ASSIGNMENT 1 FRONT SHEET

Qualification	**BTEC Level 5 HND Diploma in Business**		
Unit number and title	**Unit 30: Application Development**		
Submission date	6/10/2021	**Date Received 1st submission**	
Re-submission Date		**Date Received 2nd submission**	
Student Name	Nguyen Vu Thai	**Student ID**	GCH190530
Class	GCH0803	**Assessor name**	Le Viet Bach
Student declaration I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice.			
		Student's signature	Thai

Grading grid

P1	P2	P3	M1	M2	D1

☼ Summative Feedback:		☼ Resubmission Feedback:
Grade:	**Assessor Signature:**	**Date:**
Internal Verifier's Comments:		
Signature & Date:		

Software Requirements SpecificationS

for

<Training Online System Project>

Version <1.0>

Prepared by

Group Name: <Group 2>

Lương Ngọc Thái	**GCH190685**	**thailngch190685@fpt.edu.vn**
Ngô Tuấn Anh	**GCH190543**	**anhntgch190543@fpt.edu.vn**
Nguyễn Vũ Thái	**GCH190530**	**thainvgch190530@fpt.edu.vn**

Instructor: **Lê Việt Bách**

Course: **Application Development**

Lab Section: **<place your lab section here>**

Teaching Assistant: **<place your TA's name here>**

Date: **06/10/2021**

Contents

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Draft Type and Number	Full Name	Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded.	00/00/00

1 Introduction

1. Introduction

Today's technology is developing more and more, FPT Co wants to build a learning environment in the corporation, it is necessary to build a management system for training activities. In this report, we will present the plan. with which we will develop this project, the designs and risks or requirements of the customer

1.1 Document Purpose

The purpose of this document is for building an online training system to manage the company's internal training activities in order to easily manage courses as well as trainees. More specifically, the purpose of this SRS document will also be to provide a detailed overview of our website, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our employees see the course and its functionality. Just like the trainer can see the course he teaches.

1.2 Product Scope

This website will be the website that manages the “Training” activities for the company's internal training programs. These websites will be designed so that trainees can view and register for courses. Furthermore, this system is designed so that training staff can manage (Add, edit, delete...) trainee accounts and manage (Add, edit, delete...) existing courses. The website also contains a relational database containing lists of courses, students, and coaches.

1.3 Intended Audience and Document Overview

Intended Audience

All staff responsible for website implementation, stakeholders - and users of the website as (trainees, trainers, training staff) - should read and understand this document.

Document Overview

The remaining sections of this document provide a general description, including characteristics of the users of this project, the product's hardware, and the functional and data requirements of the product.

- General description of the project is discussed in section 2 of this document.
- Section 3 gives the functional requirements, while designing the website. It also gives the use-case model of the product. Section 3 also gives the specific requirements of the product. Section 3 also discusses the external interface requirements and gives detailed description of functional requirements.
- Section 4 is for technical design, In section 4 will show ERD and Class Diagram designs. Section 4 also gives an Activity Diagram and Gantt Chart for the project.
- Section 5 is a part of Risk Management

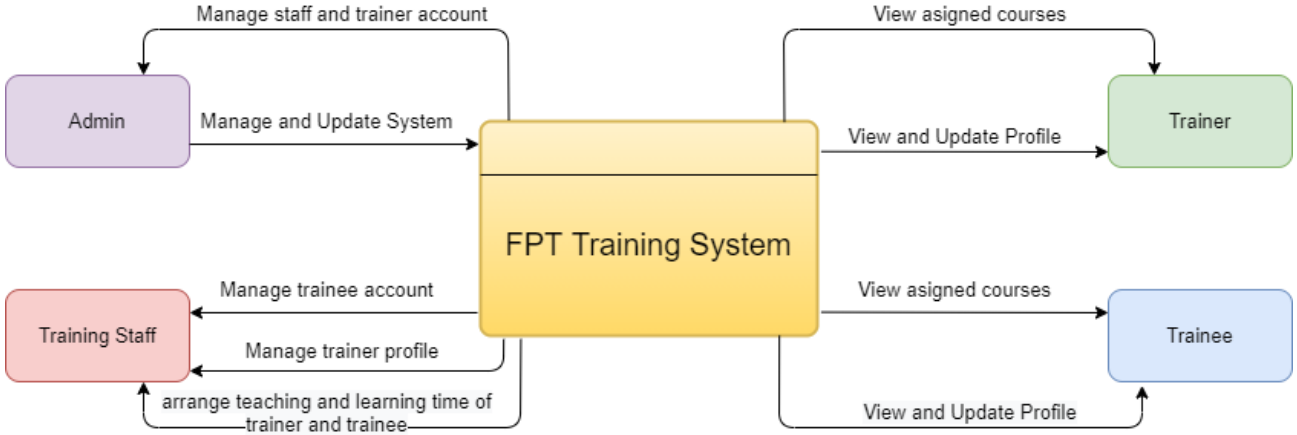
1.4 Definitions, Acronyms and Abbreviations

ERD	Entity Relationship Diagram
SRS	Software requirements specification
DB	Data Base

2 Overall Description

2.1 Product Overview

The system is used to manage training, there are 4 roles in the system including training, trainer, training staff, administrator, the roles in the system will have functions and powers. different. Administrator can edit the system about the operation of the system, the training staff can manage the trainee and trainer's information account, arrange the schedule for the system, the trainer and the trainee with less authority can only edit Personal information and course view



2.2 Product Functionality

1. Provide a bulleted list of all the major functions of the system

List of Function

Role	Function	Describe
Trainer	Login, Logout	The trainer is provided with an account to access the system to use the functions, and can exit the system if not used.
	Update profile	the trainer who can update his profile such as Trainer name, External or Internal Type, education, working place, telephone, and email address.
	View course assigned to.	trainers can see the courses they have been assigned.
Trainee	Login, Logout	The trainee is provided with an account to access the system to use the functions, and can exit the system if not used.

	Update account password	trainees can change any password they want after logging into the system
Training staff	Login, Logout	Employees will be provided with an account to access the system, and can log out at any time
	create trainee accounts	Can create trainee accounts by entering details like trainee name, trainee accounts, age, date of birth, education, main programming language, TOEIC score, experience details, department, location.
	Update, Delete trainee accounts information	Training staff will be allowed to update and delete student information accounts
	Mange courses categories	Can manage course categories such as searching, adding, updating and deleting course categories. Course category includes the information such as course category name and descriptions.

	Mange courses	Training staff can manage courses such as searching, adding, updating and deleting courses. Course includes Course Name, Course Category and Description
	Manage trainer profile	Training staff can manage trainer profile such as adding, updating and deleting the information: Trainer name, External or Internal Type, working place, telephone, and email address.
	assign trainer to courses	Staff will be responsible for managing and arranging trainers on courses
	assign trainee to topic	staff will be responsible for managing and arranging trainees into topics

	Remove courses from Trainer	Training staff can remove courses from trainer if they have ever attended the course
	Remove courses from Trainee	Training staff can remove courses from trainee if they have ever attended the course
Admin	Login, Logout	The Admin is provided with an account to access the system to use the functions, and can exit the system if not used.
	create/edit/delete new Trainer account	Admin can create, delete account for trainer
	create/edit/delete new Training Staff account	Admin can may create, delete account for training staff

2.3 Design and Implementation Constraints

To ensure the highest level of reliability, hardware, database servers, and application servers will be tightly and securely aligned. Microsoft's standards will be used in the framework. To support web programming, use Bootstrap, HTML 5, CSS, and Javascript. In terms of storage, storage devices should be able to store a wide range of information storage options and formats on demand. Options for web hosting and disconnection storage are required. When data is disconnected unexpectedly, archiving is performed programmatically to ensure that customer profile data is not lost and that data is retained for a long time. To ensure that information is stored securely online, a large amount of space is required. With online capacity, a large amount of space is required to ensure that information storage is not overloaded.

The system is supported by software that uses the company's standard MySQL Server database. Due to its outstanding features and good performance, the Google Chrome browser is the most popular browser today and is ideal for the project's application development and testing process. Because the web system will be the primary delivery protocol for applications, the system will require and support applications that use the HTTP server. Visual studio is a popular programming tool provided by Microsoft today, and the programming language is C#, which will be used in this project.

2.4 Assumptions and Dependencies

- Website is developed using ASP.Net MVC 5 and therefore requires using the latest version of Visual Studio.
- It was initially expected that each position in the team would provide the necessary human and financial resources to prepare the project implementation. Due to the impact of the Covid-19 epidemic, the Team did not have enough resources, resources, and minimum finance needed for the project implementation. This may result in system deployment later than scheduled.
- The system is deployed for internal use by the FPT company, so users need to log in to the system using the email account provided by FPT for each user role.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

Login interface for all roles

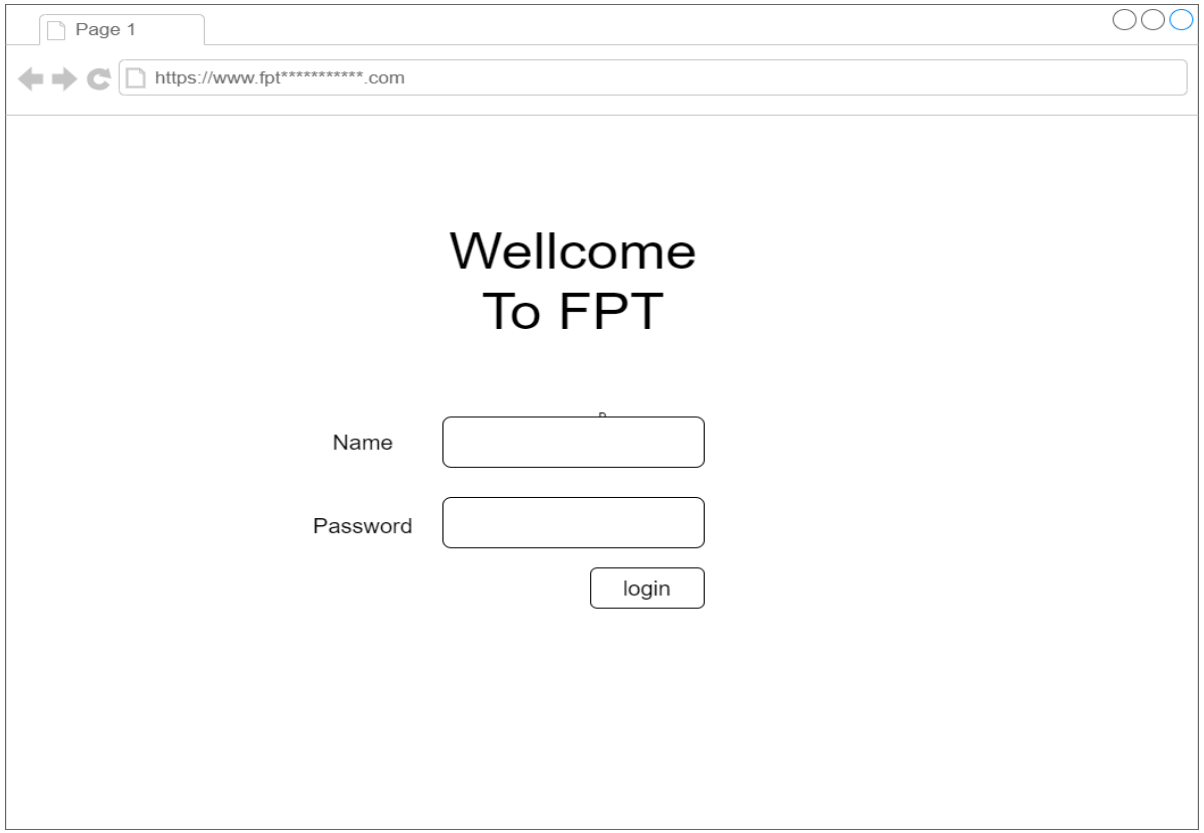


Figure 1: Login Interface

Admin's interface

Admin can create an account for the trainer, fill in the account information

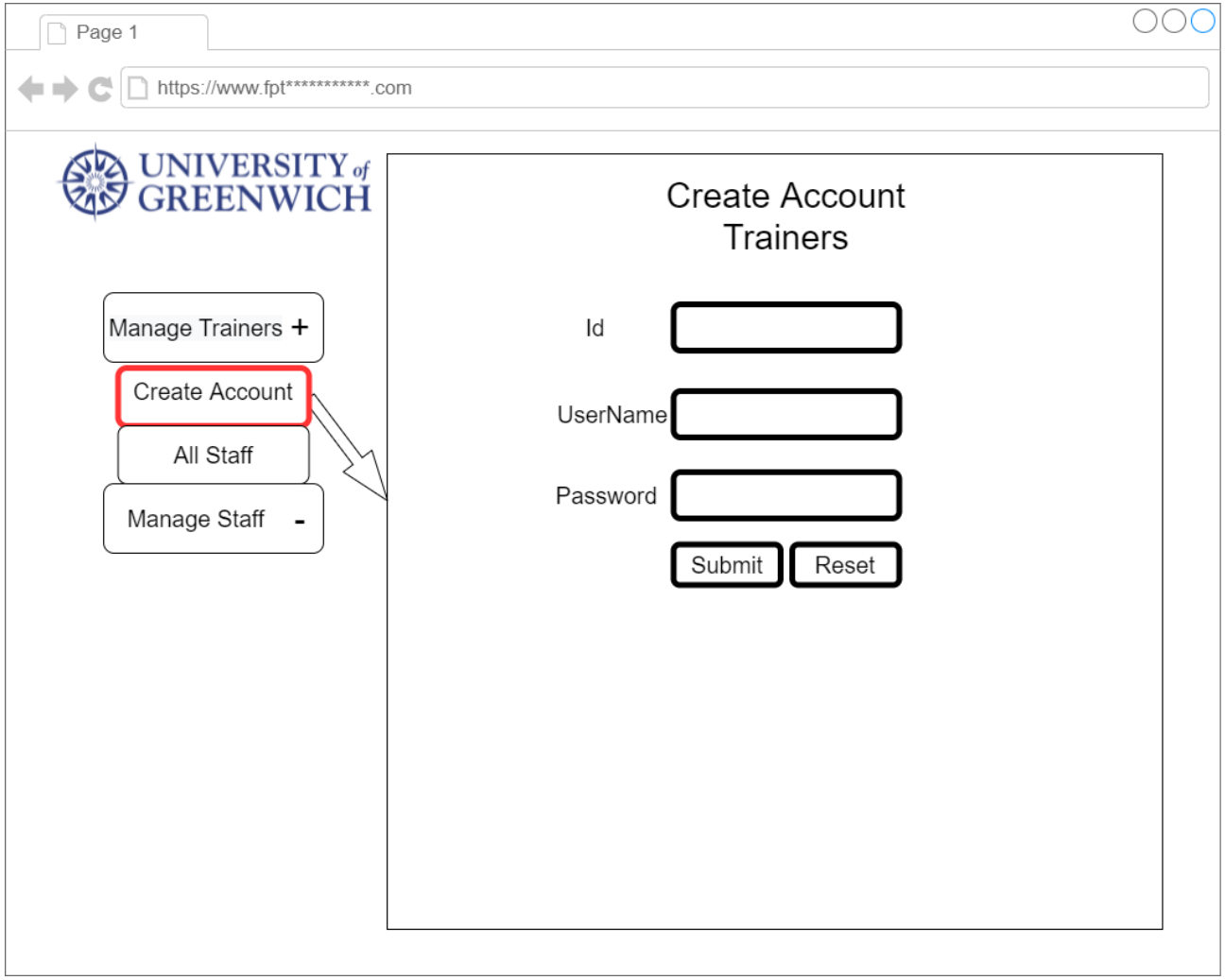


Figure 2: Admin Interface for creating trainer's account

.

Admin can view all possible accounts, edit or delete accounts.

