A Software Project

NSTU Online Course Registration

Team Members

Md. Mahbub Alam
[ASH1825003M]

Khair Ahammed
[ASH1825036M]

Akash Chandra Debnath
[ASH1825037M]

Project Mentor

Dipanita Saha
Assistant Professor
IIT, NSTU

Institute of Information Technology

October 2021

Table of Contents

Part 4: Design and Architecture

Table of Figures

Part-1
Preface

1. Preface

Software Requirement Specifications have also known as SRS, picturizes the full depth, requirements, scope, and many other criteria. Detailed and specified requirements of our project will be provided by a document of "***NSTU Online Course Registration***"

.

1.1. Purpose

The main purpose of this project is to provide the automation process of course registration for students. So that students can complete their registration process in short time.

1.2. Project Scope

We want to build a website which will have Education Branch, Departments, Students, Teachers, Section Officers and Hall Provosts. They all will be connected through a large database which will help them to store their data and get the necessary information without any hazard. We also visualize their connection through different diagrams. This project will reduce time and cost of both parties and it will be effective for our education system also.

1.3. Glossary

This subsection contains definitions of all the terms, acronyms, and abbreviations used in the document. Terms and concepts from the application domain are defined.

- NSTU – Noakhali Science and Technology University
- IIT – Institute of Information Technology
- API – Application Programming Interface
- SRS – Software Requirement Specifications
- UI – User Interface
- UX – User Experience
- SDLC – Software Development Life Cycle
- DBMS – Database Management System

Part-2
Project Plan

2. Project Plans of NSTU Online Course Registration

2.1. Overview

The students of NSTU are facing many problems during the course registration process. Such as slow registration process, mistakes in forms, unclear information, time-wasting, etc. Sometimes, they are not informed clearly about their registration.

Students have to walk from one building to another with their course registration forms. This system is very tedious and a waste of time. The forms may be damaged or lost. But in digital systems, the forms are stored on the server which is never likely to be damaged. When we need to know some kind of information, we can get it in a few seconds.

The digital system also reduces the cost of office work. Office workers, teachers can save their time.

The manual management of the whole university's registration process is very tough. So, for such many reasons, we bring the "**NSTU Online Course Registration**" project. We are sure that all problems will be removed because our project has a big ambition.

2.2. User Classes and Characteristics

In this system there are six stakeholders are Section Officers, Education Branch, Students, Course Coordinators, Hall Provosts and Director/Chairman of Departments.

Section Officer: Section officer can start the process of course registration process by sending application, publish notices, etc.

Education Branch: Education branch can reply back to the section officer's applications and send a notice of course registration.

Student: Student can apply for their course registration, download their term fee receipt, etc.

Course Coordinator, Hall Provost, Director/Chairman: They all are faculty member who approve or proceed the course registration of the students.

2.3. Project Timeline
  i.   Work Distribution

	Team Members		
Tasks	Mahbub	Akash	Khair
Planning	✓	✓	✓
Research	✓	✓	✓
Take User Interviews	✓	✓	✓
Drawing Diagrams	✓	✗	✗
Prototyping	✓	✗	✓
Design	✓	✓	✓
Development	✗	✓	✓
Testing	✓	✓	✓

  ii. Timeline

Tasks	**Timeline**
Planning	1st week
Research	2nd to 3rd week
Take User Interviews	4th week
Drawing Diagrams	5th to 6th week
Prototyping	7th to 8th week
Design	8th to 11th week
Development	11th to 16th week
Testing	17th to 18th week

Part 3

Software Requirements Specification Analysis

3. Design and Implementation Constraints

Design and implementation constraints are those that we have used to implement this project make successful. It also describes tool that enables developers and testers to view and interact with the UI/UX elements of this website.

3.1. User Interface Technology

3.1.1. Programming Language

We will use PHP language to design and develop our website. For this purpose, we will use IDE likes Visual Studio Code, Notepad++ etc.

3.1.2. Web Server

A Web server is a program that uses HTTP (Hypertext Transfer Protocol) to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computers' HTTP clients. Dedicated computers and appliances may be referred to as Web servers as well. We will use the Apache HTTP server to implement this project.

3.1.3. Database Server

We will use MySQL database server to store all of the information of this system. The reason behind to choose the database server are given below:

- Security
- Reporting and Data Mining
- Replication
- Fault tolerance
- Performance diagnostics

4. Use Case Diagram
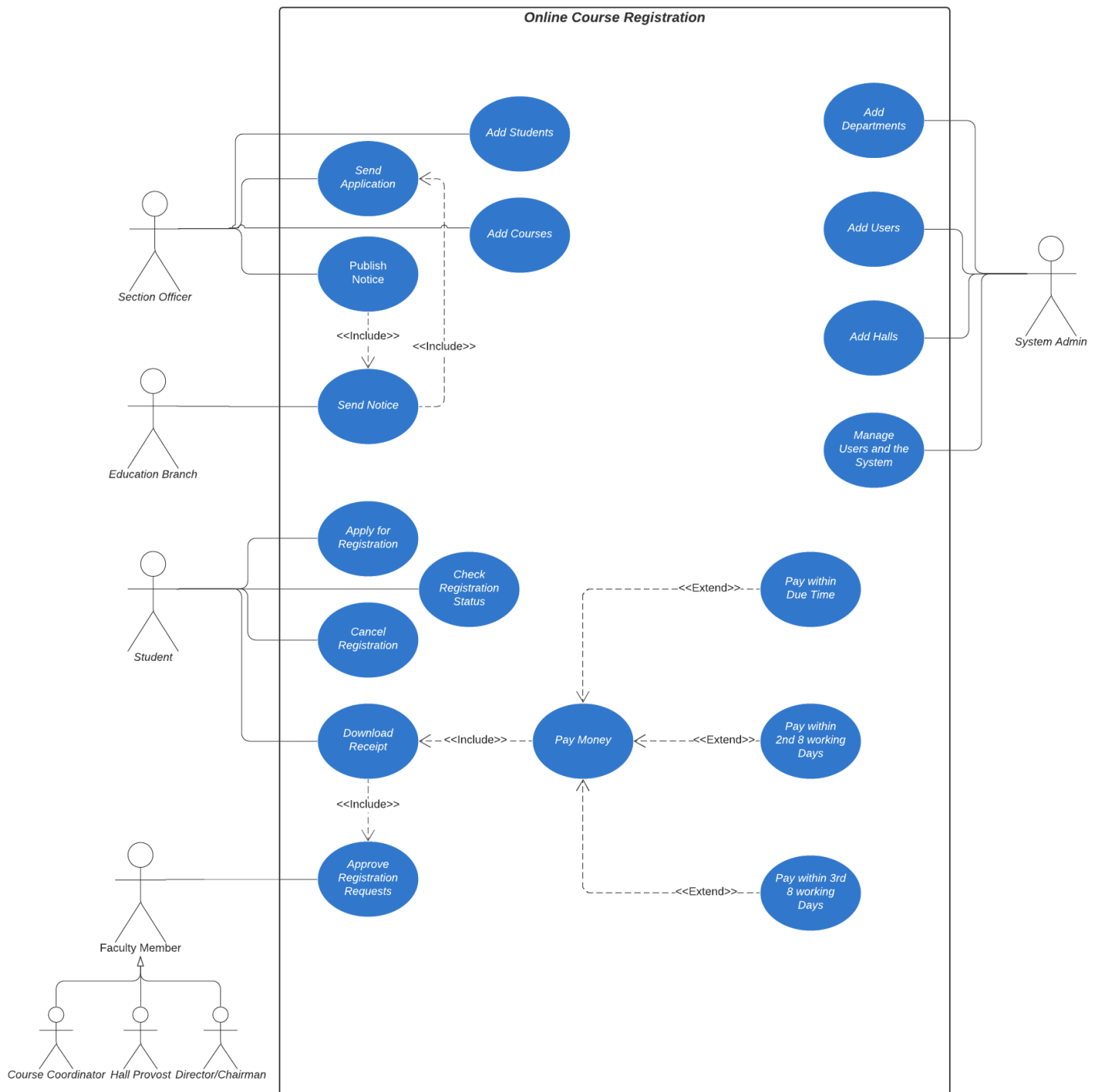
Figure 1: Use Case of NSTU Online Course Registration

5. Use Case Descriptions

Table 01: Use Case Description of Sending Application

Use Case ID	01	
Use Case Name	Send Application	
Goal	Provide an application to the Education Branch	
Preconditions	User must be registered & logged in to the system	
Success End Condition	The receiver email is correct to send the application The email is sent to the Education Branch	
Failed End Condition	The Section officer doesn't find any application file The Education Branch's email is not correct to send the application The email is not sent to the Education Branch	
Primary Actors: **Secondary Actors:**	Section Officer Education Branch, System	
Trigger	Click on Application from the navigation bar	
Main Success Flows	**Step**	**Action**
	1	The Section Officer clicks on Browse button
	2	The system shows files to select
	3	The Section Officer selects the required Application pdf file
	4	The Section Officer writes the Application Subject on Subject input field
	5	The Section Officer writes the Education Branch's email address on Send To input field
	6	The Section Officer clicks on the Send button
	7	The System shows a success message "Mail is sent"
	8	The Section officer clicks on the OK button
Alternative Flows	**Step**	**Branching Action**
	3a	The Section officer doesn't select any files
	7a	The System shows a failed message "Mail is not sent"
Quality Requirements	**Step**	**Requirement**
	3	The Section Officer should send a PDF file.
	5	The Section Officer should send email to the Education Branch

Table 02: Use Case Description of Sending Notice of Course Registration

Use Case ID	02	
Use Case Name	Send a notice	
Goal	Send an email about course registration notice to the Section Officer	
Preconditions	User must be registered & logged in to the system	
Success End Condition	The replying email is correct destination mail to send the notice The email is sent to the Section Officer	
Failed End Condition	The Education Branch doesn't find the correct notice file The Section Officer's email is not correct to send the mail The email is not sent to the Section Officer	
Primary Actors: **Secondary Actors:**	Education Branch Section Officer, System	
Trigger	Click on Application from the navigation bar	
Main Success Flows	**Step**	**Action**
	1	The Education Branch clicks on Browse button
	2	The system shows files to select
	3	The Education Branch selects the required Application pdf file
	4	The Education Branch writes the Application Subject on Subject input field
	5	The Education Branch writes the Education Branch's email address on Send To input field
	6	The Education Branch clicks on the Send button
	7	The System shows a success message "Mail is sent"
	8	The Education Branch clicks on the OK button
Alternative Flows	**Step**	**Branching Action**
	3a	The Education Branch doesn't select any files
	7a	The System shows a failed message "Mail is not sent"
Quality Requirements	**Step**	**Requirement**
	3	The Education Branch should send a PDF file.
	5	The Education Branch should send email to the Section Officer

Table 03: Use Case Description of publish a notice

Use Case ID	03	
Use Case Name	Publish a notice	
Goal	Publish a notice of Student's course registration which is sent by Education Branch	
Preconditions	User must be registered & logged in to the system	
Success End Condition	The notice is published	
Failed End Condition	The notice is not published	
Primary Actors: Secondary Actors:	Section Officer System	
Trigger	Click on the message menu bar from home page	
Main Success Flows	**Step**	**Action**
	1	The Section Officer clicks on the notice coming from the Education Branch
	2	The Section Officer clicks on the Publish button to publish the notice
	3	The System shows a confirmation message
	4	The Section officer clicks on the Ok button
Alternative Flows	**Step**	**Branching Action**
		Not Applicable
Quality Requirements	**Step**	**Requirement**
		Not Applicable

Table 04: Use Case Description of apply for semester registration

Use Case ID	04	
Use Case Name	Apply for semester registration	
Goal	Apply for semester course registration of residential and non-residential students	
Preconditions	User must be registered & logged in to the system	
Success End Condition	Application of course registration is completed	
Failed End Condition	Application of course registration is not completed	
Primary Actors: **Secondary Actors:**	Student System	
Trigger	Clicks on the Apply Now button from Home page	
Main Success Flows	**Step**	**Action**
	1	The system shows a semester registration page
	2	The student selects his/her current semester's year and term
	3	The System shows a form of full course of the selected semester
	4	Then the student clicks on the Apply button
	5	The system shows a confirmation message
	6	The student clicks on the ok button
Alternative Flows	**Step**	**Branching Action**
	1a	The student clicks on the Semester Registration from the navigation bar
Quality Requirements	**Step**	**Requirement**
		Not Applicable

Table 05: Use Case Description of the Course Coordinator to approve the course registration

