

Senior Project



Supervised By

Dr. Nouf Rahimi

Team Members

Manar Majeed	2114050
Sabreen Binsalman	2112235
Maya Karoma	2010680
Shahad Alshehri	2008426
Shomoukh Qattan	2007667

Table 1 List of Changes

Chapter	Change Name	Change Description
1	Programming Language	- Remove python language.
1	Recommendation System	- Change the methodology of the recommendation.
1	Gantt Chart	- Add second semester Gantt chart
2	Current Situation	- Add the current situation for student clubs. - Add the current situation for psychological guidance.
3	Functional Requirement	- Remove join functionality from student activity - Remove the join functionality from the study group - Change the priority of the certificate - Add login functionality for the admin panel - Add manage student clubs functionality - Add logout functionality for admin panel
3	Design constraints	- Add design constraints
3	Use-Case Diagram	- Update the diagram
4	Update the chapter	
5	Update the chapter	
6	Update the chapter	
7	Update the chapter	

Table of Contents

Chapter 1: Introduction	14
1.1 Introduction	15
1.2 Problem Definition.....	15
1.3 Proposed Solution	16
1.4 Aim and Objectives	16
1.5 Novelty/ Contribution	17
1.6 Methods.....	17
1.6.1 Flutter Framework.....	17
1.6.2 Laravel Framework.....	17
1.6.3 Programming Languages	17
1.6.4 Real-Time Communication	18
1.6.5 Recommendation System	18
1.7 Project Plan (Gantt Chart)	19
1.8 Conclusion.....	20
Chapter 2: Related Works	21
2.1 Introduction	22
2.2 Current Situation	22
2.2.1 Exclusive Offers	22
2.2.2 Events	22
2.2.3 Clinic Schedule.....	23

2.2.4 Lost and Found Items Management	23
2.2.5 Volunteering Opportunities	23
2.2.6 Student Clubs.....	23
2.2.7 Psychological Guidance.....	24
2.3 Existing System.....	24
2.4 Comparison Result	34
2.5 Conclusion.....	37
 Chapter 3: Requirement Gathering and Analysis.....	38
3.1 Introduction	39
3.2 Functional Requirement	39
3.3 Non-Functional Requirement	43
3.4 Design Constraints	45
3.5 Figma Prototype	46
3.6 Use-Case Diagram.....	56
3.7 Use-Case Specifications.....	57
3.8 Conclusion.....	68
 Chapter 4: Methodology and Tools.....	69
4.1 Introduction	70
4.2 Agile Development and System Requirement	70
4.2.1 Agile Development.....	70
4.2.2 Scrum Methodology	71

4.2.3 Scrum tool.....	71
4.2.4 Product Backlog	71
4.2.5 Sprint Backlog	79
4.3 Burndown chart	82
4.4 Task and their Allocation.....	86
4.5 Conclusion.....	92
Chapter 5: Analysis and Design.....	93
5.1 Introduction	94
5.2 Class Diagram	94
5.3 Sequence Diagram.....	96
5.4 Activity Diagram.....	99
5.5 System Architecture	101
MVC Architecture	101
5.6 Conclusion.....	103
Chapter 6: Implementation	104
6.1 Introduction	105
6.2 Engineering Standards.....	105
6.3 Tools and Languages	107
6.3.1 Flutter Framework	107
6.3.2 Dart language.....	107
6.3.3 Firebase.....	107

6.3.4 Android Studio	108
6.3.5 Visual Studio Code.....	108
6.4 System Implementation (Code Snippets of Important Functions)	108
6.4.1 Mobile Application Implementation	108
6.4.2 Admin Panel – Web Application Implementation	126
6.5 Conclusion.....	130
Chapter 7: Testing	131
7.1 Introduction	132
7.2 Unit Testing	132
7.3 Integration Testing.....	162
7.4 Acceptance Testing.....	169
7.5 System Testing	174
7.6 Usability Testing.....	178
7.7 Conclusion.....	181
Chapter 8: Conclusion.....	182
8.1 Introduction	183
8.2 Limitations	183
8.3 Future Work.....	183
8.4 Conclusion.....	183
References.....	185

List of Illustrations

Figures

Figure 1 Gantt Chart for milestone and deadlines	19
Figure 2 MyKAU Mobile Application UI	24
Figure 3 KAU Loyalty Mobile Application UI.....	25
Figure 4 UQU Mobile Application UI	26
Figure 5 UQU Alumni Mobile Application UI.....	27
Figure 6 Nafie Program Website UI	27
Figure 7 KFU Website UI	28
Figure 8 Mazaya Platform Website UI	29
Figure 9 Maharat Website UI.....	29
Figure 10 KKUx Website UI	30
Figure 11 AASTMT Student Portal Mobile Application UI.....	31
Figure 12 University of Manchester Website UI	31
Figure 13 London's Global University Website UI	32
Figure 14 University of Kent Mobile Application UI.....	32
Figure 15 Axenda Hubs Mobile Application UI	33
Figure 16 CampusGroups Mobile Application UI.....	33
Figure 17 Piogroup Mobile Application UI	34
Figure 18 Login Page.....	47
Figure 19 Profile Page	48
Figure 20 Home Page.....	49
Figure 21 Services Page	50

Figure 22 Exclusive Offers Pages	50
Figure 23 Volunteering Opportunities Page	51
Figure 24 Events Page	52
Figure 25 Students Clubs Pages.....	52
Figure 26 Study Groups Pages.....	53
Figure 27 Student Activities Pages	54
Figure 28 Clinics Page	54
Figure 29 Lost and Found Items Pages.....	55
Figure 30 Chat Pages	55
Figure 31 Saved List Page	56
Figure 32 Use-Case Diagram.....	57
Figure 33 Login Functional Requirements Product Backlog	72
Figure 34 Profile Functional Requirements Product Backlog	72
Figure 35 Exclusive Offers Functional Requirements Product Backlog	72
Figure 36 Volunteering Opportunities Functional Requirements Product Backlog	73
Figure 37 Student Clubs Functional Requirements Product Backlog	73
Figure 38 Clinic Schedule Functional Requirements Product Backlog.....	73
Figure 39 Event Functional Requirements Product Backlog.....	74
Figure 40 Lost and found items Functional Requirements Product Backlog	74
Figure 41 Student Activity Functional Requirements Product Backlog.....	74
Figure 42 Study Group Functional Requirements Product Backlog	75
Figure 43 Live Chat and Push Notification Functional Requirements Product Backlog	75
Figure 44 Save List and Psychological Guidance Functional Requirements Product Backlog ...	75
Figure 45 Certificate and Logout Functional Requirements Product Backlog.....	76

Figure 46 Admin Panel Login Functional Requirements Product Backlog.....	76
Figure 47 Manage exclusive offers Functional Requirements Product Backlog.....	76
Figure 48 Manage Event Functional Requirements Product Backlog.....	77
Figure 49 Manage Student Clubs Functional Requirements Product Backlog.....	77
Figure 50 Manage volunteering opportunity Functional Requirements Product Backlog	77
Figure 51 Clinic Schedule Functional Requirements Product Backlog.....	77
Figure 52 User Interface Non-Functional Requirements Product Backlog	78
Figure 53 Security Non-Functional Requirements Product Backlog	78
Figure 54 Interoperability Non-Functional Requirements Product Backlog	78
Figure 55 Performance Non-Functional Requirements Product Backlog	78
Figure 56 Usability Non-Functional Requirements Product Backlog	79
Figure 57 Maintainability Non-Functional Requirements Product Backlog	79
Figure 58 Availability Non-Functional Requirements Product Backlog	79
Figure 59 Sprint 1 Backlog.....	80
Figure 60 Sprint 2 Backlog.....	80
Figure 61 Sprint 3 Backlog.....	81
Figure 62 Sprint 4 Backlog.....	81
Figure 63 Sprint 5 Backlog	82
Figure 64 Sprint 6 Backlog.....	82
Figure 65 Sprint 1 Burndown chart.	83
Figure 66 Sprint 2 Burndown chart.	83
Figure 67 Sprint 3 Burndown chart.	84
Figure 68 Sprint 4 Burndown chart.	84
Figure 69 Sprint 5 Burndown chart.	85

Figure 70 Sprint 6 Burndown chart.	85
Figure 71 Task Allocation for Ideation	86
Figure 72 Tasks Allocation for Chapter 1	86
Figure 73 Tasks Allocation for Chapter 2	86
Figure 74 Tasks Allocation for Chapter 3	86
Figure 75 Tasks Allocation for Chapter 4	87
Figure 76 Tasks Allocation for Chapter 5	87
Figure 77 Tasks Allocation for Chapter 6	87
Figure 78 Tasks Allocation for Chapter 7	87
Figure 79 Class Diagram	95
Figure 80 Sequence Diagram - Login	96
Figure 81 Sequence Diagram - Found Item	97
Figure 82 Sequence Diagram - Lost Item	97
Figure 83 Sequence Diagram - Study Group	98
Figure 84 Sequence Diagram - Events	98
Figure 85 Activity Diagram - Create Lost or Found Item Report	99
Figure 86 Activity Diagram - Study Group	100
Figure 87 Activity Diagram - Event	100
Figure 88 System Architecture	101
Figure 89 MVC Architecture for Admin Panel	102
Figure 90 MVC Architecture for Mobile Application	102
Figure 91 Engineering Standard	105
Figure 92 Implementation of FirebaseAPI Class	109
Figure 93 Implementation of Theme File	110

Figure 94 Implementation of Important Functions in common_functions File.....	112
Figure 95 Implementation of Push Notification.	114
Figure 96 App Setup Functions.....	115
Figure 97 Implementation of Lunch Screen	116
Figure 98 Implementation of Login.....	117
Figure 99 Implementation of Save List	118
Figure 100 Implementation of Live Chat- ChatService Class	120
Figure 101 Implementation of Live Chat - UserChatItem Widget	121
Figure 102 Exclusive Offers Recommendation	122
Figure 103 Implementation of Lost and Found	123
Figure 104 Implementation of Study Group and Student Activity	124
Figure 105 Clinic Schedule - Filter by Branch	125
Figure 106 Implementation of Login in Admin Panel	127
Figure 107 Implementation of Workshops Controllers.....	129
Figure 108 Usability Testing - Learnability	179
Figure 109 Usability Testing - Memorability	179
Figure 110 Usability Testing - Efficiency	180
Figure 111 Usability Testing - Satisfaction.....	180

Tables

Table 1 List of Changes	2
Table 2 Similar software of Saudi universities	35
Table 3 Similar software of abroad universities.....	36
Table 4 Similar software of University Communities	37
Table 5 Functional Requirements	40
Table 6 Non-Functional Requirements	44
Table 7 Use-Case Specification - Edit User Profile	57
Table 8 Use-Case Specification - Add new Exclusive Offer	58
Table 9 Use-Case Specification - Remove Exclusive Offer	58
Table 10 Use-Case Specification - View Exclusive Offers.....	59
Table 11 Use-Case Specification - Add new Volunteering Opportunity.....	59
Table 12 Use-Case Specification - Edit Volunteering Opportunity	59
Table 13 Use-Case Specification - Remove Volunteering Opportunity.....	60
Table 14 Use-Case Specification - Register in Volunteering Opportunity	60
Table 15 Use-Case Specification - Register in Student Club	61
Table 16 Use-Case Specification - Add Clinic Schedule	61
Table 17 Use-Case Specification - Edit Clinic Schedule.....	62
Table 18 Use-Case Specification - Remove Clinic Schedule	62
Table 19 Use-Case Specification - Book an Appointment at the Clinic	62
Table 20 Use-Case Specification - Add New Event	63
Table 21 Use-Case Specification - Edit Event.....	63
Table 22 Use-Case Specification - Remove Event	64
Table 23 Use-Case Specification - Register in an Event	64

Table 24 Use-Case Specification - Report a Lost Item.....	64
Table 25 Use-Case Specification - Report a Found Item.....	65
Table 26 Use-Case Specification - Create a New Student Activity	65
Table 27 Use-Case Specification - Create a Study Group	66
Table 28 Use-Case Specification - Contact Other Users	66
Table 29 Use-Case Specification - Add an Event to Save List	66
Table 30 Use-Case Specification - Remove an Event from the Save List.....	67
Table 31 Use-Case Specification - Recommend Exclusive Offers.....	67
Table 32 Tasks Allocation and Their Assigned Team Member.....	88
Table 33 Unit Testing.....	132
Table 34 Integration Testing	162
Table 35 Acceptance Testing.....	169
Table 36 System Testing	174
Table 37 Usability Testing	178

Chapter 1: Introduction

1.1 Introduction

The university community is a flourishing ecosystem. It consists of students, professors, staff, and administrators who work together towards goals of growth, academic excellence, and positively impacting society. This community is like a world that brings together people from cultures, ideas, and experiences. They collaborate to create an environment that promotes teamwork, sharing knowledge, and personal development. [1].

Besides academics, the university community offers students opportunities to explore their interests and passions through clubs and extracurricular activities. Participating in these initiatives helps students refine their skills, develop teamwork abilities, and nurture leadership qualities. Through these experiences, students build connections with others. Develop a sense of responsibility towards their community. This lively atmosphere fosters a culture of learning that equips students with the skills to tackle challenges while seizing exciting opportunities.

1.2 Problem Definition

As a member of the UJ (University of Jeddah), it can be overwhelming to stay informed about the numerous services provided by the university. These services include lost and found items management, exclusive offers, volunteering opportunities, clinic schedules, and events, including training courses, conferences, and workshops. In addition, students are offered psychological guidance, access to student clubs, and various activities during breaks. Unfortunately, these services are frequently communicated through multiple channels such as Email, Twitter, Snapchat, or university website, and not every university community member regularly checks these platforms.

This can result in missed opportunities and valuable information for those who rely on friends to stay informed. The lack of a centralized and user-friendly platform for accessing

university services can lead to a disjointed and inefficient process, resulting in decreased member involvement and missed opportunities.

1.3 Proposed Solution

The proposed solution is to develop a comprehensive community services application exclusively for UJ members. This application will serve as a centralized platform, offering various essential services to enhance the university experience.

These services include exclusive offers, lost items management, volunteer opportunities, training courses, workshops and events announcements, and communication between students through live chat.

1.4 Aim and Objectives

The solution aims to develop and implement a community services application for UJ members, providing them with a centralized platform to access non-academic essential services. This aim can be achieved through the following objectives:

- Centralize access to essential university services.
- Increase the number of beneficiaries of the exclusive offers and events.
- Facilitate student club membership.
- Facilitates access to the clinic schedule and appointment booking.
- Optimize the process of announcing and managing lost and found items in an efficient and organized manner.
- Enhance communication between students.
- Facilitate the acknowledgment and participation of volunteering opportunities.

1.5 Novelty/ Contribution

The Mojtamai mobile application aims to assist UJ members in quickly accessing essential university services through a single platform. Moreover, it distinguishes itself from other mobile applications across Saudi Arabian universities by combining all non-academic crucial services in one application.

1.6 Methods

1.6.1 Flutter Framework

The Flutter framework will be used to facilitate mobile development. Flutter is a versatile cross-platform mobile application development tool that is both free and open-source, generously provided by Google. It allows developers to create mobile, web, and desktop applications using a single codebase. Additionally, Flutter is an object-oriented language written in Dart, which boasts a syntax comparable to that of Java and C# [2].

1.6.2 Laravel Framework

For the admin panel web page, the Laravel framework will be used. It is an open-source PHP framework recognized for its simplicity, elegance, and developer-friendly characteristics. It adheres to the MVC (Model-View-Controller) architectural pattern, which separates concerns and facilitates code organization. [3].

1.6.3 Programming Languages

▪ Dart

To build the mobile application in Flutter, Dart languages will be used as a programming language. Dart is an object-oriented language with a syntax similar to Java or JavaScript. The purpose of Dart is to develop the logic and functionality of the Flutter app [4].

- **PHP**

Developing web applications using the Laravel framework can be done efficiently with PHP, a powerful scripting language. It is an excellent choice for creating dynamic web pages and server-side scripting, and it is designed to cater to the specific needs of web developers. PHP can be relied on to create a website that is efficient and dynamic for users [5].

1.6.4 Real-Time Communication

- **Push Notification**

push notifications are a service provided by Backend-as-a-Service. Using Firebase Cloud Message is an intelligent way to send push notifications to users of an application registered on Firebase. This Firebase service allows sending push notifications to a single device or user, to a group of users with a common topic, or to a user segment based on analytical data previously collected by Firebase [6].

- **Live Chat**

Given its ability to manage real-time data in the database, Firebase has been considered the optimal solution for chat messaging needs. With its swift and seamless data exchange to and from the database, this NoSQL platform can automatically sync real-time data across all devices without requiring manual refreshments. [6], [7].

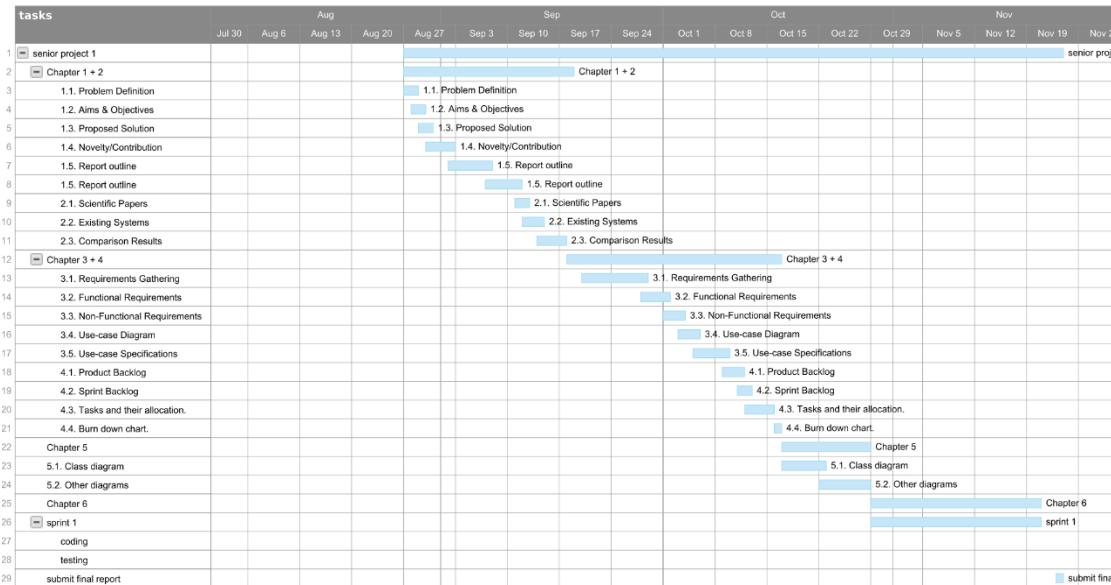
1.6.5 Recommendation System

The proposed system includes a Recommendation System module designed to provide personalized offer suggestions based on user preferences. This module aims to enhance the user experience by offering tailored recommendations that align with the user's preferences.

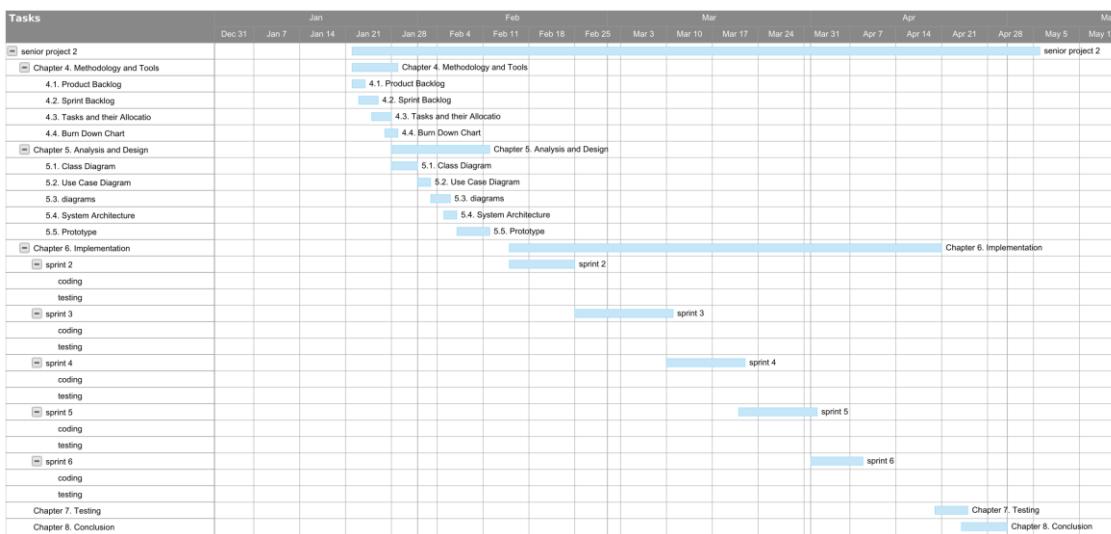
While the Recommendation System described here does not explicitly employ advanced artificial intelligence (AI) techniques, it is important to note that it still provides valuable recommendations based on user-selected preferences.

1.7 Project Plan (Gantt Chart)

The project's objectives, as represented in *Figure 1*, have been formulated with a comprehensive plan that will be executed over two academic terms.



(a) First Semester



(b) Second Semester

Figure 1 Gantt Chart for milestone and deadlines.

1.8 Conclusion

This project can potentially benefit UJ members by reducing their time and effort. Developing an app for the UJ community can transform how members interact, collaborate, and navigate their university experience. This app can meet UJ members' specific needs and challenges by utilizing technology and creating a more connected, supportive, and enriching campus environment.

Chapter 2: Related Works

2.1 Introduction

Having a university community provides a variety of ways for universities to fulfill their role as "sites of citizenship." These include contributing to community social and economic infrastructure and promoting equity and diversity in higher education. Additionally, it represents a changed social and economic environment in which ideas of "community" have replaced wider concepts of the social [8].

This chapter focuses on the current situation at UJ, and the platforms utilized to deliver its services. Specifically, it examines the provision of exclusive offers, training courses, workshops, conferences, clinic schedules, lost and found management, and volunteering opportunities. Additionally, the chapter explores existing software solutions employed by Saudi universities, international institutions, and university communities to inspire the development of a customized community platform for UJ.

2.2 Current Situation

UJ services are provided across multiple platforms. This section specifies the platform where each service is available and briefly explains it.

2.2.1 Exclusive Offers

The university's website features a range of enticing offers, but some category buttons appear inactive, leaving users uncertain about their click ability. Exclusive deals are presented solely as images without any filtering or search options. In addition, the offers are not up to date.

2.2.2 Events

It's common for announcements about events to be delivered via email. However, with the volume of emails received each day, they can easily be missed or buried. Additionally, these

messages may sometimes end up in the junk mail folder, resulting in the possibility of important information being overlooked.

2.2.3 Clinic Schedule

The clinic schedule conveniently arrives through email, but it can be a hassle to search through unrelated emails each time an appointment needs to be booked. Also, finding a specific doctor can prove challenging as their information is listed in a table.

2.2.4 Lost and Found Items Management

The management of lost and found items at UJ can be challenging. Members currently utilize various individual platforms to make announcements, which can result in some university members not having access to these groups. Additionally, there have been cases where items were handed over to security without any announcement, making it necessary to search through different departments.

2.2.5 Volunteering Opportunities

Volunteering opportunities shared through WhatsApp groups have limited reach, as they are visible only to group members. This excludes non-members from potential volunteering opportunities.

2.2.6 Student Clubs

Student clubs in UJ's community may be known from various sources such as club leaders, members, official club social media accounts, or communication among colleagues. However, because of unknown registration deadlines, students might miss specific features and activities provided by the clubs.

2.2.7 Psychological Guidance

Some students may face challenging psychological situations that are often sensitive and not usually discussed openly. This can make it difficult for them to seek the necessary help they need or even be unaware of the availability of psychological counselors who can offer support.

2.3 Existing System

This part has been designed to review similar software of the services provided by universities for their members and the university community in detail, inspiring the development of the UJ community.

▪ MyKAU

It is the official King Abdulaziz University mobile application. It enables members to access MyKAU E-Services quickly and safely, like academic services, view student financial reports and employee salaries, clinic reservations, and exchange messages between faculty members and students or vice versa [9].

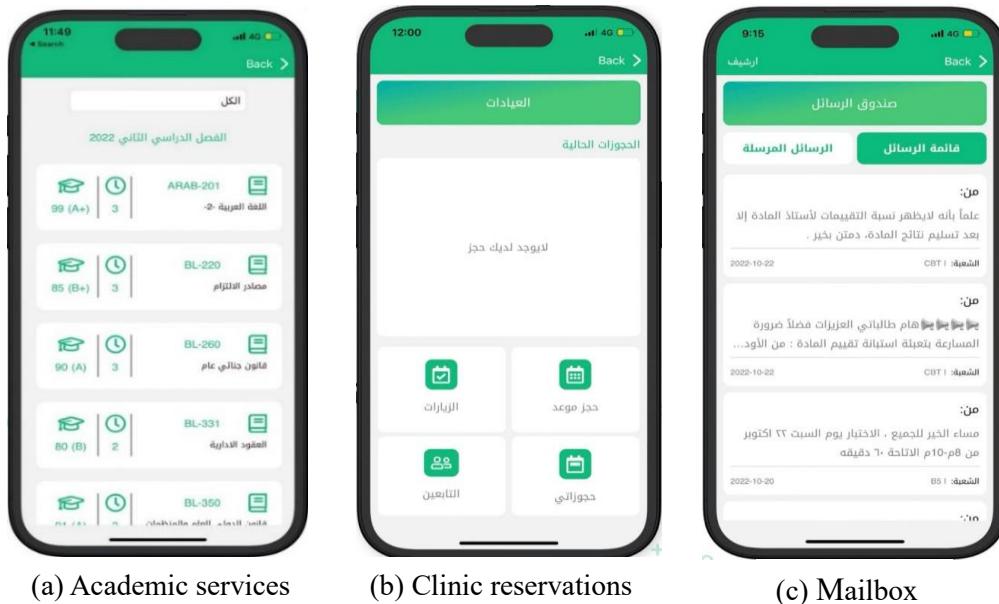
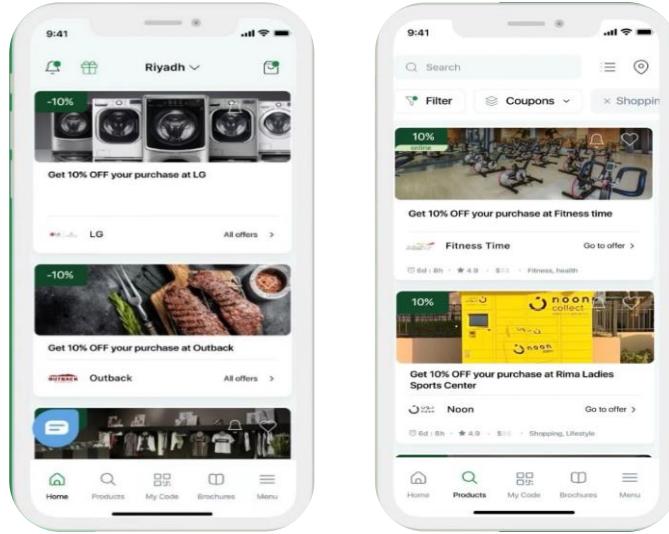


Figure 2 MyKAU Mobile Application UI

- **KAU Loyalty**

KAU Loyalty is a mobile application designed for King Abdulaziz University members who can have their account by ID to get exclusive offers like instant discounts, digital coupons, the latest promotions, and online discounts from 700+ brands in Saudi Arabia [10].



(a) Home page

(b) Search page

Figure 3 KAU Loyalty Mobile Application UI

- **UQU**

UQU is The Umm Al-Qura University mobile application for all university members. It displays the essential information about the university, allows viewing the latest news and events, training courses, workshops, conferences, and university community held at the university, and allows login and review to the electronic portal that provides academic services [11].

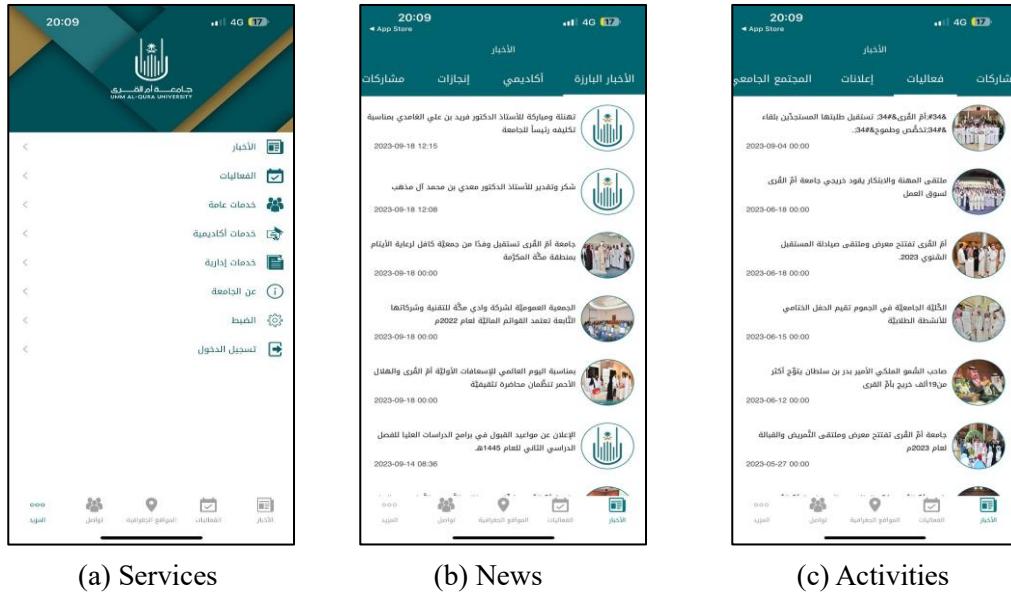
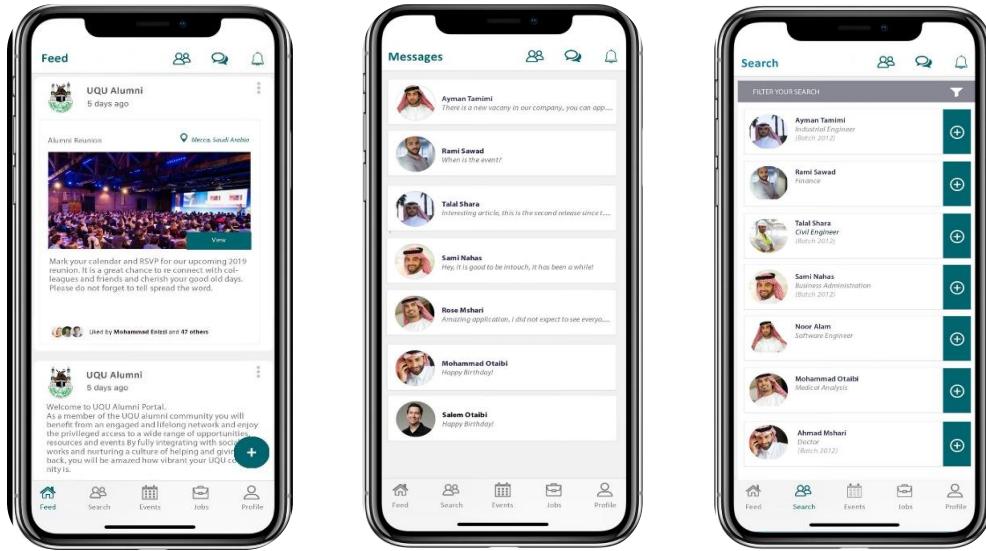


Figure 4 UQU Mobile Application UI

- **UQU Alumni**

UQU Alumni is The Umm Al-Qura University mobile application that allows users to communicate and stay in touch with classmates and friends, meet people, and find new ways to connect with the Alumni Associations and display events [11].



(a) Feed

(b) Messages

(c) Search

Figure 5 UQU Alumni Mobile Application UI

▪ Nafie Program

Nafie Program is a website for King Saud University members with exclusive offers and discounts in various fields [12].



Figure 6 Nafie Program Website UI

- **KFU Website**

KFU website for King Faisal University members has different icons for their services.

KFU Training provides a package of training courses, training programs, workshops, and a group of seminars and lectures that allow students to register and print certificates in cooperation with the deanships of the university. Mazaya Program displays exclusive offers and has a feature that lets the companies add exclusive offers to the university. In addition to KFU Volunteering, it is inactive [13].

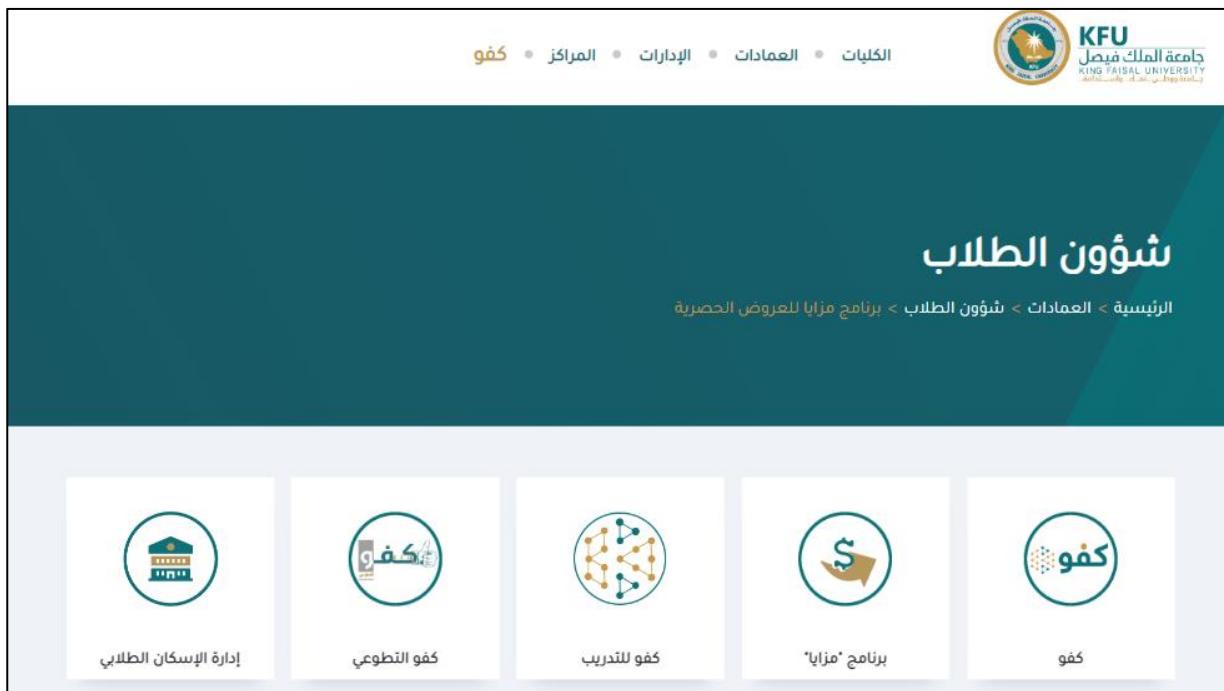


Figure 7 KFU Website UI

- **Mazaya Platform**

Mazaya Platform is a website for Taif University members. It displays exclusive offers and discounts in various fields, and they have a feature that lets companies add exclusive offers to the university [14].

