# AQU LIBRARY

- Project Title: AQU Library
- The Version: 1.0
- Date: 8/11/2022

# AQU LIBRARY

## Table of Content

# I. Introduction

## 1. Executive Summary:

The most significant development in our culture now as people travel through this digitized world is computer technology. Because they make life easier and more convenient, these innovations have a significant impact on how men live. In addition to understanding how to run a computerized library system, librarians must be aware that these improvements will greatly benefit them.

It greatly aids in achieving high-quality education through consignment and efficient learning by offering sustainable technologies. Our culture is now requiring more computer technology. It is incredibly important to our day-to-day existence. It keeps developing.

The system is an "AQU Library" a branded android mobile application that allows book borrowing as an electronic service via a mobile device and streamlines the process for students.

Systems used in libraries are often worked manually. The librarian fills out forms that are provided to them with pens. The manual library system is a laborious and time-consuming process. The tedious effort of finding records is some librarians' main gripe with this manual system. The rapid advancement of technology has had a significant impact on the development of the library system.

## 2. Audience

- Al-Quds University students.
- Al-Quds University librarians.

# II. Research and Findings

## 3. Related Topics

**I found a related paper, which I will attach in references, and the finding is, the system has three users in the system:**

- Administrator: is a user who manages the system's transactions by adding a new librarian, managing the librarians' transactions, seeing all transactions from other users, and has all permissions on the system (Usually the director of the library).

- Librarians: are users who have adding access (such as adding books, adding journals, ...etc.), manage borrowing transactions by approval or refusal of borrow requests, send the request to administrators that has requirements to buy it, and receive suggestions from the borrowers.

- Borrowers: are users who send requests to borrow the book or journal and search for the book, and can reserve the table, room, and lab in the library, in case the book was not found they can suggest providing it to the library.

**The main goals of them System were to:**

1. Computerize the manual system of keeping the records of the book.
2. Less time while searching for the book.
3. Book return of the student.
4. Each student registered in the system is provided a registered library ID.

## Scope and Limitation of them Study:

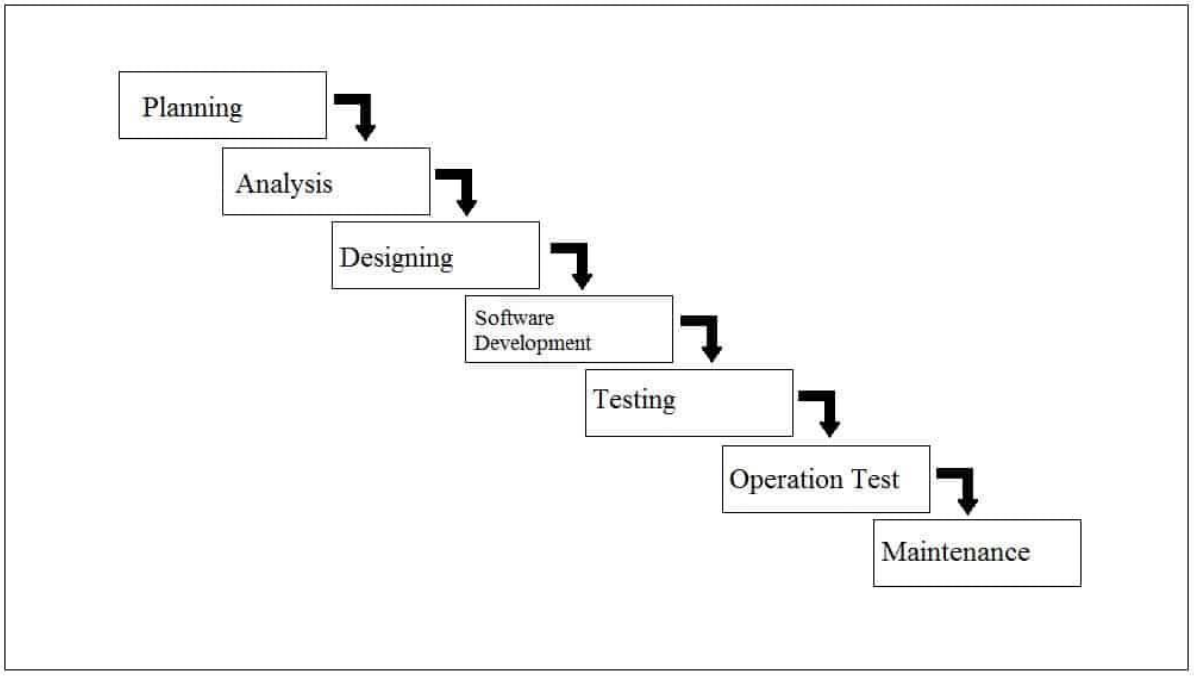
This research was restricted to St. Columban's Academy's library, which serves more than 500 pupils and has significant yearly growth.

