

Requirement & Design Specification

**Online Learning Website (SmartClass)**

**Version: 1.0**

– Hanoi, November 2024 –

# Record of Changes

| **Version** | **Date** | **A\* M, D** | **In charge** | **Change Description** |
| --- | --- | --- | --- | --- |
| V1.0 | 15/2 | A |  |  |
| Blog list | 4/11 | A | Chí Thanh | Add search tab, add soft by tag, sorf by date |
| Lesson view | 8/11 | M | Chí Thanh | Modify Lessons by Chapter ID to it can display when users click to others lesson |
| Change Role | 4/11 | M | Tuấn Hưng | User can send request to admin to change role and admin will approve or reject the request |
| User Profile | 4/11 | A | Bách Tùng | Display avatar |
| Pay | 3/11 | A | Bách Tùng | Use sandbox by VNpay to pay |
| Lecturer Subject List | 2/11 | A | Bách Tùng | Change design |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

\*A - Added M - Modified D - Deleted

Contents

[Record of Changes 2](#_heading=h.gjdgxs)

[I. Overview 4](#_heading=h.1fob9te)

[1. System Context 4](#_heading=h.3znysh7)

[2. User Requirements 4](#_heading=h.2et92p0)

[2.1 Actors 4](#_heading=h.tyjcwt)

[2.2 Diagrams 5](#_heading=h.3dy6vkm)

[2.3 Descriptions 6](#_heading=h.1t3h5sf)

[2.4 Main Workflows 6](#_heading=h.4d34og8)

[3. System Functionalities 7](#_heading=h.2s8eyo1)

[3.1 Screens Flow 7](#_heading=h.17dp8vu)

[3.2 Screen Authorization 8](#_heading=h.3rdcrjn)

[3.3 Non-UI Functions 8](#_heading=h.26in1rg)

[II. Functional Requirements 9](#_heading=h.35nkun2)

[1. <<Feature Name1>> 9](#_heading=h.44sinio)

[1.1 <<Screen/Function Name1>> 9](#_heading=h.2jxsxqh)

[1.2 User Login 9](#_heading=h.z337ya)

[1.3 … 9](#_heading=h.3j2qqm3)

[2. System Administration 9](#_heading=h.1y810tw)

[2.1 System Settings 9](#_heading=h.4i7ojhp)

[2.2 … 10](#_heading=h.1ci93xb)

[3. … 10](#_heading=h.3whwml4)

[III. System Design 11](#_heading=h.2bn6wsx)

[1. Database Design 11](#_heading=h.qsh70q)

[1.1 Database Schema 11](#_heading=h.1pxezwc)

[1.2 Table Descriptions 11](#_heading=h.49x2ik5)

[2. Code Packages 11](#_heading=h.2p2csry)

[2.1 Package Diagram 12](#_heading=h.147n2zr)

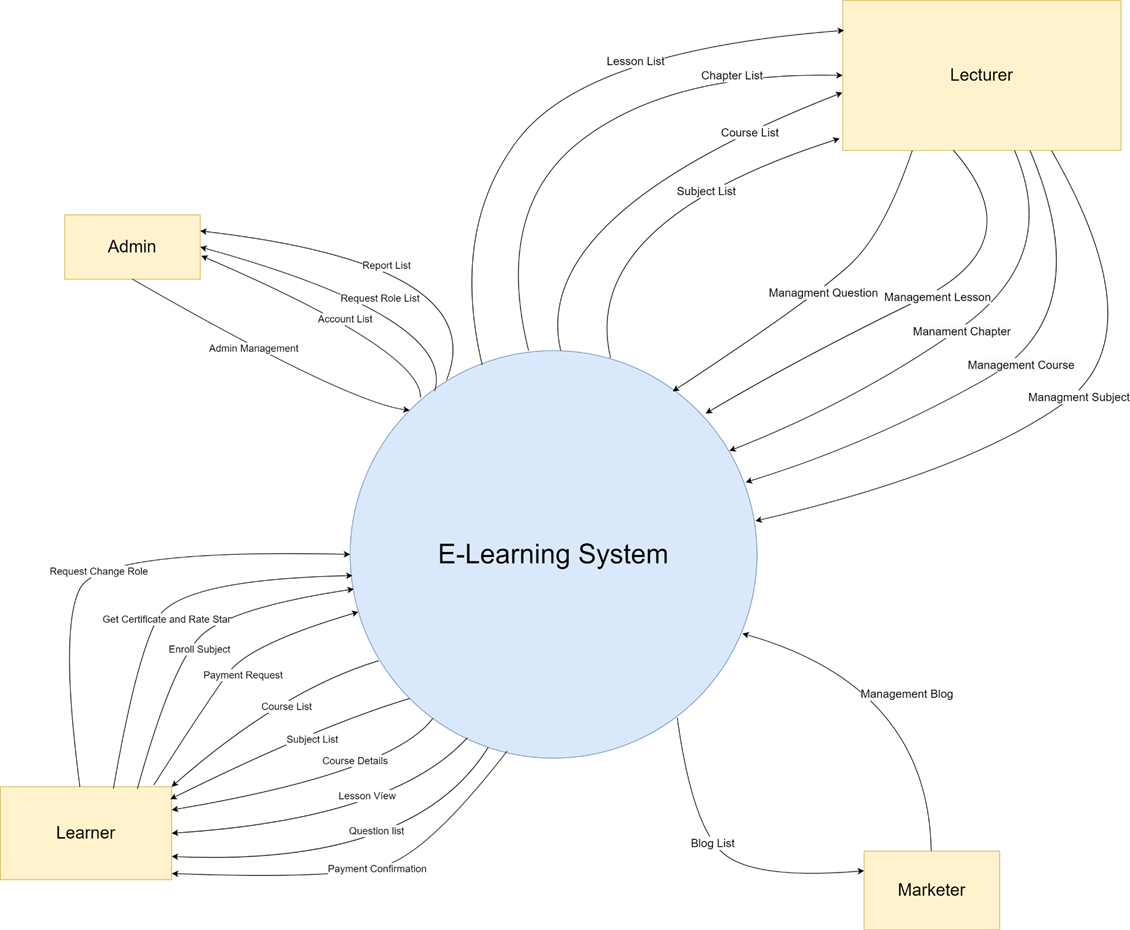
[2.2 Package Descriptions 12](#_heading=h.3o7alnk)

# I. Overview

## 1. System Context

This E-Learning system is an online learning platform with key roles including Admin, Lecturer, Learner, and Marketer. Each role interacts with the system through different functions:

* Admin: manages user accounts, permissions, and generates reports.
* Lecturer :manages courses, lessons, questions, and subjects.
* Learner : can enroll in subjects, make payments, view course content, and complete quizzes.
* Marketer : is responsible for managing blogs and marketing-related content within the system.



## 2. User Requirements

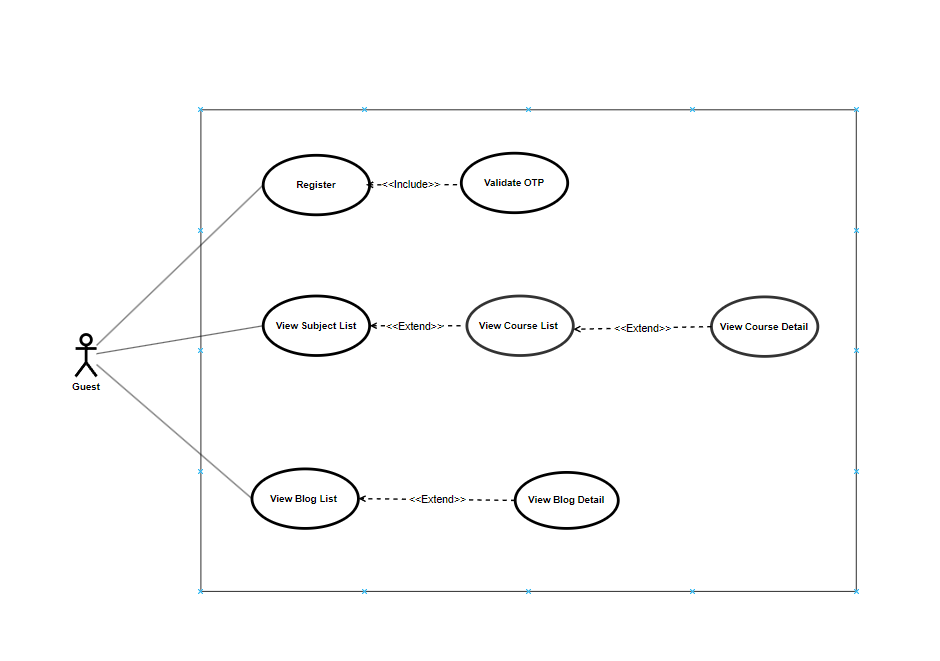
### 2.1 Actors

### 

| **#** | **Actor** | **Description** |
| --- | --- | --- |
| 1 | Admin | An administrator who manages the online learning platform. They can manage user accounts and monitor platform usage. |
| 2 | Guest | A user who has not yet logged in to the website.They can only view the homepage, view course list, the blog list but can not access the function requiring an account. |
| 3 | Learner | A learner can enroll in courses and participate in some online learning activities. |
| 4 | Lecturer | A lecturer who provides, creates and delivers course content. |
| 5 | Marketer | Responsible for creating and implementing marketing strategies to promote the online learning platform. They may create advertising campaigns, and analyze market trends. |

### 2.2 Diagram

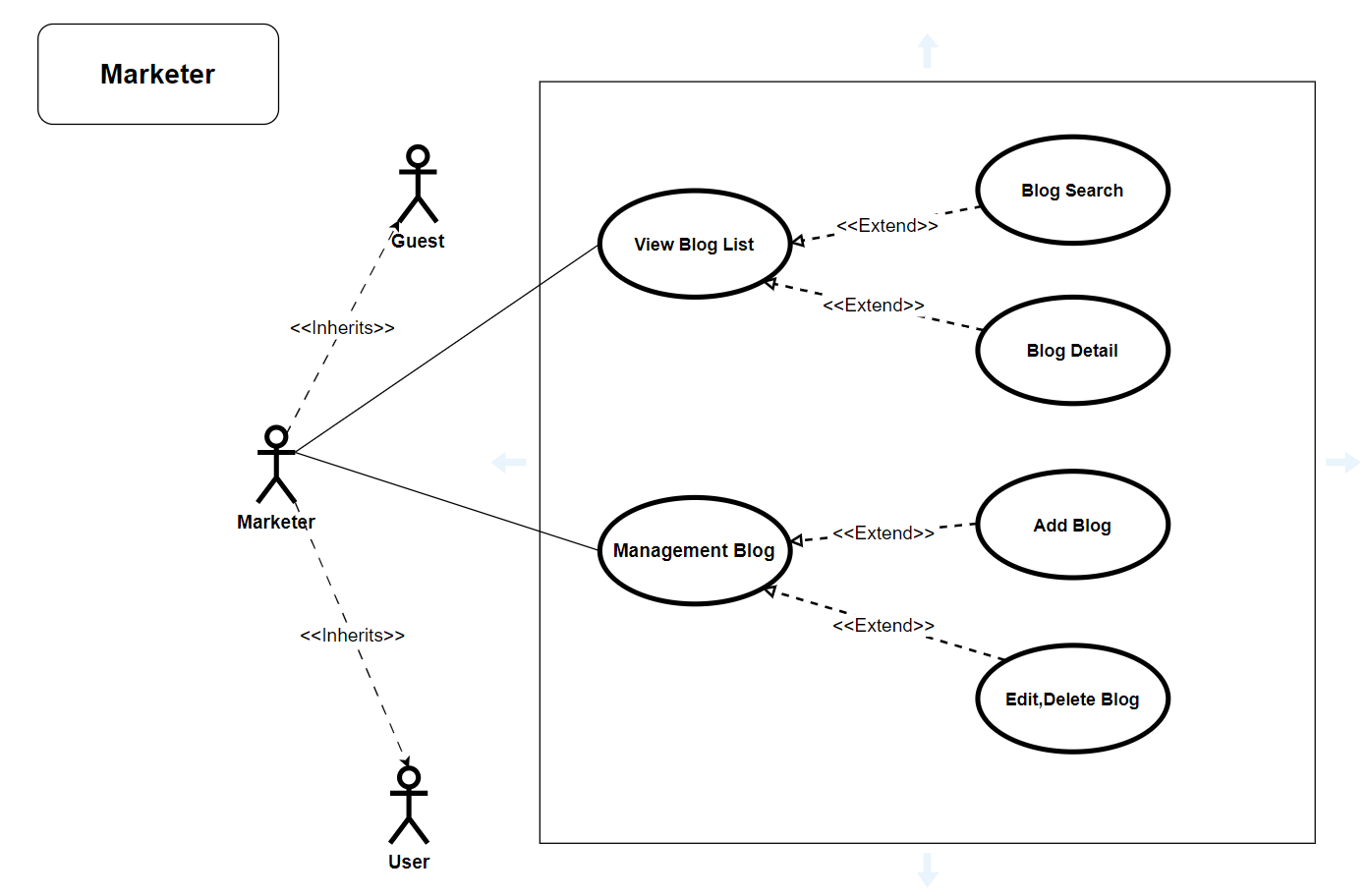
#### 2.2.1 UCs for Guest

**

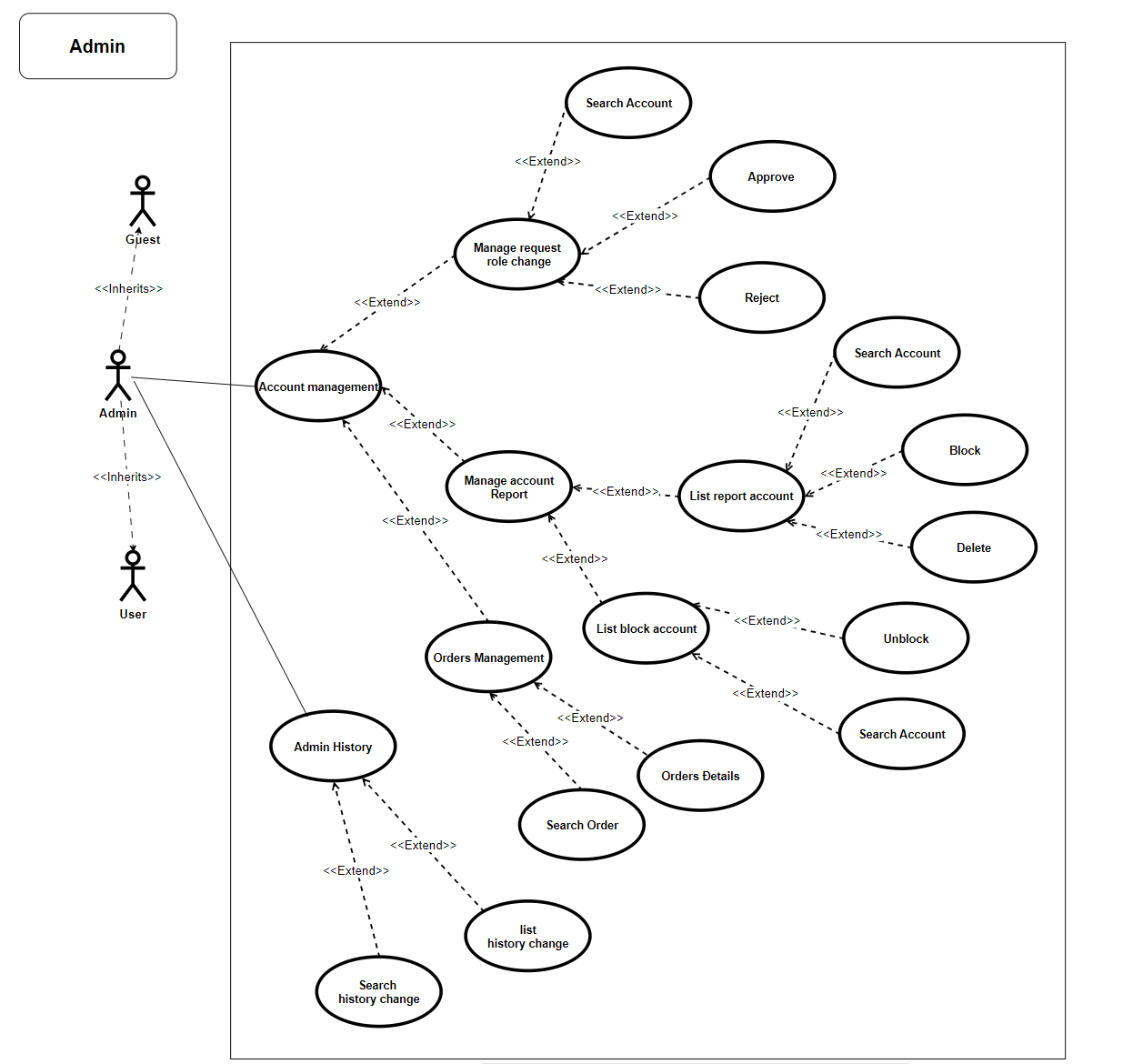
#### 2.2.2 UCs for Learner



#### 2.2.3 UCs for Makerter



#### 2.2.4 UCs for Admin



#### 2.2.5 UCs for Lecture

### 2.3 Descriptions

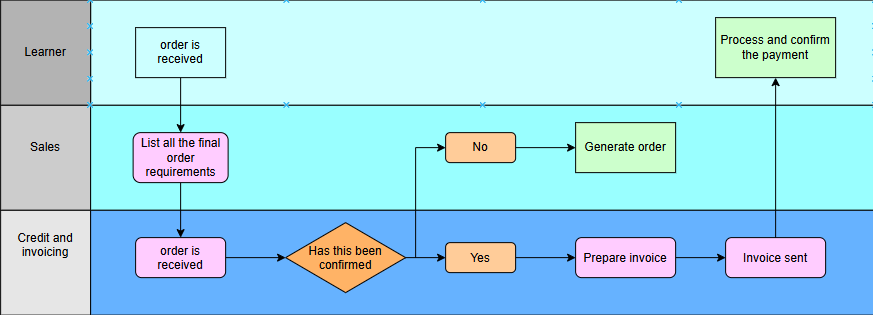
*[This part describes the use cases, you can follow the table form as below]*

| **ID** | **Use Case** | **Actors** | **Use Case Description** |
| --- | --- | --- | --- |
| 01 | User Login | User | User enter username and password if the account is in the database, you can allow access. If not you can’t |
| 02 | User Register | Guest | User will enter username, full name, email, password, mobile, date of birth to create new account to access then validate the OTP |
| 03 | User Authorization | Admin | Admin can view the account and then change their role to lecturer or to marketer |
| 04 | Forget Password |  | Allows users to recover their password when they forget it so they can continue accessing their account. |
| 05 | Change Password |  | After logging in, users can change their password in their user profile. |
| 06 | User Profile |  | Personal user information is displayed such as fullname, email, date of birth, phone number, etc. Users can view their own profile. Users can change they information, if they want. |
| 07 | View Subject List | User | Show all subjects, each subject will have subject name, description, price |
| 08 | Add Subject | Lecturer | Add a new subject including name, description, price , subject image, category. |
| 09 | Subject Management | Lecturer | Allows lecturer to add/update/block any subject in their subject’s list |
| 10 | Course List | User | The function will display subject detail,  description and Includes an image/icon  representing the subject. In subject  detail shows the number of available  courses for the subject |
| 11 | Course Detail | User | The function will display course details,  description and Includes an image/icon  representing the subject, learning  outcomes and contact information  and chapter name. |
| 12 | Home page | User | The HomePage provides an overview of the website's content and functionalities, offering access to various sections such as subject , course, and blog. It serves as the landing page for all users, including guests, learners, lecturers, marketers, and admin |
| 13 | Blog List | Users | User accesses blog page, system displays list of blogs in database. User can view or search blogs. If there are blogs in system, they will be displayed. If not, there will be nothing to display. |
| 14 | Blog Detail | Users | The user selects a blog from the list, the system displays the detailed content of that blog. If the blog exists, the content will be displayed. If the blog does not exist, the details cannot be displayed. |
| 15 | Add Blog | Admin | Admin enters new blog title and content, system saves blog to database. If information is valid, blog will be added and displayed. If invalid, blog will not be saved. |
| 16 | Add Assignment | Lecturer | Lecturer adds exercises by writing content and uploading doc files from computer |
| 17 | Update Assignment | Lecturer | Lecturer updates current assignment by editing content and uploading new doc file from computer |
| 18 | Take Assignment | Leaner | Learners can download the exercise file and do the exercise then submit it and wait for the result. |
| 19 | List Course Assignment | Lecturer | This website will display all the subjects , lecturers can access and get the assignment by student id for grading |
| 20 | View Assignment Submitted Detail | Lecturer | Teachers can view student submitted work from this website and can give comments and grades. |
| 21 | Blog Management |  | The function will display information Blogs (title, content, description, created date , status , tag) and shows the functions of Blog such as: Add blog, Delete Blog, Update Blog. |
| 22 | Edit Subject | Lecturer | The "Edit Subject" function allows lecturers to edit the information of subjects belonging to the lecturer, providing details such as subject name, description, price, image, category |
| 23 | Block Subject |  | The “Block subject” function allows instructors to edit how to view subjects for students who have not joined their course. If they click on the word block, students who have not yet participated in the course will no longer be able to see the course, and vice versa, the active button will let them bring their course back to the web |
| 24 | View Question List |  | lecturer will go to the screen to select a subject, where the question management for that subject will appear after selecting the subject |
| 25 | Add Question |  | 'Add Question' is a function that allows instructors to import questions into a chapter belonging to a subject |
| 26 | Edit Question |  | Question Edit is a function that allows Lecturers to edit question details such as content,option , answers, level |
| 27 | Delete Question |  | Allows instructors to delete questions they feel are inappropriate for the lesson |
| 29 | Chapter Management | Lecturer | Manage chapters and content groups within each course. |
| 30 | View Chapter List | Lecturer | View list of chapters within a course. |
| 31 | Create Chapter | Lecturer | Add a new chapter and set title and description. |
| 32 | Update Chapter | Lecturer | Edit chapter details |
| 33 | Delete Chapter | Lecturer | Remove a chapter from a course. Ensure no data loss. |
| 34 | Course Management |  | Manage detailed courses within each subject. |
| 35 | View Course List |  | View list of created courses within each subject. |
| 36 | Add Course |  | Add a new course including name, description, s |
| 37 | Update Course |  | Update details of an existing course like name, description, …. |
| 38 | Delete Course |  | Remove a course. Ensure no data loss for enrolled students. |
| 39 | Lesson View | Lecturer | Detailed lesson page outlining learning material, lecture video, notes, resources, attachments, notes, comments, etc. |
| 40 | Study Video Lesson |  | Interactive video player for lecture streaming. Supports pause, playback speed, closed captions, highlights, etc. |
| 41 | Read Document Lesson |  | Downloadable document lesson materials like slides, worksheets, study guides, handouts, self-paced tutorials, etc. |
| 42 | Comment Lesson |  | Students can post comments, questions, or discussion topics related to a lesson using a commenting system. |
| 43 | Take Quiz | Learner | Assess knowledge by taking chapter quizzes and graded exams. Questions are randomly generated at levels from easy to difficult |
| 44 | View Quiz Result | Learner | Students can check grades and results of all completed quizzes. Details on answers. |
| 45 | Rate Course Star | Learner | Leave a 1-5 star overall rating for completed courses along with optional text review. Used to rank courses. |
| 46 | Add to Cart |  | Users can add paid courses to a cart to enrol later. Save for later features. List saved course selections. |
| 47 | Checkout |  | Shopping cart integrated secure payment checkout process. Confirm order details before completing the purchase transaction. |
| 48 | Order View |  | Users can view their purchased course order history, status, receipts, refund policy, support docs, credentials, etc. |
| 49 | Request change role | Learner | Learner want to be lecturer or marketer can send the request to admin |
| 50 | Manage request role change | Admin | Allow Admin view list account of learner request to be Lecturer or Marketer and admin can approve or reject |
| 51 | Manage account Report |  | Allow Admin view list report account or list block account. Admin can block or delete reports and can unblock accounts again. |
| 52 | Search Account |  | Quickly find accounts based on search criteria like name, email, account type, registration date range, status, etc. Advanced search filters supported. |
| 54 | Order Details |  | Allow Admin view details orders of user |
| 55 | Admin History |  | Admin views the operating history of the admins on the system |
| 56 | Lesson Management | Lecturer | Manage lessons and content groups within each chapters. |
| 57 | View Lesson List | Lecturer | View list of lessons within a chapter. |
| 58 | Create Lesson | Lecturer | Add a new lesson and set title and description. |
| 59 | Update Lesson | Lecturer | Edit lesson details |
| 57 | Delete Lesson | Lecturer | Remove a lesson from a chapter. Ensure no data loss. |
| 58 | Subject review | Lecturer | lecturer can view the functionality of the subjects |
| 59 | Course Management (add) By Lecturer | Lecturer | The "Add Course" function allows lecturers to create and add a new course to the educational platform, providing details such as course name, description for the couse. |
| 60 | Course Management (delete, update) By Lecturer | Lecturer | Allows lecturer to update/delete any course in their subject list |
| 61 | OrderAnalysis | admin | This interface provides a comprehensive overview for order and enrollment analysis, featuring buttons to explore course popularity, order volume, user trends, and revenue data. The Enrollment Statistics table offers a clear view of each subject's enrollment numbers, allowing users to assess course engagement. The layout and color scheme make navigation and data interpretation intuitive and visually appealing. |
| 62 | Chat Bot | leaner | Chat bot is created by use gemini api key, help learner find more information fast |

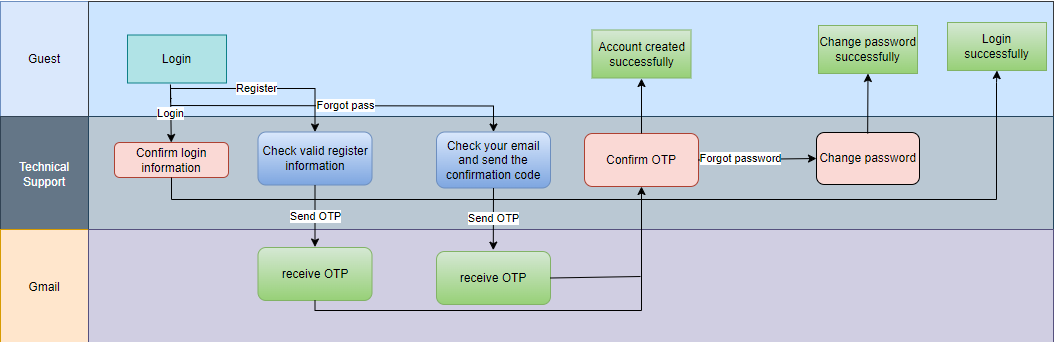
### 2.4 Main Workflows

*[Provide main workflows (business processes) using the swim-lane diagram(s) like sample below]*

