SECJ2203: Software Engineering

System Documentation SD

Equipment Usage and Booking System for University Laboratory Management Centre

Version 1.0

18th May 2021

Malaysia-Japan International Institute of Technology

Prepared by: BinaryBeasts

Revision Page

a. Overview

The current version of the document describes the introduction and specific requirements of the system. The scope, definition, reference, purpose, acronyms, abbreviation, and overview of the system are included in the introduction. External interface requirements, system features (including use case diagram, domain model, state machine diagram, activity diagram, and sequence diagram and use case specification), performance requirements, design constraints, software system attributes, and other requirements of the system are included in specific requirements.

b. Target Audience

Stakeholders and system analyst.

c. Project Team Members

Member Name	Role	Task	Status
MD Yusuf Bin Forkan	Moderator	Introduction, Table of Contents, use case specification (UC001, UC008)	✓ Complete ✓ Table of Contents-incomplete
Ruhul Quddus Tamim	Team Leader	Performance requirement, design constraint, software system attributes, other requirements, use case specification (UC002, UC009)	complete
Shafi Ahmed	Task Manager	System features, use case specification (UC005, UC006, UC007)	complete
Syafiq Ibnu Ramadhan	Secretary	External interface requirements, use case specification (UC003, UC004)	complete

d. Version Control History

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Ruhul Quddus Tamim MD Yusuf Bin Forkan Shafi Ahmed Syafiq Ibnu Ramadhan	Combined SRS, SDD, STD as SD	18/05/2021

1. Table of Contents

1. Introduction

1.1 Purpose

This SD describes the Equipment booking/borrowing system which is to make a new version of the system based on the previous manual system. SD is the description, specification, development, and deployment of a new software application. They may involve the internal production of specialized applications, the construction of a database. The requirements in this SD are elicited from the stakeholder of DVCRI and PPMU offices at UTMKL. We documented all the requirements as it is important to make sure that the project team and the stakeholders understand the project.

UTM's PPMU department decided that it is important to facilitate and ease the process of helping students/Academic staff, Lab Manager/Technicians for managing booking usage and borrowing by easing the applying process through their UTM account. This document will represent and illustrate the sub-processes of this proposed system to give a better understanding of how the system works and integrate.

1.2 Scope

The software product is an Equipment booking/borrowing system. Generally, this system will allow the Students/Academic staff to book and borrow university laboratory equipment online.

What our system can do:

- Providing a function that enables student/academic staff to add book/borrow equipment through the system for every lab and unit.
- Providing a function that enables Lab Technician/Manager to approve/cancel the pending request for booking/borrowing for approval purposes.
- Lab Technician/Manager can create a new record for a certain booking/borrowing on the specific date when equipment is available for booking/borrowing.
- Lab Technician/Manager can view and track the equipment usage through databases.
- Management (Director) can fill out laboratory details for registering labs and units and can also modify/delete the record.
- The system will let students/academic staffs view their current booking/borrowing status and return date or duration of time for use.
- The system will notify students/academic staff to return the equipment after the expiration of duration usage.

- The system will automatically analyze students' and academic staff of equipment booking/borrowing usage and record the data in the database.
- The system will show the usage report to the students/academic staff, lab manager/technician, and Management (Director).
- The system shall be able to enable all the stakeholders to download the report for booking/borrowing usage.

What our system cannot do:

- The system will not be able to let the students/academic staff book/borrow more than one different piece of equipment from different labs.
- The system will not let the students/academic staff view the reports for other students/academic staff.

The objectives and goals of this system are to upgrade the current system and design the system which meets the administrative, organizational, and management requirements of an organization. In addition, this equipment booking/borrowing system also allows the users to access the system anytime, anywhere with an internet connection.

1.3 Definitions, Acronyms, and Abbreviation

1.3.1 Definitions:

i. As-In Processes are the current existing processes in the system.
ii. Proposed (To-Be) processes are the modified and developed processes that are going to be introduced and implemented into the system.
iii. Functional Requirements are the requirements that determine how the system behaves and acts.
iv. Non-Functional Requirements are the qualities and characteristics of the system.
v. Usability is how easy the system and friendly it is.
vi. Efficiency is the quality of managing time and finishing things quicker.
vii. Swimlane is an illustration of a diagram that shows the flow process of how the system works, starting from the beginning until the end.

1.3.2 Acronyms:

i. SD - System Development DVCRI *term* is the Deputy Vice-Chancellor (Research & Innovation) at UTM KL is the office that serves as a center of Administration and Research Support, and Centre of Excellence in UTM Kuala Lumpur.

ii. PPMU *term* is the University Laboratory Management Centre that provides quality laboratory services for analysis, scientific, and engineering testing by using the latest equipment with competent and committed manpower.

iii. DVCRI *term* is the Deputy Vice-Chancellor (Research & Innovation) at UTM KL is the office that serves as a center of Administration and Research Support, and Centre of Excellence in UTM Kuala Lumpur.

1.4 References

1. HPE Apollo 20 System. (n.d.). HPE Apollo 20 System Retrieved from https://www.hpe.com/my/en/product-catalog/servers/proliant-servers/pip.1012228095.html

2. Universiti Teknologi Malaysia. (2017). UTMDigital. Retrieved from digital.UTM.my

3. Software Quality Attribute. (2018). Software Quality Attributes Retrieved from http://www.qasigma.com/2008/12/software-quality-attributes.html

1.5 Overview

The flow of the Equipment booking/borrowing system operations is determined and represented by an As-is swimlane diagram. After clarifying the current processes, the to-be system is proposed in a separate swimlane diagram. This document consists of the Proposed To-Be System which includes the To-Be Swimlane model that will explain more detail about the process of booking/borrowing equipment online. Students and Academic staff log into the system for the booking and borrowing management system. It is also a UTM single-on system, which means it shares the same information of the students and lecturers as in the MyUTM. If students or the Academic Staffs want to book the laboratory's equipment, PPMU (Lab Technician/Manager) will check the date and time of the equipment reservations for booking or borrowing and approve the reservation if the reservation's time and applied laboratory is appropriate. Then, the data from the application form will automatically store in the database. Then, a report will be generated for every month regarding the equipment used which is viewed by all the stakeholders.

2. Specific Requirements

2.1 External Interface Requirements

2.1.2 User Interfaces

When a user uses the Equipment booking/borrowing system for the first time, he or she needs to login using the same ACID ID by inserting a username and a password. After that, the user can login to the system successfully. When the user successfully login the system, the system shall display the main page that provides services that are relevant to the identity of the user. If the user is a student, he or she will get to see functions such as view labs, view units, view equipment, and view their dashboard of booking/borrowing usage. If the user is an academic staff, he or she will see the same functions as students. However, the account is different.

For the “view labs” function, the user will be brought to another page that will display the list of labs that he or she can choose. Under each lab, there are 4 units and for each unit, there is a “view equipment” function. For the “view equipment” function, the user will be bought to another page that will display the list of equipment. The interface is very easy to learn and adapt as it is quite similar to some interfaces found in other familiar software.

For booking/borrowing, the system will display to the interface where the user can view equipment and fill up the information for booking/borrowing usage. On the other hand, the booking/borrowing request will be sent to the Lab manager/technician who manages labs and he or she is responsible for the approval purposes if the equipment is available for booking/borrowing. The interface shall be clear and easy to understand what to do and when the approval will be approved. After filling approval, the user can see the booking/borrowing is approved and can proceed on the specific date to the labs for booking usage/borrowing.

As for the “view report” function, the user can simply view the report stats of every student/academic staff by finding the name or the matric number. After choosing a certain student/academic staff, the Lab Technician/Manager, as well as the Management (Director), can view the results and booking/borrowing usage report of the students/academic staff. The reports are very easy to understand so that the user can make certain decisions easily. The specific report is visible to the specific students only and specific academic staff only on their dashboard.

2.1.2 Hardware Interfaces

The University Laboratory Equipment booking/borrowing system is required to communicate with the server as the system is a web-based system. With the existence of a server, it can store, receive, and send files and data to computers through a network. For our project, we use a server made by Hewlett Packard Enterprise (HPE).

- Name: HPE Apollo 20 System
- Processor Family: Intel Xeon Platinum 9200 Processor family
- Memory: DDR4-2933 Registered Memory
- Network: 1 GB integrated with PCI-e options available (per node)
- Expansion slots: supports 4 PCIe x 16 slots (per node)
- Storage: M.2 and U.2 storage options

2.1.3 Software Interfaces

1. The University Laboratory Equipment booking/borrowing system shall require any web browser including Google Chrome, Mozilla Firefox, and Microsoft Edge to operate.

Name: Google Chrome Version: 80.0.3987.149
Source: https://www.google.com/chrome/
Name: Mozilla Firefox
Version: 73.0.1
Source: https://www.mozilla.org/en-US/firefox/new/

Name: Microsoft Edge
Version: 80.0.361.62
Source: https://www.microsoft.com/en-us/edge

2. The server needs to be taken control of by using a proper server operating system such as Windows Server.
Name: Windows Server
Version: 10.0.17763
Source: https://www.microsoft.com/en-us/cloud-platform/windows-server

3. To manage all the data of booking/borrowing and report of equipment booking/borrowing system, Database management system (DBMS) is very much needed.
Name: Oracle Database 19c
Version: 19.1.0 // 12.2.0.3
Source: https://www.oracle.com/index.html

2.1.2 Communication Interfaces

The Equipment usage and booking system for University Laboratory Management Centre shall always communicate with the server through several internet protocols such as Hypertext Transfer Protocol (HTTP), Domain Name System (DNS), and File Transfer Protocol (FTP). The UTM online equipment booking/borrowing system will interface with a Local Area Network (LAN) to maintain communication with all its devices. It should use a reliable-type IP protocol such as TCP/IP or reliable-UDP/IP for maximum compatibility and stability. All devices it will interface with should contain standard Ethernet compatible, software-accessible LAN cards to maintain communication between the server and the surface computers, tablets, mobile devices and displays. Wireless devices should also use Ethernet compatible cards, using the IEEE 802.11b/g standard and having support for WPA2-PSK encryption.

2.2 System Features

The system features include user booking/borrowing, manage lab and unit for management, manage book/borrowing for a lab manager, and download report for all the users, Admin panel for System Admin.

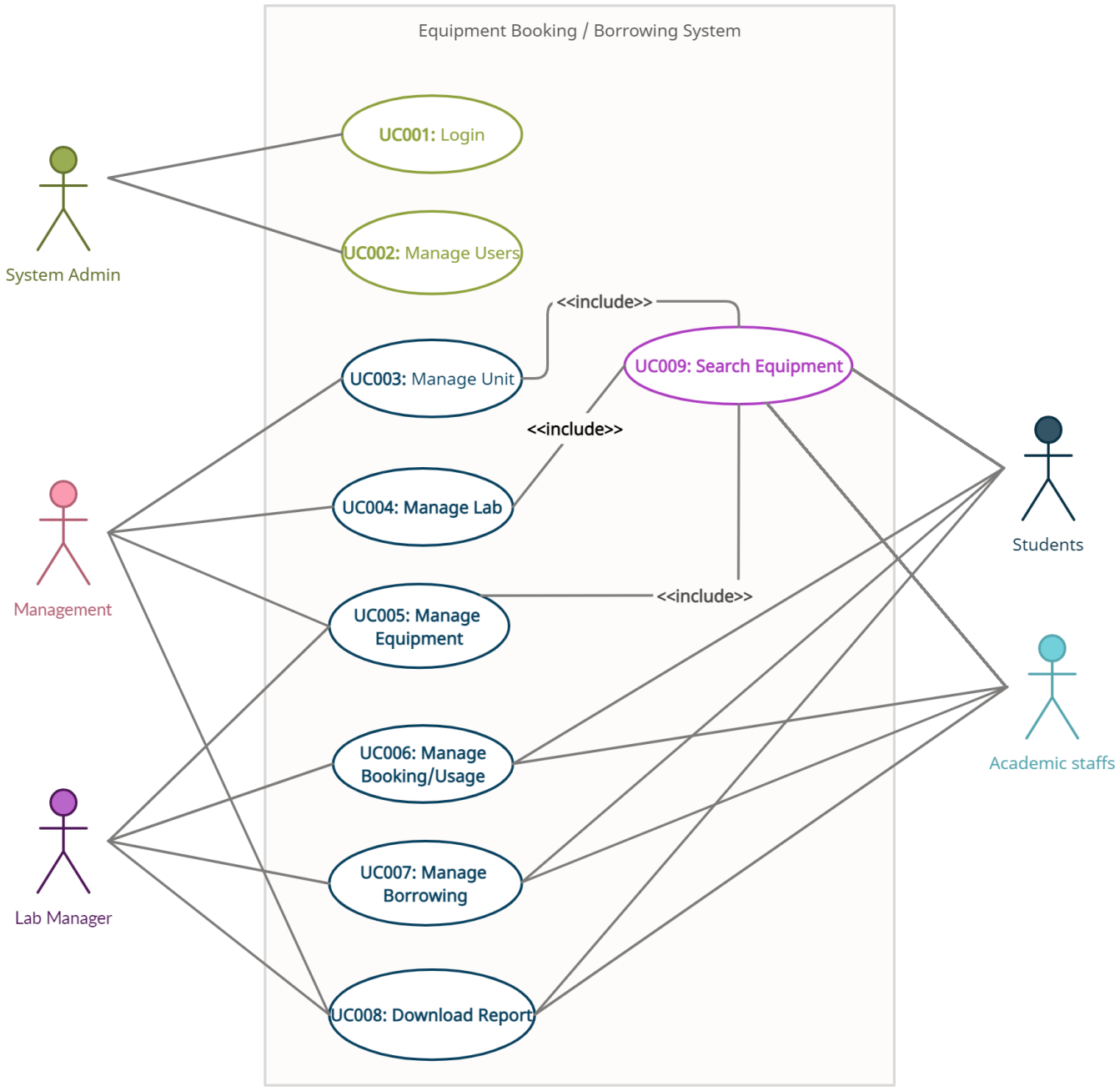


Figure 2.1: Use Case Diagram for Equipment Booking and Borrowing Online System

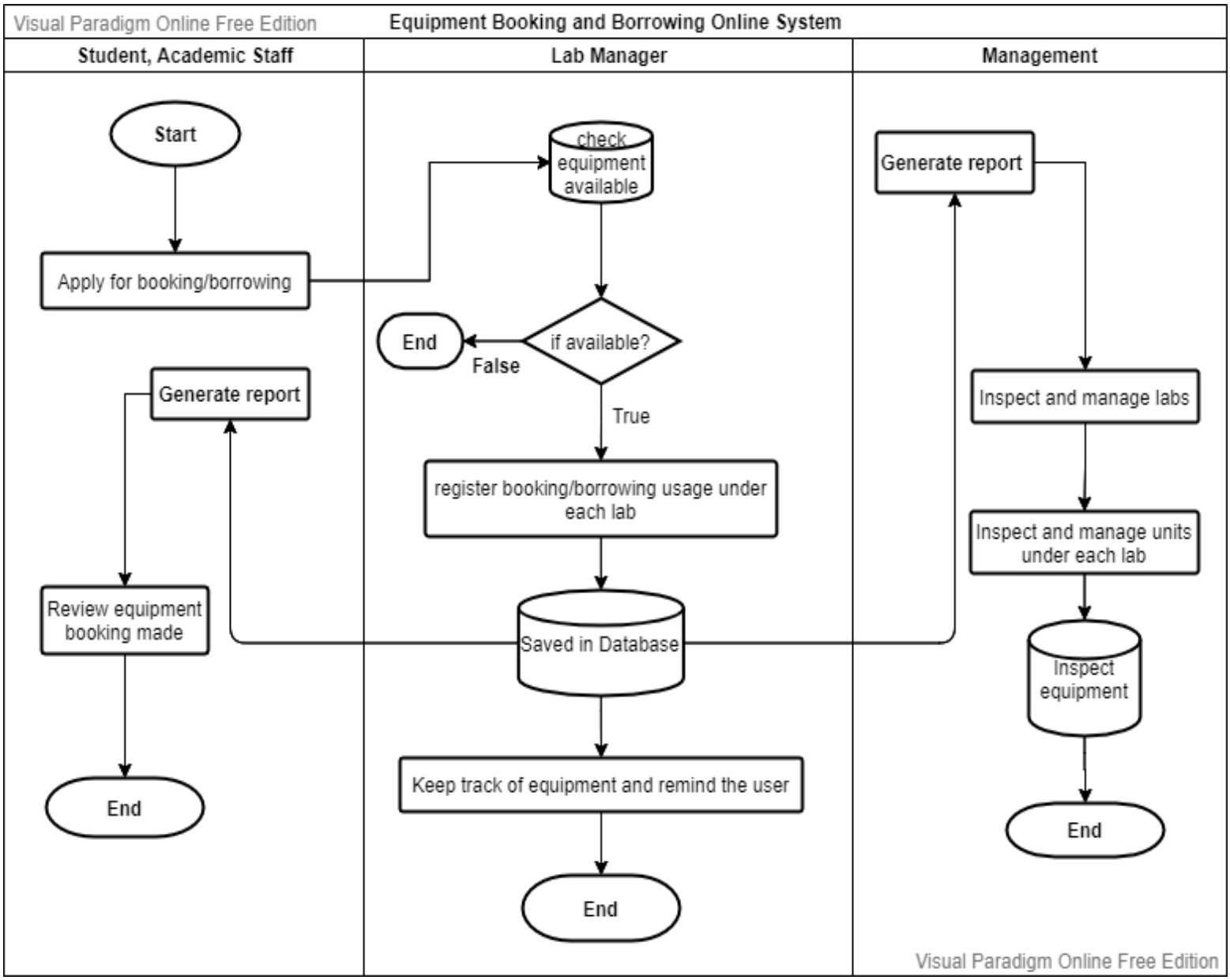


Figure 2.2: Activity Diagram for Equipment Booking and Borrowing Online System

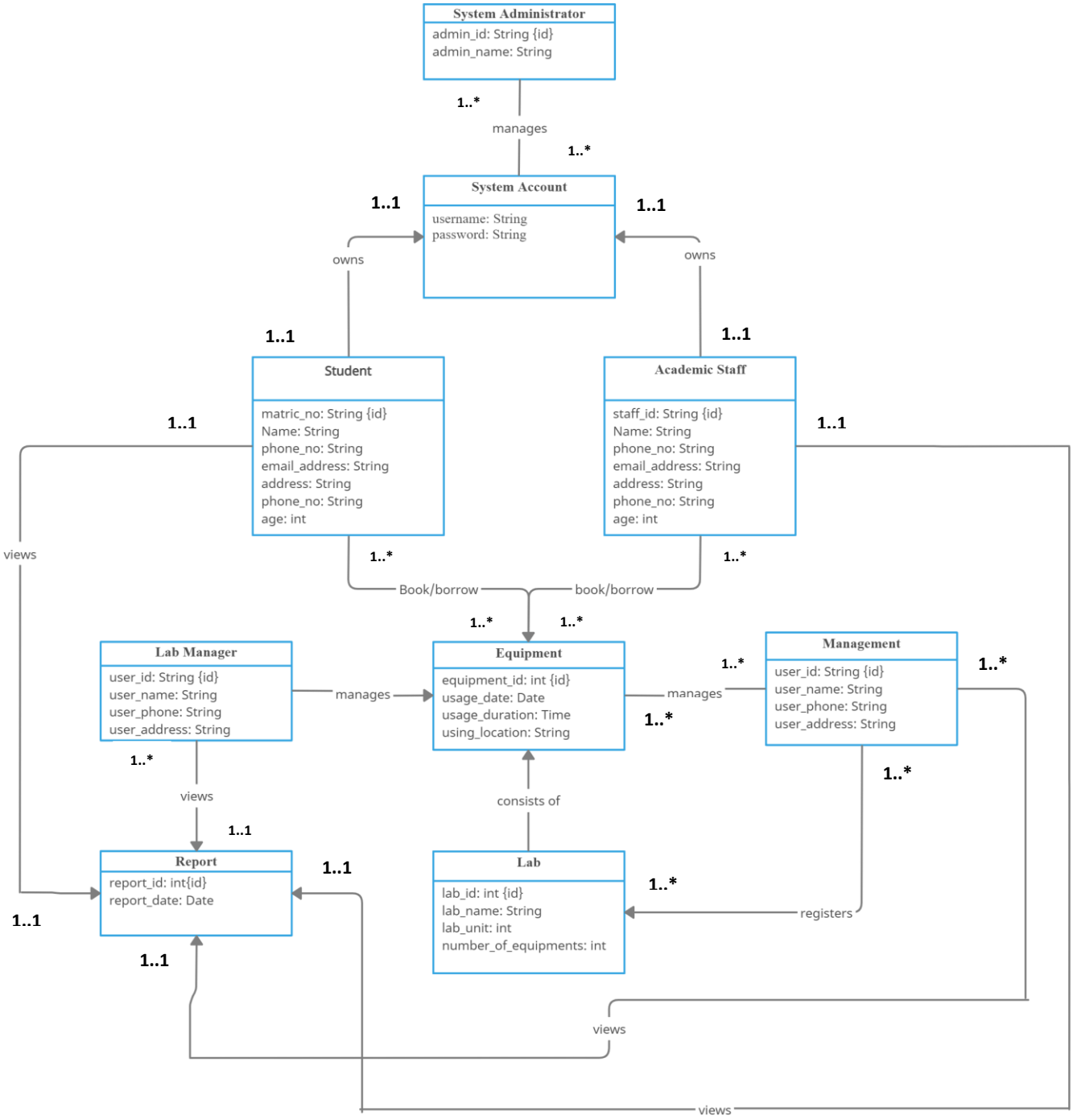


Figure 2.3: Domain Model for Equipment booking and borrowing system online

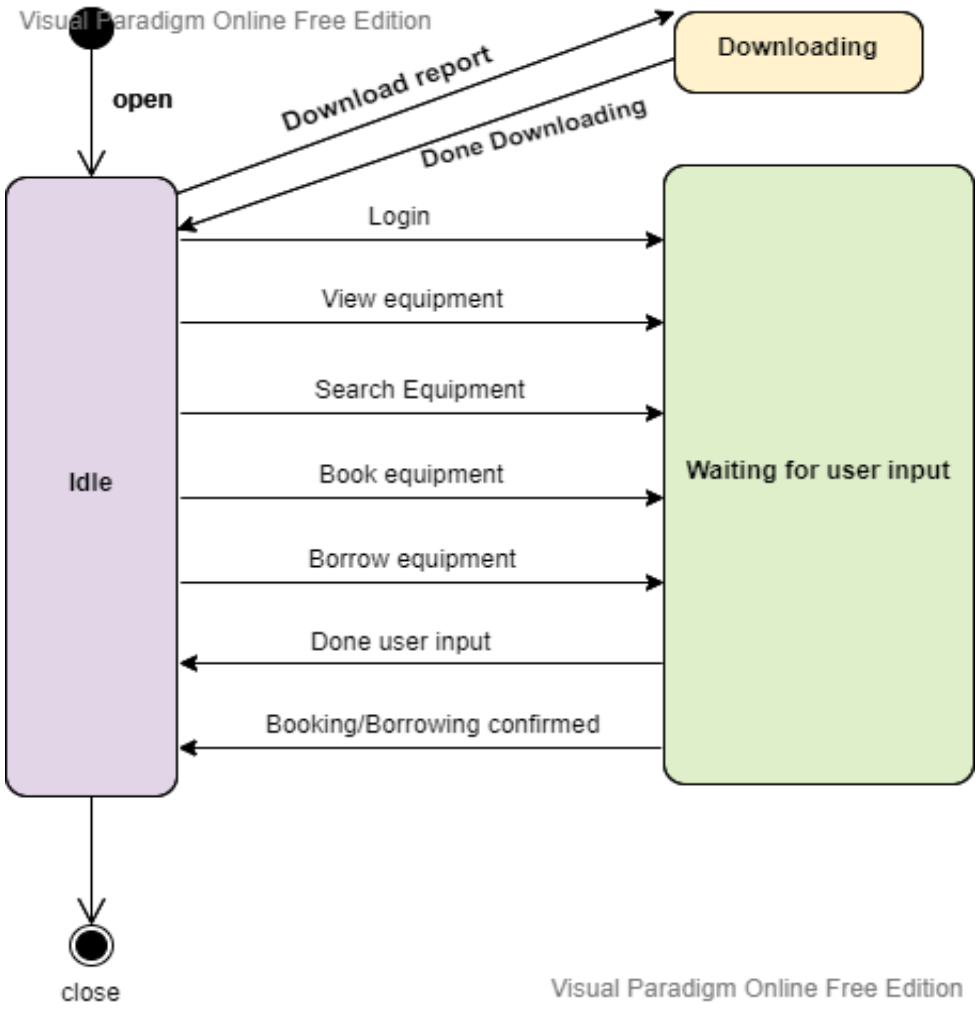
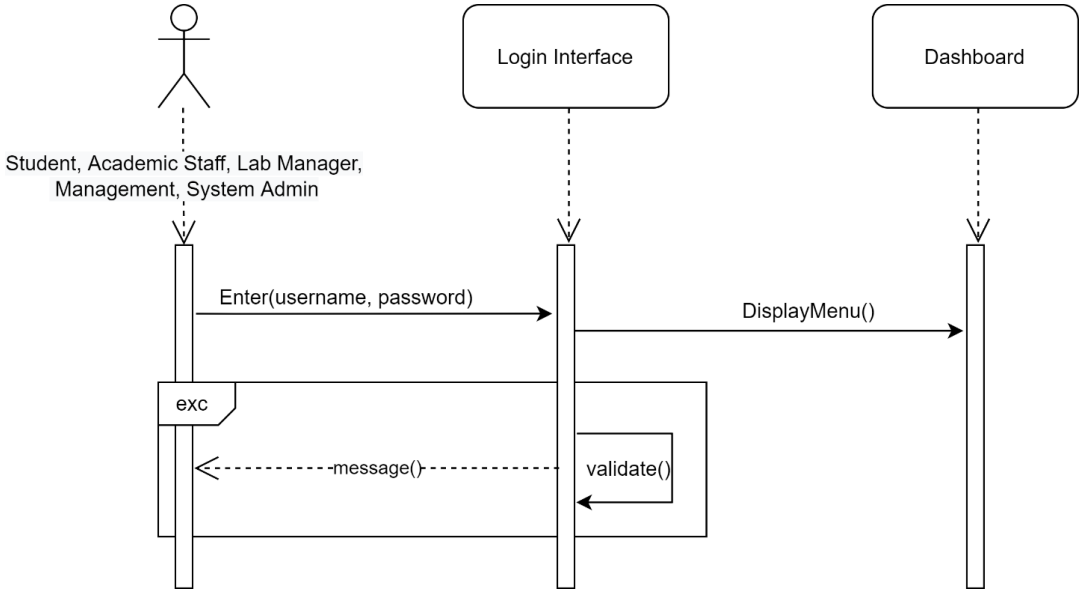