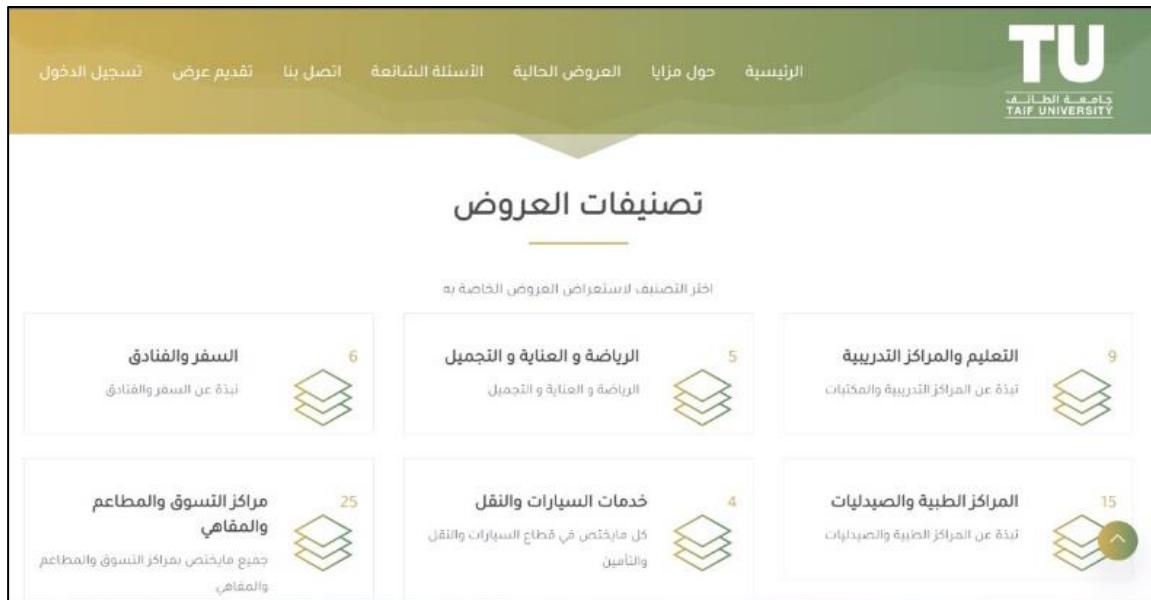


Figure 8 Mazaya Platform Website UI

▪ Maharat Platform

Maharat Platform is a website platform for Taif University students that combines digital self-paced learning and virtual learning in an integrated environment based on interactive learning methods and motivational game techniques, providing training courses and workshops [15].

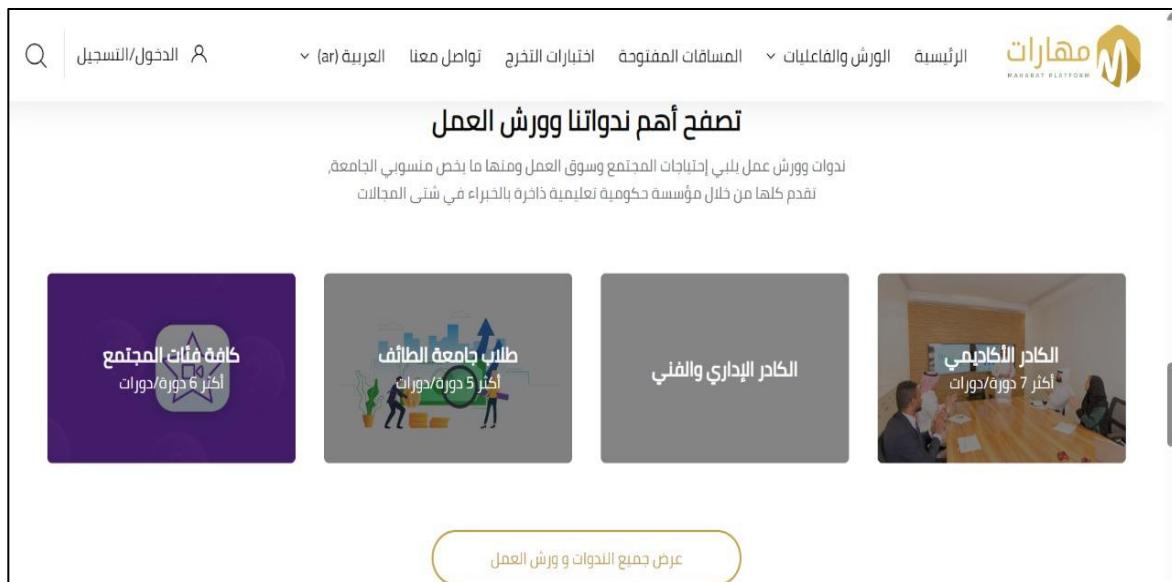


Figure 9 Maharat Website UI

- **KKUx**

KKUx is a website platform for King Khalid University students that provides all the skills necessary to prepare students for future jobs by providing training courses and programs and obtaining a certificate accredited by the university [16].

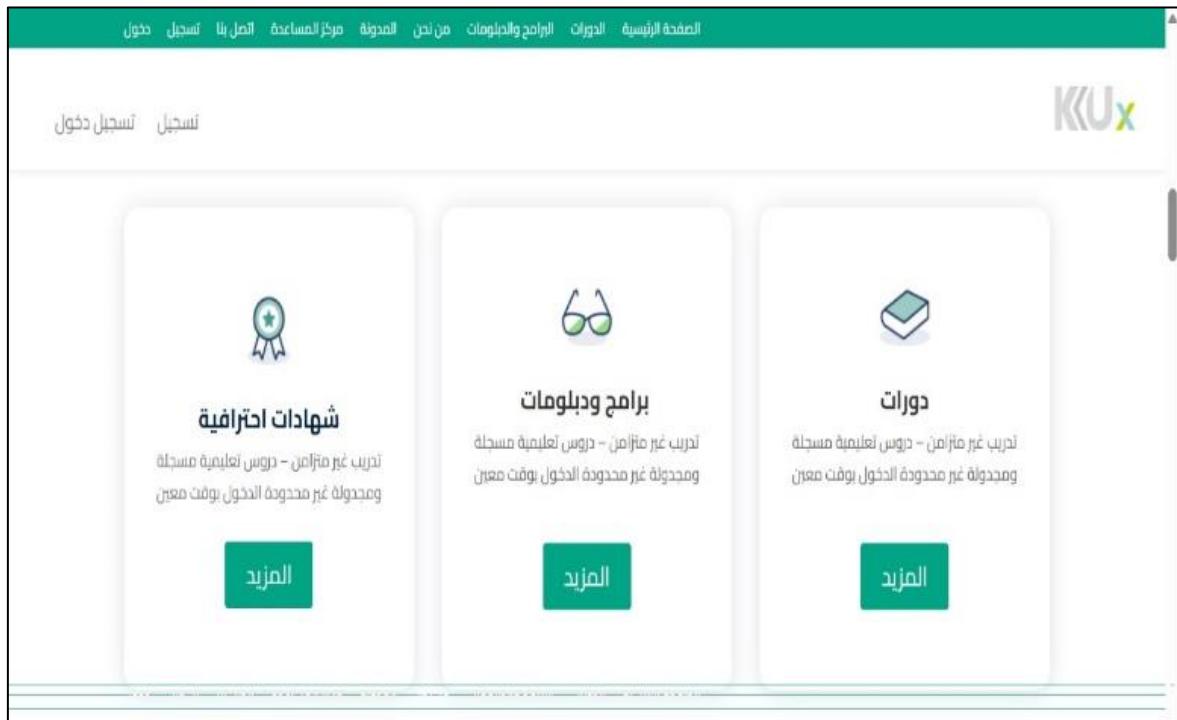


Figure 10 KKUx Website UI

- **AASTMT Student Portal**

AASTMT Student Portal is a mobile application designed for members of the Arab Academy for Science, Technology, and Maritime Transport. Besides academic services, the university community provides training courses, workshops, conferences, events, clinic reservations, and communication between members through live chat [17].

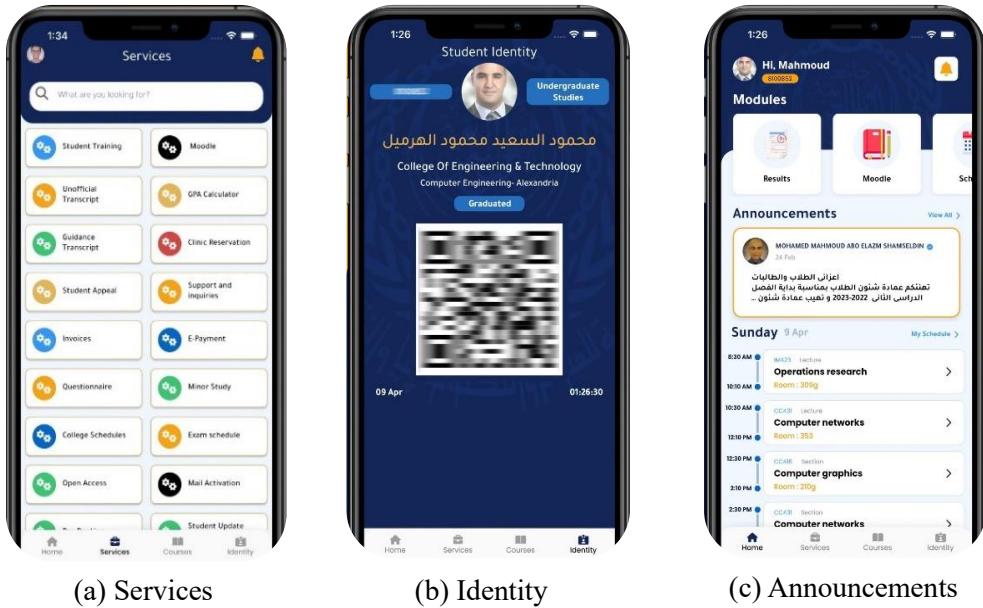


Figure 11 AASTMT Student Portal Mobile Application UI

▪ University of Manchester

The University of Manchester is a website for the University of Manchester students.

Besides academic services, the university community provides services like charity support and volunteering, arranging entertainment events like festivals and concerts, and arranging workshops and training courses [18].



Figure 12 University of Manchester Website UI

▪ London's Global University

Besides academic services, London's Global University is a website where university members can announce events and activities and join student clubs and training courses [19].

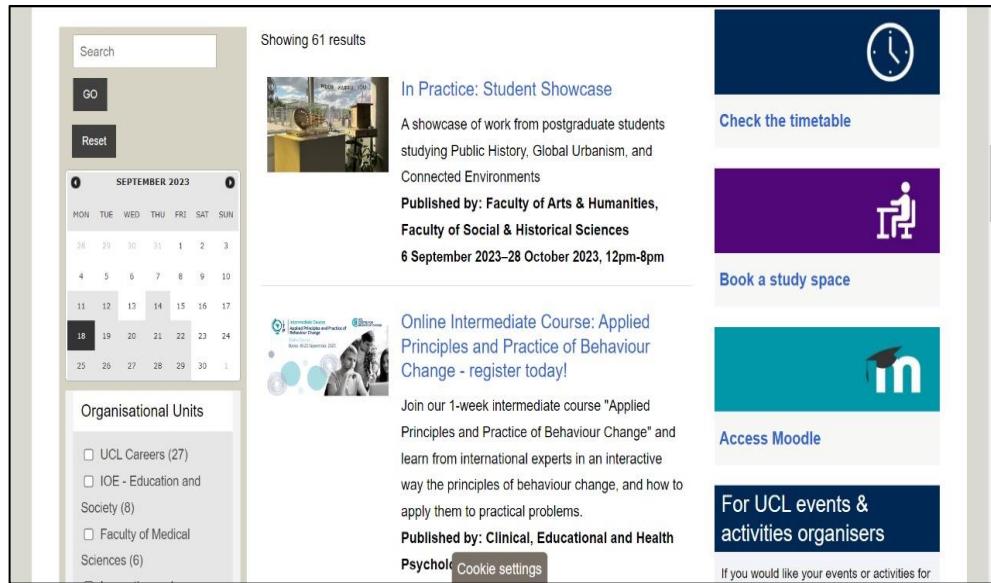


Figure 13 London's Global University Website UI

▪ University of Kent

The University of Kent is a mobile application for the University of Kent students. Besides academic services, it provides information about events and reads essential news [20].

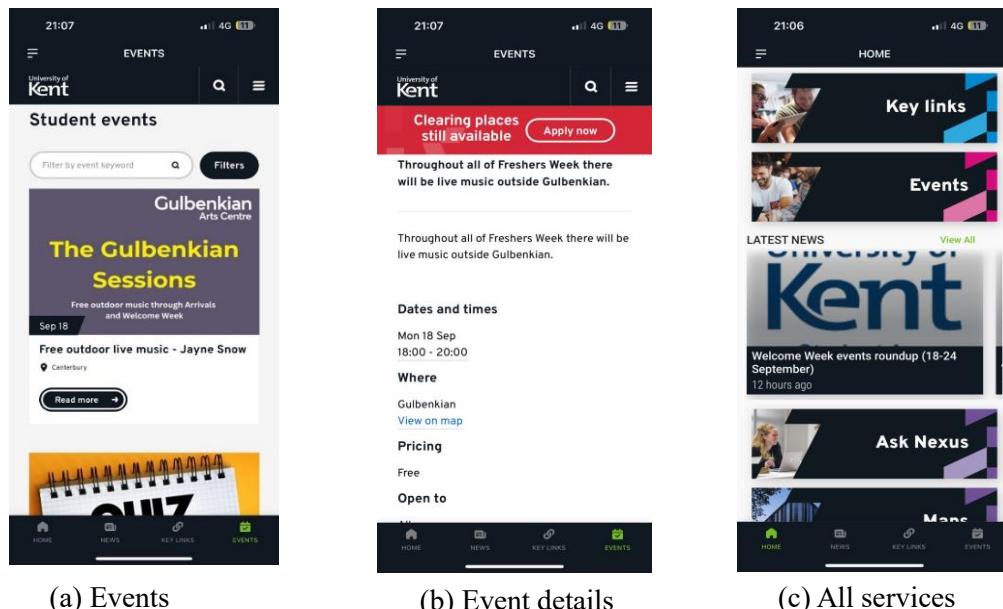


Figure 14 University of Kent Mobile Application UI

▪ Axenda Hubs

Axenda Hubs is a mobile application for all Saudi Universities. It lets students build and manage their student clubs and shows other students' clubs, events, and registers [21].

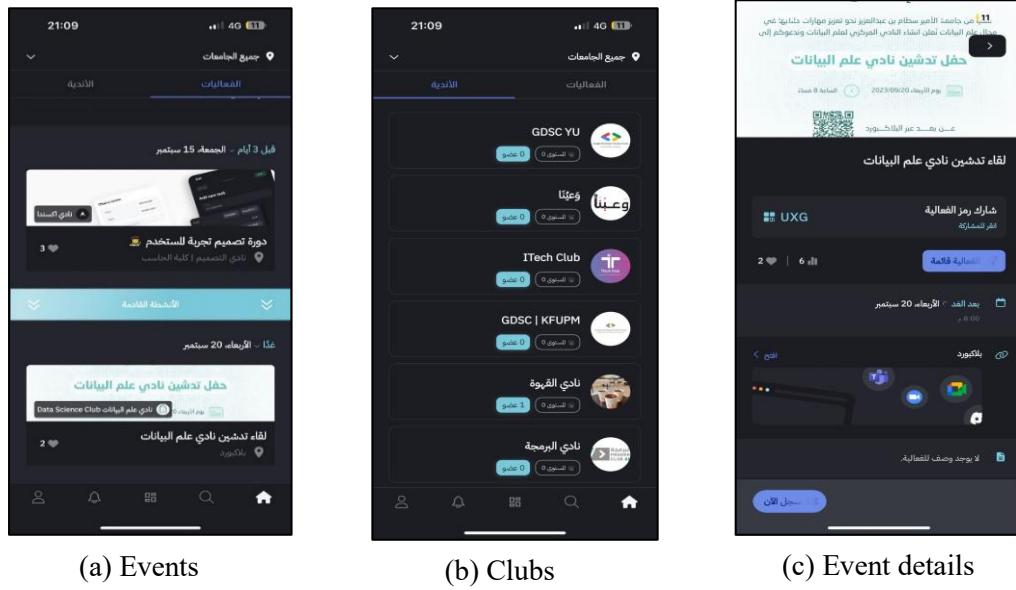


Figure 15 Axenda Hubs Mobile Application UI

▪ CampusGroups

CampusGroups lets schools and universities create a private community for their students, and it has features that include upcoming events, event registrations, and live chat [22].

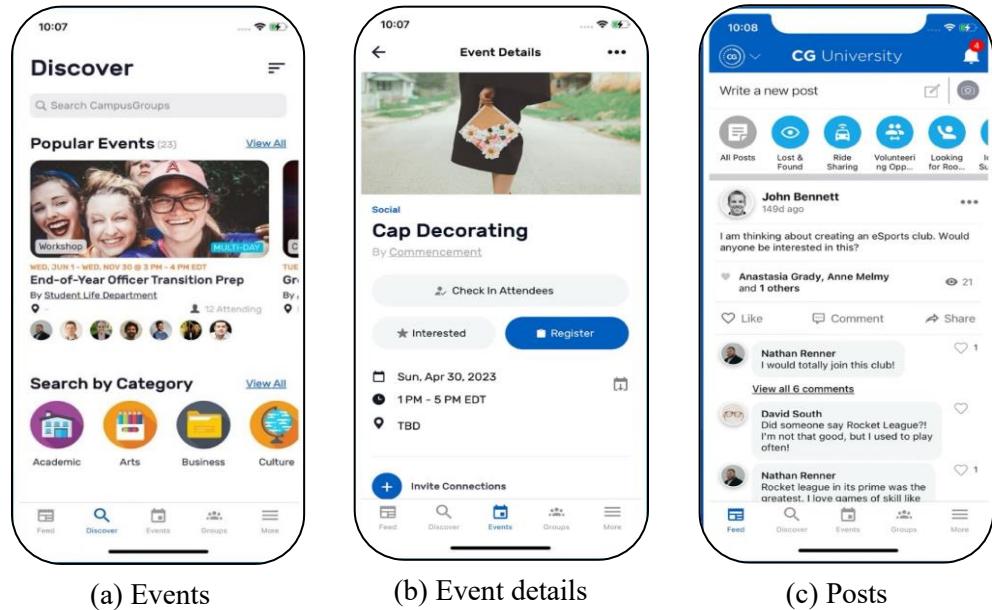


Figure 16 CampusGroups Mobile Application UI

- **Piogroup:**

Piogroup is a website that develops a university community, accesses all the community events that interest them and various financial offerings and loans, and allows eligible members to create, manage, and promote their student club events [23].

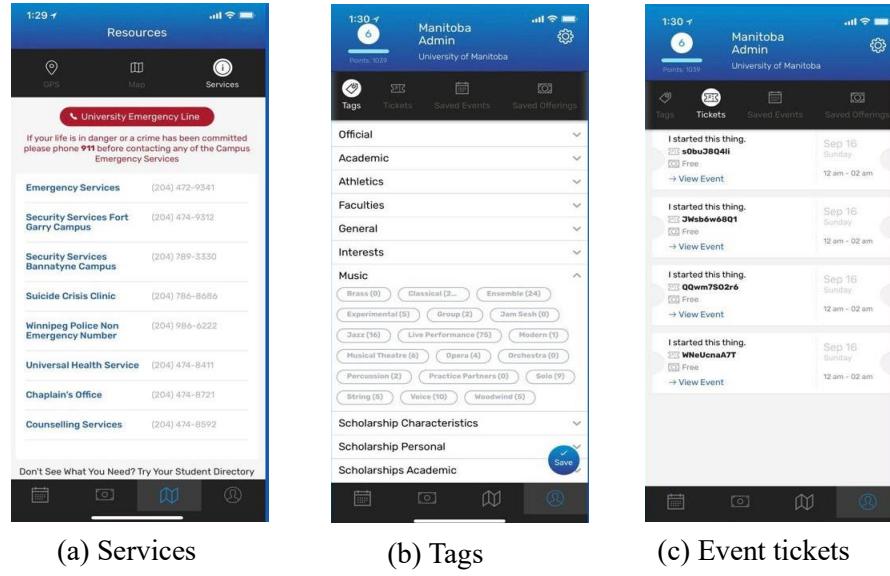


Figure 17 Piogroup Mobile Application UI

2.4 Comparison Result

This section presents a comparative analysis of the services offered by universities to their members in Saudi Arabia, as illustrated in *Table 2*. Furthermore, a comparison of services provided by universities abroad is included *Table 3*, and an overview of services offered by university communities is indicated *Table 4*.

These comparisons have inspired the Mojtamai project, which aims to improve the quality of the services provided to university members in Saudi Arabia.

Table 2 Similar software of Saudi universities

Name	Mobile Application/ Website	Services	Inspiration for Mojtamai Project
myKAU	Mobile Application	-Mostly academic services -View student financial reports and the salary of employees -Clinic reservations	-Clinic reservations
KAU Loyalty	Mobile Application	-Display exclusive offers -Provide online discounts -Provide digital coupons	-Display exclusive offers
UQU	Mobile Application	-Mostly academic services -Display news -Provide training courses, workshops, conferences, events	-Provide training courses, workshops, conferences, events
UQU Alumni	Mobile Application	-Communicate between members -Display events	-Communicate between members -Display events
Nafae Program	Website	-Display exclusive offers	-Display exclusive offers
KFU	Website	-Display exclusive offers -Let the companies add exclusive offers to the university -Training programs and seminars supervised by the deanship -Students can print certificates	- Display exclusive offers -Let the companies add exclusive offers to the university by officials -Training programs and seminars supervised by the admins -Students can print certificates
Mazaya Platform	Website	-Display exclusive offers -Let the companies add the exclusive offers to the university	-Display exclusive offers -Let the companies add exclusive offers to the university by officials

Maharat Platform	Website	-Provide training courses and workshops	-Provide training courses and workshops
KKUx	Website	-Provide training courses for all members	-Provide training courses for all members

Table 3 Similar software of abroad universities

Name	Mobile Application/ Website	Services	Inspiration for Mojtamai Project
AASTMT Student Portal	Mobile Application	-Mostly academic services -Live chat -Provide training courses, workshops, conferences, events -Clinic reservations	-Live chat -Provide training courses, workshops, conferences, events -Clinic reservations
University of Manchester	Website	-Mostly academic services -Charity support and volunteering -Provide and arrange entertainment events like festivals and concerts -Provide and arrange workshops and courses	-Provide and arrange workshops and courses -Provide volunteering opportunities
London's Global University	Website	-Mostly academic services -Provide events and activities -Join student clubs	-Provide events and activities -Join student clubs
University of Kent	Mobile Application	-Mostly academic services -Announce events -Display news	-Announce events

Table 4 Similar software of University Communities

Name	Mobile Application/ Website	Services	Inspiration for Mojtamai Project
Axenda Hubs	Mobile Application	-Provide activities -Students and student club leaders engage and manage their clubs, events, and members	-Provide activities -Provide students with clubs
CampusGroups	Website	-Provide events -Campus resources, maps, links -Live chat -Make a group chat	-Provide events -Live chat -Make a group chat
Piogroup	website	-Provide events -Engage and manage student clubs	-Provide events -Provide students with clubs

2.5 Conclusion

This chapter discusses software similar to the proposed problem in detail, and the differences are compared in a table. Additionally, it was thoroughly examined and analyzed to gain inspiration throughout the application development journey.

Chapter 3: Requirement Gathering and Analysis

3.1 Introduction

This chapter presents a formal specification of the system requirements, comprising both Functional Requirements (FRs) and Non-Functional Requirements (NFRs). The requirements are modeled using a use case diagram, with the most important specifications included to provide a comprehensive understanding of the requirements. Furthermore, a prototype of the system is developed using Figma, which enables the visualization of the system and ensures that the requirements are effectively met during the development phase. This approach will help to deliver a high-quality design that meets the needs of the stakeholders.

3.2 Functional Requirement

FRs describe the desired behavior of a system, component, or product. It articulates the specific services, tasks, or functions that the system should be capable of performing without specifying the implementation details. These requirements outline the expected behaviors and operations that the system should exhibit to fulfill its intended purpose [24].

The FRs listed *Table 5* are accompanied by their priority ranks using the MoSCoW strategy. Analysts and stakeholders use the MoSCoW technique for collaboratively prioritizing requirements. According to the Moscow mechanism, the list of conditions can be classified into the following four priority categories:

- **M – Must have.** In this group, requirements must be contained in the project. Failure to deliver these requirements means the whole project would be a failure.
- **S – Should have.** A high-priority feature that is not critical to launch. But it is supposed to be essential and of high value to users. Such requirements fill the second place on the priority list.

- **C – Could have.** This group contains the desirable requirement but not the necessary one.

But these requirements are less critical than those in the “should have” group.

- **W – Won’t have.** A requirement that will not be implemented in a current development but may be included in a future stage [25].

These priority ranks are assigned based on the needs of users and the analysis of similar applications conducted previously. The priority reflects the relative importance of each requirement to the user.

Table 5 Functional Requirements

Requirement ID	Functional Requirements Statement	Priority
FR1	Login	
FR1.1	The system must allow the user to log in to their account using UJ ID and password.	M
FR2	Profile	
FR2.1	The system should allow the user to edit their information (hobbies, interests, what they can add to the community).	M
FR2.2	The system should allow the user to add preference categories of exclusive offers.	M
FR3	Exclusive Offers	
FR3.1	The system should recommend exclusive offers for the user based on their preferences.	S
FR3.2	The system must display all exclusive offers currently available.	M
FR4	Volunteering Opportunities	
FR4.1	The system should display all available volunteering opportunities.	M
FR4.2	The system should allow the user to register for an available volunteering opportunity.	S
FR5	Student Clubs	
FR5.1	The system should display university student clubs	M

FR5,2	The system should allow the user to register in student clubs	S
FR6	Clinic Schedule	
FR6.1	The system should display the schedule of the clinic	M
FR6.2	The system should allow the user to book an appointment at the clinic	M
FR6.3	The system should allow the user to search in the clinic schedule	M
FR7	Events	
FR7.1	The system should allow the user to search for an event	S
FR7.2	The system should display all available events.	M
FR7.3	The system should allow users to register for an event.	M
FR8	Lost and found items	
FR8.1	The system should allow the user to search for a lost item.	S
FR8.2	The system should allow the user to search for found items.	S
FR8.3	The system should display all lost items	M
FR8.4	The system should display all found items	M
FR8.5	The user should be able to report a lost item.	M
FR8.6	The user should be able to report the found item.	M
FR9	Student Activity	
FR9.1	The system should display available student activates	M
FR9.2	The system should allow the user to search for student activity	S
FR9.3	The system should allow users to create a new student activity.	M
FR10	Study Group	
FR10.1	The system should display all available study groups	M

FR10.2	The system should allow the user to create a study group	M
FR10.3	The system should allow the user to search for study group	S
FR11	Live chat	
FR11.1	The system should allow the user to contact other users through live chat.	M
FR12	Push notification	
FR12.1	The system should provide push notifications to users' devices to inform them of upcoming events, exclusive offers, or activities.	C
FR13	Save list	
FR13.1	The system should allow the user to add an event, volunteering opportunity, study group or activity to their save list.	C
FR13.2	The system should allow the user to remove an event, volunteering opportunity, study group or activity from their save list.	C
FR14	Psychological guidance	
FR14.1	The system should display psychological guidance.	C
FR15	Certificate	
FR15.5	The system should allow the user to display the attendance certificate	W
FR16	Log out	
FR16.1	The user should be able to log out from their accounts	M

Admin Panel

FR17	Login	
FR17.1	The system must allow the admin to log in to their account using their ID and password.	M
FR18	Manage exclusive offers	
FR18.1	The system should allow the admin to add new exclusive offers	M
FR18.2	The system should allow the admin to edit exclusive offers information.	M
FR18.3	The system should allow the admin to remove exclusive offers.	M

FR19	Manage event	
FR19.1	The system should allow the admin to add a new event	M
FR19.2	The system should allow the admin to edit event information.	M
FR19.3	The system should allow the admin to remove the event	M
FR20	Manage volunteering opportunity	
FR20.1	The system should allow the admin to add new volunteering opportunities.	M
FR20.2	The system should allow the admin to edit volunteering opportunity information.	M
FR20.3	The system should allow the admin to remove a volunteering opportunity.	M
FR21	Manage clinic schedule	
FR21.1	The system should allow the admin to add the clinic schedule for the week.	M
FR21.2	The system should allow the admin to edit the clinic schedule.	M
FR21.3	The system should allow the admin to remove the clinic schedule.	M
FR22	Manage student clubs	
FR22.1	The system should allow the admin to add a new student club.	M
FR22.2	The system should allow the admin to edit the student club information.	M
FR22.3	The system should allow the admin to remove a student club.	M
FR23	Log out	
FR23.1	The admin should be able to log out from their accounts	M

3.3 Non-Functional Requirement

NFRs describe the characteristics and qualities of a system or software application. The priority also follows the MoSCoW technique, but only Must (M) and Should (S) are typically used for NFRs.[26][27]. *Table 6* displays NFRs for the proposed solution. It is mainly considered as the

quality attribute that will guarantee user satisfaction. the system's quality attributes as the following:

- **User interface (UI):** the access points where users interact with the system's design [28].
- **Security:** How well a system protects against unauthorized access and data breaches[29].
- **Usability:** How easy a system is to use[29].
- **Maintainability:** How easy it is to update and modify the system[29].
- **Performance:** How fast a system should respond to requests[29].
- **Interoperability:** Systems that communicate successfully through compatible formats and protocols[30].
- **Availability:** This denotes how often the system is accessible[31].

Table 6 Non-Functional Requirements

Requirement ID	Non-Functional Requirements Statement	Priority
NFR1	User Interface	
NFR1.1	The interfaces will be easy to use and user-friendly.	S
NFR2	Security	
NFR2.1	The application must ensure that user confidential data cannot be accessible by unauthorized persons.	M
NFR2.2	The application will ask permission to access the camera.	M
NFR3	Interoperability	
NFR3.1	The application should be able to run on IOS and Android.	M
NFR4	Performance	

NFR4.1	The user should be able to complete all system functions successfully within the high volume of users and maintain quick response times.	S
NFR5	Usability	
NFR5.1	The application should be easy to learn and use.	M
NFR5.2	The application should be usable for users with disabilities.	S
NFR5.3	The application should be designed to comply with accessibility standards.	S
NFR6	Maintainability	
NFR6.1	The application should support easy updates and patches without significant downtime.	M
NFR7	Availability	
NFR7.1	The application should be accessible 24/7.	S

3.4 Design Constraints

A. Ethics Standards:

Mojtamai application prioritizes user well-being by consciously avoiding design elements that exploit psychological vulnerabilities. It promotes healthy user behaviors and strives to create a positive and empowering user experience. The application is committed to transparency, providing clear explanations of its system's design choices to ensure user comprehension and foster trust.

B. Technical Design Constraints:

In software development, adhering to technical constraints imposed by the infrastructure, platform, or tools is crucial. Such limitations may include hardware, software compatibility, and performance. Consequently, Mojtamai must consider the platform's compatibility and limitations.

For instance, crafting a mobile application with restricted processing power and memory necessitates optimizing the design to ensure efficient resource utilization and smooth functionality.

C. Compatibility Design Constraints:

Software designers must ensure compatibility with existing systems, data formats, third-party software, devices and screen sizes, web browsers, required software, hardware components, and different software versions.

D. Operational Design Constraints:

When designing software, operational considerations such as scalability, maintenance, and efficiency are essential. The design must be able to handle increased loads, user growth, or expanded functionalities without disruptions or performance degradation. Designs should be modular, well-documented, and easily upgradable. The performance and efficiency of the design should be optimized by reducing response times, minimizing resource usage, and improving system reliability.

3.5 Figma Prototype

A prototype is a preliminary version of a product or system that allows stakeholders to evaluate its feasibility and gather feedback before full-scale development. It helps validate design concepts, collect user feedback for improvement, reduce risks by identifying and addressing issues early on, and facilitates stakeholder communication and collaboration. Organizations can increase the chances of developing successful and user-centric final products or systems by testing and refining prototypes[32].

The interface designs are as follows:

- Login page:

The login interface of the application comprises a single button that grants the user access to the application services upon successfully entering their current UJ ID and password.



Figure 18 Login Page

- **Profile page:**

The application's profile page allows users to update their personal information easily. This information can be used in other application features to provide a more personalized experience.



Figure 19 Profile Page

- **Home page**

The home page in *Figure 20* presents users with recommended exclusive offers and information on the latest system services and events of the day. This provides users with a comprehensive overview of the system's offerings. The home page's user-friendly design ensures users can easily navigate the various sections and quickly find the necessary information.



Figure 20 Home Page

- **Services Page**

On the Services page, see *Figure 21*; various icons represent the distinct services user can utilize to their advantage. Each icon is a gateway to the corresponding service details, enabling users to explore and evaluate the available options.



Figure 21 Services Page

▪ Exclusive Offers Pages

On the exclusive offers page, categories of offers are displayed as icons, each leading to a page showcasing one or more exclusive offers. Selecting an exclusive offer will take the user to a page with detailed information, including location, expiration date, and discount percentage.

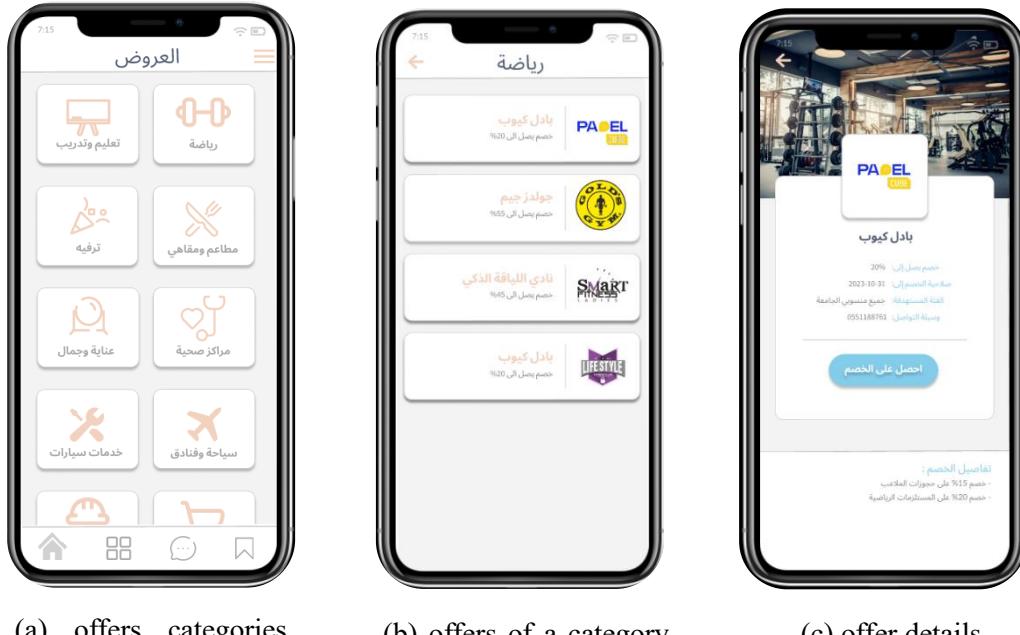


Figure 22 Exclusive Offers Pages

- **Volunteering Opportunities Page**

The page volunteering opportunities showcase the available options and a registration button to participate. This page aims to provide comprehensive information about various volunteering roles and encourage interested individuals to register.



Figure 23 Volunteering Opportunities Page

- **Events Page**

The event page features a filter to differentiate between various events, such as training courses and workshops, conferences, and other relevant occasions. Each event is accompanied by a registration form and its relevant details.



Figure 24 Events Page

▪ Students' Clubs Pages

The student club's page features icons representing various UJ clubs, each linked to their respective information page.



(a) available students' clubs

(b) club details

Figure 25 Students Clubs Pages

- **Study Group Pages**

The study group page displays the available study groups created by students. Each study group can be joined and saved for later participation. Also, students can create new study groups.



Figure 26 Study Groups Pages

- **Student Activities Pages**

The student activity page displays the available activities created by students. Each activity can be joined and saved for later participation. Also, students can create new activities by filling in the required information.



(a) available student activities (b) create student activity

Figure 27 Student Activities Pages

▪ Clinics Page

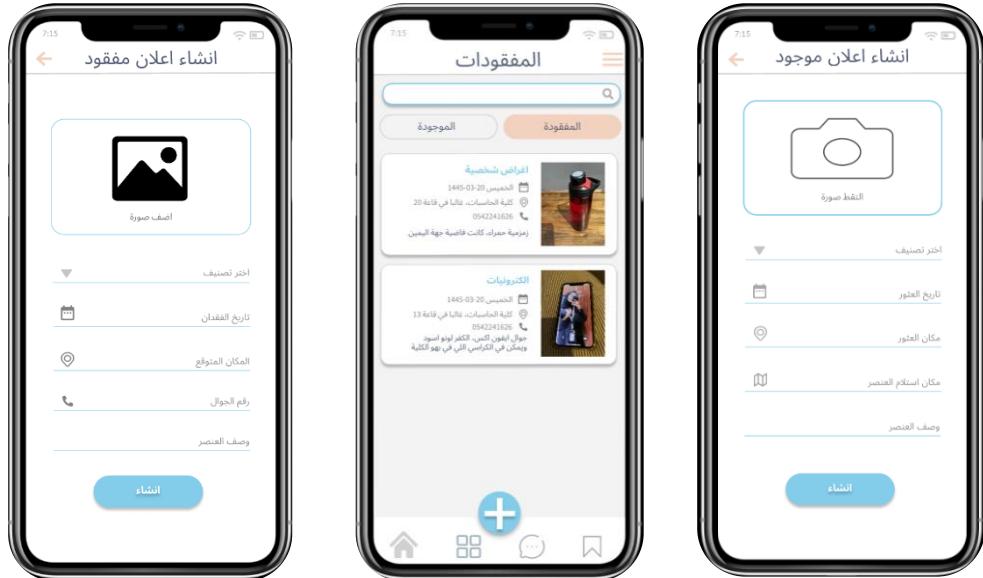
The clinic page features a filter allowing users to distinguish between different branches of the clinic schedule. This filter provides information regarding each clinic branch and a button that links to a booking form for scheduling appointments.



