Lecture 11. MVC in Nodejs

12 April 2025 14:52

MVC Architecture in Node.js – Revision Summary

✓ What is MVC?

MVC = Model + View + Controller

It's a design pattern used to separate the logic (Controller), data (Model), and presentation (View) in an application for better structure, scalability, and maintainability.

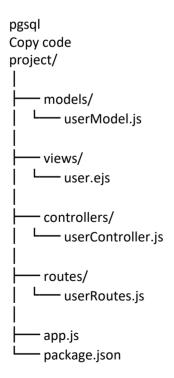
Components of MVC

Component	Role	Location
Model	Manages data and business logic	/models
View	UI/HTML templates	/views
Controller	Handles input, connects model & view	/controllers

🖫 How MVC Works

- 1. User sends a request (GET/POST)
- 2. Controller processes the request
- 3. Model is used to interact with database
- 4. Controller sends data to View
- 5. View renders HTML with data

Typical Folder Structure



Example: Simple User MVC in Node.js with Express + **MongoDB**

1 Model (models/userModel.js)

```
js
Copy code
const mongoose = require('mongoose');
const userSchema = new mongoose.Schema({
    name: String,
    email: String,
});
module.exports = mongoose.model('User', userSchema);
```

2 Controller (controllers/userController.js)

```
js
Copy code
const User = require('../models/userModel');
exports.getUsers = async (req, res) => {
  const users = await User.find();
  res.render('user', { users });
};
exports.addUser = async (req, res) => {
  const { name, email } = req.body;
  await User.create({ name, email });
  res.redirect('/users');
};
```

3 View (views/user.ejs)

```
ejs
Copy code
<!DOCTYPE html>
<html>
<head><title>Users</title></head>
<body>
<h1>User List</h1>

<% users.forEach(user => { %>
<% = user.name %> - <% = user.email %>
</wl>

</body>

Copy code

<br/>
<br/
```

4 Routes (routes/userRoutes.js)

```
js
Copy code
const express = require('express');
const router = express.Router();
const userController = require('../controllers/userController');
router.get('/users', userController.getUsers);
router.post('/users', userController.addUser);
module.exports = router;
```

5 App Entry Point (app.js)

```
js
Copy code
const express = require('express');
const mongoose = require('mongoose');
const userRoutes = require('./routes/userRoutes');
const app = express();
app.set('view engine', 'ejs');
app.use(express.urlencoded({ extended: true }));
mongoose.connect('mongodb://127.0.0.1:27017/mvcDB')
.then(() => console.log('MongoDB connected'));
app.use('/', userRoutes);
app.listen(3000, () => console.log('Server started on port 3000'));
```

Quick Benefits of MVC

- © Clean separation of concerns
- Easier to test and debug
- Ø Modular codebase
- Scalable architecture for growing apps

***** Summary Table

MVC PartRoleExample FileModelData logicuserModel.jsViewHTML rendering (e.g. EJS)user.ejsControllerLogic & routing handlinguserController.js