**1. Project Root**

* **app.js or server.js:** Entry point of your application where you set up the Express server, middleware, and routes.
* **package.json:** Lists your dependencies, scripts, and project metadata.
* **.env:** Environment variables like database connection strings.
* **config/:** Configuration files, e.g., for connecting to MongoDB.

**2. models/ (Model)**

* **models/:** This folder contains the schema and data models for your application, which interact with MongoDB.
  + **userModel.js:** Example model for user data.
  + **productModel.js:** Example model for product data.

**3. controllers/ (Controller)**

* **controllers/:** This folder holds the logic for handling requests, processing data, and interacting with models.
  + **userController.js:** Handles user-related actions (e.g., login, registration).
  + **productController.js:** Manages product-related actions.

**4. routes/**

* **routes/:** Contains the route definitions that map URLs to controllers.
  + **userRoutes.js:** Defines routes for user-related actions, e.g., /login, /register.
  + **productRoutes.js:** Defines routes for product-related actions, e.g., /products, /product/:id.

**5. views/ (View)**

* **views/:** This folder contains the HTML (or templating files) that render the user interface.
  + **index.html or index.ejs:** Main HTML file.
  + **partials/:** Common reusable components like headers, footers, etc.
  + **user/:** Views related to user actions (e.g., login.html, register.html).

**6. public/**

* **public/:** Contains static files like CSS, JavaScript, images, fonts, etc.
  + **css/:** Stylesheets (e.g., styles.css).
  + **js/:** Client-side JavaScript files (e.g., main.js).
  + **images/:** Image assets.
  + **fonts/:** Custom fonts if any.

**7. middleware/**

* **middleware/:** Custom middleware functions.
  + **authMiddleware.js:** Example for handling authentication.
  + **errorHandler.js:** Custom error handling middleware.

**8. config/**

* **config/:** Configuration settings for different environments (e.g., development, production).
  + **db.js:** Database configuration and connection to MongoDB.

**9. helpers/ (optional)**

* **helpers/:** Utility functions or classes used across the application.

**10. tests/ (optional)**

* **tests/:** Contains unit tests and integration tests for your application.