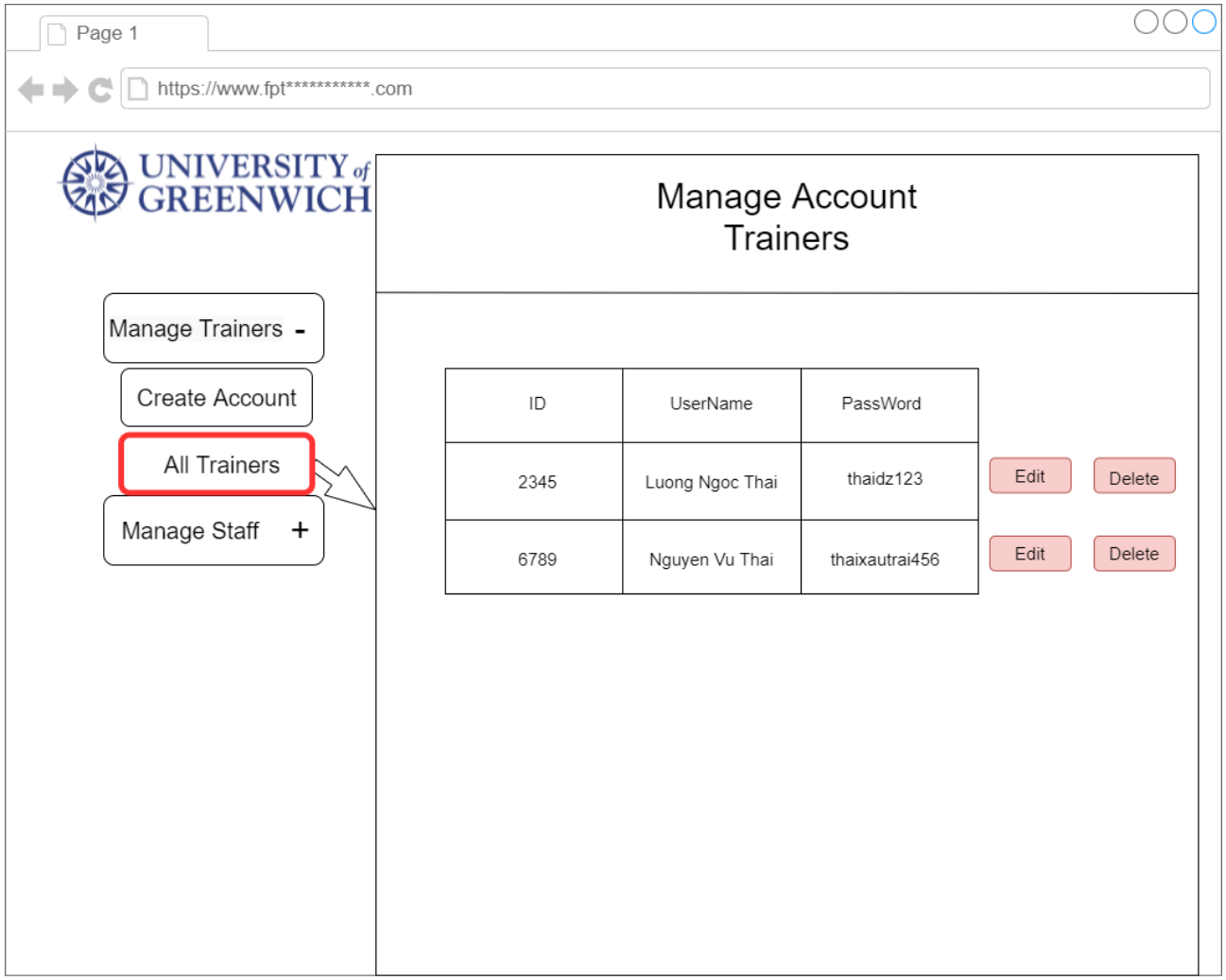


Figure 3: Admin interface for managing trainer's account

Admin can create an account for the Staff, fill in the account information.

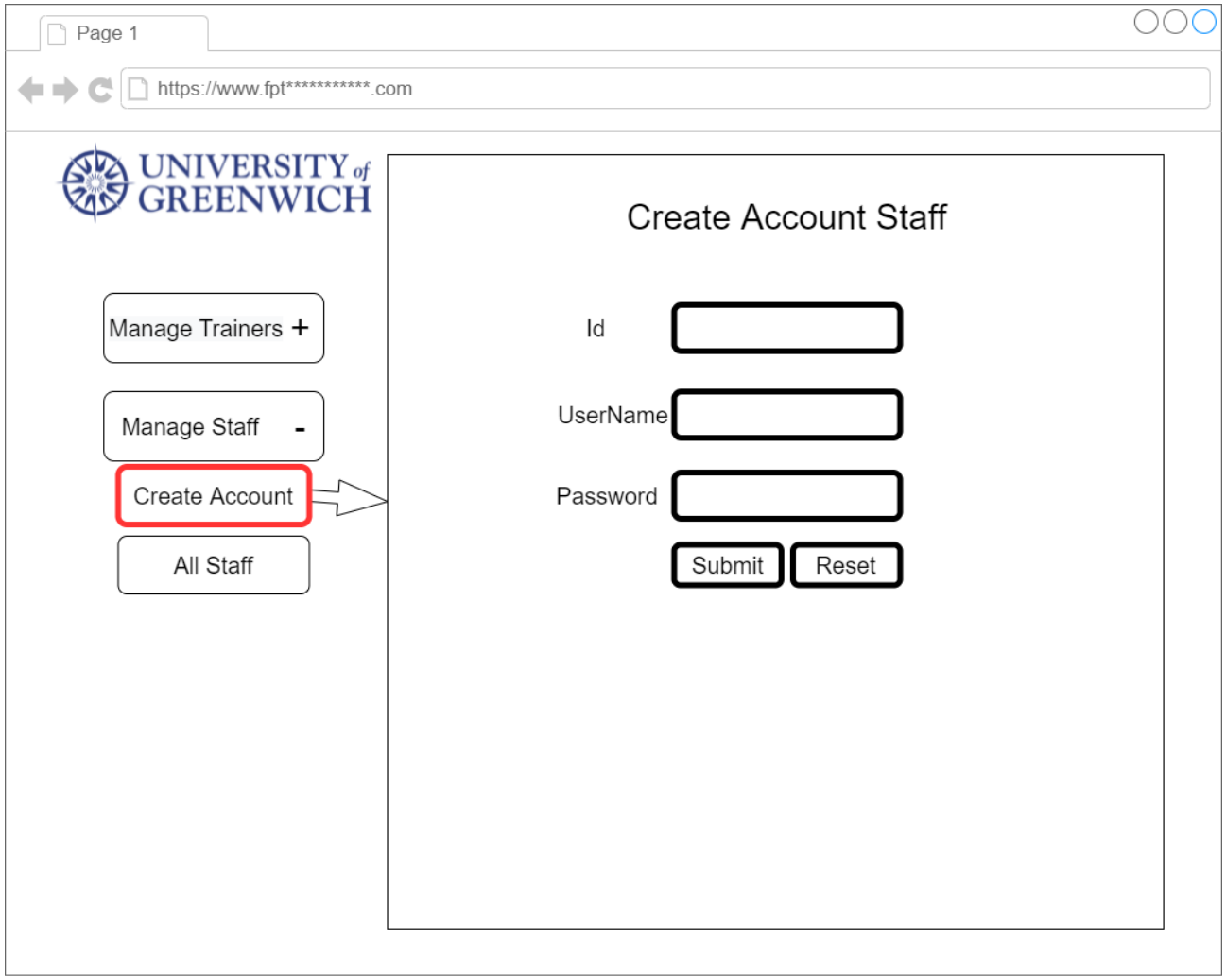


Figure 4: Admin Interface for Creating Staff Account

Admin can view all possible accounts, edit or delete accounts.

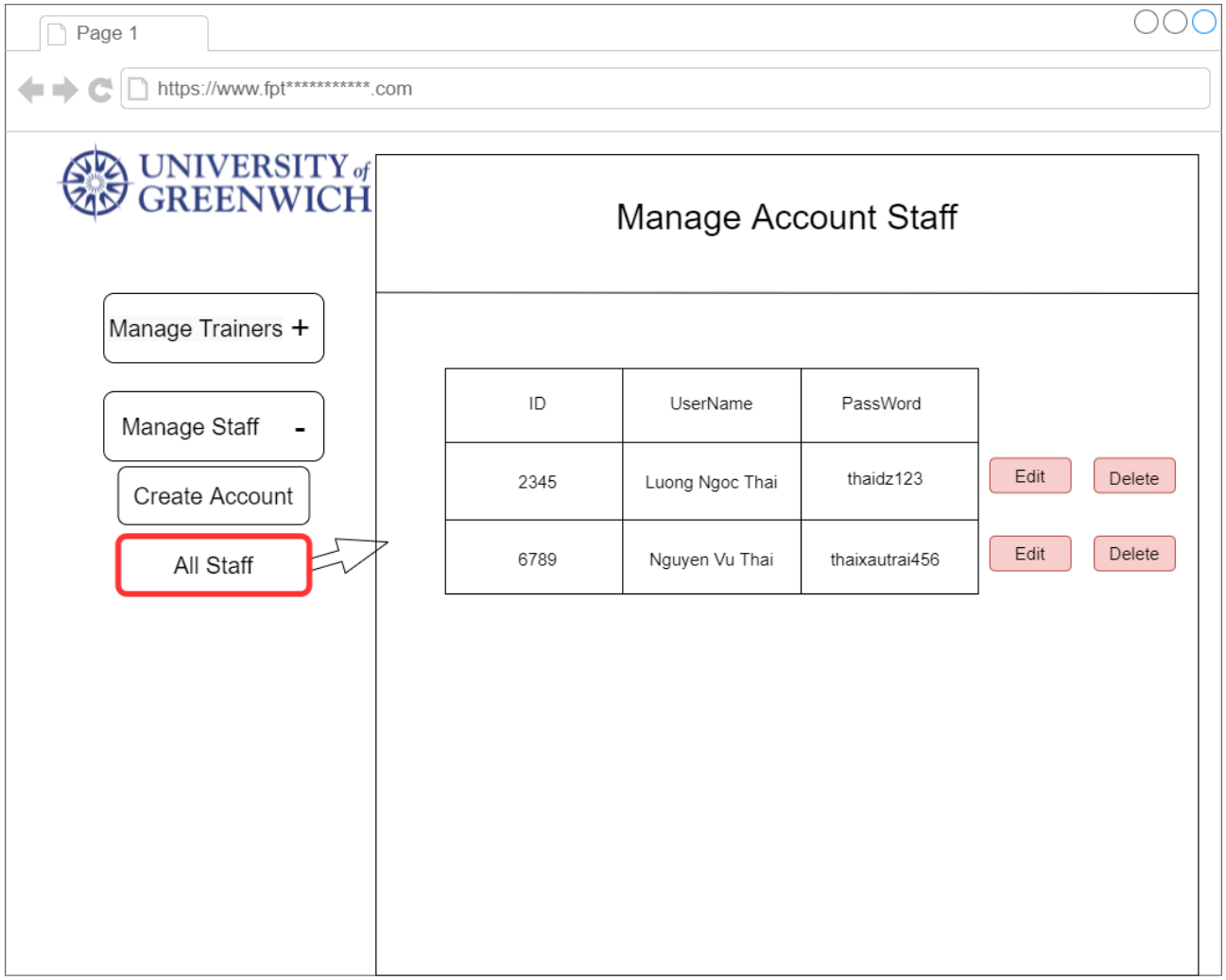


Figure 5: Admin interface for Managing Staff Account

Training Staff Interface

Staff can manage Trainer list, can Add Trainer, Edit and Delete Trainer.

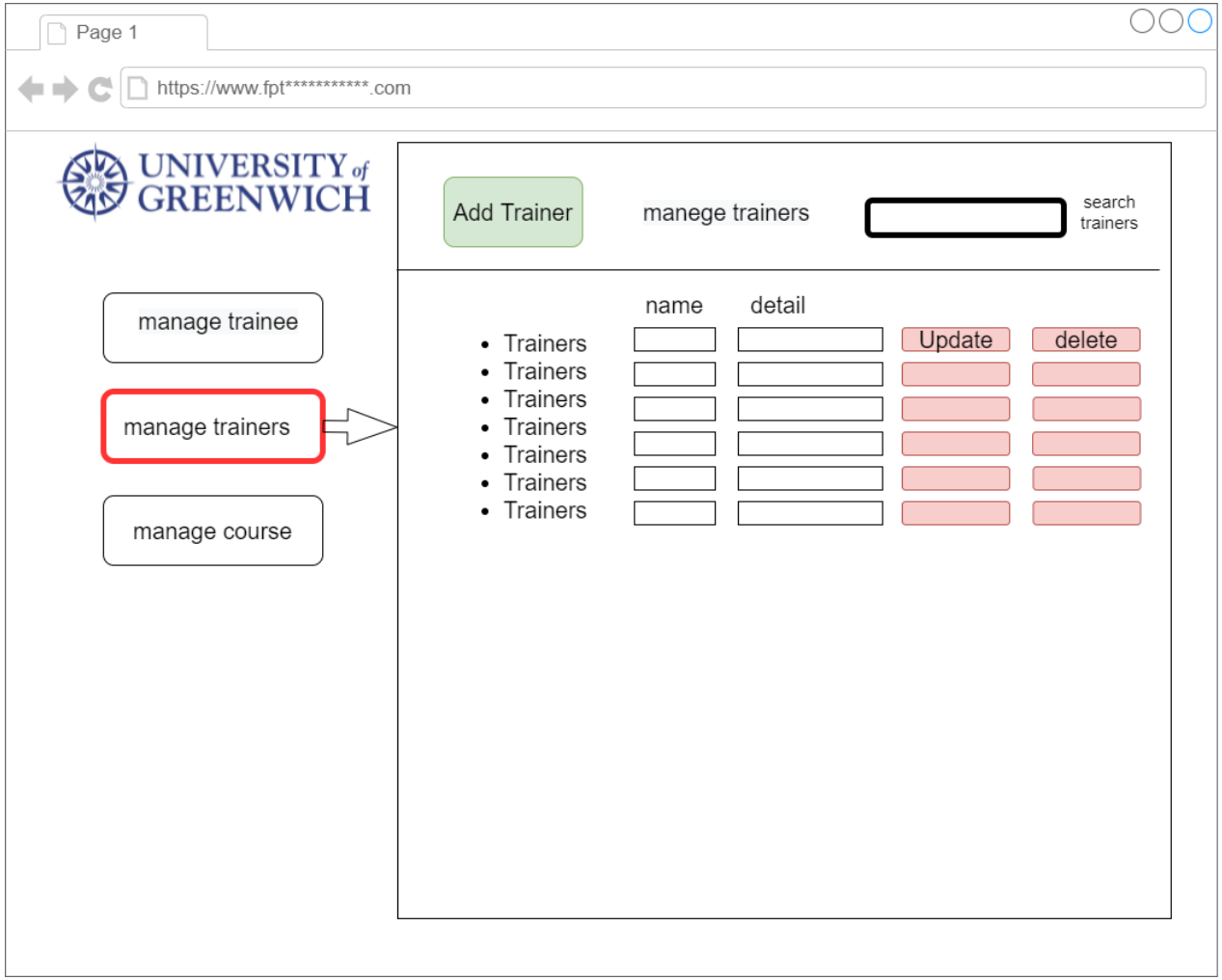


Figure 6: Staff Interface for Managing Trainers

Staff can manage Trainee list, can Add Trainer, Edit and Delete Trainee

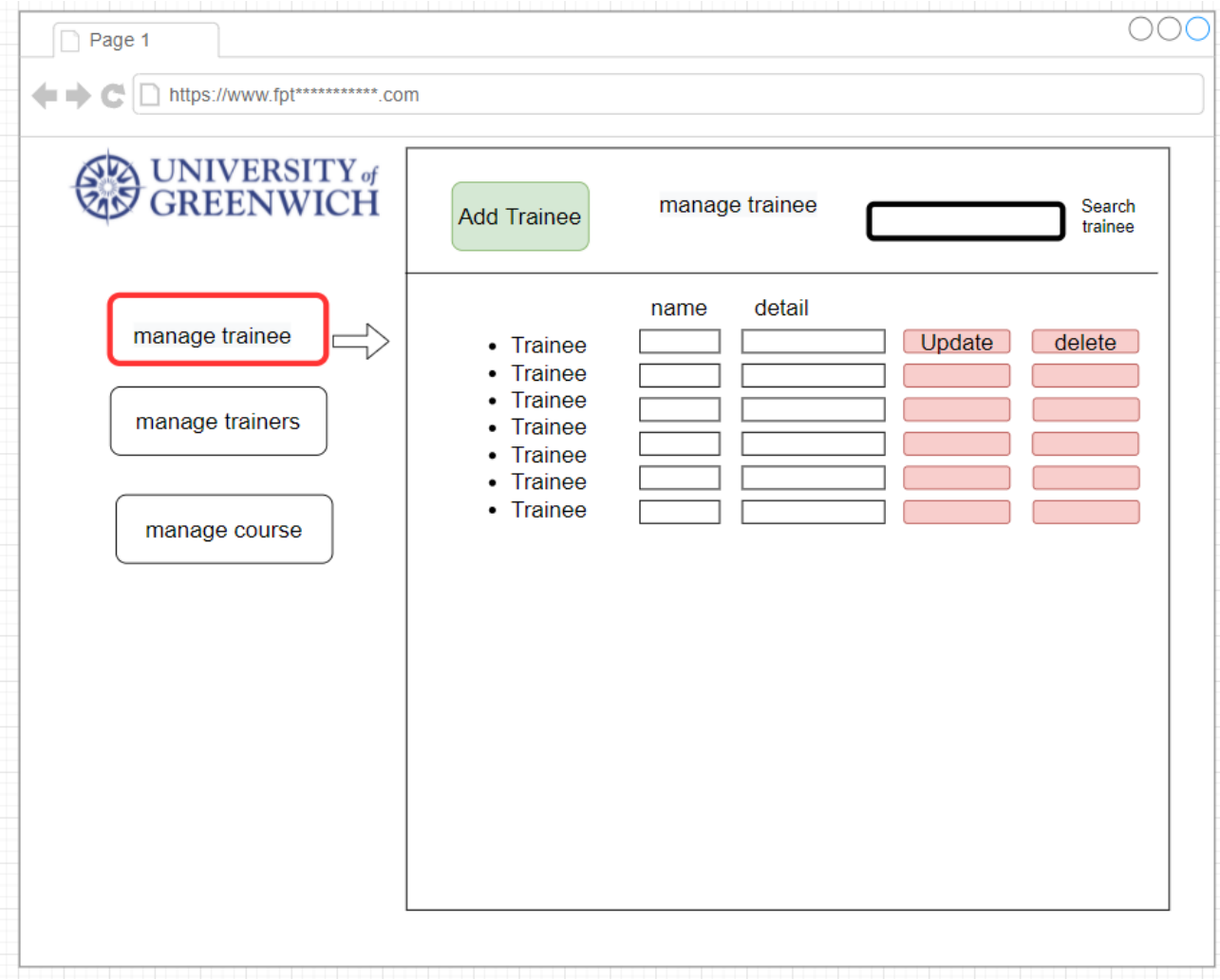


Figure 7: Staff Interface for Managing Trainee

Staff can manage Course list, can Add Trainer, Edit and Delete Course.