Figure 2.4: State Machine Diagram for equipment booking and borrowing system online

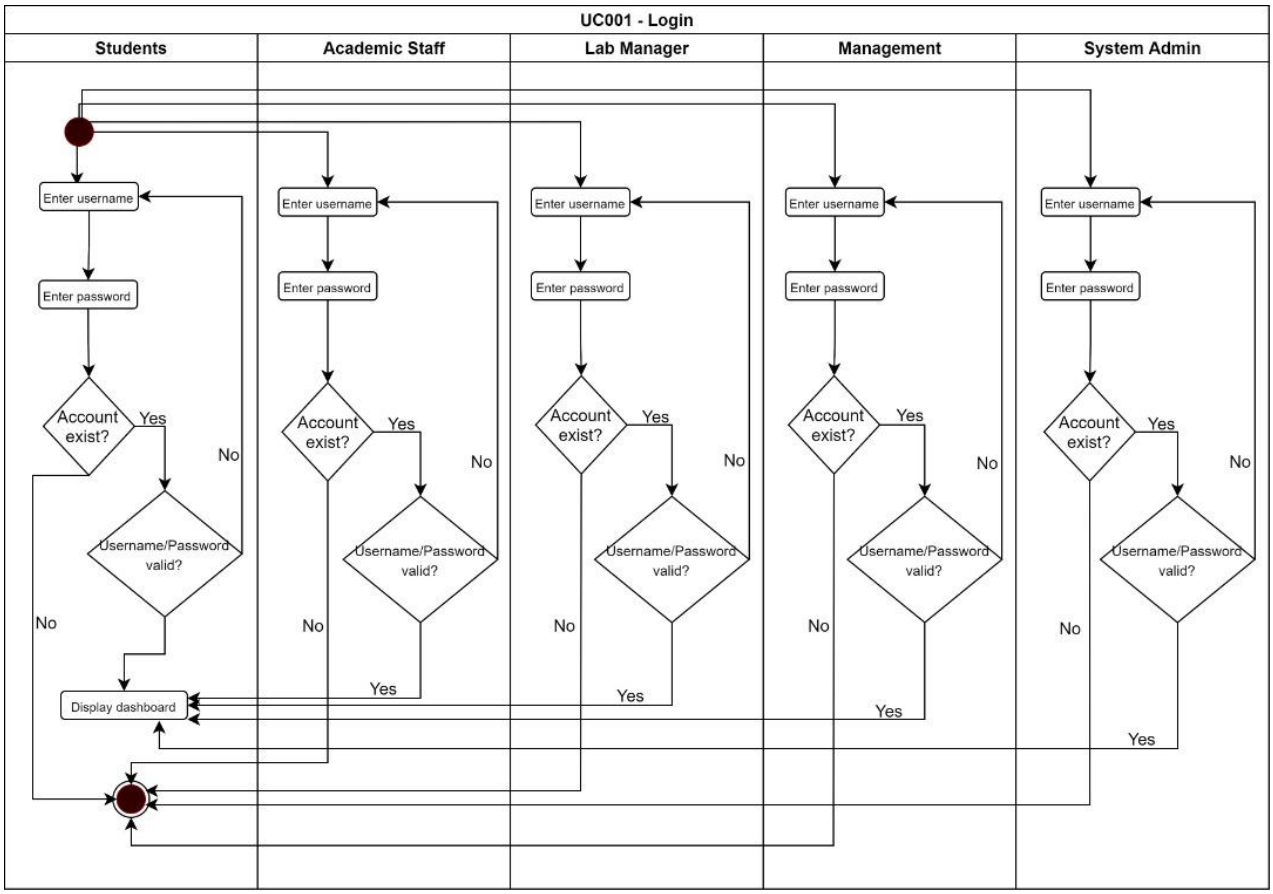
2.2.1 UC001: Login

Use case specification for UC001

Use Case name	Login
Use case ID	UC001
Actors	Student, Academic Staff, Lab Technician/Manager, Head of Units and Director (Management), System Admin
Pre-conditions	
Normal flow	1. User type in username and password based on their role/type of account. 2. The user then clicks the “Login” button. 3. If information is valid, user can view their dashboard depending on the type of account. If username and password are entered wrong, Exception E1 is performed. If the account does not exist in the system, Exception E2 is performed.
Post Conditions	Account successfully validated and the actors can view the dashboard.
Exception flow	E1. Invalid password or username is entered If in step 2 in the normal flow, the user submits the wrong password or username, then - 1. The system will display a message to indicate to the user that either the password or username is wrongly entered. 2. The user will need to return to step 2 again to enter the proper username and the password in step 3 in normal flow. E2. Username does not exist in the system If in step 4 in the normal flow, the user submits the username or password that does not exist in the system, then 1. The system will display a message to indicate to the user that he or she has no ACID account/ID.



Sequence Diagram for Login

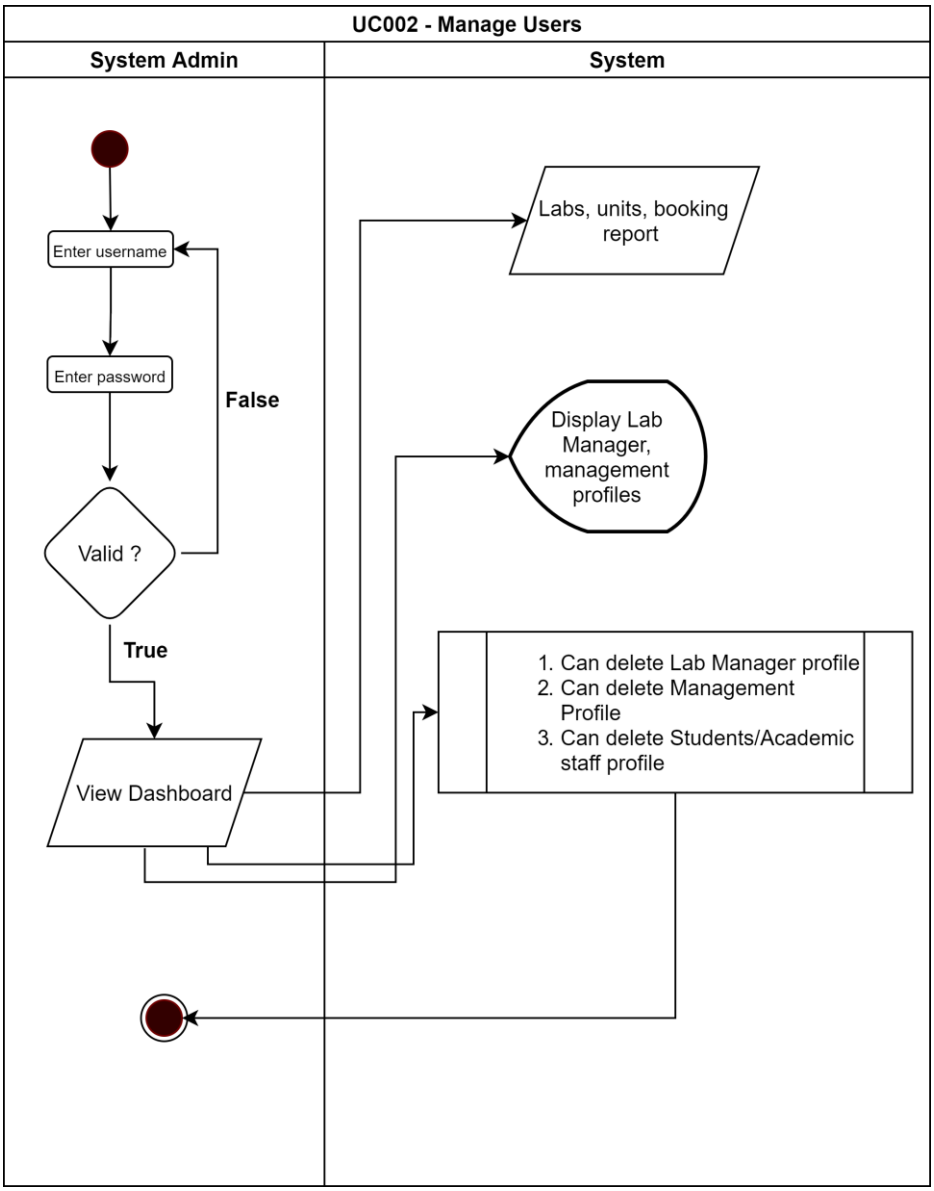


Swimlane Diagram for Login

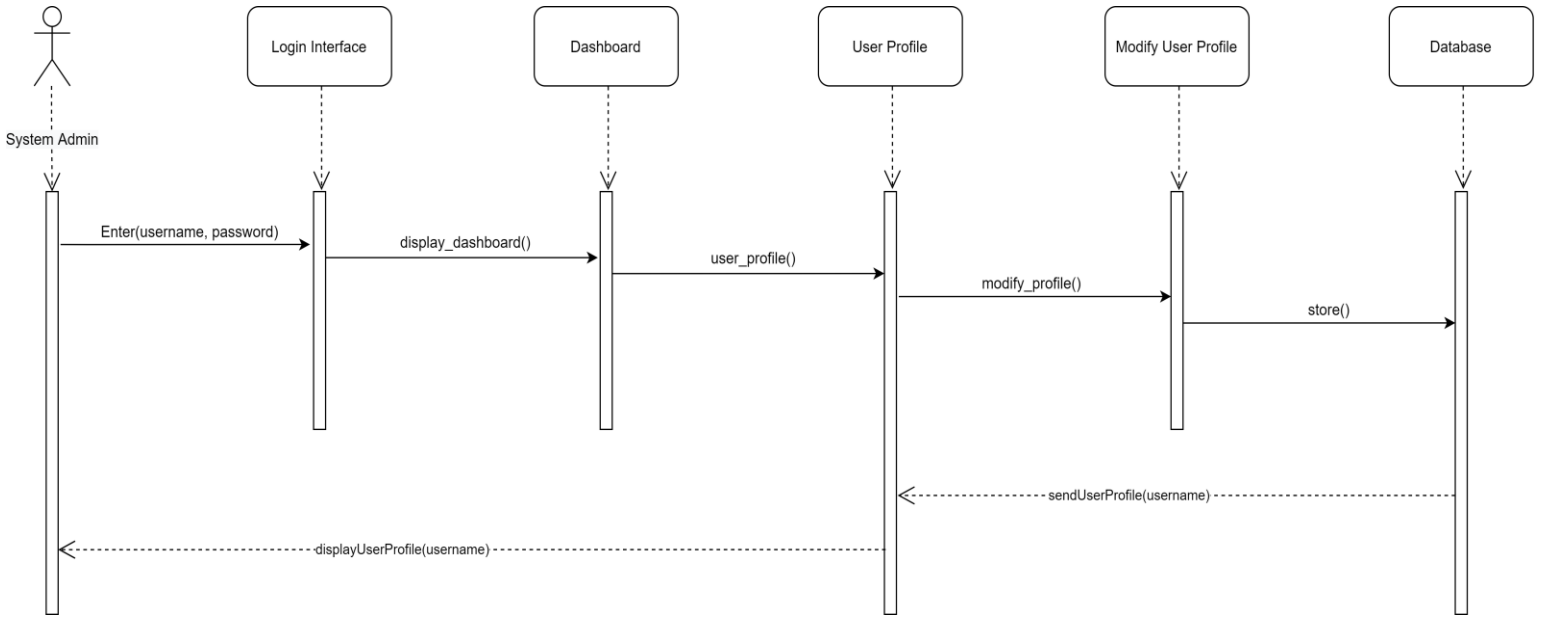
2.2.2 UC002: Manage Users

Use case specification for UC002

Use Case name	Manage Users
Use case ID	UC002
Actors	System Admin
Pre-conditions	System Admin needs to be logged in.
Normal flow	1. Admin types in username and password. 2. Then clicks on the "Login" button entering the system panel. 3. Admin can view all – registration of units, labs, equipment, review booking and borrowing of equipment and report. 4. Admin can delete the profile for Lab technician/manager, Management (Director), Students, and academic staff.
Post Conditions	Specific operation in any of the normal flow is done successfully.



Swimlane diagram for Manage Users

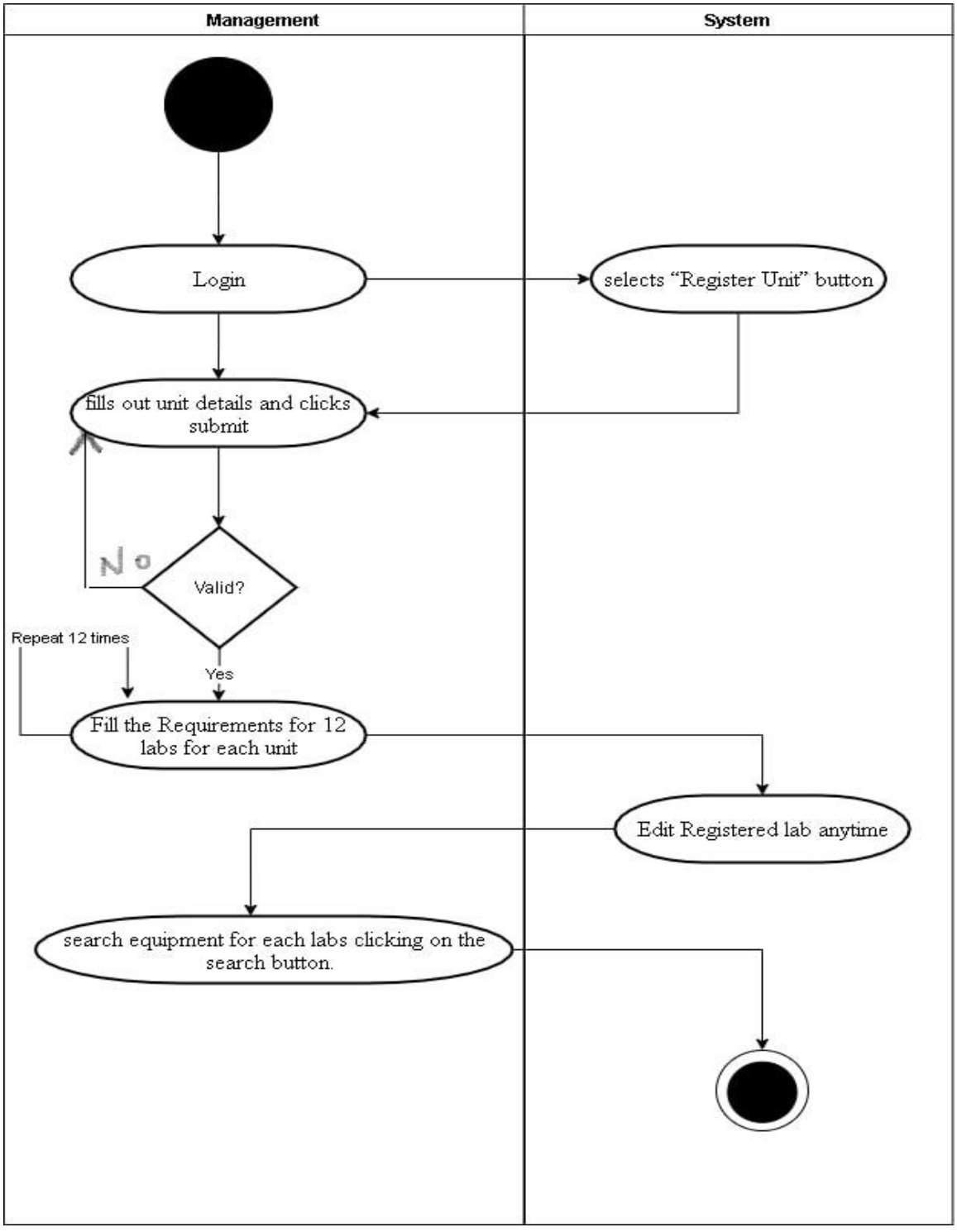


Sequence Diagram for Manage Users

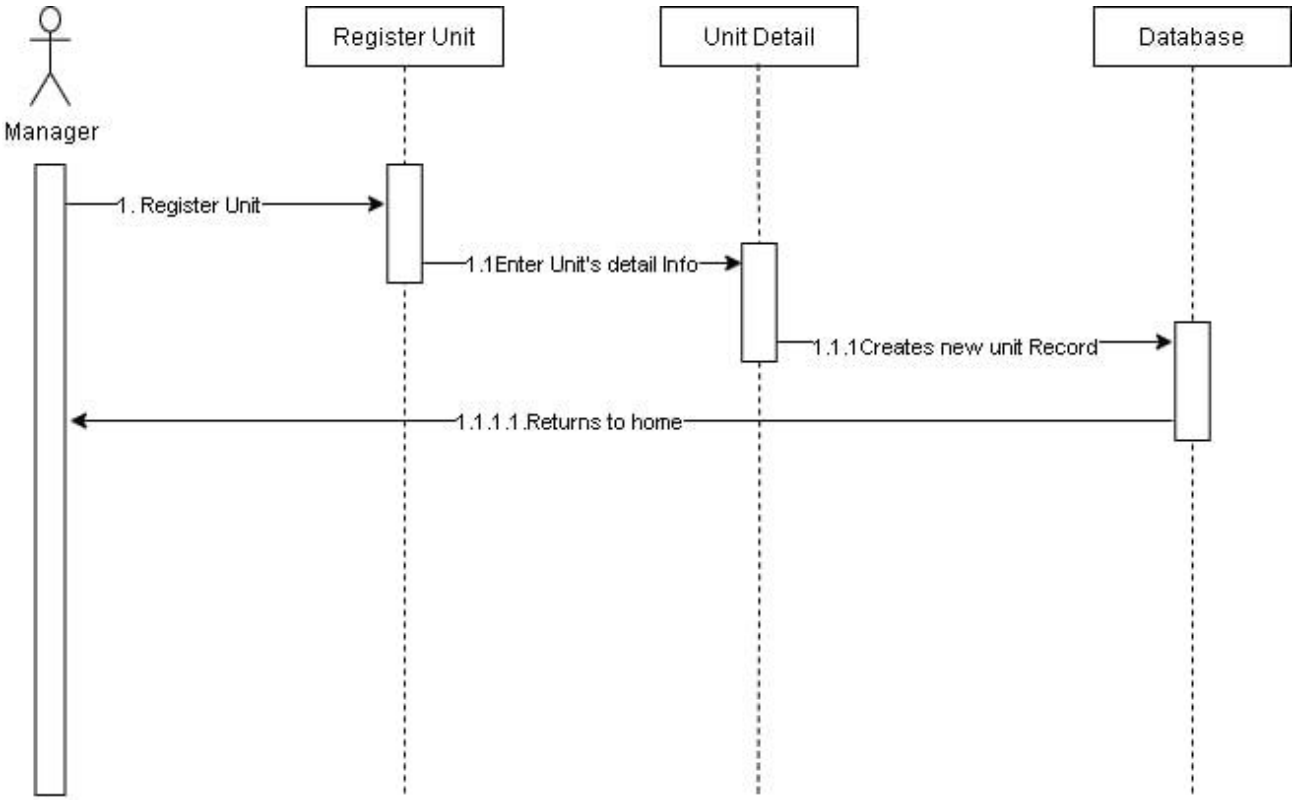
2.2.3 UC003: Manage Unit

Use case specification for UC003

Use Case name	Manage Units
Use case ID	UC003
Actors	Management
Pre-conditions	Management needs to be logged in.
Normal flow	1. Management selects the “Register Unit” button. 2. Management fills out unit details and clicks submit. 3. If information is valid, the system creates a new unit record that prompts the user that the operation has been successful returns home. 4. The user repeats the process for 4 units (Electrical, Mechanical, Civil Engineering, Service) 5. Management can search add/delete registered units anytime
Post Conditions	A new unit record is created
Related Documents	UC009
Alternative flow	In normal flow, when the user needs to repeat the process to fill 4 units the user can leave it unfinished because the previous choice will be saved into the system(more like a draft).
Exceptional flow	The system may notify the user in case incorrect information is entered and registration was not successful.



