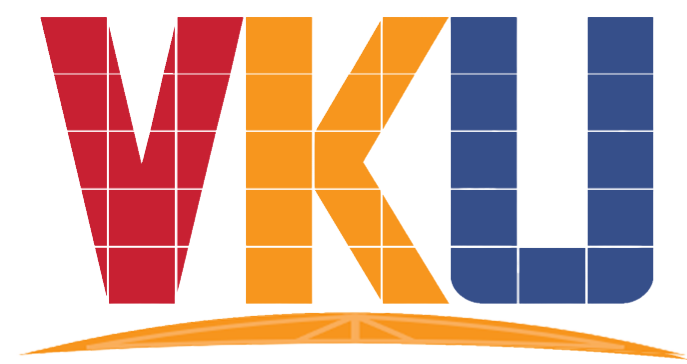
**VIETNAM – KOREA UNIVERSITY OF INFORMATION**

**AND COMMUNICATION TECHNOLOGY**

**FACULTY OF COMPUTER SCIENCE**



**GRADUATION PROJECT**

**BUILDING ONLINE COURSES SYSTEM**

Students : **ĐẶNG XUÂN LONG – 18IT079**

Class : **18IT2**

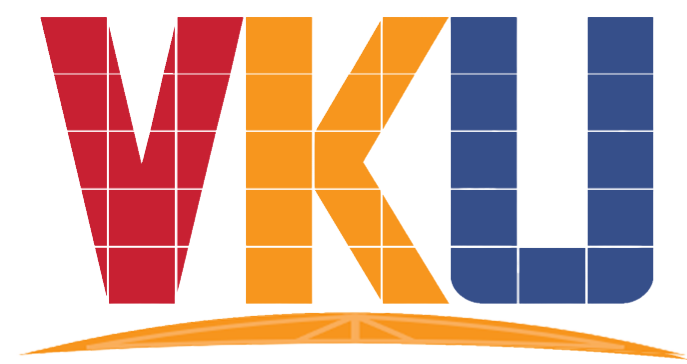
Instructor : **Dr. NGUYỄN ĐỨC HIỂN**

***Da Nang, December of 2022***

**VIETNAM – KOREA UNIVERSITY OF INFORMATION**

**AND COMMUNICATION TECHNOLOGY**

**FACULTY OF COMPUTER SCIENCE**



**GRADUATION PROJECT**

**BUILDING ONLINE COURSES SYSTEM**

Students : **ĐẶNG XUÂN LONG – 18IT079**

Class : **18IT2**

Instructor : **Dr. NGUYỄN ĐỨC HIỂN**

***Da Nang, December of 2022***

# PREAMBLE

We'd like to express our gratitude to the professors during the time of doing Graduation Project. We have received a lot of help, suggestions, and enthusiastic guidance from teachers, family, and friends. We would like to express my heartfelt gratitude to Mr. Nguyễn Đức Hiển, the instructor of Graduation Project - the Vietnam-Korea University of Information and Communication Technology, who guided us through the entire project-making process.

We would also like to thank the teachers in the Vietnam-Korea University of Information and Communication Technology who taught us the knowledge and skills and other technologies, giving us a solid foundation and helping us throughout the learning process to be able to complete this Graduation Project in the best way.

Because of the limited time, the level of knowledge of close friends is still limited. So in the project inevitably shortcomings, We look forward to receiving comments from all teachers as well as friends to make our project more complete.

We sincerely thank you!

# ACKNOWLEDGMENTS

We'd like to express our gratitude to the professors during the time of doing Graduation Project. We have received a lot of help, suggestions, and enthusiastic guidance from teachers, family, and friends. We would like to express my heartfelt gratitude to Mr. Nguyễn Đức Hiển, the instructor of Graduation Project - the Vietnam-Korea University of Information and Communication Technology, who guided us through the entire project-making process.

We would also like to thank the teachers in the Vietnam-Korea University of Information and Communication Technology who taught us the knowledge and skills and other technologies, giving us a solid foundation and helping us throughout the learning process to be able to complete this Graduation Project in the best way.

Because of the limited time, the level of knowledge of close friends is still limited. So in the project inevitably shortcomings, We look forward to receiving comments from all teachers as well as friends to make our project more complete.

We sincerely thank you!

|  |  |
| --- | --- |
|  | *Da Nang, December 30, 2022*  *Students* |
|  |  |

# COMMENT

(Of Instructor)

**………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………**

**………………………………………………………………………………………**

|  |  |
| --- | --- |
|  | ***Da Nang, December of 2022***  **Instructor**  **Dr.Nguyen Duc Hien** |

**TABLE OF CONTENTS**

[PREAMBLE 2](#_Toc126765459)

[ACKNOWLEDGMENTS 3](#_Toc126765460)

[COMMENT 4](#_Toc126765461)

[LIST OF ACRONYMS 8](#_Toc126765462)

[CHAPTER 1: OVERVIEW 9](#_Toc126765463)

[1.1 Title of the topic: 9](#_Toc126765464)

[1.2 Introduction: 9](#_Toc126765465)

[1.3 Target: 9](#_Toc126765466)

[1.4 Subjects and scope of research: 10](#_Toc126765467)

[1.4.1 Research subjects: 10](#_Toc126765468)

[1.4.2 Research scope: 10](#_Toc126765469)

[1.4.3 Research method: 10](#_Toc126765470)

[1.5 Content and implementation plan: 10](#_Toc126765471)

[1.6 Project structure: 11](#_Toc126765472)

[CHAPTER 2: THEORETICAL BASIS 12](#_Toc126765473)

[2.1 Recommend System 12](#_Toc126765474)

[2.1.1 Introduction to Recommend System: 12](#_Toc126765475)

[2.1.2 Basic components of a recommender system: 13](#_Toc126765476)

[2.2 Laravel: 14](#_Toc126765477)

[2.2.1 Introduction to Laravel: 14](#_Toc126765478)

[2.2.2 Why use Laravel: 15](#_Toc126765479)

[2.3 StarUML: 16](#_Toc126765480)

[2.4 Visual Studio Code: 16](#_Toc126765481)

[2.5 Figma: 17](#_Toc126765482)

[2.6 Postman: 17](#_Toc126765483)

[2.7 XAMPP: 18](#_Toc126765484)

[2.8 MySQL: 18](#_Toc126765485)

[2.8.1 Introduction to MySQL: 18](#_Toc126765486)

[2.8.2 Why use MySQL: 19](#_Toc126765487)

[2.9 PHP: 19](#_Toc126765488)

[2.10 Javascript: 20](#_Toc126765489)

[2.11 CSS: 21](#_Toc126765490)

[2.12 React: 21](#_Toc126765491)

[2.12.1 Introduction to React: 21](#_Toc126765492)

[2.12.2 Why use React: 22](#_Toc126765493)

[2.13 Tailwind CSS: 22](#_Toc126765494)

[CHAPTER 3: SYSTEM DESIGN ANALYSIS 24](#_Toc126765495)

[3.1 Requirements 24](#_Toc126765496)

[3.2 Use Case Diagram: 24](#_Toc126765497)

[3.3 Class Diagram: 26](#_Toc126765498)

[3.4 Activity Diagram: 28](#_Toc126765499)

[3.5 Database: 32](#_Toc126765500)

[CHAPTER 4: IMPLEMENTATION AND BUILDING 33](#_Toc126765501)

[4.1 Build courses recommend system: 33](#_Toc126765502)

[4.1.1 Analysis and data processing: 33](#_Toc126765503)

[4.1.2 Generating course recommendation: 34](#_Toc126765504)

[4.2 Build the website: 35](#_Toc126765505)

[4.3.1 Login: 35](#_Toc126765506)

[4.2.2 Home page: 35](#_Toc126765507)

[4.2.3 When entering a course: 37](#_Toc126765508)

[4.2.4 Create assignment (lecturer): 39](#_Toc126765509)

[4.2.5 Edit assignment, see submission (lecturer): 39](#_Toc126765510)

[4.2.6 Submission of assignments (students): 40](#_Toc126765511)

[4.2.7 Creating documents (instructor): 40](#_Toc126765512)

[4.2.8 Editing documents (instructor): 41](#_Toc126765513)

[4.2.9 All Courses: 41](#_Toc126765514)

[4.2.10 The teacher request to open the class: 42](#_Toc126765515)

[4.2.11 Students join/cancel classes: 42](#_Toc126765516)

[4.2.12 Forums: 43](#_Toc126765517)

[4.2.13 Comments: 44](#_Toc126765518)

[4.2.14 Student Management: 44](#_Toc126765519)

[4.2.15 Instructor management: 45](#_Toc126765520)

[4.2.16 Course Management: 45](#_Toc126765521)

[4.2.17 Browse the course: 46](#_Toc126765522)

[4.2.18 Edit personal information, avatar: 47](#_Toc126765523)

[CHAPTER 5: CONCLUSIONS AND DEVELOPMENT ORIENTATIONS 48](#_Toc126765524)

[5.1 Conclusion: 48](#_Toc126765525)

[5.1.1 Achievements: 48](#_Toc126765526)

[5.1.2 Limitations: 48](#_Toc126765527)

[5.2 Development: 48](#_Toc126765528)

[LIST OF REFERENCES 50](#_Toc126765529)

# LIST OF ACRONYMS

|  |  |  |
| --- | --- | --- |
| **STT** | **Phrase** | **Abbreviation** |
| 1 | Representational State Transfer | REST |
| 2 | Application Programming Interface | API |
| 3 | User Interface | UI |
| 4 | Model-View-ViewModel | MVVM |
| 5 | Hypertext Preprocessor | PHP |
| 6 | Model-View-Controller | MVC |
| 7 | Unified Modeling Language | UML |
| 8 | Model Driven Architecture | MDA |
| 9 | File Transfer Protocol | FTP |
| 10 | Virtual Private Server | VPS |
| 11 | Structured Query Language | SQL |
| 12 | Integrated Development Environment | IDE |
| 13 | Cascading Style Sheets | CSS |
| 14 | Hypertext Markup Language | HTML |

# CHAPTER 1: OVERVIEW

## Title of the topic:

Topic: "Online Courses System".

## Introduction:

Education systems have drastically changed during 2022 as it was imperative to keep students safe. The traditional method has been losing its relevance faster than ever. The meaning of this new-age education is getting broader by the day, and people are starting to understand its significance. It is engaging, flexible.

This type of learning has radically transformed how education works in various countries. Indeed, online education parallels the rise in technology and the internet.

Ever since the pandemic hit, we have entered an era that revolutionized education. Here are some of the unique benefits of utilizing online learning platforms:

* Self-paced learning opportunity with optimal flexibility
* Better, enhanced, and structured time management
* Demonstrated increase in self-motivation
* Wider accessibility that is global and broader than traditional learning
* Cost-effective

From those advantages, we design an online learning management to serve educational institutions. The application is an online learning environment integrated with tools to help make teaching more interesting, contributing to improving the quality of education.

## Target:

Building an online learning management, helping to quickly and effectively distribute documents to students, thereby helping lecturers can support the self-study process of students. Apply knowledge learned to implement topics such as: database, web programming, system analysis and design. Improve teamwork skills, ability to self-study and self-study new technologies.

## Subjects and scope of research:

### Research subjects:

This basic project aims to study the following subjects:

* Research and understand learning management applications.
* Research on Laravel to build the backend for a website.
* Research on Java to build an Android application.
* Research on REST API to support communication between Server and Client.
* Research on technology, operation mechanism and application MySQL database management system.
* Research and apply techniques, library support design application interface in accordance with modern design trends in the market.

### Research scope:

The project is researched and built on Java, Laravel 9 Framework and React JS framework. The project is researched and tested on a small scale.

### Research method:

The project uses practice-oriented research methods to develop and improve the application:

* Analyze, refer to the learning management.
* Collecting information, learning and researching necessary techniques, technologies and libraries will support the construction and development of the topic.
* Self-assess and compare the results achieved with the goals set out initially. Adjust, overcome the shortcomings and promote the advantages.
* Complete the application and put it into practice, conduct surveys, collect comments and reviews from users.

## Content and implementation plan:

* Content:
* System analysis and design (system design according to MVVM model)
* Learn and use frameworks: Laravel, React JS.
* Building an application’s UI and web’s UI.
* Build Backend.
* Building Database.

## Project structure:

In this report, we would like to present the knowledge and theoretical basis that the group has researched and applied. As well as the analysis and evaluation of the team in the process of working, building and developing the training management website. This project report is presented with the following structure:

* Chapter 1: An overview of the research topic, research purpose, research scope and research methods.
* Chapter 2: Introduction of necessary techniques, technologies and libraries that have been researched, studied and applied to the construction and development of the topic.
* Chapter 3: Set out the requirements specification for system design analysis, architectural design and necessary models for the whole system.
* Chapter 4: Presenting the implementation of the designs built from Chapter 3, implementing and introducing the website interface, illustrating how it works.
* Chapter 5: Summarize the project, state the meaning of the project, present the positive aspects that have been set as well as recognize and evaluate the remaining limitations. From there, give a solution, the next development direction for the application.

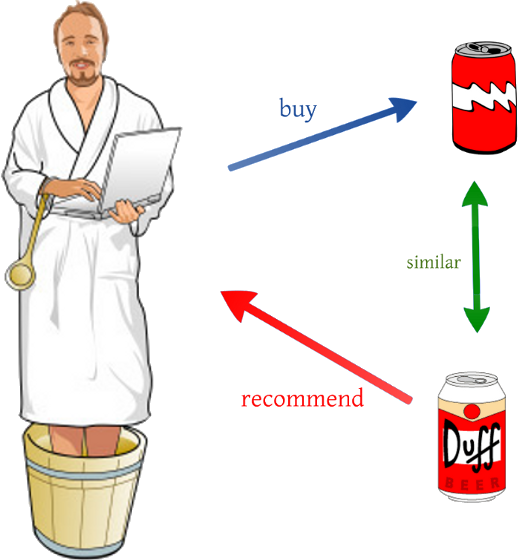
# 

# CHAPTER 2: THEORETICAL BASIS

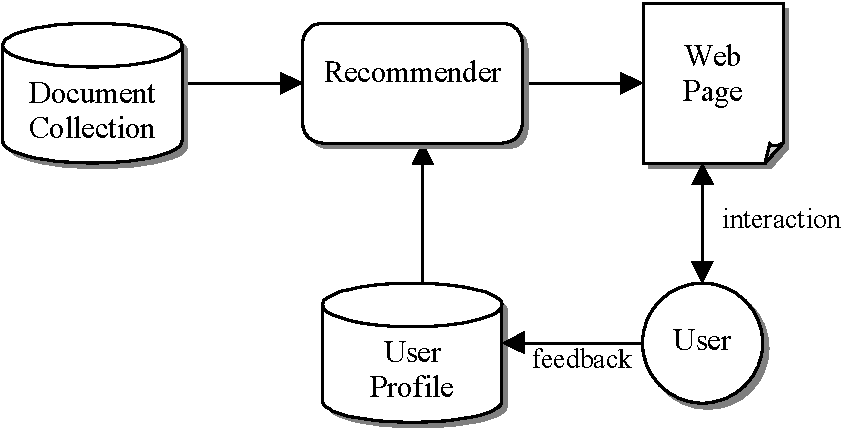
## Recommend System

### Introduction to Recommend System:

In essence we can think of a recommendation system like a matchmaker. It predicts our interests and finds potential objects that match those interests to suggest us. Assuming you are a funny single guy, you like pretty and active girls. But imagine, human strength is limited and you absolutely cannot know all the faces and faces of the beauties in the world. At this point, you really need a recommendation system like a matchmaker for you. A little easier to understand, isn't it, now we will return to an example applied in a field quite close to us IT people, which is e-commerce.



