

▼ Step 5: Model

models/user.model.js

```
p
const db = require("../config/db.config");
const User = {

Ask anything
},
getById: (id, callback) => {
    db.query("SELECT * FROM users MMERE id = ?", [id], callback);
},

create: (data, callback) => {
    db.query("INSERT INTO users SET ?", data, callback);
},

update: (id, data, callback) => {
    db.query("UPDATE users SET ? MMERE id = ?", [data, id], callback);
},

delete: (id, callback) => {
    db.query("DELETE FROM users MMERE id = ?", [id], callback);
},

module.exports = User;
```

✓ Step 6: Controller

controllers/user.controller.js

```
const User = require("../models/user.model");

Ask anything

if (err) return res.status(500).sens(err);
    res.sens(data);
));
};

exports.getUserById = (req, res) => {
    User.getByIc(req.params.id, (err, data) => {
        if (err) return res.status(500).sens(err);
        res.sens(data[0]);
));
};

exports.createUser = (req, res) => {
    User.create(req.body, (err, result) => {
        if (err) return res.status(500).sens(err);
        res.status(201).sens({ id: result.insertId, ...req.body });
));
};

exports.updateUser = (req, res) => {
    User.updateUser = (req, res) => {
    User.deleteUser = (req, res) => {
        User.deleteUser = (req, res) => {
        User.delete(req.params.id, ...req.body });
));
};

Ask anything
```

```
routes/user.routes.js

js

const express = require("express");

const router = express.Router();

const userController = require("./controllers/user.controller");

router.get("/", userController.getAllUsers);

router.get("/id", userController.getUserById);

router.post("/", userController.createUser);

router.put("/id", userController.updateUser);

router.delete("/:Id", userController.deleteUser);

module.exports = router;
```

✓ Step 8: Entry Point

```
server.js

p

const express = require("express");

Ask anything

require("dotenv").config();

const app = express();

app.use(cons());

app.use(cons());

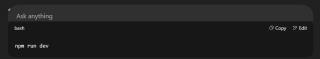
app.use('/api/users', userRoutes);
```

const PORT = process.env.PORT || 5000;
app.listen(PORT, () => console.log(' # Server is running on port \$(PORT)'));

▼ Step 9: package.json script

Update scripts:

✓ Step 10: Run Your Project



Sample API Endpoints

Method	Endpoint	Description
GET	/api/users	Get all users
GET	/api/users/:id	Get user by ID
POST	/api/users	Create new user
PUT	/api/users/:id	Update user
DELETE	/api/users/:id	Delete user

Do you want me to generate this as a downloadable ZIP file for you or need Postman collection for testing APIs?

0 0 0 0 0 0 1 C ×