

# JS Back-End Exam – Course Book

## 1. Exam Rules:

1. You have **4 hours**.
2. When you are ready, delete the **node\_modules** folder, make sure all dependencies are listed in the **package.json** file and submit your archiving project.
3. You must use the provided HTML & CSS resources.
4. You may **add attributes** (such as **class** and **dataset**), but it is forbidden to **change existing** attributes (such as **class** and **id**).
5. You may **change "href"** attributes on links and add/change the **method** and **action** attributes of HTML Forms.
6. Use **Express.js** as a back-end framework.
7. Use **MongoDB** as a database with **mongoose**.
8. You can use whatever **view engine** you like (**express-handlebars**, EJS, Pug, etc.).
9. Use **bcrypt** for hashing the password.
10. It is **forbidden** to use **React**, **Vue**, **Angular**, etc.
11. All applications will be **examined** as follows:
  - The needed **dependencies** are installed with the command "**npm install**".
  - The application is **started** with the command "**npm run start**".
12. The examiners are **NOT** allowed to **change** anything in your code and will **NOT** use any **instruments outside** of the course scope.

### IMPORTANT:

1. THE APPLICATION MUST START FROM THE FILE "INDEX.JS" ON PORT 3000!
2. ONLY THE LAST SUBMISSION WILL BE EVALUATED!

## 2. Application Overview

Get acquainted with the provided **HTML and CSS** and create an application for the **course posts**.

The visitors can **view** the **Home page** and **All Course** page with available courses. They can also **register** with an **email**, **username**, and **password**, allowing them to create their offers and to **sign up for a course they are interested in** (if the **current user** is **not** the author of the offer). Authors can edit or delete their courses at any time.

## 3. Functional Requirements

### Guest (Not logged in)

**Guest** navigation example:



The **application** should provide **Guest** (not logged in) users with the functionality to **Login**, **Register**, and **view** the **Home** page, **All Course** page, and **Details** page.

## Users (Logged in)

User navigation example:



The **application** should provide **Users** (logged in) with the functionality to:

- View the Home page and all other pages with logged-in navigation
- View Course page
- Create a new Course Offer [Create Course Offers]
- Access course details page [Details]
- Sign up for a course offer (if the current user is not the owner of the course)
- Delete or Edit the course depending on the user's authentication (only for the owner of the current course)

## 4. Database Models

The **Database** of the **Course Book** application needs to support **two entities**:

### User

- username - string (required),
- email - string (required),
- password - string (required)

### Course

- title - string (required),
- type - string (required),
- certificate - string (required),
- image: string (required),
- description: string (required),
- price – number (required),
- signUpList - a collection of Users (a reference to the User model)
- owner - object ID (a reference to the User model)


**Note:** When a user **signs up for a course**, their **id** is added to the **signUpList** collection. You can add more entities, if you need.

Implement the entities with the **correct data types**.

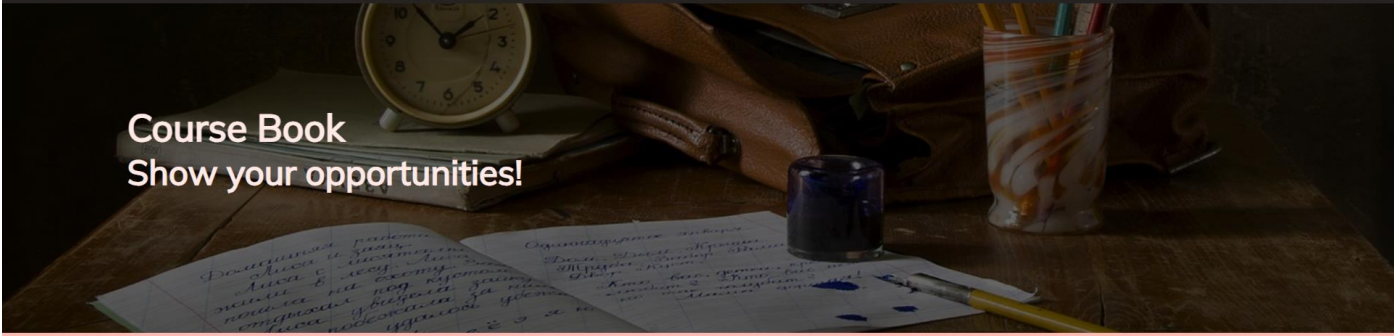
## 5. Application Pages

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

Visualize **the last 3 added course offers**. Each course must show information about the **course title**, **type**, **certificate** and **price**, as well as a page with **details** about the **specific course**.



[Home](#)
[All Courses](#)
[Profile](#)
[Create Course Offers](#)
[Logout](#)



## Course Book

### Show your opportunities!

#### Classroom Management Essentials

---

Type: Management

Certificate: Yes

Price: €35

[Learn More...](#)

#### Art History Prehistory to the Renaissance

---

Type: Art History

Certificate: Yes

Price: €17

[Learn More...](#)