Overview of the system:



A content based recommender systems will include the following components:

* User set (users)
* File items (items)
* Data for each item

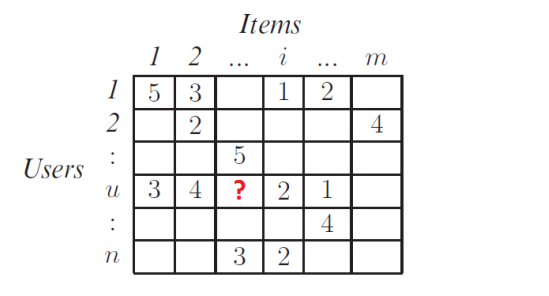
### Basic components of a recommender system:

To work or build a new information system, we need to figure out what we will need to create them. Talking about a recommendation system that is approached by Machine Learning method, we need to consider the following three basic characteristics:

* First: The first thing that needs to be taken care of is the user, of course, if there is no user, we know who to suggest in the system.
* Second: We need to pay attention to the items (items) these items can be products on sales pages, songs on music sites, another user like on social networks or an article like on our Viblo for example. Why should we care about the item because if there is no item, what do we suggest to the user.
* Third: We need to care about each user's feedback on that item. It could be a rating score, it could be an indicator of the user's interest in that item…. Simply because we have to quantify these quantities in order to have a recommendation base for users

Representing information using a users-items matrix:

* After we have gathered the above information about the system in some way we need to represent that information in a form that can be computed. A great idea is to use a matrix, a matrix is created that shows the likes of each user on the respective items represented as follows:



In the matrix, there are weighted cells that will show each user's favorite level on the items. On the other hand, there are also empty cells representing the user who has never had access to the item. It is this that shows the role of a recommender system, that is, based on the information known in the past of the user, the recommender system will suggest to that user information that the user does not know. That is, predict the values in the empty cells in the above matrix and then arrange them in descending order of preference to suggest to the user.

Recommender systems will generally be divided into two categories:

* Content based recommender systems
* Collaborative filtering recommender systems
* Hybrid method recommender systems

## Laravel:

### Introduction to Laravel:

Laravel is a free and open source PHP Framework, built to support the development of software and applications according to the MVC architecture. Currently, Laravel is the most popular and best PHP Framework.



***Figure 2.2 Laravel Introduction***

### Why use Laravel:

* Access to the latest features that PHP has to offer. This advantage is especially useful for users of Namespaces, Interfaces, Overloading, Anonymous functions and Shorter array syntax.
* Refer to diverse resources of different documents. Each release version of Laravel comes with appropriate documentation for users' convenience.
* Integrate with local or cloud-based mail services thanks to clean API on Swift Mailer library
* Fast processing speed, meeting the needs of creating websites or large projects in a short time
* Ease of use thanks to the easy-to-understand 3-layer MVC design pattern.
* High security
* Supports cache backends like Memcached and Redis out-of-the-box
* Integrated tool for command line – Artisan
* Create links to named routes
* Dramatically reduce product development cycles as Laravel helps to integrate faster with the support of the Laracasts community.

## StarUML:

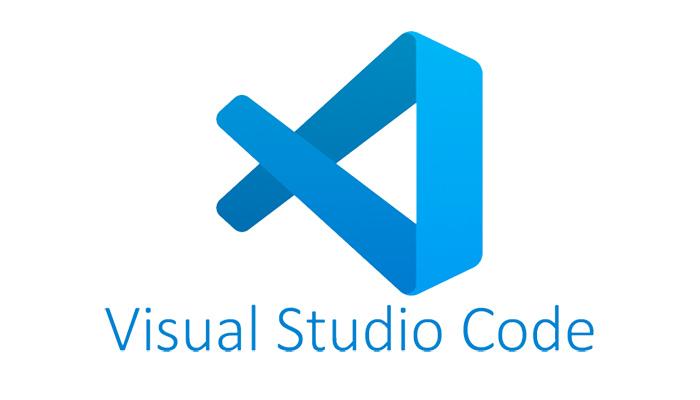
StarUML is a platform model, which is software that supports UML. It supports MDA approaches by supporting UML profile concepts. In addition, using StarUML will ensure maximum productivity and quality of software projects.



***Figure 2.3 StarUML Introduction***

## Visual Studio Code:

Visual Studio Code is a lightweight but powerful open source editor for Windows, Linux and macOS developed by Microsoft. It has support for JavaScript, Node.js, and TypeScript, as well as providing a rich extensibility ecosystem for many other programming languages. VS Code is a combination of a source code editor with powerful development tools like Git, Debug or Syntax Highlighter.



***Figure 2.4 VSCode Introduction***

## Figma:

Figma is a tool launched in 2016, with its friendly interface and ease of use, Figma has quickly emerged and become a popular user interface design tool in the global technology community. Some big brands using Figma up to now can be mentioned as Microsoft, Twitter, GitHub, Dropbox… Unlike previous design tools, Figma is designed on the cloud platform. This is a tool with similar features to Sketch, but it supports better teamwork.



***Figure 2.5 Figma Introduction***

## Postman:

Postman is a type of tool that allows users to manipulate APIs, the most popular of which is REST. Thanks to Postman programmers can call Rest API without having to write any lines of code. Postman has the ability to support all HTTP methods including: POST, PUT, DELETE, PATCH, GET,... In addition, Postman also allows programmers to save the history of requests, so it is extremely convenient for users. reuse needs.



***Figure 2.6 Postman Introduction***

## XAMPP:

XAMPP is web server software under the GNU General Public License, distributed and developed by Apache Friends. This is a program to create a Web server (Web Server) with built-in Apache, PHP, MySQL, FTP Server, Mail Server and tools like phpMyAdmin. XAMPP is a software that allows you to simulate a server hosting environment. Thanks to this hypothetical server hosting, you can run a demo of a website right on your computer without the need to buy hosting or VPS.



***Figure 2.7 XAMPP Introduction***

## MySQL:

### Introduction to MySQL:

* MySQL is the world's most popular free and open source database management system and is very popular with developers in the application development process. Because MySQL is a high-speed, stable and easy-to-use database management system that is portable, works on many operating systems and provides a large system of very powerful utility functions. With high speed and security, MySQL is well suited for applications that access databases on the internet. Users can download MySQL for free from the homepage. MySQL has many versions for different operating systems: Win32 version for operating systems Windows, Linux, Mac OS X, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS,..
* MySQL is one of the very basic examples of a Relational Database Management System using Structured Query Language (SQL).
* MySQL is used to support Node.js, PHP, Perl, and many other languages, as a place to store information on web pages written in NodeJS, PHP or Perl,...



***Figure 2.8 MySQL Introduction***

### Why use MySQL:

* High speed, stable
* High level of information security
* Ease of use and portability: despite its high functionality, MySQL is actually a very simple and minimally complex database system.
* Can work on many operating systems with a large number of utility functions: MySQL is well suited for applications that access databases on the Internet.
* As an open source, MySQL uses the GNU General Public License so it's completely free.

## PHP:

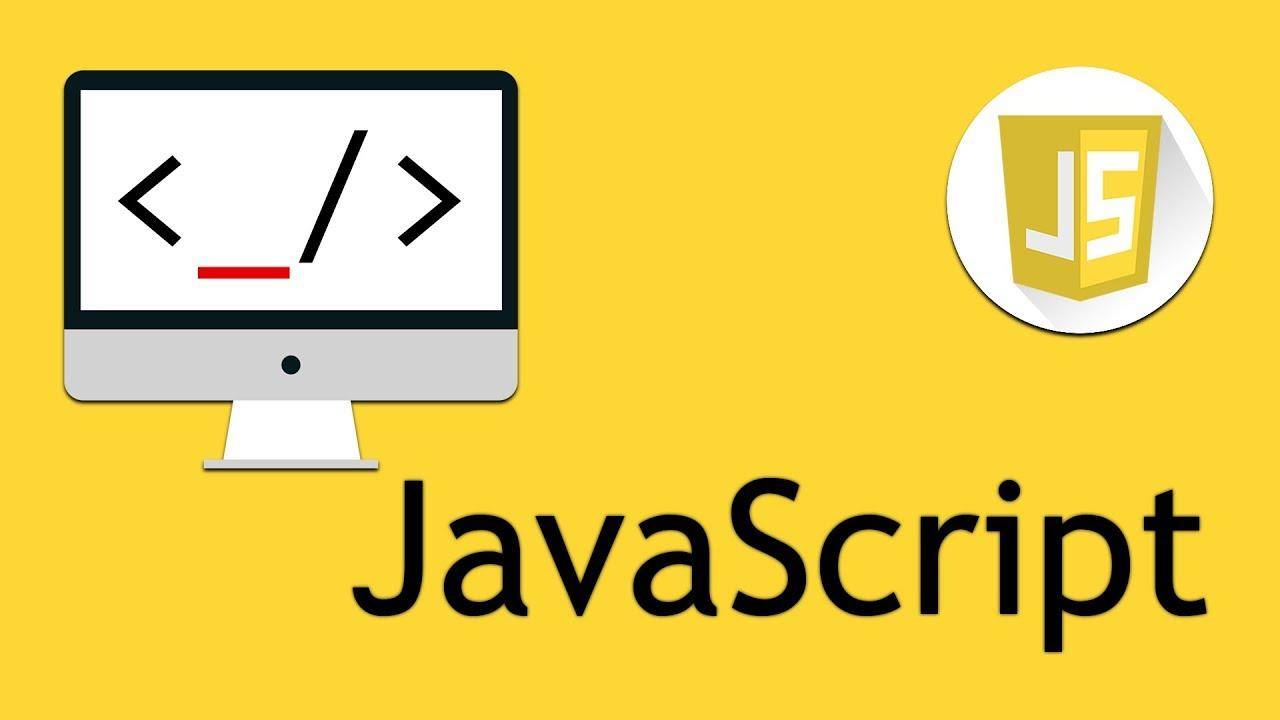
PHP is an abbreviation for the term Personal Home Page. This is a scripting language string or a form of code. In particular, the PHP language is mainly developed for server-based applications. When PHP programmers write programs, strings of commands are run on the server to generate HTML code. As a result, applications on websites can run easily.



***Figure 2.11 PHP Introduction***

## Javascript:

JavaScript is a self-developed or self-defined object-based scripting language. JavaScript is widely used in Website applications. JavaScript is supported on almost all browsers such as Firefox, Chrome, ... on computers and phones. JavaScript's job is to process HTML objects in the browser. It can interfere with actions like adding/removing/editing CSS properties and HTML tags easily. In other words, JavaScript is a client-side browser-based programming language. However, now with the appearance of NodeJS, it has made it possible for JavaScript to work in the backend HTML stands for Hypertext Markup Language which is a programming language used to build and restructure the components contained in the Website. People often use HTML in dividing paragraphs, headings, links, blockquotes, etc. A Website usually contains many subpages and each of these subpages has its own HTML file. Note, HTML is not a programming language. This means that it cannot perform “dynamic” functions. In a simpler way, similar to Microsoft Word software, HTML only works to layout and format web pages.



***Figure 2.12 JavaScript Introduction***

## CSS:

CSS stands for Cascading Style Sheets, a simple design language that handles part of the look and feel of a web page. CSS describes how HTML elements appear on screens and other media. Using CSS, you can control text color, font size, font style, spacing between paragraphs, the size of web page elements, background color, layout design, and how web pages appear. screens of different sizes as well as a variety of other effects.

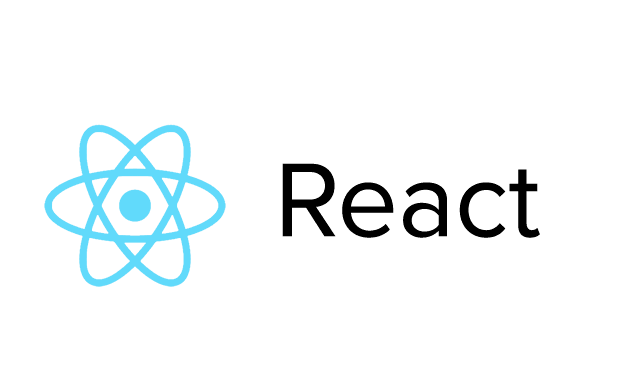


***Figure 2.13 CSS Introduction***

## React:

### Introduction to React:

React is the most popular JavaScript library for building user interfaces (UIs). It gives excellent response speed to user input using a new method of rendering web pages. Components of this tool are developed by Facebook. It was launched as an open source JavaScript engine in 2013. It is now ahead of major competitors like Angular and Bootstrap, the two best-selling JavaScript libraries of the time.



***Figure 2.14 Introduction React***

### Why use React:

* Easy to use.
* React allows to reuse components that have been developed into other applications with the same functionality.
* React components are easier to write because it uses JSX, which extends the optional syntax to JavaScript that allows HTML to be combined with JavaScript.
* Better performance with Virtual DOM.
* React allows creating user interfaces that can be accessed on different search engines. This feature is a huge advantage as not all JavaScript frameworks are SEO friendly.

## Tailwind CSS:

Tailwind is a utility-first CSS framework that helps you quickly build user interfaces. Actually it is similar to other libraries like Bootstrap, Materialize... but its main difference is that the low-level CSS framework provides us with a lot of customization functionality for component development without any problems. no need to rewrite the CSS.



***Figure 2.15 Tailwind CSS Introduction***

# CHAPTER 3: SYSTEM DESIGN ANALYSIS

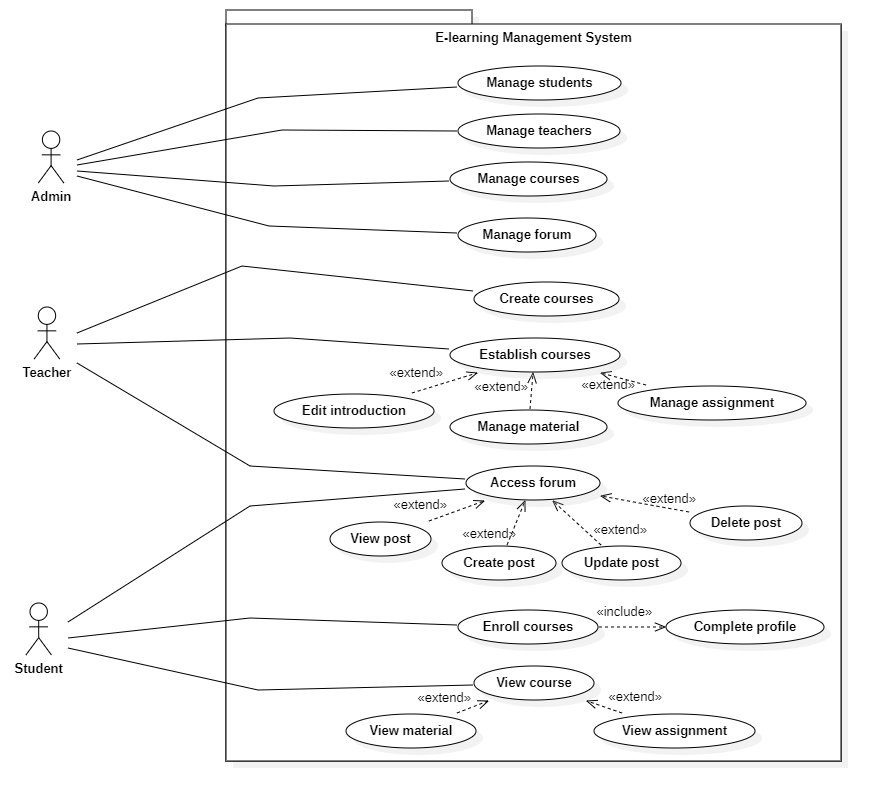
## Requirements

Based on the analysis above, we have determined the necessary requirements for our courses system.