Swimlane diagram for Manage Unit

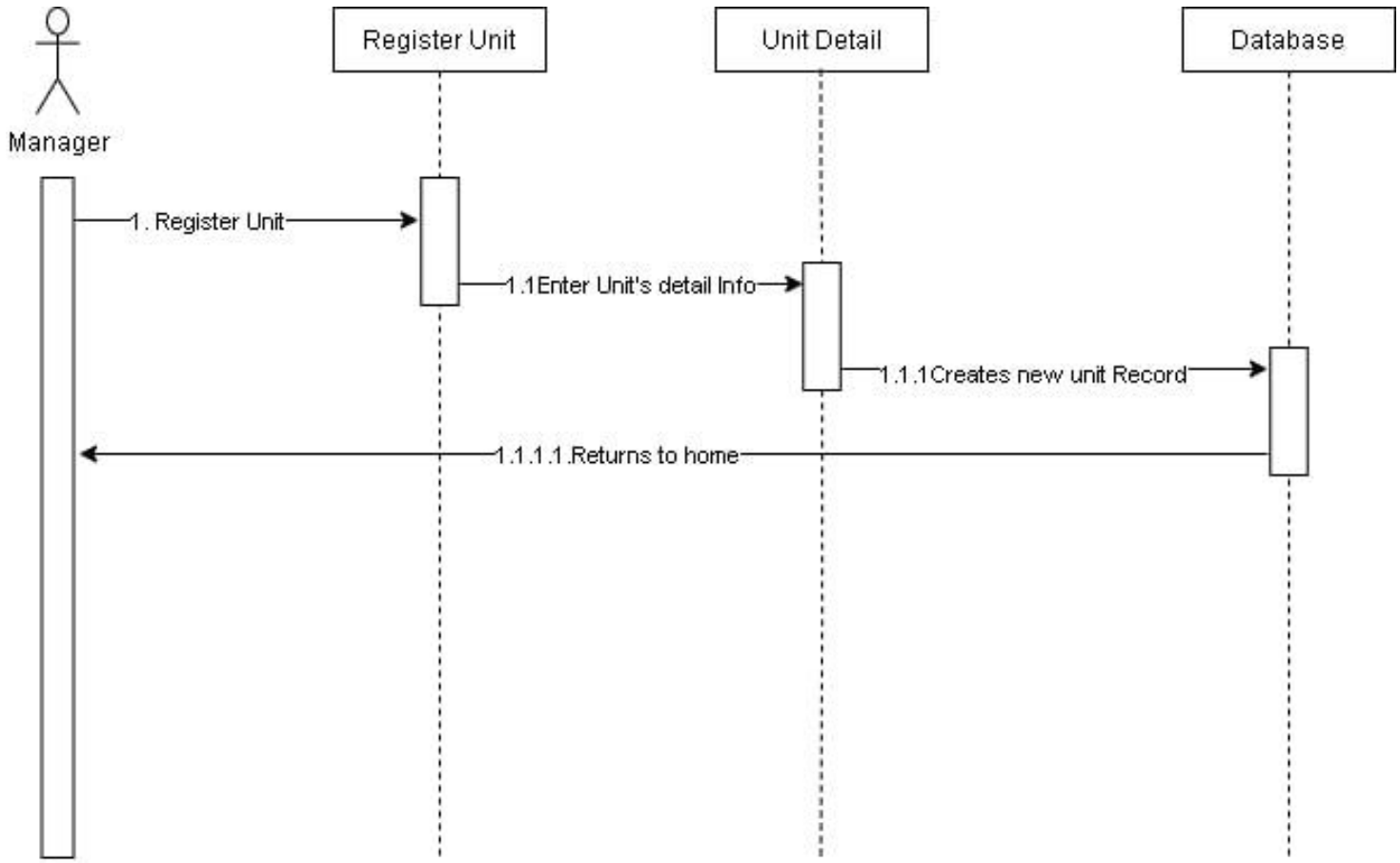


Sequence diagram for Manage Unit

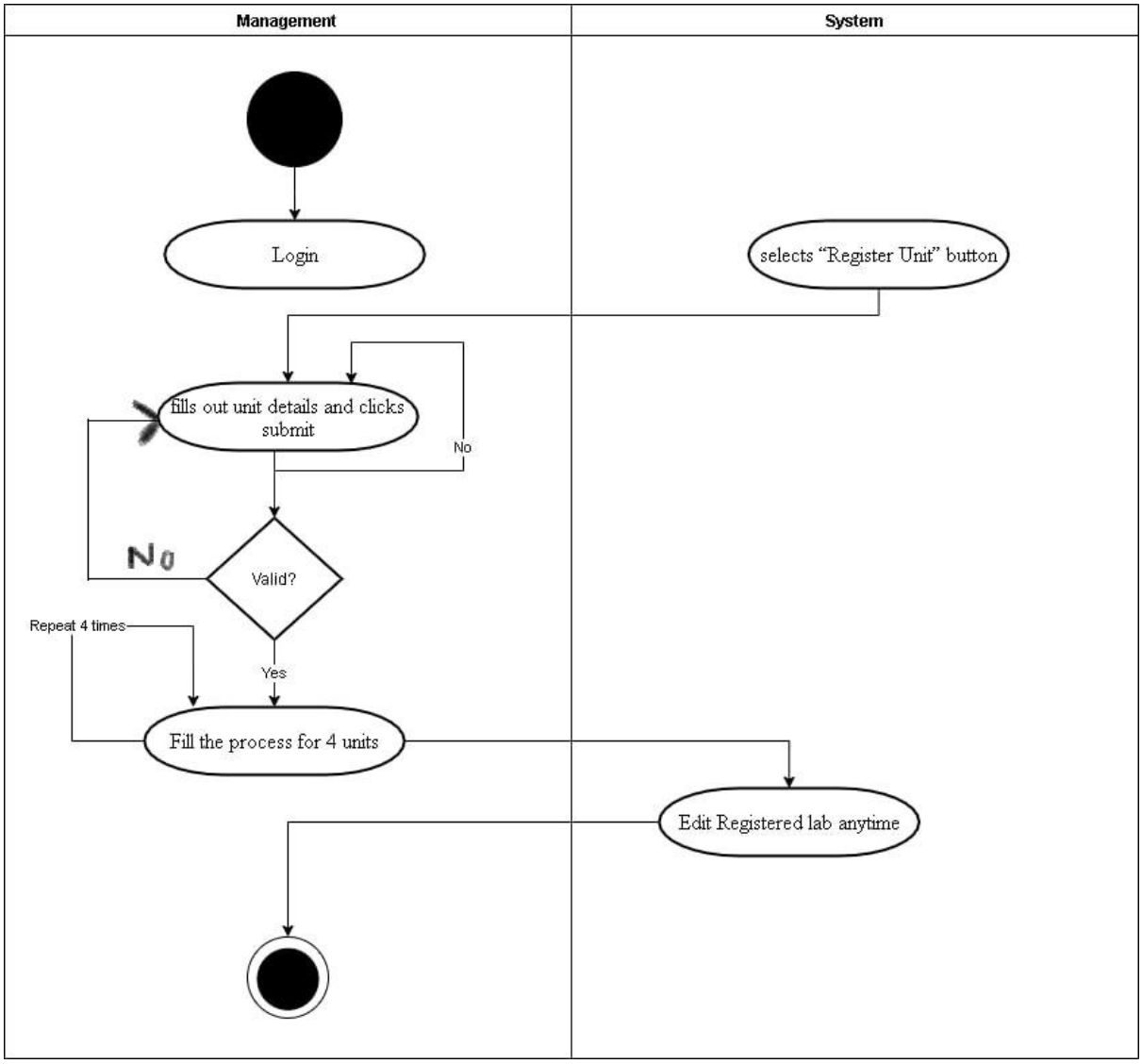
2.2.4 UC004: Manage Labs

Use specification for UC004

Use Case name	Manage Labs
Use case ID	UC004
Actors	Management
Pre-conditions	Management is on the specific unit page.
Normal flow	1. Management selects the "Register Lab" button. 2. Management fills out laboratory details and clicks submit. 3. If information is valid, the system creates a new lab record that prompts the user that the operation has been successful returns home. 4. Repeats the process for 12 labs for each unit. 5. Management can add/delete the registered labs anytime. 6. Management can search equipment for each lab by clicking on the search button.
Post Conditions	A new lab record is created
Related Documents	UC009
Alternative flow	In normal flow, when the user needs to repeat the process to fill 4 units the user can leave it unfinished because the previous choice will be saved into the system(more like a draft).
Exceptional flow	The system may notify the user in case incorrect information is entered and registration was not successful.



Sequence diagram for Manage Unit

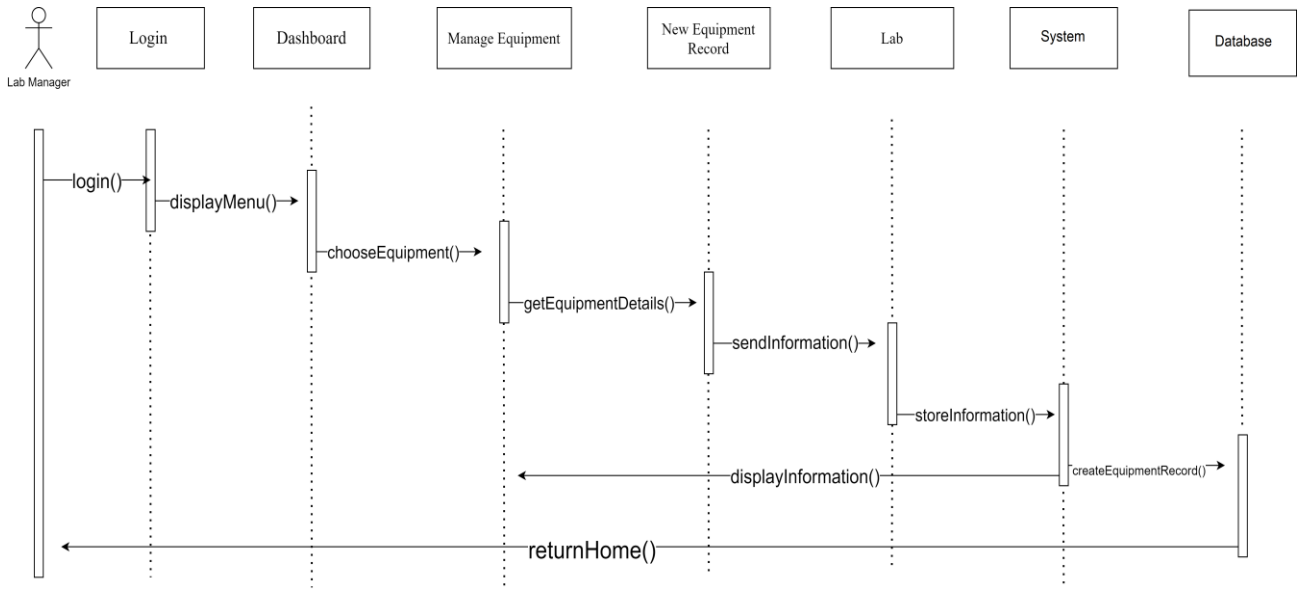


Swimlane diagram for Manage unit

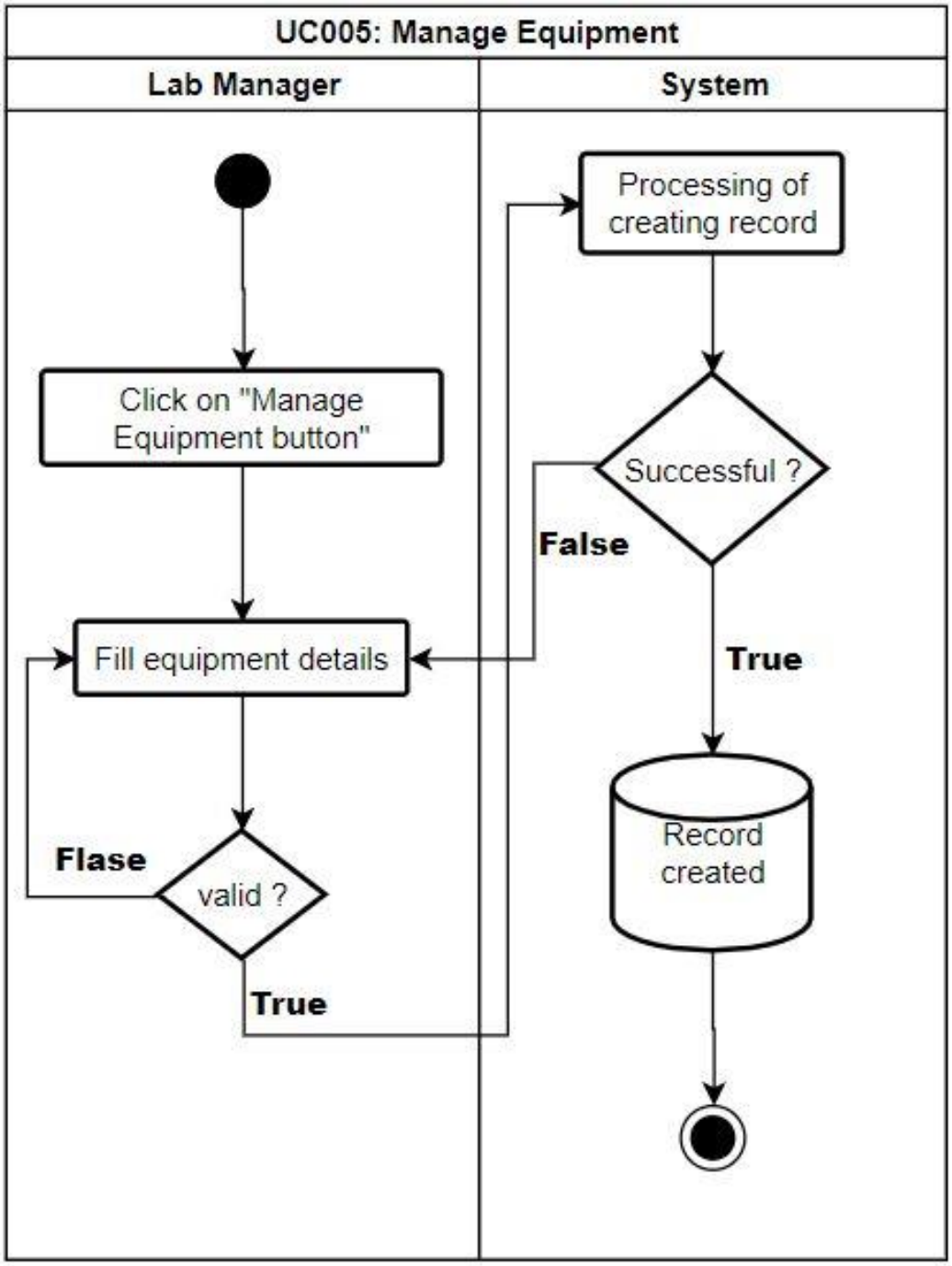
2.2.5 UC005: Manage Equipment

Use case specification UC005

Use Case name	Manage Equipment
Use case ID	UC005
Actors	Lab Manager, Management
Pre-conditions	Lab Manager/Management needs to be logged in
Normal flow	1. Manager/Management selects the "Manage Equipment" button. 2. Manager/Management fills out equipment details and clicks submit. 3. If information is valid, the system creates a new equipment record that prompts the user that the operation has been successful returns home. 4. Manager/Management repeats the process for the required amount of equipment under each lab. 5. Manager/Management deletes equipment if needed. 6. If successful, the system prompts the user that the operation was successful.
Alternative Flow	While repeating the process the user can leave the process incomplete. Just because the previous one is already saved in the system.
Post Conditions	User can view equipment usage
Related Documents	UC009
Exceptional Flow	An error message will be shown if any of the equipment details are not valid.



Sequence diagram for Manage Equipment

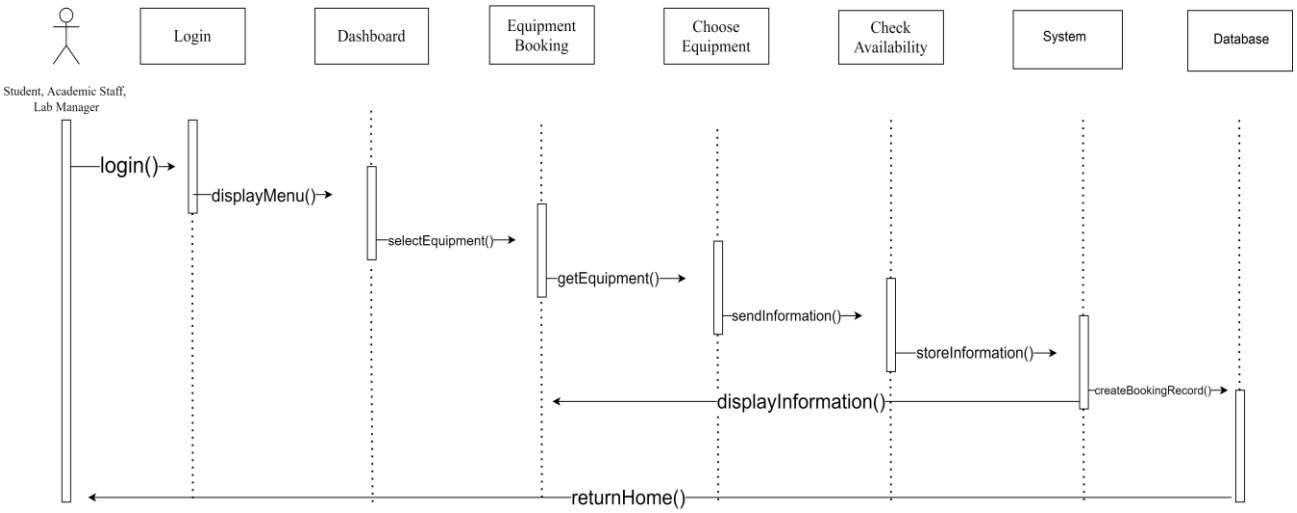


Swimlane Diagram for Manage Equipment

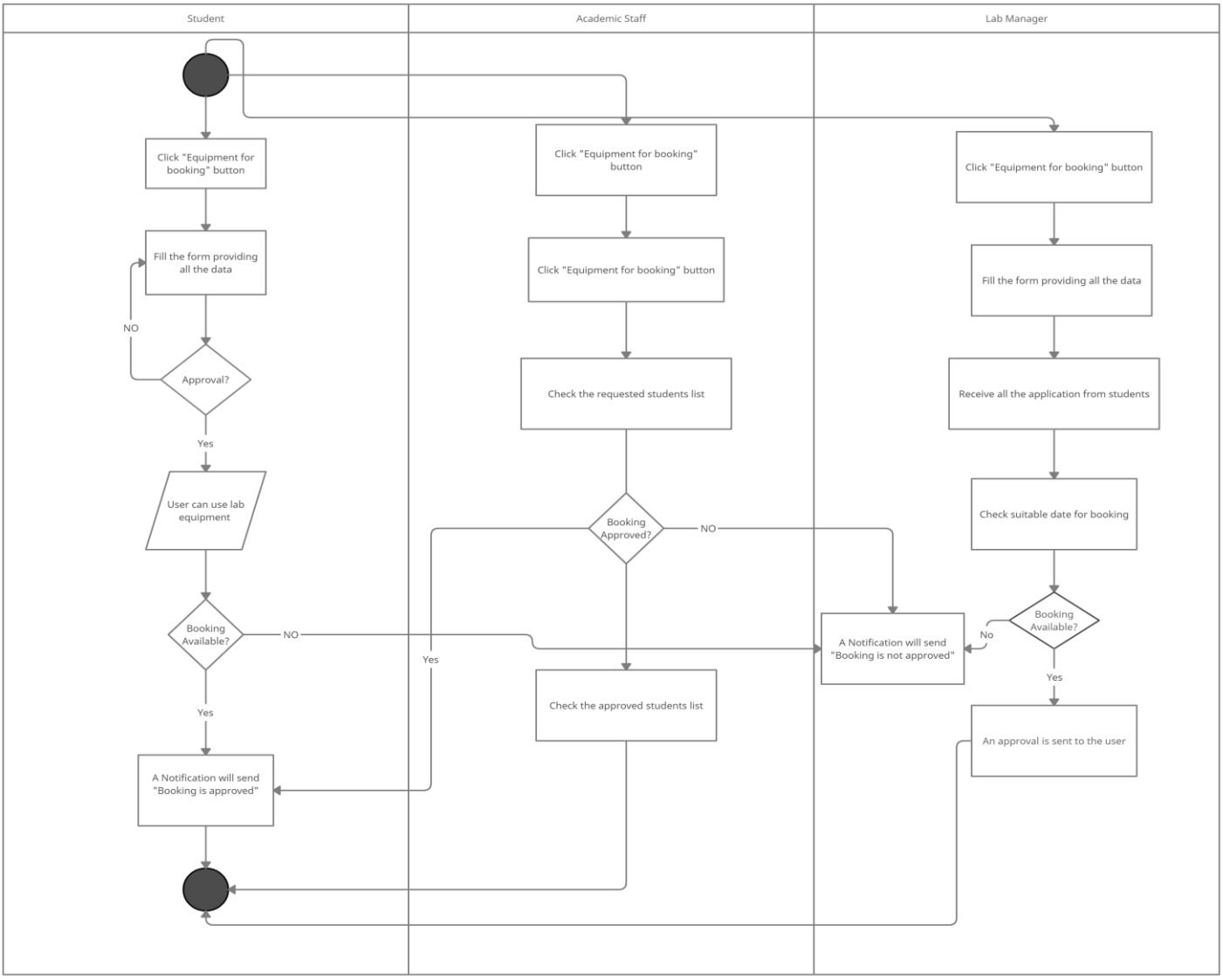
2.2.6 UC006: Manage Booking

Use case specification for UC006

Use Case name	Manage Booking
Use case ID	UC006
Actors	Student, Academic Staff, Lab Manager
Pre-conditions	User needs to log into their account with their username and password.
Normal flow	1. User clicks on the “Equipment for Booking” button. 2. User fills the form providing the date of using the lab, category of lab, and purpose of using the lab equipment. 3. User waits for approval of booking. 4. Once approved, the user can use lab equipment on the specified booking date. 5. Lab Manager checks the suitable date for booking depending on the availability of equipment. 6. If available for booking, the unique id of equipment, date of usage, and purpose of using the equipment is being recorded by the Lab Manager. 7. Then booking is confirmed, and approval is sent to the Student.
Post Conditions	A tracing record for the equipment booking has been created in the logbook.
Alternative Flow	While filling the form again the user can use the previous information which is already saved in the system.
Exception flow	Invalid information is entered If in step 5 in the normal flow, the Lab Manager founds the date of booking is not available as well as the equipment, then 1. The system will display a message to indicate to the user that either the date or equipment is wrongly entered. 2. The user will need to return to step 1 again to fill the proper Equipment booking form. 3. Use case end.