#### Spanish 1-4: Beginner, Elementary

---

Type: Language


Certificate: Yes

Price: €15

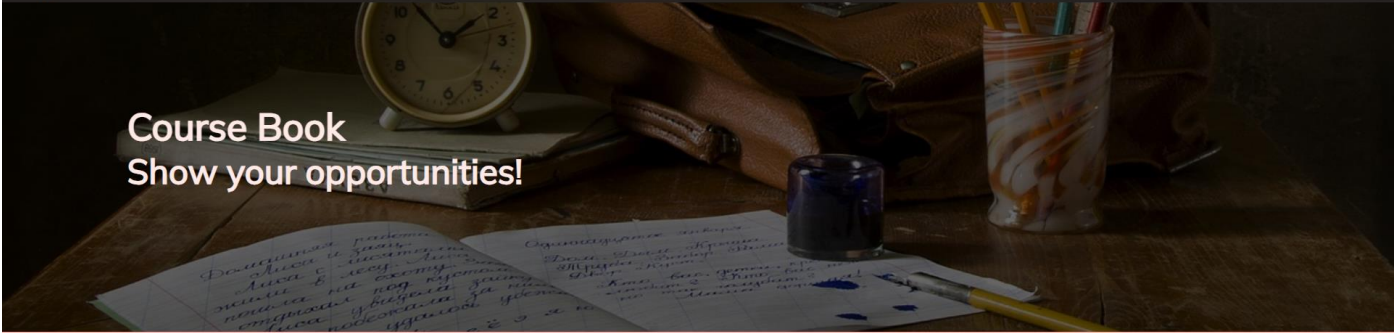
[Learn More...](#)

Back-End 2023

If there are **NO** courses in the database yet, display "No course created yet."



[Home](#)
[All Courses](#)
[Profile](#)
[Create Course Offers](#)
[Logout](#)



## Course Book

### Show your opportunities!

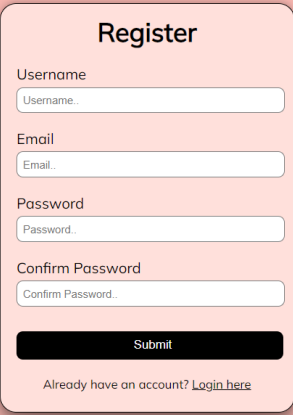
No course created yet.

Back-End 2023

## Register Page (Logged Out User)

Register a user inside the database with a **username**, **email**, and **password**. The **password** inside the **database** must be hashed (use **bcrypt**), and both passwords **must match**! After successful registration, you should **redirect** the user to the **Home page**. If the user tries to submit **invalid data**, an **error message** should be displayed and the already entered **data should be kept** in the input fields (except for the passwords).

A user who is already logged in should **NOT** be able to reach this page by typing its URL (e.g. "http://localhost:3000/register") in the browser address bar.



Register

Username

Email

Password

Confirm Password

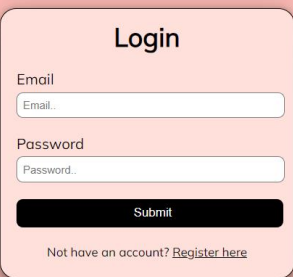
Already have an account? [Login here](#)

Back-End 2023

## Login Page (Logged Out User)

Logging an already registered user with the correct **email** and **password**. After successful login, you should **redirect the user to the Home page**. If the user tries to submit **invalid data**, an **error message** should be displayed and the already entered **data should be kept** in the input fields (except for the passwords).

A user who is already logged in should **NOT** be able to reach this page by typing its URL (e.g. "http://localhost:3000/login") in the browser address bar.



Login

Email

Password

Not have an account? [Register here](#)

Back-End 2023

## Logout (Logged in user)

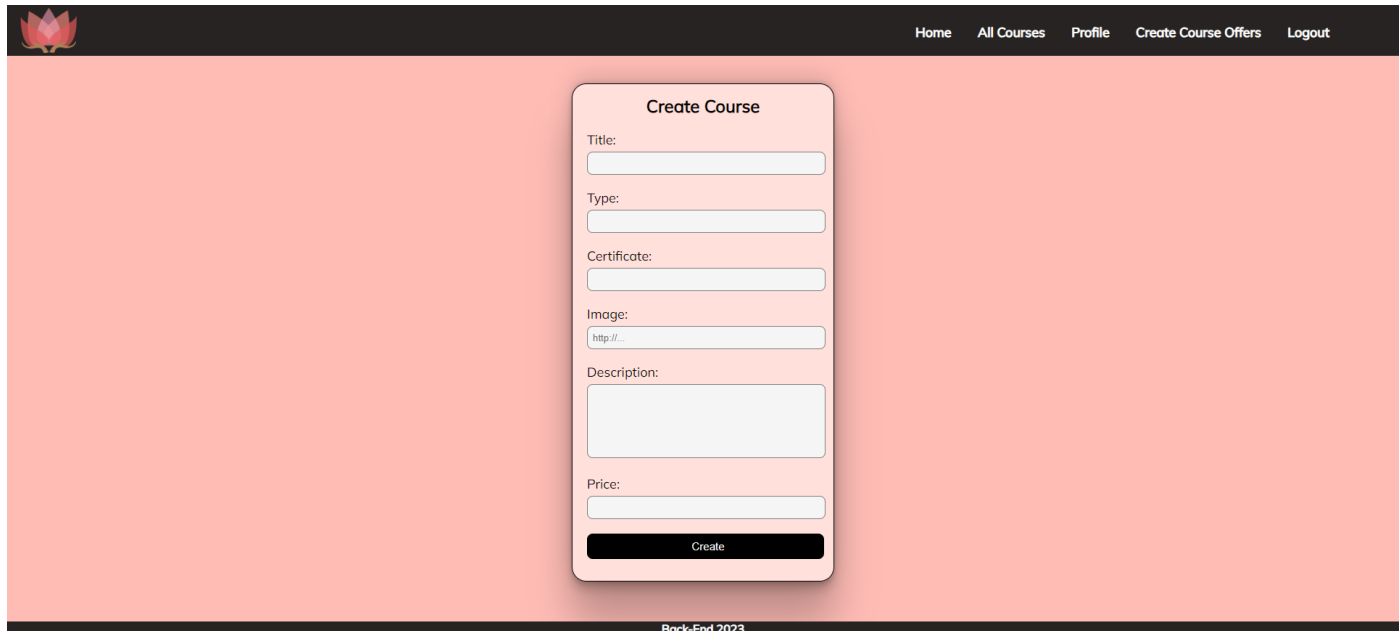
The logout action is available to **logged-in** users. Upon success, clear any session information and **redirect** the user to the **Home** page.

