Node.js and MongoDB Project Setup Guide

Here's a step-by-step guide to set up a Node.js + MongoDB project with all the necessary dependencies:

Step 1: Install Node.js and MongoDB

- Ensure that Node.js and MongoDB are installed on your system.
  - Download Node.js: https://nodejs.org/
  - Download MongoDB: https://www.mongodb.com/try/download/community (or use MongoDB Atlas for a cloud database).

Step 2: Set Up a Node.js Project

1. Initialize a Node.js project:

```
mkdir my-node-app
cd my-node-app
npm init -y
```

2. Install the required dependencies:

```
npm install express mongoose dotenv cors
```

   - express: Web framework for building RESTful APIs.
   - mongoose: Library for MongoDB interactions.
   - dotenv: For managing environment variables.
   - cors: For handling cross-origin requests from the frontend.

3. Install nodemon (optional, but useful for development):

```
npm install --save-dev nodemon
```

4. Update package.json to use nodemon:

```
"scripts": {
  "start": "node server.js",
  "dev": "nodemon server.js"
}
```

Step 3: Create a Basic Express Server

1. Create server.js:

```
const express = require('express');
const mongoose = require('mongoose');
const cors = require('cors');
const dotenv = require(‘dotenv');

dotenv.config();

const app = express();
const PORT = process.env.PORT || 8080;

app.use(cors());
app.use(express.json());

mongoose.connect(process.env.MONGO_URI, {
  useNewUrlParser: true,
  useUnifiedTopology: true,
}).then(() => console.log(‘Connected to MongoDB'))
  .catch((err) => console.log(err));
app.get('/', (req, res) => {
  res.send('Hello from the server');
});

app.listen(PORT, () => {
  console.log(`Server running on port ${PORT}`);
});
```

2. Create a .env file for environment variables:

```
  PORT=8080
  MONGO_URI=mongodb://localhost:27017/mydatabase
```

Step 4: Set Up MongoDB

1. If using local MongoDB, ensure that MongoDB is running:

```
  mongod
```

2. If using MongoDB Atlas (cloud-based):

   Sign up on MongoDB Atlas, create a cluster, and update the MONGO_URI in your .env file.

Step 5: Run the Node.js App

1. Run the server:

```
  npm run dev (for nodemon) or npm start
```

Step 6: Additional Useful Dependencies

1. bcryptjs (for password hashing):

```
  npm install bcryptjs
```

2. jsonwebtoken (for authentication):

```
   npm install jsonwebtoken
```

3. morgan (for logging HTTP requests):

```
   npm install morgan
```

4. express-validator (for validation):

```
   npm install express-validator
```

Step 7: MongoDB Model Example

```
const mongoose = require('mongoose');

const userSchema = new mongoose.Schema({
  name: String,
  email: String,
  password: String,
}, { timestamps: true });

module.exports = mongoose.model('User', userSchema);
```