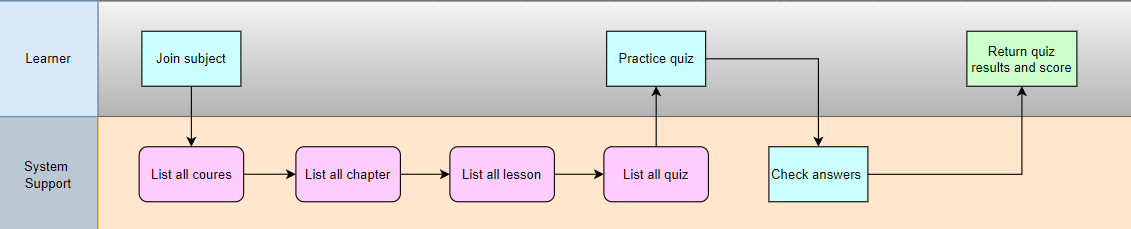
#### 2.4.1 Order Processing



#### 2.4.2 Account Processing



#### 2.4.3 Learn Processing

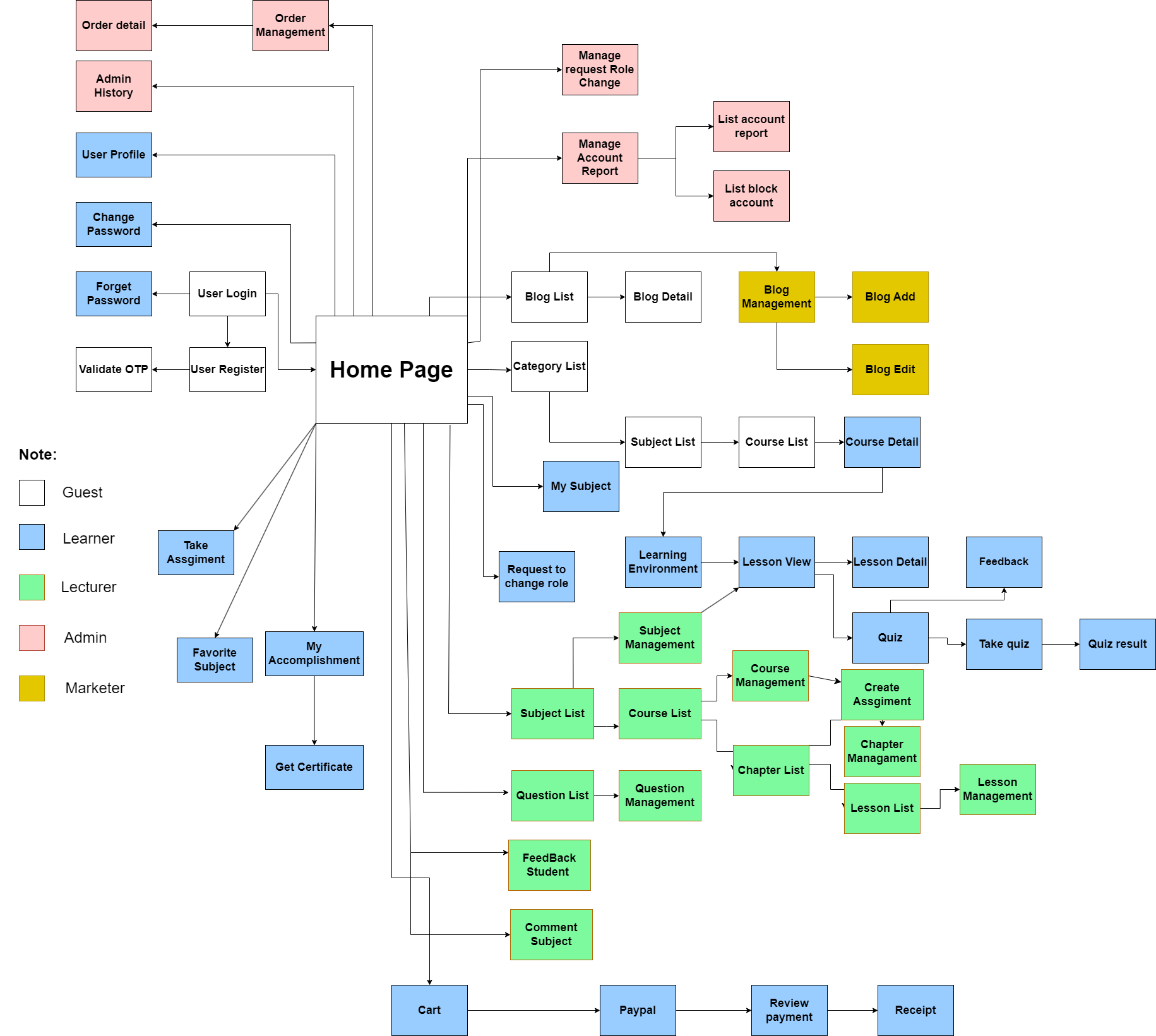


## 3. System Functionalities

*[Provide functionality overview of software system: screen flow, screen descriptions, system user roles, screen authorization, non-screen functions, ERD]*

### 3.1 Screens Flow

*[This part shows the system screens and the relationship among screens. You can draw the Screens Flow for the system in the form of diagram as below.]*



### 3.2 Screen Authorization

*[Provide the system roles authorization to the system features (down to screens, and event to the screen activities if applicable) in the table form as below – replace Role-Name1, Role-Name2,… with your specific system user role names]*

### 

| **Screen** | **Guest** | **Learner** | **Lecturer** | **Marketer** | **Admin** |
| --- | --- | --- | --- | --- | --- |
| User Login |  | x | x | x | x |
| User Register | x | x | x | x | x |
| Change Password |  | x | x | x | x |
| Forget Password |  | x | x | x |  |
| User Profile |  | x | x | x | x |
| Home page | x | x | x | x | x |
| Blog List | x | x | x | x | x |
| Blog Detail | x | x | x | x | x |
| Blog Management |  |  |  | x |  |
| Blog Edit |  |  |  | x |  |
| Blog Add |  |  |  | x |  |
| Category List | x | x | x | x | x |
| Subject List | x | x | x | x | x |
| Course List | x | x | x | x | x |
| Course Detail |  | x | x |  |  |
| Learning Environment |  | x | x |  |  |
| Lesson View |  | x | x |  |  |
| Lesson Detail |  | x | x |  |  |
| Quiz |  | x | x |  |  |
| Quiz Handle |  | x | x |  |  |
| Quiz Review |  | x | x |  |  |
| Get Certificate and Rate Star |  | x |  |  |  |
| My Course Enrollment |  | x |  |  |  |
| Subject List (Lecturer) |  |  | x |  |  |
| Subject Management |  |  | x |  |  |
| Course List (Lecturer) |  |  | x |  |  |
| Course Management |  |  | x |  |  |
| Chapter List (Lecturer) |  |  | x |  |  |
| Chapter Management |  |  | x |  |  |
| Lesson List (Lecturer) |  |  | x |  |  |
| Lesson Management |  |  | x |  |  |
| Question List |  |  | x |  |  |
| Question Management |  |  | x |  |  |
| Cart |  | x |  |  |  |
| Paypal Purchase |  | x |  |  |  |
| Review Payment |  | x |  |  |  |
| Receipt |  | x |  |  |  |
| Manage Account |  |  |  |  | x |
| Manage account Report |  |  |  |  | x |
| List Account Report |  |  |  |  | x |
| List Block Account |  |  |  |  | x |
| Accomplishment |  | x |  |  |  |
| Favorites Subject List |  | x |  |  |  |
| Admin History |  |  |  |  | x |
| Order Management |  |  |  |  | x |
| Order detail |  |  |  |  | x |
| Order history |  | x |  |  | x |
| OrderAnalysis |  |  |  |  | x |
| Chat bot | x | x | x | x | x |

### 3.3 Non-UI Functions

*[Provide the descriptions for the non-screen system functions, i.e batch/cron job, service, API, etc.]*

| **#** | **Feature** | **System Function** | **Description** |
| --- | --- | --- | --- |
| 1 | Authorization Management | Authorization | This function is used to authorize users to access authorized websites |
| 2 | User Authentication | Send OTP(API) | This function sends OTP to the email of registered and unregistered users for confirmation |
| 3 | Timeout | Timeout Function | This function ensures that the OTP code is only valid for a short period of time, and after that period ends, the OTP code will be automatically deleted or no longer accepted. |

# II. Functional Requirements

*[Provide descriptions about the system’s functions/screens. The functions/screens are grouped by the system features, and even sub-features if needed. For the screens, you need to provide the screen layouts (mock-up screens) and relevant specifications if needed]*

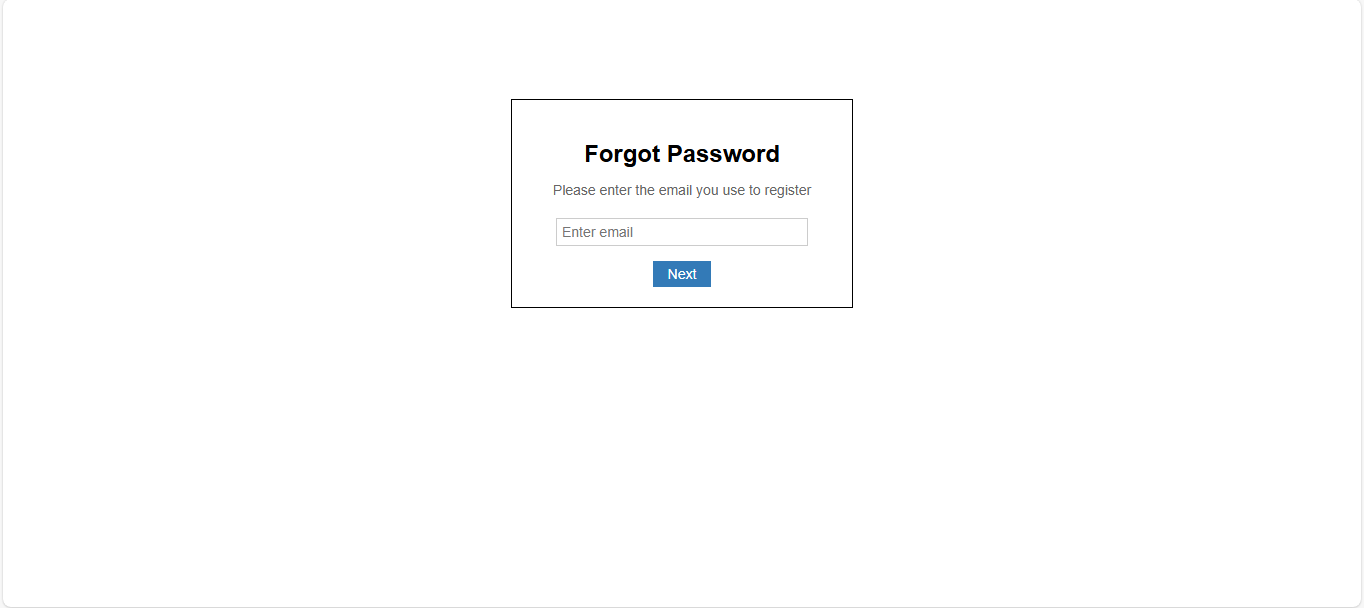
## 1. System User

### 1.1 System Subject

### 1.2 Forget Password

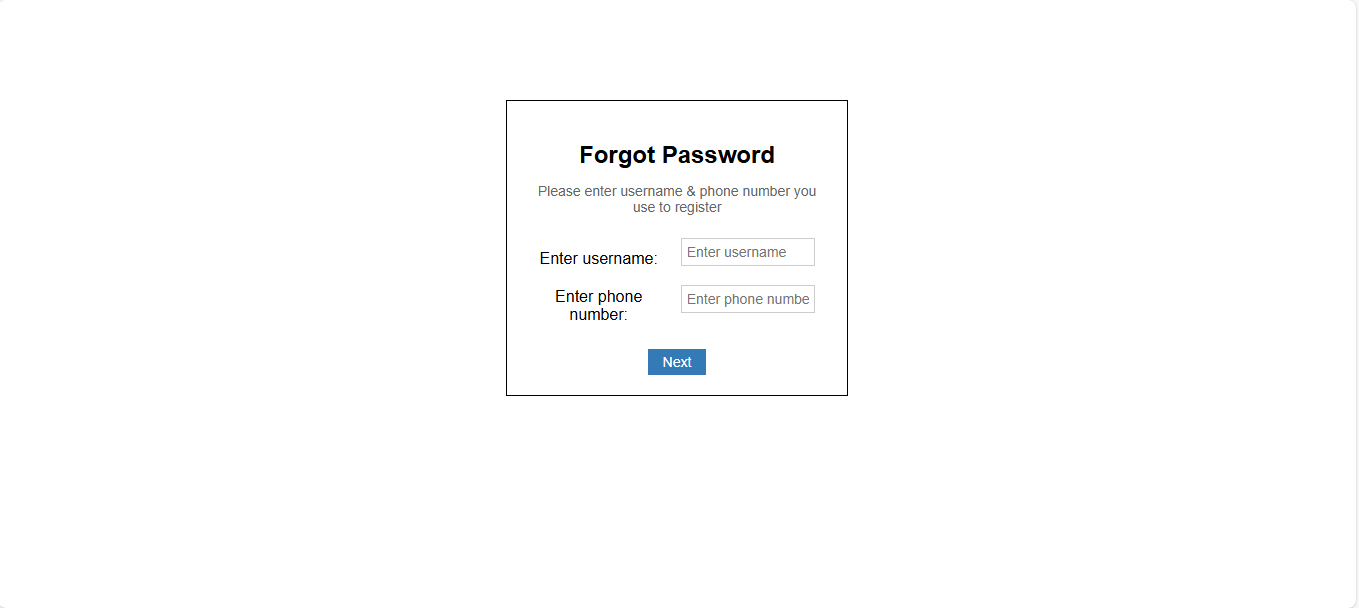
After click ForgetPassword in login page will view this page:

In this page users need enter correct email they use to register. If users enter incorrect email will view error. Users must be re-enter.



When enter correct email address click Next button will view this page:

In this page users must enter username and phone number when they use to register.



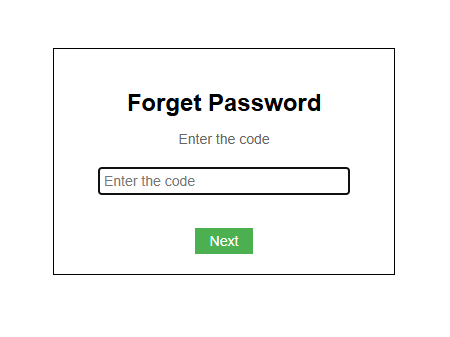
After enter correct information click on Next button will view this page:

This page is where you confirm which email the OTP code will be sent to.

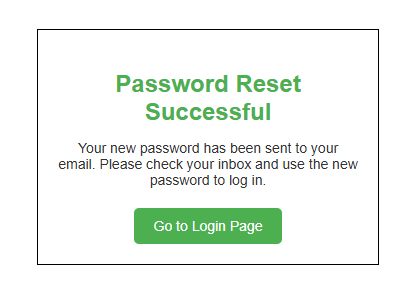


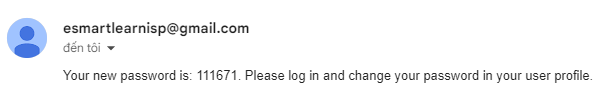
After confirm will send otp to that email, click on Send button will view this page:

In this page must be enter correct otp number send in user mail.



After enter correct otp number click on Next button will view this page:

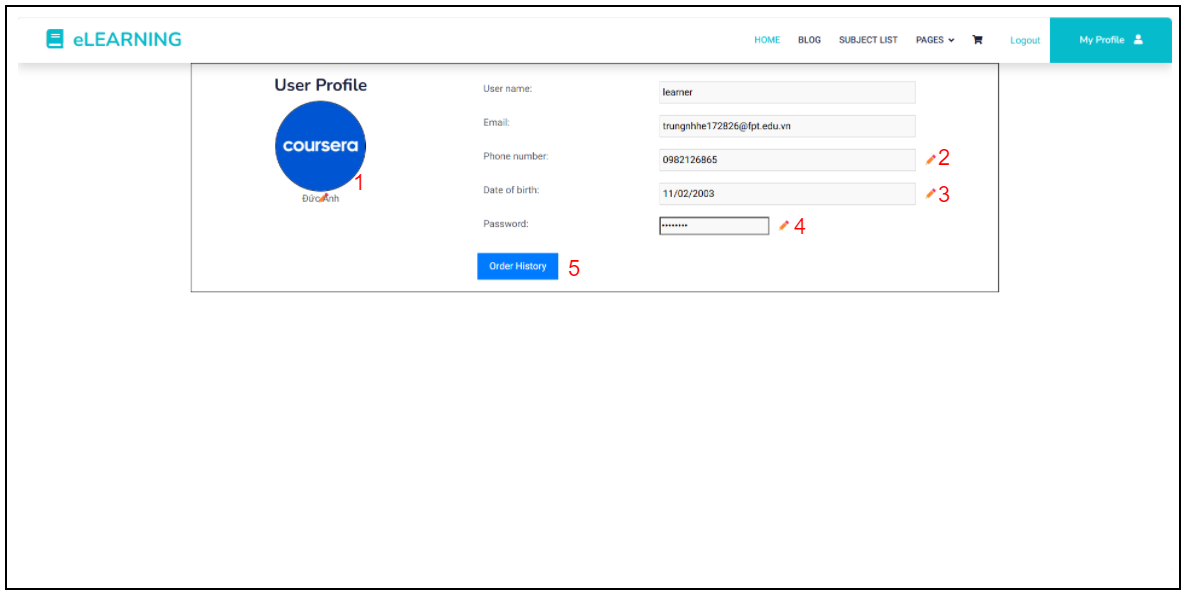


This page view it means new password sended to user mail like this

### 1.3 User profile

In this page usert can see they profile they can change phone number, date of birth, password, update avatar

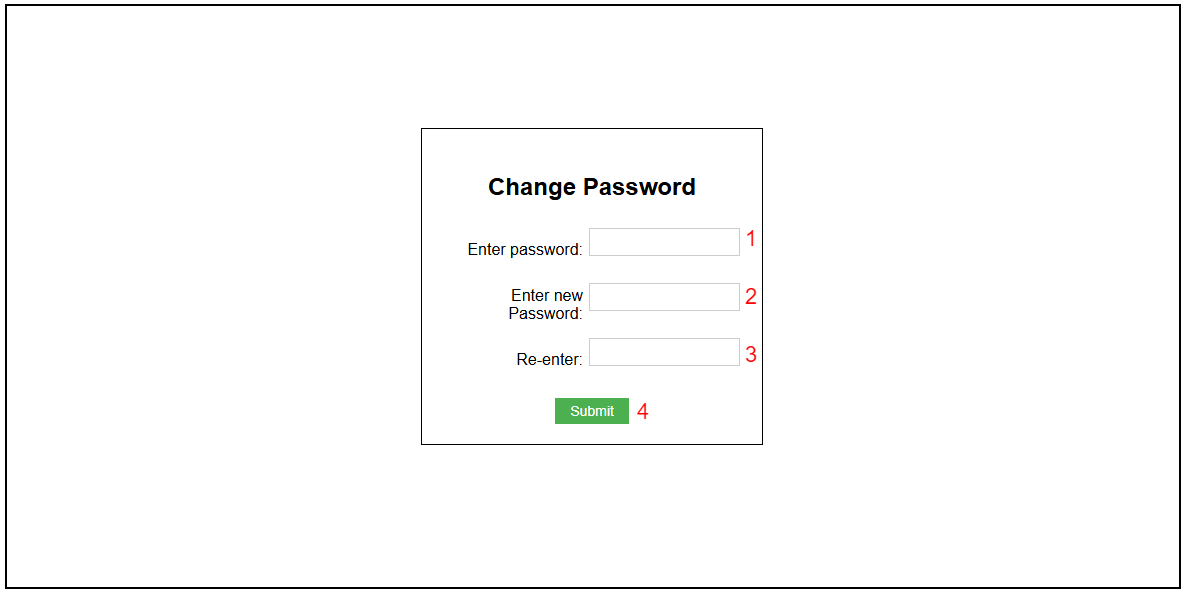
1. Change avatar
2. Change Phone number
3. Change date of birth
4. go to Order History page



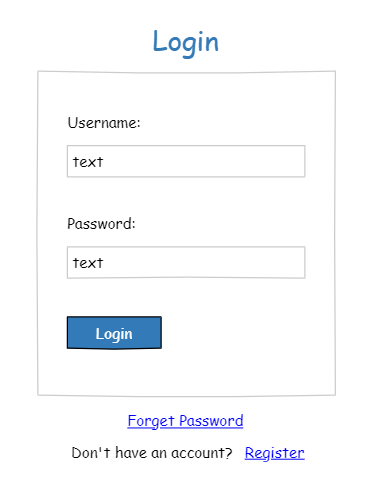
### 1.4 Change password

In this page user must enter they correct old password to update new password

1. Enter current password
2. Enter new password
3. Re-enter new password
4. Submit to save new password

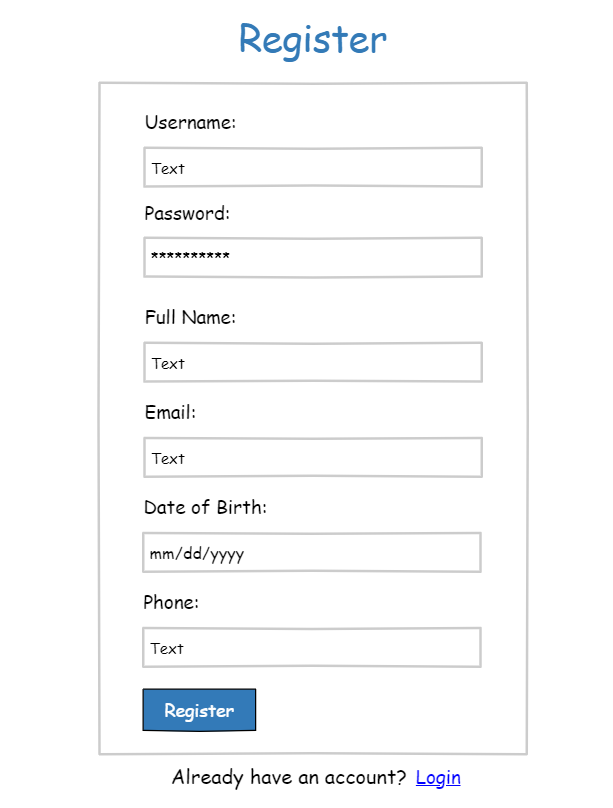


### 1.5 Login



When you access the web, you enter Username or Email, password and click the Login button to join. If you don’t have an account, you have to choose the Register button. If you don't remember your password, you can choose the Forget password button..

### 1.6 Register



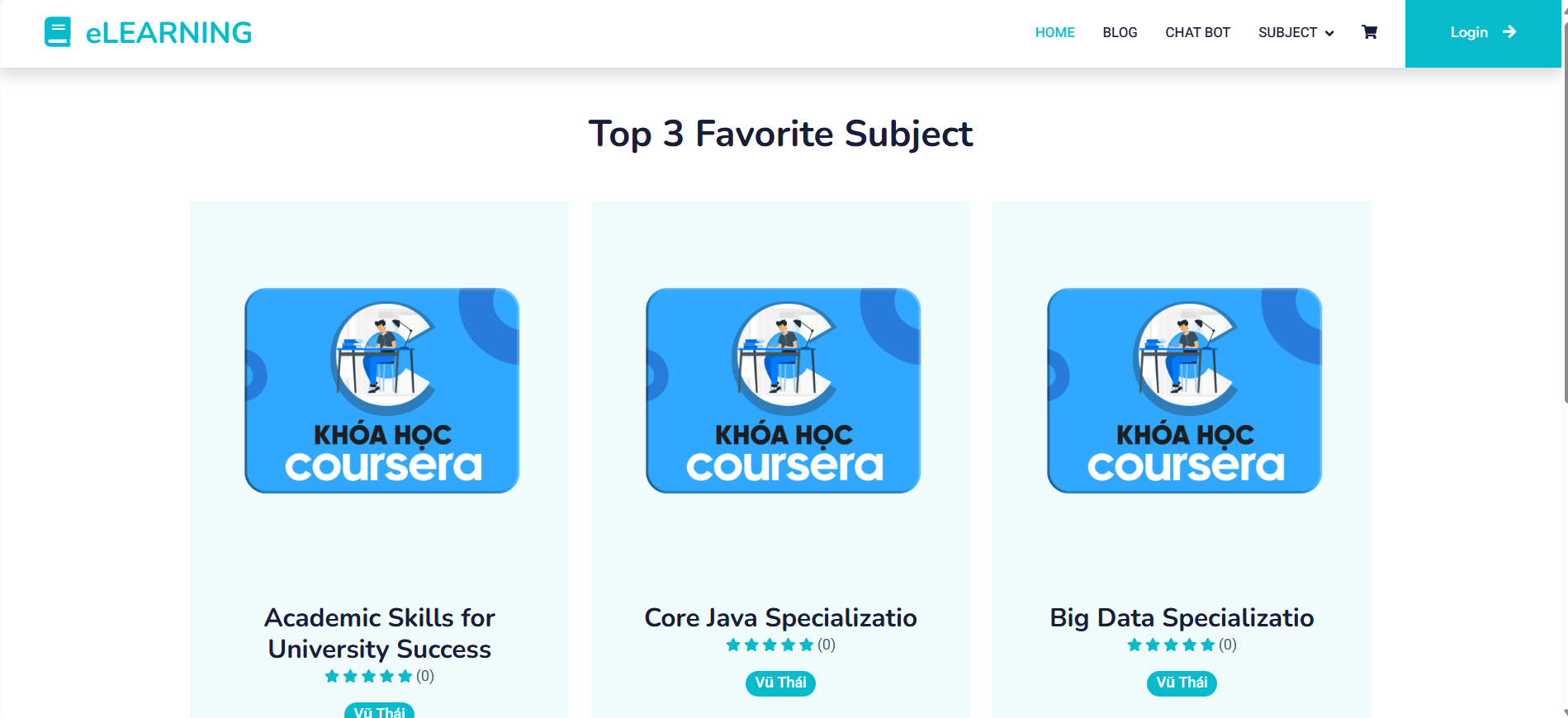
After clicking the Register button on the Login page, this page is shown. You enter your username, password, full name, email address, date of birth, phone number to create a new account with learner role as default.

When you are done, you can click the Register button. Login button used when you already registered.



After entering the Register button, this page is shown. The OTP will send to your email, you must enter the correct otp number sent to create a new account .

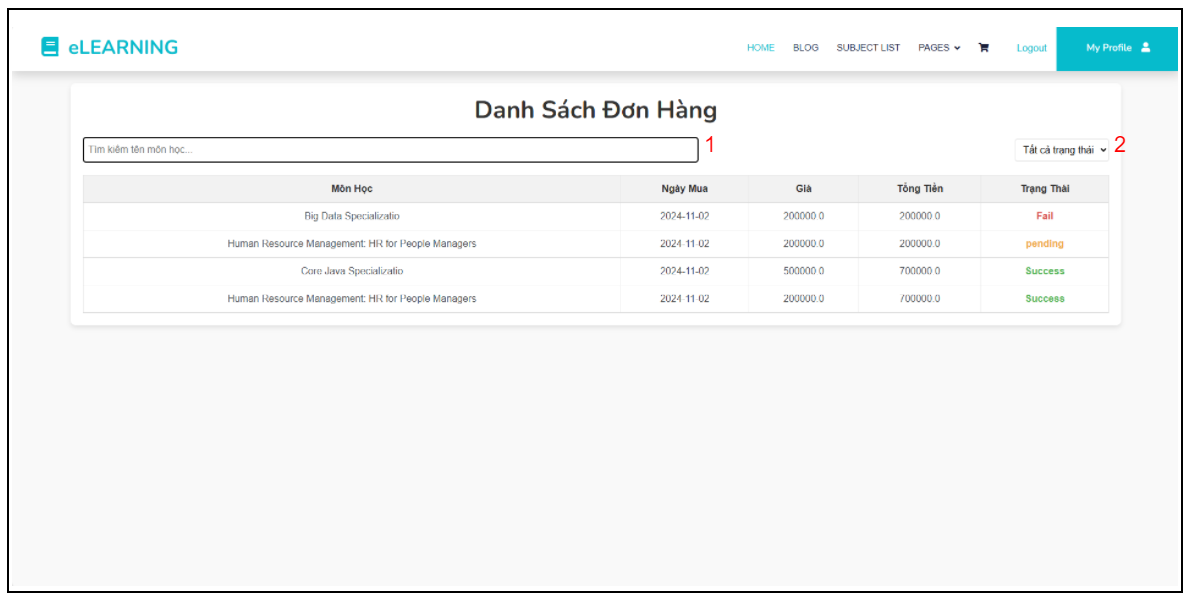
### 1.7 Home page

After running the program, it will go to the home page, the Home page is displayedBasic information of 1 website

### 1.8 Order History

In this page user is student who buy subject they can see they order history

1. To search subject name
2. To search payment status



### 1.9 View Accomplishments :

**Actors:**

* **Primary Actor**: User (Learner)
* **Supporting Actor**: System

**Description:**

This use case describes the process where a user can view their completed accomplishments (courses and certifications) and has the option to share those accomplishments on LinkedIn.

**Preconditions:**

* The user must be logged into the system.
* The user has completed one or more courses or certifications.

**Postconditions:**

* The user views their accomplishments.
* The user may share an accomplishment on LinkedIn.

**Main Success Scenario (Basic Flow):**

1. **User navigates to the Accomplishments page**:  
    The user logs into the system and navigates to the "Your Accomplishments" page.
2. **System retrieves accomplishments**:  
    The system fetches all completed accomplishments from the user’s account and displays them categorized by "Course" and "Subject".
3. **User views accomplishments**:  
    The user can see each accomplishment along with its details (name, description, start and completion dates).
4. **User shares accomplishment on LinkedIn**:  
    The user clicks on the "Share on LinkedIn" button next to an accomplishment.
   * The system redirects the user to LinkedIn with pre-populated accomplishment information for sharing.
5. **User refreshes the accomplishments**:  
    The user clicks on the "Refresh Accomplishments" button to reload and update the list of accomplishments.

**Extensions (Alternate Flows):**

3a. **No accomplishments available**:

* If the user has no completed courses or certifications, the system displays a message like "No accomplishments yet" instead of the accomplishments list.