In advance, this system improved in dependability and reliably supplied the school's library with adequate assistance to satisfy an annual increase in enrollment in pupils.

The proposed system was created based on the school's desires and requirements. It includes the registration of books with a unique book identification number so that the librarian can easily access each book, search tabs for dependable, simple, and quick monitoring of books, and—most importantly—the ability to generate reports and evaluate papers in a well-organized and well-presented manner. Every transaction may only involve the borrowing of one book per student.

## Methodology:

In them project they used Waterfall Model Project Design Model



- **Planning:** To acquire information on how the librarian manages the books, borrowed book records, returned book records, and all student records in the library, the researchers interviewed the client throughout the system development phase. As a result of the observation that the librarian manually records all transactions and maintenance information for each student who borrows or returns a book on a piece of paper, the researchers intend to improve recordkeeping by computerizing it to make it easier for the librarian and library users going forward. For the safety of their documents, the researchers also want to offer more secure and sizable database storage.

- **Analysis: The suggested computerized library borrowing system was created entirely from the users' perspective, without technological limitations (such as computer hardware and software).**

- **Design:** The researchers employed a user-friendly design to entice users to use the new library borrowing system. The program was specifically created for the school's librarian. Here, the system was designed and built only with the users in mind, putting hardware limitations aside (such as computers). Included were clarifications on the system architecture needed to implement such ideas. Following a review of the planning's need specifications, a system synopsis is given using diagrams to make it simple to guess how data processing would go. Charting of the inputs and outputs was done based on similarity, split into subsystems, and plotting.

  On a practical basis, the entire project was divided into several modules, and then each module was further subdivided into smaller entities. The boldness of the coding system was considered when developing the code. Relationships between the data were also examined.

**Reference: (**Library-Borrowing-System-Documentation.pdf )

## 4. User Analysis (Persona)

Ahmad is a student, his age 19 and he want to:

- Reading books and journals.
- Search about new book for his favourite author.

He wants to borrow a book, but he has many problems in the manual borrowing, such as went to the university to borrow and sometimes he didn't find transport, he forgets when he should return the books and journals.

# III. Feasibility Study and Proposed Solution

Here you describe the problem, they available solutions, and what the best solution for this problem is.

## 5. Alternative Solutions

When I searched about an alternative solution, I found a web version of AQU Library services, but it has some problem like:

- Design problems: for example, the text didn't regular with its parent as the pictures below, these are a picture of buttons in student portal Book Catalog (Al Quds University .::. AQU .::. Portal).


- In general state the website is relatively difficult to use with mobile, in standard mode the part in the page was overlapping, and in 'show as desktop site' the part become relatively small.

- In the exists website system you can search about the book and get details about the book and if it exists or not, but the main problem 'Online reservation' didn't exist in the system and can't reserve the book remotely (https://library.alquds.edu/ar).

## 6. Proposed Solution

1. The system will make the book borrow operation an electronic service and reserve the books, labs, and tables remotely.