Figure 28 Clinics Page

▪ Lost and Found Items Pages

The lost and found page includes a filter that distinguishes between lost and found items, thereby facilitating the display of reported items. Users can click the plus button to create a report and provide the requisite information.



(a) create a lost item report (b) lost and found items (c) create a found item report

Figure 29 Lost and Found Items Pages

▪ Chat Pages

The chat page exhibits a comprehensive list of available users for communication.



(a) all chats

(b) chat

Figure 30 Chat Pages

- **Saved List page**

The Saved list page displays saved study groups and student activities in cards with names and scheduled times.



Figure 31 Saved List Page

3.6 Use-Case Diagram

The UML's behavior diagram is called the use case diagram, which outlines the software's functional requirements. It provides a clear understanding of how the system should operate. Use case diagrams to help comprehend the system's behavior. *Figure 32 Use-Case Diagram* Displays the primary use case for the proposed solution. It has two main actors: "Admin" and "User". The "Admin" is an employee responsible for managing the university services provided to the system user. However, the "User" is UJ employees or UJ students who can benefit from the services provided by the system [33].

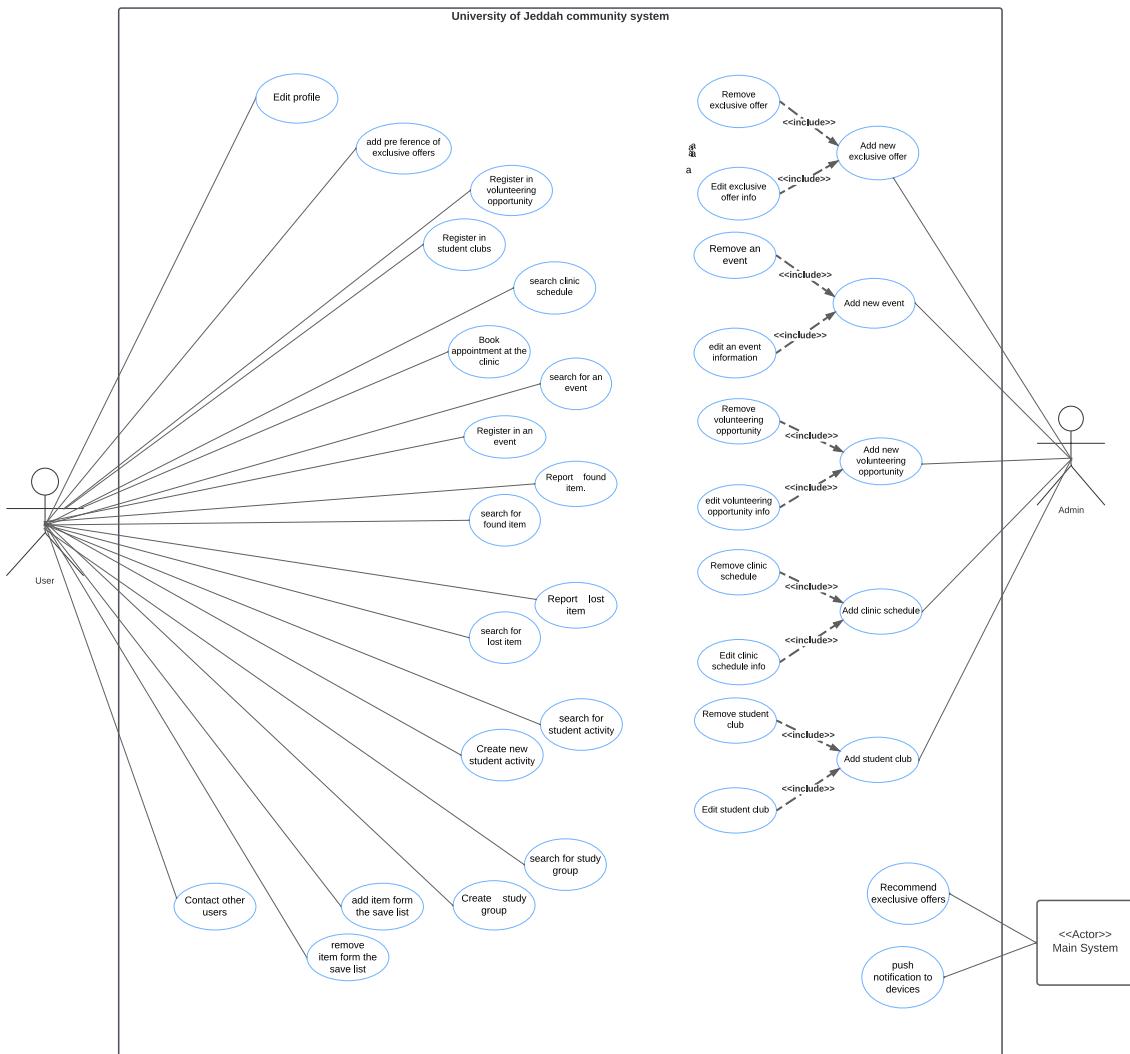


Figure 32 Use-Case Diagram

3.7 Use-Case Specifications

A use case specification describes the system's functionality. It documents the interaction between the system and the actor. Simply put, it outlines how a user engages with the system and how the system responds to the user's actions [34].

Table 7 Use-Case Specification - Edit User Profile

Use Case Name	Edit user profile
Brief Description	The system should allow the user to edit their profile.
Actors	User

Related Use Case	-	
Precondition	The user must be logged in to the application.	
Post Condition	The system updated profile information.	
Flow of Activities	Actor	System
	2- User edit profile information.	1- Display all profile information.

Table 8 Use-Case Specification - Add new Exclusive Offer

Use Case Name	Add a new exclusive offer	
Brief Description	The system should allow the admin to add new exclusive offers.	
Actors	Admin	
Related Use Case	-	
Precondition	Admin must be logged in to the system.	
Post Condition	The system will display the new offer on the application for the users.	
Flow of Activities	Actor	System
	1- Admin fill in the required information for the new exclusive offer and add the new exclusive offer	2- Store the information in the database. 3- Update the exclusive offers in the admin panel. 4- Update the exclusive offers on the mobile application

Table 9 Use-Case Specification - Remove Exclusive Offer

Use Case Name	Remove exclusive offer	
Brief Description	The system should allow the admin to remove exclusive offers information.	
Actors	Admin	
Related Use Case	Add a new exclusive offer.	
Precondition	Admin must be logged in to the system.	
Post Condition	-The system will display the updated list of offers on the application for the users -The system will remove the offer information	
Flow of Activities	Actor	System
	1- Admin removes a specific exclusive	2- Remove the offer information from the database. 3- Display the updated offer list in the panel. 4- Display an updated offer list to the user.

	offer from the panel.	
--	-----------------------	--

Table 10 Use-Case Specification - View Exclusive Offers

Use Case Name	View exclusive offers	
Brief Description	The system shall display all exclusive offers currently available.	
Actors	User	
Related Use Case	Add a new exclusive offer.	
Precondition	The user must be logged in to the application.	
Post Condition	The system displays valid available exclusive offers.	
Flow of Activities	Actor	System
	1- The user enters the exclusive offers page.	2- Displays valid available exclusive offers.

Table 11 Use-Case Specification - Add new Volunteering Opportunity

Use Case Name	Add new volunteering opportunity	
Brief Description	The system should allow the admin to add new volunteering opportunities.	
Actors	Admin	
Related Use Case	-	
Precondition	Admin must be logged in to the system.	
Post Condition	The system will display the new opportunities for users on the application.	
Flow of Activities	Actor	System
	1- Admin fills in the required information for the new volunteering opportunities and adds the volunteering opportunity.	2- Store the information in the database. 3- Display the updated volunteering opportunity list in the admin panel. 4- Display the updated volunteering opportunity list to the user,

Table 12 Use-Case Specification - Edit Volunteering Opportunity

Use Case Name	Edit volunteering opportunity
Brief Description	The system should allow the admin to edit information on volunteering opportunities.
Actors	Admin

Related Use Case	Add new volunteering opportunities.					
Precondition	Admin must be logged in to the system.					
Post Condition	<ul style="list-style-type: none"> -The system will display the updated list of volunteering opportunities on the application for the users. -The system will update the information on volunteering opportunities. 					
Flow of Activities	<table border="1"> <tr> <th>Actor</th> <th>System</th> </tr> <tr> <td>1- The admin edits specific information on volunteering opportunities.</td> <td> <ul style="list-style-type: none"> 2- Update the volunteering opportunity information in the database. 3- Display updated information on volunteering opportunities in the panel. 4- Display updated information on volunteering opportunities to the user. </td> </tr> </table>	Actor	System	1- The admin edits specific information on volunteering opportunities.	<ul style="list-style-type: none"> 2- Update the volunteering opportunity information in the database. 3- Display updated information on volunteering opportunities in the panel. 4- Display updated information on volunteering opportunities to the user. 	
Actor	System					
1- The admin edits specific information on volunteering opportunities.	<ul style="list-style-type: none"> 2- Update the volunteering opportunity information in the database. 3- Display updated information on volunteering opportunities in the panel. 4- Display updated information on volunteering opportunities to the user. 					

Table 13 Use-Case Specification - Remove Volunteering Opportunity

Use Case Name	Remove volunteering opportunity					
Brief Description	The system should allow the admin to remove information on volunteering opportunities.					
Actors	Admin					
Related Use Case	Add new volunteering opportunities.					
Precondition	Admin must be logged in to the system.					
Post Condition	<ul style="list-style-type: none"> - The system will display the updated list of volunteering opportunities on the application for the users. - The system will remove the information on volunteering opportunities. 					
Flow of Activities	<table border="1"> <tr> <th>Actor</th> <th>System</th> </tr> <tr> <td>1- Admin removes a specific volunteering opportunity from the panel.</td> <td> <ul style="list-style-type: none"> 2- Remove the volunteering opportunity information from the database. 3- Display the updated volunteering opportunity list in the panel. 4- Display the updated volunteering opportunity list to the user. </td> </tr> </table>	Actor	System	1- Admin removes a specific volunteering opportunity from the panel.	<ul style="list-style-type: none"> 2- Remove the volunteering opportunity information from the database. 3- Display the updated volunteering opportunity list in the panel. 4- Display the updated volunteering opportunity list to the user. 	
Actor	System					
1- Admin removes a specific volunteering opportunity from the panel.	<ul style="list-style-type: none"> 2- Remove the volunteering opportunity information from the database. 3- Display the updated volunteering opportunity list in the panel. 4- Display the updated volunteering opportunity list to the user. 					

Table 14 Use-Case Specification - Register in Volunteering Opportunity

Use Case Name	Register for a volunteering opportunity
Brief Description	The system should allow the user to register for an available volunteering opportunity.
Actors	User
Related Use Case	<ul style="list-style-type: none"> - Display all available volunteering opportunities. - Add a volunteering opportunity

Precondition	- The user must be logged in to the application - Admin added a volunteering opportunity	
Post Condition	-	
Flow of Activities	Actor	System
	1- The user chooses from available volunteering opportunities. 2- Enter the information required for registration and submit.	-

Table 15 Use-Case Specification - Register in Student Club

Use Case Name	Register in a student club	
Brief Description	The system should allow the user to register in student clubs.	
Actors	User	
Related Use Case	-	
Precondition	The user must be logged in to the application.	
Post Condition	-	
Flow of Activities	Actor	System
	1- User chooses from student clubs. 2- Enter the information required for registration and submit.	-

Table 16 Use-Case Specification - Add Clinic Schedule

Use Case Name	Add clinic schedule	
Brief Description	The system should allow the admin to add the clinic schedule.	
Actors	Admin	
Related Use Case	-	
Precondition	admin must be logged in to the system	
Post Condition	the system will display the new clinic schedule on the application for the users	
Flow of Activities	Actor	System
	1- Admin fills in the required information for the new clinic schedule. 2- Add the clinic schedule.	3- Store the information in the database. 4- Display the updated clinic schedule list in the admin panel. 5- Display the updated clinic schedule list to the user.

Table 17 Use-Case Specification - Edit Clinic Schedule

Use Case Name	Edit clinic schedule	
Brief Description	The system should allow the admin to edit clinic schedule information.	
Actors	Admin	
Related Use Case	Add clinic schedule	
Precondition	Admin must be logged in to the system.	
Post Condition	<ul style="list-style-type: none"> - The system will display the updated list of clinic schedules on the application for the users - The system will update the clinic schedule information 	
	Actor	System
Flow of Activities	1- Admin edits specific clinic schedule information.	<ul style="list-style-type: none"> 2- Update the clinic schedule information in the database. 3- Display updated clinic schedule information in the panel. 4- Display updated clinic schedule information to the user.

Table 18 Use-Case Specification - Remove Clinic Schedule

Use Case Name	Remove clinic schedule	
Brief Description	The system should allow the admin to remove clinic schedule information.	
Actors	Admin	
Related Use Case	Add clinic schedule	
Precondition	Admin must be logged in to the system.	
Post Condition	<ul style="list-style-type: none"> - The system will display the updated list of clinic schedules on the application for the users - The system will remove the clinic schedule information 	
	Actor	System
Flow of Activities	1- The admin removes a specific clinic schedule from the panel.	<ul style="list-style-type: none"> 2- Remove the clinic schedule information from the database. 3- Display the updated clinic schedule list in the panel 4- Display the updated clinic schedule list to the user.

Table 19 Use-Case Specification - Book an Appointment at the Clinic

Use Case Name	Book an appointment at the clinic
Brief Description	The system should allow the user to book an appointment at the clinic.

Actors	User	
Related Use Case	-	
Precondition	The user must be logged in to the application.	
Post Condition	-	
Flow of Activities	Actor	System
	2-The user book an appointment	1- Display all available appointments

Table 20 Use-Case Specification - Add New Event

Use Case Name	Add new event	
Brief Description	The system should allow the admin to add new events.	
Actors	Admin	
Related Use Case	-	
Precondition	Admin must be logged in to the system.	
Post Condition	The system will display the new event on the application for the users.	
Flow of Activities	Actor	System
	1- Admin fills in the required information for the new event. 2- Add the event	3- Store the information in the database. 4- Display the updated event list in the admin panel. 5- Display the updated event list to the user.

Table 21 Use-Case Specification - Edit Event

Use Case Name	Edit event	
Brief Description	The system should allow the admin to edit event information.	
Actors	Admin	
Related Use Case	Add new event	
Precondition	Admin must be logged in to the system.	
Post Condition	-The system will display the updated list of events on the application for the users -The system will update the event information	
Flow of Activities	Actor	System

	1- Admin edits specific event information.	2- Update the event information in the database. 3- Display updated event information in the panel. 4- Display updated event information to the user.
--	--	---

Table 22 Use-Case Specification - Remove Event

Use Case Name	Remove event	
Brief Description	The system should allow the admin to remove event information.	
Actors	Admin	
Related Use Case	Add new event	
Precondition	Admin must be logged in to the system.	
Post Condition	<ul style="list-style-type: none"> - The system will display the updated list of events on the application for the users -The system will remove the event information 	
Flow of Activities	Actor 1- Admin removes a specific event from the panel.	System 2- Remove the event information from the database. 3- Display the updated event list in the panel 4- Display the updated event list to the user

Table 23 Use-Case Specification - Register in an Event

Use Case Name	Register for an event	
Brief Description	The system should allow users to register for an event.	
Actors	User	
Related Use Case	-	
Precondition	<ul style="list-style-type: none"> - The user must be logged in to the application - There is an existing event added to the system 	
Post Condition	-	
Flow of Activities	Actor 2- The user chooses from available events 3- Enter the information for registration and submit	System 1- Display available events.

Table 24 Use-Case Specification - Report a Lost Item

Use Case Name	Report a lost item
Brief Description	The user should be able to report a lost item.

Actors	User	
Related Use Case	-	
Precondition	The user must be logged in to the application.	
Post Condition	The system displays the added lost item on the lost items page.	
Flow of Activities	<p>Actor</p> <p>1- The user enters the lost items page. 2- The user clicks on report lost item. 3- The user fills in the information about the lost item. 4- Submit the report.</p>	<p>System</p> <p>5- Store the entered information. 6- Display the added lost item on the lost items page.</p>

Table 25 Use-Case Specification - Report a Found Item

Use Case Name	Report a found item	
Brief Description	The user should be able to report a found item.	
Actors	User	
Related Use Case	-	
Precondition	The user must be logged in to the application.	
Post Condition	The system displays the added found item on the found items page.	
Flow of Activities	<p>Actor</p> <p>1- The user enters the lost items page 2- The user clicks on the plus button to report the found item 3- The user fills in the information about the found item 4- Submit the report</p>	<p>System</p> <p>5- Store the entered information 6- Display the added found item on the found items page</p>

Table 26 Use-Case Specification - Create a New Student Activity

Use Case Name	Create a new student activity
Brief Description	The system should allow the user to create a student activity.
Actors	User
Related Use Case	-

Precondition	The user must be logged in to the application.	
Post Condition	The system updates the student activity list.	
Flow of Activities	Actor	System
	1- The user fills in student activity information 2- Create a student activity	3- Store the information in the database 4- Display the updated student activity list

Table 27 Use-Case Specification - Create a Study Group

Use Case Name	Create a study group	
Brief Description	The system should allow the user to create a study group.	
Actors	User	
Related Use Case	-	
Precondition	The user must be logged in to the application.	
Post Condition	The system updated the study group list.	
Flow of Activities	Actor	System
	1- The user fills in a study group information 2- Create a study group	3- Store the information in the database. 4- Display the updated study group list

Table 28 Use-Case Specification - Contact Other Users

Use Case Name	Contact other users	
Brief Description	The system should allow the user to contact other users through live chat.	
Actors	User	
Related Use Case	-	
Precondition	The user must be logged in to the application.	
Post Condition	-	
Flow of Activities	Actor	System
	2- The user writes the message sends the message to the other user	1- Allows the message to be transmitted to the other user 3- The message reaches the user

Table 29 Use-Case Specification - Add an Event to Save List

Use Case Name	Add an event to save list	
Brief Description	The system should allow the user to add an event to save list.	
Actors	User	
Related Use Case	-	
Precondition	The user must be logged in to the application.	
Post Condition	-	
	Actor	System
Flow of Activities	1- The user adds an event to save list.	2- Store the events added to the save list in the database. 3- Display events added to the save list.

Table 30 Use-Case Specification - Remove an Event from the Save List

Use Case Name	Remove an event from the save list	
Brief Description	The system should allow the admin to remove an event from the save list.	
Actors	User	
Related Use Case	-	
Precondition	The user must be logged in to the system.	
Post Condition	-	
	Actor	System
Flow of Activities	1- The user removes an event from the save list.	2- Remove the event in the list from the database. 3- Display the updated save list to the user.

Table 31 Use-Case Specification - Recommend Exclusive Offers

Use Case Name	Recommend exclusive offers
Brief Description	The system should allow the user to recommend exclusive offers.
Actors	Main system
Related Use Case	-
Precondition	-
Post Condition	display recommended offers based on user preference

Flow of Activities	Actor	System
	2- The user selects from the list of recommended exclusive offers.	1- Displays a list of recommended exclusive offers.

3.8 Conclusion

In conclusion, this chapter comprehensively explores both FRs and NFRs. It provides an in-depth discussion covering various aspects of these requirements. A use case diagram has been carefully crafted to visually depict the system's primary actions and functionalities. Furthermore, detailed specifications for each use case are presented using specification tables, offering a deep understanding of the system's behavior and intricacies. An initial application prototype was also developed using the Figma tool, enabling stakeholders to represent the system's user interface visually. This chapter offers a comprehensive analysis of FRs and NFRs, supplemented by visual aids, detailed specifications, and a functional prototype created with the Figma tool.

Chapter 4: Methodology and Tools

4.1 Introduction

This chapter provides a comprehensive account of the organization, planning, and execution of the project's requirements, tasks, and progress. Additionally, the team utilized a project management tool to facilitate efficient collaboration and distribute responsibilities among team members. The tool was instrumental in ensuring the project was completed on time and within budget. The chapter also outlines the team's monitoring strategies to keep track of progress and identify potential bottlenecks.

4.2 Agile Development and System Requirement

The engineering design process entails a series of iterative and creative decision-making steps to develop a system, component, or process that meets the desired needs and specifications while working within specific constraints. This process involves identifying opportunities, developing requirements, performing analysis and synthesis, generating multiple solutions, evaluating those solutions against requirements, considering risks, and making trade-offs to achieve a high-quality solution under the given circumstances. The possible constraints that must be considered during the engineering design process include accessibility, aesthetics, cost, ergonomics, functionality, legal considerations, marketability, regulations, standards, sustainability, or usability [35].

4.2.1 Agile Development

Agile software development methodologies are widely accepted these days. It is an iterative approach to keep pace with dynamic development environments. Traditional software development methods are not efficient enough to cope with the rapid change in requirements and short iterations required for efficient product delivery [36].

4.2.2 Scrum Methodology

Scrum is a framework for managing knowledge work such as product development. Scrum is empirical because it allows teams to establish a hypothesis about how they believe something should function, test it, evaluate the results, and make the necessary modifications. That is, if the framework is appropriately used. Because of the way Scrum is designed, teams can use practices from other frameworks when they make sense, given the team's circumstances [37].

4.2.3 Scrum tool

Monday.com is a flexible work operating system that offers Scrum Masters a very adaptable Agile project management platform. It may be used by teams of all sizes and levels of complexity because it supports sprint planning, task assignments, and timeline tracking [38].

4.2.4 Product Backlog

A backlog is an orderly list of every modification that could be made to the product. Products listed on the backlog are choices rather than promises; merely because they are listed there does not mean they will be delivered [37]. The product backlog classified by the provided services includes Login, Profile, Exclusive offers, Volunteering Opportunities, Student Clubs, Clinic Schedule, Events, Lost and Found items, Student Activity, Study Group, Live Chat, Push Notification, Save List, Psychological Guidance, Certificate, Log Out, Manage exclusive offers, Manage Event, Manage volunteering opportunity, Manage clinic schedule. The subitems show the description of the services [37].

Product Backlog		
	Task	+
	Subitem	Priority
<input type="checkbox"/>	FR1 Login 1	<input type="button" value="⊕"/>
<input type="checkbox"/>	FR1.1 The system must allow the user to log in to their account using UJ ID, and password.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	+ Add subitem	

Figure 33 Login Functional Requirements Product Backlog

Product Backlog		
	Task	+
	Subitem	Priority
<input type="checkbox"/>	FR2 Profile 2	<input type="button" value="⊕"/>
<input type="checkbox"/>	FR2.1 The system should allow the user to edit their information (hobbies, interests, what they can add to the community).	<input type="button" value="⊕"/> M
<input type="checkbox"/>	The system should allow the user to add preference categories of exclusive offers.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	+ Add subitem	

Figure 34 Profile Functional Requirements Product Backlog

Product Backlog		
	Task	+
	Subitem	Priority
<input type="checkbox"/>	FR3 Exclusive Offers 2	<input type="button" value="⊕"/>
<input type="checkbox"/>	FR3.1 The system should recommend exclusive offers for the user based on their preferences.	<input type="button" value="⊕"/> S
<input type="checkbox"/>	FR3.2 The system must display all exclusive offers currently available.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	+ Add subitem	

Figure 35 Exclusive Offers Functional Requirements Product Backlog

Product Backlog		
	Task	+
<input type="checkbox"/>	FR4 Volunteering Opportunities 2	<input type="button" value="⊕"/>
<input type="checkbox"/>	Subitem	Priority
<input type="checkbox"/>	FR4.1 The system should display all available volunteering opportunities.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR4.2 The system should allow the user to register for an available volunteering opportunity.	<input type="button" value="⊕"/> S
	+ Add subitem	

Figure 36 Volunteering Opportunities Functional Requirements Product Backlog

Product Backlog		
	Task	+
<input type="checkbox"/>	FR5 Student Clubs 2	<input type="button" value="⊕"/>
<input type="checkbox"/>	Subitem	Priority
<input type="checkbox"/>	FR5.1 The system should display university student clubs.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR5.2 The system should allow the user to register in student clubs.	<input type="button" value="⊕"/> S
	+ Add subitem	

Figure 37 Student Clubs Functional Requirements Product Backlog

Product Backlog		
	Task	+
<input type="checkbox"/>	FR6 Clinic Schedule 3	<input type="button" value="⊕"/>
<input type="checkbox"/>	Subitem	Priority
<input type="checkbox"/>	FR6.1 The system should display the schedule of the clinic.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR6.2 The system should allow the user to book an appointment at the clinic.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR6.3 The system should allow the user to search in the clinic schedule.	<input type="button" value="⊕"/> M
	+ Add subitem	

Figure 38 Clinic Schedule Functional Requirements Product Backlog

Product Backlog		
	Task	+
<input type="checkbox"/>	FR7 Event 3	<input type="button" value="⊕"/>
<input type="checkbox"/>	Subitem	Priority
<input type="checkbox"/>	FR7.1 The system should allow the user to search for an event.	<input type="button" value="⊕"/> S
<input type="checkbox"/>	FR7.2 The system should display all available events.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR7.3 The system should allow users to register for an event.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	+ Add subitem	

Figure 39 Event Functional Requirements Product Backlog

Product Backlog		
	Task	+
<input type="checkbox"/>	FR8 Lost and found items 7	<input type="button" value="⊕"/>
<input type="checkbox"/>	Subitem	Priority
<input type="checkbox"/>	FR8.1 The system should allow the user to search for a lost item.	<input type="button" value="⊕"/> S
<input type="checkbox"/>	FR8.2 The system should allow the user to search for found items.	<input type="button" value="⊕"/> S
<input type="checkbox"/>	FR8.3 The system should display all lost items.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR8.4 The system should display all found items.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR8.5 The user should have the ability to report a lost item.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR8.6 The user should have the ability to report the found item.	<input type="button" value="⊕"/> M

Figure 40 Lost and found items Functional Requirements Product Backlog

Product Backlog		
	Task	+
<input type="checkbox"/>	FR9 Student Activity 3	<input type="button" value="⊕"/>
<input type="checkbox"/>	Subitem	Priority
<input type="checkbox"/>	FR9.1 The system should display available student activates	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR9.2 The system should allow the user to search for student activity	<input type="button" value="⊕"/> S
<input type="checkbox"/>	FR9.3 The system should allow users to create a new student activity.	<input type="button" value="⊕"/> M
<input type="checkbox"/>	+ Add subitem	

Figure 41 Student Activity Functional Requirements Product Backlog

Product Backlog 23 Tasks / 51 Subitems		
	Task	+
<input type="checkbox"/>	FR10 Study Group 3	<input type="button" value="⊕"/>
	Subitem	Priority
<input type="checkbox"/>	FR10.1 The system should display all available study groups	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR10.2 The system should allow the user to create a study group	<input type="button" value="⊕"/> M
<input type="checkbox"/>	FR10.3 The system should allow the user to search for study group	<input type="button" value="⊕"/> S
	+ Add subitem	

Figure 42 Study Group Functional Requirements Product Backlog

Product Backlog		
	Task	+
<input type="checkbox"/>	FR11 Live Chat 1	<input type="button" value="⊕"/>
	Subitem	Priority
<input type="checkbox"/>	FR11.1 The system should allow the user to contact other users through live chat.	<input type="button" value="⊕"/> M
	+ Add subitem	
<input type="checkbox"/>	FR12 Push Notification 1	<input type="button" value="⊕"/>
	Subitem	Priority
<input type="checkbox"/>	FR12.1 The system should provide push notifications to users' devices to inform them of upcoming ...	<input type="button" value="⊕"/> C

Figure 43 Live Chat and Push Notification Functional Requirements Product Backlog

Product Backlog		
	Task	+
<input type="checkbox"/>	FR13 Save List 2	<input type="button" value="⊕"/>
	Subitem	Priority
<input type="checkbox"/>	FR13.1 The system should allow the user to add an event, volunteering opportunity, study group or activity to their save li...	<input type="button" value="⊕"/> C
<input type="checkbox"/>	FR13.2 The system should allow the user to remove an event, volunteering opportunity, study group or activity from their...	<input type="button" value="⊕"/> C
	+ Add subitem	
<input type="checkbox"/>	FR14 Psychological Guidance 1	<input type="button" value="⊕"/>
	Subitem	Priority
<input type="checkbox"/>	FR14.1 The system should display psychological guidance.	<input type="button" value="⊕"/> C

Figure 44 Save List and Psychological Guidance Functional Requirements Product Backlog

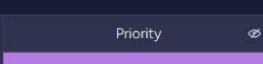
Product Backlog			
	Task	Subitem	Priority
<input type="checkbox"/>	FR15 Certificate 1	<input type="checkbox"/> FR15.5 The system should allow the user to display the attendance certificate <input type="checkbox"/> + Add subitem	 W
<input type="checkbox"/>	FR16 Log Out 1	<input type="checkbox"/> 16.1 The user should be able to log out from their accounts.	 M

Figure 45 Certificate and Logout Functional Requirements Product Backlog

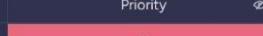
Product Backlog			
	Task	Subitem	Priority
<input type="checkbox"/>	FR17 Login (in admin panel) 1	<input type="checkbox"/> FR17.1 The system must allow the admin to log in to their account using their ID and password. <input type="checkbox"/> + Add subitem	 M

Figure 46 Admin Panel Login Functional Requirements Product Backlog

Product Backlog			
	Task	Subitem	Priority
<input type="checkbox"/>	FR18 Manage exclusive offers 3	<input type="checkbox"/> FR18.1 The system should allow the admin to add new exclusive offers <input type="checkbox"/> FR18.2 The system should allow the admin to edit exclusive offers information. <input type="checkbox"/> FR18.3 The system should allow the admin to remove exclusive offers. <input type="checkbox"/> + Add subitem	 M  M  M

Figure 47 Manage exclusive offers Functional Requirements Product Backlog

Product Backlog			
	Task	+	
<input type="checkbox"/>	FR19 Manage event 3		
<input type="checkbox"/>	Subitem		Priority
<input type="checkbox"/>	FR19.1 The system should allow the admin to add a new event.		M
<input type="checkbox"/>	FR19.2 The system should allow the admin to edit event information.		M
<input type="checkbox"/>	FR19.3 The system should allow the admin to remove the event.		M
<input type="checkbox"/>	+ Add subitem		

Figure 48 Manage Event Functional Requirements Product Backlog

Product Backlog			
	Task	+	
<input type="checkbox"/>	FR20 Manage volunteering opportunity 3		
<input type="checkbox"/>	Subitem		Priority
<input type="checkbox"/>	FR20.1 The system should allow the admin to add new volunteering opportunities.		M
<input type="checkbox"/>	FR20.2 The system should allow the admin to edit volunteering opportunity information.		M
<input type="checkbox"/>	FR20.3 The system should allow the admin to remove a volunteering opportunity.		M
<input type="checkbox"/>	+ Add subitem		

Figure 49 Manage volunteering opportunity Functional Requirements Product Backlog

Product Backlog			
	Task	+	
<input type="checkbox"/>	FR22 Manage student clubs 3		
<input type="checkbox"/>	Subitem		Priority
<input type="checkbox"/>	FR22.1 The system should allow the admin to add a new student club.		M
<input type="checkbox"/>	FR22.2 The system should allow the admin to edit the student club information.		M
<input type="checkbox"/>	FR22.3 The system should allow the admin to remove a student club.		M
<input type="checkbox"/>	+ Add subitem		

Figure 50 Manage Student Clubs Functional Requirements Product Backlog

Product Backlog			
	Task	+	
<input type="checkbox"/>	FR21 Manage clinic schedule 3		
<input type="checkbox"/>	Subitem		Priority
<input type="checkbox"/>	FR21.1 The system should allow the admin to add the clinic schedule for the week.		M
<input type="checkbox"/>	FR21.2 The system should allow the admin to edit the clinic schedule.		M
<input type="checkbox"/>	FR21.3 The system should allow the admin to remove the clinic schedule.		M
<input type="checkbox"/>	+ Add subitem		

Figure 51 Clinic Schedule Functional Requirements Product Backlog

Product Backlog(Non-Functional Requirements)			
	Task		+
<input type="checkbox"/>	▼ NFR1 User Interface 1	<input type="checkbox"/>	
	Subitem		Priority
<input type="checkbox"/>	NFR1.1 The interfaces will be easy to use and user-friendly.	<input type="checkbox"/>	S
<input type="checkbox"/>	+ Add subitem		

Figure 52 User Interface Non-Functional Requirements Product Backlog

Product Backlog(Non-Functional Requirements)			
	Task		+
<input type="checkbox"/>	▼ NFR2 Security 2	<input type="checkbox"/>	
	Subitem		Priority
<input type="checkbox"/>	NFR2.1 The application must ensure that user confidential data cannot be accessible by unauthorized perso...	<input type="checkbox"/>	M
<input type="checkbox"/>	NFR2.2 The application will ask permission to access the camera.	<input type="checkbox"/>	M
<input type="checkbox"/>	+ Add subitem		

Figure 53 Security Non-Functional Requirements Product Backlog

Product Backlog(Non-Functional Requirements)			
	Task		+
<input type="checkbox"/>	▼ NFR3 Interoperability 1	<input type="checkbox"/>	
	Subitem		Priority
<input type="checkbox"/>	NFR3.1 The application should be able to run on IOS and Android.	<input type="checkbox"/>	M
<input type="checkbox"/>	+ Add subitem		

Figure 54 Interoperability Non-Functional Requirements Product Backlog

Product Backlog(Non-Functional Requirements)			
	Task		+
<input type="checkbox"/>	▼ NFR4 Performance 1	<input type="checkbox"/>	
	Subitem		Priority
<input type="checkbox"/>	NFR4.1 The user should be able to complete all system functions successfully within high volume of users an...	<input type="checkbox"/>	S
<input type="checkbox"/>	+ Add subitem		

Figure 55 Performance Non-Functional Requirements Product Backlog

Product Backlog(Non-Functional Requirements)			
	Task		
<input type="checkbox"/>	NFR5 Usability 3	<input type="button" value="⊕"/>	
<input type="checkbox"/>	Subitem		Priority
<input type="checkbox"/>	NFR5.1 The application should be easy to learn and use.	<input type="button" value="⊕"/>	M
<input type="checkbox"/>	NFR5.2 The application should be usable for users with disabilities.	<input type="button" value="⊕"/>	S
<input type="checkbox"/>	NFR5.3 The application should be designed to comply with accessibility standards.	<input type="button" value="⊕"/>	S
	+ Add subitem		

Figure 56 Usability Non-Functional Requirements Product Backlog

Product Backlog(Non-Functional Requirements)			
	Task		
<input type="checkbox"/>	NFR6 Maintainability 1	<input type="button" value="⊕"/>	
<input type="checkbox"/>	Subitem		Priority
<input type="checkbox"/>	NFR6.1 The application should support easy updates and patches without significant downtime.	<input type="button" value="⊕"/>	M
	+ Add subitem		

Figure 57 Maintainability Non-Functional Requirements Product Backlog

Product Backlog(Non-Functional Requirements)			
	Task		
<input type="checkbox"/>	NFR7 Availability 1	<input type="button" value="⊕"/>	
<input type="checkbox"/>	Subitem		Priority
<input type="checkbox"/>	NFR7.1 The application should be accessible 24/7.	<input type="button" value="⊕"/>	S
	+ Add subitem		

Figure 58 Availability Non-Functional Requirements Product Backlog

4.2.5 Sprint Backlog

The product backlog items chosen for delivery during the sprint and any tasks that need to be identified by the team to deliver those items and meet the sprint goal make up the Sprint Backlog[37].

- ◆ Sprint 1: FR1- Login, FR8- Lost and Found Items, FR16- Log out.