* About recommendation:
* The recommendation system should suggest courses based on the course the user is viewing
* About user interface and functions
  + Basic functions of the online course system (view, search, register for a course, ...)
  + Post on the forum, comment on other posts
  + Create course lists and deadline lists
  + Platforms can work stably and have similar functions

## Use Case Diagram:

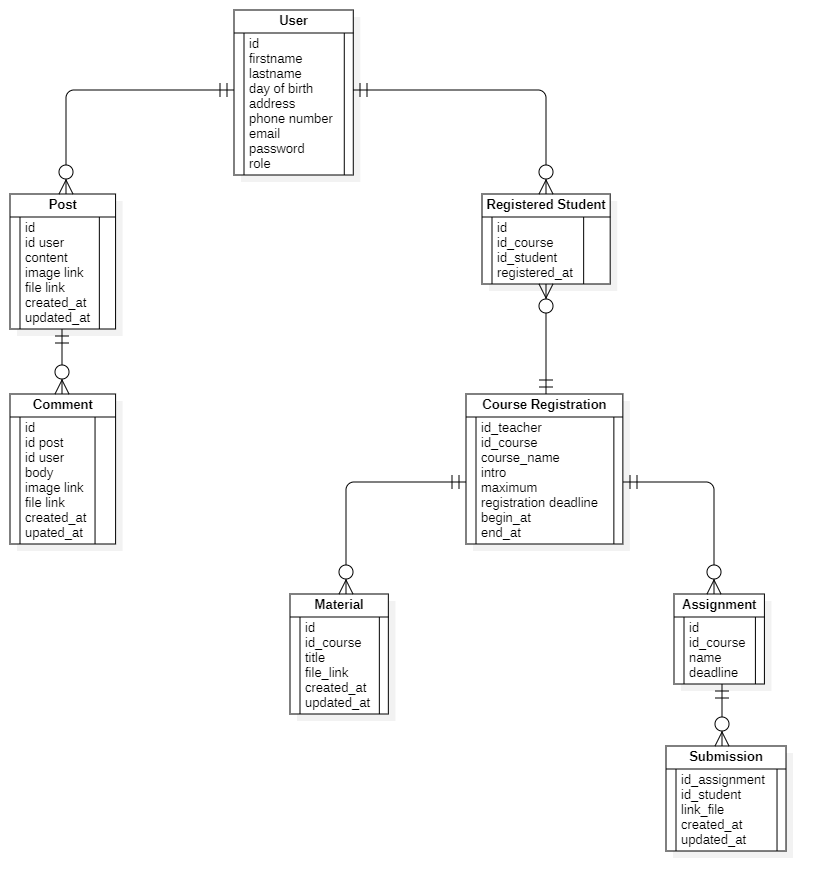
* Actor and functions:
  + Admin:
    - Manage students
    - Manage teachers
    - Manage courses
    - Manage forum
  + Teacher:
    - Create courses
    - Establish courses
    - Edit introduction
    - Manage material
    - Manage assignment
    - Access forum
* View post
* Create post
* Delete post
* Update post
  + Student:
    - Enroll courses
    - View courses
    - View material
    - View assignment
    - Access forum
* View post
* Create post
* Delete post
* Update post



***Figure 3.1 UseCase Diagram***

## Class Diagram:

After completing the use case diagram, we proceed to build a class diagram, the classes and fields of each class are represented through the following diagram:



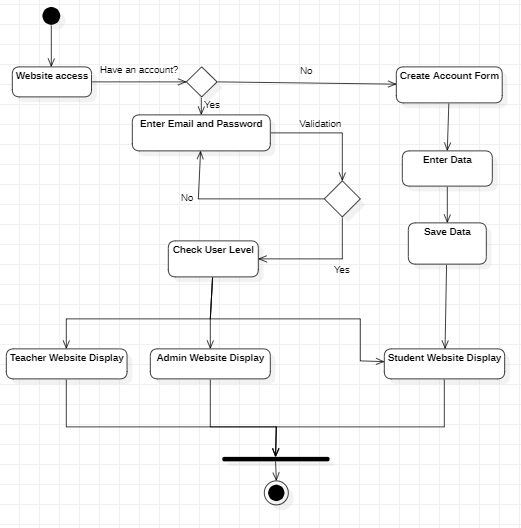
***Figure 3.2 Class Diagram***

Some of main class:

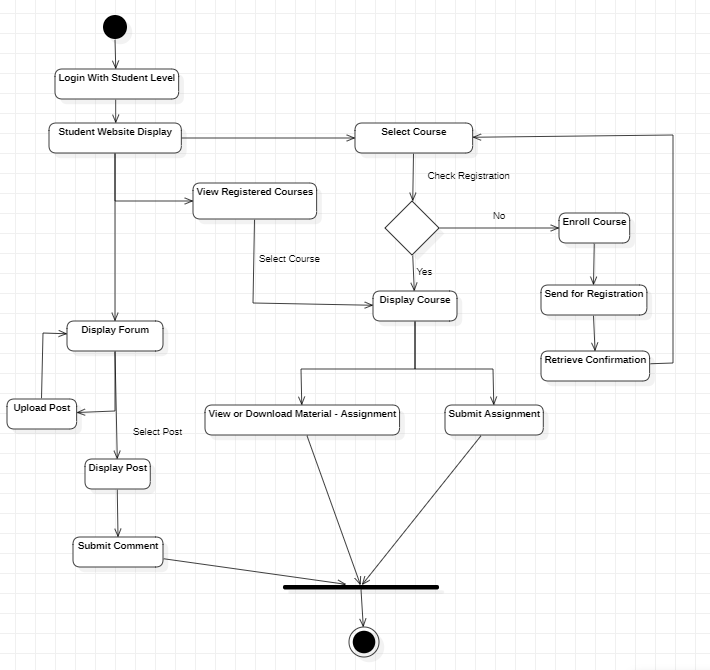
* User:
  + getUserProfile(): Get the information of logged in user
  + editProfile(): Update user profile
  + login(): Login with email and password
  + logout():
* Courses:
  + getCourses (): Return all courses
  + searchCourse(): Search course
  + createCourse(): Create a new course
  + getCourseInfo(): Return selected course infomation
* Post:
  + getPosts(): Return all posts
  + editPost(): Update post
  + addPost(): Add post to the forum
* Comment:
  + getComment(): Get a list of post comments
  + deleteComment(): Delete a comment added by the user
  + addComment(): Add comment to the post
* …

## Activity Diagram:

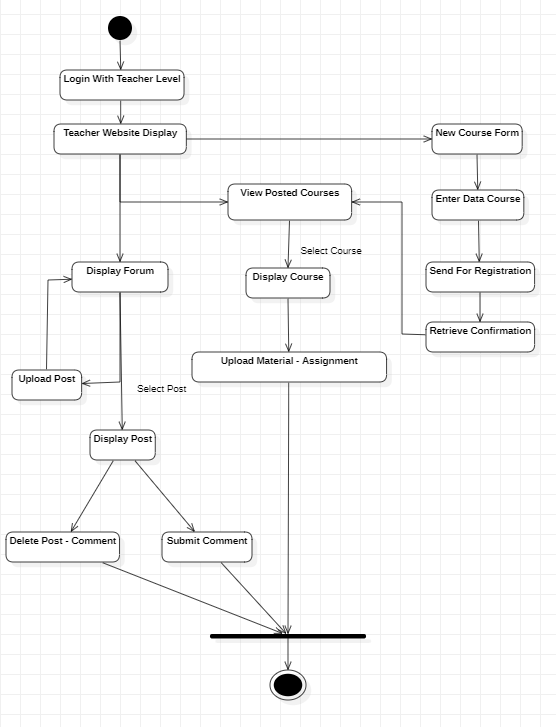
* Login and Registration Activity:



* Student Activity:



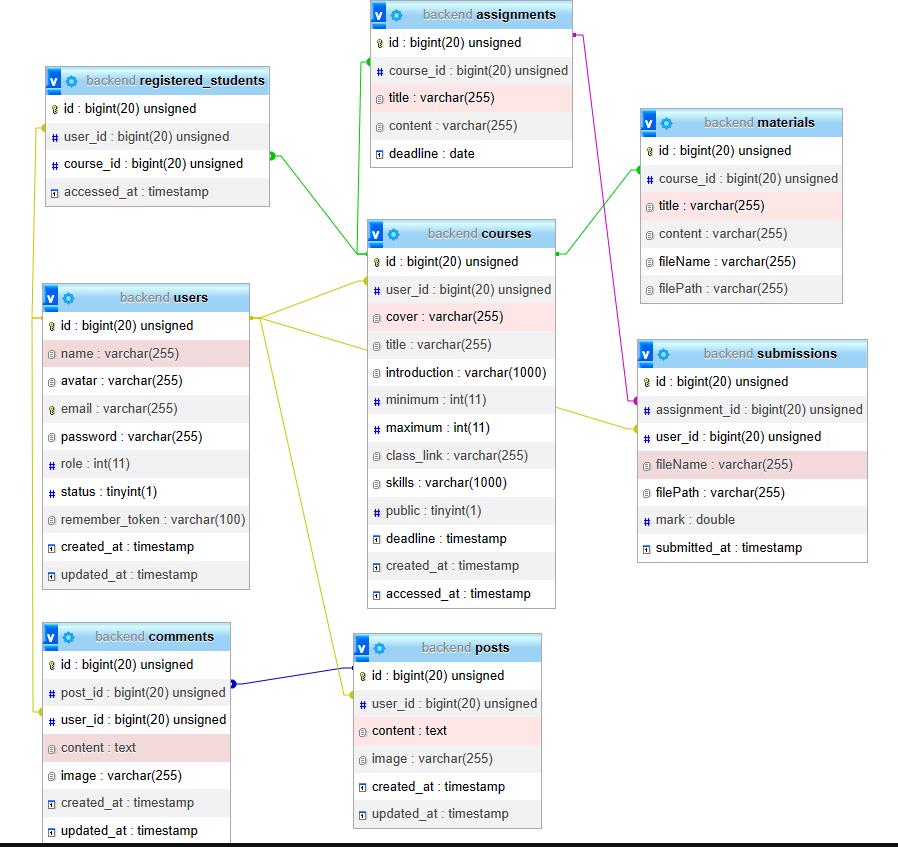
* Teacher Activity:



* Admin Activity:



## Database:

******

# CHAPTER 4: IMPLEMENTATION AND BUILDING

## Build courses recommend system:

### Analysis and data processing:

To build the course recommendation system, we pull the data from mysql into 3 columns ['id','title','skill'] then we push the data into a CSV file called products.csv. Next read the CSV file and delete all rows with NULL values from the DataFrame

db = mysql.connector.connect(host="localhost",

                    user="root",

                    passwd="",

                    db="backend")

csvHeader = ["id", "title", "skills"]

data = []

code = "SELECT \* FROM `courses`"

cur = db.cursor()

cur.execute(code)

records = cur.fetchall()

with open('./products.csv', 'w', encoding='UTF8') as f:

    f.truncate()

    f.write(csvHeader[0]+","+csvHeader[1]+","+csvHeader[2])

    for x in records:

        split = x[8].split(",")

        skill = "|".join(split)

        a = arr.array('i',range(len(split)))

        print(a)

        if len(split) > 1:

            for i in split:

                row = "\n"+str(x[0])+","+str(x[3])+","+i

            f.write(row)

        else:

            row = "\n"+str(x[0])+","+str(x[3])+","+skill

            f.write(row)

    f.close()

df1 = pd.read\_csv('products.csv',encoding='utf-8')

df1 = df1.dropna()

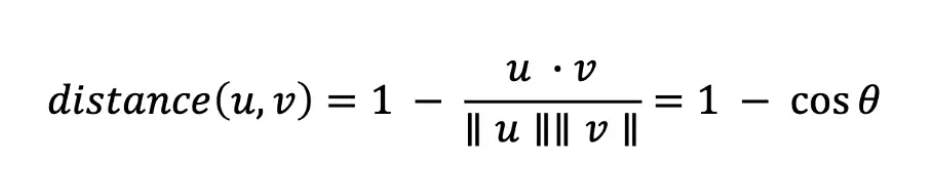
df1.dropna()

### Generating course recommendation:

We have 3 steps to achieve our goal:

* Calculate the average vector of skills and metadata for each course the user is viewing.
* Find the closest n data points in the dataset to this mean vector.
* Take these n points and recommend courses corresponding to them.

This algorithm follows a common approach used in content-based recommendation systems and is generalizable because we can mathematically determine the closest term to a range of metrics. distance, from the classical Euclidean distance to the cosine distance. For the purposes of this project, I have used the cosine distance, defined below for the two vectors u and v.



In other words, the cosine distance is one minus the cosine similarity — the cosine of the angle between the two vectors. The cosine distance is commonly used in recommender systems and can work well even when the vectors being used have different magnitudes. If the vectors for two course are parallel, the angle between them will be zero, meaning the cosine distance between them will also be zero because the cosine of zero is 1.

tdf = TfidfVectorizer(stop\_words='english')

df1['skills'] = df1['skills'].fillna('')

tdf\_matrix = tdf.fit\_transform(df1['skills'])

tdf\_matrix.shape

consine\_sim = linear\_kernel(tdf\_matrix, tdf\_matrix)

indices = pd.Series(df1.index,index=df1['title']).drop\_duplicates()

def get\_recommend(id):

df2 = df1.loc[df1['id'] == int(id)]

index = df1[df1['skills'] == df2['skills'].values[0]].index[0]

sim\_scores = list(enumerate(consine\_sim[index]))

sim\_scores = sorted(sim\_scores, key=lambda X: X[1], reverse=True)

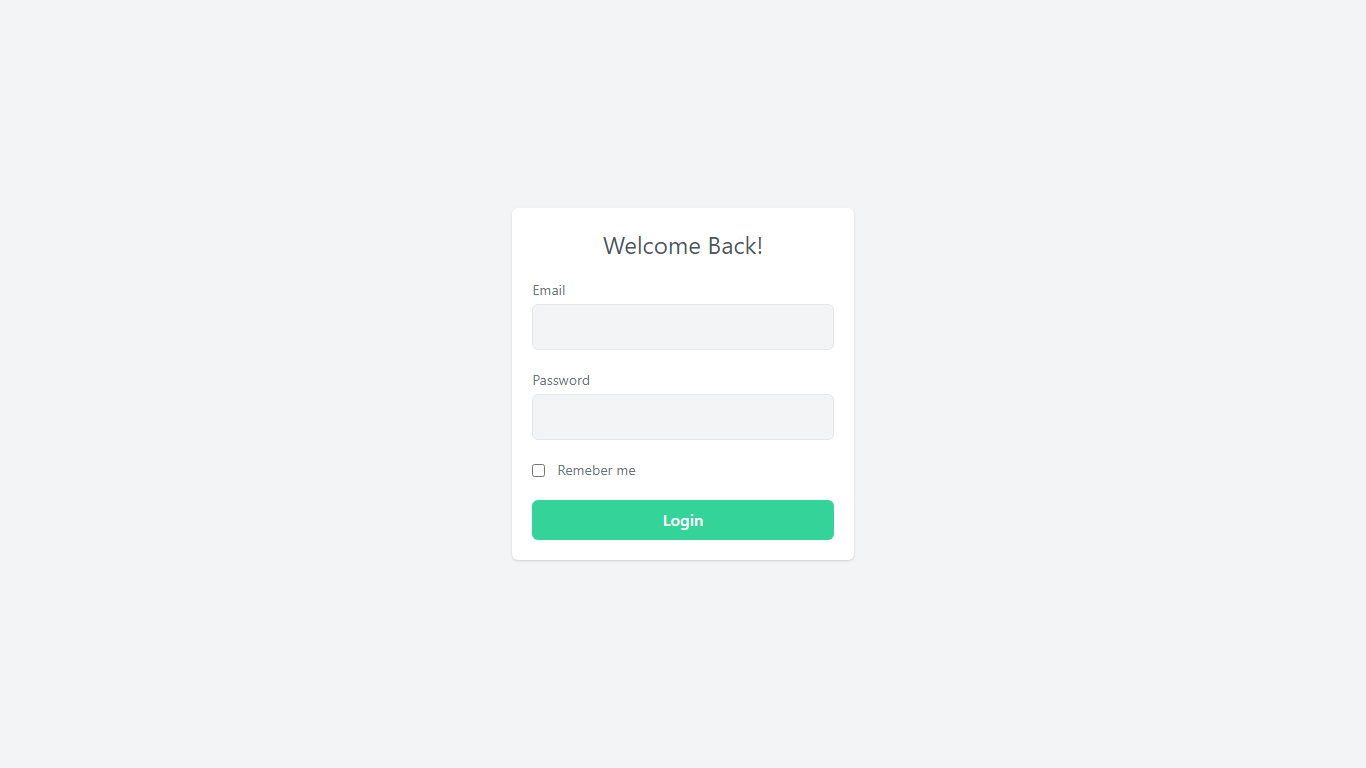
sim\_scores = sim\_scores[1:16]

tech\_indices = [i[0] for i in sim\_scores]

return df1[['title']].iloc[tech\_indices].to\_json(orient='records', lines=True)

