ASM2 Application - adsfgafasdf

Web-based Java Applications (Trường Đại học FPT)

Scan to open on Studocu

ASSIGNMENT 2 FRONT SHEET

Qualification	BTEC Level 5 HND Diploma in Computing		
Unit number and title	Unit 30: Application Development		
Submission date	7 May 2021	**Date Received 1st submission**	
Re-submission Date		**Date Received 2nd submission**	
Student Name	Ngo Dac Nhat Nam	**Student ID**	GDH190027
Class	GCH0717	**Assessor name**	Tran Quy Ban

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	**Nam Ngo**

Grading grid

P4	P5	P6	M3	M4	M5	D2	D3

☐ **Summative Feedback:** ☐ **Resubmission Feedback:**

Grade: **Assessor Signature:** **Date:**

Lecturer Signature:

Table of Contents

I. Introduction

In the last report based on the requirements my team completed the system design. Besides, we provided a number of risks when we started a project and a solution for each of them. In this report, we will talk about system development, code implementation and user evaluation after experiencing the system and making future improvements.

II. Peer review

We ran a Survey on a group of clients as well as classmates. Customers will experience using the system and then they will give comments, assess the strengths and weaknesses of this system. Our target audience is clients who have experience in the training field so that we get accurate and quality opinions from them. From there, we will listen to, take deep notes and valuable ideas from them and can improve in the future.

Question	Answer
How many points do you give to the interface of this website?	1 to 10 points
Are the student features in the system easy to use?	Easy – Medium – Hard
Are the features for teachers in the system easy to use?	Easy – Medium – Hard
Are the features for Staff in the system easy to use?	Easy – Medium – Hard
Are the features for administrator in the system easy to use?	Easy – Medium – Hard
What are the strong points of this website?	Good interactions hold roles: student, teacher, manager Good security Good training management Store information well
What is the performance of the website?	Good – Medium - Slow
In your opinion, what is the feature missing from this 'Training' web app?	Short paragraph

➕ Result of survey

How many points do you give to the interface of this website?

23 responses
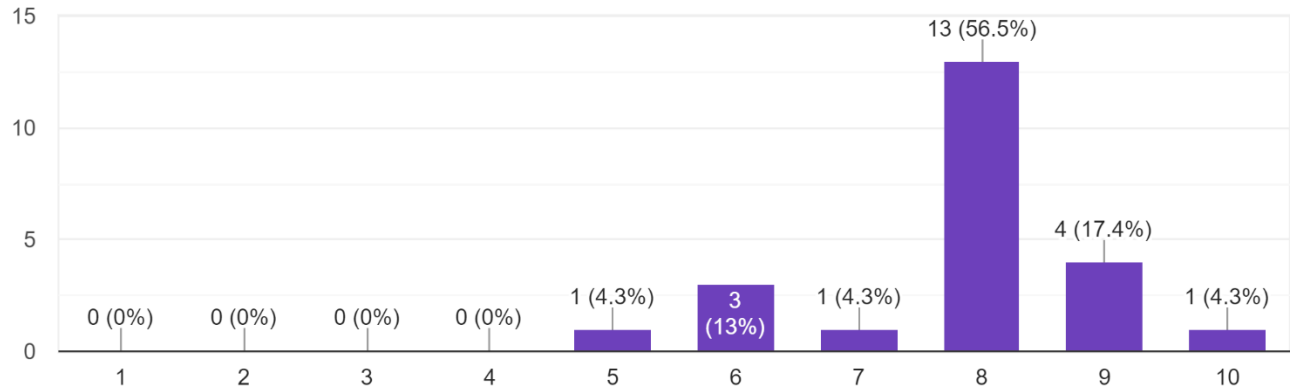
Most people rate the interface of our web application as pretty 8 points (56.5%). This is our initial success when dealing with the front end with good results with the customer. However, this is not the result we are most satisfied with, we self-evaluate our interface is average and will need to improve in the future such as: increased colors, more harmonious design.
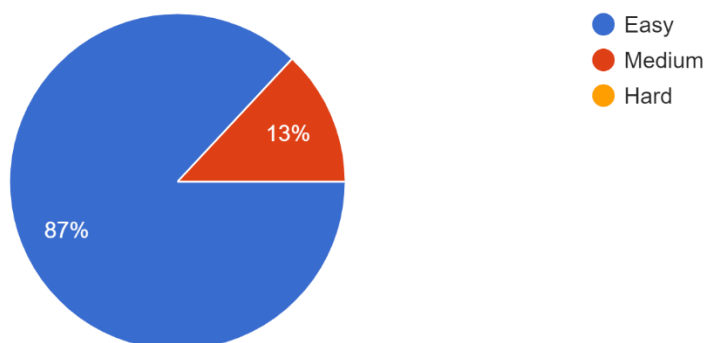
Are the student features in the system easy to use?

23 responses

- Easy
- Medium
- Hard

91.3%
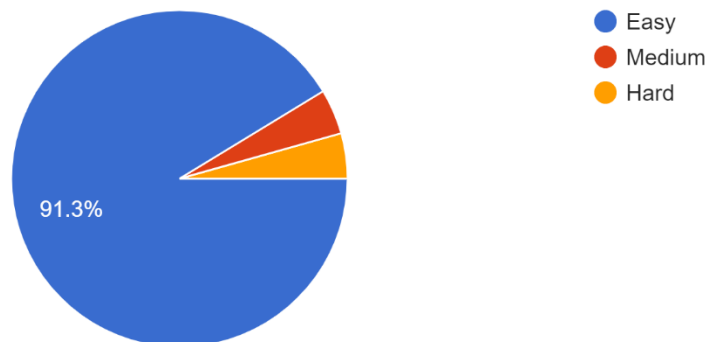
Students' features include: view the subjects being studied and view the assignments of those subjects. This feature is highly rated by users for being 91.3% easy to use.

Are the features for teachers in the system easy to use?
23 responses

Legend:
- Easy
- Medium
- Hard

13%

87%

The teacher's features include: viewing the list of subjects you teach, uploading assignments for each subject, editing uploaded assignments. This feature is highly rated by users for being 87% easy to use.

## Are the features for Staff in the system easy to use?
23 responses

Legend:
- Easy
- Medium
- Hard

17.4%

78.3%
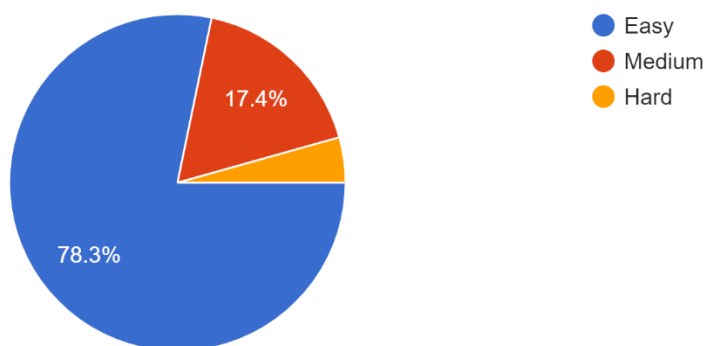
The staff's features include: adding, deleting, or deleting student accounts, adding, deleting, and deleting classes, adding and deleting courses, adding and deleting subjects. As a staff, there will be almost all system functions. However, with the design of synchronous features, easy to use with everyone, it has been highly appreciated by users: about 78.3% said it was easy to use. However, about 17.4% of respondents think that staff features are a bit difficult to use and adapt to. It can be seen that the staff features are far more than the ones of students and teachers, so this can be an issue that makes it more difficult for a small number of users to use.

Are the features for administrator in the system easy to use?

23 responses

- Easy
- Medium
- Hard

34.8%

60.9%
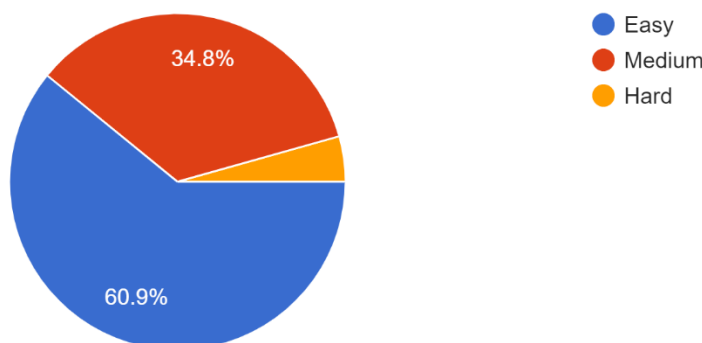
The function of the administrator is similar to the function of the staff, but as an administrator, there will be an additional feature that is adding, editing, deleting staff accounts. This is also the character with the highest management authority in the system. Through the survey results, we can see that about 60.9% think it is easy to use and 34.8% think it is a bit difficult to use -> this may be understandable when the admin manages all the functions so it will be difficult adapt in a short time.

What are the strong points of this website?

23 responses

Good interactions hold roles: student, teacher, manager	20 (87%)
Good security	5 (21.7%)
Good training management	12 (52.2%)
Store information well	8 (34.8%)
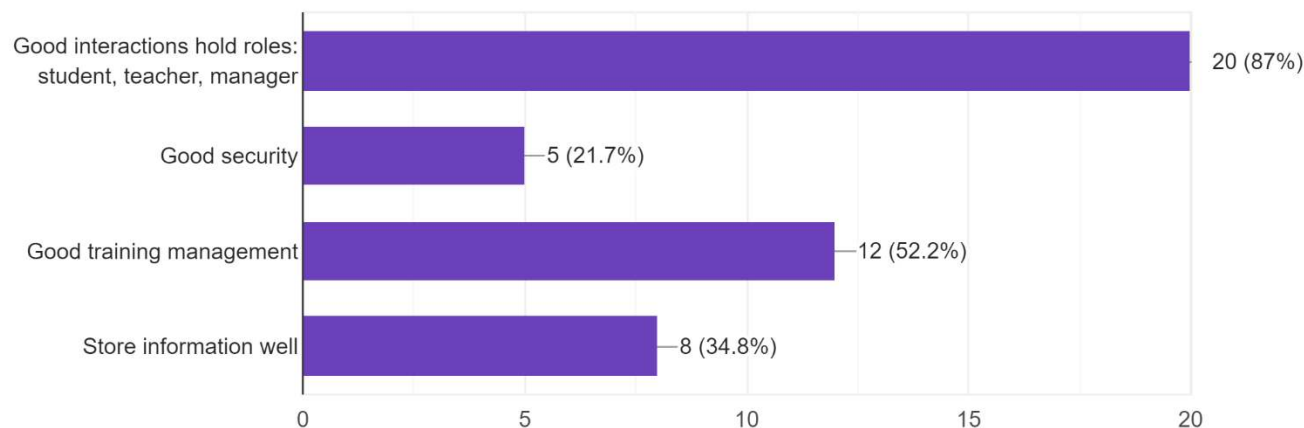
The strength of the application that is highly appreciated by users is the interactive features that keep members of the system such as: student, teacher, staff. Most people appreciate this feature about 87%. It follows the initial requirements of the system. Moreover, Training ability is also highly appreciated at 52.2% by features such as: staff assigns teachers to teach or teachers assign exercises to students. The

security factor is rated at 21.7%, relatively low compared to other factors because here we only decentralize to members without providing tools or methods to secure such as: user information confidentiality, password forgotten function.