4a. **LinkedIn sharing fails**:

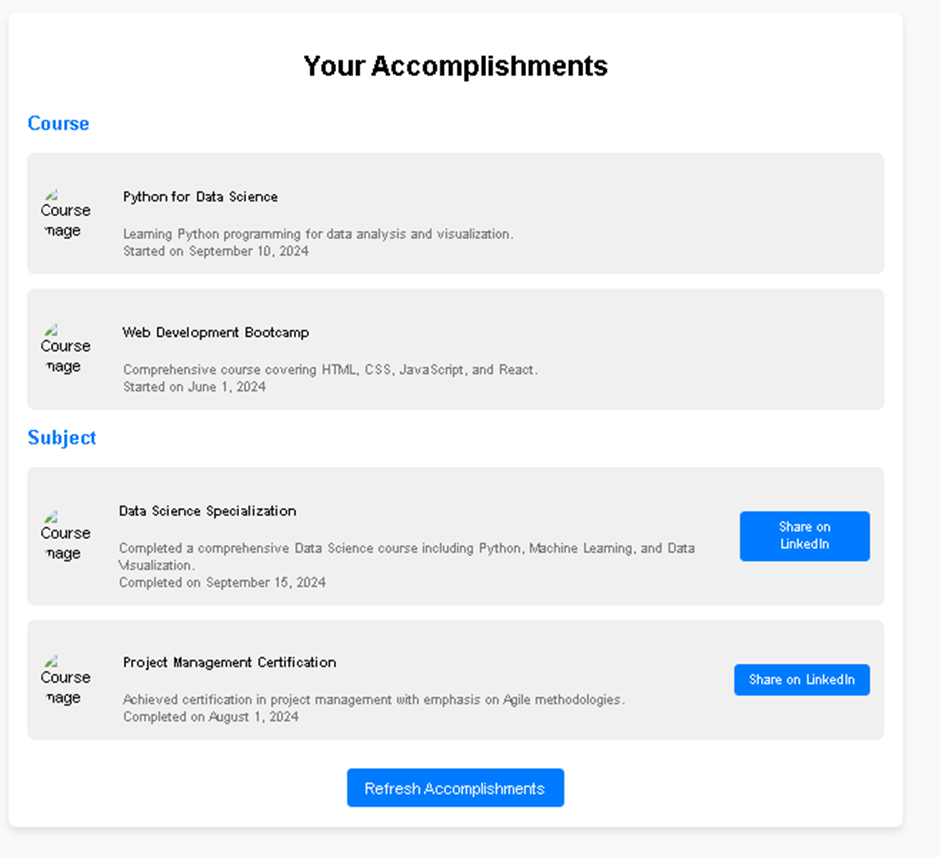
* If there is a problem with the LinkedIn integration, the system shows an error message like "Unable to connect to LinkedIn. Please try again later."

**Special Requirements:**

* The system must integrate with LinkedIn’s API for sharing accomplishments.
* The interface should be user-friendly and responsive across all devices (desktop and mobile).

**Assumptions:**

* The user has an account with the platform and has completed at least one course or certification.
* The user has a LinkedIn account and wishes to share accomplishments on that platform.



#### 1.9.1 Display Certificate of Completion :

**Actors:**

* **Primary Actor**: User (Learner)
* **Supporting Actor**: System

**Description:**

This use case describes the process where the system displays a certificate of completion to the user after successfully finishing a course. The certificate includes details such as the learner's name, course name, completion date, and signature of the instructor.

**Preconditions:**

* The user must be logged into the system.
* The user has completed a course or certification.
* The system has the user’s information (name, course details, and completion date).

**Postconditions:**

* The user can view their certificate of completion.
* The user may download or print the certificate.

**Main Success Scenario (Basic Flow):**

1. **User navigates to the Certificate page**:  
    The user logs into the system and navigates to the "Certificate" section for a specific course they have completed.
2. **System retrieves certificate details**:  
    The system fetches the user’s name, course name, completion date, instructor's signature, and logo from the database.
3. **System displays certificate**:  
    The system generates the certificate in a visually appealing format, showing the following details:
   * **User's Name**: Displayed prominently in large text (centered).
   * **Course Name**: Displayed in italic, styled text.
   * **Completion Date**: Shown clearly beneath the course name.
   * **Instructor's Signature**: Positioned at the bottom-right, alongside the instructor's name and title.
   * **Platform Logo**: Positioned at the top-left of the certificate.
4. **User views certificate**:  
    The user can view the certificate with all details properly aligned and formatted.
5. **Optional actions**:  
    The user may be provided with options to:
   * **Download**: A button to download the certificate as a PDF file.
   * **Print**: A button to print the certificate.

**Extensions (Alternate Flows):**

3a. **No certificate available**:

* If the user hasn’t completed the course, the system displays a message like "No certificate available" instead of the certificate details.

4a. **Customizing Certificate**:

* If the platform supports certificate customization (e.g., adding a different language or style), the user can select a customization option before viewing or downloading the certificate.

5a. **Download/Print error**:

* If there’s an issue downloading or printing, the system shows an error message like "Unable to download/print certificate. Please try again later."

**Special Requirements:**

* The system should support high-resolution certificate generation for clear printing.
* The certificate must be responsive and compatible with both mobile and desktop devices.

**Assumptions:**

* The user has completed at least one course that provides a certificate of completion.
* The certificate details (name, course, date, and signature) are stored in the system.



#### 1.9.2 **Lesson View (User)**

**. Use Case Name:**

* Lesson View (User)

**2. Brief Description:**

* This use case allows users to view the details of a specific lesson, including its content (e.g., text, videos, links), and track their progress through the lesson. The user may also be able to view related quizzes or assignments if they are part of the lesson.

**3. Actors:**

* **User (Student)**: The person who views the lessons.
* **System**: The system that manages and displays the lesson content and tracks user progress.

**4. Pre-conditions:**

* The user is logged into the system.
* The user is enrolled in the course or lesson.
* Lessons have been set up and are available for viewing.

**5. Post-conditions (Success):**

* The user successfully views the lesson content.
* The system updates the user's progress for the lesson.

**6. Post-conditions (Failure):**

* The user is unable to view the lesson due to missing content or system issues.
* The system fails to record the user's progress.

**7. Main Flow of Events:**

1. **User** logs into the system and navigates to the course containing the lesson they want to view.
2. **System** displays the list of lessons for the selected course.
3. **User** selects a lesson from the list.
4. **System** retrieves and displays the lesson content, including text, videos, quizzes, and any other relevant material.
5. **User** interacts with the lesson by reading, watching videos, or completing tasks within the lesson.
6. **System** tracks and updates the user's progress (e.g., percentage of the lesson completed).
7. **User** completes the lesson or exits, and the **System** records the current progress and marks it as "completed" if applicable.

**8. Alternative Flow:**

* **Step 4a**: If the lesson content is unavailable or cannot be loaded, the **System** will display an error message and prompt the user to try again later.
* **Step 6a**: If the user's progress cannot be saved due to a system issue, the **System** will notify the user and ask them to retry the lesson at a later time.

**9. Data Requirements:**

* **Lesson Table**: Stores lesson content details such as the title, content, duration, and links to media.
* **User Table**: Stores information about the user to track their progress and interaction.
* **Lesson Progress Table**: Tracks user progress through the lesson (e.g., percentage completed, time spent, last accessed).
* **Course Table** (if lessons are part of a course): Links lessons to their respective courses.
* **Quiz Table** (if applicable): Contains quiz questions related to the lesson.
* **Assignment Table** (if applicable): Stores assignment information related to the lesson.

**10. Business Rules:**

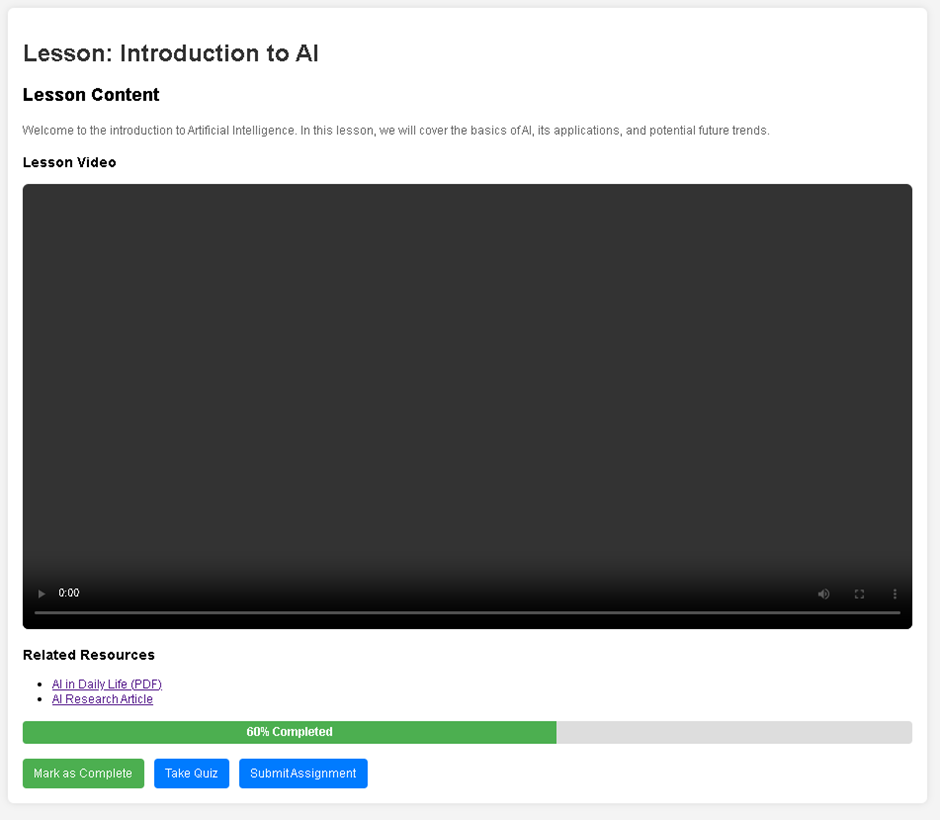
* Users can only view lessons if they are enrolled in the course.
* Lessons should be tracked, and the system must record the user's progress.
* The system should allow resumption of lessons from the last saved point.

**11. Extension Points:**

* **Quizzes**: Users may be able to take a quiz related to the lesson at the end.
* **Assignments**: Users may be prompted to submit assignments after viewing the lesson.

**12. Special Requirements:**

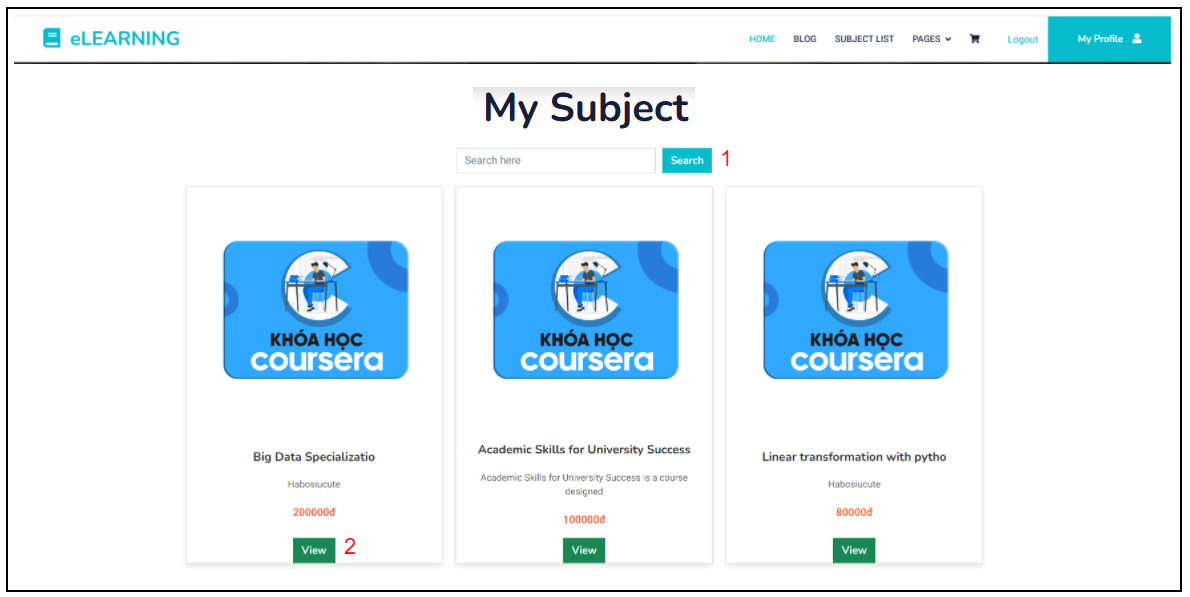
* Responsive user interface for accessing lessons on various devices (e.g., mobile, desktop).
* Offline access might be required for viewing lessons without an internet connection (if applicable).



### 1.10 My Subject

In this page, when student when they buy subject in this page will view that subject in here

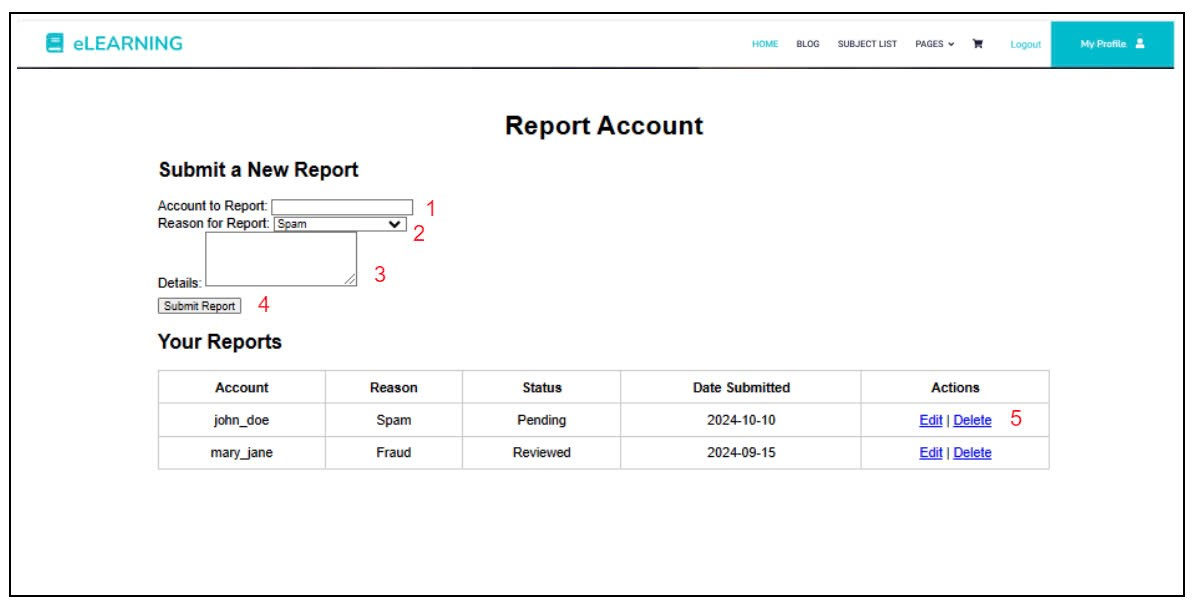
1. To search subject name
2. Go to subject information page



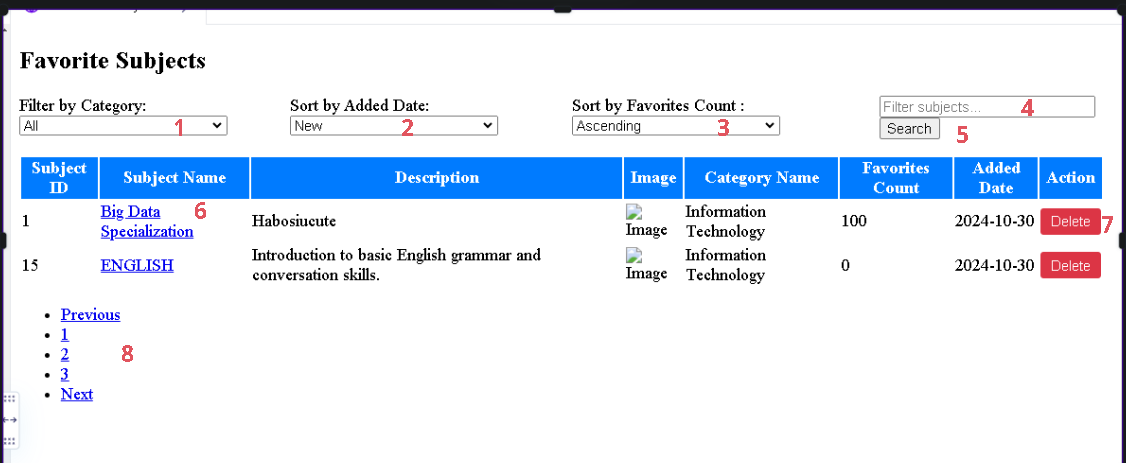
### 1.11 Report Account

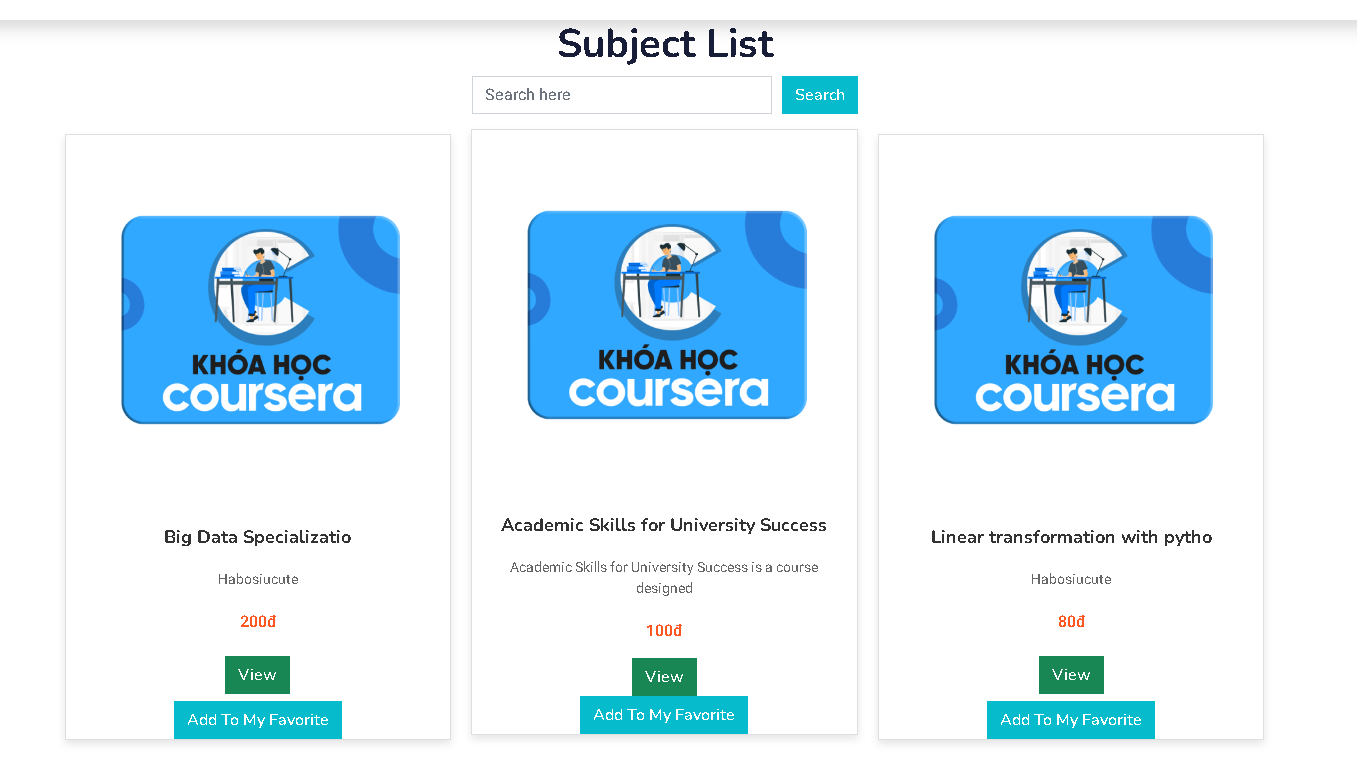
In this page, User can send to admin if they want to report someone.

1. Enter account name you want to report
2. Choose reasons you why you want to report other people
3. Enter state specifically why
4. Send report to admin
5. Edit that report if status = pending



### 1.12 Favorites Subject List





**Filter by Category**: A dropdown menu that allows users to filter subjects by category, such as "Information Technology". When a specific category is selected, only subjects in that category will be displayed.

**Sort by Added Date**: A dropdown menu that lets users sort the subject list by the date they were added. Users can choose "New" (sorting from newest to oldest) or "Old" (sorting from oldest to newest).

**Sort by Favorites Count**: A dropdown menu that enables users to sort the list based on the favorites count. Sorting can be in ascending (from fewest to most favorites) or descending (from most to fewest favorites) order.

**Filter subjects**: A search box where users can enter keywords to filter the subject list. Only subjects containing the keyword in the name or description will be displayed.

**Add favorite subject :** A User click add form subject lists to add favorite list of them.

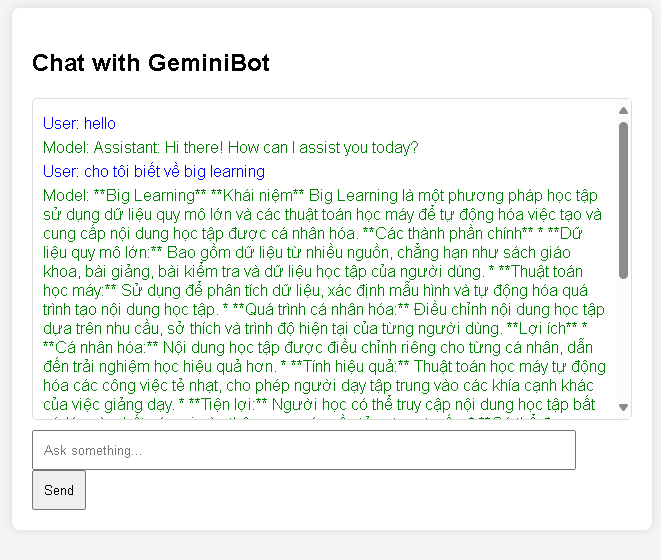
**Search Button**: The "Search" button allows users to execute a search after entering a keyword in the search box. When this button is clicked, the list is filtered based on the entered keyword.

**Subject Name Links**: Links on the subject names (e.g., "Big Data Specialization", "ENGLISH") allow users to click and view more details about the subject or navigate to a page with detailed information about it.

**Delete Button**: The red "Delete" button allows users to remove a subject from their favorites list. When clicked, the subject is removed from the table and no longer displayed in the interface.

**Pagination (Previous, Next, 1, 2, 3)**: The pagination section allows users to navigate between pages of the subject list. "Previous" and "Next" move to the previous and next pages, while page numbers allow users to jump directly to a specific page.

### 1.13 Chat Bot



Chat bot is created by use gemini api key, help learner find more infomation fast

## 

### 

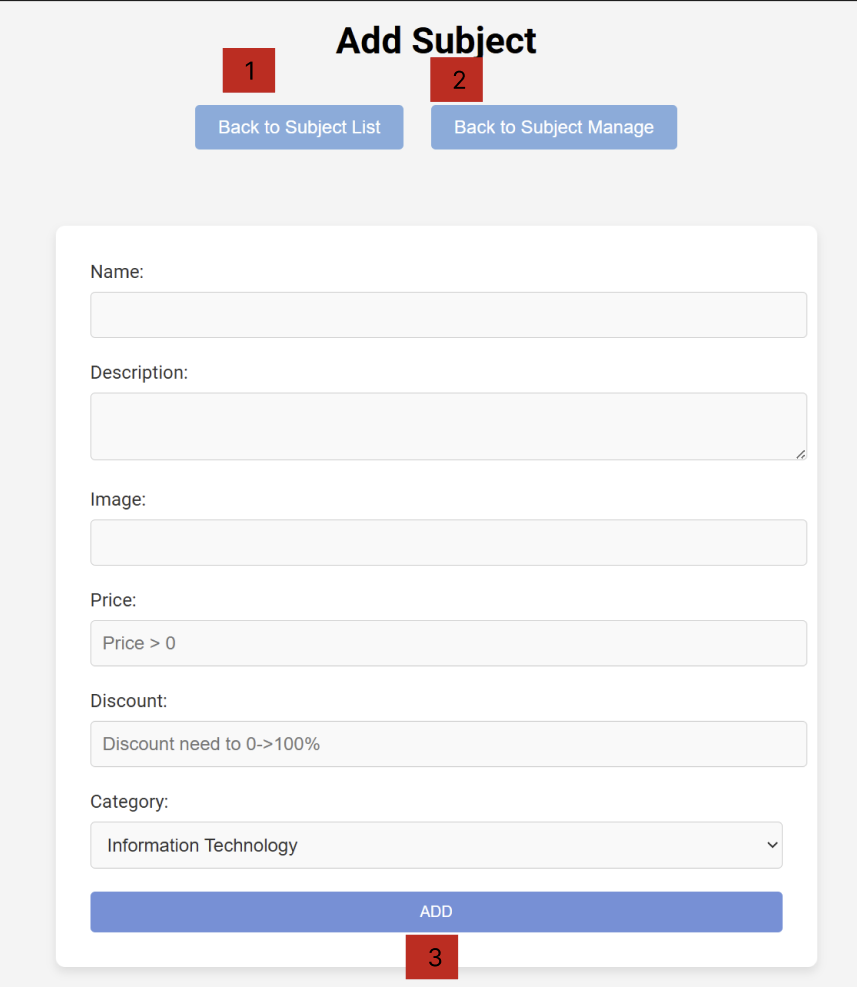
## 2. System Lecturer

### 2.1 System Subject

#### 2.1.1 Add Subject

After clicking on the Add subject button on the right corner of the Topic Management page, this page will be displayed.

This is for Lecturers to add new subject to the system



(1)Back to subject list:

* Allow users to return to the subject list page

(2) Back to subject management:

* Allow users to return to the subject manage page

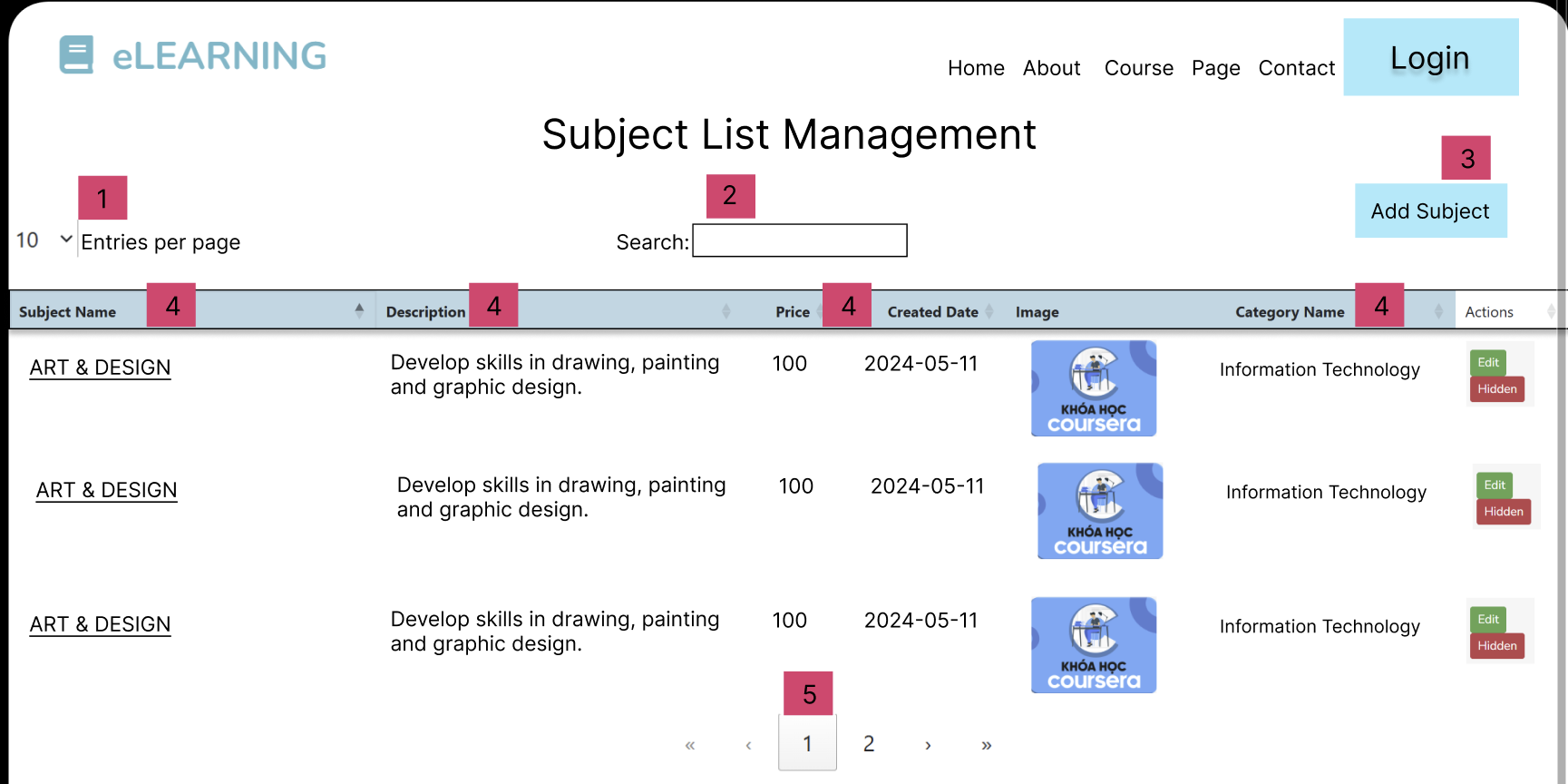
(3) Add:

* This button allows the user to add a new subject with the information fields filled in above.

