

**<<ONLINE LEARNING>>**

**Software Requirement Specification**

– Hanoi, August 2022 –

**Record of changeS**

| **Date** | **A\* M, D** | **In charge** | **Change Description** |
| --- | --- | --- | --- |
| 13/01/2023 | A | All | Create SRS v1.0. |
| 20/01/2023 | M | All | Update SRS v1.1. |
| 01/02/2023 | A | All | Create SRS v2.0. |
| 17/02/2023 | A | All | Create SRS v3.0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

\*A - Added M - Modified D - Deleted

**Table of Contents**

[I. Overview](#_heading=h.30j0zll)

[1. Introduction](#_heading=h.1fob9te)

[2. System Functions](#_heading=h.3znysh7)

[3. Entity Relationship Diagram](#_heading=h.2et92p0)

[II. Functional Requirements](#_heading=h.tyjcwt)

[1.Account Control](#_heading=h.3dy6vkm)

* Manage accounts
* Create account
* View account
* Edit accounts
* View account details

[2.](#_heading=h.2s8eyo1)Expert

* Manage course
* Add/delete course
* View course details
* Edit course details
* Manage lesson
* Create lesson
* Read lesson
* Update lesson
* Delete lesson
* Manage flashcard
* Create flashcard
* Read flashcard
* Update flashcard
* Delete flashcard

3[.](#_heading=h.2s8eyo1) Student

* Log in:
* Reset password
* Change password
* View profile
* Edit profile
* View web information
* View courses
* View blogs
* view certificate
* Enroll courses
* Study
* View
* Select lesson
* Read
* Take flashcard

4[.](#_heading=h.2s8eyo1)  Guest

* Sign up
* View web information
* View courses
* View blogs

5[.](#_heading=h.2s8eyo1)  Course seller

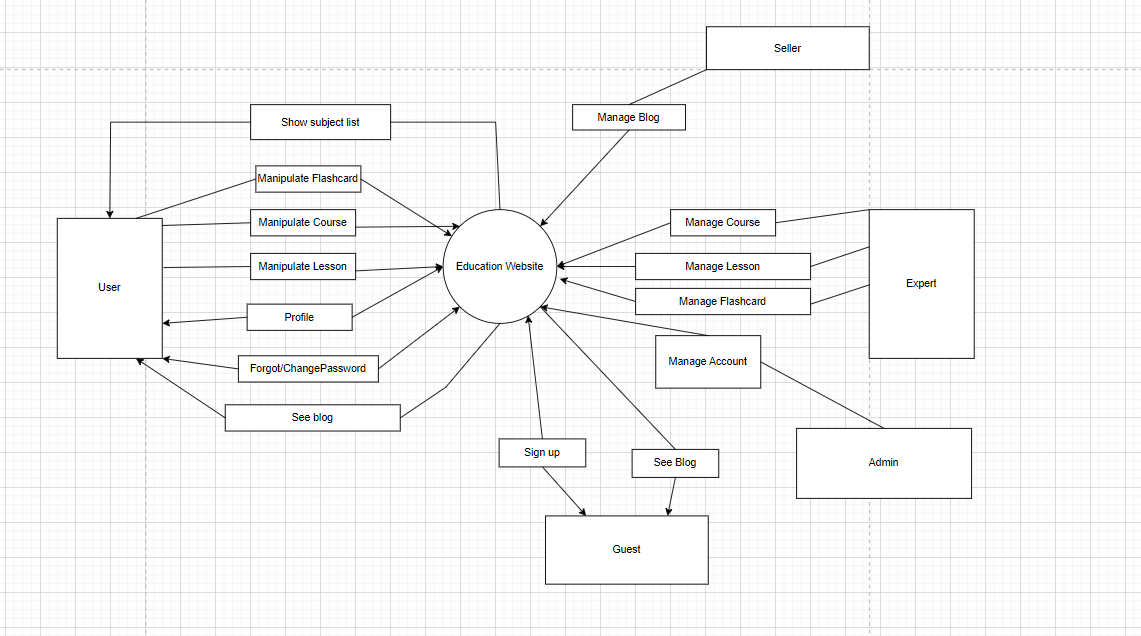
* Manage blogs
* Create blog
* View blog
* Edit Blog
* Manage sliders
* View sliders
* Process enroll

[III.](#_heading=h.30j0zll) User Group

# I. Overview

## 1. Introduction

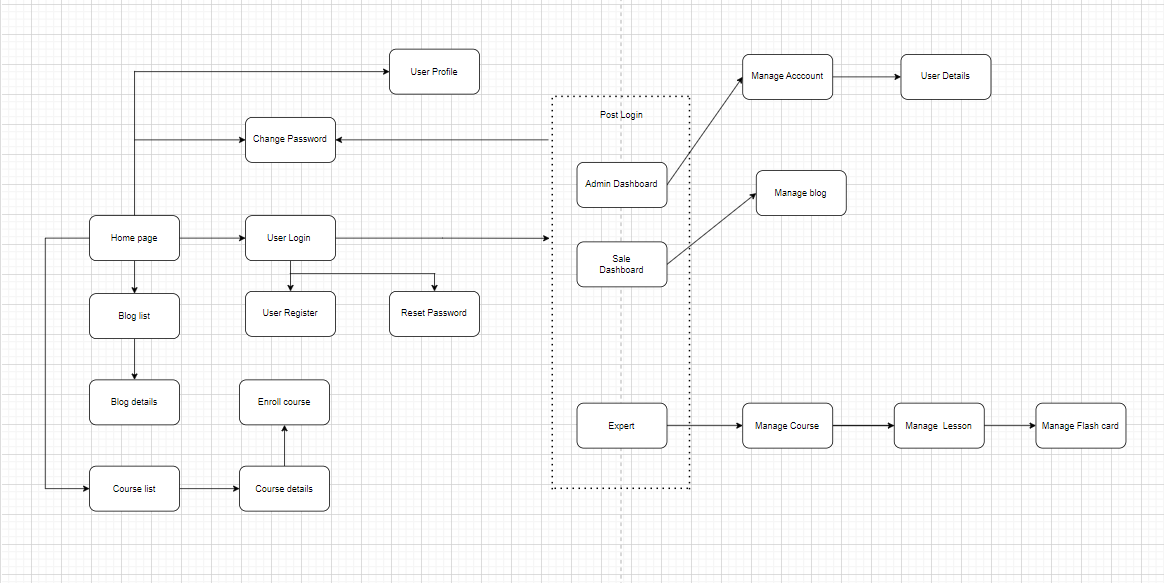
The online learning system is a software system that combines traditional forms of education and modern information technology platforms. The system is expected to grow over several releases, eventually connecting everyone to internet-based lectures and learning methods.