Sequence diagram for Manage Booking/Usage

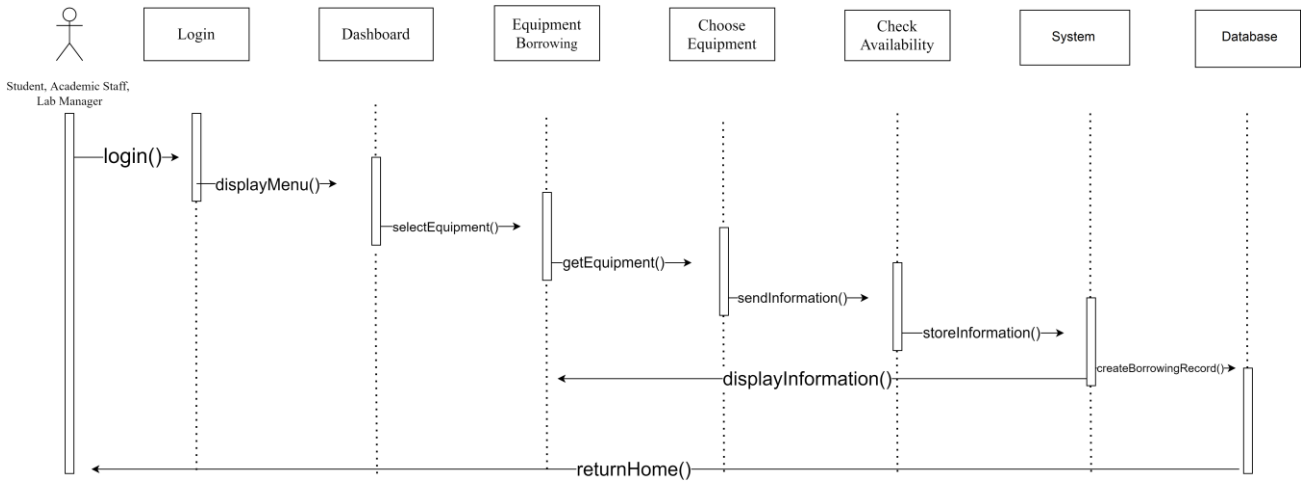


Swimlane diagram for Manage Booking/Usage

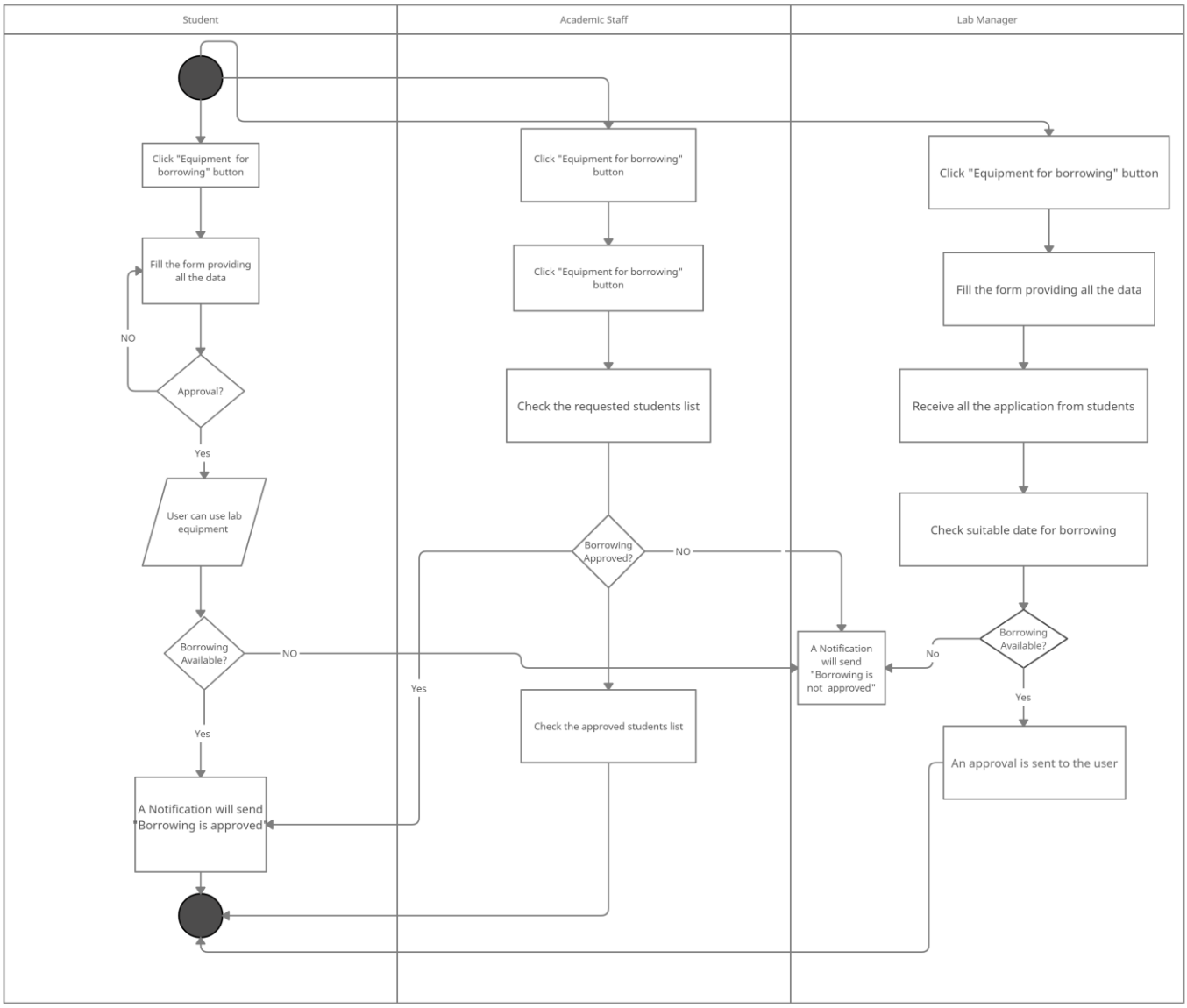
2.2.7 UC007: Manage Borrowing

Use case specification for UC007

Use Case name	Manage to borrow
Use case ID	UC007
Actors	Student, Academic Staff, Lab Manager
Pre-conditions	User needs to log into their account with their username and password.
Normal flow	1. The user clicks on the “Equipment for Borrowing” button. 2. User fills the form providing the duration of using the equipment, the purpose of using the equipment, and where the equipment will be used. 3. User waits for the approval of borrowing request. 4. Once approved, the user can borrow the lab equipment. 5. Lab Manager checks the suitable date for borrowing depending on the availability of equipment. 6. If available for borrowing, the unique id of equipment, duration, location, and purpose of using the equipment are being recorded by the Lab Manager. 7. Then borrowing is confirmed, and approval is sent to the student.
Post Conditions	Tracing records for equipment borrowing has been created in borrowing form.
Alternative Flow	While filling the form again the user can use the previous information which is already saved in the system.
Exception flow	Invalid information is entered If in step 5 in the normal flow, the Lab Manager founds the date of borrowing is not available as well as the equipment, then 1. The system will display a message to indicate to the user that either the date/equipment is wrongly entered. 2. The user will need to return to step 1 again to fill the proper Equipment borrowing form. 3. Use case end.



Sequence Diagram for Manage Borrowing

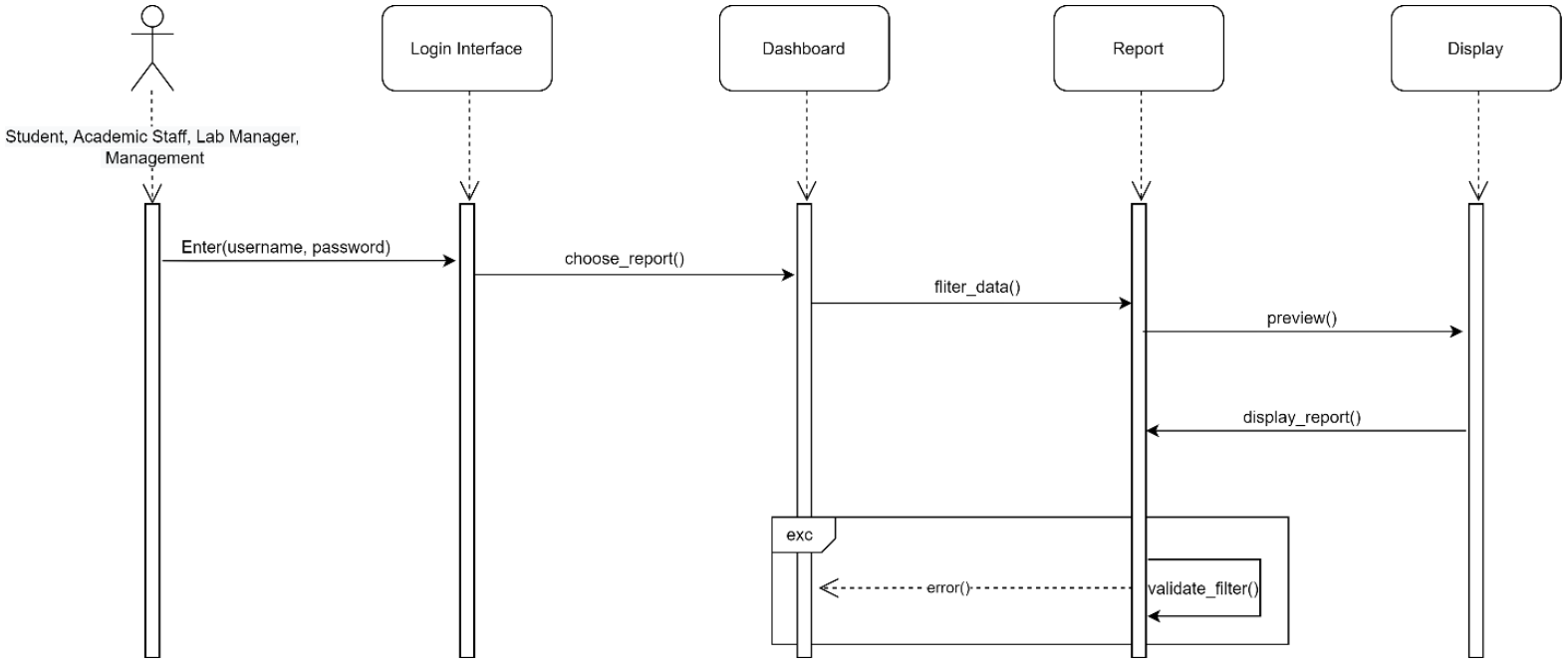


Swimlane Diagram for Manage Borrowing

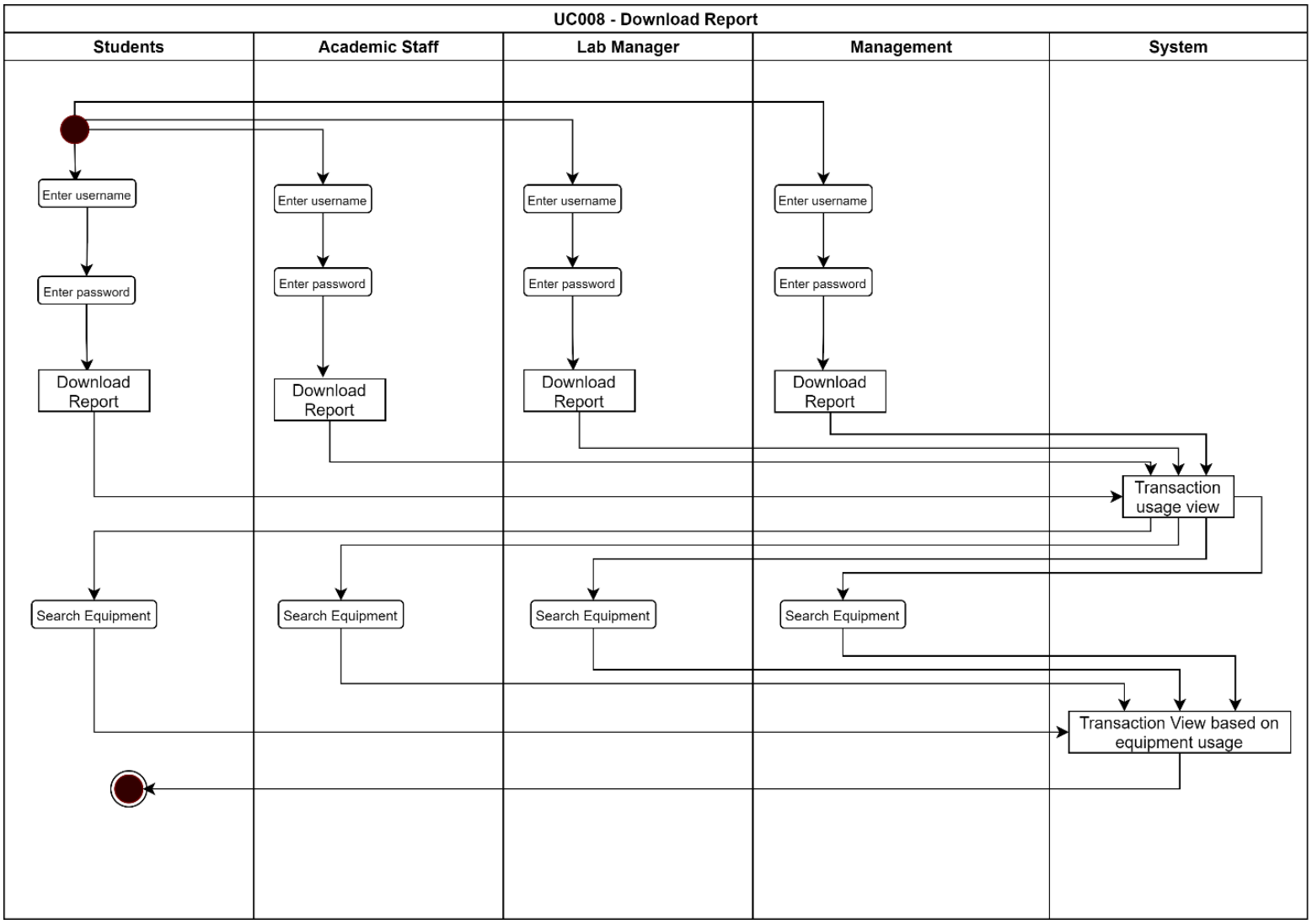
2.2.8 UC008: Download Report

Use case specification for UC008

Use Case name	View Report
Use case ID	UC008
Actors	Student, Academic Staff, Lab Manager, Management
Pre-conditions	User logged in.
Normal flow	1. The user selects the “Download Report” button. 2. User searches for the report by ID. 3. User clicks on the “Generate” button 4. If the result is shown, the user selects download report. 5. The system prompts the user that the operation was successful, and the download starts
Post Conditions	Report file ready to download
Related Documents	UC009
Exception Flow	The system may fail to download the report if any error occurs.



Sequence diagram for Download Report

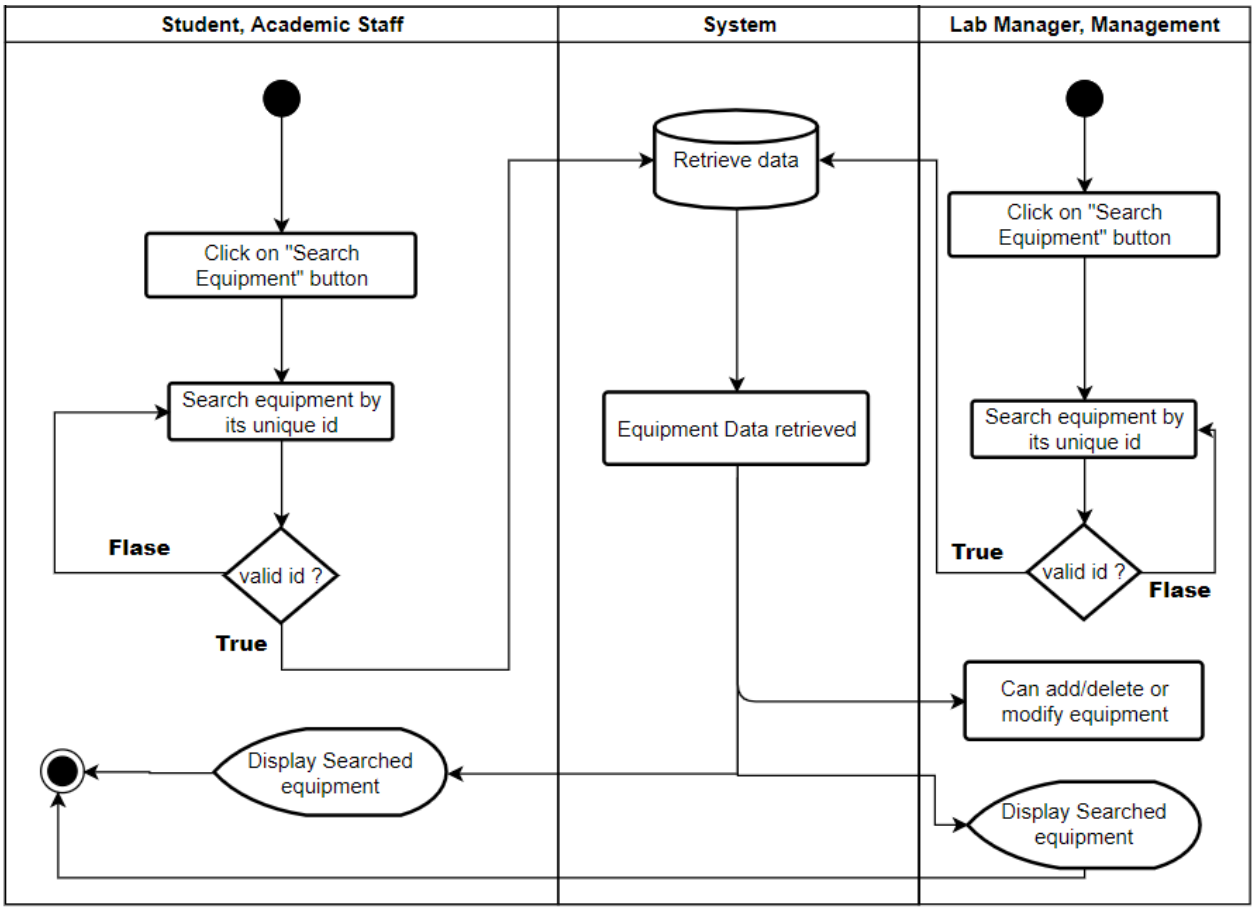


Swimlane Diagram for Download Report

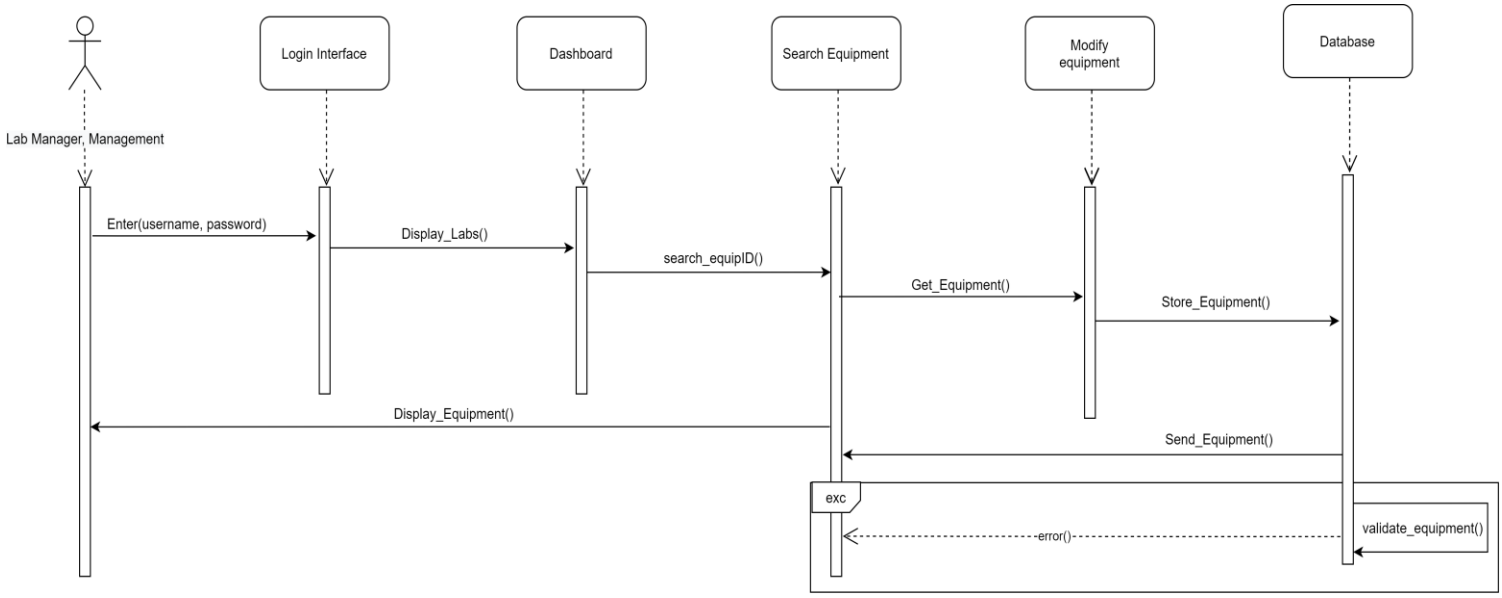
2.2.9 UC009: Search equipment

Use case specification for UC009

Use Case name	Search equipment
Use case ID	UC009
Actors	Student, Academic Staff, Lab Technician/Manager, Management
Pre-conditions	Lab Technician/Manager, management needs to be logged in and be in the lab/unit/manage page for searching equipment.
Normal flow	1. User selects the “Lab/Unit/Manage equipment” button. 2. Users can search equipment by unique equipment ID. 3. Lab Technician and Management - can add/delete the equipment under each lab and unit at any time. 4. Students and Academic staff can view the equipment report by searching specific equipment.
Post Conditions	Searched equipment is shown on the page.
Exceptional flow	The unintended path through the system usually as a result of missing information or system availability issues for searching equipment or equipment may not be available for booking/borrowing.



Swimlane diagram for Search Equipment



Sequence Diagram for Search equipment

2.3 Performance and Other Requirements

1. The system response time must be at most 8.5 to less than 5 seconds.
2. The system should be capable of supporting at least 50000 users to use the system concurrently without any error.
3. The database should be able to store limitless data and files of students, academic staff, Lab Manager, and Management.
4. The system server shall be capable of supporting an arbitrary number of active applicants' who are currently booking/borrowing equipment.
5. The system shall not disclose the user information's to any third party under any circumstances.

2.4 Design Constraints

1. For non-functional requirements, the system shall be able to authorize users via student matric number or staff id.
2. The user manual should be structured clearly and simple for example. The manual should explain to the user what the software should do and how it functions. The manual is used as a reference so that the user can understand the system easily.
3. Users should be able to access the system from any device that has an internet connection and internet browsing.
4. The information of the users is only able to be accessed by the authority that is authorized by the institution.
5. The system shall support up to 50000 users at the same time with loading time is at most 8.5 to less than 5 seconds.

The equipment booking/borrowing online system program shall be written in an object-oriented language with strong GUI links and a simple, accessible network API. The primary candidate toolchains are Java/Swing, C++/Qt, and Python/Qt. The system must provide a capacity for parallel operation and system design shall not introduce scalability issues concerning the number of devices using at any one time. The end system shall also allow for seamless recovery, without data loss from individual device failure. There must be a strong audit chain with all system actions logged. The interfaces shall be attractive but simple to use. Everything shall work as a natural flow. With that in mind, the most adaptable and portable technologies shall be used for the implementation. The system has criticality insofar as it is a live system. If the system is down, then users must not notice or notice that the system recovers quickly (in 20 milliseconds). The system must be reliable enough to run crash and glitch-free indefinitely or facilitate error recovery strong enough such that glitches are never revealed to its end-users.

