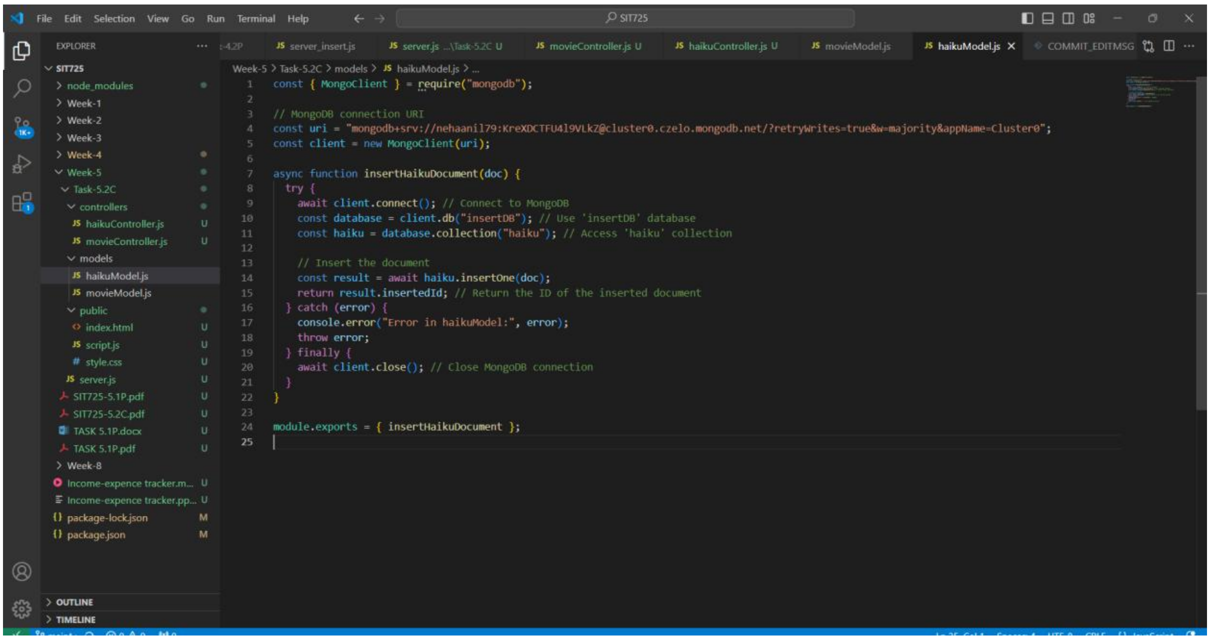
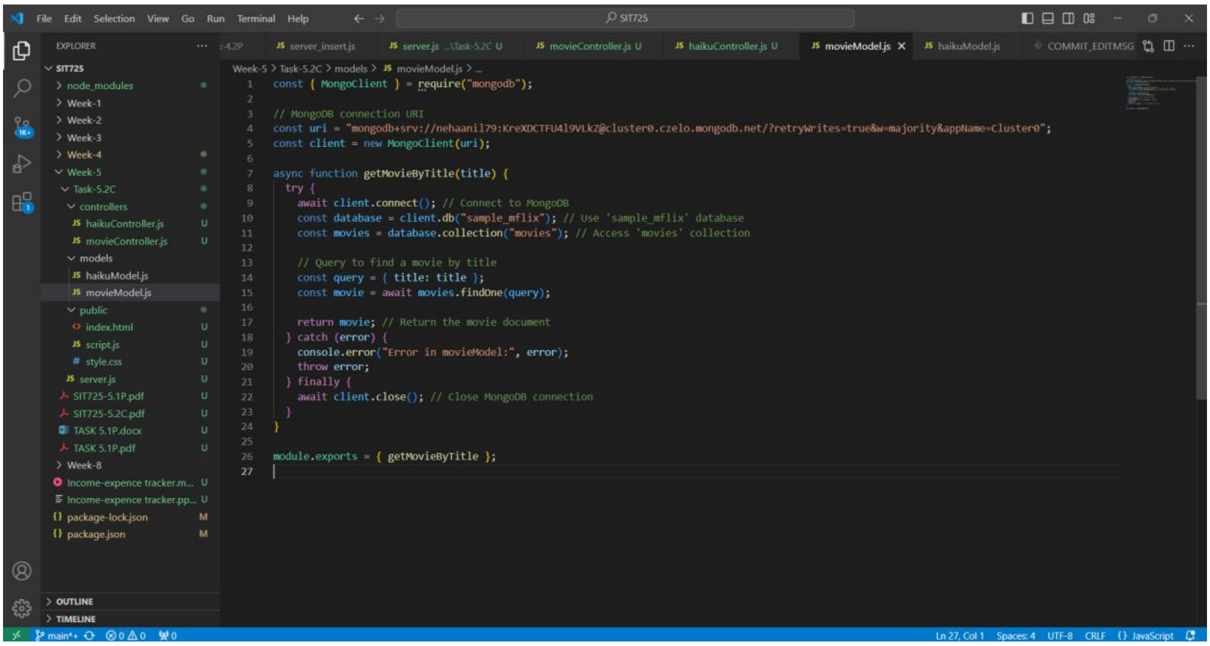
TASK 5.2C

