

Final Year Project



-----TO-----

Prof. Rajeev Puri
Computer Science Department, D.A.V. College, Jalandhar.

-----BY-----

Anurag Kaul(206301) & Himanshi(206313)
BSc IT (Information Technology)
University Roll.no-10572016161, 10572016165
Session(2020-23)

Acknowledgement

Firstly, We would like to thank our computer teacher (Mr. Rajeev Puri), because he always supported and guided us while doing this project. He very well cleared all the doubts we had regarding this project. Also, we would like to especially thank our parents and friends who helped us a lot to complete this project within the limited time. The journey of making this project has been beautiful, as well as knowledgeable for us and we have learned a lot from it. Once again, thanks to everyone who was involved with this project from beginning to end.



A Responsive Food and Recipe Web Application

Tandoori **Nights**

Content

TandoriNights

- Introduction
- Objective
- Features
- Tools and Technologies
- Hardware and Software Requirements



- Welcome to [tandoorinights](#) a food and recipe website dedicated to bringing you delicious, wholesome meals that you can prepare at home. Our goal is to provide you with a one-stop-shop for all your cooking needs, whether you're looking for quick and easy recipes for weeknight dinners, or impressive showstoppers for special occasions.
- At [tandoorinights](#) we believe that cooking is one of life's great pleasures, and that everyone can learn to create amazing meals in their own kitchen. That's why we've created a vast library of recipes, from classic comfort foods to exciting new flavor combinations, all carefully tested and adapted for home cooks of all skill levels.
- We also believe in using fresh, whole ingredients whenever possible, and in supporting local farmers and producers. So many of our recipes feature seasonal produce, sustainably-raised meats, and other wholesome ingredients that not only taste great, but are good for you and the planet.
- Whether you're a seasoned home cook or just starting out, we're confident that you'll find something to inspire you at [tandoorinights](#). So pull up a chair, browse our recipes, and get ready to take your cooking to the next level!
- [tandoorinights](#) is a place where you can please your soul and tummy with delicious food recepies of all cuisine. And our service is absolutely free. So start exploring now.

Objective

The objective of our *tandoorinights* web application is to provide a comprehensive and user-friendly platform for people who are passionate about cooking, whether they are beginners or experienced cooks. Our website aims to:

- *Inspire and educate:* We want to inspire our users to get creative in the kitchen and provide them with the tools and resources they need to expand their culinary skills and knowledge.
- *Empower and engage:* We aim to empower our users by providing them with a community of like-minded individuals who share their love of cooking and encourage them to experiment with new recipes and ingredients.
- *Simplify and streamline:* We want to make cooking and meal planning as easy and stress-free as possible by providing users with a variety of recipes, tools, and resources that help them save time and reduce waste.
- *Provide value:* We aim to provide our users with valuable content and resources that help them achieve their cooking and nutrition goals, whether that means eating healthier, cooking on a budget, or trying new cuisines.

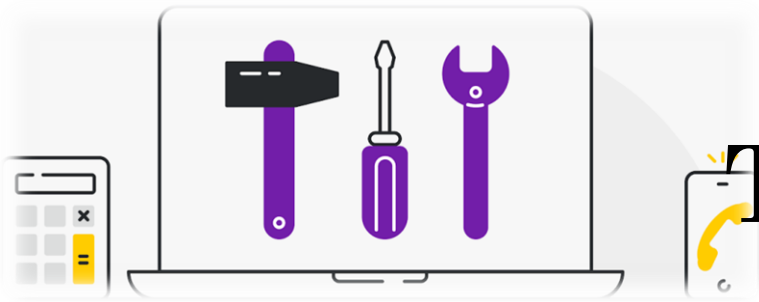
Overall, our objective is to create a welcoming and supportive community of food enthusiasts and provide them with the resources and inspiration they need to explore the world of cooking and create delicious meals that they can enjoy and share with their loved ones.

TandooriNights

FEATURES

- Recipe collection: Our website likely has a large collection of recipes that are carefully curated and tested by our team. These recipes may range from simple, everyday meals to more elaborate dishes for special occasions.
- High-quality images: Since food is a visual medium, your website may feature beautiful, high-quality photos of the dishes to entice users to try them.
- Nutritional information: If you emphasize using fresh, whole ingredients, your website may provide nutritional information for each recipe to help users make informed choices.
- Nutritional information: Many people are interested in the nutritional content of the foods they eat, so our website may provide information on the calories, macronutrients, and micronutrients of each recipe.
- Seasonal and local focus: Our website may emphasize the use of seasonal produce and locally-sourced ingredients to promote sustainability and support local food systems.
- Blog and cooking tips: In addition to recipes, Our website may feature a blog with cooking tips, ingredient guides, and other helpful resources for home cooks.
- Social media integration: Our website may have links to your social media pages where users can follow you for updates on new recipes, behind-the-scenes peeks, and more.
- Mobile responsiveness: To accommodate users who access your website from mobile devices, Our website may be optimized for mobile viewing and have a responsive design that adjusts to different screen sizes.

Tandoori Nights



Tools and Technologies

Frontend:

Our food and recipe website uses a variety of tools and technologies to create a modern, responsive, and user-friendly experience for our visitors. On the frontend, we use HTML, CSS, and JavaScript to define the structure, layout, and interactivity of our web pages, as well as the popular React.js library to build reusable UI components and manage state in complex applications.

- HTML (Hypertext Markup Language): a markup language that defines the structure of web pages.
- CSS (Cascading Style Sheets): a style sheet language that defines the presentation of web pages, including layout, typography, and colors.
- Sass (Syntactically Awesome Style Sheets) is a CSS preprocessor that allows developers to write CSS in a more efficient and maintainable way. Sass extends the functionality of CSS by providing features such as variables, nesting, mixins, functions, and more. These features help developers to write CSS code that is more reusable, modular, and easier to read and maintain.
- JavaScript: a scripting language that is commonly used to add interactivity and functionality to web pages.
- React.js: a popular JavaScript library for building user interfaces, which can be used to create reusable UI components and manage state in complex applications.

TandoriNights

Backend:



On the backend, we use Node.js as our runtime environment and rely on the MongoDB database and the Mongoose library to handle data modeling and CRUD operations. To manage file uploads in our web application, we use Multer, a middleware that allows us to upload files, save them to a specific directory, and set restrictions on the uploaded files. This ensures that our users can easily upload and access images for recipes and user profiles.

- Node.js: a JavaScript runtime that allows developers to build scalable, server-side applications using JavaScript.
- MongoDB: a NoSQL database that uses a document-oriented data model and can handle large volumes of unstructured data.

Other tools and technologies that we have used for our food and recipe website:

- Express.js: a Node.js framework for building web applications and APIs.
- Git: a version control system that allows developers to track changes to code and collaborate with others more effectively.
- Multer: is a middleware for handling file uploads in Node.js. It allows you to upload files and handle them in a convenient way, such as saving them to a specific directory, renaming them, or setting certain restrictions on the uploaded files (e.g., file type, size limit). Multer can be used in conjunction with other Node.js frameworks, such as Express.js, to handle file uploads in web applications.
- Mongoose is a Node.js library for working with MongoDB, a popular NoSQL database. Mongoose provides a way to model your data in a way that is easy to work with, by allowing you to define schemas and models that map to your MongoDB collections. This makes it easier to perform CRUD (Create, Read, Update, Delete) operations on your data, as well as other complex queries, such as aggregations and indexes.

