

# Simple Note App

This is a simple REST API application built with Node.js, Express, and MongoDB. The application allows users to create and manage notes. It demonstrates the use of a MongoDB database for storing and retrieving data, with Mongoose as the ORM.

---

## Features

- **Create Notes:** Users can create notes with a title and content.
  - **MongoDB Integration:** Notes are stored in a MongoDB database.
  - **RESTful API:** Follows REST principles for easy integration.
- 

## Prerequisites

1. **Node.js** installed on your machine.
  2. **MongoDB** installed and running locally or hosted on MongoDB Atlas.
  3. **Postman** or any other API testing tool (optional, for testing).
- 

## Installation and Setup

### 1. Clone the Repository

```
git clone https://github.com/your-username/simple-note-app.git
cd simple-note-app
```

### 2. Install Dependencies

```
npm install
```

### 3. Configure the Database

- By default, the application connects to a local MongoDB instance:

```
mongodb://localhost:27017/simple_notes
```

- To use a remote MongoDB (e.g., Atlas), update the connection string in the `server.js` file:

```
mongoose.connect('your-mongodb-connection-string', { useNewUrlParser:
true, useUnifiedTopology: true });
```

## 4. Start the Server

```
node server.js
```

The server will start and run on:

```
http://localhost:3000
```

---

# Usage

## 1. Test the API Endpoints

### Create a Note

- **URL:** POST /notes
- **Request Body (JSON):**
  - {
  - "title": "Sample Note",
  - "content": "This is a sample note."
  - }
- **Response (JSON):**
  - {
  - "\_id": "<note-id>",
  - "title": "Sample Note",
  - "content": "This is a sample note.",
  - "createdAt": "<timestamp>",
  - "\_\_v": 0
  - }

## 2. Additional Endpoints

- Additional endpoints like GET, PUT, or DELETE for notes can be added as needed.
- 

# Project Structure

```
.
├── models
│   └── Note.js      # Defines the Note schema and model
├── server.js        # Main application entry point
└── package.json     # Project dependencies and scripts
```

## Contributing

Contributions are welcome! Please follow these steps:

1. Fork the repository.
  2. Create a new branch (`git checkout -b feature-name`).
  3. Commit your changes (`git commit -m "Add feature"`).
  4. Push to the branch (`git push origin feature-name`).
  5. Open a pull request.
- 

## Contact

For any questions or support, please contact:

- **Email:** ahsanmahmood923@gmail.com
- **GitHub:** [ahsanmahmood923](#)