**Description:**

Udemy marketplace is a web application which provides multiple courses on different

domains/technologies. Learners can sign up and then enroll for the course by paying the

course fee and start learning. The requirement is to build the frontend for the online learning

marketplace application.

Design and implement a frontend ecommerce platform that needs to be implemented in Angular. This application needs to interact with a backend (see final-project-backend for more details)

**Technical Requirements:**

* Use components and databinding.
* Split a view up into multiples components
* Implement dependency injection
* Use observables if needed
* Implement Routing to build a multi-page application
* Use pipes if needed
* Implement authentication and protect frontend pages
* Apply loading control when data is loading
* Apply Error Modal to handle errors
* Connect to a backend to manage all data.
* Unit test cases are must for each component you implement.

**Functional Requirements:**

• User Management:

1. The application should support adding the users to the application. The basic details

will include information like display name, first name, last name, about yourself/bio,

areas of interest (can choose multiple interests), user type(student/professional),

experience, domain expertise, role, profile picture

2. The application should support updating user details like bio, areas of interest etc.

3. The application should support displaying the user profile and provide an option to

update it.

Udemy Management:

4. The application should support adding Udemy courses to the application. The basic

details will include information like name, domain, description, author, price, duration,

reviews, rating etc.

5. The application should support listing the courses, sorting based on price, filtering

based on the course name.

6. The application should support displaying the details of a single course.

7. The application should support adding reviews and ratings for courses by the

enrolled users.

8. Users should get the recommendation on courses based on their domain of interest.

• Cart Management:

9. The application should support adding the courses to the cart for particular users.

10. The application should support viewing the cart items for particular users.

11. The application should support deleting the courses from the cart for particular users.

12. The application should support calculating the total cart price and on checkout it

should create orders.

Order Management:

13. The user should be able to add the courses to the orders.

14. The user should be able to see all the courses they ordered. This can also have

features like sorting, filter by category/course type.

Authentication:

15. If the user is logout, he could not see anything but login page.

16. If the user is login, he could navigate between all pages, the token needs to set 1 hour of expiration and later the user will be redirect to login page.

**Project Plan:**

The delivery is expected to be high quality, iterative and showing progressive enhancements

without waiting for a last-minute integration.

The milestones below show the must-have features and how they should be planned for

delivery. The remaining time should be used to complete the functionality and prioritized

based on the business context

**Milestones:**

|  |  |  |
| --- | --- | --- |
| Milestone | Description | Estimate |
| Milestone 1 | Create a form-page to add user |  |
| Milestone 2 | Create a form-page to update user's display name, first name, last name, about  yourself/bio, areas of interest (can choose multiple interests), user type (student/professional), experience, domain  expertise, profile picture |  |
| Milestone 3 | Create a page to show all the details of a user and the course he's enrolled in |  |
| Milestone 4 | Create page to add and delete courses | Completado |
| Milestone 5 | List the available courses. There should be provision to sort by price (low to high, high  to low) and filter options based on course name, course category etc. | Completo |
| Milestone 6 | Create a page to add/remove course to cart of a given user | Completo con errores |
| Milestone 7 | Create a page to summarize what is the current content of a given user's cart, | Complete |
| Milestone 8 | Create a page to check out the cart items.  This will be creating the order.  brownie point: generate a receipt as a file | Completo, falta crear el recibo |
| Milestone 9 | Create a page to add reviews and ratings for the course. | completo |
| Milestone 10 | Create a page to show all the required details of a single course. | Completo |
| Milestone 11 | Create a page to show all courses of user along with ratings and reviews. | En progreso  punto de terminarlo |
| Milestone 12 | Create a page to show the courses in inventory  based on the Performance. Performance part can be decided on parameters like how many times the course is bought, average ratings of course |  |
| Milestone 13 | Create a page to show the recommendation on  courses based on the user's domain of  interest and by course's performance |  |
| Milestone 14 | Provision for users to stay anonymous  when their reviews are displayed to other  users. Milestone 11 must respect this. |  |

See wireframe-final-project-java.html file to get the frames needed in this project.