Sprint #1		Status	Priority
Task			
FR1.1 The system must allow the user to log in to their account using UJ ID and password.	<input checked="" type="checkbox"/>	Done	M
FR8.1 The system should allow the user to search for a lost item.	<input checked="" type="checkbox"/>	Done	S
FR8.2 The system should allow the user to search for found items.	<input checked="" type="checkbox"/>	Done	S
FR8.3 The system should display all lost items	<input checked="" type="checkbox"/>	Done	M
FR8.4 The system should display all found items	<input checked="" type="checkbox"/>	Done	M
FR8.5 The user should be able to report a lost item.	<input checked="" type="checkbox"/>	Done	M
FR8.6 The user should be able to report the found item.	<input checked="" type="checkbox"/>	Done	M
FR16.1 The user should be able to log out from their accounts	<input checked="" type="checkbox"/>	Done	M

Figure 59 Sprint 1 Backlog

- ◆ Sprint 2: FR10- Study Groups, FR9- Student Activities, FR2- Profile

Sprint #2		Status	Priority
Task			
FR10.1 The system should display all available study groups	<input checked="" type="checkbox"/>	Done	M
FR10.2 The system should allow the user to create a study group	<input checked="" type="checkbox"/>	Done	M
FR10.3 The system should allow the user to search for study group	<input checked="" type="checkbox"/>	Done	S
FR9.1 The system should display available student activates	<input checked="" type="checkbox"/>	Done	M
FR9.2 The system should allow the user to search for student activity	<input checked="" type="checkbox"/>	Done	S
FR9.3 The system should allow users to create a new student activity.	<input checked="" type="checkbox"/>	Done	M
FR2.1 The system should allow the user to edit their information (hobbies, interests, what they can add to the c...	<input checked="" type="checkbox"/>	Done	M
FR2.2 The system should allow the user to add preference categories of exclusive offers.	<input checked="" type="checkbox"/>	Done	M

Figure 60 Sprint 2 Backlog

- ◆ Sprint 3: FR17- Admin Panel Login, FR23-Admin Panel Logout, FR18- Manage exclusive offers, FR3- Exclusive Offers, FR20- Manage Volunteering Opportunities, FR4- Volunteering Opportunities in App

Sprint #3			
	Task	Status	Priority
<input type="checkbox"/>	FR17.1 The system must allow the admin to log in to their account using their ID and password.	⊕ Done	M
<input type="checkbox"/>	FR23.1 The admin should be able to log out from their accounts	⊕ Done	M
<input type="checkbox"/>	FR18.1 The system should allow the admin to add new exclusive offers	⊕ Done	M
<input type="checkbox"/>	FR18.2 The system should allow the admin to edit exclusive offers information.	⊕ Done	M
<input type="checkbox"/>	FR18.3 The system should allow the admin to remove exclusive offers.	⊕ Done	M
<input type="checkbox"/>	FR3.1 The system should recommend exclusive offers for the user based on their preferences.	⊕ Done	S
<input type="checkbox"/>	FR3.3 The system must display all exclusive offers currently available.	⊕ Done	M
<input type="checkbox"/>	FR20.1 The system should allow the admin to add new volunteering opportunities.	⊕ Done	M
<input type="checkbox"/>	FR20.2 The system should allow the admin to edit volunteering opportunity information.	⊕ Done	M
<input type="checkbox"/>	FR20.3 The system should allow the admin to remove a volunteering opportunity.	⊕ Done	M
<input type="checkbox"/>	FR4.1 The system should display all available volunteering opportunities.	⊕ Done	M
<input type="checkbox"/>	FR4.2 The system should allow the user to register for an available volunteering opportunity.	⊕ Done	S

Figure 61 Sprint 3 Backlog

- ◆ Sprint 4: FR6: Clinic Schedule, FR21- Manage Clinic Schedule, FR22- Manage Student Clubs, FR5- Student Clubs

Sprint #4			
	Task	Status	Priority
<input type="checkbox"/>	FR6.1 The system should display the schedule of the clinic	⊕ Done	M
<input type="checkbox"/>	FR6.2 The system should allow the user to book an appointment at the clinic	⊕ Done	M
<input type="checkbox"/>	FR6.3 The system should allow the user to search in the clinic schedule	⊕ Done	M
<input type="checkbox"/>	FR21.1 The system should allow the admin to add the clinic schedule for the week.	⊕ Done	M
<input type="checkbox"/>	FR21.2 The system should allow the admin to edit the clinic schedule.	⊕ Done	M
<input type="checkbox"/>	FR21.3 The system should allow the admin to remove the clinic schedule.	⊕ Done	M
<input type="checkbox"/>	FR22.1 The system should allow the admin to add a new student club.	⊕ Done	M
<input type="checkbox"/>	FR22.2 The system should allow the admin to edit the student club information.	⊕ Done	M
<input type="checkbox"/>	FR22.3 The system should allow the admin to remove a student club.	⊕ Done	M
<input type="checkbox"/>	FR5.1 The system should display university student clubs	⊕ Done	M
<input type="checkbox"/>	FR5.2 The system should allow the user to register in student clubs	⊕ Done	S

Figure 62 Sprint 4 Backlog

- ◆ Sprint 5: FR19- Manage Event, FR7- Events, FR11- Live Chat, FR14- Psychological Guidance

Sprint #5			
	Task	Status	Priority
<input type="checkbox"/>	FR19.1 The system should allow the admin to add a new event	 Done	M
<input type="checkbox"/>	FR19.2 The system should allow the admin to edit event information.	 Done	M
<input type="checkbox"/>	FR19.3 The system should allow the admin to remove the event	 Done	M
<input type="checkbox"/>	FR7.1 The system should allow the user to search for an event	 Done	S
<input type="checkbox"/>	FR7.2 The system should display all available events.	 Done	M
<input type="checkbox"/>	FR7.3 The system should allow users to register for an event.	 Done	M
<input type="checkbox"/>	FR11.1 The system should allow the user to contact other users through live chat.	 Done	M
<input type="checkbox"/>	FR14.1 The system should display psychological guidance.	 Done	C

Figure 63 Sprint 5 Backlog

- ◆ Sprint 6: FR13- Save List, FR12- Push Notification

Sprint #6			
	Task	Status	Priority
<input type="checkbox"/>	FR13.1 The system should allow the user to add an event, volunteering opportunity, study group or activity to t...	 Done	C
<input type="checkbox"/>	FR13.2 The system should allow the user to remove an event, volunteering opportunity, study group or activity ...	 Done	C
<input type="checkbox"/>	FR12.1 The system should provide push notifications to users' devices to inform them of upcoming events, exc...	 Done	C
<input type="checkbox"/>	FR15.5 The system should allow the user to display the attendance certificate	 To do	W

Figure 64 Sprint 6 Backlog

4.3 Burndown chart

The burndown chart visually represents the sprint and is an information radiator. The burndown chart assists the team in determining whether they will deliver the intended functionality identified in the forecast during sprint planning. It is based on the number of stories points an agile team has historically been able to "burn down" during each sprint. *Figure 65 to Figure 70* shows the progression of the project along 6 sprints[39].

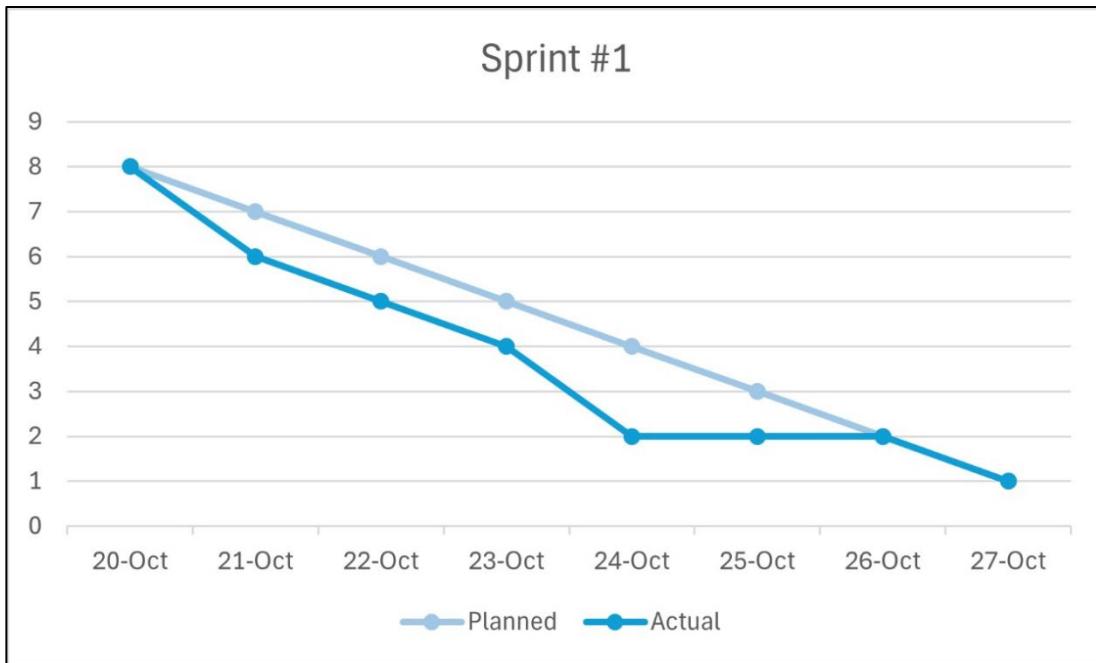


Figure 65 Sprint 1 Burndown chart.

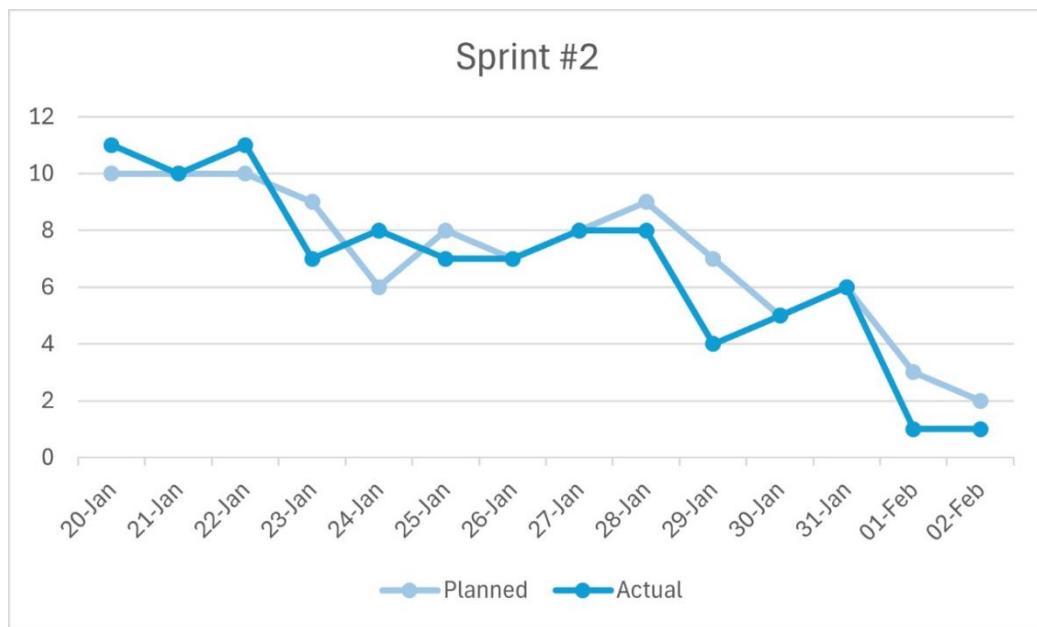


Figure 66 Sprint 2 Burndown chart.

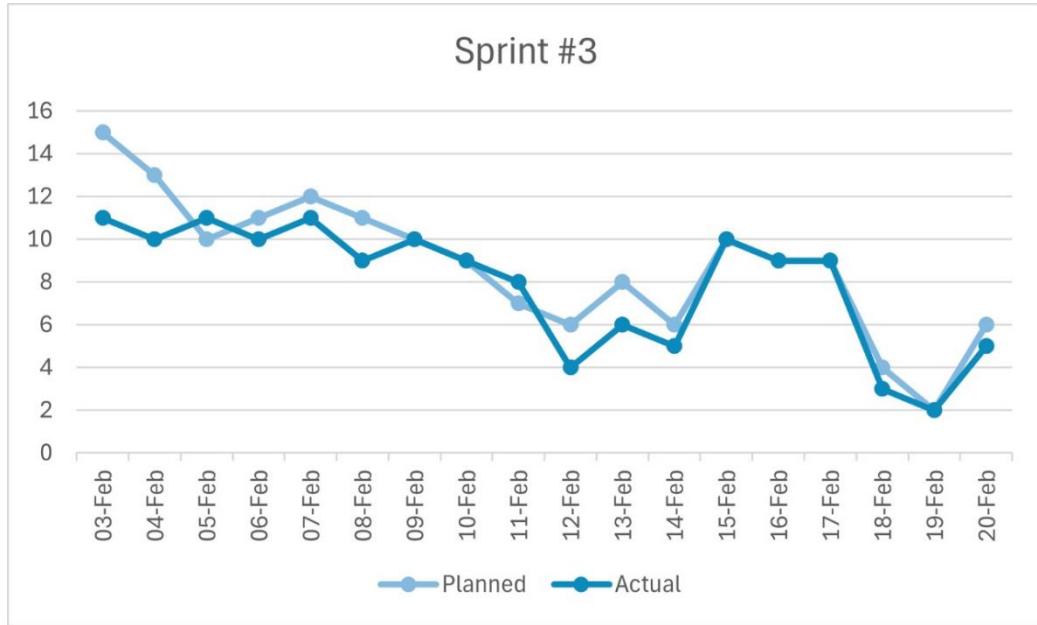


Figure 67 Sprint 3 Burndown chart.

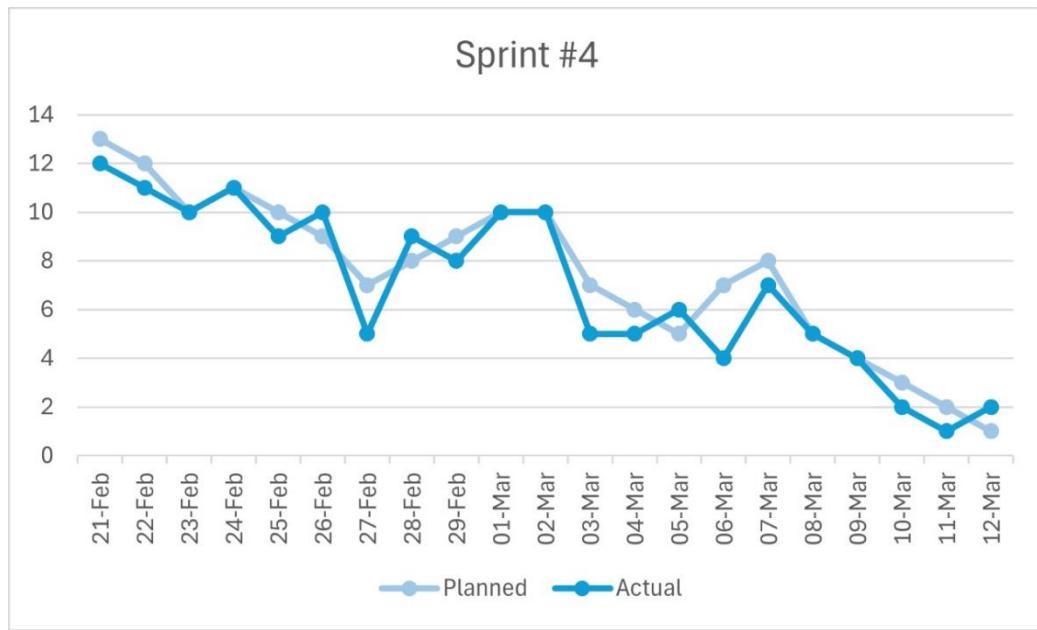


Figure 68 Sprint 4 Burndown chart.

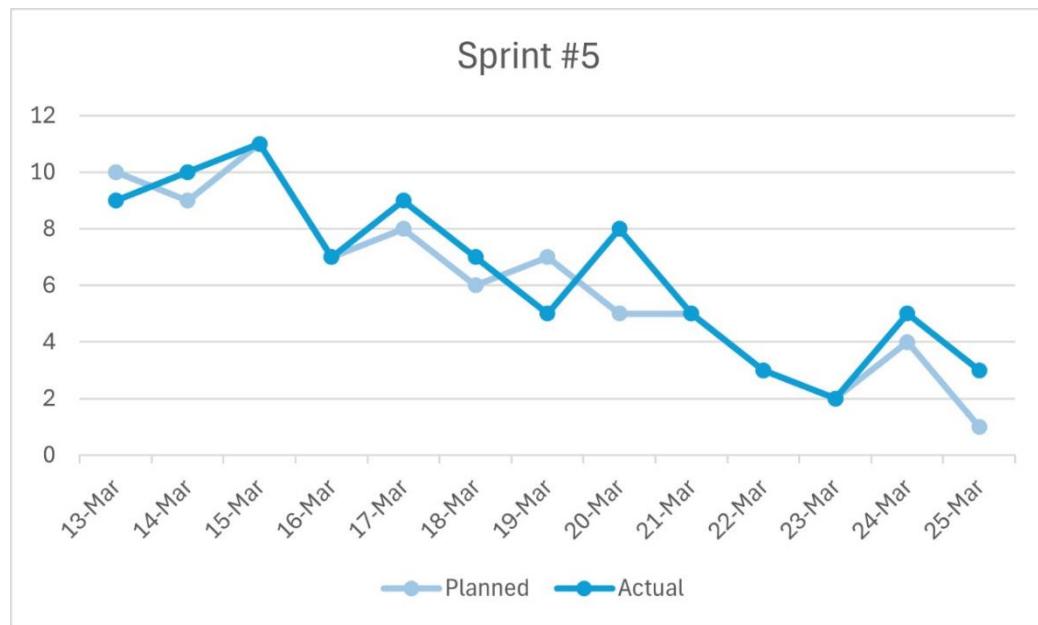


Figure 69 Sprint 5 Burndown chart.

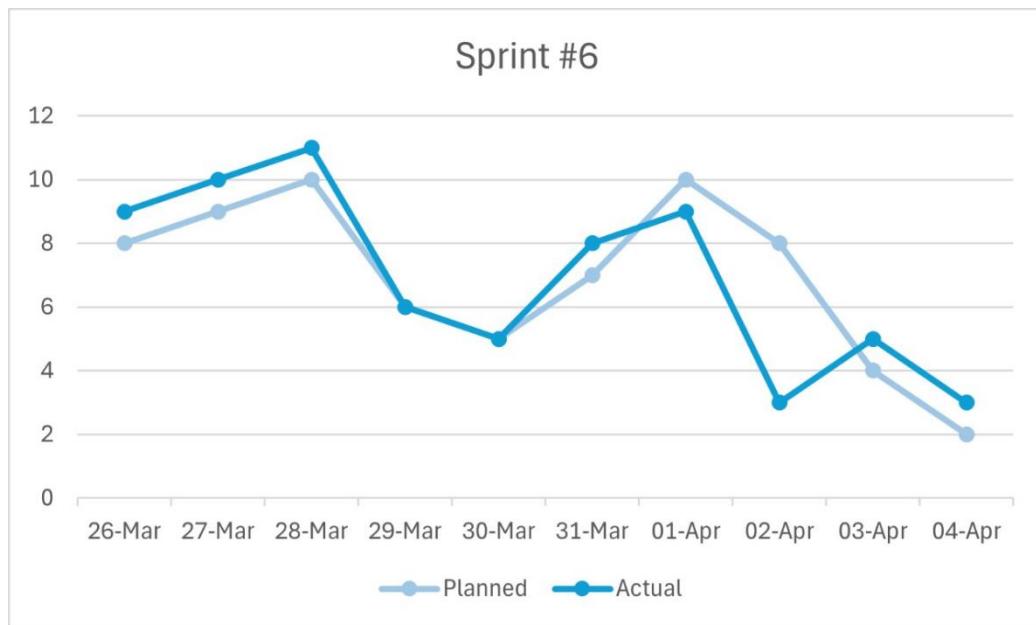


Figure 70 Sprint 6 Burndown chart.

4.4 Task and their Allocation

SubTasks		Task	Status	People	+
	Ideation		Done	SA +4	

Figure 71 Task Allocation for Ideation

SubTasks		Task	+	
		Subitem	Status	People
	Chapter 1 4			
		Definitions and objectives	Done	SB
		Search for methods	Done	SA
		Find similar applications	Done	SQ
		Gantt Chart	Done	

Figure 72 Tasks Allocation for Chapter 1

SubTasks		Task	+	
		Subitem	Status	People
	Chapter 2 2			
		Definitions and search for current situation	Done	SE
		Search for existing systems	Done	SE SA
		+ Add subitem		

Figure 73 Tasks Allocation for Chapter 2

SubTasks		Task	+	
		Subitem	Status	People
	chapter 3 4			
		Requirements gathering	Done	SA SB SE SQ
		Use cases (Diagrams, Specifications)	Done	SA SE SQ
		Analysis the requirements	Done	SE SA
		Prototyping	Done	SE SQ
		+ Add subitem		

Figure 74 Tasks Allocation for Chapter 3

SubTasks		
	Task	+
Chapter 4 [3]		
	Subitem	
	Write backlog	+ Done
	Sprint backlog	+ Done
	Burn down chart	+ Done
	+ Add subitem	

Figure 75 Tasks Allocation for Chapter 4

SubTasks		
	Task	+
Chapter 5 [3]		
	Subitem	
	Class Diagram	+ Done
	Sequence Diagram	+ Done
	Flow Diagram	+ Done
	+ Add subitem	

Figure 76 Tasks Allocation for Chapter 5

SubTasks		
	Task	+
Chapter 6 [1]		
	Subitem	
	Coding	+ Done
	+ Add subitem	

Figure 77 Tasks Allocation for Chapter 6

SubTasks		
	Task	+
Chapter 7 [1]		
	Subitem	
	Testing	+ Done
	+ Add subitem	

Figure 78 Tasks Allocation for Chapter 7

Table 32 Tasks Allocation and Their Assigned Team Member

Tasks	Assigned team member
Ideation	Manar Majeed, Sabreen Binsalman, Maya Karoma, Shahad Alshehri, Shomoukh Qattan
Chapter 1	
Definitions and objectives	Manar Majeed, Maya Karoma
Search for methods	Sabreen Binsalman
Find similar applications	Shahad Alshehri
Gantt Chart	Shomoukh Qattan
Chapter 2	
Definitions and search for current situation	Manar Majeed, Maya Karoma, Sabreen Binsalman
Search for existing systems.	Shahad Alshehri, Shomoukh Qattan
Chapter 3	
Requirements gathering	Manar Majeed, Sabreen Binsalman, Maya Karoma, Shahad Alshehri, Shomoukh Qattan,
Use cases (Diagrams, Specifications)	Shahad Alshehri, Shomoukh Qattan, Manar Majeed, Sabreen Binsalman, Maya Karoma
Analysis the requirements	Shahad Alshehri, Manar Majeed, Sabreen Binsalman
Prototyping	Manar Majeed, Sabreen Binsalman, Maya Karoma
Chapter 4	
Write backlog	Shahad Alshehri, Shomoukh Qattan, Manar Majeed, Sabreen Binsalman, Maya Karoma

Sprint backlog	Shahad Alshehri, Manar Majeed, Sabreen Binsalman
Burn down chart	Shomoukh Qattan, Maya Karoma
Chapter 5	
Class Diagram	Shahad Alshehri, Shomoukh Qattan, Manar Majeed, Sabreen Binsalman, Maya Karoma
Activity Diagram	Manar Majeed, Sabreen Binsalman
Sequence Diagram	Shahad Alshehri, Shomoukh Qattan, Maya Karoma
Chapter 6	
Admin Panel Coding	
Login	Sabreen Binsalman
Manage exclusive offers	Manar Majeed
Manage event	Manar Majeed, Sabreen Binsalman, Maya Karoma
Manage volunteering opportunity	Sabreen Binsalman
Manage clinic schedule	Maya Karoma
Manage student clubs	Sabreen Binsalman
Application Coding	
login	Manar Majeed, Sabreen Binsalman
Profile	Manar Majeed, Shomoukh Qattan
Exclusive Offers	Sabreen Binsalman

Volunteering Opportunities	Maya Karoma
Student Clubs	Sabreen Binsalman
Clinic Schedule	Maya Karoma
Events	Sabreen Binsalman, Shomoukh Qattan
Lost and found items	Manar Majeed, Sabreen Binsalman, Maya Karoma
Student Activity	Shahad Alshehri
Study Group	Shahad Alshehri, Maya Karoma
Live chat	Manar Majeed, Shahad Alshehri
Push notification	Manar Majeed
Save list	Maya Karoma, Manar Majeed
Psychological guidance	Shomoukh Qattan, Sabreen Binsalman
Log out	Sabreen Binsalman, Shahad Alshehri

Chapter 7

Admin Panel Testing

Login	Manar Majeed, Sabreen Binsalman
Manage exclusive offers	Maya Karoma
Manage event	Shomoukh Qattan, Shahad Alshehri
Manage volunteering opportunity	Maya Karoma

Manage clinic schedule	Maya Karoma
Manage student clubs	Shahad Alshehri, Sabreen Binsalman
Application Testing	
login	Sabreen Binsalman
Profile	Manar Majeed
Exclusive Offers	Maya Karoma
Recommendation System	Manar Majeed
Volunteering Opportunities	Maya Karoma
Student Clubs	Sabreen Binsalman, Shahad Alshehri
Clinic Schedule	Maya Karoma
Events	Shomoukh Qattan, Shahad Alshehri
Lost and found items	Manar Majeed, Sabreen Binsalman, Maya Karoma
Student Activity	Shomoukh Qattan, Shahad Alshehri
Study Group	Shahad Alshehri
Live chat	Shomoukh Qattan, Manar Majeed
Push notification	Manar Majeed
Save list	Shomoukh Qattan, Manar Majeed
Psychological guidance	Shahad Alshehri

[Log out](#)

Sabreen Binsalman

4.5 Conclusion

The project team successfully organized and planned the project requirements and tasks by implementing Scrum principles while maintaining visibility of the progress and addressing any issues.

Chapter 5: Analysis and Design

5.1 Introduction

This chapter provides an overview of essential UML diagrams and system architecture in software development. It begins by introducing class diagrams, which depict the static structure of a software system. Then, sequence diagram , showcasing the order of object interactions. Activity diagrams are explored to visualize processes, while the system architecture diagram illustrates the relationship between different user types and applications. Lastly, the Model-View-Controller (MVC) architecture is introduced to manage complexity and improve scalability.

The purpose of the chapter is to simplify and define the project's structure by using Unified Modeling Language (UML) diagrams that are often used to visualize, understand, and communicate the design and behavior of a system [40].

5.2 Class Diagram

One of the most essential Unified Modeling Language (UML) diagrams is the class diagram, used to model a software system's static structure. And a UML representation of the source code and the relationships and dependencies between classes [41]. *Figure 79* Represents visualization for the program class diagram.

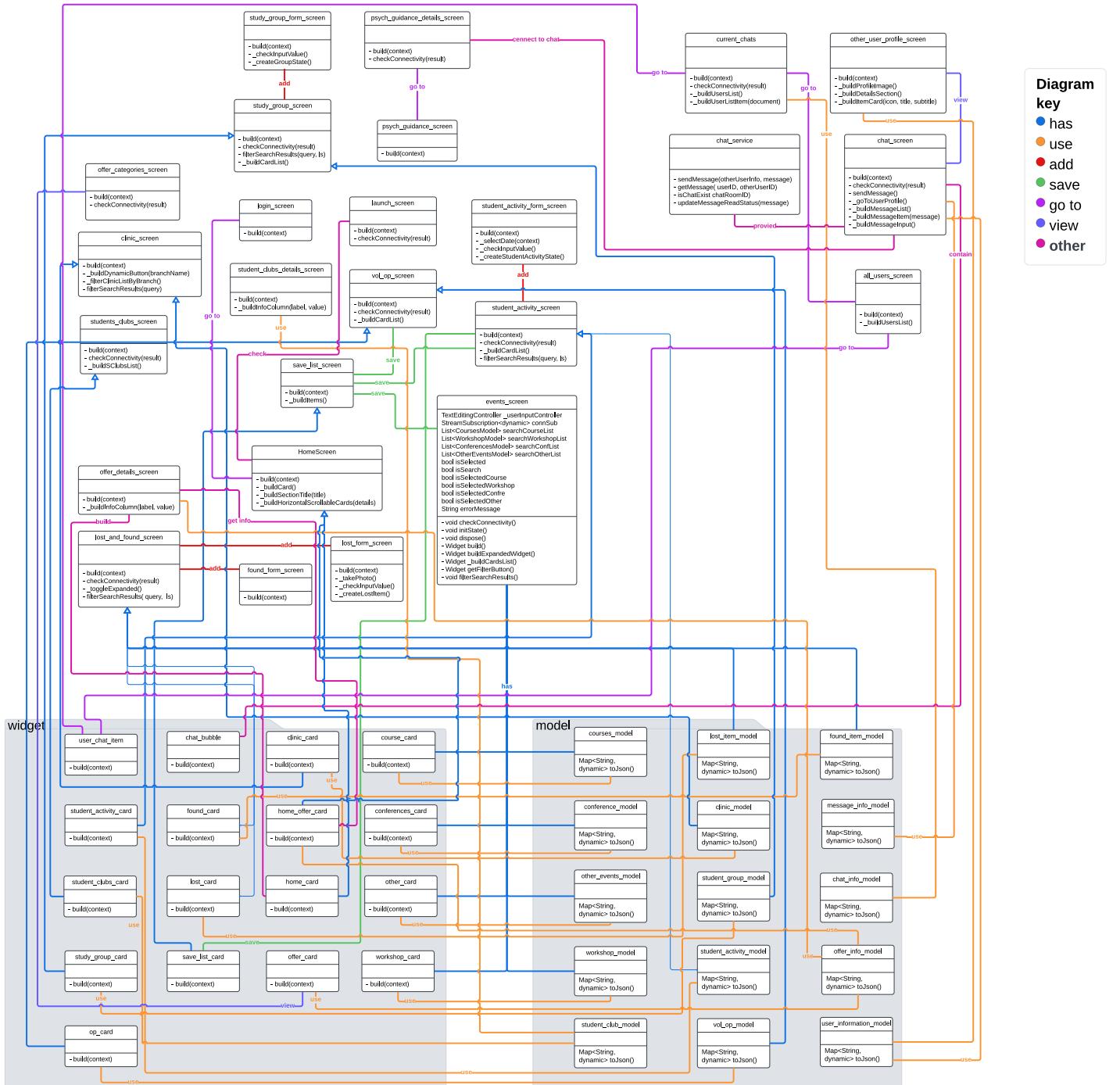


Figure 79 Class Diagram

5.3 Sequence Diagram

The primary purpose of a sequence diagram is to display object interactions in the order in which they occur [42].

Figure 80 It will show the sequence of login functionality. Illustrates the login by a user to the application where a valid number and password are required as the number and password in the university's database. The system verifies the entered login information, and the successful login will forward the user to the homepage.

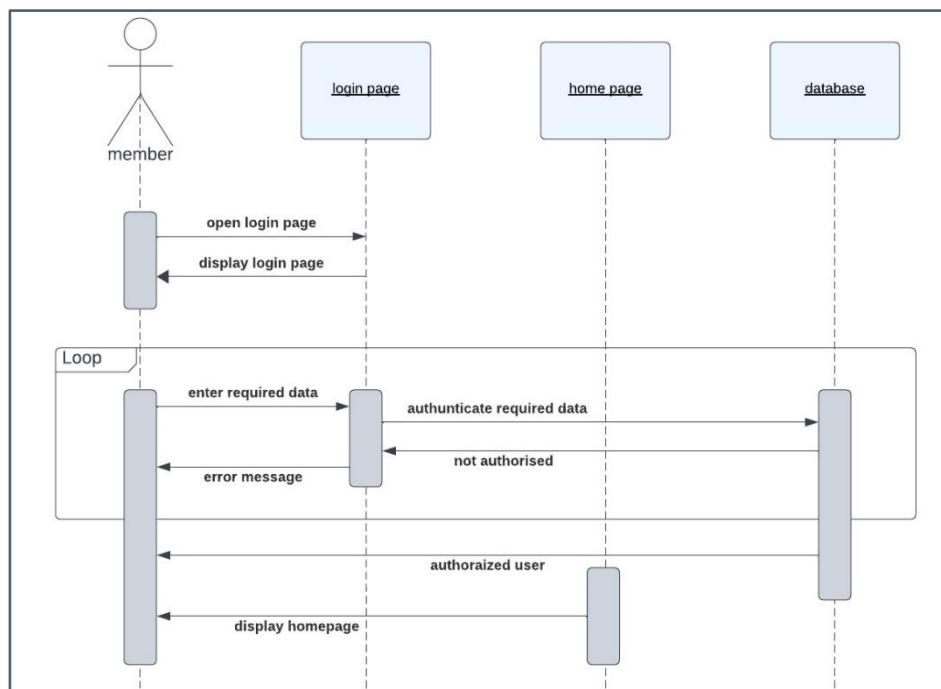


Figure 80 Sequence Diagram - Login

Figure 82 and Figure 81 will display the sequence diagram for the lost and found items, where the user needs to create a new report and add the required data to show the item on the app.

