
🔧 Steps Covered

1. **Define Mongoose Schemas** (User.js & UserDetail.js)
 2. **Set Up Express Routes & Controllers** (routes/userRoutes.js, routes/userDetailRoutes.js, controllers/userController.js, controllers/userDetailController.js)
 3. **Test API Requests**
-

1 📦 Install Dependencies

Run:

```
npm install express mongoose
```

2 📦 Define Mongoose Models

User Model (models/User.js)

```
const mongoose = require("mongoose");

const userSchema = new mongoose.Schema({
  name: String,
  email: String,
  details: { type: mongoose.Schema.Types.ObjectId, ref: "UserDetail" }, //
  // One-to-One Reference
});

const User = mongoose.model("User", userSchema);
module.exports = User;
```

UserDetail Model (models/UserDetail.js)


```
const mongoose = require("mongoose");

const userDetailSchema = new mongoose.Schema({
  age: Number,
  address: String,
  user: { type: mongoose.Schema.Types.ObjectId, ref: "User", unique: true },
  // Ensures One-to-One
});


const UserDetail = mongoose.model("UserDetail", userDetailSchema);
module.exports = UserDetail;
```

3 Create User Controller (controllers/userController.js)

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

// 1  Create User
exports.createUser = async (req, res) => {
  const { name, email } = req.body;


  try {
    const user = new User({ name, email });
    await user.save();
    res.json(user);
  } catch (error) {
    res.status(500).json({ error: error.message });
  }
};

// 2  Get User with Details (Populate)
exports.getUser = async (req, res) => {
  try {
    const user = await User.findById(req.params.id).populate("details");
    if (!user) return res.status(404).json({ message: "User not found" });

    res.json(user);
  } catch (error) {
    res.status(500).json({ error: error.message });
  }
};
```

4 Create UserDetail Controller (controllers/userDetailController.js)

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

// 1  Create UserDetail (after User is created)
exports.createUserDetail = async (req, res) => {
  const { userId, age, address } = req.body;

  try {
    const user = await User.findById(userId);
    if (!user) return res.status(404).json({ message: "User not found" });

    // Ensure user doesn't already have details
    const existingDetail = await UserDetail.findOne({ user: userId });
    if (existingDetail) {
      return res.status(400).json({ message: "User details already exist" });
    }

    const userDetail = new UserDetail({ age, address, user: userId });
    await userDetail.save();

    // Update User with details ID
    user.details = userDetail._id;
  }
};
```

```

    await user.save();

    res.json(userDetail);
  } catch (error) {
    res.status(500).json({ error: error.message });
  }
};

// 2. Get UserDetail by ID
exports.getUserDetail = async (req, res) => {
  try {
    const userDetail = await UserDetail.findOne({ user: req.params.userId });
    if (!userDetail) return res.status(404).json({ message: "Details not found" });

    res.json(userDetail);
  } catch (error) {
    res.status(500).json({ error: error.message });
  }
};

```

5. Create Routes

User Routes (`routes/userRoutes.js`)

```

const express = require("express");
const router = express.Router();
const userController = require("../controllers/userController");

router.post("/", userController.createUser);
router.get("/:id", userController.getUser);

module.exports = router;

```

UserDetail Routes (`routes/userDetailRoutes.js`)

```

const express = require("express");
const router = express.Router();
const userDetailController = require("../controllers/userDetailController");

router.post("/", userDetailController.createUserDetail);
router.get("/:userId", userDetailController.getUserDetail);

module.exports = router;

```

6. Set Up Express App (`index.js`)

```

const express = require("express");
const mongoose = require("mongoose");
const userRoutes = require("./routes/userRoutes");
const userDetailRoutes = require("./routes/userDetailRoutes");

```

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

// Connect to MongoDB
mongoose
  .connect("mongodb://localhost:27017/one-to-one-demo")
  .then(() => console.log("MongoDB Connected"))
  .catch((err) => console.error(err));

// Use Routes
app.use("/users", userRoutes);
app.use("/user-details", userDetailRoutes);

// Start Server
app.listen(3000, () => console.log("Server running on port 3000"));
```