Use Case ID	05
Use Case Name	Approve the course registration form as Course Coordinator
Goal	Proceed students by signing on the course registration form
Preconditions	User must be registered & logged in to the system Students should apply for the course registration
Success End Condition	The Course Coordinator proceeds students for course registration
Failed End Condition	The Course Coordinator doesn't proceed students for course registration
Primary Actors: Secondary Actors:	Course Coordinator System
Trigger	Clicks on the Start Process button from Home page

Main Success Flows	Step	Action
	1	The system forwards to Student Registration page
	2	The Course Coordinator clicks on the Check All checkbox
	3	The Course Coordinator clicks on the Proceed button
	4	The system shows how many students are proceeded and are remained to clear course registration process.
Alternative Flows	**Step**	**Branching Action**
	1a	The Course Coordinator clicks on the Student Registration from the navigation bar
Quality Requirements	**Step**	**Requirement**
		Not Applicable

Table 06: Use Case Description of the Hall Provost to approve the course registration form

Use Case ID	06	
Use Case Name	Approve the course registration form as Hall provost	
Goal	Proceed students by signing on the course registration form	
Preconditions	User must be registered & logged in to the system The Course Coordinators must proceed students for course registration	
Success End Condition	The Hall Provost proceeds students for course registration	
Failed End Condition	The Hall Provost doesn't proceed students for course registration	
Primary Actors: Secondary Actors:	Hall Provost System	
Trigger	Clicks on the Start Process button from Home page	
Main Success Flows	**Step**	**Action**
	1	The system forwards to Student Registration page
	2	The Hall Provost selects Department/Institute, Year, Term from dropdown.
	3	Then the Hall Provost clicks on OK button
	4	The System shows a list of students on the screen who are proceeded by Course Coordinator
	5	The Hall Provost clicks on the Check All on checkbox
	6	The Hall Provost clicks on the Proceed button
	7	The system shows how many students are proceeded and are remained to clear course registration process.
Alternative Flows	**Step**	**Branching Action**
	1a	The Hall Provost clicks on the Student Registration from the navigation bar
Quality Requirements	**Step**	**Requirement**
		Not Applicable

Table 07: Use Case Description of the Director/Chairman to approve the course registration

Use Case ID	07
Use Case Name	Approve the course registration form as Director/Chairman
Goal	Proceed students by signing on the course registration form
Preconditions	User must be registered & logged in to the system The Hall Provost must proceed students for course registration
Success End Condition	The Director proceeds students for course registration
Failed End Condition	The Director doesn't proceed students for course registration
Primary Actors: Secondary Actors:	Director System
Trigger	Clicks on the Start Process button from Home page

Main Success Flows	Step	Action
	1	The system forwards to Student Registration page
	2	The Director selects Year, Term from dropdown.
	3	Then the Director clicks on OK button
	4	The System shows a list of students on the screen who are proceeded by the Hall Provost
	5	The Director clicks on the Check All on checkbox
	6	The Director clicks on the Proceed button
	7	The system shows how many students are proceeded and are remained to clear course registration process.
Alternative Flows	Step	Branching Action
	1a	The Director clicks on the Student Registration from the navigation bar
Quality Requirements	Step	Requirement
		Not Applicable

Table 08: Use Case Description of the Checking Registration Status

Use Case ID	08	
Use Case Name	Check Registration Status	
Goal	There system should show a status of student's course registration approval.	
Preconditions	User must be registered & logged in to the system	
Success End Condition	The system shows status	
Failed End Condition	The system doesn't show any status	
Primary Actors:	Student	
Secondary Actors:	System	
Trigger	Clicks on the Registration from navigation bar	
Main Success Flows	**Step**	**Action**
	1	The system shows course registration status of the student
Alternative Flows	**Step**	**Branching Action**
		Not Applicable
Quality Requirements	**Step**	**Requirement**
		Not Applicable

Table 09: Use Case Description of the canceling registration status

Use Case ID	09	
Use Case Name	Cancel Registration	
Goal	Students can cancel his/her current course registration	
Preconditions	User must be registered & logged in to the system	
Success End Condition	The course registration is cancelled.	
Failed End Condition	The course registration is completed and cannot be cancelled.	
Primary Actors:	Student	
Secondary Actors:	System	
Trigger	Clicks on the Registration from navigation bar	
Main Success Flows	**Step**	**Action**
	1	The system shows course registration status of the student
	2	The student clicks on the Cancel Registration button
	3	The system shows a success message
Alternative Flows	**Step**	**Branching Action**
		Not Applicable
Quality Requirements	**Step**	**Requirement**
		Not Applicable

Table 10: Use Case Description of the download and printing term charge receipt

Use Case ID	10
Use Case Name	Download term charge receipt
Goal	Download term charge receipt of course registration
Preconditions	User must be registered & logged in to the system The student should be proceeded by the Director for course registration
Success End Condition	The term charge receipt is available to download
Failed End Condition	The term charge receipt is not available to download
Primary Actors:	Student
Secondary Actors:	System
Trigger	Clicks on the Payment from navigation bar

Main Success Flows	**Step**	**Action**
	1	The student selects Year and term from dropdown box.
	2	The student clicks on the OK button
	3	The system shows a receipt form
	4	The system also shows a timer to download the receipt and pay the term charge.
	5	The system shows total course fee to pay within due time
	6	Then the student clicks on a checkbox named I read it
	7	To download the term charge receipt, the student clicks on the Download button
Alternative Flows	**Step**	**Branching Action**
	3a	The system doesn't show any receipt form
	5a1	The system shows "due time is over" and adds 100 BDT fines to pay within 2nd 8 working days.
	5a2	The system shows "2nd due time is over" and adds 5000 BDT fines to pay within 3rd 8 working days.
Quality Requirements	**Step**	**Requirement**
	4	The student should pay and download his/her receipt within given due time.

Table 11: Use Case Description of the adding students

Use Case ID	11	
Use Case Name	Add Students to the system	
Goal	New students in the department and their information should be added to the system	
Preconditions	User must be registered & logged in to the system	
Success End Condition	All required field must be filled	
Failed End Condition	The required fields are not filled	
Primary Actors:	Section Officer	
Secondary Actors:	System	
Trigger	Clicks on the Students from navigation bar	
Main Success Flows	**Step**	**Action**
	1	The section officer fills all the required field of student's information
	2	The section officer clicks on the Add button
	3	The system shows a success message
	5	The section officer clicks on the OK button
Alternative Flows	**Step**	**Branching Action**
	3a	The system shows a failed message
Quality Requirements	**Step**	**Requirement**
	1.1	All the required information should be correct.
	1.2	Image size should be less than 100 KB and valid.

Table 12: Use Case Description of the adding courses

Use Case ID	10	
Use Case Name	Add Courses to the system	
Goal	Courses information should be added to the system	
Preconditions	User must be registered & logged in to the system	
Success End Condition	All required field must be filled	
Failed End Condition	The required fields are not filled	
Primary Actors:	Section Officer	
Secondary Actors:	System	
Trigger	Clicks on the Courses from navigation bar	
Main Success Flows	**Step**	**Action**
	1	The section officer fills all the required field of course's information
	2	The section officer clicks on the Add button
	3	The system shows a success message
	5	The section officer clicks on the OK button
Alternative Flows	**Step**	**Branching Action**
	3a	The system shows a failed message
Quality Requirements	**Step**	**Requirement**
	1.1	All the required information should be correct.

Table 13: Use Case Description of the adding faculty members

Use Case ID	13	
Use Case Name	Add Faculty Members	
Goal	Faculty Member's information should be added to the system	
Preconditions	User must be registered & logged in to the system	
Success End Condition	All required field must be filled	
Failed End Condition	The required fields are not filled	
Primary Actors: Secondary Actors:	System Admin System	
Trigger	Clicks on the Faculty Members from navigation bar	
Main Success Flows	**Step**	**Action**
	1	The admin fills all the required field of faculty member's information
	2	The admin clicks on the Add button
	3	The system shows a success message
	5	The admin clicks on the OK button
Alternative Flows	**Step**	**Branching Action**
	3a	The system shows a failed message
Quality Requirements	**Step**	**Requirement**
	1.1	All the required information should be correct and valid.
	1.2	Image should be less than 100KB.

Table 14: Use Case Description of the adding halls

Use Case ID	14	
Use Case Name	Add Halls	
Goal	Hall's information should be added to the system	
Preconditions	User must be registered & logged in to the system	
Success End Condition	All required field must be filled	
Failed End Condition	The required fields are not filled	
Primary Actors:	System Admin	
Secondary Actors:	System	
Trigger	Clicks on the Halls from navigation bar	
Main Success Flows	**Step**	**Action**
	1	The admin fills all the required field of hall's information
	2	The admin clicks on the Add button
	3	The system shows a success message
	5	The admin clicks on the OK button
Alternative Flows	**Step**	**Branching Action**
	3a	The system shows a failed message
Quality Requirements	**Step**	**Requirement**
	1.1	All the required information should be correct.

Table 15: Use Case Description of the adding departments

Use Case ID	15	
Use Case Name	Add Departments	
Goal	Department's information should be added to the system	
Preconditions	User must be registered & logged in to the system	
Success End Condition	All required field must be filled	
Failed End Condition	The required fields are not filled	
Primary Actors:	System Admin	
Secondary Actors:	System	
Trigger	Clicks on the Departments from navigation bar	
Main Success Flows	**Step**	**Action**
	1	The admin fills all the required field of Department's information
	2	The admin clicks on the Add button
	3	The system shows a success message
	5	The admin clicks on the OK button
Alternative Flows	**Step**	**Branching Action**
	3a	The system shows a failed message
Quality Requirements	**Step**	**Requirement**
	1.1	All the required information should be correct.

6. Requirement Specification Analysis

Before a system is designed and implemented, the requirements have to be specified in enough detail to make analysis and design possible. This is a big part of software engineering, especially for larger systems. The complete requirement specification based on the elicitation process is described in this section.

6.1. Functional Requirements

Every system must have some functional requirements. Functional requirement defines a system or its component. It describes the functions a software must perform. A function is nothing but inputs, its behavior, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform. Functional software requirements help to capture the intended behavior of the system. Now, we are going to mention functional requirements associating with our project.

Table 16: Section officer adds students

FR-1	Section officer adds students
Description	Section officer need to add students and their information to the database.
Stakeholders	Section Officer
Priority	High

Table 17: Section officer adds courses

FR-2	Section officer adds courses
Description	Section officer adds courses and course credits in the database according to the year and term.
Stakeholders	Section Officer
Priority	High

Table 18: Section officer updates courses

FR-3	Section officer update courses
Description	Section officers can make any changes to update the courses.
Stakeholders	Section Officer
Priority	High

Table 19: Section officer deletes any course

FR-4	Section officer delete any course from the database
Description	Sometimes some courses need to be deleted and the section officer can do this.
Stakeholders	Section Officer
Priority	High

Table 20: Section officer sends application

FR-5	Section officer send application
Description	Section officer can send an application to the Education Branch of NSTU about course registration.
Stakeholders	Section Officer
Priority	High

Table 21: Education Branch sends notice

FR-6	Education Branch sends notice
Description	After section officer sent the application, the education branch needs to send a notice about course registration.
Stakeholders	Education Branch
Priority	High

Table 22: Section officer publishes the notice

FR-7	Section officer publishes the notice
Description	Section officer has to publish the notice that is sent from education branch.
Stakeholders	Section Officer
Priority	High

Table 23: Section officer opens the registration process

FR-8	The section officer opens the registration process
Description	After publish the notice, the section officer has to open the registration process for students. He/she needs to set a registration due time, 2nd due time, and 3rd due time.
Stakeholders	Section officer
Priority	High

Table 24: Student applies for course registration

FR-9	Student applies for course registration
Description	After the opening of the registration process, students have to apply for their course registration. They should select the year and term to apply. In addition, they should apply for registration within the stipulated time.
Stakeholders	Student
Priority	High

Table 25: Students can see registration status

FR-10	Students can see registration status
Description	Students can see their own registration status.
Stakeholders	Student
Priority	Medium

Table 26: Course coordinator approves the registration request

FR-11	Course coordinator approves the registration request
Description	For the next step, the course coordinator must approve student's course registration requests by clicking on the "Approve" button.
Stakeholders	Course Coordinator
Priority	High

Table 27: Hall provost approves the registration request

FR-12	Hall provost approves the registration request
Description	For the next step, the hall provost must approve student's course registration requests by clicking on the "Approve" button, according to each of the department and session.
Stakeholders	Hall Provost
Priority	High

Table 28: Director/Chairman approves the registration request

FR-13	Director/Chairman approves the registration request
Description	The Director/Chairman must approve student's course registration requests by clicking on the "Approve" button, according to each of the session.
Stakeholders	Director/Chairman
Priority	High