In your opinion, what is the feature missing from this 'Training' web app?

4 responses

Add functions for students to post their homework and teachers can mark papers right on the system.

add assessment functionality so students can evaluate subjects and teachers

Added password forgotten feature
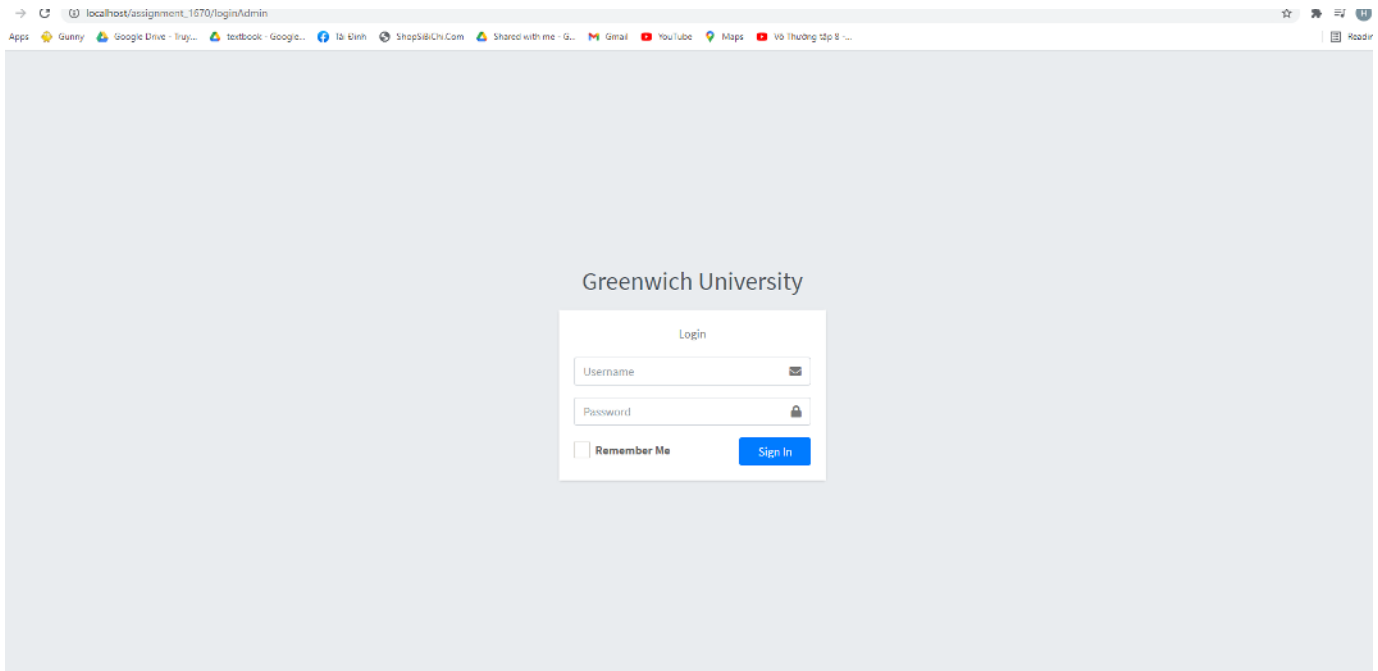
need information of person, ex: avatar

Above surveyed customers also give some missing features for the application so that we can develop and improve it in the future such as: add avatar for user, forgot password, feedback, mark point for student.