## 2. System Functions

#### a. Screen 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]*

**

#### b. Screen Details

*[Provide the descriptions for the screens in the Screens Flow above]*

| **#** | **Feature** | **Screen/Function** | **Description** |
| --- | --- | --- | --- |
| 1 |  | Home Page |  |
| 2 | Account Control | Change Password | Screen show form so that user can change their password |
| 3 | Account Control | User login | Screen show form of username and password for user to login to the system |
| 4 | Account Control | User register | Screen show registration form so user can register new account by typing all required field |
| 5 | Account Control | User profile | Screen shows information of the user that login and allows the user to update information. |
| 6 | Account Control | Reset password | Screen shows a form which contains username input fields |
| 7 | Account Control | Manage Account | Screen show list account and account control can update, delete account |
| 8 | Account Control’ | User Detail | Screen show information of user |
| 9 | Authorization | Admin Dashboard | Screen showing admin functions when logged in with admin account |
| 10 | Authorization | Sale Dashboard | Screen showing sale functions when logged in with sale account |
| 11 | Authorization | Expert | Screen showing expert functions when logged in with expert account |
| 12 | Course | Course List | Screen show all course |
| 13 | Course | Course Detail | Screen shows course information |
| 14 | Course | Enroll Course | Screen shows a form which user can enroll course to learn |
| 15 | Course | Manage Course | Screen show list course and expert can add,edit,delete course |
| 16 | Lesson | Manage Lesson | Screen show list lesson and expert can add,edit,delete lesson |
| 17 | FlashCard | Manage Flashcard | Screen show list flashcard and expert can add,edit,delete flashcard |
| 18 | Community | Blog List | The screen show the information of the blog |
| 19 | Community | Blog Detail | Screen that show detail of the blog user click on the community board and also all the replies/comment of other user to the blog and allow user to replies on the blog |
| 20 | Community | Manage Blog | the seller can add/edit/delete content of the blog |

#### c. User 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 Role1, Role2,… with the specific system user role names]*

| **Screen** | **Admin** | **Expert** | **Student** | **Seller** |
| --- | --- | --- | --- | --- |
| Home Page | X | X | X | X |
| User login | X | X | X | X |
| User register | X | X | X | X |
| Course list | X | X | X | X |
| User Profile | X | X | X | X |
| Authorization | X |  |  |  |
| Reset password | X | X | X | X |
| Change password | X | X | X | X |
| Manage Course |  | X |  |  |
| Manage Lesson |  | X |  |  |
| Manage Flashcard |  | X |  |  |
| Learn |  |  | X |  |
| Manage Blog |  |  |  | X |
| Notification | X | X | X | X |

In which:

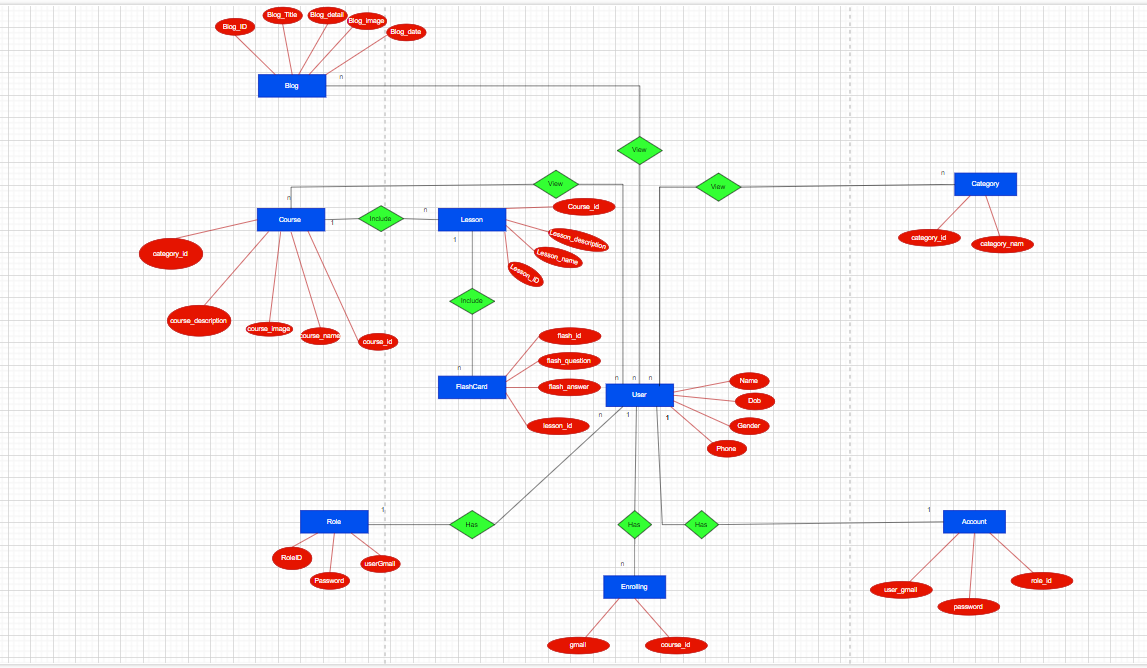
* Admin: The person who manage the website
* Expert: The person who have the right to contribute learning course to the system
* Student: The main customer of the website who can use the main functions of the website
* Seller: The person who manage blog

#### d. Non-Screen Functions

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

| **#** | **Feature** | **System Function** | **Description** |
| --- | --- | --- | --- |
| 1 | Security | Send Mail | When a user requests a reset password , the new password is sent by email. |
|  |  |  |  |

## 3. Entity Relationship Diagram

**

**Entities Description**

| **#** | **Entity** | **Description** |
| --- | --- | --- |
| 1 | Account | a record or profile that represents an individual user or entity's identity within a system or platform. |
| 2 | User | an individual who interacts with a software application, website, or system and has their own account or profile. |
| 3 | Enrolling | the process of registering or signing up for a course, program, or other educational offering to gain access to its materials and assignments. |
| 4 | Role | a set of permissions or privileges that determine what actions a user can perform within a system or platform. |
| 5 | Blog | a website or platform that hosts written content, typically in the form of posts or articles, covering a wide range of topics and purposes. |
| 6 | Course | a structured program of study that typically involves a series of lessons, assignments, and assessments, offered online or in-person. |
| 7 | Category | a grouping or classification of items based on shared characteristics or attributes, often used to organize and categorize content or data. |
| 8 | Lesson | a unit of instruction or learning that is typically part of a larger course or program, presented in a variety of formats. |
| 9 | Flashcard | a small card or piece of paper containing a question or piece of information on one side and an answer on the other, often used as a study tool to memorize or review information in digital or physical formats. |

# II. Functional Requirements

## 1. Account Control

### a.User login

***- Function trigger:***

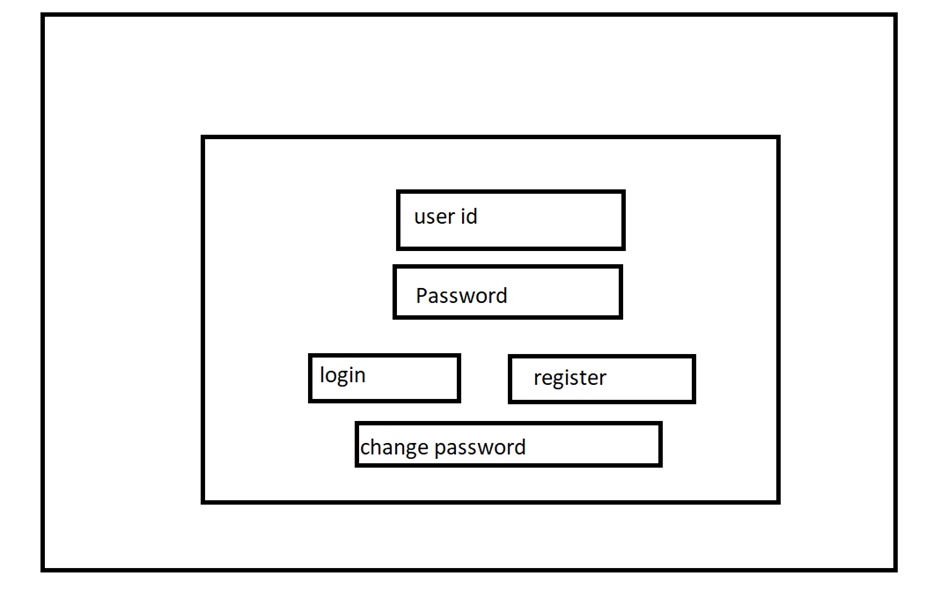
*+ Through navigation path: /login*

***- Function description:***

*+ Actor: All user of the website*

*+ Purpose: Get access to other website content*

*-* ***Screen layout:***

**

***- Function detail:***

*+ Flow:*

*1. User get to the login page by opening the website when they haven’t logged in yet or by signing out.*