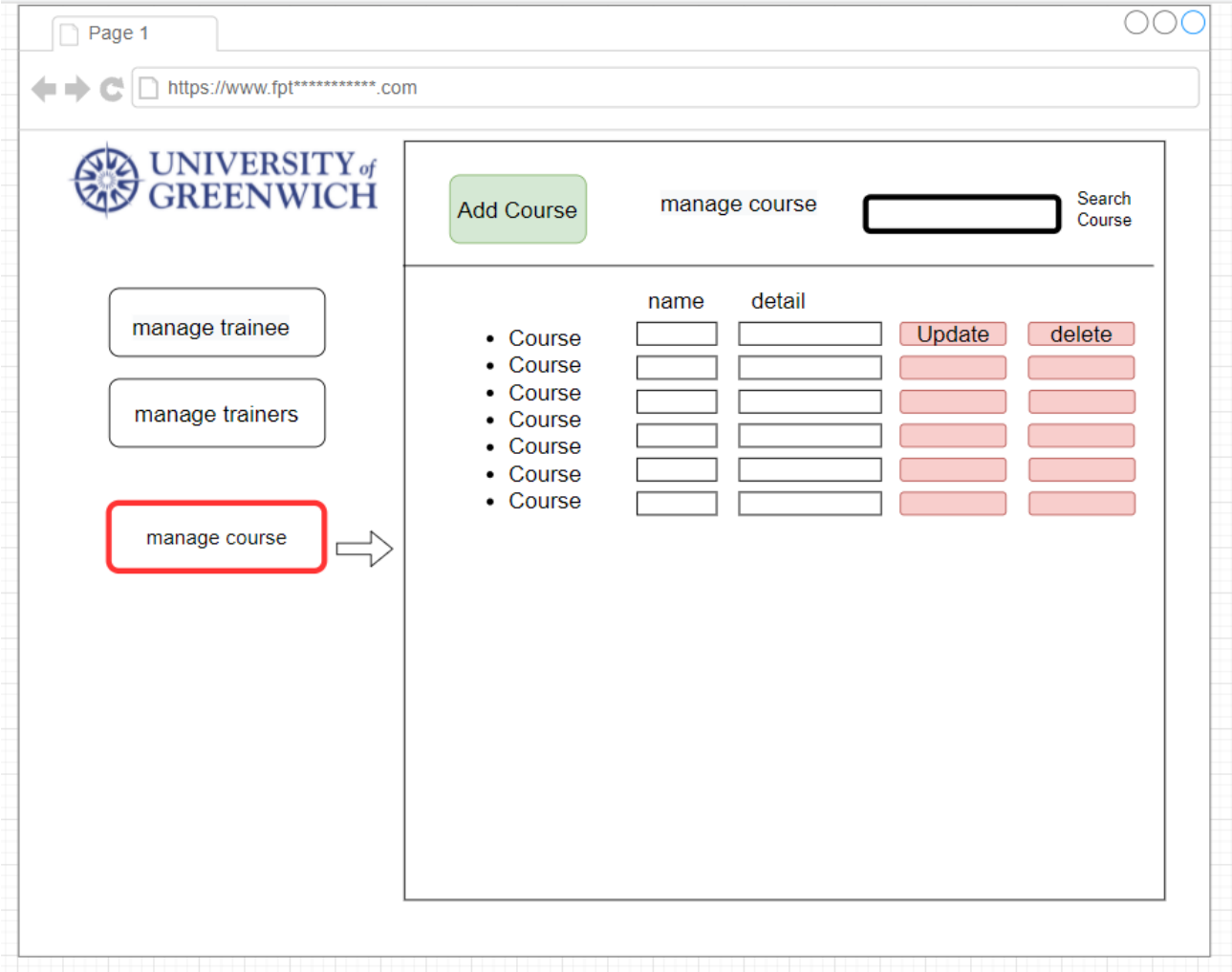


Figure 8: Staff Interface for Manage Course

Trainer' interface

Trainer can view and update my profile

Figure 9: Trainer Interface for View My Profile

Trainer can view Course and displayed so that the Trainer can know which class to teach

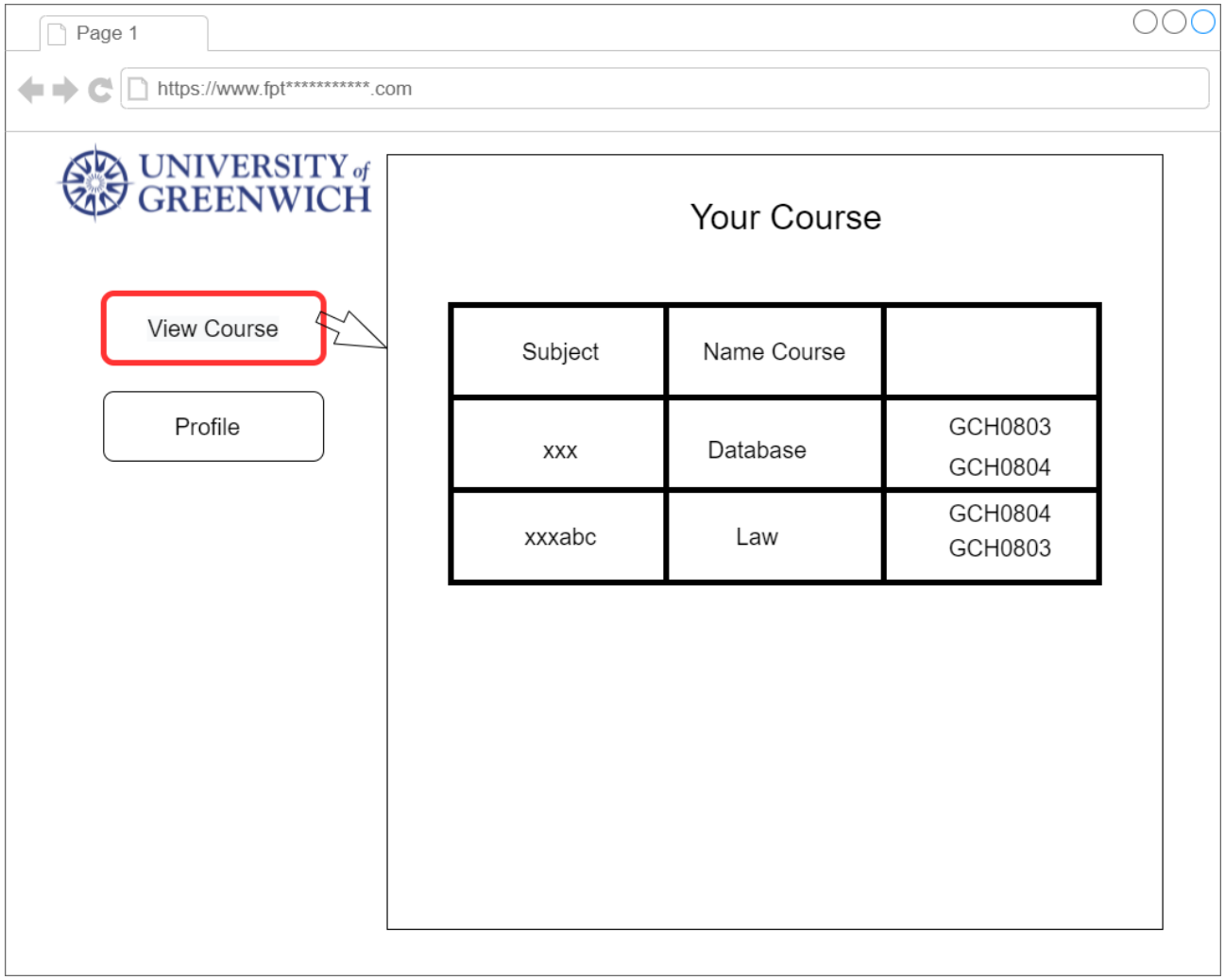


Figure 10: Trainer Interface for View Course

Trainee' interface's

Trainee can view My profile.

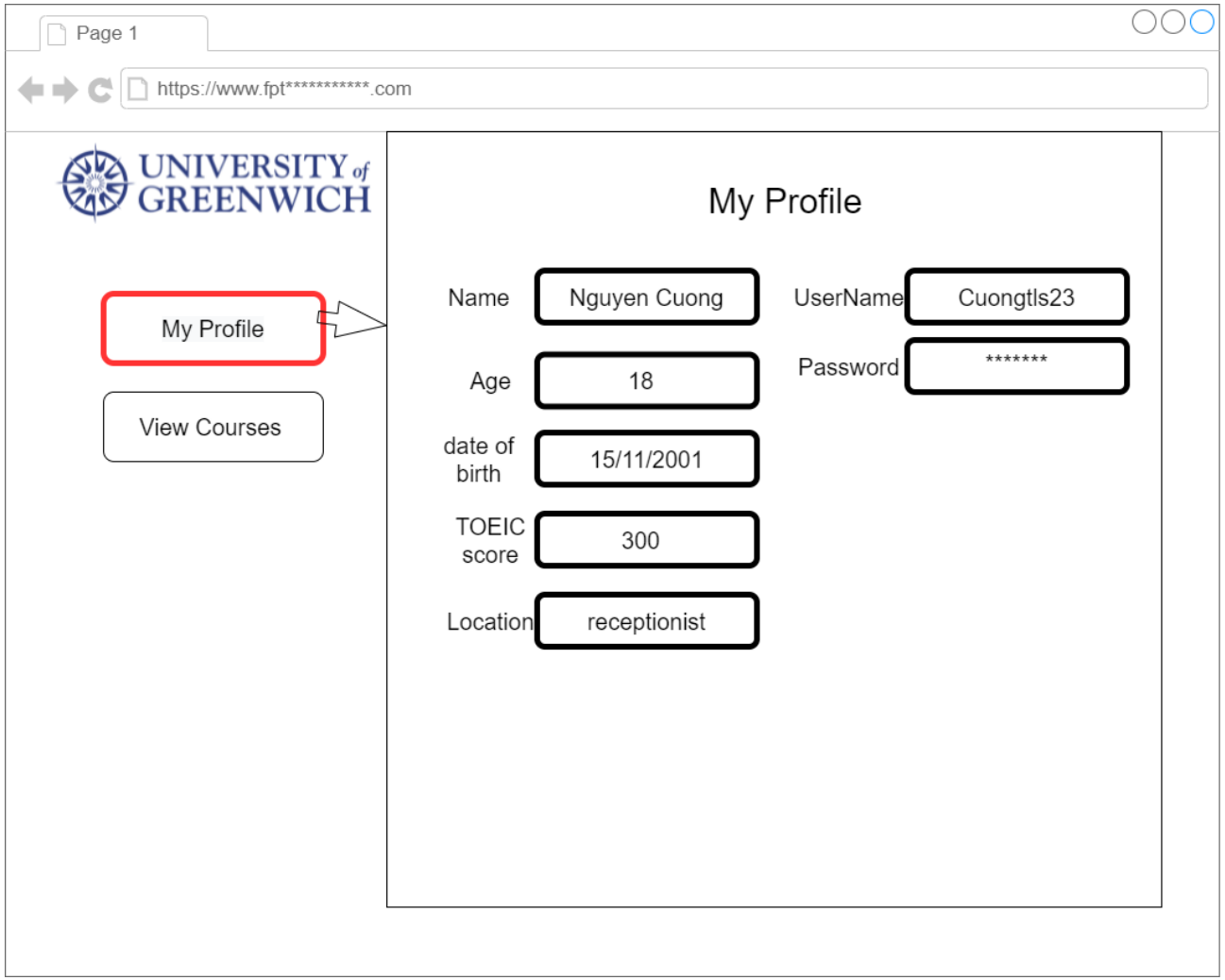


Figure 11: Trainee interface for View My Profile

Trainee can view all my registered course.

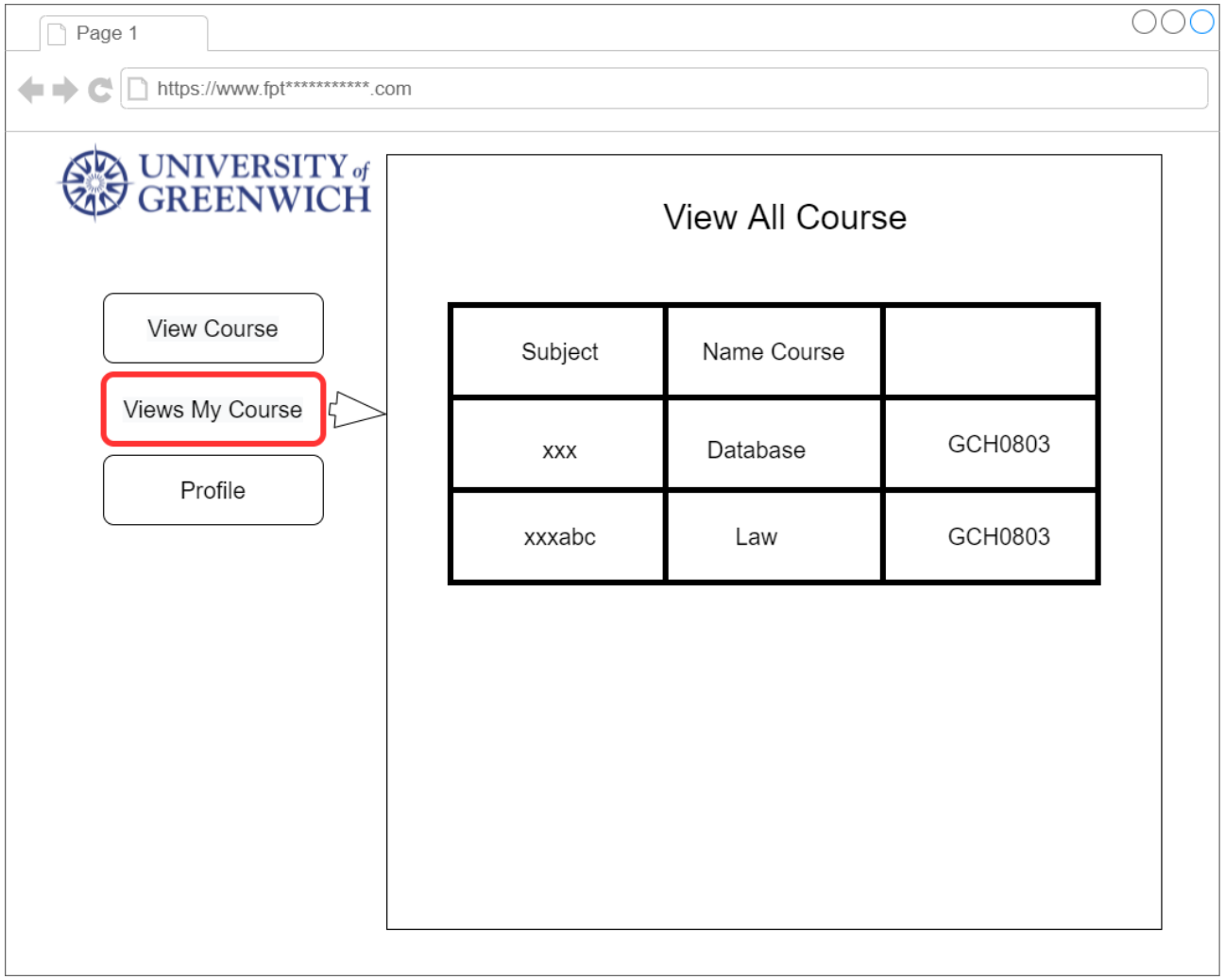


Figure 12: : Trainee interface for View My Course

Trainee can view all course to register course.

