



# PROJECT NEXTJS

**SPAN FOODS**

**BY  
SIRISHA G  
(STS-810)**



# OVERVIEW

- INTRODUCTION
- TECHNOLOGY DESCRIPTION
- BACKEND
- FRONTEND
- DATABASE CONNECTION
- REGISTER
- LOGIN
- FOOD PAGE
- MY-ORDER PAGE

# INTRODUCTION

- Using Next.js, creating the foods application user can register and login for ordering the food.
- In Food page, only the admin can add location, mealtype, food and there will be selected location which the admin added the location and mealtype with food will be displayed and added details will be stored in mongodb.
- In My-order page, both admin and the user can order the food based on date, location, mealtype and food which are considered as Order History which is listed in the table and get saved in MongoDB.

# Technology Description

## JAVASCRIPT

- JavaScript is a light-weight object-oriented programming language which is used by several websites for scripting the webpages.
- It can be used for both **Client-side** as well as **Server-side** developments.
- It turns static HTML web pages into interactive web pages by dynamically updating content, validating form data, controlling multimedia, animate images, and almost everything else on the web pages

# NODEJS

- Node JS is an open-source and cross-platform runtime environment built on Chrome's V8 JavaScript engine for executing JavaScript code outside of a browser.
- It is actually neither a framework or a library - as with traditional application software -, but JavaScript a runtime environment.

## **Advantages:**

- Easy to Learn and Quick to adapt
- Provides fast delivery

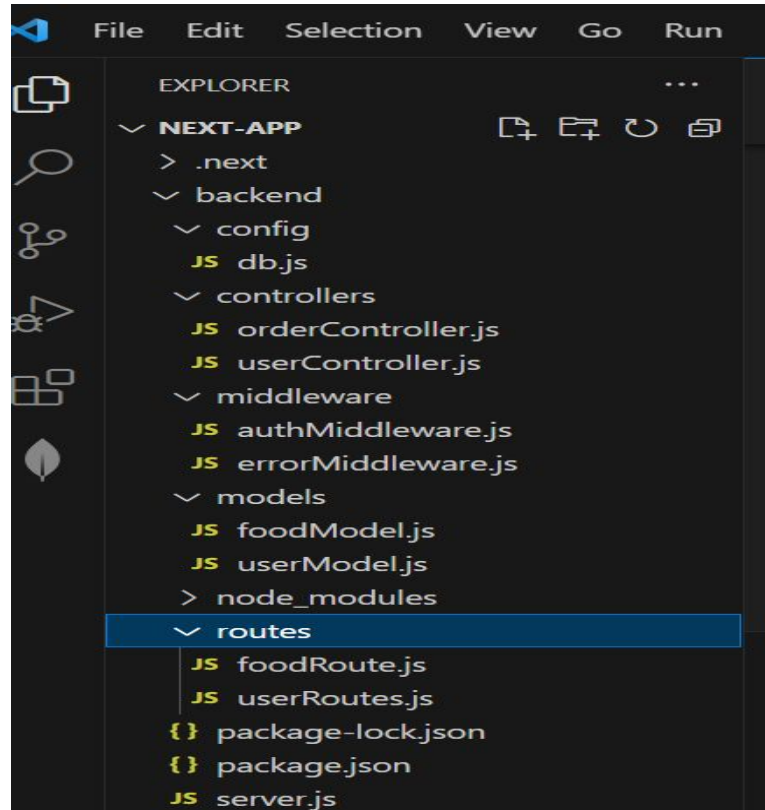
# NEXTJS

- It is an open-source web development React-based framework created by Vercel, which is famous for its unique features such as Server-side rendering and enhanced SEO.
- It is a react based framework.and It has powers to Develop beautiful Web application for different platform like Windows, Linux and mac.

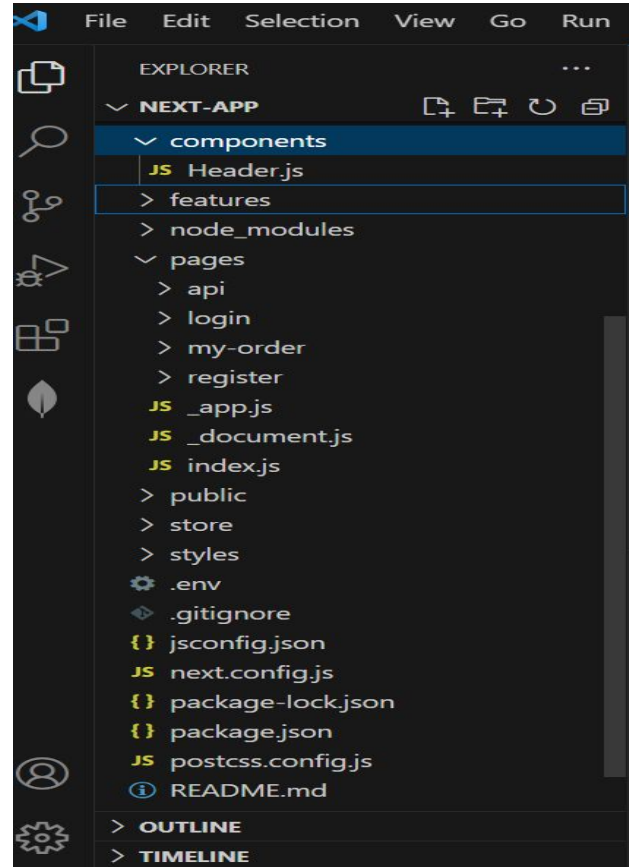
## **Advantages:**

- Faster development process
- Data Security

# BACKEND



# FRONTEND





# DATABASE CONNECTION

## Connecting with MongoDB Driver

### 1. Select your driver and version

We recommend installing and using the latest driver version.