*2. User enter user ID.*

*3. User enter password*

*4. Click the login button.*

*5. If user ID or password do not match, denied access and go back to step 1 with error message*

*6. If the user enter correct user ID and password, redirect to the home page.*

### b. Register

**- Function trigger:**

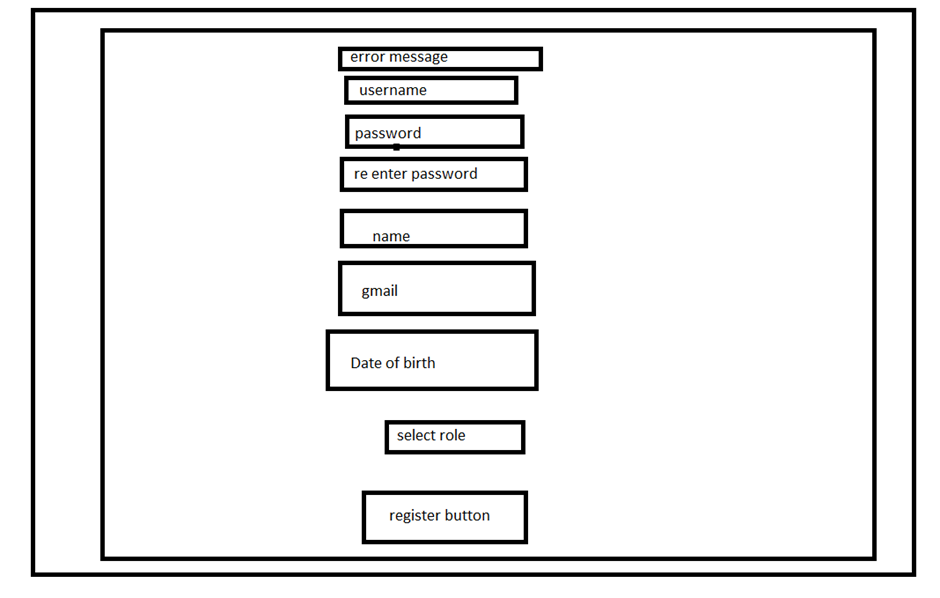
+ Through navigation path: /register

**- Function description:**

+ Actor: User

+ Purpose: To create a account to be able to login to the website

**- Screen layout:**



### - Function detail:

### + Flow:

### 1. User get to register page .

### 2. User enter User information required for registration in registration form.

### 3. Click register button.

### 4. If username or password already exist, go back to step 1 with error message

### 5. If user don’t fill the required information or not follow the data constraint, go back to step 1 with error message.

### 6. If register successful, redirect user to login page.

### + Data required:

### 1. Username

### 2. Password: must be at least 8 in length and include 1 or more lowercase letter, uppercase letter and number

### 3. Name: not empty

### 4. DOB

### 5. Gmail must be in correct gmail form

### 6. Gender

### 7. Role: Student or course creator.

### C. Change password

**- Function trigger:**

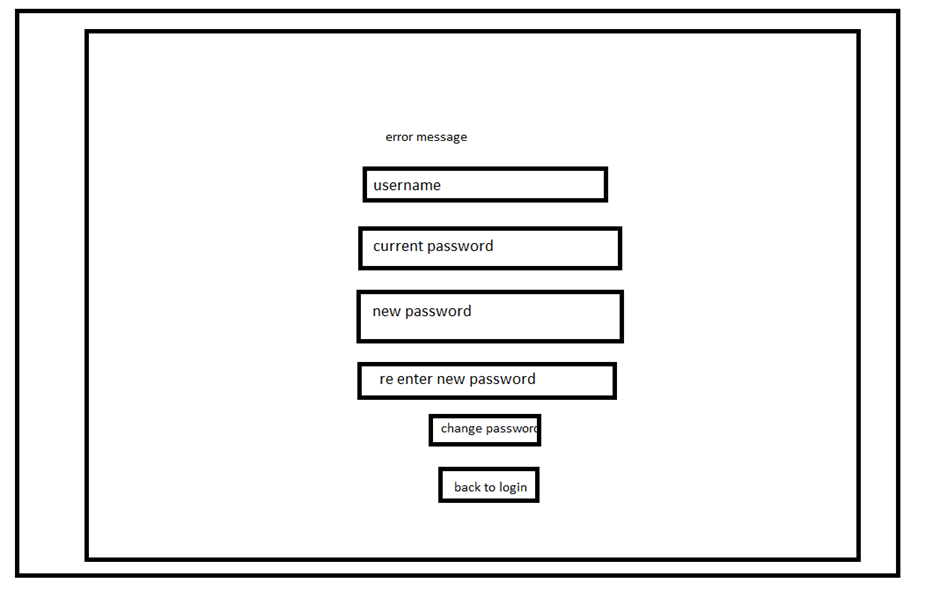
+ Through navigation path: /chgpwd

**- Function description:**

+ Actor: User

+ Purpose: To change password of the account for security reason.

**- Screen layout:**



**- Function detail:**

+ Flow:

1. User get to the change password page

2. User enter Username

3. User enter password

4. User enter new password

5. User enter retype new password.

6. Click on change password button.

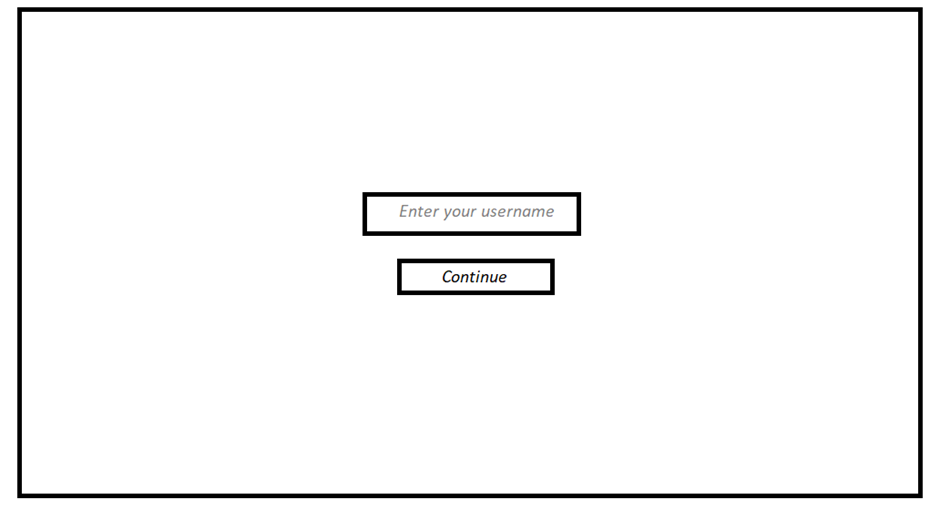
7. If user ID or password do not match, go back to step 1 with error message

8. If retype new password do not match the new password, go back to step 1 with error message.

9. If user change password successfully, log out and redirect user to log in page.

### d. Reset Password

**- Screen layout:**



**- Function trigger:**

+ Through navigation path “/resetpassword”

+ By click on a button “reset password” in login page

**- Function description:**

+ Actor: User

+Purpose: to reset account ‘s password

**- Function detail:**

+Flow:

User clicks on “reset password” button in login page

Enter email which account need to reset

Click on continue button

If the email is valid ,user must enter email registered,otherwise the user must re-enter.

Click on continue button

Current password will be changed to six random characters and the new password will be sent to the email registered previously,otherwise the user must re-enter.