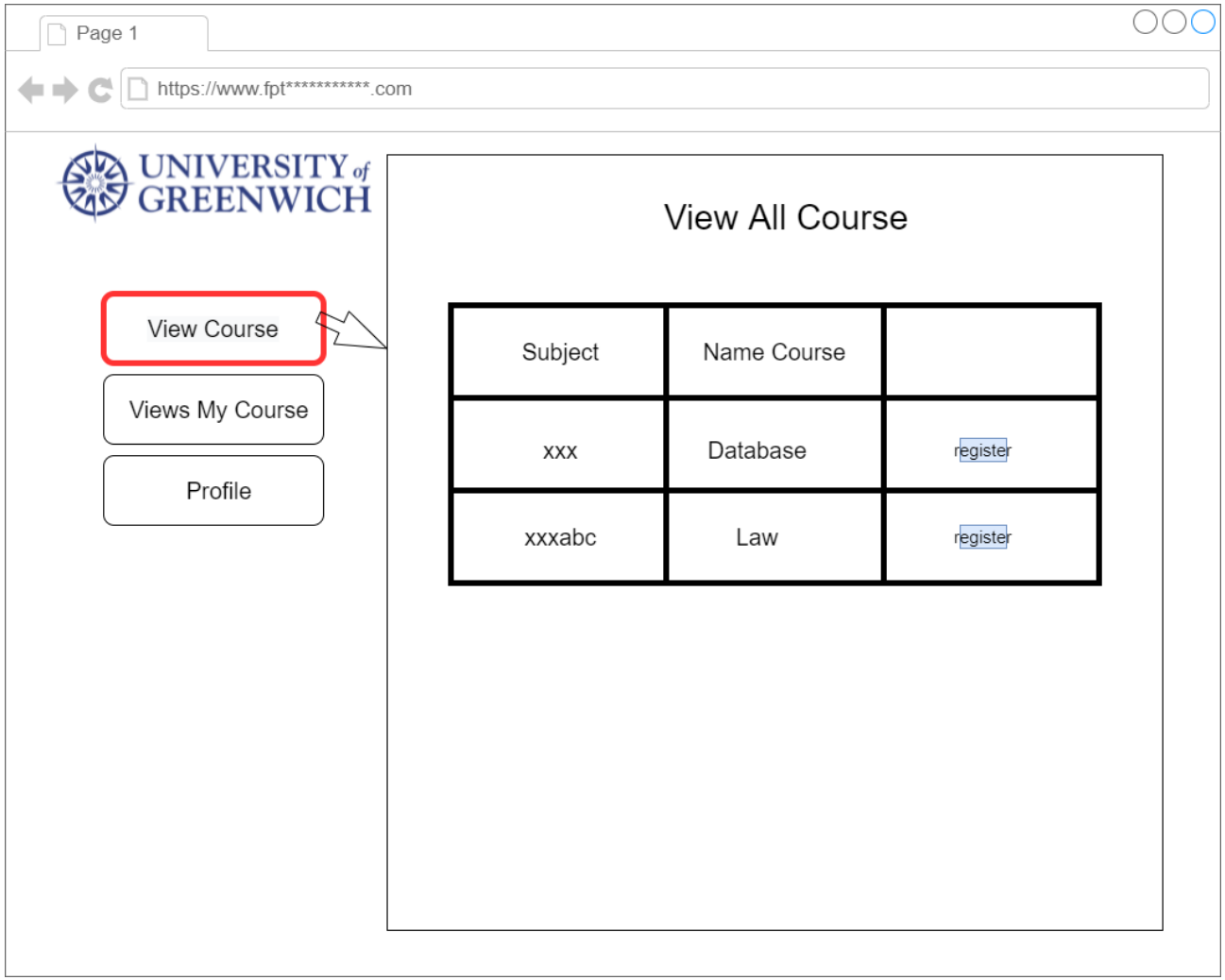


Figure 13: : Trainee interface for View Course

3.2 Functional Requirements

The system shall let:

Trainees

F1: The system shall let trainees perform the functions for the account: Login, Logout.

F2: The system shall let trainees view information such as: email address, name, address, course information, course duration, course categories, all available courses

Trainer

F3: The system shall let Trainer perform functions for the account such as: Login, Logout

F4: The system shall let Trainer edit information such as: edit course name, subject, edit personal information such as name, address.

F5: The system shall let Trainer view information such as: email, name, address, course duration, student information, teacher information, workplace, telephone.

F6: The system shall let Trainer update account password

F7: The system shall let Trainer view courses which he is assigned to such as: name's course, course categories, start time, end time.

Staff:

F8: The system shall let training staff create trainee accounts by entering details like trainee name, trainee accounts, age, date of birth, experience details, department, location, email.

F9: The system shall let training staff update, delete trainee accounts information such as: name, address, date of birth, email …

F10: The system shall let training staff manage course categories such as add course categories.

F11: The system shall let training staff manage course categories such as delete course categories.

F12: The system shall let training staff update course categories such as: name course categories, start date, end date, description …

F13: The system shall let training staff add courses including information: course name, start date, end date, description, course categories…

F14: The system shall let training staff delete courses

F15: The system shall let training staff update courses such as: course name, start date, end date, description, course categories…

F16: The system shall let training staff add trainer profiles with the information: Trainer name, External or Internal Type, working place, telephone, and email address.

F17: The system shall let training staff delete trainer profiles

F18: The system shall let training staff update trainer profiles with the information: Trainer name, External or Internal Type, working place, telephone, and email address.

F19: The system shall let training staff remove courses from trainee

F20: The system shall let training staff remove courses from trainer

F21: The system shall let training staff assign trainees to a course.

F22: The system shall let training staff assign trainers to a course.

Admin

F23: The system shall let admin login to the system

F24: The system shall let admin create/edit/delete new Trainer account and change (if existing user) its password

F25: The system shall let admin create/edit/delete new training staff account and change (if existing user) its password

3.2.1 <Functional Requirement or Feature #2>

Role	Function
Trainer	Login, Logout, Update profile, View course assigned to.
Trainee	Login, Logout, Update account password
Training staff	Login, Logout, create trainee accounts, Update , Delete trainee ,Mange courses categories, Mange courses, Manage trainer profile, assign trainer to courses, assign trainee to topic, Remove courses from Trainer, Remove courses from Trainee
Admin	Login, Logout. create/edit/delete new Trainer account, create/edit/delete new Training Staff account

3.3 Use Case Model

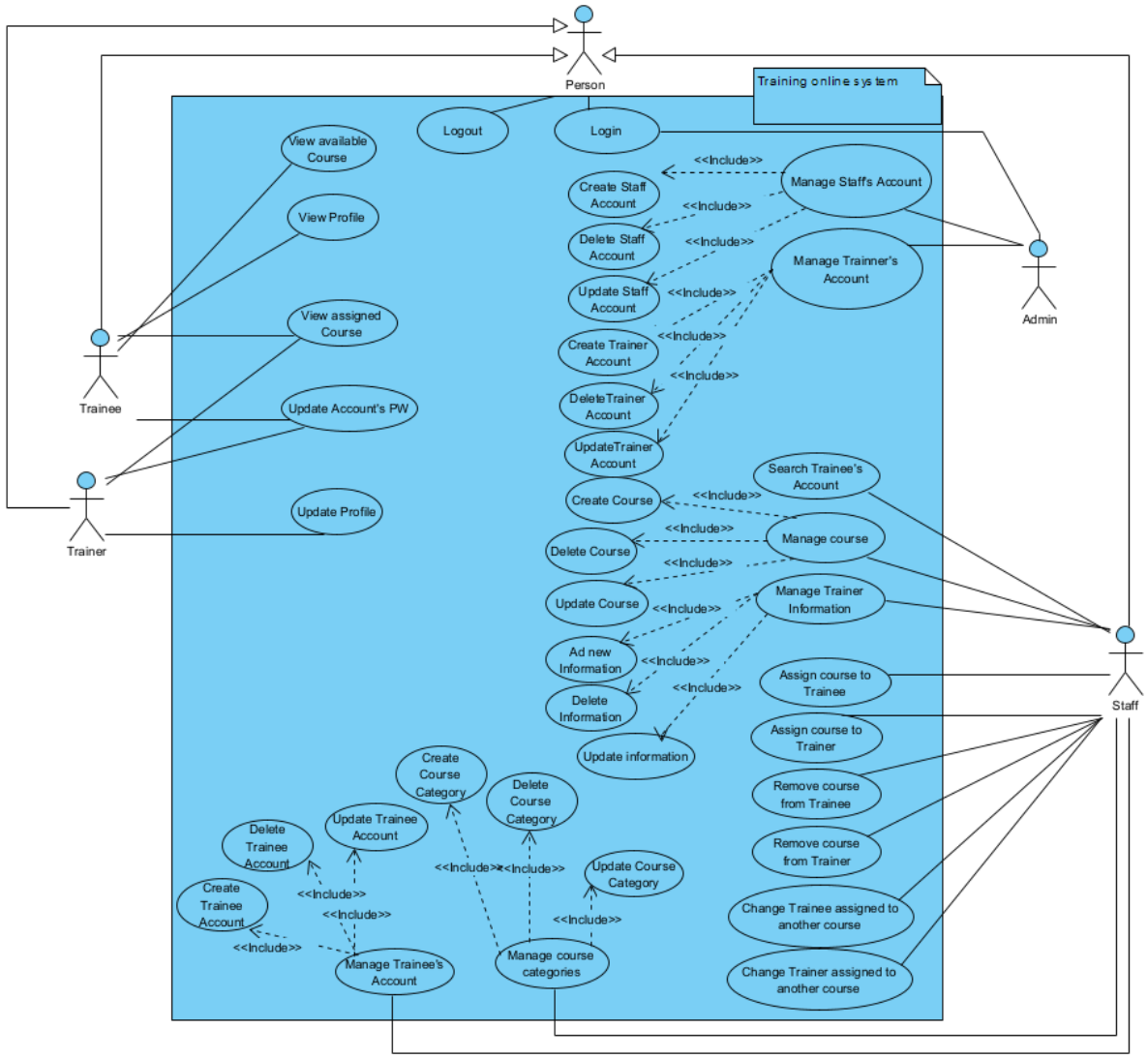


Figure 14: Use Case Model

3.3.1

Use Case #1

Use Case Name	Login
Description	As a user, we want to login to the app to use services from the app.
Actor(s)	Person
Priority	Must Have
Trigger	User wants to login to the system
Pre-Condition(s):	• User accounts already created • User account has been authorized • The user's device is already connected to the internet when logging in

Post-Condition(s):	• User successfully logged into the application • The system records successful logging in Activity Log.
Basic Flow	1. User access to the system 2. User enters account and selects login command 3. The system validates the login information successfully and allows the user to access the application 4. The system records successful login to the system
Alternative Flow	None
Exception Flow	The system validates the credentials failed and displays a message. The user chooses to cancel the login.
Business Rules	None

Non-Functional Requirement	None

Use Case #2

Use Case Name	Delete staff accounts
Description	Admin wants to delete the Staff account to the system.
Actor(s)	Admin
Priority	Must Have
Trigger	Admin wants to delete staff account
Pre-Condition(s):	• Admin account has been logged into the system • trainees' accounts have been registered in the system

	• admin must be connected to the internet before using the function • staff account has been created before
Post-Condition(s):	• admin delete staff account successfully • The list of staff accounts has been updated on the system
Basic Flow	1. Admin access to the system 2. admin perform login 3. admin performs the function of managing account staff 4. admin performs the function of delete staff accounts 5. The system records admin's management activities with the system
Alternative Flow	None
Exception Flow	update failed

Business Rules	None
Non-Functional Requirement	None

Use Case #4

Use Case Name	update staff accounts
Description	Admin wants to update the Staff account to the system.
Actor(s)	Admin
Priority	Must Have

Trigger	Admin wants to update staff account
Pre-Condition(s):	• Admin account has been logged into the system • trainees' accounts have been registered in the system • admin must be connected to the internet before using the function • staff account has been created before
Post-Condition(s):	• admin update staff account successfully • The list of staff accounts has been updated on the system
Basic Flow	1. Admin access to the system 2. admin perform login 3. admin performs the function of managing account staff 4. admin performs the function of update staff accounts 5. The system records admin's management activities with the system
Alternative Flow	None

Exception Flow	update failed
Business Rules	None
Non-Functional Requirement	None

Use Case #5

Use Case Name	Create staff accounts
Description	Admin wants to Create the Staff account to the system.

Actor(s)	Admin
Priority	Must Have
Trigger	Admin wants to Create staff account
Pre-Condition(s):	• Admin account has been logged into the system • trainees' accounts have been registered in the system • admin must be connected to the internet before using the function • staff account has been created before
Post-Condition(s):	• admin Create staff account successfully • The list of staff accounts has been updated on the system
Basic Flow	Admin access to the system admin perform login

	admin performs the function of managing account staff admin performs the function of Create staff accounts The system records admin's management activities with the system
Alternative Flow	None
Exception Flow	create failed
Business Rules	None
Non-Functional Requirement	None

Use Case #6

Use Case Name	Create trainer accounts
Description	Admin wants to create the trainer account to the system.
Actor(s)	Admin
Priority	Must Have
Trigger	Admin wants to create trainer account
Pre-Condition(s):	• Admin account has been logged into the system • trainees' accounts have been registered in the system • admin must be connected to the internet before using the function

Post-Condition(s):	• admin create trainer account successfully • The list of trainer accounts has been updated on the system
Basic Flow	1. Admin access to the system 2. admin perform login 3. admin performs the function of managing account trainer 4. admin performs the function of delete staff accounts 5. The system records admin's management activities with the system
Alternative Flow	None
Exception Flow	Create failed
Business Rules	None

Non-Functional Requirement	None

Use Case #7

Use Case Name	Delete trainer accounts
Description	Admin wants to delete a trainer account to the system.
Actor(s)	Admin
Priority	Must Have
Trigger	Admin wants to delete trainer account

Pre-Condition(s):	• Admin account has been logged into the system • trainees' accounts have been registered in the system • admin must be connected to the internet before using the function • trainer account must be created before
Post-Condition(s):	• admin delete trainer account successfully • The list of trainer accounts has been updated on the system
Basic Flow	1. Admin access to the system 2. admin perform login 3. admin performs the function of managing account trainer 4. admin performs the function of create staff accounts 5. The system records admin's management activities with the system
Alternative Flow	None
Exception Flow	

	Invalid admin information
Business Rules	None
Non-Functional Requirement	None

Use Case #8

Use Case Name	Delete Course
Description	Staff wants to delete a course to the system.