A user who is NOT logged in should NOT be able to reach this functionality by typing its URL (e.g. "http://localhost:3000/logout") in the browser address bar.

## Create Course Offer (Logged in User)

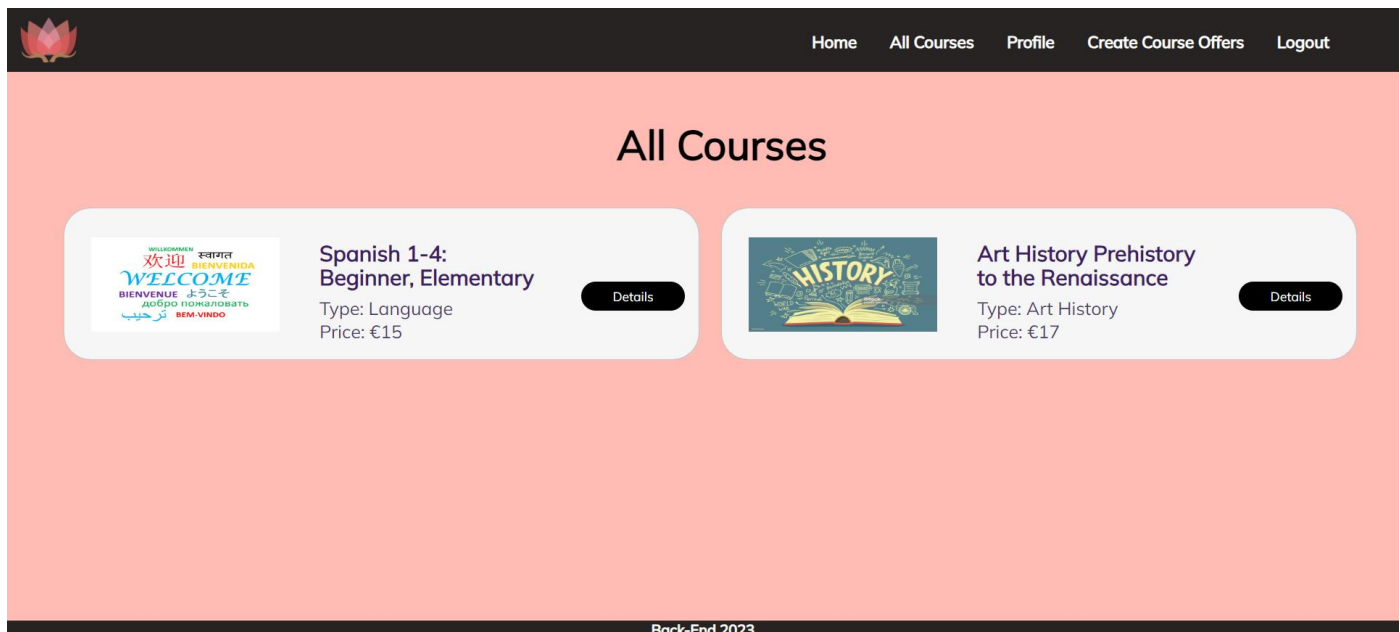
The **Create** page is available to **logged-in users**. It contains a form for adding new course offers. Upon success, **redirect** the user to the **All Course** page. If the user tries to submit **invalid data**, an **error message** should be displayed and the already entered **data should be kept** in the input fields.

A guest user (who is not logged in) should **NOT** be able to reach this page by typing its URL (e.g. "http://localhost:3000/courses/create/1234") in the browser address bar.



## Course Catalog (For logged in users and logged out users)

A list of all course offers. Each course must be displayed with the **course image**, the **title**, the **type**, the **price**, and a button for **details** about the **specific course**. As in the picture below:



The **[Details]** button should be a link to the **details page** for the current course post.

If there are **NO** offers in the database yet, display **"No course created yet."**



## All Courses

No course created yet.

Back-End 2023

### Details Page (For logged in users and logged out users)


All users should be able to **see the details about the courses**. Clicking the **Details button** on the course card should display the details page. If the currently registered user **is the creator** of the course offer, the **Edit** and **Delete** buttons should be displayed. Otherwise, they **should not be available**.

Information about the course:


- **Title**
- **Type**
- **Certificate**
- **Price**
- **Email of the creator**
- **People who have signed up for this course:**
  - If any, separate their usernames with comma and space ", "
  - If **not**, display "**Nobody has signed in for this course yet**"
- **Description**
- **Buttons** (Depending on the status of the currently logged in user)

### Details Page (Logged out users)

If the user has not **logged in**, **no buttons** should be displayed.