#### 2.1.2 Subject Manage

After clicking on the subject manage button on the home page, this page will be displayed.

This page helps lecturers to manage their subjects and can add, edit and delete subjects



(1)Entries per page:

* Allows users to adjust the number of posts displayed on a page

(2)Search:

* Allows users to search for information by name, price, date, category name

(3)Add subject:

* Allows users to navigate to the subject page

(4)Subject name,description, price, date, category name :

* Allows display in ascending order ,a-z

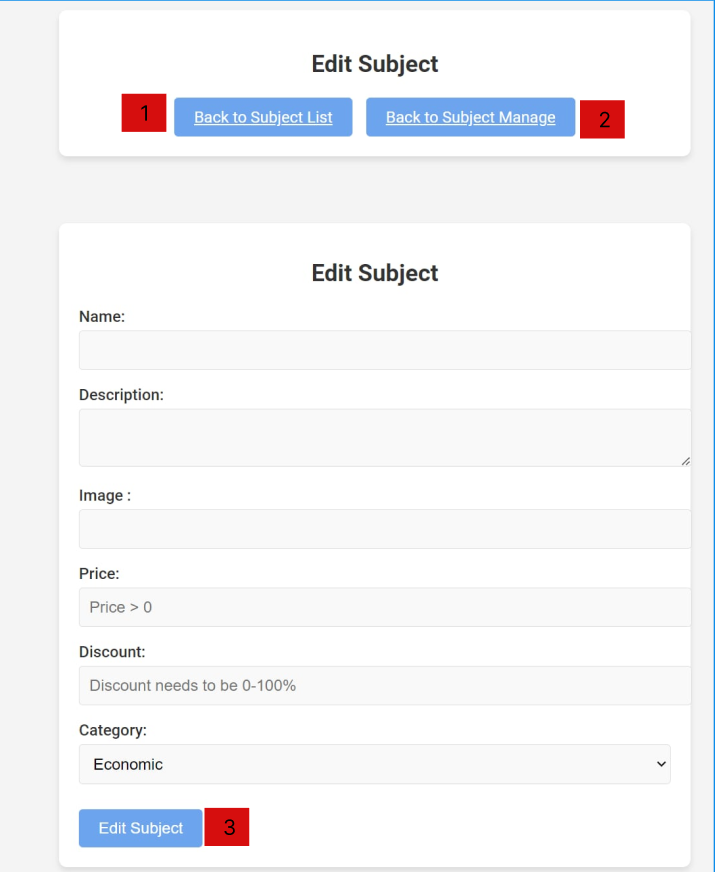
(5)Number page:

* Allows users to go to the next page or return to the last page opened

#### 2.1.3 Edit Subject

After clicking on the Add subject button on the right corner of the Topic Management page, this page will be displayed.

This is for Lecturers to edit subject to the system



(1)Back to subject list:

* Allow users to return to the subject list page

(2) Back to subject management:

* Allow users to return to the subject manage page

(3) Edit subject:

* This button allows the user to edit subject with the information fields filled in above.

#### 2.1.4 Subject Review

#### 

1. Comment Student:

This button allows the lecturer View and respond to student comment.

1. Search:

Lecturer can find subject by name

#### 2.1.5 Hiden Subject

This feature will be activated when pressing the button in the manage page. This feature helps Lecturer to hide subjects if there are any edits.

#### 

#### OK:

This button allows to hide the subject.

#### Cancel:

This button allows the lecturer can cancel hidden subject .

#### 2.1.6 Hiden Subject

This feature will be activated when pressing the button in the manage page. This feature helps Lecturer to hide subjects if there are any edits.

#### 

#### OK:

This button allows to hide the subject.

#### Cancel:

This button allows the lecturer can cancel hidden subject .

### 2.2 System Course

#### 2.2.1 List Course

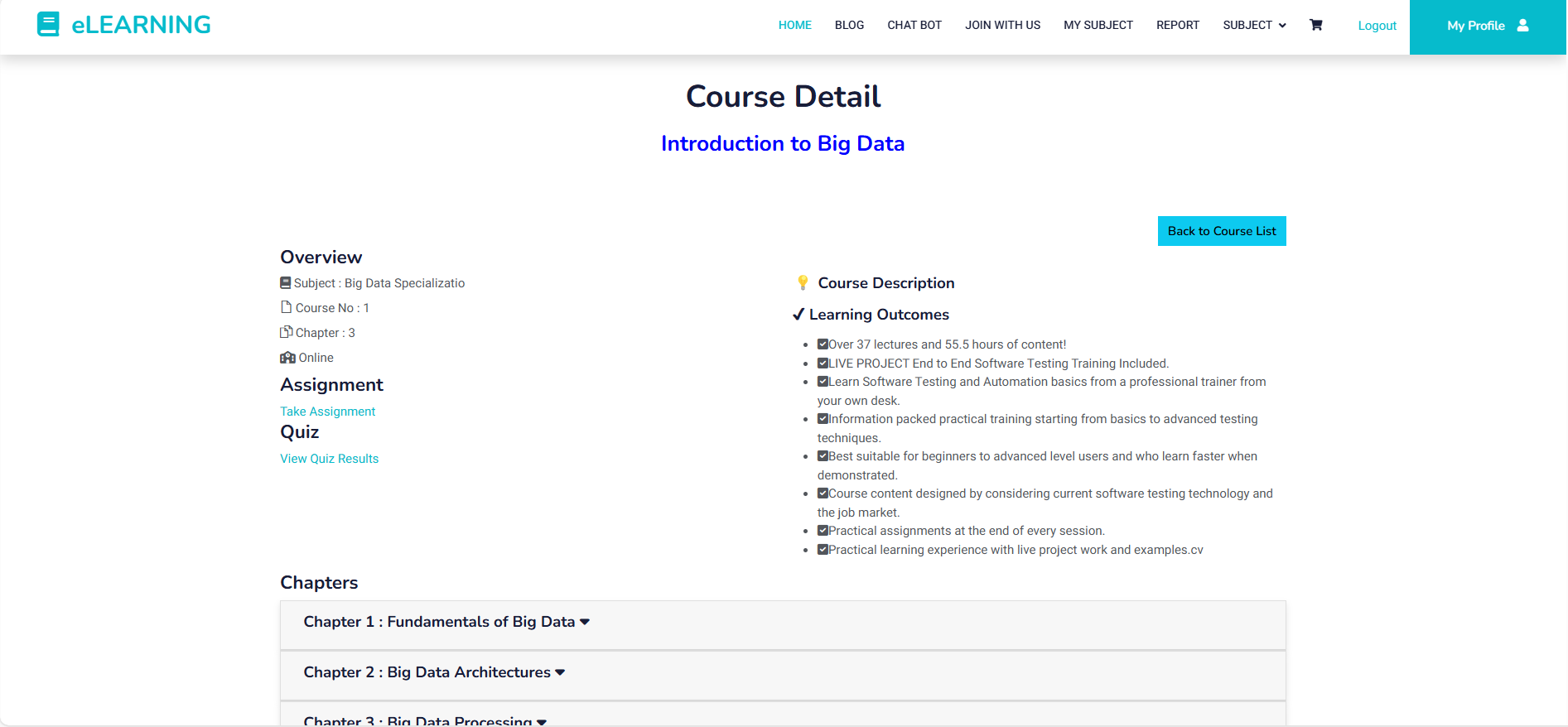
#### After clicking Course on the toolbar, you will be taken to the course list page. This page displays the basic information of the course and can be read more through the read more button ...

#### 

#### Join now: Learner can see all information about course after buy subject

#### 2.2.2 Course detail

After selecting Join Now, the Course Detail page will appear showing the details of the course

1) Take assignment:   
 learner can take assignment about thís course.

2) Chapter:  
 Learners can access chapters that are displayed inside

#### 2.2.3 Management course

#### 

1. Button 1 :

This button allows the lecturer can create new course from 1 subject .

1. Button 2:

This button allows the lecturer can back to management subject .

1. Button 3:

This button allow the lecturer update information in this course.

1. Button 4:

This button allow the lecturer delete course.

1. Button 5:

This button allows the lecturer to assign assignments to students.

#### 2.2.4 Add course

#### 

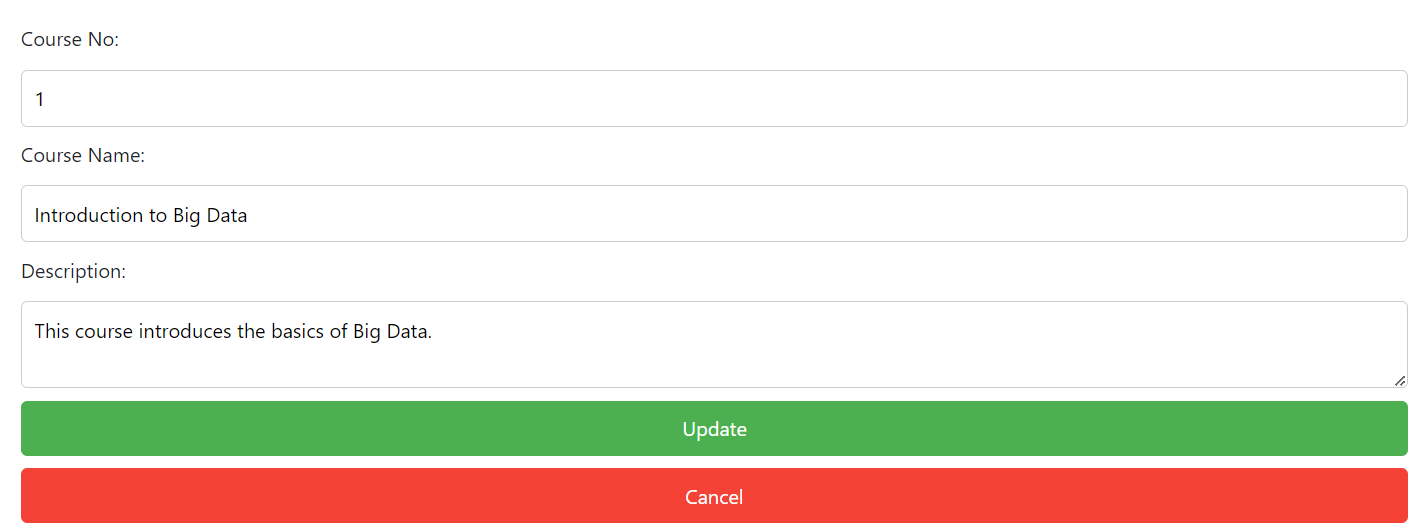
#### Add :

This button allows the lecturer can create new course .

#### Cancel:

This button allows the lecturer can cancel create new course .

#### 2.2.5 Update course



#### update:

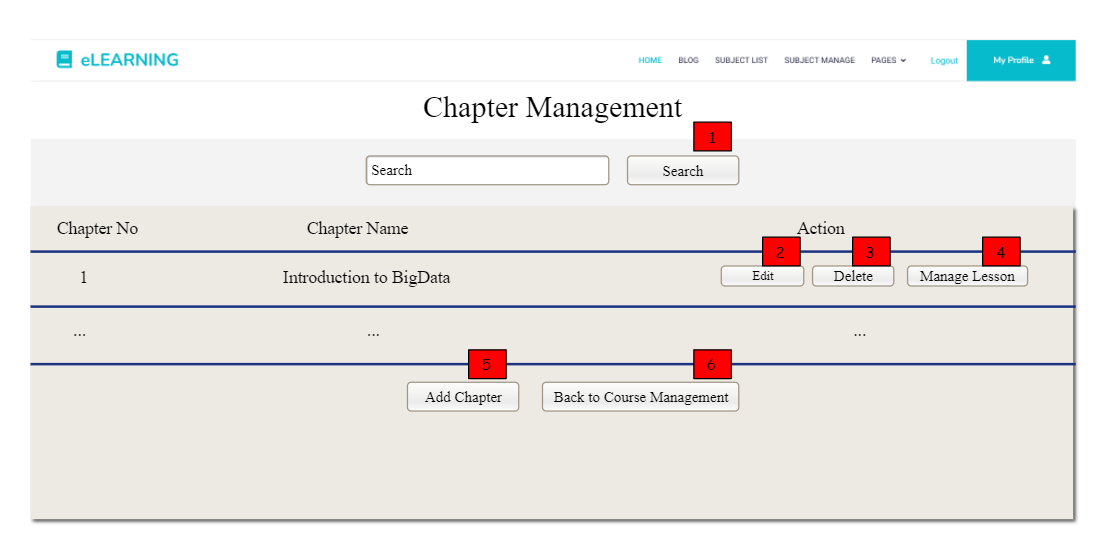
This button allows the lecturer can update information from course .

#### Cancel:

This button allows the lecturer can cancel update course .

### 2.3 System Chapter

#### 2.3.1 Chapter Manage



1. Search

Allows users to search for information by name.

1. Edit Chapter

This button allows the user to edit a chapter with the information fields filled in above.

1. Delete Chapter

This button allows the user to delete a chapter.

1. Manage Lesson

This button will show the lesson in that chapter.

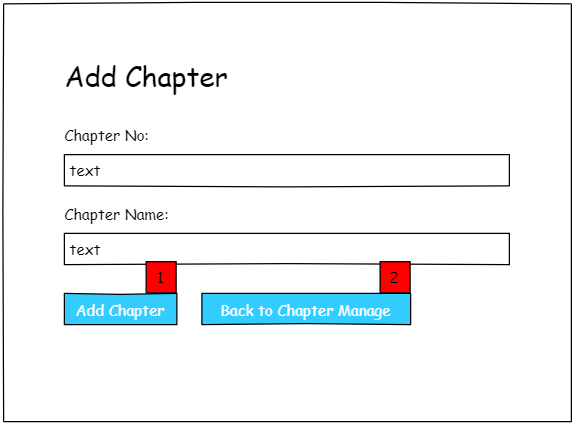
1. Add Chapter

This button allows the user to add a new chapter.

1. Back to Course Management

This button allows the user to go back to the course management page.

#### 2.3.2 Add Chapter



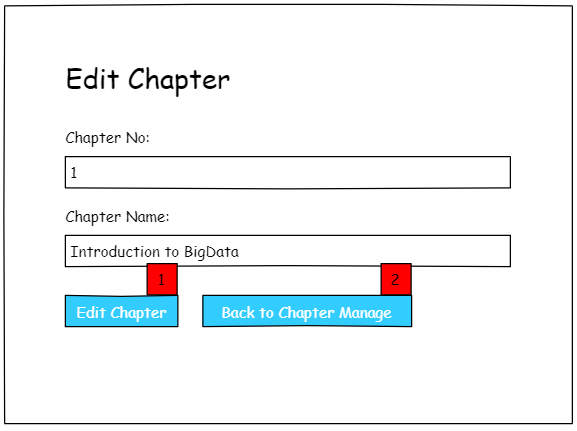
1. Add Chapter

This button allows the user to add a new chapter with the information fields filled in above.

1. Back to Chapter Management

This button allows the user to go back to the chapter management page.

#### 2.3.3 Edit Chapter



1. Edit Chapter:

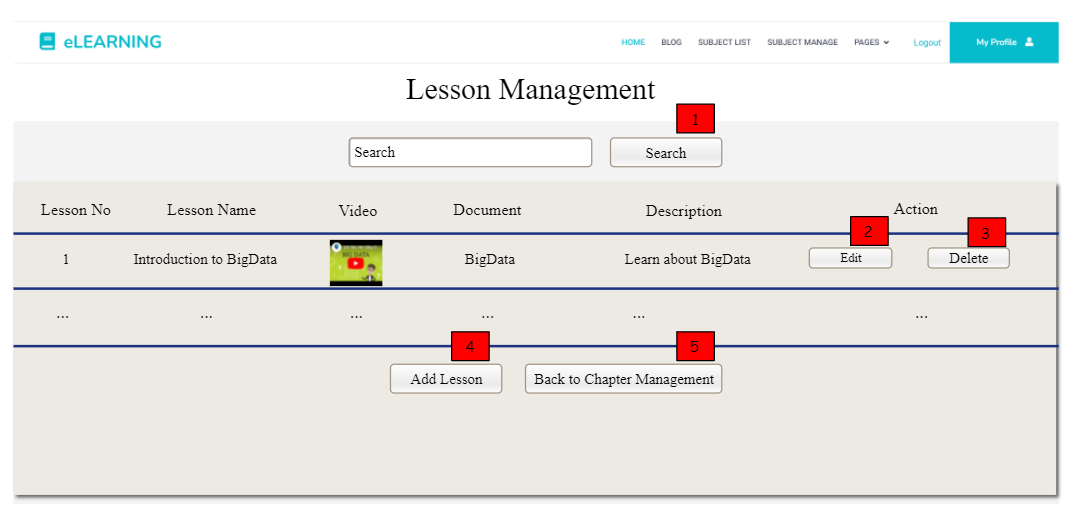
This button allows the user to edit chapter with the information fields filled in above.

1. Back to Chapter Management

This button allows the user to go back to the chapter management page.

### 2.4 System Lesson

#### 2.4.1 Lesson Manage



1. Search

Allows users to search for information by name.

1. Edit Lesson

This button allows the user to edit a lesson with the information fields filled in above.

1. Delete Lesson

This button allows the user to delete a lesson.

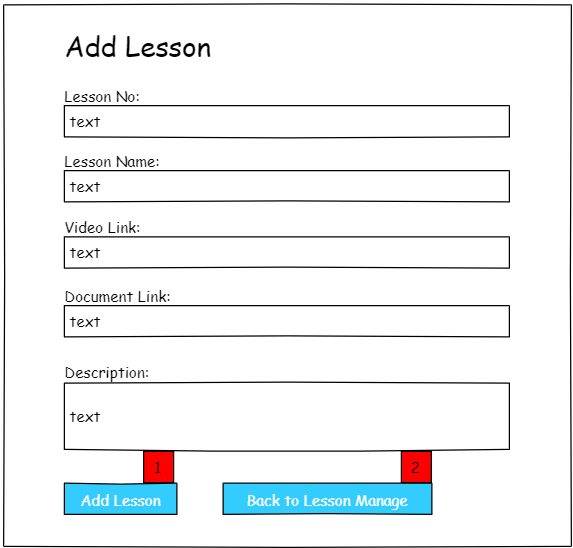
1. Add Lesson

This button allows the user to add a new lesson.

1. Back to Chapter Management

This button allows the user to go back to the chapter management page.

#### 2.4.2 Add Lesson



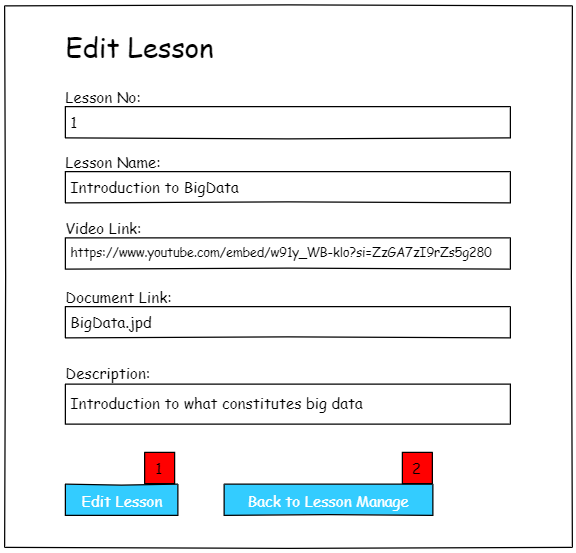
1. Add Lesson

This button allows the user to add a new lesson with the information fields filled in above.

1. Back to Chapter Management

This button allows the user to go back to the lesson management page.

#### 2.4.3 Edit Lesson



1. Edit Lesson:

This button allows the user to edit lesson with the information fields filled in above.

1. Back to Lesson Management

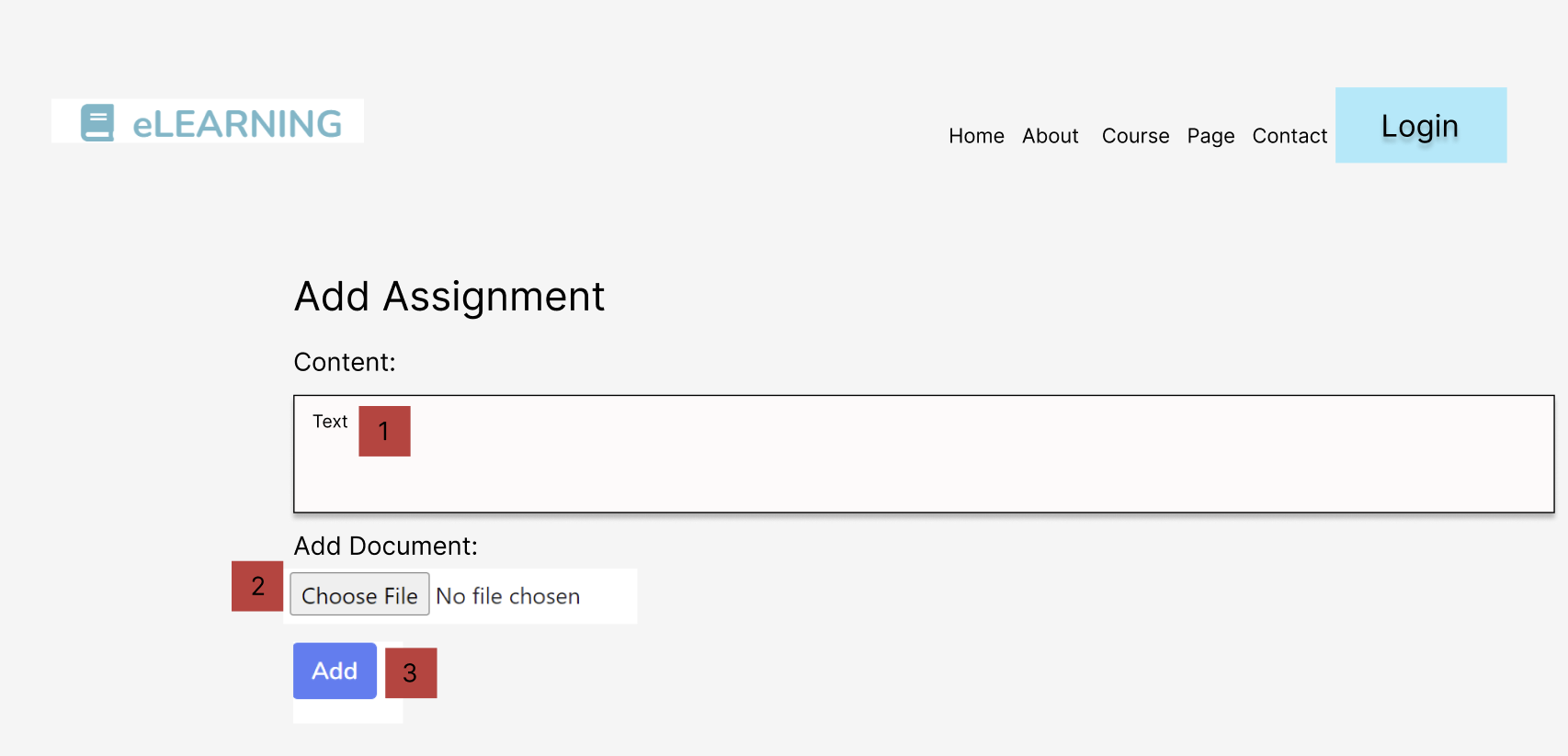
This button allows the user to go back to the lesson management page.

### 2.5 System Assignment

#### 2.5.1 Add Assignment

After you go to the course manage of a subject manage and click the add button, you will be redirected to the add assignment page.

This site can help you add exercises by uploading documents and writing content.



(1)Text:

* Allows users to enter question content

(2)Choose File:

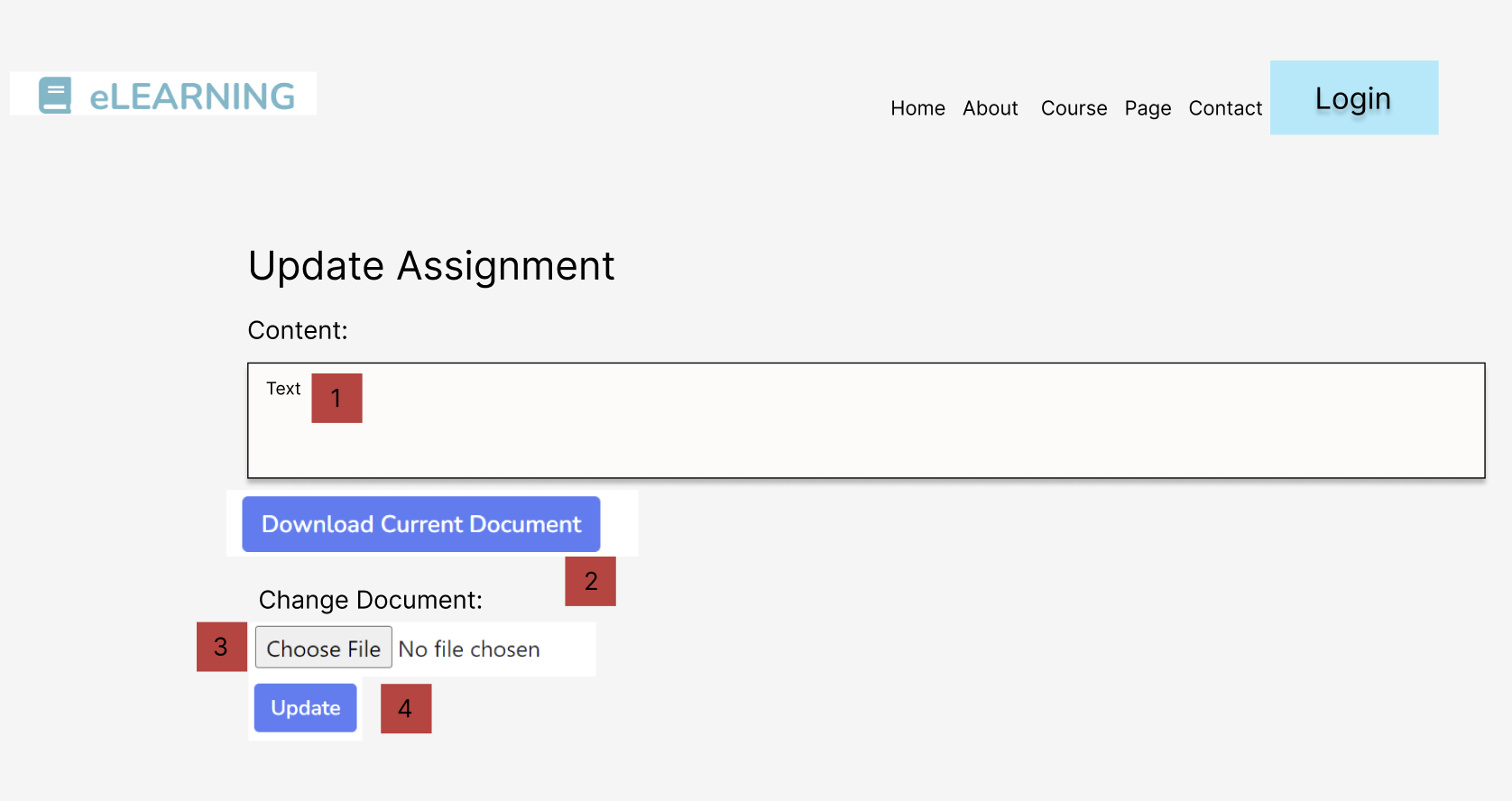
* Allows users to upload document files from their computers

(3)Add:

* Allow lecturer to add assignment to the system

#### 2.5.2 Update Assignment

Once you go to the course management of a subject and click the add button, you will be redirected to the update existing assignment page. This site can help you update your assignment by editing the assignment information.