Actor(s)	Staff
Priority	Must Have
Trigger	staff wants to delete staff new course
Pre-Condition(s):	• staff account has been logged into the system • staff must be connected to the internet before using the function • staff perform the function management course to delete course
Post-Condition(s):	• staff delete course successfully • The list of course has been updated on the system
Basic Flow	1. Staff access to the system 2. Staff perform login 3. Staff performs the function of managing course 4. Staff performs the function of delete course

Alternative Flow	None
Exception Flow	delete course failed
Business Rules	None
Non-Functional Requirement	None

Use Case #9

Use Case Name	Update Course

Description	Staff wants to Update course to the system.
Actor(s)	Staff
Priority	Must Have
Trigger	staff wants to update course
Pre-Condition(s):	• staff account has been logged into the system • staff must be connected to the internet before using the function • staff perform the function management course to create course • course was created before
Post-Condition(s):	• staff update course successfully • The list of course has been updated on the system

Basic Flow	Staff access to the system Staff perform login Staff performs the function of managing course Staff performs the function of update course
Alternative Flow	None
Exception Flow	update course failed
Business Rules	None

Non-Functional Requirement	None

USE CASE 10

Use Case Name	Create Course
Description	Staff wants to create a course to the system.
Actor(s)	Staff
Priority	Must Have
Trigger	staff wants to create staff new course
Pre-Condition(s):	• staff account has been logged into the system

	• staff must be connected to the internet before using the function • staff perform the function management course to create course
Post-Condition(s):	• staff create course successfully • The list of course has been updated on the system
Basic Flow	Staff access to the system Staff perform login Staff performs the function of managing course Staff performs the function of create course
Alternative Flow	None
Exception Flow	

	Create course failed
Business Rules	None
Non-Functional Requirement	None

Use Case 11

Use Case Name	Assign course to Trainer
Description	User wants to add trainer to the course
Actor(s)	Staff
Priority	Must Have

Trigger	User wants to add trainer to course
Pre-Condition(s):	• staff account has been logged into the system • staff must be connected to the internet before using the function • staff perform the function assign course to trainer
Post-Condition(s):	• assign trainer successfully participated in the course
Basic Flow	1. Staff access to the system 2. Staff perform login 3. Staff performs the function of assign trainer to course
Alternative Flow	None

Exception Flow	assign	failed
Business Rules	None	
Non-Functional Requirement	None	

Use Case 12

Use Case Name	Assign course to trainees
Description	User wants to add trainees to the course
Actor(s)	Staff
Priority	Must Have

Trigger	User wants to add trainees to the course
Pre-Condition(s):	• staff account has been logged into the system • staff must be connected to the internet before using the function • staff perform the function assign course to trainer
Post-Condition(s):	• assigned trainees successfully participated in the course
Basic Flow	Staff access to the system Staff perform login Staff performs the function of assign trainer to course
Alternative Flow	None

Exception Flow	assign	failed
Business Rules	None	
Non-Functional Requirement	None	

USE CASE 13

Use Case Name	remove course from trainees
Description	staff wants to remove course from trainees
Actor(s)	Staff

Priority	Must Have
Trigger	staff want to remove course from trainees
Pre-Condition(s):	• staff account has been logged into the system • staff must be connected to the internet before using the function • staff perform the function remove trainees from course
Post-Condition(s):	• remove course from trainees successfully
Basic Flow	Staff access to the system Staff perform login Staff performs the function of remove course from trainee

Alternative Flow	None	
Exception Flow	remove	failed
Business Rules	None	
Non-Functional Requirement	None	

USE CASE 14

Use Case Name	remove course from trainer
Description	staff wants to remove course from trainer

Actor(s)	Staff
Priority	Must Have
Trigger	staff want to remove course from trainer
Pre-Condition(s):	• staff account has been logged into the system • staff must be connected to the internet before using the function • staff perform the function remove trainer from course
Post-Condition(s):	• remove course from trainer successfully
Basic Flow	Staff access to the system Staff perform login

	Staff performs the function of remove course from trainer	
Alternative Flow	None	
Exception Flow	remove	failed
Business Rules	None	
Non-Functional Requirement	None	

USE CASE 15

Use Case Name	Delete Course Categories
Description	Staff wants to delete course categories to the system.
Actor(s)	Staff
Priority	Must Have
Trigger	staff wants to delete course categories
Pre-Condition(s):	• staff account has been logged into the system • staff must be connected to the internet before using the function • staff perform the function management course to delete course categories
Post-Condition(s):	• staff delete course categories successfully • The list of course has been updated on the system

Basic Flow	5. Staff access to the system 6. Staff perform login 7. Staff performs the function of managing course categories 8. Staff performs the function of delete course categories
Alternative Flow	None
Exception Flow	delete course failed
Business Rules	None

Non-Functional Requirement	None

USE CASE 16

Use Case Name	Update Course categories
Description	Staff wants to Update course categories to the system.
Actor(s)	Staff
Priority	Must Have
Trigger	staff wants to update course categories

Pre-Condition(s):	• staff account has been logged into the system • staff must be connected to the internet before using the function • staff perform the function management course categories to create course • course categories are created before
Post-Condition(s):	• staff update course categories successfully • The list of course categories has been updated on the system
Basic Flow	Staff access to the system Staff perform login Staff performs the function of managing course categories Staff performs the function of update course categories
Alternative Flow	None

Exception Flow	update course categories failed
Business Rules	None
Non-Functional Requirement	None

USE CASE 17

Use Case Name	Create Course categories
Description	Staff wants to create a new course category
Actor(s)	Staff

Priority	Must Have
Trigger	staff wants to create new course categories
Pre-Condition(s):	• staff account has been logged into the system • staff must be connected to the internet before using the function • staff perform the function management course to create course categories
Post-Condition(s):	• staff create course categories successfully • The list of course categories has been updated on the system
Basic Flow	Staff access to the system Staff perform login Staff performs the function of managing course categories Staff performs the function of create course categories

Alternative Flow	None
Exception Flow	Create course categories failed
Business Rules	None
Non-Functional Requirement	None

Use Case #18

Use Case Name	Delete trainee accounts
Description	staff wants to delete the trainee account to the system.
Actor(s)	staff
Priority	Must Have
Trigger	staff wants to delete trainee account
Pre-Condition(s):	• staff account has been logged into the system • staff accounts have been registered in the system • staff must be connected to the internet before using the function • trainee account has been created before
Post-Condition(s):	• staff delete trainee account successfully • The list of trainee accounts has been updated on the system

Basic Flow	6. staff access to the system 7. staff perform login 8. staff performs the function of managing account staff 9. staff performs the function of delete staff accounts
Alternative Flow	None
Exception Flow	delete failed
Business Rules	None

Non-Functional Requirement	None

USE CASE 19

Use Case Name	update trainee accounts
Description	staff wants to update the Staff account to the system.
Actor(s)	staff
Priority	Must Have
Trigger	staff wants to update trainee account

Pre-Condition(s):	• staff account has been logged into the system • trainees' accounts have been registered in the system • staff must be connected to the internet before using the function • trainee account has been created before
Post-Condition(s):	• staff update trainee account successfully • The list of trainee accounts has been updated on the system
Basic Flow	1. staff access to the system 2. staff perform login 3. staff performs the function of managing account staff 4. staff performs the function of update staff accounts
Alternative Flow	None
Exception Flow	

	update failed
Business Rules	None
Non-Functional Requirement	None

USE CASE 20

Use Case Name	Create staff accounts
Description	staff wants to Create trainee account to the system.
Actor(s)	staff

Priority	Must Have
Trigger	staff wants to Create trainee account
Pre-Condition(s):	• staff account has been logged into the system • staff accounts have been registered in the system • staff must be connected to the internet before using the function • trainee account has been created before
Post-Condition(s):	• staff Create trainee account successfully • The list of trainee accounts has been updated on the system
Basic Flow	1. staff access to the system 2. staff perform login 3. staff performs the function of managing account staff 4. staff performs the function of Create staff account 5. The system records staff management activities with the system

Alternative Flow	None
Exception Flow	create failed
Business Rules	None
Non-Functional Requirement	None

Use Case 21

Use Case Name	Update profile
Description	User wants to change personal information

Actor(s)	trainers
Priority	Must Have
Trigger	User wants to change personal information
Pre-Condition(s):	• trainers must log in to the system • trainers choose update profile to change the information
Post-Condition(s):	• User successfully changed information
Basic Flow	1. User access to the system 2. users log into the system with the trainer account 3. User chooses to change personal information

Alternative Flow	None
Exception Flow	change information failed
Business Rules	None
Non-Functional Requirement	None

Use Case #22

Use Case Name	Update Account PW of trainer
Description	trainer wants to change PW

Actor(s)	trainers
Priority	Must Have
Trigger	trainer wants to change PW
Pre-Condition(s):	• trainers must log in to the system • trainers choose change profile to change the information
Post-Condition(s):	• User successfully changed PW
Basic Flow	1. User access to the system 2. users log into the system with the trainer account 3. User chooses to change personal information

Alternative Flow	None
Exception Flow	change information failed
Business Rules	None
Non-Functional Requirement	None

Use Case #23

Use Case Name	Update Account PW of trainee

Description	trainee wants to change PW
Actor(s)	trainee
Priority	Must Have
Trigger	trainee wants to change PW
Pre-Condition(s):	• trainee must log in to the system • trainee choose change profile to change the information
Post-Condition(s):	• trainee successfully changed PW
Basic Flow	1. trainee access to the system 2. trainee log into the system with the trainer account

	3. trainee chooses to change personal information
Alternative Flow	None
Exception Flow	Change information failed
Business Rules	None
Non-Functional Requirement	None

Use Case #24

Use Case Name	View assigned course
Description	trainee view assigned course
Actor(s)	trainee
Priority	Must Have
Trigger	trainee view assigned course
Pre-Condition(s):	• trainee must log in to the system • trainee choose view assigned course
Post-Condition(s):	• view successfully

Basic Flow	1. trainee access to the system 2. trainee log into the system with the trainer account 3. trainee chooses view assigned course
Alternative Flow	None
Exception Flow	None
Business Rules	None
Non-Functional Requirement	None

4 Technical Design

4.1 Screen flow Diagram

Screen flow Diagram Admin:

When the admin login the system will lead to the homepage from where the admin can see the other admin's account, they are Create account Staff and the trainer

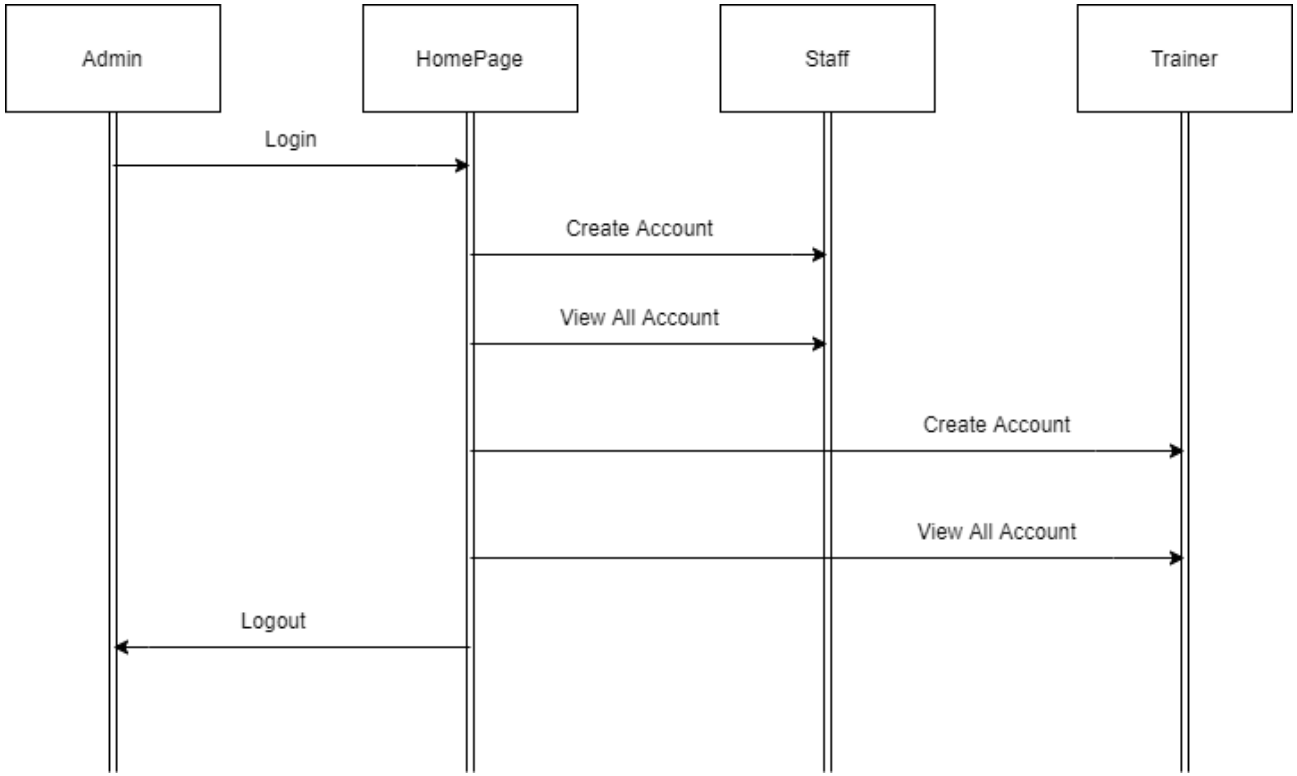


Figure 15: Screen flow Admin Diagram

Screen flow Diagram Staff:

When staff log into the system it will lead them to the home page from where they can proceed to manage functions such as adding course type, trainee, trainer, add trainer to topic, add trainee to course and from the course type. They can also go to courses and topic management.

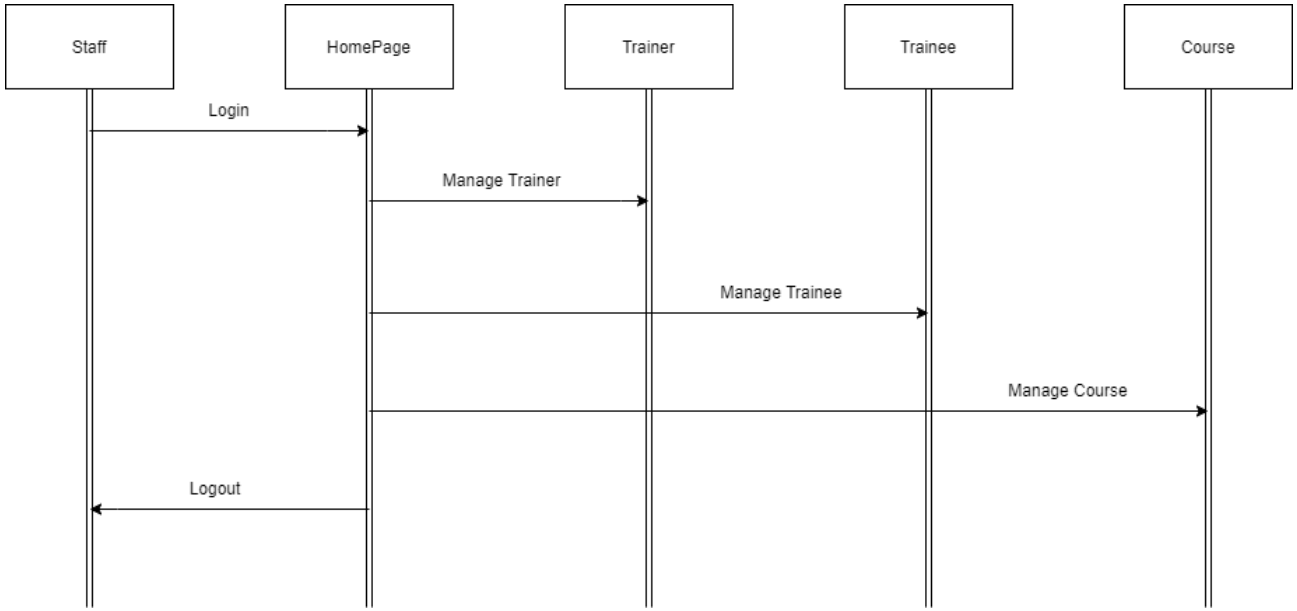


Figure 16: Screen flow Staff Diagram

Screen flow Diagram Trainer:

When the trainer log in to the system they will take them to the home page from where they can proceed to view their tasks and edit their profiles.