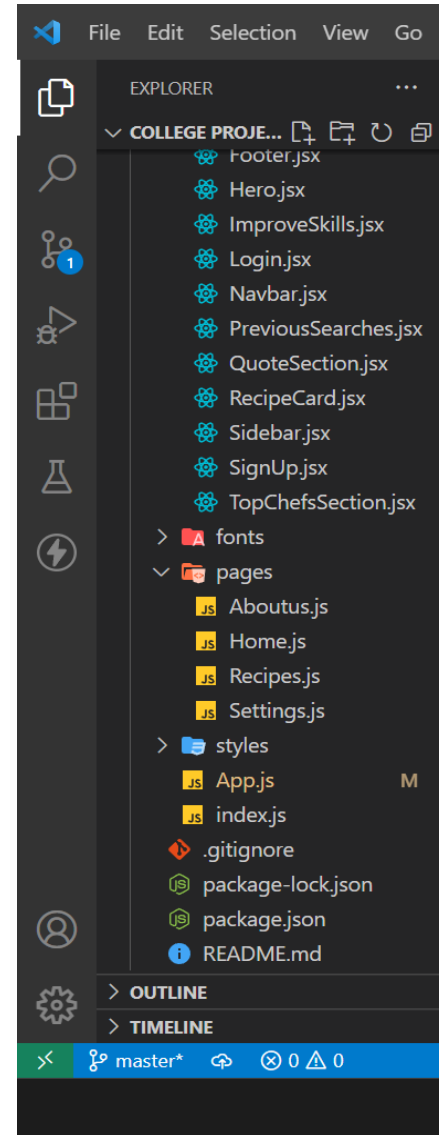
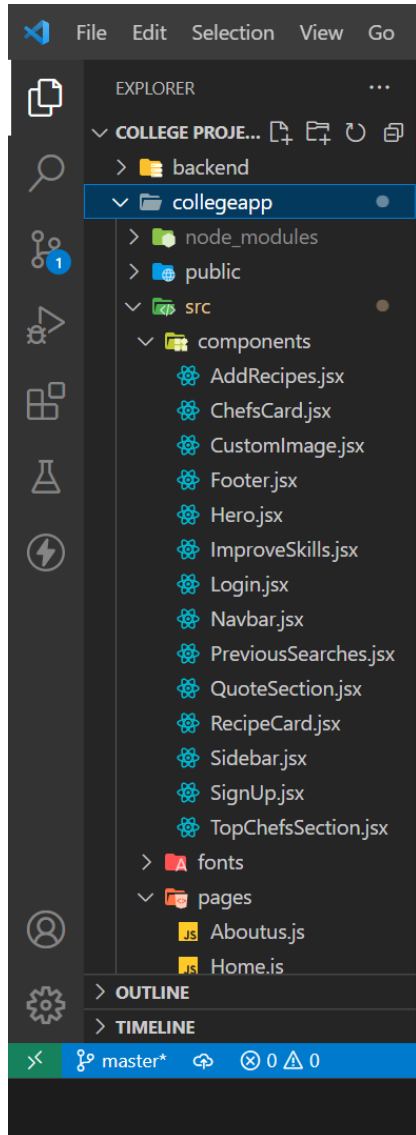
By leveraging these tools and technologies, you can build a modern, scalable, and responsive food and recipe website that meets the needs of your users and allows you to deliver high-quality content and features.

By using Multer and Mongoose in your food and recipe website, you can handle file uploads (e.g., user profile pictures, recipe images) and interact with your database in a more efficient and organized way.



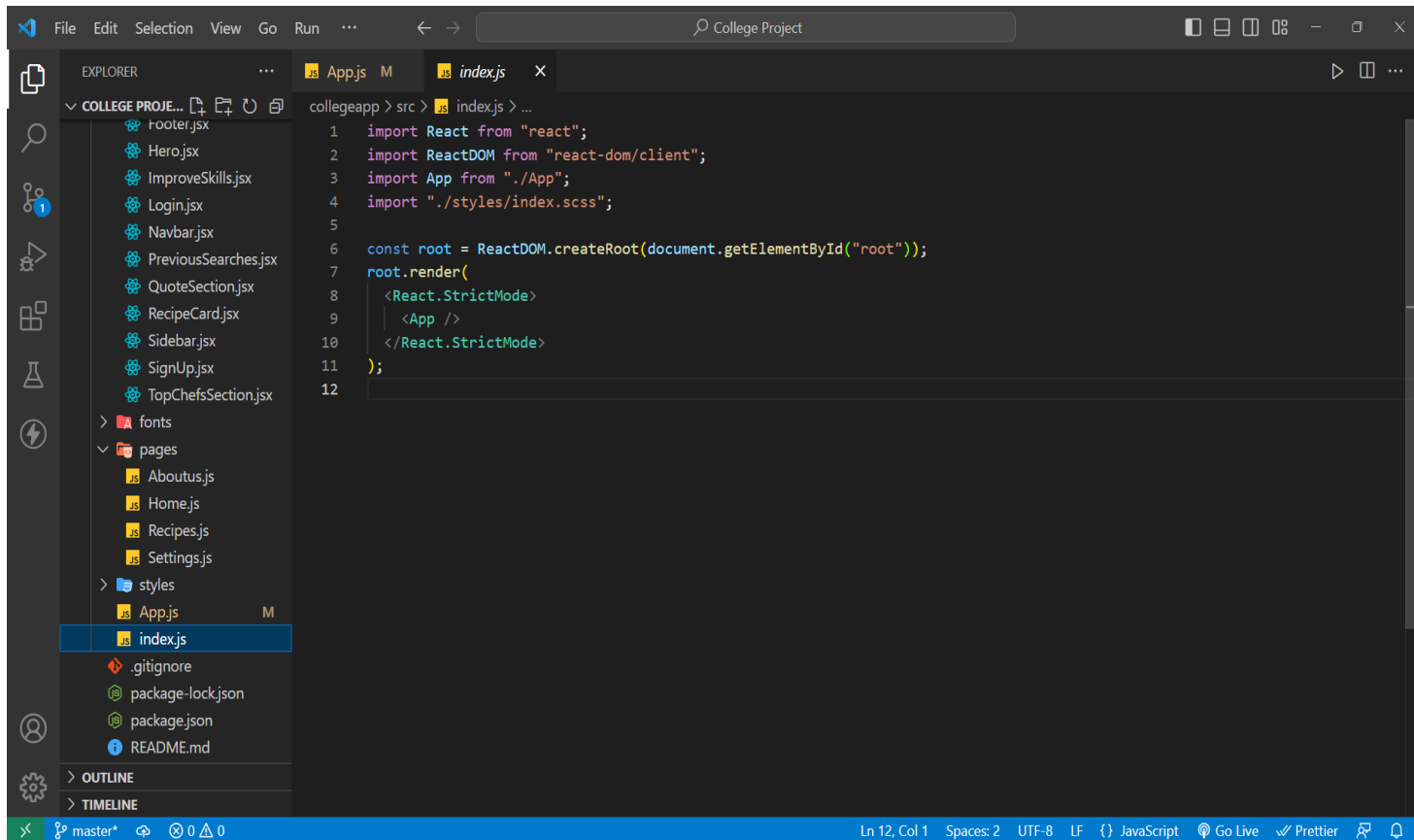
Tandoori Nights

I. Frontend: Our React Application Structure



index.js

- The index.js file is responsible for setting up the React application and rendering the root component to the HTML DOM using the ReactDOM.render() method. This method takes two arguments: the root component to be rendered and the HTML element where the component should be rendered.
- The ReactDOM.render() method renders the <App> component to the root element of the HTML DOM. The <React.StrictMode> component is a built-in component in React that helps highlight potential problems in your application during development.
- The index.js file can also include other configuration and setup code for your application, such as importing and configuring external libraries or setting up a service worker for offline support.