Table 29: Student downloads payment receipt

FR-14	Student downloads payment receipt
Description	After the director/chairman approves the registration request, students can download their payment receipt. Before download payment receipt, they should click on the "I read it" checkbox because they may need to understand the term fee receipt.
Stakeholders	Student
Priority	High

Table 30: Student prints payment receipt

FR-15	Student prints payment receipt
Description	After downloading the receipt, students should print it to pay term fee to the bank.
Stakeholders	Student
Priority	High

Table 31: Student pays term fee

FR-16	Student pays term fee
Description	Student should pay his term fee to the bank within due time.
Stakeholders	Student
Priority	High

Table 32: Section officer updates student's registration status

FR-17	Section officer updates student's registration status
Description	Section officer must update the student's course registration status in the database after student's pay his/her term fee.
Stakeholders	Section officer
Priority	High

Table 33: Admin adds department

FR-18	Admin adds department
Description	System admin needs to include all department and institutes to the database.
Stakeholders	Admin
Priority	High

Table 34: Admin manages departments

FR-19	Admin manage departments
Description	Admin can delete, add and update department or department information
Stakeholders	Admin
Priority	High

Table 35: Admin can add Faculty Members and Section Officers

FR-20	**Admin can add Faculty Members and Section Officers**
Description	Admin needs to add Faculty Members and Section Officers and their information. In the system, the Course Coordinator, Director/Chairman, and Hall provost belong to the Faculty Members.
Stakeholders	Admin
Priority	High

Table 36: Admin manages the Faculty Members and Section Officers

FR-21	**Admin manages the Faculty Members and Section Officers**
Description	On our website, the admin may need to delete, add or update the Faculty Members and Section Officers and also their information. Admin must log in to the system to apply such a manageable operation.
Stakeholders	Admin
Priority	High

Table 37: Admin can change his/her login information

FR-22	**Admin can change his/her login information**
Description	Admin needs to change his/her own account and password. For change account and password own profile, he/she needs to log in to the system.
Stakeholders	Admin
Priority	Medium

Table 38: Section officer can delete/update notice

FR-23	Section officer can delete/update notice
Description	Section officer needs to delete/update notice.
Stakeholders	Section Officer
Priority	Low

Table 39: Data retrieve from cloud server

FR-24	Data retrieve from cloud server
Description	Data must be retrieved from server as the whole system will be dynamic. It is also to be said that, all operational functionality will be occurred on server also.
Stakeholders	Admin
Priority	High

6.2. Data Requirements

For defining data requirements, we need to build the model. For our application maximum data would be loaded from remote user. And for that purpose, we need to focus on some major points. Such as:

- Types of entity of the system
- Route data locations
- Capacity and resources of the data requirements
- Data source sequence
- Data availability schedules
- Quantity of data
- Availability of data

6.3. Performance Requirements

It is very important to maintain performance of any software system. To ensure performance, we need to maintain some steps. Now, I will explain some perspective by which we are going to enhance the performance of our project.

6.3.1. Speed & Latency Requirements

Speed and latency requirements must be ensured while retrieving data from the cloud server.

Table 40: Data query must be faster

SLR-1	Data query must be faster
Description	When a student selects year and term to apply for his/her course registration, the information of courses must be shown within seconds.
Stakeholders	Student
Priority	High

6.3.2. Precision & Accuracy Requirements

Result that is to be shown to the end user is need to be accurate. Because, wrong information might be ruined the whole business process.

Table 26: Data query must be accurate

PAR-1	Data query must be accurate
Description	When a student selects year and term to apply for his/her course registration, then the query result must be accurate according to the selection input.
Stakeholders	Student
Priority	High

6.3.3. Capacity Requirements

The developed system by us must be capable to handle user data, provide accurate information, handling database, manage http request etc.

Table 27: The system will handle thousands of data

CR-1	The system will handle thousands of data
Description	The system needs to handle data thousands of data every moment.
Stakeholders	Admin
Priority	High

6.4. Dependability Requirements

The term dependability is measured based on four dimensions. Such as:

- Availability
- Reliability
- Safety
- Security

If we want to say that our application system is dependable then it must fulfil the four dimensions. But there are other tasks. Like there is no way to make mistakes or our system should have the ability to detect and then remove errors. Besides that, it is also very important to limit the damage which might be caused by system failure.

6.4.1. Reliability & Availability Requirements

Now, we will mention requirements which are related to reliability and availability.

Table 28: The system must be available on 24 X 7

RAR-1	The system must be available on 24 X 7
Description	Our system must be available all day long, every day in a week • The system must be updated regularly • System must be malware free
Stakeholders	Student, Course Coordinator, Education Branch, Hall Provost, Director/Chairman, Admin
Priority	High

6.4.2. Robustness or Fault-Tolerance Requirements

To ensure robustness and fault-tolerance facilities to the end users, it is urgent to ensure 0% crush. Moreover, it must show accurate results.

Table 29: The system handles all user access without system errors

RFT-1	The system handles all user access without system errors
Description	Thousands of users might hit our application system at a time. All their requests must be handled without any fault.
Stakeholders	N/A
Priority	High

6.4.3. Safety-Critical Requirements

There are no safety-critical requirements in our project.

6.5. Maintainability & Supportability Requirements

It is very important to provide after service or support to the end users.

6.5.1. Maintainability Requirements

Table 30: System helps to update user profile

MR-1	System helps to update user profile
Description	It is important to update user profile.
Stakeholders	Student, Course Coordinator, Education Branch, Hall Provost, Director/Chairman
Priority	Medium

6.5.2. Supportability Requirements

Supportability requirements may have related to some extends. Like:

- Testability
- Extensibility
- Adaptability
- Maintainability
- Compatibility
- Configurability

- Serviceability
- Install ability

Our application meets all of the above requirements related to supportability.

6.5.3. Adaptability Requirements

There are no adaptability requirements in our system software.

6.6. Security Requirements

Making software security as a requirement is very important. Software security requirements should be its functional requirement. Software security enforces security of an application system.

Functionality related to software security can either be directly tested or observed. Some security related requirements are given below:

- Signing in as a student
- Get access according to logged in user
- Apply for the course registration without having any issues
- Signing out as a student
- Handling encrypted passwords

While accessing to the system, each and every module must provide a central authentication mechanism. There is also a process to prevent entering into the system by ensuring hashed password for the unauthenticated users.

6.6.1. Access Requirements

For accessing to our application system, there remains some authentication and authorization techniques. And every module of our system will provide it. Now I will provide an explanation below.

Table 31: Application provides security mechanism

AR-1	Application provides security mechanism
Description	Every module is designed in such a way that it only gives access to the authorized and authenticated users.
Stakeholders	Student, Course Coordinator, Education Branch, Hall Provost, Director/Chairman, Admin
Priority	High

6.6.2. Integrity Requirements

Integrity requirements refers to a security system which ensures an expectation of data quality. It also ensures that all data of the system would never be exposed to the malicious modification or accidental destruction. For that reason, we will store our user passwords as encrypted format which is impossible to decrypt. It is also called hashed password.

6.6.3. Privacy Requirements

It is very important to ensure privacy of the system users. Privacy requirements enhances to protect stakeholder's privacy. In this way, all data or a partial part of data are going to be disclosed according to system's privacy policy. To ensure privacy, the central database should be protected by the anonymous. Users are permitted to get access to those data which are being associated by them which can be ensured by the user log in system.

6.7. Usability and Human-Interaction Requirements

The main target of developing any system is to make the system user friendly and easy to usable for the end users.

6.7.1. Ease of Use Requirements

Our application is easy to use and also easily understandable.

Table 32: Application must be usable for the end users

EUR-1	Application must be usable for the end users
Description	This website is enough usable to the all users in this system by which they can operate this system easily.
Stakeholders	Student, Course Coordinator, Education Branch, Hall Provost, Director/Chairman
Priority	High

6.7.2. Personalization and Internationalization Requirements

There are not any personalization and internationalization requirements to our system. This maiden version of our application is only be operated by Bangladesh.

6.7.3. Understand ability and Politeness Requirements

It is already said that the application which we are going to develop, is understandable enough. The system provides hints to users whether any error occurred or wrong. By reading those errors users can be able to operate the system easily.

6.7.4. Accessibility Requirements

There are no specific accessibility requirements associated to our system yet.

6.7.5. User Documentation Requirements

Documentation are mainly two types. One is internal documentation which is generally written by the application engineers. It is prepared to make development life cycle easier for the system engineers or system analysts.

Table 33: The system engineer documentation

UDR-1	The system engineer documentation
Description	To develop our website named NSTU Online Course Registration, firstly we have to make a system analysis team as well as documentation team.
Stakeholders	System analysts or software developers
Priority	Medium

6.7.6. Training Requirements

Training requirements involved in after service of any application. It is very necessary to properly train up end users to the system so that they would be capable to operate easily. After launching the full package to the market, firstly we provide training to the different end users like Student, Course Coordinator, Education Branch, Hall Provost, Director/Chairman.

6.8. Look and Feel Requirements

Look and feel requirements mainly refers how the system will look like and how the user interface or graphical user interface of our system will display to the user.

6.8.1. Appearance Requirements

Student, Section Officer, Course Coordinator, Education Branch and all other user must know which input fields are required and which are not. For that reason, we will use labels for all input fields. Input fields might be text type, radio, checkbox, spinner etc.

Table 34: Labels of mandatory fields must be bold

AR-1	Labels of mandatory fields must be bold
Description	The mandatory field's label must be bold and all input fields must have placeholder to make it easier for the users.
Stakeholders	Student, Course Coordinator, Education Branch, Hall Provost, Director/Chairman
Priority	Medium

6.8.2. Style Requirements

After keeping all contents, it is very essential to load stylesheet to the website. It is to be said that we are going to develop our system at all platform. Style makes the system lucrative.

Table 35: The appearance must be controllable using stylesheet file

SR-1	The appearance must be controllable using stylesheet file
Description	For websites, style sheet files are CSS and Bootstrap. So, all stylesheet must be controllable by the CSS and Bootstrap file.
Stakeholders	Software developer
Priority	High

6.9. Operational and Environmental Requirements

Operational and environmental requirement refers to the capabilities, performance measurements, process, measurements of effectiveness, measurements of performance, measures of sustainability, measurements of technical performances etc.

6.9.1. Expected Physical Requirements

There are no expected physical requirements in our system.

6.9.2. Requirements for Interfacing with Adjacent Systems

There are no requirements for interfacing with adjacent system for our project.

6.9.3. Release Requirements
There are no specific release requirements in our system.

6.10. Legal Requirements

Legal requirements normally refer to the terms and conditions or privacy policy of any organizations. The terms and condition of our application is that, no third-party software or person are allowed to engage to use our data for their business purpose.

6.10.1. Compliance Requirements

There are no specific compliance requirements for our system.

6.10.2. Standards Requirements

There are no specific standards requirements for our system.

7. Requirement Engineering Process

Requirement engineering refers to the process of defining, documenting and maintaining requirements in the engineering design process. It is a common role in systems engineering and software engineering.

7.1. Requirement Elicitation Techniques

Requirement elicitation is the process of collecting and refining stakeholder's requirements. It is perhaps the most difficult, most error-prone and most communication intensive software development. It can be successful only through an effective customer-developer partnership. It is needed to know what the users really need.

7.1.1. Hold Elicitation Interviews

We hold interviews that can be performed one-on-one or with a small group of stakeholders. They are an effective way to elicit requirements without taking too much stakeholder time because we meet with people to discuss only the specific requirements that are important to this system. Interviews are helpful to separately elicit requirements from members in preparation for workshops where those members of this system come together to resolve any conflicts.

7.1.2. Perform Observation

We observe the course registration system. We observe how they manage the student's course registration information, how they manage the information of faculty members and how they manage their system. Every observation must be guided by clearly stated objectives. The analyst should know what data is to be collected, how observation will be done, when and where to observe, how the data will be collected and what the data will be used for after analysis.

7.1.3. System Interface Analysis

Interface analysis can also help in determining requirements for interoperability and exposing interfacing stakeholders early on in the project. It helps to clarify the boundaries of the interacting application, identify the functionality, input and output of each interface.

7.1.4. Perform Document Analysis

Existing documentation can help reveal how systems currently work or what they are supposed to do. Documentation includes any written information about current systems, business processes, requirements specifications, competitor research. Reviewing and analysing the documents can help identify functionality that needs to remain, functionality that isn't used.

Appendix

Sample interview questions

1. What is the process of course registration?
2. How many types of users are involved in the course registration?
3. When does the course registration begin?
4. What problems generally face during course registration?
5. Who handle the course registration process?
6. How do students pay their course registration fees?
7. What steps are taken when a student can't pay registration fees?