[Home](#)
[All Courses](#)
[Login](#)
[Register](#)



**Title: Classroom Management Essentials**

Type: Management

Certificate: Yes

Price: €35

By: john@abv.bg


Signing By: Peter

Curabitur fringilla nec urna sit amet eleifend. Suspendisse sodales vitae tellus sit amet dictum. Vivamus condimentum blandit interdum. Donec ac placerat enim. Pellentesque elementum condimentum massa, id condimentum lorem maximus sit amet. Vivamus accumsan imperdiet urna, eget malesuada felis iaculis quis. Nunc orci neque, blandit nec lorem eu, mattis porta erat. Etiam et imperdiet tellus. Nulla ac gravida tellus.


Back-End 2023

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

If the **currently logged-in** user is the **owner** (the user who **created the course offer**), he should see the **[Delete]** and **[Edit]** buttons.



[Home](#)
[All Courses](#)
[Profile](#)
[Create Course Offers](#)
[Logout](#)



**Title: Classroom Management Essentials**

Type: Management

Certificate: Yes

Price: €35

By: john@abv.bg

Signing By: Peter

Curabitur fringilla nec urna sit amet eleifend. Suspendisse sodales vitae tellus sit amet dictum. Vivamus condimentum blandit interdum. Donec ac placerat enim. Pellentesque elementum condimentum massa, id condimentum lorem maximus sit amet. Vivamus accumsan imperdiet urna, eget malesuada felis iaculis quis. Nunc orci neque, blandit nec lorem eu, mattis porta erat. Etiam et imperdiet tellus. Nulla ac gravida tellus.

Edit


Delete


Back-End 2023

Details Page (Logged in user who is not signed up)

If the currently logged-in user is **not the owner** (a user who is **not** the creator of this course offer) and **has not signed up** for that course, he should see a **Sign up** button.




Home All Courses Profile Create Course Offers Logout



**Title: Classroom Management Essentials**  
 Type: Management  
 Certificate: Yes  
 Price: €35  
 By: john@abv.bg  
 Signing By: Peter


Curabitur fringilla nec urna sit amet eleifend. Suspendisse sodales vitae tellus sit amet dictum. Vivamus condimentum blandit interdum. Donec ac placerat enim. Pellentesque elementum condimentum massa, id condimentum lorem maximus sit amet. Vivamus accumsan imperdiet urna, eget malesuada felis iaculis quis. Nunc orci neque, blandit nec lorem eu, mattis porta erat. Etiam et imperdiet tellus. Nulla ac gravida tellus.


Sign up

Back-End 2023

## Details Page (Logged in user already sign up for course)

If the currently logged in user **is not the owner** and has **already sign up for that course**, he should see **[You've already signed up for this course!]**.


Home All Courses Profile Create Course Offers Logout



**Titel: Classroom Management Essentials**  
 Type: Management  
 Certificate: Yes  
 Price: €35  
 By: john@abv.bg  
 Signing By: Peter

Curabitur fringilla nec urna sit amet eleifend. Suspendisse sodales vitae tellus sit amet dictum. Vivamus condimentum blandit interdum. Donec ac placerat enim. Pellentesque elementum condimentum massa, id condimentum lorem maximus sit amet. Vivamus accumsan imperdiet urna, eget malesuada felis iaculis quis. Nunc orci neque, blandit nec lorem eu, mattis porta erat. Etiam et imperdiet tellus. Nulla ac gravida tellus.

You've already signed up for this course!

Back-End 2023

## Delete Course (Logged in user and owner of the current course)

Each **owner** of the course post must be able to click on the **[Delete]** button and **delete the current course** from the database. After this, the user must be redirected to the **All Course** page.

A user who is not the owner of the course should **NOT** be able to reach this functionality by typing its URL (e.g. "http://localhost:3000/courses/delete/1234") in the browser address bar.

## Edit Course (Logged in user and owner of the current course offer)

Each **owner** can edit their **course post**. Clicking the **[Edit]** button for a specific course on the details page should display the **Edit page**, all fields being populated with course data. It contains a form with input fields for all relevant



properties. If **successful**, redirect the user to the **current course post details page**. If the user tries to submit **invalid data**, an **error message** should be displayed and the already entered **data should be kept** in the input fields.

A user who is not the owner of the course should **NOT** be able to reach this page by typing its URL (e.g. "http://localhost:3000/courses/edit/1234") in the browser address bar.