MVC model

Code in **Model** folder:

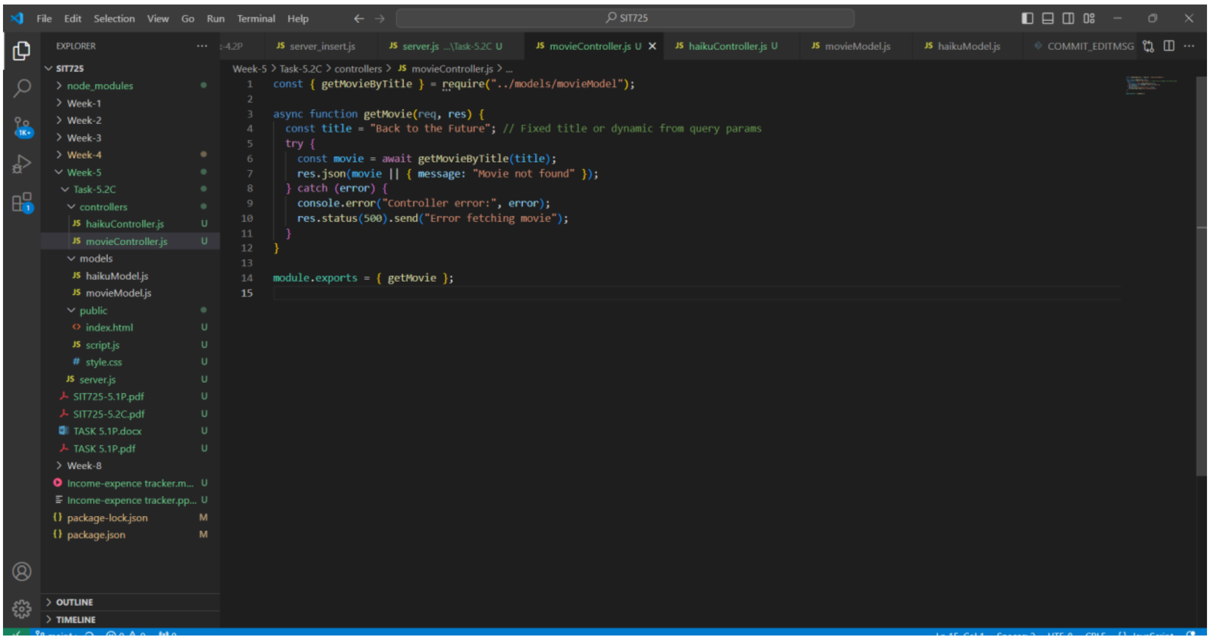

File Edit Selection View Go Run Terminal Help
SIT725
EXPLORER
SIT725
Week-5 > Task-5.2C > models > haikuModel.js > ...
haikuModel.js
OUTLINE
TIMELINE

```javascript
const { MongoClient } = require("mongodb");

// MongoDB connection URI
const uri = "mongodb+srv://nehaanil79:KreXDCTFU4l9VLkZ@cluster0.czelo.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0";
const client = new MongoClient(uri);

async function insertHaikuDocument(doc) {
  try {
    await client.connect(); // Connect to MongoDB
    const database = client.db("insertDB"); // Use 'insertDB' database
    const haiku = database.collection("haiku"); // Access 'haiku' collection

    // Insert the document
    const result = await haiku.insertOne(doc);
    return result.insertedId; // Return the ID of the inserted document
  } catch (error) {
    console.error("Error in haikuModel:", error);
    throw error;
  } finally {
    await client.close(); // Close MongoDB connection
  }
}

module.exports = { insertHaikuDocument };
```