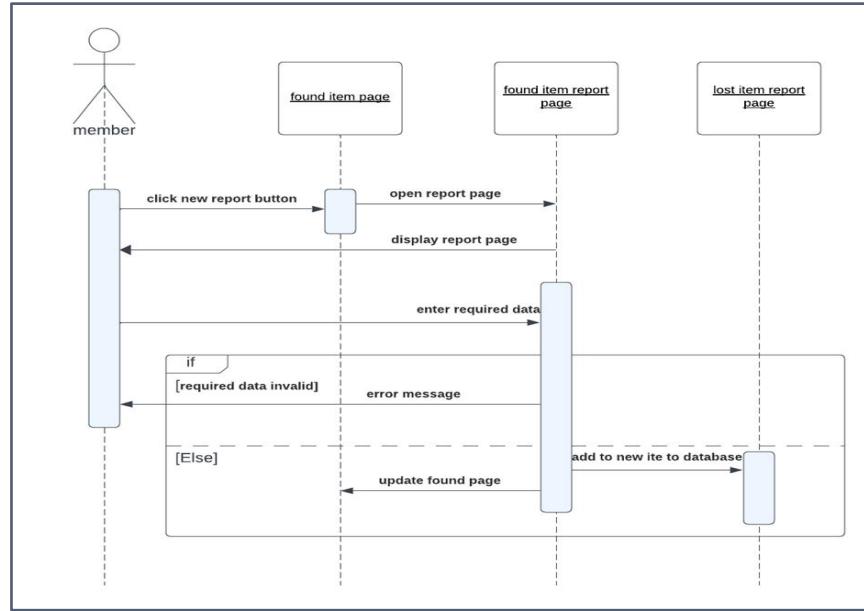


Figure 81 Sequence Diagram - Found Item

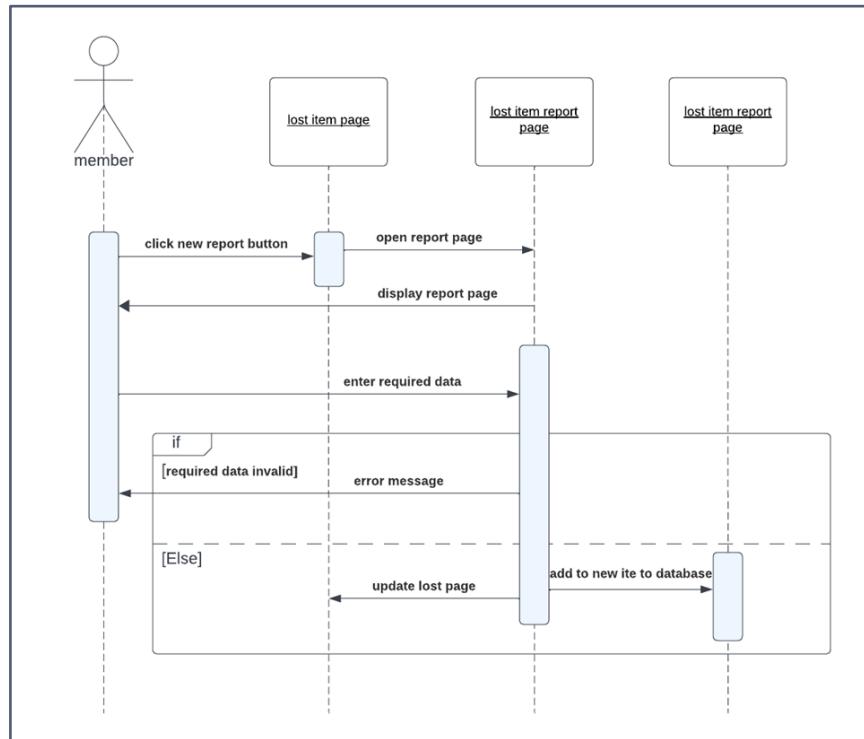


Figure 82 Sequence Diagram - Lost Item

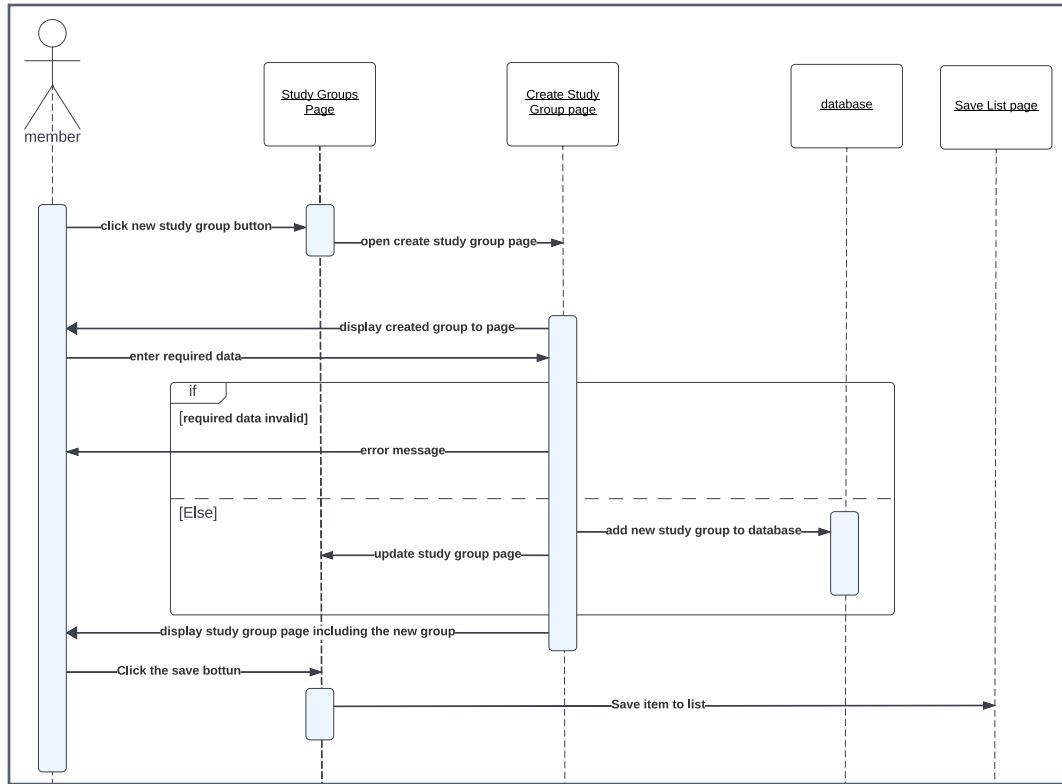


Figure 83 Sequence Diagram - Study Group

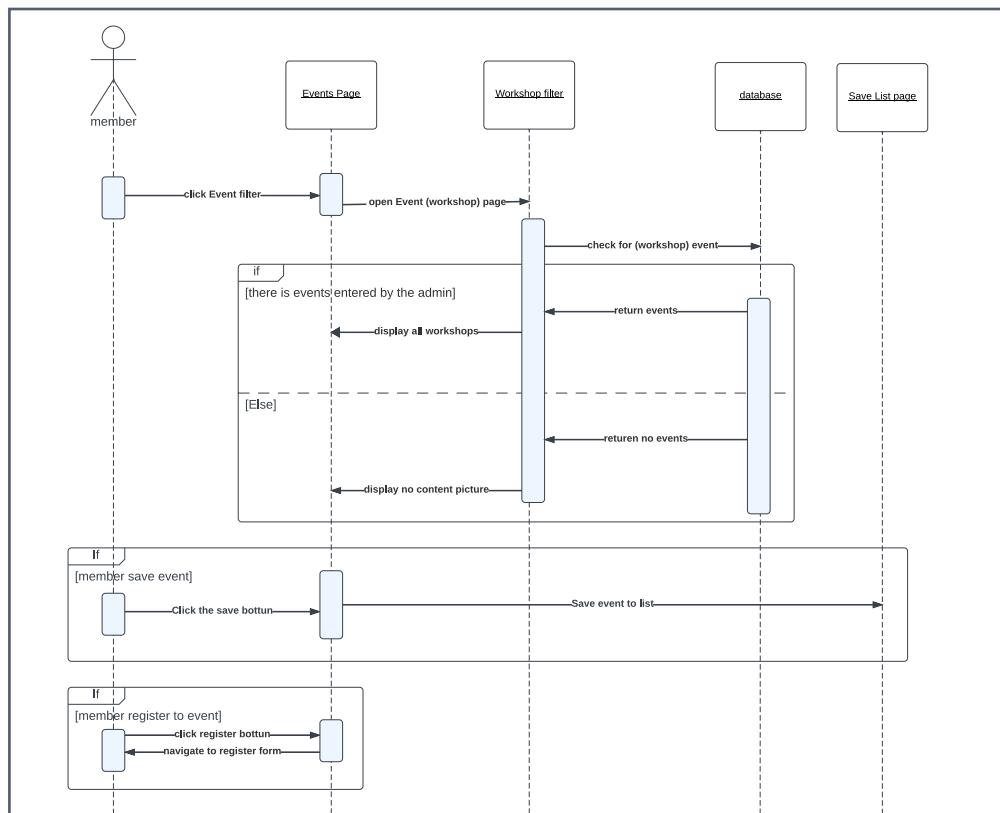


Figure 84 Sequence Diagram - Events

5.4 Activity Diagram

Individual use cases are frequently explained and visualized using UML activity diagrams (ADs). Due to their higher level of abstraction and process-oriented perspective, UML ADs are also highly suitable for model-based test generation [40].

Figure 85 It will display the process of reporting lost and found items by the user.

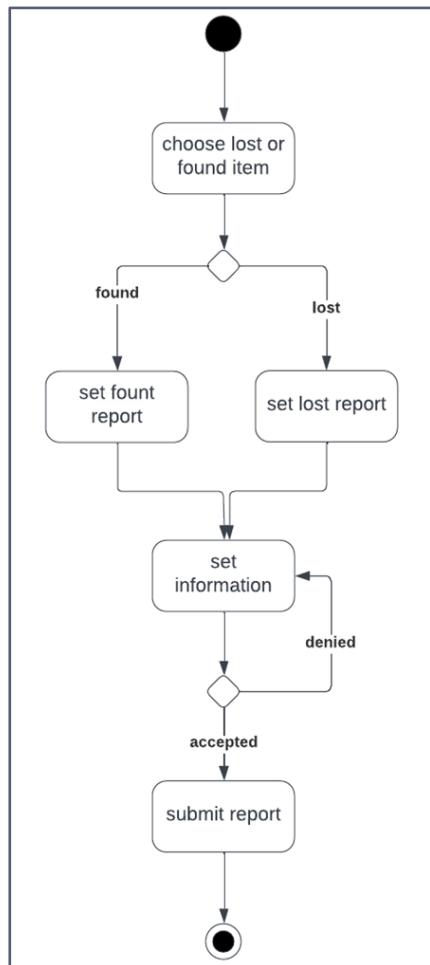


Figure 85 Activity Diagram - Create Lost or Found Item Report

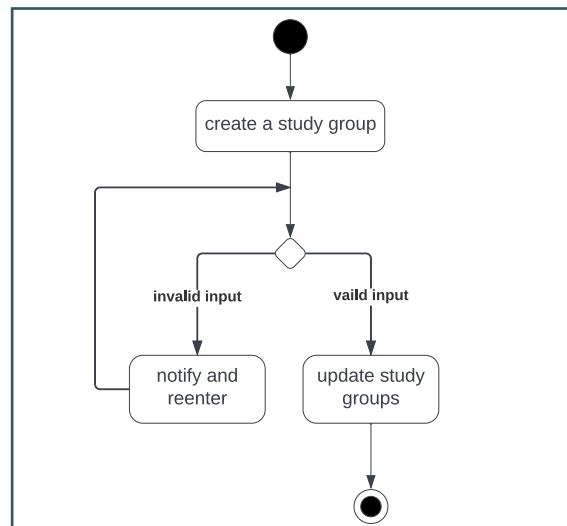


Figure 86 Activity Diagram - Study Group

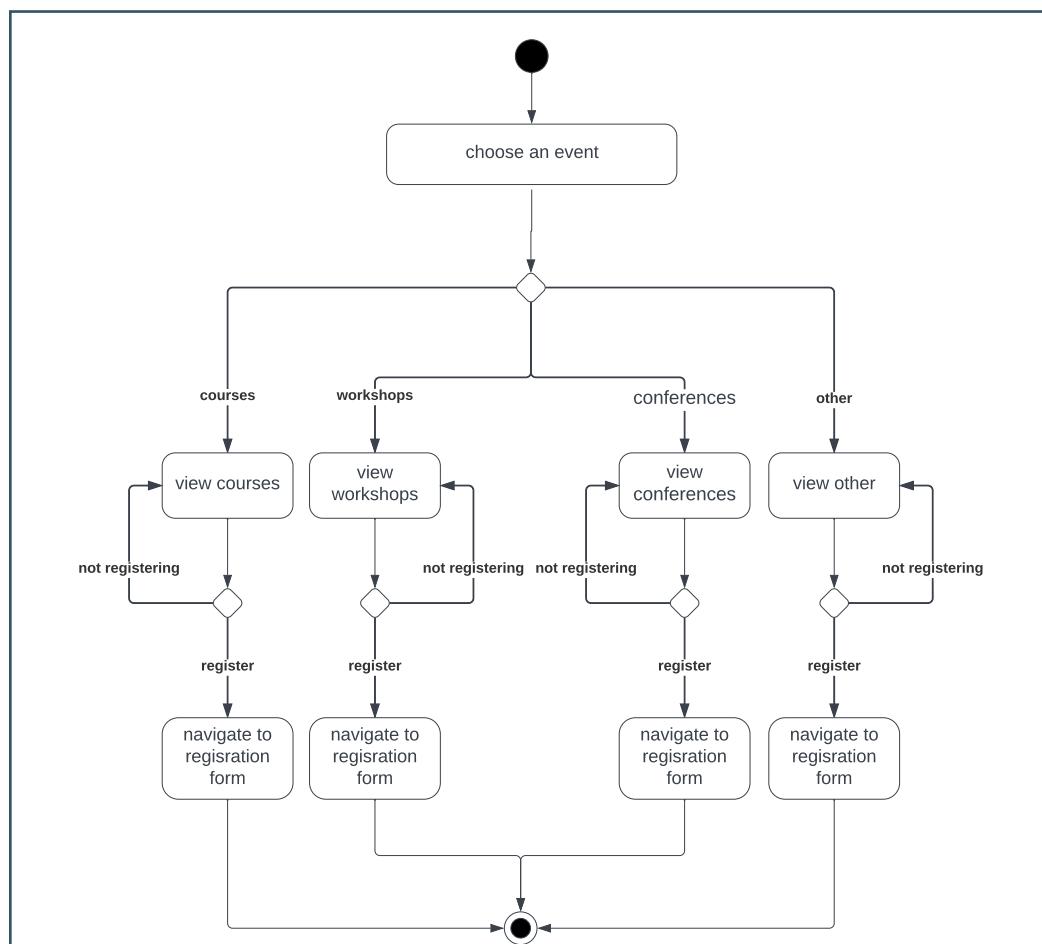


Figure 87 Activity Diagram - Event

5.5 System Architecture

The system will have two types of users: mobile application users and admins who will manage the admin panel. The architecture of both mobile and web applications is depicted in *Figure 88*. Both applications will use the same Firebase database for real-time updates.

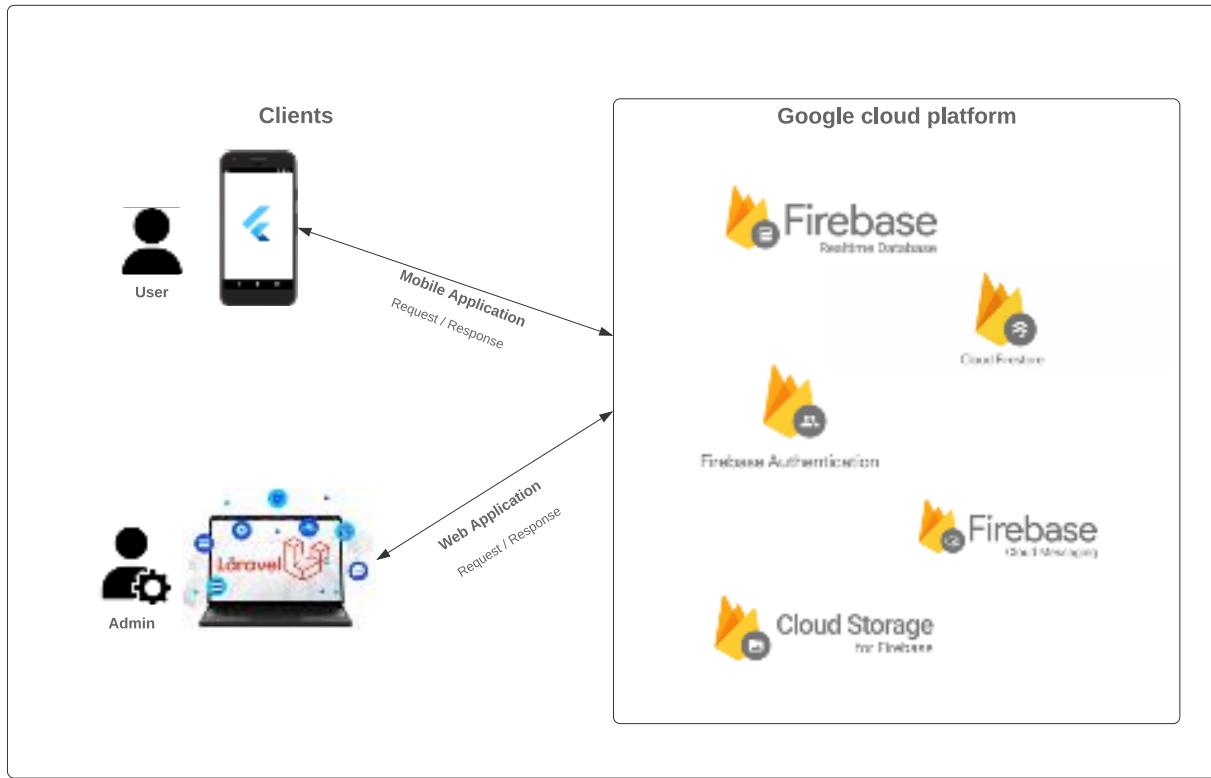


Figure 88 System Architecture

MVC Architecture

The Model-View-Controller (MVC) architecture is a software design pattern that separates data management (Model), user interface (View), and user input processing (Controller). This division helps manage complexity and improve scalability, making development and maintenance more efficient by isolating different aspects of the application.[43]

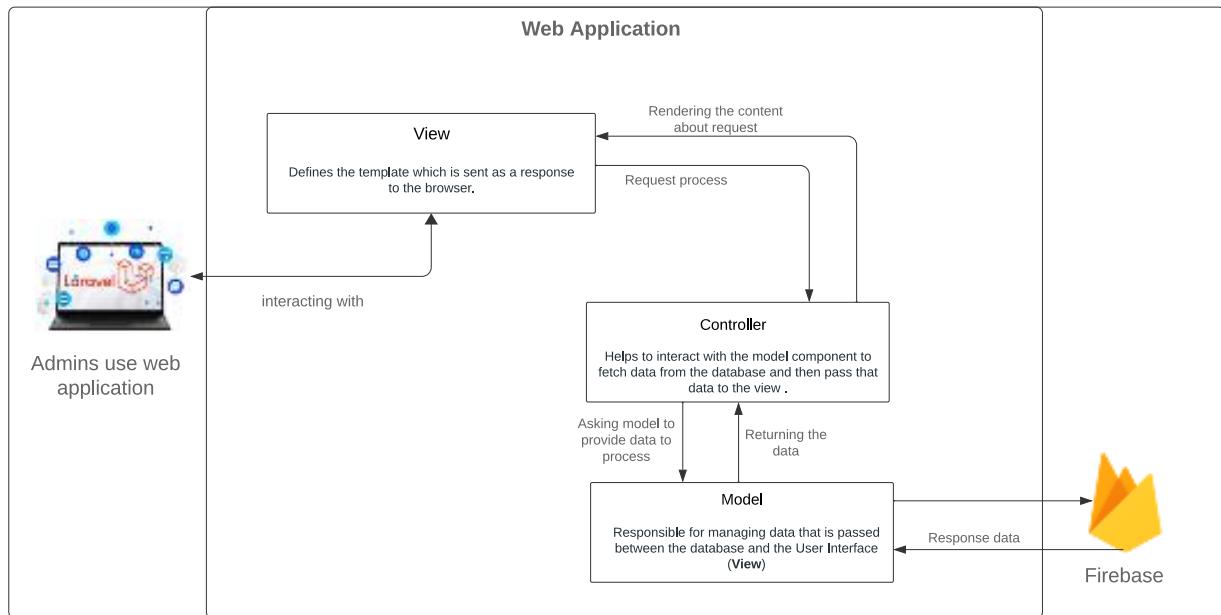


Figure 89 MVC Architecture for Admin Panel.

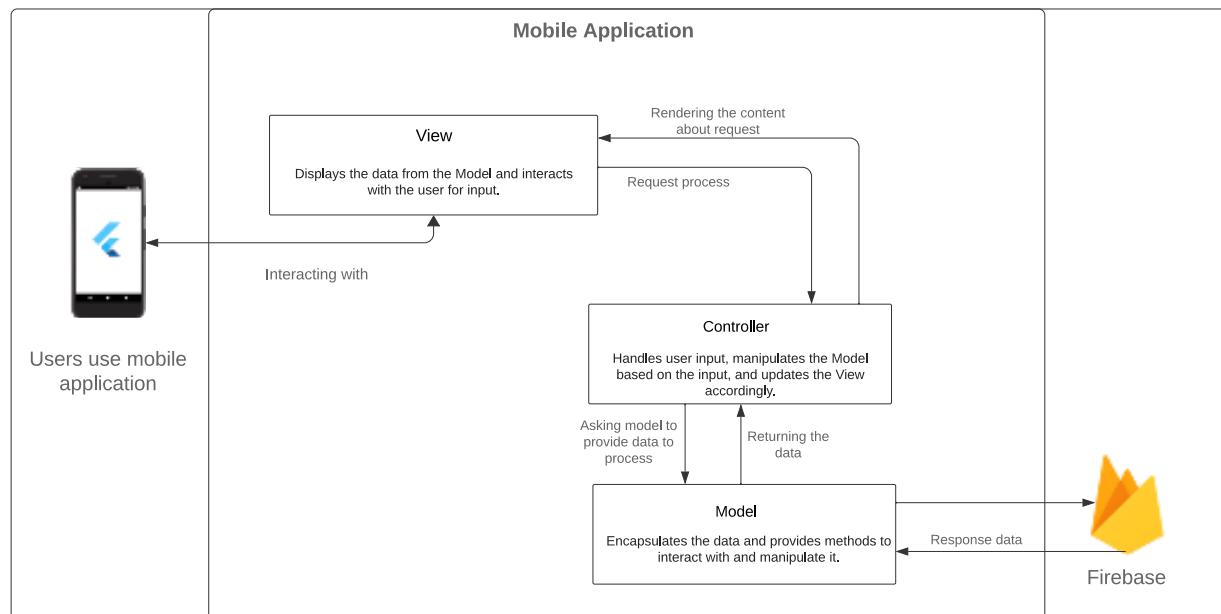


Figure 90 MVC Architecture for Mobile Application.

5.6 Conclusion

In conclusion, this chapter has provided an overview of important UML diagrams and architectural considerations in software development. Class diagrams offer insights into the static structure of a software system, while sequence diagrams depict the order of object interactions. Activity diagrams aid in visualizing and understanding processes, and the system architecture diagram showcases the relationship between different types of users and applications.

The chapter also highlighted the significance of the Model-View-Controller (MVC) architecture, which separates data management, user interface, and user input processing to enhance scalability and maintainability. By utilizing these diagrams and architectural patterns, software developers can effectively model, analyze, and design complex systems, leading to more efficient and robust software solutions. It's crucial to understand the urgency and necessity of using these tools in your software development process.

Chapter 6: Implementation

6.1 Introduction

This chapter focuses on the practical application of the design by creating a fully functional system. The tools and programming languages used in the development process will be specified, providing insight into the technical aspects of the project. Additionally, the chapter will demonstrate several code fragments that contain essential functions, giving a comprehensive overview of the system's capabilities.

6.2 Engineering Standards

The top priority in developing the Mojtamai application was to ensure that we followed the best practices and engineering standards. To evaluate the application's quality, we have integrated the ISO/IEC 25010 standard, which includes strict benchmarks for performance and effectiveness. By not just meeting, but exceeding these benchmarks, we aim to guarantee that the application performs with optimal efficiency and effectiveness, as shown in the *Figure 91*.[44], [45]

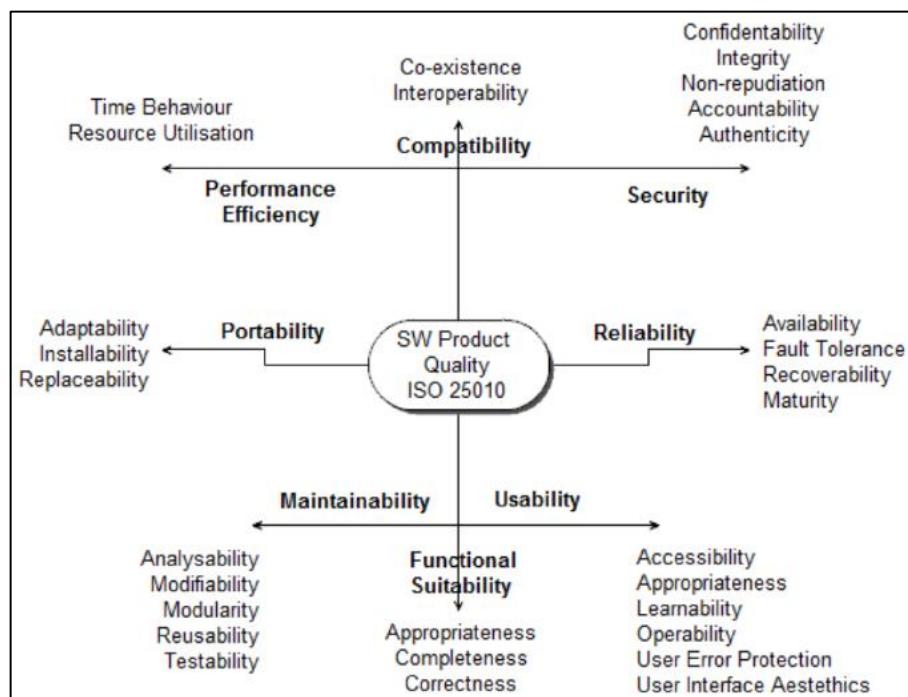


Figure 91 Engineering Standard

1. Functional Suitability:

Functional completeness: ensure the real-time features work correctly and cover all functionality.

2. Reliability:

Availability: ensure that the application is accessible and functional 24/7.

3. Usability:

Learnability: ensure the application has a user-friendly interface that is easy to navigate and understand.

4. Performance Efficiency:

Time behaviour: Efficient algorithms and technologies are utilized to achieve the specified processing time for real-time tasks.

5. Maintainability:

Reusability: ensure that the dart-written code and created dataset can be effectively utilized in other systems, promoting efficiency, and reducing redundancy.

Modifiability: make the application easy to modify without compromising the existing product quality. This includes employing modular design principles, clear documentation, and well-structured code to facilitate future enhancements and updates while maintaining the application's reliability and functionality.

6. Portability:

Adaptability: Ensure that the application runs seamlessly on different platforms and environments, especially as a mobile application. Considering the diverse technological landscape,

we aim to give users the flexibility to access and utilize the Mojtamai application on various devices and operating systems, enhancing its usability and reach.

6.3 Tools and Languages

In this section, we will show the software and programming languages used in the development phase/sprint of mobile application:

6.3.1 Flutter Framework

Flutter is the mobile UI toolkit that Google provides to create wonderfully natively compiled smartphone applications, web pages, and desktops from one codebase. Flutter allows developers and organizations worldwide to work with current code while being accessible and freely available [46].

6.3.2 Dart language

The Dart programming language for building mobile and web apps on different platforms is an Object-Oriented Client Programming Optimized Language. It allows you to use applications easily at both the user and server levels. To create front-end user interfaces for mobile applications, Dart uses the C style [47].

6.3.3 Firebase

Firebase is a Backend-as-a-Service (BaaS) app development platform that offers hosted backend services such as real-time database, cloud storage, authentication, crash reporting, machine learning, remote setup, and static file hosting [48].

6.3.4 Android Studio

Android Studio is a powerful and popular application development environment for Flutter apps. This package provides an extensive development environment to create, test, and analyze Flutter applications. There are several core features in the Android Studio for Flutter development package, including real-time analysis of code, access to creating and managing Flutter Projects and debugging solid tools such as Firebase or Gradle integration [49].

6.3.5 Visual Studio Code

Visual Studio Code is an easy-to-use, flexible text editor commonly used for Flutter development. It provides a range of features and extensions that can help streamline the development process, including syntax highlighting, code completion, debugging, and Git integration. Visual Studio Code also supports hot reloads, allowing developers to see the results of their code changes without delay. There is also a vast and active community for Visual Studio Code that offers valuable resources and support [49].

6.4 System Implementation (Code Snippets of Important Functions)

6.4.1 Mobile Application Implementation

Firstly, “common” folder, which contains essential files used throughout the entire application. These files are designed to provide shared functionality and resources, which helps to code reusability, modular and organized structure, consistency and standardization, and simplified maintenance and updates. These files are “constant”, “theme”, “common_functions”, “firebase_api” and “network_page”.

❖ Firebase API

The "FirebaseAPI" class utilizes the Firebase packages to interact with the Firebase services. Here is a brief description of the functions:

1. currentUserInfo: This method returns a 'DocumentReference' pointing to the current user's profile document in the "userProfile" collection.
2. url: This method returns a 'Uri' object representing the URL for a specific path in the Firebase Realtime Database.
3. databaseReference: This method returns a 'Stream<DatabaseEvent>' representing a real-time stream of events for a specific path in the Firebase Realtime Database. It listens for changes on the specified path and returns the stream of events.
4. fireStorageRef: This method returns a 'Reference' object representing a reference to a specific file in the Firebase Storage service. It constructs the reference using the provided path and unique file name.

```
6  class FirebaseAPI {  
7      static const String projectDatabase =  
8          'senior-project-72daf-default-rtdb.firebaseio.com';  
9      static DocumentReference<Map<String, dynamic>> currentUserInfo() {  
10         return FirebaseFirestore.instance  
11             .collection("userProfile")  
12             .doc(FirebaseAuth.instance.currentUser!.email!.split("@")[0]);  
13     }  
14  
15     static Uri url(String path) {  
16         return Uri.https(projectDatabase, '$path.json');  
17     }  
18  
19     static Stream<DatabaseEvent> databaseReference(String path) {  
20         return FirebaseDatabase.instance.ref().child(path).onValue;  
21     }  
22  
23     static Reference fireStorageRef(String path, String uniqueFileName) {  
24         return FirebaseStorage.instance  
25             .ref()  
26             .child(path)  
27             .child('$uniqueFileName.jpg');  
28     }  
29 }
```

Figure 92 Implementation of FirebaseAPI Class.

❖ Theme

The "theme" file contains two classes, "CustomColors" and "TextStyles," which define custom colors and text styles for use in Mojtamai application's visual theme. This helps in consistent visual appearance, easy customization, code reusability and improved readability.

```
4  class CustomColors {
5      static const Color red = Colors.red;
6      static const Color black = Colors.black;
7      static const Color blackLowTrans = Color(0x42000000);
8      static const Color lightBlue = Color(0xff83CCEA);
9      static const Color darkGrey = Color(0xff53D74);
10     static const Color pink = Color(0xffF2D1BE);
11     static const Color lightGrey = Color(0xFFABABAB);
12     static const Color backgroundColor = Color(0xFFFF1EFF1);
13     static const Color white = Colors.white;
14     static const Color realBlue = Colors.blue;
15     static const Color lightGreyLowTrans = Color.fromRGBO(171, 171, 171, 0.5);
16     static const Color lightBlueLowTrans = Color.fromRGBO(168, 131, 205, 234);
17     static const Color noColor = Color.fromRGBO(171, 171, 171, 0.0);
18     static const Color highlightColor = Color(0xFFE0E0E0);
19 }
20
21 class TextStyles {
22     static TextStyle pageTitle = GoogleFonts.getFont("Almarai",
23         color: CustomColors.darkGrey, fontWeight: FontWeight.w400, fontSize: 30);
24     static TextStyle pageTitle2 = GoogleFonts.getFont("Almarai",
25         color: CustomColors.darkGrey, fontWeight: FontWeight.w400, fontSize: 25);
26     static TextStyle subtitlePink = GoogleFonts.getFont("Almarai",
27         color: CustomColors.pink, fontWeight: FontWeight.bold, fontSize: 25);
28     static TextStyle subtitleGrey = GoogleFonts.getFont("Almarai",
29         color: CustomColors.darkGrey, fontWeight: FontWeight.bold, fontSize: 25);
30     static TextStyle menuTitle = GoogleFonts.getFont("Almarai",
31         color: CustomColors.darkGrey, fontWeight: FontWeight.bold, fontSize: 20);
32     static TextStyle profileTitle = GoogleFonts.getFont("Almarai",
33         color: CustomColors.lightBlue, fontWeight: FontWeight.bold, fontSize: 20);
34
35     static TextStyle heading1D = GoogleFonts.getFont("Almarai",
36         color: CustomColors.darkGrey, fontWeight: FontWeight.bold, fontSize: 18);
37     static TextStyle heading3D = GoogleFonts.getFont("Almarai",
38         color: CustomColors.darkGrey, fontWeight: FontWeight.bold, fontSize: 16);
```

Figure 93 Implementation of Theme File.

❖ Common Functions

The functions in the file include:

1. buildBottomBar: A widget function that builds a bottom app bar with a bottom navigation bar inside it. It handles navigation to different screens based on the selected index.
2. loadingFunction: A widget function that displays a loading indicator.
3. checkNetworkConnectivity: A function that checks the network connectivity.
4. isToday: A function that checks if a given date is today's date.
5. isFutureValidity: A function that checks if a given date is in the future.
6. sortItemsByTimestamp: A function that sorts a list of items based on their timestamps in descending order.
7. homeCards: A function that populates a combined list with items from different categories and sorts them by timestamp.
8. getTodayList: A function that filters the combined list to get only the items with today's date.
9. goToProfilePage: A function that navigates to the profile screen and pops the current screen.
10. showNetWidgetDialog: A function that shows an alert dialog indicating an error in internet connectivity.
11. launchURL: A function that launches a URL using the url_launcher package.

Figure 94 shows the implementation of important functions in this file.

```
⑧ senior_project > lib > common > common.functions.dart > ...
  1 > import 'package:connectivity_plus/connectivity_plus.dart'; ...
  12
  13 > Widget buildBottomBarWF(BuildContext context, int index) { ...
  172
  173 > Widget buildBottomBar(BuildContext context, int index, bool isService, ...
  179
  180 > Widget loadingFunction(BuildContext context, bool load) { ...
  181
  174 |> Future<void> checkNetworkConnectivity() async { ...
  175
  176 |> bool isToday(String time) { ...
  177
  178 |> bool isFutureValidity(String time) { ...
  179
  201 > void sortItemsByTimestamp(List<dynamic> combined) { ...
  202
  203 > void homeCards() async { ...
  204
  205 > void getTodayList() { ...
  206
  207 > void goToProfilePage(BuildContext context) { ...
  208
  209 > void showNetWidgetDialog(BuildContext context) { ...
  210
  211 > Future<void> launchURL(String? urlString, BuildContext context) async { ...
  212
  213
```

```
73 Widget buildBottomBar(BuildContext context, int index, bool isService,
74   {bool isSaved = false}) {
75   return BottomAppBar(
76     color: CustomColors.white,
77     shape: const CircularNotchedRectangle(),
78     notchMargin: 0.1,
79     clipBehavior: Clip.none,
80     child: SizedBox(
81       height: kBottomNavigationBarHeight * 1.2,
82       width: MediaQuery.of(context).size.width,
83       child: Container(
84         decoration: const BoxDecoration( // BoxDecoration ...
85         child: BottomNavigationBar(
86           showSelectedLabels: isService ? false : true,
87           landscapeLayout: BottomNavigationBarLandscapeLayout.linear,
88           onTap: (int newIndex) {
89             if (index == newIndex && !isService) { ...
90             }
91             if (newIndex == 0) {
92               Navigator.pushReplacement(context, ...
93             } else if (newIndex == 1) {
94               Navigator.pushReplacement(context, ...
95             } else if (newIndex == 2) {
96               Navigator.pushReplacement(context, ...
97             } else if (newIndex == 3) {
98               Navigator.pushReplacement(context, ...
99             }
100           },
101           unselectedItemColor: CustomColors.darkGrey,
102           selectedItemColor: isService ? CustomColors.darkGrey : CustomColors.lightBlue,
103           currentIndex: index,
104           items: [...,
105             ],
106           ), // BottomNavigationBar
107         ), // Container
108       ), // BottomAppBar
109     ), // Scaffold
110   ); // MaterialApp
111 }
```

(a) Functions Overview

```
338 Future<void> launchURL(String? urlString, BuildContext context) async {
339   if (urlString == null || urlString.isEmpty) {
340     ScaffoldMessenger.of(context).showSnackBar(
341       const SnackBar(content: Text('The URL is not available.')),
342     );
343     return;
344   }
345   final Uri url = Uri.parse(urlString);
346
347   if (!await launchUrl(url)) {
348     if (!context.mounted) {
349       return;
350     }
351     ScaffoldMessenger.of(context).showSnackBar(
352       SnackBar(content: Text('Could not launch $urlString')),
353     );
354   }
355 }
```

```
190 |     bool isFutureValidity(String time) {  
191 |         DateTime expiryDate = DateTime.parse(time);  
192 |         DateTime now = DateTime.now();  
193 |  
194 |         if (expiryDate.difference(now).inDays >= 0) {  
195 |             return true;  
196 |         } else {  
197 |             return false;  
198 |         }  
199 |     }
```

(c) launchURL Function

(d) `isFutureValidity` Function

```
174 Future<void> checkNetworkConnectivity() async {
175   final connectivityResult = await Connectivity().checkConnectivity();
176   isOffline = (connectivityResult[0] == ConnectivityResult.none);
177 }
```

(e) checkNetworkConnectivity Function

```
201 void sortItemsByTimestamp(List<dynamic> combined) {  
202     combined.sort((a, b) {  
203         DateTime timestampA = DateTime.parse(a.item.timestamp);  
204         DateTime timestampB = DateTime.parse(b.item.timestamp);  
205         return timestampB.compareTo(timestampA); // Sort in descending order  
206     });  
207 }
```

(f) sortItemsByTimestamp Function

Figure 94 Implementation of Important Functions in common functions File.

❖ Push Notification

The PushNotification class handles push notifications in the application, specifically for chat services. It handles the retrieval and management of Firebase Messaging tokens, the initialization and display of local notifications, and the sending of push notifications to specific users.

1. **FirebaseMessaging:** An instance of the FirebaseMessaging class is created to handle Firebase Cloud Messaging (FCM) functionality.
2. **FlutterLocalNotificationsPlugin:** An instance of the FlutterLocalNotificationsPlugin class is created to handle local notifications within the app.
3. **getFirebaseMessagingToken:** This method requests permission for receiving push notifications and retrieves the Firebase Messaging token.
4. **updatePushToken:** This method updates the pushToken field in the user's profile document in the "userProfile" collection of the Firestore database.
5. **firebaseInit:** This method initializes Firebase messaging by listening to incoming messages. It registers a callback function to handle incoming messages and triggers local notifications using the initLocalNotifications() method.
6. **initLocalNotifications:** This method initializes local notifications by configuring the AndroidNotificationChannel and AndroidNotificationDetails. It also initializes the FlutterLocalNotificationsPlugin with the provided settings.
7. **showNotification:** This method displays a local notification when a push notification is received. It checks if the app is in the foreground and, if not, creates a customized notification appearance.