2.5 Software System Attributes

1. The system must be able to make users recognize the symbol and icon used in the system, hence the system must be easy to use by having a simple but attractive interface so that the user will be able to know what to do.
2. The system must provide users with a valid id and password with no duplication so that users can access the system.
3. The information must be protected from unauthorized users to block from accessing the system.
4. The language used in the system should be English or Malay so it can be understood by all users based on the user's choice the language.
5. The system code should be written clearly and easy for students, staff, managers, and management to understand and be able to modify the system with new requirements when needed.

2.6 Other Requirements

Usability

The GUI (Graphical User Interface) shall be easy to learn and use by users of any technical background. A built-in help feature shall be available on all pages, to guide the users with the available functions on that page. Easy-to-understand documentation must be provided with the system. The system shall support several languages.

Reliability

The system shall never crash or hang, other than as the result of an operating system error. The system shall provide graceful degradation in the face of network delays and failures, or when handling large graphs.

Security and Privacy

The system, at any time, shall be accessed only by the authenticated users. The system is required to end the session automatically when an open session is not used for a specific period.

Maintainability
The document shall be easy for the users who execute the system day to day, for the system admin who wishes to edit or develop further, and for the personnel who oversees the maintenance.

3. System Architectural Design

3.1 Architecture Style and Rationale

The chosen architectural design pattern is the model view controller (MVC). The main reason that causes us to choose this model is due to the system has multiple ways to view and interact with data and the future requirements for interaction and presentation of data are unknown. MVC is an acronym for model, view, and controller and it is a product development architecture. By using MVC, we could create an application that separates the different aspects of the application. Since MVC allows the data to change independently of its representation and vice versa, it enables us to focus on one aspect of the implementation at a time. For example, we can focus on the controller (input logic) without depending on models and views (business logic and UI logic). By using MVC allows us to modify our system easily and multiple developers can work simultaneously on the model, controller, and views.

3.2 Component Model

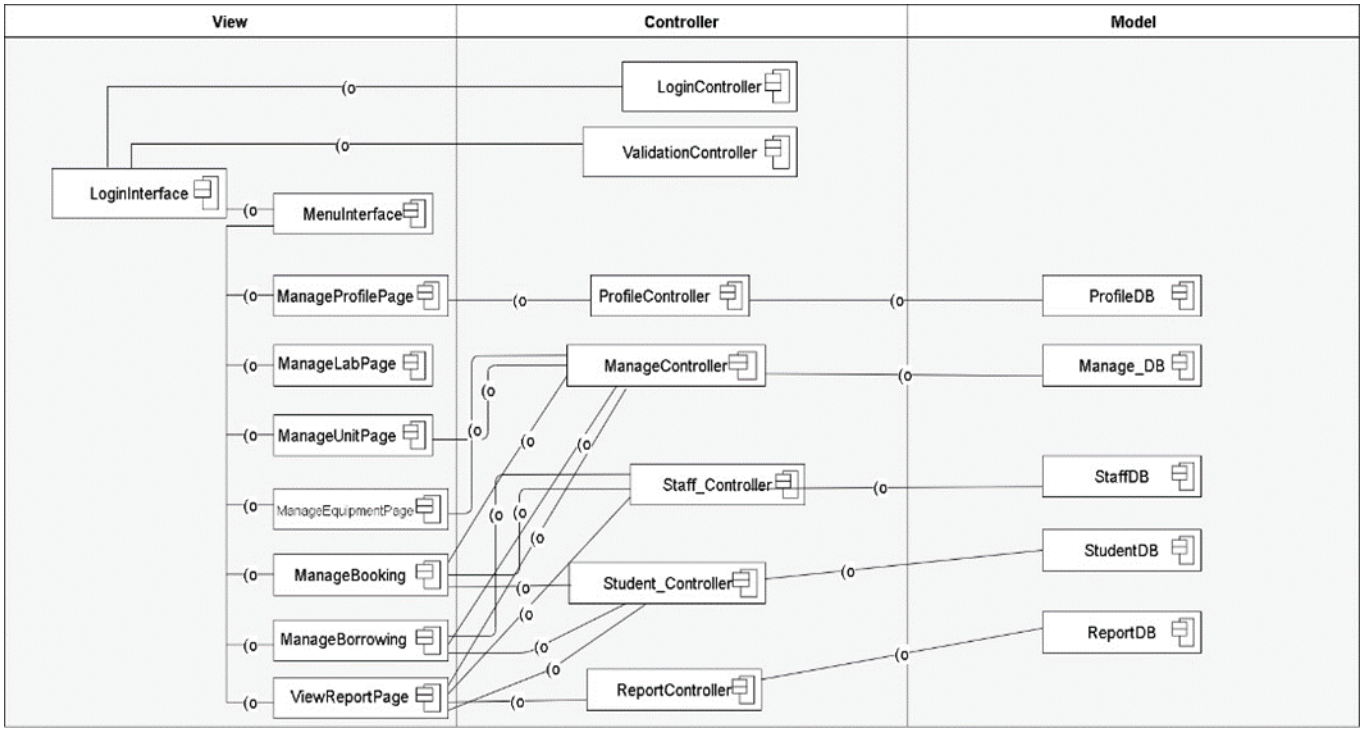


Figure 3.1: Component Diagram of Equipment booking/borrowing System

4. Detailed Description of Components

4.1 Complete Package Diagram

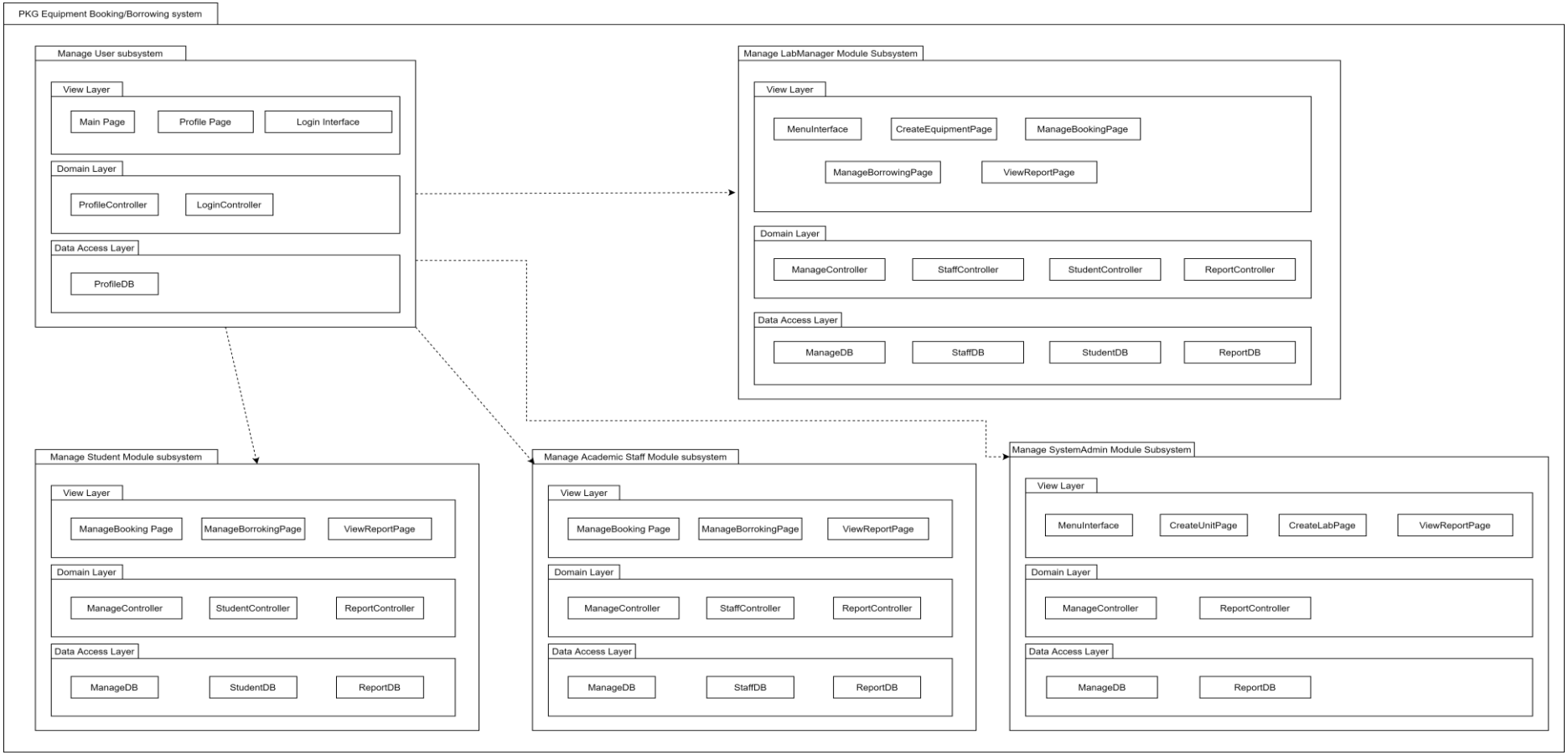


Figure 4.1: Complete Package Diagram

4.2 Detailed Description

The proposed Equipment booking/borrowing system consists of five subsystems that are managed Lab Manager Module subsystem, the manage student module subsystem, the manage Academic Staff module subsystem, and the system admin subsystem. Every subsystem consists of the view layer, domain layer, and data access layer. The view layer is a layer that allows the user to interact with the system. The domain layer includes controller classes that are related to the main function and logic of the subsystem. The data access layer is a layer that accessing the data from the database.

In Manage Lab Manager Module Subsystem (P001), involves UC005 Manage Equipment, UC006 Manage Booking/Usage, UC007 Manage Borrowing, and UC008 Download Report. This subsystem provides functions for the Lab Manager and the System Admin. In this subsystem, it has a MenuInterface that interacts with CreateLabPage, CreateUnitPage, CreateEquipmentPage, ManageBookingPage, ManageBorrowingPage, and ViewReportPage. When a user chooses a function, he/she wishes in the MenuInterface, the related page will be displayed. In the domain layer, the subsystem consists of ManageController, Staff_Controller, Student_Controller, and ReportController. Each of the controllers related to its perspective entity in the data access layer. For example, if the user wishes to get the report, the ReportController will handler this task and retrieve the data from the ReportDB.

4.2.2 P001: Manage Lab Manager and System Admin Module Subsystem

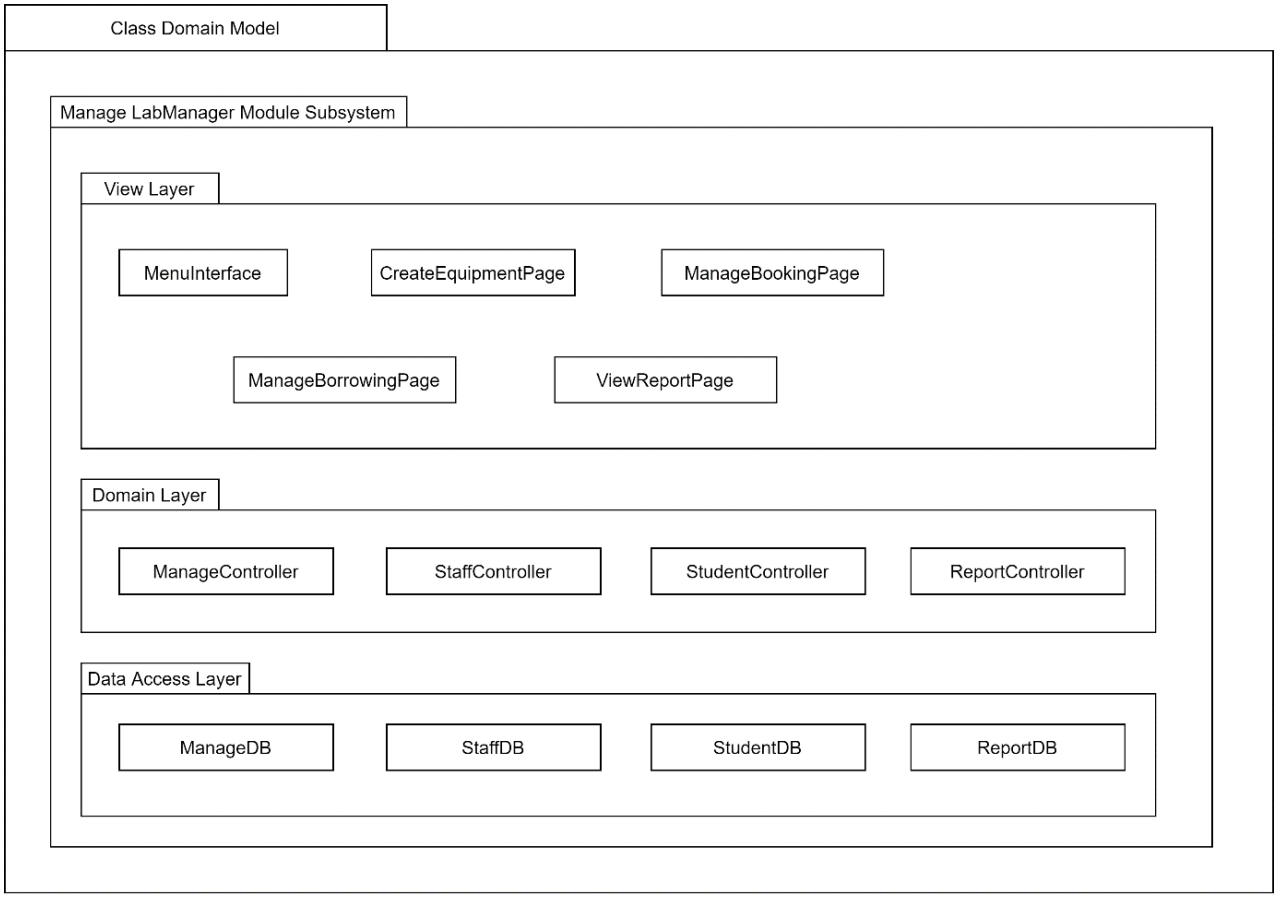


Figure 4.2: Package Diagram for Managing Lab Manager Module Subsystem

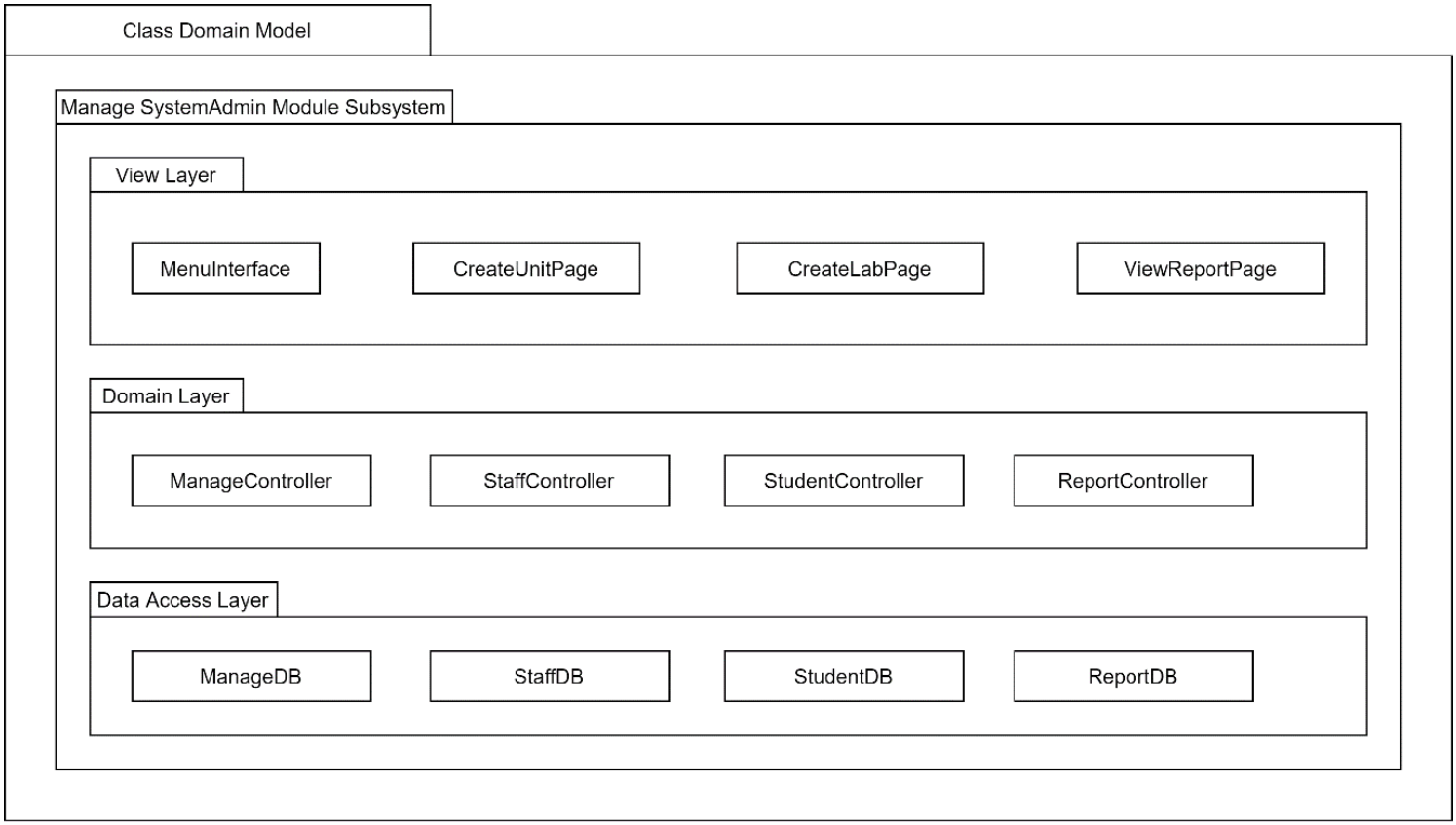
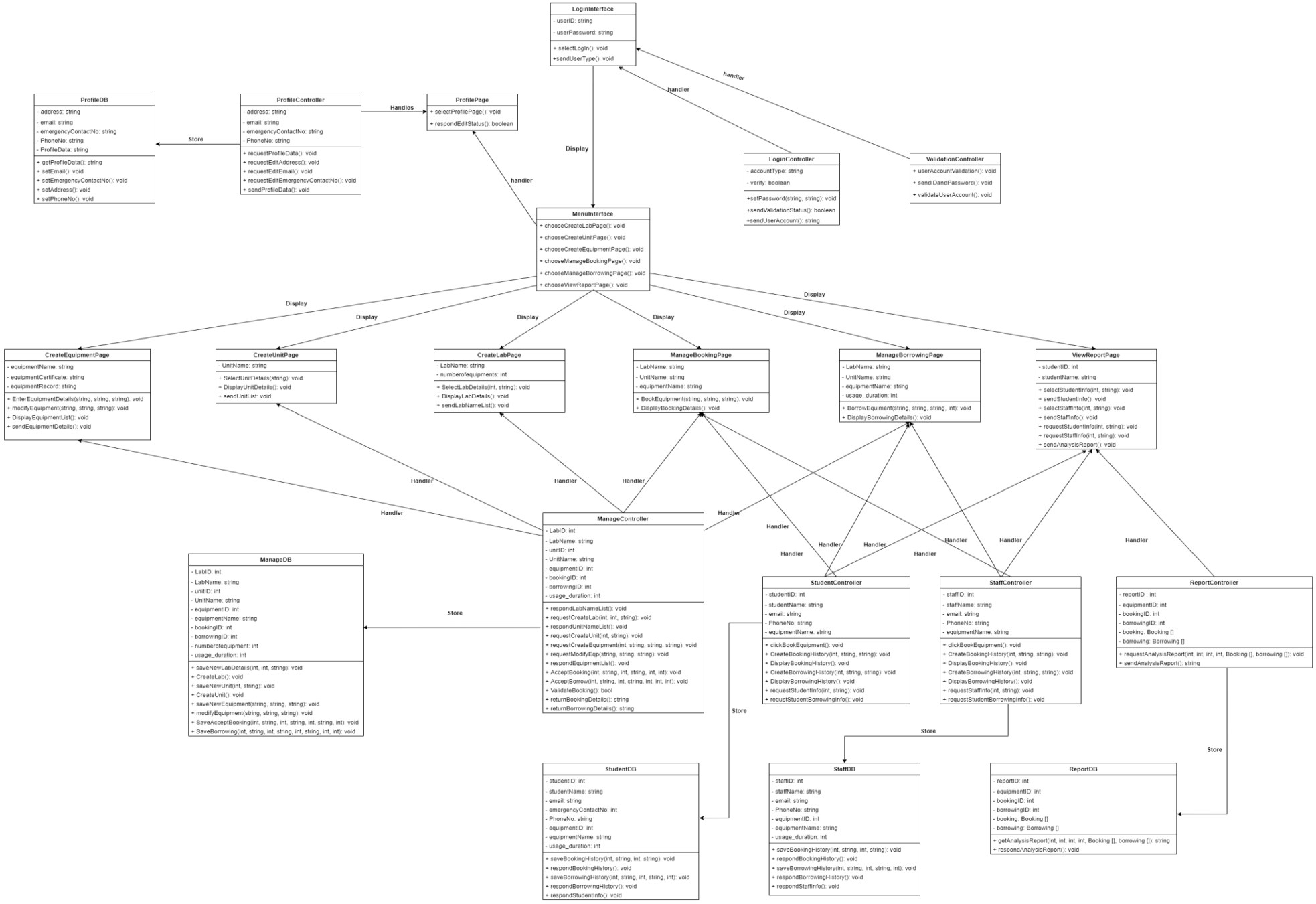


Figure 4.3: Package Diagram for Managing System Admin Module Subsystem

3.2.2.1 Class Diagram (Overall)



Entity Name	Create Unit
Method Name	selectUnitDetails(), requestCreateUnit(), SaveNewUnit()
Input	Unit Details
Output	Display Successful message that a new unit is created
Algorithm	1. Choose to create the unit page from the menu interface. 2. Selecting unit details calls the selectUnitDetails() method and choose details from the dropdown list. 3. Fill in unit details and click on create unit button to create a new unit. Then it will call the method requestCreateUnit() to create a new unit and save into the database by calling the method SaveNewUnit().

