## Exam Preparation – Video Tutorials

## Application Overview

Get familiar with the provided **HTML & CSS** and create an application for **video tutorials.**

## Functional Requirements

The Functionality Requirements describe the functionality that the **application** must support.

**Guest (Not Logged In)**

**Guest** navigation example:



The **application** should provide Guest (not logged in) users with the functionality to login, register, **view** the Home page and **details page of all listed courses**.

**Users (Logged In)**

**User** navigation example:

****

The **application** should provide **Users** (logged in) with the functionality to **view the Home page**, **course details page of all listed courses,** and **they should be able to create courses, enroll in courses**. Courses can be **public** or **not.** Users and Guests can access **only** **public courses.**

## Database Models (10 Pts)

The **Database** of the Tutorials application needs to support **2 entities**

### User

* Username - string (required),
* Password - string (required),
* **Enrolled Courses - a collection of** Courses

### Course

* **Title - string (required),**
* **Description - string (required),**
* **Image Url - string (required),**
* **Is Public - boolean, default - false,**
* **Created at – Date or String, required,**
* **Users Enrolled - a collection of Users**

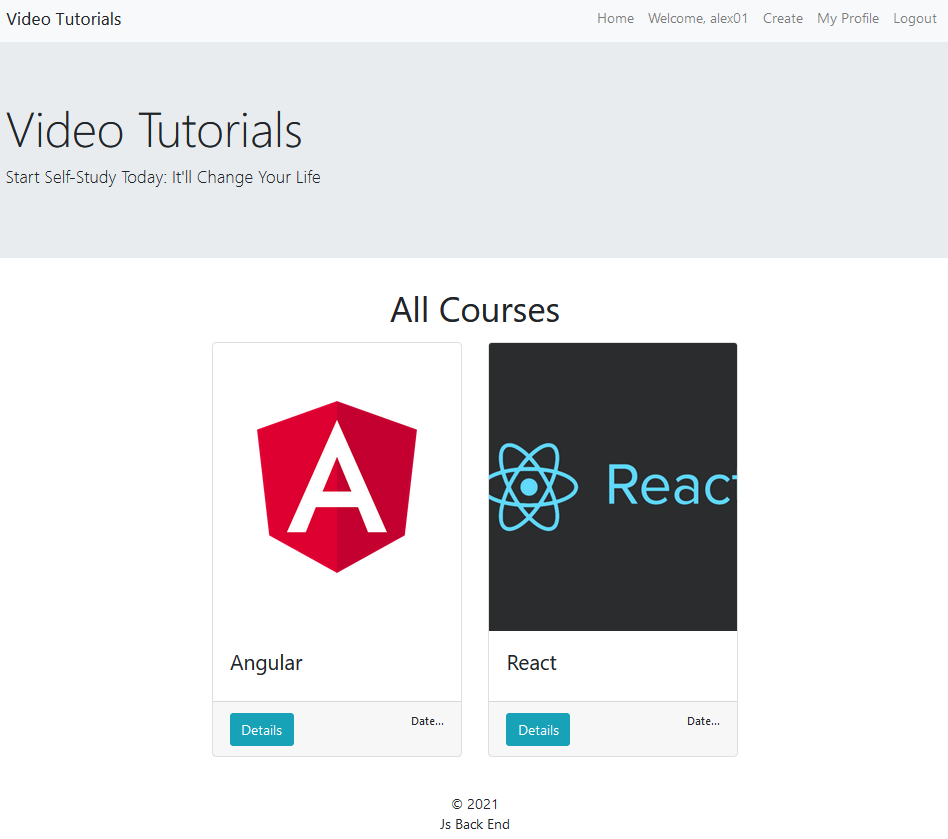
**Note:** When a user **enrolls** in a course, their **id** is added to that collection (**Users Enrolled**)

Implement the entities with the correct data types.

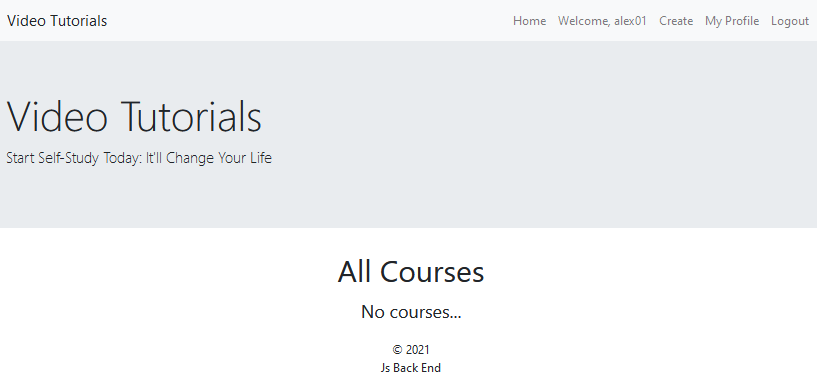
## Application Pages

**Home Page (For logged in users and logged-out users)**

List all public courses. Each **course** must show information about the **image**, **title**, **date** as well as a page with details about the specific course with a button [**Details**].