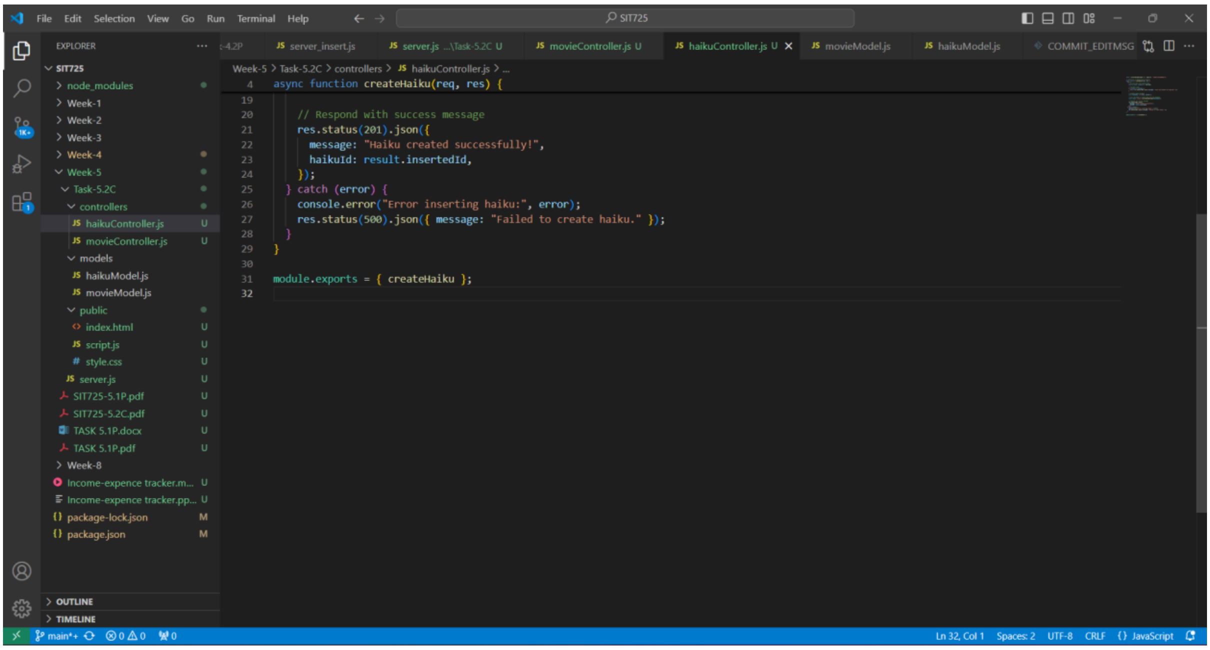

File Edit Selection View Go Run Terminal Help
SIT725
EXPLORER
SIT725
Week-5 > Task-5.2C > models > movieModel.js > ...
movieModel.js
OUTLINE
TIMELINE
Ln 27, Col 1 Spaces: 4 UTF-8 CRLF {} JavaScript

```javascript
const { MongoClient } = require("mongodb");

// MongoDB connection URI
const uri = "mongodb+srv://nehaanil79:KreXDCTFU4l9VLkZ@cluster0.czelo.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0";
const client = new MongoClient(uri);

async function getMovieByTitle(title) {
  try {
    await client.connect(); // Connect to MongoDB
    const database = client.db("sample_mflix"); // Use 'sample_mflix' database
    const movies = database.collection("movies"); // Access 'movies' collection

    // Query to find a movie by title
    const query = { title: title };
    const movie = await movies.findOne(query);

    return movie; // Return the movie document
  } catch (error) {
    console.error("Error in movieModel:", error);
    throw error;
  } finally {
    await client.close(); // Close MongoDB connection
  }
}

module.exports = { getMovieByTitle };
```

