

```
1  const mongoose = require("mongoose");
2  require("dotenv").config();
3
4  const connectDB = async () => {
5      try {
6          await mongoose.connect(process.env.MONGO_URI, {
7              useNewUrlParser: true,
8              useUnifiedTopology: true,
9          });
10         console.log("MongoDB Connected...");
11     } catch (error) {
12         console.error("MongoDB Connection Failed:", error);
13         process.exit(1);
14     }
15 };
16
17 module.exports = connectDB;
18
```

```
1  const mongoose = require("mongoose");
2
3  const groupMessageSchema = new mongoose.Schema({
4      from_user: {
5          type: String,
6          required: true
7      },
8      room: {
9          type: String,
10         required: true
11     },
12     message: {
13         type: String,
14         required: true
15     },
16     date_sent: {
17         type: Date,
18         default: Date.now
19     }
20 });
21
22 const GroupMessage = mongoose.model('GroupMessage', groupMessageSchema);
23 module.exports = GroupMessage;
24
```

The screenshot shows a code editor with a file tree on the left and the content of `PrivateMessage.js` on the right.

```
1  const mongoose = require('mongoose');
2
3  const privateMessageSchema = new mongoose.Schema({
4      from_user: {
5          type: String,
6          required: true
7      },
8      to_user: {
9          type: String,
10         required: true
11     },
12     message: {
13         type: String,
14         required: true
15     },
16     date_sent: {
17         type: Date,
18         default: Date.now
19     }
20 });
21
22 const PrivateMessage = mongoose.model('PrivateMessage', privateMessageSchema);
23 module.exports = PrivateMessage;
24
```

The screenshot shows a code editor with a file tree on the left and the content of `User.js` on the right.

```
1  const mongoose = require('mongoose');
2
3  const userSchema = new mongoose.Schema({
4      username: {
5          type: String,
6          required: true,
7          unique: true, // Ensure the username is unique
8      },
9      firstname: String,
10     lastname: String,
11     password: {
12         type: String,
13         required: true
14     },
15     createon: {
16         type: Date,
17         default: Date.now
18     }
19 });
20
21 const User = mongoose.model('User', userSchema);
22 module.exports = User;
23
```

File tree:

```

    < models
      < User.js
    > node_modules
    < public
      < css
        # style.css
      < js
        JS chat.js
      < routes
        JS authRoutes.js
        JS chatRoutes.js
      < views
        <> chat.html
        <> login.html
        <> signup.html
      .env
    {} package-lock.js...
    {} package.json
    JS server.js
  > config
  ...
```

> OUTLINE

> TIMELINE

> CS-SCRIPT - ACTIVE

```

2   body {
3     font-family: Arial, sans-serif;
4     background-color: #f5f5f5;
5     color: #333;
6   }
7
8   #chat-window {
9     background-color: #fff;
10    padding: 10px;
11    border-radius: 5px;
12    max-width: 800px;
13    margin: 0 auto;
14  }
15
16  #messages {
17    max-height: 400px;
18    overflow-y: scroll;
19  }
20
21  .message {
22    margin: 5px 0;
23  }
24
25  #typing-indicator {
26    font-style: italic;
27    color: gray;
28 }
```

File tree:

```

> OPEN EDITORS 1 unsaved
< LABTE... 📁 🚫 C ⌂
< 101431885_lab_t...
  < models
    JS User.js
  > node_modules
  < public
    < css
      # style.css
    < js
      JS chat.js
  < routes
    JS authRoutes.js
    JS chatRoutes.js
  < views
    <> chat.html
    <> login.html
    <> signup.html
  .env
  {} package-lock.js...
  {} packagejson
  JS server.js
> config
...
```