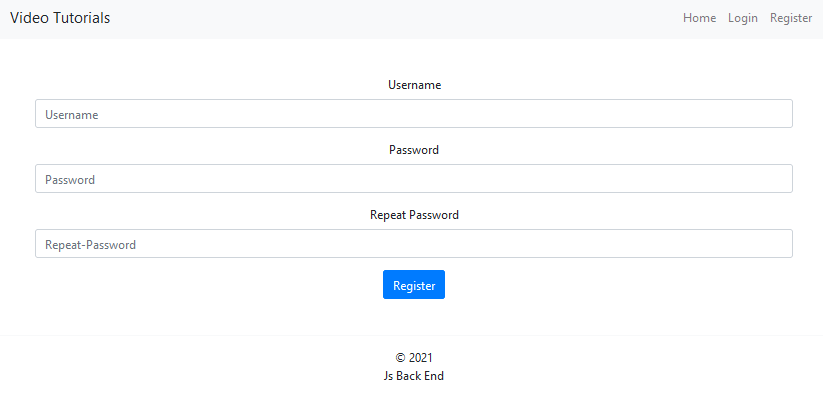


If there are no courses in the database yet, display **"No courses..."**



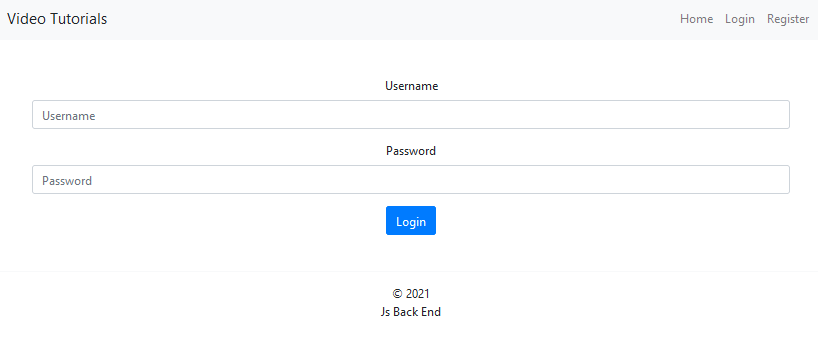
**Register Page (Logged Out User)**

**Register** a user inside the database with **username, and password**. **Password** inside the **database** must be hashed (**use bcrypt**) and both passwords **must match**! After successful registration, you should **redirect** to **Home page**, **with an already logged in user**.



**Login Page (Logged Out User)**

Logging an already registered user with the correct **username** and **password**. After successful login, you should **redirect to Home page.**

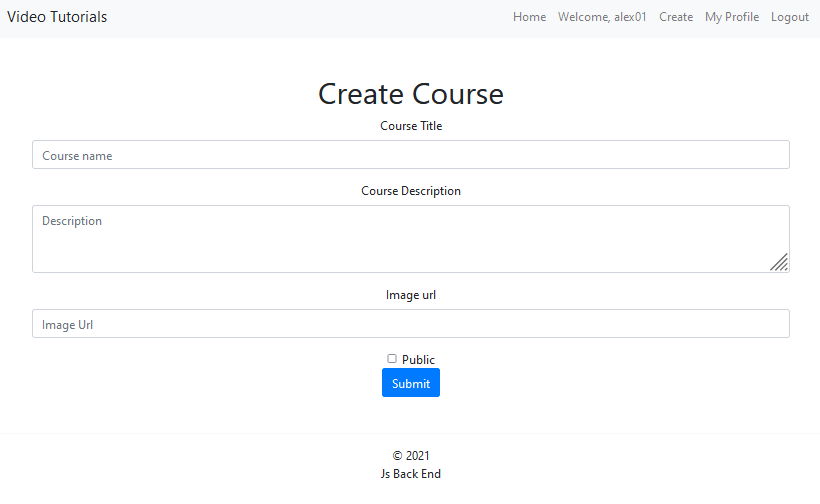


#### Logout (logged in user)

The logout action is available to **logged-in** users. Upon success, clear any session information and **redirect** the user to the **Home** page.