(1)Text:

* Allow users to edit question content

(2)Download Current Document:

* Allow users to download uploaded documents

(3)Choose File:

* Allows users to upload document files update from their computers

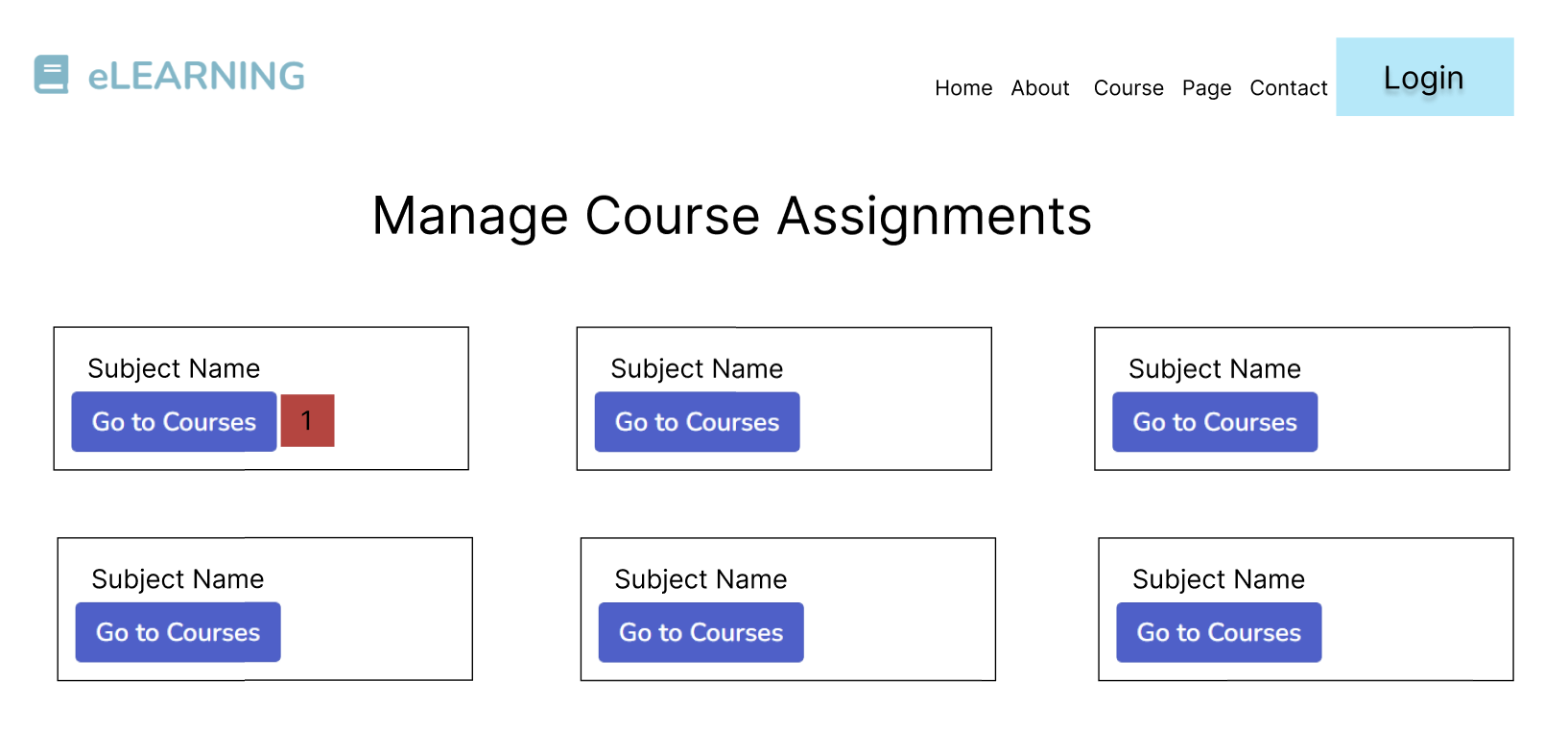
(4)Update:

* Allow lecturer to update assignment to the system

#### 2.5.3 List Cousre Assignment

Lecturer can access this website by clicking on the subject management page. This website will display the subject list from which lecturer can access the courses and view student work for grading.

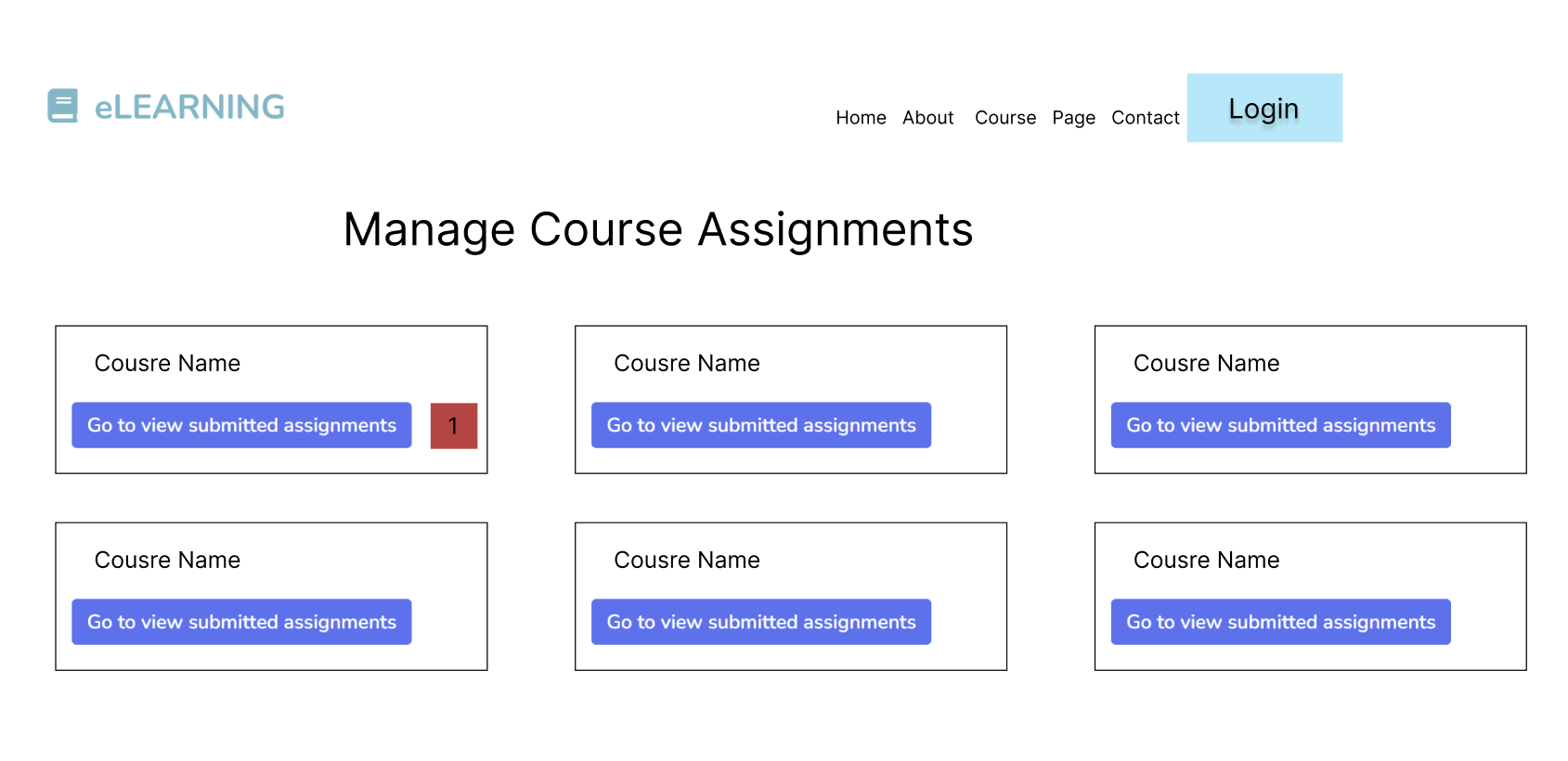
**Page (1)**



(1)Go to Cousre:

* Access the course contained in the subject

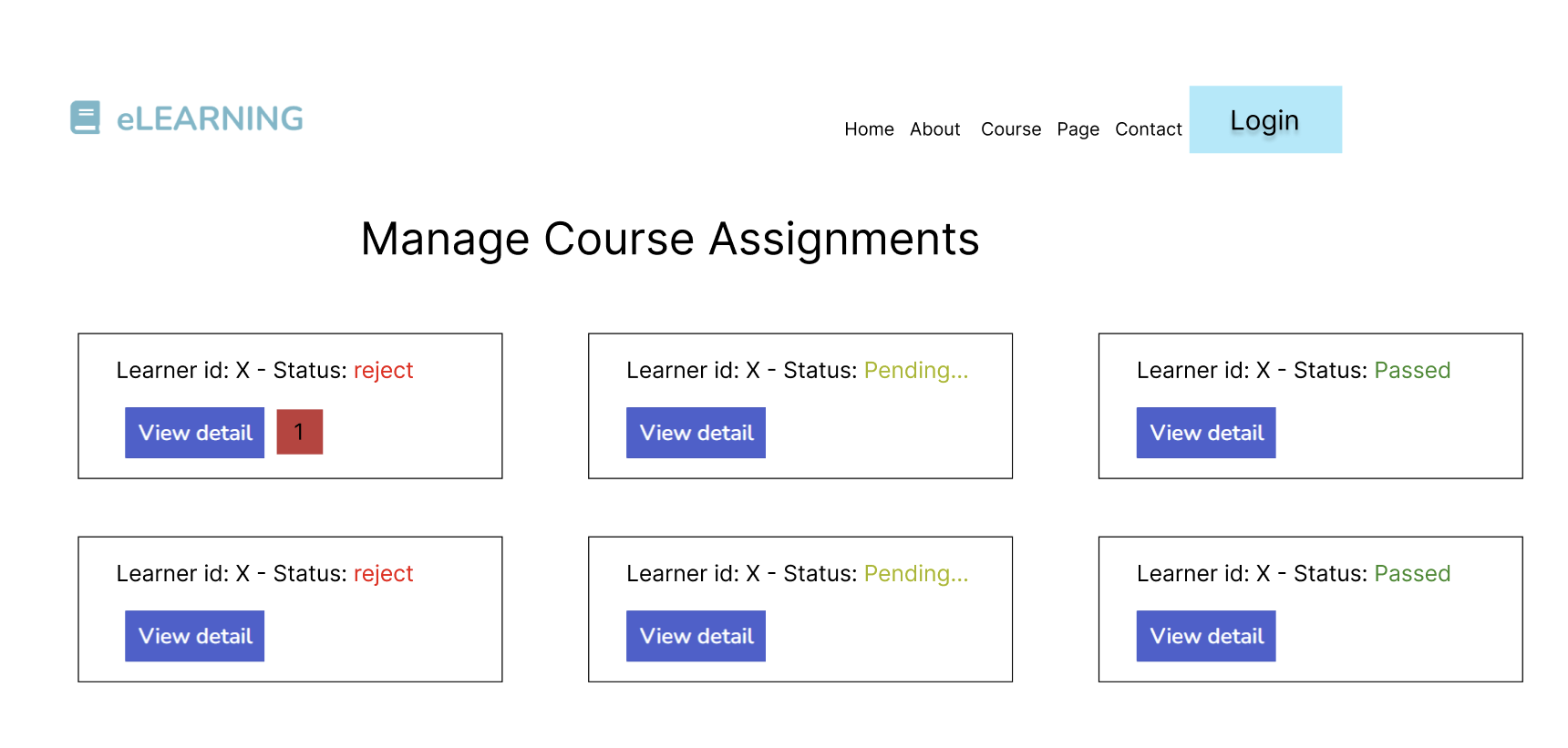
**Page (2)**



(1)Go to view submitted assignments:

* Access to view submitted assignments in cousre

**Page (3)**

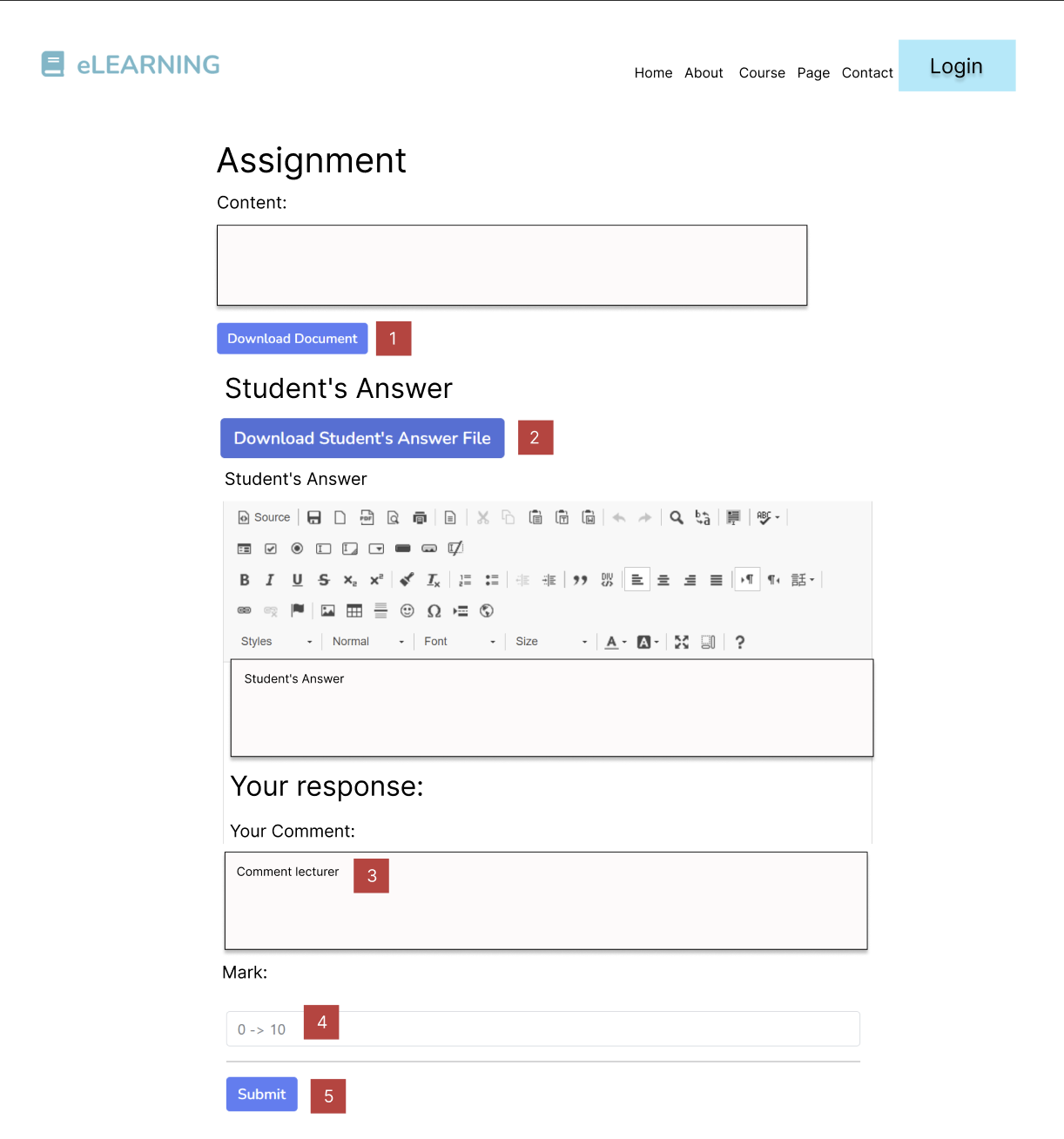
****

(1)View Detail:

* Lecturer can access to view details of learner set

#### 2.5.4 View Assignment Submitted Detail

After accessing the Manage Course Assignments page, the lecturer can click on view detail to see the learner's assignment by viewing the answer or downloading the learner's document, from which they can give comments and scores.



(1)Download Document:

* Lecturer can download and view the assignment I have given.

(2)Download Student’s Aswer File:

* Lecturer can download answers in learner's file

(3)Comment:

* Lecturer can comment on learner's assignments

(4)Mark:

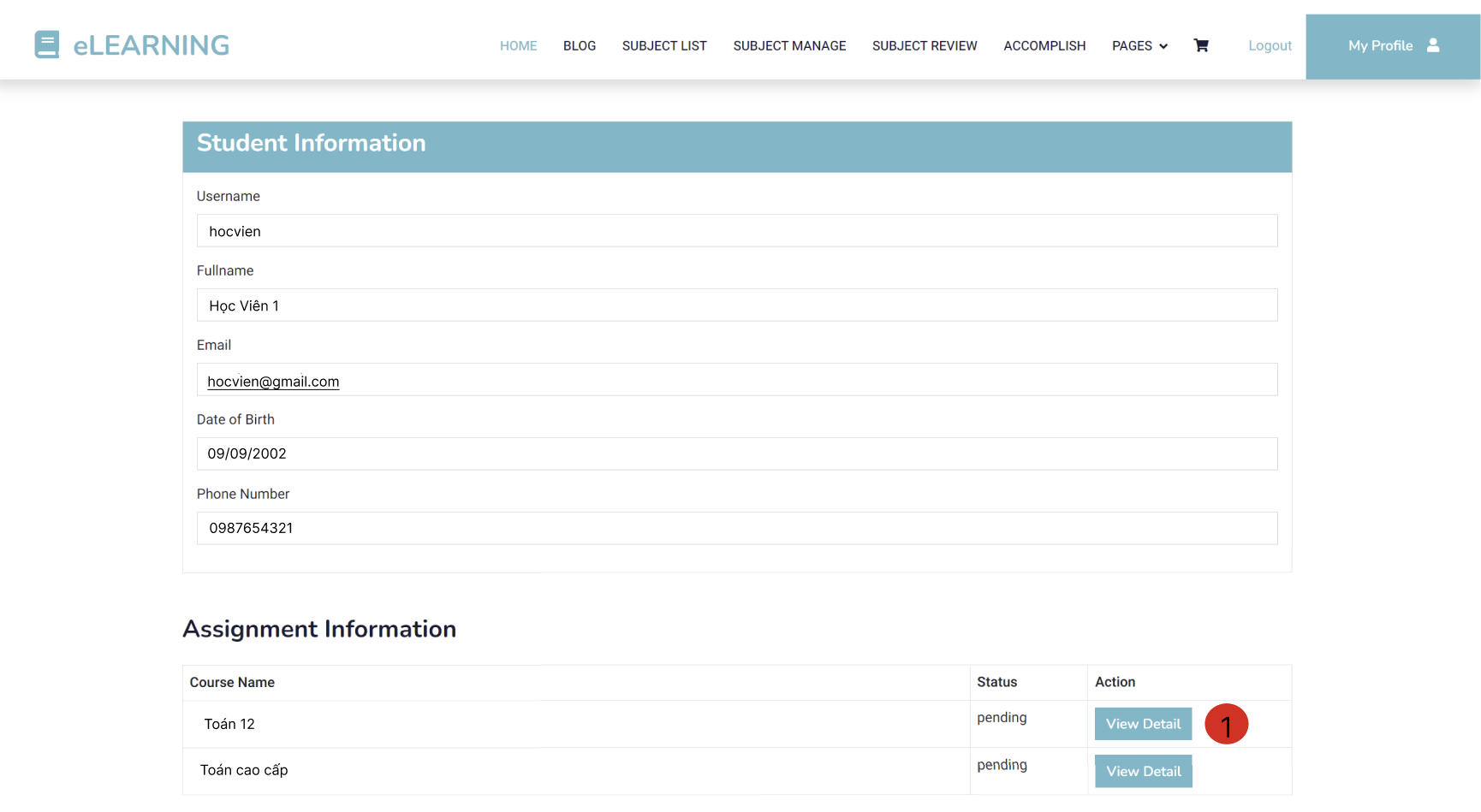
* Lecturer can rate 0 points - 10 points

(5)Submit:

* Lecturer can submit and send for evaluation.

#### 2.5.5 View Learner Information in Assignment

After entering the course assignment list, Lecturer can click on the ID to see the learner's information and the assignments in the subject that the learner has done.



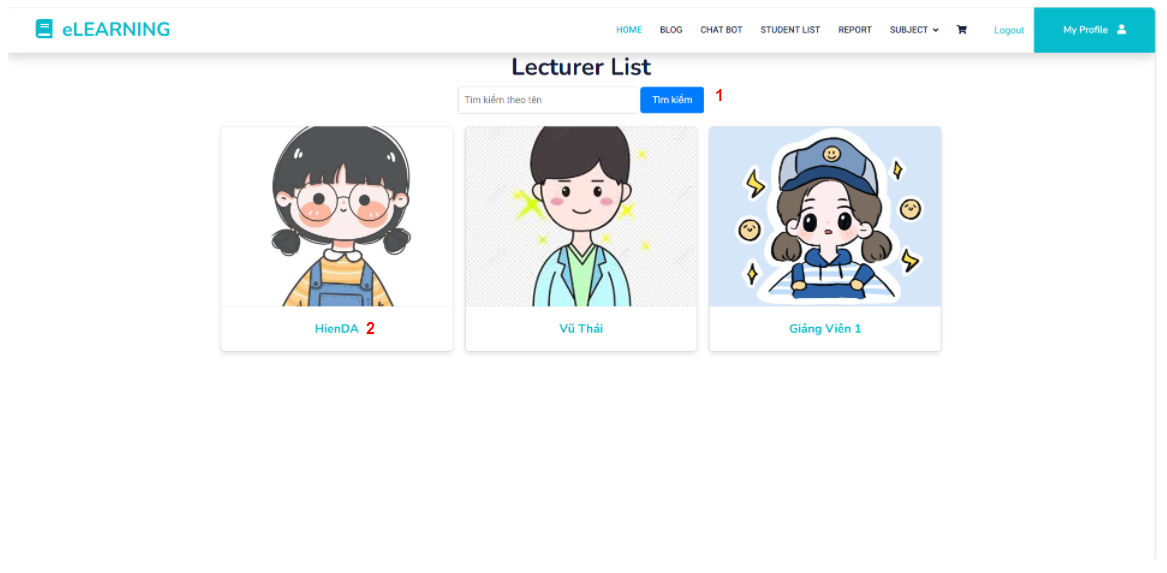
(1)View Detail:

* Lecturer can access to view details of learner set

### 2.6 Lecturer List

In this page user can see Which teacher created which subject, which course?

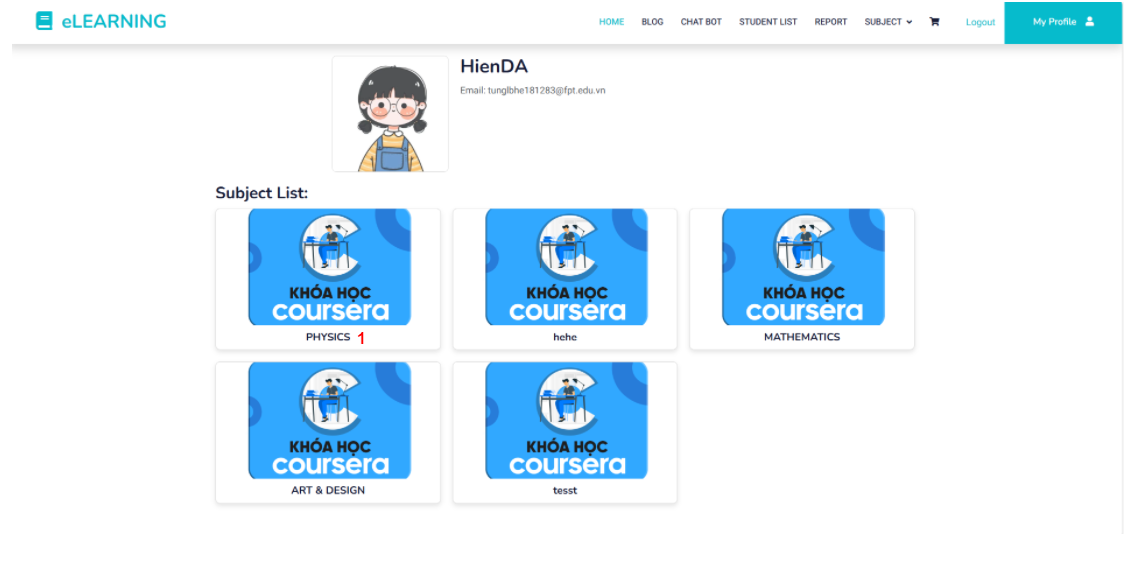
1. To search lecturer name
2. Click will go to next page



In this page user can see all lecturer list in web. User can search name to find they one lecturer

### 2.7 Lecturer Subject List

1. Click will go to subject information page

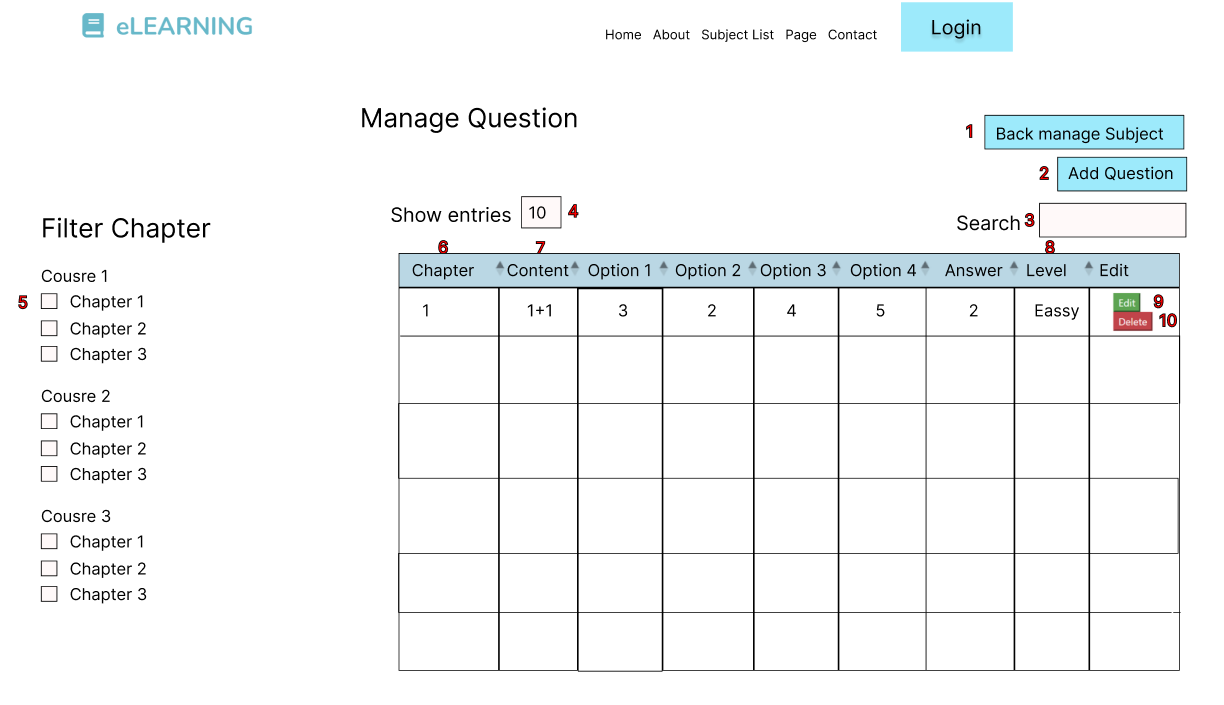


In this page will view lecturer name, avatar, email address and what subject lecturer made. Click on (1) user can see that subject information page.

### 2.8 System Question

#### 2.8.1 Question Manage

After clicking on the managed question in the manage Subject, we will see the managed question with the interface from which we can add edit delete questions.



## 

## 

(1)Back Manage subject:

* Access the in the subject manage

(2)Add Question

* After clicking you can add questions.

(3)Search

* Enter text to search by content.

(4)Show entries

* Select the number of questions to display.

(5)Show Chapter

* Filter questions in selected chapter.

(6)Chapter arrangement

* Chapters can be viewed in ascending or descending order.

(7)Content arrangement

* Can arrange content in order from a-z.

(8)Level arrangement

* Levels can be arranged from easy to difficult or vice versa.