8. **sendPushNotification**: This method sends a push notification to a specific user.

9. **foregroundMessage()**: This method configures the foreground notification presentation options for the FirebaseMessaging instance.

```
19 // for getting firebase messaging token
20 Future<void> getFirebaseMessagingToken() async {
21     await fMessaging.requestPermission();
22     await fMessaging.getToken().then((t) async {
23         if (t != null) {
24             userInfo.pushToken = t;
25         }
26     });
27 }
28 }
29 );
30 }
31
32 Future<void> updatePushToken() async {
33     FirebaseFirestore.instance
34         .collection('userProfile')
35         .doc(userInfo.userID)
36     .update({ ...
37 })
38 }
39
40 void firebaseInit(BuildContext context) {
41     FirebaseMessaging.onMessage.listen((message) {
42         initLocalNotifications(context, message);
43         showNotification(message);
44     });
45 }
46 }
```

(a) Tokens manage Functions and firebaseInit

```
// for sending push notification
Future<void> sendPushNotification(
    UserInformationModel chatUser, String msg) async {
try {
    final body = {
        "to": chatUser.pushToken,
        "notification": {
            "title": userInfo.name, //our name should be send
            "body": msg,
            "android_channel_id": "chat_rooms"
        },
        "android": {...}
    };

    var res = await post(Uri.parse('https://fcm.googleapis.com/fcm/send'),
        headers: {...},
        body: jsonEncode(body));
} catch (e) {
    if (kDebugMode) {...}
}
}

Future foregroundMessage() async {
    await fMessaging.setForegroundNotificationPresentationOptions(
        alert: true,
        badge: true,
        sound: true,
    );
}
```

(b) send notification and foreground message Function

```
48 Future<void> initLocalNotifications(
49     BuildContext context, RemoteMessage message) async {
50     var androidInit =
51         const AndroidInitializationSettings('@mipmap/launcher_icon');
52     var initSet = InitializationSettings(...);
53
54     await _flutterLocalNotificationsPlugin.initialize(initSet, ...);
55 }
56
57 Future<void> showNotification(RemoteMessage message) async {
58     if (isForeground) // App is in the foreground, do not show the notification
59     return;
60 }
61
62 AndroidNotificationChannel channel = AndroidNotificationChannel( // Android
63
64     AndroidNotificationDetails androidNotificationDetails =
65         AndroidNotificationDetails(...),
66
67     NotificationDetails notificationDetails =
68         NotificationDetails(android: androidNotificationDetails),
69
70     Future.delayed(Duration.zero, () { ...
71 })
72 }
```

(c) initLocalNotification and show notification Function

Figure 95 Implementation of Push Notification.

❖ App Setup

The ‘app_setup’ file contains Setup class, which is a controller class. It is responsible for setting up and initializing various components in the application and loading data from firebase.

```
1 > import 'dart:convert'; ...
24
25 > class Setup {
26 >   Future<void> build() async { ...
35
36 >   Future<void> build2() async { ...
46
47 >   Future<void> loadPsychGuidance() async { ...
71
72 >   static Future<void> loadUserData(String enteredID) async { ...
146
147 >   Future<void> loadAllUsers() async { ...
182
183 >   Future<void> loadOffers() async { ...
238
239 >   Future<void> loadCourses() async { ...
288
289 >   Future<void> loadWorkshops() async { ...
338
339 >   Future<void> loadConferences() async { ...
388
389 >   Future<void> loadOtherEvents() async { ...
439
440 >   Future<void> loadVolOp() async { ...
497
498 >   Future<void> loadSClubs() async { ...
530
531 >   Future<void> loadSActivities() async { ...
585
586 >   Future<void> loadStudyGroups() async { ...
635
636 >   Future<void> loadFoundItems() async { ...
663
664 >   Future<void> loadLostItems() async { ...
690
691 >   Future<void> loadClinics() async { ...
718 }
719 }
```

Figure 96 App Setup Functions.

❖ Lunch Screen

The LaunchScreen class handles the initial setup and rendering of the application's launch screen. It checks connectivity, verifies the user's authentication token, and navigates to the appropriate screens based on the authentication status and device connectivity.

In lunch screen there is several important functions:

1. `checkConnectivity`: This method checks the device's connectivity status using the Connectivity plugin. It updates the value of the "isOffline" variable based on the connectivity result, indicating whether the device is offline or not.
2. `_checkToken`: This method checks the user's authentication token and navigates to the appropriate screen based on the token's availability and the device's connectivity status.

```
46 void checkConnectivity(List<ConnectivityResult> result) {
47     switch (result[0]) {
48         case ConnectivityResult.mobile || ConnectivityResult.wifi:
49             if (isOffline != false) {
50                 setState(() {
51                     isOffline = false;
52                 });
53             }
54             break;
55         case ConnectivityResult.none:
56             if (isOffline != true) {
57                 setState(() {
58                     isOffline = true;
59                 });
60             }
61             break;
62         default: ...
63     }
64 }
```

(a) check connectivity function

```
Future<void> _checkToken() async {
try {
    final user = FirebaseAuth.instance.currentUser;
    if (isOffline) {
        await Navigator.pushReplacement(context,
            MaterialPageRoute(builder: (context) => const NetworkConnection()));
    } else {
        if (user != null) {
            if (!isOffline) {
                Future.delayed(const Duration(seconds: 1), () {
                    Navigator.pushReplacement(context,
                        MaterialPageRoute(builder: (context) => const HomeScreen()));
                }); // Future.delayed
            } else {
                Future.delayed(const Duration(seconds: 1), () {
                    Navigator.pushReplacement(context,
                        MaterialPageRoute(builder: (context) => const LoginScreen()));
                }); // Future.delayed
            }
        } catch (e) {
            if (kDebugMode) { ... }
        }
        Navigator.pushReplacement(context,
            MaterialPageRoute(builder: (context) => const LoginScreen()));
    }
}
```

(b) check token Function

Figure 97 Implementation of Lunch Screen

❖ Login Screen

Mobile application login is made by submitting form with text fields for a university ID and password. It also handles the situation of absence of internet connectivity.

The `_submit` method is called when the login button is pressed. It performs form validation and attempts to sign in the user using the `firebase_auth` package. If the sign-in is successful, it loads user data and navigates to the home screen (`HomeScreen`). If there is an authentication exception, an error message is displayed.



```
void _submit() async {
    final isValid = _formKey.currentState!.validate();
    setState(() {
        errorMessage = '';
    });
    if (!isValid) {
        return;
    }
    _formKey.currentState!.save();
    try {
        setState(() => isLoading = true);
        final userCredential = await firebase.signInWithEmailAndPassword(
            email: _enteredID,
            password: _enteredPass,
        );

        await Setup.loadUserData(_enteredID);
        await Setup().build();
        Setup().build2();
        if (mounted) {
            Navigator.pushReplacement(
                context,
                MaterialPageRoute(builder: (_) => const HomeScreen()),
            );
        }
    } on FirebaseAuthException {
        setState(() {
            errorMessage = 'الرقم الجامعي أو الرقم السري غير صحيح، حاول مرة أخرى';
        });
        if (kDebugMode) {
            print('Authentication Failed');
        }
    } finally {
        setState(() => isLoading = false);
    }
}
```

(a) network handling

(b) submit Function

Figure 98 Implementation of Login

❖ Save List

The purpose of the save list is allowing the user to save events, volunteering opportunities, study groups, student activities they are interested in to remember it later. First, retrieve items previously saved from database (in app setup). And to make save list functionally support reusability and maintainability, it separated into:

1. Model: responsible for managing the saving and retrieving of either locally or with database.
Most important function is handleSaveItem, which is used to add or remove an item from the saveList. It takes a boolean parameter isSaved that indicates whether the item is currently saved or not. It toggles the value of isSaved and performs the necessary operations to add or remove the item from the saveList based on the updated isSaved value. It also calls the addItem or removeItem methods to update the item in the Firestore database.
2. Widget: clickable card represents a single item.
3. Screen: handle situation of empty list and display list of saved items.

```
class SavedListModel {
    final String serviceName;
    final dynamic dynamicObject;
    final String icon;
    SavedListModel({
        required this.serviceName,
        required this.dynamicObject,
        required this.icon,
    });
    Map<String, String> convertServiceName = {...};

    // check item id
    static bool findId(String id) {...}

    bool handleSaveItem(bool isSaved) {
        isSaved = !isSaved;
        if (isSaved) {
            saveList.add(
                DynamicItemModel(
                    serviceName: convertServiceName[serviceName].toString(),
                    item: dynamicObject,
                    icon: icon),
            );
            addItem(dynamicObject.id.toString());
        } else {
            saveList.removeWhere((item) =>
                item.serviceName == convertServiceName[serviceName].toString() &&
                item.item.id == dynamicObject.id &&
                item.icon == icon);
            removeItem(dynamicObject.id.toString());
        }
        return isSaved;
    }

    //add item to firebase
    Future<void> addItem(String itemId) async {...}

    //remove item from firebase
    Future<void> removeItem(String itemId, {bool isSaveScreen = false}) async {...}
}
```

(a) handle save item function

```
class _SaveCardState extends State<SaveCard> {
    Widget build(BuildContext context) {
        bool isSaved;
        SavedListModel savedItem = SavedListModel(
            serviceName: widget.serviceName,
            dynamicObject: widget.dynamicObject,
            icon: widget.icon);
        isSaved = SavedListModel.findId(widget.dynamicObject.id);

        return isSaved
            ? InkWell(
                onTap: () {
                    if (widget.serviceName == 'فرصه خطيوهين') {
                        Navigator.push(...);
                    } else if (widget.serviceName == 'الذكاء الاصناف') {
                        Navigator.push(...);
                    } else if (widget.serviceName == 'حلقة مذاكرات') {
                        Navigator.push(...);
                    } else {
                        Navigator.push(...);
                    }
                },
                child: Card( // Card ...
            ) // InkWell
            : Container();
    }
}
```

(b) navigation handling

Figure 99 Implementation of Save List

❖ Live Chat

Live chat service includes several screens, widgets, and models to achieve a high degree of usability, reusability, readability, and maintainability.

- **Current chats screen:** display current chat conversations. It fetches data from a Firestore collection named "chat_rooms" and displays the list of users and their chat information in form of "UserChatItem" widget card. This list is dynamically updated as new chat rooms are added or existing ones are modified. This done by help of StreamBuilder widget that listens to changes in the Firestore collection "chat_rooms".

- **All users screen:** display a list of all users in form of "UserChatItem" widget card.

- **Chat screen:** display a list of messages in form of "ChatBubble" widget card. It uses a StreamBuilder widget to listen to real-time updates from the ChatService and displays the messages accordingly.

- **Chat services:** is it a controller class that encapsulates various methods for interacting with Firebase Firestore to send and receive chat messages. It extends the ChangeNotifier class because it needs to notify its listeners when its internal state changes.

- sendMessage method: responsible for sending a chat message to another user. It takes the "UserInformationModel" object representing the other user and the message as parameters. in the method, it constructs a new "MessageInfoModel" object with the message details and the current timestamp. If the chat room does not exist, it creates a new chat room document in Firestore with initial chat information. It updates the chat room document with the latest message details. Finally, it adds the message to the chat room's messages subcollection and sends a push notification to the other user.

- getMessage method: retrieve the chat messages between two users. It takes the user IDs of the two users as parameters. The method returns a Stream<QuerySnapshot> that listens for real-time updates to the chat room's messages subcollection, ordered by the message timestamp in descending order.
- updateMessageReadStatus method: update the read status of a message. It takes a "MessageInfoModel" object representing the message as a parameter. It updates the corresponding message document in the chat room's messages subcollection with the current timestamp as the read status. Additionally, it updates the chat room document with the current timestamp as the read status.

```
class ChatService extends ChangeNotifier {
  //send
  Future<void> sendMessage(
    UserInformationModel otherUserInfo, String message) async {
  //get current user info ----
  try {
    final time = DateTime.now().millisecondsSinceEpoch.toString();

    //create new message
    MessageInfoModel newMessage = MessageInfoModel(...);

    //construct chat room id -> sender+receiver
    List<String> ids = [userInfo.userID, otherUserInfo.userID];
    ids.sort();
    String chatRoomID = ids.join("_");

    if (!await isChatExist(chatRoomID)) {
      ChatInfo newChat = ChatInfo(...);

      await _firestore
        .collection("chat_rooms")
        .doc(chatRoomID)
        .set(newChat.toMap());
    }

    await _firestore.collection("chat_rooms").doc(chatRoomID).update({ ... });
    //add the message to firestore
    await _firestore
      .collection("chat_rooms")
      .doc(chatRoomID)
      .collection("messages")
      .doc(time)
      .set(newMessage.toMap())
      .then((value) => notificationServices.sendPushNotification(...));
  } catch (error) {
    if (kDebugMode) { ...
  }
}
```

(a) send message function

```
class ChatService extends ChangeNotifier {
  //get
  Stream<QuerySnapshot> getMessage(String userID, String otherUserID) {
    List<String> ids = [userID, otherUserID];
    ids.sort();
    String chatRoomID = ids.join("_");

    return _firestore
      .collection("chat_rooms")
      .doc(chatRoomID)
      .collection("messages")
      .orderBy('time', descending: true)
      .snapshots();
  }

  Future<void> updateMessageReadStatus(MessageInfoModel message) async {
    List<String> ids = [message.senderID, message.receiverID];
    ids.sort();
    String chatRoomID = ids.join("_");
    var time = DateTime.now().millisecondsSinceEpoch.toString();
    _firestore
      .collection("chat_rooms")
      .doc(chatRoomID)
      .collection("messages")
      .doc(message.time)
      .update({ 'readF': time });

    _firestore.collection("chat_rooms").doc(chatRoomID).update({ 'readF': time });
  }
}
```

(b) get message and update message status Function

Figure 100 Implementation of Live Chat- ChatService Class

- User chat item widget: It represents a single item in the user's chat list. It contains a user profile picture and other information related to the user. When the user taps on the chat item, it will navigate to the chat screen. And when the user taps on the profile picture, it will navigate to the other user's profile screen.

```
class UserChatItem extends StatelessWidget {
    Widget build(BuildContext context) {
        child: InkWell(
            onTap: () {
                Navigator.push(
                    context,
                    MaterialPageRoute(
                        builder: (context) => RealChatPage( // RealChatPage ...
                    ), // MaterialPageRoute
                );
            },
            child: ListTile(
                title: Text( // Text ...
                subtitle: Text( // Text ...
                trailing: chatInfo == null
                    ? null //show nothing when no message is sent
                    : (chatInfo!.lastMsgSender != userInfo.userID &&
                        | chatInfo!.readF.isEmpty)
                    ? //show for unread message
                        Container( // Container ...
                            // message sent time
                            : Text( // Text ...
                leading: GestureDetector(
                    onTap: () {
                        Navigator.push(
                            context,
                            MaterialPageRoute(
                                builder: (context) => OtherUserProfileScreen( // ...
                            ), // MaterialPageRoute
                        );
                    };
                ),
            ),
        ),
    
```

Figure 101 Implementation of Live Chat - UserChatItem Widget

❖ Exclusive Offers

The exclusive offers are categorized into groups. Each group contains a list of offers that belongs to that category. Each item is clickable and will display the details of the offer. This process is done through several components (OfferCategoriesScreen, OffersListScreen, OfferDetailsScreen, OfferInfoModel, OfferCard) to ensure highest degree of usability, readability, and maintainability.

The exclusive offers recommendation is made by allowing the user to choose his preferences from offers categories in his profile, then according to it the recommended offers will show in the home page.

```
int falseNum = 0;
List<dynamic> tempOffer = [];
recommendedOffers = [];
for (Map<String, dynamic> item in offers) {
    if (userInfo.offersPreferences[item['offerCategory']] == true) {
        recommendedOffers.addAll(item['categoryList']);
    } else {
        tempOffer.addAll(item['categoryList']);
        falseNum++;
    }
}
if (falseNum > 9 || recommendedOffers.isEmpty) {
    recommendedOffers = tempOffer;
}
```

Figure 102 Exclusive Offers Recommendation

❖ Lost and Found

This service uses one screen for displaying the lost and found items. Two screens are forms for creating lost and found reports. Here the common important functions between the lost and found items.

1. selectDate: This function is used to show a date picker dialog and update the selected date based on the user's input.
2. takePhoto: This function is used to take a photo using the device's camera or gallery and set it as the selected image.
3. checkInputValue: This function is used to validate the form input values and perform additional actions.
4. createFoundItem: This function is used to create a new found or lost item by making an HTTP POST request to a specific URL to firebase.

```
Future<void> _selectDate(BuildContext context) async { ...  
  
void _takePhoto() async { ...  
  
void _checkInputValue() async { ...  
  
void _createFoundItem() async { ...  
  
@override  
Widget build(BuildContext context) { ...
```

Figure 103 Implementation of Lost and Found

❖ Student Activity and Study Group

The StudentActivityScreen and StudyGroupScreen classes in the app include several screens, widgets, and models to achieve a high degree of usability, reusability, readability, and maintainability.

These classes allow users to manage and explore student activities and study groups efficiently. They provide real-time updates from Firebase using StreamBuilder widget, enable users to search and filter specific activities or groups. Additionally, the classes incorporate connectivity checks. In both classes, there are several important functions:

1. `selectDate`: This function is used to show a date picker dialog and update the selected date based on the user's input.
2. `selectTime`: This function is used to show a time picker dialog and update the selected time based on the user's input.
3. `checkInputValue`: This function is used to validate the form input values and perform additional actions.
4. `createStudentActivityState`: This function is used to create a new student activity by making an HTTP POST request to a specific URL.

```
Future<void> _selectDate(BuildContext context) async { ... }

Future<void> _selectTime(BuildContext context) async { ... }

void _checkInputValue() async { ... }

void _createStudentActivityState() async { ... }
```

Figure 104 Implementation of Study Group and Student Activity

❖ Clinic Schedule

The Clinics service is designed to enhance user interaction by providing a comprehensive overview of available clinics related to specific branches, organized, and sorted by date for easy navigation. Users can explore detailed listings of clinics, each topped with a date header to clearly distinguish between different sessions. The class allows users to filter clinics by various criteria, ensuring they find exactly what they need. Additionally, interactive elements like search bars and dynamic selection chips make the interface user-friendly and responsive. For added convenience, a direct link to book an appointment is also provided, facilitating a smooth and efficient user experience.

`_filterClinicListByBranch`. This function sorts and displays clinics based on the selected branch and arranges them by date. This helps users easily find the most relevant and timely clinic information, enhancing their experience and making the app easier to navigate.

```
void _filterClinicListByBranch() {
    if (selectedBranch.isEmpty) {
        filteredClinicList = clinicItems
            .where((clinic) => clinic.clBranch == "المقر الرئيسي")
            .toList();
    } else {
        filteredClinicList = clinicItems
            .where((clinic) => clinic.clBranch == selectedBranch)
            .toList();
    }
    filteredClinicList.sort((a, b) => DateFormat('yyyy-MM-dd')
        .parse(a.clDate!)
        .compareTo(DateFormat('yyyy-MM-dd').parse(b.clDate!)));
}
```

Figure 105 Clinic Schedule - Filter by Branch

6.4.2 Admin Panel – Web Application Implementation

❖ Login and Logout

The Login Controller handles the application login and logout functionality. It validates user input, authenticates users using Firebase authentication, and checks if the user is an admin by querying the Firebase Realtime Database. The controller interacts with the session to store and retrieve user data.

- Connect function: establishes a connection to the Firebase Realtime Database. It uses the Factory class from the kreait/firebase-php package to create a database instance with the provided Firebase service account credentials and database URL.

- Index function: renders the login view and returns the login view file.

- Logout function: handles the user logout process. It clears the session data and redirects the user to the login page.

- Login function: handles the user login process. It expects a Request object that contains the user's login credentials (email and password). To ensure the security of the user's information

- It first validates the user's input using the Validator class. It redirects to the login page if validation fails, displaying the validation errors and input data.
- Once validation is successful, it attempts to authenticate the user by calling the signInWithEmailAndPassword() method from the Firebase authentication service. It uses the provided email and password and appends the '@uj.edu.sa' domain to the email.
- If the authentication process is successful, it retrieves the user's email and checks if the user is an admin by querying the Firebase Realtime Database.

- If the user is an admin, the function stores the admin data in the session and redirects it to the home view.
- If the user is not an admin, it sends an error message and redirects to the login page.
- If any exception occurs during the authentication process, it sends an error message and redirects to the login page.

```

class LoginController extends Controller
{
    public function connect()
    {
        $firebase = (new Factory)
            ->withServiceAccount(base_path(env('FIREBASE_CREDENTIALS')))
            ->withDatabaseUri(env("FIREBASE_DATABASE_URL"));
        return $firebase->createDatabase();
    }
}

public function index() ...
}

public function logout() ...
}

public function login(Request $request)
{
    $messages = [ ... ];
    $validator = Validator::make($request->all(), [ ... ],
        $messages);

    if ($validator->fails()) { ... }

    try {
        $message = 'الرقم الجامعي او كلمة المرور غير صحيحة';
        $firebaseAuth = Firebase::auth();

        $signInResult = $firebaseAuth->signInWithEmailAndPassword(
            $request->email . '@uj.edu.sa', $request->password);
        $user = $signInResult->data();
        $email = $request->email;
        // Check user's role
        $adminRef = $this->connect()->getReference('Admins')->getChild($email);
        $isAdmin = $adminRef->getValue();
        if ($isAdmin) {
            // User is an admin, After successful authentication
            session(['user' => $isAdmin]);
            return view('home');
        } else { ... }
    } catch (\Exception $e) { ... }
}

```

Figure 106 Implementation of Login in Admin Panel

The admin panel has seamlessly incorporated the MVC architecture, providing all controllers with a standard set of functions. These functions, namely index(), create(), store(), edit(), update(), and destroy(), interact with different database sources for each controller. We have confidently identified the most critical functions for one controller and found that they are equally applicable to other controllers.

The important functions from the workshop event controller:

- ◆ Connect function: initializes and returns a connection to the Firebase Realtime Database using the provided credentials and database URL. This connection can interact with the database and perform operations like reading and writing data.
- ◆ Index function: retrieves workshop data from the Firebase Realtime Database and passes it to a view for display. This allows users to view the list of workshops stored in the database.
- ◆ Store function: responsible for storing a new workshop in the Firebase Realtime Database. It does this by validating the data that is submitted through a request, adding a timestamp to the data, and then storing the information in the database if the validation is successful.
- ◆ Create function: displays the form view that allows users to create a new workshop. It also provides the id variable to the view, which is set to null by default. The view can utilize this variable to handle display requirements related to creating a new workshop.
- ◆ Edit function: retrieves the data of a specific workshop from the Firebase Realtime Database based on the given ID. Then, it renders the form view (eventsWorkshop-form) with the retrieved workshop data. This allows the user to edit the workshop information using the pre-filled form fields.

- ◆ Update function: responsible for modifying an existing workshop in the Firebase Realtime Database, using the data provided through a request. Before updating the workshop, the update() function validates the input data, adds a timestamp to the data, and only proceeds with the update if the validation is successful.
- ◆ Destroy function: used to remove a workshop from the Firebase Realtime Database. If there are no workshops left, a placeholder value is set to maintain the reference. Once the workshop is deleted, the user is redirected to the previous page.

```
class WorkshopsController extends Controller
{
    public function connect()
    {
        $firebase = (new Factory)
            ->withServiceAccount(base_path(env('FIREBASE_CREDENTIALS')))
            ->withDatabaseUri(env("FIREBASE_DATABASE_URL"));
        return $firebase->createDatabase();
    }
0 references | 0 overrides
public function index()
{
    $eventsWorkshopsDB = $this->connect()->
        getReference('eventsWorkshopsDB')->getSnapshot()->getValue();
    return view('eventsWorkshops-list')
        ->with([
            'eventsWorkshops' => $eventsWorkshopsDB
        ]);
}
0 references | 0 overrides
public function store(Request $request)
{
    $messages = [...];
    $validator = Validator::make($request->all(), [...], $messages);

    if ($validator->fails()) {
        return redirect()->back()->withErrors($validator)->withInput();
    }
    $requestData = $request->except(['_token']);
    $requestData['timestamp'] = Carbon::now();

    $this->connect()->getReference('eventsWorkshopsDB')
        ->push($requestData);
    return redirect()->route('workshops.index');
}
0 references | 0 overrides
public function create() ...
}
```

(a) connect, store and create functions

```
class WorkshopsController extends Controller
{
    public function edit($id)
    {
        $workshop = $this->connect()->getReference('eventsWorkshopsDB')
            ->getChild($id)->getValue();
        return view('eventsWorkshop-form')->with([
            'workshop' => $workshop,
            'id' => $id
        ]);
    }
0 references | 0 overrides
public function update($id, Request $request)
{
    $messages = [...];
    $validator = Validator::make($request->all(), [...], $messages);

    if ($validator->fails()) {
        return redirect()->back()->withErrors($validator)->withInput();
    }
    $requestData = $request->except(['_token', '_method']);
    $requestData['timestamp'] = Carbon::now();

    $this->connect()->getReference('eventsWorkshopsDB/' . $id)
        ->update($requestData);
    return redirect()->route('workshops.index');
}
0 references | 0 overrides
public function destroy($id)
{
    $this->connect()->getReference('eventsWorkshopsDB/' . $id)->remove();
    // Check if there are no items left
    $eventsWorkshopsDB = $this->connect()->getReference('eventsWorkshopsDB')
        ->getSnapshot()->getValue();

    if ($eventsWorkshopsDB === null) {
        // Set a placeholder value to keep the reference
        $this->connect()->getReference('eventsWorkshopsDB')->set('placeholder');
    }
    return back();
}
```

(b) edit, update, and destroy functions

Figure 107 Implementation of Workshops Controllers

6.5 Conclusion

The main functionalities of our system have been represented in this chapter by language, tools, and a code snippet. In the next chapter, we will test each function of this system through unit tests, acceptance tests, integration tests, System Testing, and Usability Tests.

Chapter 7: Testing

7.1 Introduction

These days, testing has become an essential procedure that all mobile developers must follow to ensure that the features function as intended. This is because software development, particularly mobile development, constantly evolves due to the constant need to add new features and update outdated technologies to assist our clients in their daily lives. The testing process is represented in this chapter. An application must pass a rigorous testing process before being released to guarantee it functions as intended. Unit testing, integration testing, system testing, and acceptance testing are the four primary testing phases that must be finished before a program is approved for usage [50] [51].

7.2 Unit Testing

Unit testing assesses specific components of software, such as functions or procedures, to ensure they function as designed. It allows quick issue resolution by testing each unit independently and can be executed whenever the code is modified [50].

Table 33 Unit Testing

Test ID	Test Case Description	Test Data	Expected Results	Actual Results	Pass /Fail
Mobile Application					
Login					
TC1	Check user login with a valid ID and password	ID: 2112235 Password: 123456	Navigate to the next page	As expected,	pass
TC2	Check user login with an invalid ID and password	ID: 211223 Password: 123457	display an error message	A message indicates that the ID should be seven digits	pass
TC3	Check user login with null ID and password	ID: Password:	Display an error message	A message indicates that the ID and password	pass

				should not be empty	
TC4	Check user login with does not exist ID and password	ID: 1234567 Password:987654	display an error message	A message indicates that the ID or password is not correct	pass
TC5	Check user login with an ID that contains character and password	ID: 123457, Password:987654	display an error message	There is a restriction on the input values. The characters are not clickable,	pass

Profile Page

TC6	Change profile picture from camera	-Take a photo from the device. -Click update button	Picture updated successfully	As expected,	pass
TC7	Change profile picture from galley	-Choose a photo from the device. -Click update button	Picture updated successfully	As expected,	pass
TC8	Update user profile with no internet	Interest: المكياج، تصميم الازياء Hobbies: السباحة، القراءة Skills: عمل دورات، تقديم والقاء، البرمجة Offers preferences: Choose education and training, remove travel and hotels. -Click update button	Informatic message indicates that the update failed because the internet connection.	Message: لم يتم تحديث البيانات لتعذر الاتصال بالإنترنت	pass
TC9	Update user interest, hobbies, skills, and offers preferences	Interest: المكياج، تصميم الازياء Hobbies: السباحة، القراءة Skills:	Informatic message indicates that the information updated successfully	Message: تم تحديث البيانات بنجاح	pass

		<p>عمل دورات، تقديم والقاء، البرمجة</p> <p>Offers preferences: Choose education and training, remove travel and hotels. -Click update button</p>			
--	--	--	--	--	--

Exclusive Offers

TC10	Check the Exclusive Offers page with no cards	-	Display no-content image	As expected,	pass
TC11	Check the ability to click the cards and see the details related to the selected card	-	View full details of the selected card	As expected,	pass
TC12	Check the ability to click on the Get Discount button and see the discount code	-	View the discount code once click on get Discount button	As expected,	pass
TC13	Check the internet connectivity when it's turning off on the Exclusive Offers page	-	A message indicates that the device is not connected to the internet	Display an image that shows there is no internet connection	pass

Volunteering Opportunity

TC14	Check the Volunteering Opportunities page with no cards	-	Display no-content image	As expected,	pass
TC15	Check the ability to save the Volunteering Opportunities card and move it to the saved list	-	The card is saved successfully and it appears in the saved list	As expected,	pass

TC16	Check the Volunteering Opportunities سجل button	-	The link is working properly and is ready to fill out the form	As expected,	pass
TC17	Check the internet connectivity when it's turning off on the Volunteering Opportunities page	-	A message indicates that the device is not connected to the internet	Display an image that shows there is no internet connection	pass

Student Clubs

TC18	Check the internet connectivity while its off on the Student Clubs page	-	A message indicates that the device is not connected to the internet	Display an image that shows there is no internet connection	pass
TC19	Check choosing a specific Student Clubs	Select data: نادي هندسة البرمجيات	Navigate to the Student Clubs page and display Student Clubs information and registration links	As expected,	pass
TC20	Check the internet connectivity while its off on the Student Clubs page	-	A message indicates that the device is not connected to the internet	Display an image that shows there is no internet connection	pass

Clinic Schedule

TC21	Check the clinic page with no cards	-	Display no-content image	As expected,	pass
TC22	Check clinic search with valid input	Search input: طب عام	List of clinics that contain the search input in specific branch	As expected,	pass
TC23	Check clinic search with null input	Search input:	Display a list of all available clinics	As expected,	pass
TC24	Check clinic search with does not exist search input	Search input: الطب النفسي in المقر الرئيسي	A message indicates that the input does not exist	The image shows that there is no clinic data found	pass
TC25	Check the clinic booking button	-	Open the link correctly	As expected,	pass

Events					
Coerces					
TC26	Check Courses filter when empty	-	Display an image that shows there is no Courses	As expected,	pass
TC27	Check Courses search with valid input	Search input: اساسيات البرمجة ال شيئاً	List of Course that contain the search input	As expected,	pass
TC28	Check Courses search with null input	Search input:	Display all Courses	As expected,	pass
TC29	Check Courses search with does not exist search input	Search input: jsdkjdj	A message indicates that the input does not exist	An image shows that there is no Course with this name	pass
TC30	Check Course registration button	-	Navigate the user to the google form	As Expected,	pass
TC31	Check Couse save to list	-	The save icon change to blue and displayed at the Save List page	As Expected,	pass
TC32	Check the internet connectivity while its off on the Courses page	-	A message indicates that the device is not connected to the internet	Display an image that shows there is no internet connection	pass
TC33	Check the internet connectivity while its off in the Courses search	Search input: اساسيات البرمجة ال شيئاً	Display an image that shows there is no internet connection when the internet is off then when it's on it returns to the same search input and content on the page	As expected,	pass

Workshops

TC34	Check Workshops filter when empty	-	Display an image that shows there is no Workshops	As expected,	pass
TC35	Check Workshops search with valid input	Search input: التوثيق والاقتراض في البحث العلمي	List of Workshops that contain the search input	As expected,	pass

TC36	Check Workshops search with null input	Search input:	Display all Workshops	As expected,	pass
TC37	Check Workshops search with does not exist search input	Search input: jsdkjdj	A message indicates that the input does not exist	An image shows that there is no Workshops with this name	pass
TC38	Check Workshop registration button	-	Navigate the user to the google form	As Expected,	pass
TC39	Check Workshop save to list	-	The save icon change to blue and displayed at the Save List page	As Expected,	pass
TC40	Check the internet connectivity while its off on the Workshops page	-	A message indicates that the device is not connected to the internet	Display an image that shows there is no internet connection	pass
TC41	Check the internet connectivity while its off in the Workshops search	Search input: التوثيق والاقتباس في البحث العلمي	Display an image that shows there is no internet connection when the internet is off then when it's on it returns to the same search input and content on the page	As expected,	pass

Conferences

TC42	Check Conferences filter when empty	-	Display an image that shows there is no Conferences	As expected,	pass
TC43	Check Conferences search with valid input	Search input: جودة الحياة	List of Conferences that contain the search input	As expected,	pass
TC44	Check Conferences search with null input	Search input:	Display all Conferences	As expected,	pass
TC45	Check Conferences search with does not exist search input	Search input: jsdkjdj	A message indicates that the input does not exist	An image shows that there is no Conferences with this name	pass

TC46	Check Conference registration button	-	Navigate the user to the google form	As Expected,	pass
TC47	Check Conference save to list	-	The save icon change to blue and displayed at the Save List page	As Expected,	pass
TC48	Check the internet connectivity while its off on the Conferences page	-	A message indicates that the device is not connected to the internet	Display an image that shows there is no internet connection	pass
TC49	Check the internet connectivity while its off in the Conferences search	Search input: جودة الحياة	Display an image that shows there is no internet connection when the internet is off then when it's on it returns to the same search input and content on the page	As expected,	pass

Other Events

TC50	Check Other Events filter when empty	-	Display an image that shows there is no Other Events	As expected,	pass
TC51	Check Other Events search with valid input	Search input: اليوم العالمي للتوحد	List of Other Events that contain the search input	As expected,	pass
TC52	Check Other Events search with null input	Search input:	Display all Other Events	As expected,	pass
TC53	Check Other Events search with does not exist search input	Search input: jsdkjdj	A message indicates that the input does not exist	An image shows that there is no Other Events with this name	pass
TC54	Check event registration button	-	Navigate the user to the google form if registration is needed for the event	As Expected,	pass
TC55	Check events save to list	-	The save icon change to blue and displayed at the Save List page	As Expected,	pass

TC56	Check the internet connectivity while its off on the Other Events page	-	A message indicates that the device is not connected to the internet	Display an image that shows there is no internet connection	pass
TC57	Check the internet connectivity while its off in the Other Events search	Search input: اليوم العالمي للتوحد	Display an image that shows there is no internet connection when the internet is off then when it's on it returns to the same search input and content on the page	As expected,	pass

Lost and Found Items

TC58	Check lost items search with valid input	Search input: المستندين	List of lost items that contain the search input	As expected,	pass
TC59	Check lost items search with null input	Search input:	Display a list of all lost items	As expected,	pass
TC60	Check lost items search with does not exist search input	Search input: i9ji9do	A message indicates that the input does not exist	The image shows that there is no item found	pass
TC61	Check found items search with valid input	Search input: المستندين	List of found items that contain the search input	As expected,	pass
TC62	Check search with null input	Search input:	Display a list of all found items	As expected,	pass
TC63	Check search with does not exist search input	Search input: i9ji9do	A message indicates that the input does not exist	The image shows that there is no item found	pass
TC64	Check the creation of a lost item report with valid inputs	Image: Select valid image Category: مجهرات Lost date: 9-9-2023 Expected place: قاعة 9 Phone number: 0542241626	Back to the lost and found items page and display the new lost item report in the list	As expected,	pass