> OUTLINE

> TIMELINE

> CS-SCRIPT - ACTIVE

```

101431885_lab_test1_chat_app > public > js > JS chat.js > ...
1  // public/js/chat.js
2  const socket = io(); // Connect to WebSocket
3
4  const token = localStorage.getItem("token");
5  if (!token) {
6    window.location.href = "/login";
7  }
8
9  const sendBtn = document.getElementById("sendMessageBtn");
10 const roomSelect = document.getElementById("roomSelect");
11 const messagesDiv = document.getElementById("messages");
12 const messageInput = document.getElementById("messageInput");
13 const typingIndicator = document.getElementById("typingIndicator");
14
15 let typingTimer;
16
17 // Send message
18 sendBtn.addEventListener("click", async () => {
19   const message = messageInput.value.trim();
20   const room = roomSelect.value;
21   const from_user = localStorage.getItem("username");
22
23   if (!message) return;
24
25   await fetch("/chat/sendMessage", {
26     method: "POST",
27     headers: { "Content-Type": "application/json" },
28     body: JSON.stringify({ from_user, room, message }),
29   });

```

```

101431885_lab_t... ●
  ↴ models
    JS User.js
  > node_modules ●
  ↴ public
    ↴ css
      # style.css
    ↴ js
      ●
        JS chat.js U
  ↴ routes
    JS authRoutes.js M
    JS chatRoutes.js M
  ↴ views
    <> chat.html M
    <> login.html M
    <> signup.html M
  ↶ .env
  ↶ package-lock.json M
  ↶ package.json M
  JS server.js M
  > config
  ... ...
  > OUTLINE
  > TIMELINE
  > CS-SCRIPT - ACTIVE

  50   messageInput.value = "";
  51   socket.emit("stopTyping", { room });
  52 });
  53
  54 // Listen for new messages in real-time (room-filtered)
  55 socket.on("newMessage", (data) => {
  56   const currentRoom = roomSelect.value;
  57   if (data.room !== currentRoom) return;
  58
  59   messagesDiv.innerHTML += `<p><strong>${data.from_user}</strong> ${data.message}</p>`;
  60   messagesDiv.scrollTop = messagesDiv.scrollHeight;
  61 });
  62
  63 // Load messages when the room changes
  64 async function loadMessages(room) {
  65   const response = await fetch(`/chat/messages/${room}`);
  66   const messages = await response.json();
  67
  68   messagesDiv.innerHTML = messages
  69     .map((msg) => `<p><strong>${msg.from_user}</strong> ${msg.message}</p>`)
  70     .join("");
  71
  72   messagesDiv.scrollTop = messagesDiv.scrollHeight;
  73
  74 // Join the room using Socket.io
  75   socket.emit("joinRoom", room);
  76 }

  77 // Change room
  78 roomSelect.addEventListener("change", () => {
  79   const room = roomSelect.value;
  80   loadMessages(room);
  81 });
  82 // Load messages when page loads
  83 window.onload = () => {
  84   const room = roomSelect.value;
  85   loadMessages(room);
  86 };
  87
  88 // Typing indicator (required)
  89 messageInput.addEventListener("input", () => {
  90   const room = roomSelect.value;
  91   const username = localStorage.getItem("username");
  92
  93   socket.emit("typing", { room, username });
  94
  95   clearTimeout(typingTimer);
  96   typingTimer = setTimeout(() => {
  97     socket.emit("stopTyping", { room });
  98   }, 700);
  99 });
  100
  101 socket.on("typing", ({ username }) => {
  102   if (typingIndicator) typingIndicator.textContent = `${username} is typing...`;
  103 });
  104
  105 socket.on("stopTyping", () => {
  106   if (typingIndicator) typingIndicator.textContent = "";
  107 });

  108
  109
  110
  111
  112
  113
  114
  115
  116
  117
  118
  119
  120
  121
  122
  123
  124
  125
  126
  127
  128
  129
  130
  131
  132
  133
  134
  135
  136
  137
  138
  139
  140
  141
  142
  143
  144
  145
  146
  147
  148
  149
  150
  151
  152
  153
  154
  155
  156
  157