2. The system has a login page, the students will log in to their university accounts and confirm if the login data was right or not by searching in the database and confirm it.

3. The system will notify the borrower before the end of the borrowing period (for example you must return the book after 1 week, the system will calculate the date (based on server date) for return the book and will let you know before 2 days).

4. After the period ends the system will calculate how much fine by day and how much amount must pay (by mathematical operation, for example you late 4 days after return date, the system will calculate the day by getting the day from the server date, and you must pay 5NIS per day for late, so the system will find the result (20NIS) and print it in the app) and notify the user how he must pay.

5. When the user tries to reserve the book, the application will find the fitting periods (which the book was not borrowed in this) throw searching by tag, if the book borrowed the system will make its tag as unavailable and if it didn’t borrow the system will make the tag available.

6. The system will sort the books as a category by type or authors.

7. The system has a search by book name, by its ISBN, or author name.

8. The system will view all information about the book.

# IV. Technical Approach

## 7. Tools

- (https://online.visual-paradigm.com/) and (diagrams.net) for building the software diagrams.
- (Proto.io - Prototyping for all) to create a prototype for the system.
- Jira tool to manage tasks and time.
- Adobe XD to design the UIs for the system.
- XML to create the UIs in the android studio.
- Java for android to develop the backend for the application in android studio.
- JUnit test to test the methods, check, and making find the bugs in the code easy.
- PHP CodeIgniter to develop a backend for the system.

## 8. Environment

In the current system, the environment will completely be compatible with any android device, so if someone have an android device, he will be able to take the full features, hopefully in the feature it will be compatible with the iPhone devices in the future, and we will try to make sure that every device will have the full features without any different, and the full functionality will be on every device too.

# V. Final SRS

## 9. Requirements

### 1.1. Functional Requirements

1. Login: The student must login in his/her university account can make any action on the system.
2. Select Search: The student can search about the book in many methods such as:
    - By Name.
    - By Category.

   - By ISBN.
   - By Author name.
   - By Classification number.
3. Show Book details: The system must show the status of book if it borrowed or available in the library, and its details, such as (title, author/s, published date, version, page number).
4. Reserve the book: The student can reserve the book when its status is available.
5. Cancel reservation: When the student reserve a book and isn't come to take it from library, the system will cancel his request.
6. Send notifications: The system will send the notification in two cases:
   - When the borrowed period comes to end.
   - When the borrowed period ends and how much fine must pay.
7. Extends borrowed request: Sometimes the student need more time to use the book so he/she can request to extend the period.
8. View borrowed history: Display the book which the student borrowed and view it status if he returns it, late for return, or still in borrowed period.
9. View fine history: Display how much fines a student gets and its status if paid or not.
10. Suggest book: the student can suggest a book if it does not exist in the library branch.
11. View reservation history: view all reserved book from the student.
12. View late submitted book history.

### 1.2. Non-Functional Requirements

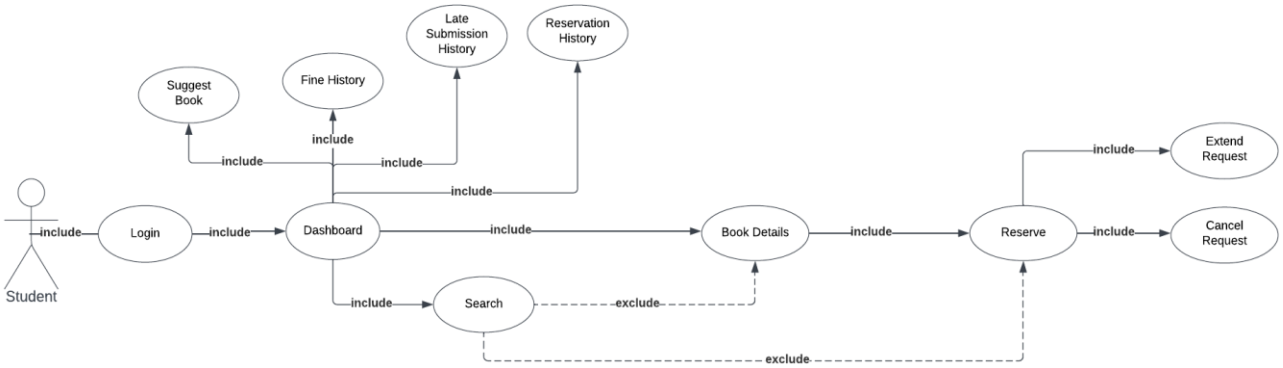
a. Availability: System will be always available, so the student can send requests at any time.