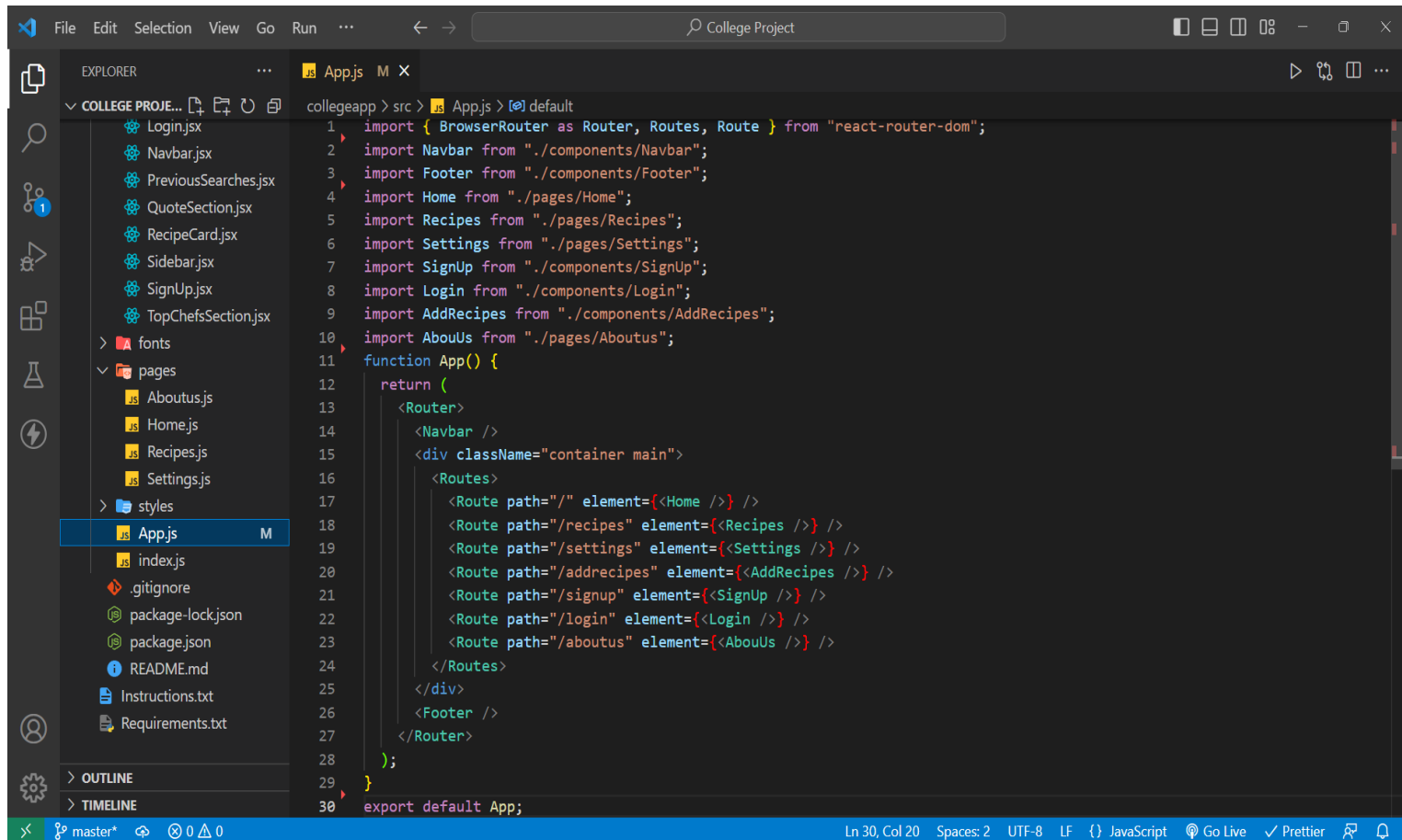
The screenshot shows a code editor with a dark theme. The Explorer panel on the left shows a project structure for 'COLLEGE PROJ...'. The file 'index.js' is selected in the 'styles' folder. The main editor area displays the code for 'index.js'.

```
1 import React from "react";
2 import ReactDOM from "react-dom/client";
3 import App from "./App";
4 import "./styles/index.scss";
5
6 const root = ReactDOM.createRoot(document.getElementById("root"));
7 root.render(
8   <React.StrictMode>
9     <App />
10  </React.StrictMode>
11 );
12
```

The status bar at the bottom indicates the file is at line 12, column 1, with 2 spaces, UTF-8 encoding, LF line endings, and is using JavaScript, Go Live, and Prettier.

App.js

- In React, App.js is a convention for the main entry point of your React application. It typically represents the top-level component of your application and is responsible for rendering all other components that make up the UI.
- The App.js file usually contains the main code for configuring your application, such as importing other components, setting up routes, and managing state. It can also contain other helper functions or components that are used throughout your application.



The screenshot shows a code editor with a dark theme. The Explorer panel on the left shows a project structure for 'COLLEGE PROJECT'. The file 'App.js' is selected and highlighted in blue. The main editor area displays the code for 'App.js', which imports various components and pages, and defines the 'App' function. The code is as follows:

```
1 import { BrowserRouter as Router, Routes, Route } from "react-router-dom";
2 import Navbar from "../components/Navbar";
3 import Footer from "../components/Footer";
4 import Home from "../pages/Home";
5 import Recipes from "../pages/Recipes";
6 import Settings from "../pages/Settings";
7 import SignUp from "../components/SignUp";
8 import Login from "../components/Login";
9 import AddRecipes from "../components/AddRecipes";
10 import AboutUs from "../pages/Aboutus";
11 function App() {
12   return (
13     <Router>
14       <Navbar />
15       <div className="container main">
16         <Routes>
17           <Route path="/" element={<Home />} />
18           <Route path="/recipes" element={<Recipes />} />
19           <Route path="/settings" element={<Settings />} />
20           <Route path="/addrecipes" element={<AddRecipes />} />
21           <Route path="/signup" element={<SignUp />} />
22           <Route path="/login" element={<Login />} />
23           <Route path="/aboutus" element={<AboutUs />} />
24         </Routes>
25       </div>
26       <Footer />
27     </Router>
28   );
29 }
30 export default App;
```

The status bar at the bottom shows the file is at line 30, column 20, with 2 spaces, UTF-8 encoding, LF line endings, and is a JavaScript file. It also indicates that Go Live, Prettier, and other extensions are active.

Pages in our Web Application:

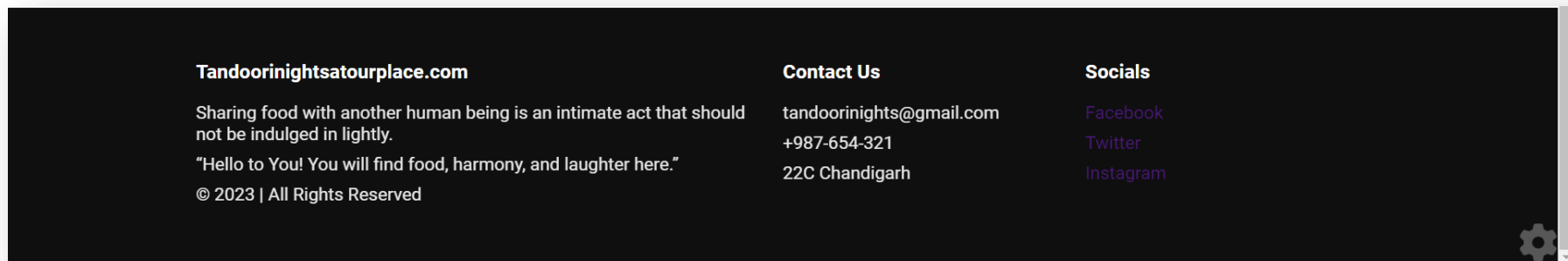
- In a React application, pages are typically organized as components that represent different views or sections of the user interface. Each page component can be further broken down into smaller, reusable components that represent specific parts of the page's UI.
1. Home Page("/")
 2. Recipes Page("/recipes")
 3. About Us Page("/aboutus")
 4. Signup Page("/signup")
 5. Login Page("/login")
 6. Settings Page("/settings")
 7. Add Recipes Page(Conditionally rendered, only if the user is logged in) ("/addrecipes")

Note: Navbar and Footer Component will be rendered in every Page.(See App.js)

<Navbar />



<Footer />



1.Home Page

Home Page React Components

- `<HeroSection />`
- `<ImproveSkills />`
- `<QuoteSection />`
- `<TopChefsSection />`

1.<HeroSection />

What Are We About?

