figure 4.57: UI Design for Account Status Management (Administrator) 136

Figure 4.58: MVC components of Laravel 137

Figure 5.1: Academic staff’s side bar 139

Figure 5.2: Section of code for verification of requester flag 139

Figure 5.3: Requester registration view 139

Figure 5.4: Staff directory view 140

Figure 5.5: Staff directory with filtered academic staff 140

Figure 5.6: First tab of booking form 141

Figure 5.7: Section of code for obtaining free time slot(s) 141

Figure 5.8: Second tab of booking form 142

Figure 5.9: Third tab of booking form 142

Figure 5.10: Section of code for adding appointment 143

Figure 5.11: Appointment details view 143

Figure 5.12: Side bar 144

Figure 5.13: Section of codes for extracting filtered appointments 144

Figure 5.14: Appointment calendar view 145

Figure 5.15: Appointment list view 145

Figure 5.16: Appointment details view 146

Figure 5.17: Modal for new selection date, start time and end time 147

Figure 5.18: Model for input of cancellation reason 148

Figure 5.19: Section of codes for verifying appointment validation period 148

Figure 5.20: Appointment details view with validation button 149

Figure 5.21: Availabilities view 150

Figure 5.22: Modal for confirming to add new availability 150
Figure 5.23: Modal for confirming the update of availability 151
Figure 5.24: Modal for confirming the deletion of availability 151
Figure 5.25: Section of codes for obtaining latest availabilities 152
Figure 5.26: First tab for first time login guideline 153
Figure 5.27: Second tab for first time login guideline 153
Figure 5.28: Section of codes for update of flag 153
Figure 5.29: Third tab of first time login guideline 154
Figure 5.30: Setting dropdown in side bar 154
Figure 5.31: Synchronization enable page 155
Figure 5.32: Google consent screen 155
Figure 5.33: Synchronization disable page 155
Figure 5.34: Appointment rules enable page 156
Figure 5.35: Section of codes for enabling appointment rules 156
Figure 5.36: Appointment rules update and disable page 156
Figure 5.37: Add form for appointment type 157
Figure 5.38: Update form for appointment type 157
Figure 5.39: Prompt for deletion of appointment type 158
Figure 5.40: Default reminder notices setting view 158
Figure 5.41: Customization of reminder notices setting view 159
Figure 5.42: Modal for guideline in getting appointment data 159
Figure 5.43: Sample of approval notice from system 160
Figure 5.44: Add form for intake question 161
Figure 5.45: Update form for intake question 161
Figure 5.46: Prompt for deletion of intake question 162
Figure 5.47: Reports generation view 162
Figure 5.48: Sample of report based on filtration on booking period 163

Figure 5.49: Profile dropdown in side bar 163

Figure 5.50: Profile view 163

Figure 5.51: Edit profile view 164

Figure 5.52: Change password view 164

Figure 5.53: Requester’s side bar 165

Figure 5.54: Staff Directory with filtered Academic Staff 165

Figure 5.55: First tab of booking form 166

Figure 5.56: Section of codes for getting free time slot(s) 166

Figure 5.57: Second tab of booking form 167

Figure 5.58: Third tab of booking form 167

Figure 5.59: Section of codes for adding appointment 168

Figure 5.60: Appointment list view 168

Figure 5.61: Appointment details view 169

Figure 5.62: Modal for new selection date, start time and end time 169

Figure 5.63: Modal for input of cancellation reason 170

Figure 5.64: Profile dropdown in side bar 171

Figure 5.65: Profile view 171

Figure 5.66: Edit profile view 171

Figure 5.67: Change password view 172

Figure 5.68: Section of codes for change password 172

Figure 5.69: Login box in login page 173

Figure 5.70: Registration box in registration page 173

Figure 5.71 Navigation bar in administrator page 174
Figure 5.72: Account creation view for academic staff 174
Figure 5.73: Account list view of academic staff 175
Figure 5.74: Detailed view of academic staff account 175
Figure 5.75: Sorting of priorities among requesters 176
Figure 5.76: Booking form with higher priority 176
Figure 5.77: Modal for appointment bring forward/postpone 177
Figure 5.78: Section of codes for checking priority and appointment time slots 177
Figure 5.79: Section of codes for sending mails 178
Figure 5.80: Example of automated reminder email 179
Figure 5.81: Section of codes for automated reminder feature 179
Figure 5.82: Example of recap 180
Figure 5.83: Section of codes for daily recap feature 180
Figure 6.1: Section of codes for unit testing 202
Figure 6.2: Result of the unit testing 202
Figure 6.3: Example of test cases in selenium 215
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Abbreviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICT</td>
<td>Information and Communications Technology</td>
</tr>
<tr>
<td>UTAR</td>
<td>Universiti Tunku Abdul Rahman</td>
</tr>
<tr>
<td>HTML</td>
<td>HyperText Markup Language</td>
</tr>
<tr>
<td>PHP</td>
<td>Hypertext Preprocessor</td>
</tr>
<tr>
<td>CSS</td>
<td>Cascading Style Sheets</td>
</tr>
<tr>
<td>PC</td>
<td>Personal Computer</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modeling Language</td>
</tr>
<tr>
<td>SQL</td>
<td>Structured Query Language</td>
</tr>
<tr>
<td>MVC</td>
<td>Model-View-Controller</td>
</tr>
</tbody>
</table>
LIST OF APPENDICES

APPENDIX A: Interview Questions 223
APPENDIX B: Questionnaire Questions 224
APPENDIX C: System Installation Manual 228
CHAPTER 1

INTRODUCTION

1.1 Background

Academic staff such as professors, lecturers and researchers are a significant part of a university. Each member of academic staff has the responsibility to improve the learning environment through teaching, research and service that support the institutional mission. It is a fundamental principle that each academic staff shall be held responsible for competent and effective performance of his/her workload/duties expectations at all times.

Setting up and attending appointments is one of the academic staff duties. It is essential to have the most effective and efficient technique to manage and arrange for the appointment schedule.

In the 21st century, technology has changed the ways in which humans communicate, go about human lives and taking place in teaching and learning. Information and Communications Technology (ICT) has become a major part of our education system. And with the growth of internet, using computers in online appointment scheduling is just one of the major uses of ICT.

1.2 Problem Statement

In UTAR, the academic staff members are still using manual technique for appointment scheduling. The members arrange and set up the academic related appointment from the request of requester such as student, staff or external visitor.

Currently, there is no centralized system for appointment booking to be in place. Requester in current workflow request an appointment through different platforms. For instance, the request can be sent through UTAR email, personal phone call, phone message, WhatsApp messenger, Facebook messenger, face to face and many more. Academic staff require to manually check through every possible platforms to ensure no appointment request will be skipped. A manual reply about acceptance or rejection of appointment will be returned to the requester after staff check with the existing time schedule. And if there is any change on the agreed appointment, the staff will also send the notification for either rescheduling or cancellation of appointment to the parties involved.
An academic staff normally will have numerous appointments to be attended. In order to keep track on their appointment schedule, majority of the staff make their own time schedule on paper and only the minority use paper along with calendar application for recording appointment schedule.

Using manual system to create a fair and suitable appointment schedule to all parties involved is indeed a very difficult task. But difficulties can be solved by replacing the manual technique with an automated and computerized system. By using system, the human involvement will be minimized as the appointment preferences, timing constraints and so on are not needed to be taken into consideration for acceptance, reschedule and cancellation of appointment (Sankaranarayanan and Parchment 2012).

In the traditional, manual way of making appointment, students and staff need to have a long wait for the confirmation of the appointment. In contrast, a computerized system will reduce the long waiting time as the confirmation can be done immediately and automatically. Besides that, the system is able to provide features which eliminate the disadvantages of old manual system such as:

1. Decrease the burden or mistake of academic staff as he/she does not need to arrange appointment one after another.
2. Improve appointment attendance through automated messages or emails as reminder.
3. Provide a centralized platform for appointment booking.
4. Keep track on the appointment activity and store the appointment details for future analysis.
5. Automated rescheduling the appointment to other period when there is a request for appointment which have more priority.
6. Prevent double booking which always been a mistake in manual appointment scheduling.

1.3 Proposed solution

Through the surveys, it is proved that problems emerged when using manual system to manage appointment scheduling. A computerized system named online academic appointment scheduling system is introduced as a proposed solution in order to eliminate the problems encountered by the stakeholders in UTAR. This system is a
great helper to the individuals involved in both appointment booking and appointment acknowledgement as it is full computerized and automated. The human involvement to appointment scheduling will be minimized eventually as appointment preferences, timing constraints and so are not need to be taken into consideration for acceptation, reschedule and cancellation of appointment. (Sankaranarayanan & Parchment, 2012)

1.4 Project Objectives
The objectives of the project are:

1. To conduct literature review of existing online scheduling systems.
2. To conduct interview and survey to stakeholders such as lecturers and students who are involved in appointments in order to determine the important features of online academic scheduling system.
3. To develop a web-based academic appointment scheduling system that can schedule and maintain priority of requests via Google Calendar system.

1.5 Project scope
In order to solve the problems encountered by current manual system, a computerized online academic appointment system is proposed as a solution. The appointment scheduling system is a web-based system which using HTML5, CSS, JavaScript and PHP programming language with suitable web framework and suitable plug-ins. It is only available in PC platform. This web-based system will be integrated with external calendar application, google calendar by using google calendar API.

1.5.1 The users of the system
1. Academic staff (Main user)
   - This user synchronizes his/her google calendar to this system in order to access the automating features where his/her academic appointment schedule is managed and arranged automatically.
   - This user can use this system as requester to make an appointment to another academic staff.
   - This user can make a log for each appointment (duration, content and etc.)
   - This user can generate several types of reports from log.
2. Student/ external visitor/ staff
   • This user uses the system as the platform for appointment booking with academic staff.
3. ICT staff
   • This user acts as a main administrator who take charge of account management.

1.5.2 Scope coverage

As the product of this project, an online academic appointment scheduling system, it is expected:

- Academic staff members can keep their appointment schedule on track.
- Academic staff members can focus on other academic activities.
- Individuals are coordinated in booking their appointments.
- Individuals can book their own appointments and reservations at their own convenience, anywhere and any period of time.
- Academic staff members make no mistake in double booking.
- Significantly reduce the number of “no-shows”.
- Time-saving for appointment scheduling.
- Enable record-keeping and reporting capabilities for appointments.

In order to provide these functionality, the system covers the following 10 modules:

1.5.2.1 Appointment booking

This module can be separate into two different parts, e.g., part for confirmer (academic staff) and part for requester (student/staff/external visitor).

1. Confirmer

   This system is mainly designed to help the academic staff in the appointment scheduling process. The following are the features provided in this module for the staff:
   • Staff can manually add new, change or cancel an appointment through Google Calendar after the synchronization process.
   • Staff can define a range of times for requester to make bookings.
   • Staff can define the prioritization of each requester.
Staff can design their own content of email and message notification for acceptation, rejection, rescheduling and cancellation of appointment.

Staff can design their own content of email and message reminders. (Refer to appointment reminders module for the feature of reminders)

Staff can set the when reminders will be sent before the appointment meeting.

2. Requester

This system allows the requester to make appointment booking through the selecting of a particular staff in staff directory.

A form will be appeared during the process of making appointment booking. The requester requires to complete the form to request for an appointment. The following are the steps to complete the form:

- Select the date/time.
- Provide contact information
  i. Contact Number
  ii. Email
  iii. Other information will be extract from account’s profile.
- Select the appointment types
- Answer specific questions from academic staff.

1.5.2.2 Appointment acknowledgement

The acknowledgement of appointment is mainly based on the prioritization according to group which selected by main user. This system able to acknowledge the appointment in different scenarios:

1. Only one individual requests an appointment for that particular date and time. Acceptance email and message will be sent to the requester.
2. Multiple individuals request appointment for the same particular date and time.
3. Some individuals request the appointment for the time period which already reserved for other requester.
1.5.2.3 Appointment reminders
The main function of this module is to reduce no-shows for every appointment. It provides automatic reminders feature to ensure the both parties (confrimer and requester) always alert for appointment. The automated reminders will be sent in email format.
In this system, the reminders will be sent in three scenarios:
   1. Staff (confrimer) and requester will be notified when appointment are confirmed, modified or cancelled.
   2. Staff (confrimer) and requester will be reminded when the appointment date and time is near.
This system will allows academic staff (confrimer) to tailor his/her reminders:
   1. Change the content of email notifications.
   2. Set how far in advance the reminders are sent.

1.5.2.4 Synchronization
This system provides the syncing feature with Google Calendar in real-time. Therefore, academic staff are allowed to use Google Calendar as a platform to manually change and manage their academic appointment schedule. The staff is only requires to link his/her Google account in the setting of this system. This synchronization helps in prevent double booking of appointment.

1.5.2.5 Search Facility
This system has a search facility which allows the requester to search a particular staff profile by the staff name. If the name of that particular staff match the available names in the database, the system will pop out all the possible staff’s record for requester to choose.

1.5.2.6 Daily recap
This module will aimed to keep academic staff update to date with appointment every day. The recap will be delivered every morning at 6am before the working hours.
1.5.2.7  Appointment log
This module provides the tracking feature which it allows academic staff to keep track on appointment history. This system provides a form for academic staff to fill-in after each appointment is completed. The form will then save as a new log.

The following queries will be in the form:

➢ The attendance of requester: present or absence.
➢ If absence, state the reason.
➢ If present, state the punctuality of requester.

Moreover, there is an addition feature which allows academic staff to view the full appointment history.

1.5.2.8  Profile/ Directory
Every profile/ directory of academic staff in all faculties, departments and institute will be available in this system. In each profile of academic staff, it includes:

1. Basic information
   a. Name
   b. Faculty/ Institute/ Centre/ Division
   c. Department
   d. Designation
   e. Administrative Post
   f. Telephone No
   g. Email Address
   h. Qualification
   i. Area of Expertise
   j. Homepage URL

Other than academic staff, requesters will also be having their profile and all of them are able to make changes of their personal information.

1.5.2.9  Reports generation
This module aim to print the report based on the analysis result from the log. Each academic staff can generate report based on:

1. The list of no-shows appointments on day, week or month basis.
2. The list of completed appointments on day, week or month basis.
3. The list of cancelled appointments on day, week or month basis.
4. The list of appointments filtered by period of times (morning, afternoon, evening).
5. The list of appointments filtered by type on day, week or month basis.

1.5.2.10 Authentication and administration

There will be a main administrator, a member of ICT team in university who take charge of account management. The account of every academic staff member will be created by administrator. Academic staff members are able to change the password once log-in to the system. If the member forget the password of their account, they require to obtain the new password from the administrator.

1.5.3 Out of Scope

Following are the items that are not covered in this project:

1. Compare each academic staff’s schedule to output a suitable period of time for general meeting between staff.
2. Generate all kind of report(s) based on the logs from every academic staff.
3. Request a new password through online.
4. Generate academic time schedule such as timetable for classes as it is only use for managing the appointment booking between academic staff and requester.
5. Use system for other non-academic related appointment scheduling.

1.5.4 Conclusion

In general, the proposed system is able to bring solutions to solve the issues in the current workflow. By using this system:

1. **Academic staff members can keep their appointment schedule on track.**
   Staff members are able to keep track on their appointment schedule at any time and at any place using computers, tablets or phones. They are able to view their schedule through this web-based system or Google Calendar. Through this functionality, issues encountered using pen-and-paper technique can be eliminated.
2. Academic staff members can focus on other academic activities.
   Every appointment booking is managed by this web-based system without any human involvement. The system has automated notification to requesters about the acceptation, rejection, reschedulation and cancellation of appointment through email and message. It directly reduces the workload of staff members therefore they can focus on other academic related activities.

3. Individuals are coordinated in booking their appointments.
   In the system, there is a defined form for each appointment booking. This form will coordinate the requester to fill in the required information for appointment booking.

4. Individuals can book their own appointments and reservations at their own convenience, 24 hours a day.
   Usually, each request of appointment can only be confirmed during working hours. With the automated feature for appointment booking, individuals can make their appointments at any time and get an immediate reply for every request.

5. Academic staff members make no mistake in double booking.
   This system provides flexibility by enables the synchronization with Google Calendar. Although the system requires less human involvement in managing the time schedule, staff can choose to add in/ modify/ delete any appointment through Google Calendar. The system will automatically sync with Google Calendar to get the latest time schedule to prevent double booking is made from the requester.

6. Significantly reduce the number of “no-shows”.
   The system provides automated reminders to every party that involved in the appointment. The default reminders will be sent to staff members and requester one hour before the appointment is hold in order to remind them. The reminders will be in the form of email or message. To provide extra flexibility to the main user, staff members are able to change the time for reminders to be sent and add more reminders.
7. **Time-saving.**

In old manual system, requester requires to have a long wait for the reply from staff member about the appointment request. Multiply with the workload of staff, they are not able to reply on each request immediately and cause time-consuming issue to the requester. With the automated feature, the requester is no longer need to have wait for the confirmation from staff members.

Moreover, this system includes a full appointment schedule of every staff members, requester is now able to know the free period of the staff for appointment, thus the process of requesting time schedule from the staff can be eliminated and time is saved indirectly.

8. **Enable record-keeping and reporting capabilities.**

In current workflow, there is no log for every appointment. Academic staff is not able to keep track on the appointment history such as the contact information of the requester, details of discussion and so. With the data storage features which provided by the system, staff are able to record the information about every appointment. The record in database of the system can be then analysed and generate a few version of report that provided by the system.
CHAPTER 2

LITERATURE REVIEW AND FACT FINDING

2.1 Existing Online Appointment Scheduling System
In this section, three related online appointment scheduling system will be studied, analysed and evaluated on their key features and functionality. Although those systems are non-relate to academic, but the features provided are quite similar and applicable to my proposed web-based system. The appointment scheduling systems that will be discussed in the following are Acuity Scheduling, FlexBooker, and Square.

2.1.1 Acuity Scheduling
In Acuity Scheduling, there are two types of users: administrators (main) and clients.

2.1.1.1 Key Features
1. Appointment booking
   Acuity Scheduling allows clients to make appointment by filling information in selected administrator’s profile. The information requires including name, age, identity card number, gender, along with booking date and time.

2. Automated confirmations, reminders and follow-ups
   Acuity Scheduling will automatically send confirmations, reminders and follow-ups on appointment scheduled to clients via email or text message. Customization of email and text message are allowed.

Figure 2.1: Sample of Confirmations message
3. **Intake form**

Administrators can customize their specific questions to ask clients at the end of scheduling process for necessary information for early preparation or so.

4. **Synchronization with external calendars**

Two-way sync with external calendars is available for Google Calendar, iCloud, Office 365, Outlook.com and Exchange. The appointment scheduled by clients will be automatically integrated in admin’s preferred calendar. Double booking can be prevented.

5. **Integrations**

Acuity scheduling is integrated with different tools such as MailChimp for email marketing service, Freshbooks for cloud-based accounting service and many more.

6. **Online Payments**

Payments can be made through Acuity Scheduling as this software is integrated with Stripe, Paypal and Authorize.net. This allows the clients to pay the bill with credit card directly from the scheduling page. The billing is for appointments such as appointment for haircut, hair treatment, foot massage and many more.

7. **Combine multiple schedules**

Acuity Scheduling’s appointment scheduling software allow admins to manage all their staff’s schedules together without the need to manage multiple calendars. The software consists of filtering feature to filter the schedule by day, week, or employee.
8. Real-time availability
Administrators are able to set their own availability in either based on regular weekly hours or specific days setting. Clients can see the administrator’s real-time calendar availability through this software and self-select the time that works best for them.

![My Calendar](image)

Figure 2.3: Settings of availability in Acuity Scheduling

9. Report generation
Reports can be generated in different areas: appointment report, revenue, admins and so. Appointment report is the main report which will covers the total number of appointments, total duration of appointments and total price of appointments.

10. Responsive layout
Although Acuity Scheduling is a fully web-based software, admins and clients can access it from anywhere on their phone or tablet as this software is completely responsive on any mobile device.

2.1.1.2 Evaluation
Acuity Scheduling is counted as an online scheduling program that is simple to set up to help administrators to manage their schedule easily. As the first-time login administrators, Acuity Scheduling provides a checklist with progress bar to lead administrators through each step to ensure them to fully utilize the system. The
checklist includes setting availability period of time, setting scheduling rules such as prevent cancellation of appointment 24 hours before it take place, designate specific appointment types, setting pricing for each appointment types, and customize client’s scheduling page.

Although the checklist is not one of the key features but it is a good feature which lead/ teach users to learn each function in the application quickly. I would attempt to include all the items in the checklist which mentioned earlier my application except the item which probe the users to set the pricing for each appointment types into as my proposed application is mainly used for academic-related appointments.

One of the best features Acuity Scheduling has is the responsive web-based system. Although Acuity Scheduling does not provide any mobile applications, but their web-based system has been optimize to adapt to different layouts which enable users to get access to their schedule in any devices. This is one of the good solution for me as my proposed solution is mainly a web-based system without providing a mobile application either in IOS or Android version.

Furthermore, intake form is the another feature which will be adapted to my application as that the intake form is a feature which allows users to ask questions to get a more detailed information from the clients. As if the feature is applied to my system, academic staff is able to ask specific questions based on their preferences to prepare for every appointment. Aside with the features mentioned earlier, all basic key features which provided by Acuity Scheduling can be applied to my proposed solution except for the integrations and online payment features. Integration feature is not needed as most of the integration that have been done is for the business field which is not relevant to my academic-related project.

There is one limitation of this application which it does not have a wait list for clients. Clients are not able to receive any notification when a desired appointment time becomes available. While in terms of interface design, Acuity Scheduling has provided an interactive calendar that was simple to use while booking an appointment online but unfortunately there is one drawback which is that each step in booking process, it brings the users to another page rather than allowing them to complete the whole booking process within one view window.
In conclusion, almost all of the basic key features provided by Acuity Scheduling: real-time availability, intake form, synchronization with external calendars, automated confirmations, reminders and follow-ups, report generation, and responsive layout can be implemented to my project except for online payment and integration features.

2.1.2 FlexBooker
In FlexBooker, there are two types of users: administrators (main) and clients.

2.1.2.1 Key Features
1. Appointment booking
   FlexBooker allows clients to make appointment by filling information in selected administrator’s profile. The information requires including name, age, identity card number, gender, booking date and time and answering intake questions.

2. Automated confirmations, reminders and follow-ups
   FlexBooker will automatically send confirmations, reminders and follow-ups on appointment scheduled to both users – administrators and clients via email or text message.
3. **Daily Recap**

FlexBooker will keep admins update with the latest bookings and cancellations that have been made in the last 24 hours, along with a clipboard view of next 7 days of booking. The recap normally will be delivered every morning before 6 AM.

![Figure 2.5: Clipboard view of Daily Recap](image)

4. **Automated Wait List**

Clients can place themselves into a wait list for any full appointments. A wait list is not a booking or a confirmation but is a way to automatically get a reserved for appointment if space opens up for them. If the event is opening up for a wait list party, FlexBooker will send the client a text message which ask the client to turn his/her wait list request into a confirmed booking.

5. **Online Payments**

FlexBooker allows online payments using credit card for appointment bookings.

6. **Client Management**

This feature has the similar function as report generation. It allows administrator to keep track of several critical metrics: the busiest appointment type, the most productive employee, the clients who book the most and many more.

![Great news! Space has opened up for your 'Biceps and Back' Class at 1/31/2014 1:00 PM with ABC Widgets Inc.. Do you want to book? Reply "YES 22" or "NO 22".](image)

![Figure 2.6: Sample of notifications about availability of time slot](image)
7. **Fully Customizable**

Administrators can edit confirmation and reminder emails using HTML editor provided. Moreover, admins can edit their booking form according to their own preferences.

8. **Intake form**

Administrators can customize their specific questions to ask clients at the end of scheduling process for necessary information for early preparation or so.

9. **Calendar Integrations**

FlexBooker enables two-way sync with external calendars for Google Calendar, Apple Calendar, Office 365, and Outlook. Double booking will be prevented as the appointment scheduled by clients will be automatically integrated in administrator’s preferred calendar.

10. **Combine multiple schedules**

FlexBooker’s appointment scheduling software allow admins to manage all their staff’s schedules together without the need to manage multiple calendars. The software consists of filtering feature to filter the schedule by day, week, or employee.

11. **Responsive layout**

FlexBooker is a fully web-based software but it is able to detect the type of device administrator or client used to visit and then adapt to the layout of that specific device.

2.1.2.2 **Evaluation**

FlexBooker has provide a variety of important features and tools which simplify the whole scheduling process. One of the best features is the wait lists. This feature allows clients to receive notification when the desired time which indicated earlier in the online schedule becomes available. It will ask the client to turn his/her wait list request into a confirmed booking. This feature will be the best feature for my application too as this would benefits the requesters. For instance, there are many students under
supervised by an academic advisor, they require to meet with advisor near the end of the semester. It is possible that students will book for the same time slot to meet with their advisor, therefore, a wait list is important for the students as if there is a student suddenly reschedule the time slot for appointment, then the student in the wait list will be notified.

The other best feature is the daily recap that will be sent through email. This feature is work to update the users at every 6AM in the morning with their latest bookings and cancellations that have been made in the last 24 hours, along with a nice clipboard view of next 7 days of booking. I would like to adapt this feature in my application with the extra feature on sending the recap in text message. Furthermore, the basic key features will be adapted and implemented in my application such as intake form and many more except the feature related to payment and the feature related to time zone conversion.

Unfortunately, FlexBooker has one limitation in which that it only integrate with iCal compared to other online appointment scheduling programs which integrate with a variety of other calendars and business programs. This limitation may cause problem to those who are using Google Calendar, Outlook Calendar and so as their calendar are not synchronized.

In conclusion, basic key features such as appointment booking, automated confirmations, automated reminders and follow-ups, synchronization with external calendar, daily recap and intake form can be implemented to my system. Other than those, for my opinion is out of scope for my project.

