**ASSIGNMENT 2 FRONT SHEET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification** | **BTEC Level 5 HND Diploma in Computing** | | |
| **Unit number and title** | Unit 30: Application Development | | |
| **Submission date** |  | **Date Received 1st submission** | 11/19/2019 |
| **Re-submission Date** |  | **Date Received 2nd submission** | 11/19/2019 |
| **Student Name** | Nguyen Trung Duc | **Student ID** | GCH17377 |
| **Class** | GCH0705 | **Assessor name** | Truong Cong Doan |
| **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** |  |

**Grading grid**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| P4 | P5 | P6 | M3 | M4 | M5 | D2 | D3 |
|  |  |  |  |  |  |  |  |

Table of Contents

[List of tables: 2](#_Toc28944263)

[List of figures: 2](#_Toc28944264)

[Introduction: 2](#_Toc28944265)

[I Peer review 3](#_Toc28944266)

[Formal questionanarie 3](#_Toc28944267)

[Feedback analysis 5](#_Toc28944268)

[Suggestions and possible improvement 8](#_Toc28944269)

[II Application development 9](#_Toc28944270)

[2.1 Current system interface 9](#_Toc28944271)

[2.2 Test case: 20](#_Toc28944272)

[2.2.1 Admin function testing: 20](#_Toc28944273)

[2.2.2 Trainer function testing: 21](#_Toc28944274)

[2.2.3 Trainee function testing: 21](#_Toc28944275)

[III Application review 21](#_Toc28944276)

[3.1 Comparison to initial requirements 21](#_Toc28944277)

[3.1.1 Functional features: 21](#_Toc28944278)

[3.1.2 Non-functional features: 23](#_Toc28944279)

[3.2 Assessment of the system’s strength and weakness 24](#_Toc28944280)

[3.3 Indentified risks 24](#_Toc28944281)

[Conclusion 24](#_Toc28944282)

[References: 24](#_Toc28944283)

# List of tables:

# List of figures:

# Introduction:

In assignment 1, our team discussed and made decisions about the appropriate technologies and methods that will be applied in the FPT co (CMS) system. It is a blueprint that includes sections such as requirements, existing technologies and methods for managing and developing a cms system and, ultimately, an objective evaluation of the selection, to make the most appropriate decision.

 We have built a system based on the plan given earlier. In this section, we act as testers, perform system performance tests against initial requirements, conduct a small survey by the programmers, and evaluate the strengths. weaknesses as well as giving solutions.

# I Peer review

## Formal questionanarie:

The following is a set of questions for survey research. Our target audience is IT students (They are technologically savvy people, aged 18-25). The questionnaire includes both quantitative and qualitative questions. We hope this study will gain the necessary evaluation information for our system.

1. If you are to rate the cleanliness of the code, what will it be?

a.1\*

b.2\*

c.3\*

d.4\*

e.5\*

2. On a scale of 1 to 5 how will you rate the web interface?

a.1\*

b.2\*

c.3\*

d.4\*

e.5\*

3. On a scale of 1 to 5 how will you rate the web features?

a.1\*

b.2\*

c.3\*

d.4\*

e.5\*

4. Presentation Skill

a.1\*

b.2\*

c.3\*

d.4\*

e.5\*

5. Personal recommendation

6. What system features need improvements?

7. Problems needed to changed when selecting tools?

8. Is the scope and purpose of the system reasonable?

a. Yes

b. No

9. Evaluation of the display of the group on a scale of 1 - 5.

a.1\*

b.2\*

c.3\*

d.4\*

e.5\*

10. Is the SRS Good enough?

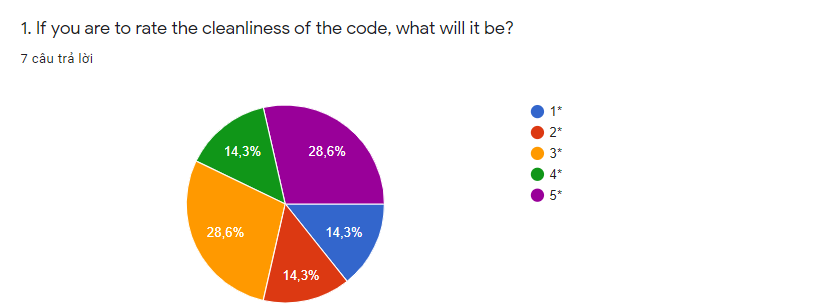
a. Yes

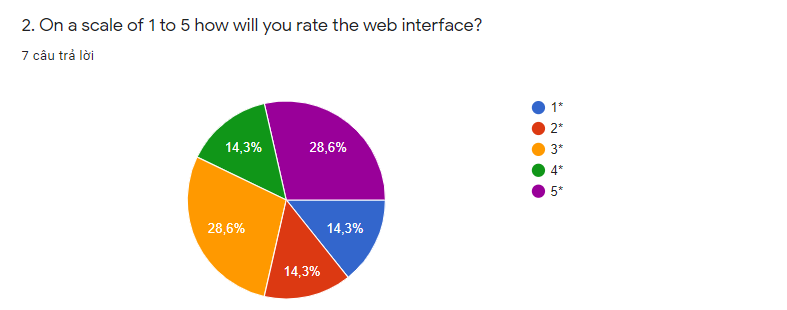
b. No

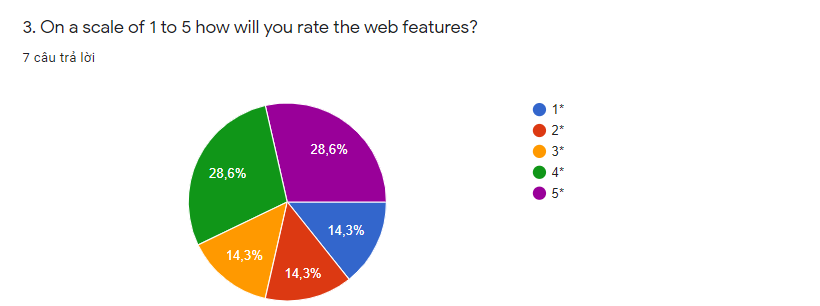
c. May be

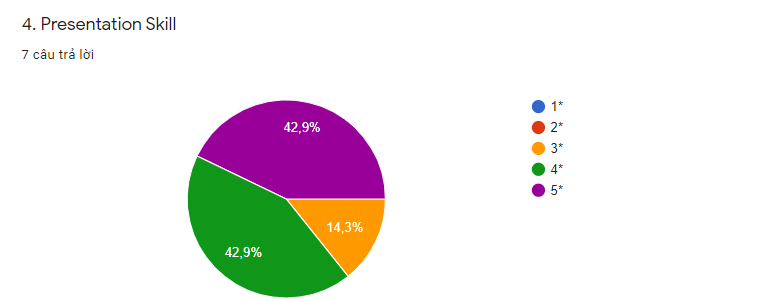
## Feedback analysis

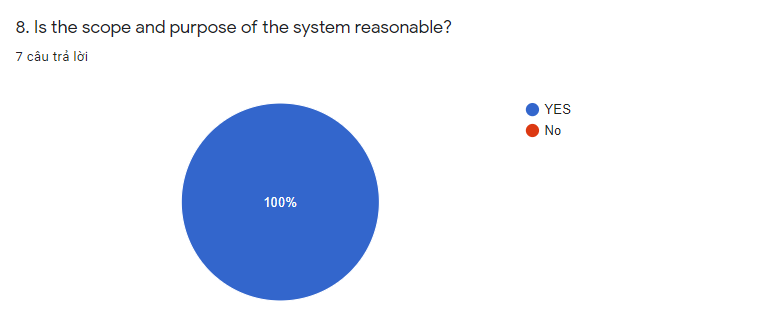
Here are the results we have obtained after conducting survey research:

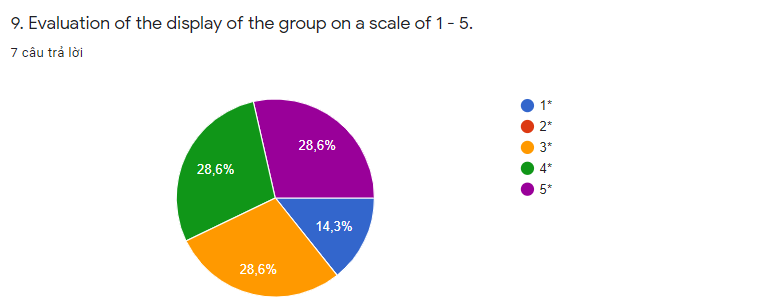


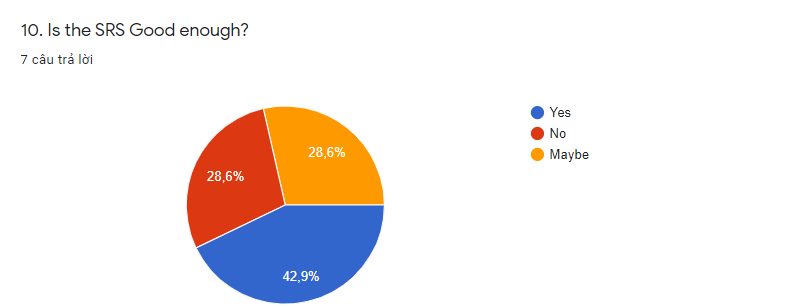




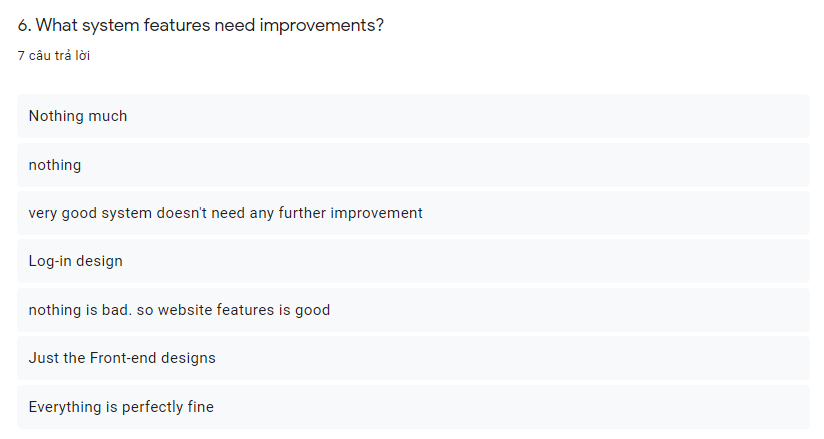


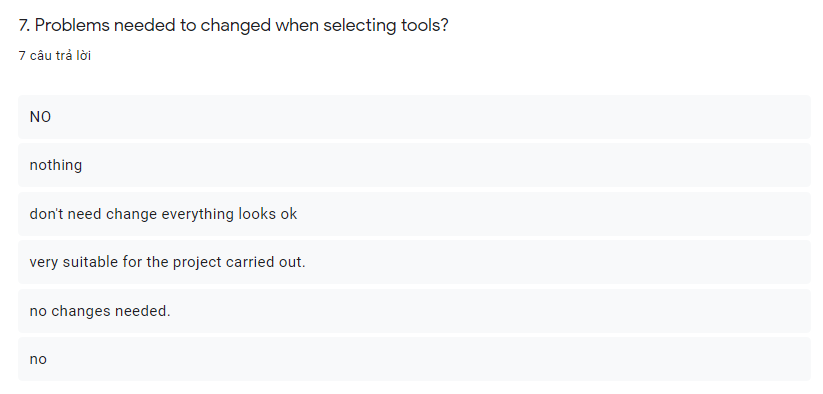












## Suggestions and possible improvement

# II Application development

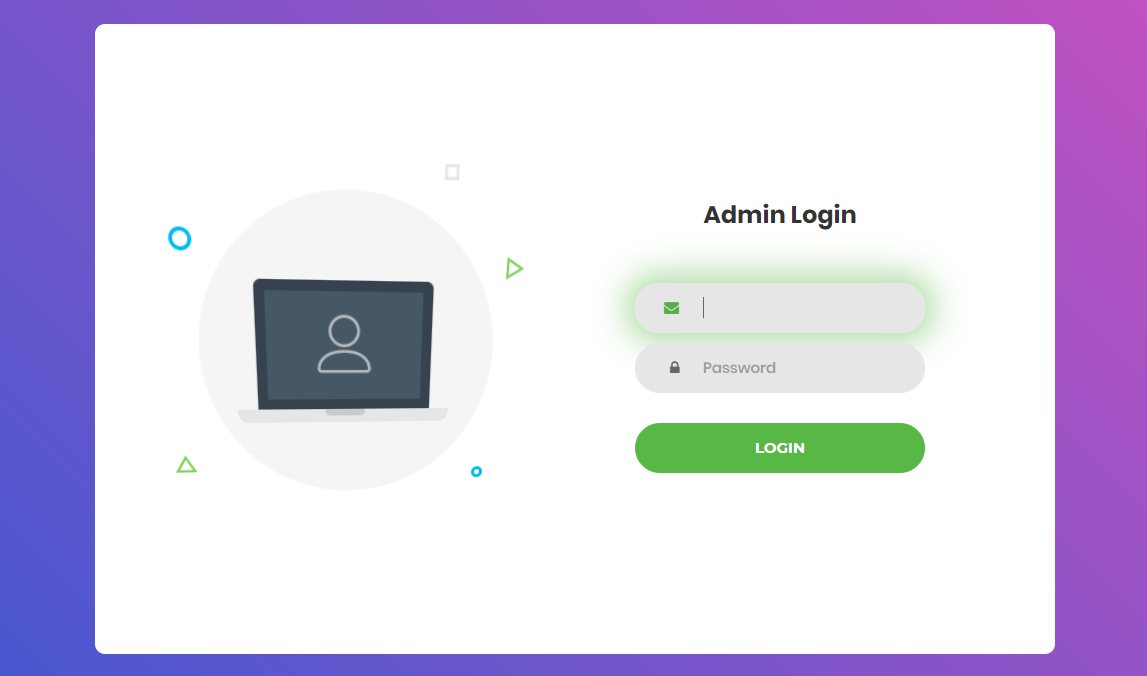
## 2.1 Current system interface

The following is the educational management system for FPT employees. The interface is divided into 3 parts with specific roles

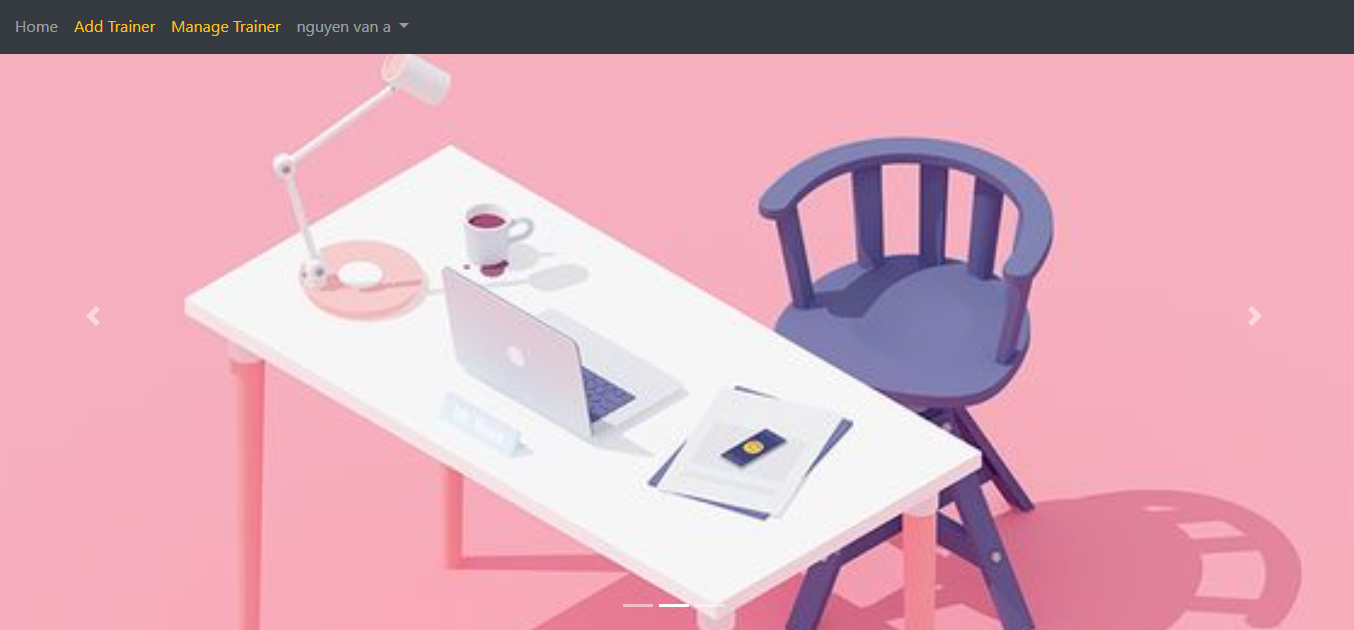


Admin:

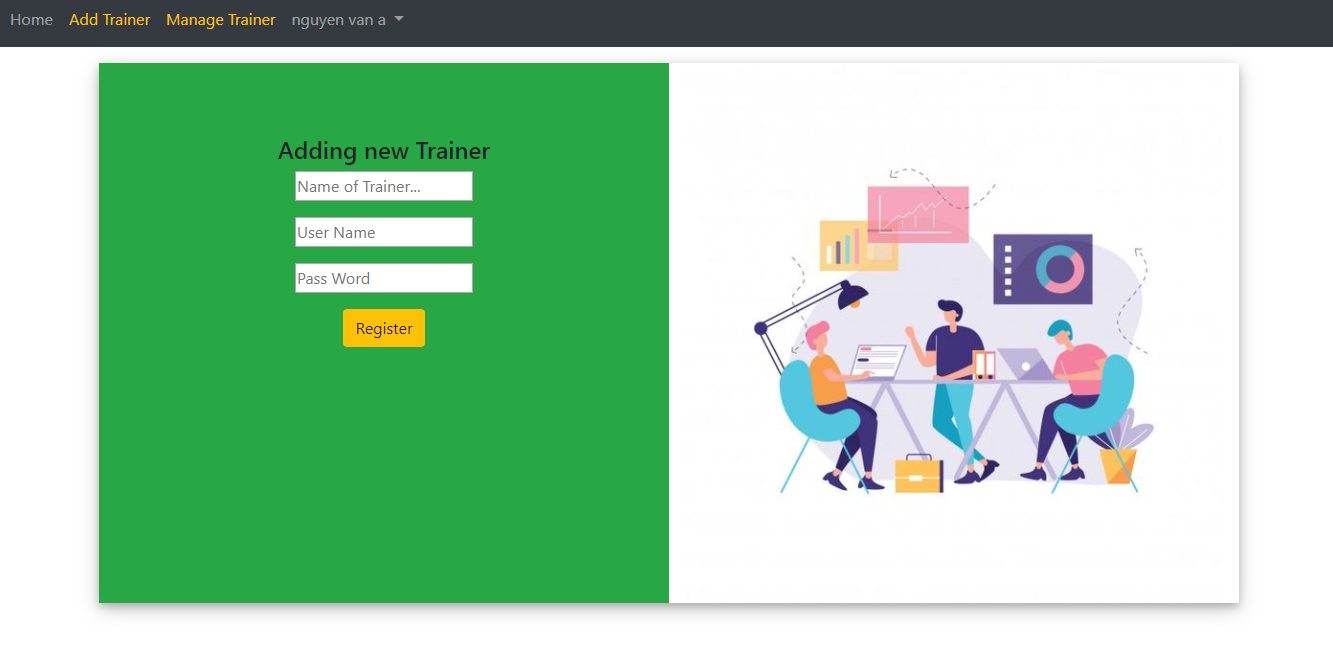
Login function:



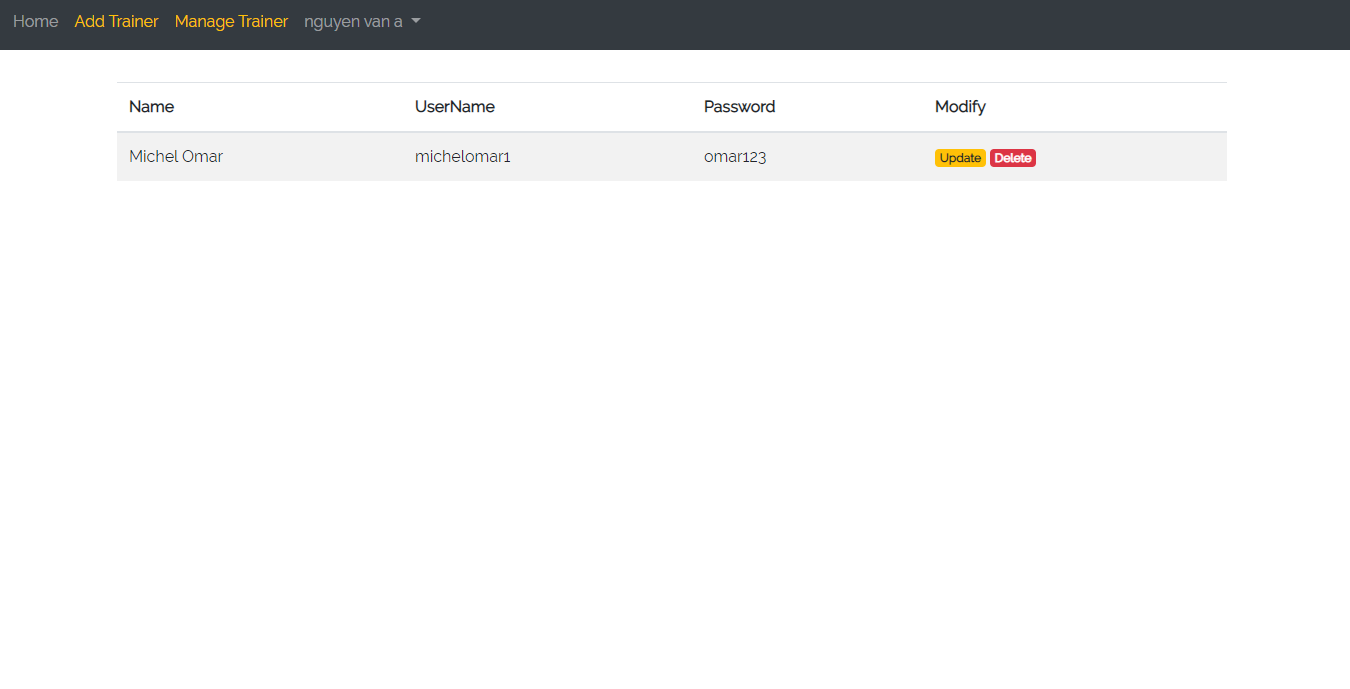
Home page of admin:

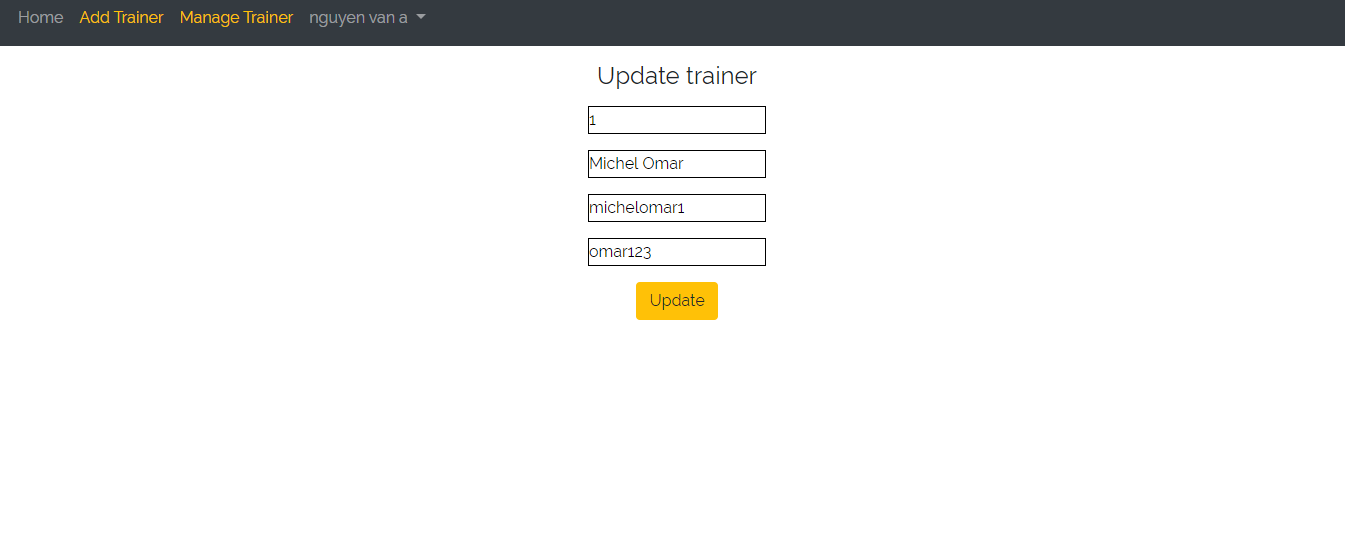


Adding new trainer account:



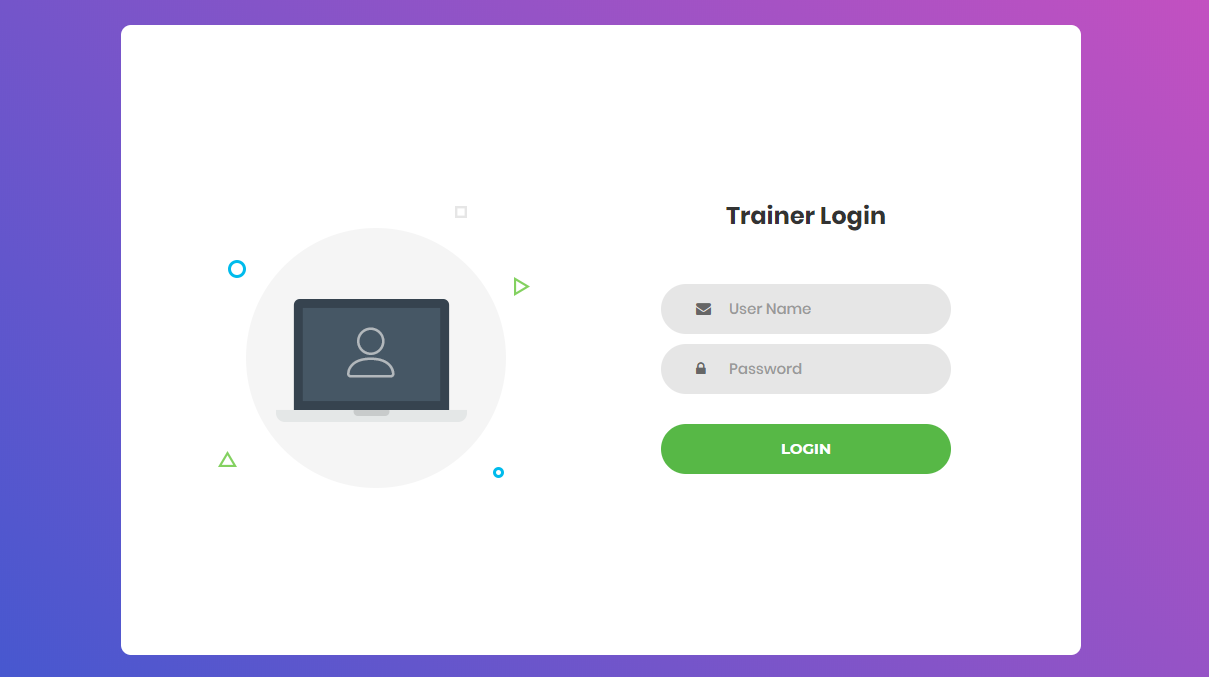
Management Trainer account:





Trainer:

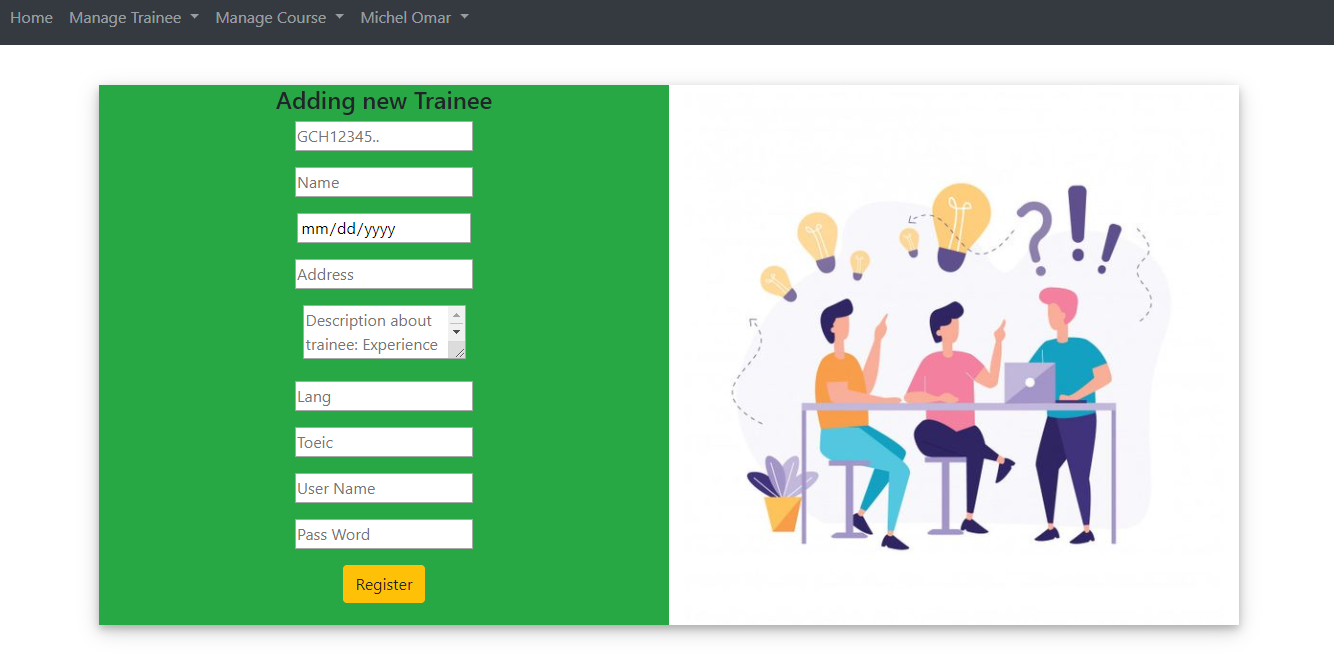
Login form of trainer:



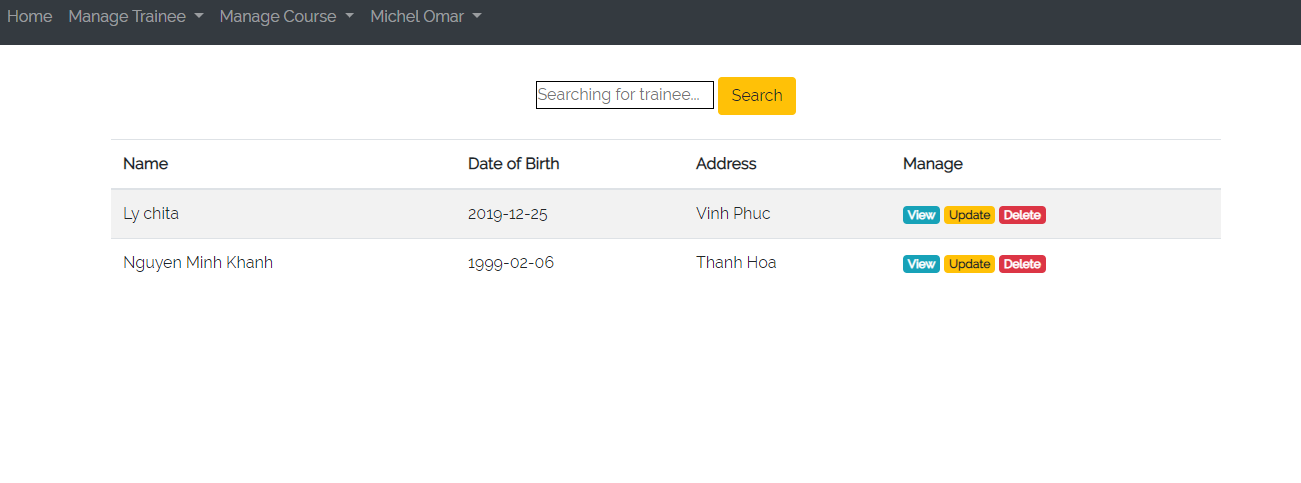
Home page of trainer:



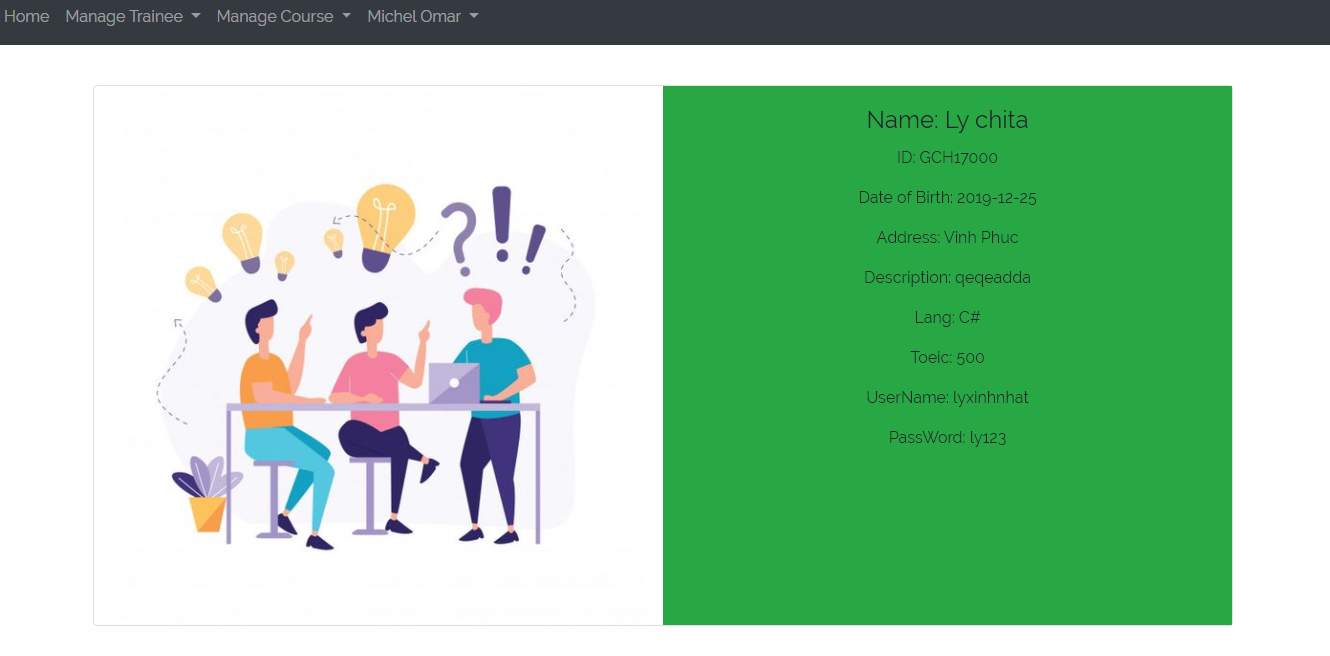
Adding new Trainee account page:



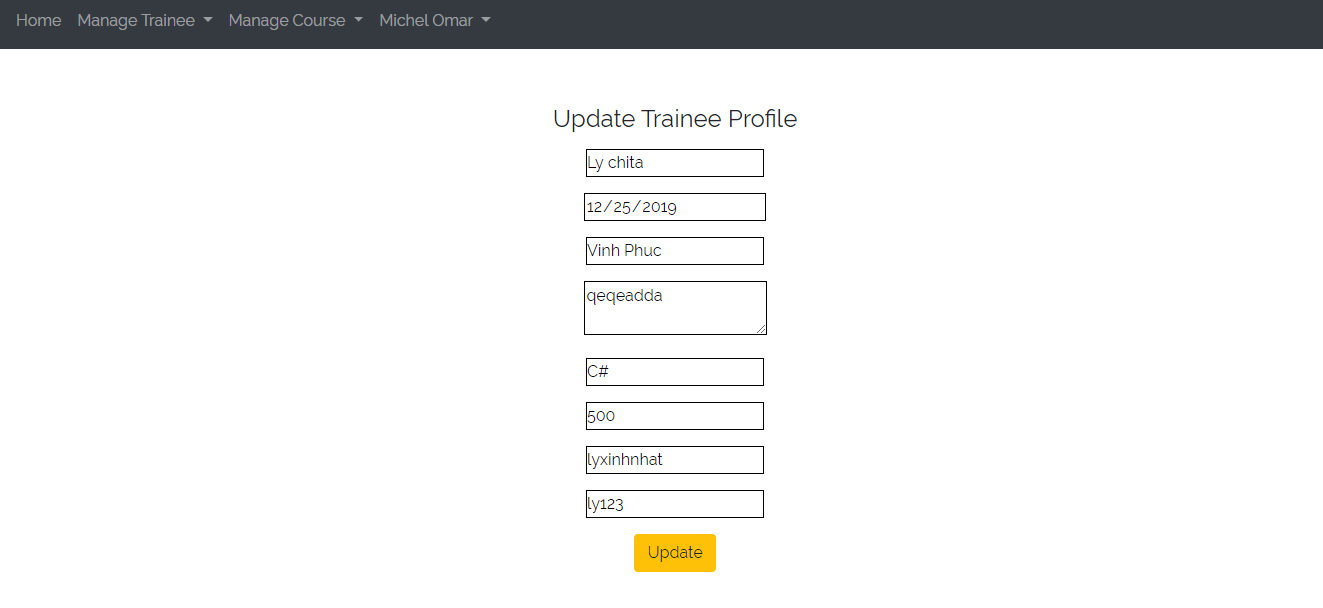
Management Trainee page:



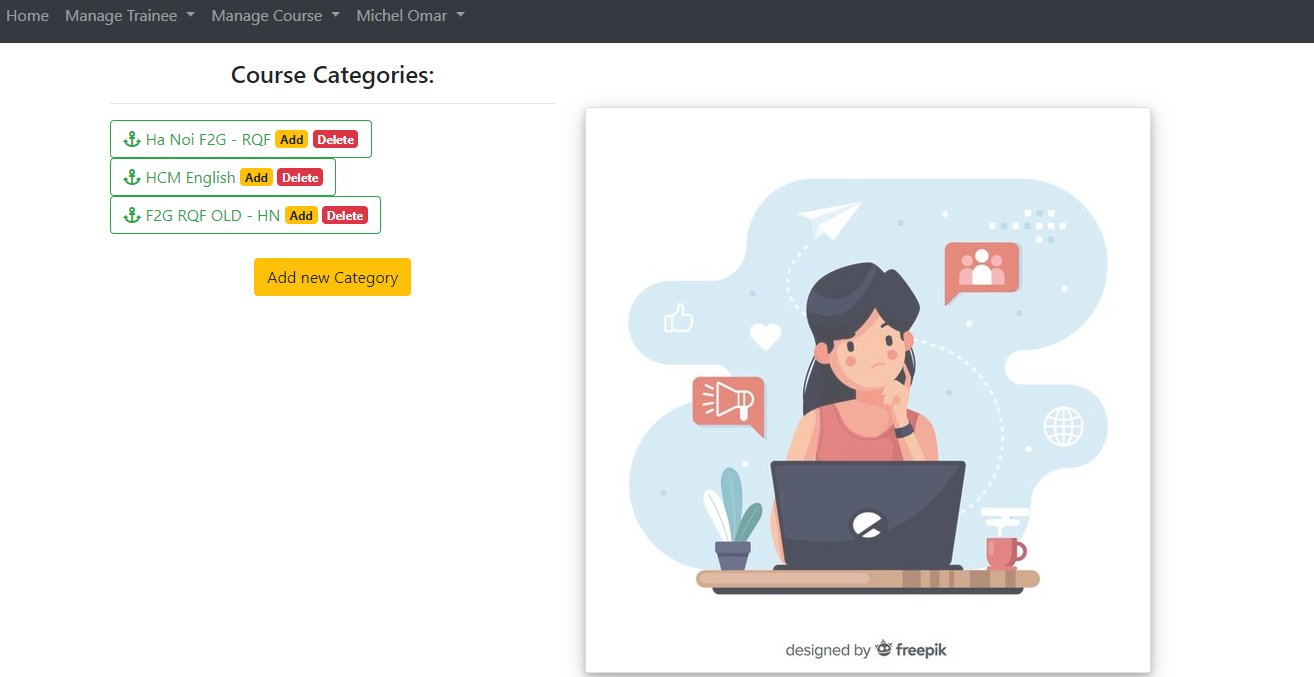
View detail trainee profile page:



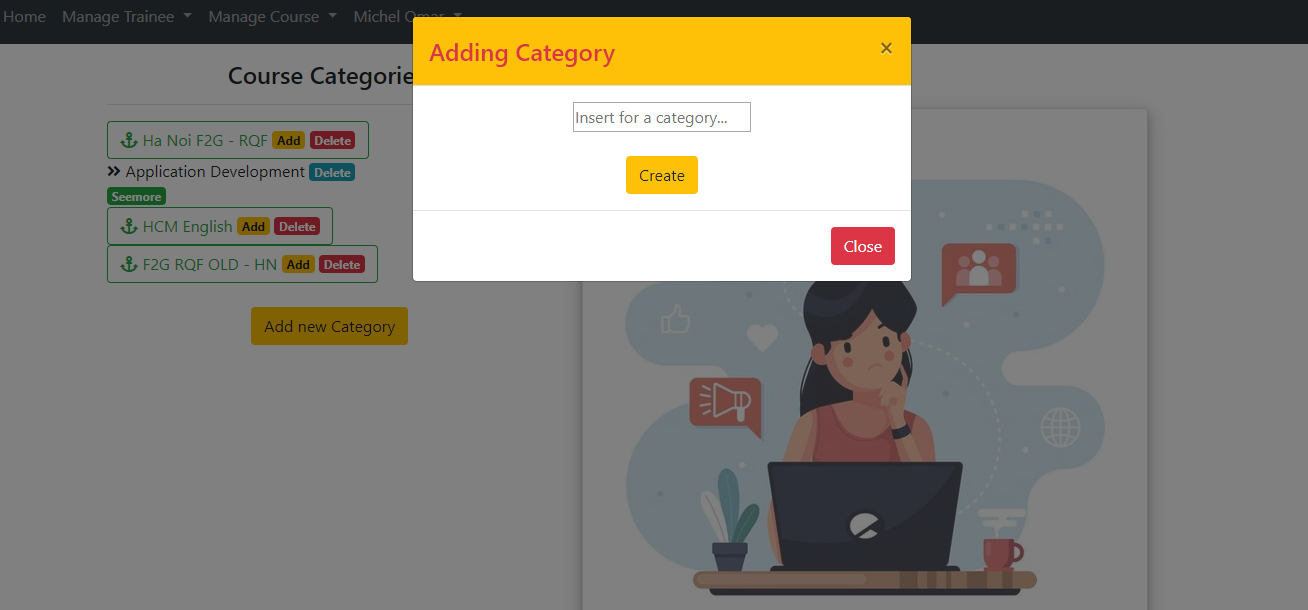
Update trainee profile page:



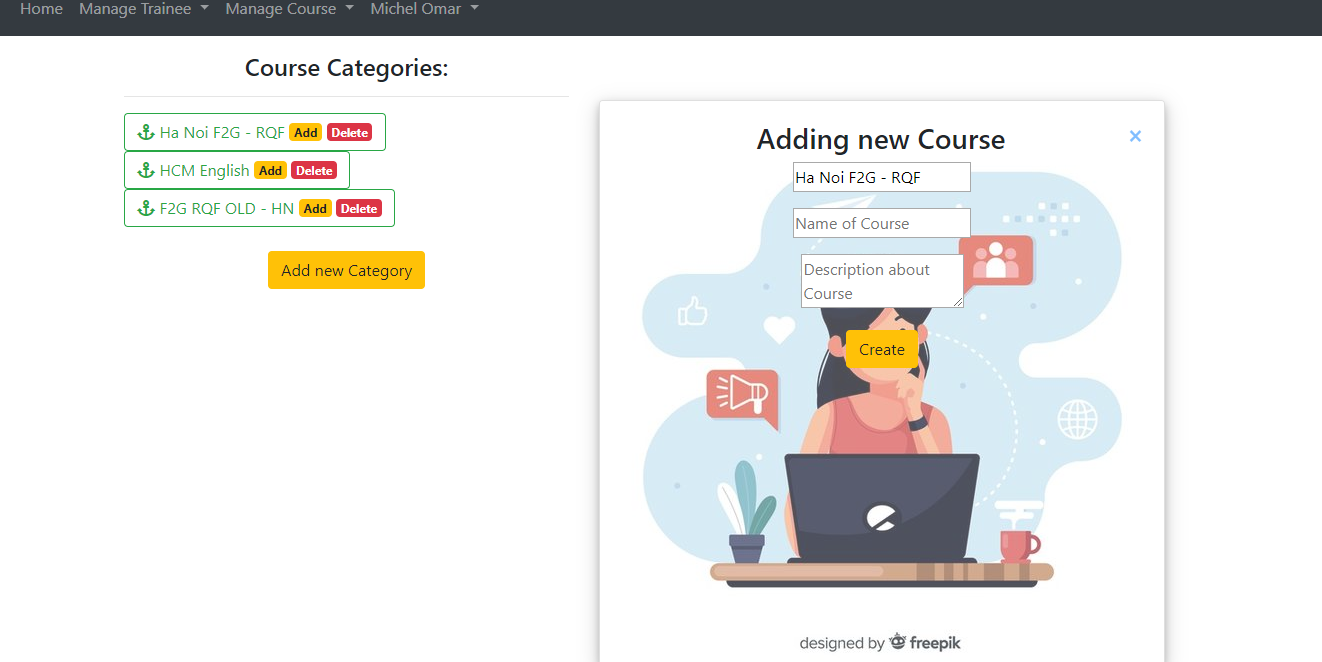
Management page for categories, courses:



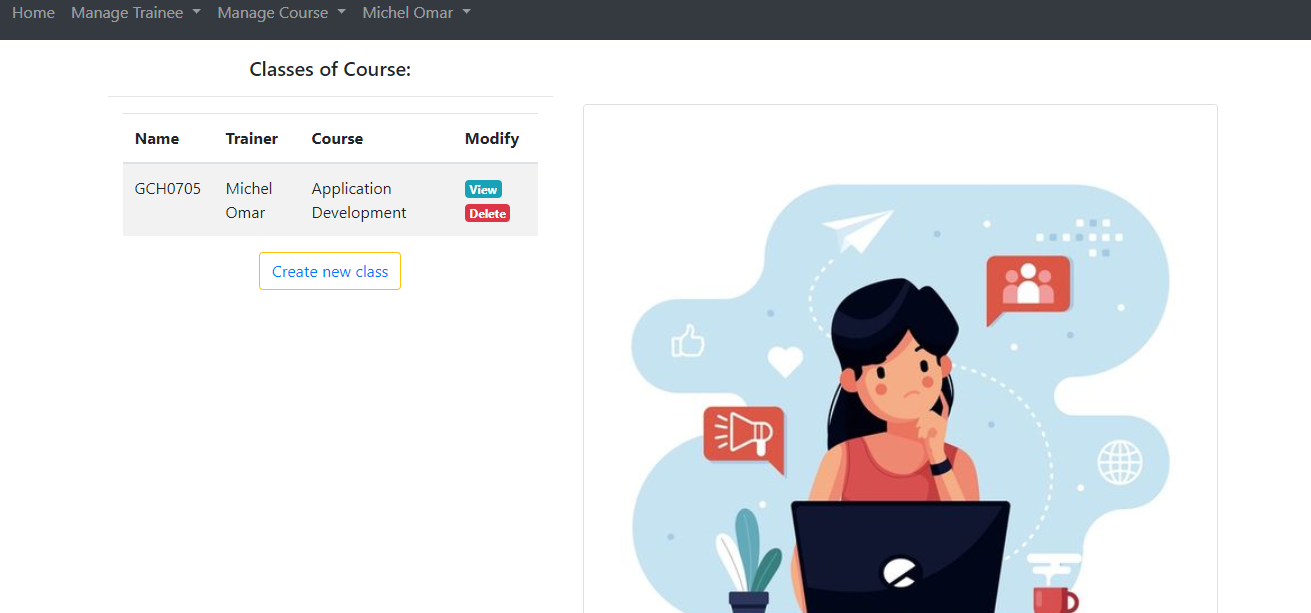
Adding new category:



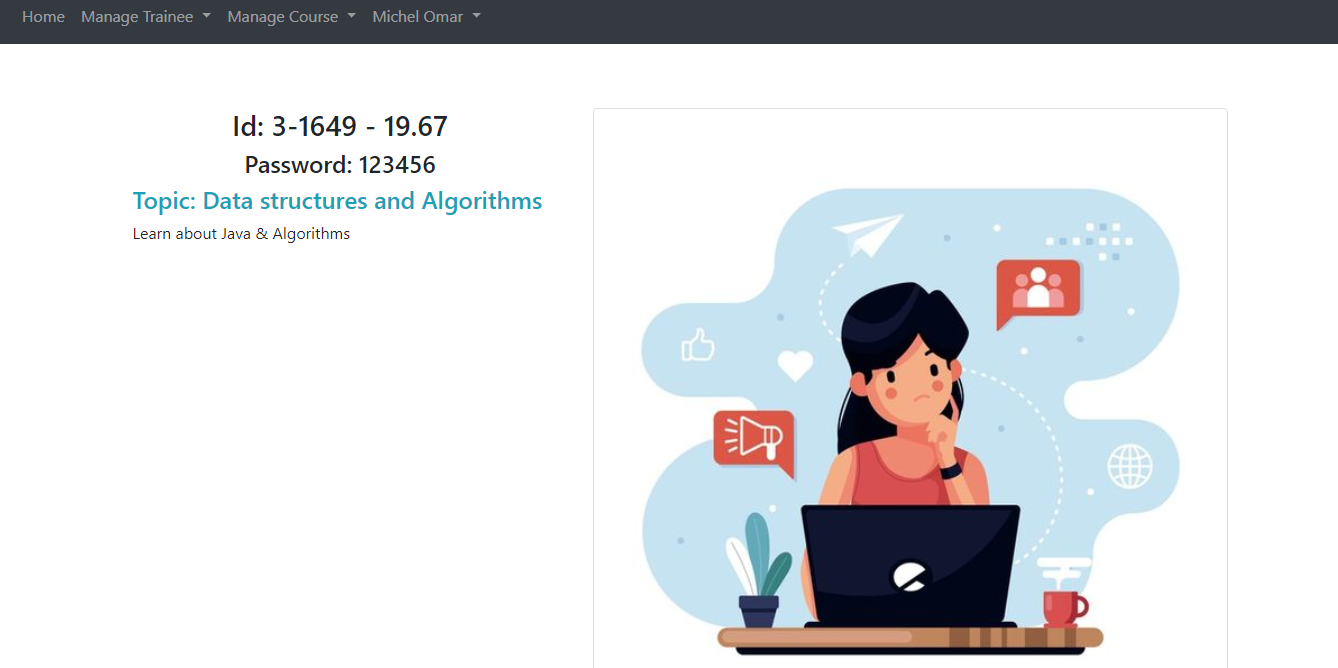
Adding new course:



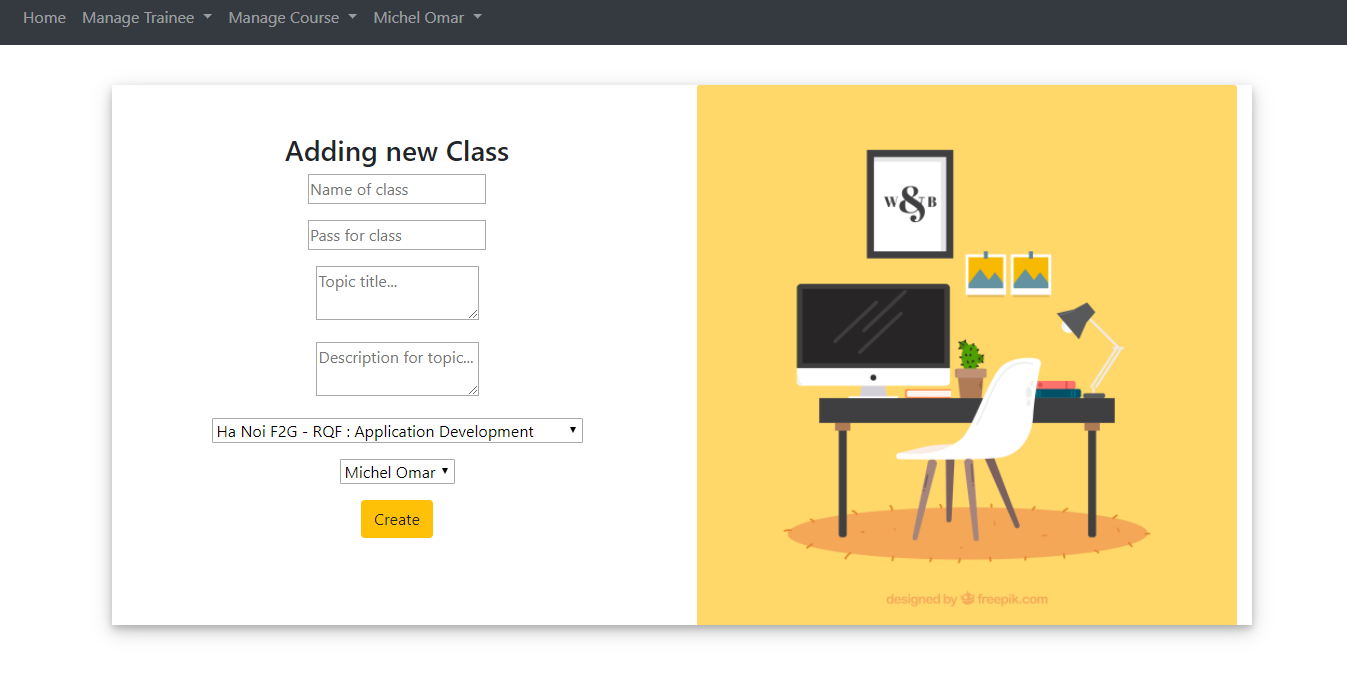
Management classes in course:



View class detail:

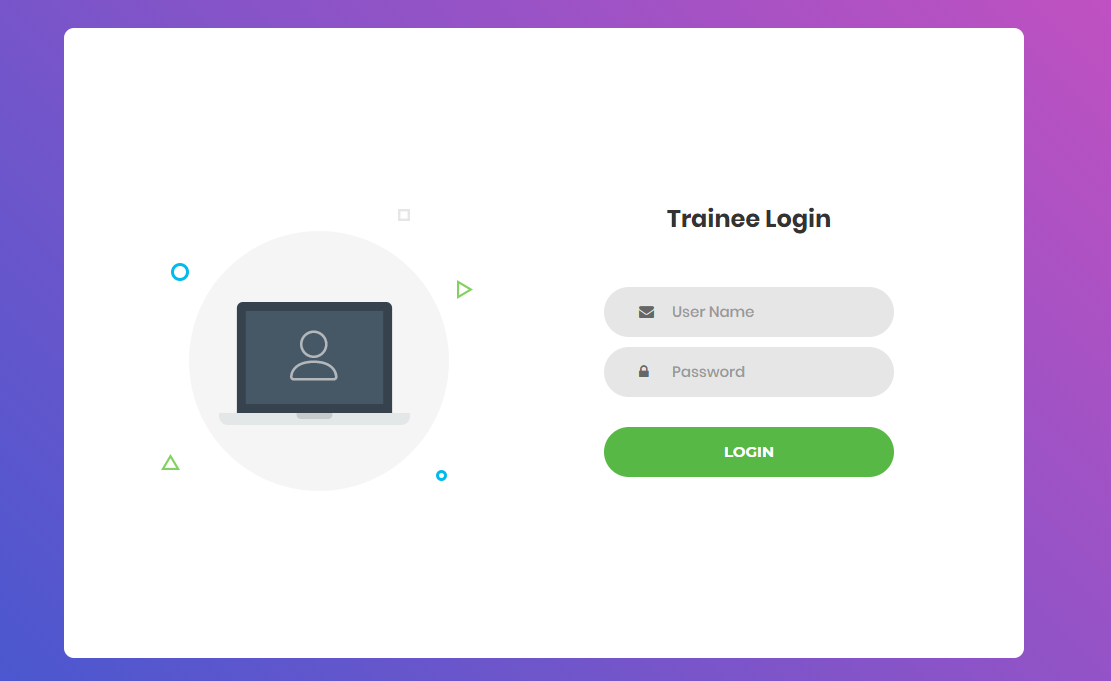


Adding new class:



Trainee:

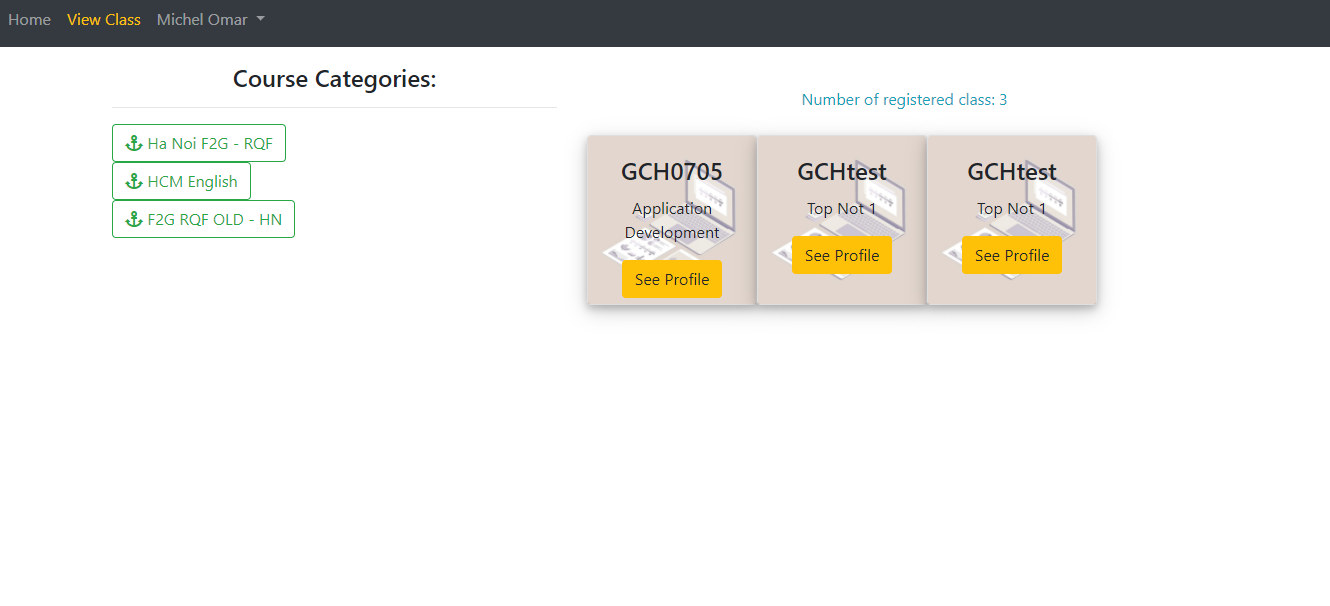
Login form of trainee:



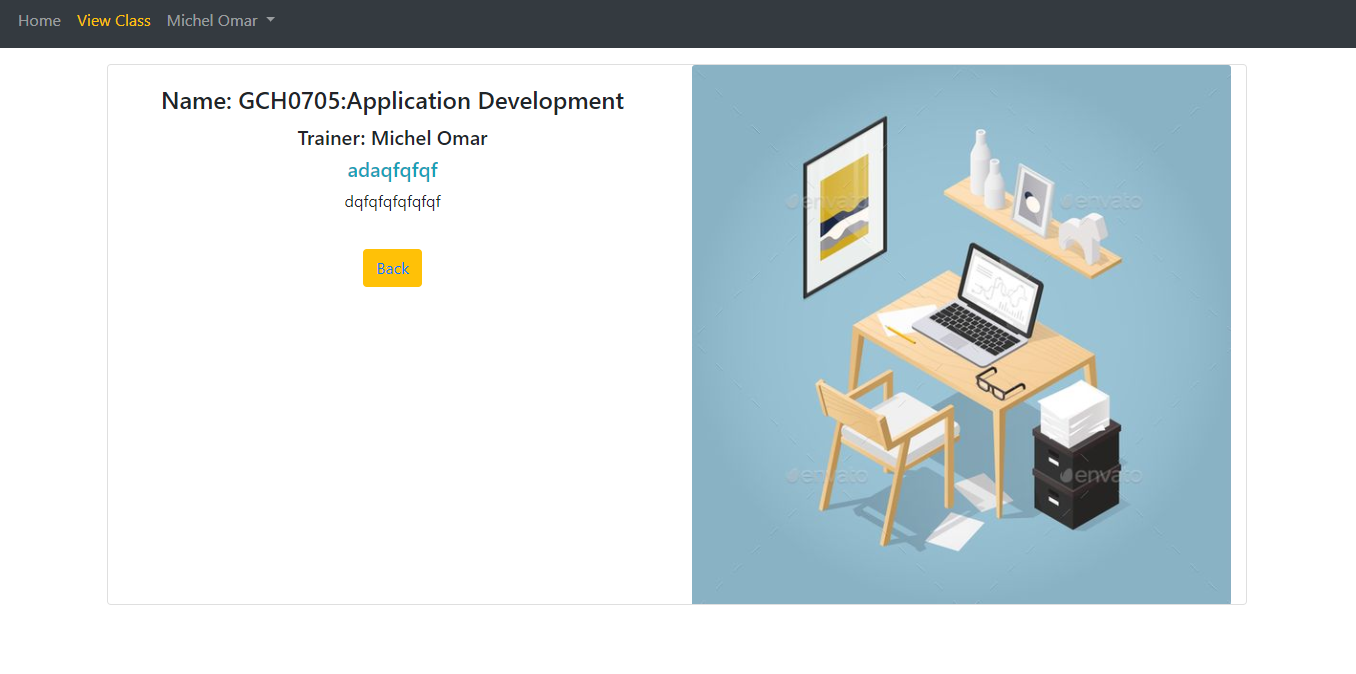
Home page of trainee:



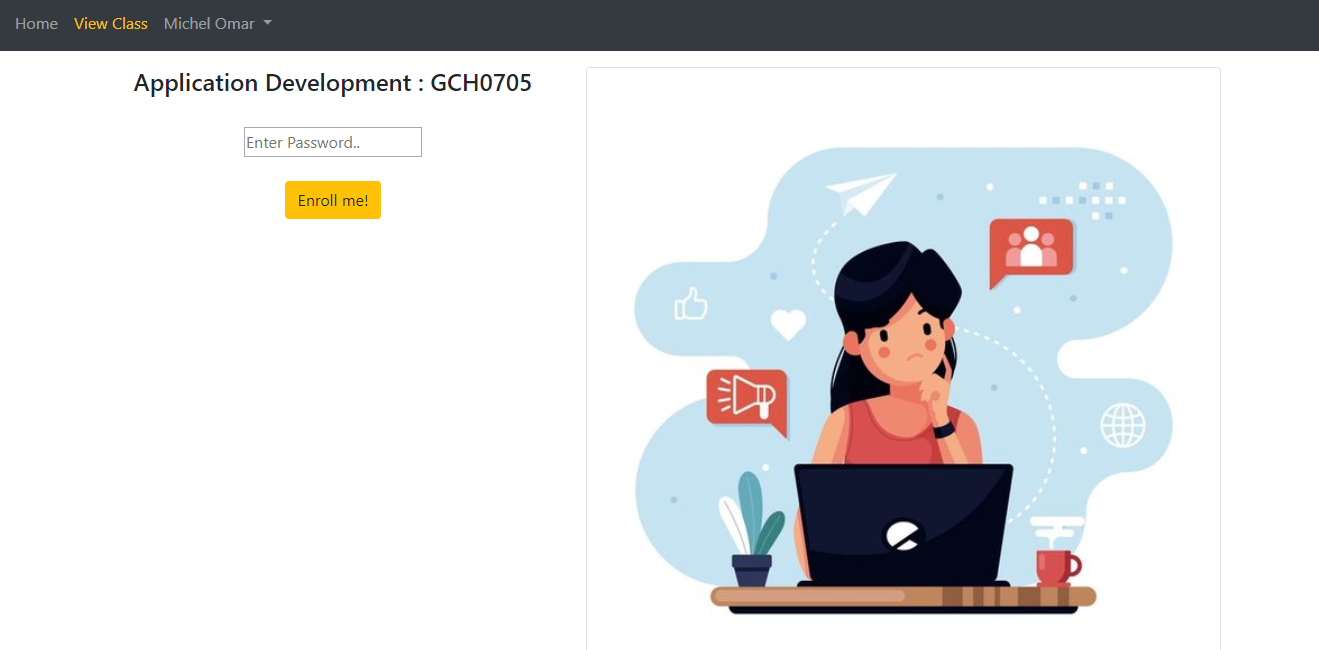
View categories, courses, classes of course and registered class:



View topic in class:



Register for a class:



## 2.2 Test case:

## 2.2.1 Admin function testing:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Function | What is being tested | How | Test data  used | Expected results | Time | Actual result | Action taken |
| Login function | Order of input  on data entry  screen | Enter the data following the instructions from the screen into the form with correct data | Data set:  Username: admin  Pass: admin123 | Logged in successfully | 12/31/2019 | OK | None |
| Login function | Validation of input | Performs a login without leaving any input values | Data set:  Leave the username blank | The system will issue a prompt to provide sufficient information | 12/31/2019 | OK | None |
| Login function | Sql injection | Enter values that generate sql errors | Data set:  Username:1' or '1' = '1  Password: 1' or '1' = '1 | Login unsuccessful | 12/31/2019 | OK | None |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

## 2.2.2 Trainer function testing:

## 2.2.3 Trainee function testing:

# III Application review

3.1 Comparison to initial requirements

After the development of the first version of the system has been completed, we will continue to sit back to check that the system's functions and non functional features meet the needs of FPT. The below directly compares the current system requirements and functions.