One of the reasons that people enjoy coming to a great restaurant is that when an extraordinary meal is placed in front of them, they feel honored, respected, and even a little bit loved.

EXPLORE



2.<ImproveSkills />



Improve Your Culinary Skills

- Learn new recepies
- Experiment with food
- Write your own recepies
- Know nutrition facts
- Get cooking tips
- Get ranked

SIGNUP NOW

3.<QuoteSection />

“ Although the skills aren’t hard to learn, finding the happiness and finding the satisfaction and finding fulfillment in continuously serving somebody else something good to eat, is what makes a really good restaurant.

- Fivel Stewart, Chef.

4.<TopChefsSection />

Our Top Chefs:



Fivel Stewart
Recipes: 10
Cuisine: Chinese





Patrick Bateman
Recipes: 05
Cuisine: Mexican





Mathew Murdock
Recipes: 13
Cuisine: Italian





Keira Knightley
Recipes: 08
Cuisine: American





Joe Goldberg
Recipes: 09
Cuisine: French





Daisy Edgar-Jones
Recipes: 04
Cuisine: Indian

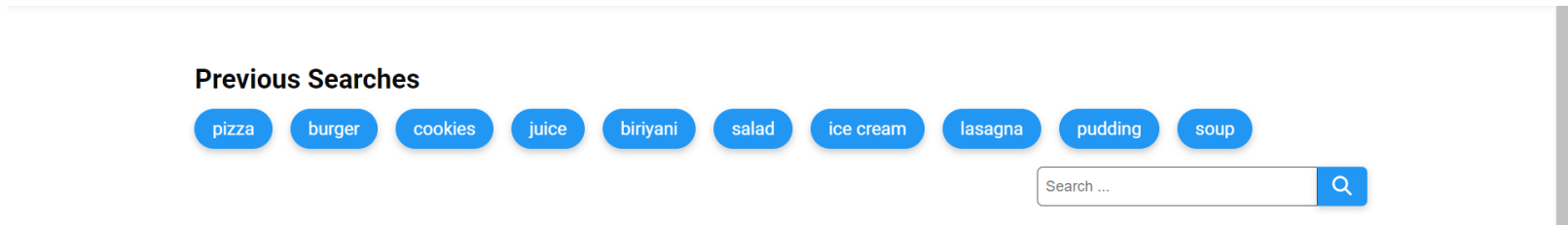


2.Recipes Page



Recipes Page React Components

- `<PreviousSearches />`
- `<RecipeCard />`

1.<PreviousSearches />





2. <RecipeCard />



Risotto

Risotto is a northern Italian rice dish cooked with broth until it reaches a creamy consistency. The broth can be derived from meat, fish, or vegetables.



[VIEW RECIPE](#)



Chicken Pan Pizza

Pan pizza is a pizza baked in a deep dish pan or sheet pan. Italian tomato pie, Sicilian pizza, Chicago-style pizza and Detroit-style pizza may be considered forms of pan pizza.



[VIEW RECIPE](#)



Chocolate soufflé

A dark chocolate ganache, deepened with the addition of unsweetened chocolate, makes a simple base that bakes up into a moist, not dry, soufflé.



[VIEW RECIPE](#)



Dal Makhni

Dal makhani is a dish originating in Punjab, India. A relatively modern variation of traditional lentil dishes, it is made with urad dal and other pulses, and includes butter and cream.



[VIEW RECIPE](#)



Japanese Sushi

Sushi is a Japanese dish of prepared vinegared rice, usually with some sugar and salt, accompanied by a variety of ingredients, such as seafood—often raw—and vegetables.



[VIEW RECIPE](#)



White Sauce Pasta

White Sauce Pasta is simply cooked pasta mixed with a silky smooth & decadent white sauce made of milk, butter and flour.



[VIEW RECIPE](#)



Napoletana Pizza

Neapolitan pizza, also known as Naples-style pizza, is a style of pizza made with tomatoes and mozzarella cheese.



[VIEW RECIPE](#)



Pancakes

A pancake is a flat cake, often thin and round, prepared from a starch-based batter that may contain eggs, milk and butter and cooked on a hot surface such as a griddle or frying pan, with oil or butter.



[VIEW RECIPE](#)



Salade Niçoise

Salade niçoise, salada nissarda in the Niçard dialect of the Occitan language, insalata nizzarda in Italian, is a salad that originated in the French city of Nice.



[VIEW RECIPE](#)



Spaghetti and Meatballs

Spaghetti and meatballs is an Italian-American dish consisting of spaghetti, tomato sauce and meatballs.



[VIEW RECIPE](#)



American Cheese Burger

Enjoy a mouth-watering meat-free burger that's full of flavor with BOCA Veggie Burgers. Made with soy protein, garlic, onion and cheddar cheese, e.t.c.

[VIEW RECIPE](#)



Spanish Tortilla

Spanish omelette or Spanish tortilla is a traditional dish from Spain. Celebrated as a national dish by Spaniards, it is an essential part of the Spanish cuisine.

[VIEW RECIPE](#)



3.About Us Page

About Us Page React Components

- `<AboutUs />`

About Us



Nimanshi

I'm the creator, promotor and passionate designer behind [tandoorinights](#). I want to change the way you use your recipes so you can spend more time cooking. If you have any feedback or need help getting setup, I'd love to hear from you.



Anurag Kaul

As the developer of [tandoorinights](#), I handle all of the technical details and making sure you have a smooth experience. If you run into any bugs or issues send me an email to get it fixed.



4.Signup Page

Signup Page React Components

1. `<SignUp />`

Sign up for an account

Username:

Email:

Password:

City:

[Sign Up](#)

5.Login Page

Login Page React Components

1. <Login />

Login to your account

Email:

Enter your email

Password:

