Online Learning App

using Angular

Problem Statement

Develop a Comprehensive Single Page Application (SPA) in Angular Utilizing the Following Concepts:

- Components
- Parent-Child Components
- Directives (Structural, Attribute)
- @Input and @Output
- Routing and Routing Parameters
- Local Storage
- Routing Guards
- Services and Observables
- API Calls
- Form Handling (Template and Model-Based Approaches)
  Inbuilt Pipes   Custom Pipes

Website Features

01.

Listing Top course;
The home page, built with Angular, utilizes distinct components for each section, providing a modular and organized layout. It features a prominent hero section with a call-to-action, seamless navigation through Angular's routing,

02.

User Authentication Signup & Login
– Users must register and log in before accessing destinations. Session Persistence – Users remain logged in until they manually log out. Auth Guards – Prevent unauthorized access to itinerary and booking pages.
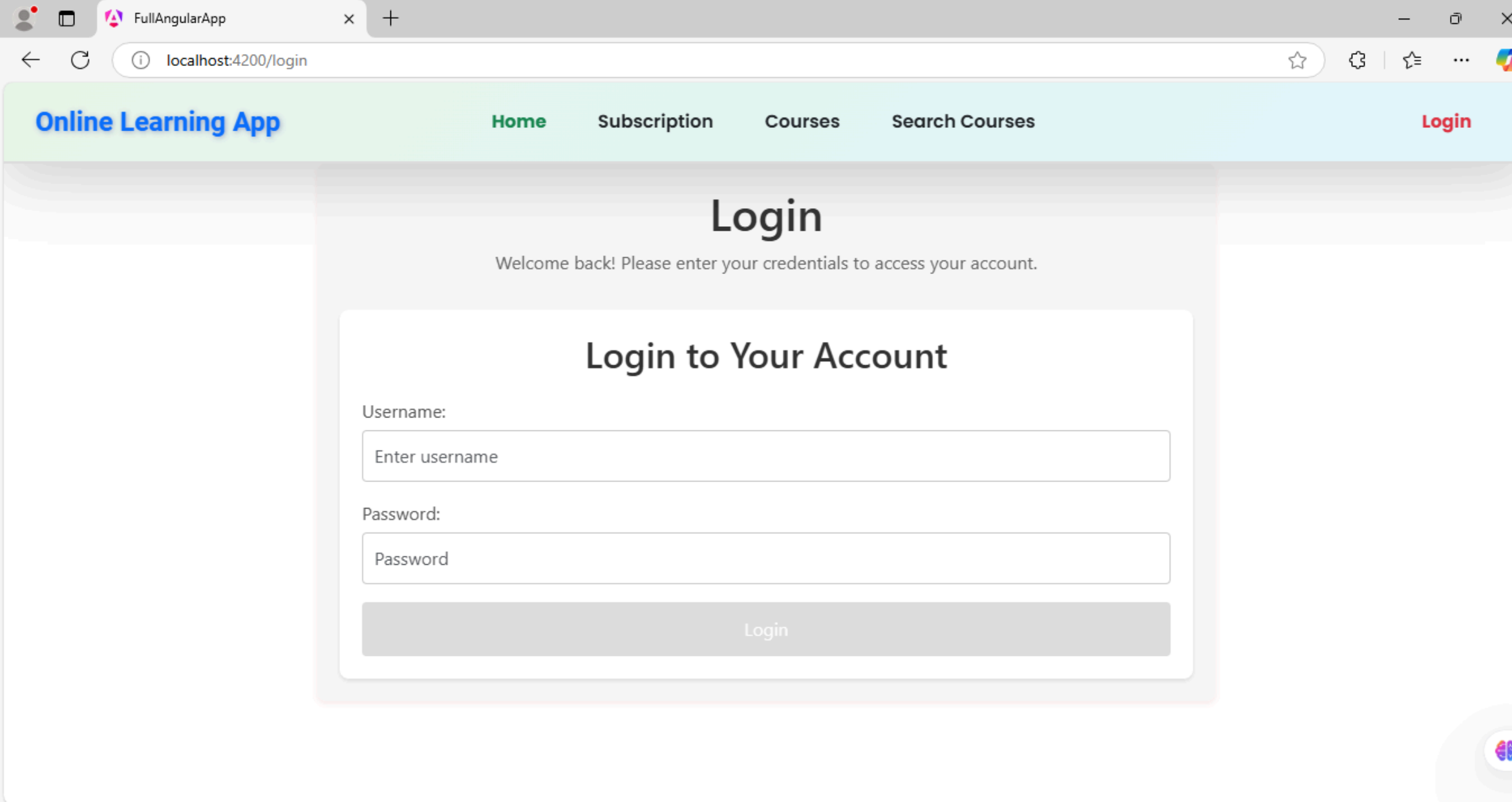
03.