2.1.3  Square
In Square, there are two types of users: administrators (main) and clients.

2.1.3.1  Key Features
1.  Appointment booking
    Square allows clients to make appointment by filling information in selected administrator’s profile. The information requires including name, age, identity card number, gender, booking date and time and answering intake questions.
2. **Automated confirmations, reminders and follow-ups**
   Square’s appointment scheduling software will automatically send confirmations, reminders and follow-ups on appointment scheduled and to clients via email or text message.

3. **Synchronization with external calendar**
   Square’s appointment scheduling software enables two-way synchronization with Google Calendar. The availability of admins is then always up to date.

4. **Online Payments**
   This software allows online payments using credit card for appointment bookings.

5. **Intake form**
   Administrators can customize their specific questions to ask clients at the end of scheduling process for necessary information for early preparation or so.

6. **Custom cancellation policy**
   Admins are allowed to request prepayment or a credit card from clients. Timeframe can be set for late cancellations and apply a fee for any missed appointments or last-minute cancellations.

7. **Book multiple services at one**
   Clients can select several services to schedule all at once. Square’s appointment scheduling software will block off the time on admin’s calendar so that his/her clients can get everything done in one sitting. For example
8. **Sync schedule across Apple devices**

Mobile application is available for iPhone and iPad app.

9. **Combine multiple schedules**

Square’s appointment scheduling software allow admins to manage all their staff’s schedules together without the need to manage multiple calendars. The software consists of filtering feature to filter the schedule by day, week, or employee.

2.1.3.2 **Evaluation**

Comparing Square to FlexBooker and Acuity Scheduling, Square does not have as many functionality and features available to clients. Square does not have the services of wait lists, daily recaps and intake form. However, the administrative portal of
Square indeed has the better interface design compared to the others. It is because Square has more cleanly and clear interface whereas other portal has more complex and pack design on interface. Square also provides instructions to lead clients in the steps of setting up available time slot, setting up appointment types such as discussion meeting and many more.

One of the good feature is that book multiple services in one sitting. Clients are given the option of booking multiple services as a part of a single appointment. For instance, a client can book an appointment for multiple services such as long hair cut along with service of short haircut. But this feature has less meaning for academic-related application.

![Figure 2.9: Book multiple services view](image)

Although Square has multiple great features to simplify appointment scheduling but the focus is more on businesses. Most of the features are not able to be applied as they are not suitable for academic-related application such as key features of custom cancellation policy and online payment which involves payment. Furthermore, academic staff in UTAR should arrange their appointment schedule on their own perspective, therefore, the key feature of combine multiple schedules in one view is not needed as well.

In conclusion, Square’s appointment scheduling software can only provide basic appointment scheduling features as references to my proposed application such as appointment booking, appointment reminders, synchronization of external calendars and custom cancellation policy.
2.1.4 Comparison between the scheduling system

The Table 2.1 compares the features provided between the scheduling systems: Acuity Scheduling, FlexBooker and Square.

Table 2.1: Comparison between online scheduling systems

<table>
<thead>
<tr>
<th></th>
<th>Acuity Scheduling</th>
<th>FlexBooker</th>
<th>Square</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appointment Booking</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>Automated confirmation,</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>reminders and follow-ups</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Daily Recap</td>
<td>Not available</td>
<td>Available</td>
<td>Not available</td>
</tr>
<tr>
<td>Automated wait list</td>
<td>Not available</td>
<td>Available</td>
<td>Not available</td>
</tr>
<tr>
<td>Checklist/ guidelines for</td>
<td>Available</td>
<td>Not Available</td>
<td>Available</td>
</tr>
<tr>
<td>first time user</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fully customization</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>Synchronization with external</td>
<td>Available</td>
<td>Available</td>
<td>Only Google</td>
</tr>
<tr>
<td>calendars</td>
<td></td>
<td></td>
<td>Calendar</td>
</tr>
<tr>
<td>Intake form</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>Integration with external tools</td>
<td>Available</td>
<td>Not available</td>
<td>Not available</td>
</tr>
<tr>
<td>Real-time availability</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>Online Payments</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>Reports generation</td>
<td>Available</td>
<td>Available</td>
<td>Not available</td>
</tr>
<tr>
<td>Manage multiple schedules</td>
<td>Available</td>
<td>Available</td>
<td>Available</td>
</tr>
<tr>
<td>(a team schedule)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Responsive web layout</td>
<td>Available</td>
<td>Available</td>
<td>Not available</td>
</tr>
<tr>
<td>Mobile platform</td>
<td>Not available</td>
<td>Not available</td>
<td>Only IOS mobile</td>
</tr>
</tbody>
</table>

2.1.5 Conclusion

In conclusion, the existing online appointment scheduling systems have provide a numerous amount of features that are suitable for my project. The features that I found to be useful for my project:
1. **Appointment booking**
   Requesters which are visitors, students and staff can use the system to book appointment for discussion, meeting and any other academic-related appointments. Real-time availability can be view through the process of appointment booking.

2. **Automated confirmation, reminders and follow-ups**
   Reduce the human involvement in confirmation and follow ups and even remind both parties about the appointments to reduce no-shows rate.

3. **Daily Recap**
   Academic staff can be up to date with all the appointments that have been attended and those that have not.

4. **Intake form**
   Academic staff can use intake form to obtain the specific information to prepare for any appointments.

5. **Checklist/ guidelines for first time user**
   Checklist or guidelines provide the first time login academic staff for easy setup of the system.

6. **Synchronization with external calendar**
   Synchronization with external calendar allows academic staff to have the latest appointment schedule and prevent double booking.

7. **Reports generation**
   Reports can be generated by academic staff to keep track on appointment history. For instance, report can be generated on who has missed/ no-shows for appointments more than 3 times.

Other than features, those systems have provided some interface design concept for my project. For instance, Acuity Scheduling has provided an idea that my
proposed system should allow requester to complete the booking process within one instead of multiple view window.

2.2 Research Methodologies
Research is a process to gain a great understanding on a phenomenon where we are interested or concerned through the steps of collecting, analysing and interpreting information or data collected (Creswell 2008). A research process is systematic in terms of managing the data or information, defining the objective and communicating the findings which occur within an established frameworks and also in accordance with existing guidelines.

Research has several characteristics. Williams (Williams 2007) asserts that one of the characteristic is that research is originates with at least one problem or one question about one phenomenon of interest. A research question is necessary to start up a research as it is a vital for a research project, study, or review of literature which helps researchers to focus on their studies, reviews, and opinions. Moreover, it could help them to select the appropriate approach for their research on the phenomenon of interest. My research questions can be found in the appendix of this project.

There are two common approaches available for conducting research: quantitative and qualitative research. Each approach is used to respond to research questions which requiring different types of data. For instance, quantitative research approach is designed for responding numerical data type while qualitative research approach is designed for responding textural data type.

2.2.1 Quantitative Research Approach
Quantitative Research - Definition
Quantitative research is a method which emphasizes on objective, measurements and analysis of data. There are 3 types of analytical method: statistical, mathematic or numerical which used to analyse the data collected through different data collection methods: polls, surveys, and questionnaires, or by manipulating pre-existing statistical data using computational techniques. Quantitative research approach is focusing on gathering numerical data and interpreting it to explain a particular phenomenon. While, Creswell (Creswell 2008), the author of Education Research explained quantitative research as a type of educational research in which allows researcher to decide on what to be studied; an educational research which allows researcher to ask specific
questions; an educational research which collects quantifiable or measurable data from participants; an educational research which analyse data using statistical analytical method; and an educational research which conducts the inquiry in a biased and subjective way.

**Quantitative Research - Methodology**

In quantitative research, there are three broad classification (Bhawna and Gobind 2015): descriptive, experimental and causal-comparative.

1. **Descriptive**

   Descriptive is a research approach that describes and analyses the current status of an identified variable or phenomenon. This type of research identify and classify of attributes of a particular phenomenon based on observation, or exploration of correlation between two or more phenomena.

2. **Experimental**

   Experimental research approach is an approach which implements to investigate the treatment of an intervention into the study group and also measures the outcomes of the investigation on the treatment. Bhawna and Gobind (Bhawna and Gobind 2015) asserts that there are three types of exploratory approaches: quasi-experimental, pre-experimental, and true experimental.

   a. **Quasi-experimental**

      Quasi-experimental is a design which involves non-random selection of study participants.

   b. **Pre-experimental**

      Pre-experimental is design which consists of an independent variable that does not vary or a control group that is not randomly chosen.

   c. **True experimental**

      True experimental is a design which provides a higher degree of control in the experiment and produces a high degree of validity. The designs result in a systemic approach to quantitative data collection involving mathematical models in the analysis.
3. **Causal-Comparative**

Causal Comparative research is used to determine and to examine on how the dependent variables reflect the independent. The research is also involves cause and effect relationships between the variables. Generally, there are three types of causal comparative research: exploration of causes, exploration of effects, and exploration of consequences. As for the design of this research, it provides solution to determine and test the interaction between dependent variables and independent variables.

**Quantitative Research - Data Collection**

There are a variety of techniques can be used as the data collection methods for quantitative research approach. However, all of the techniques are tends to be numerical and structured. The measurable data can be gathered through common data collection such as questionnaires and structured interviews.

1. **Questionnaires**

Surveys are typically selected as the data collection method when information is required to be gathered from a large number of groups of people or when answers are needed to clearly defined set of questions (Frechtling 2002). In general, questionnaires are one of the instruments to conduct surveys. Questionnaires are cheaper and effective in measuring the attitudes, preferences, thoughts, behaviours and intentions of large number of people when compared to other data collection methods for quantitative research. Questionnaires can be divided into two types: paper-and-pencil questionnaires and web-based questionnaires. In questionnaires, there are two distinctive questions: open-ended and closed-ended questions. Questions which are open-ended allow respondent to express their answer in their own words while closed questions allow only fix answers which provided in advanced by the researcher.

As questionnaires are one of the surveys’ instruments, the following are the strength and limitations:

**Strengths**

- Time-saving as large amount of data can be gathered within a short period.
- A wide range of topics can be covered in questions
• Cost-saving
• Results can be easily analysed and quantified either by researchers or through software.

Limitation
• Data collected may only provide a general view which lack of adequate information.
• Data is lack of validity as respondent may not be honest.

2. Structured interviews
Structured interview is also known as a formal interview. The questions in structured interviews are usually structured, closed-ended questions and they are being asked in a set or standardized order (Frechtling 2002). Normally, interviewer/researcher will not probe beyond the answers received.

Strengths
• A fixed set of closed questions allows easy replication of structured interviews and easy quantification of data collected.
• Multiple interviews can take place within a short period of time which means that a large sample can be obtained resulting in the findings being representative and having the ability to be generalized to a large population.

Limitations
• Not flexible as interview schedule must be followed. New questions are not allowed to be asked during the interview section.
• Closed questions are not able to collect detailed answers.
Quantitative Research - Strengths and Limitations

Table 2.2: Strengths and limitations of quantitative research

<table>
<thead>
<tr>
<th>Strengths</th>
<th>Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Findings can be generalised if selection process is well-designed and sample is representative of study population</td>
<td>Related secondary data is sometimes not available or accessing available data is difficult/impossible</td>
</tr>
<tr>
<td>Data can be very consistent, precise and reliable</td>
<td>Data may not be robust enough to explain complex issues</td>
</tr>
<tr>
<td>Relatively easy to analyse</td>
<td>Difficult to understand context of a phenomenon</td>
</tr>
</tbody>
</table>

2.2.2 Qualitative Research Approach

Qualitative Research - Definition

Qualitative research is a type of research which used to obtain a deep understanding of underlying reasons, thoughts, opinions and motivations of a phenomenon. Furthermore, this research provides better pictures or insights into the setting of a problem or it could help in generating ideas and hypothesis for potential quantitative research. Based on Creswell (Creswell 2008), qualitative research is a type of educational research in which the researcher is deeply depends on the opinions or views of participants; an education research that requires to ask broad, general questions; an education research that requires to collect data consisting largely of words from participants; an education research that can describe and analyse the data collected for themes; and an education that can conduct the inquiry in a subjective, biased way.

Qualitative Research - Methodology

Qualitative research is usually conducted within a poststructuralist paradigm. Williams (Williams 2007) asserts that there are five common methods that can be used to conduct qualitative research: ethnography, phenomenological, case study, grounded theory, and content analysis.
1. **Ethnography**
   Ethnography emphasizes on culture. Participant observation is the most common approach of ethnographic as a part of field research. Creswell (Creswell 2008) defines that ethnography is an approach which allows researcher to study on a group of people that shares a common culture by collecting primarily on observational data. This approach is focusing on everyday behaviours in order to understand the changes in culture over time by figure out the beliefs, norms, social structures and other related factors.

2. **Phenomenology**
   Phenomenology is a study that emphasizes on subjective experiences from participants and interpretations of the world to others. According to Creswell (Creswell 2008), the purpose of phenomenology approach is to understand the underlying meaning of participant’s experience and interpret about the both outward appearance and inward consciousness of experience based on image, meaning and memory.

3. **Case Study**
   Case study emphasizes on a person, activity, process or event. There are two different types of case study: a single case or cases that consists of a defined place and time frame (Williams 2007). Case study can also be defined as a study to gain a better understanding or learn more on a poorly understood situation. As for the data collection of a case study, it is said to be extensive as it requires to draw from multiple sources such as direct or participant observations, interviews, physical artefacts and so which is time-consuming as researcher must spend time on-site to interact with the target people for research purpose.

4. **Grounded Theory Study**
   Grounded theory research is an approach used to develop a theory about phenomenon of interest based on the views of participants. The theory in this study is developed from the data collected in the field such as data collected from interview with participants, data collected from observation while on-site and many more rather than the data collected from literature review or research literature. Furthermore, due to the ability of this method which able to determine
and examine people’s actions and interactions, this study has been used primarily in the sociology discipline.

5. Content Analysis

Content analysis is a systematic method that examine and explore the patterns, themes, verbal, visual or biases. This method is designed to identify the special characteristics from the content in the human communication by reviewing books, newspapers, films and other related forms.

Qualitative Research - Data Collection

Data collection approaches for qualitative research usually involves direct interaction with individuals in a group setting or with individuals on one to one basis. These data collection approaches are said to be time-consuming and more expensive as the data or information collected is able to provide a deeper insight into the phenomenon of interest. The common data collection methods are in-depth interview, focus groups, and participant observation (Remler and Van Ryzin 2011).

1. In-depth interview

In-depth interview is usually conducted on one to one basis and is used to probe or obtain qualitative data. The questions for this type of interview are usually in the form of open-ended questions. There are two basic approaches for in-depth interview: informal conversational and guided interviews.

i. Informal conversational interview

Interview is start with normal conversation. In normal work flow, the questions will be asked from immediate context.

ii. Guided interview

A basic checklist will be prepared in advanced before the interview. The checklist is prepared to make sure that all relevant topics are covered. However, the interviewer is free to probe or ask questions which deemed interesting to him/her. He/She is not necessary to follow the checklist or guideline.
Strengths
- Flexible as questions can be changed and adjusted depending on the respondents’ answer.
- More valid and qualitative data can be obtained from respondents as they are allowed to talk in depth and respond in their own words.
- Provides higher validity as interviewer is able to probe and ask additional questions for deeper understanding and clarification.

Limitations
- Time-consuming to conduct an in-depth interview as one can last for a long time to collect qualitative data.
- Time consuming for data analysis

2. Focus group
Focus group is also known as a group interview. This data collection approach can be used when a collective discussion is needed to research to gain understanding on behaviour or opinions, or when there are limited resources in time, finances and manpower (Frechtling 2002).

Strengths
- Time-saving compare to individual interviews
- Qualitative data can be generated through the open-ended questions. This allows respondents to talk in depth by respond with their own words. Discussion is also allowed in this method.

Limitations
- Disagreement and irrelevant on discussion may occurred which distract from the main focus.
- Hard to control and manage

3. Participant Observation
Participant observation is a type of data collection which adapt to ethnographic approach (Remler and Van Ryzin 2011). Participant observation is accomplished through observation and participating. It is usually take place in community
settings to study and get a deeper insight into participant lives. There are two types of participant observations: overt or covert. Overt observations refer to the researcher being open about their intentions in the field and ensuring all members of the social group are aware of what is happening whereas covert observations involve the researcher keeping the reason for their presence as secret.

Strengths
- Provides a deep insight into participants’ contexts, relationships and behaviour.

Limitations
- Observations lack of structure.
- Only a small number can be observed at one time.

**Qualitative Research - Strengths and Limitations**

Table 2.3: Strengths and limitations of qualitative research

<table>
<thead>
<tr>
<th>Strengths</th>
<th>Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provide more detailed information to explain complex problems and issues.</td>
<td>More difficult to analyse; don’t fit neatly in standard categories.</td>
</tr>
<tr>
<td>Multiple methods for gathering data on sensitive subjects.</td>
<td>Data collection is time-consuming.</td>
</tr>
<tr>
<td>Complement and refine quantitative data.</td>
<td>Findings usually cannot be generalised to the study population or community.</td>
</tr>
<tr>
<td>Data collection is cost efficient.</td>
<td></td>
</tr>
</tbody>
</table>
### 2.2.3 Comparison between quantitative and qualitative research

In brief, quantitative research is based on numerical data while qualitative research is based on descriptive data.

The table below shows a more detailed comparison between quantitative and qualitative research:

Table 2.4: Comparison between quantitative and qualitative research (Castellan 2010)

<table>
<thead>
<tr>
<th></th>
<th>Quantitative</th>
<th>Qualitative</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Purpose</strong></td>
<td>Examine phenomena through focused collection of structured, numerical data.</td>
<td>Gain a deep insight and understanding of a phenomena of interest through intensive collection of narrative data.</td>
</tr>
<tr>
<td><strong>Methods of reasoning</strong></td>
<td>Deductive</td>
<td>Inductive</td>
</tr>
<tr>
<td><strong>Approach to Inquiry</strong></td>
<td>Subjective</td>
<td>Objective</td>
</tr>
<tr>
<td><strong>Sampling</strong></td>
<td>Usually a small number of non-representative cases. Respondents selected to fulfil a given quota.</td>
<td>Usually a small number of non-representative cases. Respondents selected to fulfil a given quota.</td>
</tr>
<tr>
<td><strong>Measurement</strong></td>
<td>Standardized, numerical</td>
<td>Non-standardized, narrative (written word)</td>
</tr>
<tr>
<td><strong>Generalization</strong></td>
<td>Generalizable</td>
<td>Not generalizable</td>
</tr>
<tr>
<td><strong>Flexibility</strong></td>
<td>Inflexible as it is structured</td>
<td>Flexible</td>
</tr>
<tr>
<td><strong>Methodology</strong></td>
<td>Descriptive, Experimental and Causal-comparative.</td>
<td>Ethnography, Phenomenology, Grounded Theory, Case Study and Content Analysis</td>
</tr>
<tr>
<td><strong>Data Collection Strategies</strong></td>
<td>Structured interviews and surveys (questionnaires)</td>
<td>In-depth interview, focus group, and participant observation.</td>
</tr>
</tbody>
</table>
### Data Analysis

<table>
<thead>
<tr>
<th></th>
<th>Non-statistical</th>
<th>Statistical data is usually in the form of tabulations (tabs). Findings are conclusive and usually descriptive in nature.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Outcome</strong></td>
<td>Can be used to recommend a final course of action.</td>
<td>Findings are not conclusive and cannot be used to make generalizations about the population of interest. However, it has developed an initial understanding for further decision making.</td>
</tr>
</tbody>
</table>

#### 2.2.4 Conclusion

Based on the review on research methodologies along with their data collection methods, one data collection in both research methods: qualitative and quantitative will be selected to collect information for my project.

The first data collection method that will be implemented is in-depth interview. As mentioned in previous section, there are two types of interviews: structured and in-depth interviews. Structured interview is being eliminated because it requires an interviewer to ask structured, close-ended questions to participant in standardised order. It is impossible for me to identify all possible questions with static possible answers. Moreover, this method is unable to obtain result that is valuable as participant can only answer by selecting the possible answers that I have provided. Therefore, in-depth interview is selected. There are 2 main reasons for this method is being used:

1. I am free to probe or ask any questions to gain further understanding on topic that I am preferred on and relate to my project without the need to follow interview schedule.
2. Questions are all in open-ended which allows participant to respond with their own words, thus more valuable data can be collected.

The second data collection method that will be carried out is surveys by using questionnaires. Questionnaires data collection method is chosen because it can collect
large amount of data from a large number of people in a very short period of time even though it affect the validity and reliability of data as many people do not answer the questions honestly. In my project, questionnaires are used to clarify the information gathered from interviews and to find out possible new information from other possible stakeholders.

2.2.4.1 Application of in-depth interview
In the beginning of the project, before conducting interviews, several tasks have been done: identify possible stakeholders, design questions and checklist, and contact with stakeholders to arrange appointments for interviews. A total of 6 in-depth interviews have been conducted to the stakeholders: 3 interviews to academic staff and 3 interviews to students in UTAR.

In the interviews, the questions that being probe are mostly in open-ended questions instead of closed-ended questions. The main reason for why closed-ended questions are not favourable because the answer choices are given to respondent. The respondent can only answer using what we have given. While, open-ended questions are the questions that allow the respondent to answer based on their own words and thoughts. The questions that have been asked to academic staff and students can be referred to appendix A in this project.

After collecting the data and the data is then grouped as possible answer for closed-ended questions in the next surveys.

2.2.4.2 Application of questionnaires
Surveys are implemented after data collected and analysed from interviews with stakeholders. In this project, the surveys is done by distributing questionnaires to a large number of stakeholders. A total number of 40 questionnaires has been sent through email to academic staff and a total number of 40 questionnaires has been sent through different platforms to students in UTAR.

In the questionnaires, different types of questions are included in the questionnaires:

1. Dichotomous Questions
   A simple question which consists only two possible answers for the question.
   For instance, yes or no.
2. Likert Scaled Questions
   A question which used to rate or measure how the respondent feels towards certain issue. For instance, a question that ask the level of agreement for an automated system is better than manual system.

3. Multiple choice questions
   A closed-ended question which allows respondent to select only one or multiple answers. It can be open-ended which allows respondents to express their thought as they may have different answer for the question.

The questionnaires that are used in this project can be viewed at appendix B. Fact finding is done after questionnaires are collected and the answers are analysed.

2.2.5 Data Collection and Analysis
2.2.5.1 In-depth interview
   In this section, the information collected from the interviews will be discussed, summarised and further analysed. Interview questions can be referred in appendix A.

Interviews with academic staff
   There are 3 in-depth interviews that have been conducted with academic staff for this particular project starting from week 2 to week 3 of the semester.

   From the interviews, it shows that majority of academic staff are still using pen-and-paper technique on appointment scheduling instead of using computerized system such as google calendar to manage on their appointment schedule. Interviewees who prefers pen-and-paper technique claimed that managing appointment schedule using that technique is easier than using computerised system as pen-and-paper does not requires internet, login and a series of confirmation steps. However, scheduling appointments using pen-and-paper technique brought a lot of problems: missing of the paper which appointments are scheduled, the paper or note are not around when there is an appointment request, the time schedule may be messy and hard to read, and many other demerits.

   Appointment is scheduled based on the request from either students, staff or external visitors. In UTAR, the request can be made from different platforms: email, phone call, phone text message, WhatsApp and even through face to face. Interviewees
claimed that there is no much obvious problem by having multiple platforms but having problems in the process of requesting appointment. Most of the problems are caused by students. The problems are as below:

- Students sent request at very last minute.
- Students do not provide feedback as if academic staff reschedule or cancel or reject the appointment.
- Students do not state the time slot they want to have appointments and meet up academic staff at any period of time that they want.

Further questioning is done on the problems that will be faced during the whole appointment process. More problems are found and summarised as below:

- Inefficient using manual technique to manage appointments: time-consuming, late acknowledgement and mistakes in appointment arrangement.
- No-shows by requesters without any reasons.

Next, academic staff do prefer to have online computerized system which is automated for appointment scheduling. Academic staff prefer that the computerized system is able to automatically acknowledge the appointment request from requesters, along with automated notification about the appointment details sent in either email, text messages or both. Although they prefer to have automated feature, however they do want the system to provide flexibility which the system can be selected for manual acknowledgement of appointment by academic staff themselves.

Furthermore, they also prefer to have appointment reminders which will reminds them and the requester about the appointment before it is held. I do agreed on these automated notification and reminder features as it could really help on notify and keep us alert on appointments and prevent no-shows to an appointment. Next, 2 out of 3 of the academic staff emphasized on the tracking of appointment history. The tracking of appointment history can be very important as they can keep track on the requester who always missing for his/ her appointments, the discussion details in appointment for reference in future and many more.
Last but not least, academic staff who are using calendar system, although in minority, they prefer that the system will provide synchronization with the calendar system that they are currently using to access to their personal calendar.

**Interviews with students**

There are 3 in-depth interviews that have been conducted with students (requesters) for this project starting from week 2 to week 3 of the semester.

From interviews, students are found to use multiple platforms to make appointment: email, text messages, WhatsApp and face to face. As requesters, there are also several issues faced for appointment process: before, during and after.

Before:

- Students are unable to see the appointment schedule of academic staff. Therefore, they are unable to determine what time does academic staff is free for appointment.
- Students claimed that academic staff does not reply on their appointment request.

During:

- Academic staff are no-shows for the appointment or students forgot to attend the appointment.
- Academic staff having meeting with other students/staff/visitors which cause the delay or the cancellation of appointment.

Students do prefer an online appointment scheduling system. The main reason is that it will be easy as it can get notification and reminder about the appointment. Students would like to be able to view the availability of academic staff as it ease the students to arrange their time before requesting the appointment.

**2.2.5.2 Questionnaires**

**Surveys on academic staff**

There are a total of 40 questionnaires sent to academic staff and 23 of them have provided feedback on the questionnaires. Questions for questionnaire can be referred in appendix B. The feedback are summarised and analysed as below:
1. The most widely used appointment scheduling technique

![Pie chart showing appointment scheduling technique used by academic staff.](image)

Figure 2.10: Appointment scheduling technique used by academic staff

Based on the pie charts above, it shows majority of academic staff are using pen-and-paper technique for appointment scheduling. In my view, pen-and-paper technique is widely used as it is simpler compared to computerised technique. Pen-and-paper technique does not require complex procedures to manage appointments such as login, input appointment details, and many more.