(8)Edit question

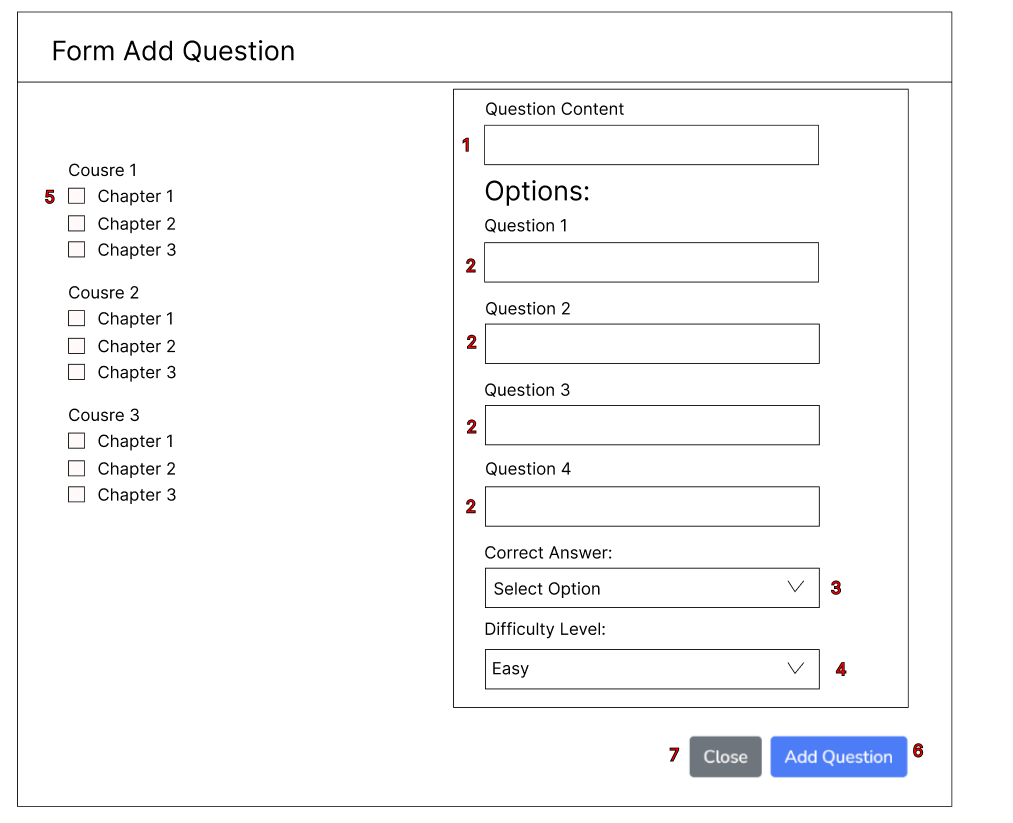
* Get the currently selected question into edit.

(9)Delete question

* Get the currently selected question into Delete.

#### 2.8.2 Add Question

After clicking on the Add Question button on the right corner of the Topic Management page, this page will be displayed.This website allows us to add questions into a chapter of a course.



(1)Question Content:

* Fill in the question

(2) Question

* Fill in the answer to the question above.

(3)Correct Answer:

* Select the answer.

(4)Difficulty Level:

* Select the difficulty of the question.

(5)Select Chapter

* Select chapter to add questions.

(6)Add Question

* Add questions to chapter.

(7)Close

* Close add question

#### 2.8.3 Edit Question

After clicking on the edit Question button next to the question to be edited, this page will be displayed. This website allows us to edit existing questions.

## 

## 

(1)Question Content:

* Fill in the question edit

(2) Question

* Fill in the answer to the question edit above.

(3)Correct Answer:

* Select the answer edit.

(4)Difficulty Level:

* Select the difficulty of the question.

(5)Select Chapter

* Select chapter to add questions.

(6)Edit Question

* Edit selected questions.

(7)Close

* Close Edit question

#### 2.8.4 Delete Question

This feature will be enabled when clicking the button in the admin page. This feature helps Lecturer to delete questions in the chapter.

#### 

#### OK:

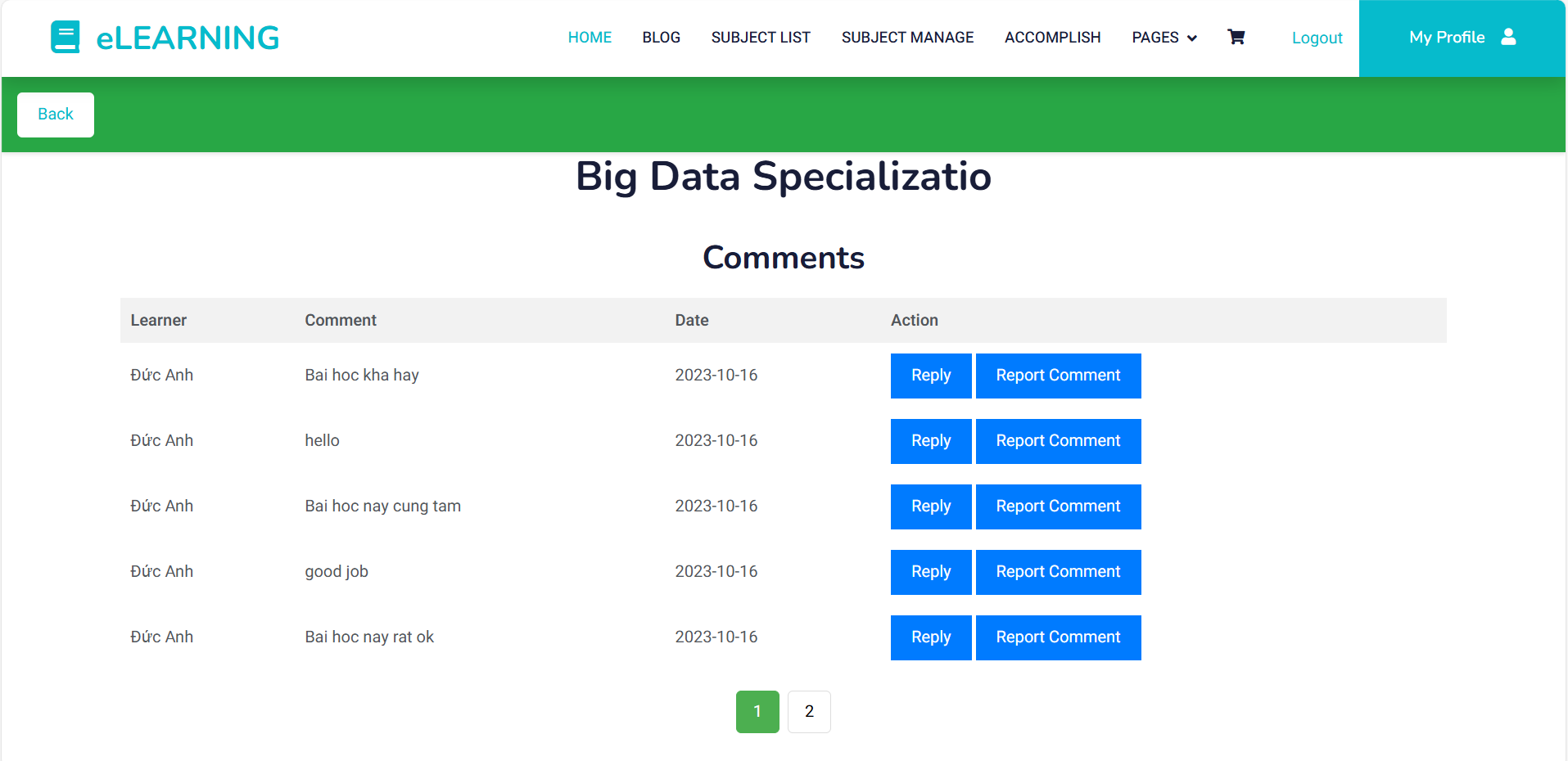
This button allows to delete question.

#### Cancel:

This button allows the lecturer can canceldelete question .

### 2.9 Comment subject

The Comment subject show page shows the lecturer all the comments in the subject .

  
  
(1). Back:

This botton allows lecturer going to subject review page.

(2). Reply:

This botton allows lecturer can respond to learner comments.

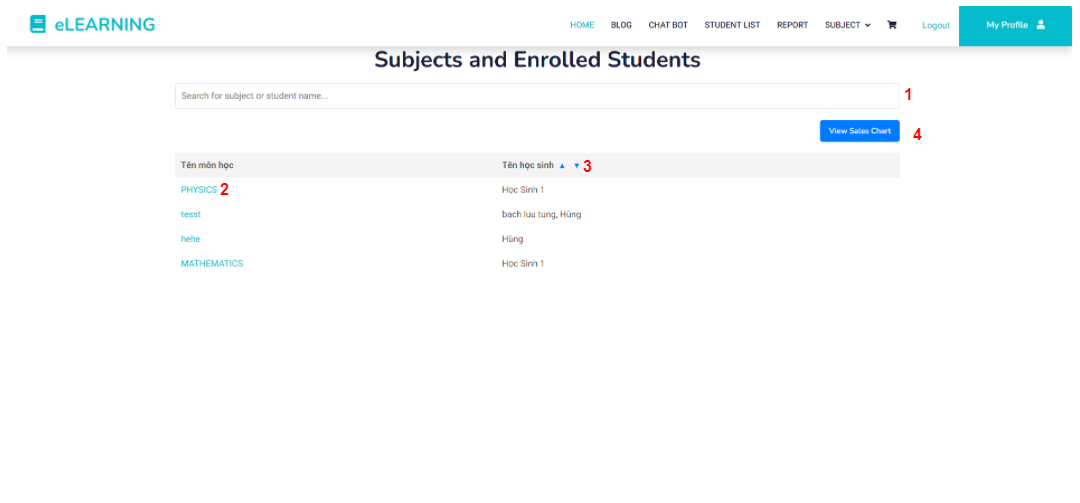
(3). Report comment:

This botton allows lecturer can report comment to learner.

### 2.10 Subjects and Enrolled Students

In this page lecture can see have any one buy they subject

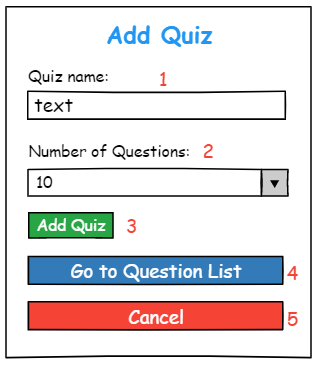
1. To search Subject name or student name
2. Click will view Course manage page
3. Arrange in ascending order of number of buyers and vice versa
4. go to sale chart page



In this page , lecturer can see who buy they subject and know they name. Lecturer can search they subject name, or student name. Beside, lecturer can click on subject name to go to that subject course manage page.

### 2.11 System Quiz

#### 2.11.1 Add Quiz for Chapter



1. Quiz name:

This field allows the lecturer to enter a quiz name.

1. Number of question:

This combo box allow lecturer choose number of questions 10 or 15 or 20

1. Add Quiz

This button will submit the quiz

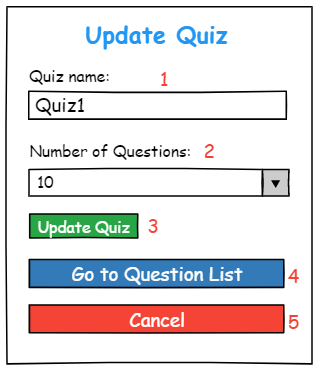
1. Go to Question List

This button will send the lecturer to question list to add question

1. Cancel

This button will cancel and send lecturer to the manage Lesson page

#### 2.11.2 Update Quiz



1. Quiz name:

This field allows the lecturer to rename the quiz.

1. Number of question:

This combo box allows the lecturer to choose 10 or 15 or 20 questions for the quiz.

1. Update Quiz

This button will submit the update quiz.

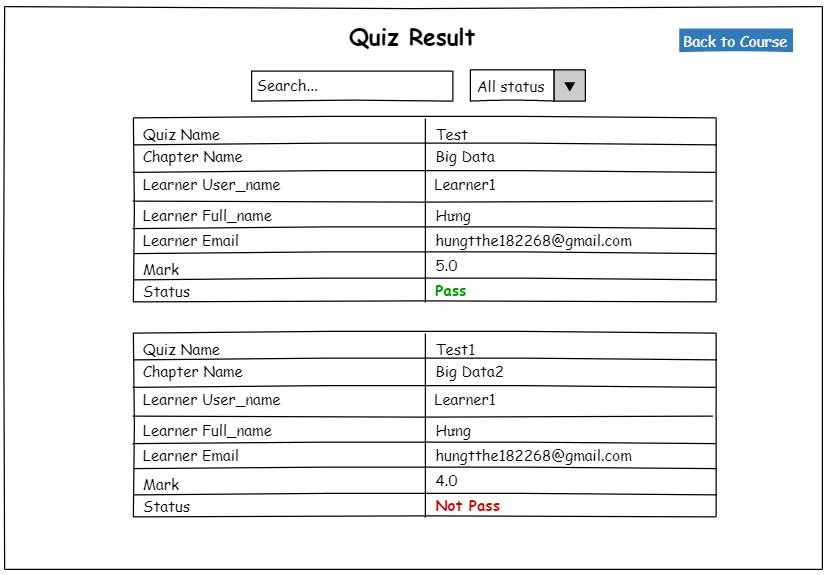
1. Go to Question List

This button will send the lecturer to the question list to add questions.

1. Cancel

This button will cancel and send the lecturer to the manage Lesson page.

#### 2.11.3 View All Quiz Result



Lecturers can view all quiz results that learners have taken.

### 2.12 Sale Chart

1. back to subject and enrolled student page

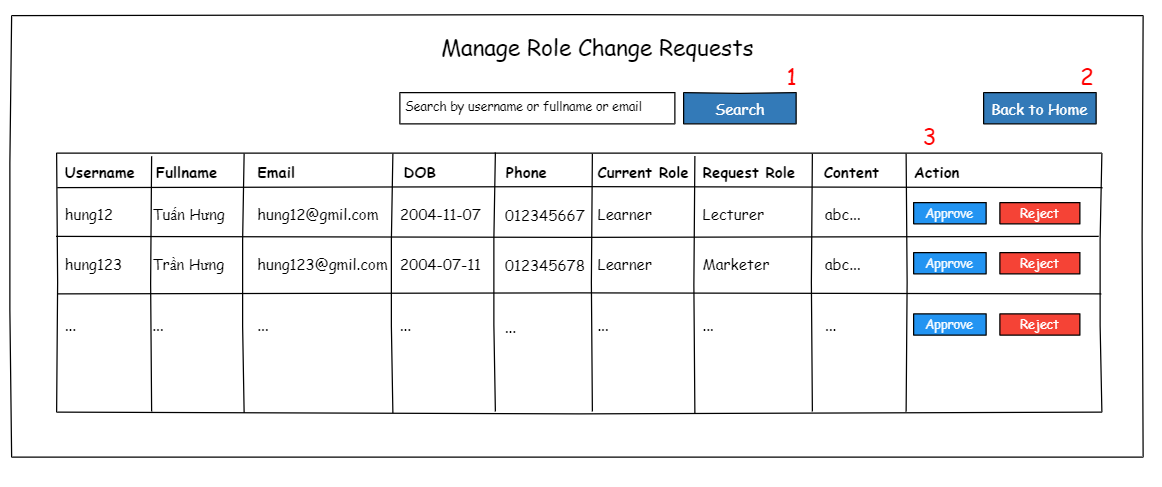


In this page, lecturer can see they sale chart. Revenue is displayed in a bar chart.

## 3. System Admin

### 3.1 System account

#### 3.1.1 Admin manage role change request



1. Search:

This button allows admin to search by username or full name or email.

1. Back to home

This button allows the admin to go back to the home page.

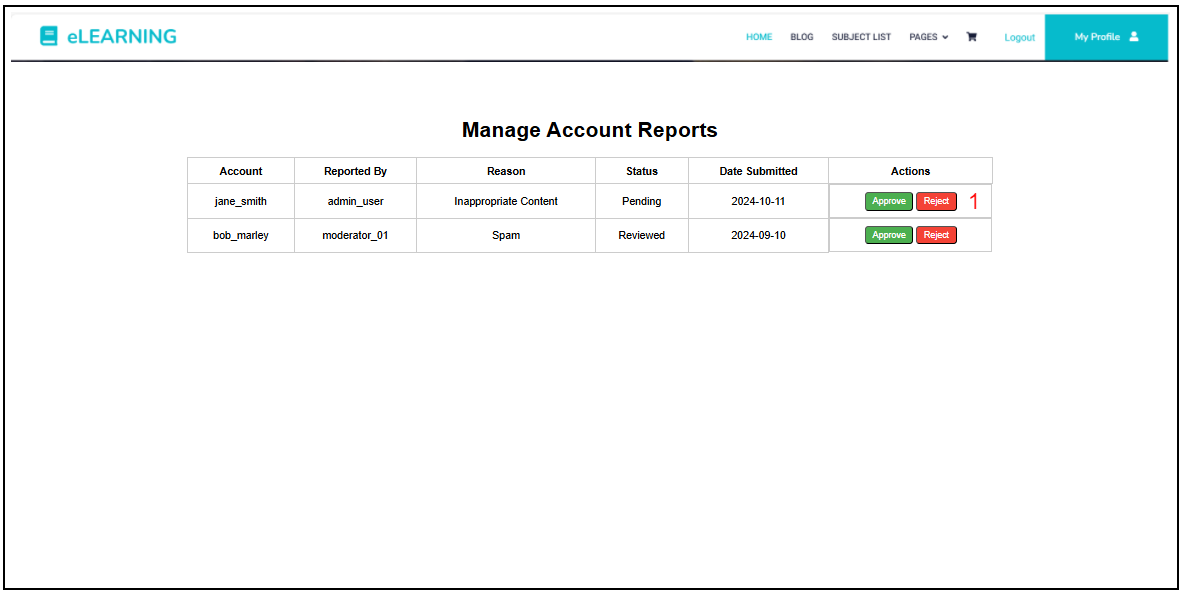
1. Action

Allows admin to approve or reject the request.

#### 3.1.2 Manage Report Account

In this page, if admin approve with report, the reported account wil be block, they will can’t login to they account. In addition, admin can also remove block, or reject that report.

1. Admin approve to report or no



### 3.2 System Order

#### 3.2.1 Orders Management

Admin login header.jsp will display buton OrderManagament

### 

**Search by Order ID**: An input field where users can enter an Order ID to search for a specific order.

**Search by Order Date**:Users select the date and click the corresponding button to initiate the search.

**Search by Account ID**: Users can perform the search by clicking the button below this field.

**Order ID Sorting**: A dropdown menu allowing users to sort the list of orders by Order ID, with options such as "Newest First" to order from the most recent.

**Order Date Sorting**:. Users can select options like "Newest First" to organize the list based on the order date.

**Total Money Sorting**: Sorting options may include highest to lowest or vice versa.

**Search by Order ID Button**: A button that initiates the search based on the Order ID entered in field (1.

**Search by Date & Account ID Button**: A button that triggers a search based on the date in field (2) and the Account ID in field (3).

**Payment Status Filter**: A dropdown menu allowing users to filter the list by Payment Status, such as "Pending" or "Success." This narrows down the displayed orders based on their payment status.

**Analysis Order Button**: This button likely initiates an analysis or generates a report on the displayed or selected orders.

**Account ID Column**: Displays the Account ID for each order. Each Account ID is displayed as a clickable link, highlighted in blue, which directs users to detailed information about that specific account. By clicking on an Account ID, users can quickly access the associated account’s information page.

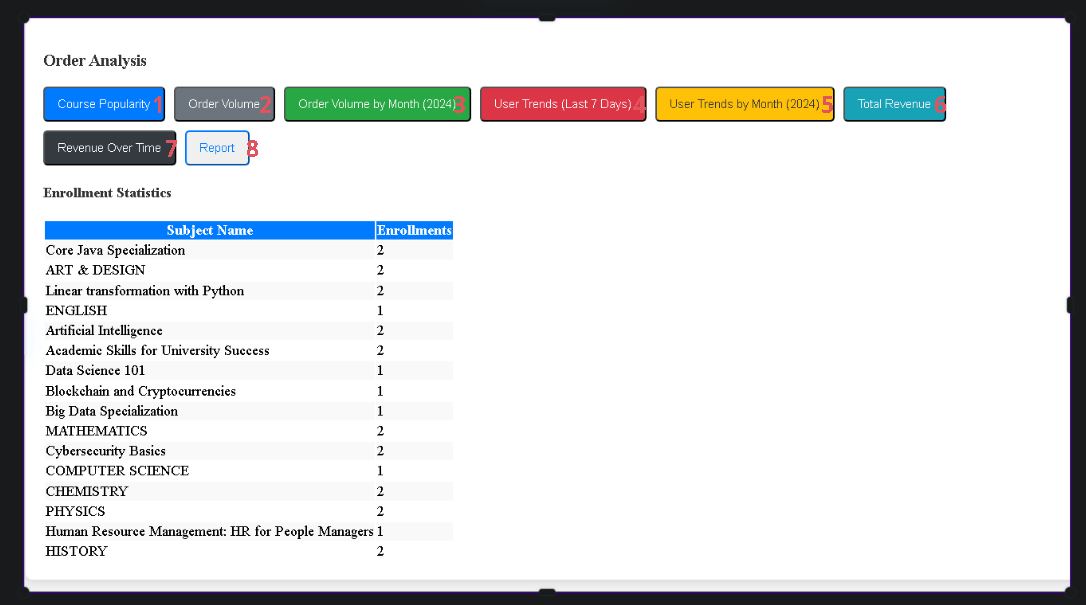
**Payment Status Dropdown**: A dropdown in each row that shows the current payment status of an order (e.g., "Pending" or "Success"). Users can change the payment status directly within this dropdown.

**Action - View Link**: A clickable link labeled "View" that likely opens additional details for the specific order in that row.

**Save Button**: This button saves any changes made to the payment statuses or other editable fields in the order list.

**Pagination**: Allows users to navigate between pages of the order list. Users can click on specific page numbers or "Next" to move through the paginated list.

#### **3.2.2 Orders Analysis**



**Analysis Buttons**:

* The interface provides several buttons for different types of data analysis:
  + **Course Popularity**: Displays the popularity of each course based on enrollments.
  + **Order Volume**: Shows the total number of orders.
  + **Order Volume by Month (2024)**: Analyzes order volume by month for the year 2024.
  + **User Trends (Last 7 Days)**: Shows user activity and trends over the past 7 days.
  + **User Trends by Month (2024)**: Displays user trends and engagement on a monthly basis for 2024.
  + **Total Revenue**: Calculates and displays the total revenue generated.
  + **Revenue Over Time**: Shows revenue changes over time, useful for identifying growth patterns.
  + **Report**: Generates a detailed report based on selected data.

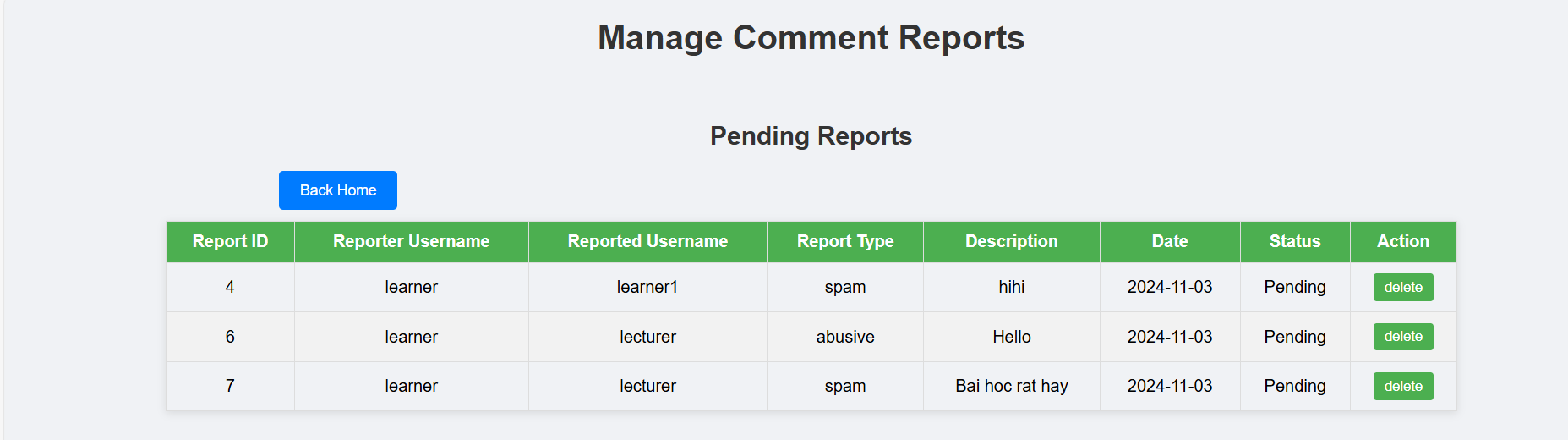
**Enrollment Statistics Table**:

* A table that lists various subjects and the number of enrollments for each.
* The **Subject Name** column displays the names of subjects or courses, while the **Enrollments** column shows the number of users enrolled in each.
* Alternating row colors improve readability, and a blue header distinguishes the column titles.

### 3.3 System Comment

#### 3.3.1 Manage Comment Reports

This is the Manage Comment Reports page to show us the information of the reporter, the person being reported, and related information

  
(1). Back Home:

This button is used to return to the admin admin page .

(2). Delete:

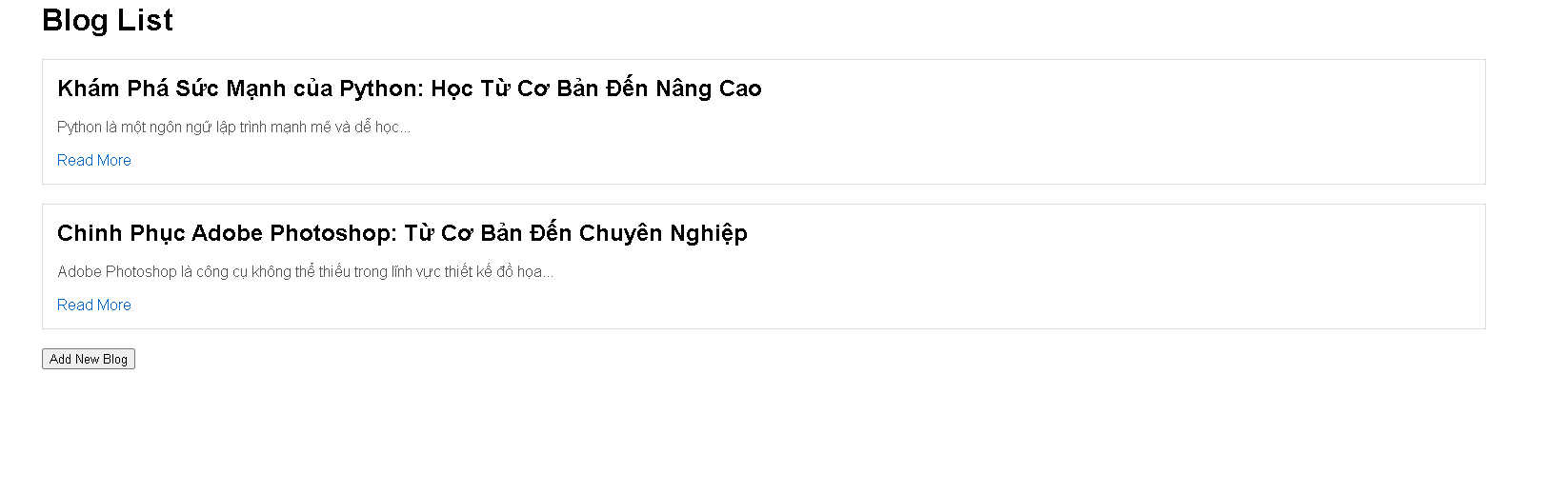
Click this button to allow admins to delete reported comments .

## 4. System marketer

### 4.1 System blog

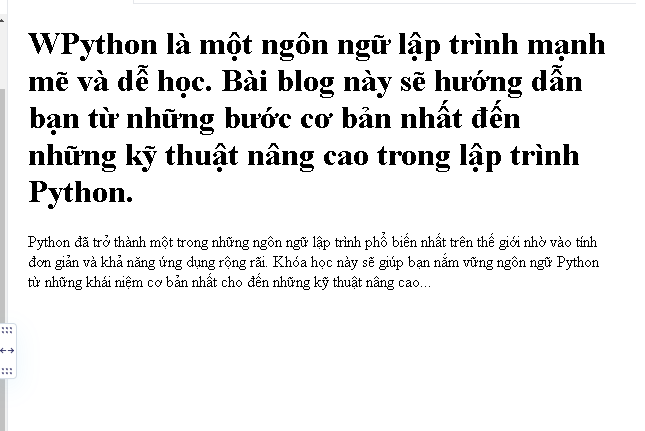
#### 4.1.1 Blog List

Users access the blog list page and can view blogs displayed in a certain order (e.g. by date, blogviews, topic).



#### 4.1.2 Blog Detail

When the user clicks on readmore, the screen will switch to the detail block, which will display the course details.



4.3 Add Blog

Admin can add new blog content to the system by entering the required information and saving.



#### 4.1.3 Edit Blog Posts

**Actors:**

* **Primary Actor**: User (Admin/Marketer)
* **Supporting Actor**: System

**Description:**

This use case describes the process where a user views, updates, or deletes blog posts using a paginated blog management interface.

**Preconditions:**

* The user is logged in as an admin or content creator.
* There are existing blog posts in the system.
* The system displays the list of blog posts with the option to update or delete each post.

**Postconditions:**

* The user can successfully update a blog post.
* The user can successfully delete a blog post.
* The list of blog posts is paginated, allowing easy navigation between pages.