Code in **Controller** folder:

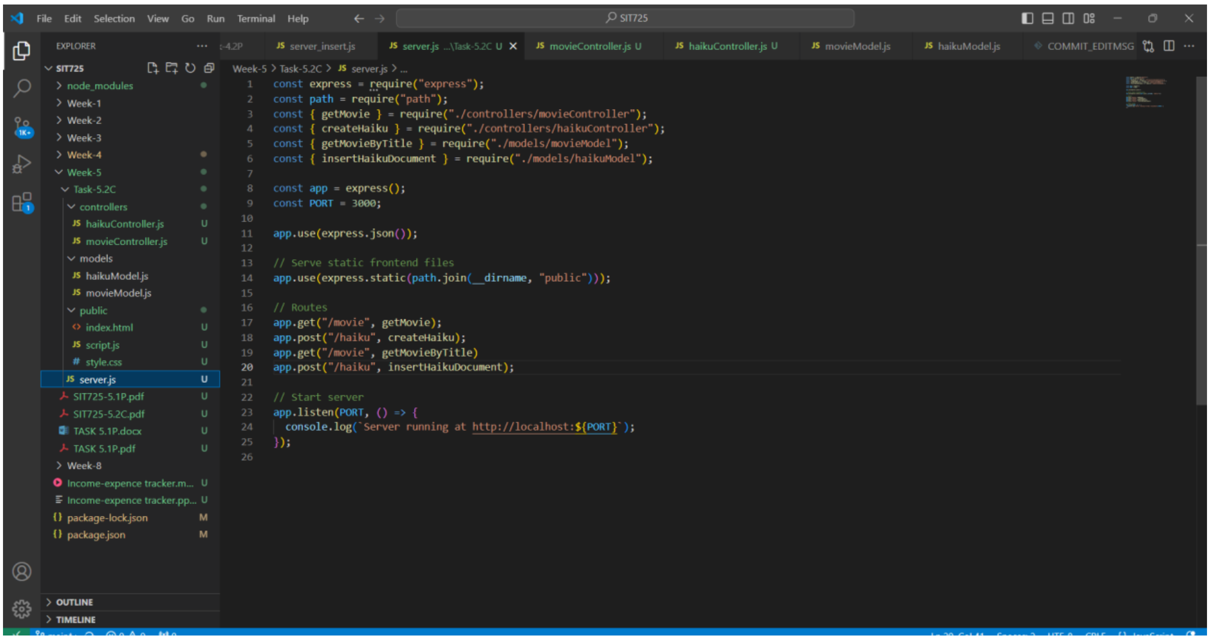


```javascript
const { getMovieByTitle } = require("../models/movieModel");

async function getMovie(req, res) {
  const title = "Back to the Future"; // Fixed title or dynamic from query params
  try {
    const movie = await getMovieByTitle(title);
    res.json(movie || { message: "Movie not found" });
  } catch (error) {
    console.error("Controller error:", error);
    res.status(500).send("Error fetching movie");
  }
}

module.exports = { getMovie };
```



```javascript
async function createHaiku(req, res) {

    // Respond with success message
    res.status(201).json({
      message: "Haiku created successfully!",
      haikuId: result.insertedId,
    });
  } catch (error) {
    console.error("Error inserting haiku:", error);
    res.status(500).json({ message: "Failed to create haiku." });
  }
}

module.exports = { createHaiku };
```

Server.js



```javascript
const express = require("express");
const path = require("path");
const { getMovie } = require("./controllers/movieController");
const { createHaiku } = require("./controllers/haikuController");
const { getMovieByTitle } = require("./models/movieModel");
const { insertHaikuDocument } = require("./models/haikuModel");

const app = express();
const PORT = 3000;

app.use(express.json());

// Serve static frontend files
app.use(express.static(path.join(__dirname, "public")));

// Routes
app.get("/movie", getMovie);
app.post("/haiku", createHaiku);
app.get("/movie", getMovieByTitle)
app.post("/haiku", insertHaikuDocument);

// Start server
app.listen(PORT, () => {
  console.log(`Server running at http://localhost:${PORT}`);
});
```

My repo link:

https://github.com/nehaanil12345/SIT725.git