## Build the website:

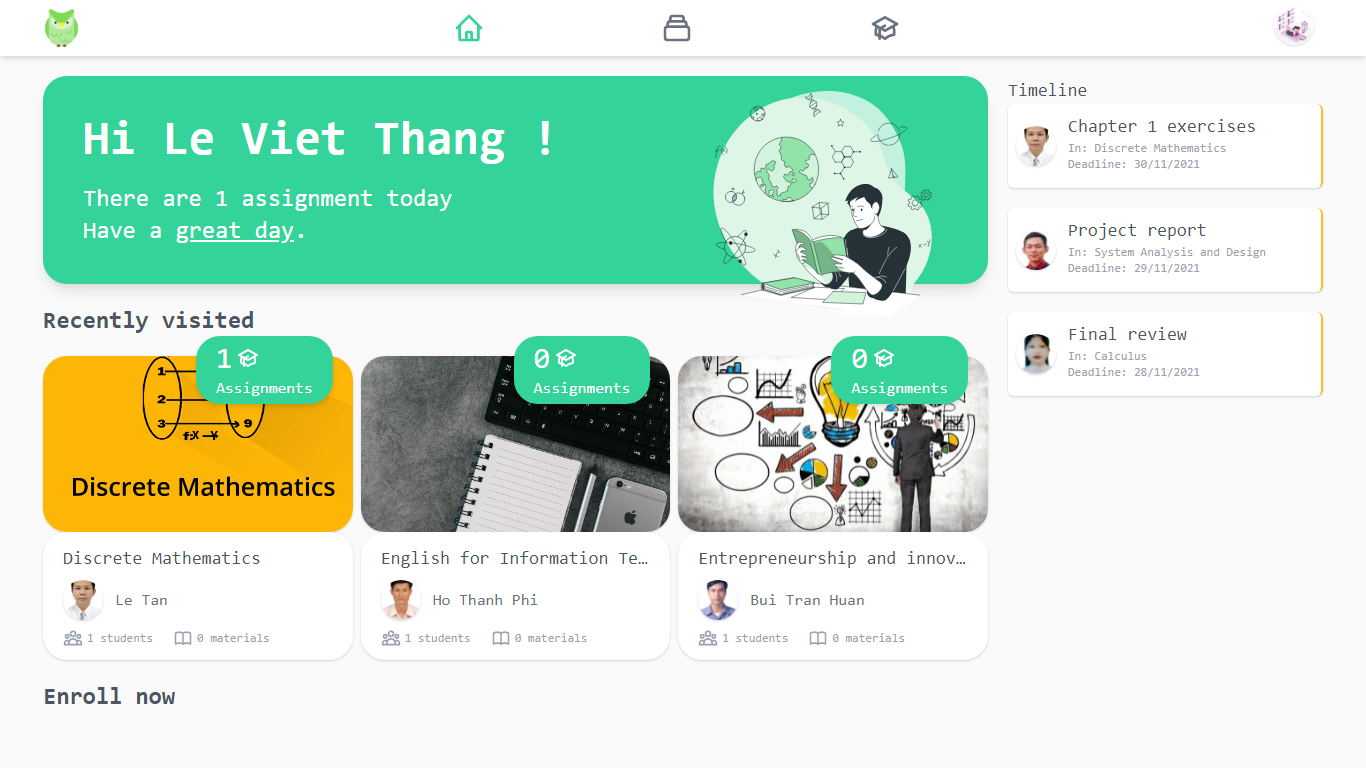
### 4.3.1 Login:



***Figure 4.2 Login page***

### 4.2.2 Home page:

* Student:



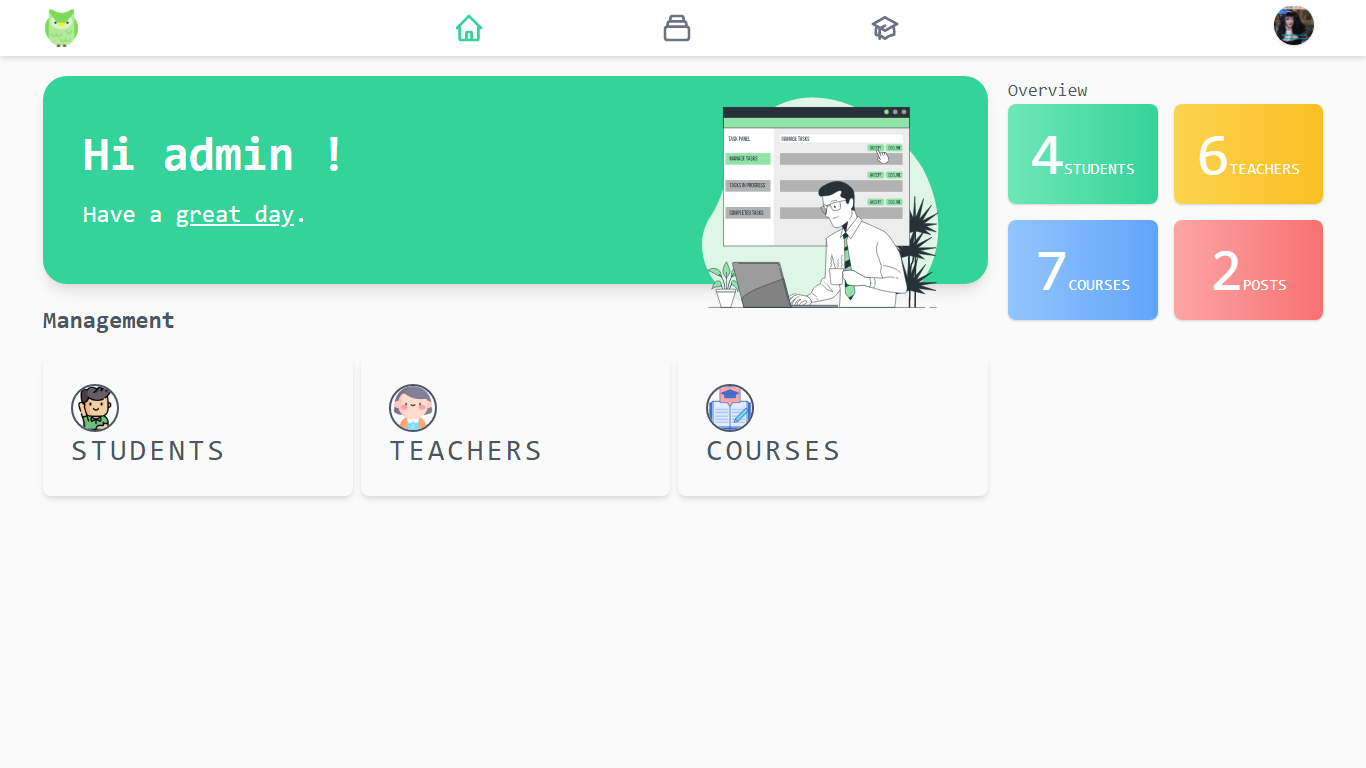
***Figure 4.3 Student homepage interface***

* Instructor:



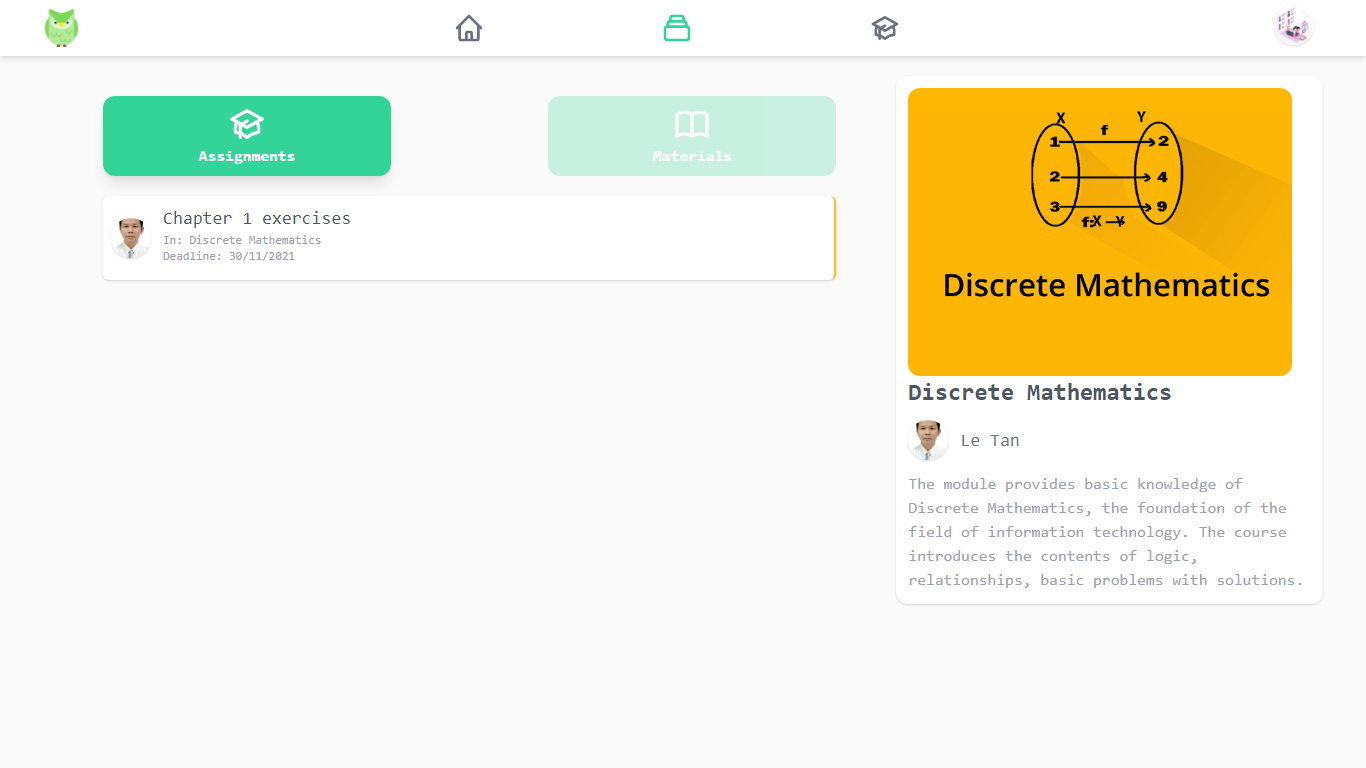
***Figure 4.4 Instructor homepage interface***

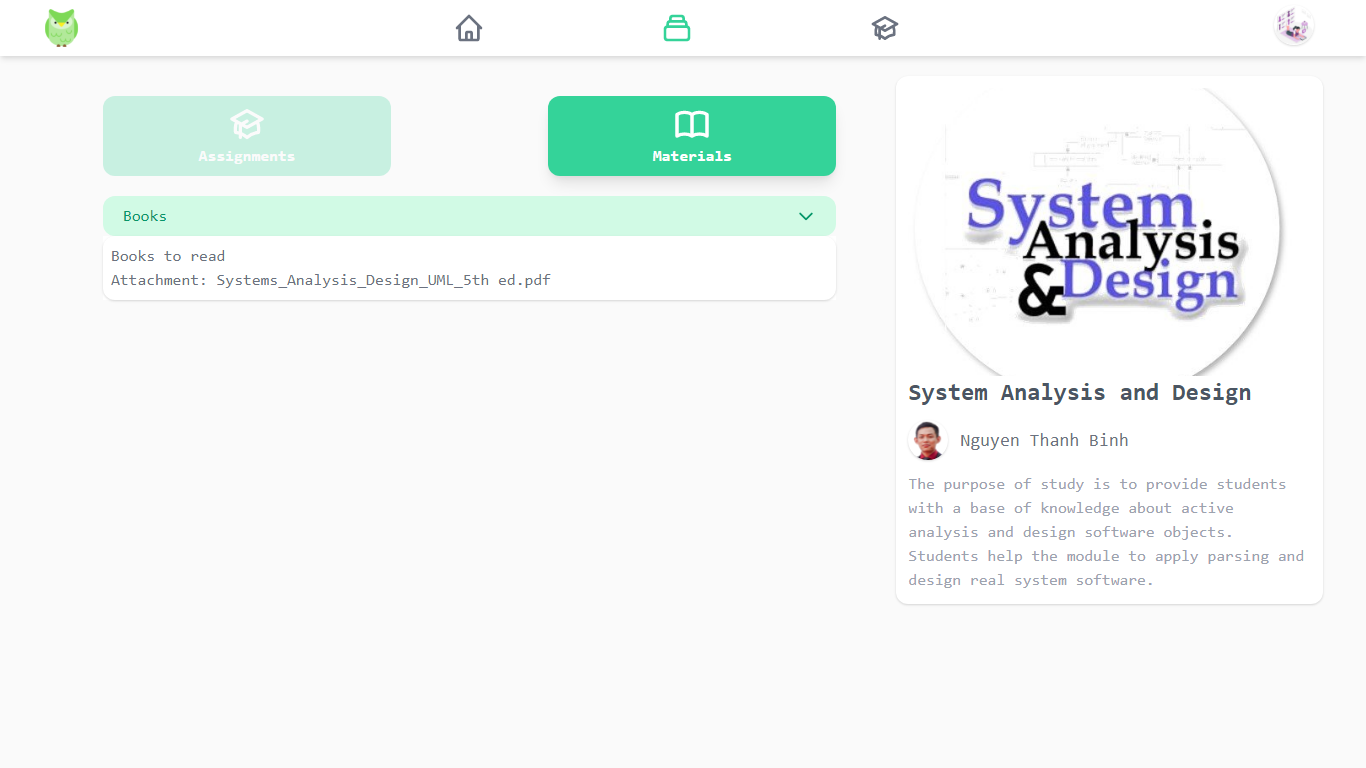
* Admin:

***Figure 4.5 Admin homepage interface***

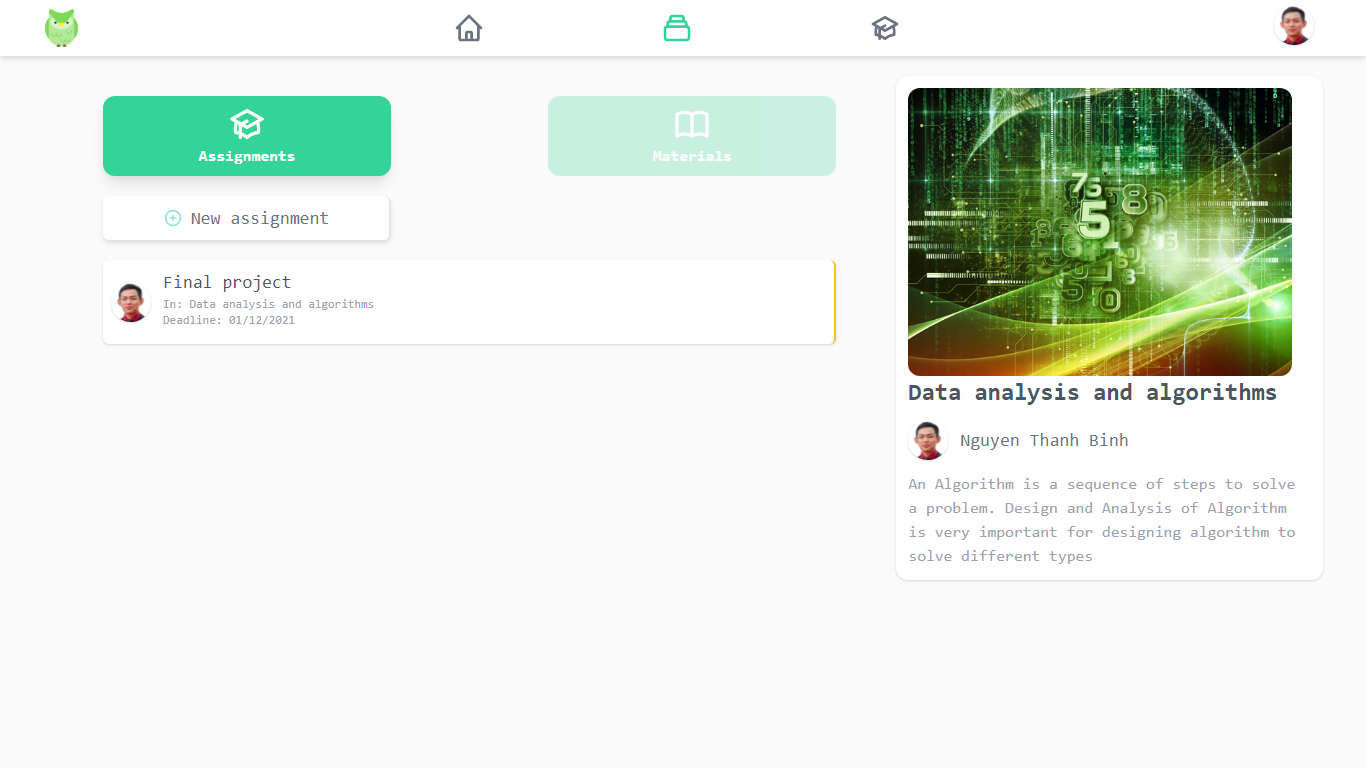
### 4.2.3 When entering a course:

* Student:

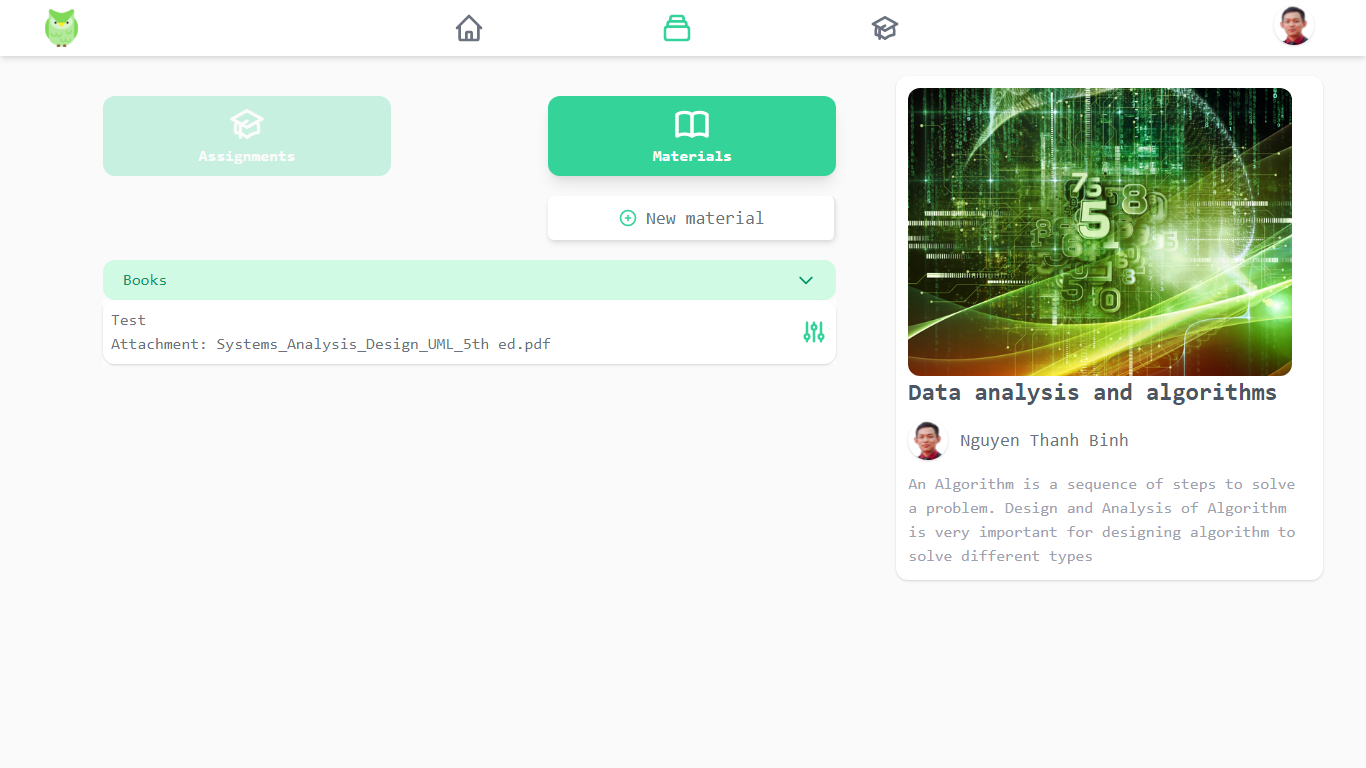
***Figure 4.6 Assignments interface(Students)***

***Figure 4.7 Materials interface(Students)***

* Instructor:

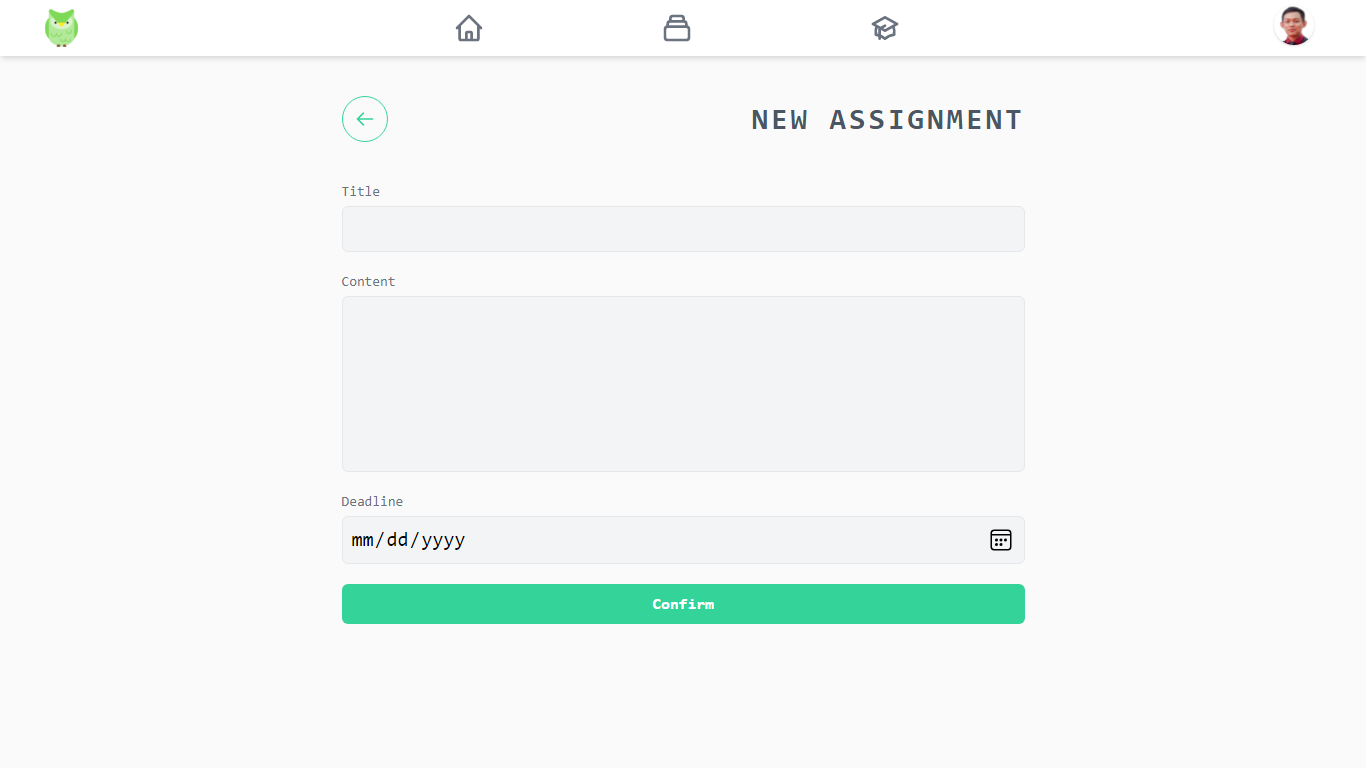


***Figure 4.8 Assignments interface (Instructor***

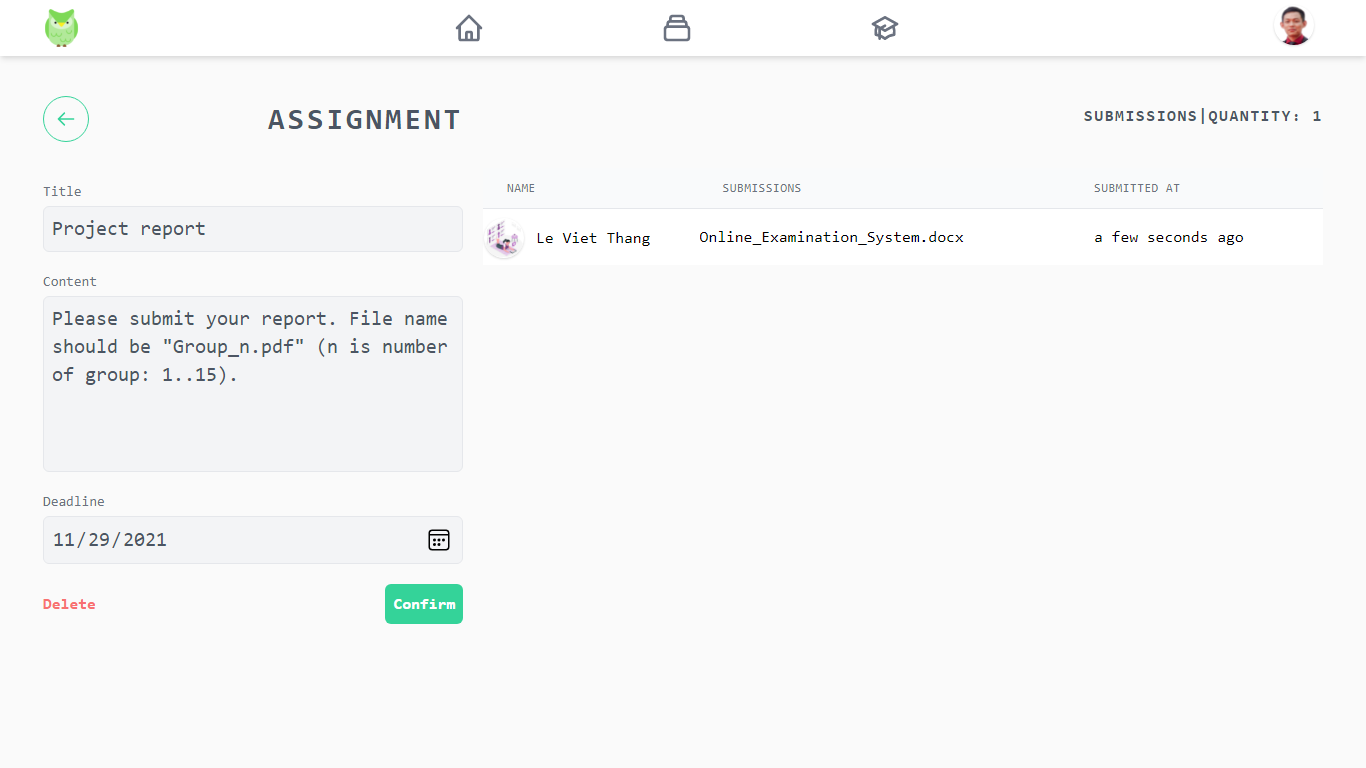


***Figure 4.9 Materials interface (Instructor)***

### 4.2.4 Create assignment (lecturer):

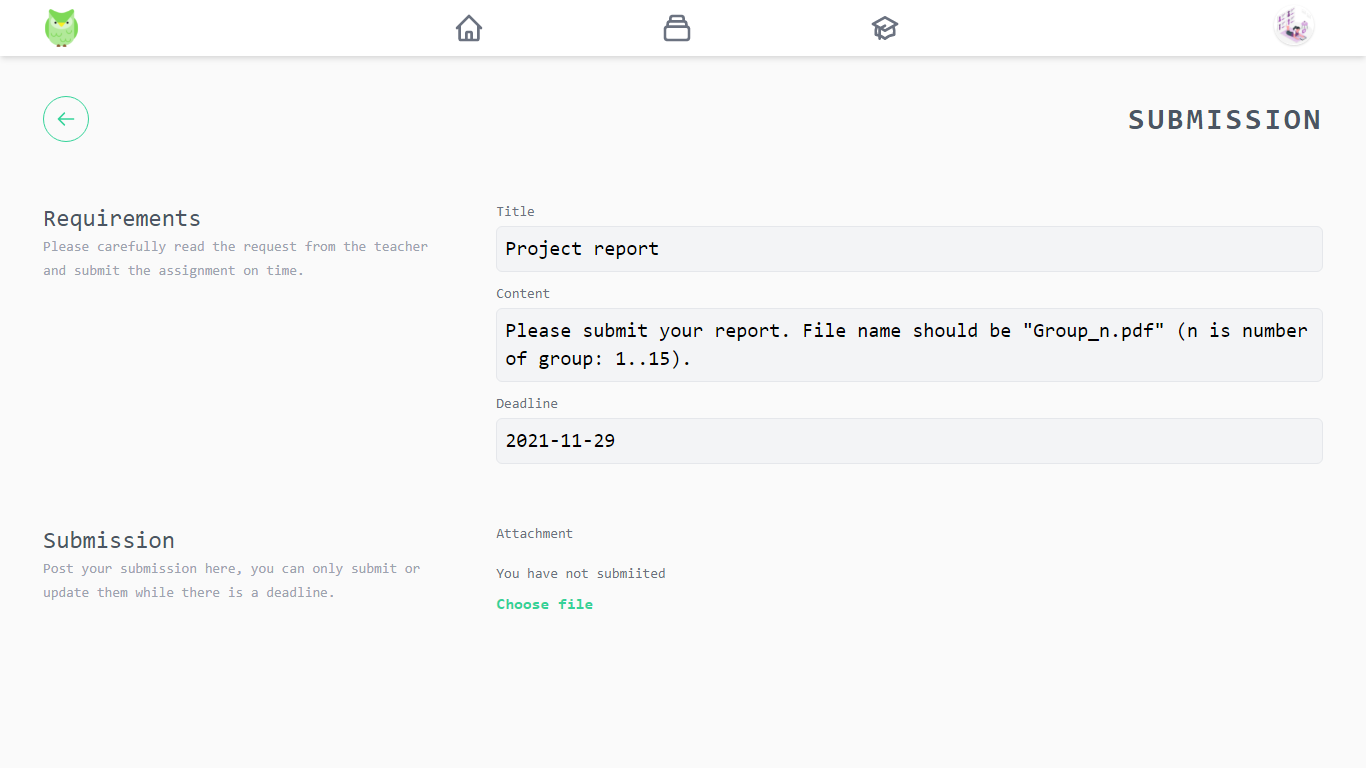
***Figure 4.10 Assignment creation form***