Advanced Angular Features
1. Components & Routing: Organized navigation across various pages for a seamless user experience.
2. Pipes for Data Formatting: Formats costs, dates, and filters for clear presentation of information.

Login page

The login page ensures user authentication. It also helps in enabling the premium features to the logged in users.Angular's guard feature is used to achieve this.
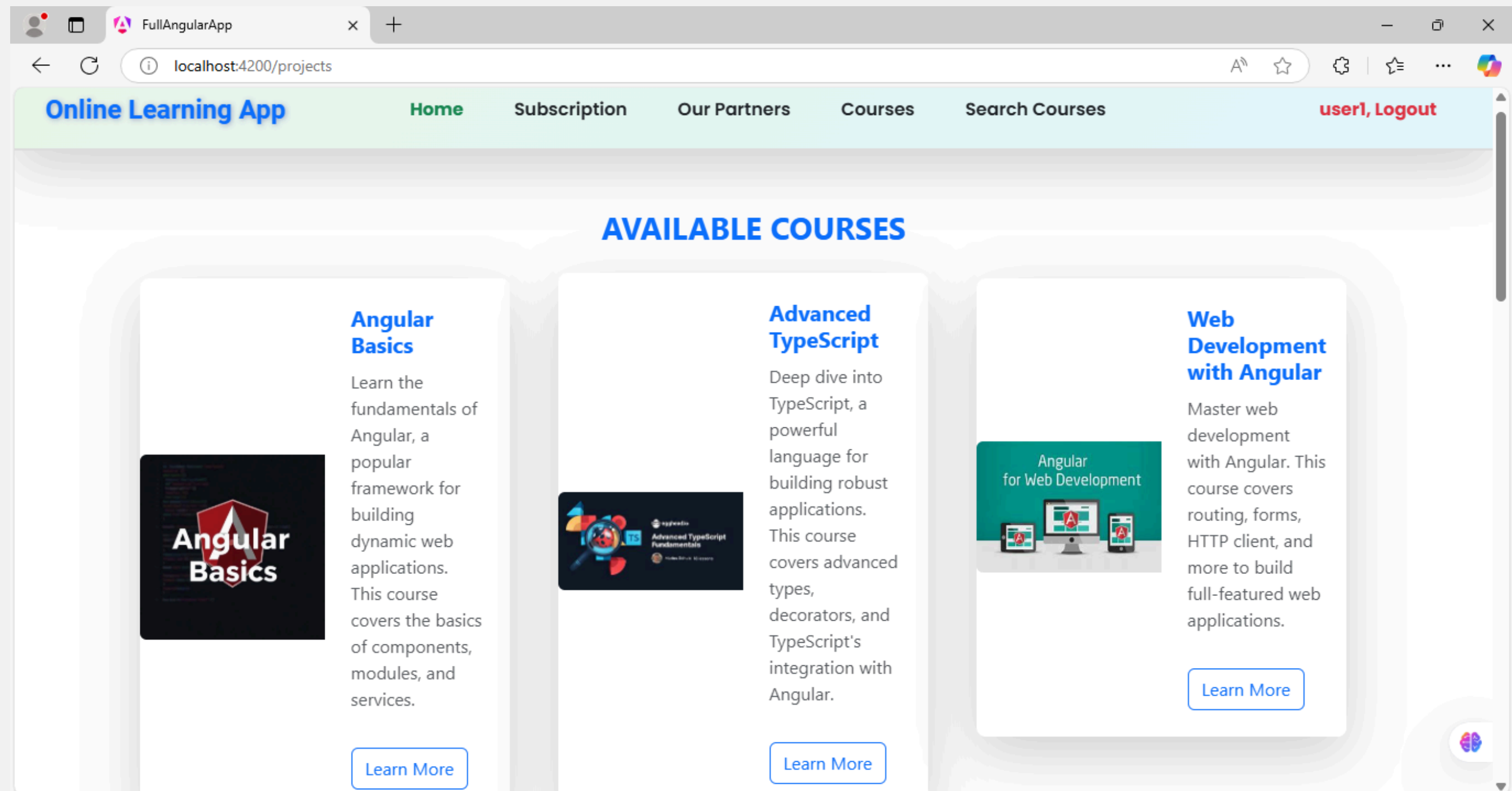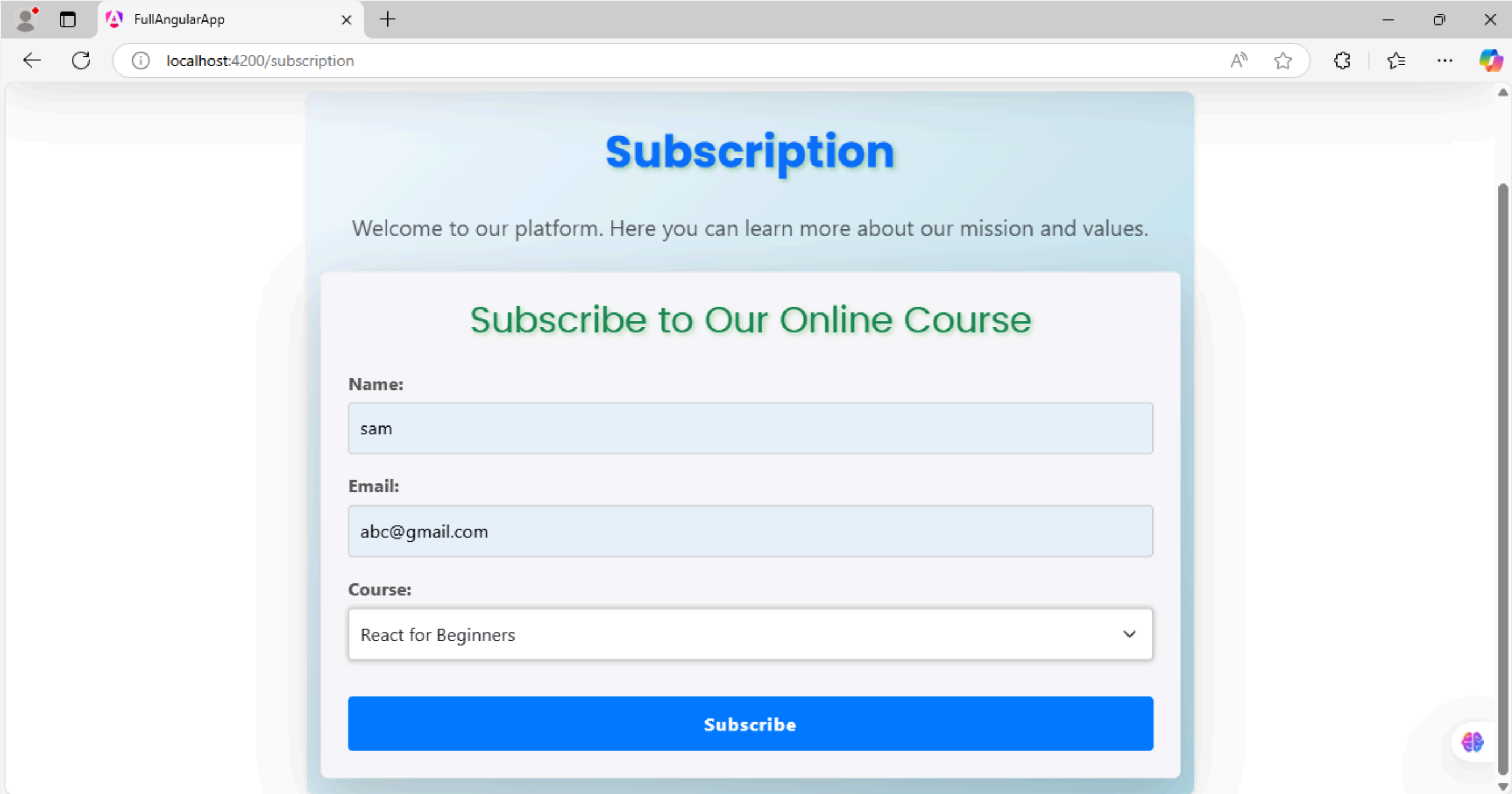
Course page

This page list all the courses available.To achieve this I have used api service to fetch the rael time data using json server to dynamically display the courses with title ,description and Learn more button.