**2. *Admin***

. **a. Manager user**

**- Function trigger:**

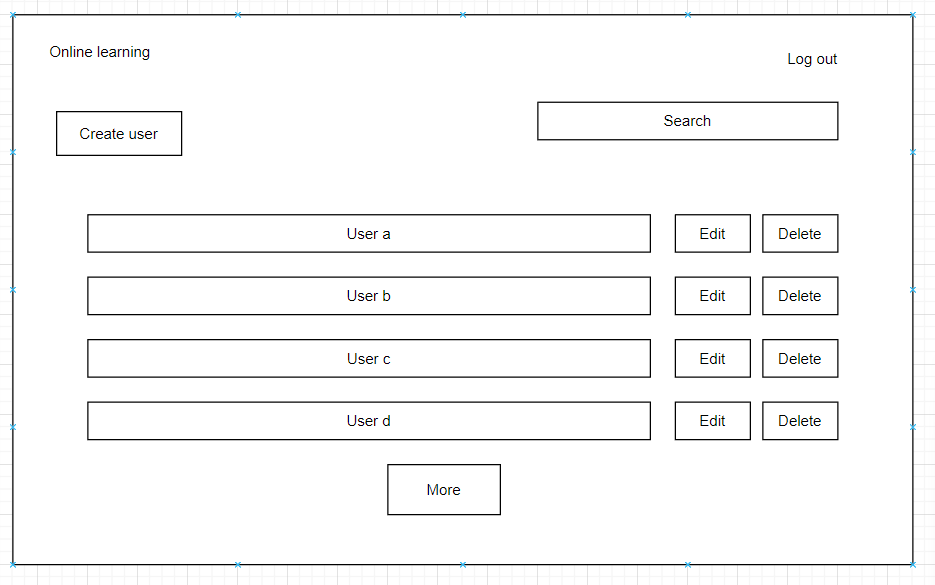
+ Through navigation path: /admin

**- Function description:**

+ Actor: Administrator

+ Purpose: To manage user account

- **ScreenLayout:**

****

**Function details:**

+More: Show more user’s account

+Create user: create new user’s account

+delete: delete user’s account

+Edit: Edit user’s account details

### c. Edit user

**- Function trigger:**

+ Through navigation path: /admin/edituser

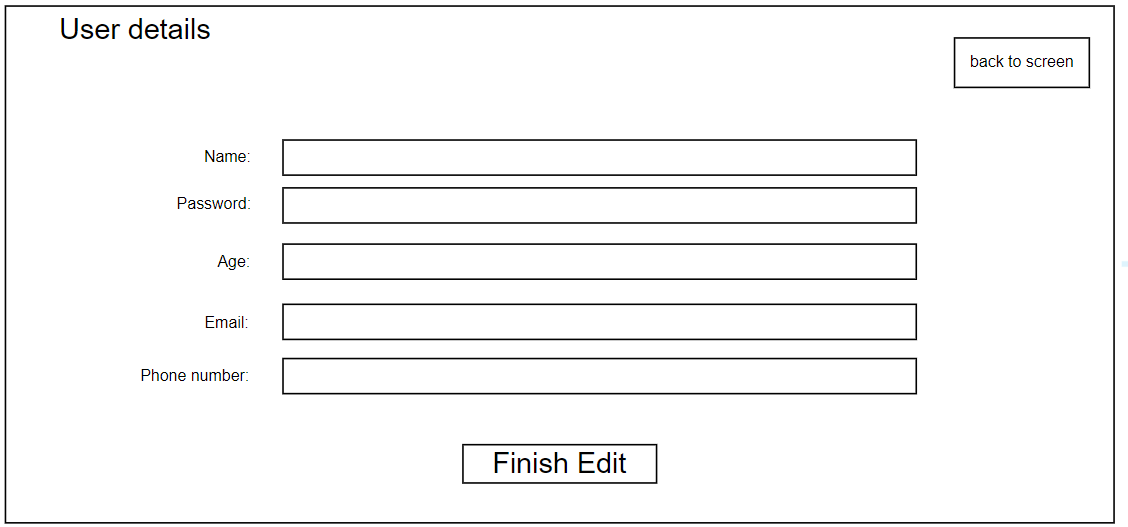
+ By click on a button “ Edit” in manage admin screen