		Description: اسورة عليها تعليقة فيل			
TC65	Check the creation of a new lost item report with null inputs	Image: Category: Lost date: Expected place: Phone number: Description:	Display informative error message	Error message on each input field indicates that the field should not be empty, except that the image can be null	pass
TC66	Check the creation of a new lost item report with an invalid lost date and phone number	Image: Select an image from the device Category: إلكترونيات Lost date: 9-9-2024 Expected place: قاعة 23 Phone number: 5432898 Description: سماعة ايربودز بـكفر وردي	Display informative error message	Error message on each input field indicates that: Lost date: restriction on future dates Phone number: Length should be 10	pass
TC67	Check the creation of new found item report with valid inputs	Image: Select valid image Category: مجوهرات Found date: 9-9-2023 Found place: قاعة 9 Receive Place: أمانات كلية الحاسوبات Description: اسورة عليها تعليقة فيل	Back to the lost and found items page and display the new found item report in the list	As expected,	pass
TC68	Check the creation of new found item report with null inputs	Image: Category: Found date: Found place: Receive Place: Description:	Display informative error message	Error message on each input field indicates that the field should not be empty	pass
TC69	Check the creation of a new found item report with an invalid found date	Image: Select image Category: مجوهرات	Display informative error message	Error message on each input field indicates that:	pass

		Found date: 9-9-2024 Found place: قاعة 9 Receive Place: أمانات كلية الحاسوبات Description: اسورة عليها تعليقة فيل		Found date: restriction on a future date.	
--	--	---	--	--	--

Students Activities

TC70	Check the internet connectivity while its off on the Student Activity page	-	A message indicates that the device is not connected to the internet	Display an image that shows there is no internet connection	pass
TC71	Check the internet connectivity while it's off in the creation of a Study Group report with valid inputs	The activity: اهمية الامن السيبراني في حياتنا اليومية The activity date: 2-05-2024 The activity time: 12:20م The activity place: بهو كلية الحاسوبات Number of people: 12	A message indicates that the device is not connected to the internet	As expected,	pass
TC72	Check Student Activity search with valid input	Search input: العقل السليم	List of created Student Activity that contain the search input	As expected,	pass
TC73	Check Student Activity search with null input	Search input:	Display a list of all created Student Activity	As expected,	pass
TC74	Check Student Activity search with does not exist search input	Search input: jgvbuvugvm	A message indicates that the input does not exist	The image shows that there is no created Student Activity	pass
TC75	Check the creation of a Student Activity report with valid inputs	The activity: اهمية الامن السيبراني في حياتنا اليومية The activity date: 30-05-2024 The activity time: 12:20م	Back to the Student Activity page and display the new Student Activity report in the list	As expected,	pass

		The activity place: بيت كلية الحاسوب Number of people: 12			
TC76	Check the creation of a Student Activity report with null inputs	The activity: The activity date: The activity time: The activity place: Number of people:	Display informative error message	Error message on each input field indicates that the field should not be empty	pass
TC77	Check the creation of a Student Activity report with an invalid number of people	The activity: ادارة الاعمال والاداريين The activity date: 06-13-2024 The activity time: ص8:55 The activity place: كلية الاعمال 1040 غرفة رقم 1040 Number of people: اربعين شخص	Display information error message	Error message on the Number of people field that should be only numeric value	pass
TC78	Check the creation of a Student Activity report with small number of people	The activity: نهضتي بثقافي The activity date: 05-26-2024 The activity time: ص9:31 The activity place: مبنى 14 Number of people: 2	Display informative error message	Error message on the Number of people field that should be more than 2 person	pass
TC79	Check the creation of a Student Activity report with large number of people	The activity: فعالية رحلة رياضية The activity date: 06-20-2024 The activity time: ص8:05 The activity place: كلية العلوم - 5 Number of people: 100	Display informative error message	Error message on the Number of people field that should be less than 100 person	pass

TC80	Check the creation of a Student Activity report with an invalid session date	The activity: محاور اللغة العربية The activity date: 11-02-2019 The activity time: 11:39ص The activity place: كلية الاداب Number of people: 20	Display informative error message	Error message on the Activity date field that the date should not be in the past	pass
------	--	---	-----------------------------------	--	------

Study Group

TC81	Check the internet connectivity while it's off in the Study Group page	-	A message indicates that the page isn't connect to the internet	The image shows that there is no connection	pass
TC82	Check the internet connectivity while it's off in the Study Group page search	Search input: مفاهيم اللياقة البدنية والمصحة	Display an image that shows there is no internet connection when the internet is off. once it is on, it returns to the same search input and content on the page	As expected,	pass
TC83	Check Study Group search with valid input	Search input: مواضيع متقدمة في هندسة برمجيات	List of created Study Groups that contain the search input	As expected,	pass
TC84	Check Study Group search with null input	Search input:	Display a list of all created Study Groups	As expected,	pass
TC85	Check Study Group search with does not exist search input	Search input: ffffhgjj	A message indicates that the input does not exist	The image shows that there is no created Study Groups	pass
TC86	Check the creation of a Study Group report with valid inputs	The subject: CCCY 225 - أمن برمجيات The session date: 29-04-2024 The session time: 1:15م The session place: قاعة رقم 24 مبني الحاسوبات	Back to the Study Group page and display the new study group report in the list	As expected,	pass

		Number of people: 4			
TC87	Check the creation of a Study Group report with null inputs	The subject: The session date: The session time: The session place: Number of people:	Display information error message	Error message on each input field indicates that the field should not be empty	pass
TC88	Check the creation of a Study Group report with an invalid number of people	The subject: SCMT 221 - جبر خطى The session date: 24-04-2024 The session time: 10:50ص The session place: بهو مبني العلوم Number of people: شخصان	Display informative error message	Error message on the Number of people field that should be only numeric value	pass
TC89	Check the creation of a Study Group report with small number of people	The subject: لغة انجليزية - ELPR 101 The session date: 21-04-2024 The session time: 8:16ص The session place: مبني كلية الاداب - 1 Number of people: 2	Display informative error message	Error message on the Number of people field that should be more than 2 people	pass
TC90	Check the creation of a Study Group report with large number of people	The subject: اخرى The session date: 04-07-2024 The session time: 2:00م The session place: مبني المعامل Number of people:	Display informative error message	Error message on the Number of people field that should be less than 100 persons	pass

		150			
TC91	Check the creation of a Study Group report with an invalid session date	The subject: BACA 211 - محاسبة مالية The session date: 17-08-2020 The session time: 8:01 ص The session place: مبنيٌ التحضيري 17 Number of people: 6	Display informative error message	Error message on the Session date field that the date should not be in the past	pass

Live Chat

TC92	Check users create a new chat	Select user: شهد الشهري	Navigate to the selected student chat as a receiver	As expected,	pass
TC93	Check user ability send a message to another student	Sent message: السلام عليكم	Display the message in a blue message bubble	As expected,	pass
TC94	Check receives messages from other students	-	Display the sent message in a gray message bubble on the left	As expected,	pass
TC95	Check displays the other user profile information	-	Display the profile information	As expected,	pass
TC96	Check the internet connectivity while its off on the live chat	-	A message indicates that the device is not connected to the internet	Display an image that shows there is no internet connection	pass
TC97	Check the internet connectivity and send a message on live chat	-	A message indicates that the device is not connected to the internet	As Expected,	pass

Save List

TC98	Check saved items (volunteer opportunities, events, study)	-	Display list of the saved items by the user	As expected,	pass
------	--	---	---	--------------	------

	groups, student activity)				
TC99	Check un-save items	-	Icon deactivate and become uncolored	As expected,	pass
TC100	Check navigation of saved item to its category	-	The application navigates to the item category	As expected,	pass

Admin Panel

Login

TC101	Check admin login with an authorized ID (role: admin) and valid password	ID: 0010050 Password: 123456	Navigate to the services page	As expected,	pass
TC102	Check admin login with an unauthorised ID (role: user) and password	ID: 2114050 Password: Aa123456	display an error message	A message indicates that the user is not authorised to login	pass
TC103	Check admin login with null ID and password	ID: Password:	Display informative error message	A message indicates that the ID and password should not be empty	pass
TC104	Check admin login with does not exist ID	ID: 1122334 Password:987654	Display informative error message	A message indicates that the ID or password is not correct	pass
TC105	Check user login with an ID that contains character and password	ID: 123456* Password:987654	Display informative error message	A message indicates that the ID format is not correct	pass

Manage exclusive offers

TC106	Check the creation of new Exclusive Offers with valid inputs	التصنيف: مراكز صحية شعار الشركة: Deryaq.JPG اسم الشركة: جمع رعاية درياق الطبي مقدار الخصم: %30	Navigate to the Exclusive Offers page and display a new Exclusive Offer in the list	As expected,	pass
-------	--	--	---	--------------	------

		صلاحية الخصم: 2024-5-30 الفئة المستهدفة: الموظفين والطلاب وسيلة التواصل: 0530034244 كود الخصم: Deryaq_clinics تفاصيل الخصم : يقدم مجمع درياق %30 طبي خصم على جميع الكشوفات و20% خصم على جميع الخدمات			
TC107	Check the creation of new Exclusive Offers with null inputs	التصنيف: شعار الشركة: اسم الشركة: مقدار الخصم: صلاحية الخصم: الفئة المستهدفة: وسيلة التواصل: كود الخصم: تفاصيل الخصم:	Display informative error message	Error message for each input field indicates that the field should not be empty	pass
TC108	Check the creation of a new Exclusive Offer with an invalid data	التصنيف: مراكز صحية شعار الشركة: Deryaq.JPG اسم الشركة: مجمع رعاية درياق الطبي مقدار الخصم: صلاحية الخصم: 2023-5-30 الفئة المستهدفة: الموظفين والطلاب وسيلة التواصل: 0530034244 كود الخصم: ثلاثون بالمائة تفاصيل الخصم: يقدم مجمع درياق %30 طبي خصم على جميع الكشوفات و20% خصم على جميع الخدمات	Display informative error message	Error message on each input field indicates that: Exclusive Offer date: restriction on a future date, The discount amount must be a number..	pass

TC109	<p>Check the creation of the new Exclusive Offer when entering a discount amount that is less than 1 and more than 100</p>	<p>التصنيف: مراكز صحية شعار الشركة: Deryaq.JPG اسم الشركة: مجمع رعاية درياق الطبي مقدار الخصم: 102% صلاحية الخصم: 2023-5-30 الفئة المستهدفة: الموظفين والطلاب وسيلة التواصل: 0530034244 كود الخصم: Deryaq_clinics تفاصيل الخصم : يقدم مجمع درياق الطبي خصم %30 على جميع الكشوفات و 20% خصم على جميع الخدمات</p>	<p>Display informative error message</p>	<p>Error message indicates that: The discount amount must be between 1 and 100</p>	pass
TC110	<p>Check the availability of editing the existing Exclusive Offer information</p>	<p>التصنيف: مراكز صحية شعار الشركة: Deryaq.pdf اسم الشركة: مجمع رعاية درياق الطبي مقدار الخصم: From 30% to 35% صلاحية الخصم: 2023-5-30 الفئة المستهدفة: الموظفين والطلاب وسيلة التواصل: 0530034244 كود الخصم: Deryaq_clinics تفاصيل الخصم : يقدم مجمع درياق الطبي خصم %30 على جميع الكشوفات و 20% خصم على جميع الخدمات</p>	<p>The discount amount has been successfully changed and the data has been updated in the application</p>	<p>As expected,</p>	pass

TC111	Check the availability of deleting the existing Exclusive Offer	Delete specific Exclusive Offer	The data has been successfully removed and the Exclusive Offer has been removed from the application	As expected,	pass
Manage events					
Courses					
TC112	Check the creation of Courses with valid input	Course title: "اساسيات البرمجة الشيئية" Course Date: 30-05-2024 Course Time: 10:55 Course Location: "كلية الحاسوبات-معلم 117" Course presenter: "أ. سمية بفلح" Course Link: "https://forms.gle/kDftNG8jvHkuTQPp9"	Display the Course information on the Courses table and on the mobile application	As expected,	pass
TC113	Check the creation of Courses with invalid link	Course title: "اساسيات البرمجة الشيئية" Course Date: 29-04-2024 Course Time: 10:55 Course Location: "كلية الحاسوبات-معلم 117" Course presenter: "أ. سمية بفلح" Course Link:"asdfghjkl"	Display an error message for invalid link	As expected,	pass
TC114	Check the creation of Courses with null	Add input: Course title: "" Course Date: Course Time:	Display an error message for every null field	As expected,	pass

		Course Location: Course Presenter:" " Course Link:" "			
TC115	Check the editing the Courses	Course title: "اساسيات البرمجة ال شيئاً" Course Date: 30- 05-2024 Course Time: 10:55 Course Location: "كلية الحاسوبات- معلم 116" Course Presenter: "أ. سمية بفلح" Course Link: "https://forms.gl e/kDftNG8jvHk uTQPp9 "	Save the edit and display the new information on the Courses table and the mobile application	As expected,	pass
TC116	Check removing a Course from the Courses filter	-	The course gets removed from the Courses table and the mobile application	As expected,	pass

Workshops

		Workshop title: "الترخيص والاقتباس في البحث العلمي" Workshop Date: 30-07-2024 Workshop Time: 19:00 Workshop Location: "عن بعد - عبر منصة تيمز" Workshop Presenter: "أ. د. خالد الوكيل" Workshop Link: "https://forms.gl e/kDftNG8jvHk uTQPp9 "			
TC117	Check the creation of Workshops with valid input		Display the Workshop information on the Workshops table and on the mobile application	As expected,	pass

TC118	Check the creation of Workshops with invalid link	Add input: Workshop title: "الوثيق والاقتباس في البحث العلمي" Workshop Date: 30-07-2024 Workshop Time: 19:00 Workshop Location: "عن بعد - عبر منصة تيمز" Workshop Presenter: "أ. د. خالد الوكيل" Workshop Link: "adsadsasd"	Display an error message for invalid link	As expected,	pass
TC119	Check the creation of Workshops with null	Workshop title: Workshop Date: Workshop Time: Workshop Location: Workshop Presenter: Course Link:	Display an error message for every null field	As expected,	pass
TC20	Check the editing the Workshops	Workshop title: "الوثيق والاقتباس في البحث العلمي" Workshop Date: 30-08-2024 Workshop Time: 19:00 Workshop Location: "عن بعد - عبر منصة تيمز" Workshop Presenter: "أ. د. خالد الوكيل" Workshop Link: "https://forms.gle/kDftNG8jvHkuTQPp9"	Save the edit and display the new information on the Workshops table and the mobile application	As expected,	pass

TC121	Check removing a Workshop from the Workshops filter	-	The course gets removed from the Workshop table and the mobile application	As expected,	pass
Conferences					
TC122	Check the creation of Conferences with valid input	Conference title: "جودة الحياة" Conference Date: 29-08-2024 Conference Time: 10:00 Conference Location: "مبني 17" Conference Presenter: "أ. دعاء زهران" Conference Link: "https://forms.gle/kDftNG8jvHkuTQPp9"	Display the Conference information on the Conferences table and on the mobile application	As expected,	pass
TC123	Check the creation of Conferences with invalid link	Conference title: "جودة الحياة" Conference Date: 29-08-2024 Conference Time: 10:00 Conference Location: "مبني 17" Conference Presenter: "أ. دعاء زهران" Conference Link: "asdfghjkl"	Display an error message for invalid link	As expected,	pass
TC124	Check the creation of Conferences with null	Conference title: Conference Date: Conference Time: Conference Location: Conference Presenter: Conference Link:	Display an error message for every null field	As expected,	pass

TC125	Check the editing the Conferences	Conference title: "جودة الحياة" Conference Date: 29-08-2024 Conference Time: 10:00 Conference Location: "مبني 17" Conference Presenter: "أ. دعاء زهران" "د. عاطف خليل" Conference Link: "https://forms.gle/kDftNG8jvHkuTQPp9"	Save the edit and display the new information on the Conferences table and the mobile application	As expected,	pass
TC126	Check removing a Conference from the Conferences filter	-	The course get removed from the Conferences table and the mobile application	As expected,	pass

Other Events

TC127	Check the creation of Other Events without presenter and link	Other Event title: "اليوم العالمي للتوحد" Other Event Date: 05-09-2024 Other Event Time: 11:30 Other Event Location: "بهو مبنى 17"	Display the Other Event information on the Other Events table and on the mobile application	As expected,	pass
TC128	Check the creation of Other Events with valid input	Other Event title: "فعالية اليوم العالمي للعصا البيضاء" Other Event Date: 02-06-2024 Other Event Time: 07:00 Other Event Location: "عن بعد - عبر منصة تيمز" Other Event Presenter:	Display the Other Event information on the Other Events table and on the mobile application	As expected,	pass

		<p>"أ. سندس قوقدني"</p> <p>Other Event Link: "https://forms.gle/kDftNG8jvHkuTQPp9 "</p>			
TC129	Check the creation of Other Events with invalid link	<p>Other Event title: "فعالية اليوم العالمي للعصا البيضاء"</p> <p>Other Event Date: 02-06-2024</p> <p>Other Event Time: 07:00</p> <p>Other Event Location: "عن بعد - عبر منصة تيزير"</p> <p>Other Event Presenter: "أ. سندس قوقدني"</p> <p>Other Event Link: " asdfghjkl "</p>	Display an error message for invalid link	As expected,	pass
TC130	Check the creation of Other Events with null	<p>Other Event title: Other Event Date: Other Event Time: Other Event Location: Other Event Presenter: Other Event Link:</p>	Display an error message for every null field	As expected,	pass
TC131	Check the editing the Other Events	<p>Edit input: Other Event title: "فعالية اليوم العالمي للعصا البيضاء"</p> <p>Other Event Date: 02-06-2024</p> <p>Other Event Time: 11:00</p>	Save the edit and display the new information on the Other Events table and the mobile application	As expected,	pass

		<p>Other Event Location: "عن بعد - عبر منصة تيمز"</p> <p>Other Event Presenter: "أ. سندس قرقندي"</p> <p>Other Event Link: "https://forms.gle/kDftNG8jvHkuTQPp9"</p>			
TC132	Check removing a Workshop from the Workshops filter	-	The course gets removed from the Other Events table and the mobile application	As expected,	pass

Manage volunteering opportunity

		<p>عنوان الفرصة ال التطوعية: زراعة 1139 شتلة التاريخ: 25-5-2024 الوقت: 11:00 الموقع: الساحات الخارجية خلف مني 14 المتطوعين: 9 رابط نموذج التسجيل: https://forms.gle/kDftNG8jvHkuTQPp9</p>	<p>Navigate to the Volunteering Opportunities page and display a new Volunteer opportunity in the list</p>		
TC133	Check the creation of new Volunteering opportunities with valid inputs			As expected,	pass
TC134	Check the creation of new Volunteering opportunities with null inputs	<p>عنوان الفرصة التطوعية: التاريخ: الوقت: الموقع: عدد المتطوعين: رابط نموذج التسجيل:</p>	<p>Display informative error message</p>	Error message for each input field indicates that the field should not be empty	pass

TC135	Check the creation of a new Volunteer Opportunity with an invalid date	عنوان الفرصة: التطوعية: ماراثون التطوع التاريخ: 22-5-2023 الوقت: 9:00 الموقع: امام مبني 5 عدد المتطوعين: 7 رابط نموذج التسجيل: https://forms.gle/kDftNG8jvHkuTQPp9	Display informative error message	Error message on each input field indicates that: Volunteer Opportunity date: restriction on a future date.	pass
TC136	Check the creation of a new Volunteer Opportunity when typing letters in the number of volunteers field	عنوان الفرصة: التطوعية: زراعة 1139 شنطة التاريخ: 25-5-2024 الوقت: 11:00 الموقع: الساحات الخارجية خلف مبني 14 المتطوعين: تسعة رابط نموذج التسجيل: https://forms.gle/kDftNG8jvHkuTQPp9	Display informative error message	Error message indicates that: The number of volunteers must be a number.	pass
TC137	Check the creation of the new Volunteer Opportunity when entering any information that is not a valid link	عنوان الفرصة: التطوعية: زراعة 1139 شنطة التاريخ: 25-5-2024 الوقت: 11:00 الموقع: الساحات الخارجية خلف مبني 14 المتطوعين: 9 رابط نموذج التسجيل: forms.gle/kDftNG8jvHkuTQPp9	Display informative error message	Error message indicates that: The link must be valid.	pass
TC138	Check the availability of	عنوان الفرصة: التطوعية: زراعة 1139 شنطة	The number of volunteers has	As expected,	pass

	editing the existing Volunteer Opportunity information	التاريخ: 25-5-2024 الوقت: 11:00 الموقع: الساحات الخارجية خلف مبني 14 المتطوعين: from 9 to 12 رابط نموذج التسجيل: forms.gle/kDftNG8jvHkuTQPp9	been successfully changed and the date has been updated in the application		
TC139	Check the availability of deleting the existing Volunteer Opportunity	-	The data has been successfully removed and the Volunteer Opportunity has been removed from the application	As expected,	pass

Manage clinic schedule

TC140	Check the creation of a new clinic with valid inputs	الفرع: المقر الرئيسي الشخص: الطب العام اسم الطبيب: د.مهند الحربي التاريخ: 2024, 22 مايو وقت البداية: 10:00 وقت النهاية: 12:00	Navigate to the clinic page and display a new clinic in the list	As expected,	pass
TC141	Check the creation of a new clinic with null inputs	الفرع: الشخص: اسم الطبيب: التاريخ: بداية الوقت: نهاية الوقت:	Display informative error message	Error message for each input field indicates that the field should not be empty	pass
TC142	Check the creation of a new clinic with an invalid date	الفرع: المقر الرئيسي الشخص: الطب العام اسم الطبيب: د.محمد القرني التاريخ : 24 مايو 2023	Display informative error message	Error message on each input field indicates that: clinic date: restriction on a future date.	pass

		وقت البداية: 11:00 وقت النهاية: 12:00			
TC143	Check the creation of a new clinic with an end time before the start time	الفرع: الفيصلية التخصص: طب الأطفال اسم الطبيب: د.محمد البافعي التاريخ: 22 مايو, 2024 وقت البداية: 10:00 وقت النهاية: 9:00	Display informative error message	Error message indicates that: The end time must be after the start time.	pass
TC144	Check the creation of the new clinic with time after 03:00 pm	الفرع: الفيصلية التخصص: الطب النفسي اسم الطبيب: د.احمد النجار التاريخ: 22 مايو, 2024 وقت البداية: 01:00 وقت النهاية: 3:30	Display informative error message	Error message indicates that: End time must not exceed 3:00 PM.	pass
TC145	Check the availability of editing the existing clinic information	الفرع: الفيصلية التخصص: الطب العام اسم الطبيب: د. محمد القرني التاريخ: 2024, 24 مايو to 2024, 25 مايو وقت البداية: 11:00 وقت النهاية: 12:00	The date has been successfully changed and the date has been updated in the application	As expected,	pass
TC146	Check the availability of deleting the existing clinic	Delete specific clinic	The data has been successfully removed and the clinic has been removed from the application	As expected,	pass

Manage student clubs					
TC47	Check the creation of Student Club with valid input	<p>Student Clubs title: "هندسة برمجيات"</p> <p>Student Clubs description: "أسس نادي هندسة البرمجيات على ركيزة جوهرية واحدة"</p> <p>Student Club Registration Date: 22-06-2024</p> <p>Student Club leader: "امامي الحازمي"</p> <p>Contact information: "X : @SESC_UJ"</p> <p>Join as a member Link: "https://linktr.ee/SESC.UJ "</p> <p>Join as a management member Link: "https://linktr.ee/SESC.UJ "</p>	Display the club information on the Student Club table and on the mobile application	As expected,	pass
TC07	Check the creation of Student Clubs join as a member with invalid link	<p>Student Club title: "الذكاء الاصطناعي"</p> <p>Student Club description: "نهدف في هذا النادي ...".</p> <p>Student Club Registration Date: 01-08-2024</p> <p>Student Club leader: "ليان"</p> <p>Contact information: "X : @AiClub_UJ"</p> <p>Join as a member Link:</p>	Display an error message for invalid link.	As expected,	pass

		"asmdlksad " Join as a management member Link: -			
TC08	Check the creation of Student Clubs join as a management member with invalid link	Student Club title: "نادي الشطرنج" Student Club description: "رؤيتنا هي نشر رياضة الشطرنج..." Student Club Registration Date: 30-04- 2024 Student Club leader: "رند الشهري" Contact information: " X : @ChessclubUJ " Join as a member Link: "https://docs.goo gle.com/forms/d/ e/1FAIpQLSe7O IXsL8- QkPENU0tY0k KoJ3xVX_H_b MgsFzcGz3rjN1 MWZA/viewfor m " Join as a management member Link: "lajhdkjsahdksjh d "	Display an error message for invalid link.	As expected,	pass
TC09	Check the creation of Student Clubs with null	Student Club title: Student Club description: Student Club Registration Date: Student Club leader:	Display an error message for empty fields.	As expected,	pass

		Contact information: Join as a member Link: Join as a management member Link:			
TC10	Check the editing in Student Clubs	Add input: Student Clubs title: "هندسة" "برمجيات" Student Clubs description: "أسس نادي هندسة البرمجيات على ركيزة جوهرية واحدة ، مساعها الدؤوب اولاً" Student Club Registration Date: 30-06-2024 Student Club leader: "اماسي الحازمي" Contact information: "X : @SESC_UJ" Join as a member Link: "https://linktr.ee/SESC.UJ " Join as a management member Link: "https://linktr.ee/SESC.UJ "	Save the edit and display the new information in the Student Clubs table and the mobile application	As expected,	pass
TC11	Check removing a Student Club from the Student Clubs	-	The Student Club gets removed from the Student Clubs table and the mobile application	As expected,	pass

7.3 Integration Testing

Integration testing enables the combination of program units for group testing, evaluating how effectively these units operate together. It focuses on identifying interface defects between modules or functions within the system [50].

Table 34 Integration Testing

Test Case No.	Scenario	Status
Mobile Application		
1	If the user wants to Log out: Login Page → Click the login button → Navigate to the services page → Click the menu icon → Click the logout button → Navigate to the Login Page	Success
2	If the user wants to update his profile: Login Page → Click the login button → Navigate to user profile by click on the menu then user profile → Change wanted cell to update → Click update Button	Success
3	If the user wants to get a Discount code: Login Page → Click the login button → Navigate to the services page → Click the offers button in the services category → Selected a desired offers category → Click on the selected offer card within the selected offer category → View the full details of the selected card and get discount code once click on it	Success
4	If the user wants to scroll the page down to see the discount details: Login Page → Click the login button → Navigate to the services page → Click the offers button in the services category → Selected a desired offers category → Click on the selected offer card within the selected offer category → View the full details of the selected card and get discount code once click on it → View the discount details by simply scrolling the screen down	Success
5	If the user wants to return from the offer details page to the cards page for the selected category: Login Page → Click the login button → Navigate to the services page → Click the offers button in the services category → Selected a desired offers category → Click on the selected offer card within the selected offer category → View the full details of the selected card and get discount code once click on it → View the discount details by simply scrolling the screen down → Return to the cards page for the selected category by simply pressing the back button	Success

6	If the user wants to register in the volunteer opportunity: Login Page → Click the login button → Navigate to the services screen → Click the volunteer opportunity category button → Navigate to the volunteer opportunity page → Click the register button → Navigate to the registration form page → Fill the required information then click submit → Successfully registered for the volunteer opportunity	Success
7	If the user wants to save cards in a volunteer opportunity: Login Page → Click the login button → Navigate to the services screen → Click the volunteer opportunity category button → Navigate to the volunteer opportunity page → Click the save icon for desired cards → Successfully saved the cards	Success
8	If the user wants to register as a member to a Student Clubs: Login Page → Click the login button → Navigate to the services screen → Click the Student Clubs category button → Navigate to the Student Clubs page → Click the wanted Student Clubs → Click on the register to join as members button → Navigate to registration form.	Success
9	If the user wants to register as a management member to a Student Clubs: Login Page → Click the login button → Navigate to the services screen → Click the Student Clubs category button → Navigate to the Student Clubs page → Click the wanted Student Clubs → Click on the join to management members button → Navigate to registration form.	Success
10	If the user wants to book an appointment in the clinic: Login Page → Click the login button → Navigate to the services page → Click the clinic category button → Navigate to the clinic page → Click on specific branch → Click the booking button → Navigate to book an appointment page → choose branch and date → fill personal information then click book → The appointment has been booked successfully	Success
11	If the user wants to search in the clinic: Login Page → Click the login button → Navigate to the services page → Click the clinic category button → Navigate to the clinic page → Click on specific branch → Click the search button → Search by specialty or doctor's name → See all search result	Success
12	If the user wants to register to a Course: Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Events page → Navigate to the Courses filter → Click on the register button → Navigate to registration form.	Success
13	If the user wants to search for a specific Course: Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Courses filter → Click the search field → Type the name of the Course in the search field → Click the search icon	Success

14	If the user wants to register to a Workshops: Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Events page → Navigate to the Workshops filter → Click on the register button → Navigate to registration form.	Success
15	If the user wants to search for a specific Workshops: Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Workshops filter → Click the search field → Type the name of the Workshops in the search field → Click the search icon	Success
16	If the user wants to register to a Conferences: Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Events page → Navigate to the Conferences filter → Click on the register button → Navigate to registration form.	Success
17	If the user wants to search for a specific Conferences: Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Conferences filter → Click the search field → Type the name of the Conferences in the search field → Click the search icon	Success
18	If the user wants to register to a Other Events: Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Events page → Navigate to the Other Events filter → Click on the register button → Navigate to registration form.	Success
19	If the user wants to search for a specific Other Events: Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Other Events filter → Click the search field → Type the name of the Other Events in the search field → Click the search icon	Success
20	If the user wants to create a lost item report: Login Page → Click the login button → Navigate to the services page → Click the Lost Items category button → Navigate to the Lost and Found page → Click on the (+) icon → Click the Create Lost Item Report button → Navigate to create the Lost item report page → Click create button → Navigate back to Lost and found page.	Success
21	If the user wants to create a found item report: Login Page → Click the login button → Navigate to the services page → Click the Lost Items category button → Navigate to the Lost and Found page → Click on the (+) icon → Click the Create Found Item Report button. → Navigate to the Create Found Item report page → Click the Create button → Navigate back to the Lost and Found page	Success

22	If the user wants to search for a specific lost item: Login Page → Click the login button → Navigate to the services screen. → Click the Lost Items category button → Navigate to the Lost and found page → Click the Lost filter → Click the search field → Click the search icon	Success
23	If the user wants to search for a specific found item: Login Page → Click the login button → Navigate to the services screen. → Click the Lost Items category button → Navigate to the Lost and Found page → Click the found filter → Click the search field → Click the search icon	Success
24	If the user wants to create a Student Activity report: Login Page → Click the login button → Navigate to the services page → Click the Student Activity category button → Navigate to the Student Activity page → Click on the (+) icon → Navigate to create the Student Activity report page → Click create button → Navigate back to Student Activity page.	Success
25	If the user wants to search for a specific Student Activity: Login Page → Click the login button → Navigate to the services page → Click the Student Activity category button → Navigate to the Student Activity page → Click the search field → Type the name of the activity in the search field → Click the search icon	Success
26	If the user wants to create a Study Group session report: Login Page → Click the login button → Navigate to the services page → Click the Study Group category button → Navigate to the Study Group page → Click on the (+) icon → Navigate to create the Study Group report page → Click create button → Navigate back to Study Group page.	Success
27	If the user wants to search for a specific Study Group session: Login Page → Click the login button → Navigate to the services page → Click the Study Group category button → Navigate to the Study Group page → Click the search field → Type the name of the session in the search field → Click the search icon	Success
28	If the user wants to search for a specific Study Group session: Login Page → Click the login button → Navigate to the services page → Click the Study Group category button → Navigate to the Study Group page → Click the search field → Type the code of the session in the search field → Click the search icon	Success
29	If the user wants to create a new live chat with another user: Login Page → Click the login button → Navigate to the live chat page → Click on the (+) icon → Navigate to all users page → Click on the student name → Type a message on the message field → Click on the send icon → Navigate back to live chat page.	Success
30	If the user wants to view other user profile information: Login Page → Click the login button → Navigate to the live chat page → Click on the user profile picture → Navigate to the information page → Navigate back to the live chat page.	Success

31	If the user wants to save an item to the save list: Login Page → Click the login button → Navigate to the services page → Click on a category button → Click on a category button → Navigate to the category page → Click on the save icon → Navigate back to the services page → Navigate to the save list page	Success
32	If the user wants to un-save an item from the save list: Login Page → Click the login button → Navigate to the save list page Click on the save icon → refresh page → Navigate to the save list page	Success
33	If the user wants to view counselor information: Login Page → Click the login button → Navigate to the services page → Click the Psychological Guidance category button → Navigate to the Psychological Guidance page → Click on the (requesting help) button → Navigate to the counselor information page.	Success
34	If the user wants to chat with counselor information: Login Page → Click the login button → Navigate to the services page → Click the Psychological Guidance category button → Navigate to the Psychological Guidance page → Click on the (requesting help) button → Click on the chat button → Navigate to the counselor information page → Navigate to the live chat.	Success
35	If the user wants to copy the counselor E-mail: Login Page → Click the login button → Navigate to the services page → Click the Psychological Guidance category button → Navigate to the Psychological Guidance page → Click on the (requesting help) button → Long press on the E-mail.	Success

Admin Panel

36	If the admin wants to Log in: Login Page → Click the login button → Navigate to the services page	Success
37	If the user wants to Log out: Login Page → Click the login button → Navigate to the services page → Click the user icon → Click the logout button → Navigate to the Login Page	Success
38	If the admin wants to create a new Exclusive offer: Login Page → Click the login button → Navigate to the services page → Click the Exclusive offers category button → Navigate to the Exclusive offers page → Click the Create Exclusive offers button. → Navigate to the Exclusive Offers page → Fill the required information → Click the save button → Navigate back to the Exclusive Offers page	Success
39	If the admin wants to Edit an existing Exclusive offer: Login Page → Click the login button → Navigate to the services page → Click the Exclusive offers category button → Navigate to the Exclusive offers page → Click the edit Exclusive offers button. → Navigate to the existing Exclusive offers page → Edit the information → Click the save	Success

	button → Navigate back to the Exclusive offers page → See the updated information	
40	If the admin wants to Remove an existing Exclusive offer: Login Page → Click the login button → Navigate to the services page → Click the Exclusive Offers category button → Navigate to the Exclusive Offers page → Click the remove Exclusive Offers button. → The Exclusive offers have been removed from the list → See the updated list after removing the Exclusive offers	Success
41	If the admin wants to create a new volunteer opportunity: Login Page → Click the login button → Navigate to the services page → Click the volunteer opportunity category button → Navigate to the volunteer opportunity page → Click the Create volunteer opportunity button. → Navigate to the volunteer opportunity page → Fill the required information → Click the save button → Navigate back to the volunteer opportunity page	Success
42	If the admin wants to Edit an existing volunteer opportunity: Login Page → Click the login button → Navigate to the services page → Click the volunteer opportunity category button → Navigate to the volunteer opportunity page → Click the edit volunteer opportunity button. → Navigate to the existing volunteer opportunity page → Edit the information → Click the save button → Navigate back to the volunteer opportunity page → See the updated information	Success
43	If the admin wants to Remove an existing volunteer opportunity: Login Page → Click the login button → Navigate to the services page → Click the volunteer opportunity category button → Navigate to the volunteer opportunity page → Click the remove volunteer opportunity button. → The volunteer opportunity has been removed from the list → See the updated list after removing the volunteer opportunity	Success
44	If the admin wants to create a new clinic: Login Page → Click the login button → Navigate to the services page → Click the clinic category button → Navigate to the clinic page → Click the Create clinic button. → Navigate to the clinic page → Fill the required information → Click the save button → Navigate back to the clinic page	Success
45	If the admin wants to Edit an existing clinic: Login Page → Click the login button → Navigate to the services page → Click the clinic category button → Navigate to the clinic page → Click the edit clinic button. → Navigate to the existing clinic page → Edit the information → Click the save button → Navigate back to the clinic page → See the updated information	Success
46	If the admin wants to Remove an existing clinic: Login Page → Click the login button → Navigate to the services page → Click the clinic category button → Navigate to the clinic page → Click the remove clinic button. → The clinic has been removed from the list → See the updated list after removing the clinic	Success

47	<p>If the admin wants to Create a new Event:</p> <p>Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Events page → Choose a specific branch → Click the Create Events button. → Navigate to the Events page → Fill the required information → Click the save button → Navigate back to the Events page</p>	Success
48	<p>If the admin wants to Edit an existing Event:</p> <p>Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Events page → Choose a specific branch → Click the edit Events button. → Navigate to the existing Events page → Edit the information → Click the save button → Navigate back to the Events page → See the updated information</p>	Success
49	<p>If the admin wants to Remove an existing Event:</p> <p>Login Page → Click the login button → Navigate to the services page → Click the Events category button → Navigate to the Events page → Choose a specific branch → Click the remove Events button. → The Events have been removed from the list → See the updated list after removing the Events</p>	Success
50	<p>If the admin wants to Create a new Student Club:</p> <p>Login Page → Click the login button → Navigate to the services page → Click the Student Clubs category button → Navigate to the Student Clubs page → Click the Create Student Clubs button. → Navigate to the Student Clubs page → Fill the required information → Click the save button → Navigate back to the Student Clubs page</p>	Success
51	<p>If the admin wants to Edit an existing Student Club:</p> <p>Login Page → Click the login button → Navigate to the services page → Click the Student Clubs category button → Navigate to the Student Clubs page → Click the edit Student Clubs button. → Navigate to the existing Student Clubs page → Edit the information → Click the save button → Navigate back to the Student Clubs page → See the updated information</p>	Success
52	<p>If the admin wants to Remove an existing Student Club:</p> <p>Login Page → Click the login button → Navigate to the services page → Click the Student Clubs category button → Navigate to the Student Clubs page → Click the remove Student Clubs button. → The Student Clubs have been removed from the list → See the updated list after removing the Student Club</p>	Success

7.4 Acceptance Testing

Acceptance testing, a type of software testing carried out by end-users or customer representatives in a real-world environment, aims to assess if a system is prepared for release. Users evaluate the system in this phase to ensure it aligns with their business requirements. Upon successful completion, the software proceeds to production delivery [50], [52].