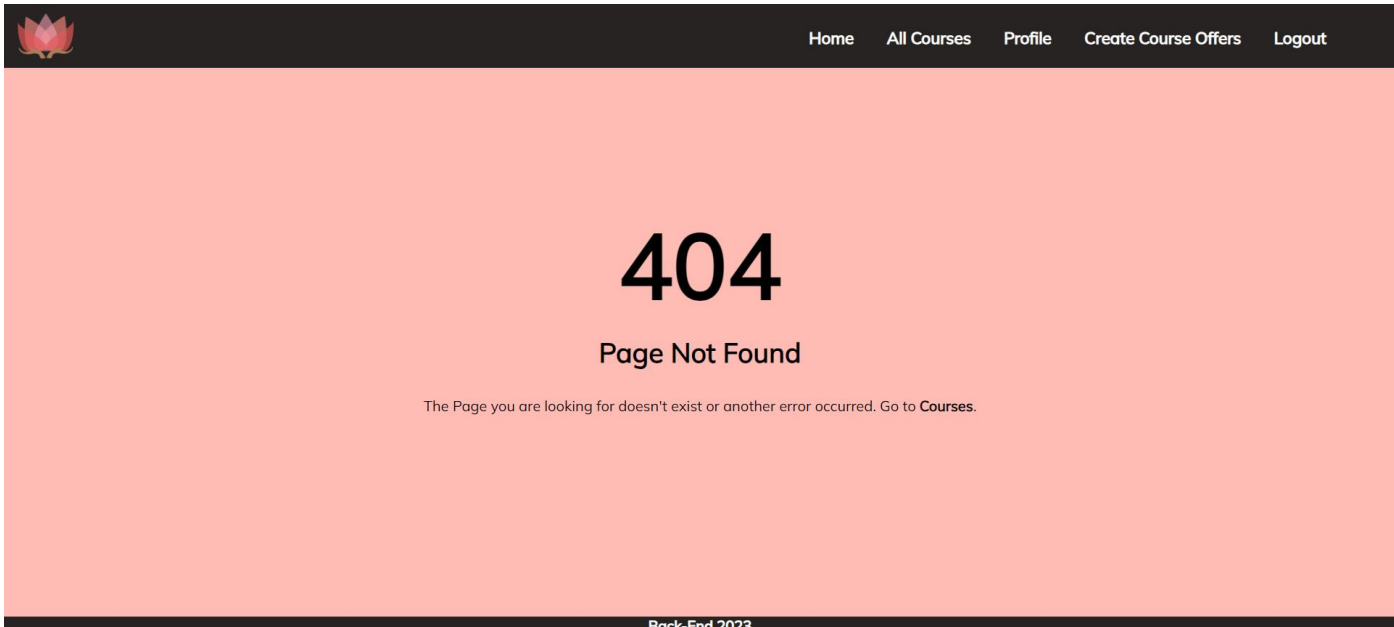
Back-End 2023

## 6. Security Requirements (Routes Guards)

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

- **Guest** (not logged in) users can access the **Home** page.
- **Guest** (not logged in) users can access the **Login** page and functionality.
- **Guest** (not logged in) users can access the **Register** page and functionality.
- **Guest** (not logged in) and **Users** (logged in) can access the **All Course Page (Listed all course offers)**.
- **Guest** (not logged in) can access the **Details** page without functionality.
- **Users** (logged in) can access the **Home** page.
- **Users** (logged in) can access the **Details** page and functionality.
  - **Users** (not course post owners) can **sign up for other courses**.
  - **Users** (course post owners) can **Edit** and **Delete** the current course post
- **Users** (logged in) can access the **Profile** page and functionality.
- **Users** (logged in) can access the **Create Course Offers** page and functionality.
- **Users** (logged in) can access the **Logout** functionality.

Use the following view for **invalid paths**:



## 7. Validation and Error Handling

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

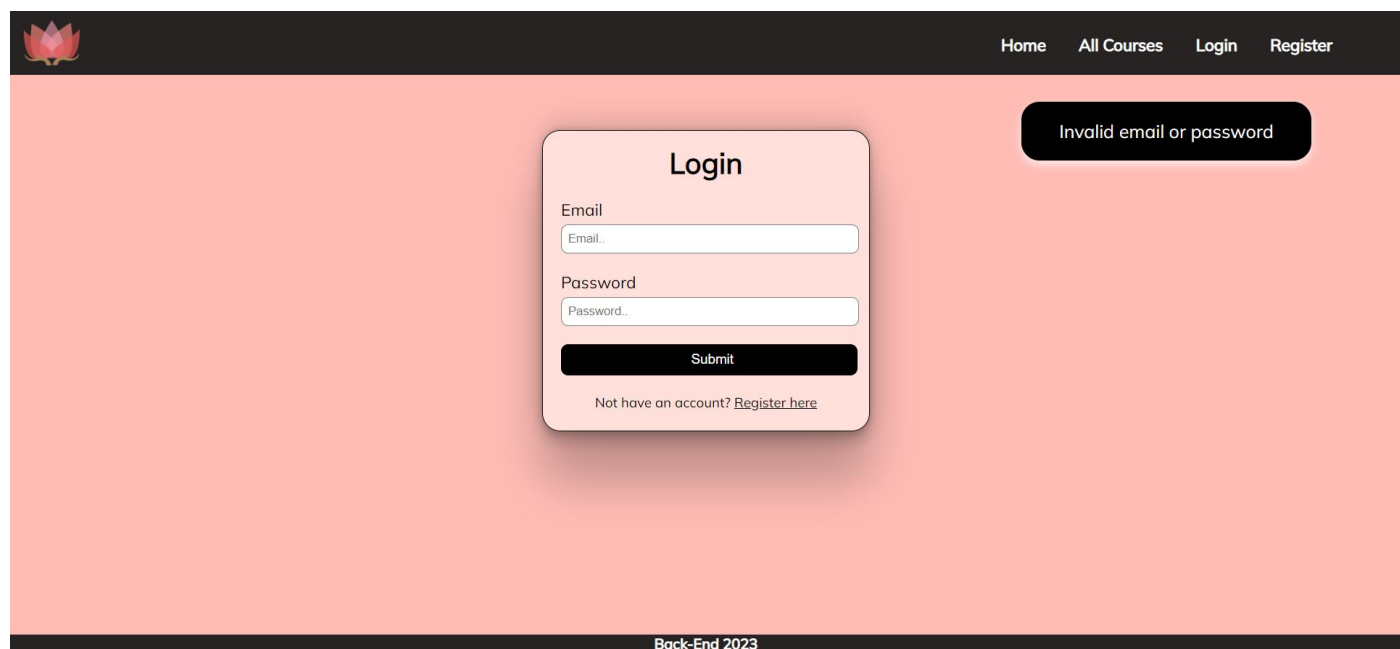
In case of error, you should display div with class "errorContainer"

You can choose to display the first error or all of them. You have complete freedom to choose the content of the error message you will display, as long as it is clear what exactly the error is.