Driver

Node.js

Version

5.5 or later

### 2. Install your driver

Run the following on the command line

```
npm install mongodb
```



[View MongoDB Node.js Driver installation instructions.](#)

### 3. Add your connection string into your application code

☐ View full code sample

```
mongodb+srv://sirisha:<password>@cluster0.cp3qlbh.mongodb.net/?  
retryWrites=true&w=majority
```



Replace **<password>** with the password for the **sirisha** user. Ensure any option params are [URL encoded](#).

# Backend DB Connection

```
backend > config > JS db.js > [?] <unknown>
1  const mongoose = require('mongoose')
2  const MONGO_URI = "mongodb+srv://sirisha:Sirisha28@cluster0.cp3qlbh.mongodb.net/?retryWrites=true&w=majority"
3  const connectDB = async () => {
4    try {
5      const conn = await mongoose.connect(MONGO_URI)
6      console.log(`MongoDB Connected`)
7    } catch (error) {
8      console.log(`Error: ${error.message}`)
9    }
10  }
11
12
13  module.exports = connectDB
```

# Backend API Endpoints

```
backend > JS server.js > ...
1  const path = require('path')
2  const express = require('express')
3  require('dotenv').config()
4  const { errorHandler } = require('./middleware/errorMiddleware')
5  const connectDB = require('./config/db')
6  const PORT = process.env.PORT || 5000
7  const cors = require('cors');
8  // Connect to database
9  connectDB()
10
11  const app = express()
12
13  app.use(express.json())
14  app.use(cors());
15  app.use(express.urlencoded({ extended: false }))
16  app.use('/api/foods', require('./routes/foodRoute'))
17  // Routes
18  app.use('/api/users', require('./routes/userRoutes'))
19  // app.use('/api/foods', require('./routes/foodRoute'))
20
21  app.use(errorHandler)
22
23  app.listen(PORT, () => console.log(`Server started on port ${PORT}`))
```

# FRONTEND -HOME PAGE

pages > JS index.js > Foods

```
1  import React, { useState } from 'react';
2  import { toast } from 'react-toastify';
3  import { useRouter } from 'next/router';
4  import Header from '@components/Header';
5  function Foods() {
6    const [location, setLocation] = useState('');
7    const [locationError, setLocationError] = useState('');
8    const [locations, setLocations] = useState([]);
9    const [mealType, setMealType] = useState('');
10   const [mealTypeError, setMealTypeError] = useState('');
11   const [mealTypes, setMealTypes] = useState([]);
12   const [selectedLocation, setSelectedLocation] = useState('');
13   const [selectedMealType, setSelectedMealType] = useState('');
14   const [food, setFood] = useState('');
15   const [foodError, setFoodError] = useState('');
16   const [foods, setFoods] = useState([]);
17   const [mealTypeItems, setMealTypeItems] = useState({
18     breakfast: ['idly', 'dosa', 'pongal', 'upma', 'semiya'],
19     morningsnacks: ['Tea', 'Coffee', 'biscuit', 'milk'],
20     lunch: ['sambar rice', 'rasam rice', 'curd rice', 'briyani', 'fried rice'],
21     eveningSnacks: ['samosa', 'puffs', 'tea', 'coffee'],
22     dinner: ['chapathi', 'parota', 'dosa', 'poori', 'plain roast']
23   });
```

# REGISTER PAGE

Span Foods

Login

Register

## Register

**Username**

**Email**

**Emp ID**

**Password**



**Confirm Password**



**Admin** ☐

Register

# LOGIN PAGE

Span Foods

Login

Register

## Login

Email

Emp ID

Password



Submit

# FOOD PAGE

Span Foods

Login

Register

## Food page

Location:

Submit Location

Meal Type:

Submit Meal Type

Select Location:

Select Meal Type:

Select MealType

Select Meal Type:

Food:

Submit Food

View my Orders

# MY-ORDER PAGE

Span Foods

Login

Register

## My Order Page

Date:

dd-mm-yyyy



Select Location:

Select Location



Select Meal Type:

Select MealType



Food Items:

Select Food



Submit Order

Date	Location	Meal Type	Food Items
18-11-2023	Coimbatore	Breakfast	idly
18-11-2023	Chennai	Lunch	briyani
19-11-2023	Coimbatore	Breakfast	idly



A large, irregular watercolor splash in various shades of teal and blue, centered on a white background. The splash has a soft, painterly texture with darker blue in the center and lighter teal towards the edges.

*Thank You*