2. The main problem encountered using pen-and-paper technique

![Bar chart showing problems encountered using pen-and-paper technique.](image)

Figure 2.11: Problems encountered using pen-and-paper technique

According to the bar chart above, it shows that the main problem is that most of the academic staff does not have appointment schedule on their side at anytime and anywhere. In my view, this issue became the main problem as academic staff is not
able to know their upcoming appointments and keep themselves up to date. Moreover, they are not able to record new appointment into the schedule when it is not around.

3. The main problem encountered during appointment process

<table>
<thead>
<tr>
<th>Common problems faced during appointment process</th>
<th>Level of Influence</th>
<th>Weightage (Number of students * Level of Influence)</th>
<th>Order of Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>No-shows for appointments</td>
<td>Most (5)</td>
<td>11</td>
<td>97</td>
</tr>
<tr>
<td></td>
<td>(4)</td>
<td>8</td>
<td>74</td>
</tr>
<tr>
<td></td>
<td>(3)</td>
<td>2</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>(2)</td>
<td>0</td>
<td>65</td>
</tr>
<tr>
<td></td>
<td>Least (1)</td>
<td>0</td>
<td>61</td>
</tr>
<tr>
<td>Unable to acknowledge appointments on time</td>
<td></td>
<td>2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>10</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>No feedback from requesters about acknowledgement of request</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>9</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Mistake occurs in arranging appointment (double booking)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Requesters request appointment at last minute</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

Based on the result shown in the table above, the problem of no-shows is identified as the most critical problem encountered by the academic staff. In my perspective, no-shows is indeed an irresponsible action which cause time-consuming to academic staff which spend his/her time to wait for the requester. Furthermore, the absent requester has occupied the appointment time slot which may be used by other requester for any meaningful discussion.
4. The most important feature for online appointment scheduling system

Table 2.6: Recommended features for appointment scheduling project

<table>
<thead>
<tr>
<th>Features</th>
<th>Level of Influence (Importance)</th>
<th>Weightage (Number of students * Level of Influence)</th>
<th>Order of Importance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Most (5)</td>
<td>(4)</td>
<td>(3)</td>
</tr>
<tr>
<td>Automated notification/reminders</td>
<td>8</td>
<td>15</td>
<td>0</td>
</tr>
<tr>
<td>Tracking on appointment history</td>
<td>4</td>
<td>6</td>
<td>10</td>
</tr>
<tr>
<td>Automated acknowledgement of appointment requests</td>
<td>2</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>Automated prioritization on requester types</td>
<td>0</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>Synchronization with external calendar</td>
<td>0</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>Mobile platform provided</td>
<td>0</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

5. The

6. Table 2.6 shows that automated notification and reminders is the most important feature that must be included into the system. In my point of view, the main reason is that it is used to reduce and eventually prevent the no-shows of both parties: academic staff and requesters. Moreover, it prevents students to have chance to make any excuses for any no-shows.

**Surveys on students**
There are a total of 40 questionnaires sent to students and all of them have provided feedback on the questionnaires. The feedback are summarised and analysed as below:

1. The most used appointment booking platform

![Most used appointment booking platform](image)

**Figure 2.12: Platforms used for appointment booking**

Based on the pie chart above, it shows that email method is the frequent method that is used for appointment booking. In the perspective of a student, email method is preferred as it is more formal and it is easier to get reply from academic staff.

2. The main problem encountered during appointment process

Based on the table shown below, the surveys shows that unable to view the appointment schedule of academic staff is the main problem facing during appointment process. To explain on why most of the students rate highly on this problem is that students require to know the availability time of the academic staff in order to pick the time slot which is more favourable to them. This problem is also prolong the acknowledgement of an appointment request as additional step is required to reaffirm the availability between both parties.

<table>
<thead>
<tr>
<th>Common problems faced during appointment process</th>
<th>Level of Influence</th>
<th>Weightage (Number of students * Level of Influence)</th>
<th>Order of Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Face to face 31%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Email 46%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Text messages 5%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WhatsApp 18%</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.7: Common problems encountered during appointment process
3. The most important feature for online appointment scheduling system

Table 2.8: Recommended features for appointment scheduling project

<table>
<thead>
<tr>
<th>Features</th>
<th>Level of Influence (Importance)</th>
<th>Weightage (Number of students * Level of Influence)</th>
<th>Order of Importance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Most (5)</td>
<td>(4)</td>
<td>(3)</td>
</tr>
<tr>
<td>Automated notification/reminders</td>
<td>21</td>
<td>15</td>
<td>4</td>
</tr>
<tr>
<td>View on appointment schedule of academic staff</td>
<td>15</td>
<td>18</td>
<td>7</td>
</tr>
<tr>
<td>Instant acknowledgement</td>
<td>16</td>
<td>15</td>
<td>9</td>
</tr>
</tbody>
</table>

Lastly, all of the features above are vital to students as all of them have rate the features with a score of 3 and above. Automated notification and reminders feature
is the most vital as it is able to reduce no-shows especially for the students themselves by notify and remind them through email and text messages.

In conclusion, based on the feedback from respondents in interviews and surveys, six problems as below has been found in the current workflow of appointment booking.

I. No dedicated appointment scheduling system available for academic staff

In UTAR, there is no existing academic appointment scheduling system that can be accessed by the academic staff at any place and any time. In the current workflow, most of the academic staff use old pen-and-paper technique to manage their own academic related timetable manually on paper. Since there is no dedicated system provided for managing the appointment schedule, it causes uneasiness to the staff:

1. Schedule made on paper does not have backup. If the paper is missing, an academic staff may lost his/her track on the schedule.
2. An academic staff may not have the paper onside all the time. Sometimes, an appointment for meeting or consultation is made when the paper is not around, therefore he/she may forget to add in that particular appointment into the schedule.

II. Lack of centralized coordination in appointment booking

Due to the lack of a system to coordinate the whole process of appointment booking such as booking, acknowledgement of booking, recording, feedback and etc., therefore multiples problems and misunderstanding were occurred between academic staff and students.

The following are the problems identified through interviews and surveys to a few academic staff and students on the current workflow of appointment booking:

Problems based on academic staff’s perspectives

1. There are no automatic reminder notification for the appointment. Therefore, academic staff or student may forget about the appointment if he/she does not check on their time schedule or he/she does not make a record for the appointment date and time.
2. In some cases, academic staff do informed students about their appointments have been delayed due to some emergency meetings or classes but students do not check on their emails frequently. Besides, there is no notification about the new email which sent by the staff.

3. There are cases that there are some opinionated students who thought their requested appointment have been agreed before the particular academic staff reply on his/her email/message. And if the staff are not around, the students will be eventually wasting their time waiting for unconfirmed appointment.

Problems based on students’ perspectives

1. Some students do not know how to make appointment with academic staff.

2. Some students thought that writing an email for appointment is quite annoying as they are confused about what content need to be written.

3. There is no notification system that can notify student whether his/her requested appointment or consultation has been approved or rejected. There are some cases where the students forget to check for their emails and eventually no-shows on the appointments that they have made.

4. There is no reminder notification that can remind student about the appointment or consultation that he/she has made. There are cases which students often forget about their appointments or consultations and eventually cause the trust from academic staff to theirs declined.

5. There are some complains about there are some of the academic staff do no reply on students’ appointment or consultation requests.

III. Time-consuming in manual appointment acknowledgement

Academic staff are the staff which directly involved with academic programming and research for the university. The workload of the staff are massive and their timetable are mostly fully packed during the working hours. Although acknowledgement process may take only a few minutes to complete, but multiply with the workload of staff and the inefficient method for scheduling management, the easy process will turn out to be time-consuming.

The following are the issues:
1. The current workflow is not convenient to requesters as they are unable to view the appointment schedule of staff members. So, they are not able to arrange a more suitable period of time for appointment before the booking is taken in place. Thus, they may need to send multiple request to get a perfect time for appointment.

2. Due to the massive workload such as attending classes, making consultation to students and etc., academic staff are not able to check on the email, phone and social network frequently. Since all the appointment must be acknowledge manually, the process of appointment booking will be prolong if the acknowledgement delayed.

3. In current workflow, appointments are make through different platforms such as Facebook messenger, phone call, phone message, WhatsApp application and email. It is quite time-consuming as the staff are require to check through multiple platforms for any request about appointment booking.

4. In past events, there are cases where different parties book appointment with the same academic staff at the same period of time and date. The academic staff require to manually accept the appointment request from the party who has the most priority. This cause inconvenient to them as doing this is quite time-consuming when the staff require to manually reply their request one by one.

5. The working hours for academic staff are normally starts from 9 to 5, therefore the staff may not bother to look for the requests of appointment booking after working hours. The appointment date and time may not be confirmed immediately since all works need to be done manually. The requester requires a long wait for the reply.

IV. Common mistake in appointment scheduling

Human make mistakes and in current workflow, mistake may be made if manually manage and arrange the requests of appointment booking. For instance, an academic staff may accidentally arrange two different appointments to be held at the same time period (double booking).

V. No centralized platform for appointment booking
In UTAR, the requesters such as student, staff and external visitor request their appointment through multiple platforms such as email, phone call, phone message, WhatsApp communication application and so. There is no proper platform available and cause inconvenient to academic staff as he/she requires to frequently check on multiple platforms to ensure all appointment requests are acknowledged.

VI. Lack of records on appointment

In a common situation, a student should attend every appointment that he/she has made if the request has been approved. But in contrast, there are some situations where students did not show up for the appointment without any notification once or more than once. This bad behaviour from students do cause discomfort to academic staff as they have to sacrifice their time on waiting the appointments where no outcome will be produced.

Unfortunately, in UTAR, there is no system that allows academic staff to provide feedback on this behaviour from students in order to set up a penalty in order to prevent these situations to reoccur. Besides feedback, there is no system that will keep track on every appointment and keep the appointment details (duration, title, content) for future analysis.
CHAPTER 3

METHODOLOGY AND DEVELOPMENT TOOLS

3.1 System Development Methodology

A system development methodology can be defined as the framework that is used to control, plan, and structure the whole development process of an information system. A variety of models have evolved over the years and each of them have their own recognized strengths and limitations. Each of the available methodologies is designed to suit different types of projects which based on various organizational, project, technical and team considerations. It is vital to choose the correct methodology that suit the project as it has very high impact on the whole development process that is carried out.

The following are some of the software development methodologies/models that will be discussed such as traditional software development life cycle methodologies (waterfall, incremental, spiral and prototyping model) and agile methodologies.

3.1.1 Waterfall Model

The waterfall model is a traditional model that approaches system development in a linear and sequential manner, completing one activity before the other. Ian Sommerville (Sommerville 2011), the authors of “Software Engineering” explained that waterfall model takes the fundamental process activities of specification, development, validation, and evolution and represents them as separate process phases such as requirements definition, system and software design, implementation and unit testing and so on as the Figure 3.1 Error! Reference source not found. shown below.

![Figure 3.1: The phases of Waterfall model](image-url)
In each phase of a waterfall model, it carries specific goal. All the development activities need to be planned and scheduled before starting working on them (Sommerville 2011). Pressman (Pressman 2010) also stresses that a phase requires to be completed before the development proceeds to the next phase. The developments shall be proceed without overlapping between stages. And once the phase is completed, it is not recommended to go back and revert to the earlier phase for amendment as depicted in Figure 3.1. The reason is because iterations can be costly and involves significant rework by repeating previous process stages for any adjustment (Sommerville 2011).

The following are the advantages of waterfall model:

<table>
<thead>
<tr>
<th>Advantage</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Waterfall model is simple to implement since it is a linear and structured model.</td>
<td>(Adenowo and Adenowo 2013)</td>
</tr>
<tr>
<td>2. Structure for organizing and controlling a software development is provided to ensure the adequacy of documentation and design reviews which in the end helps to ensure the reliability, quality and maintainability of the developed software.</td>
<td>(Kannan, Jhajharia and Verma 2014)</td>
</tr>
<tr>
<td>3. Waterfall style conserves resources. Waterfall model use lesser resources compare to other models.</td>
<td></td>
</tr>
<tr>
<td>4. A lot of emphasis is done on paperwork to ensure quality of product. Furthermore, the paperwork allows new worker to catch up with the progress of development.</td>
<td></td>
</tr>
</tbody>
</table>

The following are the disadvantages of waterfall model:

<table>
<thead>
<tr>
<th>Disadvantage</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Inflexible, low adaptability, costly and slow due to significant structure and tight controls.</td>
<td>(Kannan, Jhajharia and Verma 2014)</td>
</tr>
<tr>
<td>2. Hard to measure progress within stages or phases.</td>
<td></td>
</tr>
</tbody>
</table>
3. It is a poor model for large project as it takes a lot of time for the development of SRS.

4. Model mainly depends upon early identification and specification of requirements which is not possible as users may not be able to clearly define what they need early in the project. (Pressman 2010)

5. A working version of product may not be available for user until late in the project time span.

6. A model can accommodate iteration but indirectly. Therefore, when iterating a phase, the iteration takes considerable effort for rework.

7. High effort and costs for writing and approving documents for each development phase. (Sommerville 2011)

8. Linear model is extremely hard to respond to changes. Changes that occur later in life cycle are more costly and therefore discouraged.

9. Problems of finished phases are left for later phases to solve.

10. Problems are often not discovered until system testing. (Petersen, Wohlin and Baca 2009)

11. Big-bang integration and test of the whole system in the end of the project can lead to unexpected quality problems, high costs, and schedule overrun as no technology, business bottleneck or challenges can be identified earlier.

The situations where most appropriate for waterfall model to be applied:

Table 3.3: Appropriate situations for waterfall model

<table>
<thead>
<tr>
<th>Situation</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Project requirements can be stated comprehensively, unambiguously and well understood</td>
<td>(Sommerville 2011)</td>
</tr>
</tbody>
</table>
| 2. Project requirements are stable or unlikely to change during the system development life cycle. | }
3. Product definition is stable. (Kannan, Jhajharia and Verma 2014)

4. Ample resources with required expertise are available to support the product.

3.1.2 **Spiral Model**

Spiral model is a risk-driven software process model that was introduced by Boehm at 1988 (Sommerville 2011). Pressman (Pressman 2010) explained that the spiral model is an evolutionary software process model that couples the iterative nature of prototyping with the systematic and controlled aspects of the waterfall model. The model is called as spiral as a software project will repeatedly passes through these phases in iterations. In this model, there is 4 phases such as planning, risk analysis, engineering and evaluation (Govardhan and Nabil Mohammed Ali Munassar 2010) as the Figure 3.2 shown below.

![Figure 3.2: The phases of spiral model](image)

<table>
<thead>
<tr>
<th>Phase</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning</td>
<td>Gather requirements</td>
</tr>
<tr>
<td>Risk Analysis</td>
<td>Identify risk and alternate solutions</td>
</tr>
<tr>
<td></td>
<td>Produce prototype at the end of the phase</td>
</tr>
<tr>
<td>Engineering</td>
<td>Produce software</td>
</tr>
<tr>
<td></td>
<td>Test software at the end of the phase</td>
</tr>
<tr>
<td>Evaluation</td>
<td>Let customer to evaluate the output of the project</td>
</tr>
</tbody>
</table>
The spiral model starts with baseline spiral in planning phase. And after the pass through every phases, a subsequent spiral will be built on the baseline spiral. The process will be repeated after the whole phases are passed.

The following are the advantages of spiral model:

<table>
<thead>
<tr>
<th>Advantage</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. High amount of risk analysis which enhance risk avoidance.</td>
<td>(Govardhan and Nabil Mohammed Ali Munassar 2010)</td>
</tr>
<tr>
<td>2. Software is produced early in software development lifecycle.</td>
<td></td>
</tr>
<tr>
<td>3. Project monitoring is easy and effective as it is done in each phase of iteration.</td>
<td></td>
</tr>
<tr>
<td>4. Model is flexible as additional functionality can be added at a later date.</td>
<td>(Adel Alshamrani and Abdullah Bahattab 2015)</td>
</tr>
<tr>
<td>5. Suitable for high risk projects where business needs may be unstable.</td>
<td></td>
</tr>
<tr>
<td>6. Suitable to develop a highly customized product.</td>
<td></td>
</tr>
</tbody>
</table>

The following are the disadvantages of spiral model:

<table>
<thead>
<tr>
<th>Disadvantage</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Cost involved in the model is usually high.</td>
<td>(Adel Alshamrani and Abdullah Bahattab 2015)</td>
</tr>
<tr>
<td>2. Risk analysis usually requires highly specific expertise.</td>
<td></td>
</tr>
<tr>
<td>3. Amount documentation required in intermediate stages makes the management of a project very complex.</td>
<td></td>
</tr>
<tr>
<td>5. Success rate of a project is highly dependent on the risk analysis phase</td>
<td></td>
</tr>
</tbody>
</table>
The situations where most appropriate for spiral model to be applied:

<table>
<thead>
<tr>
<th>Situation</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Risk management and avoidance is a high priority.</td>
<td>(Sommerville 2011)</td>
</tr>
<tr>
<td>Costs are high in priority.</td>
<td>(Adel Alshamrani and Abdullah Bahattab 2015)</td>
</tr>
<tr>
<td>For medium to high-risk projects.</td>
<td></td>
</tr>
<tr>
<td>Significant changes are expected.</td>
<td></td>
</tr>
</tbody>
</table>

### 3.1.3 Incremental Model

The incremental model combines elements of linear and parallel process flows in an iterative fashion (Pressman 2010). The basic requirements are usually addressed in the first increment which is the core product. Although the basic requirements are addressed but many supplementary features remain undelivered at the increment. A plan will be developed after undergoes an evaluation on core product by customers. Modification of the core product normally will be made to better meet the needs of customer and the delivery of additional features. This process will be repeated following the delivery of each increment until the complete product is produced (Adel Alshamrani and Abdullah Bahattab 2015).

![Incremental model](image-url)
The following are the advantages of incremental model:

Table 3.8: Advantages of incremental model

<table>
<thead>
<tr>
<th>Advantage</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Less risk as the risk is spread across smaller increments instead of concentrating in one large development.</td>
<td>(Adel Alshamrani and Abdullah Bahattab 2015)</td>
</tr>
<tr>
<td>2. Lesson learned about issues, challenges and risks from each increment can be utilized or applied to the next increment.</td>
<td></td>
</tr>
<tr>
<td>3. Generate working software quickly and early during the software development life cycle.</td>
<td>(Pressman 2010)</td>
</tr>
<tr>
<td>4. Customer able to provide feedback and evaluation as software is built early during life cycle.</td>
<td></td>
</tr>
<tr>
<td>5. Less costly to change the scope or requirements. The amount of analysis and documentation that has to be redone is much lesser.</td>
<td>(Sommerville 2011)</td>
</tr>
</tbody>
</table>

The following are the disadvantages of incremental model:

Table 3.9: Disadvantages of incremental model

<table>
<thead>
<tr>
<th>Disadvantage</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Model requires good planning and design.</td>
<td>(Adel Alshamrani and Abdullah Bahattab 2015)</td>
</tr>
<tr>
<td>2. The model does not allow for overlapping/ iterations within each increment.</td>
<td></td>
</tr>
<tr>
<td>3. A clear and complete definition of the whole functional system is required before broken down and built incrementally.</td>
<td>(Sommerville 2011)</td>
</tr>
<tr>
<td>4. Process is not visible as the managers need regular deliverables to measure the progress.</td>
<td></td>
</tr>
</tbody>
</table>
The situations where most appropriate for incremental model to be applied:

Table 3.10: Appropriate situations for incremental model

<table>
<thead>
<tr>
<th>Situation</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. The model has clear and defined requirements of a functional system.</td>
<td></td>
</tr>
<tr>
<td>3. There is a high risk features and goals to develop the system.</td>
<td></td>
</tr>
<tr>
<td>4. A new technology is being used.</td>
<td></td>
</tr>
<tr>
<td>5. A project with lengthy development schedule.</td>
<td></td>
</tr>
</tbody>
</table>

3.1.4 Prototyping model

Prototyping model is a system development model in which a prototype (working model with limited features) is developed, tested and then refined as necessary until an acceptable and satisfied prototype is built from which the complete system can now be developed (Pressman 2010). There are 4 general steps to design a software prototype: design, build the initial prototype, customer evaluation, and review and refining the prototype.

Evolutionary and throwaway prototyping are the two common methods used in the industry. Evolutionary prototyping is a lifecycle model in which the system is developed in incremental manners so that it can readily be modified in response to end-user or customer feedback (McConnell 1996). The prototype will undergo a series of refinements and should eventually become the solution. While, Throwaway Prototyping is refers to the creation of model that will eventually be discarded rather
than be reused for the final delivered software after getting feedback which can be incorporated into the development of main system (Sommerville 2011).

The following are the advantages of prototyping model:

Table 3.11: Advantages of prototyping model

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Missing features or errors can be detected earlier which helps to reduce time and cost for rework.</td>
<td>(Pressman 2010)</td>
</tr>
<tr>
<td>2. A working prototype provides developers a better insight of the system being developed.</td>
<td></td>
</tr>
<tr>
<td>3. High user involvement in the product before actual implementation.</td>
<td>(Sommerville 2011)</td>
</tr>
<tr>
<td>4. Provides quick implementation of an incomplete but functional application</td>
<td></td>
</tr>
</tbody>
</table>

The following are the disadvantages of prototyping model:

Table 3.12: Disadvantages of prototyping model

<table>
<thead>
<tr>
<th>Disadvantages</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Users may get confuse with the prototype and actual system</td>
<td>(Sommerville 2011)</td>
</tr>
<tr>
<td>2. Increase complexity of system as it may expand beyond original plans</td>
<td></td>
</tr>
<tr>
<td>3. Developer may try to reuse the prototype to build the actual system.</td>
<td>(Pressman 2010)</td>
</tr>
</tbody>
</table>

The situations where most appropriate for incremental model to be applied:

Table 3.13: Appropriate situations for prototyping model

<table>
<thead>
<tr>
<th>Situation</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. When desired system requires a lot of interaction with the end users.</td>
<td>(Sommerville 2011)</td>
</tr>
<tr>
<td>2. When project objectives are blur and unclear.</td>
<td></td>
</tr>
</tbody>
</table>
3.1.5 Rapid Application Development

Rapid Application Development (RAD) model is a software development model that stresses on rapid delivery of product within short time period by minimize the preplanning process. RAD model is designed based on rapid prototyping. A functional prototype is developed as the working model which equivalent to a component of product. Each prototype represent a part of modules of a software which developed in parallel and all of them integrated to make a complete product at the end of development phase.

According to James Martin, RAD model divides the SDLC processes into four phases such as requirements planning phase, user design phase, construction phase and cutover phase.

![Figure 3.5 Rapid Application Development Phases](image)

The following are the advantages of RAD model:

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Reduced development time</td>
<td>(Sommerville 2011)</td>
</tr>
<tr>
<td>2. High reusability of components</td>
<td></td>
</tr>
<tr>
<td>3. Enable quick initial reviews</td>
<td>(Pressman 2010)</td>
</tr>
<tr>
<td>4. Encourages customer feedback</td>
<td></td>
</tr>
<tr>
<td>5. Less Integration issues, higher quality</td>
<td></td>
</tr>
</tbody>
</table>
The following are the disadvantages of RAD model:

Table 3.15: Disadvantages of RAD model

<table>
<thead>
<tr>
<th>Disadvantages</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Fast development speed might compromise quality.</td>
<td>(Pressman 2010)</td>
</tr>
<tr>
<td>2. Difficulties to modularize software.</td>
<td></td>
</tr>
<tr>
<td>3. Requires strong commitment from team member</td>
<td>(Sommerville 2011)</td>
</tr>
</tbody>
</table>

3.1.6 Agile Model

Agile development model is a combination of incremental and iterative process models with the emphasis on process adaptability and customer satisfaction by rapid delivery of working product (Pressman 2010). Agile model is designed to be flexible and adaptable to handle projects with different requirements (Kannan, Jhajharia and Verma 2014). It provides methods to access the development and risks along with the direction throughout the development lifecycle. In brief, the method will break the product into small incremental builds which are provided in iterations. Each of the iteration will eventually involves cross functional teams working simultaneously on various areas such as requirement gathering, planning, design and so for two to three weeks. At the end of each iteration, the working product will be shown to customers for evaluation. And each build is incremental in terms of features based on the request from customers after evaluation and thus final working product should hold all the features required by them.

Figure 3.6: Agile model
The agile model works following the Agile Manifesto principles (Sommerville 2011):  

<table>
<thead>
<tr>
<th>Principle</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Individuals and interactions over processes and tools</td>
<td></td>
</tr>
<tr>
<td>2. Working software over comprehensive documentation</td>
<td></td>
</tr>
<tr>
<td>3. Customer collaboration over contract negotiation</td>
<td></td>
</tr>
<tr>
<td>4. Responding to change over following a plan</td>
<td></td>
</tr>
</tbody>
</table>

Pressman (Pressman 2010) emphasis that an agile development model covers different kinds of agile process models such as Extreme programming (XP), Scrum, Crystal, Feature Drive Development, Adaptive Software Development and so. Extreme programming (XP) is the most well-known and widely used agile method (Pressman 2010). It is organized as 4 basic framework activities such as planning, design, coding and testing. It integrated a good range of programming practices and a number of innovative techniques that allows the development team to develop a frequent software releases that deliver the features that have been described and then prioritized by the customers. Besides XP, the other agile process models also stress team self-organization and human collaboration but they did define their own framework activities and select different points of emphasis.