Prioritization Technique

We've prioritized the functional requirements by following Three-level Scale technique.

Three-level Scale:

When a BA categorizes the requirements in any of the ordering or ranking scale, it is subject to the analyst's understanding of the business. Many analysts suggest that this method has some drawbacks and advocate methods that have more than one scale. Covey, Rebecca and Merrill would have never in their wildest dreams have thought that their "The four-quadrant 'Eisenhower Decision Matrix' for importance and urgency", from their self-help book First things First, would become one of the most widely used prioritization techniques in the IT space.
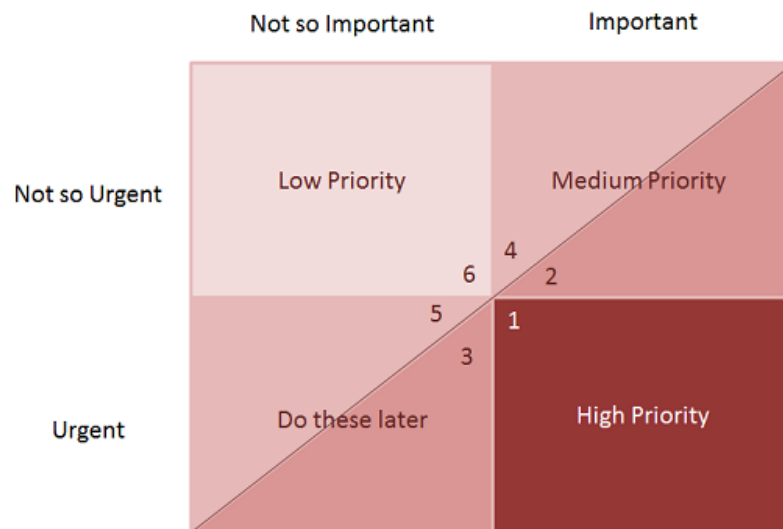

Figure 2: Eisenhower Decision Matrix – Lower the number, higher the priority of the section

With the numbering on the different sections of the diagram, the priority of the sections is implicit.

Important items have the highest preference, while urgent items have lower preference.

1. High Priority – These requirements are urgent and important. These are requirements that are generally with respect to compliance or contract that cannot be left out. These requirements need to be implemented in the current release and not implementing the same will have some adverse effect on the business.

2. Medium Priority – These requirements are important but not as urgent. Implement these after you implement the high priority items. If you see closely there is a line that splits this quadrant into 2 parts. Implement the items that are on the right side of the line first as they are relatively of higher medium priority.

3. Do these later – These items are urgent but do not have a lot of effect on the business. Hence do it after completing the more important medium priority items. Similar to the medium priority items, this quadrant has also been split into two; the items on the right side have a higher priority relatively to the items on the left.

4. Low Priority – These items are neither important nor are they urgent.

The items on the righthand side of the diagonal have higher priority. Start with the bottom-right corner of the high-priority quadrant and work your way up and left.

Prioritization of the requirements of NSTU Online Course Registration:

FR-1: Section officer adds students

This priority is high because section officer needs to add students.

FR-2: Section officer adds courses

This priority is high because section officer needs to add courses to the database.

FR-3: Section officer update courses

This priority is high because course information needs to be updated.

FR-4: Section officer delete any course from the database

This priority is high because sometimes there needs many changes in course.

FR-5: Section officer send application

This priority is high because course registration must be started.

FR-6: Education Branch sends notice

This priority is high because section officer needs to publish a notice of course registration.

FR-7: Section officer publishes the notice

This priority is high because students need to apply for their course registration.

FR-8: The section officer opens the registration process

This priority is high because apply for course registration must start with an exact date.

FR-9: Student applies for course registration

This priority is high because without apply for course registration he can't attend to any course.

FR-10: Students can see registration status

This priority is high because he/she needs to know registration status for paying term fee to the bank.

FR-11: Course coordinator approves the registration request

This priority is high because course registration system has an order to approve registration of students.

FR-12: Hall provost approves the registration request

This priority is high because course registration system has an order to approve registration of students.

FR-13: Director/Chairman approves the registration request

This priority is high because course registration system has an order to approve registration of students.

FR-14: Student downloads payment receipt

This priority is high because he/she needs to print the payment receipt.

FR-15: Student prints payment receipt

This priority is high because he/she needs to pay term fee to the bank.

FR-16: Student pays term fee

This priority is high because he/she needs to complete his/her registration in due time.

FR-17: Section officer updates student's registration status

This priority is high because student's registration status must be updated to confirm their course registration completion.

FR-18: Admin adds department

This priority is high because admin needs to add departments.

FR-19: Admin manage departments

This priority is high because admin needs to manage department's information.

FR-20: Admin can add Faculty Members and Section Officers

This priority is high admin needs to add Faculty Members and Section Officers.

FR-21: Admin manages the Faculty Members and Section Officers

This priority is high because the admin needs to manage faculty members and section officers.

FR-22: Admin can change his/her login information

This priority is medium because it's not so important to change login information.

FR-23: Section officer can delete/update notice

This priority is medium because delete/update a notice isn't urgent but important.

FR-24: Data retrieve from cloud server

This priority is high because data is an important thing and any time users need data as urgent.

SLR-1: Data query must be faster

This priority is high because query of the data query must be faster to make registration process faster.

PAR-1: Data query must be accurate

This priority is high because without an accurate data, user cannot do anything with the registration system.

CR-1: The system will handle thousands of data

This priority is high because many users in the university may interact with the system at the same time.

RAR-1: The system must be available on 24 X 7

This priority is high because users may interact with the system at any time.

RFT-1: The system handles all user access without system errors

This priority is high because users need to log in to the system without any errors. Errors can cause any mail functionality. So, there shouldn't be any errors during log in.

MR-1: System helps to update user profile

This priority is medium because updating the user profile and information is very important.

AR-1: Application provides security mechanism

This priority is high because security mechanism in the system is very important to secure user data and registration information

EUR-1: Application must be usable for the end users

This priority is high because users will feel better using the website if they can easily use the system.

UDR-1: The system engineer documentation

This priority is medium because software development team may need to make document their works.

AR-1: Labels of mandatory fields must be bold

This priority is medium because the UI may help users to develop the useability.

SR-1: The appearance must be controllable using stylesheet file

This priority is high because the UI/UX must be designed in a professional way. For this purpose, web designer needs to have a stylesheet file to control the UI/UX of the system.

Part 4
Design and Architecture

8. Activity Diagrams

User is registered

Open NCR website

Click on drop-down box

Select user type

Enter email and password

Is email and password correct?

No → Invalid email and password

Yes

User successfully logs in

User's homepage is displayed

User logs into the system
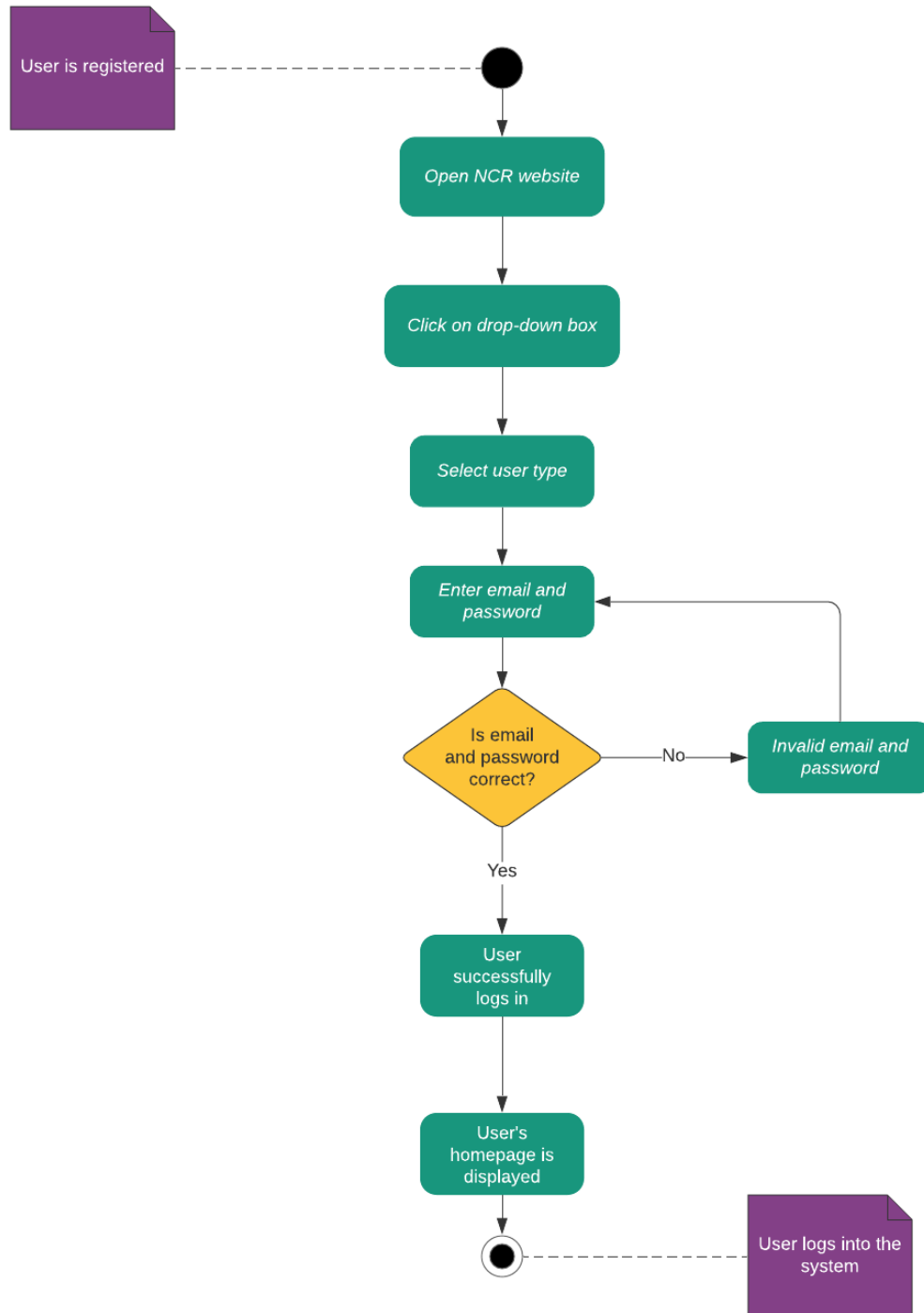
Figure 3: Login to the system