b. Security: All of users must authenticate to borrowed book, so it will not accept any unauthorized borrowed, and the borrowed operation must be verified by email.

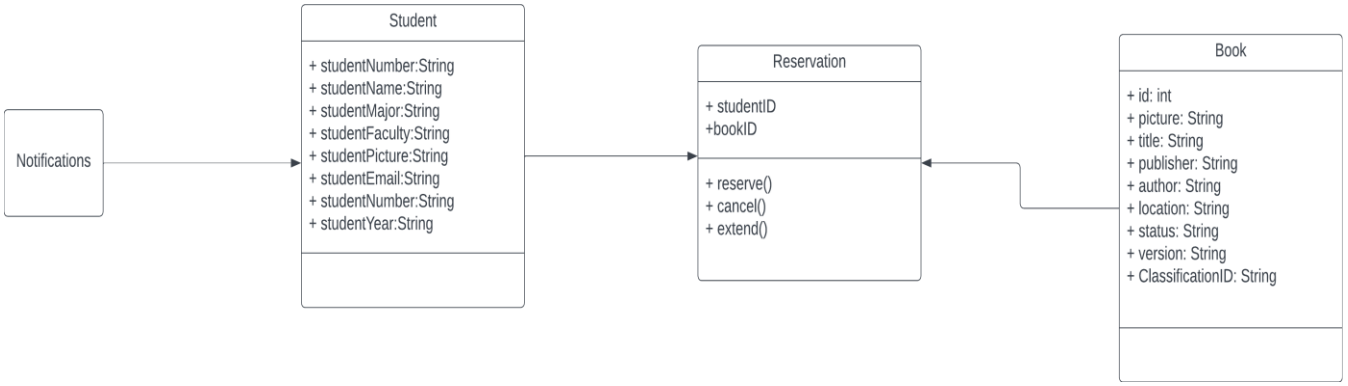
c. Usability: The system will have simple interfaces that will interact with students.

d. Reliability: The student reserve a book online (without be inside the university) so the student can rely on the system.