The following are the advantages of agile model:

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Working software product is available in the early phases of the software as agile method focus on rapid development and frequent releases of the software.</td>
<td>(Sommerville 2011)</td>
</tr>
<tr>
<td>2. People and interactions are emphasized rather than process and tools. Developers, testers and customers constantly communicate with each other.</td>
<td>(Pressman 2010)</td>
</tr>
<tr>
<td>3. Team work is promoted Everyone involved work together daily throughout the project.</td>
<td></td>
</tr>
<tr>
<td>4. Allows changing of requirements even late in development.</td>
<td></td>
</tr>
</tbody>
</table>
5. This model does not dictate on documentation and minimum rules employed. (Kannan, Jhajharia and Verma 2014)

6. Provides flexibility and the model is easy to manage.

The following are the disadvantages of agile model:

Table 3.18: Disadvantages of agile model

<table>
<thead>
<tr>
<th>Disadvantages</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. This model is not suitable for handling complex dependencies.</td>
<td>(Kannan, Jhajharia and Verma 2014)</td>
</tr>
<tr>
<td>2. Provides more risk of extensibility, sustainability and maintainability.</td>
<td></td>
</tr>
<tr>
<td>3. Depends heavily on customer interaction. Without the cooperation of users</td>
<td></td>
</tr>
<tr>
<td>in development process, the process would be hampered and may lead to failure.</td>
<td>(Kannan, Jhajharia and Verma 2014)</td>
</tr>
<tr>
<td>4. The model is not suit for projects with strictly requirements and scopes.</td>
<td></td>
</tr>
</tbody>
</table>

The situations where most appropriate for agile model to be applied:

Table 3.19: Appropriate situations for agile model

<table>
<thead>
<tr>
<th>Situation</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. When requirements are not well specified or when the requirements</td>
<td>(Sommerville 2011)</td>
</tr>
<tr>
<td>are expected to alter during later phases of the development.</td>
<td></td>
</tr>
<tr>
<td>2. The product needs to be developed in a limited time.</td>
<td>(Kannan, Jhajharia and Verma 2014)</td>
</tr>
<tr>
<td>3. Customer is mostly involved.</td>
<td></td>
</tr>
</tbody>
</table>

3.1.7 Comparison between system methodologies

Table 3.20: Comparison between system methodologies

<table>
<thead>
<tr>
<th>Usefulness in Developing Systems</th>
<th>Waterfall</th>
<th>Spiral</th>
<th>Incremental</th>
</tr>
</thead>
<tbody>
<tr>
<td>With unclear user requirements</td>
<td>Poor</td>
<td>Poor</td>
<td>Good</td>
</tr>
<tr>
<td>With unfamiliar technology</td>
<td>Poor</td>
<td>Poor</td>
<td>Good</td>
</tr>
<tr>
<td>Complex</td>
<td>Good</td>
<td>Good</td>
<td>Good</td>
</tr>
</tbody>
</table>
3.1.8 Conclusion

Based on the review and comparison between several development methodologies, Rapid application development model is selected to be adopted as the methodology for the development of this project entitled Online Academic Appointment Scheduling System. Rapid application development model is defined as a software development methodology which focuses on the process of development instead of the preplanning process. This methodology is chosen rather than the other methodologies because of the benefit of shorter development time. It suits my project which requires a working product within a short time period (2 to 3 months) and it can be modularized.

RAD consists of phases such as requirements planning phase, user design phase, construction phase and cutover phase. My actions on those phases will be explained in the following subsections.

| Reliable                      | Good | Good | Good |
| With short time schedule      | Poor | Good | Excellent |
| With schedule visibility      | Poor | Poor | Excellent |

<table>
<thead>
<tr>
<th>Usefulness in Developing Systems</th>
<th>Prototyping</th>
<th>RAD</th>
<th>Agile</th>
</tr>
</thead>
<tbody>
<tr>
<td>With unclear user requirements</td>
<td>Excellent</td>
<td>Excellent</td>
<td>Excellent</td>
</tr>
<tr>
<td>With unfamiliar technology</td>
<td>Good</td>
<td>Excellent</td>
<td>Poor</td>
</tr>
<tr>
<td>Complex</td>
<td>Poor</td>
<td>Excellent</td>
<td>Poor</td>
</tr>
<tr>
<td>Reliable</td>
<td>Poor</td>
<td>Excellent</td>
<td>Good</td>
</tr>
<tr>
<td>With short time schedule</td>
<td>Excellent</td>
<td>Good</td>
<td>Excellent</td>
</tr>
<tr>
<td>With schedule visibility</td>
<td>Excellent</td>
<td>Good</td>
<td>Good</td>
</tr>
</tbody>
</table>

Figure 3.7 Rapid Application Development Phases
3.1.8.1 Action of Requirement Planning Phase

In this phase, an investigation is going to be carried out to obtain more understanding on the basic requirements of my project’s proposed system. In my perspective, the investigation is carried out to obtain several information:

1. The primary stakeholders of proposed system development.
2. The current workflow of appointment scheduling.
3. The problems existed in current workflow.
4. The objectives and scopes of the proposed system.

In order to gather these information, a few requirements gathering methods: interviews, surveys (using questionnaires), and literature reviews are implemented.

Face-to-face contact with users through individual interviewing is the primary source of requirements and an important way to gather and validate the requirements as interviewing will provide rich qualitative data about projects, key stakeholders, and end users. In this project, unstructured interview is chosen to be the requirements gathering method instead of semi or structured interview. The reason is that an unstructured interview is a spontaneous conversation, not a specific set of questions asked in a predetermined order. It allows interviewer to probe more questions beyond expected answer for more understanding. In order to obtain more in-depth understanding on my project, an interview guide and checklist is prepared along with several open-ended questions used to probe for current workflow of appointment scheduling, existing problems in current workflow of appointment scheduling, the possible solution for the current workflow, expected features in solution and many more related to appointment scheduling. Up to six interviews is done with key stakeholders, academic staff and requester such as staff and students.

Besides interviews, surveys using questionnaires are also selected as one of the requirements gathering methods in my project. The main reason for me to use questionnaires is that this method is the quick and efficient to obtain large amount of data from large sample of people. Although questionnaires bring a lot of demerits such as respondents may lie to present a positive image of themselves and many more, but this method is used to verify or affirm the information that I gathered from interviews and to obtain other possible answers from the respondents. The questionnaires is
design using different types of questions such as multiple choice questions, dichotomous questions, scaled questions, likert questions and so.

Literature review is also implemented on several systems which related to my proposed system – Online Academic Appointment Scheduling System. Through the review, I can obtain large amount of information which are important to my project. For instance, the basic features that should provide by an appointment scheduling system, the additional features which can be implemented in my system, and the features that should not be implement in the system. Furthermore, a few development tools will be identified for the development of application.

After gathering all necessary information, analysis is done along with several UML diagrams such as use case diagram. The use case diagram is used to represent the behaviour of the system so that stakeholders and I can have a better understanding on the system. The requirement gathering phase is ended after all required information has been gathered and analysed.

3.1.8.2 Action of User Design Phase
In this phase, diagrams will be drawn based on the requirements gathered in previous phase. UML diagrams such as use case diagram and sequence diagram and entity relationship diagram are the diagrams that will be drawn to provide an in-depth understanding on how the activities of the system works. The diagrams will be produced by using Enterprise Architecture.

Other than the diagrams, a simple story board will be designed based on the modules that will be included in the actual system. The story board will show the basic user interface design which provide ideas during the development phase. The story board will be produced by using a rapid prototyping software named Axure.

3.1.8.3 Action of Construction Phase
In this phase, the planned system will break down into smaller modules and developed into several prototypes throughout the construction phase. For instance, there will be modules such as appointment management module, availability management module, appointment type management module, appointment rule management module and many more.
The prototypes produced in this phase will be functional and workable which equivalent to a component of product. The modules will be implemented by using web programming languages such as PHP, JavaScript, JQuery, HTML, CSS, SQL and many more. Moreover, testing such as unit testing, integration testing and system testing will be carried out.

3.1.8.4 Action of Cutover Phase
In this phase, an integration will be carried out for all the prototypes developed from previous phase into a complete system. Testing will be carried out once more to ensure the system is able to perform perfectly.

3.2 Development Tools
Web development tools and resources are the necessary items that will aid developer in the development workflow of web application. There are numerous tools and resources available in market: JavaScript libraries, front-end frameworks, web application frameworks, languages, databases, and local dev environment. The following are the tools and resources selected for this project.

3.2.1 JQuery
JavaScript is one of the most popular programming languages on the webA JavaScript library is a library of pre-written JavaScript which allows easier access throughout the development of the website or application. JQuery is selected as the JavaScript library that will be used in this project. The reason that jQuery is chosen is that it is a small, fast in performance and feature rich JavaScript library. It consists of API that works across a multitude of browsers which makes HTML document traversal and manipulation, event handling, Ajax and animation much simpler.

3.2.2 Bootstrap
Front-end frameworks is also known as client-side development tool. It uses HTML, CSS and JavaScript for developing a website or web application that allows user interaction. Bootstrap is chosen as the front-end framework of my project. There are several reasons to choose Bootstrap: it is easy to use, it provide features which aid to build responsive website and many more.
3.2.3 Laravel
A web application framework is a software framework designed to help and aid the development of web applications and services. Laravel is chosen as the web application framework in my project as it is a free, open-source PHP web application framework which build web applications on MVC architecture design pattern.

3.2.4 PHP, HTML5, JavaScript, CSS3 and SQL
Programming language is one of the important development tools as it is the fundamental core to build up an application. A programming language is defined as a formal constructed language designed to communicate instructions to a machine and can be used to create programs to control the behaviour of a machine or to express algorithms. In this project, a few languages will be used: PHP, HTML5, JavaScript, CSS and SQL.

- **PHP**
  PHP is also known as Hypertext Pre-processor. It is a widely-used open source general-purpose scripting language that is especially suited for web development. One of the benefits is that it can be embedded into HTML.

- **HTML5**
  HTML5 is a type of mark-up language used for structuring and presenting content on the web.

- **JavaScript**
  JavaScript is an object-oriented computer programming language commonly used to create interactive effects within web browsers.

- **CSS**
  CSS stands for Cascading Style Sheets. It is a style sheet language used for describing the presentation of a document written in a mark-up language.

- **SQL**
  SQL stands for Structured Query Language which is designed for managing data held in a relational database management system (RDBMS).
3.2.5 MySQL
In every web application, there is a back-end that uses database to store the information that obtained from user through front-end. Same as my application, a relational database is required to store the required data. MySQL is selected as the relational database management system in my project to manage and control the database. MySQL is one of the popular open source databases available in the market.

3.2.6 Axure RP Pro
Axure RP Pro is a software tool for wire-framing, rapid prototyping, documentation and specification. It is selected to develop prototype for basic user interface design as it provide clear visual guidance to me in the process of prototype development.

3.3 Project Plan
This project is planned using Work Breakdown Structure (WBS) and scheduled using Gantt chart.

3.3.1 Work Breakdown Structure
A Work Breakdown Structure (WBS) allows work or tasks of the project to be planned more efficiently. For the WBS in this project, top-down approach is used in which bigger tasks are broke down in smaller tasks to reduce the complexity of the activities. The project is basically divided in 6 different phases: preliminary, requirement gathering, prototype development, development, testing, and deployment and maintenance phases. In prototype development phase, there are 4 sub-phases: design, prototyping, customer evaluation and refinement phases which will be repeated for 3 iteration.

Each phase in the whole project will be further broken down into tasks to be done. The tasks of each phase is shown in the Figure 3.8 below.
Figure 3.8 Work Breakdown Structure diagram
3.3.2 Gantt Chart

Gantt chart provides an overview of the schedules for tasks in a chart. The tasks for each phase is scheduled and shown in table and Gantt charts below.

Table 3.21: Project schedule table

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Duration</th>
<th>Start</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preliminary Phase</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Background Research</td>
<td>3d</td>
<td>06/06/16</td>
<td>06/08/16</td>
</tr>
<tr>
<td>Determine current workflow</td>
<td>6d</td>
<td>06/09/16</td>
<td>06/16/16</td>
</tr>
<tr>
<td>Determine problem statement</td>
<td>3d</td>
<td>06/17/16</td>
<td>06/21/16</td>
</tr>
<tr>
<td>Determine purpose solution</td>
<td>3d</td>
<td>06/22/16</td>
<td>06/24/16</td>
</tr>
<tr>
<td>Determine project objectives</td>
<td>3d</td>
<td>06/27/16</td>
<td>06/29/16</td>
</tr>
<tr>
<td>Determine project scopes</td>
<td>3d</td>
<td>06/30/16</td>
<td>07/01/16</td>
</tr>
<tr>
<td>Requirement Planning Phase</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Literature review</td>
<td>10d</td>
<td>07/05/16</td>
<td>07/18/16</td>
</tr>
<tr>
<td>Fact Findings</td>
<td>4d</td>
<td>07/19/16</td>
<td>07/22/16</td>
</tr>
<tr>
<td>Choose appropriate Methodologies</td>
<td>3d</td>
<td>07/25/16</td>
<td>07/27/16</td>
</tr>
<tr>
<td>Choose appropriate development tools</td>
<td>2d</td>
<td>07/28/16</td>
<td>07/29/16</td>
</tr>
<tr>
<td>User Design Phase</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UML diagram</td>
<td>125d</td>
<td>08/01/16</td>
<td>01/20/17</td>
</tr>
<tr>
<td>Use case diagram</td>
<td>5d</td>
<td>08/01/16</td>
<td>08/05/16</td>
</tr>
<tr>
<td>Use case descriptions</td>
<td>6d</td>
<td>08/08/16</td>
<td>08/15/16</td>
</tr>
<tr>
<td>Sequence diagrams</td>
<td>5d</td>
<td>01/16/17</td>
<td>01/20/17</td>
</tr>
<tr>
<td>ERD diagram</td>
<td>5d</td>
<td>01/23/17</td>
<td>01/27/17</td>
</tr>
<tr>
<td>Logical ERD diagrams</td>
<td>2d</td>
<td>01/23/17</td>
<td>01/24/17</td>
</tr>
<tr>
<td>Physical ERD diagrams</td>
<td>3d</td>
<td>01/25/17</td>
<td>01/27/17</td>
</tr>
<tr>
<td>Data flow diagram</td>
<td>4d</td>
<td>01/30/17</td>
<td>02/02/17</td>
</tr>
<tr>
<td>Storyboard</td>
<td>3d</td>
<td>02/03/17</td>
<td>02/07/17</td>
</tr>
<tr>
<td>Construction Phase</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Construction of modules for academic staff</td>
<td>35d</td>
<td>02/08/17</td>
<td>03/28/17</td>
</tr>
<tr>
<td>Appointment Management Module</td>
<td>31d</td>
<td>02/08/17</td>
<td>03/22/17</td>
</tr>
<tr>
<td>Appointment Making Module</td>
<td>5d</td>
<td>02/08/17</td>
<td>02/14/17</td>
</tr>
<tr>
<td>View Appointments Module</td>
<td>3d</td>
<td>02/15/17</td>
<td>02/17/17</td>
</tr>
<tr>
<td>Appointment Postpone Module</td>
<td>3d</td>
<td>02/20/17</td>
<td>02/22/17</td>
</tr>
<tr>
<td>Module</td>
<td>Duration</td>
<td>Start Date</td>
<td>End Date</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
<td>----------</td>
<td>-------------</td>
<td>------------</td>
</tr>
<tr>
<td>Appointment Cancellation Module</td>
<td>3d</td>
<td>02/20/17</td>
<td>02/22/17</td>
</tr>
<tr>
<td>Appointment Validation Module</td>
<td>3d</td>
<td>02/20/17</td>
<td>02/22/17</td>
</tr>
<tr>
<td>Availability Management Module</td>
<td>2d</td>
<td>02/23/17</td>
<td>02/24/17</td>
</tr>
<tr>
<td>Google Calendar Synchronization Module</td>
<td>2d</td>
<td>02/27/17</td>
<td>02/28/17</td>
</tr>
<tr>
<td>Appointment Rules Management Module</td>
<td>2d</td>
<td>02/27/17</td>
<td>02/28/17</td>
</tr>
<tr>
<td>Appointment Types Management Module</td>
<td>2d</td>
<td>02/27/17</td>
<td>02/28/17</td>
</tr>
<tr>
<td>Intake Questions Management Module</td>
<td>2d</td>
<td>02/27/17</td>
<td>02/28/17</td>
</tr>
<tr>
<td>Reminder Notices Management Module</td>
<td>2d</td>
<td>02/27/17</td>
<td>02/28/17</td>
</tr>
<tr>
<td>Report Generation Module</td>
<td>1d</td>
<td>03/01/17</td>
<td>03/01/17</td>
</tr>
<tr>
<td>Profile Management Module</td>
<td>1d</td>
<td>03/02/17</td>
<td>03/02/17</td>
</tr>
<tr>
<td>Construction of modules for requester</td>
<td>4d</td>
<td>03/03/17</td>
<td>03/08/17</td>
</tr>
<tr>
<td>Appointment Management Module</td>
<td>2d</td>
<td>03/03/17</td>
<td>03/06/17</td>
</tr>
<tr>
<td>Profile Management Module</td>
<td>1d</td>
<td>03/07/17</td>
<td>03/07/17</td>
</tr>
<tr>
<td>Registration Module</td>
<td>1d</td>
<td>03/08/17</td>
<td>03/08/17</td>
</tr>
<tr>
<td>Construction of modules for requester</td>
<td>3d</td>
<td>03/09/17</td>
<td>03/13/17</td>
</tr>
<tr>
<td>Account Management Module</td>
<td>3d</td>
<td>03/09/17</td>
<td>03/13/17</td>
</tr>
<tr>
<td><strong>Cutover Phase</strong></td>
<td><strong>15d</strong></td>
<td><strong>03/14/17</strong></td>
<td><strong>04/03/17</strong></td>
</tr>
<tr>
<td>Prototypes Integration</td>
<td>5d</td>
<td>03/14/17</td>
<td>03/20/17</td>
</tr>
<tr>
<td>Testing</td>
<td>10d</td>
<td>03/21/17</td>
<td>04/03/17</td>
</tr>
<tr>
<td>Unit Testing</td>
<td>3d</td>
<td>03/21/17</td>
<td>03/23/17</td>
</tr>
<tr>
<td>Integration Testing</td>
<td>2d</td>
<td>03/24/17</td>
<td>03/27/17</td>
</tr>
<tr>
<td>System Testing</td>
<td>2d</td>
<td>03/28/17</td>
<td>03/29/17</td>
</tr>
<tr>
<td>User Acceptance Test</td>
<td>2d</td>
<td>03/31/17</td>
<td>04/03/17</td>
</tr>
</tbody>
</table>
Figure 3.9: Project 1 Gantt chart
Figure 3.10: Project 2 Gantt chart (part 1)
Figure 3.11: Project 2 Gantt chart (part 2)
4.1 Functional Requirements

Functional Requirements for Academic Staff

1. Appointment Management Module
   a. The system must be able to perform appointment making, appointment viewing, appointment time bring forward/postpone, appointment cancelling and appointment validating.
   b. The system should provide automated approval for appointment making, appointment time bring forward/postpone and appointment cancelling.
   c. The system must be able to save, update and delete the appointment’s record to database.
   d. The system must be able to send reminder notices through email to both parties: academic staff and requester.

2. Availability Management Module
   a. The system must allow academic staff to manage their availabilities in the system either create new, update existing and delete existing availabilities.
   b. The system must be able to create a booking form with free time slots using the availabilities information saved by academic staff.
   c. The system must be able to save, update and delete the record of availability to database.

3. Settings Module
   a. The system must allow academic staff to add new, update existing or delete existing appointment type which used to identify the purpose of an appointment.
   b. The system must allow academic staff to modify appointment rules for appointment management. It should restrict on the time for appointment booking, appointment bring forward/postpone, appointment cancellation.
Moreover, it should allow the configuration on prioritization of appointment management.

c. The system must allow academic staff to manage on google calendar synchronization. The system should be able to save the google access information and able to add new, update existing and delete existing google event for academic appointments.

d. The system must allow academic staff to add new, update existing or delete existing intake questions which used to obtain more information about the purpose of appointment.

e. The system must allow academic staff to manage on the reminder notices. It should allow academic staff to choose either using default notice or customize their on reminders.

f. All the settings should be able to save to database.

4. Generate Reports Module
   a. The system must be able to generate different types of reports by using records available in database.

5. Profile Management Module
   a. The system must allow academic staff to view or modify his/her account’s information.
   b. The system must allow academic staff to change the account’s password.

**Functional Requirements for Requester**

1. Appointment Management Module
   a. The system must be able to perform appointment making, appointment time bring forward/postpone and appointment cancelling.
   b. The system should provide automated approval for appointment making, appointment time bring forward/postpone and appointment cancelling.
   c. The system should able to display all appointments to requester whenever he/she needed.
   d. The system must be able to save, update and delete the appointment’s record to database.
e. The system must be able to send reminder notices through email to both parties: academic staff and requester.

2. Profile Management Module
   a. The system must allow academic staff to view or modify his/her account’s information.
   b. The system must allow academic staff to change the account’s password.

3. Account Registration Module
   a. The system must allow requester to create his/her own account.
   b. The account’s information must be saved to database for profile management.

**Functional Requirements for Administrator**

1. Account Management Module
   a. The system must allow administrator to create accounts for academic staff and administrator.
   b. The system must allow administrator to view on the accounts’ information of academic staff, requesters and administrators.
   c. The system must allow administrator to activate or deactivate the account of academic staff, requester and administrator.
   d. The system must allow administrator to change account’s password for the academic staff, requester and administrator account.

4.2 Use Case

Each use case describe a scenario in which the use interacts with the system to achieve a specific goal or to accomplish a particular task (Wiegers 1997). A use case diagram and use case descriptions are included in this section. A use case diagram represent the user’s interaction with the system by showing the relationship between the user and the different use cases whereas a use case description is a list which describe each use case in detail of how users will perform the tasks.
4.2.1 Use Case Diagram

Figure 4.1: Use Case Diagram
4.2.2 Use Case Description

4.2.2.1 Manage Appointments

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Manage Appointments</td>
</tr>
<tr>
<td>Actors</td>
<td>Academic staff</td>
</tr>
<tr>
<td>Description</td>
<td>Academic staff can make, view, postpone, cancel and validate appointments</td>
</tr>
</tbody>
</table>

**Preconditions**
- Make appointment
  Academic staff must register himself/herself as requester before making any appointment.

**Post Conditions**
- Normal Flow
  1. Academic staff log into the system by using given username and password.
  2. Academic staff select on make appointment feature.
  3. System will extract all the other staff available for him/her to choose.
  4. Academic staff choose one of the other staff.
  5. System will extract free time slots of the targeted staff and display to the academic staff.
  6. Academic staff is then select the preferred time slot and insert required information.
  7. System will validate the input.
  8. Add new appointment function will be computed if the input is valid.
  9. Confirmation notice will be sent to both parties: academic staff and targeted staff.

- View appointments
  1. Academic staff log into the system by using given username and password.
  2. Academic staff select on view appointments feature.
3. System will extract all appointments from database and display them to academic staff.
4. Academic staff is then able to view on details of every appointment.

- **Bring forward/Postpone Appointment**
  1. Academic staff log into the system by using given username and password.
  2. Academic staff perform view appointments feature.
  3. Academic staff select on bring forward/postpone feature in the view of appointment details.
  4. Academic staff submit the postpone request with the new appointment date and time.
  5. System will validate the input.
  6. Postpone of existing appointment function will be computed if the input is valid.
  7. System will then send postpone notice to involved parties.

- **Cancel Appointment**
  1. Academic staff log into the system by using given username and password.
  2. Academic staff perform view appointments feature.
  3. Academic staff select on cancellation feature in the view of appointment details.
  4. Academic staff submit the cancellation request with the cancellation reason.
  5. System will validate the input.
  6. Cancellation of existing appointment function will be computed if the input is valid.
  7. System will then send cancellation notice to involved parties.
• Validate Appointment
  1. Academic staff log into the system by using given username and password.
  2. Academic staff perform view appointments feature.
  3. System display validation feature if the appointment period has passed.
  4. Academic staff select on validation feature in the view of appointment details.
  5. Academic staff submit the validation request with the information about attendance and punctuality of requester.
  6. System will validate the input.
  7. Validation of existing appointment function will be computed if the input is valid.
  8. System will then save the information for future reference.

Alternative Flow
• Bring forward/Postpone Appointment
  3.1 System will extract own availabilities from database if the appointment is requested by others.
  3.2 System will extract the target availabilities from database if the appointment is requested by own self to targeted staff.

4.2.2.2 Manage Availabilities

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Manage Availabilities</td>
</tr>
<tr>
<td>Actors</td>
<td>Academic staff</td>
</tr>
<tr>
<td>Description</td>
<td>Availability is stands for the availability time period of academic staff. This use case describes academic staff can view, add, update and delete their available time for appointment making section.</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Academic staff log into the system by using given username and password.</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Post Conditions</td>
<td>Academic staff is available for appointments.</td>
</tr>
</tbody>
</table>
| Normal Flow        | - View Availability  
|                    | 1. Academic staff select on the feature of view availabilities.  
|                    | 2. System will execute the function of getting all the availabilities of the academic staff.  
|                    | 3. Academic staff will be able to view his/her availabilities in calendar format.  
|                    | - Add Availability  
|                    | 1. Academic staff perform feature of view availabilities.  
|                    | 2. Academic staff drag a new time range for his/her availability.  
|                    | 3. System check on the input and save it to database.  
|                    | - Update Availability  
|                    | 1. Academic staff perform feature of view availabilities.  
|                    | 2. Academic staff drag the existing availability to a new time range in the calendar.  
|                    | 3. System check on the input and save it to database.  
|                    | - Delete Availability  
|                    | 1. Academic staff perform feature of view availabilities.  
|                    | 2. Academic staff click on the existing availability.  
|                    | 3. System will delete the availability after academic staff confirms on it.  |
| Alternative Flow   | -                                                                       |
### 4.2.2.3 Manage Google Calendar Synchronization

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Manage Google Calendar Synchronization</td>
</tr>
<tr>
<td>Actors</td>
<td>Academic staff</td>
</tr>
<tr>
<td>Description</td>
<td>Academic staff can either connect to his/her google calendar or disconnect from it.</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Academic staff log into the system by using given username and password.</td>
</tr>
</tbody>
</table>
| Post Conditions | 1. Connect  
   1. Time schedule of academic staff stays up to date with schedule in Google Calendar.  
   2. Disconnect  
   1. Time schedule of academic staff in system will not be up to date with schedule in Google Calendar. |
| Normal Flow | Connect  
1. Academic staff select on the feature of google calendar synchronization.  
2. Academic staff perform the connection with google by entering his/her google account information at the consent screen which redirected by the system.  
3. System will then save the connection details after the connection completed.  