### 4.2.5 Edit assignment, see submission (lecturer):



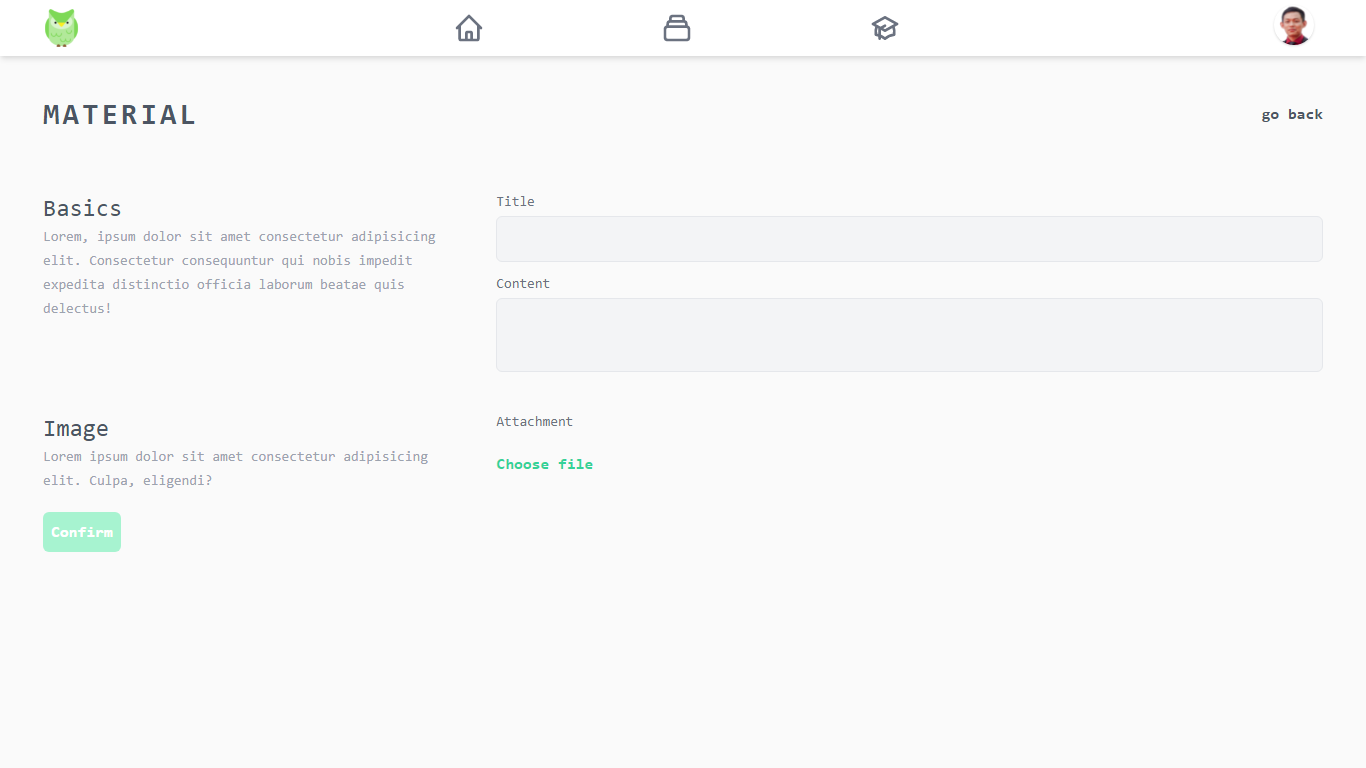
***Figure 4.11 Edit Assignment and view Submit list***

### 4.2.6 Submission of assignments (students):



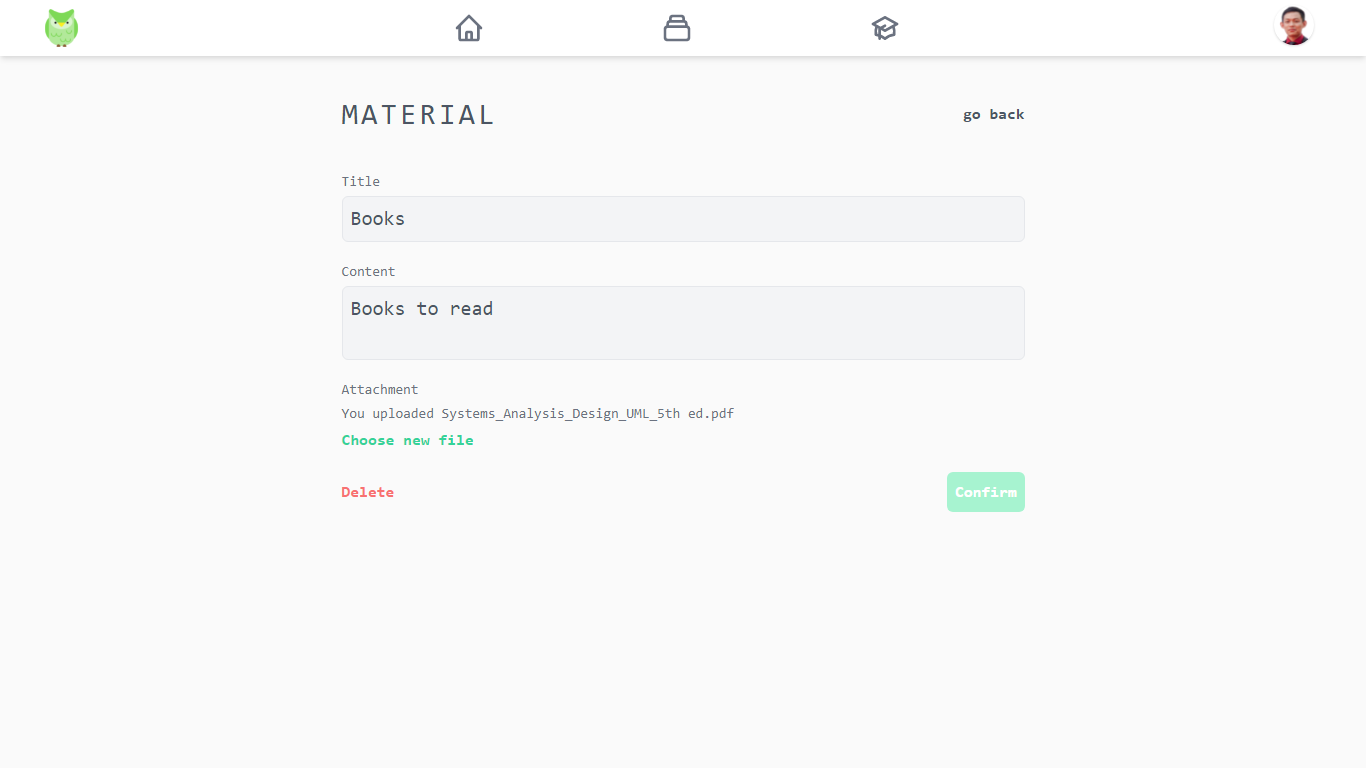
***Figure 4.12 Submission form***

### 4.2.7 Creating documents (instructor):



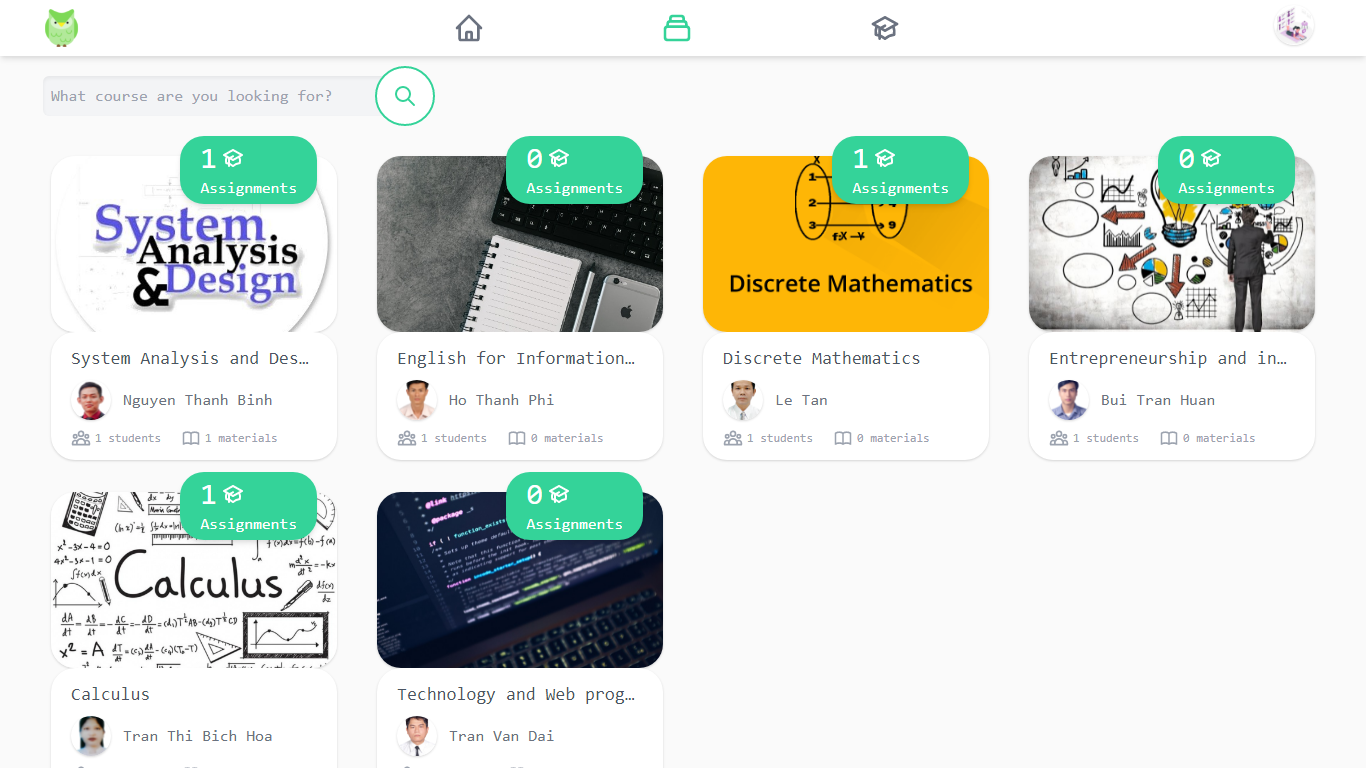
***Figure 4.13 Material creation form***