4.2.1.1 Sequence Diagram

a) SD001: Sequence diagram for Create New Unit

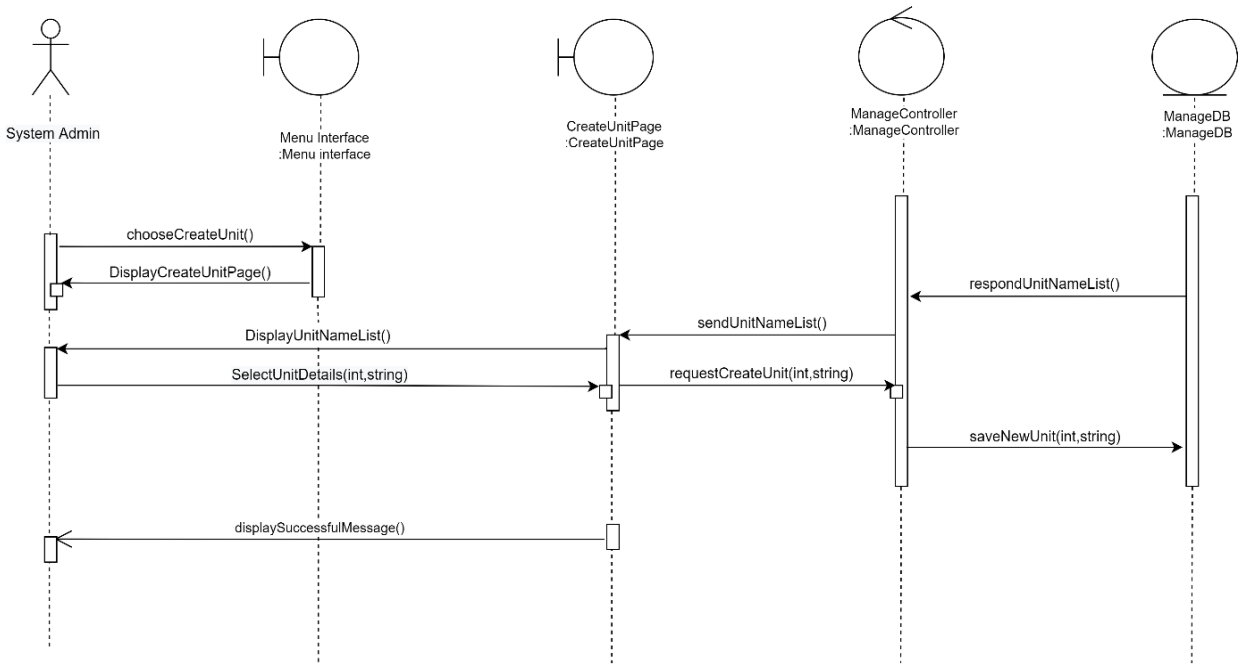


Figure 4.4: Sequence Diagram for Create New Unit

Entity Name	Create Lab
Method Name	selectLabDetails(), requestCreateLab(), SaveNewLab()
Input	Lab Details
Output	Display Successful message that a new lab is created
Algorithm	1. Choose to create a lab page from the menu interface. 2. Selecting lab details calling the method name selectLabDetails() and choose details from the dropdown list. 3. Fill up lab details and click on create lab button to create a new lab by calling the method name requestCreateLab() and a new Lab details are stored in database by calling the method SaveNewLab().

b) SD002: Sequence diagram for create new Lab

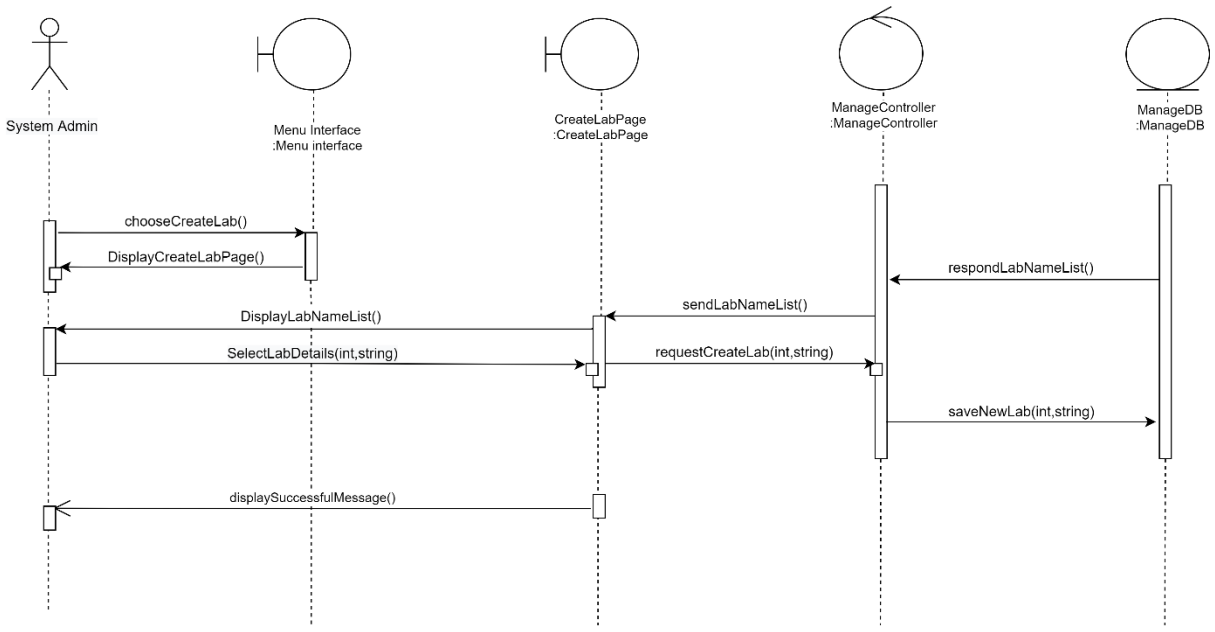


Figure 4.5: Sequence Diagram for Create new Lab

Entity Name	Create Equipment
Method Name	✓ EnterEquipmentDetails() ✓ ModifyEquipment() ✓ requestModifyEquipment() ✓ requestCreateEquipment() ✓ modifyEquipment() ✓ saveNewEquipment()
Input	• Name of equipment. • Equipment Certificate and record of when it is added into the system. • To modify the existing equipment or to delete, rename all the previous details entered or remove the equipment.
Output	Display Successful message that new equipment is created/modified.
Algorithm	1. To create new equipment, enter all the details of the new equipment and click on the submit button by calling the method EnterEquipmentDetails() to add new equipment. Then it will call the method requestCreateEquipment() to create a new equipment. Then it will call the SaveNewEquipment() to save the equipment into database. 2. To modify existing equipment, click on the modify button by calling the method ModifyEquipment() to rename all the previous details and click on the submit button to modify the equipment by calling the method requestModifyEquipment(). This will modify the equipment into the database by calling the method modifyEquipment(). 3. Display a successful message when new equipment is added/modified.

c) SD003: Sequence diagram for Add/Modify Equipment

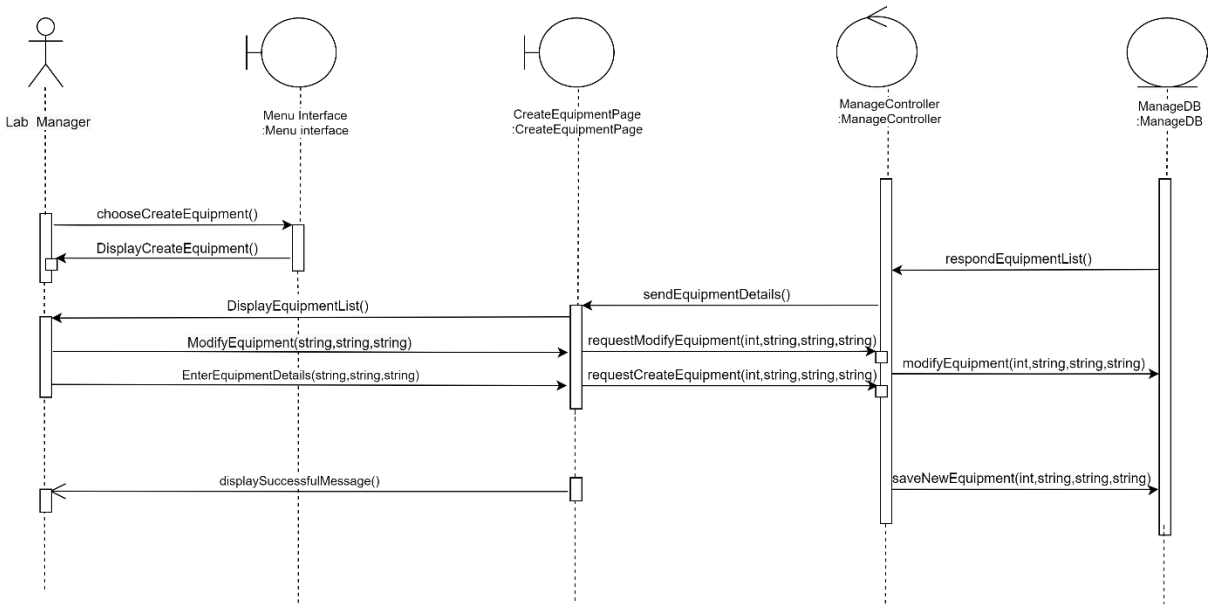


Figure 4.6: Sequence Diagram for Add/Modify Equipment

Entity Name	Manage Booking
Method Name	✓ BookEquipment() ✓ AcceptBooking() ✓ ValidateBooking() ✓ CreateBooking() ✓ saveBookingHistory()
Input	Select equipment to book.
Output	Display the booking details.
Algorithm	1. Choose equipment to book. 2. Click on the book button. 3. After booking, the system will call the method called AcceptBooking(), after accepting it will validate the booking, if available for booking, it will call the method name called CreateBooking(), after create booking, it will save into the booking history by calling the method name called saveBookingHistory()

	then it will show the output of the booking details. If not available, then it will show the booking details that not available for booking.

d) i) SD004: Sequence diagram for Manage Booking – Student, and Academic Staff

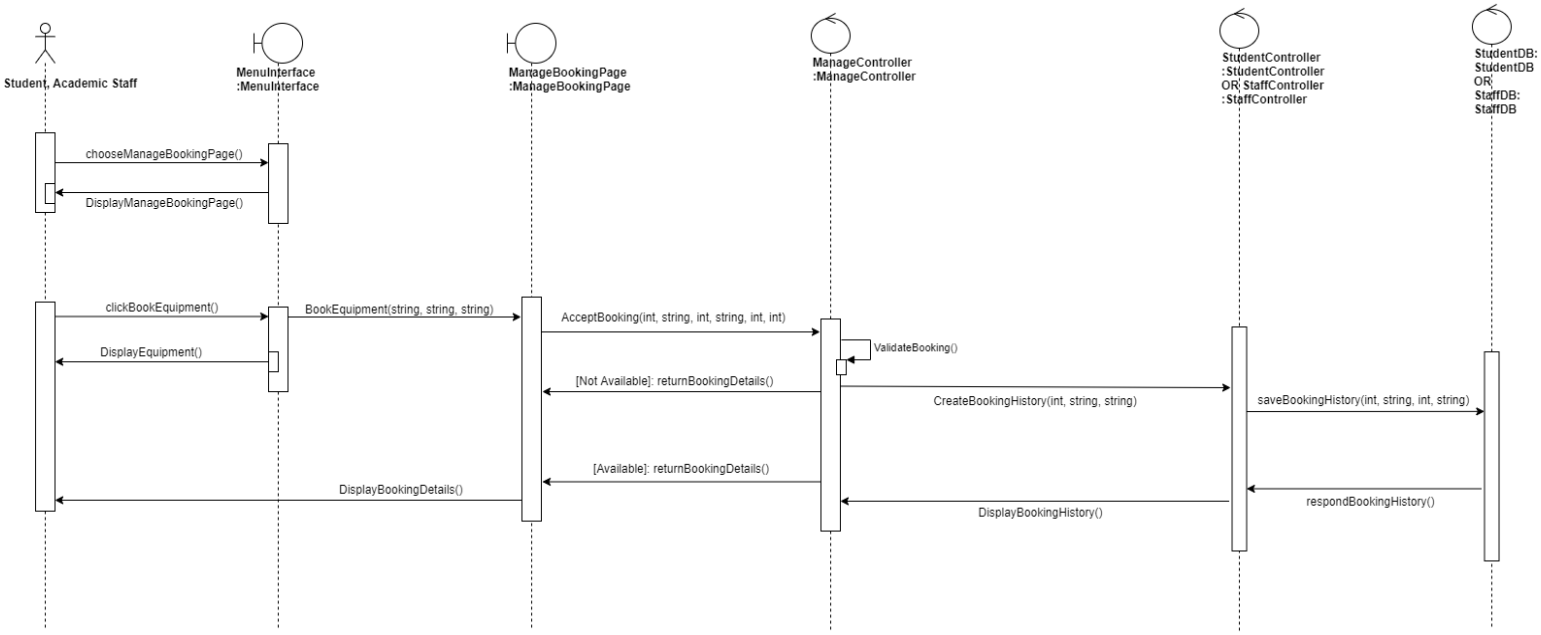


Figure 4.7: Sequence Diagram for Manage Booking for Student and Academic Staff

Entity Name	Manage Booking
Method Name	✓ BookEquipment() ✓ AcceptBooking() ✓ saveAcceptBooking()
Input	Click on the accept button to accept booking.
Output	Display Successful message that the booking is confirmed.

Algorithm	1. Click on the confirm button to accept booking for the pending requests. 2. After accepting booking, it will call the method to AcceptBooking() to create booking. 3. After calling the AcceptBooking() method, it will save the booking history into the database through the method saveAcceptBooking().

d) ii) SD005: Sequence diagram for Manage Booking – Lab Manager

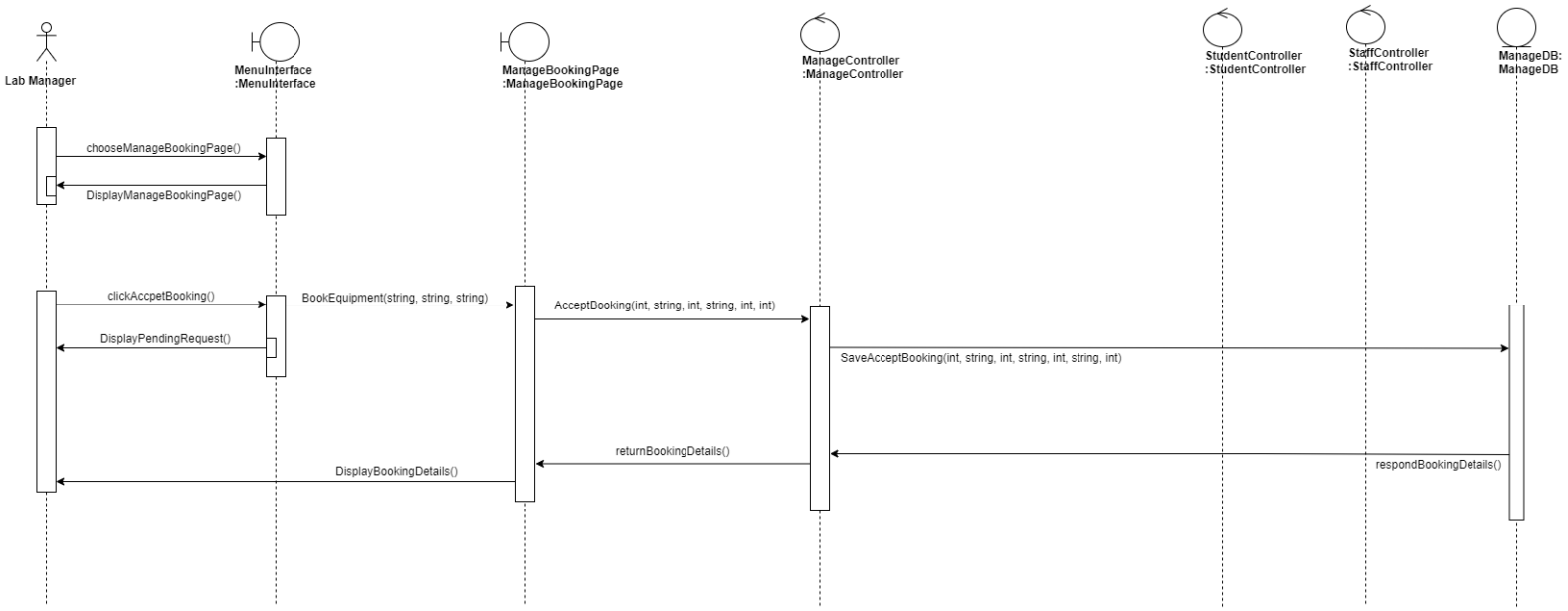


Figure 4.8: Sequence Diagram for Manage Booking for Lab Manager

Entity Name	Manage Borrowing
Method Name	✓ BookEquipment() ✓ AcceptBorrow() ✓ ValidateBooking() ✓ CreateBorrowingHistory() ✓ saveBorrowingHistory()
Input	• Select equipment to book. • Duration Usage • Location/Address
Output	Display a Successful message that the equipment is available to book and then wait for confirmation by the lab manager.
Algorithm	1. Choose equipment to book. 2. Fill up the usage duration and location of using the equipment. 3. Fill up the purpose of using the equipment. 4. Click on the book button. 5. After borrowing, the system will call the method called AcceptBorrow(), after accepting it will validate the booking, if available for booking, it will call the method name called CreateBorrowingHistory(), after create booking, it will save into the booking history by calling the method name called saveBorrowingHistory() then it will show the output of the borrowing details. If not available, then it will show the borrowing details that not available for borrowing.

e) i) SD006: Sequence diagram for Borrow Equipment – Student and Academic Staff

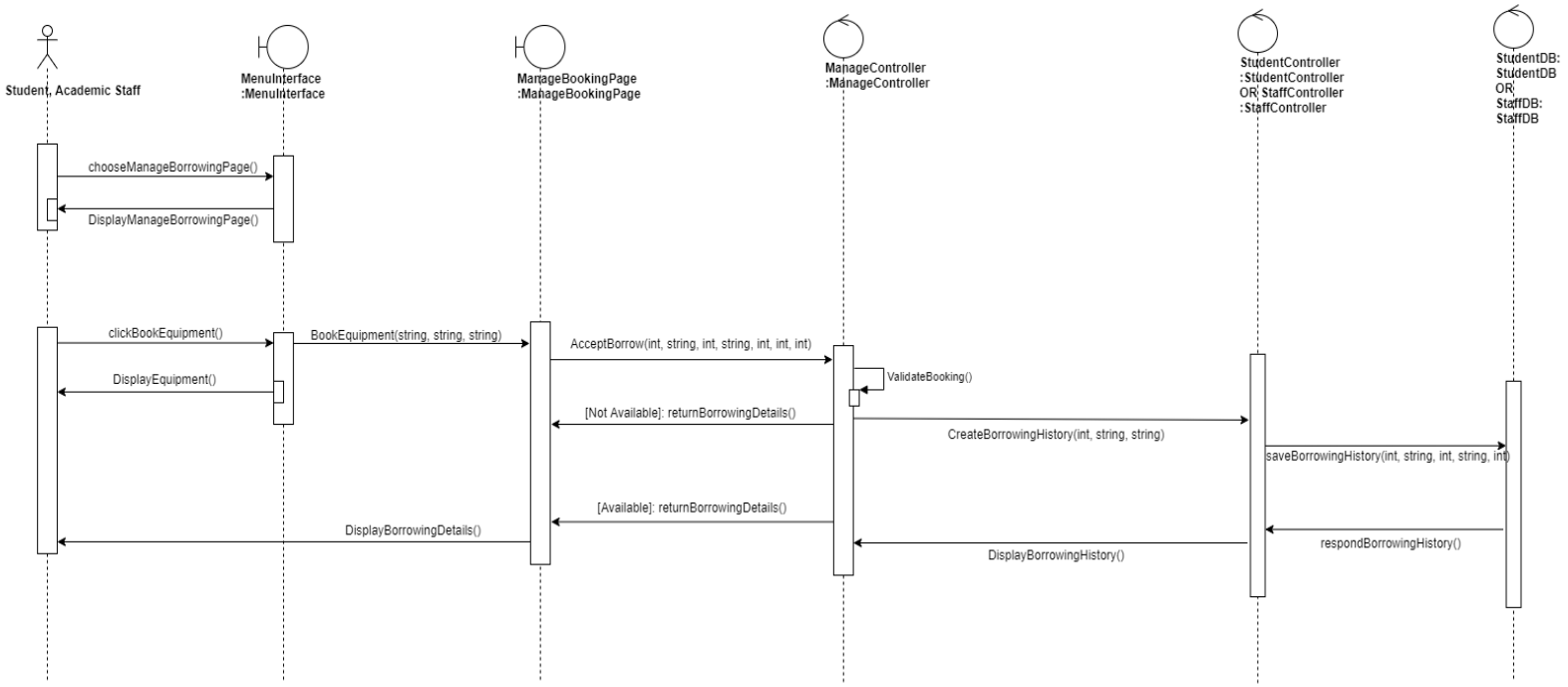


Figure 4.9: Sequence Diagram for Borrowing Equipment for Student and Academic Staff

Entity Name	Manage Borrowing
Method Name	✓ BookEquipment() ✓ AcceptBorrow() ✓ SaveBorrowing()
Input	Click on the confirm button to accept borrowing.
Output	Display Successful message that the equipment is confirmed for borrowing and display the borrowing details that have been confirmed.
Algorithm	1. Click on the Borrowing page. 2. After accepting booking, it will call the method to AcceptBorrow() to create booking. 3. After calling the AcceptBorrowing() method, it will save the borrowing history into the database through the method SaveBorrowing().

e) ii) SD007: Sequence diagram for Borrow Equipment – Lab Manager

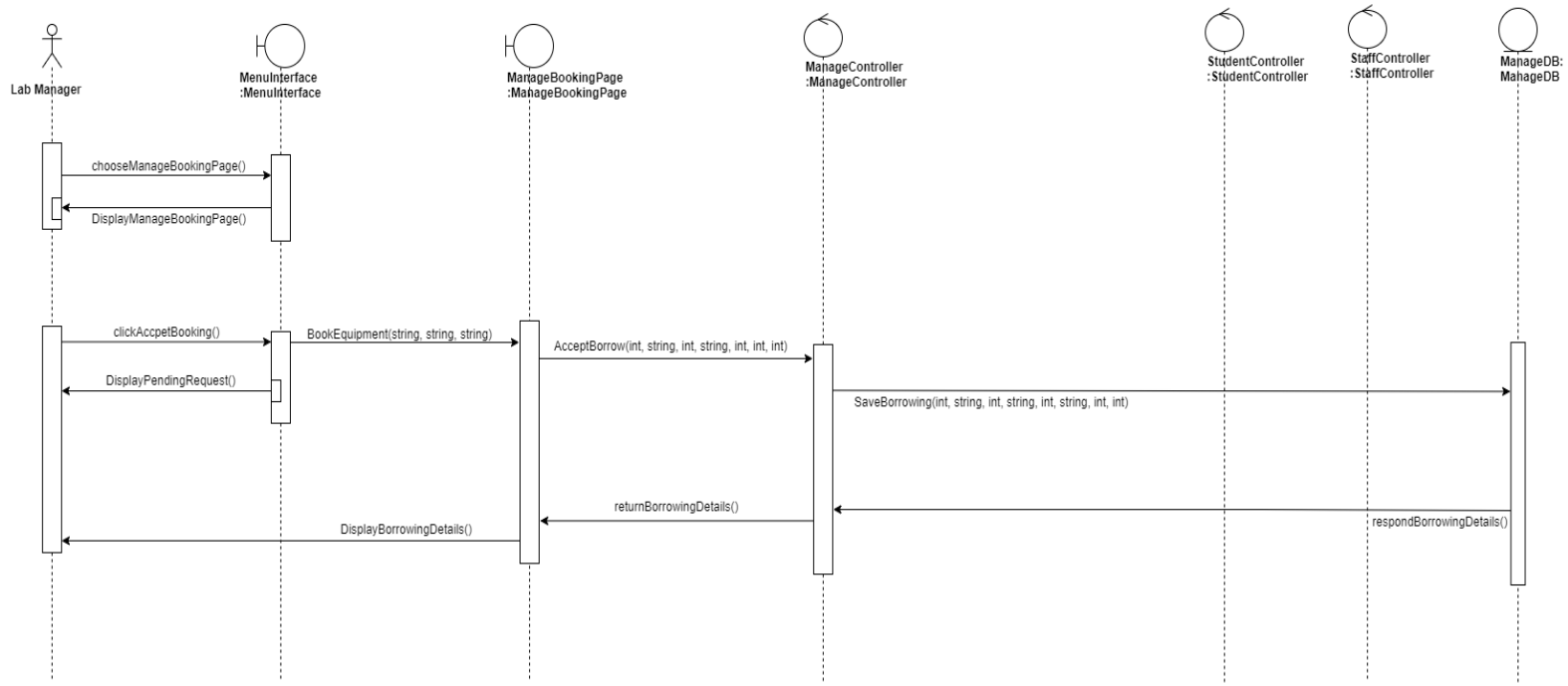


Figure 5.0: Sequence Diagram for Borrowing Equipment for Lab Manager

Entity Name	View Report
Method Name	selectStudentInfo() requestStudentInfo() getStudentInfo() selectStaffInfo() requestStaffInfo() getStaffInfo() selectStudent() requestAnalysisReport() getAnalysisReport() displayStudentReport() displayStaffReport()
Input	• Select Student or Staff. • Select Booking/Borrowing. • Select the name of the student or staff.
Output	Display the generated report for specific student or staff
Algorithm	1. Select user type by calling the method selectStudentInfo() or selectStaffInfo(). 2. Select booking or borrowing. 3. Select a user name from the dropdown list. 4. The system shall call the method requestAnalysisReport() to produce the report for specific user and generate the report by calling the method getAnalysisReport() and then it is displayed.