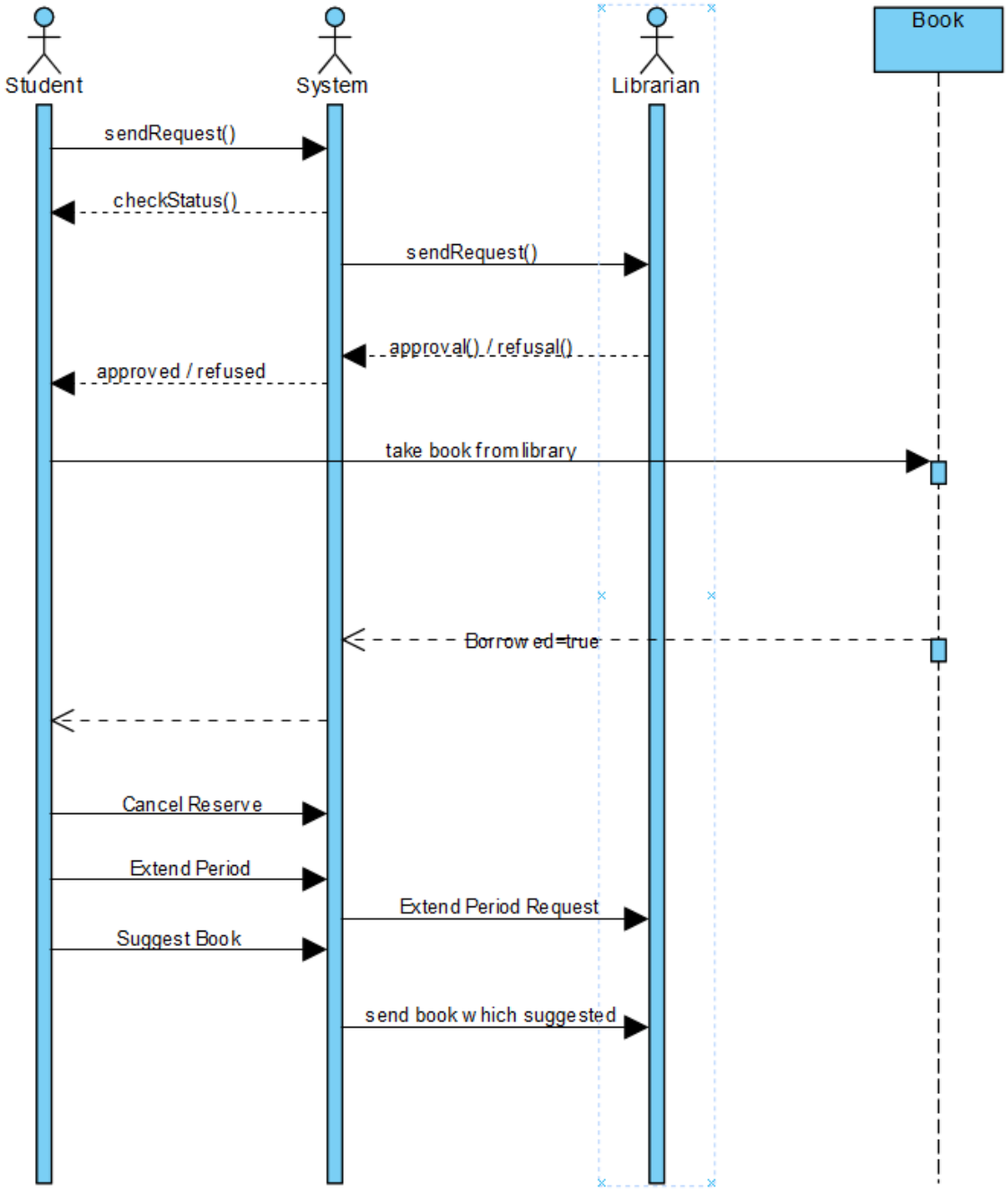
## 10. Use Case Diagram



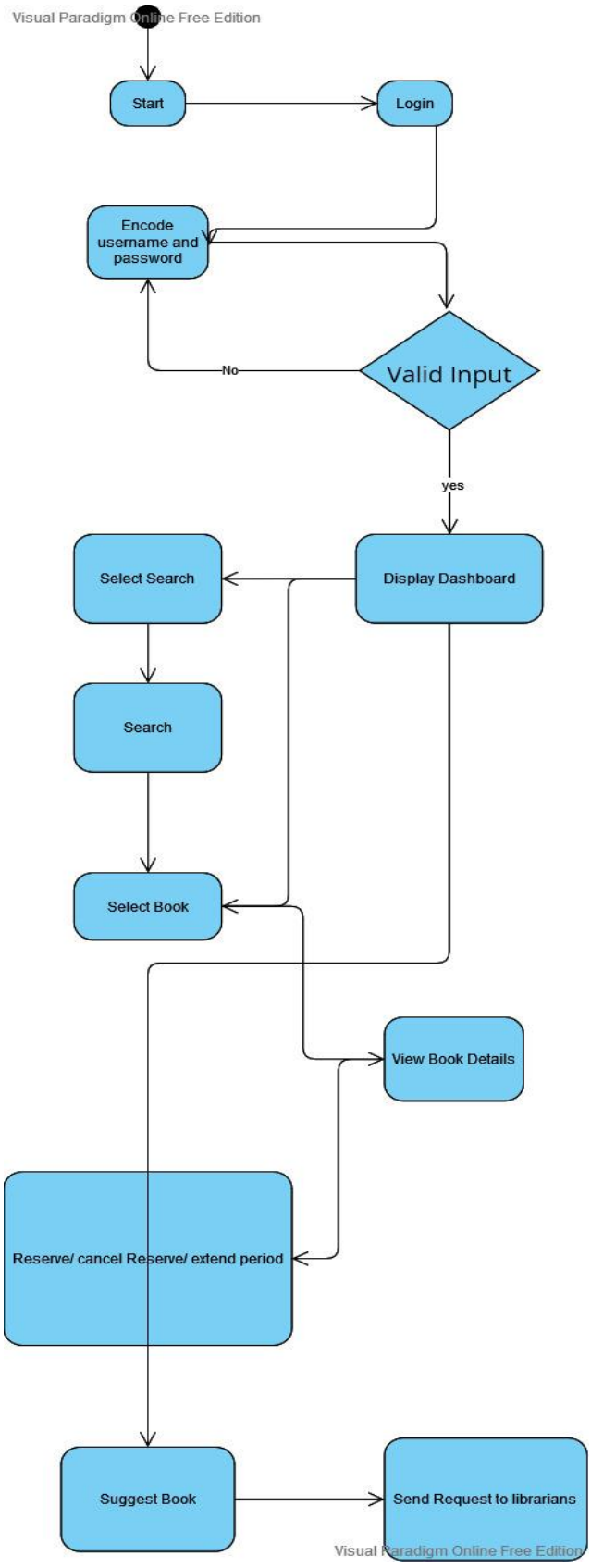

## 11. Class Diagram



## 12. Sequence Diagram



## 13. Activity Diagram



# VI. Testing Process (2nd Semester)

Test cases is here

| Sign In | | | | |
|---|---|---|---|---|
| Field Name | Type | Description of Test Case | Example Input | Example Output |
| Username | NC | Enter username with right format | s1910885 | Nothing shows |
| | EC | Enter username without 's' in the first | 21910885 | Username error format |
| | EC | Enter username without numbers | ahmad | Username error format |
| | EC | Enter the part of username | s1910 | Username error format |
| | EC | Enter student user not in Al-Quds university | s5522997 | User not found |
| Password | NC | Enter password that meet the portal password | Tamer0110** | Nothing shows |
| | EC | Enter password does not match with the username | s1910885/AW1111111*7 | Password is not correct |
| | EC | Enter an escape char to make SQL Injection or XSS Attack | "+union select username from table +" | Illegal input |
| Sign In Button | NC | Send request with all user credentials was right | s1910885/********** | Nothing shows |
| | EC | Send request with all fields was empty | | Required |
| | EC | Send request in offline status | | Error while connection |

| Dashboard | | | | |
|---|---|---|---|---|
| Field Name | Type | Description of Test Case | Example Input | Example Output |
| Search Icon | NC | Search Icon is existed | The user should see the icon | The user finds the icon |
| | EC | Search Icon isn't existed | The user should see the icon | Icon not found |
| Sign Out Icon | NC | Sign Out Icon is existed | The user should see the icon | The user finds the icon |
| | EC | Sign Out Icon isn't existed | The user should see the icon | Icon not found |

| Suggest Book | | | | |
|---|---|---|---|---|
| Field Name | Type | Description of Test Case | Example Input | Example Output |
| Suggest Service | NC | Submit full form | The user clicks on the service icon | Successful Suggestion |
| | EC | Fill part of form | The user clicks on the service icon | form not complete |