### 4.2.8 Editing documents (instructor):



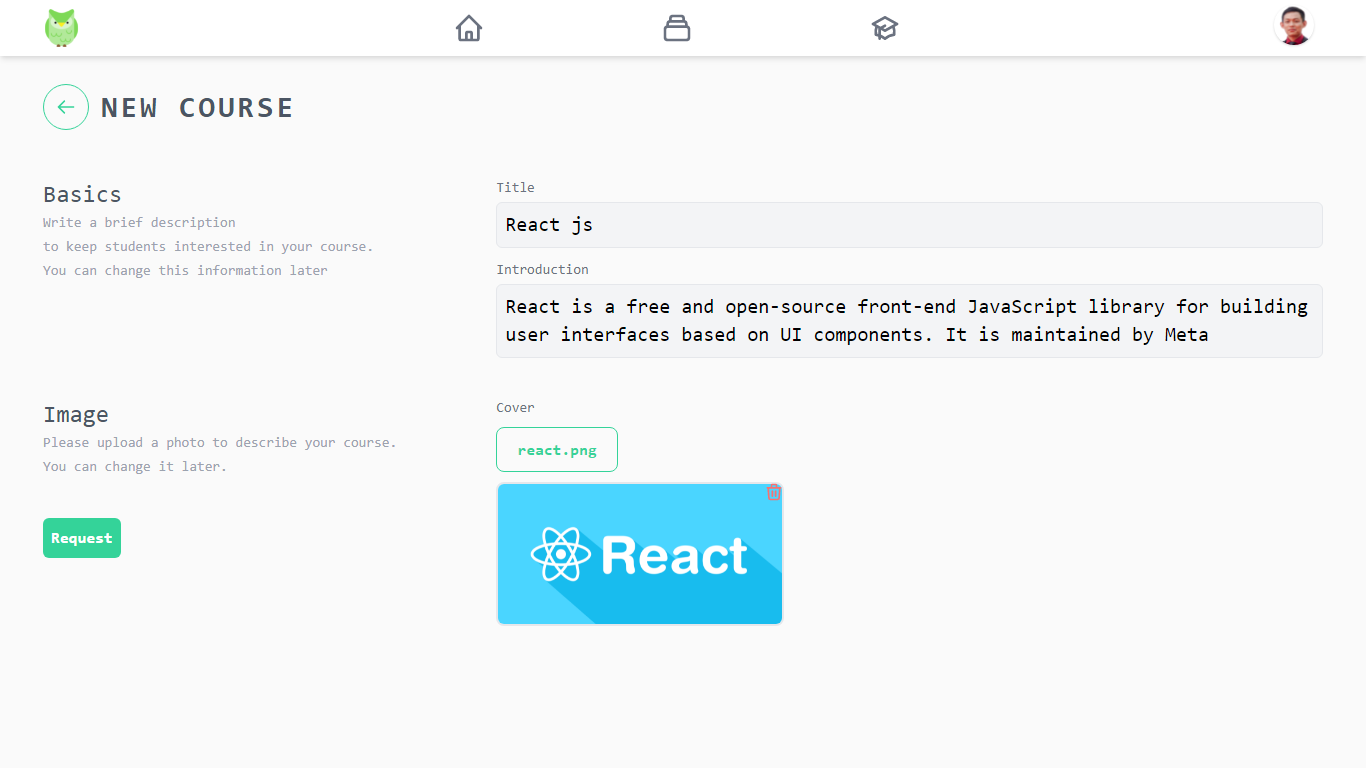
***Figure 4.14 Material editing form***

### 4.2.9 All Courses:



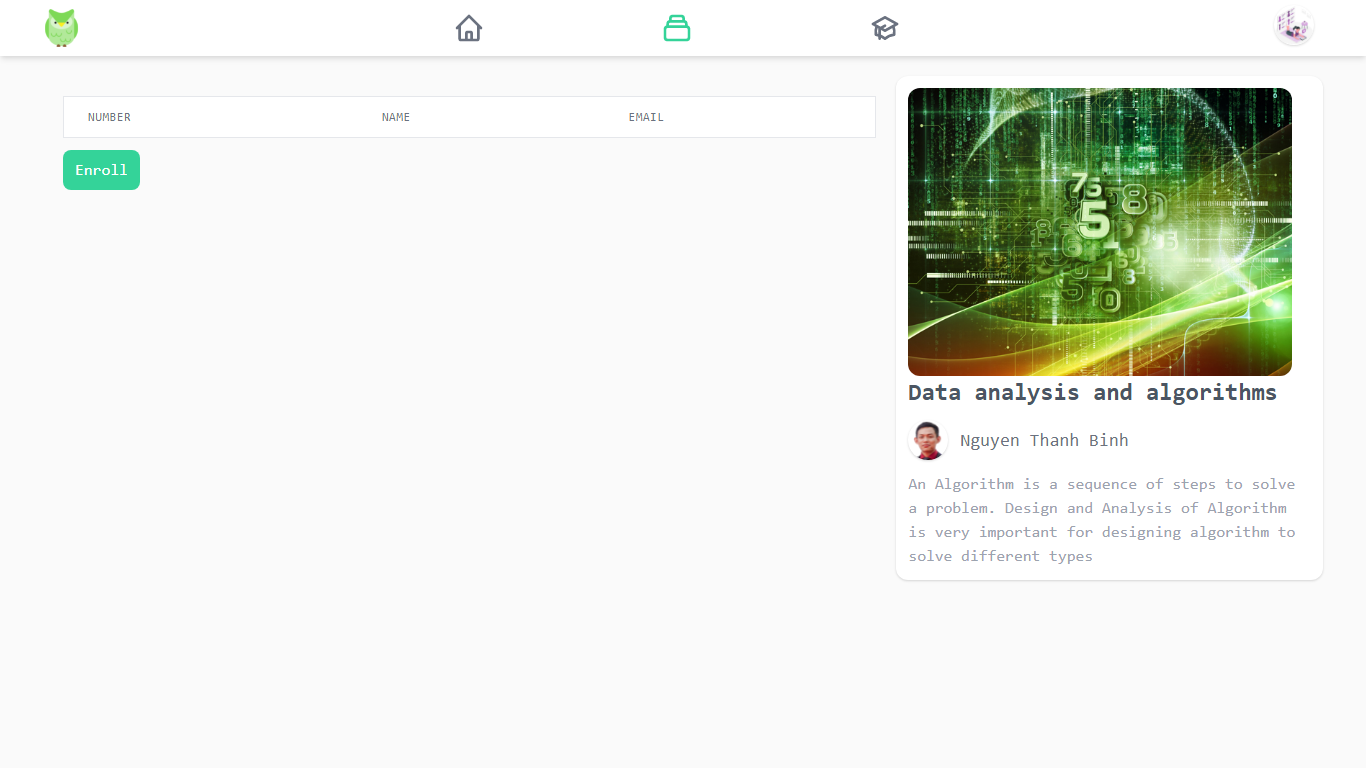
***Figure 4.15 Course List***

### 4.2.10 The teacher request to open the class:

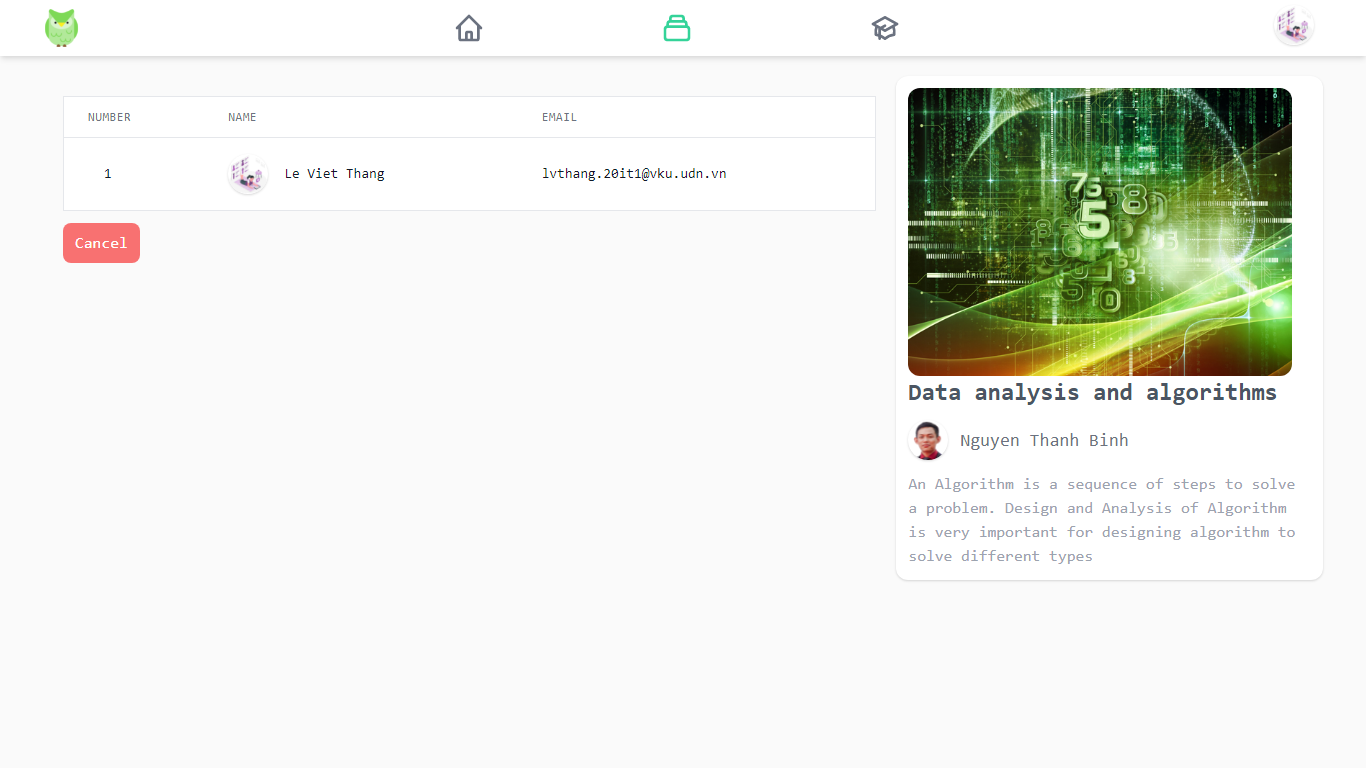


***Figure 4.16 Form request to open class***

### 4.2.11 Students join/cancel classes:

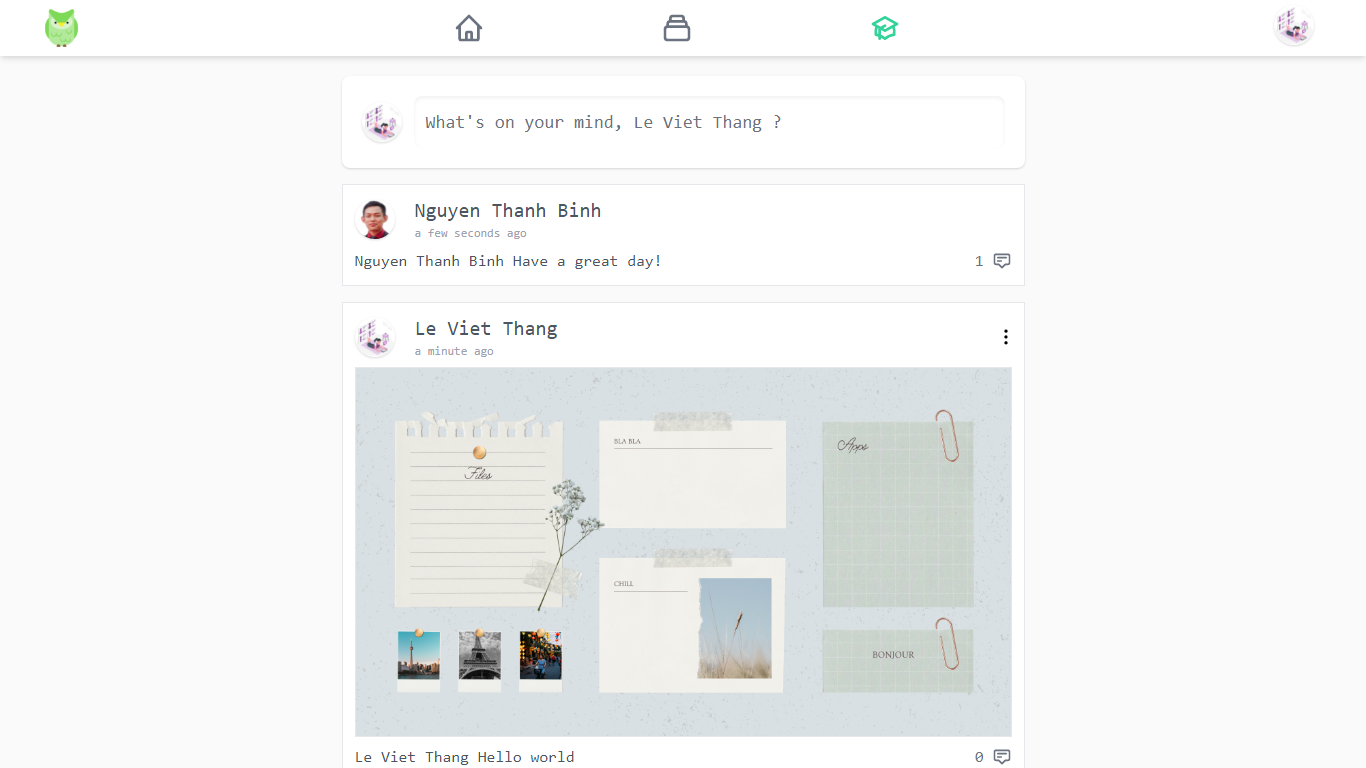


***Figure 4.17 Students join an open class***



***Figure 4.18 The student cancels his/her participation in the registered class***

### 4.2.12 Forums:



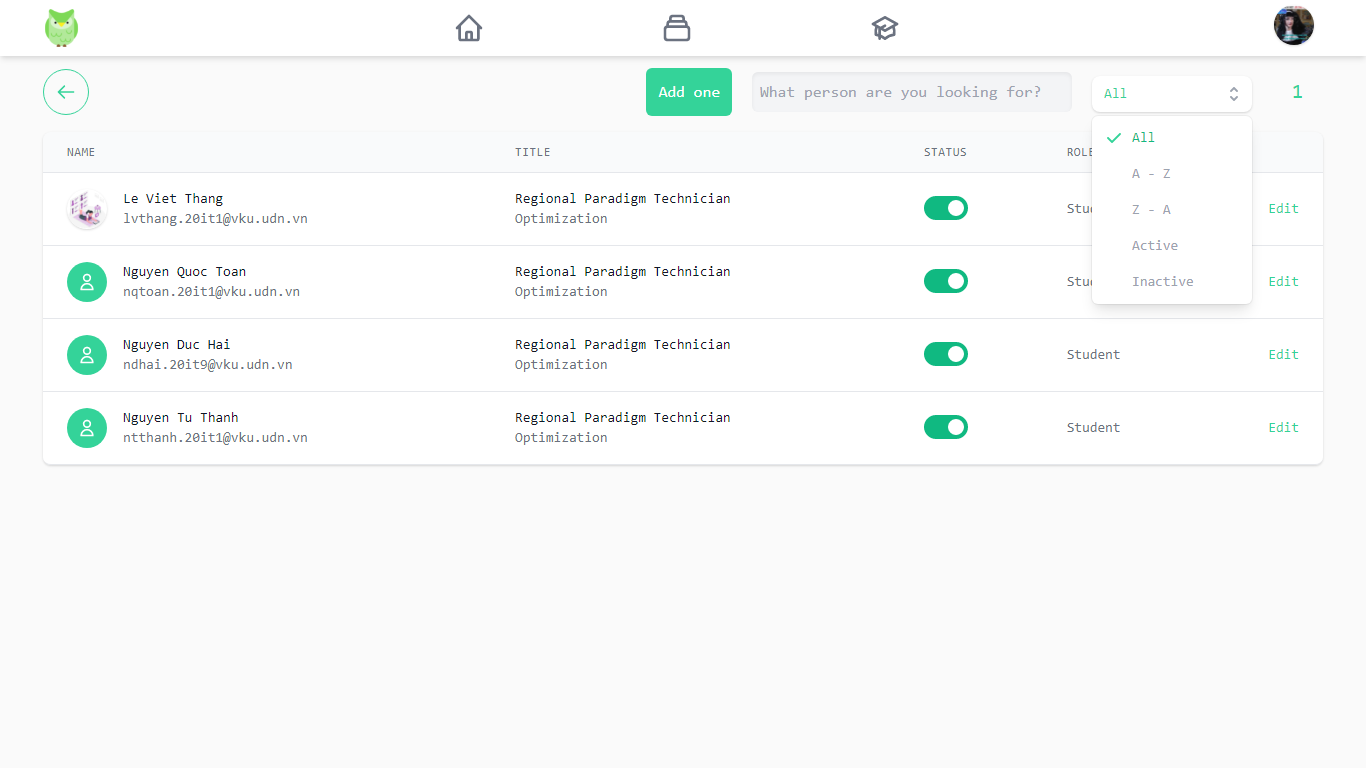
***Figure 4.19 Forum***

### 4.2.13 Comments:



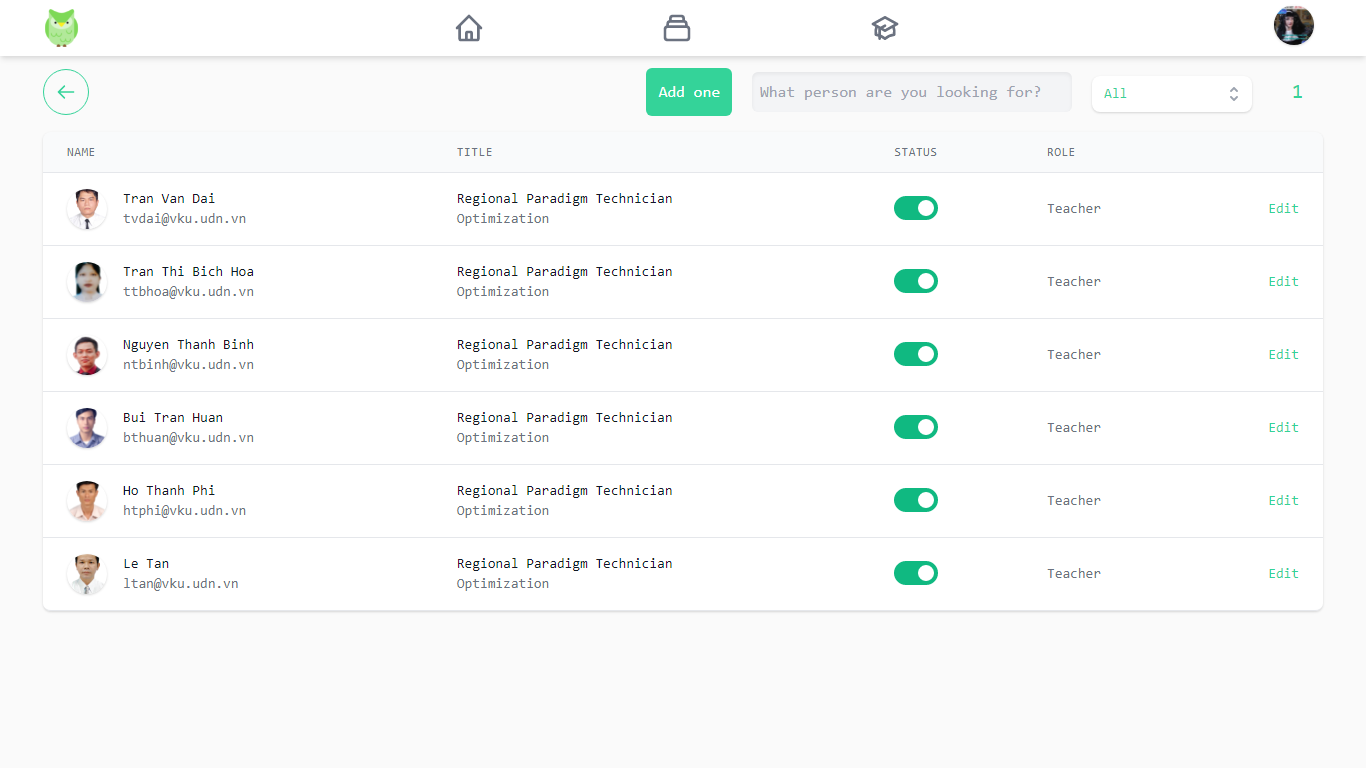
***Figure 4.20 Posts and comments section***

