

The screenshot shows the Visual Studio Code editor with the `server.js` file open. The Explorer sidebar on the left shows the project structure for `ASG3B`, including `config`, `database.config.js`, `controllers`, `user.controller.js`, `models`, `user.model.js`, `node_modules`, `public`, `index.html`, `script.js`, `style.css`, `routes`, `user.routes.js`, `package-lock.json`, `package.json`, and `server.js`. The `server.js` file contains the following code:

```
11 server.js > ...
12 // MongoDB Connection
13 mongoose.connect(dbConfig.url, { useNewUrlParser: true, useUnifiedTopology: true })
14   .then(() => console.log('Database Connected'))
15   .catch(err => console.log('DB Connection Error:', err));
16
17 // Serve static files (index.html, css, js) from ./public folder
18 app.use(express.static(path.join(__dirname, 'public')));
19
20 // API Routes
21 app.use('/users', require('./routes/user.routes'));
22
23 // Fallback to index.html for root URL
24 app.get('/', (req, res) => {
25   res.sendFile(path.join(__dirname, 'public', 'index.html'));
26 });
27
28 // Start Server
29 app.listen(5000, () => console.log('Server running on http://localhost:5000'));
30
```

The TERMINAL panel at the bottom shows the output of running `node server.js` in a PowerShell window:

```
PS C:\Users\hp\OneDrive\Desktop\wdp\asg3b> node server.js
(node:24844) [MONGODB DRIVER] Warning: useNewUrlParser is a deprecated option: useNewUrlParser has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
(node:24844) [MONGODB DRIVER] Warning: useUnifiedTopology is a deprecated option: useUnifiedTopology has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
Server running on http://localhost:5000
Database Connected
[]
```

The screenshot shows the Visual Studio Code editor with the `user.model.js` file open. The Explorer sidebar on the left shows the project structure for `ASG3B`, including `config`, `database.config.js`, `controllers`, `user.controller.js`, `models`, `user.model.js`, `node_modules`, `public`, `index.html`, `script.js`, `style.css`, `routes`, `user.routes.js`, `package-lock.json`, `package.json`, and `server.js`. The `user.model.js` file contains the following code:

```
1 models > JS user.model.js > ...
2 const mongoose = require('mongoose');
3
4 const UserSchema = mongoose.Schema({
5   email: { type: String, required: true, unique: true },
6   firstName: { type: String, required: true },
7   lastName: { type: String, required: true },
8   phone: { type: String, required: true },
9   timestamps: true });
10
11 module.exports = mongoose.model('User', UserSchema);
```

```
File Edit Selection View Go Run Terminal Help asg3b
EXPLORER server.js index.html style.css package.json script.js database.config.js user.model.js user.controller.js user.routes.js
ASG3B
  config
  database.config.js
  controllers
  user.controller.js
  models
  user.model.js
  node_modules
  public
    index.html
    script.js
    style.css
  routes
  user.routes.js
  package-lock.json
  package.json
  server.js
  OUTLINE
  TIMELINE

public > index.html > html > body
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>User Management</title>
7   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
8   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
9
10  <link rel="stylesheet" href="style.css">
11 </head>
12 <body>
13   <div class="container mt-5">
14     <h2 class="text-center">User Management</h2>
15     <form id="userForm">
16       <div class="mb-3">
17         <input type="text" id="firstName" class="form-control" placeholder="First Name" required>
18       </div>
19       <div class="mb-3">
20         <input type="text" id="lastName" class="form-control" placeholder="Last Name" required>
21       </div>
22       <div class="mb-3">
23         <input type="email" id="email" class="form-control" placeholder="Email" required>
24       </div>
25       <div class="mb-3">
26         <input type="text" id="phone" class="form-control" placeholder="Phone Number" required>
27       </div>
28       <button type="submit" class="btn btn-primary">Add User</button>
29       <button type="button" id="resetBtn" class="btn btn-secondary">Reset</button>
30     </form>
31
32     <h3 class="mt-5">Find User</h3>
33     <div class="input-group mb-3">
34       <input type="email" id="searchEmail" class="form-control" placeholder="Enter Email to Search">
35       <button class="btn btn-success" id="searchBtn">Find</button>
36     </div>
37     <div id="searchResult"></div>
38
39   </div>
40 </body>
```

```
File Edit Selection View Go Run Terminal Help asg3b
EXPLORER server.js index.html style.css package.json script.js database.config.js user.model.js user.controller.js user.routes.js
ASG3B
  config
  database.config.js
  controllers
  user.controller.js
  models
  user.model.js
  node_modules
  public
    index.html
    script.js
    style.css
  routes
  user.routes.js
  package-lock.json
  package.json
  server.js
  OUTLINE
  TIMELINE

controllers > user.controller.js >
1 const User = require("../models/user.model");
2
3 exports.create = async (req, res) => {
4   try {
5     const user = new User(req.body);
6     await user.save();
7     res.status(201).json({ message: 'User created!', user });
8   } catch (error) {
9     res.status(500).json({ message: error.message });
10  }
11 };
12
13 exports.findAll = async (req, res) => {
14   try {
15     const users = await User.find();
16     res.status(200).json(users);
17   } catch (error) {
18     res.status(500).json({ message: error.message });
19   }
20 };
21
22 exports.findOne = async (req, res) => {
23   try {
24     const user = await User.findById(req.params.id);
25     if (!user) return res.status(404).json({ message: 'User not found' });
26     res.status(200).json(user);
27   } catch (error) {
28     res.status(500).json({ message: error.message });
29   }
30 };
31
32 exports.update = async (req, res) => {
33   try {
34     const user = await User.findByIdAndUpdate(req.params.id, req.body, { new: true });
35     if (!user) return res.status(404).json({ message: 'User not found' });
36     res.status(200).json({ message: 'User updated!', user });
37   } catch (error) {
38     res.status(500).json({ message: error.message });
39   }
40 }
```

🔍 (1) Maharashtra Hasra... x User Management x +

localhost:5000

Gmail dev course english art CS learn now frontend Community Exchan... D4E - Django for E... Responsive Web De... shreyagopal/Phishin... SQLZoo All Bookmarks

User Management

Add User

Reset

Find User

Find

Users List

Hide Users

Name	Email	Phone	Actions
Rajvardhan Deshmukh	mathematicsworld4256@gmail.com	9309815333	<div>Delete</div>
Atharv Desai	desaiatharv907@gmail.com	9552848132	<div>Delete</div>
Shivaji Gadekar	rajvardhanhd6403@gmail.com	8767729499	<div>Delete</div>