| Book Details | | | | |
|---|---|---|---|---|
| Field Name | Type | Description of Test Case | Example Input | Example Output |
| | NC | All details are found | The user should show all details | Book Detail |
| | EC | Some details aren't found | The user search about the author for the book | Auther not found |

| Fine History | | | | |
|---|---|---|---|---|
| Field Name | Type | Description of Test Case | Example Input | Example Output |
| Search | NC | Enter book is exist in fine history | Ethical Hacking | Found |
| | EC | Enter book isn't exist in fine history | خوف | Doesn't Found |
| | EC | Enter an escape char to make SQL Injection or XSS Attack | "+union select username from table +" | Illegal input |
| Paid Status Icon | NC | Paid Status Icon is existed | Paid fine user should see green icon | The user finds the icon |
| | NC | Paid Status Icon is existed | Unpaid user should see red icon | The user finds the icon |
| | EC | Paid Status Icon is existed | Paid users see red icon vice versa | Wrong status |
| | EC | Paid Status Icon isn't existed | The user should see the icon | Icon not found |

| Search | | | | |
|---|---|---|---|---|
| Field Name | Type | Description of Test Case | Example Input | Example Output |
| Search Field | NC | Enter search method and enter the text in the field | Ethical Hacking | Showing search result |
| | EC | Enter book is not exist in library | خوف | Showing nothing |
| | EC | Enter an escape char to make SQL Injection or XSS Attack | "+union select username from table +" | Illegal input |
| Select Search | NC | View all search methods | The user should all search methods | Nothing shows |
| | EC | View part of search method | The user should all search methods | Method not found |
| More Details Icon | NC | More Details Icon is existed | The user should see the icon | The user finds the icon |
| | EC | More Details Icon isn't existed | The user should see the icon | Icon not found |
| Reserve Icon | NC | Reserve Icon is existed | The user should see the icon | The user finds the icon |
| | EC | Reserve Icon isn't existed | The user should see the icon | Icon not found |

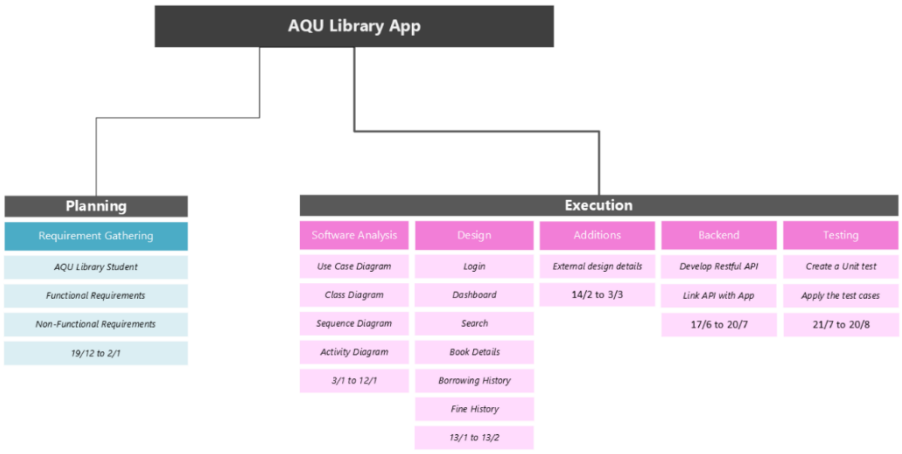
| Borrowed History | | | | |
|---|---|---|---|---|
| Field Name | Type | Description of Test Case | Example Input | Example Output |
| Search | NC | Enter book is exist in borrowed history | Ethical Hacking | Found |
| | EC | Enter book isn't exist in borrowed history | خوف | Doesn't Found |
| | EC | Enter an escape char to make SQL Injection or XSS Attack | "+union select username from table +" | Illegal input |
| Borrowed Status Icon | NC | Borrowed Status Icon is existed | Late submission user should see red icon | The user finds the icon |
| | NC | Borrowed Status Icon is existed | Normal submission user should see green icon | The user finds the icon |
| | EC | Borrowed Status Icon is existed | Normal submission user sees red icon vice versa | Wrong status |
| | EC | Borrowed Status Icon isn't existed | The user should see the icon | Icon not found |