   Disconnect  
1. Academic staff select on the feature of google calendar synchronization.  
2. Academic staff perform the disconnect feature with google.  
3. System will then remove all the connection details after the disconnection is completed. |
| Alternative Flow | - |
### 4.2.2.4 Manage Setting of Reminder Notice

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Manage Setting of Reminder Notice</td>
</tr>
<tr>
<td>Actors</td>
<td>Academic staff</td>
</tr>
<tr>
<td>Description</td>
<td>Academic staff can configure notification content for approval, postpone, cancellation and amendment notices.</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Academic staff log into the system by using given username and password.</td>
</tr>
<tr>
<td>Post Conditions</td>
<td>The content for notices are in the academic staff’s preferences.</td>
</tr>
</tbody>
</table>
| Normal Flow | 1. Academic staff select to feature of customization of reminder notice.  
2. Academic staff can select either keep the default notice content provided by the system or customize the content by himself/herself.  
3. Academic staff saves the setting. |
| Alternative Flow | - |

### 4.2.2.5 Manage Appointment Types

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Manage Appointment Types</td>
</tr>
<tr>
<td>Actors</td>
<td>Academic staff</td>
</tr>
<tr>
<td>Description</td>
<td>Appointment type is used to determine the purpose for each appointment. Academic staff can either add new, update existing or delete existing appointment type.</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Academic staff log into the system by using given username and password.</td>
</tr>
<tr>
<td>Post Conditions</td>
<td>Appointment made can be differentiate using appointment type.</td>
</tr>
</tbody>
</table>
| Normal Flow | 1. Academic staff select on the feature of appointment types setting.  
2. System will extract all the appointment types from database and display them in list.  
3. Academic staff can choose to either add new, update existing or delete existing appointment types. |
### Alternative Flow

<table>
<thead>
<tr>
<th>3.1 Add New</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.1 Academic staff inserts new appointment type.</td>
</tr>
<tr>
<td>3.1.2 Academic staff saves the input.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3.2 Update Existing</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2.1 Academic staff select the existing appointment type.</td>
</tr>
<tr>
<td>3.2.2 Academic staff change the value for existing appointment type.</td>
</tr>
<tr>
<td>3.2.3 Academic staff saves the input.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3.3 Delete Existing</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.1 Academic staff select the existing appointment type.</td>
</tr>
<tr>
<td>3.3.2 Academic staff perform deletion of the existing appointment type.</td>
</tr>
</tbody>
</table>

### 4.2.2.6 Manage Appointment Rules

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Manage Appointment Rules</td>
</tr>
<tr>
<td>Actors</td>
<td>Academic staff</td>
</tr>
<tr>
<td>Description</td>
<td>Appointment rules are used to define time limit for appointment making, postpone and cancellation. It also includes the setting of prioritization for appointment making.</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Academic staff log into the system by using given username and password.</td>
</tr>
<tr>
<td>Post Conditions</td>
<td>Appointment management includes appointment rules.</td>
</tr>
<tr>
<td>Normal Flow</td>
<td>1. Academic staff select on the feature of appointment types rules.</td>
</tr>
<tr>
<td></td>
<td>2. System will extract all the appointment rules information from database and display them in list.</td>
</tr>
<tr>
<td></td>
<td>3. Academic staff select the unit time and time range for each rules for appointment management process.</td>
</tr>
<tr>
<td></td>
<td>4. Academic staff save the setting.</td>
</tr>
<tr>
<td>Alternative Flow</td>
<td>-</td>
</tr>
</tbody>
</table>
### 4.2.2.7 Manage Intake Form

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Manage Intake Form</td>
</tr>
<tr>
<td>Actors</td>
<td>Academic staff</td>
</tr>
<tr>
<td>Description</td>
<td>Intake form contains question(s) that has the functionality to obtain specific information from requester. Academic staff can customize his/her own questions.</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Academic staff wants to obtain specific information for early preparation for the appointment.</td>
</tr>
<tr>
<td>Post Conditions</td>
<td>-</td>
</tr>
</tbody>
</table>
| Normal Flow | 1. Academic staff log into the system by using given username and password.  
2. Academic staff select on the feature of intake form setting.  
3. Academic staff can choose to either add new, update existing or delete existing intake question. |
| Alternative Flow | 3.1 Add New  
3.1.1 Academic staff inserts new question and select he answer format of that particular question.  
3.1.2 Academic staff saves the input.  
3.2 Update Existing  
3.2.1 Academic staff select the existing intake question.  
3.2.2 Academic staff change the value for existing intake question in terms of question itself and answer format.  
3.2.3 Academic staff saves the input.  
3.3 Delete Existing  
3.3.1 Academic staff select the existing intake question.  
3.3.2 Academic staff perform deletion of the existing intake question. |
### 4.2.2.8 Generate Reports

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Generate Reports</td>
</tr>
<tr>
<td>Actors</td>
<td>Academic staff</td>
</tr>
<tr>
<td>Description</td>
<td>Academic staff can generate different types of reports.</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Academic staff want to generate report to track on specific information.</td>
</tr>
<tr>
<td>Post Conditions</td>
<td>Requested report is generated for academic staff.</td>
</tr>
</tbody>
</table>
| Normal Flow   | 1. Academic staff log into the system by using username and password.  
                  2. Academic staff select the type of report he/she want to generate.  
                  3. Academic staff will then select the date range for the reports.  
                  4. System will filtered out the report based on the type and date range and display to academic staff. |
| Alternative Flow | - |

### 4.2.2.9 Manage Profile

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Manage Profile</td>
</tr>
<tr>
<td>Actors</td>
<td>Academic staff, requester</td>
</tr>
<tr>
<td>Description</td>
<td>User can view and edit profile, along with the change password feature.</td>
</tr>
<tr>
<td>Preconditions</td>
<td>User log into the system by using given username and password.</td>
</tr>
<tr>
<td>Post Conditions</td>
<td>-</td>
</tr>
</tbody>
</table>
| Normal Flow   | 1. User select on manage profile feature.  
                  2. User can choose to either view and edit profile, or change password for its account. |
| Alternative Flow | View Profile  
                  2.1.1 Academic staff select to view profile |
2.1.2 System extracts the current user details and display to the user.

Edit Profile
2.2.1 Academic staff select to edit profile
2.2.2 System extracts the current user details and display to the user.
2.2.3 User can change any information based on his/her preferences.
2.2.4 User saves the input and system will store it to database.

Change Password
2.3.1 User select to change account’s password.
2.3.2 User enter current password, along with new password and confirmation password.
2.3.3 System saves the password after it verifies the correctness of inputs.

4.2.2.10 Manage Appointments

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Manage Appointments</td>
</tr>
<tr>
<td>Actors</td>
<td>Requester</td>
</tr>
<tr>
<td>Description</td>
<td>Requester can make, view, postpone and cancel appointments</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Requester log into the system by using username and password.</td>
</tr>
<tr>
<td>Post Conditions</td>
<td>-</td>
</tr>
<tr>
<td>Normal Flow</td>
<td>• Make Appointments</td>
</tr>
<tr>
<td></td>
<td>1. Requester select on make appointment feature.</td>
</tr>
<tr>
<td></td>
<td>2. System will extract all the academic staff information and display in staff directory for him/her to choose.</td>
</tr>
<tr>
<td></td>
<td>3. Requester choose one of the staff.</td>
</tr>
<tr>
<td></td>
<td>4. System will extract free time slots of the targeted staff and display to him/her.</td>
</tr>
<tr>
<td></td>
<td>5. Requester is then select the preferred time slot and insert required information.</td>
</tr>
</tbody>
</table>
6. System will validate the input.
7. Add new appointment function will be computed if the input is valid.
8. Confirmation notice will be sent to both parties: targeted staff and requester.

- View appointments
  1. Requester select on view appointments feature.
  2. System will extract all appointments from database and display them to him/her.
  3. Requester is then able to view on details of every appointment.

- Postpone Appointment
  1. Requester perform the view appointments feature.
  2. Requester select on postpone feature in the view of appointment details.
  3. System will extract the availabilities of targeted staff from database.
  4. Requester submit the postpone request with the new appointment date and time.
  5. System will validate the input.
  6. Postpone of existing appointment function will be computed if the input is valid.
  7. System will then send postpone notice to involved parties.

- Cancel Appointment
  1. Requester perform the view appointments feature.
  2. Requester select on cancellation feature in the view of appointment details.
  3. Requester submit the cancellation request with the cancellation reason.
4. System will validate the input.
5. Cancellation of existing appointment function will be computed if the input is valid.
6. System will then send cancellation notice to involved parties.

4.2.2.11 Register Account

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Register Account</td>
</tr>
<tr>
<td>Actors</td>
<td>Requester</td>
</tr>
<tr>
<td>Description</td>
<td>Requester can create an account to start making appointment with academic staff in UTAR.</td>
</tr>
<tr>
<td>Preconditions</td>
<td>-</td>
</tr>
<tr>
<td>Post Conditions</td>
<td>Requester able to start manage appointment.</td>
</tr>
</tbody>
</table>
| Normal Flow | 1. Requester visits the web page of online academic appointment scheduling system.  
2. Requester select the feature of account registration on the requester’s login page.  
3. Requester insert the required information for account creation.  
4. System will create an account for the requester after submission of his/her information. |
| Alternative Flow | - |

4.2.2.12 Manage Accounts

<table>
<thead>
<tr>
<th>Use Case ID</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name</td>
<td>Manage Accounts</td>
</tr>
<tr>
<td>Actors</td>
<td>Administrator</td>
</tr>
<tr>
<td>Description</td>
<td>The administrators, ICT staff manage on accounts from all users. Administrator can create new accounts for academic staff</td>
</tr>
</tbody>
</table>
and administrator while he/she can activate or deactivate accounts for all users.

<table>
<thead>
<tr>
<th>Preconditions</th>
<th>Administrator is a legit ICT staff member.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Post Conditions</td>
<td>-</td>
</tr>
<tr>
<td>Normal Flow</td>
<td>1. Administrator can choose either create account or activate/deactivate account or change password.</td>
</tr>
<tr>
<td>Alternative Flow</td>
<td>2.1 Account creation</td>
</tr>
<tr>
<td></td>
<td>2.1.1 Administrator select to create account for either academic staff or administrator.</td>
</tr>
<tr>
<td></td>
<td>2.1.2 Administrator inserts required information for account creation.</td>
</tr>
<tr>
<td></td>
<td>2.1.3 Administrator confirms on the creation and system will create a new account.</td>
</tr>
<tr>
<td></td>
<td>2.2 Account activation/termination</td>
</tr>
<tr>
<td></td>
<td>2.2.1 Administrator select to activate/deactivate account.</td>
</tr>
<tr>
<td></td>
<td>2.2.2 Administrator select an account and perform the activation/termination of account.</td>
</tr>
<tr>
<td></td>
<td>2.2.3 System will then set the status of the account to activated/terminated.</td>
</tr>
<tr>
<td></td>
<td>2.3 Change account’s password</td>
</tr>
<tr>
<td></td>
<td>2.3.1 Administrator select to change password feature.</td>
</tr>
<tr>
<td></td>
<td>2.3.2 Administrator select an account and key in new password.</td>
</tr>
<tr>
<td></td>
<td>2.3.3 System will then updated the password.</td>
</tr>
</tbody>
</table>

### 4.3 Sequence Diagram
A sequence diagram is defined as an integration diagram that describe the interactions among classes in terms of an exchange of message over time. The figures below shows all the sequence diagram for each use case described in section 4.1. The figures are arranged in the order of users: academic staff, requester and administrators.
4.3.1 Sequence Diagrams for Academic Staff

4.3.1.1 Manage Appointments

Figure 4.2 Sequence diagram of Academic Staff (Manage Appointments) Part 1
Figure 4.3 Sequence diagram of Academic Staff (Manage Appointments) Part 2
Figure 4.4 Sequence diagram of Academic Staff (Manage Appointments) Part 3
Figure 4.5 Sequence diagram of Academic Staff (Manage Appointments) Part 4
Figure 4.6 Sequence diagram of Academic Staff (Manage Appointments) Part 5
4.3.1.2  Manage Availabilities

Figure 4.7 Sequence diagram of Academic Staff (Manage Availabilities)
4.3.1.3 Manage Google Calendar Synchronization

Figure 4.8 Sequence diagram of Academic Staff (Manage Google Calendar Synchronization)
4.3.1.4 Manage Setting of Reminder Notice

Figure 4.9 Sequence diagram of Academic Staff (Manage Setting of Reminder Notice)
4.3.1.5 Manage Appointment Types

Figure 4.10 Sequence diagram of Academic Staff (Manage Appointment Types)
4.3.1.6 Manage Appointment Rules

![Sequence diagram of Academic Staff (Manage Appointment Rules)](image)

Figure 4.11 Sequence diagram of Academic Staff (Manage Appointment Rules)
4.3.1.7 Manage Intake Form

Figure 4.12 Sequence diagram of Academic Staff (Manage Intake Form)
4.3.1.8 Generate Reports

Figure 4.13 Sequence diagram of Academic Staff (Generate Reports)
4.3.1.9  Manage Profile

Figure 4.14 Sequence diagram of Academic Staff (Manage Profile)
4.3.1.10 Change Account Password

Figure 4.15 Sequence diagram of Academic Staff (Change Account Password)
4.3.2 Sequence Diagrams for Requester

4.3.2.1 Manage Appointments

Figure 4.16 Sequence diagram of Requester (Manage Appointments) Part 1
Figure 4.17 Sequence diagram of Requester (Manage Appointments) Part 2
Figure 4.18 Sequence diagram of Requester (Manage Appointments) Part 3
Figure 4.19 Sequence diagram of Requester (Manage Appointments) Part 4
Figure 4.20 Sequence diagram of Requester (Manage Appointments) Part 5
4.3.2.2 Register Account

Figure 4.21 Sequence diagram of Requester (Register Account)

4.3.2.3 Manage Profile

Figure 4.22 Sequence diagram of Requester (Manage Profile)
4.3.2.4 Change Account Password

Figure 4.23 Sequence diagram of Requester (Change Account Password Account)
4.3.3 Sequence Diagram for Administrator

4.3.3.1 Manage Accounts

Figure 4.24 Sequence diagram of Administrator (Manage Accounts) Part 1
Figure 4.25 Sequence diagram of Administrator (Manage Accounts) Part 2
Figure 4.26 Sequence diagram of Administrator (Manage Accounts) Part 3
4.4 Entity Relationship Diagram
An entity relation diagram (ERD) shows the relationships of entity (component of data) sets stored in a database. The figures below shows the logical and physical entity relationship diagrams.

4.4.1 Logical Entity Relationship Diagram

![Logical Entity Relationship Diagram](image)

Figure 4.27 Logical Entity Relationship Diagram
4.4.2 Physical Entity Relationship Diagram

Figure 4.28 Physical Entity Relationship Diagram
The ERD diagram for online academic appointment system contains 9 tables as shown in Figure 4.28. Each table has its own function which will be described in the table below.

<table>
<thead>
<tr>
<th>Table Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>staff</td>
<td>Contains the information for every academic staff.</td>
</tr>
<tr>
<td>requesters</td>
<td>Contains the information for every requester.</td>
</tr>
<tr>
<td>admins</td>
<td>Contains the information for every administrator.</td>
</tr>
<tr>
<td>appointments</td>
<td>Contains the information for every appointment made by either academic staff or requester.</td>
</tr>
<tr>
<td>availabilities</td>
<td>Contains the information for every availability declared by academic staff.</td>
</tr>
<tr>
<td>appointment_types</td>
<td>Contains the information for every appointment type declared by academic staff.</td>
</tr>
<tr>
<td>appointment_rules</td>
<td>Contains the information for appointment rules that declared by academic staff.</td>
</tr>
<tr>
<td>intake_questions</td>
<td>Contains the information for every intake question declared by academic staff.</td>
</tr>
<tr>
<td>reminders_setting</td>
<td>Contains the information for every setting of reminder notice of academic staff.</td>
</tr>
</tbody>
</table>
### 4.4.3 Data Dictionary

Entity name: staff

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
<th>Data Type</th>
<th>PK/FK</th>
<th>Nulls</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Unique identification for academic staff</td>
<td>Integer</td>
<td>PK</td>
<td>N</td>
</tr>
<tr>
<td>username</td>
<td>Username for the academic staff’s account access</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>password</td>
<td>Password for the academic staff’s account access</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of academic staff</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>working_id</td>
<td>Working id of academic staff in UTAR</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>email</td>
<td>Email address of academic staff</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>office</td>
<td>Office phone number of academic staff In UTAR</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>mobile</td>
<td>Mobile phone number of academic staff</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>campus_location</td>
<td>Current working location of academic staff</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>faculty</td>
<td>Faculty of academic staff is working in</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>department</td>
<td>Department of academic staff is working in</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>designation</td>
<td>Current designation/ permanent position</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>admin_post</td>
<td>Current administrative position</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>qualification</td>
<td>Qualification in study experience</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>area_expertise</td>
<td>Subject area that academic staff knows a lot about</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>home_page_url</td>
<td>Biography/CV of academic staff</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Data Type</td>
<td>PK/FK</td>
<td>Nulls</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>profile_pic</td>
<td>Location of profile picture stored in the system</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>google_access_token</td>
<td>Google access token for accessing google api such as google calendar</td>
<td>Text</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>google_refresh_token</td>
<td>Google refresh token for revoking the access token which expired</td>
<td>Text</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>has_requester</td>
<td>Flag to check whether academic staff has or does not have requester account</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>ftl_status</td>
<td>Flag for first time login guideline, used to track whether academic staff has or has not finish the guideline.</td>
<td>Varchar</td>
<td></td>
<td>N</td>
</tr>
<tr>
<td>status</td>
<td>Flag to determine the account status: 0 for terminated, 1 for activated</td>
<td>Integer</td>
<td></td>
<td>N</td>
</tr>
</tbody>
</table>

Entity name: requesters

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
<th>Data Type</th>
<th>PK/FK</th>
<th>Nulls</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Unique identification for requester</td>
<td>Integer</td>
<td>PK</td>
<td>N</td>
</tr>
<tr>
<td>username</td>
<td>Username for requester’s account access</td>
<td>Varchar</td>
<td></td>
<td>N</td>
</tr>
<tr>
<td>password</td>
<td>Password for requester’s account access</td>
<td>Varchar</td>
<td></td>
<td>N</td>
</tr>
<tr>
<td>name</td>
<td>Name of the requester</td>
<td>Varchar</td>
<td></td>
<td>N</td>
</tr>
<tr>
<td>email</td>
<td>Email address of the requester</td>
<td>Varchar</td>
<td></td>
<td>N</td>
</tr>
<tr>
<td>mobile</td>
<td>Mobile phone number of the requester</td>
<td>Varchar</td>
<td></td>
<td>N</td>
</tr>
<tr>
<td>type</td>
<td>Type of requester: Student, Academic staff, Non-academic staff and Visitor</td>
<td>Varchar</td>
<td></td>
<td>N</td>
</tr>
<tr>
<td>student_id</td>
<td>Student Id of a student</td>
<td>Varchar</td>
<td></td>
<td>N</td>
</tr>
<tr>
<td>student_current_sem</td>
<td>Current study semester of the student</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>Table Name</td>
<td>Description</td>
<td>Data Type</td>
<td>Foreign Key</td>
<td>Required</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------</td>
<td>-------------</td>
<td>----------</td>
</tr>
<tr>
<td>student_department</td>
<td>Department of which the student is studying</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>student_faculty</td>
<td>Faculty of which the student is studying</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>student_course</td>
<td>Course of which the student is studying</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>as_id</td>
<td>Academic staff Unique Identification Number</td>
<td>Integer</td>
<td>FK</td>
<td>Y</td>
</tr>
<tr>
<td>as_working_id</td>
<td>Working Id of an academic staff</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>as_campus_loc</td>
<td>Working location of academic staff in UTAR</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>as_department</td>
<td>Academic staff’s working department</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>as_faculty</td>
<td>Academic staff’s working faculty</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>as_designation</td>
<td>Academic staff’s designation/permanent position</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>as_admin_post</td>
<td>Academic staff’s administrative position</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>non_as_working_id</td>
<td>Working Id of a non-academic staff</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>non_as_campus_loc</td>
<td>Current working location of non-academic staff in UTAR</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>non_as_department</td>
<td>Non-academic staff’s working department</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>non_as_faculty</td>
<td>Non-academic staff’s working faculty</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>non_as_designation</td>
<td>Non-academic staff’s designation/permanent position</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>non_as_admin_post</td>
<td>Non-academic staff’s administrative position</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>company_name</td>
<td>Visitor’s company name</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>company_tel</td>
<td>Visitor’s company telephone number</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>company_address</td>
<td>Visitor’s company address</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Data Type</td>
<td>PK/FK</td>
<td>Nulls</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------</td>
<td>------------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>company_designation</td>
<td>Visitor’s designation in his/her company</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>profile_pic</td>
<td>Location of requester’s profile picture stored in the system</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>Flag to determine the account status: 0 for terminated, 1 for activated</td>
<td>Integer</td>
<td>N</td>
<td></td>
</tr>
</tbody>
</table>

**Entity name: admins**

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
<th>Data Type</th>
<th>PK/FK</th>
<th>Nulls</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Unique identification for administrator</td>
<td>Integer</td>
<td>PK</td>
<td>N</td>
</tr>
<tr>
<td>username</td>
<td>Username for the administrator’s account access</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>password</td>
<td>Password for administrator’s account access</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of administrator</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>working_id</td>
<td>Working Id of administrator in UTAR</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>email</td>
<td>Email address of administrator</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>Flag to determine the account status: 0 for terminated, 1 for activated</td>
<td>Integer</td>
<td>N</td>
<td></td>
</tr>
</tbody>
</table>

**Entity name: appointments**

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
<th>Data Type</th>
<th>PK/FK</th>
<th>Nulls</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Unique identification for appointment</td>
<td>Integer</td>
<td>PK</td>
<td>N</td>
</tr>
<tr>
<td>old_date</td>
<td>Previous booked date for appointment</td>
<td>Date</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>old_start_time</td>
<td>Previous booked starting time for appointment</td>
<td>Time</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Column</td>
<td>Description</td>
<td>Data Type</td>
<td>Nullable</td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-----------------</td>
<td>----------</td>
<td></td>
</tr>
<tr>
<td>old_end_time</td>
<td>Previous booked ending time for appointment</td>
<td>Time</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>new_date</td>
<td>Current booked date for appointment</td>
<td>Date</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>new_start_time</td>
<td>Current booked starting time for appointment</td>
<td>Time</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>new_end_time</td>
<td>Current booked ending time for appointment</td>
<td>Time</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>Type of appointment</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>intake_question_answers</td>
<td>Questions along with answers, separated with delimiters</td>
<td>Text</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>staff_id</td>
<td>Unique identification number of academic staff</td>
<td>Integer</td>
<td>FK</td>
<td></td>
</tr>
<tr>
<td>requester_id</td>
<td>Unique identification number of requester</td>
<td>Integer</td>
<td>FK</td>
<td></td>
</tr>
<tr>
<td>mobile</td>
<td>Mobile phone number of requester</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>email</td>
<td>Email address of requester</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>Flag to determine the appointment status: Pending, Completed, Cancelled</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>attendance</td>
<td>Flag to determine the attendance of requester: Present, Absent</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>punctuality</td>
<td>Flag to determine the punctuality of requester: Punctual, Late</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>cancel_by</td>
<td>Name of the person who cancel the appointment</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>cancel_reason</td>
<td>Reason for appointment cancellation</td>
<td>Text</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>from_google</td>
<td>Flag to determine the whether the appointment is made in google calendar</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>staff_google_event_id</td>
<td>Google Event Id for particular appointment that has record in academic staff’s google calendar</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
<td>Data Type</td>
<td>PK/FK</td>
<td>Nulls</td>
</tr>
<tr>
<td>------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>staff_google_event_desc</td>
<td>Google event description for particular appointment that has record in academic staff’s google calendar</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>requester_google_event_id</td>
<td>Google Event Id for particular appointment that has record in requester’s google calendar</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
<tr>
<td>requester_google_event_desc</td>
<td>Google event description for particular appointment that has record in requester’s google calendar</td>
<td>Varchar</td>
<td></td>
<td>Y</td>
</tr>
</tbody>
</table>

Entity name: availabilities

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
<th>Data Type</th>
<th>PK/FK</th>
<th>Nulls</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Unique identification for availability</td>
<td>Integer</td>
<td>PK</td>
<td>N</td>
</tr>
<tr>
<td>staff_id</td>
<td>Unique identification number of academic staff</td>
<td>Integer</td>
<td>FK</td>
<td>N</td>
</tr>
<tr>
<td>start</td>
<td>Starting Date Time of an availability</td>
<td>DateTime</td>
<td></td>
<td>N</td>
</tr>
<tr>
<td>end</td>
<td>Ending Date Time of an availability</td>
<td>DateTime</td>
<td></td>
<td>N</td>
</tr>
</tbody>
</table>

Entity name: appointment_types

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
<th>Data Type</th>
<th>PK/FK</th>
<th>Nulls</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Unique identification for appointment type</td>
<td>Integer</td>
<td>PK</td>
<td>N</td>
</tr>
<tr>
<td>staff_id</td>
<td>Unique identification number of academic staff</td>
<td>Integer</td>
<td>FK</td>
<td>N</td>
</tr>
<tr>
<td>name</td>
<td>Value for appointment type</td>
<td>Varchar</td>
<td></td>
<td>N</td>
</tr>
</tbody>
</table>
Entity name: appointment_rules

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
<th>Data Type</th>
<th>PK/FK</th>
<th>Nulls</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Unique identification for appointment rule</td>
<td>Integer</td>
<td>PK</td>
<td>N</td>
</tr>
<tr>
<td>staff_id</td>
<td>Unique identification number of academic staff</td>
<td>Integer</td>
<td>FK</td>
<td>N</td>
</tr>
<tr>
<td>ab_unit_time</td>
<td>Unit of time for rule that limit appointment booking</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>ab_time</td>
<td>Exact time in number for rule that limit appointment booking</td>
<td>Integer</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>pa_unit_time</td>
<td>Unit of time for rule that limit postpone of appointment</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>pa_time</td>
<td>Exact time in number for rule that limit postpone of appointment</td>
<td>Integer</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>ca_unit_time</td>
<td>Unit of time for rule that limit cancellation of appointment</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>ca_time</td>
<td>Exact time in number for rule that limit cancellation of appointment</td>
<td>Integer</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>allow_prior</td>
<td>Flag to determine the academic staff allows or not the prioritization in appointment booking</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>prior_1</td>
<td>The type of requester who has the highest priority</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>prior_2</td>
<td>The type of requester who has the 2nd rank of priority</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>prior_3</td>
<td>The type of requester who has the 3rd rank priority</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>prior_4</td>
<td>The type of requester who has the lowest priority</td>
<td>Varchar</td>
<td>Y</td>
<td></td>
</tr>
</tbody>
</table>
Entity name: intake_questions

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
<th>Data Type</th>
<th>PK/FK</th>
<th>Nulls</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Unique identification for each intake question</td>
<td>Integer</td>
<td>PK</td>
<td>N</td>
</tr>
<tr>
<td>staff_id</td>
<td>Unique identification number of academic staff</td>
<td>Integer</td>
<td>FK</td>
<td>N</td>
</tr>
<tr>
<td>question</td>
<td>The question that intend to be asked</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>answer_format</td>
<td>The answer format for the question</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
</tbody>
</table>

Entity name: reminders_setting

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Description</th>
<th>Data Type</th>
<th>PK/FK</th>
<th>Nulls</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Unique identification for reminder setting</td>
<td>Integer</td>
<td>PK</td>
<td>N</td>
</tr>
<tr>
<td>staff_id</td>
<td>Unique identification number of academic staff</td>
<td>Integer</td>
<td>FK</td>
<td>N</td>
</tr>
<tr>
<td>approval_notice</td>
<td>Flag to determine the academic staff whether using default notice or</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>customized notice for approval of appointment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>approval_content</td>
<td>Customized content for approval notice in html</td>
<td>Text</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>postpone_notice</td>
<td>Flag to determine the academic staff whether using default notice or</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>customized notice for postpone of appointment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>postpone_content</td>
<td>Customized content for postpone notice in html</td>
<td>Text</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>cancellation_notice</td>
<td>Flag to determine the academic staff whether using default notice or</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td></td>
<td>customized notice for cancellation of appointment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cancellation_content</td>
<td>Customized content for cancellation notice in html</td>
<td>Text</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------</td>
<td>------</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>priority_cancellation_notice</td>
<td>Flag to determine the academic staff whether using default notice or customized notice for priority cancellation of appointment</td>
<td>Varchar</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>priority_cancellation_content</td>
<td>Customized content for priority cancellation notice in html</td>
<td>Text</td>
<td>Y</td>
<td></td>
</tr>
</tbody>
</table>
4.5 User Interface Design
In this system, there are 3 different types of users such as academic staff, requester and administrator. They will be having 3 different types of interface design except login page of the system.