Md. Mahbub Alam | October 3, 2021
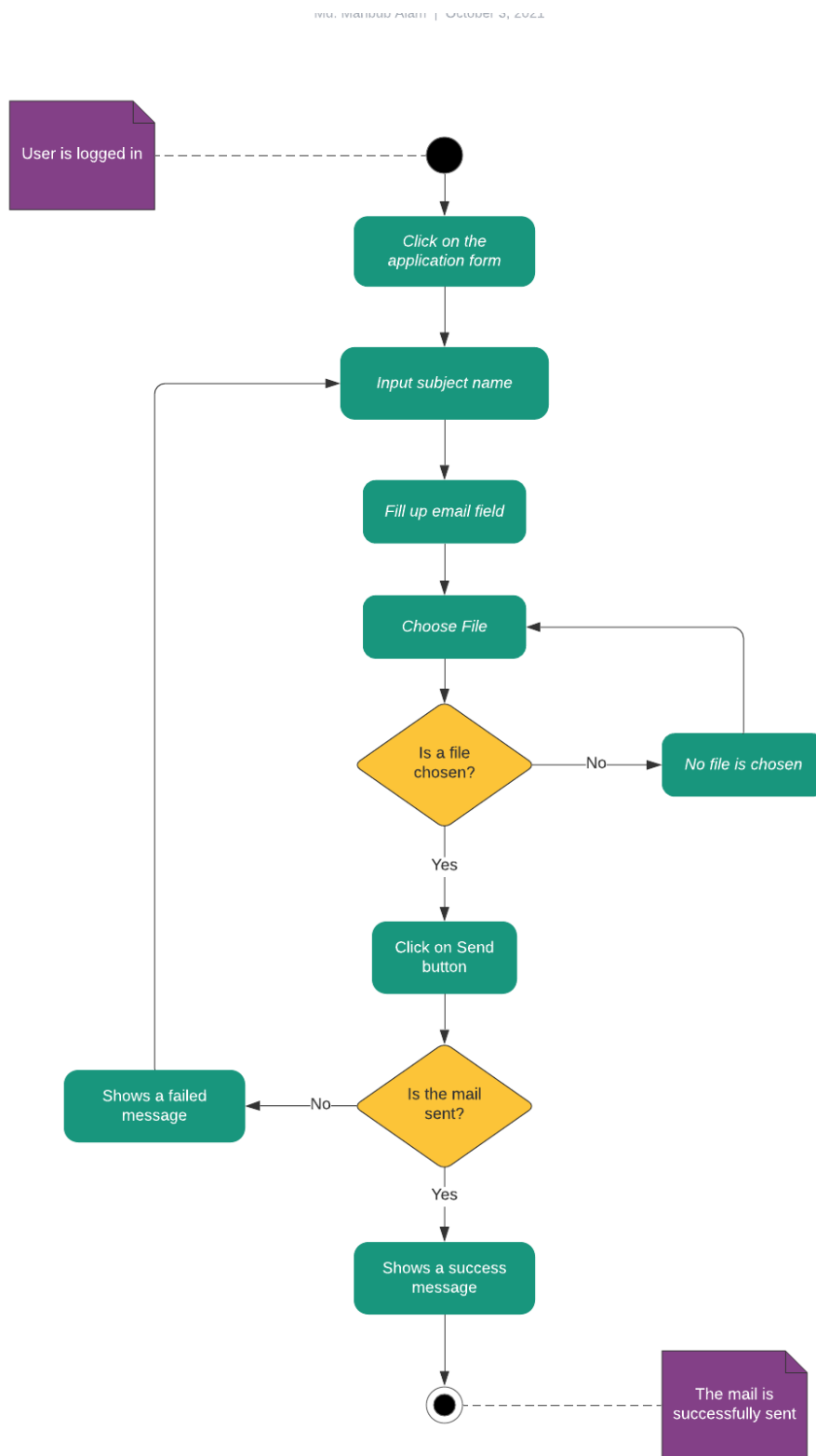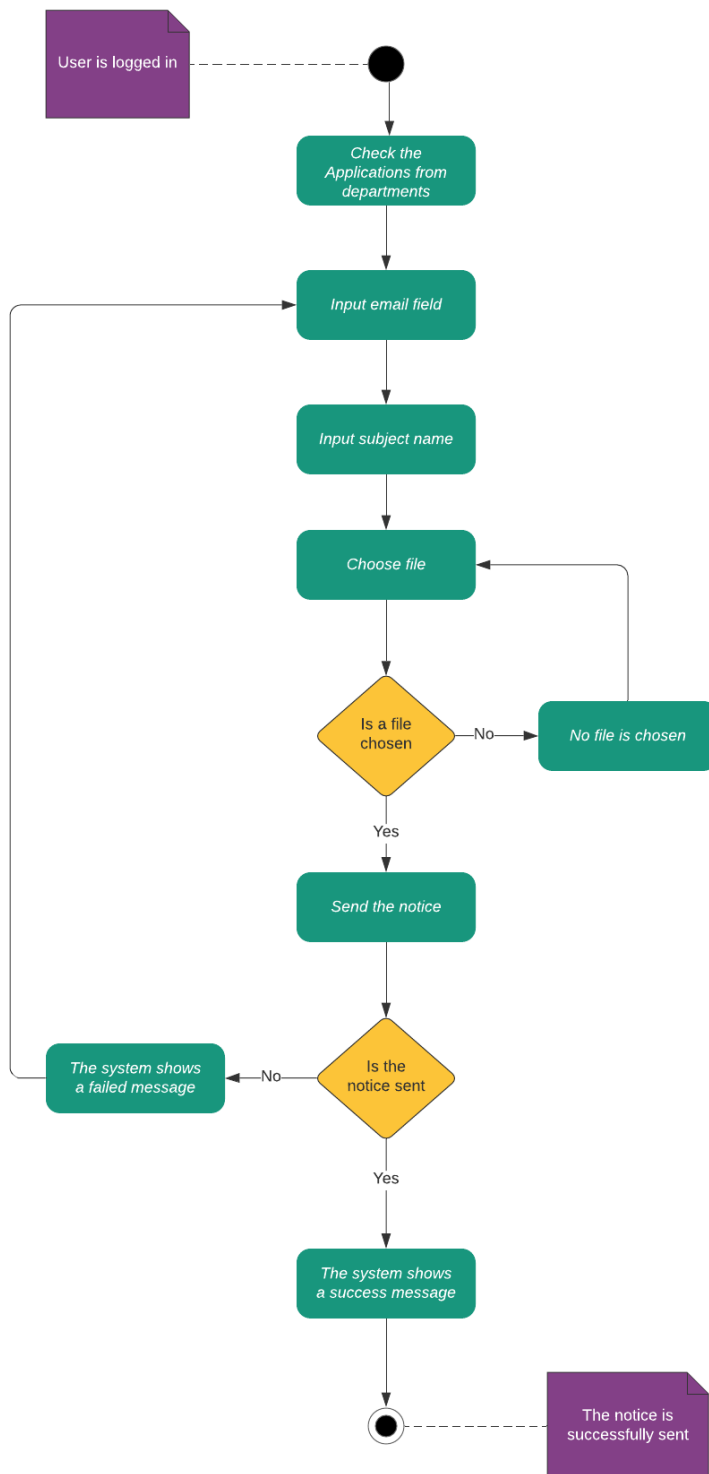