**Main Success Scenario (Basic Flow):**

1. **User navigates to the Blog Management page**:  
    The user logs into the system and navigates to the "Edit Blog" page.
2. **System retrieves blog posts**:  
    The system retrieves and displays a list of blog posts (e.g., "Blog1", "Blog2", "Blog3"), each containing a title, body text, and two action buttons (Update, Delete).
3. **User views blog post details**:  
    The user can view the title and body text of each blog post on the current page.
4. **User updates a blog post**:  
    The user clicks the "Update" button next to a specific blog post.
   * The system redirects the user to an editing interface where they can modify the title and content.
   * The user makes changes and clicks "Save".
   * The system updates the post in the database and refreshes the list.
5. **User deletes a blog post**:  
    The user clicks the "Delete" button next to a specific blog post.
   * The system asks for confirmation.
   * The user confirms deletion, and the system removes the blog post from the database and refreshes the list.
6. **User navigates between pages**:  
    The user clicks the "Previous" or "Next" buttons or selects a specific page number to view more blog posts.
   * The system retrieves and displays the corresponding page of blog posts.

**Extensions (Alternate Flows):**

4a. **Update blog post fails**:

* If the system fails to save the updated post (e.g., due to a server error or missing input), the system shows an error message and allows the user to retry.

5a. **Delete blog post fails**:

* If the system fails to delete the post (e.g., due to a server error), the system displays an error message and the post remains in the list.

6a. **No blog posts to display**:

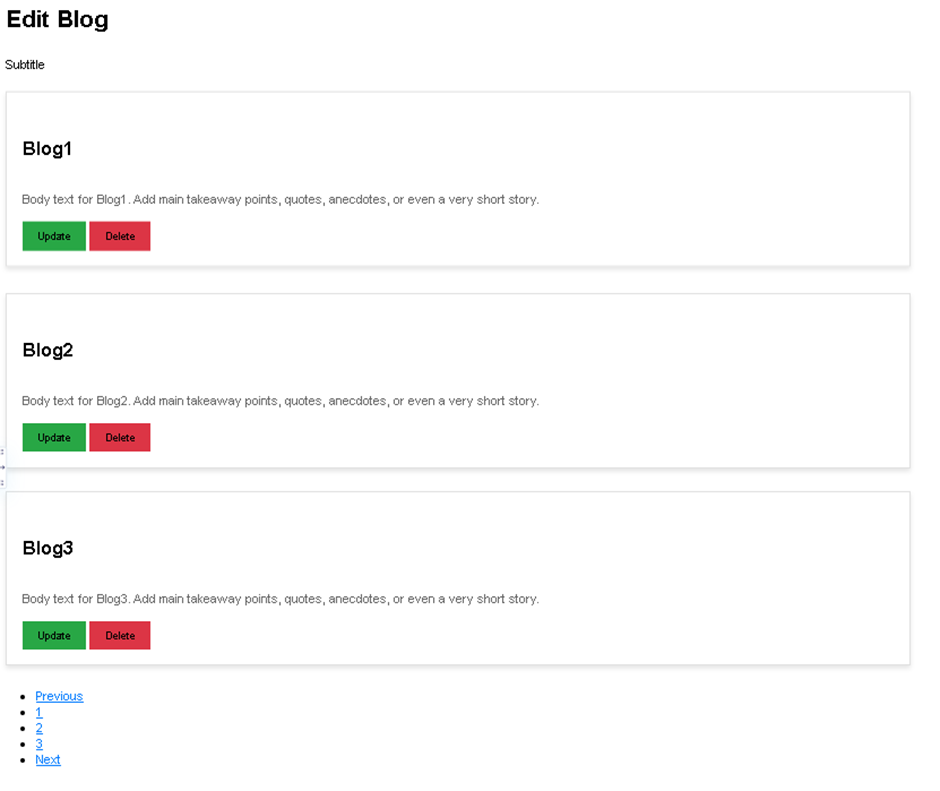
* If the user navigates to a page with no blog posts, the system shows a message like "No blog posts available on this page."

**Special Requirements:**

* The system must paginate the blog posts efficiently to ensure smooth navigation.
* The system should confirm actions like delete with a dialog box to prevent accidental deletions.
* Blog post updates must be saved in real-time, with feedback given to the user upon success or failure.

**Assumptions:**

* The system holds a list of blog posts stored in a database.
* The user has the appropriate permissions to update and delete blog posts.



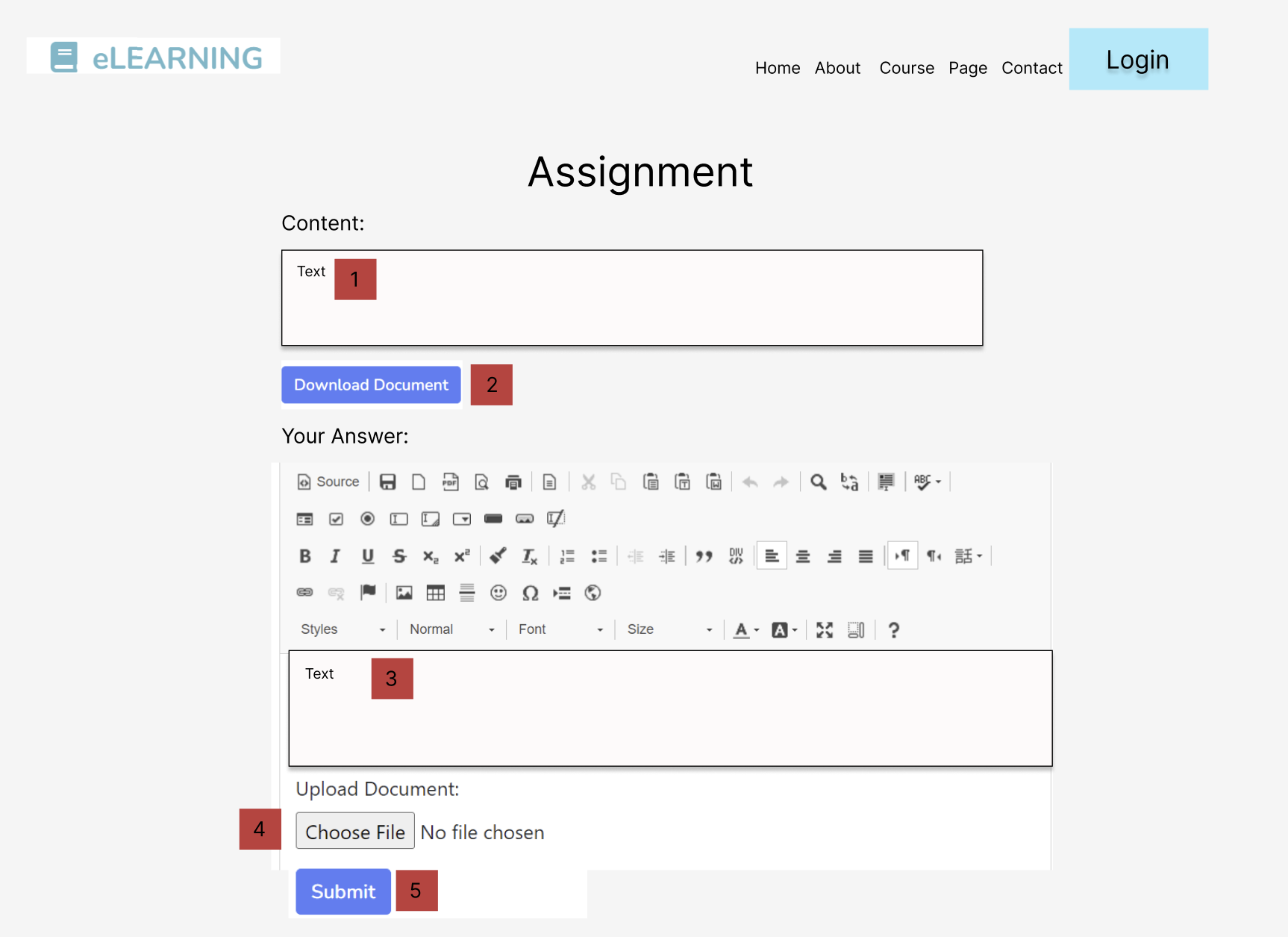
## 5. System Learner

### 5.1 System Assignment

#### 5.1.1 Take Assignment

After entering the course detail, click on the take assignment button to be redirected to the take assignment page.

This allows us to download the assignments that the teacher has assigned in document form.



(1)Content:

* Display content provided by lecturer

(2)Download Document:

* Allow students to download documents provided by the lecturer

(3)Your Answer:

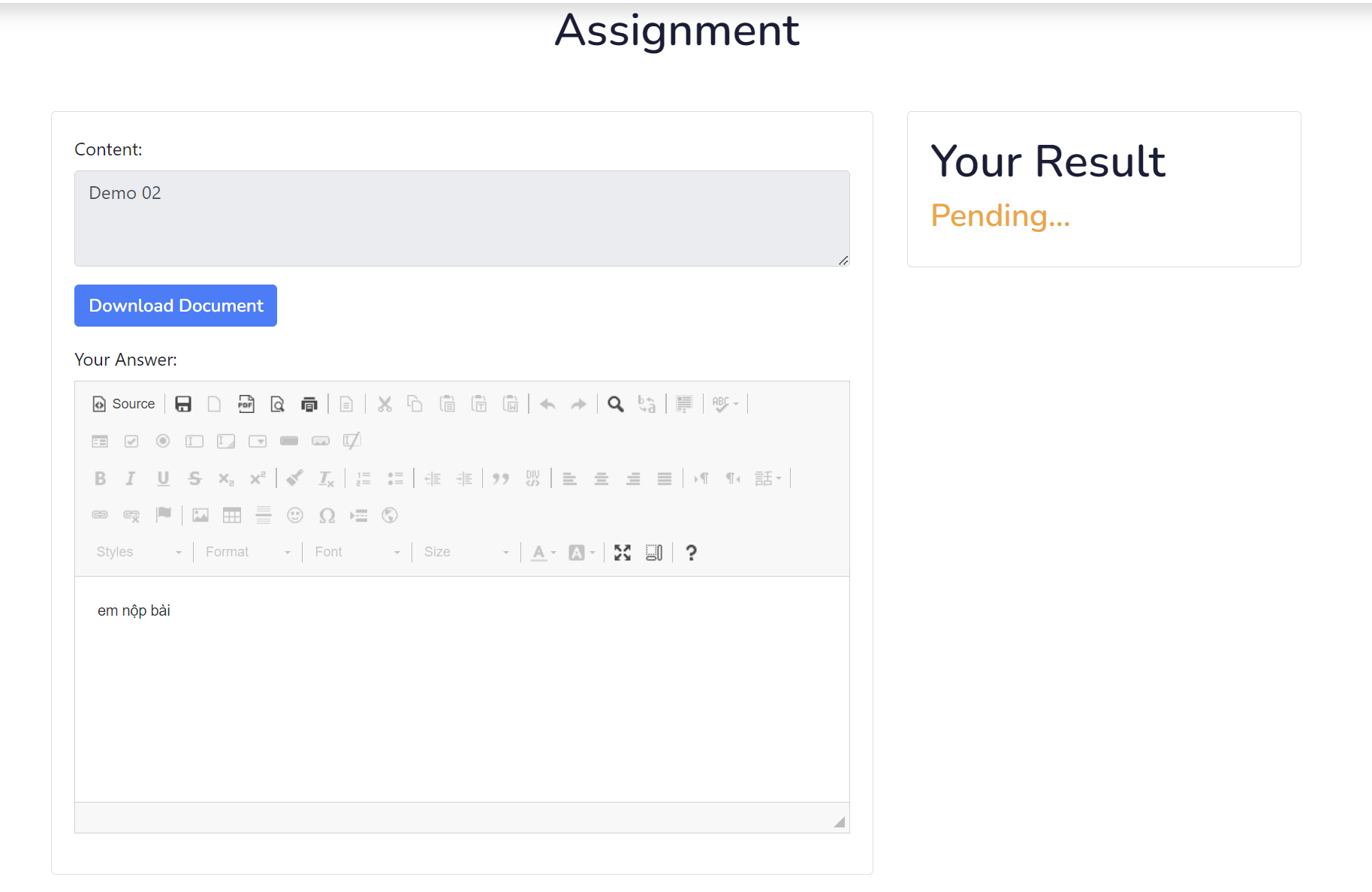
* Allow students to provide answers

(4)Choose File:

* Allow students to upload completed homework files

(5)Submit:

* Allow students to submit assignments



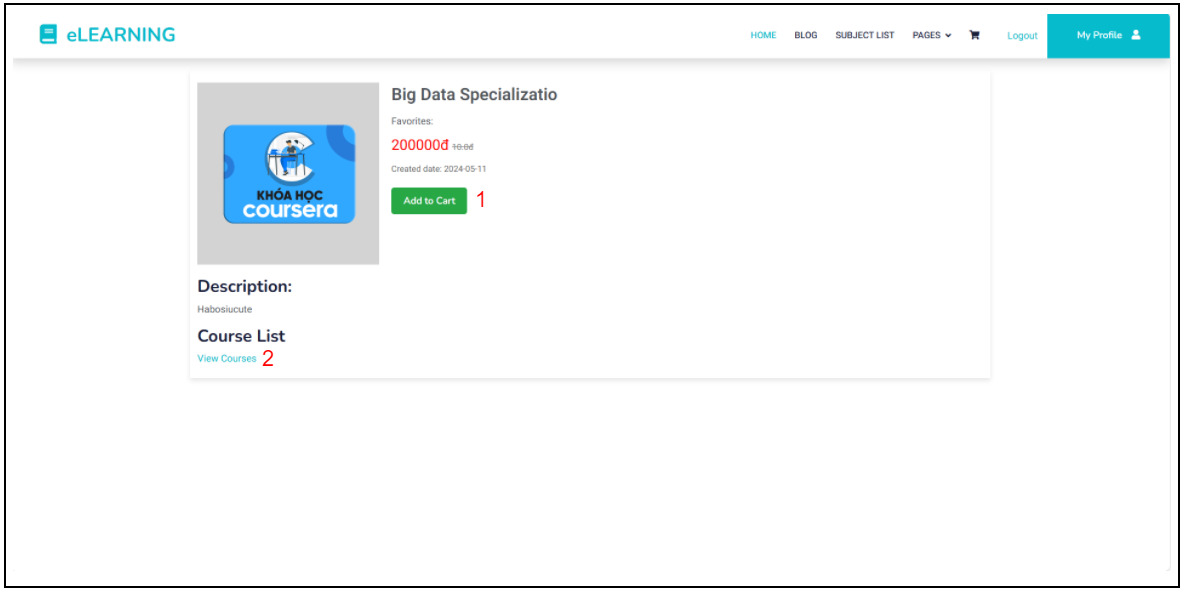
After submitting the assignment, a page will appear stating that the student will wait for the teacher to grade it.Here students can also download files they have previously submitted.

### 5.2 Cart

When click on View buttom in Subject list page will view this page:

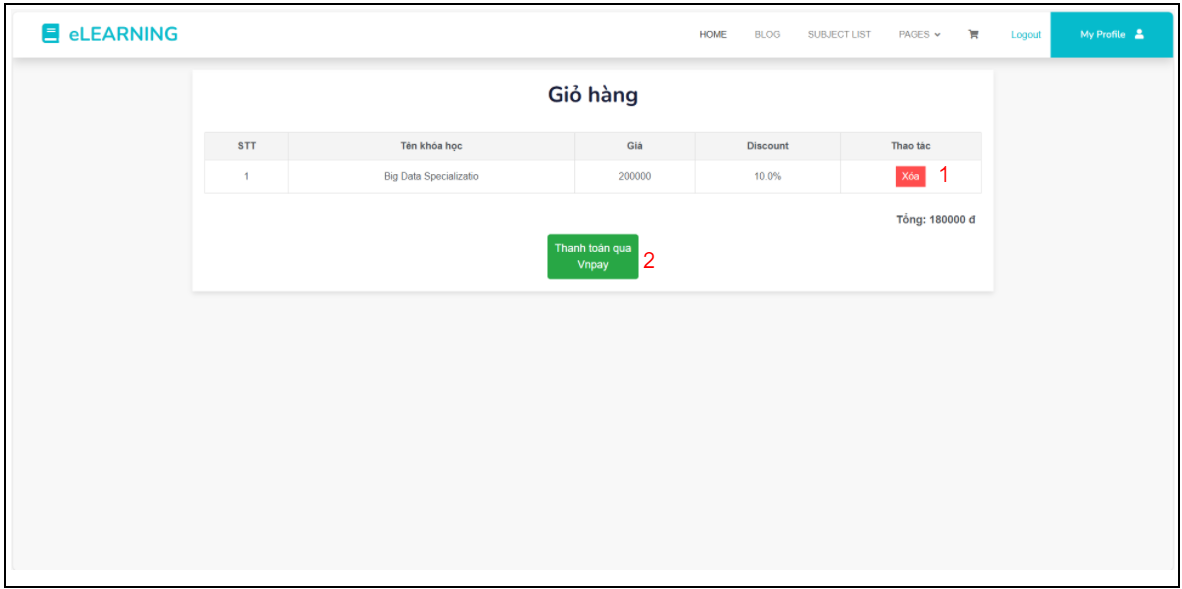
In this page screen subject ava, subject name, favorities count, Price, discount, created date, Thêm vào giỏ hàng buttom, subject descriptions, course list.

1. when user click to this buttom this subject will go to user Cart, to wait user payment that subject.
2. To view Course list page



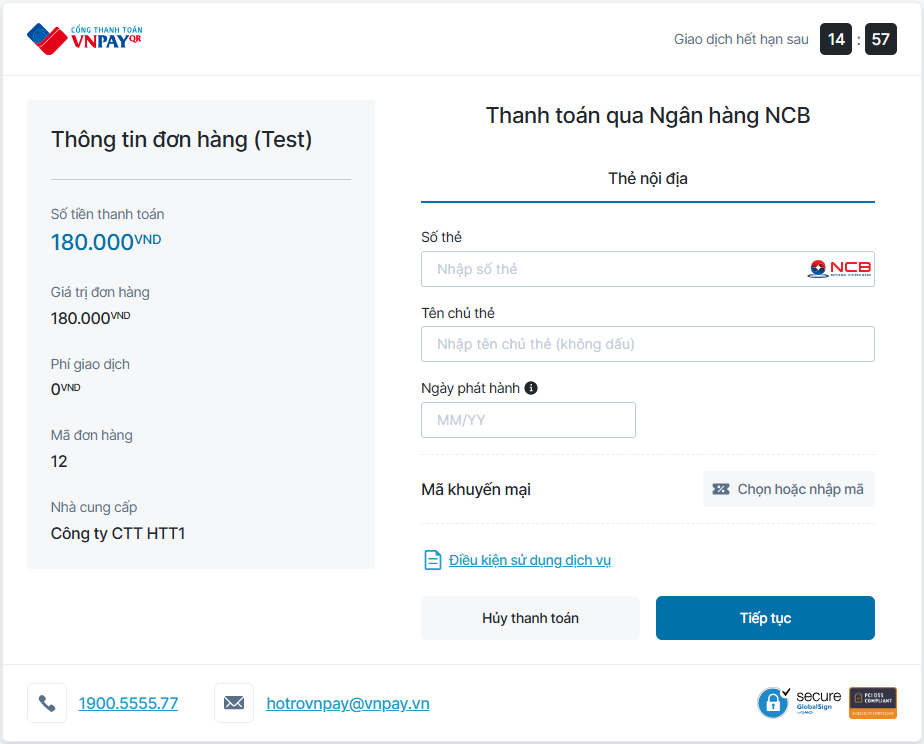
After click on Thêm vào giỏ hàng buttom subject will add to Cart. And user can go to other subject and add them to Cart. When user go to Cart , user can delete subject thay don’t want to buy in (1).

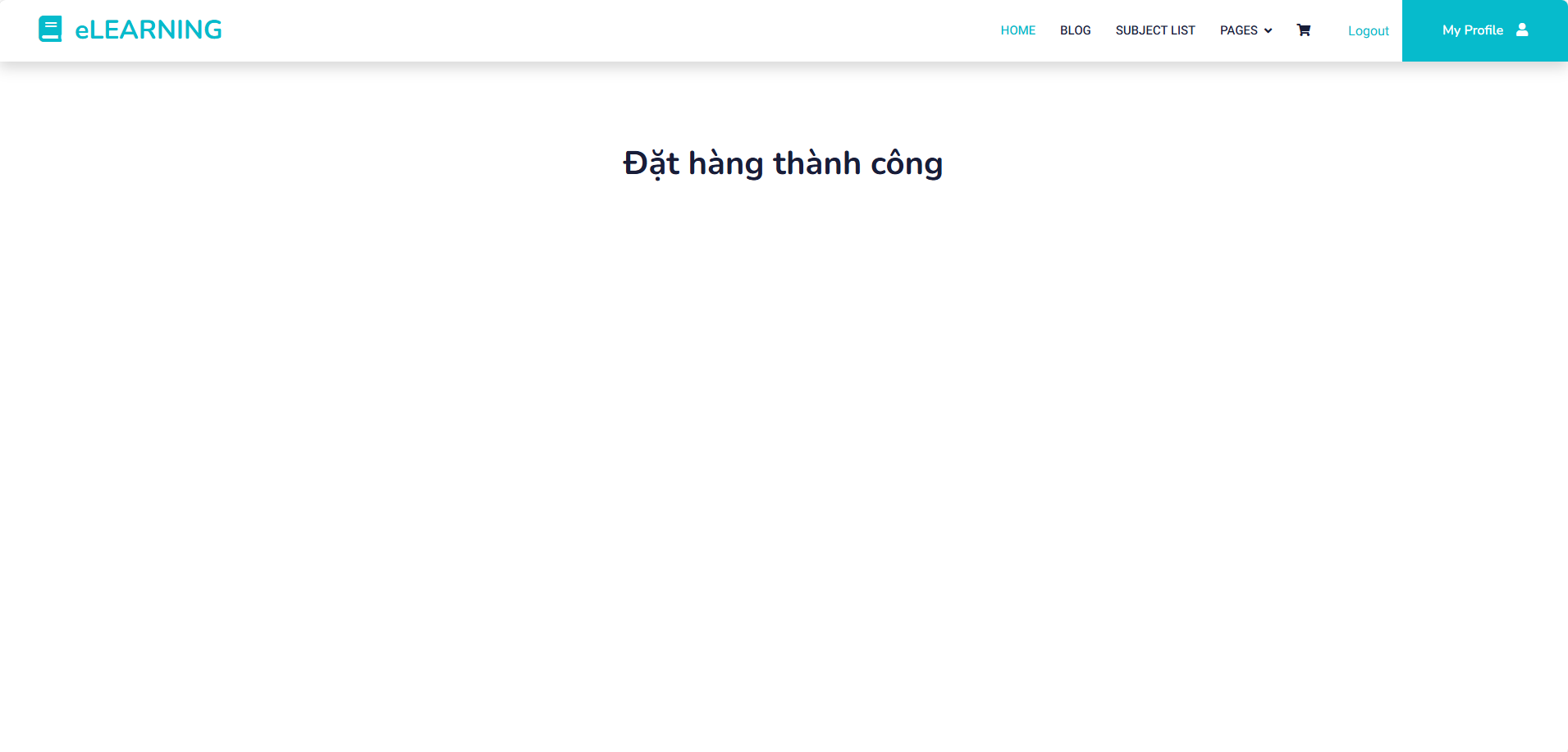
When click on (2) will change to Pay page.



### 5.3 Pay

In this page, we wil have vn pay account to pay, if pay success, user can go to star learn they buy subject.

In this page, User must to enter the information cart after pay done wil go back to web page.



### 5.4 Role Change Request



1. Select new role:

This combo box allows users to choose to be a lecturer or marketer.

1. Content:

This field allows the user to send a reason.

1. Submit Request button:

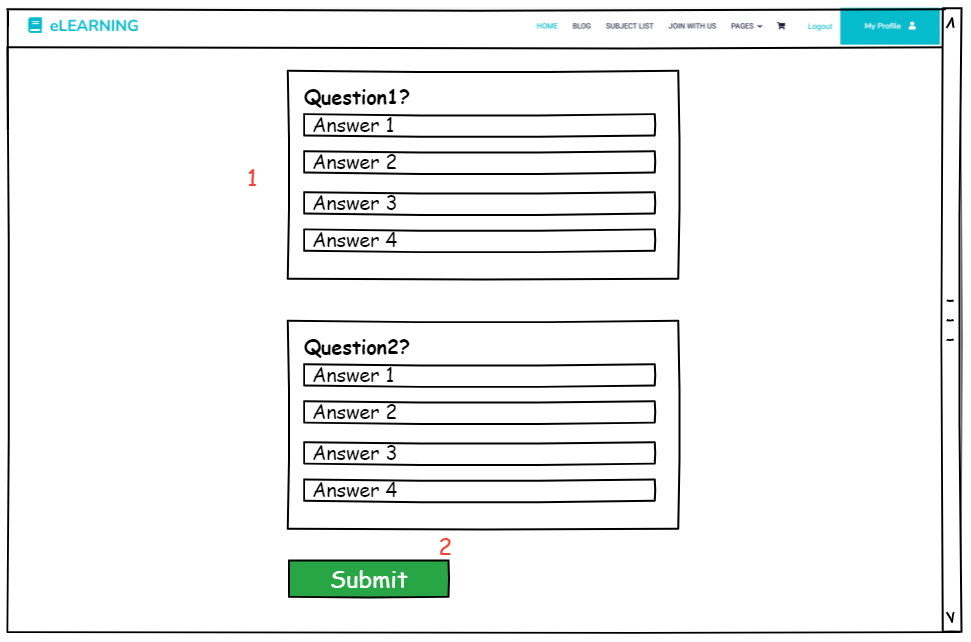
This button will send your request to the admin.

1. Cancel:

This button cancels your request and sends you back to the home page.

### 5.5 System Quiz

#### 5.5.1Take Quiz



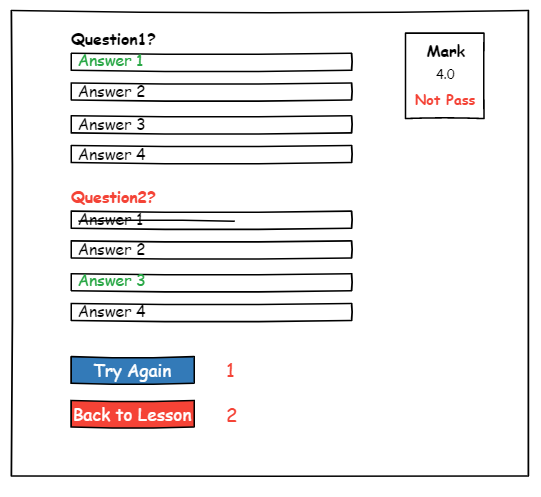
1. Answer field

Learner can choose 1 in 4 answer for that question

1. Submit

This button will submit and send you to the quiz result

#### 5.5.2 Quiz Result



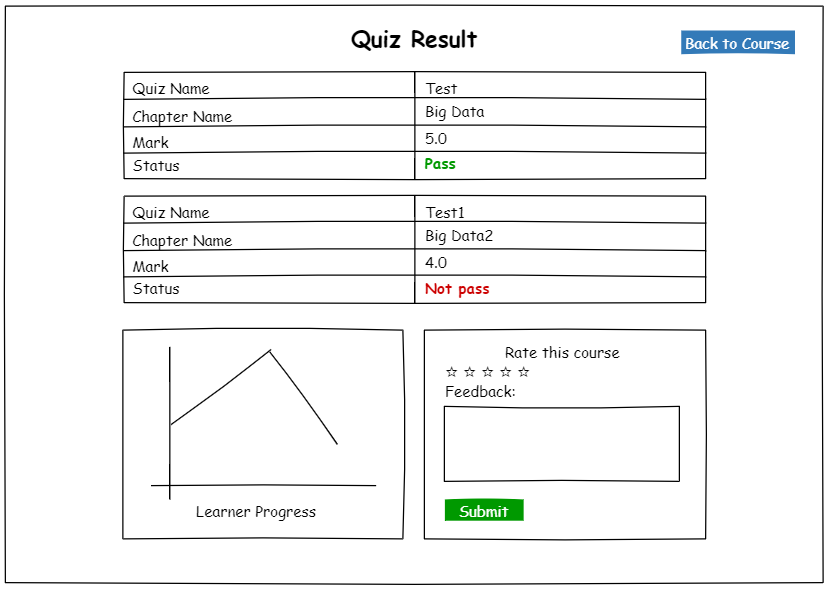
1. Try Again

This button allows learner to take another quiz if they do not pass the quiz. If they pass, this button will not show out.

1. Back to Lesson

This button will send learner back to lesson list.