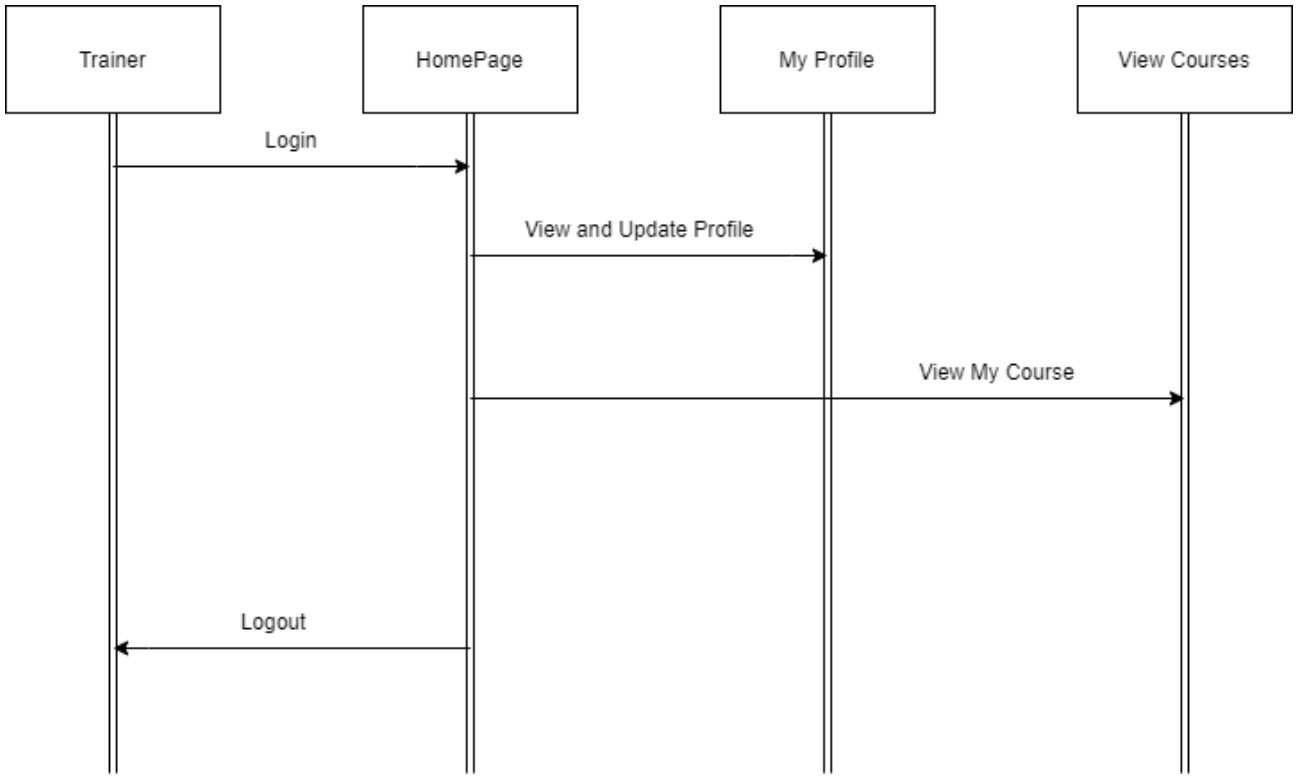


Figure 17: Screen flow Trainer Diagram

Screen flow Diagram Trainee:

When the trainee logs in to the system it will lead them to the home page from where they can proceed to view their tasks and profiles.

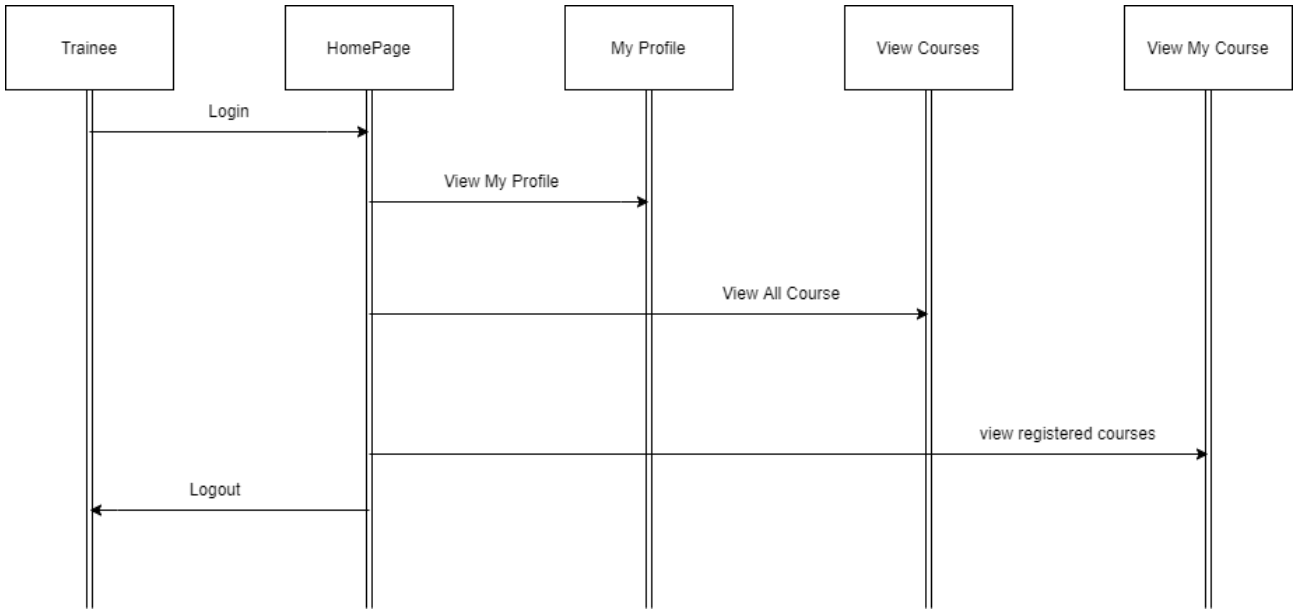


Figure 18:Screen flow Traiee Diagram

4.2 Entity Relationship Diagram (ERD)

Figure 19: Entity Relationship Diagram (ERD)

4.3 Class Diagram

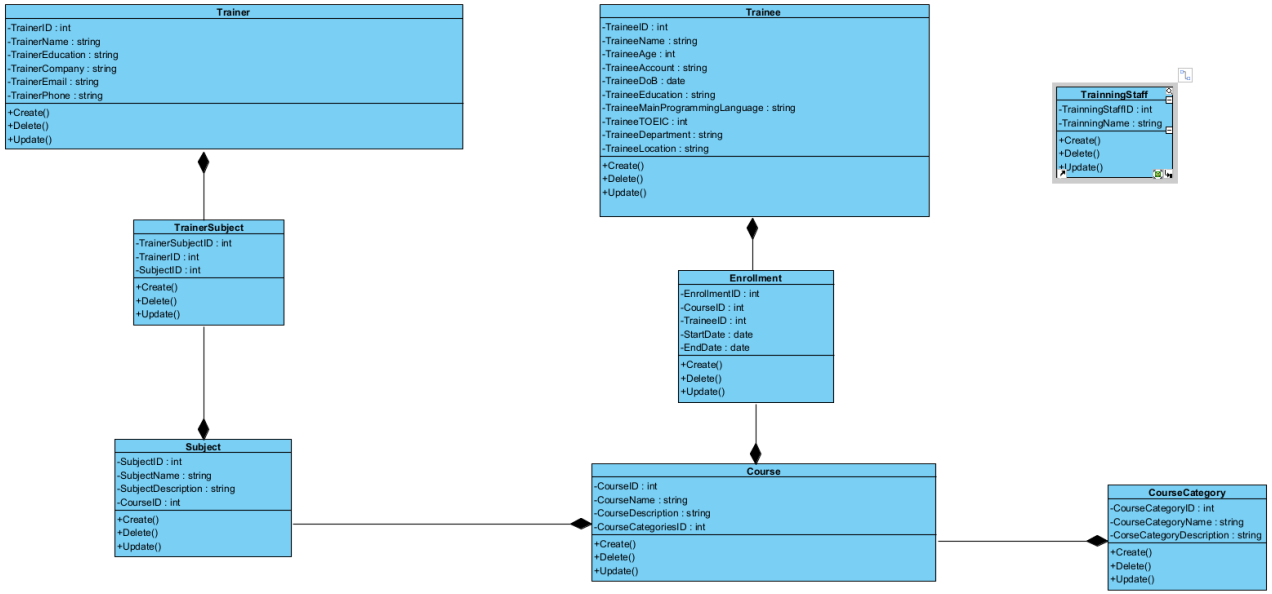


Figure 20: Class Diagram

4.4 Activity Diagram

Below is the operation diagram of the system that outlines each of the activities and roles of the user in the system

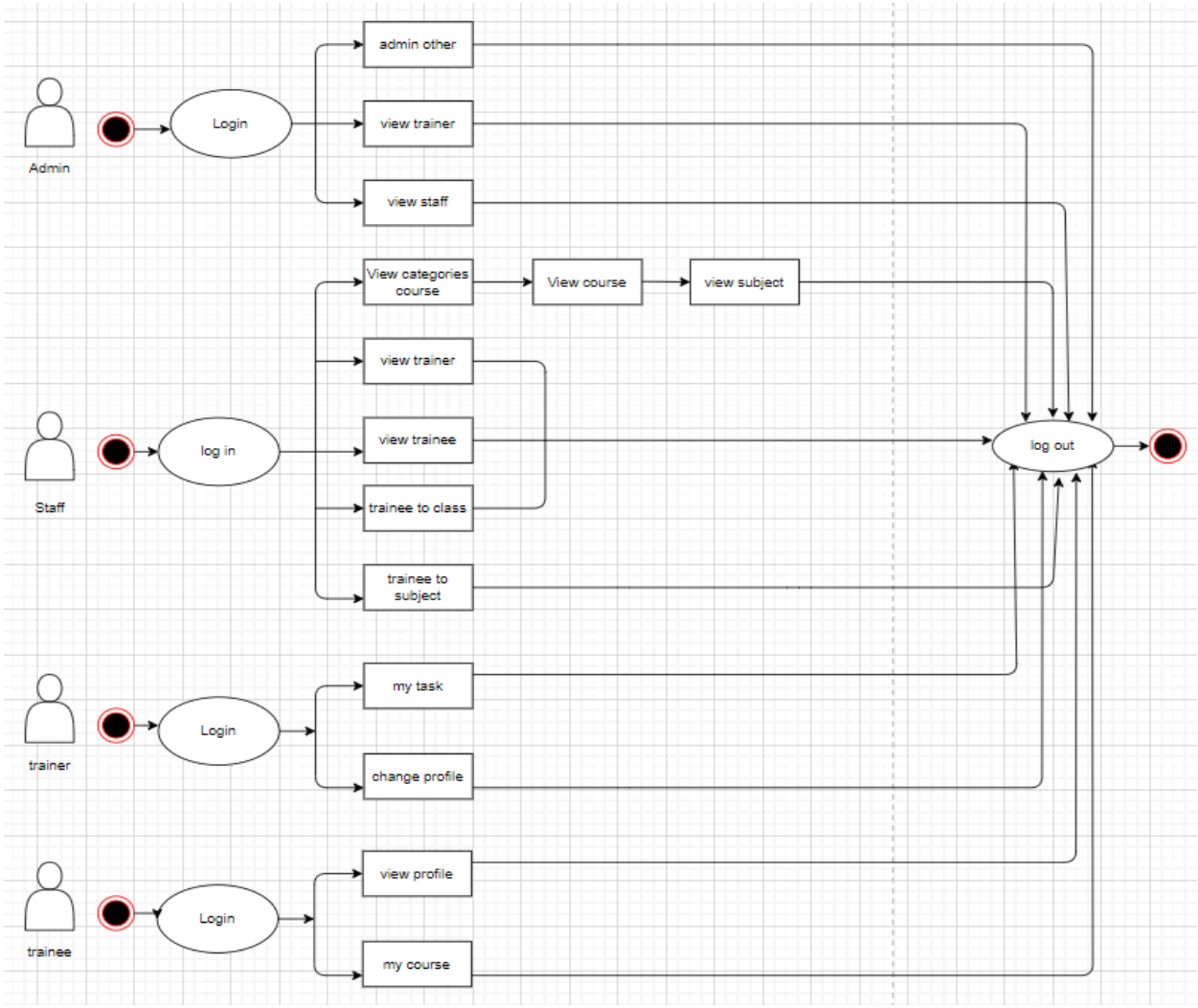


Figure 21: Activity Diagram 1

Below is the operation diagram of the admin, the admin will be able to manage the staff, trainer and other admin accounts

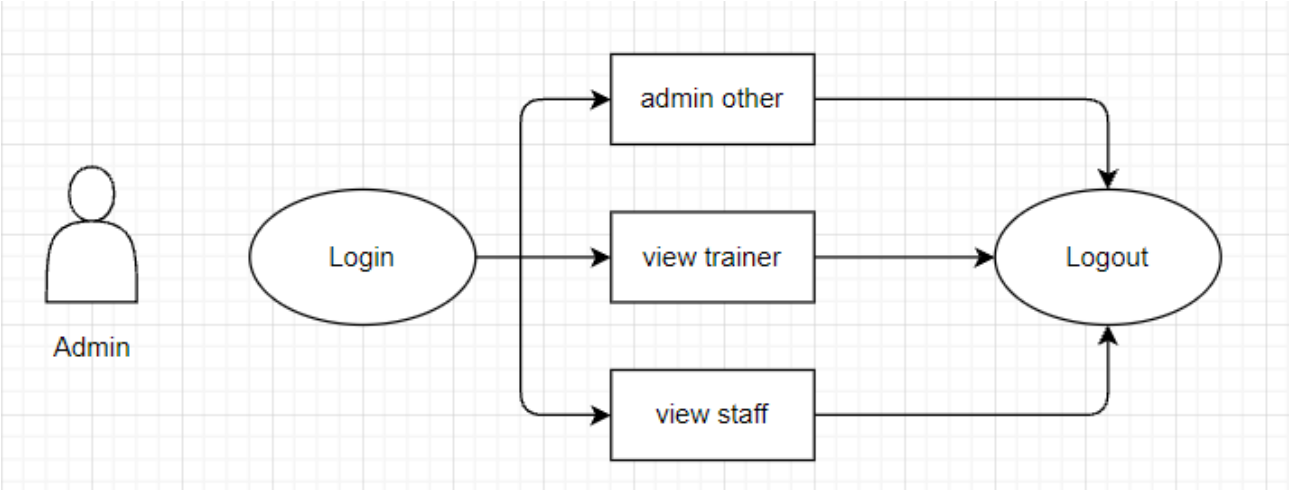


Figure 22: Activity Diagram 2

Below is the staff activity diagram, the staff will be able to manage the trainer account, trainee, course, course type, topic, add trainee to course, add trainer to topic

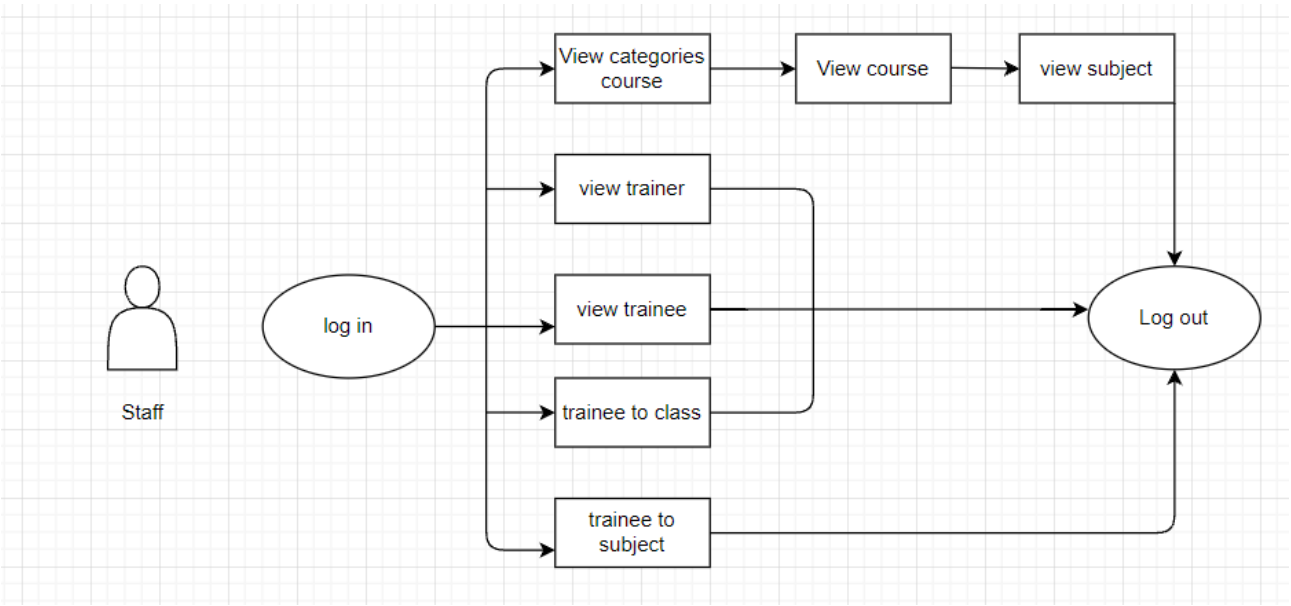


Figure 23: Activity Diagram 3

Below is a chart of the trainer activity, the trainee will be able to see what they teach and change their personal information.