### 3.1.1 Functional features:

Admin:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Function | Initial requirement | On track | Conclusion about function of system | |
| 1 | Can login to the system through the first  page of the application | 100% | | We have designed a login interface exclusively for the admin. Here with username and password, admin can log into the system easily. In addition, the system has features that validate not allowed and prevent sql injection | |
| 2 | Can create and manage new user account for trainer (edit/delete) | 100% | | After making login. List of trainer accounts is displayed with delete or update options with information such as name, username, password. The function meets the requirement | |

Trainer:

|  |  |  |  |
| --- | --- | --- | --- |
| Function | Initial requirement | On Track | Conclusion about function of system |
| 1 | Can login to the system through the first  page of the application | 100% | We have designed a login interface exclusively for the Trainer. Here with username and password, trainer can log into the system easily. In addition, the system has features that validate not allowed and prevent sql injection |
| 2 | Can add new user account for trainee | 100% | After making login. Administrators can create a new trainee account with information such as trainee name, trainee accounts, age, date of birth, education, main programming language, TOEIC score, experience details, department, location, etc. on an easy web interface easy. Besides, the error values are also valid at the input. |
| 3 | Can save the information of trainee accounts in the database and from the list of trainees searchable by programming language keywords, TOEIC score etc | 80% | All information from the trainee account after being created by the trainer is saved in the database. When needed, it is easy to search by the name of the trainee. However, we still lack the invalid value validating features. |
| 4 | Can update, delete trainee accounts | 0% | Trainee cannot edit his own profile, he has to rely on the admin |
| 5 | Can manage categories such as searching, adding, updating and deleting course categories | 75% | The system provides trainers with features for managing adding and deleting categories. But there is no updating feature yet |
| 6 | Can manage courses such as searching, adding, updating and deleting courses | 75% | The system provides trainers with features for managing adding and deleting courses. But there is no updating feature yet |
| 7 | Can add topics such as topic name and topic descriptions into a course, add courses into a category | 100% | The trainer can create new topics and descriptions for a new class after performing the login session. |
| 8 | Can manage trainer profile such as adding, updating and deleting the information | 0% | The Trainer cannot directly edit his profile on our system but must rely on the admin. |
| 9 | Can assign trainer to a topic | 100% | Trainer can register teachers for specific classes on the system |
| 10 | Can assign trainee to a course | 0% | The system does not have this feature |

Trainee:

|  |  |  |  |
| --- | --- | --- | --- |
| Function | Initial requirement | On track | Conclusion about function of system |
| 1 | Can login to the system through the first  page of the application | 100% | We have designed a login interface exclusively for the Trainee. Here with username and password that resigntered by trainer, trainee can log into the system easily. In addition, the system has features that validate not allowed and prevent sql injection |
| 2 | Can view courses which have a topic he is assigned to | 75% | Trainee can view the class that you have registered on the system. There is no feature registered by the trainer |
| 3 | Can update own profile | 0% | The system does not have this feature |

### 3.1.2 Non-functional features:

In this section we will continue to compare the existing non-funtioncal features in the system with the initial requirements in terms of performance, security, UX / UI, etc:

|  |  |  |  |
| --- | --- | --- | --- |
| Non-function features | Initial requirement | On track | Conclusion |
| UX/UI | The system needs to provide a nice user friendly interface. | 100% | Our system provides a beautiful interface, selected colors are very bright, using trendy designs such as gadient, bootrap, jquery. Besides, the interface is easy to use and not tangled, users always take the control role with the nav which is located at the top of the website. Logical user script, and there's always a navigation button to go back or show what to do next. |
| Performance | Website response time is short | 75% | Using MVC model helps the website wait time for response from 0.2 to 0.5 seconds. However, the code is not really clean, increasing the highest speed |
| Security | Ensuring information security and safety features, preventing hackers from infiltrating and stealing information. | 50% | The security features available on the site are prevention of sql injection, html and javascript code. However, the biggest minus point is that it has not done data encryption before putting into the database, which poses great risks to customer privacy. |
| Capacity and expandability | The system can cater to about 1,000 people or more | 10% | Currently the system has not provided the domain name, so it is not possible to grasp the ability to store and expand on the cloud |

## 3.2 Assessment of the system’s strength and weakness:

After completing the work of developing the learning environment management system within FPT co. Now is the time for us to make an objective assessment of what the system has done and has not done (strengths and weaknesses) in the first version of the system to make improvements in the next versions.

Advantage:

* In the front-end, the system possesses a beautiful interface but it is very simple and easy to use for users. Using technologies like bootrap, jquery makes the interface more interesting.
* The user script is simple, there are always navigation buttons that make the user feel in control of everything. Short website processing times increase the experience.
* Regarding security, the system implements features to prevent html and sql injection to avoid the risk of hackers logging directly into the system.

Disadvantage:

* The system did not fully meet the requirements put forward. The simple design is not really attractive to users; sometimes complex designs are difficult to understand.
* In the back-end, using the php language to design is somewhat outdated compared to the current trend.
* In terms of security, there are no steps to encrypt data before saving to the database, as well as decryption when using data. This poses a huge risk when hackers can use the data directly in case of theft.
* The system still lacks some validation features for the input, which makes the data not clean.
* In terms of coding, the functions are not clean, the code is still quite long, making it difficult to reuse.
* The system has not provided backup data in case of emergency, backup features to restore information have not been provided.
* The scalability of the system is limited. Only meet for more than 1000 people.

## 3.3 Indentified risks

According to Cody, L (2016, online), The launch is the crucial first step in any website or web app’s journey. The truth is not so, your website or web application is successful and growing, then you need to constant update and improve your website before the risks may occur. Here are the risks to avoid and solutions:

|  |  |  |
| --- | --- | --- |
| Number | Risks | Solution |
| 1 | **Site Security**: every day thousands of websites are under attack, especially large or new ones. Personal and financial information are most at risk of being stolen. (Rean, J, 2015, online) | Hiring an expert to check your website’s source code for potential security flaws: the latest hacks, cracks, SQL injections, and XSS vulnerabilities. “If you are using a WordPress website, I highly recommend installing WordFence Security.” (Rean, J, 2015, online) |
| 2 | **Legal Pages:** Risks when users, developers and managers do not implement or violate legal policies (information security, etc.) (Rean, J, 2015, online) | Set up pages to commit legality regarding people who use your site, from copyrights, cookies and information about your company. If you are using WordPress, you can install a plugin called Terms of Service and Automated Privacy Policy. (Rean, J, 2015, online) |
| 3 | **Website Copy:** Risks from information or content on the website that has not been proofread before launching. Mistakes in the content or information provided, however small, will take away the reputation of the entire site. | Don't launch if you don't have untested content or for important pages like About. Also provides full information about copyright, version, author (Rean, J, 2015, online) |
| 4 | **JavaScript Independence:** Some browsers may have JavaScript disabled implicitly for any reason but do not indicate how to enable it. If you do not guarantee the independence of JavaScript, your website may look like it is broken. (Rean, J, 2015, online) | “Try turning off your browser’s JavaScript and browse around your website. If it looks broken, then you need to fix that.” (Rean, J, 2015, online) |
| 5 | **Backup:** Most web hosts are not good at keeping backups of websites. This may result in loss or inability to recover data in an emergency.(Rean, J, 2015, online) | Pay them to set up recurring jobs for backups depending on your needs and save backups on a separate location independent of your server. (Rean, J, 2015, online) |
| 6 | **Unexpected risk:** Unexpected risk: Sometimes a problem in your site will occur and people will visit pages that do not exist. Sometimes, they even visit pages they shouldn't. (Rean, J, 2015, online) | “Check for all possible security risks on your site and block public access to those pages, and of course, when you block people on specific pages, you need to let them know why.” (Rean, J, 2015, online) |

# Conclusion

As such, our group is under the help of lecturers and the spirit of cooperation and cooperation of our team members, our team has completed the process of developing the educational environment system within FPT co. We have really come a long way with a lot of new things in managing a project for the first time. We start with making a plan for the right tools and methods, then applying that plan to develop and manage the entire project throughout its life cycle. Finally, as testers we learn how to test and evaluate system features and offer improved solutions. Because this is the first time, we have been exposed to a large project, so it is inevitable that mistakes. If given the opportunity, we hope that there will be conditions to develop the next version of the system.

# References:

Cody, L. (2016) *4 Keys to Effectively Managing Your Website After Launch*. Avaiable from: https://modeeffect.com/4-keys-managing-website-after-launch/

[Access 25/02/2019]

Rean, J. (2015) *15 Things to Check Before Launching a New Website*. Avaiable from: https://onextrapixel.com/15-things-to-check-before-launching-a-new-website/

[Access 25/02/2019]

|  |  |  |
| --- | --- | --- |
| **❒ Summative Feedback: ❒ Resubmission Feedback:** | | |
| **Grade:** | **Assessor Signature:** | **Date:** |
| **Lecturer Signature:** | | |