### Login / Register

You should make the following validations:

- The **username** should be **at least 2 characters** long
- The **email** should be **at least 10 characters** long
- The **password** should be **at least 4 characters** long
- The **repeat password** should be **equal to the password**



## Course

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

- The **Title** should be **at least 5 characters**
- The **Course Image** should start with **http://** or **https://**
- The **Description** should be a minimum of **10 characters long**
- The **Type** should be a minimum of **3 characters long**
- The **Certificate** should be a minimum of **2 characters long**
- The **Price** should be a **positive number**

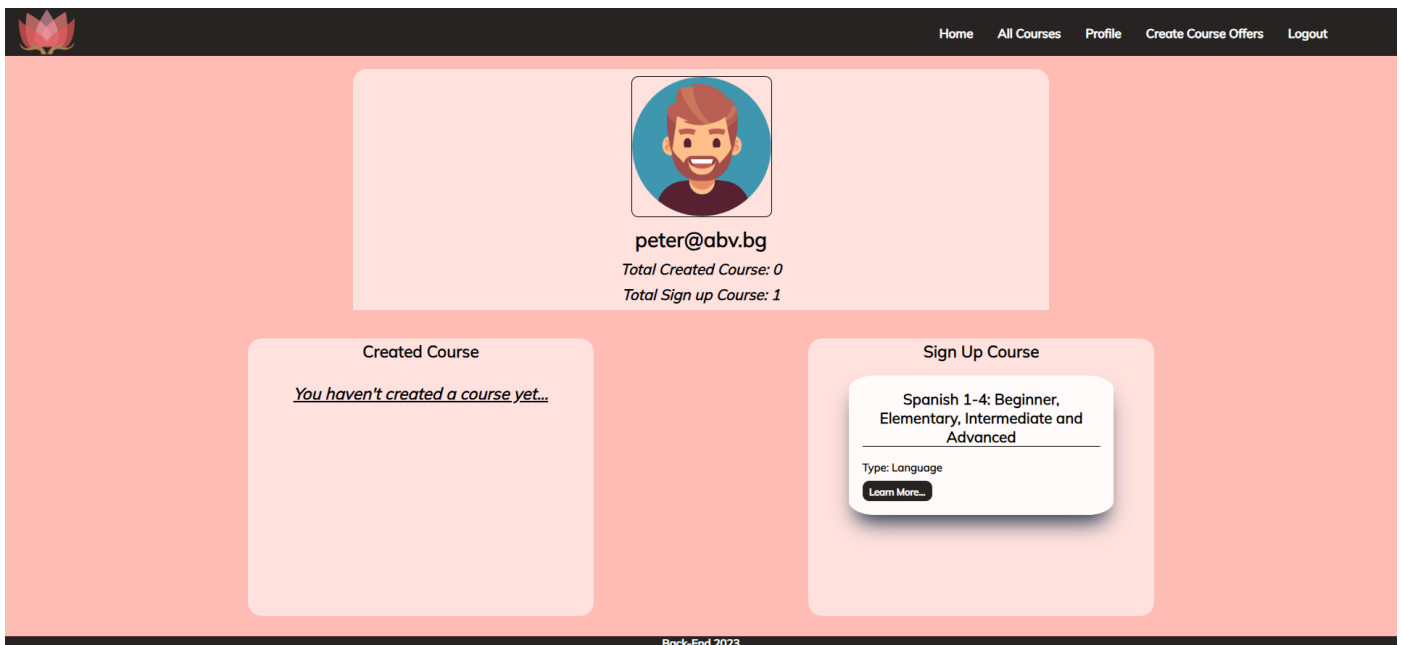
Back-End 2023

## 8. \*Bonus – Profile

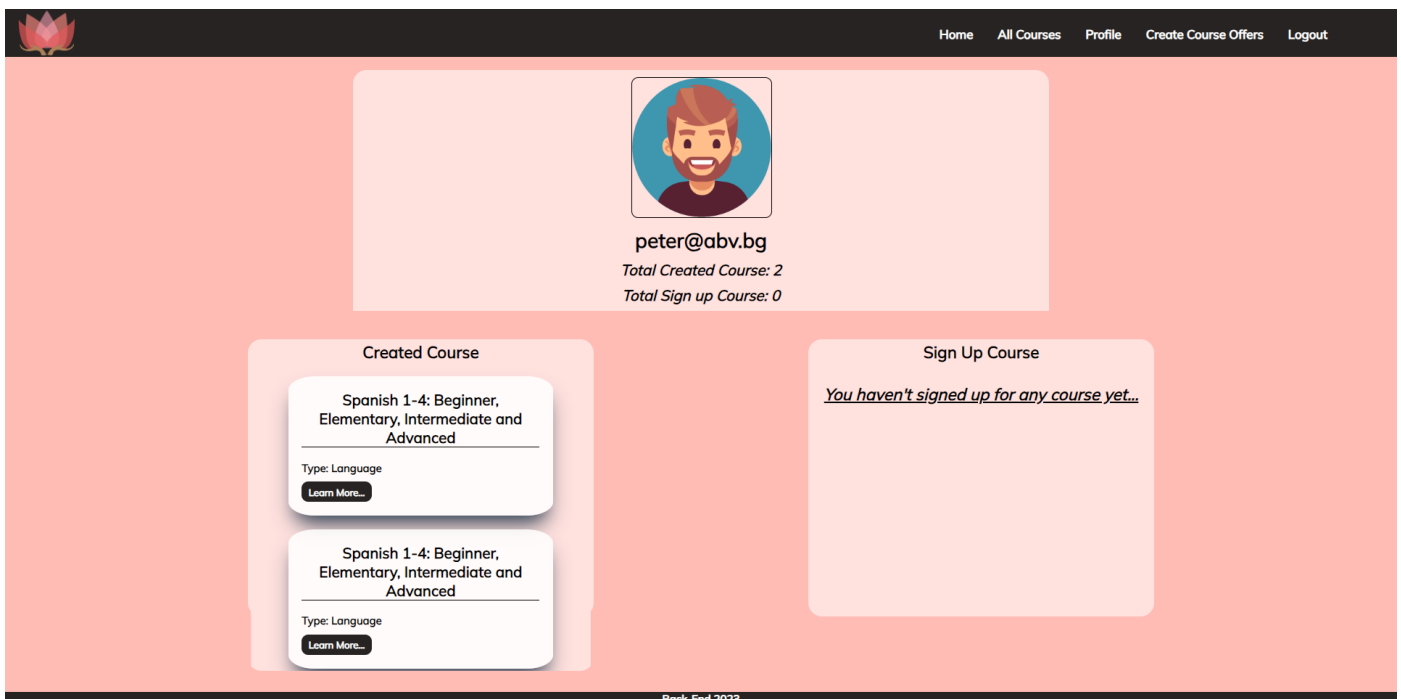
Each **logged-in user** should be able to view their profile information by clicking on **[Profile]**. On the profile page, the user should see their **email** and the **courses** they have **created**, as well as the courses they have **signed up for**. Under the **email**, the user has a **total count of the created courses** and the ones they have **signed up for**.

Back-End 2023

If there are **no created courses** yet, the message "**You haven't created a course yet...**" should be visualized.