Enter your password

Login

6.Settings Page

Settings Page React Components

1. <Settings/>

Primary Theme



Preferred color



Font size






Animation speed



6.Add Recipes Page

Add Recipes Page React Components

1. <AddRecipes/>



recipe card

Recipe Title
Enter your title

Description
Enter your description

Author
Enter your name

Ingredients
Enter your ingredients


Instructions
Enter your directions

Prep-Time
mins

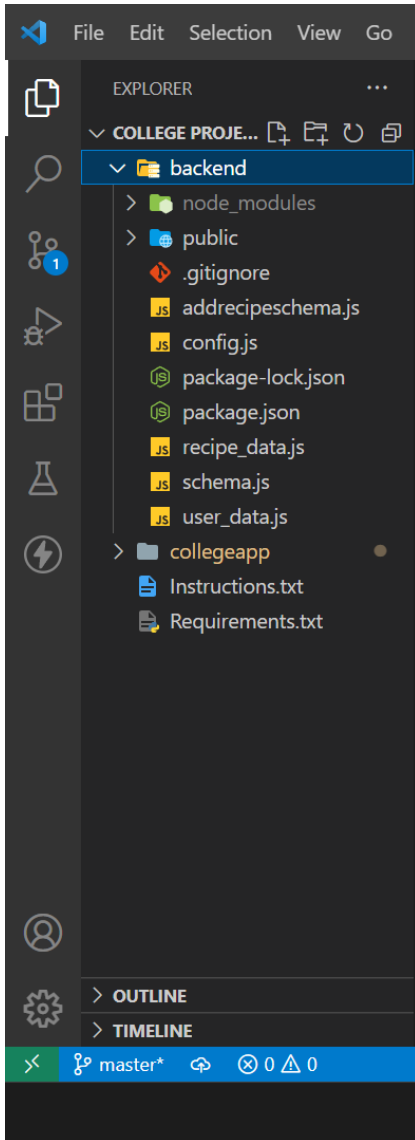
Cook-Time
mins

Total-Time
mins

Add Image
 No file chosen



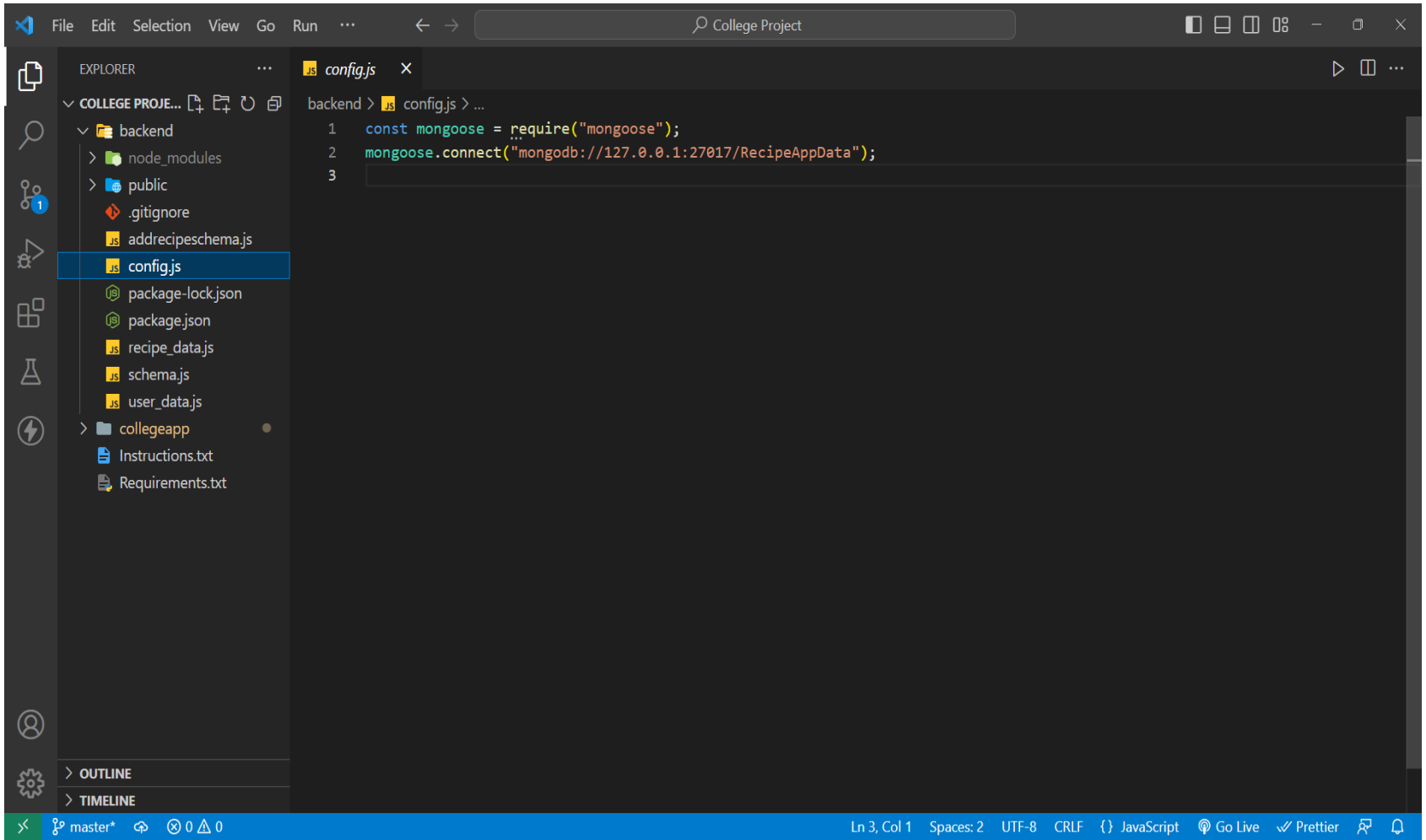
II. Backend:File Structure



We need to run two files from the backend(using node):

1. user_data.js
2. recipe_data.js

1.config.js- To establish a connection with mongodb.

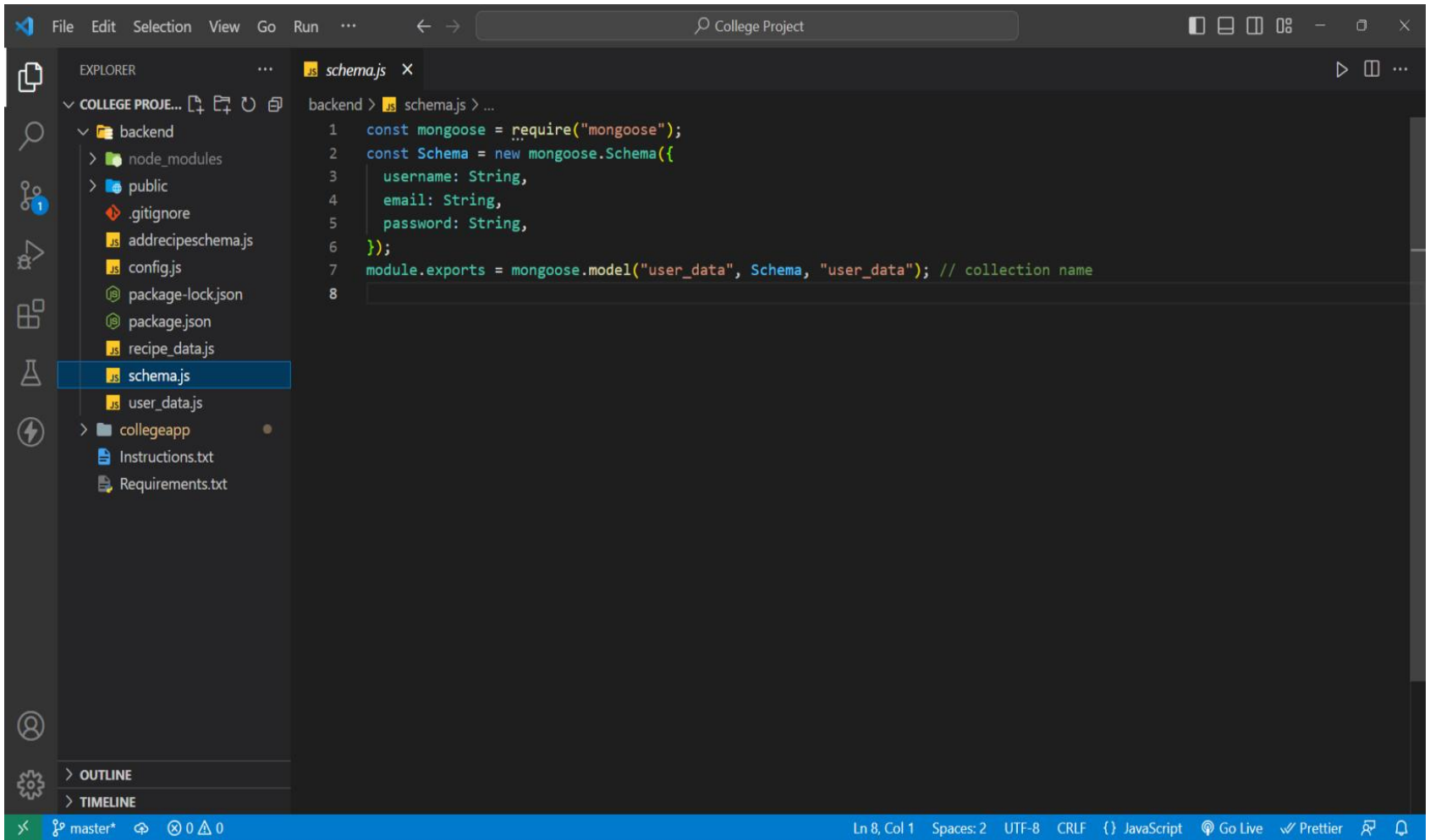


The image shows a screenshot of the Visual Studio Code (VS Code) editor interface. The title bar at the top indicates the project name is "College Project". The Explorer sidebar on the left shows the project structure, with the "backend" folder expanded. Inside "backend", the "config.js" file is selected and highlighted. The main editor area displays the content of "config.js", which contains the following code:

```
backend > JS config.js > ...  
1  const mongoose = require("mongoose");  
2  mongoose.connect("mongodb://127.0.0.1:27017/RecipeAppData");  
3
```

The status bar at the bottom of the editor shows the current file is "Ln 3, Col 1", the encoding is "UTF-8", the line endings are "CRLF", and the language is "JavaScript". It also includes icons for "Go Live", "Prettier", and other extensions.