  158
  159
  160
  161
  162
  163
  164
  165
  166
  167
  168
  169
  170
  171
  172
  173
  174
  175
  176
  177
  178
  179
  180
  181
  182
  183
  184
  185
  186
  187
  188
  189
  190
  191
  192
  193
  194
  195
  196
  197
  198
  199
  200
  201
  202
  203
  204
  205
  206
  207
  208
  209
  210
  211
  212
  213
  214
  215
  216
  217
  218
  219
  220
  221
  222
  223
  224
  225
  226
  227
  228
  229
  230
  231
  232
  233
  234
  235
  236
  237
  238
  239
  240
  241
  242
  243
  244
  245
  246
  247
  248
  249
  250
  251
  252
  253
  254
  255
  256
  257
  258
  259
  260
  261
  262
  263
  264
  265
  266
  267
  268
  269
  270
  271
  272
  273
  274
  275
  276
  277
  278
  279
  280
  281
  282
  283
  284
  285
  286
  287
  288
  289
  290
  291
  292
  293
  294
  295
  296
  297
  298
  299
  300
  301
  302
  303
  304
  305
  306
  307
  308
  309
  310
  311
  312
  313
  314
  315
  316
  317
  318
  319
  320
  321
  322
  323
  324
  325
  326
  327
  328
  329
  330
  331
  332
  333
  334
  335
  336
  337
  338
  339
  340
  341
  342
  343
  344
  345
  346
  347
  348
  349
  350
  351
  352
  353
  354
  355
  356
  357
  358
  359
  360
  361
  362
  363
  364
  365
  366
  367
  368
  369
  370
  371
  372
  373
  374
  375
  376
  377
  378
  379
  380
  381
  382
  383
  384
  385
  386
  387
  388
  389
  390
  391
  392
  393
  394
  395
  396
  397
  398
  399
  400
  401
  402
  403
  404
  405
  406
  407
  408
  409
  410
  411
  412
  413
  414
  415
  416
  417
  418
  419
  420
  421
  422
  423
  424
  425
  426
  427
  428
  429
  430
  431
  432
  433
  434
  435
  436
  437
  438
  439
  440
  441
  442
  443
  444
  445
  446
  447
  448
  449
  450
  451
  452
  453
  454
  455
  456
  457
  458
  459
  460
  461
  462
  463
  464
  465
  466
  467
  468
  469
  470
  471
  472
  473
  474
  475
  476
  477
  478
  479
  480
  481
  482
  483
  484
  485
  486
  487
  488
  489
  490
  491
  492
  493
  494
  495
  496
  497
  498
  499
  500
  501
  502
  503
  504
  505
  506
  507
  508
  509
  510
  511
  512
  513
  514
  515
  516
  517
  518
  519
  520
  521
  522
  523
  524
  525
  526
  527
  528
  529
  530
  531
  532
  533
  534
  535
  536
  537
  538
  539
  540
  541
  542
  543
  544
  545
  546
  547
  548
  549
  550
  551
  552
  553
  554
  555
  556
  557
  558
  559
  560
  561
  562
  563
  564
  565
  566
  567
  568
  569
  570
  571
  572
  573
  574
  575
  576
  577
  578
  579
  580
  581
  582
  583
  584
  585
  586
  587
  588
  589
  590
  591
  592
  593
  594
  595
  596
  597
  598
  599
  600
  601
  602
  603
  604
  605
  606
  607
  608
  609
  610
  611
  612
  613
  614
  615
  616
  617
  618
  619
  620
  621
  622
  623
  624
  625
  626
  627
  628
  629
  630
  631
  632
  633
  634
  635
  636
  637
  638
  639
  640
  641
  642
  643
  644
  645
  646
  647
  648
  649
  650
  651
  652
  653
  654
  655
  656
  657
  658
  659
  660
  661
  662
  663
  664
  665
  666
  667
  668
  669
  670
  671
  672
  673
  674
  675
  676
  677
  678
  679
  680
  681
  682
  683
  684
  685
  686
  687
  688
  689
  690
  691
  692
  693
  694
  695
  696
  697
  698
  699
  700
  701
  702
  703
  704
  705
  706
  707
  708
  709
  710
  711
  712
  713
  714
  715
  716
  717
  718
  719
  720
  721
  722
  723
  724
  725
  726
  727
  728
  729
  730
  731
  732
  733
  734
  735
  736
  737
  738
  739
  740
  741
  742
  743
  744
  745
  746
  747
  748
  749
  750
  751
  752
  753
  754
  755
  756
  757
  758
  759
  760
  761
  762
  763
  764
  765
  766
  767
  768
  769
  770

