**File Structure for Todo App**

project-root/

|— conn/

| |— conn.js # Database connection file

|— models/

| |— user.js # User model

| |— list.js # List model

|— routes/

| |— auth.js # Authentication routes

| |— list.js # Task-related routes

|— frontend/

| |— build/ # React frontend build files

|— node\_modules/ # Node.js dependencies

|— .env # Environment variables (hidden, not shared)

|— app.js # Main server file

|— package.json # Project configuration and dependencies

|— package-lock.json # Dependency tree lock file

**How to Run the Project**

**1. Setup Prerequisites**

* Make sure you have Node.js installed. [Download Node.js](https://nodejs.org/)
* Install MongoDB and ensure it is running locally or provide a MongoDB Atlas URI.

**2. Clone the Repository**

Clone the project repository from your version control system:

git clone <repository-url>

cd project-root

**3. Install Dependencies**

Install the required Node.js dependencies:

npm install

**4. Set Environment Variables**

Create a .env file in the project root with the following variables:

PORT=1000

MONGO\_URI=<your-mongodb-connection-string>

JWT\_SECRET=<your-secret-key>

Replace <your-mongodb-connection-string> with your MongoDB URI and <your-secret-key> with a secure string for JWT authentication.

**5. Build the Frontend**

If you are using a React frontend, navigate to the frontend directory and build it:

cd frontend

npm install

npm run build

This will generate the build/ folder inside the frontend/ directory.

**6. Start the Server**

Start the server using one of the following commands:

* **For Development:**

npm run dev

This uses nodemon for automatic server restarts on file changes.

* **For Production:**

npm start

**7. Access the Application**

Open your browser and go to:

http://localhost:1000

**Key Commands**

* Install dependencies:

npm install

* Start the server in development mode:

npm run dev

* Build the React frontend:

npm run build

* Start the server in production mode:

npm start

**Additional Notes**

* **Database Configuration:** Ensure your MongoDB service is running locally or provide a valid connection string for a remote MongoDB instance in the .env file.
* **React Development Server:** If you want to test the React app separately, navigate to the frontend/ directory and run:

npm start

This will start the React development server on http://localhost:3000.

* **Static Serving:** Ensure the frontend/build/ folder is correctly generated to serve the React app through the Node.js server.

By following these steps, you should be able to run and test your Todo App successfully.