2.schema.js-Schema for /signup data.

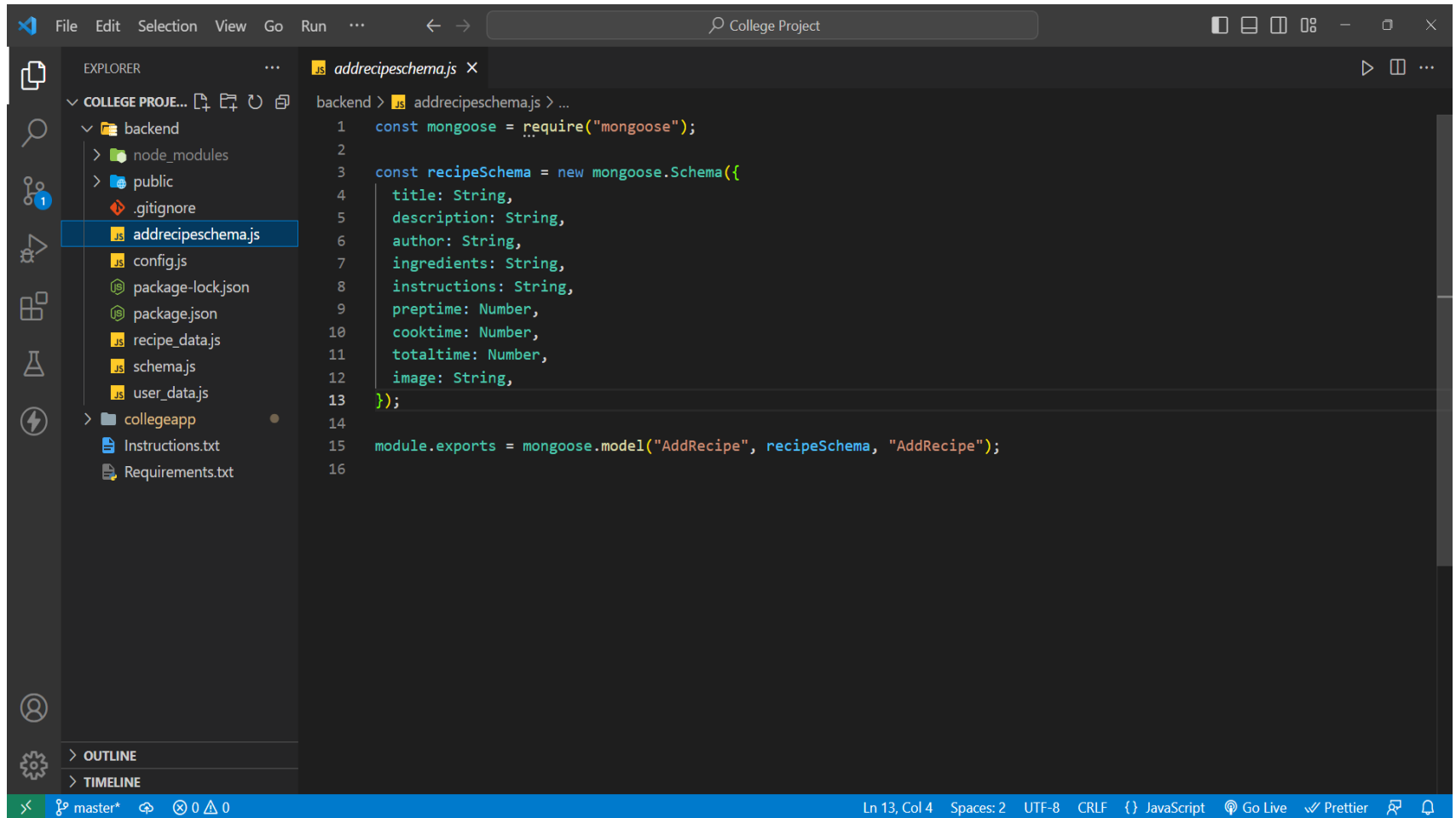


The screenshot shows the Visual Studio Code editor interface. The Explorer panel on the left displays the project structure for 'COLLEGE PROJE...'. The 'backend' directory is expanded, showing files like 'node_modules', 'public', '.gitignore', 'addrecipesschema.js', 'config.js', 'package-lock.json', 'package.json', 'recipe_data.js', 'schema.js' (selected), and 'user_data.js'. Below these are 'collegeapp' and 'Instructions.txt', 'Requirements.txt'. The main editor area shows the content of 'schema.js' with the following code:

```
backend > schema.js > ...
1  const mongoose = require("mongoose");
2  const Schema = new mongoose.Schema({
3    username: String,
4    email: String,
5    password: String,
6  });
7  module.exports = mongoose.model("user_data", Schema, "user_data"); // collection name
8
```

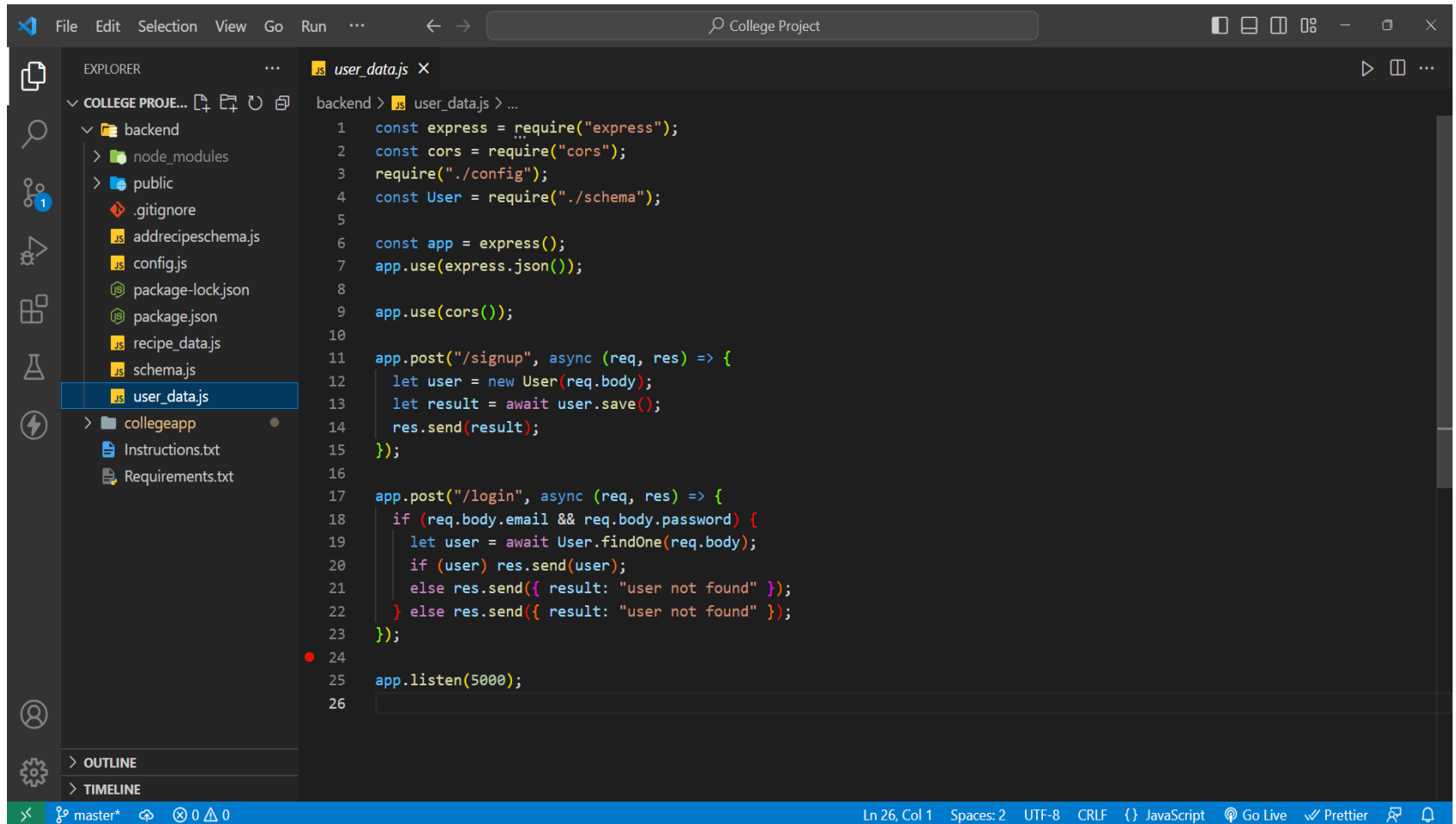
The status bar at the bottom indicates the current file is 'master*' with 0 errors, 0 warnings, and 0 info messages. It also shows the cursor position as 'Ln 8, Col 1', the encoding as 'UTF-8', and the line endings as 'CRLF'. The language is set to 'JavaScript', and the editor has 'Go Live' and 'Prettier' extensions installed.