```

OPEN EDITORS 2 unsaved

LABTE... 🌐 C ⚙️

- ✓ 101431885_lab_t... ●
- models
 - JS User.js
- > node_modules
- ✓ public
 - css
 - # style.css
 - js
 - JS chat.js U
- ✓ routes
 - JS authRoutes.js M
 - JS chatRoutes.js M
- views
 - chat.html M
 - login.html M
 - signup.html M
- .env
- { package-lock.json M
- { package.json M
- JS server.js M
- config

OUTLINE

TIMELINE

CS-SCRIPT - ACTIVE

```
You, 1 hour ago | 1 author (You)
1 // routes/authRoutes.js
2 const express = require("express");
3 const bcrypt = require("bcryptjs");
4 const jwt = require("jsonwebtoken");
5 const User = require("../models/User");
6
7 const router = express.Router();
8
9 /*
10 =====
11 Signup Route
12 POST /auth/signup
13 =====
14 */
15 router.post("/signup", async (req, res) => {
16   const { username, firstname, lastname, password } = req.body;
17
18   try {
19     // Check if user exists
20     const existingUser = await User.findOne({ username });
21     if (existingUser) {
22       return res.status(400).json({ error: "Username already taken" });
23     }
24
25     // Hash password
26     const hashedPassword = await bcrypt.hash(password, 10);
27
28     // Create user
29     const user = new User({
30       username,
31       firstname,
32       lastname,
33       password: hashedPassword
34     });
35     await user.save();
36
37     const token = jwt.sign({ user }, process.env.JWT_SECRET);
38     res.status(200).json({ token });
39   } catch (err) {
40     console.error(err);
41     res.status(500).json({ error: "Server error" });
42   }
43 })
44
45 module.exports = router;
```

OPEN EDITORS 2 unsaved

LABTE... 🌐 C ⚙️

- ✓ 101431885_lab_t... ●
- models
 - JS User.js
- > node_modules
- ✓ public
 - css
 - # style.css
 - js
 - JS chat.js U
- ✓ routes
 - JS authRoutes.js M
 - JS chatRoutes.js M
- views
 - chat.html M
 - login.html M
 - signup.html M
- .env
- { package-lock.json M
- { package.json M
- JS server.js M
- config

OUTLINE

TIMELINE

CS-SCRIPT - ACTIVE

```
You, 1 hour ago | 1 author (You)
1 // routes/chatRoutes.js
2 const express = require("express");
3 const GroupMessage = require("../models/GroupMessage");
4
5 const router = express.Router();
6
7 module.exports = (io) => {
8   // Send message (stores in MongoDB)
9   router.post("/sendMessage", async (req, res) => {
10     const { from_user, room, message } = req.body;
11
12     // Basic validation
13     if (!from_user || !room || !message) {
14       return res.status(400).json({ error: "Missing required fields" });
15     }
16
17     try {
18       const newMsg = new GroupMessage({ from_user, room, message });
19       await newMsg.save();
20
21       // Realtime emit to the room
22       io.to(room).emit("newMessage", {
23         from_user,
24         message,
25         room,
26         date_sent: newMsg.date_sent,
27       });
28
29     } catch (err) {
30       console.error(err);
31       res.status(500).json({ error: "Server error" });
32     }
33   });
34 }
35
36 module.exports = router;
```

OPEN EDITORS 2 unsaved

LABTEST... 101431885_lab_t...