| Reservation | | | | |
|---|---|---|---|---|
| Field Name | Type | Description of Test Case | Example Input | Example Output |
| Reserve Button | NC | The user sends reservation request for the book and admin approve the request | The user must get notification in email | The user goes to take the book |
| | NC | The user sends reservation request for the book and admin decline the request | The user must get notification in email | No action |
| | EC | The user sends reservation request for the book and admin approve the request | The user must get notification in email | The user doesn't go to take the book, so the system cancels his request over than 24 h |
| Cancel Button | NC | The user doesn't take the book and he try to cancel | Cancel request | Verification message to apply the cancelation process |
| | EC | The user take the book and he try to cancel | Cancel request | Cannot cancel after taking the book |
| Extend Period Button | NC | The button must appear after user take the book | the user wants to extend period after taking the book | Extend Button is appeared |
| | EC | The button must appear after user take the book | the user wants to extend period before taking the book | Extend Button isn't appeared |

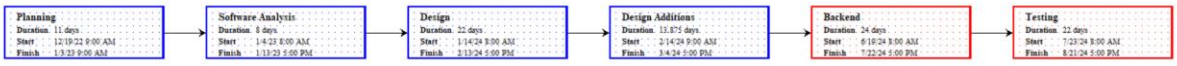
# VII. Project Management (2nd Semester)

## 14. Work Breakdown Structure

Work Breakdown Structure (WBS) is here

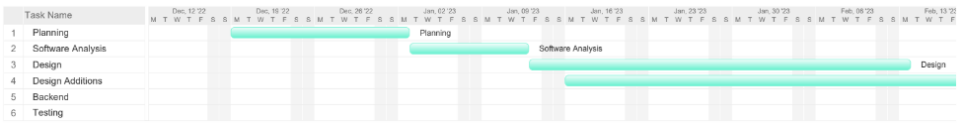



## 15. Network Diagram




## 16. Gantt Chart Diagram

Gantt Chart in here



## 17. Cost Estimation

| | | |
|---|---|---|
| 1. | XD for school and universities subscription | 70$ for 2 months |
| 2. | Computer (CPU: i7 127000, Ram: 16 GB, M.2: 512 GB) | 806$ |
| 3. | Server (Dell PowerEdge R350 Rack Server Data Center Bundle - 8GB - 2T) | 3,000$ |
| 4. | Windows Server OS | 2,200$ |
| 5. | Android Device | 400$ |
| 6. | 2 Developers | 2,000$ |
| 7. | Designer | 200$ |
| 8. | IT Manager | 3,000$ |
| | Sum | 9,876$ |

## 18. Risk Management

| Num | Risk | Level | Solution |
|---|---|---|---|
| **1** | SQL Injection or XSS Attack | Medium | Handle any illegal characters |
| **2** | Low performance | Medium | Use parallelism to do network operations |
| **3** | Design issues | High | Use responsive layouts |
| **4** | Database management challenges | Medium | Use Django framework |
| **5** | Third-Party Libraries Issues | Low | Keep update the dependency |
| **6** | Failed in resources | Medium | Alternative resources |

## 19. Project Management Team

| | | |
|---|---|---|
| Hana Nayroukh | Supervisor | hshabaneh@staff.alquds.edu |
| Motasem Radaydah | Leader | malradaideh@staff.alquds.edu |
| Ahmad Jber | Developer | ahmad.jber2@students.alquds.edu |