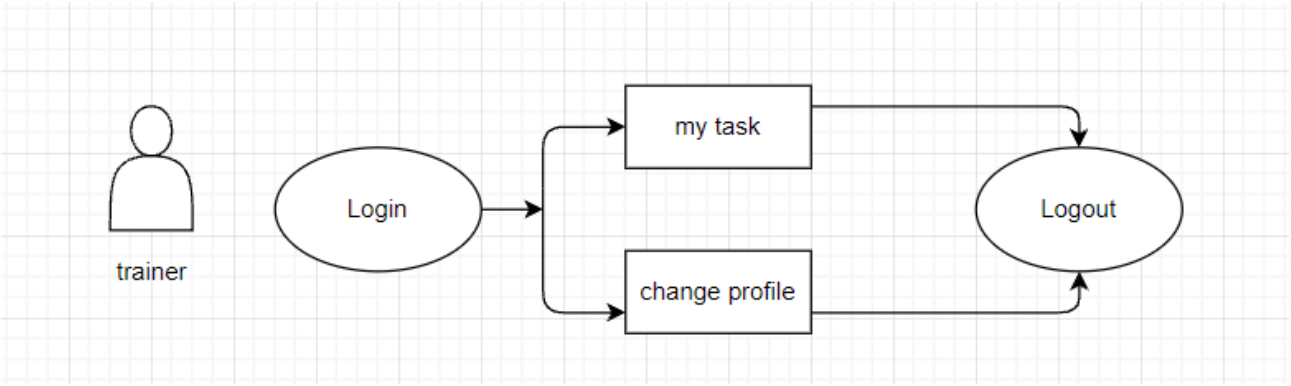


Figure 24: Activity Diagram 4

Below is a chart of the trainee's activities, the trainee will be able to see which class they belong to and their personal information.

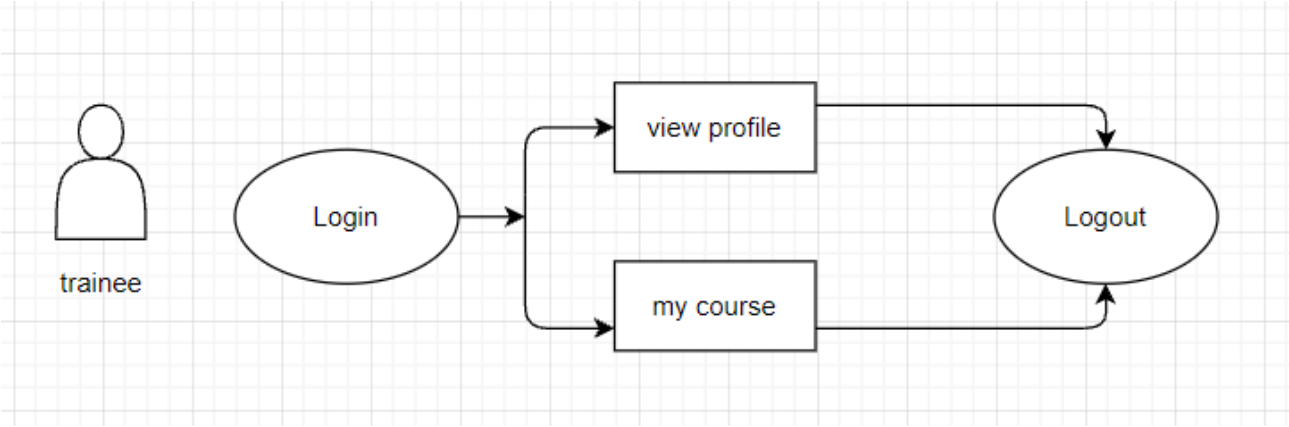


Figure 25: Activity Diagram 5

5 Risk Management

5.1 Risk Management Matrix

NAME	FPT System	OBJECTIVE	

REF / ID	PRE-MITIGATION				DEPARTMENT / LOCATION	MITIGATIONS / WARNINGS / REMEDIES	POST-MITIGATION			
	RISK	RISK SEVERITY	RISK LIKELIHOOD	RISK LEVEL			RISK SEVERITY	RISK LIKELIHOOD	RISK LEVEL	ACCEPTABLE TO PROCEED?
		– ACCEPTABLE – TOLERABLE – UNDESIRABLE – INTOLERABLE	– IMPROBABLE – POSSIBLE – PROBABLE	**– LOW** **– MEDIUM** **– HIGH** **– EXTREME**			– ACCEPTABLE – TOLERABLE – UNDESIRABLE – INTOLERABLE	– IMPROBABLE – POSSIBLE – PROBABLE	**– LOW** **– MEDIUM** **– HIGH** **– EXTREME**	**YES / NO**
1	Risk of nature	INTOLERABLE	IMPROBABLE	**MEDIUM**	Greenwich University	There should be data backup plans	TOLERABLE	IMPROBABLE	**LOW**	**NO**
2	Money of risk	ACCEPTABLE	PROBABLE	**HIGH**	Greenwich University	There should call for more investors	ACCEPTABLE	PROBABLE	**MEDIUM**	**YES**
3	Lack of times	ACCEPTABLE	POSSIBLE	**HIGH**	Greenwich University	Should plan the time carefully, make a schedule	TOLERABLE	POSSIBLE	**MEDIUM**	**YES**

4	Lack of technical skills	UNDESIRABLE	PROBABLE	**HIGH**	Greenwich University	Should learn and learn from people who have experience in that field	UNDESIRABLE	PROBABLE	**MEDIUM**	**YES**
5	Misunderst and the request	UNDESIRABLE	POSSIBLE	**HIGH**	Greenwich University	Should create regular meetings with customers to get the requirements in the most clear way	TOLERABLE	POSSIBLE	**MEDIUM**	**YES**
6	Human resource risk	UNDESIRABLE	IMPROBABLE	**MEDIUM**	Greenwich University	There should be backup people for emergencies	ACCEPTABLE	PROBABLE	**LOW**	**NO**

6 Technical Evaluation

6.1. Design Tool

Tools to design UML

UML stands for Unified Modeling Language. Developers created UML in the late 1990s to better map out complex projects. Software development has become more complex. UML still holds an important place in any software development project. UML provides teams with a valuable tool for mapping complex software projects. UML is a popular, flexible visual modeling language. UML creates a standard form for visualizing systems.

With the basic intro and understanding out of the way, let us focus on some of the best UML diagramming tools that you can look into: Lucidchart, Visual Paradigm, Gliffy, Creately....

Tools to design User Interface

By definition, UI design is the process designers use to build interfaces in software or computerized devices, focusing on looks or style. These are the best UI design tools to help you create an interface: Sketch, Adobe XD, Draw.io, UXpin…

Apply for this project

In this project, we use Visual Paradigm to design UML diagrams. The reason we choose the visual paradigm is:

- Powerful visual modeling tools that help you build and manage your diagrams and model elements
- End-to-end business tools that help you improve your business's efficiency and productivity.
- Achieve better results by managing your software projects with full-fledged project management tools.

For UI design, we choose draw.io to design because:

- Easily produces good quality diagrams
- Has a rich set of predefined shapes for all sorts of different diagramming needs
- Allows grouping of shapes
- Smart connectors
- Integrates with Google Drive
- Conveniently exports to a variety of formats
- Allows for collaborative development of diagrams

6.2 Front End technology Stack

The front-end part of a website is the part that interacts with the user. Everything you see when navigating the Internet, from fonts and colors to drop-down menus and sliders... Primary front-end programming languages such as HTML, CSS, JS…

In addition to being fluent in those languages, front-end developers need to be familiar with frameworks like Bootstrap, AngularJS, and EmberJS, etc. to ensure that content always displays well on all different devices, and libraries like jQuery and LESS, encapsulate code into a more useful and time-saving form.

For our project, we use HTML5 as the main language to design and structure my website. The reasons are:

- No need to create cookies: In versions prior to HTML5, if we wanted to save any information, we had to create cookies. However, with this version, we do not need to create cookies.
- Customizable Data Attributes: With the HTML5 language, data can be customized. We do not need to learn about servers or Ajax when renting a server, they can also program a highly compatible website.

- Menu Element: Added to increase web interactivity.
- Convenience when designing mobile web: HTML5 helps us easily manipulate when building or designing mobile web interfaces.
- Increased compatibility for web applications: HTML5 allows the browser to act as an application platform, helping us improve website performance management.

Besides, We choose bootstrap 4 to help improve my website, there are some typical reasons why we chose this framework:

- Because Bootstrap is designed as a module, it easily integrates with most popular open-source codes on the market today such as WordPress, Joomla,
- Besides Bootstrap is also very easy to use, as long as you have basic knowledge of HTML, CSS or more JavaScript, you can use Bootstrap quite professionally.
- In addition, Bootstrap also has a very powerful Responsive feature that helps your website to automatically adjust the website size to match the screen size of devices such as phones, tablets, etc.
- Finally, it is compatible with many popular browsers on the market such as: chrome, Firefox, MS edge, ...

6.3 Backend Technology Stack

The server-side of software development is the backend tech stack. It refers to the inner workings of a website or app that users cannot see. Think of it like the electric power stations that generate electricity for your home. They may seem invisible in the background, but they are important to keep the operations running smoothly.

The backend stack includes these elements:

1. Programming languages – This creates logic for apps and websites. The codes link the web to a database. Some examples are JavaScript, PHP, and Python, C#-ASP.Net, Prominent among them is C#: this is a language with an extremely large online community, easy to learn, easy to understand, developed by Microsoft, can create cross-platform software, besides it We also work with ASP.Net framework and have high security, so that's why We chose C# as the programming language for this project.

2. Frameworks – it provides support of applications based on a single programming language. Laravel, Django, and Ruby, .Net, Entity,. Among them are ASP.Net and Entity: with ASP.net being considered as a good technology to secure the web language. good XML support and data accessibility via ADO.Net. works well in 1 application for high performance. can split the interface and code into two completely different parts, making code management easier. Optimized page load speed to reduce loading time. About Entity Framework, help users increase productivity through reducing their code.

3. Servers – You need backend servers to manage client requests. Apache, Nginx, and Microsoft's Internet Information Server (IIS) are great examples of web servers. With the Windows operating system, ASP.Net technologies are very suitable for this platform. If you need an enterprise level database there are some different features that run on Linux platform but if you need to use the base MSSQL database, Windows server is the best choice. Access to the database is only possible on Windows server. Access can only run in Microsoft's Windows platform and therefore it will not be usable on Linux servers. If your website specifically needs to use this access database then you will have to choose to use Windows server.

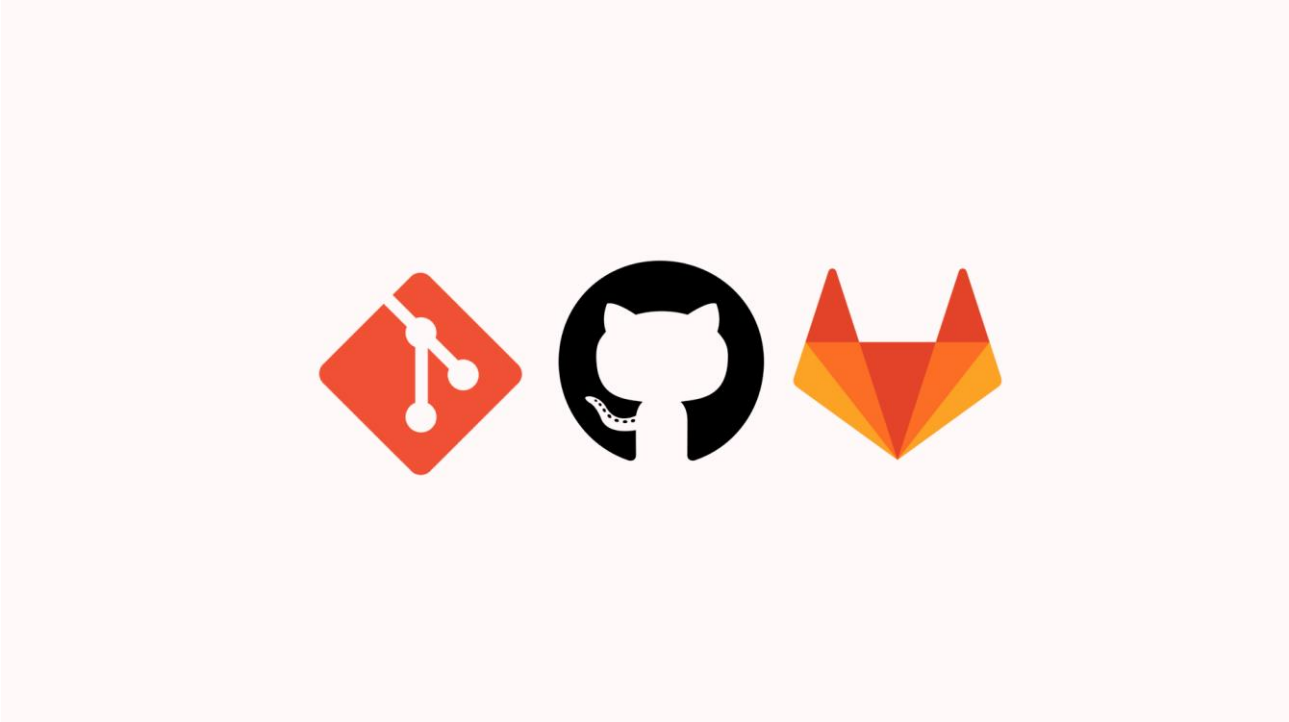
4. Databases – It's a digital space to store data (word document, web page, MP3 file). MongoDB, PostgreSQL, and MySQL are common databases. The highlight is SQL server: making it easy to add, edit and delete data. This is a software supported by many big companies. Besides, it makes programming easier.

5. Hosting: Popular hostings trusted by users include: Hawk Host, Stable Host, AZDIGI, Heroku. The most prominent and trusted by everyone is probably Heroku. This is a tool capable of managing groups, combining many programmers together to be able to build better software, Add-ons You can scale, enhance and manage your applications with built-in services such as: New Relic, MongoDB, SendGrid, Searchify, Fastly, Papertrail, ClearDB MySQL, Treasure Data, Complete Database Completely free, SSL is free to use, Able to support team work, Can link with the simplest Github types.

6. System Operating: There are 4 popular operating systems for users including MAC, Linux, Windows, Android. With windows operating system, it is equipped for many types of machines, such as asus, Acer, HP, Dell, so it will be very suitable for all users on this operating system. Besides, it has a rich application store for users to use for many different jobs and purposes and can emulate Android or IOS applications, which will be superior to other operating systems such as MAC or iOS. Linux.

7. In conclusion, with the above data, we will use C# language to program the project, use frameworks like ASP.Net, Entity Framework, use SQL Server to store data, use Windows Server Run and develop the project, use Windows Operating system , and use Heroku hosting to develop the project.

6.4 Tools for source control management



Git

Git is a Distributed Version Control System (DVCS), it is one of the most popular distributed version management systems today. Git provides each developer with its own repository ({repository) containing the entire change history.

The Version Control System is an open-source distributed version control system. The VCS will store all files in the entire project and record the entire file change history. Each saved change will be converted into a version

GitHub

GitHub is a project management and code versioning system that acts like a social network for developers. Developers can clone source code from a repository and GitHub is a public repository server service, each person can create an account on it to create their own repositories to work with.

GitHub is a well-known service that provides Git source code repositories for software projects. GitHub has all the features of Git, in addition to adding social features for developers to interact with each other.

GitLab

GitLab is an open-source self-hosted system based on the Git server system used to manage your source code. With GitLab, organizations, individuals, and businesses can store and manage code in a scientific, secure, and quickly accessible manner via an internet connection. GitLab offers free storage for users, and you can pay extra if you want to increase the storage.

In our application, we decided to choose GitHub for development because of the following advantages.