**- Function description:**

+ Actor: Admin

+ Purpose: To show list and manage user as a admin

- **ScreenLayout:**



**- Function details:**

+Name: Edit user name

+Password: Edit user password

+Age:Edit user age

+Email: Edit user email

+Phone number: Edit user phone number

## 

## 

## 

## 3. *Expert*

### a. Manage course

**- Function trigger:**

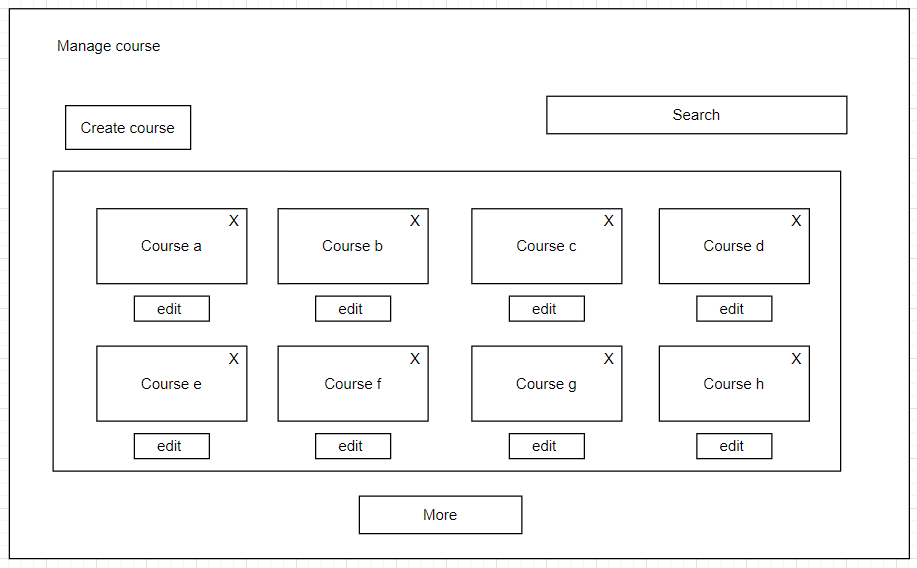
+ Through navigation path: /course

**- Function description:**

+ Actor:Expert

+ Purpose: To manage course as a course creator

- **ScreenLayout:**

****

### b. Create Course

**- Function trigger:**

+ Through navigation path: /course/crcourse

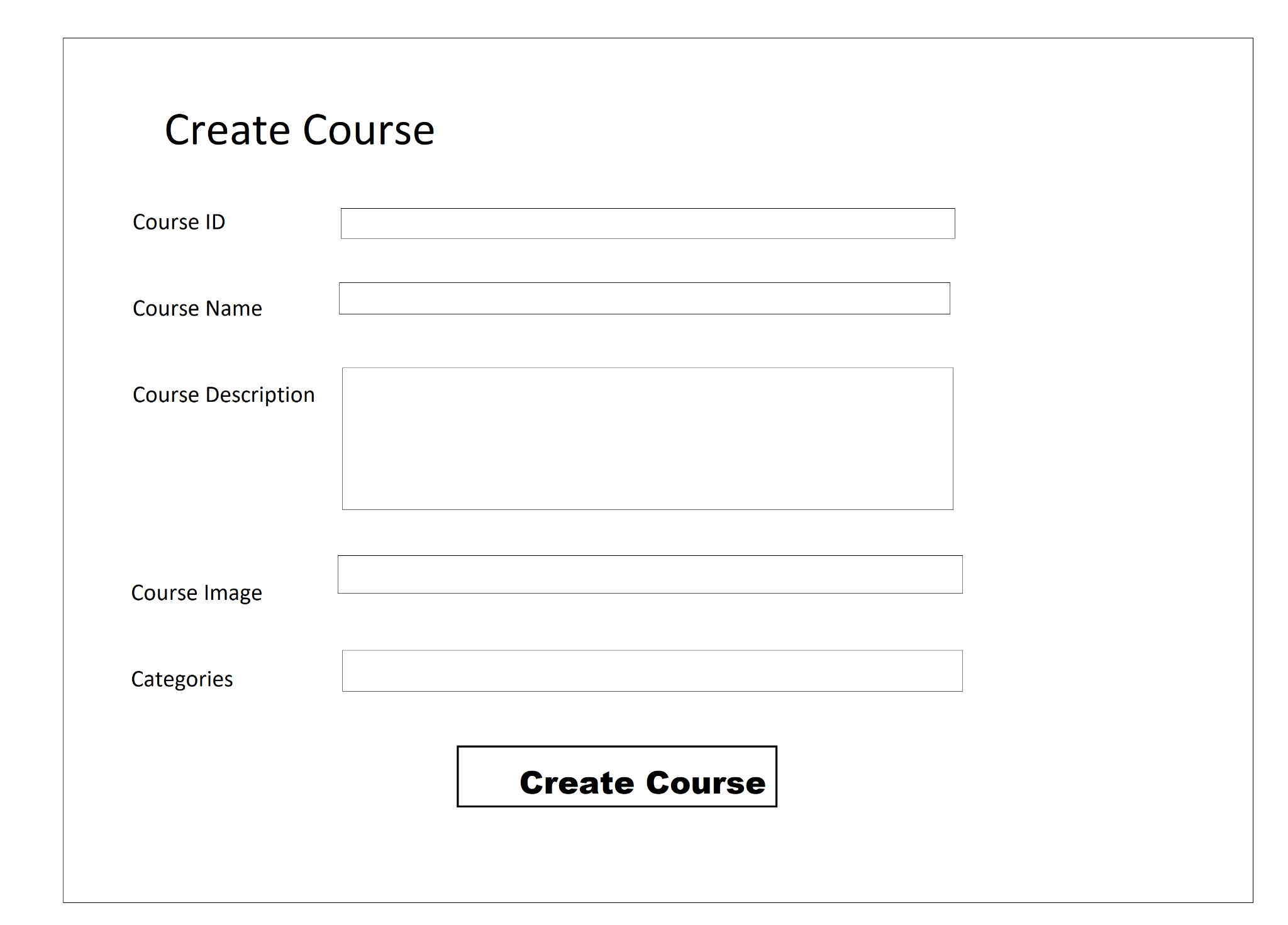
+ By click on a button “ create” in manage course screen

**- Function description:**

+ Actor: Expert

+ Purpose: To create course as a expert

- **ScreenLayout:**

****

**- Function details:**

+Course id: enter course id

+Course name: enter course name

+Course description : enter course description

+Course image: enter url image

+Categories: enter Categories

### c. Edit Course

**- Function trigger:**

+ Through navigation path: /course/mgcourse

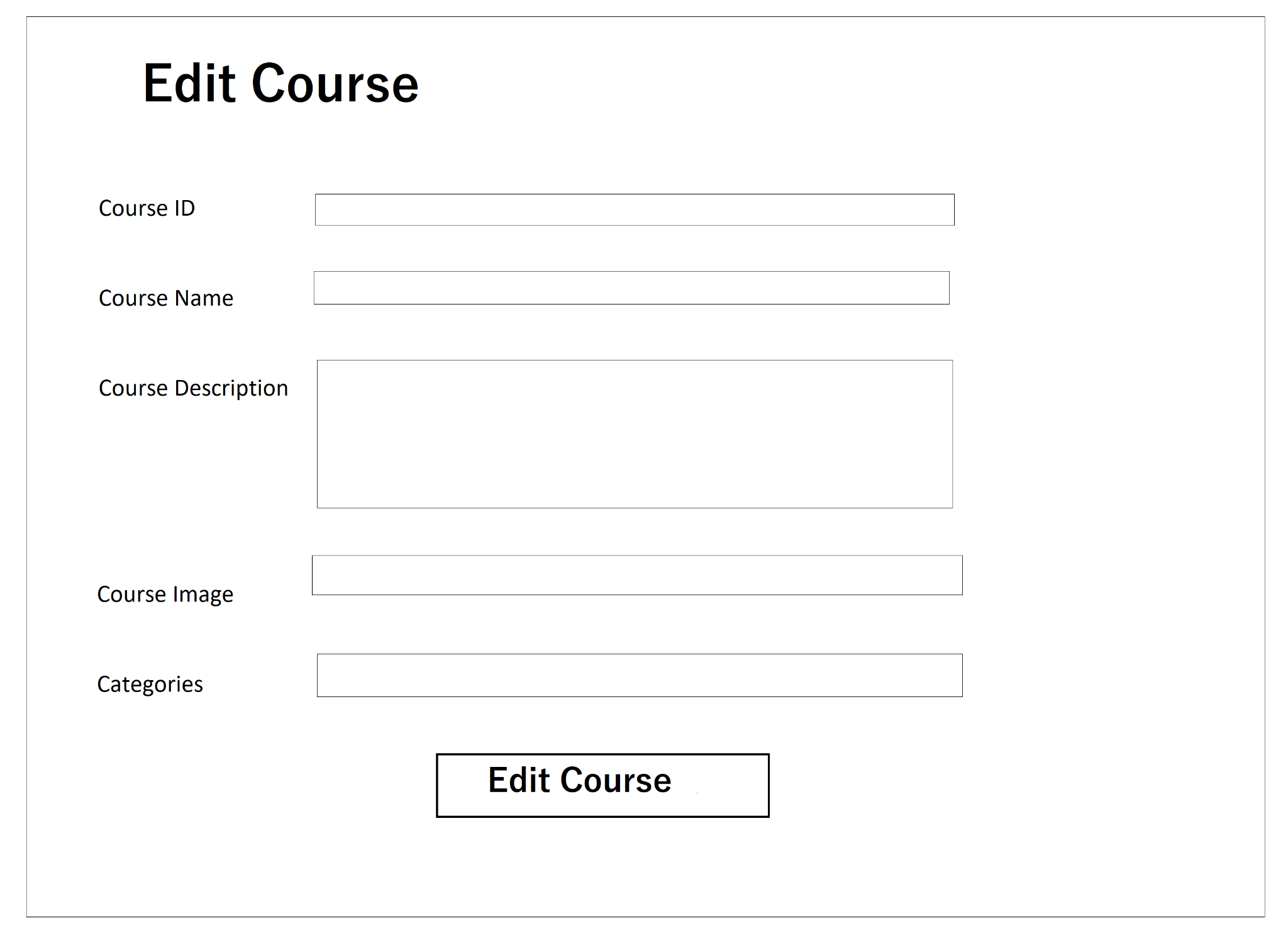
+ By click on a button “ edit” in manage course screen