3.addrecipeshema.js-Schema for /addrecipes data.



```
1  const mongoose = require("mongoose");
2
3  const recipeSchema = new mongoose.Schema({
4    title: String,
5    description: String,
6    author: String,
7    ingredients: String,
8    instructions: String,
9    preptime: Number,
10   cooktime: Number,
11   totaltime: Number,
12   image: String,
13 });
14
15 module.exports = mongoose.model("AddRecipe", recipeSchema, "AddRecipe");
16
```

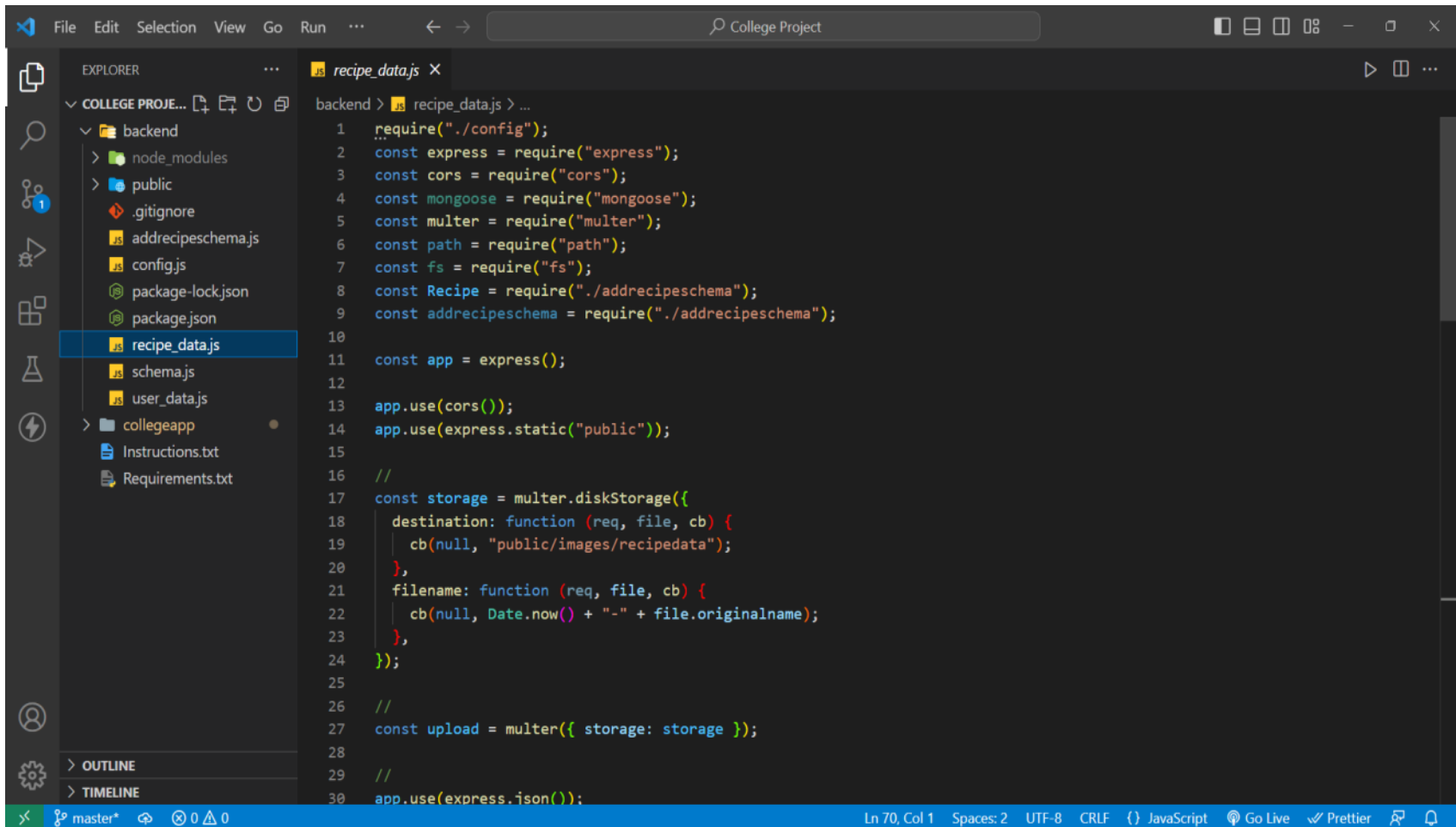

3.user_data.js-To store signup data in mongodb.



The screenshot shows the Visual Studio Code editor interface. The Explorer panel on the left displays the project structure for 'COLLEGE PROJE...'. The 'backend' folder is expanded, showing files like 'node_modules', 'public', '.gitignore', 'addrecipescema.js', 'config.js', 'package-lock.json', 'package.json', 'recipe_data.js', 'schema.js', and 'user_data.js'. The 'user_data.js' file is selected and its content is displayed in the main editor area. The code is a JavaScript file for an Express.js application, defining routes for signup and login. The status bar at the bottom indicates the file is at line 26, column 1, with 2 spaces, UTF-8 encoding, and CRLF line endings. It also shows the active language as JavaScript and various extension icons like Go Live and Prettier.

```
1  const express = require("express");
2  const cors = require("cors");
3  require("./config");
4  const User = require("./schema");
5
6  const app = express();
7  app.use(express.json());
8
9  app.use(cors());
10
11 app.post("/signup", async (req, res) => {
12   let user = new User(req.body);
13   let result = await user.save();
14   res.send(result);
15 });
16
17 app.post("/login", async (req, res) => {
18   if (req.body.email && req.body.password) {
19     let user = await User.findOne(req.body);
20     if (user) res.send(user);
21     else res.send({ result: "user not found" });
22   } else res.send({ result: "user not found" });
23 });
24
25 app.listen(5000);
26
```

4.recipe_data.js-To store Recipe data in mongodb.