GitHub is considered as the largest and easiest to use social network for programmers with core features such as:

- Wiki, issue, statistics, project rename, project is placed in the namespace as user.

-Watch project: track the activities of other people's projects. See how people develop software, how projects develop.

-Follow user: track other people's activities.

There are 2 GitHub approaches: Create your own project Contribute to an existing project: fork someone else's existing project, modify it, then ask them to update their edit.

Benefits of GitHub

Manage source code easily:

When you create a repo, the entire source code of that repo is stored on GitHub. Here, you can review the process you have worked on through comments after each commit. And the beauty here is that many people can do the same repo.

Tracking changes across versions:

When there are many members working on a project, it can be quite complicated to keep track of revisions – who changed what, when and where those files are stored. Don't worry because GitHub has taken care of this for you, by always saving the changes you have pushed to the repository. Similar to Microsoft Word or Google Drive, you have a version history in case previous versions are lost or not saved.

Markdown:

Markdown is a way of formatting text on the web. You can edit the appearance of documents, format words like bold or italic, add images, and create lists of things you can do with Markdown. Most of the time, Markdown is just plain text with special characters inserted, like # or *. In GitHub, you can use Markdown in places: Git, Comments at Issues and Pull Requests, files with .md or markdown extension.

GitHub helps improve coding skills, even tracking bugs:

There are thousands upon thousands of ways to learn, learning on GitHub would be a good idea in this day and age. With thousands of open-source projects, hundreds of thousands of contributors, billions of commits every day, just by watching. In comparison, learning from those changes has given you tons of good things to improve your own coding skills.

“Bug tracking” is a feature integrated by GitHub to simplify the process of “finding and killing bugs”. To understand the process, all you need to do is open the dashboard of each project and filter the information. After that, the questions will be systematically sorted by popularity, update time or future. This software also has a pretty smooth interface, so it is always ranked high in the IT dev community.

6.5 Software Development Models

1. Introduce several SDLC models: Scrum, Waterfall, V-model, etc.

SCRUM

Scrum is a framework for sustainable development of complex products. This can be understood as a general framework for organizing work towards the development of complex products, mainly software. However, Scrum can be used as a foundation for organizing tasks ranging from flexible project management in general, to product development, marketing campaigns, teaching organization, production. modular cars or other personal work.

Currently, the Scrum definition is documented in the Scrum Guide and is regularly updated by the ScrumGuides authors themselves. It should be noted that Scrum is a framework, not a specific method. When applied, Scrum provides the basic foundation, combined with other methods or practices to work.

SCRUM'S THREE LEGS

Scrum is an agile methodology, so it adheres to the principles of the Agile Manifesto (Manifesto for Agile Software Development). In addition, Scrum operates on three core values, also known as the Three Legs of Scrum including Transparency, Inspection and Adaptation.

Transparency: In Scrum, transparency is promoted as the most basic core value. To be successful with Scrum, information related to the development process must be transparent and transparent. Such information can be: product vision, customer requirements, work progress, problems and barriers, etc. From there people in different roles have enough information needed to make valuable decisions to improve work efficiency. Scrum tools and meetings ensure that information is transparent to all parties.

Inspection: Continuous inspection of activities in Scrum ensures the discovery of problems and solutions so that diverse and useful information reaches project stakeholders. Thorough and continuous review is the starting mechanism for adaptation and continuous improvement in Scrum.

Adaptation: Scrum is as flexible as other agile software development methodologies. As a result, it offers a very high adaptability. Based on transparent information from inspection and work processes, Scrum can respond to changes positively, thereby bringing success to the project.

Waterfall

The Waterfall model is also known as the Waterfall model. The Waterfall model is one of the easiest project management models to understand today. The Waterfall model is a project management methodology based on a sequential and sequential design process.

In the Waterfall model, the phases of the project are executed one after the other. The new phase is only started when the previous one has been completed.

A simple waterfall model has 6 phases: requirement, design, implementation (build), verification, deployment, and maintenance.

Stages of the Waterfall model

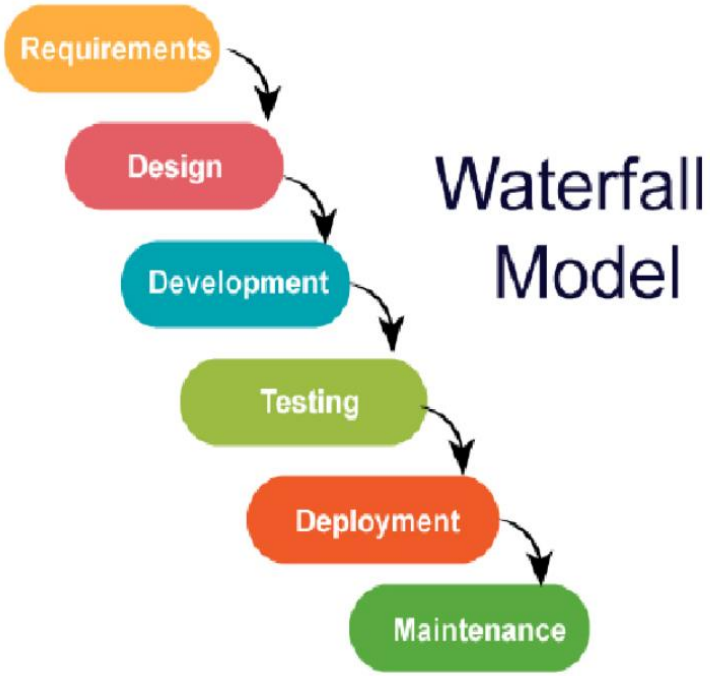


Figure 26: Waterfall Model

V-model

Model V is also known as Verification and Authentication Model. Every test done has to follow a certain sequence and V Model is one of the perfect methods to conduct software testing. In V-Model there are certain steps or sequences that need to be followed while executing the test. When a step is complete, it is moved to the next step. V-shaped test execution chains. In the software development life cycle, V-Model testing should start at the very first steps of the project to begin the requirements analysis. During the development and testing of the project V Model should be monitored in parallel.

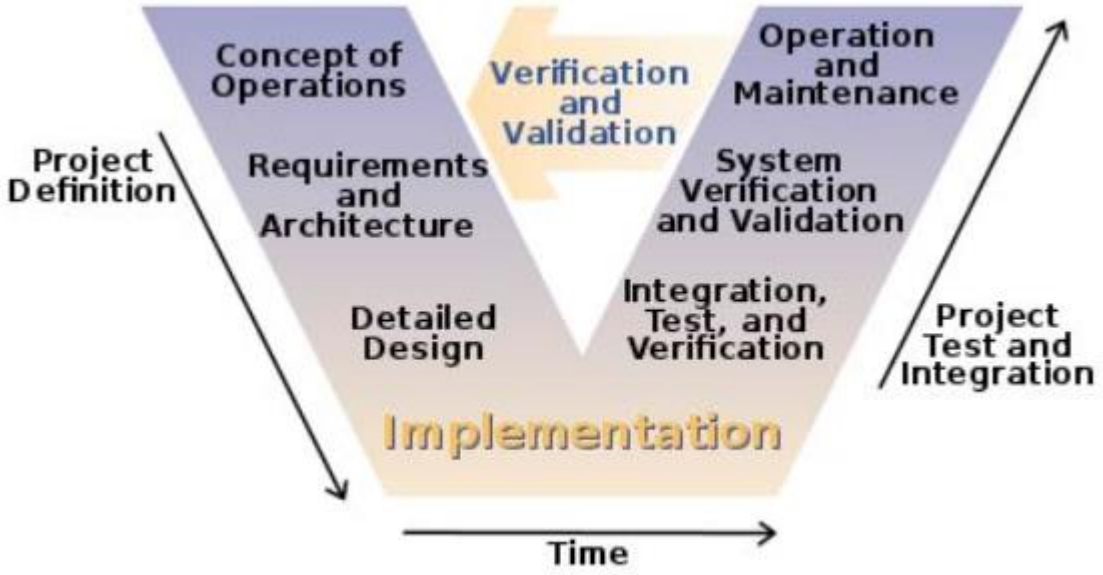


Figure 27: V-Model

2. **Conclude which SDLC model will be used for the development (Waterfall, Scrum, etc.) with explanations**

SDLC Waterfall model will be used for development because: Analysts and users go from one step to the next in a waterfall development process. As the project is approved from phase to phase, the major deliverables for each phase are generally extensive and given to the committee and project for approval. As soon as the work performed in one phase is accepted, the phase comes to a close and the next begins. As the project progresses, it progresses in a waterfall-like fashion from one phase to the next. Nevertheless, although it is possible to travel backwards through the stages, it is quite difficult.

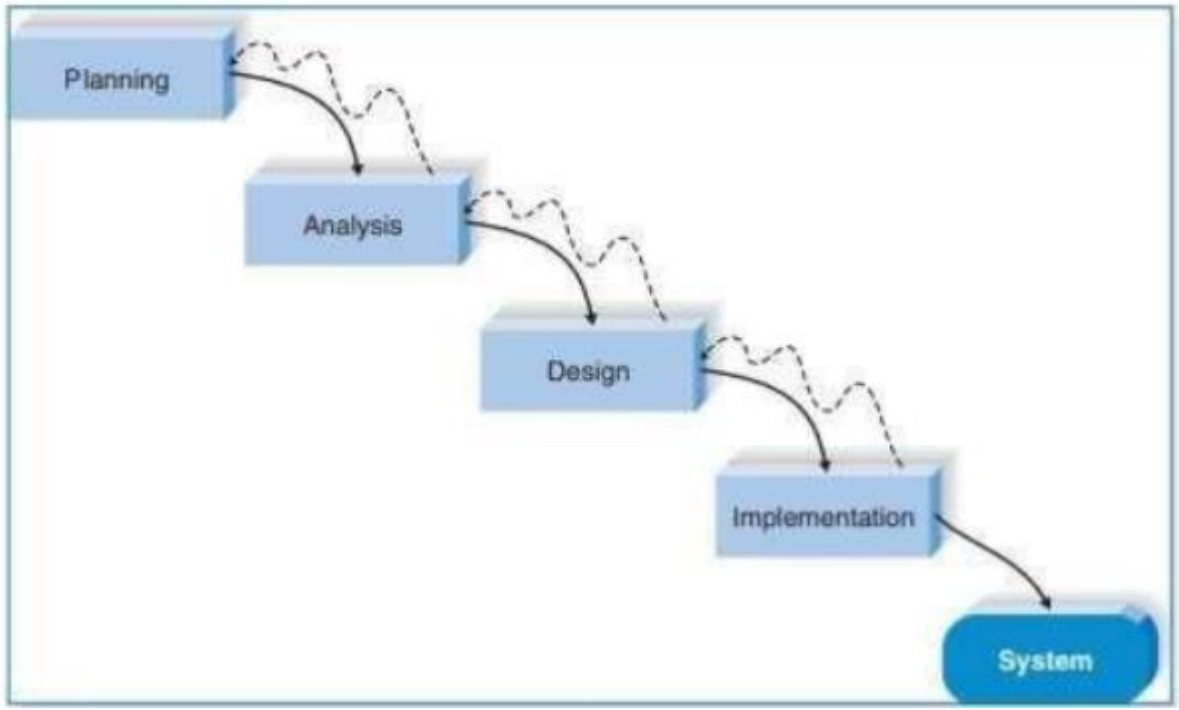
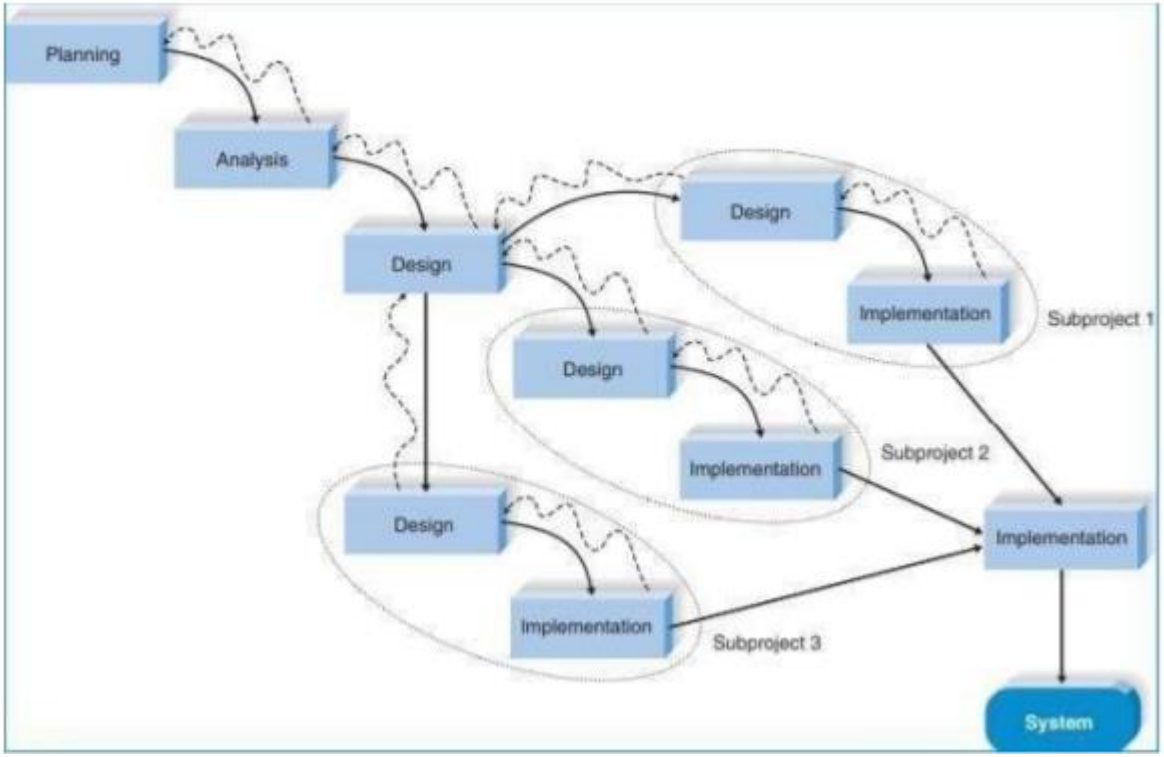


Figure 1 Waterfall Model

Waterfall development techniques offer the benefit of defining requirements well before programming begins and minimizing needed modifications as the project progresses. The main drawbacks are that the design must be fully described before programming can begin, that there is a lengthy period between the completion of the system proposal in the analysis phase and the delivery of the system, and that testing is virtually an afterthought in the implementation phase. Furthermore, because deliverables are frequently a poor means of communication, key needs may be ignored in the voluminous paperwork. If a key need is overlooked by the project team, costly post-implementation programming may be required. Because so much time has passed between the initial idea and the actual implementation, users may forget the system's original goal. Furthermore, in today's changing business world, a system that fits the current environmental circumstances during the analysis phase may require significant modification when it is implemented to meet the environment. This rework necessitates returning to the first step and making necessary modifications in each of the following phases. (Dennis, 2009).



Here are two significant waterfall development variations. Parallel development techniques arose in response to the lengthy time frames associated with waterfall development. Instead of designing and implementing the system in order, a general design for the entire system is carried out. The project is then split into a number of subprojects, each of which may be planned and implemented independently. When all of the sub projects are finished, the system is provided after a final integration of the various components. Because parallel development decreases the time it takes to deploy a system, rework is less likely as a result of changes in the business environment. The method still has issues due to the large number of deliverables. It also introduces a new issue: if the subprojects are not fully autonomous, design decisions in one subproject may have an impact on another, and merging the subprojects at the conclusion of the project may be difficult. (Dennis, 2009).

7. References

Học viện Agile. 2021. *Scrum là gì? Vai trò và ý nghĩa của Scrum - Học viện Agile*. [online] Available at: <https://hocvienagile.com/scrum-la-gi/> [Accessed 5 October 2021].

gocnhintangphat.com. 2021. *Mô Hình V Model Là Gì ? Tìm Hiểu Với Ví Dụ Sdlc& Stlc Mô Hình Phát Triển Phần Mềm V Là Gì*. [online] Available at: <https://gocnhintangphat.com/v-model-la-gi/> [Accessed 5 October 2021].

Gaille, B., 2021. *19 HTML5 Advantages and Disadvantages*. [online] BrandonGaille.com. Available at: <https://brandongaille.com/19-html5-advantages-and-disadvantages/> [Accessed 5 October 2021].

Dennis, A. a. H. W., 2009. Systems Analysis and Design. s.l.:s.n