Subscription page

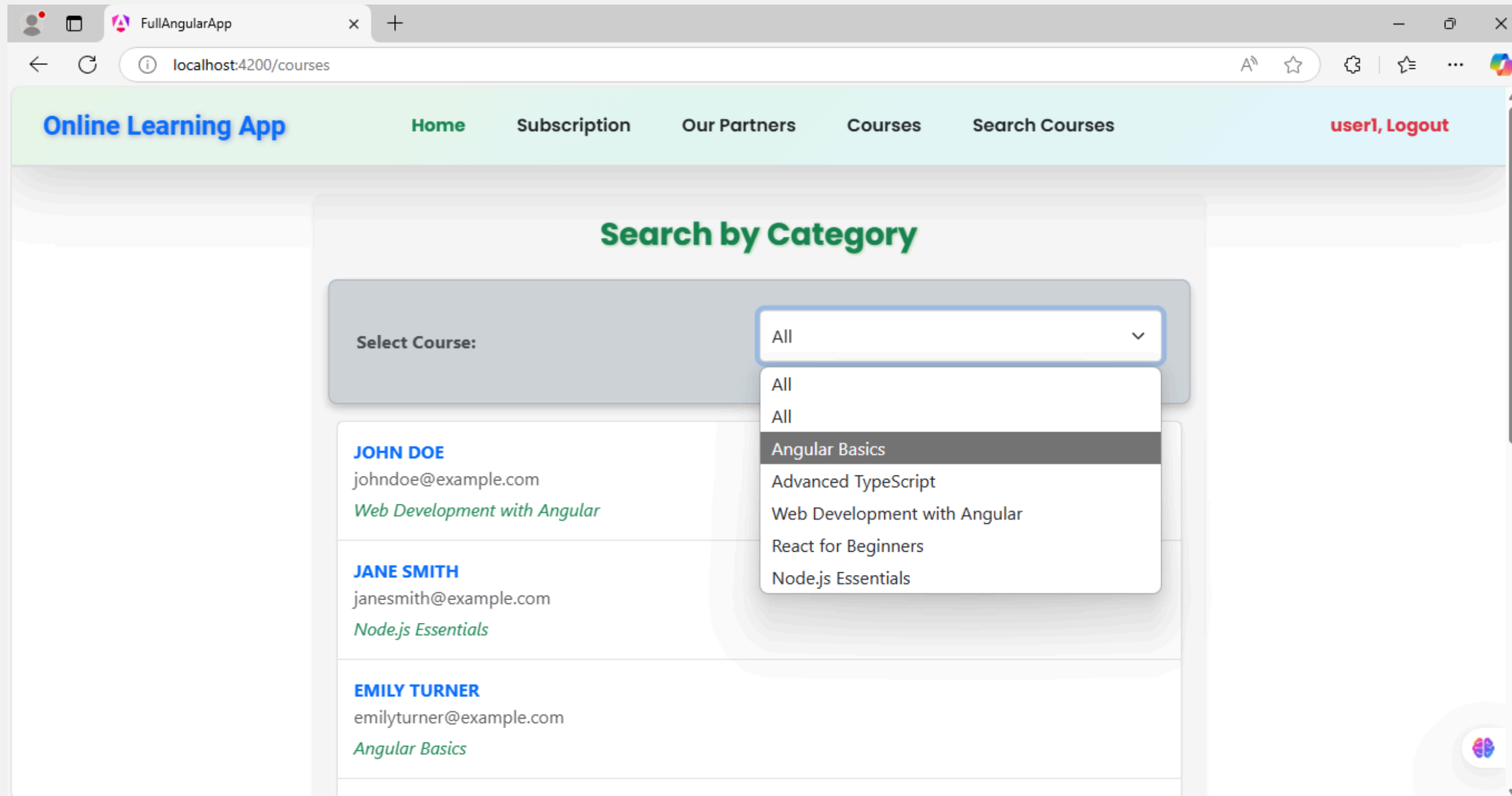This page enables users to subscribe to our online courses, granting them access to course materials.
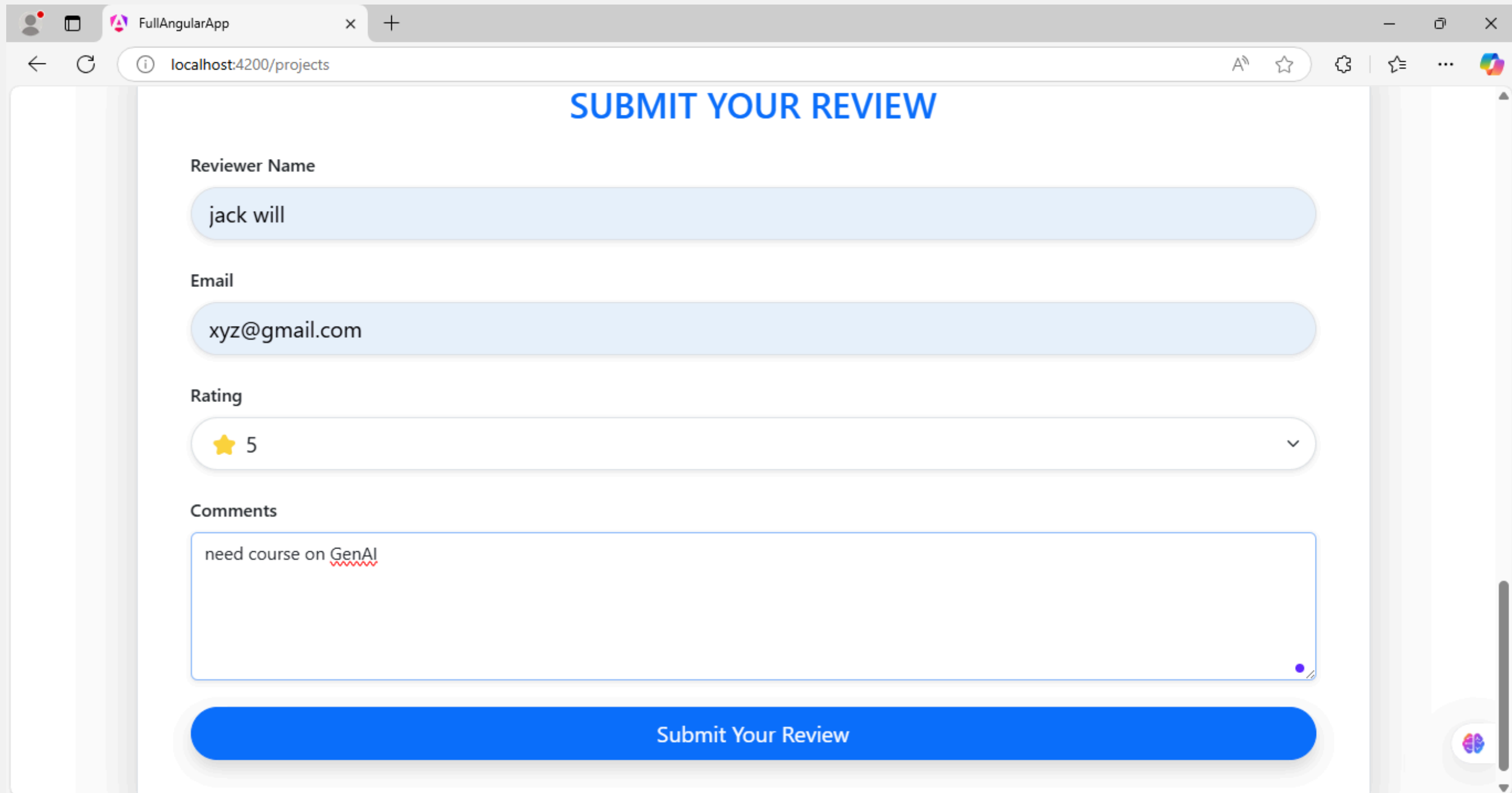
Search   page

This page allows users to search for courses based on the author's name or by technology category. To improve user experience, I have utilized Angular's built-in pipes along with my own custom pipe features.

Review   page

This is where users can leave their reviews, sharing their experiences regarding the ease of use of this site and expressing their needs. To create this review form, I utilized a model-based approach.

Conclusion

1. *Component-Based Architecture:* The application is structured using reusable components, allowing   for modular development and easier maintenance.
2. *Seamless Routing:* It provides organized navigation between different pages, enhancing user experience and accessibility.
3. *Data Formatting with Pipes:* Angular pipes are utilized to display formatted information, such as costs and dates, clearly and effectively.
4. *Real-Time Updates:* Observables and services enable real-time data updates, ensuring users receive instant notifications about changes.
5. *User-Centric Design:* The overall design focuses on enhancing user engagement through responsive features and intuitive navigation.

Thank you very much!