The screenshot shows the Visual Studio Code editor interface. The Explorer panel on the left displays the project structure for 'COLLEGE PROJ...', with the 'backend' folder expanded. Inside 'backend', the file 'recipe_data.js' is selected. The main editor area shows the code for 'recipe_data.js', which includes imports for 'express', 'cors', 'mongoose', 'multer', 'path', and 'fs'. It also imports local modules 'addrecipescema' and 'addrecipescema'. The code sets up an Express application, uses CORS, serves static files from the 'public' directory, and configures Multer for disk storage. The storage destination is set to 'public/images/recipedata', and the filename is generated using the current date and the original filename. The Multer middleware is applied to the upload route, and the application is configured to use JSON for parsing requests.

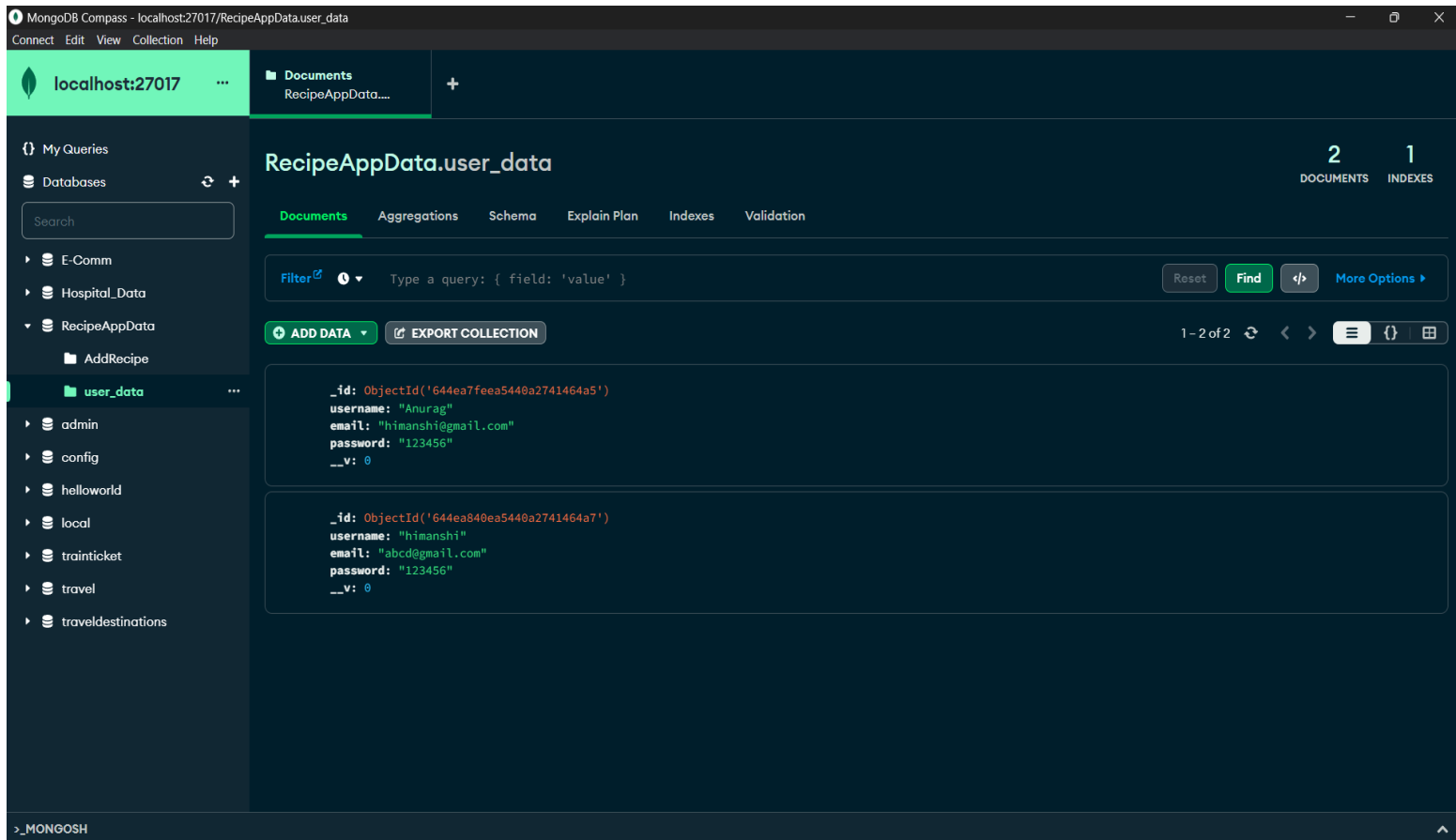
```
1  require("../config");
2  const express = require("express");
3  const cors = require("cors");
4  const mongoose = require("mongoose");
5  const multer = require("multer");
6  const path = require("path");
7  const fs = require("fs");
8  const Recipe = require("../addrecipescema");
9  const addrecipescema = require("../addrecipescema");
10
11  const app = express();
12
13  app.use(cors());
14  app.use(express.static("public"));
15
16  //
17  const storage = multer.diskStorage({
18    destination: function (req, file, cb) {
19      cb(null, "public/images/recipedata");
20    },
21    filename: function (req, file, cb) {
22      cb(null, Date.now() + "-" + file.originalname);
23    },
24  });
25
26  //
27  const upload = multer({ storage: storage });
28
29  //
30  app.use(express.json());
```

We need to run two files from the backend(cd /backend):

1. node user_data.js
2. node recipe_data.js

III. Database: mongodb

- MongoDB is a popular open-source document-oriented database that is widely used in modern web development. It is designed to store and manage large volumes of data with high performance and scalability, and is known for its flexibility and ease of use.
 - MongoDB uses a flexible document data model that allows you to store data in a variety of structures, including nested arrays and objects. This makes it well-suited for storing data in a way that reflects the structure of your application, and can help simplify data modeling and improve performance.
1. Name of the db: RecipeAppData
 2. Collection Name: user_data (user_data will store the /signup data).



1. Name of the db: RecipeAppData
2. Collection Name: AddRecipe (AddRecipe will store the /addrecipes data).

The screenshot shows the MongoDB Compass interface. The top bar indicates the connection to 'localhost:27017/RecipeAppData.AddRecipe'. The left sidebar shows the database structure with 'RecipeAppData' selected and 'AddRecipe' highlighted. The main panel displays the 'AddRecipe' collection with a single document. The document contains fields for _id, title, description, author, ingredients, instructions, preptime, cooktime, totaltime, and image. The right side of the document shows the data types for each field.

RecipeAppData.AddRecipe

DOCUMENTS: 1, INDEXES: 1

Filter: Type a query: { field: 'value' }

ADD DATA, EXPORT COLLECTION

1-1 of 1

Field	Value	Type
_id	ObjectId('644ea911d2dd3bd86408f804')	ObjectId
title	"Egg Curry,"	String
description	"This is the best and easiest egg curry available on the internet. Lets enjoy."	String
author	"Anurag Kaul,"	String
ingredients	"1/2cup neutral oil, such as grapeseed or canola 2medium yellow onions, chopped 6garlic cloves, finely chopped 1(2-inch) piece fresh ginger, peeled and finely chopped (about 2 tabl 2small cinnamon sticks 6green cardamom pods 1teaspoon ground coriander 1/2teaspoon ground turmeric 1/2teaspoon whole black peppercorns 6green cardamom pods"	String
instructions	"Step 1 In a large, heavy skillet, heat the oil over medium. Add the onions & Step 2 Add the tomatoes, salt and 1 cup water. Cook, stirring occasionally, Step 3 Add 1/2 cup of the cream and stir until it is the consistency of a"	String
preptime	15	Int32
cooktime	25	Int32
totaltime	40	Int32
image	"1682876689240-egg-curry.jpg,"	String

Hardware and Software Requirements:

1. Hardware Requirements:

- Processor: Intel i3 6th gen (minimum)
- RAM :2 GB or above
- Hard disk :40 GB or above

2. Software Requirements:

- Microsoft Visual Studio Code
- Node js
- Mongodb and mongodb compass
- **Languages used :** **Front-end:** HTML, CSS, SASS, Javascript and React js
- **Back-end:** Node js
- **DATABASE:** mongodb

Thank you!