f) SD008: Sequence diagram for View Report

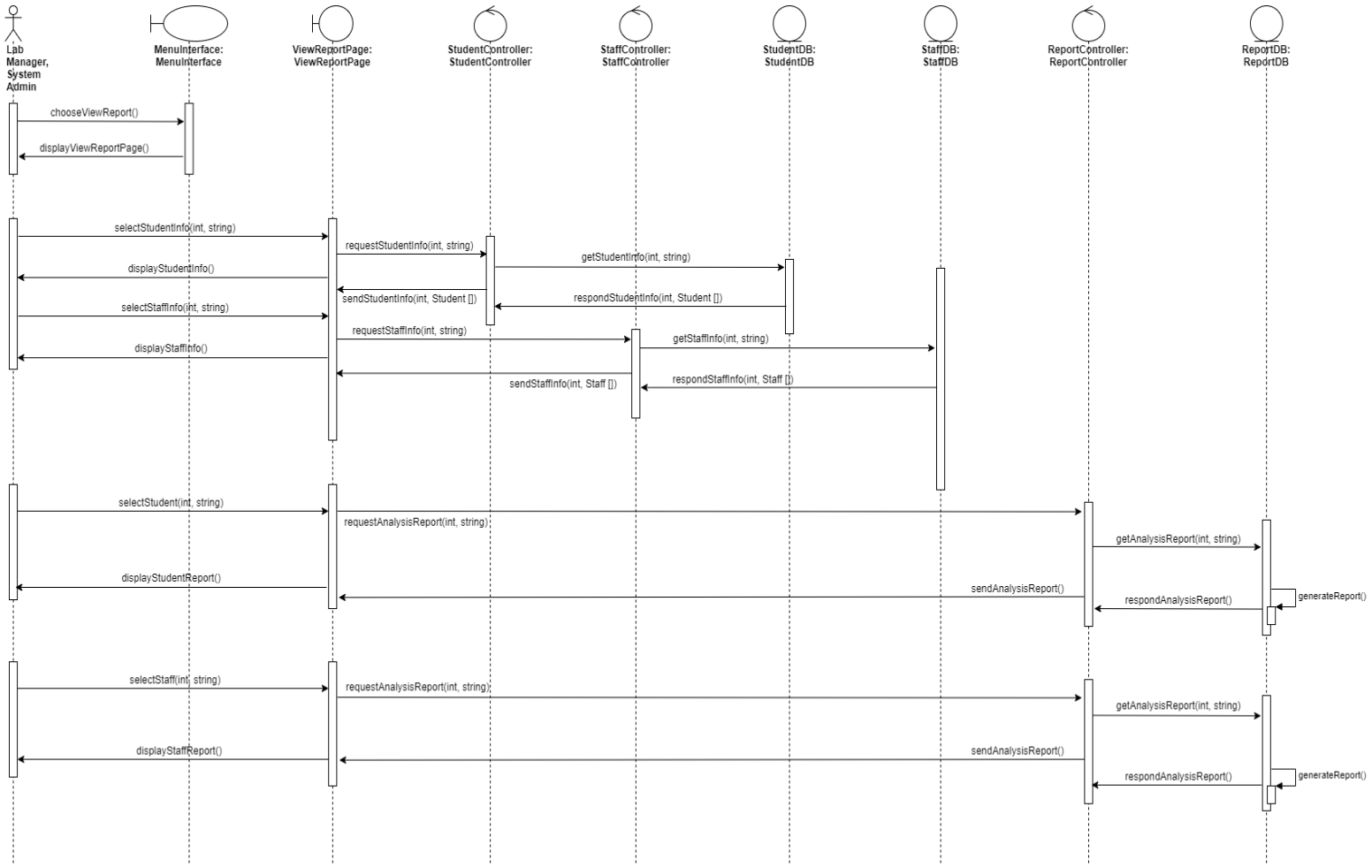


Figure 5.10: Sequence Diagram for View Report

Data Design

4.3 Data Description

The major data or systems entities are stored into a relational database named as Equipment booking/borrowing database and comprise 4 entities (for manage lab manager module subsystem). The data retrieved or stored by the system through the controllers from the entities or to the entities. The data is organized into attributes with recognizable names. Each of the entities has its contents and all of these entities have a different unique name. Each of these unique names enables the system to search and retrieve the data correctly when needed. Each of the entities consists of its attributes.

Table 5.1: Description of Entities in the Database

No.	Entity Name	Description
1.	ManageDB	This entity contains the ID of units, labs, and equipment. Whenever a new unit and lab are created, or when equipment is added or modified, or when a new booking/borrowing is made are stored in this entity.
2.	StudentDB	This entity contains the student's bookings and borrowings that are made are stored in this entity.
3.	StaffDB	This entity contains the staff's bookings and borrowings that are made are stored in this entity.
4.	ReportDB	This entity contains the number of bookings or borrowings done by the student or staff are stored in this entity.

4.4 Data Dictionary

5.2.1 Entity: ManageDB

Attribute Name	Constraints and Tags
LabID **int private**	*default:*
LabName **string private**	*default:*
UnitId **int private**	*default:*
UnitName **string private**	*default:*
equipmentID **int private**	*default:*
equipmentName **string private**	*default:*
bookingID **int private**	*default:*
borrowingID **int private**	*default:*

numberofquipment **int private**	*default:*
usage_duration **int private**	*default:*

Operations

Methods	**Notes**	**Parameters**
saveNewUnit () void public	SD001 Create Unit	**int** [in] UnitID **string** [in] LabName
CreateUnit() void public	SD001 Create Unit	
saveNewLabDetails() void public	SD002 Create Lab	**int** [in] LabID **int** [in] equipmentID **string** [in] LabName
CreateLab() void public	SD002 Create Lab	
saveNewEquipment() void public	SD003 Add/Modify Equipment	**Int** [in] equipmentID **string** [in] equipmentName
modifyEquipment() void public	SD003 Add/Modify Equipment	**int** [in] UnitID **string** [in] LabName
SaveAcceptBooking() void public	SD004 Manage Booking SD005 Manage Booking	**int** [in] UnitID **string** [in] UnitNameName **int** [in] LabID **string** [in] LabName **int** [in] equipmentID **string** [in] equipmentName **int** [in] bookingID
SaveAcceptBorrowing() void public	SD006 Manage Borrowing SD007 Manage Borrowing	**int** [in] UnitID **string** [in] UnitNameName **int** [in] LabID **string** [in] LabName **int** [in] equipmentID **string** [in] equipmentName **int** [in] borrowingID **int** [in] usage_duration

5.2.2 Entity: StudentDB

Attribute Name	Constraints and Tags
studentID **int private**	*default:*
studentName **string private**	*default:*
Email **string private**	*default:*
emergencyContactNo **int private**	*default:*
PhoneNo **int private**	*default:*
equipmentID **int private**	*default:*
equipmentName **string private**	*default:*
usage_duration **int private**	*default:*

Operations

Methods	Notes	Parameters
saveBookingHistory () void public	SD004 Manage Booking SD005 manage Booking	**int** [in] studentID **string** [in] studentName **int** [in] equipmentID **string** [in] equipmentName
saveBorrowingHistory() void public	SD006 Manage Borrowing SD007 Manage Borrowing	**int** [in] studentID **string** [in] studentName **int** [in] equipmentID **string** [in] equipmentName **int** [in] usage_duration
respondStudentInfo() void public	SD008 View Report	**int** [in] studentID **string** [in] studentName

5.2.3 Entity: StaffDB

Attribute Name	Constraints and Tags
staffID **int private**	*default:*
staffName **string private**	*default:*
Email **string private**	*default:*
PhoneNo **int private**	*default:*
equipmentID **int private**	*default:*
equipmentName **string private**	*default:*
usage_duration **int private**	*default:*

Operations

Methods	Notes	Parameters
saveBookingHistory () void public	SD004 Manage Booking SD005 Manage Booking	**int** [in] staffID **string** [in] staffName **int** [in] equipmentID **string** [in] equipmentName
saveBorrowingHistory() void public	SD006 Manage Borrowing SD007 Manage Borrowing	**int** [in] staffID **string** [in] staffName **int** [in] equipmentID **string** [in] equipmentName **int** [in] usage_duration
respondStaffInfo() void public	SD008 View Report	**int** [in] staffID **string** [in] staffName

5.2.4 Entity: ReportDB

Attribute Name	Constraints and Tags
reportID **int private**	*default:*
equipmentID **int private**	*default:*
bookingID **int private**	*default:*
borrowing **int private**	*default:*
equipmentID **int private**	*default:*
booking Booking[] **private**	*default:*
borrowing Borrowing[] **private**	*default:*

Operations

Methods	Notes	Parameters
getAnalysisReport () void public	SD008 View Report	**int** [in] reportID **int** [in] equipmentID **int** [in] bookingID **int** [in] borrowingID Booking[] in booking Borrowing[] in borrowing
respondAnalysisReport()	SD008 View Report	

5. User Interface Design

5.1 Overview of User Interface

To manage Lab manager module subsystem as well as the System Admin module subsystem, after the lab manager and system admin log into their account, the system will display a menu that only shows functions that are relevant to the lab manager and system admin. For the lab manager, the menu will show five functions such as Add/Modify Equipment, Manage Booking, Manage Borrowing, and view student and academic staff booking/borrowing report. For System Admin, the menu will show three functions as Create Unit, and Create Lab. For Create Unit and Lab, the system will create a new unit and lab for the user to book. To create equipment, the system will create new equipment or modify the existing equipment for each lab. To manage booking/borrowing, the user will book/borrow the equipment and the lab manager need to accept the booking/borrowing from the “Manage Booking” and “Manage Borrowing” pages. Lastly, the user, as well as the lab manager and the system admin, may view each student or academic staff booking/borrowing usage report.