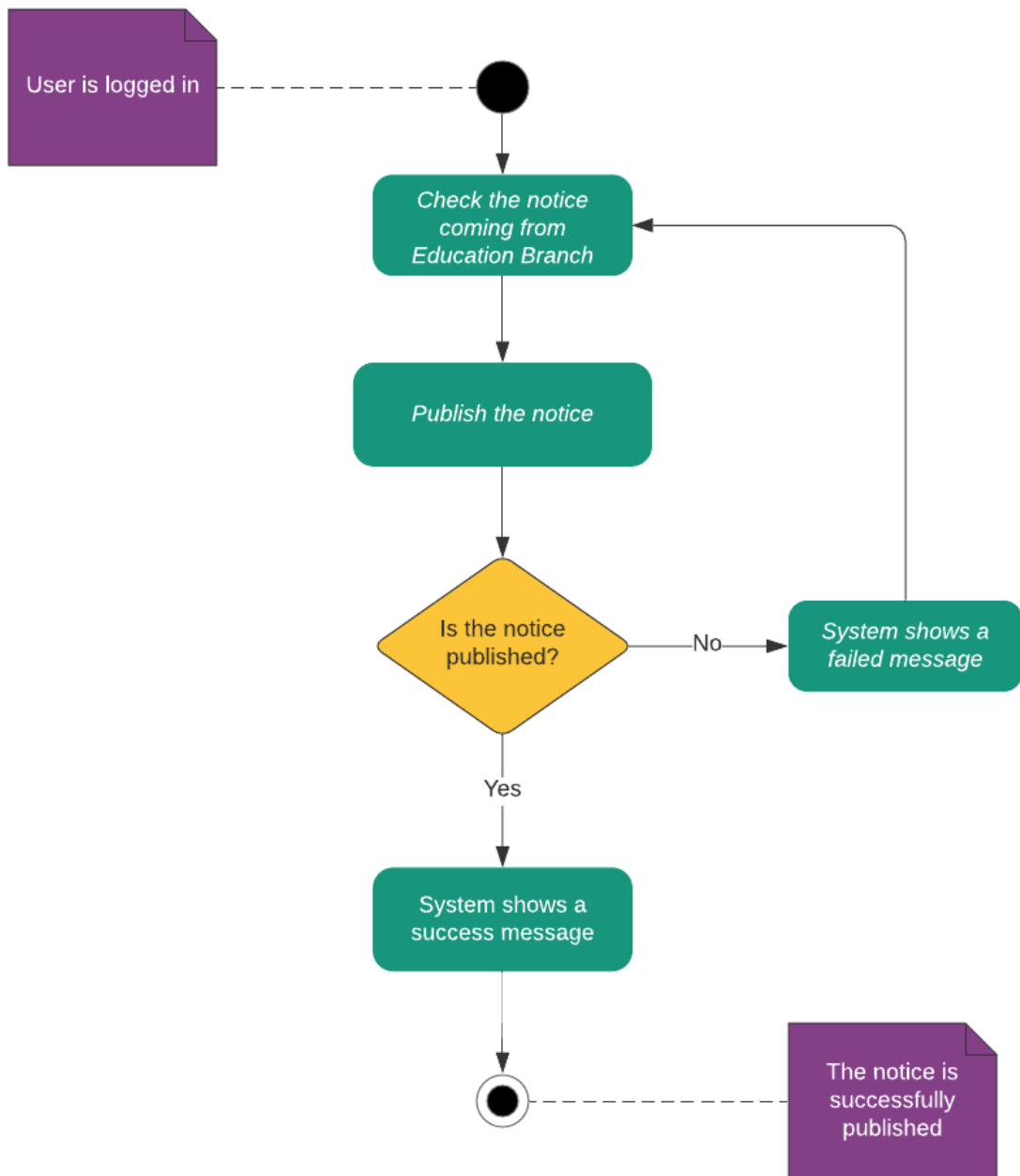*Figure 4: Send Application*

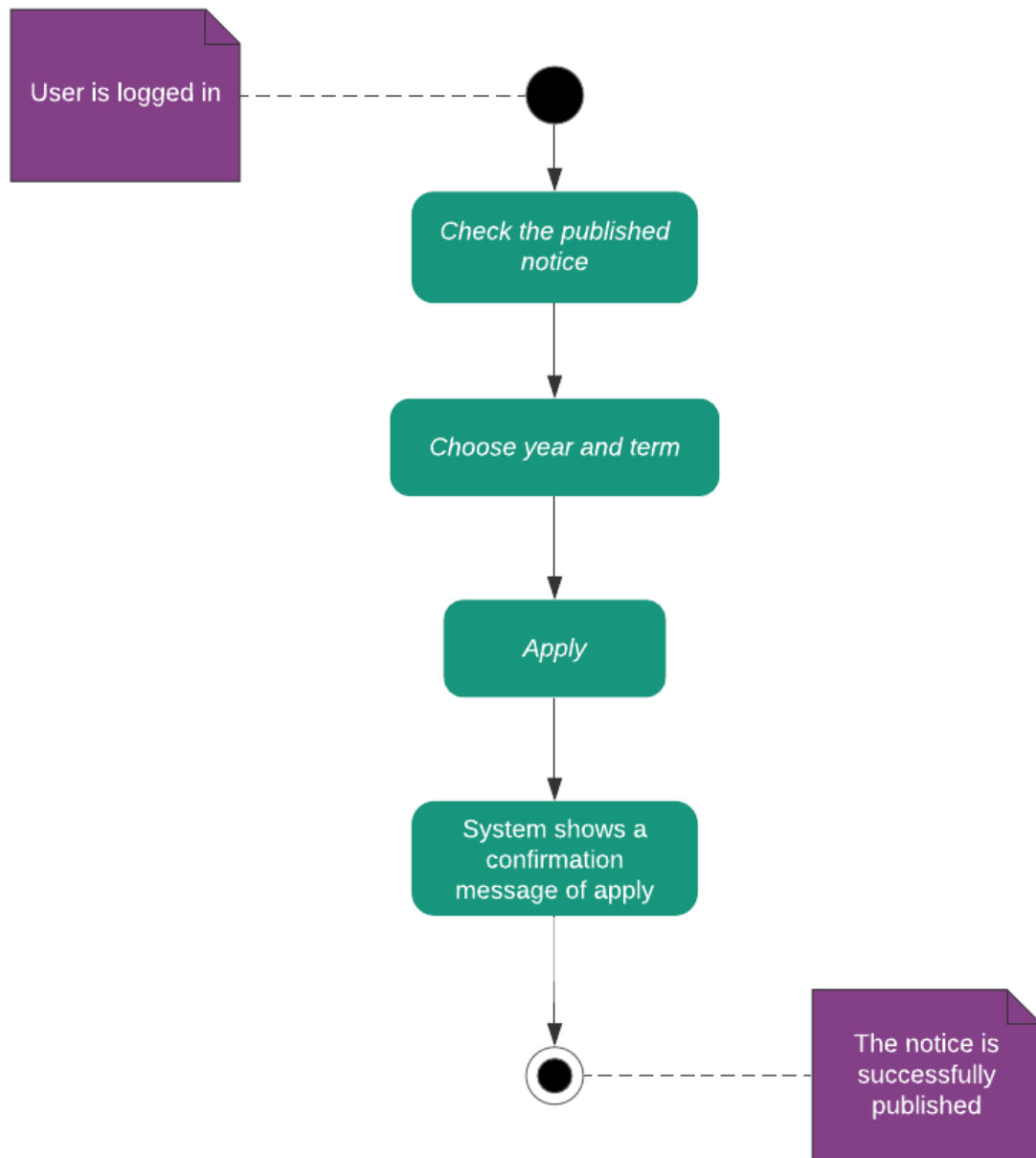*Figure 5: Send Notice*

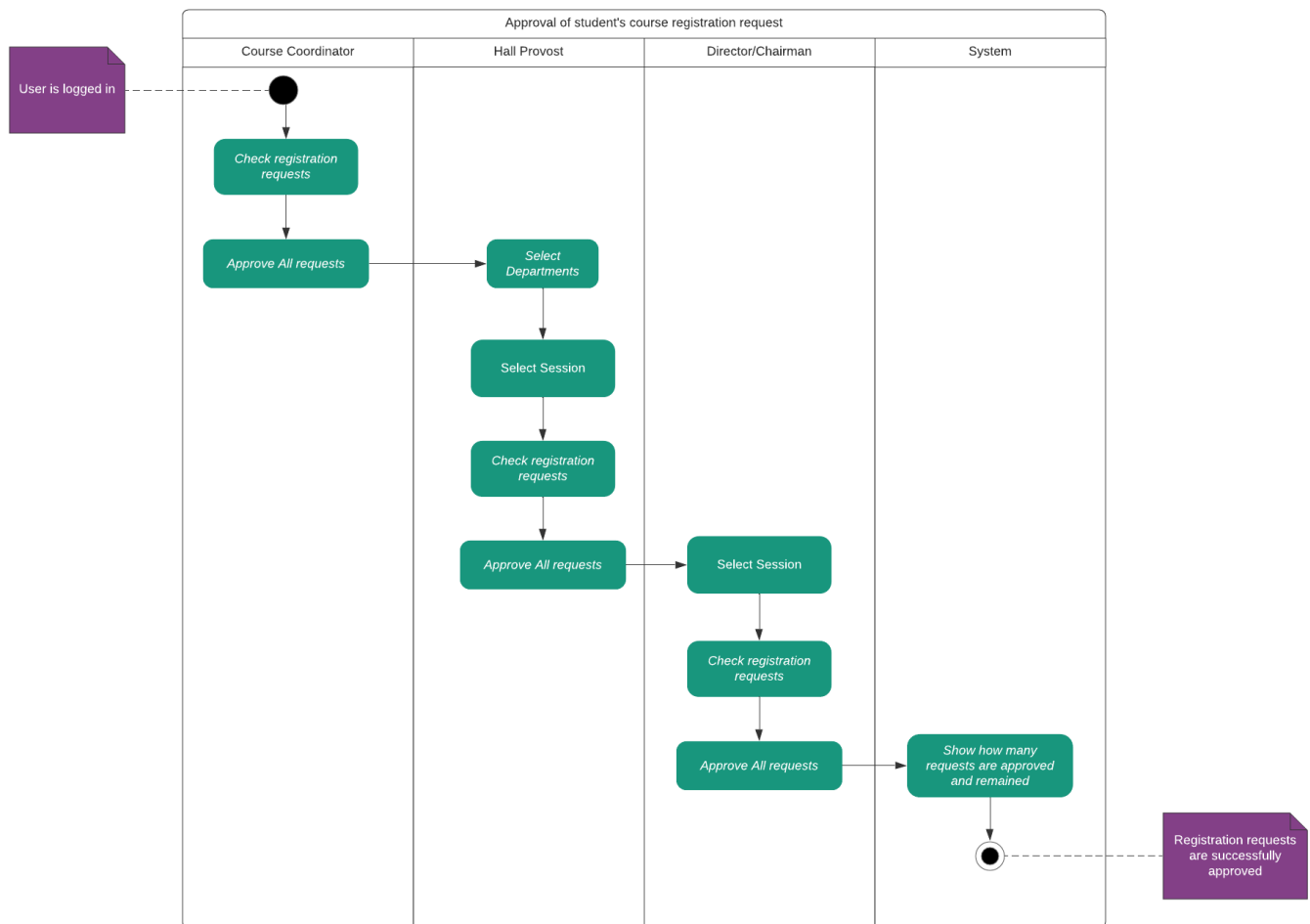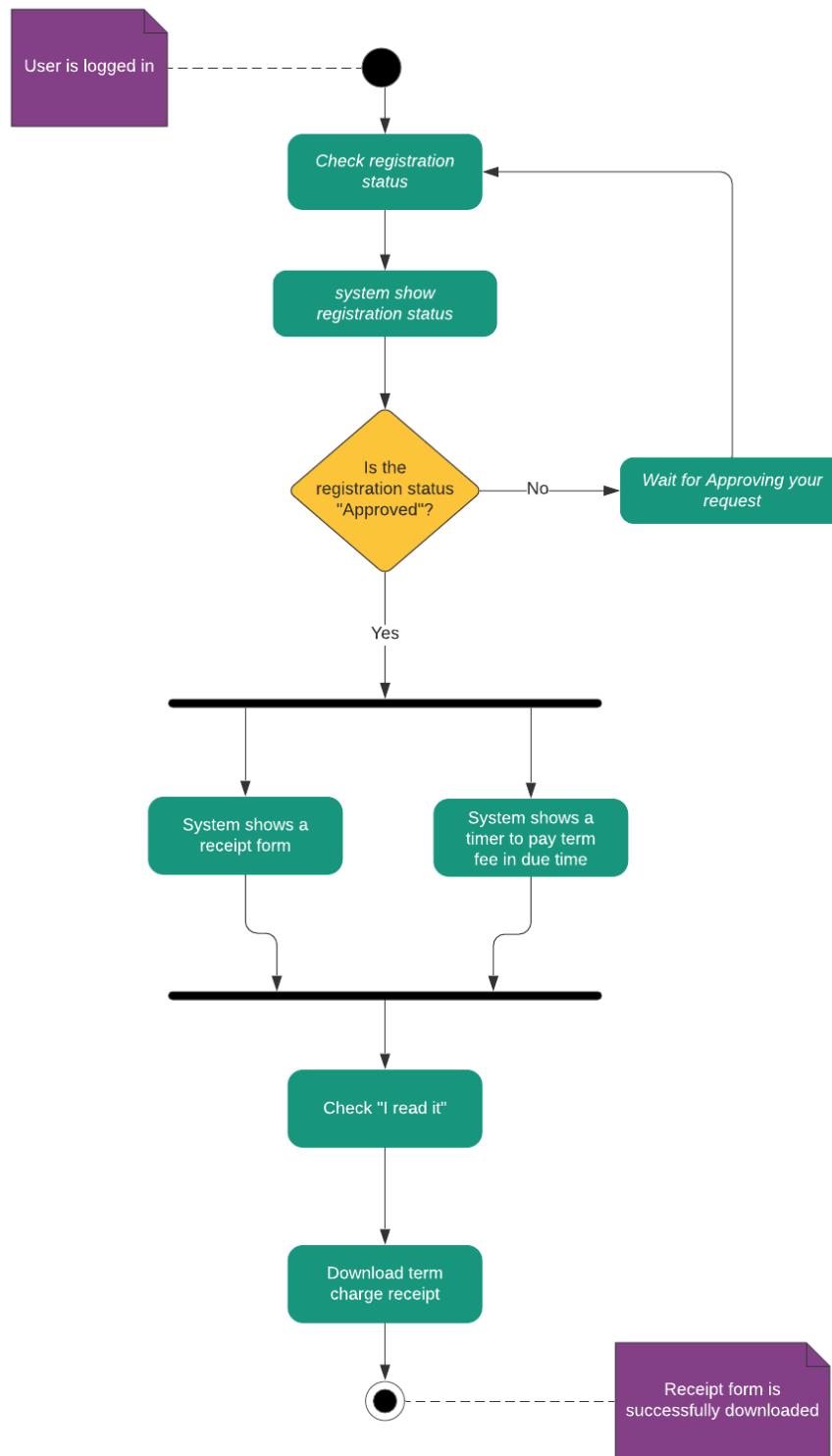*Figure 6: Publish Notice*