III. Implement web application
1. implement code
1. Login interface admin
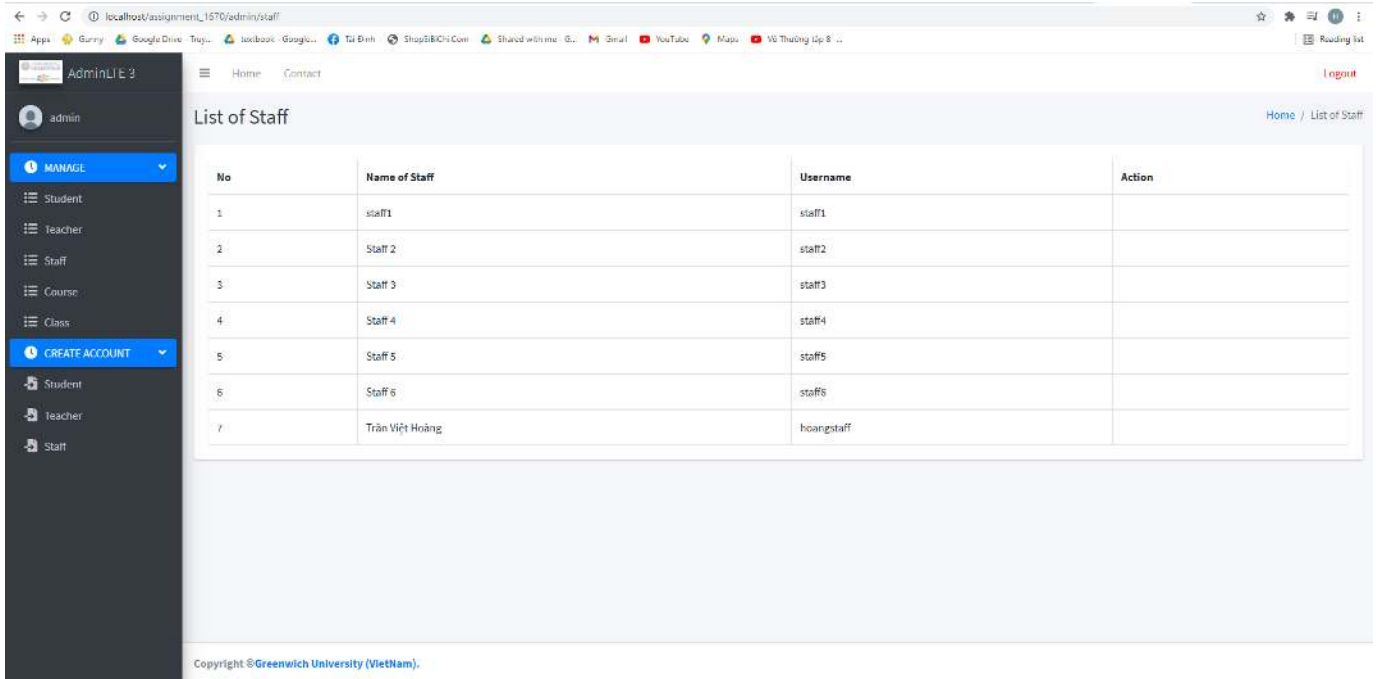
The login form will occur after the user has first entered the system; in order to login, the user must first input their username and password, which the system will check by comparing the data in the database. If all is right, the user will be routed to the menu's homepage.
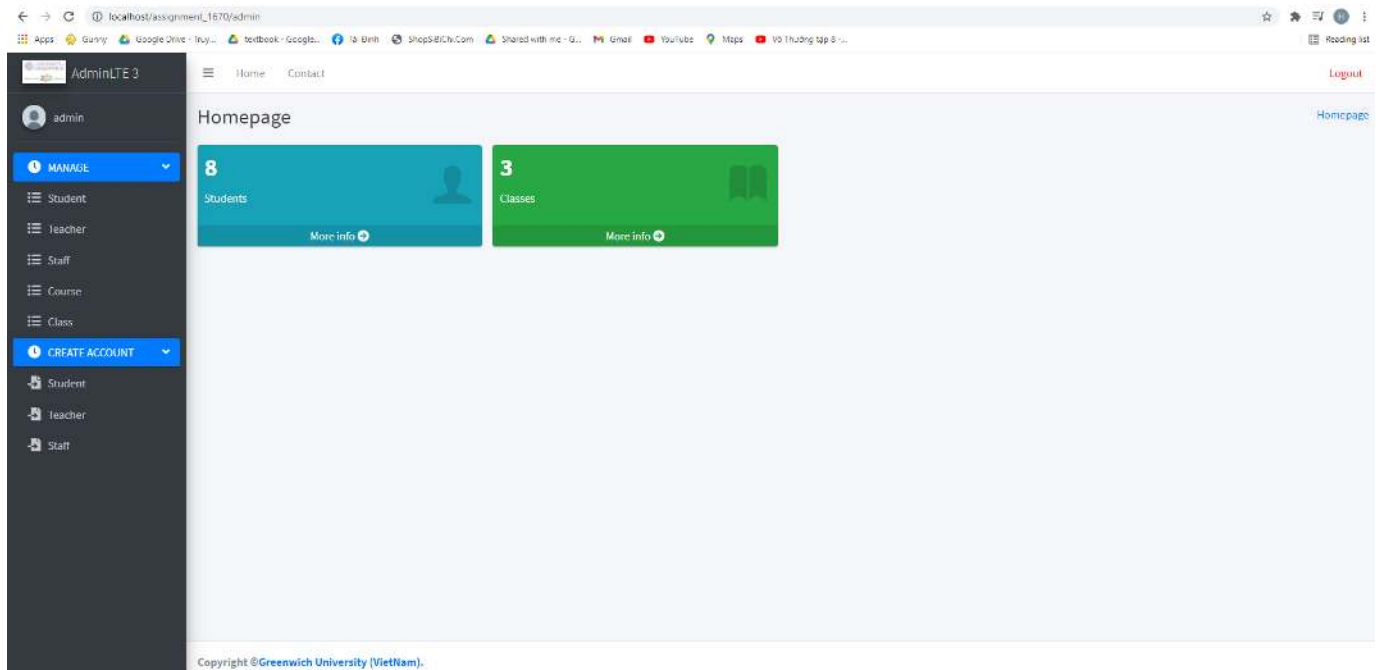
When login Homepage of admin is show. The admin home page will have the right to perform all functions on the system. Along with that it will show the number of classes available and the number of students in the system.
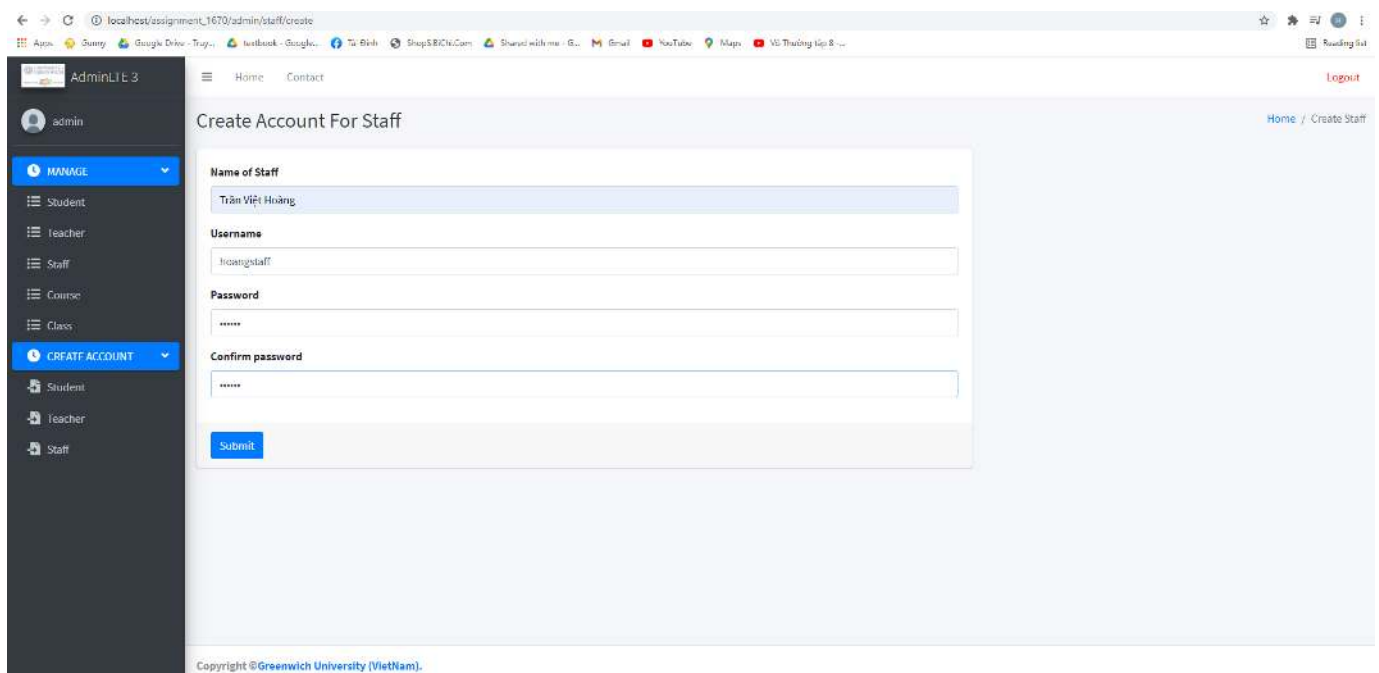
Administrator can create accounts for Staff. Here the administrator will create basic staff information such as name, account and password.
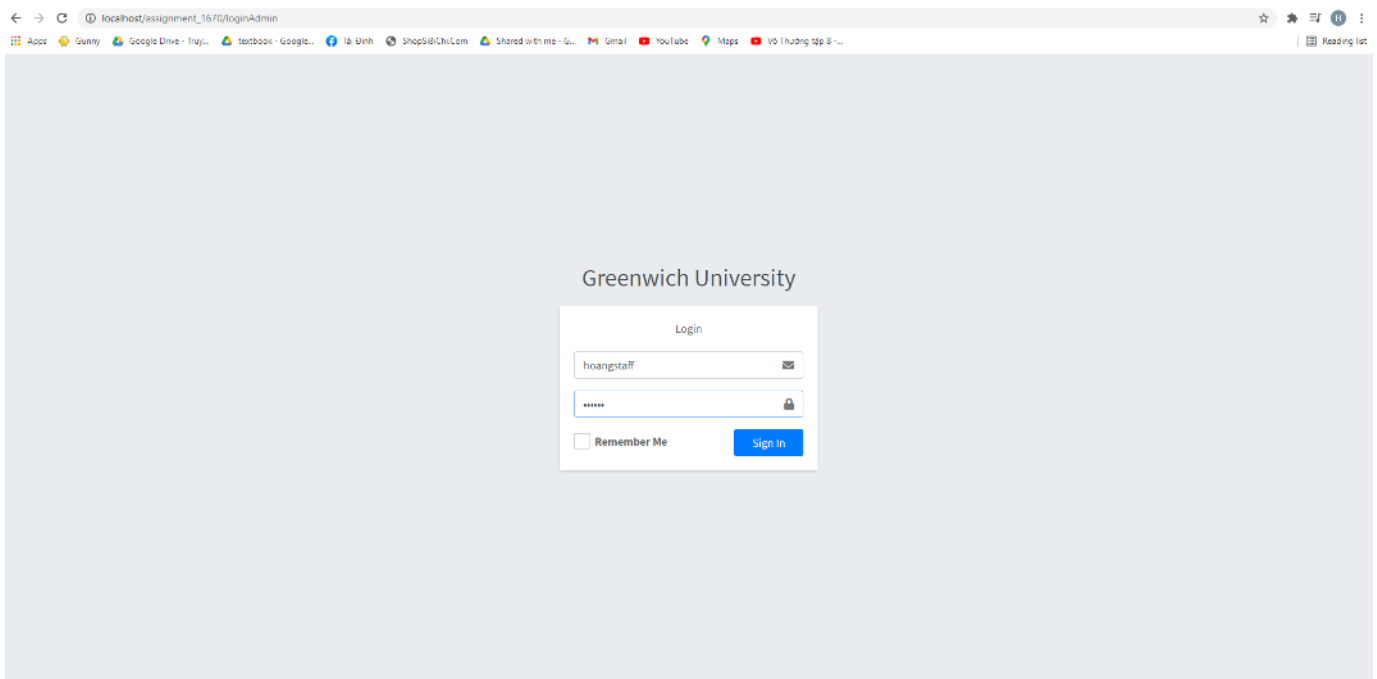
After creating an account for Staff, the administrator can see a list of staff, the administrator can edit or delete the staff if desired.
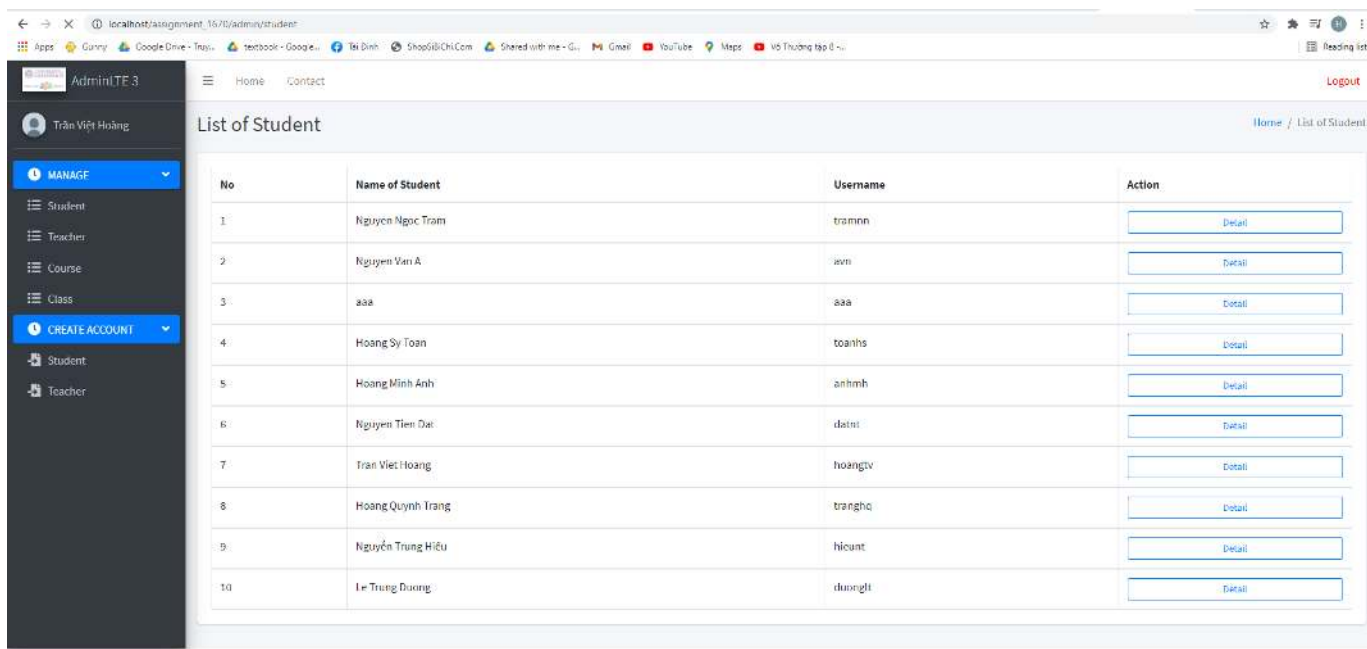
2 Login interface Staff