4.5.1 Login Page UI
The figure below shows the login page for this system. It requires the input of username and password for authentication.

![Login Page UI Design](image)

Figure 4.29 UI Design for Login Page

4.5.2 User Interfaces for Academic Staff

4.5.2.1 Side Navigation Bar
The figure below shows the side navigation bar in academic staff’s page.

![Side Navigation Bar Design](image)

Figure 4.30 UI Design for side navigation bar (Academic Staff)
4.5.2.2 Appointment Management UI

The figure below shows the view under navigation link named as “Appointment Calendar”. This UI is used for appointment booking, along with the features of appointment details viewing, appointment postpone and appointment cancellation.

Figure 4.31 UI Design for Appointment Calendar (Academic Staff)

The figures below show the all the appointments in list under navigation link named as “Appointment Log”. These UI are used for the viewing of appointment details and validate appointment after clicking on the edit button.

Figure 4.32 UI Design for Appointment List (Academic Staff)

Figure 4.33 UI Design for Validation of Appointment (Academic Staff)
4.5.2.3 Availability Management UI

The figure below shows the view for availability management. Academic staff will require to add in their availabilities in the box given for each day.

Figure 4.34 UI Design for Availability Management (Academic Staff)

4.5.2.4 Appointment Types Management UI

The figures below show the view for the management of appointment types under the navigation link named “Appointment Types”. The design allows academic staff to view existing appointment type list, along with create new and edit existing appointment type.

Figure 4.35 UI Design for Appointment Type Listing (Academic Staff)

Figure 4.36 UI Design for Create New Appointment Type (Academic Staff)
4.5.2.5 Appointment Rules Management UI

The figures below show the view for the management of appointment rules under the navigation link named “Appointment Rules”. The design allows academic staff to view the setting of appointment rules, along with the update feature.

4.5.2.6 Intake Form Questions Management UI

The figures below show the view for the management of intake questions under the navigation link named “Intake Form Questions”. The design allows academic staff to view existing intake questions, along with create new and edit existing questions.
4.5.2.7 Google Calendar Synchronization Management UI

The figures below show the view for the management of google calendar synchronization under the navigation link named “Synchronization of Google Calendar”. The design allows academic staff to connect or disconnect google account from the system.

Figure 4.40 UI Design for Create/ Update Form of Intake Questions (Academic Staff)

Figure 4.41 UI Design for Google Calendar Synchronization (Academic Staff)

Figure 4.42 UI Design for Google Calendar Desynchronization (Academic Staff)
4.5.2.8 Reminder Notice Management UI

The figures below show the view for the management of reminder notice under the navigation link named “Email Setting”. The view allows academic staff to configure the notice content that will be sent to both parties: academic staff and requester. The notice content can be either stick to the default which designed in the system or customized by the academic staff himself/herself.

![UI Design for Reminder Notice Management](image)

Figure 4.43 UI Design for Reminder Notice Management (Academic Staff)

4.5.2.9 Report Generation UI

The figure below shows the user interface design for report generation under the navigation link named “Reports”. This UI is used to generate report based on the report type and selected time range.

![UI Design for Report Generation](image)

Figure 4.44 UI Design for Report Generation (Academic Staff)
4.5.2.10 Profile Management UI

The figures below shows the user interface design for feature of viewing and editing user profile.

![Profile Management UI](image)

Figure 4.45 UI Design for viewing of user profile (Academic Staff)

![Profile Management UI](image)

Figure 4.46 UI Design for editing of user profile (Academic Staff)

4.5.2.11 Change Password UI

The figures below shows the user interface design for feature of change password.

![Change Password UI](image)

Figure 4.47 UI Design for Change Password (Academic Staff)
4.5.3 User Interfaces Design for Requester

4.5.3.1 Appointment Management UI

The figures below show the view for appointment booking. These UI is used for appointment booking. The first view will be the staff directory which allows requester to search for the targeted staff, along with the appointment booking form.

![UI Design for Staff Directory (Requester)](image1)

![UI Design for Booking Form (Requester)](image2)
The figure below shows the views for appointment viewing, postpone and cancellation.

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Appointment Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Aug 15</td>
<td>3:30PM - 3:30PM</td>
</tr>
<tr>
<td>2</td>
<td>Aug 15</td>
<td>3:30PM - 4:30PM</td>
</tr>
<tr>
<td>3</td>
<td>Aug 15</td>
<td>10:30AM - 10:50AM</td>
</tr>
</tbody>
</table>

Figure 4.50 UI Design for Appointment Listing (Requester)

![Appointment Details](image)

Figure 4.51 UI Design for Appointment Details (Requester)

### 4.5.3.2 Profile Management UI

The figures below shows the user interface design for feature of viewing and editing user profile.

![Profile](image)

Figure 4.52 UI Design for viewing of user profile (Requester)
4.5.3.3 Change Password UI

The figures below shows the user interface design for feature of change password.

4.5.3.4 Account Registration UI

The figure below shows the user interface design for feature of account registration. There will be several input fields to obtain the required information for requester account.
4.5.4  **User Interfaces for Administrator**

4.5.4.1  **Account Management UI**

The figures below show the view of account creation and account listing. These UI will allow administrator to perform their responsibility to create account for users and managing the activation/deactivation of account.

![Account Creation (Academic Staff)](image)

**Figure 4.56 UI Design for Account Creation (Administrator)**

![Faculty](image)

**Figure 4.57 UI Design for Account Status Management (Administrator)**

4.6  **Architecture Pattern**

Laravel application is selected as the framework of this project. The framework follows the model-view-controller (MVC), a popular architecture pattern for developing web applications as it is made up of the following three parts: Model, View and Controller. Figure 4.58 shows the MVC components of Laravel framework.
The following table explains the responsibilities and displays the collaborators for each class in MVC architecture pattern:

Table 4.2: Class-Responsibility-Collaborator Card for MVC architecture pattern

<table>
<thead>
<tr>
<th>Class</th>
<th>Responsibilities</th>
<th>Collaborators</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>• Provides core data and functionality.</td>
<td>View, Controller</td>
</tr>
<tr>
<td></td>
<td>• Informs the components about data modification.</td>
<td></td>
</tr>
<tr>
<td>View</td>
<td>• Displays data to user.</td>
<td>Model, Controller</td>
</tr>
<tr>
<td></td>
<td>• Retrieves data from the model.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Implementation of data update procedure.</td>
<td></td>
</tr>
<tr>
<td>Controller</td>
<td>• Accepts user handling</td>
<td>Model, View</td>
</tr>
<tr>
<td></td>
<td>• Translate user events</td>
<td></td>
</tr>
</tbody>
</table>
5.1 Implementation of Modules

In this system, there are 3 types of users such as academic staff, requester and administrator. The users have their respective modules for accessing different functionality provided. The list of modules for each user is shown as below:

<table>
<thead>
<tr>
<th>User</th>
<th>Modules</th>
</tr>
</thead>
</table>
| Academic Staff    | Appointment Management  
|                   | Availability Management  
|                   | First Time Login Guideline  
|                   | Google Calendar Synchronization Management  
|                   | Appointment Rules Management  
|                   | Appointment Types Management  
|                   | Appointment Reminder Management  
|                   | Intake Form Management  
|                   | Reports Generation  
|                   | Profile Management  |
| Requester         | Appointment Management  
|                   | Profile Management  
|                   | Account Registration  |
| Administrator     | Account Management  |

Besides the modules for each user, the system also provides a module called prioritization module, automated reminder module and daily recap module.

5.1.1 Modules for Academic Staff

5.1.1.1 Appointment Management Module

This module is defined as the process of dealing with appointments. For an academic staff, it consists of 5 sub-modules such as Appointment Making, View Appointments, Appointment Bring Forward/Postpone, Appointment Cancellation and Appointment Validation.
5.1.1.1 Appointment Making Module

In this system, academic staff is not restricted only for managing appointment requests while this user is also allows to register himself/herself as a requester directly using the given academic staff’s account in order to make appointment request to other academic staff. The registration for requester will only require to be completed once at the first stage of appointment making process.

In order to perform the appointment making process, academic staff requires to direct himself/herself to a view by clicking the navigation link which named as “Appointment Booking” in the side bar shown in Figure 5.1. The link will trigger a backend method to check and verify academic staff has its own requester account. The checking will be done on a flag recorded in his/her account.

If the account does not contain any requester information on the flag, the backend method will redirect him/her to a registration view. To simplify the registration process, the system has been computed to register the requester account itself by using the existing information after the register button is clicked.
After the registration is completed or the verification of requester flag in user’s account is completed, he/she will be redirected again to a staff directory shown in Figure 5.4. The filtration of all academic staff can be done by inserting the targeted academic staff’s name, faculty and department.

![Figure 5.4 Staff directory view](image)

After the filtration, by clicking the box where the details of targeted academic staff is shown, the academic staff who is acting as requester will be directed to a booking form which requires he/she to fill the appointment information such as the appointment date with start time and end time, the appointment type, the answers for any intake questions, the contact number and email address.

![Figure 5.5 Staff directory with filtered academic staff](image)

In the first tab of the booking form, the academic staff who is acting as requester requires to choose a suitable date among the dates in light blue color which indicates
the free time of targeted academic staff for an appointment. When a date is clicked, the backend mechanism will once be trigger to obtain the free time slot in that particular day after eliminate the other time slot(s) that has been filled by someone else and verify that the free time slot(s) fulfills the appointment rules.

![First tab of booking form](image1)

**Figure 5.6 First tab of booking form**

```
public function GetTimes() {
    $selectedDay = Input::get('selectedDay');
    $staff = Staff::getStaffDetails(Input::get('id'));
    if($staff['requester_id'] != null){
        $AppointmentTimes = Appointment::getAppointmentTimes($selectedDay, Input::get('id'), $staff['requester_id']);
    }else{
        $AppointmentTimes = Appointment::getAppointmentTimes($selectedDay, Input::get('id'), null);
    } $availableTimes = Availability::getAvailableTimes(Input::get('id'));
    $rules = AppointmentRule::getAppointmentRules(Input::get('id'));
    $count = 0;
    $newAvailability = null;
    if($availableTimes != null){
        foreach($availableTimes as $i => $value) {
            $startTime = new DateTime($value->start);
            if (strtotime($selectedDay) != $startTime) {
                unset($availableTimes[$i]);
            }else{
                $endTime = new DateTime($value->end);
                $intervalTime = new DateTime($endTime->format('Y-m-d H:i:s'));
                
```

![Section of code for obtaining free time slot(s)](image2)

**Figure 5.7 Section of code for obtaining free time slot(s)**

In the second tab, the academic staff who act as the requester will requires to select the appointment type for that particular appointment and answer the intake questions from the targeted academic staff. If there is any intake question, it can be either be answered or left empty.
In the third tab, the information of the academic staff who act as the requester is shown. The academic staff who act as the requester may choose to keep or change the contact number and email address for future follow-ups of appointment.

The validation of data inputs will then be triggered after the book button is clicked. If the data inserted is fulfilled with the requirement, the appointment will be made and confirmation notice will then be sent to both parties. Moreover, the system creates
new google event for both parties if they have synchronize their timetable with google calendar. If the data inserted is failed to fulfil the requirement, he/she will be redirected back to the booking form which has error message shown around the wrong input field.

```java
public static void addAppointment() {
    spot = spot + 1;
    replacement = false;
    data = array;
    'date' => spot['date'],
    'start_time' => spot['start_time'],
    'end_time' => spot['end_time'],
    'old_appointment_id' => spot['old_appointment_id'],
    'new_start_time' => spot['new_start_time'],
    'new_end_time' => spot['new_end_time'],
    'requester_id' => spot['requester_id'],
    'staff_id' => spot['staff_id'],
    'mobile' => spot['mobile'],
    'email' => spot['email'],
    'type' => spot['type'],
    'total_questions' => spot['total_questions']
};

if (spot.start_time != '' && spot.end_time != '' && spot.old_appointment_id != '' && spot.new_start_time != '' && spot.new_end_time != '')
        {array = array;
        'date' => 'required',
        'start_time' => 'required',
        'end_time' => 'required',
        'mobile' => 'required',
        'email' => 'required',
        'type' => 'required'}
```

Figure 5.10 Section of code for adding appointment

After the appointment is successfully made, the academic staff will be redirected to a view that displays appointment details

![Appointment Details](image)

Figure 5.11 Appointment details view
5.1.1.1.2 View Appointments Module

In this system, academic staff is provided with two different types of view either in list or in calendar. Both of the views can be accessed through the navigation link under a dropdown list which named as “Appointment [Requested]”.

![Side bar](image)

Figure 5.12 Side bar

By clicking the navigation link which named as “Appointments [Calendar View]”, it will display appointments in calendar. After the link is clicked, a backend mechanism will be performed to extract only appointments without the status of being cancelled.

```php
public function GetAllAppointments()
{
    $staff = Staff::getStaffDetails(Auth::id());
    if($staff['requester_id'] !== null)
        $appointments = Appointment::getStaffAppointmentsWithoutCancelled($staff['id'], $staff['requester_id']);
    else
        $appointments = Appointment::getStaffAppointmentsWithoutCancelled($staff['id'], $staff['requester_id']);

    $calendarAppointments = array();
    foreach($appointments as $app) {
        if($app['status'] == "Pending"){
            $title = '';
            if($app['from_google'] == 'Yes')
                $title = 'Appt from Google';
            else{
                if($app['id'] != $app['staff_id']){
                    $target_staff = Staff::getStaffDetails($app['staff_id']);
                    $title = 'Appt with ' . $target_staff['name'];
                }else{

```
As there would be many appointments available in the calendar, the academic staff can click on either one of it and the details of that particular appointment will be displayed on the right side of the screen.

![Appointment calendar view](image)

Figure 5.14 Appointment calendar view

While clicking on the navigation link which named as “Appointments [List View]”, it will display appointments in a list. After the link is clicked, a backend mechanism will be performed to extract all appointments including appointments with the status of being cancelled.

![Appointment list view](image)

Figure 5.15 Appointment list view

As there would be many appointments available in the list, the academic staff can click on either one of it and the details of that particular appointment will be displayed in a new view.
5.1.1.3 Appointment Bring Forward/ Postpone Module

In order to perform the changing of date and time either bring forward or postpone, the academic staff is required to perform the viewing of appointments in either calendar or list view.

In calendar view, he/she requires to click on the appointment that he/she attempts to change the date and time. The details of the appointment will be displayed on the right side of the application shown in Figure 5.14 along with two buttons named as “bring forward/postpone” and “cancel”.

In list view, he/she requires to click on the row (appointment) of the table to direct himself/herself to view the details of appointment as shown in Figure 5.16 which he/she wish to change the date and time. The appointment details view comes with two buttons named as “change date & time” and “cancel” as shown in Figure 5.16.

A modal with a calendar for selection of new date, start time and end time is shown after the “bring forward/postpone” or “change date & time” button is clicked.
The date, start time and end time are obtained through backend mechanism similar to which computed during appointment making process. After the yes button is clicked, the changing of appointment date and time will be confirmed and follow-ups notice will be sent to both parties, along with the update of google calendar event for the person who has synchronize google calendar with the time schedule in this system.

5.1.1.1.4 Appointment Cancellation Module

In this system, academic staff is allowed to cancel any appointment requested by someone else or the appointment requested to someone else. In order to cancel appointment, academic staff requires to perform the viewing of appointments in either calendar or list view.

In calendar view, he/she requires to click on the appointment that he/she attempts to change the date and time. The details of the appointment will be displayed on the right side of the application shown in Figure 5.14 along with two buttons named as “bring forward/postpone” and “cancel”.

Figure 5.17 Modal for new selection date, start time and end time
In list view, he/she requires to click on the row (appointment) of the table to direct himself/herself to view the details of appointment as shown in Figure 5.16 which he/she wish to change the date and time. The appointment details view comes with two buttons named as “change date & time” and “cancel” as shown in Figure 5.16.

A modal with an input field for cancellation reason will be shown after the “cancel” button is clicked.

![Cancel Appointment](image)

Figure 5.18 Model for input of cancellation reason

The appointment will then be cancelled and cancellation notice will be sent to both parties, along with the deletion of google event for the person who has synchronize google calendar with the time schedule in this system.

5.1.1.1.5 Appointment Validation Module

In this system, the academic staff is requires to validate the appointment period which has passed for record purpose. The validation can only be done after performing the viewing of appointments in either calendar or list view. A backend mechanism will be performed every time viewing is being performed to check whether the appointment has reach the appointment validation period.

![Section of code](image)

Figure 5.19 Section of codes for verifying appointment validation period
In calendar view, the buttons for “bring forward/postpone” and “cancel” will be then changed to a button named as “validate”. The same goes to appointment details view.

![Appointment Details](image)

Figure 5.20 Appointment details view with validation button

A modal with an input field for attendance and punctuality of requester will be prompt after the “validate” button is clicked. The record is then saved to the database of this system for future reference or report generation.

5.1.1.2 Availability Management Module

For appointment making process, the most important aspect is that an academic staff requires to have free time slot(s), in other words, the academic staff requires to be available. Therefore, every academic staff requires to declare their availabilities in this system by adding in, and later update or even delete their own availabilities.

To manage the availabilities, academic staff requires to click on the navigation link which named as “Availability” in the side bar shown in Figure 5.1. The link will direct the academic staff to availabilities view as shown in Figure 5.21.
In this view, there are three types of management methods such as add, update and delete. The way to carry out the management methods is very simple as only requires clicking, dragging and dropping.

In order to add a new availability time period, academic staff is only requires to start dragging at any period and stop at the preferred ending period. A modal will then pop out and prompt academic staff to ensure whether the availability should be inserted. The availability will be added after he/she is confirmed by clicking “yes” button.

---

**Figure 5.21 Availabilities view**

**Figure 5.22 Modal for confirming to add new availability**
Whereas, in order to update an availability, academic staff is required to click, hold and drag the existing availability to a new suitable time period or click at the end of the existing availability and drag to a new suitable ending period. A modal will then pop out and prompt academic staff to ensure whether the existing availability should be updated to the new time period. The availability will be changed after academic staff is confirmed by clicking “yes” button.

![Modal for confirming the update of availability](image)

**Figure 5.23** Modal for confirming the update of availability

As for delete an availability, academic staff is required only to click on the existing availability. A modal will then pop out and prompt academic staff to ensure whether the existing availability should be deleted.

![Modal for confirming the deletion of availability](image)

**Figure 5.24** Modal for confirming the deletion of availability
Besides that, the backend of the system will automatically obtain the latest availabilities of the academic staff after any of the management of the methods has been performed and shows them in the calendar.

```javascript
function confirmAvailability()
    eventData = {start: startDate, end: endDate};
$.ajax({
    type: "POST",
    url: url + '/api/set-availability',
    data: eventData,
    success: function(data) {
        reset();
        swal({title: "Sweet!", text: "New availability is added.", type: "success"});
    },
    error: function(data) {
        errorMsg();
        reset();
    },
    dataType: "json",
});
```

Figure 5.25 Section of codes for obtaining latest availabilities

### 5.1.1.3 First Time Login Guideline Module

In this system, it provides first time login guideline for every academic staff to insert necessary information for appointment management and indirectly increase their understanding on the system. The guideline will be shown when academic staff login to its respective account for the very first time. There will have 3 steps before completion and a flag which determine the step that academic staff has reached.

This first step will be the tab which configure the appointment types for appointment making process. This process is important which it prevent academic staff has forgotten to create appointment type which will cause no appointment can be made successfully as there would be error occurs. The next button is disabled to prevent academic staff to proceed to another step before he/she adds in at least one appointment types.
The second step will be the tab which configure the appointment rules for appointment management process. This process is important to allow academic staff to set limit to each appointment making such as prevent requester to make, bring forward/postpone or cancel appointment before 1 day of the appointment is taking place. Academic staff can select to enable the rules or skip the step. After either skip or next step button is clicked, the flag will be updated to indicate the staff has finished this step.

```php
public function UpdateFTLStatus(){
    $staff = Staff::getStaffDetails(Auth::id());
    $success = Staff::updateFTLStatus($staff);
    if(!$success){
        return redirect('staff/dashboard');
    }else{
        return response()->json('error', 500);
    }
}
```
The last step of the guideline will be the explanation of other features available for academic staff. The staff can proceed to use the full functionality after click on the proceed button. He/she will be directed to the dashboard of this system.

Figure 5.29 Third tab of first time login guideline

5.1.1.4 Google Calendar Synchronization Management Module

In this system, it provides the functionality of synchronize google calendar with the time schedule of the academic staff. Beforehand, academic staff requires to direct himself/herself to view of setting for google synchronization by click the navigation link which named as “Synchronization” in side bar.

Figure 5.30 Setting dropdown in side bar
A backend mechanism will be triggered to check whether google synchronization has or has not been carried out. If the google synchronization has not been carried out, a connect button will be shown to bring academic staff to a consent screen to login their google account.

![Google Calendar Sync](image)

**Figure 5.31 Synchronization enable page**

If the connection is completed, a disconnect button will be shown to allow he/she to disconnect the synchronization between google calendar and time schedule in this system.

![Google Calendar Sync](image)

**Figure 5.33 Synchronization disable page**

### 5.1.1.5 Appointment Rules Management Module

In this system, appointment rules is created to prevent the appointment to be made, brought forward/postponed or cancelled by very last minute. Academic staff is allowed to enable appointment rules or disable the rules based on his/her preferences.
The appointment rules can be enabled/disabled at the second step in first time login guideline or access by clicking the navigation link which named as “Appointment Rules” shown in Figure 5.30. A backend mechanism will be trigger to check whether the staff has or has not enable the appointment rules in his/her account.

If the appointment rules has not been enabled. He/she will be directed to a page with enable button.

![Appointment rules enable page](image)

Figure 5.34 Appointment rules enable page

After the enable button is clicked, the rules will be created with limit of all appointments must be made, brought forward/postponed or cancelled by at least one day before it takes place while the prioritization still remained disabled.

```php
public static function enableAppointmentRules($staff_id){
    $rule = new AppointmentRule;
    $rule->staff_id = $staff_id;
    $saved = $rule->save();
    if($saved){ return true; } 
    else{ return false; }
}
```

Figure 5.35 Section of codes for enabling appointment rules

![Appointment rules update and disable page](image)

Figure 5.36 Appointment rules update and disable page
The rules for the time limit of appointment making, bring forward/postpone or cancellation can be configured for unit time or time period. If academic staff is decided to disable the appointment rules, he/she can just directly click on the disable button.

