

Here is a skeleton code for a Node.js project using Express, ES6, EJS, environmental variables, and MongoDB:

Directory Structure:

my-node-project/

??? config/

? ??? db.js

??? controllers/

? ??? exampleController.js

??? models/

? ??? Example.js

??? routes/

? ??? exampleRoutes.js

??? views/

? ??? index.ejs

??? .env

??? .gitignore

??? package.json

??? server.js

1. package.json

Run `npm init -y` to create a package.json file, then install necessary dependencies:

`npm install express dotenv mongoose ejs`

`npm install --save-dev nodemon`

Update package.json scripts:

"scripts": {

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

2. .env

PORT=3000

MONGO_URI=mongodb://localhost:27017/mydatabase

3. .gitignore

node_modules/

.env

4. config/db.js

```
import mongoose from 'mongoose';
```

```
const connectDB = async () => {  
  
  try {  
  
    await mongoose.connect(process.env.MONGO_URI, {  
  
      useNewUrlParser: true,  
  
      useUnifiedTopology: true,  
  
    });  
  
    console.log('MongoDB connected...');  
  
  } catch (err) {  
  
    console.error(err.message);  
  
    process.exit(1);  
  
  }  
  
};
```

```
export default connectDB;
```

5. models/Example.js

```
import mongoose from 'mongoose';
```

```
const exampleSchema = new mongoose.Schema({  
  
  name: {  
  
    type: String,  
  
    required: true,  
  
  },  
  
  createdAt: {  
  
    type: Date,  
  
    default: Date.now,  
  
  },  
  
});
```

```
const Example = mongoose.model('Example', exampleSchema);
```

```
export default Example;
```

6. controllers/exampleController.js

```
import Example from '../models/Example.js';
```

```
export const getExamples = async (req, res) => {  
  
  try {  
  
    const examples = await Example.find();
```

```

    res.render('index', { examples });

  } catch (err) {

    res.status(500).send('Server Error');

  }

};

export const createExample = async (req, res) => {

  try {

    const { name } = req.body;

    const newExample = new Example({ name });

    await newExample.save();

    res.redirect('/');

  } catch (err) {

    res.status(500).send('Server Error');

  }

};

```

7. routes/exampleRoutes.js

```

import express from 'express';

import { getExamples, createExample } from '../controllers/exampleController.js';

const router = express.Router();

router.get('/', getExamples);

router.post('/', createExample);

export default router;

```

8. views/index.ejs

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Example App</title>

</head>

<body>

  <h1>Examples</h1>

  <ul>

    <% examples.forEach(example => { %>

      <li><%= example.name %></li>

    <% }) %>

  </ul>

  <form action="/" method="POST">

    <input type="text" name="name" placeholder="Add Example" required>

    <button type="submit">Submit</button>

  </form>

</body>

</html>
```

9. server.js

```
import express from 'express';

import dotenv from 'dotenv';

import connectDB from './config/db.js';
```

```
import exampleRoutes from './routes/exampleRoutes.js';
```

```
dotenv.config();
```

```
// Connect to MongoDB
```

```
connectDB();
```

```
const app = express();
```

```
// Middleware
```

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

```
app.use(express.urlencoded({ extended: true }));
```

```
app.use(express.static('public'));
```

```
// Set EJS as the templating engine
```

```
app.set('view engine', 'ejs');
```

```
// Routes
```

```
app.use('/', exampleRoutes);
```

```
const PORT = process.env.PORT || 5000;
```

```
app.listen(PORT, () => {
```

```
  console.log(`Server running on port ${PORT}`);
```

```
});
```

Run the Project

1. Start MongoDB.

2. Run the server in development mode:

```
npm run dev
```

3. Visit <http://localhost:3000>.

This setup provides a basic structure for a Node.js project with Express, ES6, EJS, environment variables, and MongoDB.