This is the login interface of the staff
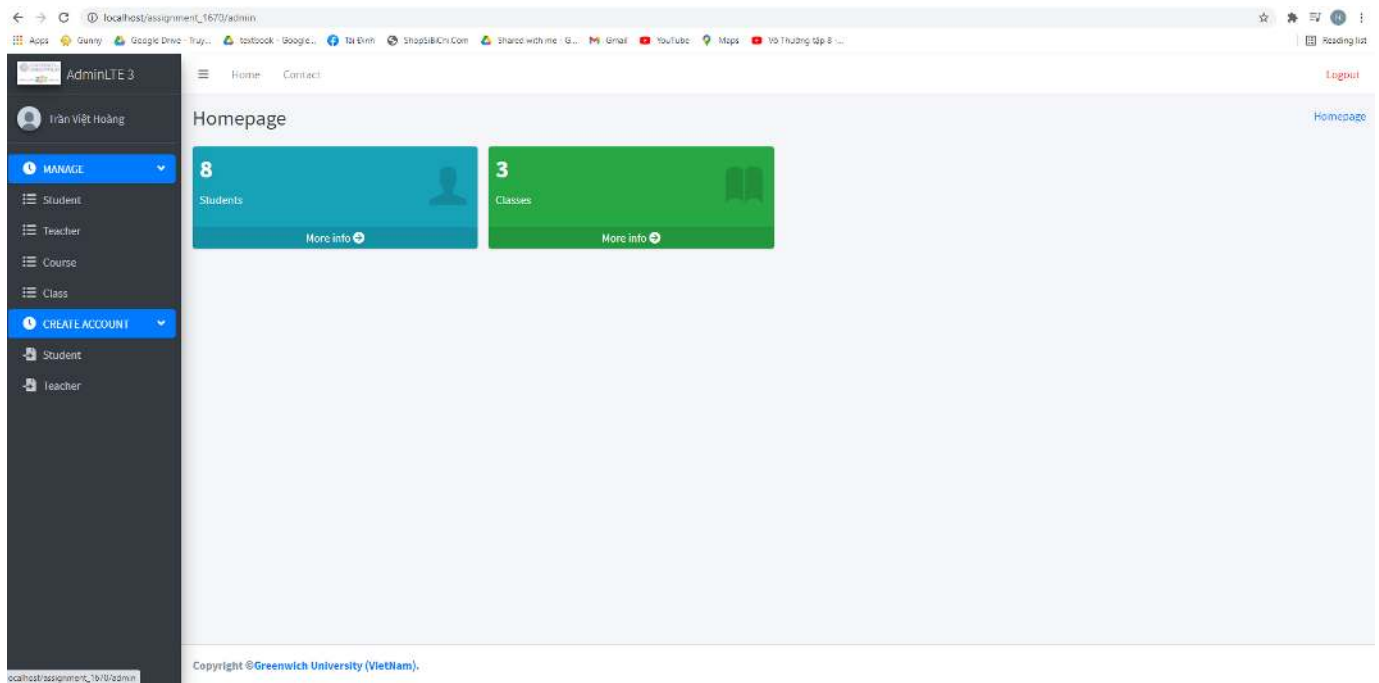
Similar to the administrator, the Staff can perform all functions in the administrative system, but the Staff cannot create an account for other Staff and cannot modify or delete another Staff account.
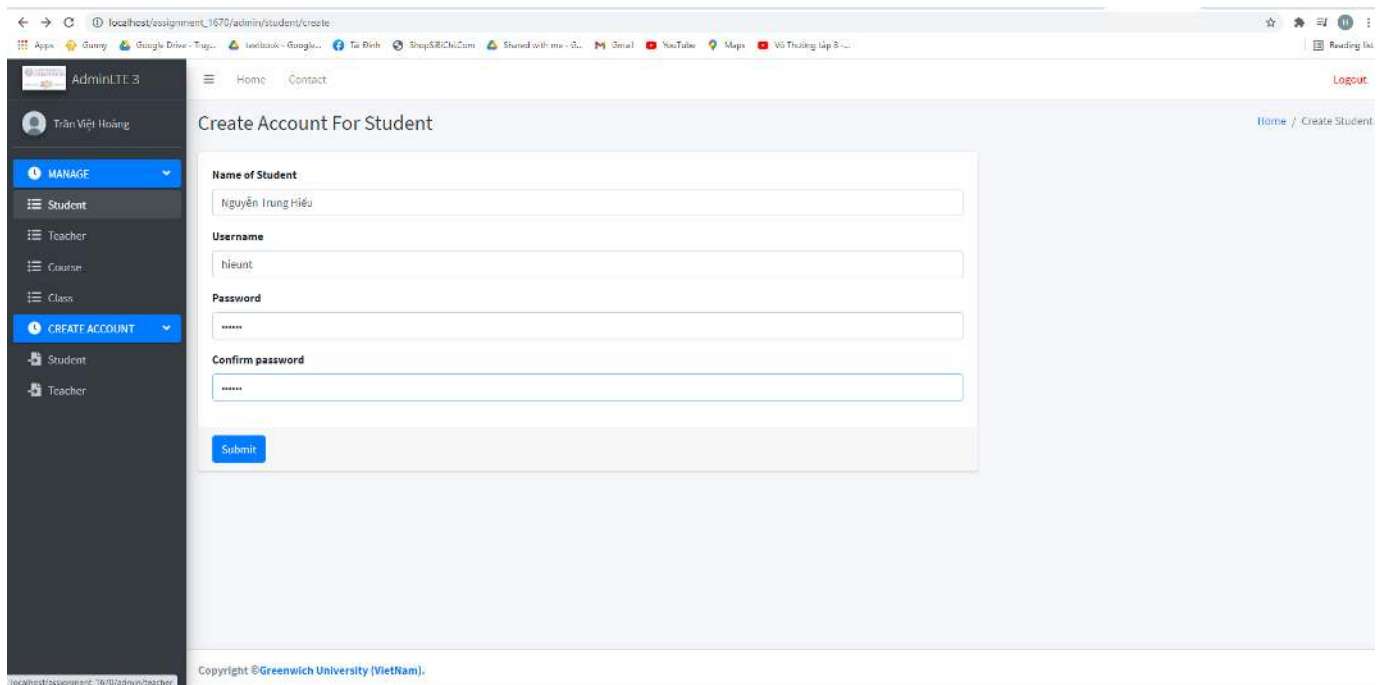
Staff can create accounts for trainee. Trainee account will include the name of the trainee, the login information on the system.

After creating the talent for the trainee, the system will display a list of the available trainees. Staff can observe preliminary information such as username and login account. To see more details click on the Detail button.

After clicking on the details, the staff can see the trainee's details including personal information and the class that the trainee is attending. The training status of the trainee is also displayed and the staff can delete the trainee at the same time.

When you click on the Information button, the basic information of the trainee will be displayed so that the Staff can edit and save that information.

Staff can add new trainer

Same strainee, system will show a list when add successfully.

When click button detail, After clicking on the Course button will display information about the courses that the trainer is teaching and the classes of trainers therein.

Class can edit info trainer
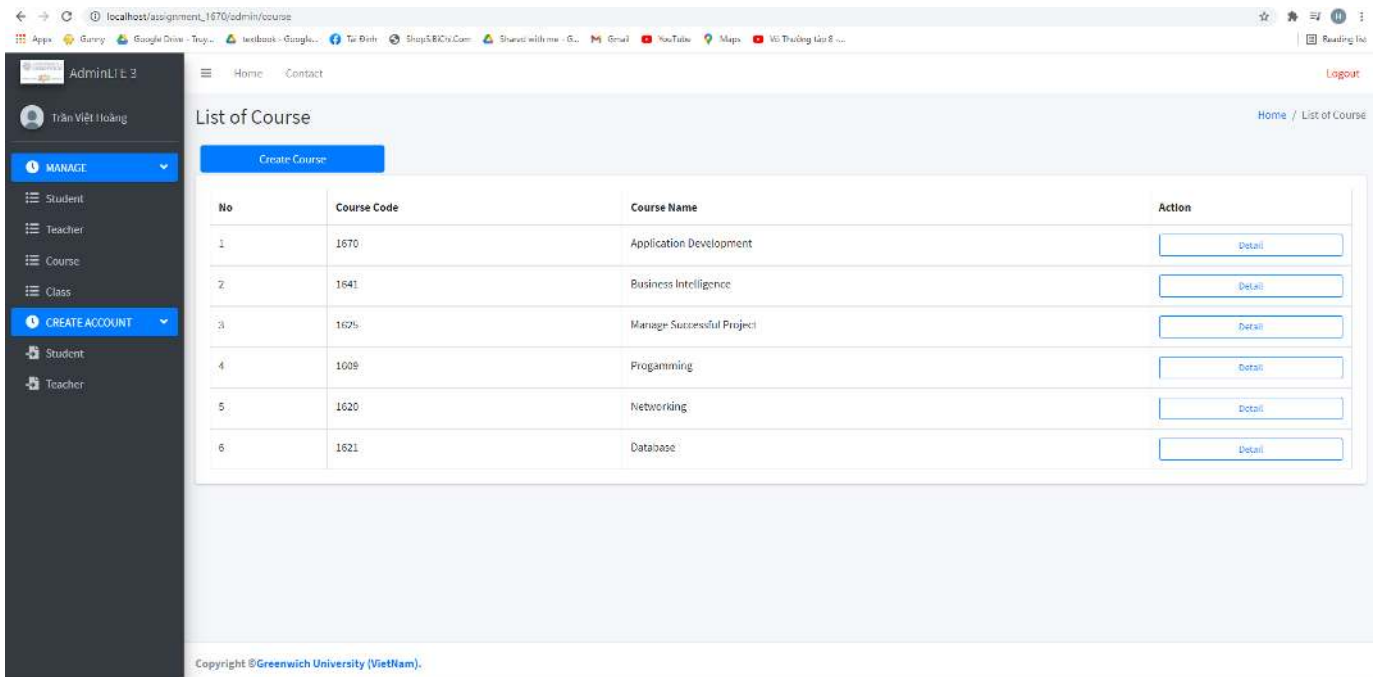
Staff can create a new course and add a name for the Course

After adding Course, Staff can observe a series of Course list that I have created. Staff can also manage the Course by deleting or modifying it.

Staff can create new class.

Similar to Course, Staff will have a Class list after class has been successfully added