5.1.1.6 Appointment Types Management Module

In this system, appointment types are important to differentiate the purpose for each appointment. It is defined by academic staff under the view of setting for appointment types which can be accessed through clicking the navigation link named as “Appointment Types” shown in Figure 5.30. In this module, academic staff can choose to add new, update existing or delete existing appointment type.

The adding of new appointment type can be easily done by click add button after the input is filled.

![Add form for appointment type](image)

Figure 5.37 Add form for appointment type

The update of existing appointment type can be easily done by clicking the update of the targeted type. An update form will then be shown by hiding the add form. The update of existing type by input will be completed after click on the update button in update form.

![Update form for appointment type](image)

Figure 5.38 Update form for appointment type
The delete of existing appointment type can be easily done by clicking on the delete button of the targeted type. A prompt will be pop out to confirm the deletion of the existing appointment type. Deletion will be carried out after the prompt is confirmed.

![Prompt for deletion of appointment type](image)

**Figure 5.39 Prompt for deletion of appointment type**

5.1.1.7 Reminder Notice Management Module

In this system, it has automated follow-ups on each making, bring forward/postponing and cancelling of appointment by sending mails to both parties, the academic staff and the requester. There will be a default format for every types of reminders but academic staff is allowed to customize its own content for the notice by following the guideline provided.

The customization page can be assessed by clicking on the navigation link which named as “Reminder Notices” under setting as shown in Figure 5.30. Academic staff will be directed to the view of reminder notices setting as shown in Figure 5.40.

![Default reminder notices setting view](image)

**Figure 5.40 Default reminder notices setting view**
Academic staff can select to keep the default setting for the notice or click customize to customize his/her own notice.

![Customization of reminder notices setting view](image)

Figure 5.41 Customization of reminder notices setting view

The customization of notice can follow the guideline in a modal to get the information about the appointment. The modal will be displayed by clicking the question mark symbol at the top-right of tab bar.

![Modal for guideline in getting appointment data](image)

Figure 5.42 Modal for guideline in getting appointment data

Any changes in the reminder notices setting will be saved by just clicking on the save button.
5.1.1.8 Intake Form Management Module

In this system, academic staff is provided with the feature of creating questions to be asked during appointment making process of any requester. The intake questions are set to be optional in case of the requester has no answer for any of the questions.

In this module, it allows the academic staff to add new, update existing or delete existing intake questions that will be shown in intake form (tab 2 of booking form). To access this functionality, academic staff is required to click on the navigation link which named as “Intake Form” under setting as shown in Figure 5.30.

The adding of new intake question is quite simple as academic staff is only required to click the add button after question is inserted and type is selected.
The update of the existing intake question is quite simple as academic staff is only required to click on the update button of targeted intake question. An update form will be displayed and prompt him/her the new question and new answer format for replacing the existing intake question.
The deletion of targeted intake question can be easily done by just click on the delete button. A prompt will be pop out to confirm the deletion of the question. Deletion will then be carried out if the prompt is confirmed.

![Prompt for deletion of intake question](image)

Figure 5.46 Prompt for deletion of intake question

5.1.1.9 Reports Generation Module

In this system, academic staff is allowed to generate report or in other words, filtered out the data according to the report type. The report generation feature can be accessed through the navigation link which named as “Reports” in side bar shown in Figure 5.1.

![Reports generation view](image)

Figure 5.47 Reports generation view

There are only 5 types of reports available in current system such as incomplete appointments, completed appointments, cancelled appointments, appointments based on booking period and appointments based on appointment type. The report can only be generated by adding the date range.
5.1.1.10 Profile Management

In this system, it provides features such as view and edit profile, along with the feature of change password. All of the features can be assessed through navigation links under a tree named “Profile” in side bar shown in Figure 5.49.

![Profile dropdown in side bar](image)

Academic staff is allowed to view his/her personal profile by clicking on the navigation link named as “My Profile”.

![Profile view](image)
Academic staff is allowed to edit his/her personal information in profile by clicking on the edit profile button.

![Edit Profile](image1)

Figure 5.51 Edit profile view

Academic staff is also allowed to change his/her account’s password by clicking the “Change Password” navigation link under profile tree as shown in Figure 5.49.

![Change Password](image2)

Figure 5.52 Change password view

5.1.2 Modules for Requester

5.1.2.1 Appointment Management Module

This module is defined as the process of dealing with appointments. For requester, it consists 4 sub-modules such as Appointment Making, View Appointments, Appointment Bring Forward/Postpone and Appointment Cancellation.

5.1.2.1.1 Appointment Making Module

In this system, a requester is defined as the person who makes a request for appointment from academic staff. In order to perform the appointment making process,
the requester is required to direct himself/herself to a staff directory by clicking navigation link which named as “Appointment Booking” in side bar shown in Figure 5.53.

![Figure 5.53 Requester’s side bar](image)

The steps after the link is clicked will be much the same as the steps in appointment making process of academic staff (Section 5.1.1.1.1).

After the link is clicked, the requester will be directed to staff directory which requires inputs from him/her. Requester will require to fill in the academic staff’s name, faculty and department or just directly click on search button. The filtered academic staff’s details will be displayed in boxes as shown in Figure 5.54.

![Figure 5.54 Staff Directory with filtered Academic Staff](image)
The requester will then need to click on the box where the details of targeted academic staff is displayed for accessing the booking form.

In the first tab of the booking form, the requester requires to choose a suitable date among the dates in light blue color which indicates the free time of targeted academic staff for an appointment. When a date is clicked, the backend mechanism will once be trigger to obtain the free time slot in that particular day after eliminate the other time slot(s) that has been filled by someone else and verify that the free time slot(s) fulfills the appointment rules.

![Booking Form](image)

**Figure 5.55 First tab of booking form**

![Section of codes for getting free time slot(s)](image)

**Figure 5.56 Section of codes for getting free time slot(s)**

In the second tab, the requester will requires to select the appointment type for that particular appointment and answer the intake questions from the targeted academic staff. The intake questions can be either be answered or left empty.
In the third tab, the information of the requester is shown. He/she may choose to keep or change the contact number and email address for future follow-ups of appointment.

The validation of data inputs will then be triggered after the book button is clicked. If the data inserted is fulfilled with the requirement, the appointment will be made and confirmation notice will then be sent to both parties. If the data inserted is failed to fulfil the requirement, he/she will be redirected back to the booking form which has error message(s) shown around the wrong input field.
After the adding process is completed, the requester will be redirected to a view that displays appointment details.

### 5.1.2.1.2 View Appointments Module

In this system, requester is allowed to view appointments in list. The view can be access through the navigation link named as “Appointment list” in side bar as shown in Figure 5.53. A backend mechanism is triggered to obtain all appointments and then display all of them in a list.

![Appointment list view](image)

If there is any appointment available in the list, the requester can click on either one of it and the details of the particular appointment will be displayed in a new view as shown in Figure 5.61.
5.1.2.1.3 Appointment Bring Forward/Postpone Module

In order to perform the changing of date and time, the requester is required to perform the viewing of appointments in list view and deep into appointment details.

In the appointment details view, there will be a button which named as “change date & time”. By clicking it, it will shows up a modal with a calendar for selection of new date, start time and end time.

The date, start time and end time are obtained through backend mechanism similar to which computed during appointment making process. After the yes button is clicked,
the changing of appointment date and time will be confirmed and follow-ups notice will be sent to both parties, along with the update of google calendar event for the person who has synchronize google calendar with the time schedule in this system.

5.1.2.1.4 Appointment Cancellation Module

In this system, the requester is allowed to cancel his/her appointment if the appointment date and time does not violate the time limit rules for appointment cancellation. In order to perform this feature, requester requires to direct himself/herself to the appointment details view of that particular appointment.

In the appointment details view, there has a cancel button for appointment cancellation as shown in Figure 5.61. A modal with an input for cancellation reason will be pop out after the cancel button is clicked.

![Figure 5.63 Modal for input of cancellation reason](image)

The appointment will then be cancelled after the confirmation of appointment cancellation has been done. Cancellation notice will be sent to both parties, along with the deletion of google event for the person who has synchronize google calendar with the time schedule in the system.

5.1.2.2 Profile Management Module

In this system the feature of profile management for requester is similar with what has given to academic staff. In this module, features such as view and edit profile, along with the feature of change password are included.
All of the features can be assessed through navigation links under a dropdown list which named as “Profile” in side bar shown in Figure 5.64.

![Figure 5.64 Profile dropdown in side bar](image)

Requester is able to view his/her personal profile by clicking on the “My Profile” navigation link. The link will trigger the backend mechanism to extract the requester’s information.

![Figure 5.65 Profile view](image)

In order to edit the profile, requester is required to click on the edit profile button that shown in profile view.

![Figure 5.66 Edit profile view](image)
Lastly, in this module, the requester is also allowed to change his/her account’s password. A backend mechanism will be triggered to validate the correctness of the current password and validate whether the new password fulfills the requirement rule.

![Change Password](image)