- models
- User.js
- node_modules
- public
- css # style.css
- js chatjs
- routes authRoutes.js M
- chatRoutes.js M
- views
- chat.html M
- login.html M
- signup.html M
- .env
- package-lock.json M
- package.json M
- server.js M
- config

OUTLINE

TIMELINE

CS-SCRIPT - ACTIVE

```

101431885_lab_test1_chat_app > views > chat.html > ...
You, 1 hour ago | 1 author (You)

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>Chat - Chat App</title>
7   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
8   <link rel="stylesheet" href="/css/style.css">
9 </head>
10 <body>
11   <div class="container">
12     <h2 class="mt-5">Welcome to the Chat</h2>
13
14     <button id="logoutBtn" class="btn btn-warning">Logout</button>
15     <button id="leaveRoomBtn" class="btn btn-danger">Leave Room</button>
16
17     <div class="mt-4">
18       <label for="roomSelect">Select a Room:</label>
19       <select id="roomSelect" class="form-select">
20         <option value="devops">DevOps</option>
21         <option value="cloud">Cloud Computing</option>
22         <option value="covid19">COVID-19</option>
23         <option value="sports">Sports</option>
24         <option value="nodejs">Node.js</option>
25       </select>
26     </div>
27
28     <div id="messages" class="mt-3"></div>

```

OPEN EDITORS 2 unsaved

LABTEST1 101431885_lab_t...

- models
- User.js
- node_modules
- public
- css # style.css
- js chatjs
- routes authRoutes.js M
- chatRoutes.js M
- views
- chat.html M
- login.html M
- signup.html M
- .env
- package-lock.json M
- package.json M
- server.js M
- config

OUTLINE

TIMELINE

```

101431885_lab_test1_chat_app > views > login.html > ...
You, 1 hour ago | 1 author (You)

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Login</title>
6   <link rel="stylesheet" href="/css/style.css">
7 </head>
8 <body>
9
10 <div id="chat-window">
11   <h2>Login</h2>
12
13   <form id="loginForm">
14     <input type="text" id="username" placeholder="Username" required />
15     <br><br>
16     <input type="password" id="password" placeholder="Password" required />
17     <br><br>
18     <button type="submit">Login</button>
19   </form>
20
21   <p>No account? <a href="/signup">Signup here</a></p>
22 </div>
23
24 <script src="/js/login.js"></script>
25
26 </body>
27 </html>

```

> OPEN EDITORS 2 unsaved

✓ LABTEST1 101431885_lab_test1_chat_app

- ✓ models
 - JS User.js
- > node_modules
- ✓ public
 - ✓ css
 - # style.css
 - ✓ js
 - JS chat.js U
- ✓ routes
 - JS authRoutes.js M
 - JS chatRoutes.js M
- ✓ views
 - chat.html M
 - login.html M
 - signup.html M

.env

{ package-lock.json... M

{ package.json M

JS server.js M

> config

OUTLINE

TIMELINE

```
You, 1 hour ago | 1 author (You)
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>Signup</title>
7   <link rel="stylesheet" href="/css/style.css" />
8 </head>
9 <body>
10 <div id="chat-window">
11   <h2>Signup</h2>
12
13   <form id="signupForm">
14     <input type="text" id="username" placeholder="Username" required />
15     <br><br>
16
17     <input type="text" id="firstname" placeholder="First name" required />
18     <br><br>
19
20     <input type="text" id="lastname" placeholder="Last name" required />
21     <br><br>
22
23     <input type="password" id="password" placeholder="Password" required />
24     <br><br>
25
26     <button type="submit">Create Account</button>
27 </form>
28
```

/

> OPEN EDITORS 2 unsaved

✓ LABTEST1 101431885_lab_test1_chat_app

- ✓ models
 - JS User.js
- > node_modules
- ✓ public
 - ✓ css
 - # style.css
 - ✓ js
 - JS chat.js U
- ✓ routes
 - JS authRoutes.js M
 - JS chatRoutes.js M
- ✓ views
 - chat.html M
 - login.html M
 - signup.html M