It that the Staff presses specifically on the detail button, detailed information about the Class will appear. Here we have shown the details of the class name (GCH0717). The Staff can also add students to a Class as well as add a Course to a Class. Staff can also delete Class. Specifically, the class will display information about the number of trainees in the class, Course that that class is studying.
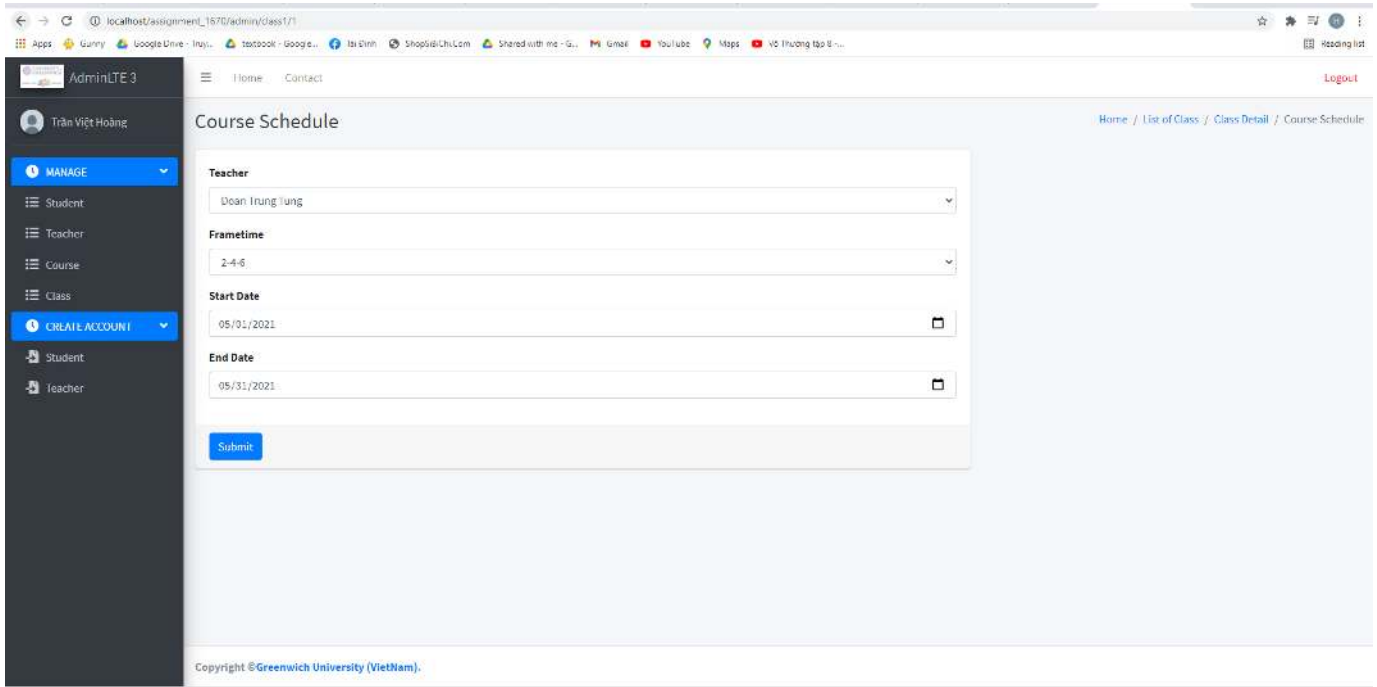
The example above Class will display the specific number of trainee currently in class is 1. Specifically, that trainee is named Nguyen Tien Dat. At the same time, GCH 0719 class has 1 Course which is Programing.
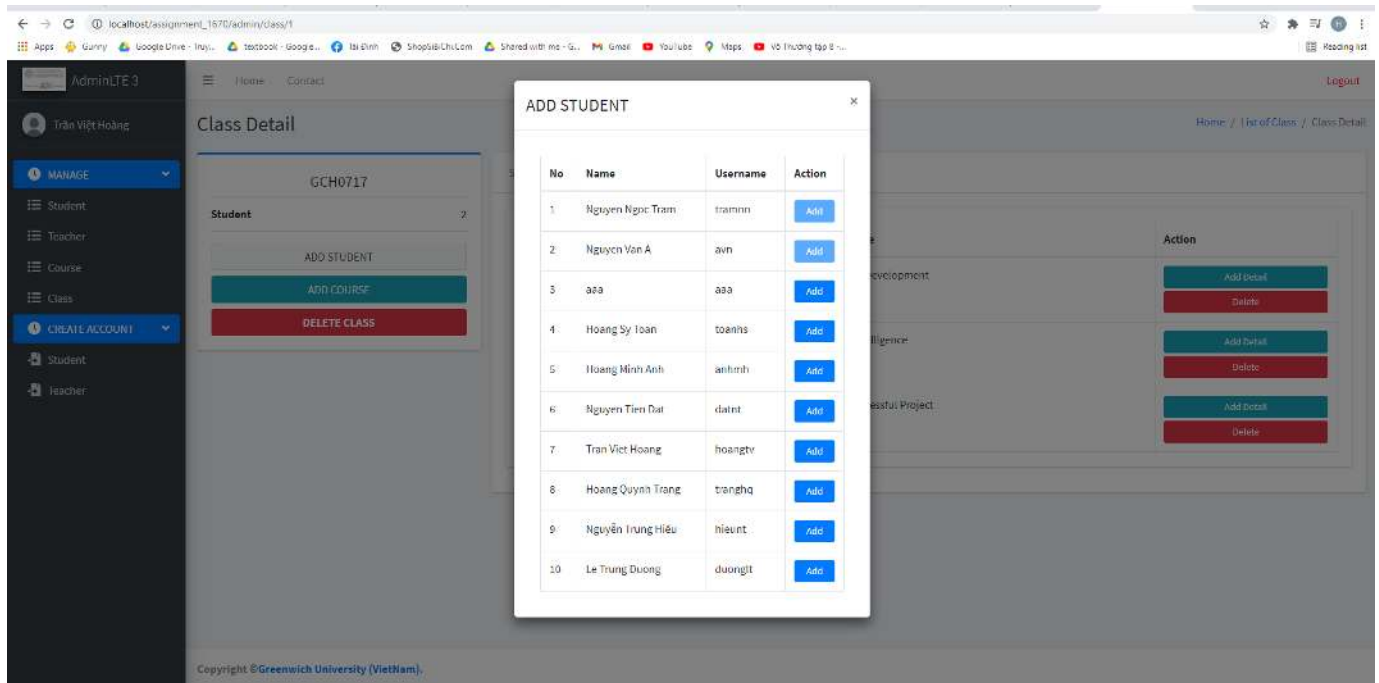
Staff performs the function of adding trainee to the class. Students who are already in the Class will not be able to show any buttons anymore.
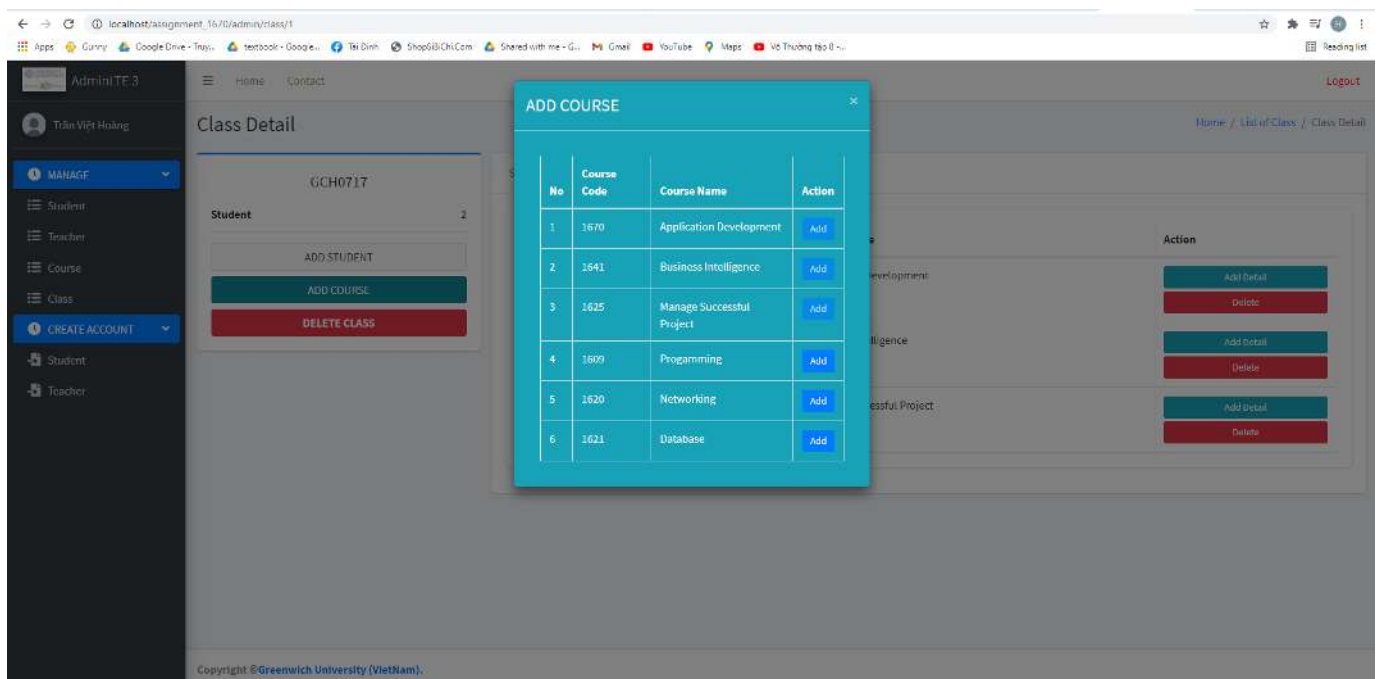
The staff can add the Course that the class needs to learn. The subjects that the class is studying cannot be added.

Staff can also add a Trainer and create a specific schedule for that Class.

3. Login interface trainer

Trainer can login system same Staff

After successful login, the trainer home page will display the classes he is teaching. By clicking on the view exercise button, the trainer can add exercises to that class for the trainee to view and do