**- Function description:**

+ Actor: Expert

+ Purpose: To show list and manage course as a expert

- **ScreenLayout:**

****

**- Function details:**

+Course id: enter course id

+Course name: enter course name

+Course description : enter course description

+Course image: enter url image

+Categories: enter Categories

### d. Manage lesson

**- Function trigger:**

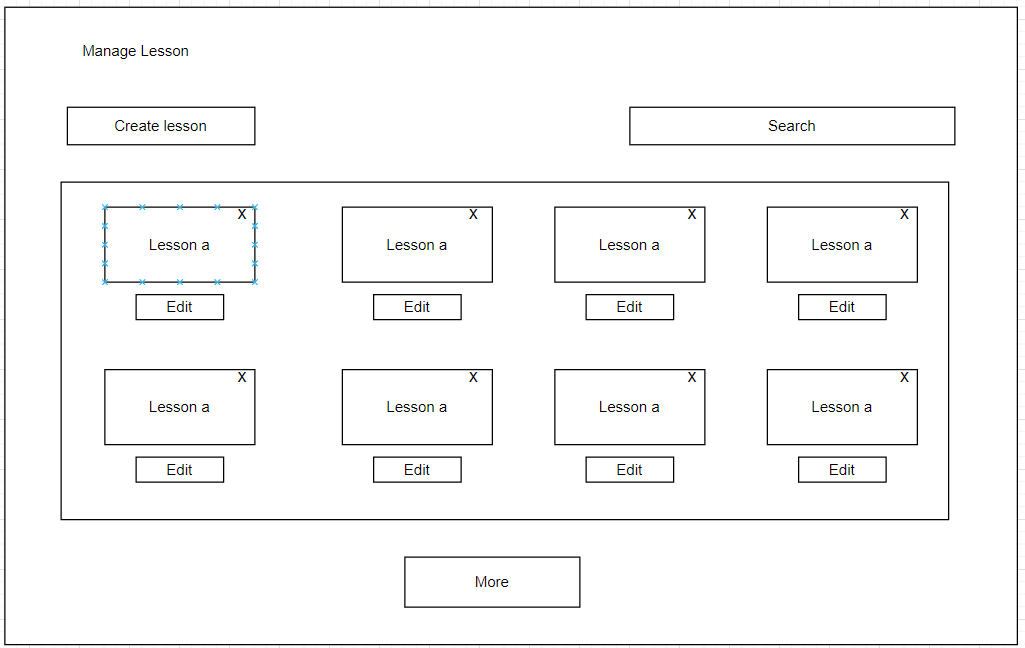
+ Through navigation path: /managelesson

**- Function description:**

+ Actor: Expert

+ Purpose: To manage lesson as a expert

- **ScreenLayout:**

****

### 

### 

### e. Create lesson

* **Function trigger:**

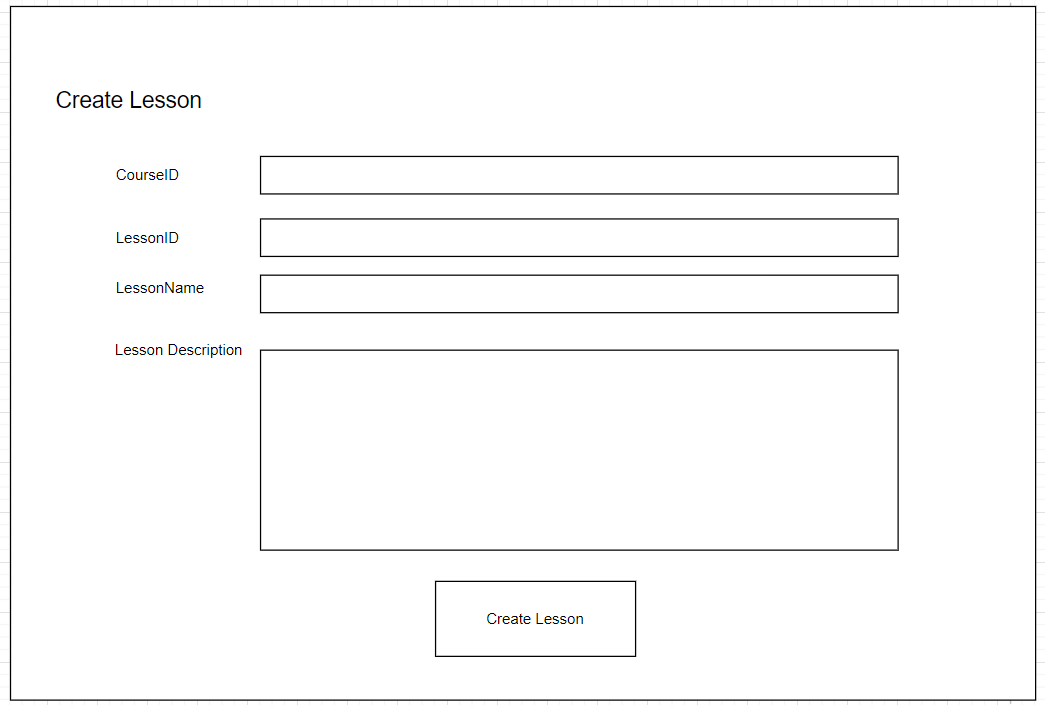
+ Through navigation path: /createlesson