.env

{ package-lock.json... M

{ package.json M

JS server.js M

> config

OUTLINE

TIMELINE

```
You, 54 minutes ago | 1 author (You)
1 {
2   "name": "101431885_lab_test1_chat_app",
3   "version": "1.0.0",
4   "lockfileVersion": 3,
5   "requires": true,
6   "packages": {
7     "": {
8       "name": "101431885_lab_test1_chat_app",
9       "version": "1.0.0",
10      "dependencies": {
11        "bcryptjs": "^2.4.3",
12        "core": "^2.8.5",
13        "dotenv": "^16.4.7",
14        "express": "^4.21.2",
15        "jsonwebtoken": "^9.0.2",
16        "mongoose": "^6.21.0",
17        "mongoose": "^8.10.0",
18        "socket.io": "^4.8.1"
19      }
20    }
21  }
22 }
23
24 "node_modules/@mongodb-js/saslprep": {
25   "version": "1.4.6",
26   "resolved": "https://registry.npmjs.org/@mongodb-js/saslprep/-/saslprep-1.4.6.tgz",
27   "integrity": "sha512-y+x3H1xBZd38n10NZF/rEB1vDOOMQ6LKUTHqr8R9Vkj+mmQOYtJFxI1kkK8fZrtOiL6VixbOBWMbZGBdal3Z1g="
28   "license": "MIT"
29 }
```

/

The screenshot shows the VS Code interface with the following details:

- OPEN EDITORS:** 2 unsaved
- LABTEC...** (selected)
- 101431885_lab_test1_chat_app**
- models**: User.js
- public**: style.css
- js**: chat.js (U), authRoutes.js (M), chatRoutes.js (M), chat.html (M), login.html (M), signup.html (M), .env, package-lock.json... (M), package.json (M), server.js (M)
- routes**: authRoutes.js (M), chatRoutes.js (M)
- views**: chat.html (M), login.html (M), signup.html (M)
- config**
- OUTLINE**
- TIMELINE**

package.json content (Line numbers 1-23):

```
1  {
2    "name": "101431885_lab_test1_chat_app",
3    "version": "1.0.0",
4    "main": "server.js",
5    "scripts": {
6      "start": "node server.js",
7      "dev": "nodemon server.js"
8    },
9    "dependencies": {
10      "bcryptjs": "^2.4.3",
11      "cors": "^2.8.5",
12      "dotenv": "^16.4.7",
13      "express": "^4.21.2",
14      "jsonwebtoken": "^9.0.2",
15      "mongoose": "^6.21.0",
16      "mongoose": "^8.10.0",
17      "socket.io": "^4.8.1"
18    },
19    "devDependencies": {
20      "nodemon": "^3.0.0"
21    }
22  }
```

You, 54 minutes ago | 1 author (You)

You, 54 minutes ago • Uncommitted changes

1

The screenshot shows a code editor interface with the following details:

- Open Editors:** 2 unsaved
- LABELED:** 101431885_lab_t...
- models:** Userjs
- node_modules:** (empty)
- public:** style.css
- css:** style.css
- js:** chat.js
- routes:** authRoutes.js, chatRoutes.js
- views:** chat.html, login.html, signup.html
- .env**
- package-lock.json**
- package.json**
- server.js** (selected)
- config**
- OUTLINE**
- TIMELINE**
- CS-SCRIPT - ACTIVE**

File Content (server.js):

```
101431885_lab_test1_chat_app > JS server.js > ...
You, 1 hour ago | 1 author (You)
1 const path = require("path");
2 const express = require("express");
3 const dotenv = require("dotenv");
4 const http = require("http");
5 const cors = require("cors");
6 const { Server } = require("socket.io");
7
8 const connectDB = require("./config/db");
9 const authRoutes = require("./routes/authRoutes");
10 const chatRoutes = require("./routes/chatRoutes");
11
12 dotenv.config();
13
14 const app = express();
15 const server = http.createServer(app);
16 const io = new Server(server, { cors: { origin: "*" } });
17
18 // DB
19 connectDB();
20
21 // Middleware
22 app.use(cors());
23 app.use(express.json());
24 app.use(express.urlencoded({ extended: true }));
25
26 // Serve static files (JS/CSS)
27 app.use(express.static(path.join(__dirname, "public")));
28
29 // Socket.io
```