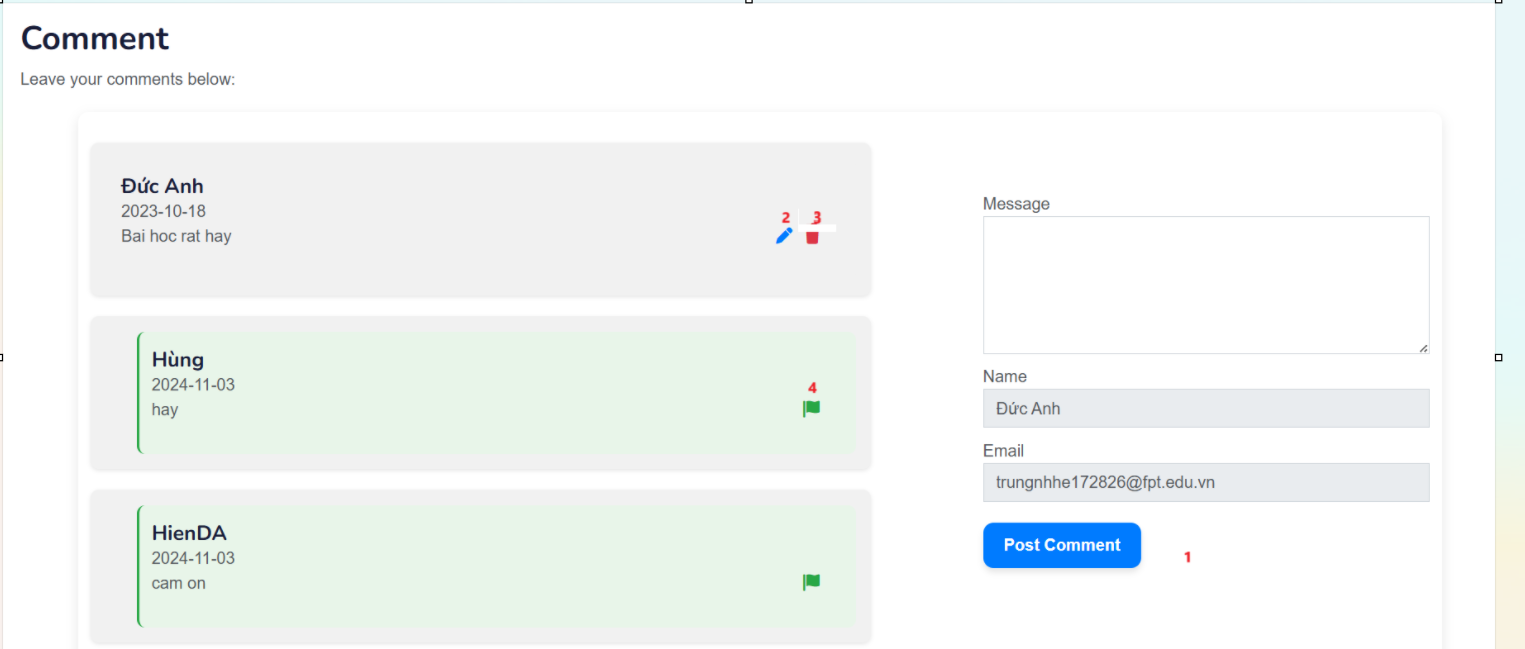
#### 5.5.3 Track learning progress



Learner can view their result for each quiz they have taken, their Learner Progress and Rate feedback course.

### 5.7 Comment

The comment page is used to allow learners to comment below courses .



(1). Post Comment:

This button is used to allow learners to comment below the lesson .

(2). Edit comment:

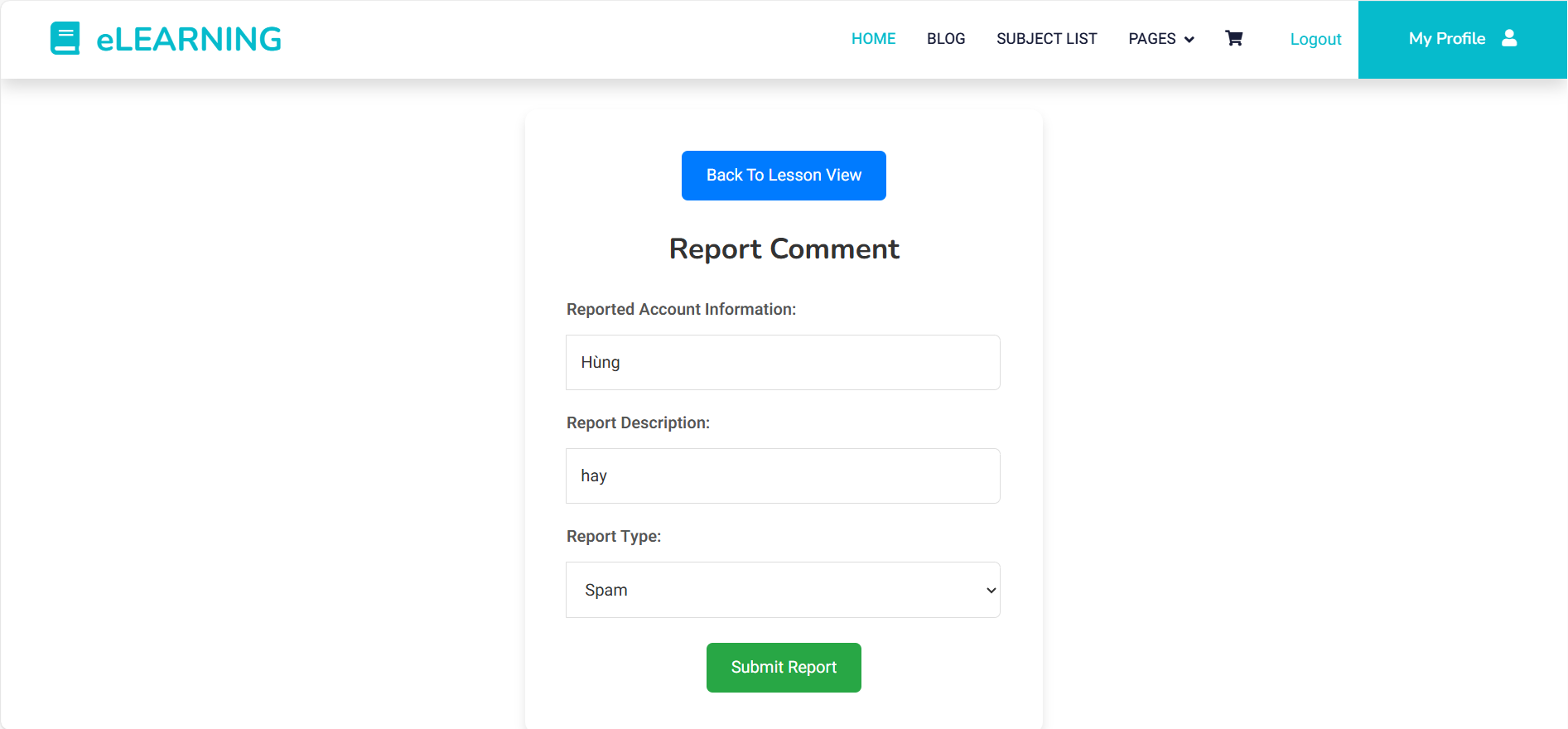
This button is used to allow learners to edit the commented content

(3). Delete comment:

This button is used to allow learners to delete commented content .

### 5.8 Report

The report page can show us information related to the reported comment .



(1). Back to lession view:

This button can allow the user to return to the Lession View page .

(2). Report account information:

This text show information being reported.

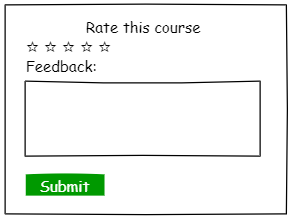
(3). Report description:

This text show comment being reported.

(4). Type report:

We can choose the report type .

### 5.9 Feedback and Rate Course



1. Rate Star

Allow learner rate star for the course they learn

1. Feedback

Allow learner send a feedback for the course they learn

1. Submit

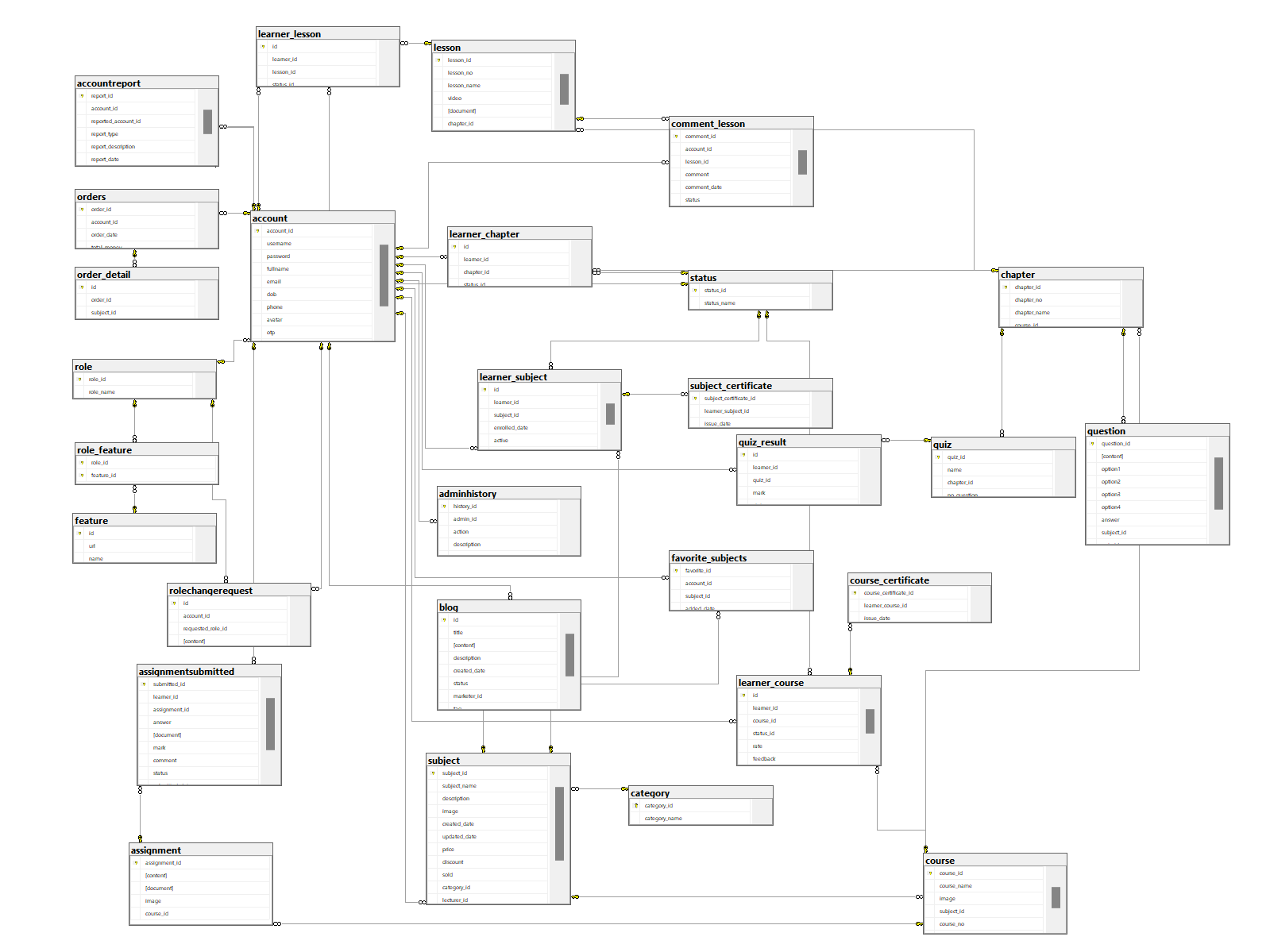
This button will submit the rate and feedback

# III. System Design

## 1. Database Design

*[Provide the tables relationship like example below]*

### 1.1 Database Schema



### 1.2 Table Descriptions

#### 

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 01 | Account | This is a table to store basic information and user distribute authorization  - Primary keys: <account\_id>  - Foreign keys: <role\_id (table.Role)> |
| 02 | Role | This is a table that stores user authorizations available on the website  - Primary keys: <role\_id>  - Foreign keys: <NONE> |
| 03 | Blog | This is a table that stores basic information about the blog  - Primary keys: <blog\_id>  - Foreign keys: <marketer\_id (table.Account)> |
| 04 | Subject | This is a table that stores basic information about the subject such as id, name, created lecturer,...  - Primary keys: <subject\_id>  - Foreign keys: <lecturer\_id (table.Account)> |
| 05 | Learner\_ Subject | This table stores information about which learner studies which subject  - Primary keys: <id>  - Foreign keys: <learner\_id (table.Account),  subject\_id(table.Subject)> |
| 06 | Course | This is a table that stores basic information about course  - Primary keys: <subject\_id>  - Foreign keys: <lecturer\_id (table.Account)> |
| 07 | Learner\_ Course | This table stores basic information about which student takes which course  - Primary keys: <learner\_course\_id>  - Foreign keys: <learner\_id (table.Account),  course\_id (table.Course)> |
| 08 | Category | This is a table that stores information about categories on the website  - Primary keys: <category\_id>  - Foreign keys: <NONE> |
| 09 | Chapter | This table stores basic information of the chapter  - Primary keys: <chapter\_id>  - Foreign keys: <course\_id (table.Course)> |
| 10 | Learner\_Chapter | This table stores basic information about which student takes which chapter  - Primary keys: <learner\_chapter\_id>  - Foreign keys: <learner\_id (table.Account), chapter\_id (table.Chapter)> |
| 11 | Quiz | This is a table that stores information about which quiz belongs to which chapter  - Primary keys: <quiz\_id>  - Foreign keys: <chapter\_id (table.Chapter)> |
| 12 | Question | This is a table that stores information about questions including questions, and answers, as well as which question belongs to which quiz.  - Primary keys: <question\_id>  - Foreign keys: <quiz\_id (table.Quiz)> |
| 13 | Quiz\_ Result | This is a table that stores information, results, and status of quizzes for each learner  - Primary keys: <quiz\_result\_id>  - Foreign keys: <learner\_id (table.Account),  quiz\_id (table.Quiz)> |
| 14 | Lesson | This is a table that stores information about which lessons belong to which chapter as well as which subject  - Primary keys: <lesson\_id>  - Foreign keys: <chapter\_id (table.Chapter)> |
| 15 | Learner\_Lesson | This is a table that stores status information of each lesson for each learner  - Primary keys: <learner\_lesson\_id>  - Foreign keys: <learner\_id (table.Account),  lesson\_id (table.Lesson)> |
| 16 | Comment\_Lession | This is a table that stores comment information of each lesson for each learner  - Primary keys: <comment\_id>  - Foreign keys: <account\_id (table.Account),  lesson\_id (table.Lesson)> |
| 17 | Role\_ Feature | This table stores information about which features users can access  - Primary keys: <role\_id, feature\_id>  - Foreign keys: <role\_id (table.Role), feature\_id (table.Feature)> |
| 18 | Feature | This table stores information about visited links and path names describing the links  - Primary keys: <feature\_id>  - Foreign keys: <NONE> |
| 19 | Order | This table stores user information once payment has been completed  - Primary keys: <order\_id>  - Foreign keys: <account\_id(table.Account)> |
| 20 | Order\_Detail | This table stores information about which subjects to buy and the price for each subject  - Primary keys: <id>  - Foreign keys: <order\_id(table.Order)> |
| 21 | Status | This table stores the different statuses that can be assigned.  -Primary keys: status\_id |
| 22 | role change request | This table stores the request to change the role of the learner.  -Primary keys: <id>  -Foreign keys: <Account(account\_id)>  -Foreign keys: <Role(role\_id)> |
| 23 | AccountReport | This table stores the report the violating account of user .  -Primary keys: <report\_id>  -Foreign keys: <Account(account\_id)> |
| 24 | AdminHistory | This table stores the activities of admin.  -Primary keys: <history\_id>  -Foreign keys: <Account(account\_id)> |
| 25 | subject\_certificate | This table stores the certificate of the subject after completion.  -Primary keys: <subject\_certificate id>  -Foreign keys: <learner\_subject\_id**>** |
| 26 | course\_certificate | This table stores the certificate of the course after completion.  -Primary keys: <course\_certificate id>  -Foreign keys: <learner\_subject\_id**>** |
| 27 | favorite\_subject | This table stores favorite subjects.  -Primary keys: <favorite\_id>  -Foreign keys: <account\_id**>** |
| 28 | assignment | This table stores assignment.  -Primary keys: <assignment\_id >  -Foreign keys: <course\_id**>** |
| 29 | assignment submitted | This table stores assignment submitted  -Primary keys: <submitted\_id>  -Foreign keys: <account\_id**> ,** <assignment\_id**>** |

## 2. Code Packages

*[Provide the package diagram for the system (or sub systems) and package description similar to sample diagram and using description table format below]*

### 1. Code Packages Lecturer

### 1.1 Packages Assignment

#### 1.1.1 Add Assignment

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.1.2 Update Assignment

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.1.3 List Cousre Assignment

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.1.4 Take Assignment

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.1.5 View Assignment Submitted

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 1.2 Packages Subject

#### 1.2.1 Add Subject

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.2.2 Subject manage

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.2.3 Edit Subject

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.2.4 Hiden Subject

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 

### 1.3 package Chapter

#### 1.3.1 Add Chapter



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 

#### 1.3.2 Chapter manage



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.3.3 Edit Chpater



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 1.4 Package Lesson

#### 1.4.1 Add Lesson



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.4.2 Lesson Manage



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.4.3 Edit Lesson

#### 

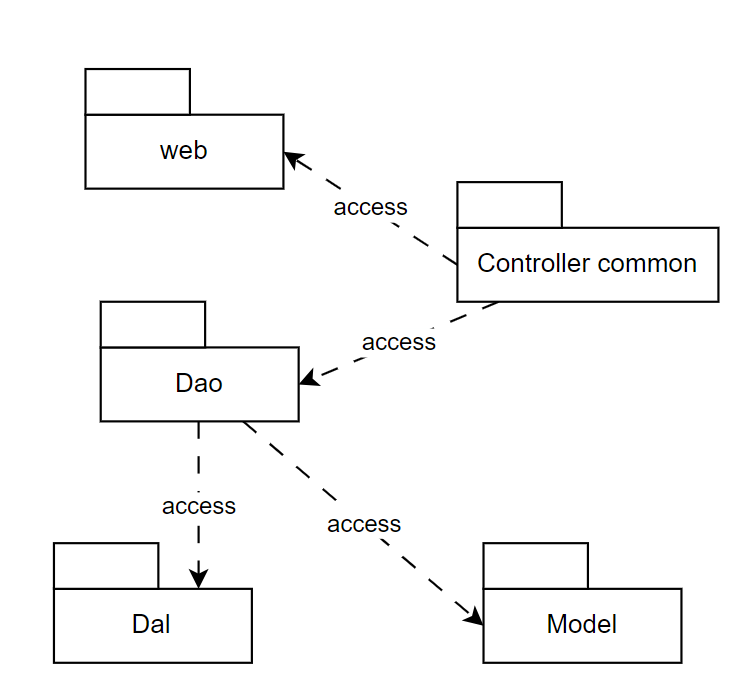
Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 

### 1.5 Package Login and Register

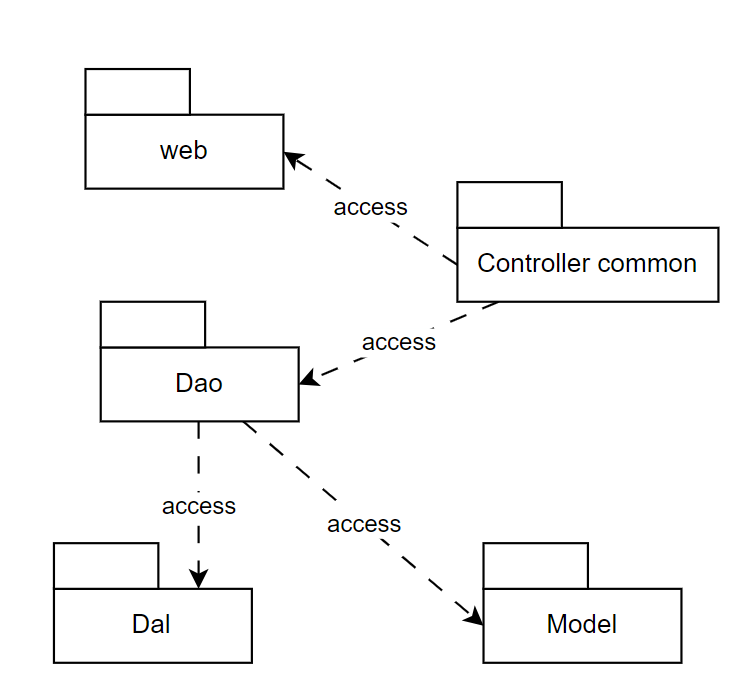
#### 1.5.1 Login



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.5.2 Register



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 1.6 package Admin

#### 1.6.1 Manage Change Role

#### 

Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.6.2 Manage comment report

#### 

Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 

### 1.7 Packages Question

#### 1.7.1 Add Question

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.7.2 Question manage

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.7.3 Edit Question

Package Diagram



Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.7.4 Delete Question

Package Diagram



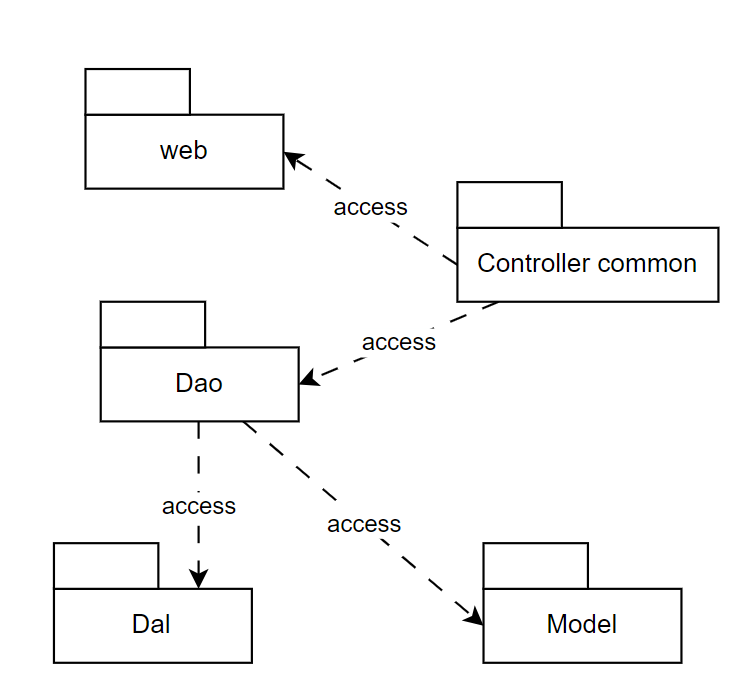
Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 

### 1.8 Packages Course

#### 1.8.1 Course list



#### 

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |

#### 1.8.2 Course details



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.8.3 Add course



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.8.4 Update course



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.8.5 Delete course



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 1.9 Packages Comment

#### 1.9.1 Comment subject



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.9.2 Comment



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 1.10 Packages Quiz

#### 1.10.1 Add Quiz



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.10.2 Edit Quiz



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 1.10.3 View All Quiz Result



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

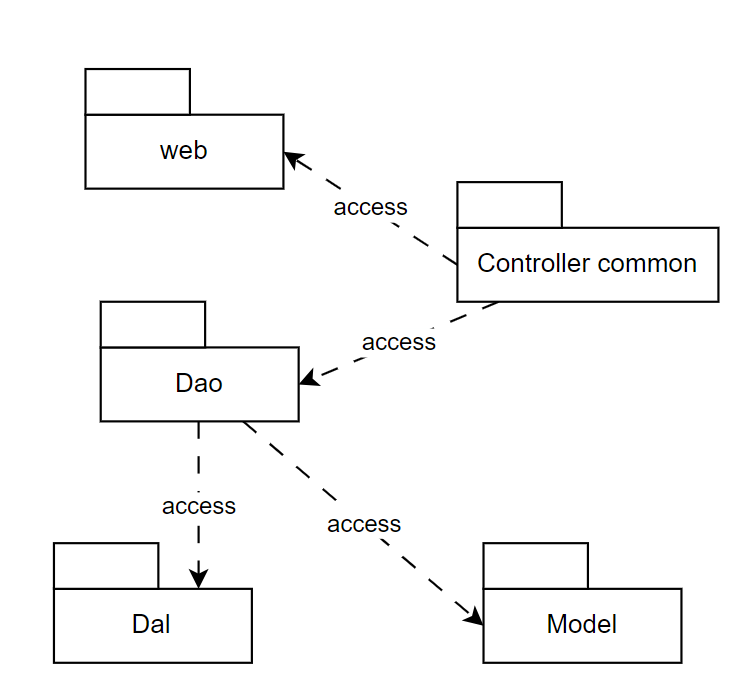
### 

### 2. Code Packages User

### 2.1 Packages Subject

#### 2.1.1 Subject List

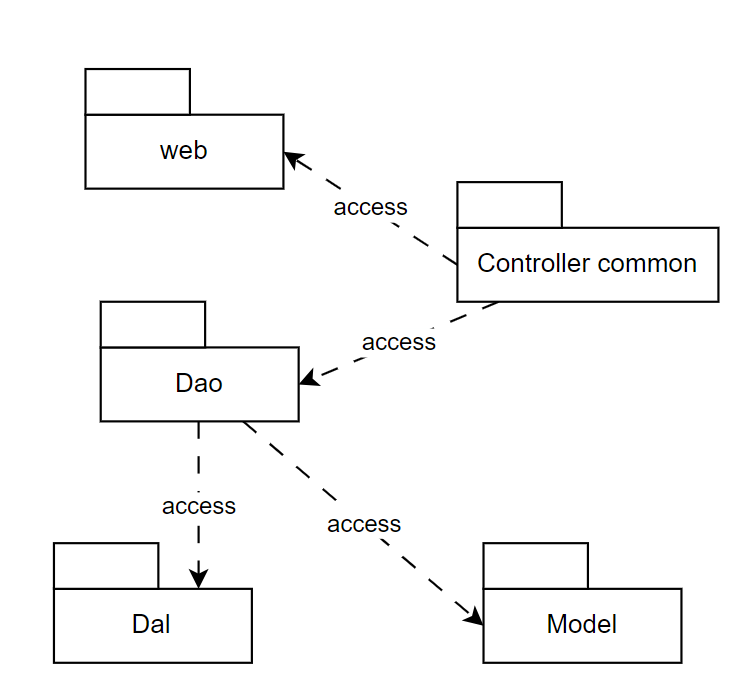
Package Diagram

Package Descriptions

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 2.2 Packages Chapter

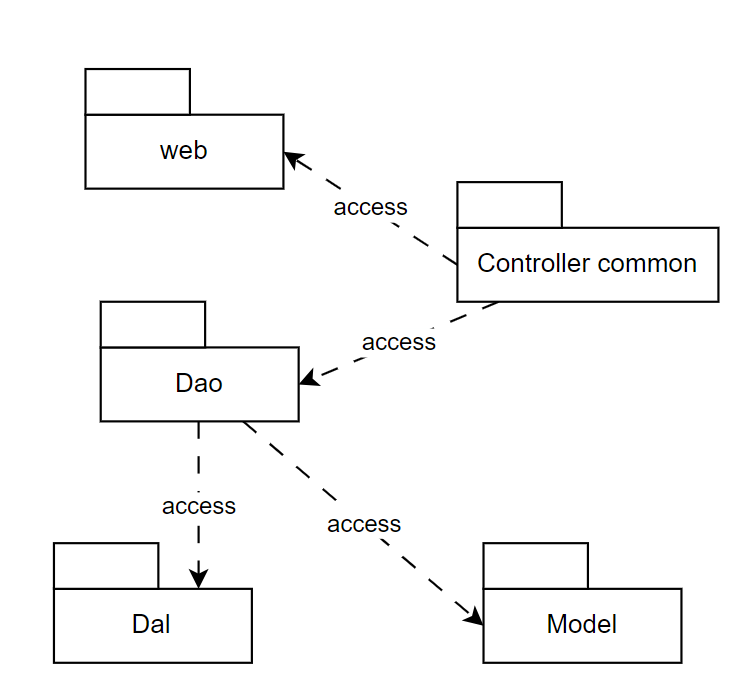
#### 2.2.1 Chapter List

s

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 2.3 Packages Lesson

#### 2.3.1 Lesson List

s

| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 2.4 Packages Role Change Request



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 

### 3. Code Packages Learner

### 3.1 Packages Quiz

#### 3.1.1 Take Quiz



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

#### 3.1.2 View Quiz Result



| **No** | **Package** | **Description** |
| --- | --- | --- |
| 01 | web page | Contain all JSP files providing the UI/UX of the web |
| 02 | controller auth | Contains all process-partitioned servlet controllers |
| 03 | controller common | Contain all servlet controllers handling operational logic of the web |
| 04 | dao | Manage the database |
| 05 | dal | Access the database |
| 06 | model | Represent entities in the database |
| 07 | utils | Some basic utilities uses by other Packages |

### 