**- Function description:**

+ Actor: Expert

+ Purpose: To manage lesson as a expert

- **ScreenLayout:**

****

**- Function details:**

+LessonID: enter lesson ID

+CourseID: enter course ID

+LessonName : enter lesson name

+LessonDescription : enter lesson description

### f. Edit lesson

* **Function trigger:**

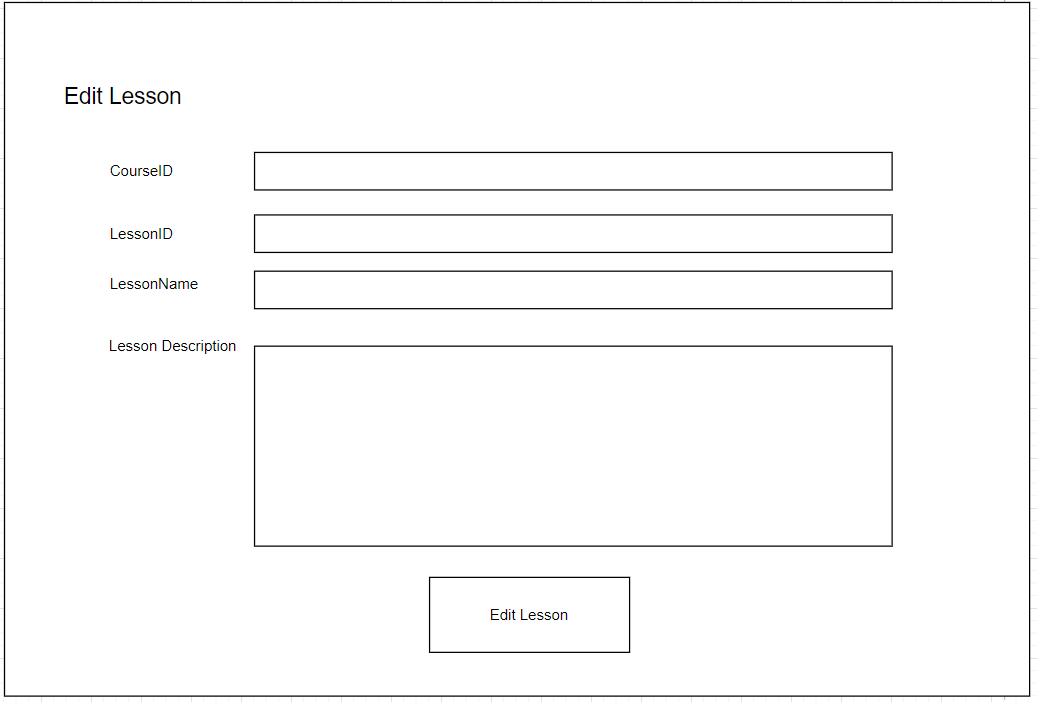
+ Through navigation path: /editlesson

**- Function description:**

+ Actor: Expert

+ Purpose: To manage lesson as a expert

- **ScreenLayout:**

****

**- Function details:**

+LessonID: enter lesson ID

+CourseID: enter course ID

+LessonName : enter lesson name

+LessonDescription : enter lesson description

### g. Manage flashcard

* **Function trigger:**

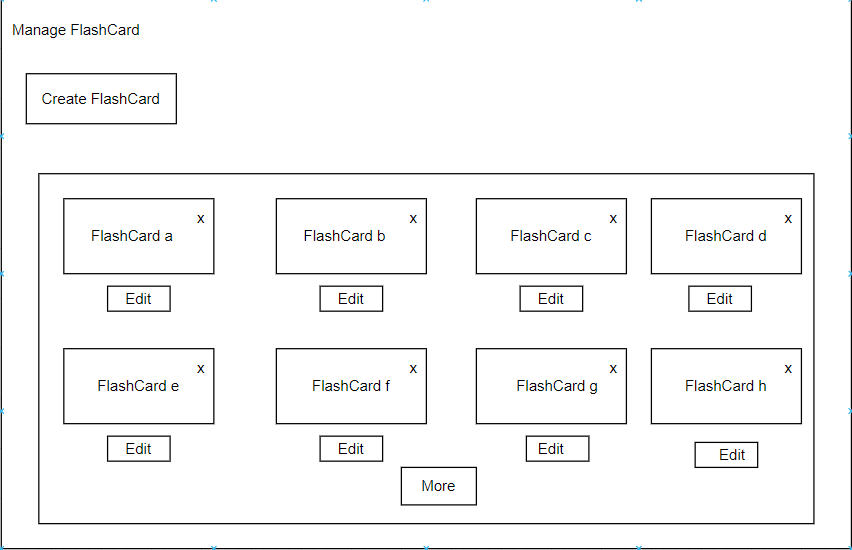
+ Through navigation path: /manageflashcard

**- Function description:**

+ Actor: Expert

+ Purpose: To manage flashcard as a expert

- **ScreenLayout:**

****

### h. Create flashcard

* **Function trigger:**

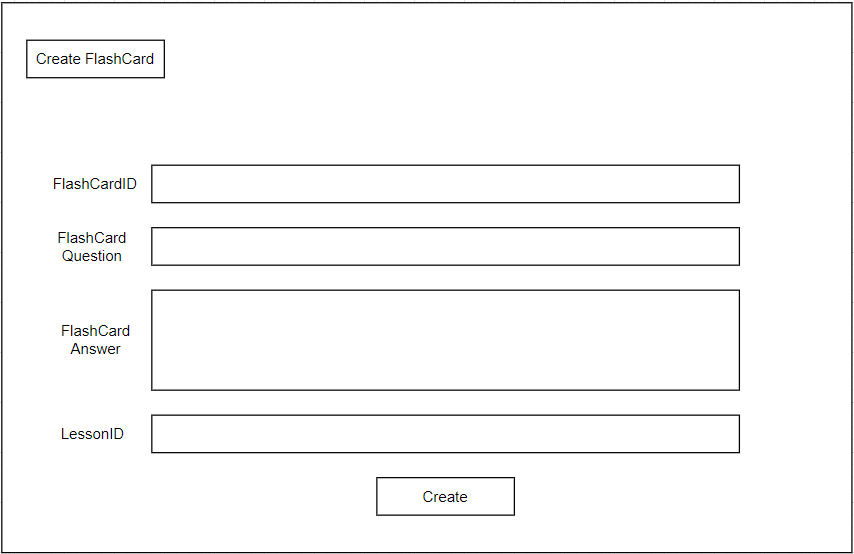
+ Through navigation path: /createflashcard

**- Function description:**

+ Actor: Expert

+ Purpose: To manage flashcard as a expert

- **ScreenLayout:**

****

**- Function details:**

+FlashCardID: enter flashcard ID

+FlashCard Question: enter flashcard question

+ FlashCard Answer : enter flashcard answer

+LessonID : enter lesson ID

### j. Edit flashcard

* **Function trigger:**

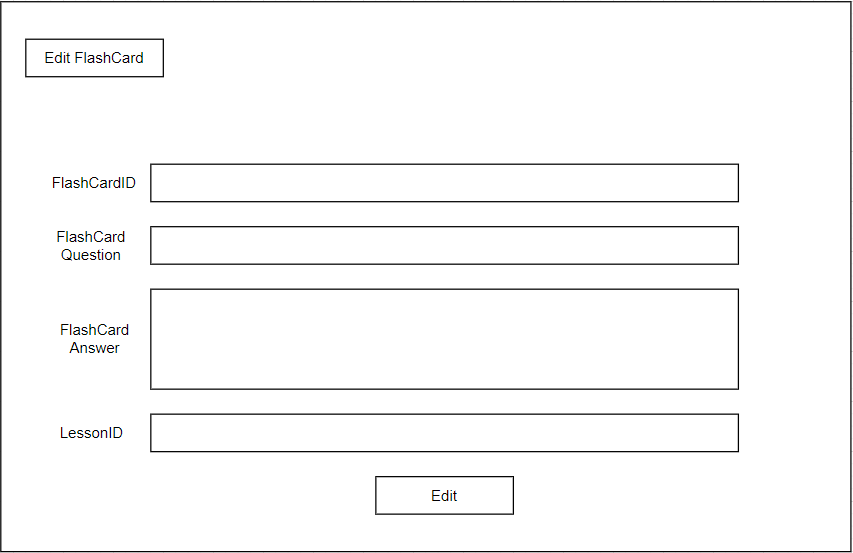
+ Through navigation path: /editflashcard

**- Function description:**

+ Actor: Expert

+ Purpose: To manage flashcard as a expert

- **ScreenLayout:**

****

**- Function details:**

+FlashCardID: enter flashcard ID

+FlashCard Question: enter flashcard question

+ FlashCard Answer : enter flashcard answer

+LessonID : enter lesson ID

## 4. *Student*

### a. Enroll course

**- Function trigger:**

+ Through navigation path: /course

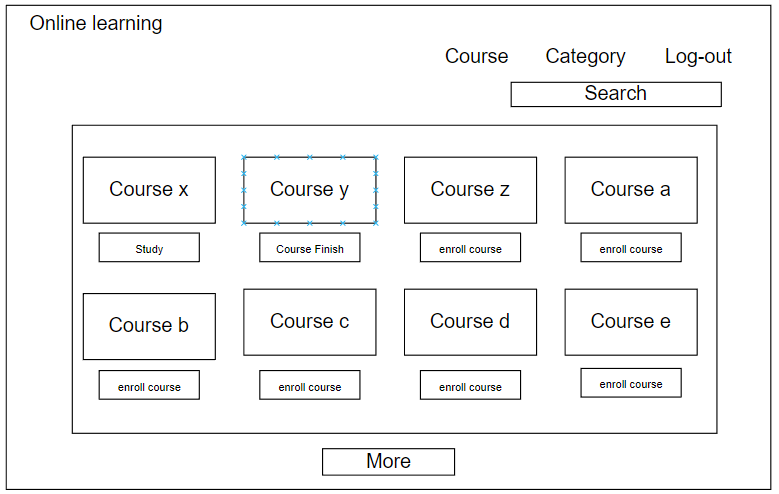
+ By click on a button “course” in home page