Trainer can add exercises

Details of the exercise link will be displayed here, and their start and end times.
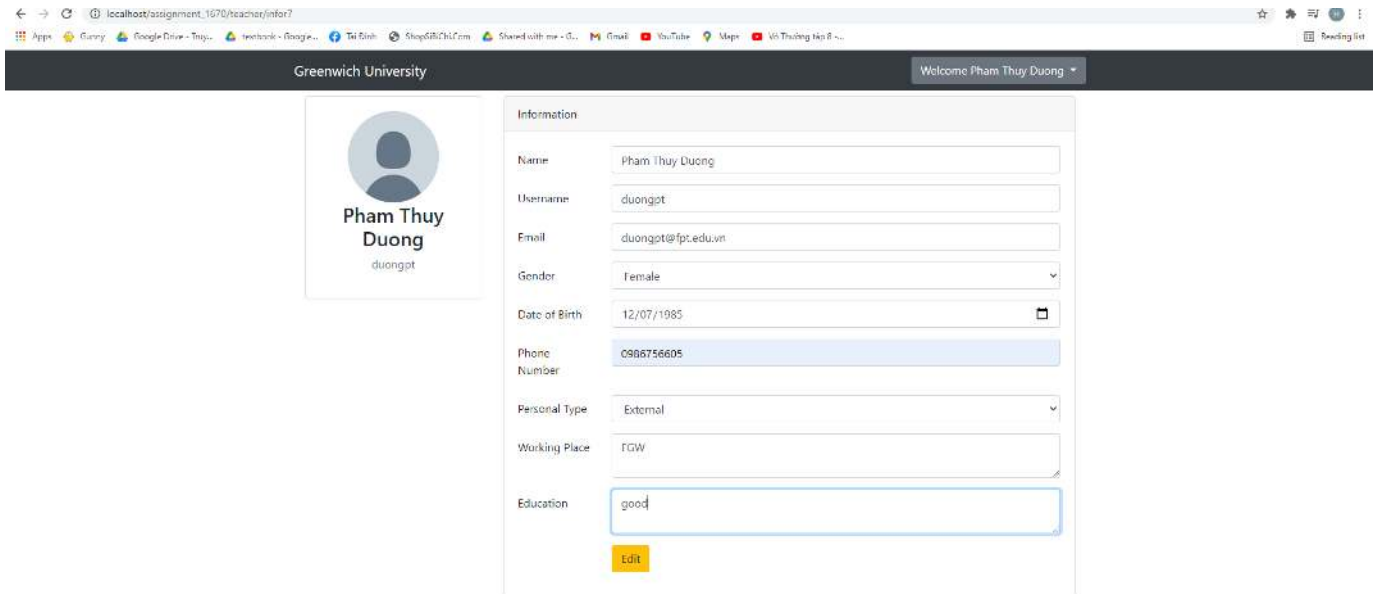
The Trainer can also review the work he uploaded and edit it if needed
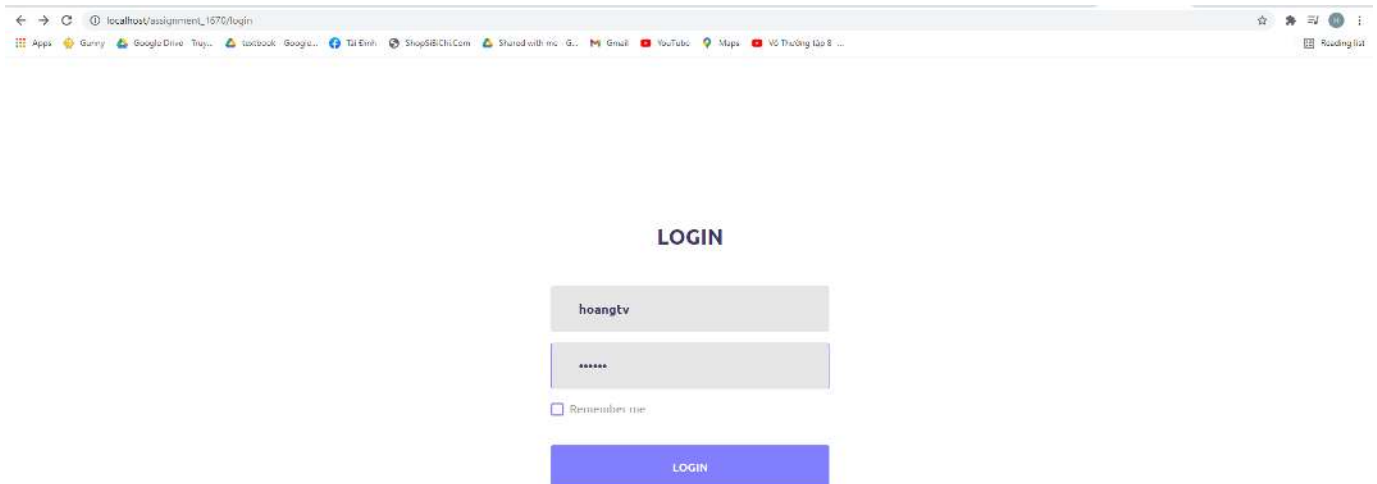
Trainer is also update information

4. Login interface trainee

Trainee can login the system
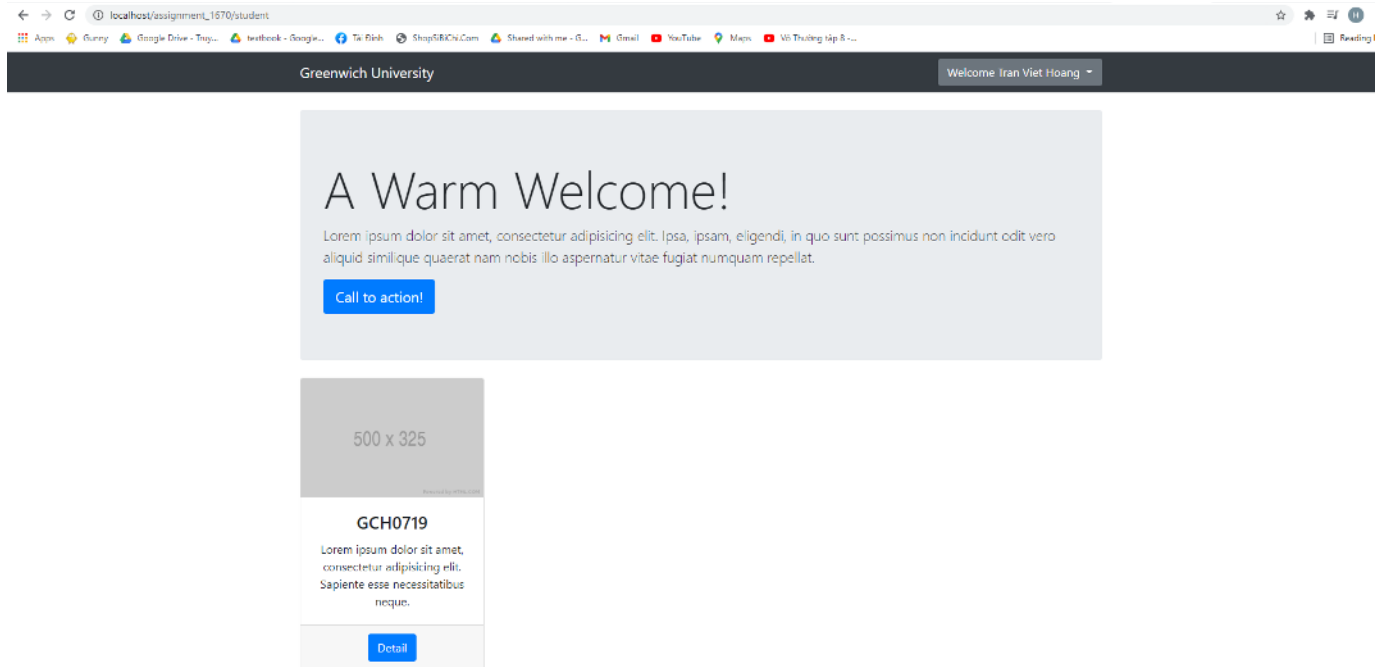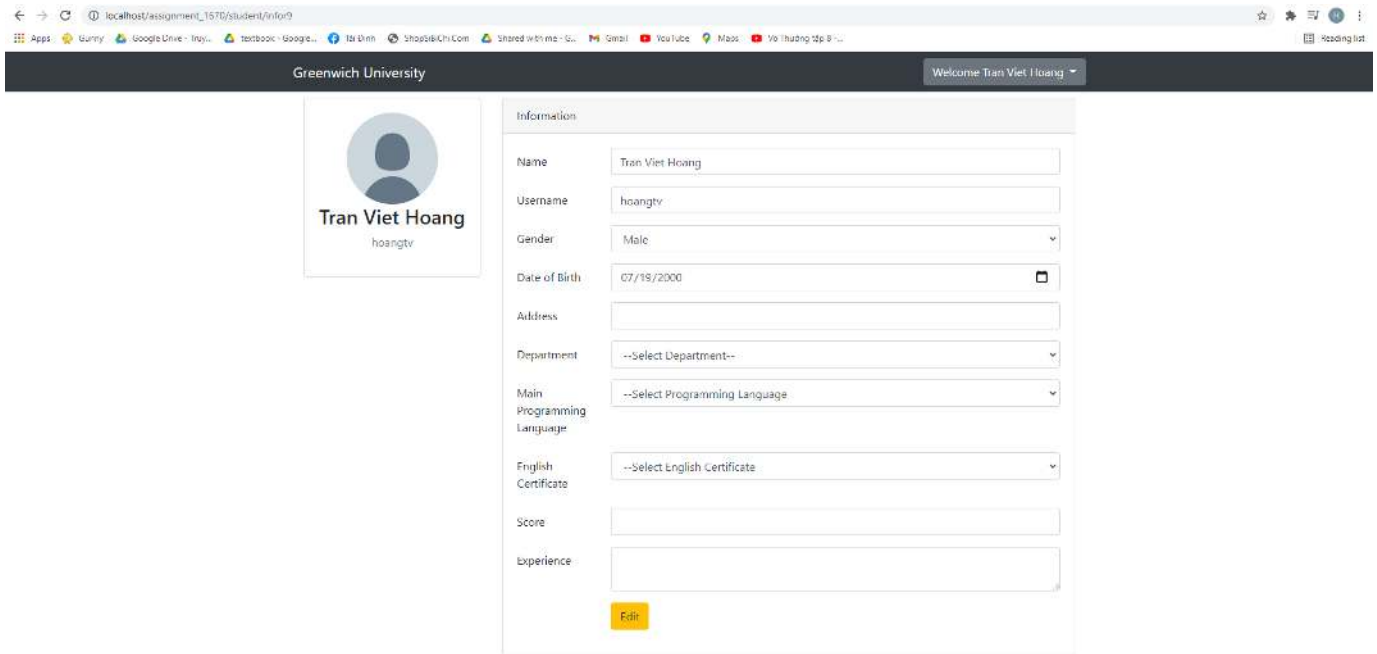
After successfully logging in, Trainee can see the classes she is attending.
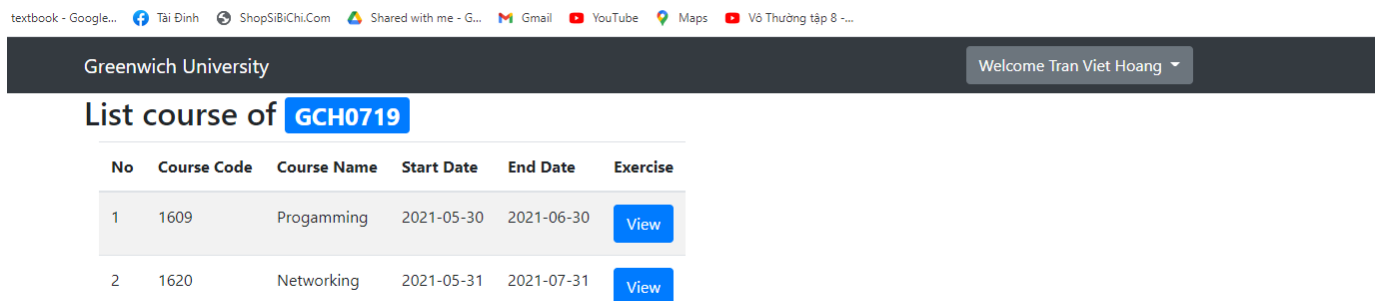
When you click on the details of a class, it will display the Course that the class is studying, detailed information about the class: start and end time. Also see the assignments for each class to be able to do
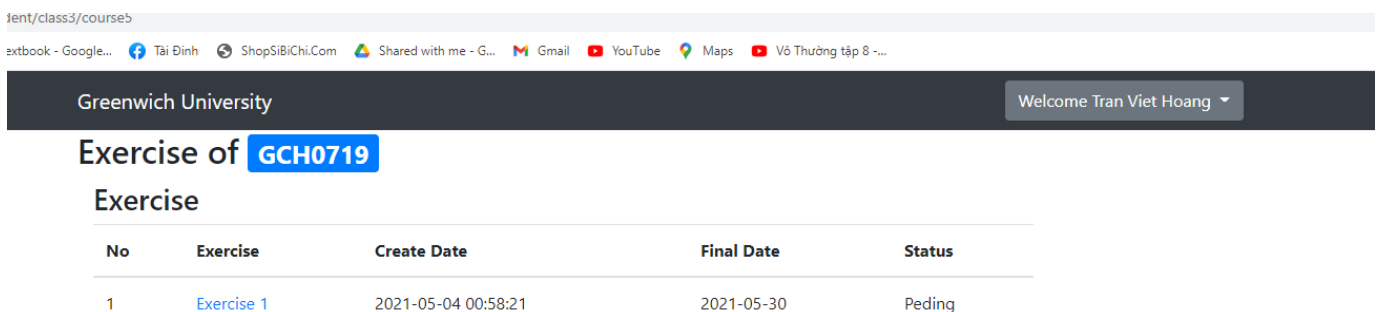
Assignment details: start time, assignment deadline, link to view assignments.