Table 35 Acceptance Testing

NO.	Test Case Description	Expected Results	Actual Results	Pass/Fail
Mobile Application				
TC1	Log in to the application with valid credentials.	The user should be successfully logged in and directed to the Services page.	As Expected	Pass
TC2	Log out of the application.	The user should be successfully logged out and directed to the login page.	As Expected	Pass
TC3	Update profile info	The user should update the information successfully.	As Expected	Pass
TC4	View details related to the specific offer	The user should be able to view the full details of the selected card	As Expected	Pass
TC5	Get discount of an exclusive offer	The user should be able to view the discount code once click on get Discount button	As Expected	Pass
TC6	Save the volunteer opportunity cards	The user should be able to save volunteer opportunity cards and see the cards in the saved list	As expected,	Pass
TC7	Register for the volunteer opportunity	The user should be able to register for the volunteer opportunity from the سجل link successfully	As Expected	Pass
TC8	Click on the Student Clubs	The user should be viewing Student Clubs information and be able to register to join as a management member or only a member to the Student Club	As Expected,	Pass
TC9	Register to the Student Clubs as a management member	The user should be able to click the button and go to the registration form	As Expected,	Pass
TC10	Register to the Student Clubs as a member	The user should be able to click the button and go to the registration form	As Expected,	Pass

TC11	Search for clinic	The user should be able to view the search result	As expected,	Pass
TC12	Book an appointment	The user should be able to book an appointment from the الحجز link successfully	As Expected	Pass
TC13	Register to the course	The user should be able to go to the registration from the mobile application	As Expected,	Pass
TC14	Save the course to list	The user should be able to save the course for easy access later	As Expected,	Pass
TC15	Search for a course	The user should be able to view the search result	As Expected,	Pass
TC16	Register to the workshop	The user should be able to go to the registration from the mobile application	As Expected,	Pass
TC17	Save the workshop to list	The user should be able to save the workshop for easy access later	As Expected,	Pass
TC18	Search for a workshop	The user should be able to view the search result	As Expected,	Pass
TC19	Register to the conference	The user should be able to go to the registration from the mobile application	As Expected,	Pass
TC20	Save the conference to list	The user should be able to save the conference for easy access later	As Expected,	Pass
TC21	Search for a conference	The user should be able to view the search result	As Expected,	Pass
TC22	Register to the other events if needed	The user should be able to go to the registration from the mobile application if there was a registration for the event	As Expected,	Pass
TC23	Save the other events to list	The user should be able to save the other event for easy access later	As Expected,	Pass
TC24	Search for other events	The user should be able to view the search result	As Expected,	Pass
TC25	Create a Lost Item Report	The user should be directed to Lost and Found Page. Then, see the new lost item added to the list.	As Expected,	Pass
TC26	Create a Found Item Report	The user should be directed to Lost and Found Page. Then, see the new found item added to the list.	As Expected,	Pass

TC27	Search for a lost item.	The user should be able to view the search result	As Expected,	Pass
TC28	Search for a found item.	The user should be able to view the search result	As Expected,	Pass
TC29	Create a Student Activity Report	The user should be directed to Student Activity Page. Then, see the new Student Activity added to the list.	As Expected	Pass
TC30	Search for a Student Activity session	The user should be able to view the search result	As Expected	Pass
TC31	Create a Study Group Report	The user should be directed to Study Group Page. Then, see the new Study Group session added to the list.	As Expected	Pass
TC32	Search for a Study Group session	The user should be able to view the search result	As Expected	Pass
TC33	Create a new chat.	The user should be able to send messages to the selected user	As Expected,	pass
TC34	Send a message to another student.	The user should be able to see the message on the chat screen	As Expected,	pass
TC35	View other students profile information.	The user should be able to view the other user's profile information	As Expected,	pass
TC36	Un-save items from the save list	The save icon should be inactive and the item should be removed from the list after refreshing the page	As Expected,	pass
TC37	Go so the save item	The user should be able to see all related items in the category	As Expected,	pass
TC38	Communicating with the counselor	The user should be able to navigate to a private live chat with the counselor assigned for them	As Expected,	Pass
TC39	View all counselor information	The user should be able to view all the counselor information (name, office location and email)	As Expected,	Pass
TC40	Copy the counselors E-mail	The user should be able to copy the counselors E-mail	As Expected,	Pass
Admin Panel				
TC41	Log in to the admin panel with valid credentials	The admin should be successfully logged in and directed to the Services page.	As Expected	Pass

TC42	Create a new Exclusive Offer	The admin should be directed to the Exclusive Offer page Then, fill in the required information, and click save to see the added Exclusive Offer	As Expected	Pass
TC43	Edit an Exclusive Offer information	The admin should be directed to the Exclusive Offer Page. Then, see all Exclusive Offer lists and edit the information of the specific Exclusive Offer	As Expected	Pass
TC44	Remove a specific Exclusive Offer	The admin should be directed to the Exclusive Offers Page. Then, see all Exclusive Offer lists and remove specific Exclusive Offer	As Expected	Pass
TC45	Create a new Volunteer Opportunity	The admin should be directed to the Volunteer Opportunity page Then, fill in the required information, click save to see the added Volunteer Opportunity	As Expected	Pass
TC46	Edit a Volunteer Opportunity information	The admin should be directed to the Volunteer Opportunity Page. Then, see all Volunteer Opportunity lists and edit the information of the specific Volunteer Opportunity	As Expected	Pass
TC47	Remove a specific volunteer opportunity	The admin should be directed to the Volunteer Opportunity Page. Then, see all Volunteer Opportunity lists and remove specific Volunteer Opportunity	As Expected	Pass
TC48	Create a new clinic	The admin should be directed to the clinic page Then, fill in the required information and add a new clinic	As Expected	Pass
TC49	Edit a clinic information	The admin should be directed to the clinic Page. Then, see all clinic lists and edit the information of the specific clinic	As Expected	Pass
TC50	Remove a specific clinic	The admin should be directed to the clinic Page. Then, see all clinic lists and remove specific clinic	As Expected	Pass
TC51	Create a new Events	The admin should be directed to the Events page Then, choose a specific branch, fill in the required information, and click save to see the added Events	As Expected	Pass
TC52	Edit an Event information	The admin should be directed to the Events Page. Then, choose a specific branch, see all Events list, and edit the information of the specific Events	As Expected	Pass

TC53	Remove a specific Events	The admin should be directed to the Events Page. Then, choose a specific branch, see all Events list, and remove specific Events	As Expected	Pass
TC54	Create a new Student Clubs	The admin should be directed to the Student Clubs page Then, fill in the required information, and click save to see the added Student Clubs	As Expected	Pass
TC55	Edit a Student Club information	The admin should be directed to the Student Clubs Page. Then, see all Student Clubs list and edit the information of the specific Student Clubs	As Expected	Pass
TC56	Remove a specific Student Clubs	The admin should be directed to the Student Clubs Page. Then, see all Student Clubs list and remove specific Student Clubs	As Expected	Pass

7.5 System Testing

System testing involves testing the entire application as a unified entity to assess compliance with specified requirements and ensure adherence to Quality Standards. It verifies that the application meets customer-set technical, functional, and business requirements [50].

Table 36 System Testing

Test Scenario	Result Status
Mobile Application	
Login page → Login button → Services page → Menu button → Logout button → Login page	Pass
Login page → Login button → Menu button → Profile page → Update button → Back button	Pass
Login page → Login button → Services page → Exclusive offers category button → Exclusive offers category page → Exclusive offers page	Pass
Login page → Login button → Services page → Exclusive offers category button → Exclusive offers category page → Exclusive offers page → Services icon → Services Page	Pass
Login page → Login button → Services page → Volunteer opportunity category button → volunteer opportunity page	Pass
Login page → Login button → Services page → Volunteer opportunity category button → volunteer opportunity page → Select the desired volunteer opportunity → Click register button → Back button → volunteer opportunity page	Pass
Login page → Login button → Services page → volunteer opportunity category button → volunteer opportunity page → Select the desired volunteer opportunity → Click save icon → saved list page	Pass
Login page → Login button → Services page → Student Clubs category button → Student Clubs page	Pass
Login page → Login button → Services page → Student Clubs category button → Student Clubs page → Software engineering Student Clubs button → Register to join as a member button → Form page	Pass
Login page → Login button → Services page → Student Clubs category button → Student Clubs page → Software engineering Student Clubs button → Register to join as a management member button → Form page	Pass
Login page → Login button → Services page → Clinic category button → Clinic page	Pass
Login page → Login button → Services page → Clinic category button → Clinic page → Services icon → Services Page	Pass

Login page → Login button → Services page → Clinic category button → Clinic page → Select the clinic branch button → Click book button → Back button → Clinic page	Pass
Login page → Login button → Services page → Events category button → Events page → Courses filter	Pass
Login page → Login button → Services page → Events category button → Events page → Courses filter → Register button → Form page	Pass
Login page → Login button → Services page → Events category button → Events page → Workshops filter	Pass
Login page → Login button → Services page → Events category button → Events page → Workshops filter → Register button → Form page	Pass
Login page → Login button → Services page → Events category button → Events page → Conferences filter	Pass
Login page → Login button → Services page → Events category button → Events page → Conferences filter → Register button → Form page	Pass
Login page → Login button → Services page → Events category button → Events page → Other Events filter	Pass
Login page → Login button → Services page → Events category button → Events page → Other Events filter → Register button → Form page	Pass
Login page → Login button → Services page → Lost category button → Lost and found page	Pass
Login page → Login button → Services page → Lost category button → Lost and found page → Services icon → Services Page	Pass
Login page → Login button → Services page → Lost category button → Lost and found page → Create Lost item report button → Lost item report Page → back button → Lost and found page	Pass
Login page → Login button → Services page → Lost category button → Lost and found page → Create Lost item report button → Lost item report Page → Create button → Lost and found page	Pass
Login page → Login button → Services page → Lost category button → Lost and found page → Create Found item report button → Found item report Page → Create button → Lost and found page	Pass
Login page → Login button → Services page → Student Activity category button → Student Activity page → Services icon → Services Page	Pass
Login page → Login button → Services page → Student Activity category button → S Student Activity page → Create Student Activity report button → Student Activity report Page → back button → Student Activity page	Pass
Login page → Login button → Services page → Student Activity category button → Student Activity page → Create Student Activity report button → Student Activity report Page → Create button → Student Activity page	Pass
Login page → Login button → Services page → Study Group category button → Study Group page → Services icon → Services Page	Pass
Login page → Login button → Services page → Study Group category button	Pass

→ Study group page → Create Study group report button → Study Group report Page → Back button → Study Group page	
Login page → Login button → Services page → Study Group category button → Study group page → Create Study group report button → Study group report Page → Create button → Study group page	Pass
Login page → Login button → Live chat page → Select student → Send message	Pass
Login page → Login button → Live chat page → New chat button → All students page → Select student → Send message	Pass
Login page → Login button → Live chat page → Select student → profile information page	Pass
Login page → Login button → Save list page	Pass
Login page → Login button → Services page → (event) category button → (courses) page → Save to list button → Save list Page	Pass
Login page → Login button → Services page → Select category button → Chosen category page → Save to list button → Save list Page	Pass
Login page → Login button → Services page → Psychological Guidance category button → Psychological Guidance page	Pass
Login page → Login button → Services page → Psychological Guidance category button → Psychological Guidance page → counselor information button → counselor information page	Pass
Login page → Login button → Services page → Psychological Guidance category button → Psychological Guidance page → counselor information button → counselor information page → live chat button	Pass
Admin Panel	
Login page → Login button → Services page → user button → Logout button → Login page	Pass
Login page → Login button → Services page → Exclusive offers category button → Exclusive offers page → Create Exclusive offers button → Save button → Exclusive offers page	Pass
Login page → Login button → Services page → Exclusive offers category button → Exclusive offers page → Edit button → Edit Exclusive offers information Page → Save button → Exclusive offers page	Pass
Login page → Login button → Services page → Exclusive offers category button → Exclusive offers page → Edit button → Edit Exclusive offers information Page → Back button → Exclusive offers page	Pass
Login page → Login button → Services page → Exclusive offers category button → Exclusive offers page → Delete button → Exclusive offers page	Pass
Login page → Login button → Services page → Volunteer opportunity category button → volunteer opportunity page → Create volunteer opportunity button → Save button → volunteer opportunity page	Pass

Login page → Login button → Services page → Volunteer opportunity category button → volunteer opportunity page → Edit button → Edit volunteer opportunity information Page → Save button → volunteer opportunity page	Pass
Login page → Login button → Services page → Volunteer opportunity category button → volunteer opportunity page → Edit button → Edit volunteer opportunity information Page → Back button → volunteer opportunity page	Pass
Login page → Login button → Services page → Volunteer opportunity category button → volunteer opportunity page → Delete button → volunteer opportunity page	Pass
Login page → Login button → Services page → Clinic category button → Clinic page → Create Clinic button → Save button → Clinic page	Pass
Login page → Login button → Services page → Clinic category button → Clinic page → Edit button → Edit Clinic information Page → Save button → Clinic page	Pass
Login page → Login button → Services page → Clinic category button → Clinic page → Edit button → Edit Clinic information Page → Back button → Clinic page	Pass
Login page → Login button → Services page → Clinic category button → Clinic page → Delete button → Clinic page	Pass
Login page → Login button → Services page → Events category button → Events page → Choose a specific branch → Create Events button → Save button → Events page	Pass
Login page → Login button → Services page → Events category button → Events page → Courses page → Choose Conferences branch → Return to the Courses page → Create Courses button → Save button → Courses page	Pass
Login page → Login button → Services page → Events category button → Events page → Choose a specific branch → Edit button → Edit Events information Page → Save button → Events page	Pass
Login page → Login button → Services page → Events category button → Events page → Choose a specific branch → Edit button → Edit Events information Page → Back button → Events page	Pass
Login page → Login button → Services page → Events category button → Events page → Choose a specific branch → Delete button → Events page	Pass
Login page → Login button → Services page → Student Clubs category button → Student Clubs page → Create Student Clubs button → Save button → Student Clubs page	Pass
Login page → Login button → Services page → Student Clubs category button → Student Clubs page → Edit button → Edit Student Clubs information Page → Save button → Student Clubs page	Pass
Login page → Login button → Services page → Student Clubs category button → Student Clubs page → Edit button → Edit Student Clubs information Page → Back button → Student Clubs page	Pass
Login page → Login button → Services page → Student Clubs category button → Student Clubs page → Delete button → Student Clubs page	Pass

7.6 Usability Testing

Usability testing in software testing is conducted from the end user's viewpoint to assess the system's ease of use. Scenarios were presented to 10 UJ students of diverse ages, colleges, and majors. Tasks were then assigned to explore Mojtamai, which was done by a survey administered through Google Forms containing closed-ended questions. *Table 37* delineates the usability test questions and satisfaction levels.[53].

Table 37 Usability Testing

Measure	Strongly agree	Agree	Neutral	Disagree	Strongly disagree
Learnability: Is Mojtamai easy to use to accomplish the task when you use it for the first time?					
Efficiency: Does Mojtamai help you perform tasks quickly when you learn to use the application?					
Memorability: Is Mojtamai easy to use to accomplish the task when you return after a period of time?					
Satisfaction: In general, will you be satisfied using the Mojtamai application to perform the task?					

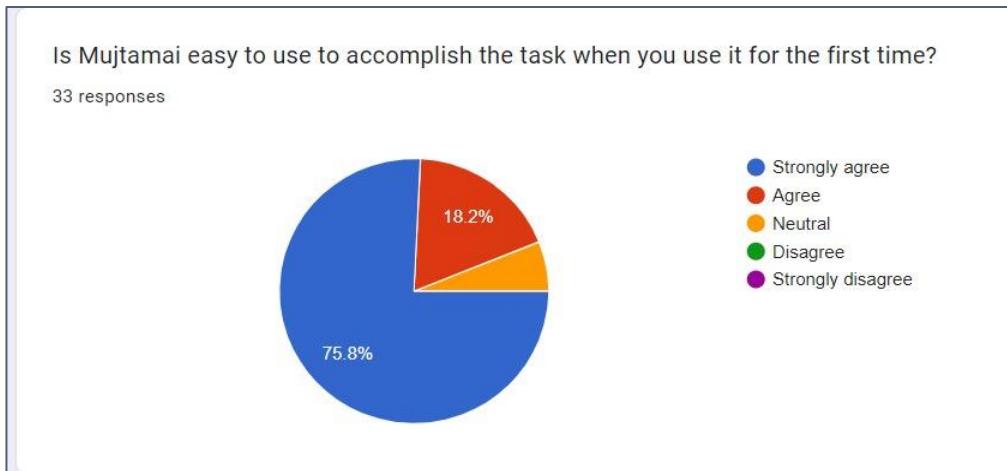


Figure 108 Usability Testing - Learnability

Figure 108 shows that the majority (about 75.8%) were strongly agreed while (about 11.8%) were agree. However, the minority (about 2.9%) were neutral.

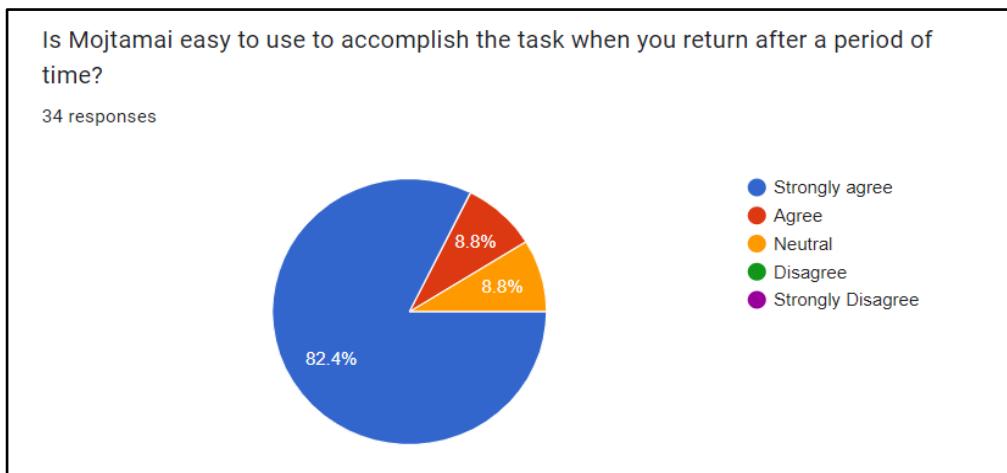


Figure 109 Usability Testing - Memorability

Figure 109 shows that the majority (about 82.4%) were strongly agreed while (about 8.8%) were agree. However, the minority (about 8.8%) were neutral.

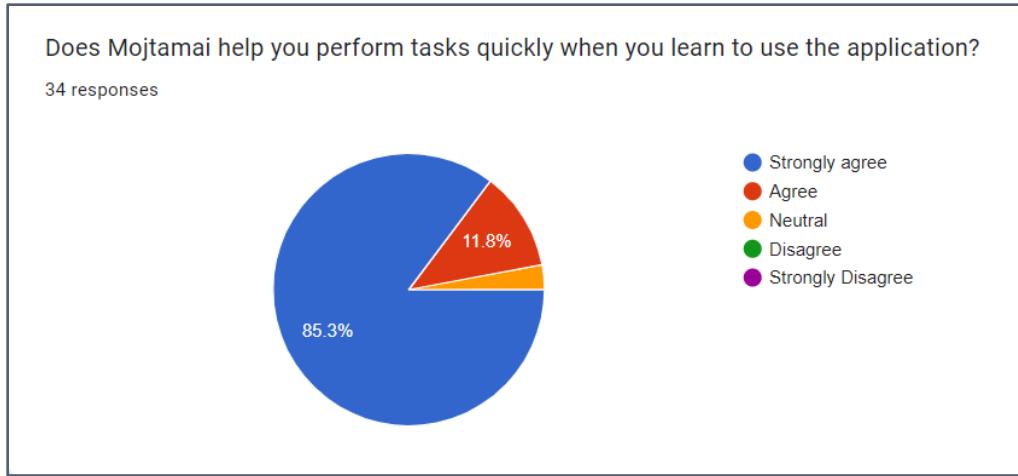


Figure III Usability Testing - Efficiency

Figure 110 shows that the majority (about 85.3%) were strongly agreed while (about 11.8%) were agree. However, the minority (about 1.8%) were neutral.

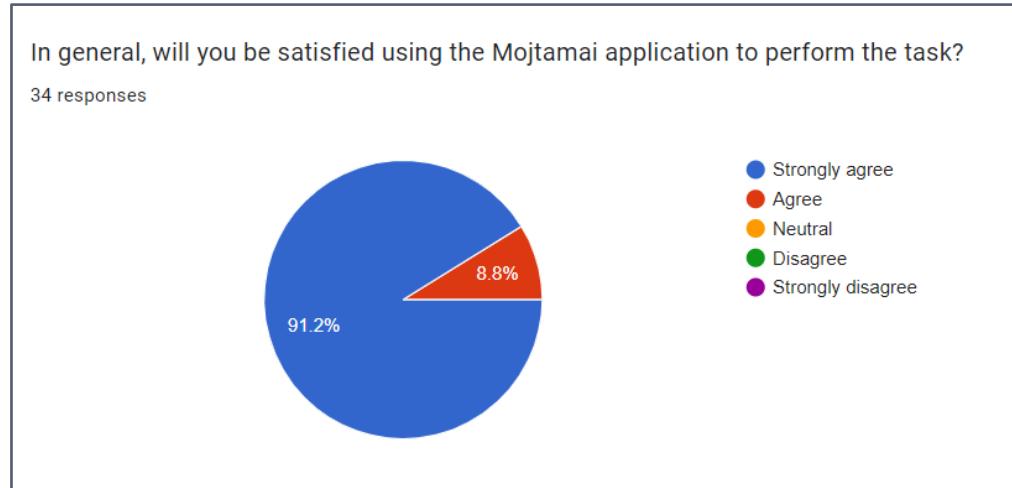


Figure 111 Usability Testing - Satisfaction

Figure 111 shows that the majority (about 91.2%) were strongly agreed while (about 8.8%) were agree.

Upon analysis and evaluation of the users' responses, it was observed that there were no instances of disagreement or strong disagreement. The majority of responses fell under the categories of strongly agree or agree, indicating a high level of satisfaction with the experience and efficiency of Mojtamai.

7.7 Conclusion

This chapter shows the four main stages of testing that need to be completed before a program can be cleared for use: unit testing, integration testing, acceptance testing, system testing, and usability testing. Each result proves that Mojtamai is effective, error-free, and easy for target users.

Chapter 8: Conclusion

8.1 Introduction

This chapter aims to illuminate the limitations encountered during the project development process, including the challenges faced while creating the application. It will also discuss future work that could be carried out to overcome these limitations.

8.2 Limitations

The limitations encountered during the project will be summarized as the following:

- ◆ The application currently only supports the Arabic language.
- ◆ Limited user's data
- ◆ The application was not revealed to people.

8.3 Future Work

Given the time constraints and resource limitations, certain features will be progressively introduced to augment the application's functionality and user experience.

- ◆ The application needs to be connected to the university database.
- ◆ The application has the ability to provide users with certificates they receive from their respective participants.
- ◆ The push notifications notify users of new events or offers upon expansion.
- ◆ The application must be designed to support multiple languages.
- ◆ The application will be uploaded to both Google Play and Apple Store.

8.4 Conclusion

This chapter discussed the primary challenges and obstacles faced during the project's development and highlighted its limitations. In addition, the steps needed to enhance the application's quality, efficiency, and effectiveness among university members have been outlined. The project's primary aim is to assist users in their daily lives.

In conclusion, the application has accomplished the core functionality of this project. Members can now benefit from the app by reducing time and effort. The app can revolutionize how students communicate, cooperate, and manage their educational experience.

References

- [1] V. Trowler, “Student engagement literature review,” 2010.
- [2] “A comparison study between mobile cross platform frameworks Flutter and React Native,” 2023.
- [3] O. M. A. AL-atraqchi, “A Proposed Model for Build a Secure Restful API to Connect between Server Side and Mobile Application Using Laravel Framework with Flutter Toolkits,” *Cihan University-Erbil Scientific Journal*, vol. 6, no. 2, pp. 28–35, Aug. 2022, doi: 10.24086/cuesj.v6n2y2022.pp28-35.
- [4] T. Huynh, “A flashcard mobile application development with Flutter,” 2021.
- [5] R. Lerdorf, K. Tatroe, and P. Macintyre, *Programming Php*. O’ Reilly Media, Inc, 2006.
- [6] H. Allain, P. Mario Di Francesco, and P. Mario Di Francesco Advisor, “’s Programme in ICT Innovation Improving productivity and reducing costs of mobile app development with Flutter and Backend-as-a-Service Title: Improving productivity and reducing costs of mobile app development with Flut-ter and Backend-as-a-Service,” 2020.
- [7] S. John, “CHAT APP WITH REACT JS AND FIREBASE CASE: CHAT APP FOR A COMPANY,” 2010.
- [8] A. Winter, J. Wiseman, and B. Muirhead, “university-community engagement in Australia: practice, policy and public good.,” 2006.

- [9] “MyKAU.” Accessed: Dec. 02, 2023. [Online]. Available: <https://mykau.kau.edu.sa/Pages-about.aspx>
- [10] “KAU Loyalty.” Accessed: Dec. 02, 2023. [Online]. Available: <https://www.kau.edu.sa/Pages-loyalty-and-affiliation.aspx>
- [11] “UQU.” Accessed: Dec. 02, 2023. [Online]. Available: <https://uqu.edu.sa/it/1066#:~:text=%D8%AA%D8%B7%D8%A8%D9%8A%D9%82%D8%AC%D8%A7%D9%85%D8%B9%D8%A9%20%D8%A3%D9%85%D20%D8%A7%D9%84%D9%82%D8%B1%D9%89%3A%20%D9%87%D9%88%D20%D8%A8%D8%B1%D9%86%D8%A7%D9%85%D8%AC%D20%D9%8A%D8%B3%D8%AA%D8%B9%D8%B1%D8%B6%D20%D8%A3%D9%87%D9%85,%D8%B9%D9%84%D9%89%D20%D8%A2%D8%AE%D8%B1%D8%A7%D9%81%D9%8A%D20%D8%AD%D8%A7%D9%84%D20%D8%AA%D9%81%D8%B9%D9%8A%D9%84%D20%D8%AE%D8%AF%D9%85%D8%A9%D20%D8%A7%D9%84%D8%A5%D8%B4%D8%B9%D8%A7%D8%B1%D8%A7%D8%AA>
- [12] “King Saud University -Nafie,” King Saud University. Accessed: Sep. 20, 2023. [Online]. Available: <https://nafae.ksu.edu.sa/>
- [13] “King Faisal University -KFU,” KFU -Development by Deanship of Information Technology. Accessed: Sep. 20, 2023. [Online]. Available: <https://services.kfu.edu.sa/Services/Account/Login?ReturnUrl=%2fservices>
- [14] Taif University, “Mazaya platform,” Deanship of E-Learning and Information Technology - Web Applications Department. Accessed: Sep. 20, 2023. [Online]. Available: [جامعة الطائف - منصة مزايا \(tu.edu.sa\)](http://tu.edu.sa)

- [15] Taif University, “Maharat ,” Deanship of E-Learning and Information Technology - Web Applications Department. Accessed: Sep. 20, 2023. [Online]. Available: [Maharat | مهارات \(tu.edu.sa\)](https://tu.edu.sa/Maharat)
- [16] KKUx, “KKUx,” KKUx . Accessed: Sep. 20, 2023. [Online]. Available: <https://kkux.org/about>
- [17] “AASTMT Student Portal.” Accessed: Dec. 02, 2023. [Online]. Available: https://aast.edu/en/news.php?page=0&unit_id=1&language=1&event=1348&get_event_type=1
- [18] “The University of Manchester,” The University of Manchester. Accessed: Sep. 20, 2023. [Online]. Available: <https://www.manchester.ac.uk/collaborate/communities/#:~:text=Our%20community%20promise%20to%20improve,people%20in%20our%20neighbouring%20communities>.
- [19] “ucl - london’s global university.” Accessed: Sep. 20, 2023. [Online]. Available: <https://www.ucl.ac.uk/>
- [20] “university of kent.” Accessed: Dec. 02, 2023. [Online]. Available: <https://www.kent.ac.uk/student/app>
- [21] “Axenda Hubs.” Accessed: Dec. 02, 2023. [Online]. Available: <https://hubs.axenda.io/>
- [22] “CampusGroups.” Accessed: Dec. 02, 2023. [Online]. Available: <https://blog.campusgroups.com/campusgroups/2016/1/29/your-own-dedicated-campus-app>

- [23] “Piogroup ,” Piogroup Software, Corp. Accessed: Sep. 20, 2023. [Online]. Available: <https://piogroup.net/project/post-ed-student-app>
- [24] R. Malan, H.-P. Company, and D. Bredemeyer, “Functional Requirements and Use Cases Functional Requirements.”
- [25] A. Hudaib, R. Masadeh, M. H. Qasem, and A. Alzaqebah, “Requirements Prioritization Techniques Comparison,” *Mod Appl Sci*, vol. 12, no. 2, p. 62, Jan. 2018, doi: 10.5539/mas.v12n2p62.
- [26] L. Chung, J. C. Sampaio, and P. Leite, “On Non-Functional Requirements in Software Engineering,” 2009. [Online]. Available: www.utdallas.edu/~chung/, www.inf.puc-rio.br/~julio
- [27] A. Jarzebowicz and P. Weichbroth, “A Qualitative Study on Non-Functional Requirements in Agile Software Development,” *IEEE Access*, vol. 9, pp. 40458–40475, 2021, doi: 10.1109/ACCESS.2021.3064424.
- [28] “User Interface (UI) Design.” Accessed: Nov. 11, 2023. [Online]. Available: <https://www.interaction-design.org/literature/topics/ui-design>
- [29] “Nonfunctional Requirements ,” © Scaled Agile, Inc. . Accessed: Nov. 11, 2023. [Online]. Available: <https://scaledagileframework.com/nonfunctional-requirements/>
- [30] Sarah Lewis, “interoperability.” Accessed: Nov. 11, 2023. [Online]. Available: <https://www.techtarget.com/searchapparchitecture/definition/interoperability>
- [31] C. C. Hamid Akhtar, “NFRs: What is Non Functional Requirements (Example & Types).” Accessed: Nov. 11, 2023. [Online]. Available: <https://www.browserstack.com/guide/non-functional-requirements-examples>

- [32] J. Smith and A. Johnson, “The Benefits of Prototyping in Product Development,” *Journal of Engineering Design*, vol. 10, pp. 123–136, 2022.
- [33] R. Fauzan, D. Siahaan, S. Rochimah, and E. Triandini, “A Different Approach on Automated Use Case Diagram Semantic Assessment,” *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 1, pp. 496–505, 2021, doi: 10.22266/IJIES2021.0228.46.
- [34] S. Tiwari and A. Gupta, “Use case specifications: How complete are they?,” *Journal of Software: Evolution and Process*, vol. 32, no. 1, Jan. 2020, doi: 10.1002/smrv.2218.
- [35] Dennis M. Buede and William D. Mille, *The Engineering Design of Systems: Models and Methods*. 2016. Accessed: Dec. 02, 2023. [Online]. Available: <https://books.google.com.sa/books?id=RxeRCgAAQBAJ>
- [36] A. Ahmed, S. Ahmad, N. Ehsan, E. Mirza, and S. Z. Sarwar, “Agile software development: Impact on productivity and quality,” 2010 IEEE International Conference on Management of Innovation & Technology.
- [37] “scrum.” Accessed: Nov. 11, 2023. [Online]. Available: <https://www.agilealliance.org/glossary/scrum/>
- [38] Mary Iqbal, “Scrum Master’s Toolkit: Essential Tools and Software for Agile Success.” Accessed: Nov. 11, 2023. [Online]. Available: <https://www.scrum.org/resources/blog/scrum-masters-toolkit-essential-tools-and-software-agile-success>

- [39] Jeff Dalton, “Burn Down Chart,” Apress, Berkeley, CA. Accessed: Nov. 11, 2023. [Online]. Available: https://link.springer.com/chapter/10.1007/978-1-4842-4206-3_19
- [40] T. Ahmad, J. Iqbal, A. Ashraf, D. Truscan, and I. Porres, “Model-based testing using UML activity diagrams: A systematic mapping study,” *Comput Sci Rev*, vol. 33, pp. 98–112, Aug. 2019, doi: 10.1016/J.COSREV.2019.07.001.
- [41] R. Shmallo and T. Shrot, “Constructive Use of Errors in Teaching the UML Class Diagram in an IS Engineering Course,” 2020, [Online]. Available: <http://jise.org/Volume31/n4/JISEv31n4p282.html>
- [42] D. Bell, “<http://www.ibm.com/developerworks/rational/library/3101.html>.” [Online]. Available: <http://www.ibm.com/developerworks/rational/library/3101.html>
- [43] “MVC - MDN Web Docs Glossary: Definitions of Web-related terms | MDN.” Accessed: May 05, 2024. [Online]. Available: <https://developer.mozilla.org/en-US/docs/Glossary/MVC>
- [44] “ISO/IEC 25010:2023.” Accessed: May 05, 2024. [Online]. Available: <https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:en>
- [45] R. Dami Ristanto, A. Dwinanto, dan Nawasyarif, and A. korespondensi, “Edu Komputika Journal Analisis Software Product Quality ISO/IEC 25010 pada Pengembangan Tes Bakat Menggunakan Sistem Computer-Based Test (CBT),” 2020, [Online]. Available: <http://journal.unnes.ac.id/sju/index.php/edukom>
- [46] “FAQ | Flutter.” Accessed: Nov. 30, 2023. [Online]. Available: <https://docs.flutter.dev/resources/faq>

- [47] “Dart Programming Language: A Guide to its Applications.” Accessed: Nov. 30, 2023. [Online]. Available: <https://emeritus.org/blog/coding-dart-programming-language/>
- [48] “Firebase | Flutter.” Accessed: Dec. 02, 2023. [Online]. Available: <https://docs.flutter.dev/data-and-backend/firebase>
- [49] “6 Best Flutter IDE & Text Editors for App Development | Blog - BairesDev.” Accessed: Nov. 30, 2023. [Online]. Available: <https://www.bairesdev.com/blog/best-flutter-ide-text-editors/>
- [50] “Critical to the Quality of Software Products.” [Online]. Available: www.seguetech.com
- [51] Moataz Nabil, “What is unit testing in mobile development?” Accessed: Dec. 02, 2023. [Online]. Available: <https://bitrise.io/blog/post/what-is-unit-testing-in-mobile-development>
- [52] STEN PITTEL, “The different types of software testing.” Accessed: Dec. 02, 2023. [Online]. Available: <https://www.atlassian.com/continuous-delivery/software-testing/types-of-software-testing>
- [53] “Usability Testing.” Accessed: Dec. 02, 2023. [Online]. Available: <https://www.geeksforgeeks.org/usability-testing/>