**Create Course Page (Logged in User)**

Enter course **title**, **description**, **image URL** and choose if the course will be **public** or not. You will receive the value of the checkbox as a string **"on"** if it's checked or **undefined** if it's not. You have to convert the received value as Boolean **true** or **false** and save the course in the database.

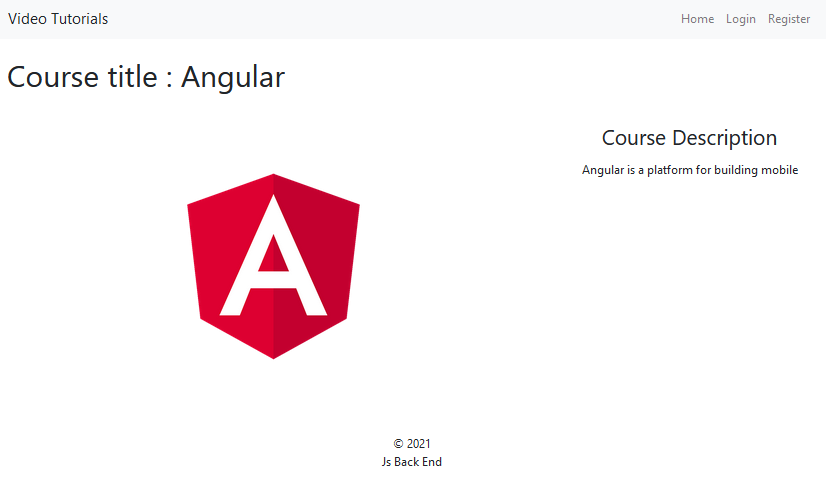


**Details Page - (for logged in users and logged out users)**

All users should be able **to see details** of the course. In the Details of a course section, you should load the course **title, image** and **description**. If the currently registered user **is the creator** of the course, the **Edit** and **Delete** buttons should be displayed, otherwise they **should not be available**.

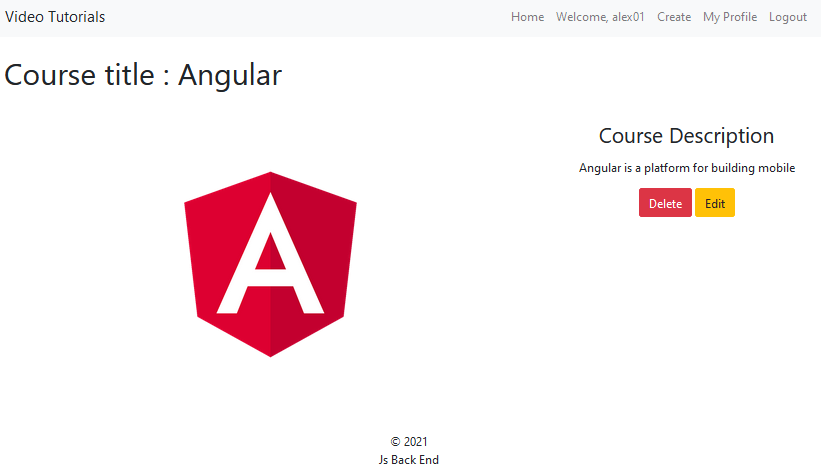
#### Details Page (logged out users)

If there are **no logged** in user, **no buttons** should be displayed.