Trainee can also re-upload her personal information in the system.

2. Test case

No	Test case	Test detail		Input data	Expected output	Actual output	Status
1	Test Authorization function	Login with Correct password and username	Login as admin	Username: tuananhyht Password: tuananh123	Login successfully as admin	As expected	Pass
			Login as Staff	Username: toannt Password: toan123	Login successfully as staff	As expected	Pass
			Login as Teacher	Username: teacher1 Password: 123456	Login successfully as teacher	As expected	Pass
			Login as Student	Username: hoangnt Password:trumtinder123	Login successfully as student	As expected	Pass
		Login with incorrect password and username	Login as admin	Username: tuananhyht Password: quocanh123	Login Failed	As expected	Pass
			Login as Staff	Username: toannt1 Passsword: amdaldal	Login Failed	As expected	Pass
			Login as Teacher	Username: teacher1	Login Failed	As expected	Pass
			Login as Student	Username: hoangnt Password: fkboilamdong	Login Failed	As expected	Pass
		Logout	Logout as admin	Click logout	Logout successfully	As expected	Pass

			Logout as Staff	Click logout	Logout successfully	As expected	Pass
			Logout as Teacher	Click logout	Logout successfully	As expected	Pass
			Logout as Student	Click logout	Logout successfully	As expected	Pass
		Open login form before logout	as admin	Localhost:3000/	Redirect to homepage	As expected	Pass
			as Staff	Localhost:3000/	Redirect to homepage	As expected	Pass
			as Teacher	Localhost:3000/	Redirect to homepage	As expected	Pass
			as Student	Localhost:3000/	Redirect to homepage	As expected	Pass
		Open homepage without login	Admin homepage	Localhost:3000/admin	Throw exception "Unauthorized" "statusCode": **403**	As expected	Pass
			Staff homepage	Localhost:3000/staff	Throw exception "Unauthorized" "statusCode": **403**	As expected	Pass
			Student homepage	Localhost:3000/homepage/student	Throw exception "Unauthorized" "statusCode": **403**	As expected	Pass
			Teacher homepage	Localhost:3000/homepage/teacher	Throw exception "Unauthorized"	As expected	Pass

2	Create new Staff	Open create form	As admin	Localhost:3000/staffs/create	Display create staff interface	As expected	Pass
			As other role	Localhost:3000/staffs/create	Throw exception "Unauthorized" "statusCode": **403**	As expected	Pass
		Create new staff	Normal case	Staff_address = "30 Nguyen Du"	Create successfully	As expected	Pass
			Validation	Staff_first name = "" Staff_last name = ""	Throw exception "must enter name"	As expected	Pass
3	View staff list			As admin	Localhost:3000/staffs/index	Create successfully	As expected	Pass
				As other role	Localhost:3000/staffs/index	Throw exception "Unauthorized" "statusCode": **403**	As expected	Pass
4	Update Staff	Update information	Change something	Staff_address = "124 Nguyen Trai"	Update successfully	As expected	Pass
			Change something	Staff_first name = "" Staff_last name = ""	Throw exception "must enter name"	As expected	Pass

5	Delete Staff	Choose delete or not		Choose ok	Delete successfully	As expected	Pass
				Choose cancel	Delete canceled	As expected	Pass
6	Create new teacher	Open create form	As admin	Localhost:3000/teacher/create	Display create teacher interface	As expected	Pass
			As other role	Localhost:3000/teacher/create	Throw exception "Unauthorized" "statusCode": **403**	As expected	Pass
		Create new teacher	Normal case	Teacher_address = "124 Nguyen Trai"	Create successfully	As expected	Pass
			Validation	Teacher_firstname ="" Teacher_lastname = ""	Throw exception "must enter name"	As expected	Pass
7	View teacher list		As admin	Localhost:3000/teacher/index	Create successfully	As expected	Pass
			As other role	Localhost:3000/teacher/index	Throw exception "Unauthorized" "statusCode": **403**	As expected	Pass
8	Update teacher	Update information	Change something	Teacher _address = "20 Kim Ma"	Update successfully	As expected	Pass
			Change something	Teacher _firstname = "" Teacher _lastname = ""	Throw exception "must enter name"	As expected	Pass

9	Delete teacher	Choose delete or not		Choose ok	Delete successfully	As expected	Pass
				Choose cancel	Delete canceled	As expected	Pass
10	Create new Student	Open create form	As Staff	Localhost:3000/Student/create	Display create trainee interface	As expected	Pass
			As other role	Localhost:3000/Student/create	Throw exception "Unauthorized" "statusCode": **403**	As expected	Pass
		Create new Student	Normal case	student_address = "36 Hoang Cau"	Create successfully	As expected	Pass
			Validation	student_firstname ="" student_lastname = ""	Throw exception "must enter name"	As expected	Pass
11	View student list		As admin And staff	Localhost:3000/student/index	Create successfully	As expected	Pass
			As other role	Localhost:3000/student/index	Throw exception "Unauthorized" "statusCode": **403**	As expected	Pass
12	Update student	Update information	Change something	student_address = "124 Nguyen Trai"	Update successfully	As expected	Pass
			Change something	student_firstname ="" student_lastname = ""	Throw exception "must enter name"	As expected	Pass

13	Delete student	Choose delete or not		Choose ok	Delete successfully	As expected	Pass
				Choose cancel	Delete canceled	As expected	Pass
14	Create new Course	Open create form	As Staff	Localhost:3000/course/create	Display create categories interface	As expected	Pass
			As other role	Localhost:3000/class/create	Throw exception "Unauthorized" "statusCode": 403	As expected	Pass
		Create new course	Normal case	Class_id = "GCH0717"	Create successfully	As expected	Pass
			Validation	Class_id = ""	Throw exception "must enter name"	As expected	Pass
15	Create new Subject	Open create form	As Staff	Localhost:3000/subject/create	Display create topics interface	As expected	Pass
			As other role	Localhost:3000/subject/create	Throw exception "Unauthorized" "statusCode": 403	As expected	Pass
		Create new subject	Normal case	Subject_name = "Programing"	Create successfully	As expected	Pass
			Validation	Subject_name = ""	Throw exception "must enter name"	As expected	Pass

16	Create new Course_detail	Open create form	As Staff	Localhost:3000/course_detail/create	Display create course_detail interface	As expected	Pass
			As other role	Localhost:3000/course_detail/create	Throw exception "Unauthorized" "statusCode": **403**	As expected	Pass
		Create new category	Normal case	Select Teacher,Subject and course from dropdown	Create successfully	As expected	Pass
17	View profile	View profile of teacher	As teacher	Localhost:3000/homepage/profile	Show all incorrect information of logged user	As expected	Pass
		View profile of student	As student	Localhost:3000/homepage/profile	Show all incorrect information of logged user	As expected	Pass
18	Change password	User change their password	As teacher	Password = "newpassword"	Current password ="newpassword"	As expected	Pass
19	Change password	User change their password	As student	Password = "newpassword2"	Current password ="newpassword2"	As expected	Pass
20	View My Course	View enrolled courses	As teacher	http://localhost:3000/homepage/courseteacher?teacher_id=4	Show all course of teacher_id = 4	As expected	Pass

21	Create exercise	Upload link exercise	As teacher	https://docs.google.com/forms/d/1Qvle hjlfzxl9m5mf1NJrp hBbxLMrKfVy7P5 GrNCC-uM	showing 1 link to the exercise	As expected	Pass
22	View exercuse	View exercuse	As teacher	Link1	showing 1 link to the exercise	As expected	Pass
			As student	Link1			