Figure 7: Apply for Registration

Figure 8: Approve Registration Requests

Figure 9: Download Receipt
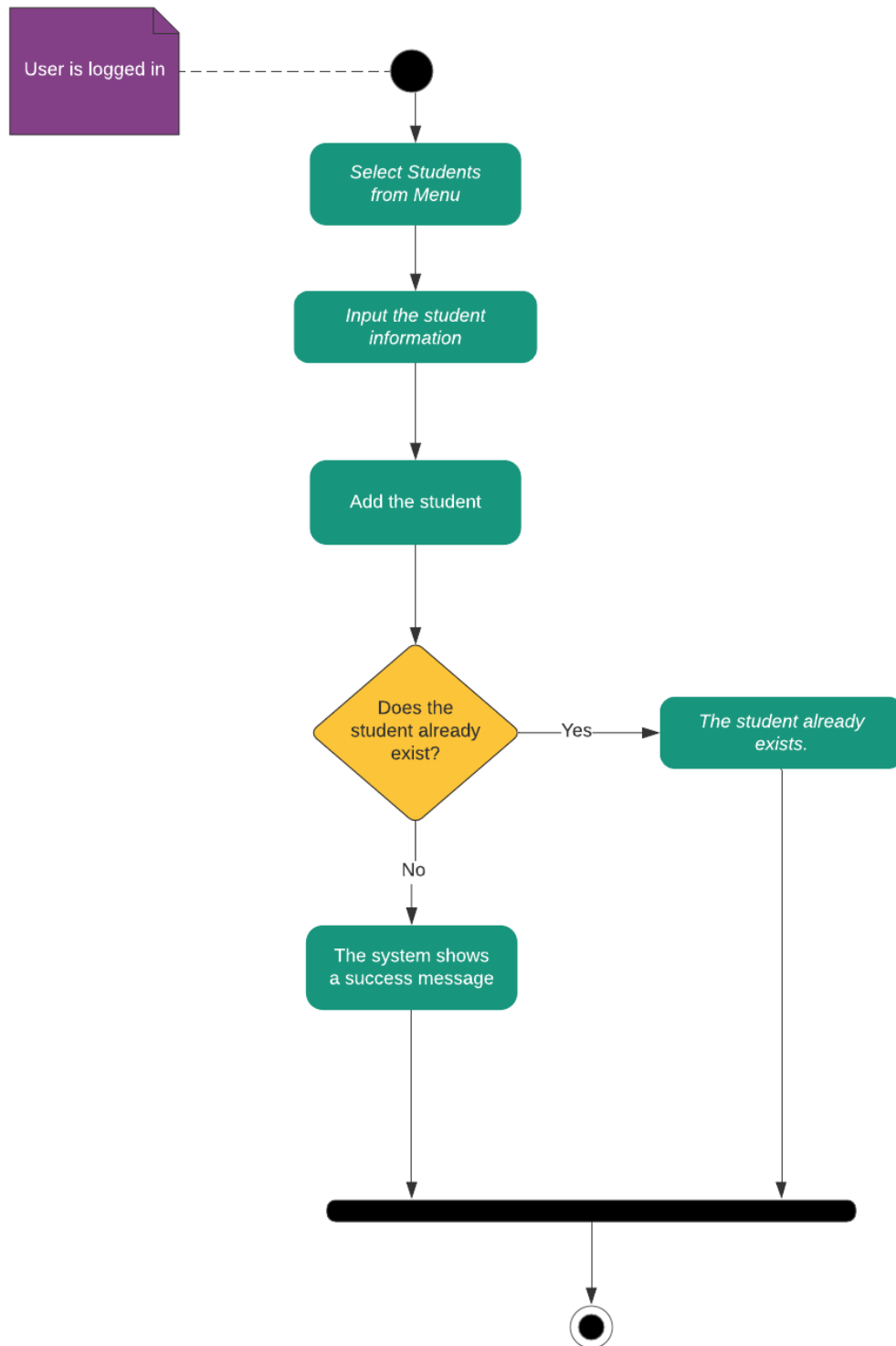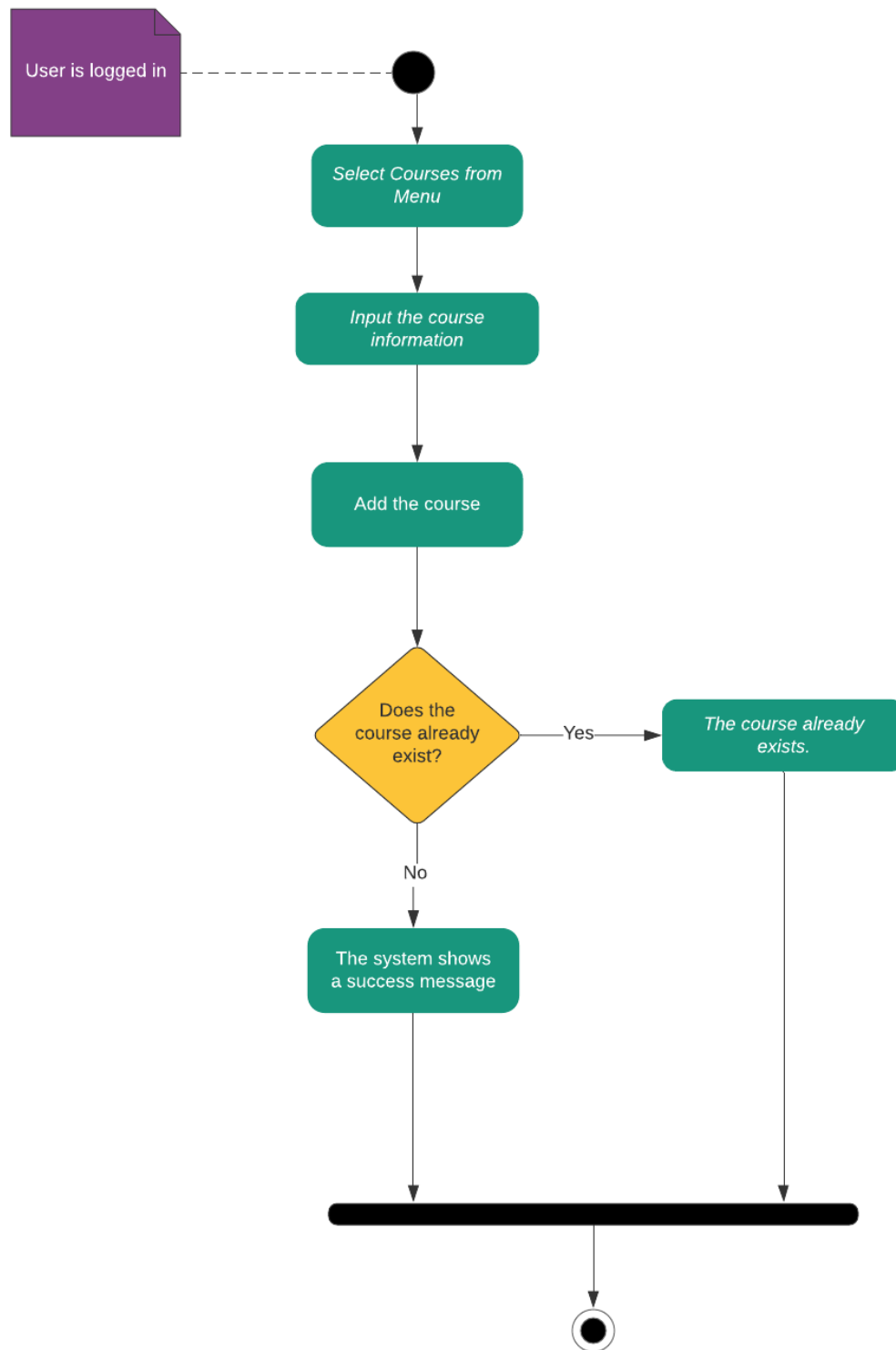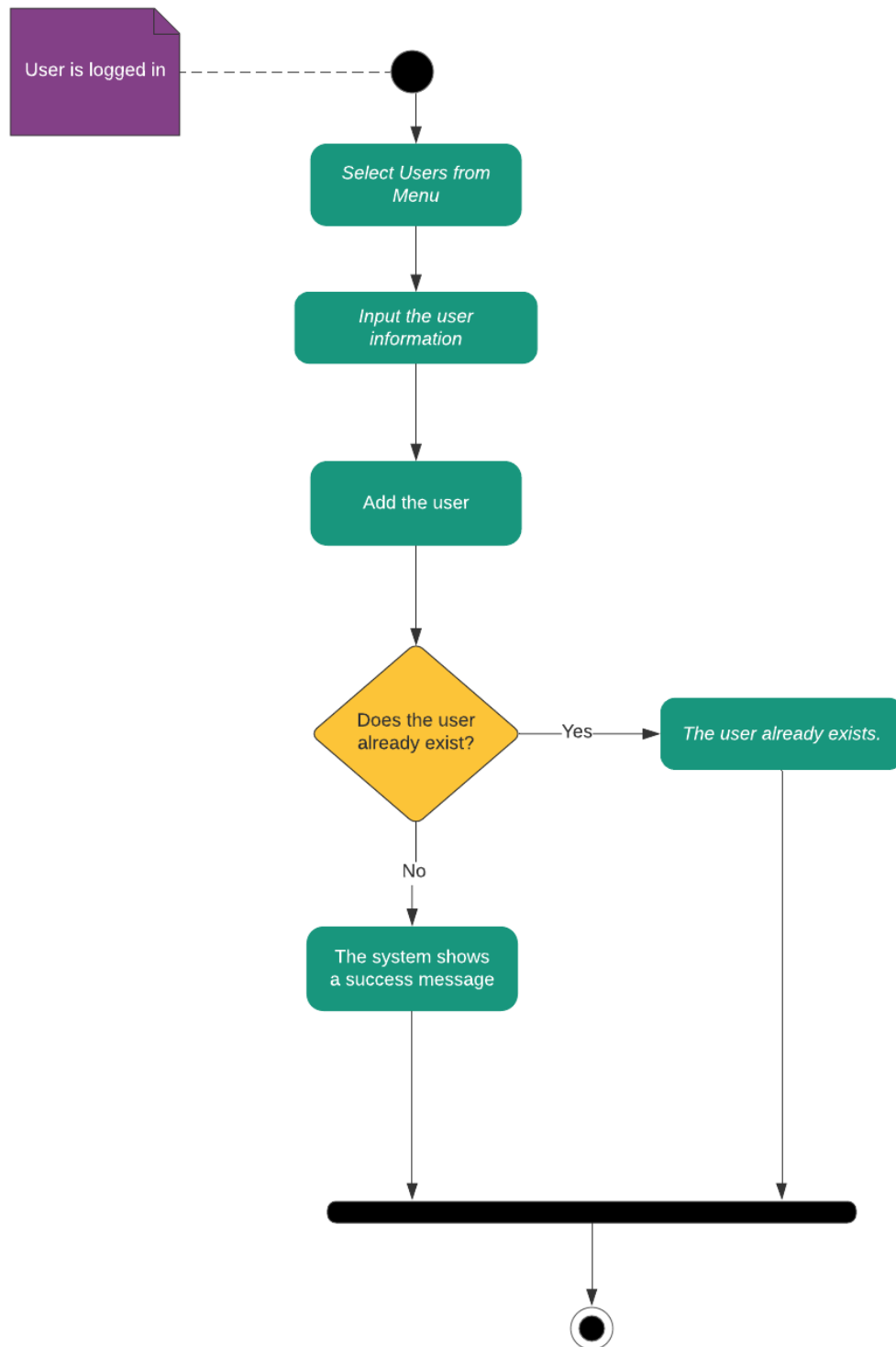
Figure 11: Add Course
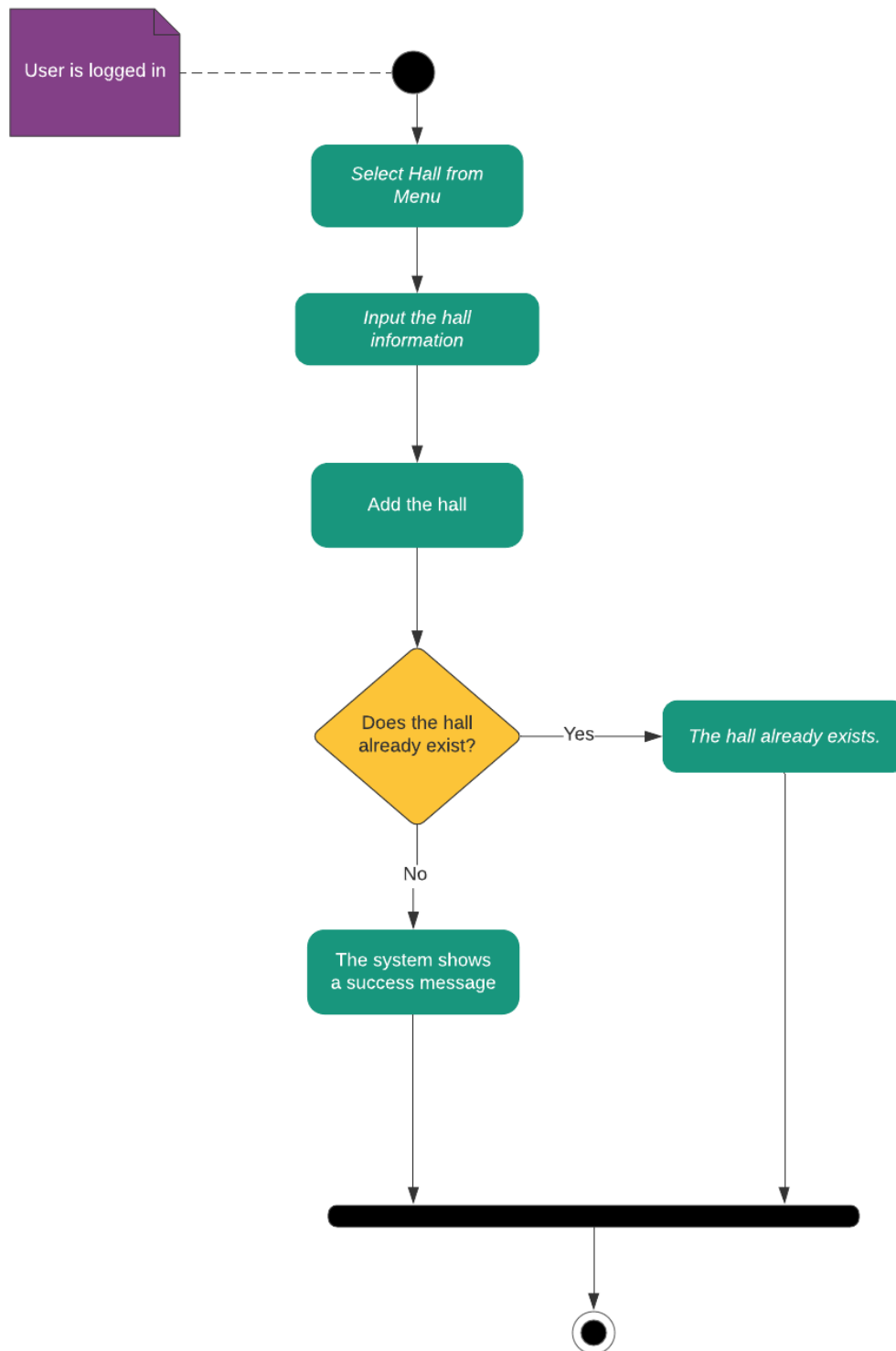
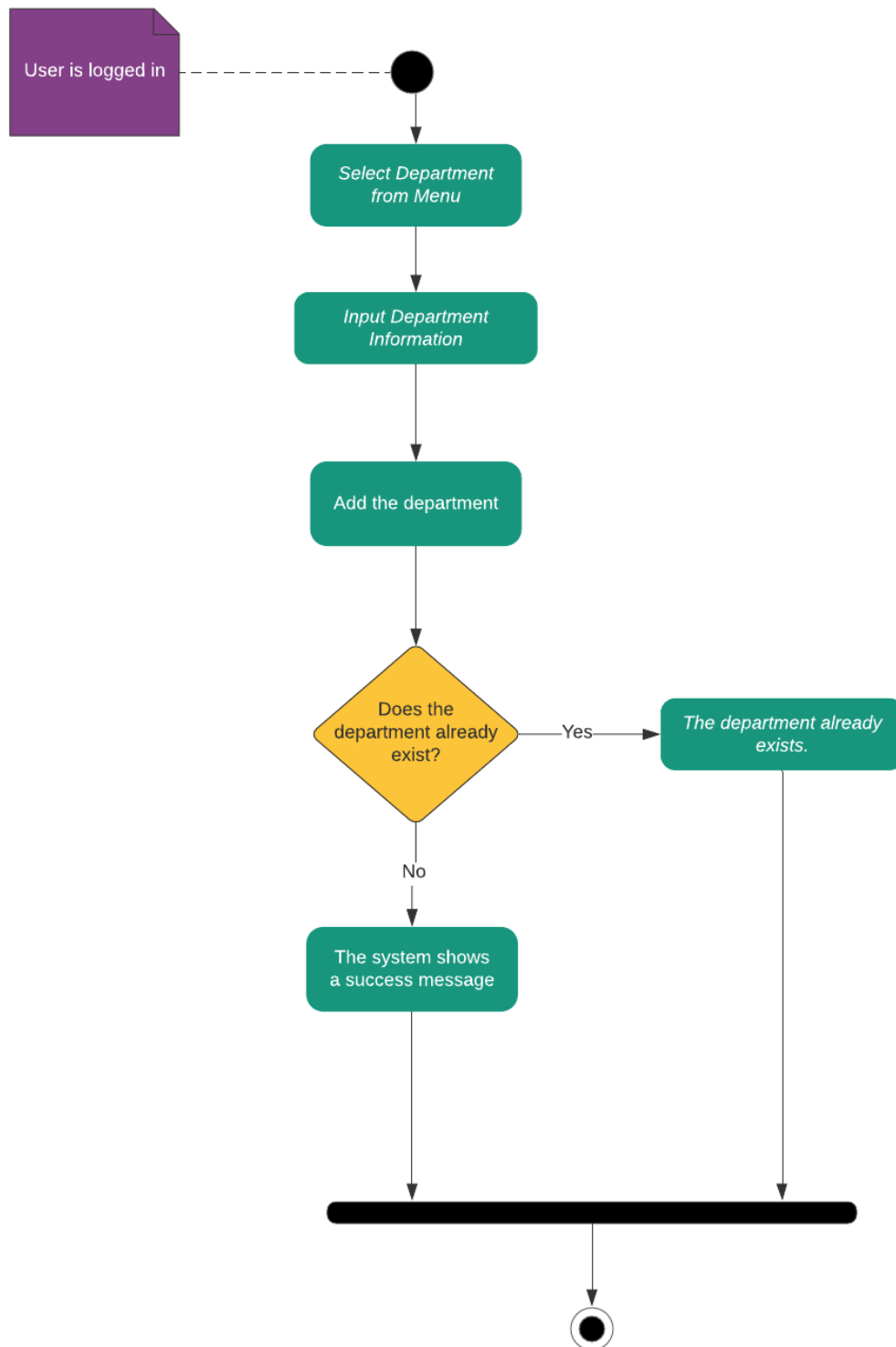*Figure 12: Add Faculty Members*