#### Details Page (logged in user and creator of the current course)

The creator of the course should see **[Delete] [Edit].**

****

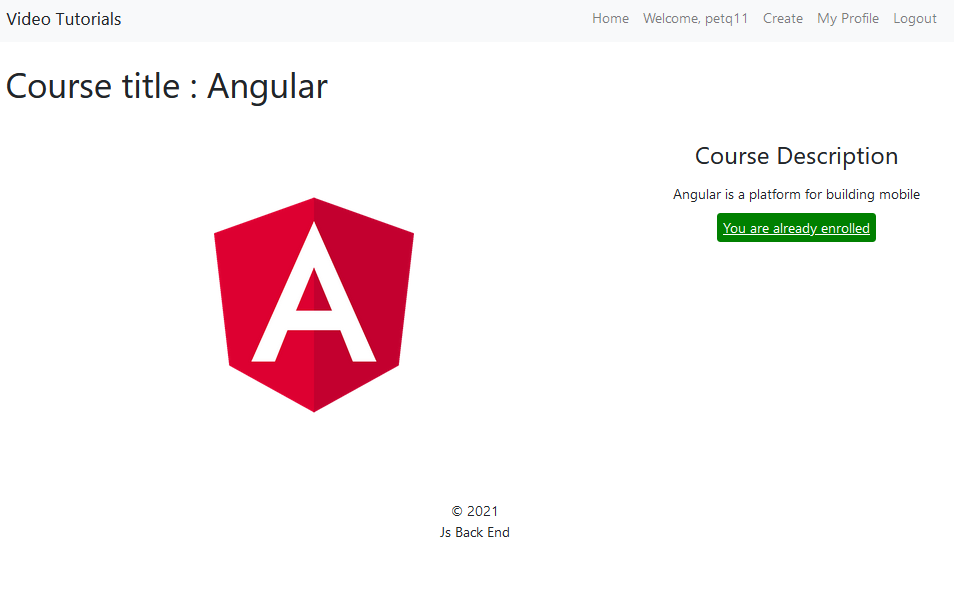
**Details Page (logged in user that is not enrolled in course)**

If the user is **not enrolled** in the course, you should display **[Enroll] button.**

****

**Details Page (logged in user that is enrolled in course)**

**After** the user has enrolled in the course, the **"You are already enrolled" span element** should be **shown instead of the [Enroll] button.**



#### Enroll course (logged in user which is not the current creator)

Any registered user who is **not** the current creator of the course must be able **to enroll in the course**.

If he manages to **enroll**, his **userId** must be added to the collection of **Users Enrolled**, and **redirect** the user to the **Details** page for the current **course**.

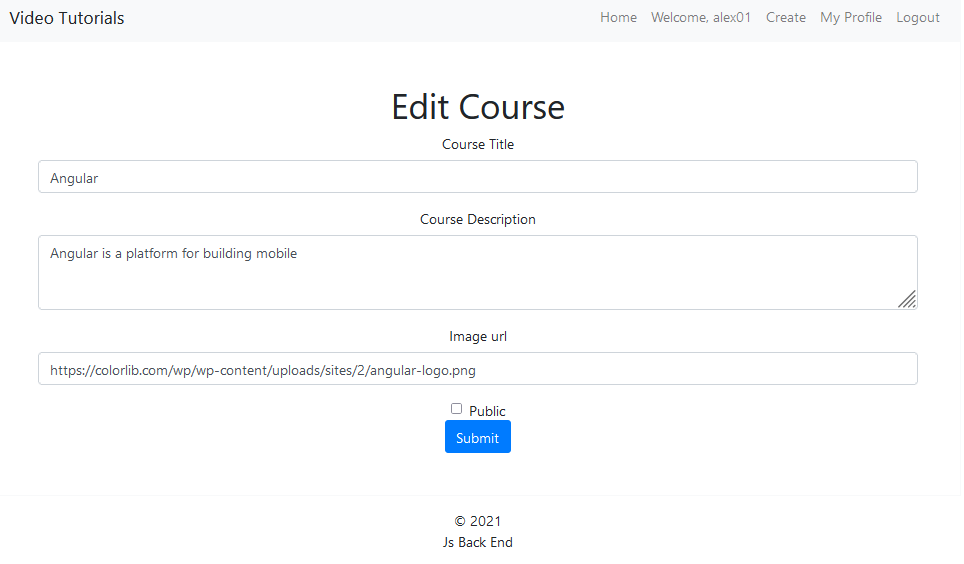
If a user has once enrolled in the current course, he should see "**You are already enrolled**".

**Delete Course (logged in user and creator of the current course)**

Each **creator** of the course must be able to click on the **[Delete]** button and **delete the current course** from the database and the user must be redirected to the **Home** page.

**Edit Course (logged in user and owner of the current course)**

Each **creator** can edit their **course**. All form fields should be **filled** with the corresponding information of the selected course. If successful, redirect the user to the current **course** details page.



**Security Requirements (10 Pts)**

The Security Requirements are mainly **access** requirements. Configurations about which users can access specific functionalities and pages.

* Guest (not logged in) users can access Home page (Listed all Courses).
* Guest (not logged in) users can access Login page and functionality.
* Guest (not logged in) users can access Register page and functionality.
* Guest (not logged in) can access the **Details** page without functionality.
* Users (logged in) can access Home page (Listed all Courses).
* Users (logged in) can access **Course** Details page and functionality.
  + **Users** (not course creator) can **Enroll** **once** for the course
  + **Users** (course creator) can **Edit** and **Delete** the current course
* Users (logged in) can access **Create Course** page and functionality.
* Users (logged in) can access Logout functionality.