**Figure 5.67** Change password view

```php
public function UpdatePassword(){
    $data = array(
        'current_password' => $post['current_password'],
        'password' => $post['password'],
        'password_confirmation' => $post['password_confirmation'],
    );

    $rules = array(
        'password' => 'required|alpha_num|min:8|confirmed',
        'password_confirmation' => 'required|alpha_num|min:8',
    );

    $v = Validator::make($data, $rules);
    $check = true;

    if ($v->passes()) {
        if (Requester::setPassword($session->get('id'), $data['password'])){
            $info = 'The password is successfully changed.';
            return view('Requester/profile/change_password')->with(["requester" => Requester::getRequesterDetails($session->get('id')),
                'check' => false,
                'info' => $info,
                'data' => null
            ]);}
    }

7.2.3 Account Registration Module

In this system, the requester has different functionality compared to academic staff as he/she is allowed for account registration. If the requester does not have any account, he/she is allowed to carry out the account registration through clicking the link “Register a new membership” on the login page for guest.
The link will direct the requester to a registration page. Normal input such as username, password, name, email, phone number and requester type will be prompted in the page.

Registration will be successful if the page does not show any errors and is directed back to login page.

5.1.3 Modules for Administrator

5.1.3.1 Account Management Module
An administrator is needed for this system to manage on accounts for all types of users such as academic staff, requester and administrator. To be more detailed, he/she requires to manage the account in terms of creation, activation and termination.
After login for an administrator, one will see there is three types of managements in the navigation bar such as Staff Management, Admin Management and Requester Management. In each dropdown list, it contains navigation links of account creation and account listing.

Figure 5.71 Navigation bar in administrator page

Navigation link that named as “Account Creation” will direct the administrator to a account creation view for either type of user depending on which navigation link is being clicked.

Figure 5.72 Account creation view for academic staff

Navigation link that named as “Account Listing” will direct the administrator to an account list view for either type of user depending on which navigation link is being clicked. For example, the “Account Listing” navigation link of academic staff is being clicked, backend mechanism will be triggered to extract all account information of academic staff from database and display them in a list.
By click on either row in the list will direct the administrator to a detailed view of that particular account. A backend mechanism will triggered to check the status of the account and then display either activation or termination button depends on the status of that particular account. The button for change password will also be shown to allow administrator to modify the account password.

5.1.4 System Modules

5.1.4.1 Prioritization Module

In this system, prioritization means to arrange appointment in order of their relative importance. In other words, a type of requester can replace the time slot taken of appointment by another type of requester if he/she has a higher priority which defined by the targeted academic staff.
There are 4 types of requesters: academic staff, non-academic staff, student and visitor. Their priorities are declared in appointment rules by the targeted academic staff as shown in Figure 5.75.

![Figure 5.75 Sorting of priorities among requesters](image.png)

If the requester has a higher priority, he/she is able to see the appointment time period that the other requester(s) has made and decide whether going to replace it or not. The existing appointment type period can be seen in the first tab of booking form. It will be located below the available time slots.

![Figure 5.76 Booking form with higher priority](image.png)
Other than prioritization on appointment making, it is also available for appointment bring forward/postpone.

![Figure 5.77 Modal for appointment bring forward/postpone](image)

![Figure 5.78 Section of codes for checking priority and appointment time slots](image)
If the requester has successfully replace the appointment time period of the others for himself/herself, the system will then send a cancellation notice to previous requester with stated reason, an amendment notice to the targeted academic staff and indeed the confirmation notice to the current requester. Never the less, the existing google event will be either updated or deleted and a new google event will be created if they have synchronize their google calendar with time schedule in this system.

Figure 5.79 Section of codes for sending mails

5.1.4.2 Automated Reminder Module

This system provides an automated reminder feature. This feature will trigger the system to send reminders automatically to academic staff and requester. In default setting, the reminder will be sent 1 hour before the appointment is take place. The reminder will include all the necessary appointment information such as appointment date, start time and end time. Moreover, the automated reminder feature is configured to cooperate with appointment rules set by academic staff. In more detail, the academic staff can customize the time period for sending the reminder instead of following default setting.

The following is the example of reminder email and section of codes for configuring the automated reminder feature.
5.1.4.3 Daily Recap Module

This system provides a daily recap feature. This feature will trigger the system to send a recap about daily appointments to academic staff automatically. In default setting, the recap will be sent at 6 a.m. every day. The feature can be used to cooperate with appointment rules which set by academic staff. In more detail, the academic staff can customize the time period for sending the recap instead of following default setting.

The following is the example of recap and section of codes for configuring the daily recap feature.
Figure 5.82 Example of recap

Figure 5.83 Section of codes for daily recap feature
CHAPTER 6

TESTING

6.1 Unit Testing
Unit testing is a level of software testing where individual units or components of software are tested to determine whether they are perform as designed. If the test fails, the issue will need to be identified, resolved and then it will be retested until success.

6.1.1 Unit Test Cases
Unit testing should be involves in each individual unit of software, small until clicking links, working with forms, working with attachment and many more. The following will be briefly describe the unit test cases for the important modules in this system.
6.1.1.1 Unit Test Cases (Modules for Academic Staff)

**Appointment Management Module**

I. Appointment Making

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appointment Booking Navigation Link</td>
<td>Click on the navigation link named “Appointment Booking” which located in side bar menu.</td>
<td>Link to staff directory page.</td>
</tr>
</tbody>
</table>
| Staff Directory Submit Button                 | 1. Insert any data needed for input fields.  
2. Click on the submit button.                                                                                                                                | Academic staff details appear in box form or error message appears.                                                                                                                                   |
| Clickable display box for academic staff details | Click on the one of the boxes which display the academic staff details                                                                                                                                           | Link to booking form of that particular staff                                                                                                                                                     |
| Clickable day of calendar                     | 1. Choose a month.  
2. Click on a day (blue in colour) for appointment making.                                                                                                  | Availabilities of time slots will be shown on the selection input for starting time and ending time.                                                                                                   |
| Continue Button                               | Click on continue button                                                                                                                                                                                                 | The booking form will display the on tab ahead of current tab.                                                                                                                                 |
| Previous Button                               | Click on previous button                                                                                                                                                                                               | The booking form will display previous tab.                                                                                                                                                             |
1. Input all necessary fields such as appointment date, starting time, ending time, appointment types, contact number and email address.
2. Click on the book button.

1. New appointment is added to the database.
2. Approval notice is sent to both parties.
3. Google Event is added to google calendar if the party has connect to google calendar.
4. Redirect to appointment details page.

## II. View Appointment Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appointment [Calendar View] Navigation Link</td>
<td>Click on the navigation link named “Appointment [Calendar View]” which located in side bar menu.</td>
<td>Link to a page which view all appointments without the status of cancelled in calendar format.</td>
</tr>
<tr>
<td>Coloured time slot in calendar</td>
<td>Click on the coloured time slot in calendar.</td>
<td>Details of appointment will be displayed at the right hand side of the calendar.</td>
</tr>
<tr>
<td>Appointment [List View] Navigation Link</td>
<td>Click on the navigation link named “Appointment [List View]” which located in side bar menu.</td>
<td>Link to a page which view all appointments in list.</td>
</tr>
<tr>
<td>Clickable Table Row</td>
<td>Click on the targeted appointment in table row.</td>
<td>Link to a page which display appointment details</td>
</tr>
</tbody>
</table>
III. Appointment Bring Forward/Postpone Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bring Forward/Postpone Button</td>
<td>Click on the “bring forward/postpone” button in appointment calendar view/ appointment details view.</td>
<td>A modal will pop out with a calendar.</td>
</tr>
</tbody>
</table>
| Clickable day of calendar        | 1. Choose a month.  
2. Click on a day (blue in colour) for appointment making. | Availabilities of time slots will be shown on the selection input for starting time and ending time. |
| Confirm Button in modal          | Click on confirm button in modal.                                                   | 1. Update on the appointment will be completed.  
2. Bring forward/Postpone notice will be sent to both parties.  
3. Google Event in google calendar will be updated if the party has connect to google calendar.  
4. Refresh the appointment details page/appointment calendar page. |
## IV. Appointment Cancellation Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancellation Button</td>
<td>Click on the cancel button in appointment calendar view/appointment details view.</td>
<td>A modal will pop out with an input for cancel reason.</td>
</tr>
<tr>
<td>Confirm Button in modal</td>
<td>1. Fill in the cancel reason. &lt;br&gt;2. Click on confirm button</td>
<td>1. Update of status on the appointment will be completed. &lt;br&gt;2. Cancellation notice will be sent to both parties. &lt;br&gt;3. Google Event in google calendar will be deleted if the party has connect to google calendar. &lt;br&gt;4. Refresh the appointment details page/appointment calendar page.</td>
</tr>
</tbody>
</table>
### V. Appointment Validation Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Validation Button</td>
<td>Click on the validate button in appointment calendar view/ appointment details view.</td>
<td>A modal will pop out with inputs of radio button for the data of attendance and punctuality of requester.</td>
</tr>
<tr>
<td>Confirm Button in modal</td>
<td>1. Fill in the attendance and punctuality. 2. Click on confirm button</td>
<td>1. Update of attendance and punctuality on the appointment will be completed. 2. Refresh the appointment details page/appointment calendar page.</td>
</tr>
</tbody>
</table>

### Availability Management Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability Navigation Link</td>
<td>Click on the navigation link named “Availability” which located in side bar menu.</td>
<td>Link to a page which display the academic staff availabilities.</td>
</tr>
<tr>
<td>Calendar in the page for availabilities</td>
<td>Drag on the calendar (on blank time slot).</td>
<td>A modal pop out for prompting academic staff to add new availability.</td>
</tr>
</tbody>
</table>
### Google Calendar Synchronization Management Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synchronization Navigation Link</td>
<td>Click on the navigation link named “Synchronization” which located in side bar menu.</td>
<td>Link to a page which check on whether academic staff has connect google calendar with time schedule of this system. A connect button will be displayed if there is no synchronization happen before while a disconnect button will be displayed if synchronization has happen before.</td>
</tr>
<tr>
<td>Connect Button</td>
<td>Click on the connect button.</td>
<td>Link to a consent screen which require academic staff to fill in google account details and allow access to his/her google calendar.</td>
</tr>
</tbody>
</table>
### Disconnect Button

**Test**

Click on the disconnect button.

**Expected Output**

Google account information is removed for the staff’s account.

### Appointment Rules Management Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appointment Rules Navigation Link</td>
<td>Click on the navigation link named “Appointment Rules” which located in side bar menu.</td>
<td>Link to a page which shows all appointment rules if the appointment rules has been enabled or else an enable button is shown.</td>
</tr>
<tr>
<td>Enable Button</td>
<td>Click on the enable button.</td>
<td>Refresh the page and show all appointment rules.</td>
</tr>
<tr>
<td>Save Button</td>
<td>Click on the save button.</td>
<td>Appointment rules setting will be saved.</td>
</tr>
</tbody>
</table>

### Appointment Types Management Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appointment Types Navigation Link</td>
<td>Click on the navigation link named “Appointment Types” which located in side bar menu.</td>
<td>Link to a page which shows all appointment types in list along with two types of forms: add form and update form.</td>
</tr>
</tbody>
</table>
| Add Button in add form | 1. Fill in the name of the appointment type.  
       2. Click on add button. | 1. New appointment type will be added to the database.  
       2. The page will be refreshed to get latest appointment types. |
|------------------------|--------------------------------------------------|------------------------------------------------------------------|
| Update Button in list  | Click the update button of any appointment type  | 1. Update form will be appeared while add form is hidden.  
       2. The details of that particular appointment type will be displayed in update form |
| Update Button in update form | 1. Fill in the updated name of the appointment type.  
       2. Click on update button. | 1. Existing appointment type will be updated to the database.  
       2. The page will be refreshed to get latest appointment types. |
| Cancel Button in update form | Click the cancel button. | Update form will be hidden while add form will be displayed. |
| Delete Button in list | Click the delete button | A prompt will pop out to get the confirmation for deletion on appointment type. |
### Reminder Notice Management Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reminder Notices Navigation Link</td>
<td>Click on the navigation link named “Reminder Notices” which located in side bar menu.</td>
<td>Link to a page which shows multiple tabs for approval, bring forward/postpone, cancellation, priority cancellation notice respectively.</td>
</tr>
<tr>
<td>Submit button</td>
<td>1. Select either default or customize for notice.</td>
<td>1. Customize info is saved to database.</td>
</tr>
<tr>
<td></td>
<td>2. Input customization content if customize radio button is selected.</td>
<td>2. The page will be refreshed to get latest reminders approval setting.</td>
</tr>
<tr>
<td></td>
<td>3. Click on save button.</td>
<td></td>
</tr>
</tbody>
</table>

### Intake Form Management Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intake Form Navigation Link</td>
<td>Click on the navigation link named “Intake form” which located in side bar menu.</td>
<td>Link to a page which shows all intake questions for intake form, along with two types of forms: add form and update form.</td>
</tr>
<tr>
<td>Add Button in add form</td>
<td>1. Fill in the intake questions and other necessary information.</td>
<td>1. New intake question will be added to the database.</td>
</tr>
<tr>
<td></td>
<td>2. Click on add button.</td>
<td></td>
</tr>
</tbody>
</table>
| Update Button in list | Click the update button of any intake question. | 1. Update form will be appeared while add form is hidden.  
2. The details of that particular intake question will be displayed in update form |
|-----------------------|-------------------------------------------------|--------------------------------------------------------------------------------------------------|
| Update Button in update form | 1. Fill in the update details for the intake question.  
2. Click on update button. | 1. Existing question will be updated to the database.  
2. The page will be refreshed to get latest intake questions. |
| Cancel Button in update form | Click the cancel button. | Update form will be hidden while add form will be displayed. |
| Delete Button in list | Click the delete button | A prompt will pop out to get the confirmation for deletion on intake question. |
### Reports Generation Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reports Navigation Link</td>
<td>Click on the navigation link named “Reports” which located in side bar menu.</td>
</tr>
<tr>
<td></td>
<td>Link to a page which shows input fields that obtain information for generation of reports.</td>
</tr>
</tbody>
</table>
| Submit Button              | 1. Insert all necessary information to the input fields available.  
2. Click the submit button                                                 |
|                            | Report will be generated and show on the page.                                                                                                                |

### Profile Management Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>My Profile Navigation Link</td>
<td>Click on the navigation link named “My Profile” which located in side bar menu.</td>
</tr>
<tr>
<td></td>
<td>Link to a page which display the academic staff personal information.</td>
</tr>
<tr>
<td>Edit Profile Button</td>
<td>Click the edit profile button in the page which display the information of academic staff.</td>
</tr>
<tr>
<td></td>
<td>Redirect academic staff to a page where they can edit their profile.</td>
</tr>
</tbody>
</table>
| Save Button in Edit Profile Page | 1. Input all necessary information/ Change any information.  
2. Click save button.                                                                   |
|                            | 1. Information of academic staff is updated.  
2. Redirect academic staff to view profile page.                                             |
Change Password Navigation Link

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Click on the navigation link named “Change Password”</td>
<td>Link to a page which enquire academic staff to input current password and new password.</td>
</tr>
</tbody>
</table>

Submit Button in Change Password Page

| 1. Input current password, new password and confirmation password. |
| 2. Click on submit button.                                             |
| 1. Password changes will saved to database.                           |
| 2. Refresh the page and display success message.                     |

6.1.1.2 Unit Test Cases (Modules for Requester)

Appointment Management Module

I. Appointment Making

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appointment Booking Navigation Link</td>
<td>Click on the navigation link named “Appointment Booking” which located in side bar menu.</td>
<td>Link to staff directory page.</td>
</tr>
<tr>
<td>Staff Directory Submit Button</td>
<td>1. Insert any data needed for input fields.</td>
<td>Academic staff details appear in box form or error message appears.</td>
</tr>
<tr>
<td>Clickable display box for academic staff details</td>
<td>Click on the one of the boxes which display the academic staff details</td>
<td>Link to booking form of that particular staff</td>
</tr>
<tr>
<td>--------------------------------------------------</td>
<td>---------------------------------------------------------------------</td>
<td>---------------------------------------------</td>
</tr>
</tbody>
</table>
| Clickable day of calendar                        | 1. Choose a month.  
2. Click on a day (blue in colour) for appointment making. | Availabilities of time slots will be shown on the selection input for starting time and ending time. |
| Continue Button                                  | Click on continue button                                             | The booking form will display the on tab ahead of current tab. |
| Previous Button                                  | Click on previous button                                             | The booking form will display previous tab. |
| Book Button                                      | 1. Input all necessary fields such as appointment date, starting time, ending time, appointment types, contact number and email address.  
2. Click on the book button.                      | 1. New appointment is added to the database.  
2. Approval notice is sent to both parties.  
3. Google Event is added to google calendar if the party has connect to google calendar.  
4. Redirect to appointment details page.           |
### II. View Appointment Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appointment List</td>
<td>Click on the navigation link named “Appointment List” which located in side bar menu.</td>
<td>Link to a page which view all appointments in list.</td>
</tr>
<tr>
<td>Clickable Table Row</td>
<td>Click on the targeted appointment in table row.</td>
<td>Link to a page which display appointment details.</td>
</tr>
</tbody>
</table>

### III. Appointment Bring Forward/Postpone Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bring Forward/Postpone Button</td>
<td>Click on the “bring forward/postpone” button in appointment calendar view/ appointment details view.</td>
<td>A modal will pop out with a calendar.</td>
</tr>
<tr>
<td>Clickable day of calendar</td>
<td>1. Choose a month. 2. Click on a day (blue in colour) for appointment making.</td>
<td>Availabilities of time slots will be shown on the selection input for starting time and ending time.</td>
</tr>
<tr>
<td>Confirm Button in modal</td>
<td>Click on confirm button in modal.</td>
<td>1. Update on the appointment will be completed. 2. Bring Forward/Postpone notice will be sent to both parties.</td>
</tr>
</tbody>
</table>
3. Google Event in google calendar will be updated if the party has connect to google calendar.
4. Refresh the appointment details page/appointment calendar page.

### IV. Appointment Cancellation Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancellation Button</td>
<td>Click on the cancel button in appointment calendar view/appointment details view.</td>
<td>A modal will pop out with an input for cancel reason.</td>
</tr>
</tbody>
</table>
| Confirm Button in modal | 1. Fill in the cancel reason.  
2. Click on confirm button | 1. Update of status on the appointment will be completed.                        |
|                    |                                                           | 2. Cancellation notice will be sent to both parties.                             |
|                    |                                                           | 3. Google Event in google calendar will be deleted if the party has connect to google calendar. |
4. Refresh the appointment details page/appointment calendar page.

Profile Management Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>My Profile Navigation Link</td>
<td>Click on the navigation link named “My Profile” which located in side bar menu.</td>
<td>Link to a page which display the requester’s personal information.</td>
</tr>
<tr>
<td>Edit Profile Button</td>
<td>Click the edit profile button in the page which display the information of requester.</td>
<td>Redirect requester to a page where he/she can edit profile.</td>
</tr>
</tbody>
</table>
| Save Button in Edit Profile Page | 1. Input all necessary information/ Change any information.  
2. Click save button.                                                                                                                                           | 1. Information of requester is updated.  
2. Redirect requester to view profile page.                                                                                 |
| Change Password Navigation Link | Click on the navigation link named “Change Password”                                                                                                                                                                   | Link to a page which enquire requester to input current password and new password.                                        |
| Submit Button in Change Password Page | 1. Input current password, new password and confirmation password.  
2. Click on submit button.                                                                                                                                               | 1. Password changes will saved to database.  
2. The page will be refreshed and display success message.                                                                                                                 |
### Registration Module

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login Page</td>
<td>Key in URL for the web application named UTAR OASS.</td>
<td>UTAR OASS academic staff’s login page will be displayed.</td>
</tr>
<tr>
<td>Guest Navigation Link</td>
<td>Click on the navigation link named “Requester” which located in top menu.</td>
<td>Link to requester’s login page.</td>
</tr>
<tr>
<td>Registration link</td>
<td>Click on the link named as “Register a new membership” in the login box.</td>
<td>Link to registration page for requester that display multiple input fields.</td>
</tr>
</tbody>
</table>
| Register Button         | 1. Input necessary information.  
                          | 2. Click on register button.                                                      | 1. New requester account is created.  
                          | 2. Redirect to login page.                                                      |

### 6.1.1.3 Unit Test Cases (Modules for Administrator)

**Account Management Module**

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Academic Staff Account</td>
<td>Click on the navigation link named “Account Creation” which located in dropdown</td>
<td>Link to account creation page for academic staff.</td>
</tr>
<tr>
<td>Navigation Link</td>
<td>of Staff Management at top menu.</td>
<td></td>
</tr>
</tbody>
</table>
| Create Button in Account Creation Page for Academic Staff | 1. Input necessary information.  
2. Click on create button. | 1. New academic staff account is created.  
2. Redirect to view account summary of created account. |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrator Account Creation Navigation Link</td>
<td>Click on the navigation link named “Account Creation” which located in dropdown of Administrator Management at top menu.</td>
<td>Link to account creation page for administrator.</td>
</tr>
</tbody>
</table>
| Create Button in Account Creation Page for Administrator | 1. Input necessary information.  
2. Click on create button. | 1. New administrator account is created.  
2. Redirect to view account summary of created account. |
| Academic Staff Account List Navigation Link | Click on the navigation link named “Account Listing” which located in dropdown of Staff Management at top menu. | Link to a page which shows all academic staff accounts in list. |
| Clickable table row of brief academic staff’s info in account listing. | Click on the table row of any account info displayed. | Direct to a page which display account summary of that particular account with activate/deactivate button depends on its status. |
| Activate button in academic staff’s account summary page | Click on the activate button. | 1. The particular account is activated.  
2. Page refreshed. |
| Deactivate button in academic staff’s account summary page | Click on the deactivate button. | 1. The particular account is deactivated.  
2. Page refreshed. |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrator Account List Navigation Link</td>
<td>Click on the navigation link named “Account Listing” which located in dropdown of Administrator Management at top menu.</td>
<td>Link to a page which shows all administrator accounts in list.</td>
</tr>
<tr>
<td>Clickable table row of brief administrator’s account info in account listing.</td>
<td>Click on the table row of any account info displayed.</td>
<td>Direct to a page which display account summary of that particular account with activate/deactivate button depends on its status.</td>
</tr>
</tbody>
</table>
| Activate button in administrator’s account summary page | Click on the activate button. | 1. The particular account is activated.  
2. Page refreshed. |
| Deactivate button in administrator’s account summary page | Click on the deactivate button. | 1. The particular account is deactivated.  
2. Page refreshed. |
| Requester Account List Navigation Link                    | Click on the navigation link named “Account Listing” which located in dropdown of Requester Management at top menu. | Link to a page which shows all requester accounts in list. |
| Clickable table row of brief requester’s account info in account listing. | Click on the table row of any account info displayed. | Direct to a page which display account summary of that particular account with |
| Activate button in requester’s account summary page | Click on the activate button. | 1. The particular account is activated.  
2. Page refreshed. |
| Deactivate button in requester’s account summary page | Click on the deactivate button. | 1. The particular account is deactivated.  
2. Page refreshed. |
6.1.2 **Unit Test Scripts**

PHPUnit, a PHP programmer-oriented testing framework is used to create unit test codes for the system of this project as the construction is mainly using PHP programming language.

Figures below show several sections of test codes for unit testing.

![Figure 6.1 Section of codes for unit testing](image1)

Figure below shows the result of testing for the section of codes shown above.

![Figure 6.2 Result of the unit testing](image2)
6.2 Integration Testing
Integration testing is performed after unit testing. It is performed when individual modules are combined as a group. Integration testing is crucial to discover error when different modules are integrated to perform functionalities.

6.2.1 Integration Test Cases
The following shows several integration test cases involves combination of modules to perform functionalities. The execution steps contain roles to let tester understand where to perform the step by acting as that particular user.

<table>
<thead>
<tr>
<th>Modules Involved</th>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
</table>
| 1. Google        | Add new google events of academic appointments in google calendar. | 1. Academic staff click on the navigation link “Synchronization” under dropdown of settings in side navigation bar.  
2. Academic staff click on the connect button.  
3. Academic staff log into his/her google account and allow privilege for the system to access google calendar.  
4. Requester search for that particular academic staff booking form. | 1. New appointment has been made and saved to database.  
2. Notice will be sent through email to both parties.  
3. Google event will be added to the google calendar that the person has registered with. |
<p>| Synchronization Management | | | |
| 2. Appointment Making | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>
| 1. **Google Synchronization Management** | 5. Requester fill in all the information needed into the booking form such as appointment date, starting time, ending time and etc.  
| 2. **View Appointments** | Update google events of academic appointments in google calendar. | 1. Academic staff click on the navigation link “Synchronization” under dropdown of settings inside navigation bar.  
2. Academic staff click on the connect button.  
3. Academic staff log into his/her google account and allow privilege for the system to access google calendar.  
4. System will automatic insert the existing appointments to google account.  
5. Confirmer/Requester search and view for the appointment details.  
6. He/she click on the “bring forward/postpone” button and select a new date and time.  
7. He/she click confirm button. | 1. Details of appointment has been updated to database.  
2. Notice will be sent through email to both parties.  
3. Google event will be updated to the google calendar that the person has registered with. |
| 1. Google Synchronization Management | Delete google events of academic appointments in google calendar. | 1. Academic staff click on the navigation link “Synchronization” under dropdown of settings in side navigation bar.  
2. Academic staff click on the connect button.  
3. Academic staff log into his/her google account and allow privilege for the system to access google calendar.  
4. System will automatic insert the existing appointments to google account.  
5. Confirmer/Requester search and view for the appointment details.  
6. He/she click on the cancel button. | 1. The appointment status will set to “cancelled” and saved to database.  
2. Cancellation Notice will be sent through email to both parties.  
3. Google event will be deleted in the google calendar that the person has registered with. |
| 2. View Appointments | | | |
| 1. Reminder notices management  
2. Appointment making/postpone/cancellation | Send customized notice to both parties: confirmer and requester. | 1. Academic staff click on the navigation link “reminder notices” under dropdown of settings in side navigation bar.  
2. Academic staff input customized content for approval, postpone, cancellation and priority cancellation.  
3. Academic staff saves the setting. | 1. The appointment will then be added/updated into the database.  
2. Customized notice will be sent to both parties. |
<table>
<thead>
<tr>
<th>1. Availability management</th>
<th>Make appointment with targeted academic staff. (If no existing availability, appointment cannot be made)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. Appointment making</td>
<td>4. An appointment is later made/ brought forward/ postponed/ cancelled.</td>
</tr>
<tr>
<td></td>
<td>1. Academic staff click on the navigation link named “Availability” in side navigation bar.</td>
</tr>
<tr>
<td></td>
<td>2. Academic staff perform drag function on the calendar shown.</td>
</tr>
<tr>
<td></td>
<td>3. Academic staff saves the availability.</td>
</tr>
<tr>
<td></td>
<td>4. Requester search for that particular academic staff booking form.</td>
</tr>
<tr>
<td></td>
<td>A calendar will be shown along with blue colour days which represent that days that academic</td>
</tr>
<tr>
<td></td>
<td>staff is free.</td>
</tr>
<tr>
<td>1. Appointment Rules</td>
<td>Appointment is not allowed if the appointment is made one day before it takes place</td>
</tr>
<tr>
<td></td>
<td>1. Academic staff click on the navigation link named “Appointment Rules” under dropdown of</td>
</tr>
<tr>
<td></td>
<td>setting in side navigation bar.</td>
</tr>
<tr>
<td></td>
<td>2. Academic staff enabled the appointment rules.</td>
</tr>
<tr>
<td></td>
<td>3. Academic staff set the appointment booking time limit to one day.</td>
</tr>
<tr>
<td></td>
<td>4. Academic staff saves the setting.</td>
</tr>
<tr>
<td></td>
<td>5. Requester search for that particular academic staff booking form.</td>
</tr>
<tr>
<td></td>
<td>Error message appears to state no available time slots as appointment rules is enabled.</td>
</tr>
</tbody>
</table>
6. Requester click on the day (blue colour) on the same day that he/she look for the booking form.

6.3 System Testing

System testing is carried out after integration testing to evaluate whether the application is in compliance with the requirements. System testing is conducted on fully integrated software.

6.3.1 System Test Cases

The following shows system test cases on fully combined modules. The execution steps contain roles to let tester understand where to perform the step by acting as that particular user.

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Test Execution Steps</th>
<th>Expected Output</th>
</tr>
</thead>
</table>
| Appointment made for FYP Meeting with rule that it cannot take place within one day. Appointment is then added as google event in google calendar. | 1. Academic staff click on the navigation link “Appointment Type” in side navigation bar.  
2. Academic staff add new appointment type by clicking add button after filling “FYP Meeting” into the input field.  
3. Academic staff click on the navigation link “Appointment Rules” in side navigation bar.  
4. Academic staff enables the appointment rules. | 1. Availabilities of that particular academic staff shall only start on one day.  
2. Appointment type named “FYP Meeting” is available.  
3. Appointment is added to database after book button is clicked. |
| Requester is able to replace one’s appointment time slot as having higher priority. Appointment will be updated with new requester info. Google event will be then updated in google calendar. | 1. Academic staff click on the navigation link “Appointment Rules” in side navigation bar.  
2. Academic staff enables the appointment rules.  
3. Academic staff enables the prioritization.  
4. Academic staff arranges the priority for requester based on the type: academic staff, non-academic staff, student and visitor.  
5. Academic staff saves the settings.  
6. Academic staff click on the navigation link “Synchronization” under dropdown of settings in side navigation bar.  
7. Academic staff click on the connect button. | 1. Appointments which booked by requester that has lower priority than the current requester will be shown.  
2. Time slots of those appointments will be shown after one of them is selected.  
3. Approval notice will be sent to current requester, priority cancellation notice will be sent to previous requester and amendment |
8. Academic staff log into his/her google account and allow privilege for the system to access google calendar.
9. Previous appointments in system will be added to google calendar.
10. A requester with highest priority search that particular academic staff booking form.
11. Requester select for a day that has available time slots.

4. Google event will be updated to new date and time for the academic staff.

notice will be sent to the targeted academic staff after the book button is clicked.
### 6.4 User Acceptance Test

#### 6.4.1 User Acceptance Test Cases (for Academic Staff)

<table>
<thead>
<tr>
<th>Test Modules</th>
<th>Test Description</th>
<th>Date Tested</th>
<th>Sign-off by</th>
<th>Sign-off Date</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appointment Making</td>
<td>Able to add appointment to database by complete the booking form.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>View Appointments</td>
<td>Able to view appointments in calendar view / list view.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Appointment Bring Forward/Postpone</td>
<td>Able to change date and time for existing appointment by clicking “bring forward/postpone” button in the view of appointment details.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Appointment Cancellation</td>
<td>Able to cancel existing appointment by clicking cancel button in the view of appointment details.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feature</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Appointment Validation</td>
<td>Able to validate appointment by entering attendance and priority of requester by clicking validate button in the view of appointment details.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Availability Management</td>
<td>Able to add new/ update existing/ delete existing availability.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Google Calendar Synchronization Management</td>
<td>Able to connect/ disconnect google calendar with the time schedule of the system.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Appointment Rules Management</td>
<td>Able to set appointment rules on appointment booking, appointment postpone, appointment cancellation and prioritization.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Appointment Types Management</td>
<td>Able to add new/ update existing/ delete existing appointment type.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reminder Notice Management</td>
<td>Able to customize notice content for reminder after adding/ updating/ cancelling of appointment.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feature</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Intake Form Management</td>
<td>Able to add new/ update existing/ delete existing intake questions.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reports Generation</td>
<td>Able to produce reports for:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1. No-shows appointments</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Completed appointments</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Cancelled appointments</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4. Filtered appointments with booking period</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5. Filtered appointments with appointment types.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Profile Management</td>
<td>Able to view/ edit personal information and change account’s password.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
6.4.2 User Acceptance Test Cases (for Requester)

<table>
<thead>
<tr>
<th>Test Modules</th>
<th>Test Description</th>
<th>Date Tested</th>
<th>Sign-off by</th>
<th>Sign-off Date</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appointment Making</td>
<td>Able to add appointment to database by complete the booking form.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>View Appointments</td>
<td>Able to view appointments in calendar list view.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Appointment Bring Forward/ Postpone</td>
<td>Able to change date and time for existing appointment by clicking “bring forward/postpone” button in the view of appointment details.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Appointment Cancellation</td>
<td>Able to cancel existing appointment by clicking cancel button in the view of appointment details.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Profile Management</td>
<td>Able to view/ edit personal information and change account’s password.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test Modules</td>
<td>Test Description</td>
<td>Date Tested</td>
<td>Sign-off by</td>
<td>Sign-off Date</td>
<td>Comments</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>----------------------------------------------------------------------------------</td>
<td>-------------</td>
<td>-------------</td>
<td>---------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Account Management</td>
<td>Able to create account for academic staff and administrator.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Account Creation)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Account Management</td>
<td>Able to view accounts in list for academic staff, administrator and requester.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Account Listing)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Account Management</td>
<td>Able to activate/deactivate account in view of account’s summary for academic staff, administrator and requester.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Account Activation/Termination)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
6.5 Software Testing Tools

6.5.1 Selenium IDE

Selenium is a portable software testing framework for web applications. It provides a record tool for authoring tests without the need to learn a test scripting languages. It is suitable for my application as it support PHP programming languages and it is able to run against most modern web browsers.

The following is the example of test cases recorded into the selenium.

![Example of test cases in selenium](image)

**Figure 6.3 Example of test cases in selenium**
CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 Contribution of the Application
This system helps academic staff to manage their academic appointments with automated approval for appointment making, appointment bring forward/postponing and appointment cancelling from any requester. Academic staff shall only require to insert their availabilities (free time slots) into the system and it will then be displayed to requester for appointment booking or appointment bring forward/postponing. This feature has prevent academic staff to waste their time on manual approval for any appointment request. Furthermore, the system will store all appointment records for further reference which helps to eliminate manual record keeping by using pen and paper technique. It also beneficial to those who using google calendar as it can sync with the respective calendar to keep them up to date on academic appointments.

Other than benefits to academic staff, the system helps requester to obtain immediate approval of appointment request. Compare to request method in current workflow, it is more efficient by using this system as requester does not need to wait for late approval or rejection from academic staff. Furthermore, requester can request appointment in 24/7 rather than only restricted to office hours.

The most important benefit is that it can reduce no-show for appointment by using excuse such as forgotten due to no reminder notice. The reason is that the system will send reminder notice to both parties after the appointment has been made, brought forward/postponed or cancelled and to both parties 1 hour before it took place.

7.2 Limitation and Future Enhancements
Although the system is ready for real-time operating, but it still consists of multiple limitations. The following shows the limitations, along with suggestions for future enhancement which could improve the effectiveness and provide more functionality to users.
<table>
<thead>
<tr>
<th>#</th>
<th>Limitation</th>
<th>Suggestion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Current system does not have notification feature.</td>
<td>Develop a notification feature to immediate notify academic staff that he/she has new appointment request, update of existing appointment or cancellation of existing appointment.</td>
</tr>
<tr>
<td>2</td>
<td>Current system has only 5 different kinds of reports.</td>
<td>Develop more selection of types of reports for academic staff.</td>
</tr>
<tr>
<td>3</td>
<td>Current system does not consist of feedback feature for requester.</td>
<td>Develop a feedback feature for requester to commend on the academic staff. It could later merge with UTAR existing system to compute the KPI for each academic staff.</td>
</tr>
<tr>
<td>4</td>
<td>Current system does not have wait list for appointment making.</td>
<td>Develop a wait list for requester to wait for the time slot obtained by the others. It should then allow the requester in wait list to fill in the slots if slot is empty out due to cancellation or brought forward/ postponed of appointment.</td>
</tr>
<tr>
<td>5</td>
<td>Current system is synchronize with only one external calendar</td>
<td>Construct coding to allow user to connect to multiple calendars such as iCal and etc.</td>
</tr>
<tr>
<td>6</td>
<td>Current system only allows academic staff to synchronize with external calendar</td>
<td>Develop the synchronization feature for requester to allow them to connect their external calendar with the time schedule in system.</td>
</tr>
<tr>
<td>7</td>
<td>Current system has only few rules available for appointment management</td>
<td>Compute more rules to ensure the appointment management is fair for every users.</td>
</tr>
</tbody>
</table>
8. Current system allows requester to view the availabilities in booking form only. Construct a new view which allows requester to view the academic staff availabilities or time schedule directly.

9. Current system only allows administrator to create, activate or deactivate accounts, along with change password feature. Compute a feature to allow administrator to edit the information of all accounts, along with feature that can restrict the functionality used by users.

7.3 Conclusion

This project has been developed over a period of 6 months, starting from the planning to testing phase in software development life cycle.

At the very beginning of the development, a qualitative and a quantitative research has been carried out to obtain the information of current workflow for appointment scheduling, along with the suggestions of functionality of the system that is going to be built. Results collected from the research were then be consolidated and analysed to produce functional and non-functional requirements. After the production of requirement list, timeline was then developed in Gantt chart to ensure the development can be finished before submission date.

In later phase, design was done by developing case diagram, sequence diagrams, entity relationship diagrams and user interface design. The web-based academic appointment scheduling system was then be developed by referencing the requirements and designs from previous phases. Testing was the final phase that used to verify the system is working according to what it is needed.

In conclusion, an online academic appointment scheduling system is delivered at the end of software development life cycle and all the objectives of the project has been fulfil as below:

- To conduct literature review of existing online scheduling systems.
- To conduct interview and survey to stakeholders such as lecturers and students who are involved in appointments in order to determine the important features of online academic scheduling system.
• To develop a web-based academic appointment scheduling system that can schedule and maintain priority of requests via Google Calendar system.
REFERENCES


APPENDICES

APPENDIX A: Interview Questions

**Interview questions for academic staff**

1. How do you manage on your appointment schedule?
2. Why are you preferring that technique?
3. What are the problems faced by using current technique in manage appointment schedule?
4. What are the platform(s) that you usually receive appointment requests?
5. What are the common problems that you have faced for the appointment process: before, during and after?
6. Do you prefer to have an automated online scheduling system?
   a. If yes, why do you prefer? Do you have any recommendations? What are your expectation on the system?
   b. If no, why do you deny the system?

**Interview questions for students**

1. How do you make appointment with academic staff?
2. What are the common problems that you faced for whole appointment process: before, during and after?
3. Do you prefer to have an automated online appointment scheduling system for appointment booking?
   a. If yes, why do you prefer? Do you have any recommendations? What are your expectation on the system?
   b. If no, why do you deny the system?
APPENDIX B: Questionnaire Questions

Questionnaire questions for academic staff

1. How many appointments do you have to attend in a month?

   [Only one answer is allowed.]

   - Less than 10
   - In between 10 to 20
   - More than 20

2. What are the appointment types that you usually have?

   [Only one answer is allowed.]

   - Advisory section
   - Assignment discussion
   - Consultation section
   - Department Meeting
   - Project discussion
   - Other, please specify: ________________________________

3. What technique do you applied to manage your appointment schedule?

   [Only one answer is allowed.]

   - Pen-and-paper technique (Continue to Question 3)
   - Computerised technique  (Continue to Question 4)

4. What are the common problems encountered using pen-and-paper technique? [Only one answer is allowed.]

   - No problem occurs
   - Multiple appointment schedules cause confusion
   - Messiness cause mistake
   - Unable to access appointment schedules if it is not around
   - Missing appointment schedule
   - Other, please specify: ________________________________
5. What are the common problems encountered using computerised technique? [Only one answer is allowed.]

- No problems
- Requires to fill in multiple information (time-consuming)
- Other, please specify: ________________________________

6. What are the common problems encountered during appointment process (before, during or after appointment is held)? [Please tick one from each row.]

<table>
<thead>
<tr>
<th>Common problems faced during appointment process</th>
<th>Level of Influence (Importance)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No-shows for appointments</td>
<td>Least (1)</td>
</tr>
<tr>
<td>Mistake in arranging appointment (double booking)</td>
<td></td>
</tr>
<tr>
<td>Requesters request appointment at last minute</td>
<td></td>
</tr>
<tr>
<td>Unable to acknowledge appointments on time</td>
<td></td>
</tr>
<tr>
<td>No feedback from requesters about acknowledgement of request</td>
<td></td>
</tr>
</tbody>
</table>

7. If there is an online appointment scheduling system available for UTAR, what are your expectation on the features of the system? [Please tick one from each row.]

<table>
<thead>
<tr>
<th>Features</th>
<th>Level of Influence (Importance)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Automated prioritization on requester types</td>
<td></td>
</tr>
<tr>
<td>Synchronization with external calendar</td>
<td></td>
</tr>
</tbody>
</table>
Mobile platform provided
Automated notification/reminders
Tracking on appointment history
Automated acknowledgement of appointment requests

**Questionnaire questions for students**

1. How many appointments do you have with academic staff in a month?  
   *Only one answer is allowed.*
   - Less than 3
   - In between 3 to 5
   - More than 5

2. What are the appointment types that you usually have with academic staff?  
   *Only one answer is allowed.*
   - Advisory section
   - Assignment discussion
   - Consultation section
   - Other, please specify: ________________________________

3. What are the common problems encountered during appointment process (before, during or after appointment is held)?  
   *Please tick one from each row.*

<table>
<thead>
<tr>
<th>Common problems faced during appointment process</th>
<th>Level of Influence (Importance)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Academic staff do not reply on the appointment booking</td>
<td>Most (5) (4) (3) (2) Least (1)</td>
</tr>
<tr>
<td>Do not have guideline for appointment booking</td>
<td></td>
</tr>
</tbody>
</table>
4. If there is an online appointment scheduling system available for UTAR, what are your expectation on the features of the system?

[Please tick one from each row.]

<table>
<thead>
<tr>
<th>Features</th>
<th>Level of Influence (Importance)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Most (5)</td>
</tr>
<tr>
<td>Unable to view academic staff’s timetable</td>
<td></td>
</tr>
<tr>
<td>Academic staff having other meetings</td>
<td></td>
</tr>
<tr>
<td>No-shows for the appointments</td>
<td></td>
</tr>
</tbody>
</table>

Automated notification/ reminders
View on appointment schedule of academic staff
Instant acknowledgement
APPENDIX C: System Installation Manual

Installation on Windows Operating System

1) Install PHP version with at least version 5.6.
   - http://php.net/downloads.php
2) Install Composer.
   - https://getcomposer.org/download/
3) Copy the folder named “utaroass” to any preferred location.
4) Open the .env file in “utaroass” folder and modify database setting to your preferred database name, username and password.
5) Create a database with the name which configured in .env to MySQL server.
6) Open command prompt.
7) Direct yourself to the “utaroass” folder.
8) Type the following code to migrate database tables to the database that you have created.
   - php artisan migrate
9) Type the following code to migrate some hard coded data to the database tables.
   - php artisan db:seed
10) Type the following to start up the server
    - php artisan serve
11) Open a browser and type the following code to access the web system.
    - localhost::8000