Figure 13: Add Hall
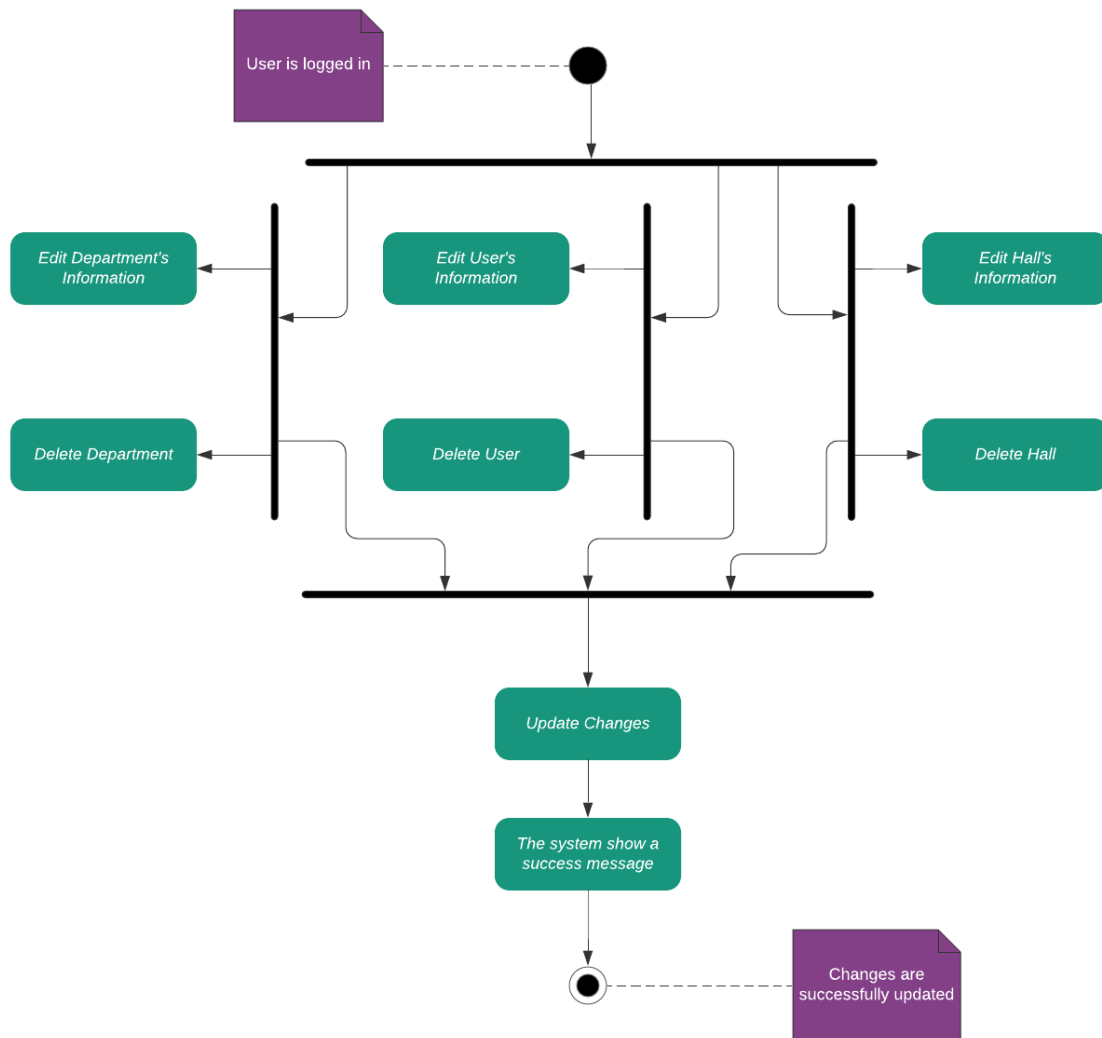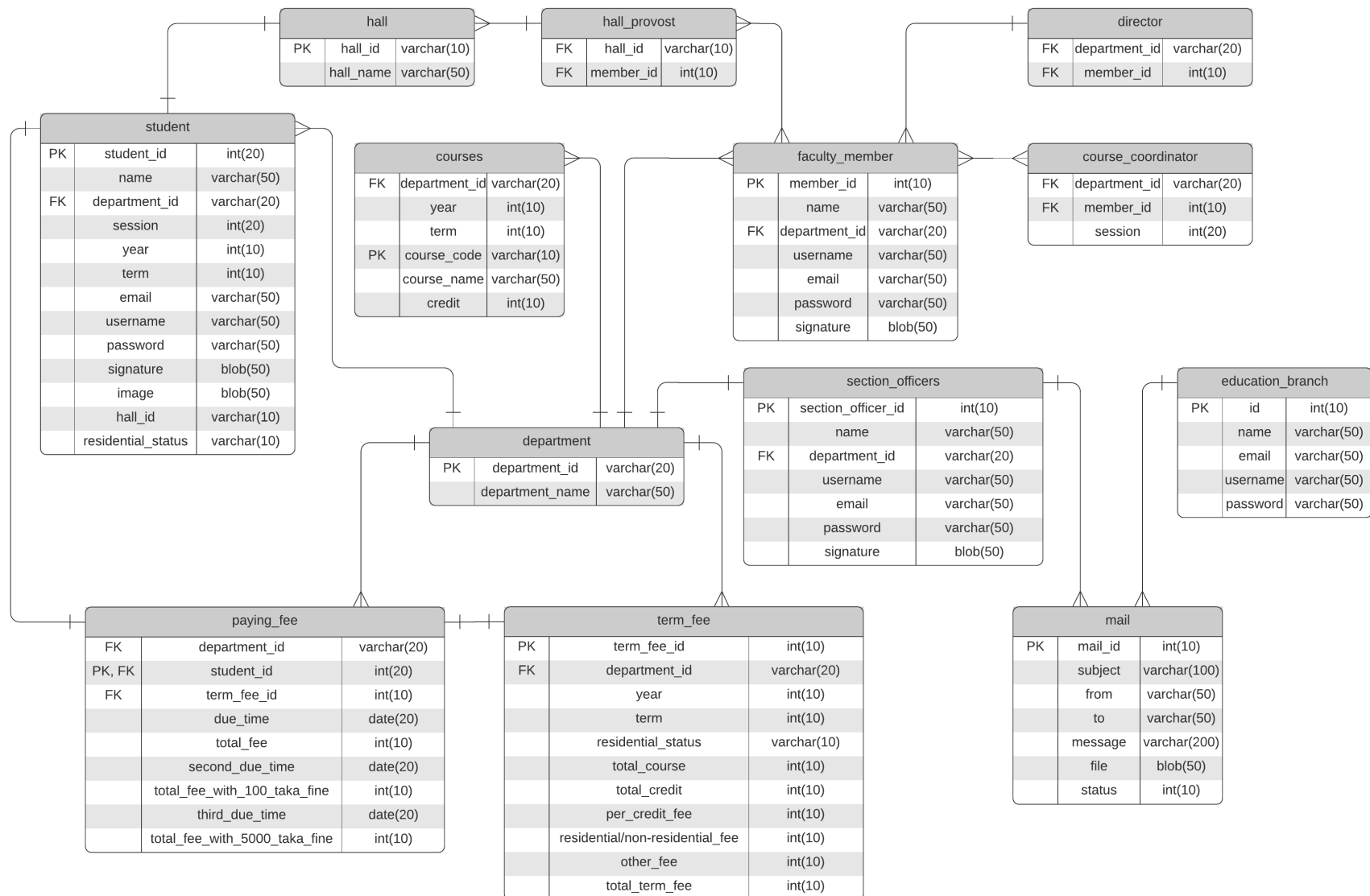
Figure 14: Add Department

Figure 15: Manage Users and the System

9. Relational Database Schema

Relational Database Schema
of NSTU Online Course Registration

hall

PK	hall_id	varchar(10)
	hall_name	varchar(50)

hall_provost

FK	hall_id	varchar(10)
FK	member_id	int(10)

director

FK	department_id	varchar(20)
FK	member_id	int(10)

student

PK	student_id	int(20)
	name	varchar(50)
FK	department_id	varchar(20)
	session	int(20)
	year	int(10)
	term	int(10)
	email	varchar(50)
	username	varchar(50)
	password	varchar(50)
	signature	blob(50)
	image	blob(50)
	hall_id	varchar(10)
	residential_status	varchar(10)

courses

FK	department_id	varchar(20)
	year	int(10)
	term	int(10)
PK	course_code	varchar(10)
	course_name	varchar(50)
	credit	int(10)

faculty_member

PK	member_id	int(10)
	name	varchar(50)
FK	department_id	varchar(20)
	username	varchar(50)
	email	varchar(50)
	password	varchar(50)
	signature	blob(50)

course_coordinator

FK	department_id	varchar(20)
FK	member_id	int(10)
	session	int(20)

section_officers

PK	section_officer_id	int(10)
	name	varchar(50)
FK	department_id	varchar(20)
	username	varchar(50)
	email	varchar(50)
	password	varchar(50)
	signature	blob(50)

education_branch

PK	id	int(10)
	name	varchar(50)
	email	varchar(50)
	username	varchar(50)
	password	varchar(50)

department

PK	department_id	varchar(20)
	department_name	varchar(50)

paying_fee

FK	department_id	varchar(20)
PK, FK	student_id	int(20)
FK	term_fee_id	int(10)
	due_time	date(20)
	total_fee	int(10)
	second_due_time	date(20)
	total_fee_with_100_taka_fine	int(10)
	third_due_time	date(20)
	total_fee_with_5000_taka_fine	int(10)

term_fee

PK	term_fee_id	int(10)
FK	department_id	varchar(20)
	year	int(10)
	term	int(10)
	residential_status	varchar(10)
	total_course	int(10)
	total_credit	int(10)
	per_credit_fee	int(10)
	residential/non-residential_fee	int(10)
	other_fee	int(10)
	total_term_fee	int(10)

mail

PK	mail_id	int(10)
	subject	varchar(100)
	from	varchar(50)
	to	varchar(50)
	message	varchar(200)
	file	blob(50)
	status	int(10)

References

1. **SRS Template**

    IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

2. **Diagrams**

    www.lucidchart.com

3. **Online Communication**

    https://meet.google.com

    https://www.messenger.com/

4. **Coding problem solutions**

    www.stackoverflow.com