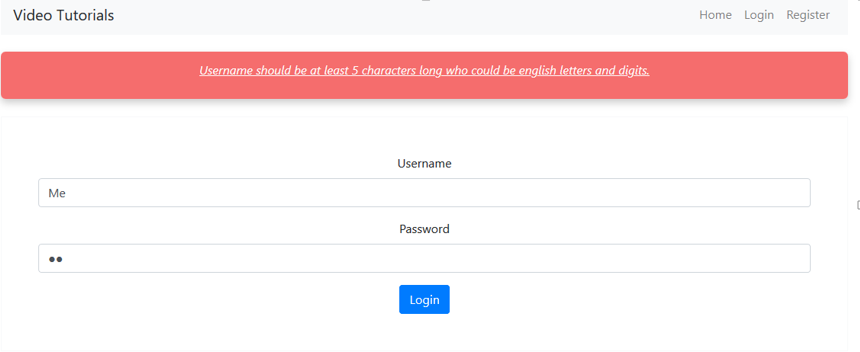
**Validation and Error Handling (10 Pts)**

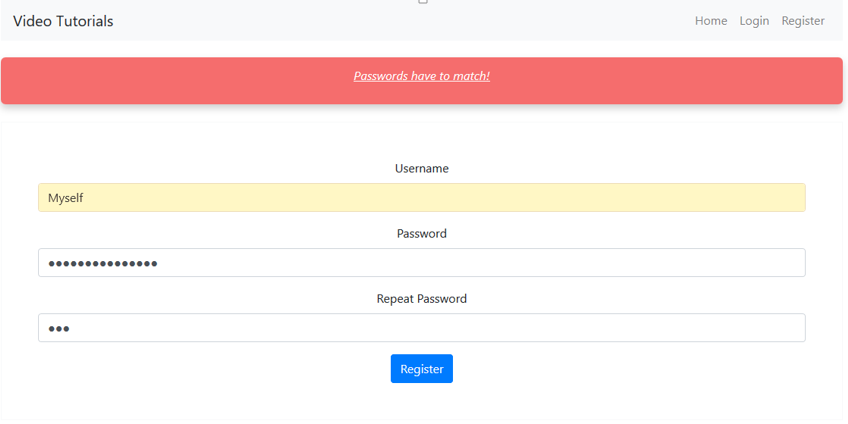
The application should notify the users about result of their actions.

### Login / Register

You should make the following validations:

* The **username** should be **at least 5 characters** long and should consist only **english** **letters** and **digits**
* The **password** should be **at least 5 characters** long and should consist only **english** **letters** and **digits**
* The **repeat password** should be **equal to the password**

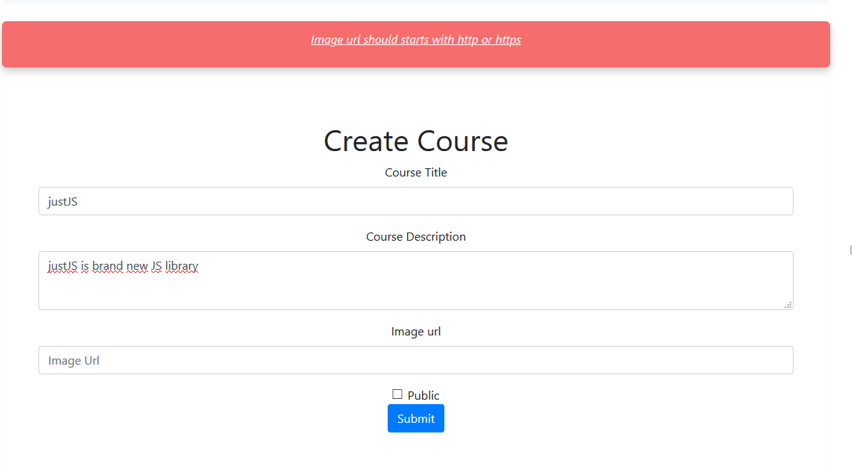




### Course

You should make the following validations while **creating** or **editing a** **course**:

* The **title** should be **at least 4 characters**
* The **description** should be **at least 20 characters** long
* The **imageUrl** should **start with http or https**



**\* Bonus (10 Pts)**

Each logged-in user should be able to view his own profile by clicking **[My Profile].**

**Profile Page** shows the information of the user as shown in the picture.

**Enrolled Courses** – is a **string** containing all the **titles** of the enrolled courses of the **user**, separated by comma and space(**,** ).