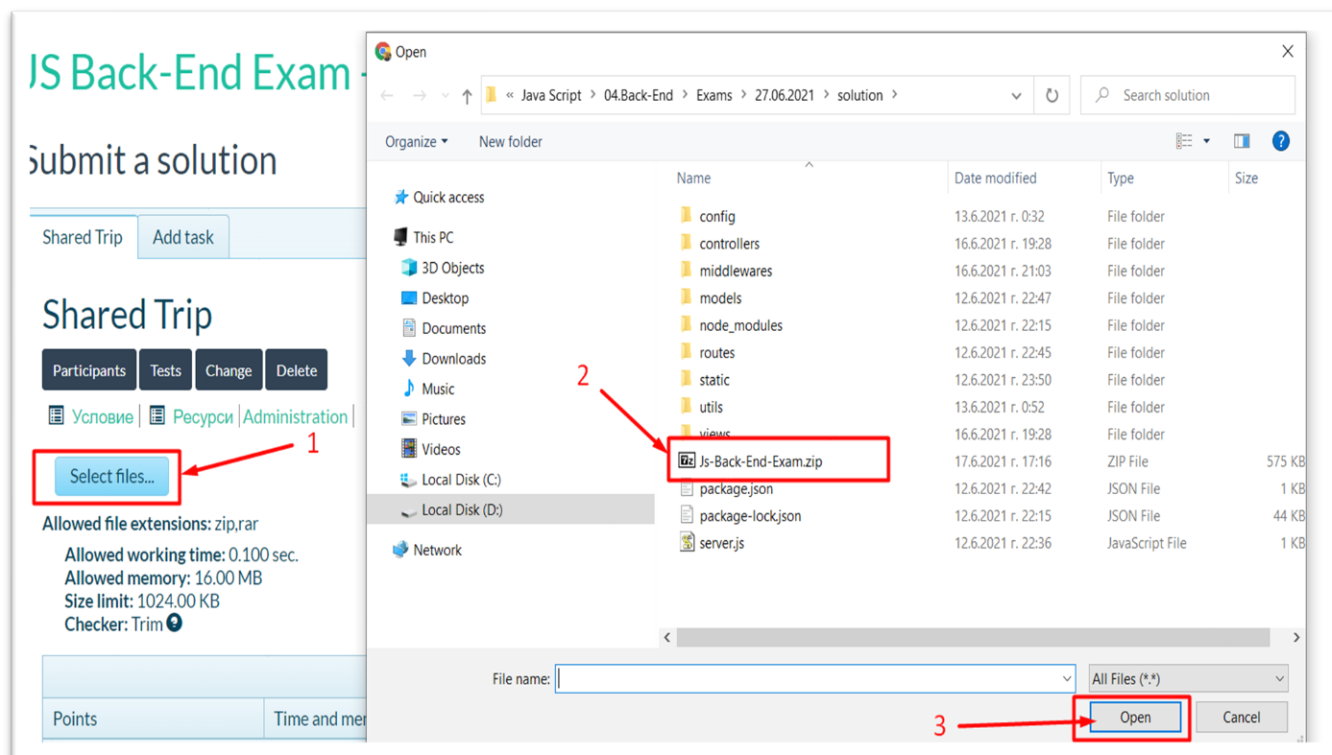
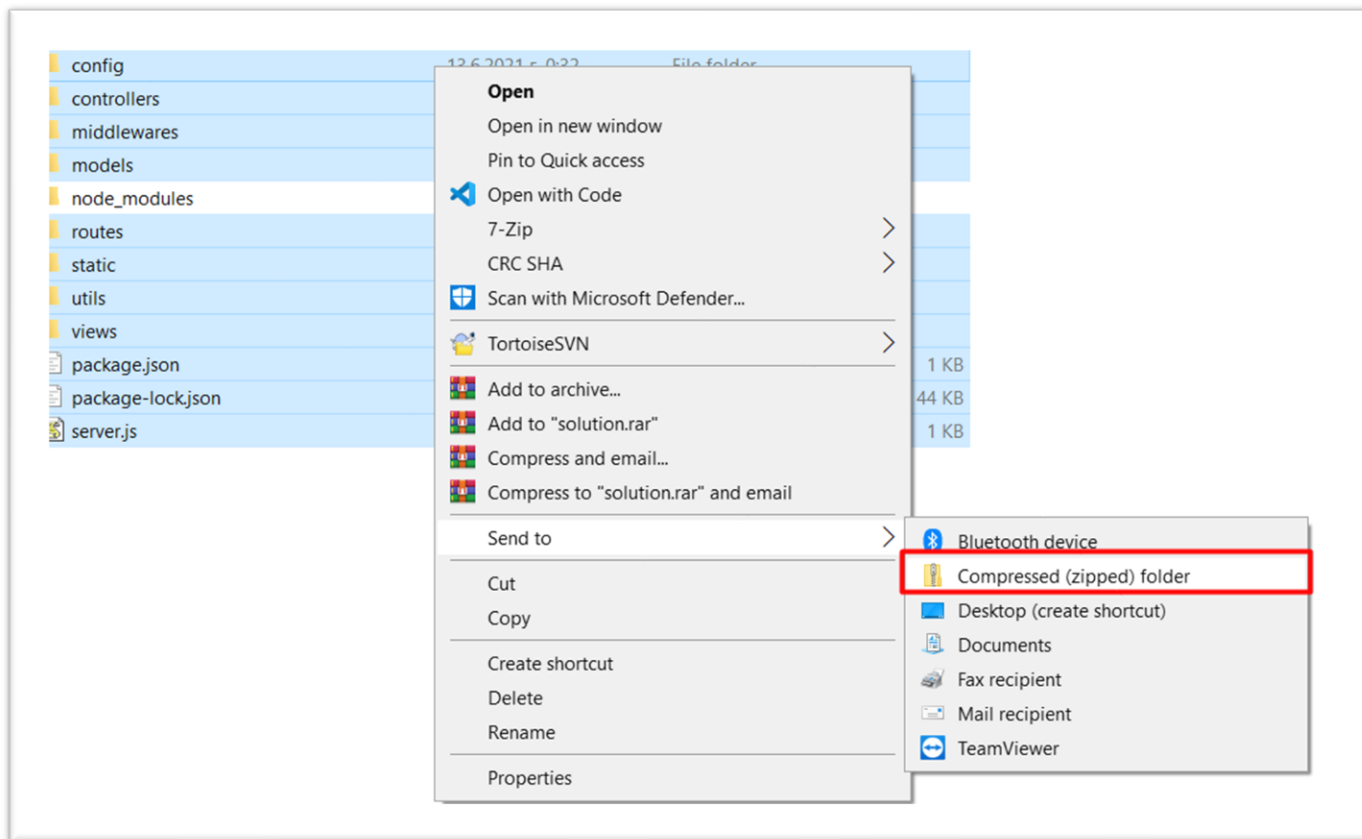


If the user **hasn't signed up for any course** yet, the message "**You haven't signed up for any course yet...**" should be visualized.



## 9. Submitting Your Solution

Place a **ZIP** file in your project folder. Exclude the **node\_modules** folder. Upload the archive to Judge.



Shared Trip

Add task

## Shared Trip

Participants

Tests

Change

Delete

Условие

Ресурсы

Administration

Select files...

Js-Back-End-Exam.zip

Allowed file extensions: zip,rar  
 Allowed working time: 0.100 sec.  
 Allowed memory: 16.00 MB  
 Size limit: 1024.00 KB  
 Checker: Trim

File upload

Submit

Shared Trip

Participants

Tests

Change

Delete

Условие

Ресурсы

Administration

Select files...

Allowed file extensions: zip,rar

Allowed working time: 0.100 sec.

Allowed memory: 16.00 MB

Size limit: 1024.00 KB

Checker: Trim

File upload

Submit

Problem results

Participant	Result
shampion	0 / 100

Submissions

Points	Time and memory used	Submission date
Compile time error		17:24:04 17.06.2021

This is not a problem !

GOOD LUCK! 😊