### 4.2.14 Student Management:



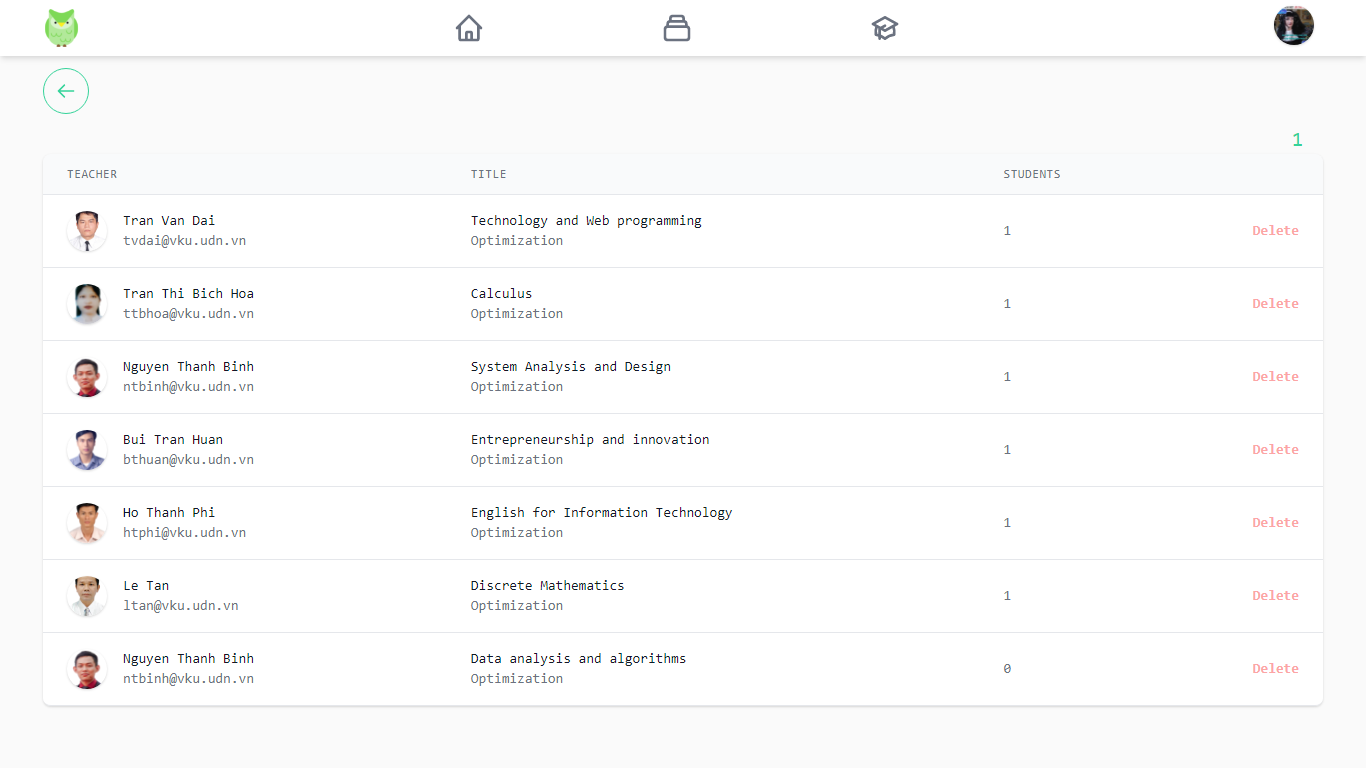
***Figure 4.21 Manage student list***

### 4.2.15 Instructor management:



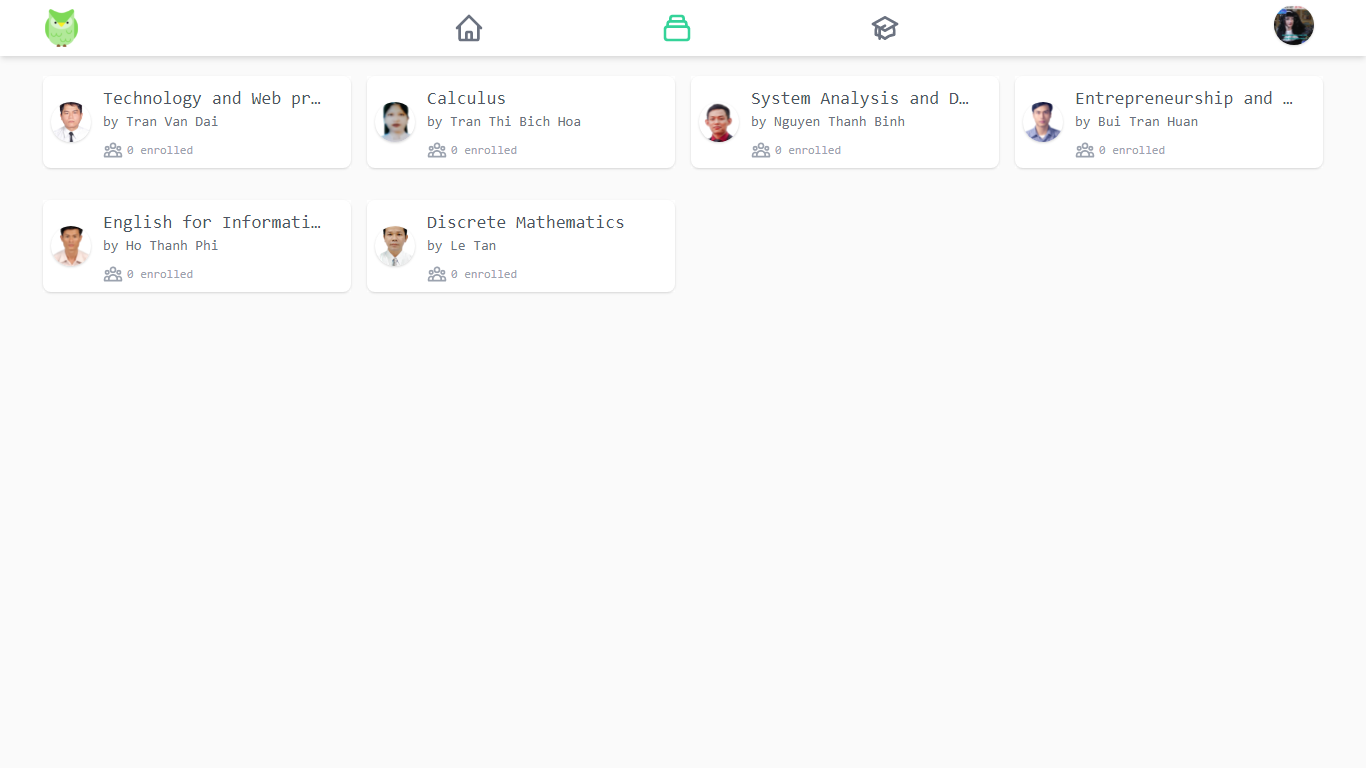
***Figure 4.22 Manage the list of lecturers***

### 4.2.16 Course Management:

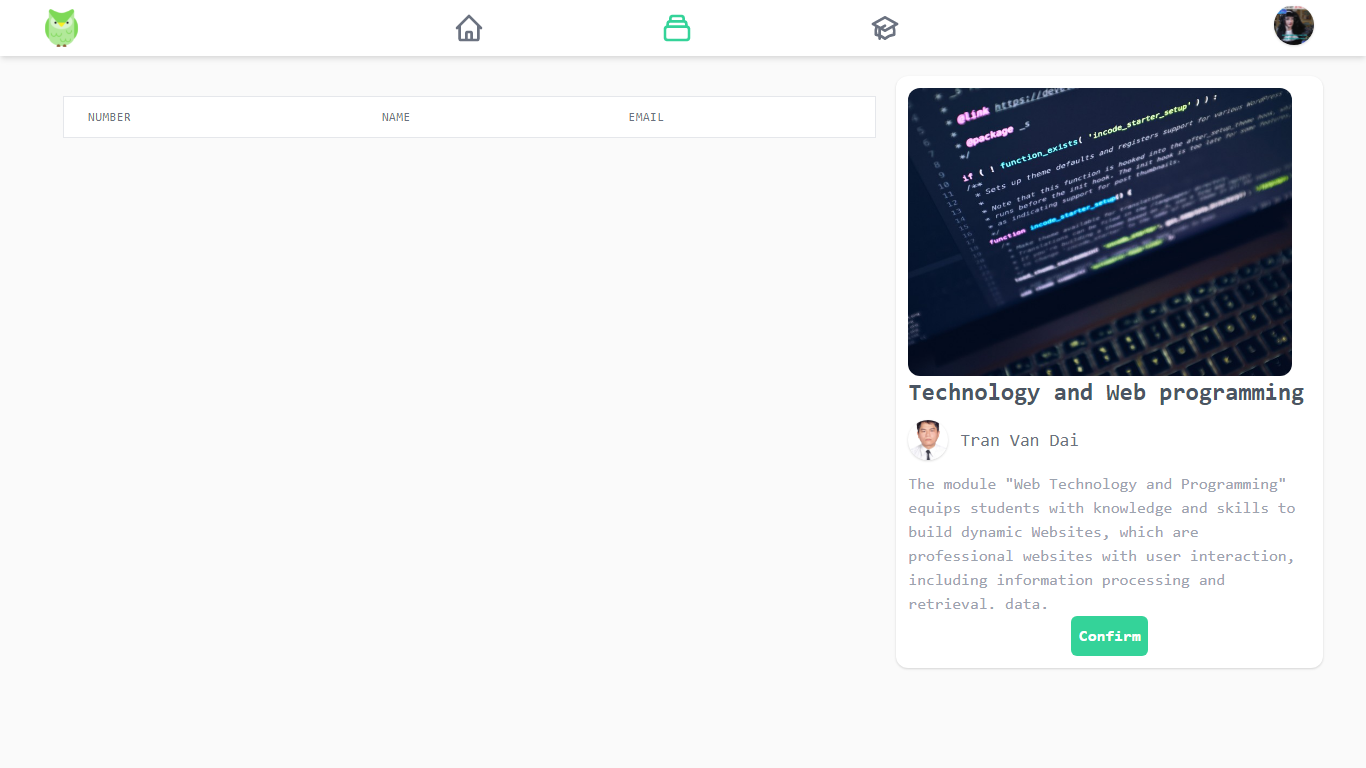


***Figure 4.23 Manage your course list***

### 4.2.17 Browse the course:

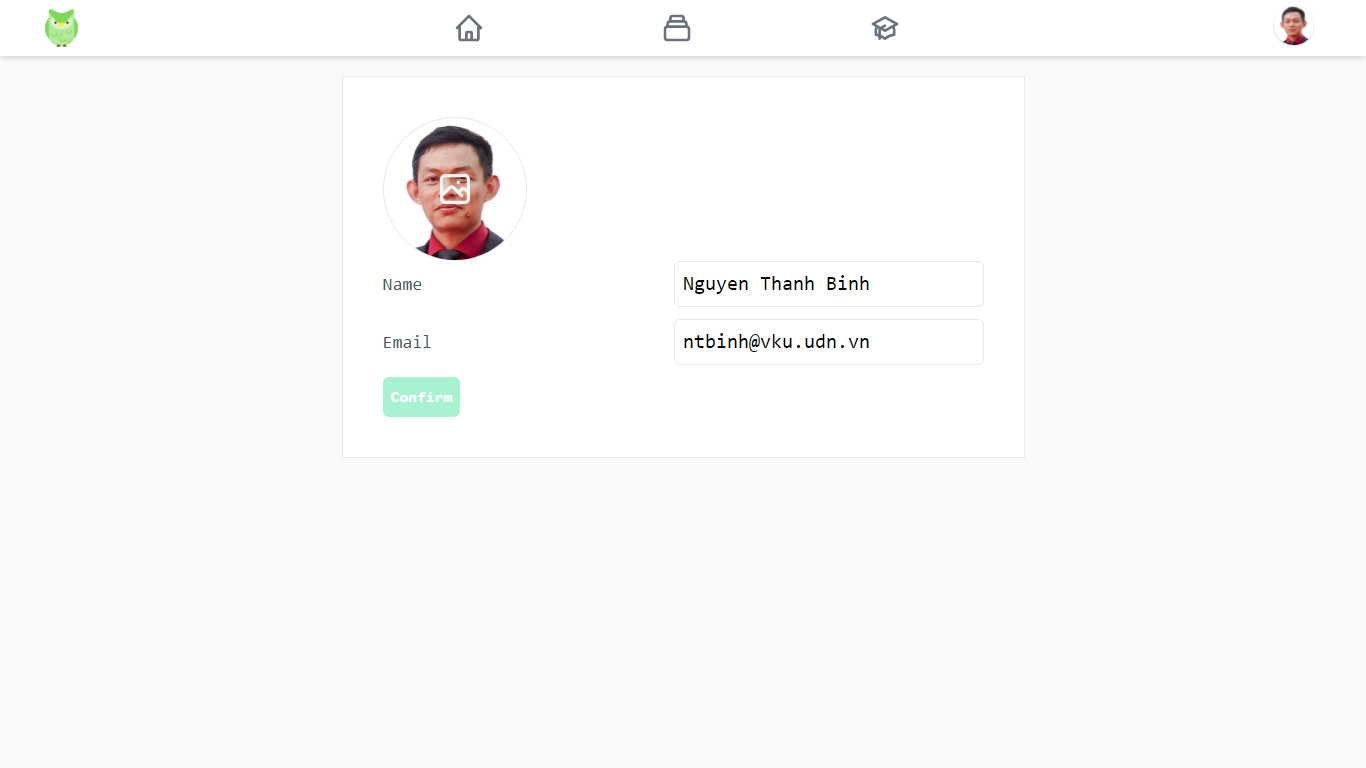
****

***Figure 4.24 Manage classes that are requesting opening***

****

***Figure 4.25 View information about the class you are requesting to open***

### 4.2.18 Edit personal information, avatar:



***Figure 4.26 Edit personal information***

# CHAPTER 5: CONCLUSIONS AND DEVELOPMENT ORIENTATIONS

## 5.1 Conclusion:

### 5.1.1 Achievements:

* Easy-to-use application, user-friendly interface.
* Complete system design analysis.
* Design a learning management system with all the necessary functions.
* Access and familiarize with new frameworks and libraries.

### 5.1.2 Limitations:

* Application response speed is slow.
* There are still some small errors during use.
* The aesthetic is not high.
* The application does not have a messaging function that limits the ability to exchange between users.
* Because the time to perform system analysis and design is relatively limited compared to a relatively broad and rich topic, certain shortcomings cannot be avoided. In addition, the program has some incomplete and inaccurate functions.
* This is an application based on the client/server model with the number of users. To meet this, the system requires a strong and stable server.
* However, building and operating a server takes a lot of effort and cost. This is a real difficulty.

## 5.2 Development:

In order to further develop this topic and be able to apply it in practice, we realize that we need to continue to do the following tasks:

* Improve the response speed of the website to limit the interference in the user's experience.
* Test and continuously fix existing bugs.
* Upgrade more messaging and calling functions to improve online interaction.
* Handle errors more closely before putting them into use.

# LIST OF REFERENCES

1. <https://laravel.com/docs/9.x>
2. <https://tailwindcss.com/docs/installation>
3. <https://reactjs.org/docs/getting-started.html>