5.2 Screen Images

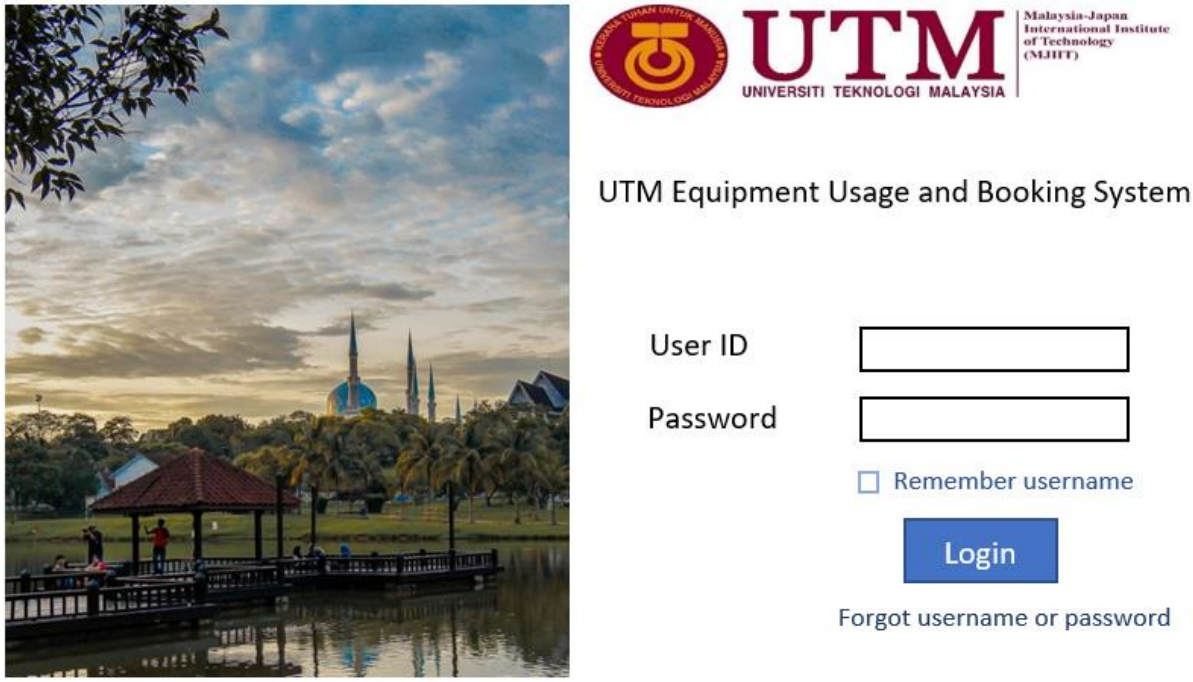


Figure 6.1: Interface for <Login>

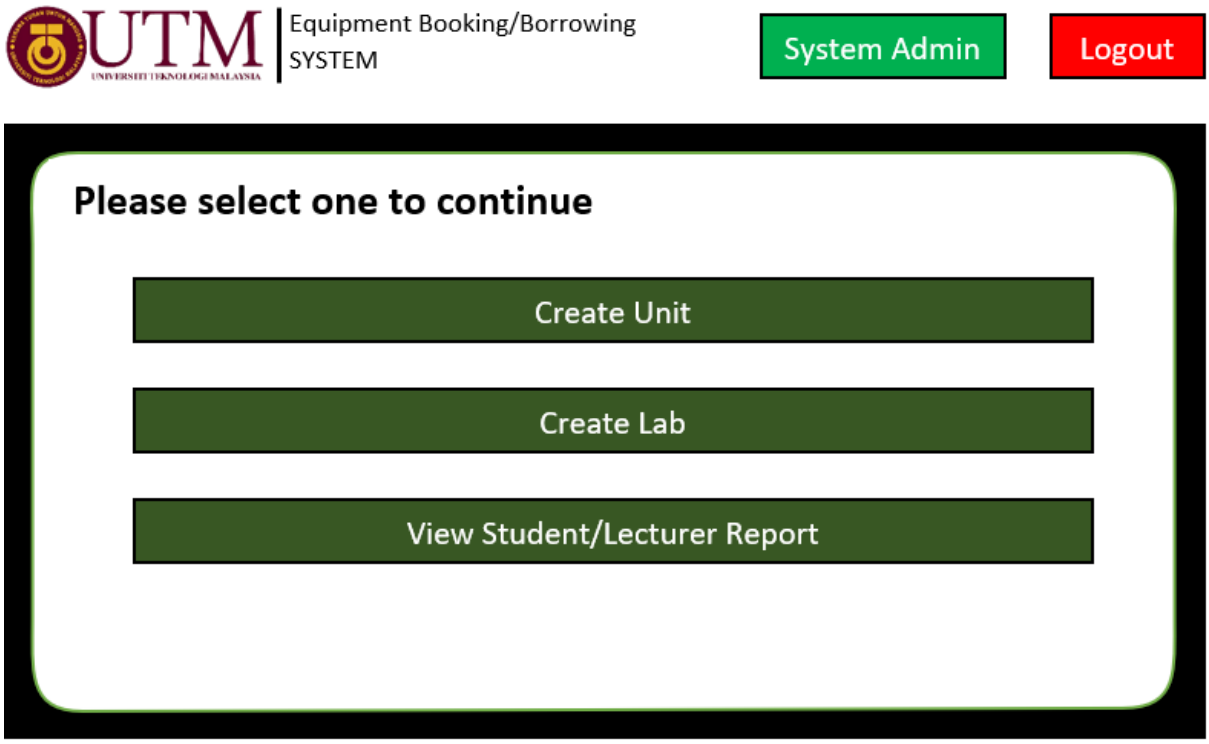


Figure 6.2: Interface for <System Admin Homepage>

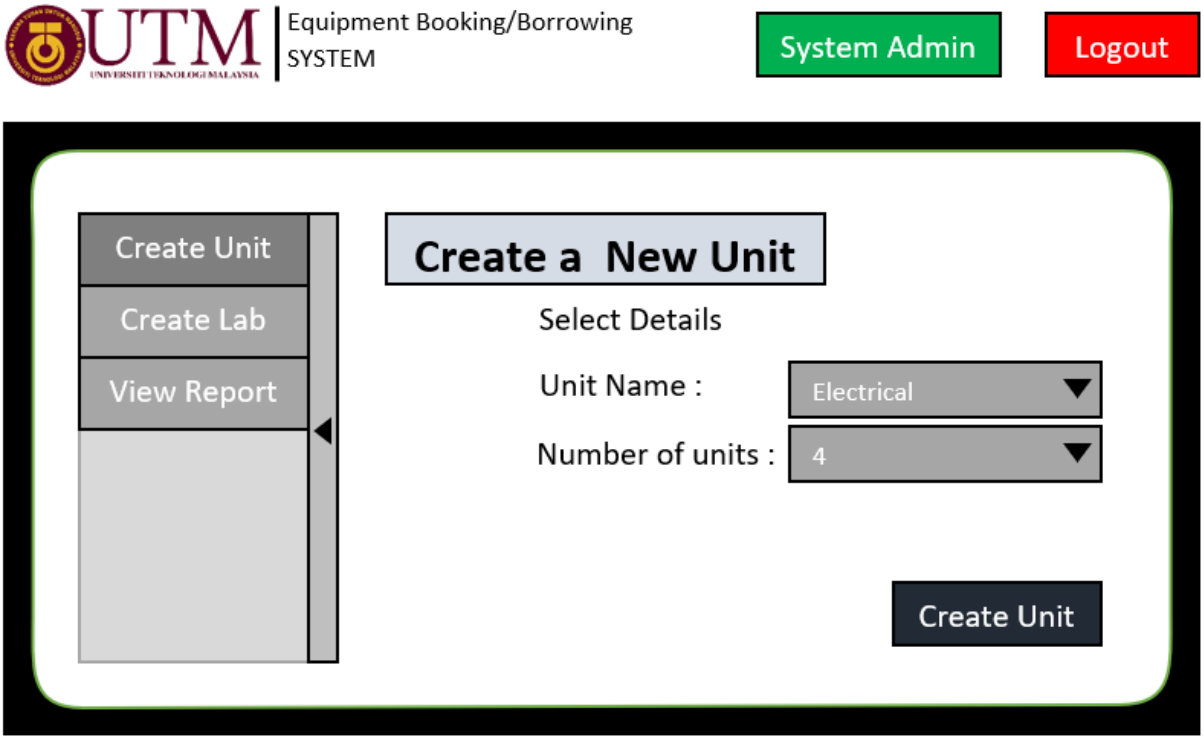


Figure 6.3: Interface for <Create Unit>

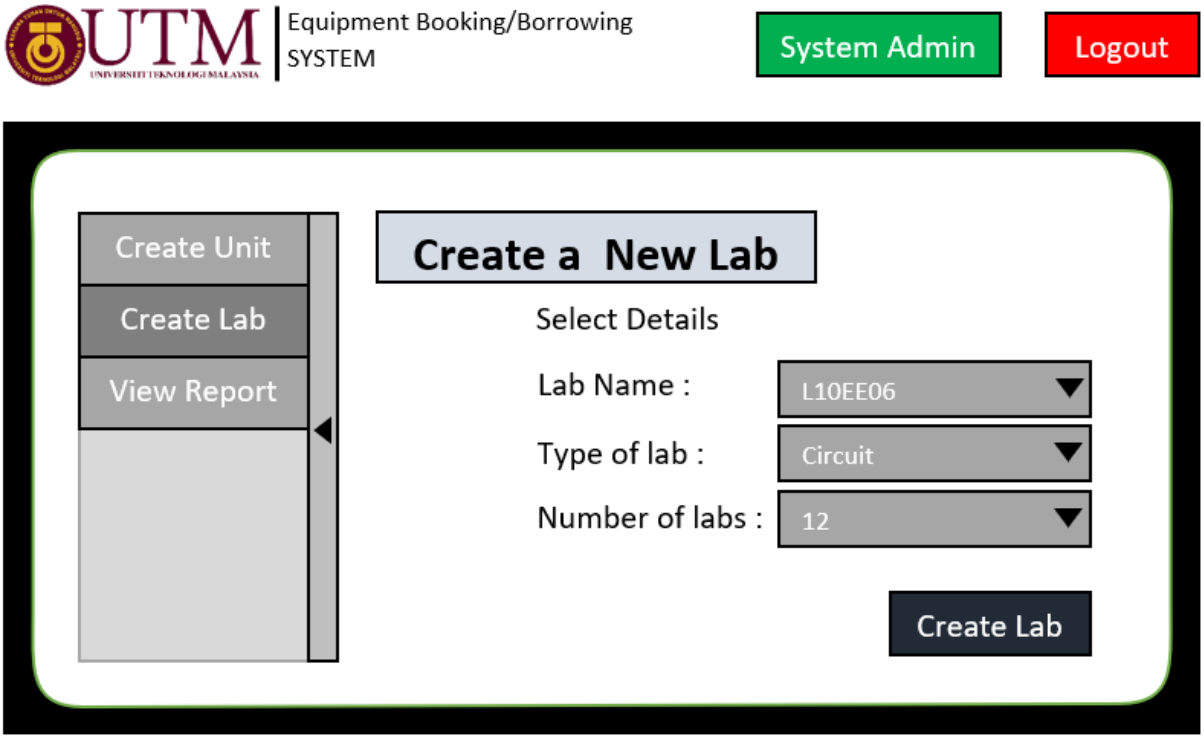


Figure 6.4: Interface for <Create Lab>

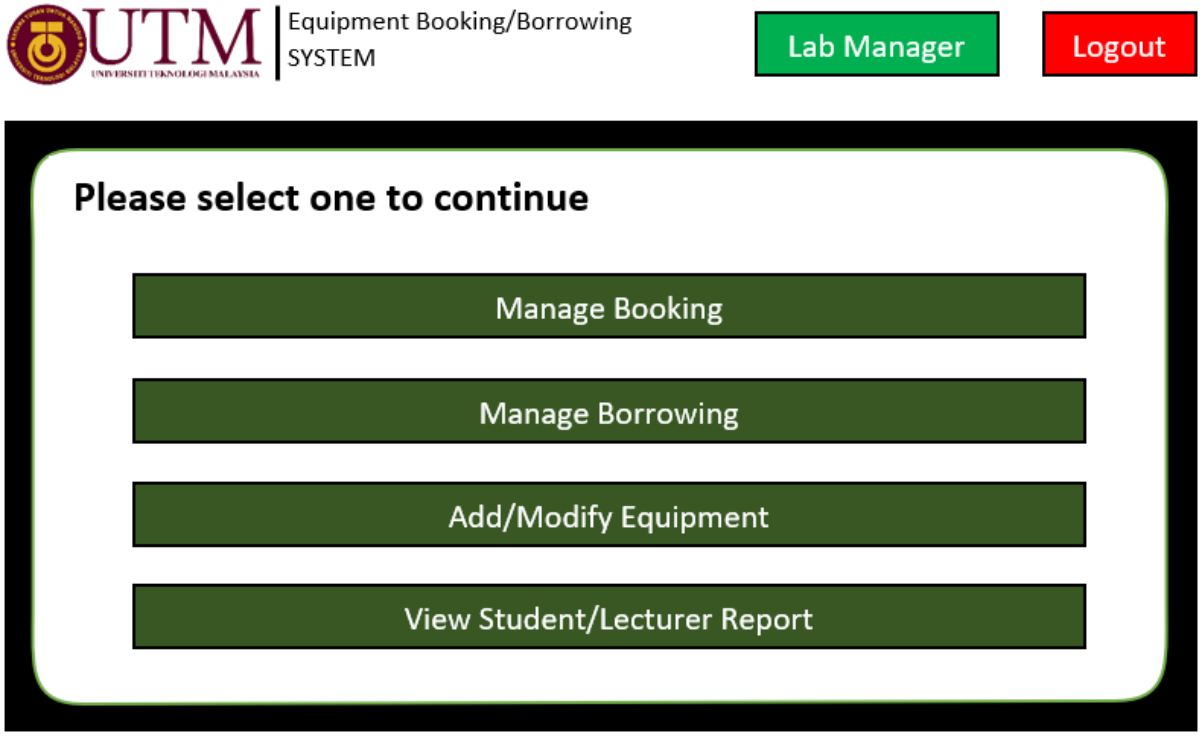


Figure 6.5: Interface for <Lab Manager Homepage>

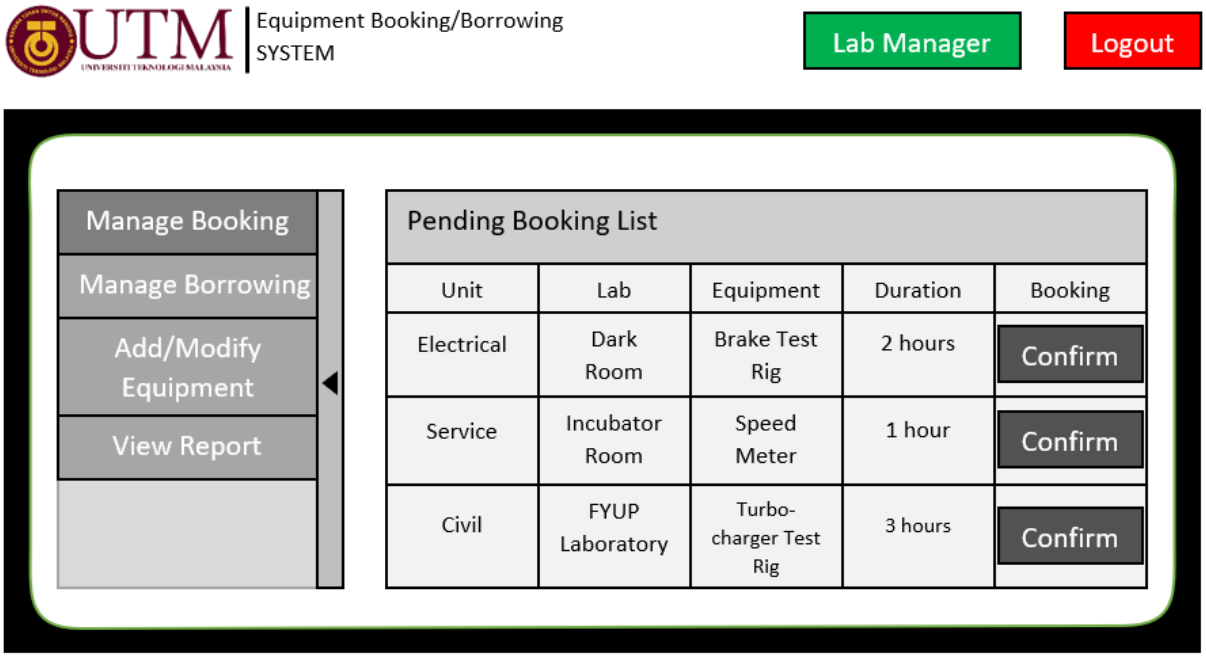


Figure 6.6: Interface for <Manage Booking>

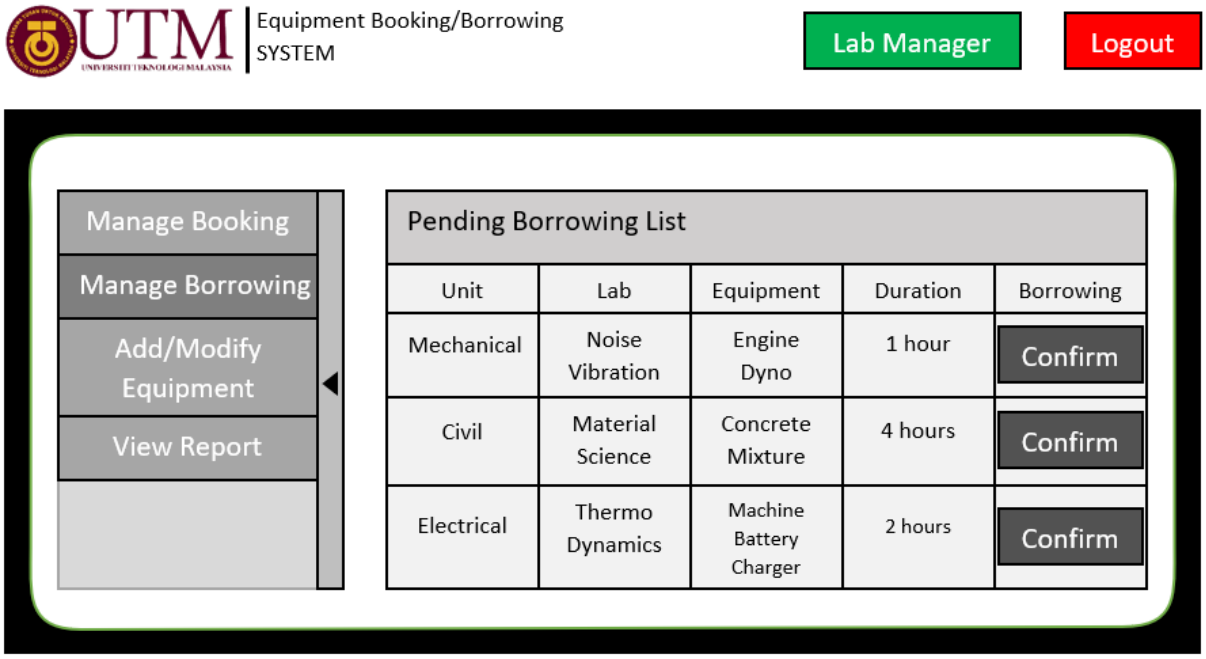


Figure 6.7: Interface for <Manage Borrowing>

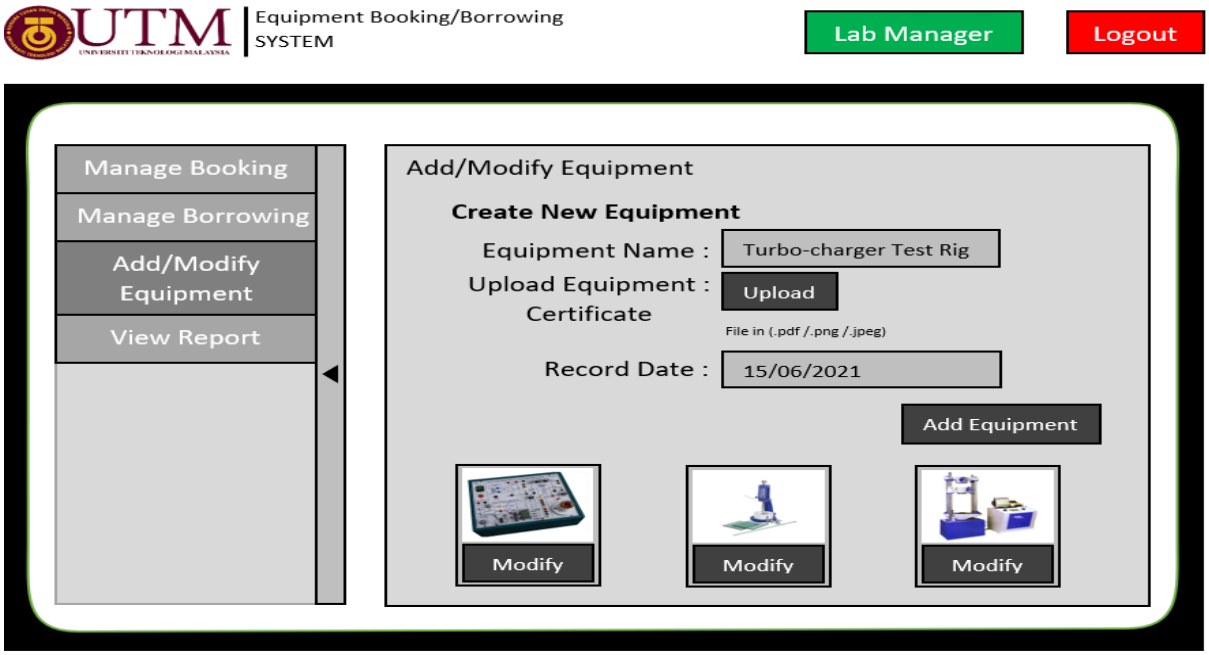


Figure 6.8: Interface for <Add/Modify Equipment>

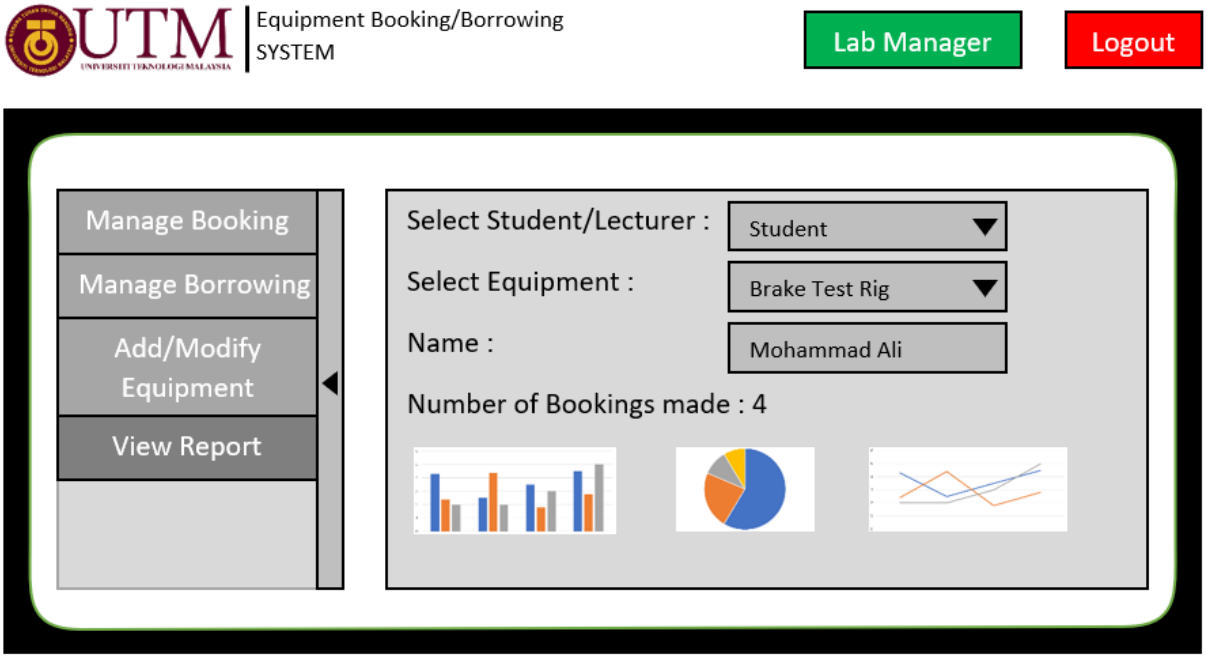


Figure 6.9: Interface for <View Report>

6. Requirements Matrix

The sequence diagrams for each use case vs. corresponding classes (entities) are listed as in Table 7.1.

Table 7.1: Description of Entities in the Database

	Create Unit	Create Lab	Create Equipment	Manage Booking	Manage Borrowing
P002, UC003, SD001	X				
P002, UC004, SD002		X			
P001, UC005, SD003			X		
P001, UC006, SD004, SD005				X	
P001, UC007, SD006, SD007					X

7. Test Cases

7.1 TC001: Test System Admin and Lab Manager Subsystem: <(UC003, UC004, UC005, UC006, UC007)>

This test contains the following test cases:

(a) TC001_01: Test <Manage Unit (SD001)>

Test Case ID	TC001_01	Test Case Description	Manage Unit		
Created By	Syafiq Ibnu Ramadhan	Reviewed By	Syafiq Ibnu Ramadhan	Version	1

QA Tester's log

Tester's Name	Syafiq	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	System Admin has logged in to the system
2	Account belongs to System Admin
3	
4	
5	
6	
7	

S #	Test Data Requirement
1	UnitID
2	UnitName
3	
4	
5	
6	
7	

Test Conditions	Process of creating a unit.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	System Admin select unit name	System display list of unit name.		
2	System Admin select number of units	System display the number of units.		
3	System Admin clicks on the "Create Unit" button	System display the unit created.		
4				

(b) TC001_01_01 Test <exception scenario1 of (SD001)>

Test Case ID	TC001_01	Test Case Description	Manage Unit _Exception		
Created By	Syafiq Ibnu Ramadhan	Reviewed By		Version	1.0

QA Tester's log

Tester's Name	Syafiq	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	System Admin has logged in to the system
2	Account belongs to System Admin
3	
4	
5	
6	
7	

S #	Test Data Requirement
1	UnitID
2	UnitName
3	
4	
5	
6	
7	

Test Conditions	Process to create

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	Click "Create Unit" button to create a unit	Failed to create a Unit		
2				
3				
4				

7.1.2 TC001_02: Test <Manage Lab (SD002)>

Test Case ID	TC002	Test Case Description	Manage Lab		
Created By	Syafiq Ibnu Ramadhan	Reviewed By		Version	1.0

QA Tester's log

Tester's Name	Syafiq	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	User has logged in to the system
2	The account belongs to System Admin
3	
4	
5	
6	
7	

S #	Test Data Requirement
1	LabID
2	LabName
3	
4	
5	
6	
7	

Test Conditions	Process to create a Lab

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	Select Lab Name	The system displays the list of lab names		
2	Select Lab Type	The system displays the list of lab types		
3	Select number of Labs	The system displays the list of number of labs		
4				

(a) TC001_02_01: Test <exception scenario1 of sequence diagram2 (SD002)>

Test Case ID	TC002_02	Test Case Description	Manage Lab (Exception)		
Created By	Syafiq Ibnu Ramadhan	Reviewed By	Syafiq Ibnu Ramadhan	Version	1.0

QA Tester's log

Tester's Name	Syafiq	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	System Admin has logged in to the system
2	The account belongs to System Admin
3	
4	
5	
6	
7	

S #	Test Data Requirement
1	LabID
2	LabName
3	
4	
5	
6	
7	

Test Conditions	Process to create lab

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	System Admin clicks on "Create Lab" button	Failed to create a Lab		
2				
3				
4				

8.1.3 TC001_03: Test <Manage Booking – Student (SD004)>

Test Case ID	TC001_03	Test Case Description	Manage Booking - Student		
Created By	Shafi Ahmed	Reviewed By	Shafi Ahmed	Version	1.0

QA Tester's log

Tester's Name	Shafi	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	Student has logged in to the system with their ACID account.
2	The account belongs to the "Student" catagory.
3	Student has an active connection to the system.
4	The student has set up the booking link.

S #	Test Data Requirement
1	LabName
2	LabID
3	unitName
4	unitId
5	equipmentName
6	eqipmentId
7	bookingId
8	studentID
9	StudentName

Test Conditions	Verify the system can manage booking by student.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	Select Manage Booking page	System should display all the details of booking		
2	Select Equipment	System should display the list of the equipments		
3	Book Equipment	The system will display "Your booking is confirmed"		

TC001_03_01: Test <Manage Booking – Academic Staff (SD004)>

Test Case ID	TC001_04	Test Case Description	Manage Booking - Staff		
Created By	Shafi Ahmed	Reviewed By	Shafi Ahmed	Version	1.0

QA Tester's log

Tester's Name	Shafi	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	Staff has logged in to the system with their ACID account.
2	The account belongs to the "Staff" catagory.
3	Staff has an active connection to the system.
4	The staff has set up the booking link.

S #	Test Data Requirement
1	LabName
2	LabID
3	unitName
4	unitId
5	equipmentName
6	eqipmentId
7	bookingId
8	staffID
9	staffName

Test Conditions	Verify the system can manage booking by staff.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	Select Manage Booking page	Site should display all the details of booking		
2	Select Equipment	Site should display the availability of the equipments		
3	Book Equipment	The system will display "Your booking is confirmed"		

TC001_03_02: Test <Manage Booking – Lab Manager (SD004)>

Test Case ID	TC001_04	Test Case Description	Manage Booking - Staff		
Created By	Shafi Ahmed	Reviewed By	Shafi Ahmed	Version	1.0

QA Tester's log

Tester's Name	Shafi	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	Staff has logged in to the system with their ACID account.
2	The account belongs to the "Staff" catagory.
3	Staff has an active connection to the system.
4	The staff has set up the booking link.

S #	Test Data Requirement
1	LabName
2	LabID
3	unitName
4	unitId
5	equipmentName
6	eqipmentId
7	bookingId
8	staffID
9	staffName

Test Conditions	Verify the system can manage booking by staff.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	Select Manage Booking page	Site should display all the details of booking		
2	Select Equipment	Site should display the availability of the equipments		
3	Book Equipment	The system will display "Your booking is confirmed"		

a) TC001_03_03: Test <Exception Scenario1 of Manage Booking – Student (SD004)>

Test Case ID	TC001_03_01	Test Case Description	Manage Booking (Exception Scenario) - Student		
Created By	Shafi Ahmed	Reviewed By		Version	

QA Tester's log

Tester's Name	Shafi	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	Student has logged in to the system with their ACID account.
2	The account belongs to the "Student" catagory.
3	Student has an active connection to the system.
4	The student has set up the booking link.

S #	Test Data Requirement
1	LabName
2	LabID
3	unitName
4	unitId
5	equipmentName
6	eqipmentId
7	bookingId
8	studentID
9	StudentName

Test Conditions	Verify the system can process booking.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	If the eqipment is not available	The system will display " Booking the equipment is not available"		

a) TC001_03_04: Test <Exception Scenario2 of Manage Booking – Staff (SD004)>

Test Case ID	TC001_03_02	Test Case Description	Manage Booking (Exception Scenario) by Staff		
Created By	Shafi Ahmed	Reviewed By	Shafi Ahmed	Version	1.0

QA Tester's log

Tester's Name	Shafi	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	Staff has logged in to the system with their account.
2	The account belongs to the "Staff" catagory.
3	Lecturer has an active connection to the system.
4	The staff has set up the booking link.

S #	Test Data Requirement
1	LabName
2	LabID
3	unitName
4	unitId
5	equipmentName
6	eqipmentId
7	bookingId
8	staffID
9	staffName

Test Conditions	Verify the system can process booking.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	If the eqipment is not available	The system will display " Booking the equipment is not available"		

8.1.4 TC001_04: Test <Manage Borrowing - Student (SD006)>

Test Case ID	TC001_04	Test Case Description	Manage Borrowing by Student		
Created By	Shafi Ahmed	Reviewed By	Shafi Ahmed	Version	1.0

QA Tester's log

Tester's Name	Shafi	Date Tested	Shafi	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	Student has logged in to the system with their ACID account.
2	The account belongs to the "Student" catagory.
3	Student has an active connection to the system.
4	The student has set up the borrowing link.

S #	Test Data Requirement
1	LabName
2	LabID
3	unitName
4	unitId
5	equipmentName
6	eqipmentId
7	borrowingId
8	studentID
9	StudentName
10	usage_duration

Test Conditions	Verify the system can manage borrowing by student.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	Select Manage borrowing page	Site should display all the details of borrowing		
2	Select Equipment	Site should display the availability of the equipments		
3	Borrowing Equipment	The system will display "Your borrowing is confirmed"		

TC001_04_01: Test <Manage Borrowing – Academic Staff (SD006)>

Test Case ID	TC001_04_01	Test Case Description	Manage Borrowing by Staff		
Created By	Shafi Ahmed	Reviewed By	Shafi Ahmed	Version	1.0

QA Tester's log

Tester's Name	Shafi	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	Staff has logged in to the system with their ACID account.
2	The account belongs to the "Staff" catagory.
3	Staff has an active connection to the system.
4	The staff has set up the borrowing link.

S #	Test Data Requirement
1	LabName
2	LabID
3	unitName
4	unitId
5	equipmentName
6	eqipmentId
7	borrowingId
8	staffID
9	staffName
10	usage_duration

Test Conditions	Verify the system can manage borrowing by staff.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	Select Manage borrowing page	Site should display all the details of borrowing		
2	Select Equipment	Site should display the availability of the equipments		
3	Borrowing Equipment	The system will display "Your borrowing is confirmed"		

TC001_04_02: Test <Exception Scenario1 of Manage Borrowing – Student (SD006)>

Test Case ID	TC0006_04_02	Test Case Description	Manage Borrowing (Exception Scenario) by Student		
Created By	Shafi Ahmed	Reviewed By	Shafi Ahmed	Version	1.0

QA Tester's log

Tester's Name	Shafi Ahmed	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	Student has logged in to the system with their ACID account.
2	The account belongs to the "Student" catagory.
3	Student has an active connection to the system.
4	The student has set up the borrowing link.

S #	Test Data Requirement
1	LabName
2	LabID
3	unitName
4	unitId
5	equipmentName
6	eqipmentId
7	borrowingId
8	studentID
9	StudentName
10	usage_duration

Test Conditions	Verify the system can process borrowing.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	If the eqipment is not available	The system will display " Borrowing the equipment is not available"		

TC001_04_03: Test <Exception Scenario2 of Manage Borrowing – Staff (SD006)>

Test Case ID	TC0006_04_02	Test Case Description	Manage Borrowing (Exception Scenario) by Staff		
Created By	Shafi Ahmed	Reviewed By	Shafi Ahmed	Version	1.0

QA Tester's log

Tester's Name	Shafi Ahmed	Date Tested	22/06/2021	Test Case (Pass/Fail/NotExecuted)	

S #	Prerequisites:
1	Student has logged in to the system with their ACID account.
2	The account belongs to the "Student" catagory.
3	Student has an active connection to the system.
4	The student has set up the borrowing link.

S #	Test Data Requirement
1	LabName
2	LabID
3	unitName
4	unitId
5	equipmentName
6	eqipmentId
7	borrowingId
8	studentID
9	StudentName
10	usage_duration

Test Conditions	Verify the system can process borrowing.

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not excuted / Suspended
1	If the eqipment is not available	The system will display " Borrowing the equipment is not available"		