3. Supportive evidence of using the preferred tools, techniques and methodologies

All functions in the use case diagram (drawn with the Draw.io tool) have been implemented and thoroughly tested above. It includes all the mentioned features like: login, logout and some CRUD functions for each role. Authorization is also handled so that each user can only use the authorized functions.
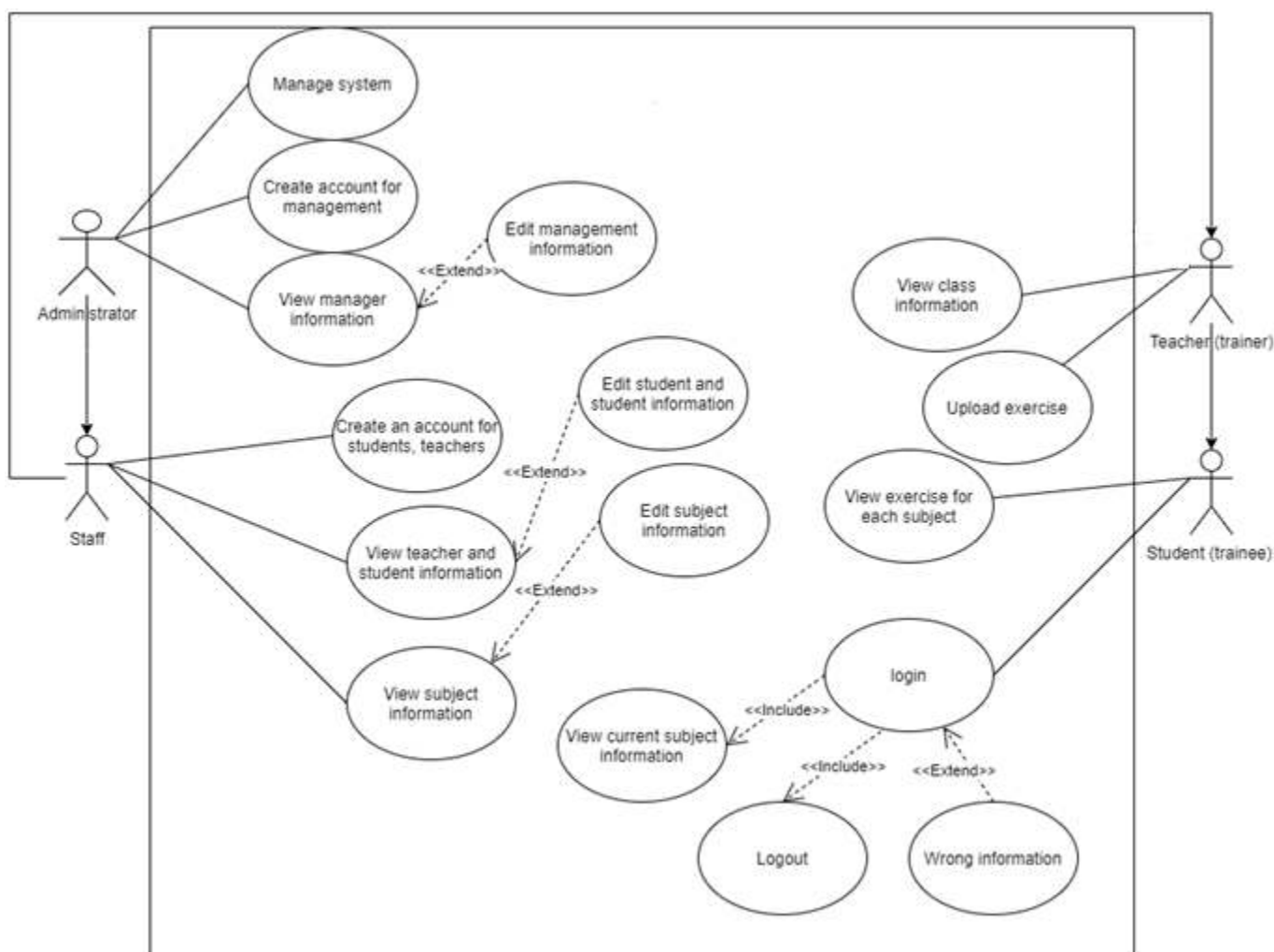
Entities designed in class diagrams have also been implemented in code. Each entity will also be designed to include the properties and methods mentioned in the class diagram.
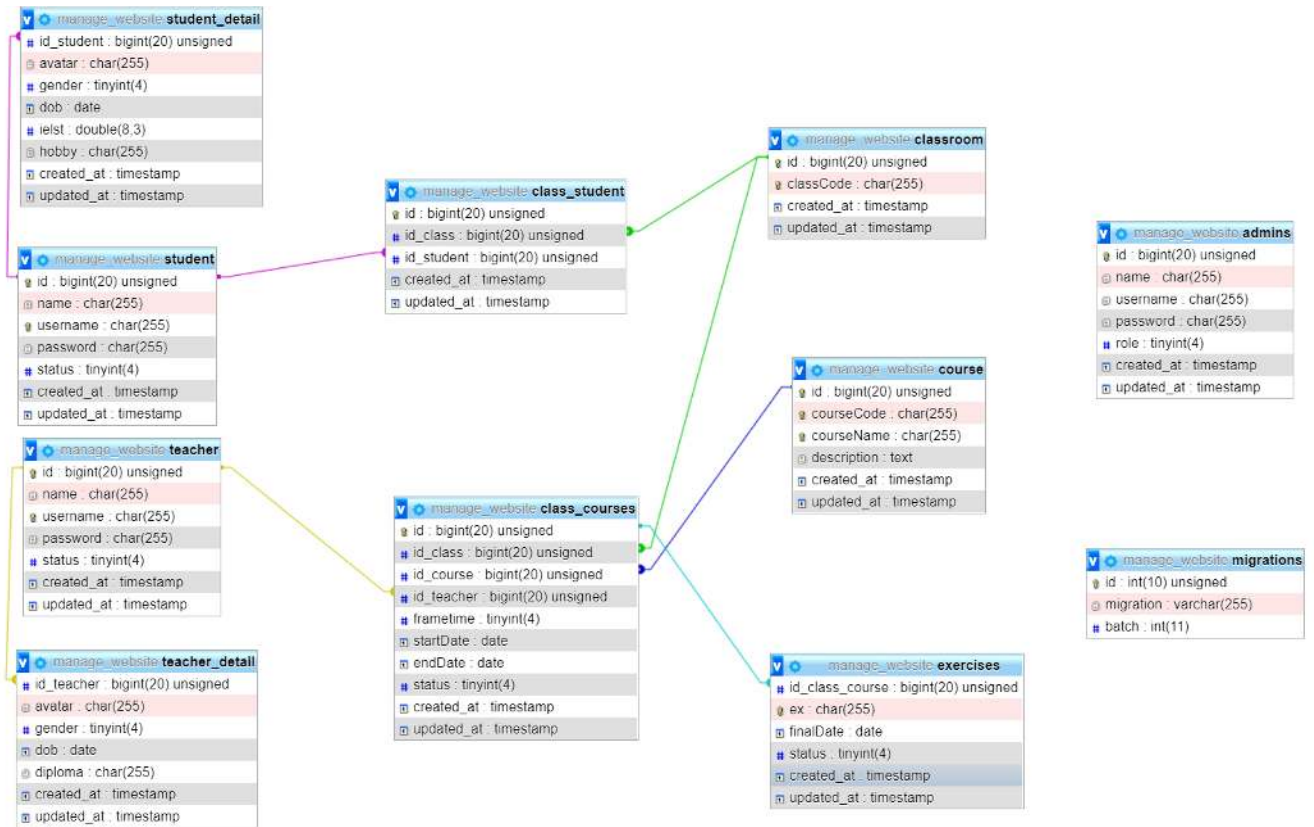
The above is Advanced Entity-Relationship. I used PhpMyAdmin software with automatic drawing function based on existing database. In addition, during development, I added a number of attributes to better suit the needs of customers such as semester properties, information users, ...

My software is also completed on the lavavel framework's support project and main IDE is VS code

Start using the command line to initialize the project with the above command.

The project will be automatically initialized in the VS code part according to the MVC model. From there, users can rely on the supported parts to develop their own code.

IV. application review

1. Comparison to initial requirements

As the app developer, we research all the specifications first after analyzing and comparing the device with the generated properties. We have completed a product with full features for 4 roles: administrator, staff, trainer, trainee. The features have also been decentralized with suitable roles to be used: add, edit, delete accounts, courses. During other login attempts, we do not see any illegal authority. This also shows that we have a good mix of decentralization and adherence to the previous criteria.

In employee, log in and manage courses, coaches and topics without problems. This also proves that we have a good mix of decentralization and adherence to previous criteria as admin.

In Teachers (Instructors) and Students (Students), they are ready to log in and view the course, updating their profiles.

Role	Function	Implement during	Initial	Result

			project	Requirement	
Admin	Login	Create interface and back-end based on wireframe. Must have an admin account to login	Login to system through first page	Pass
	Create, edit, delete Account for trainer, training staff	Can create, edit, delete accounts for trainer, staff, trainee	Can be accessed to management system to control trainer / staff /trainee account	Pass
Staff	Login	Create interface and back-end based on wireframe. Must have an staff account to login	Login to system through first page	Pass
	Add information for trainer, trainee	Information fields such as: trainee name, trainee accounts, age, date of birth, education, main programming language, TOEIC score, experience details, department, location	trainee name, trainee accounts, age, date of birth, education, main programming language, TOEIC score, experience details, department, location	Pass
	Create, edit, delete Student, Teacher Account	Can create, edit, delete accounts for teacher and student	Can be accessed to management system to control teacher / staff account	Pass
	Create, edit, delete subject	Subjects can be added, edited and deleted as needed.	Can be accessed to management system to control subject	Pass
	Add topics	topic name and topic descriptions	Add topics	Pass
	Create, edit, delete class	Add student in class	Add class	Pass

	Manage trainer profile	adding, updating and deleting the information: Trainer name, Personal type, working place, telephone, and email address.	Trainer name, External or Internal Type, working place, telephone, and email address	Pass
	assign trainer to a topic	Add trainer into subject	assign trainer to a topic	Pass
	assign trainee to a course	Add trainee into class have course	assign trainee to a course	Pass
Teacher(trainer)	Login	Create interface and back-end based on wireframe. Must have an teacher account to login	Login to system through first page	Pass
	Edit information	trainer can be added, edited information	Update profile	Pass
	View course	View assigned course	View course	Pass
	Upload exercise	Update assignments to students	None	Pass
Student(trainee)	Login	Create interface and back-end based on wireframe. Must have an student account to login	Login to system through first page	Pass
	View class, exercise, study time	Displays detailed information about classes, assignments and time for students	none	Pass

2. Analysis the factor

The factors affecting the performance of a business application have been objectively considered and analyzed by us. During the design process, I took into account a lot of situations and the ability of the user during use to affect the performance of the program. Here I will talk about it:

- The case of data validation can affect the integrity of the authenticated data. Such as leaving age information, language of instruction, ... All authentication cases are processed on the server side, the application requires to fill in all information, avoiding the user forgetting.

- I have handled all authorization and authentication and security cases for each function separately. For example, if someone tries to access the link of the function that their account does not have permission to go to, it will be blocked and an exception will be reported. In case they are not logged in but want to go to the path, although only in the view, it is still blocked and return to the login form.

- With creating accounts for students, teachers, and staff, it is essential that the accounts will not overlap. For example, students with the same name when the staff create an account for them will have to take a different username. If the same matches will be sent to the account that existed before. This allows staff and teachers to differentiate students based on their username even if they have identical names.

- While evaluating the code, we identified the factors that make the code duplicate, complex, and difficult to debug. We used the MVC pattern and the lava framework. All my source code is modular and functional. when implementing code we will start based on the modules generated by lavavel. This makes it easier for me to grasp the flow of data and more flexible in finding and fixing errors. In addition, functional modularization makes code clean, tidy, and reusable avoiding memory-consuming repetition. Such as the everywhere in the program import and export functions.

V. Conclusion

My system has met most of the requirements outlined in report 1. Our system has passed all the test cases and test plans presented in the test cases section and determine the factors affecting the performance of the application. We need to improve and upgrade the system to be able to respond and enhance the user experience.