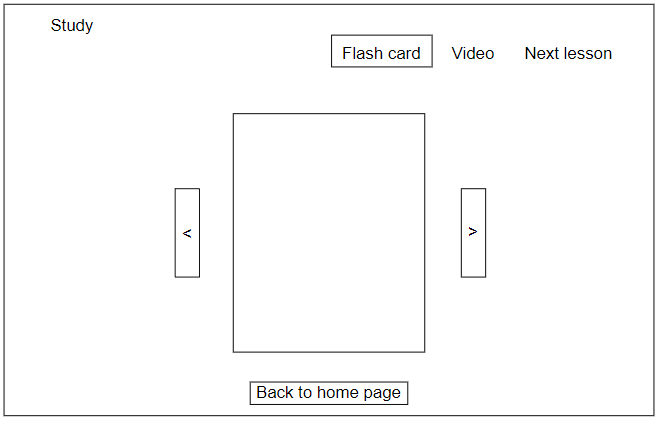
**- Function description:**

+ Actor: User

+ Purpose: To show list and enroll course as a student

- **ScreenLayout:**



`

**- Function description:**

+Course screen

* Course: Show list of courses to view or enroll
* Category: Show different course
* Log-out: Log-out of account
* More: Show more course

+Study screen

* Flash card: Learn by Flash card
* Next lesson: Next lesson of course
* Back to home page: Go back course screen

## 5. *Seller*

### a.Manage Blog

**- Function trigger:**

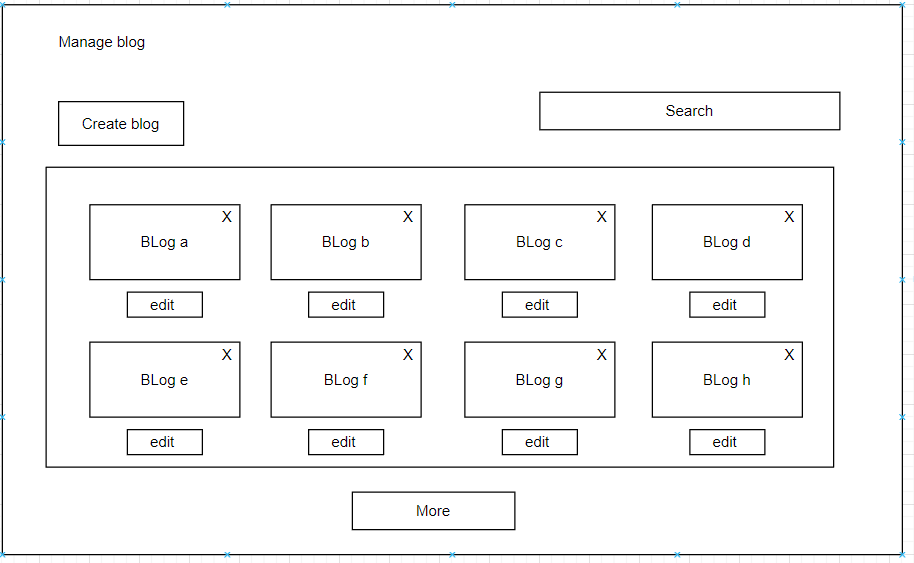
+ Through navigation path: /blog

**- Function description:**

+ Actor: Seller

+ Purpose: To show list and manage blog as a seller

- **ScreenLayout:**

****

**- Function details:**

+Create blog: create new blog

+X: delete blog

+Edit: Edit blog details

### b. Create Blog

**- Function trigger:**

+ Through navigation path: /blog/createblog

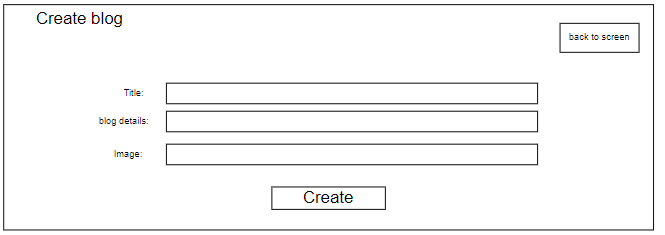
+ By click on a button “ create blog” in manage blog screen

**- Function description:**

+ Actor: Seller

+ Purpose: To create blog as a seller

- **ScreenLayout:**

****

**- Function details:**

+Blog Title: enter blog name

+Blog details: enter blog details

+Blog Image: enter url image

+ Create :

+Back to screen: go back to screen

### 

### 

### c. Edit Blog

**- Function trigger:**

+ Through navigation path: /blog/editblog

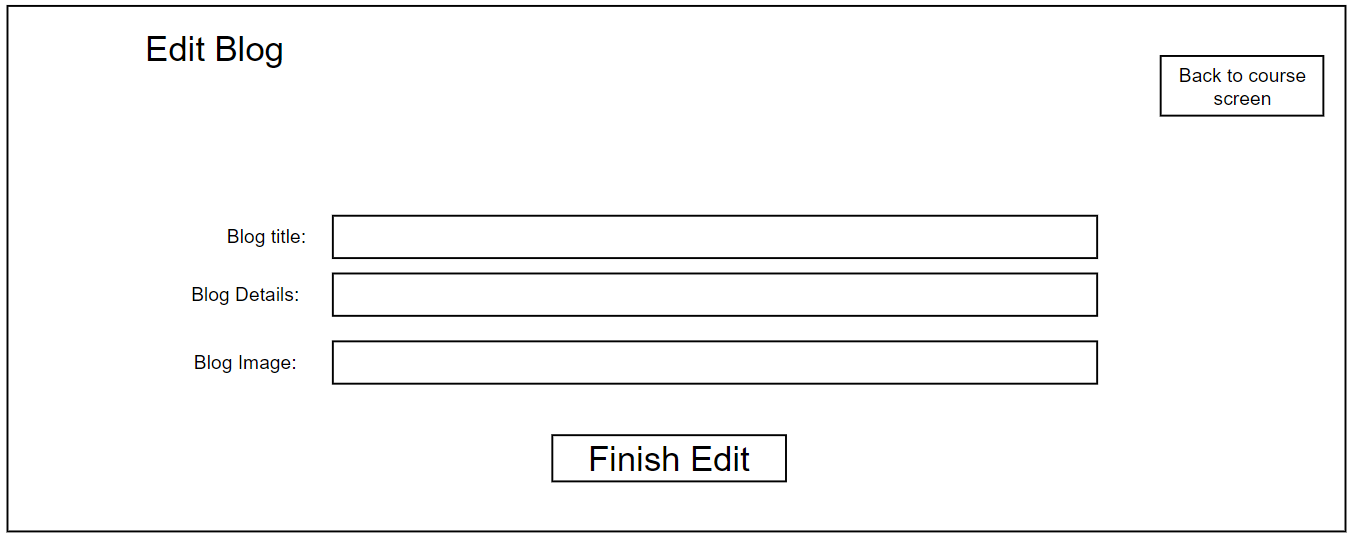
+ By click on a button “ edit” in manage blog screen

**- Function description:**

+ Actor: User

+ Purpose: To edit blog as a seller

- **ScreenLayout:**

****

**- Function details:**

* Back to course screen: back to seller screen
* Finish edit: save edit

# II. USER GROUP

[1.Account Control](#_heading=h.3dy6vkm)

* Manage accounts
* Create account
* View account
* Edit accounts
* View account details

[2.](#_heading=h.2s8eyo1)Expert

* Manage course
* Add/delete course
* View course details
* Edit course details
* Manage lesson
* Create lesson
* Read lesson
* Update lesson
* Delete lesson
* Manage flashcard
* Create flashcard
* Read flashcard
* Update flashcard
* Delete flashcard

3[.](#_heading=h.2s8eyo1) Student

* Log in:
* Reset password
* Change password
* View profile
* Edit profile
* View web information
* View courses
* View blogs
* view certificate
* Enroll courses
* Study
* View
* Select lesson
* Read
* Take flashcard

4[.](#_heading=h.2s8eyo1)  Guest

* Sign up
* View web information
* View courses
* View blogs

5[.](#_heading=h.2s8eyo1)  Course seller

* Manage blogs
* Create blog
* View blog
* Edit Blog
* Manage sliders
* View sliders
* Process enroll