

CSL308 SE S25 Mini Project - STEPS

Deliverable No. 1

Project Details

- Group Number: G9
- Group Name (Optional): Synergy Squad
- Mini-Project Title (Tentative): Vehicle Inventory Management System

Group Members

S. No. Enrollment No Name of the Student

- 1 BT22CSE025 NIKITA NAIR
- 2 BT22CSE103 LANJE PRASHIK SANJAY
- 3 BT22CSE038 SANKALP GHANSHAM MESHRAM
- 4 BT22CSE019 DUDEKULA SHAHINA
- 5 BT22CSE123 MEHTAB SINGH

GIT HUB Code

<https://github.com/nikinair2004/Vehicle-Management/upload>

STEP 1 : Data Base

1. Install MySQL Server on Laptop
2. Install MYSQL Work Bench on Laptop and start server
3. Create Schema : test / with password (S*****12)
4. Create tabel vehicle and insert data
5. Alter privilege password.

All the scripts are below

STEP 2: Node JS

1. Install Visual Studio
2. On VS install Node JS
3. MYSQL 1 had connectivity issue with so installed MYSQL2
4. In main users folder ie (Users/Nikitanair) create two folders
 - a. Backend (for node js)
 - b Client (for react)
5. one VS create DB connection , connection screen shot below
run following command if you get error
ALTER USER 'root'@'localhost'
IDENTIFIED BY 'S**12';***
5. On terminal do the following only if you get error
nikitanair@Nikitas-MacBook-Air ~ % **brew install node**
6. Go to backend folder and install node js and connect to mysql
 - a. ***npm init -y (only once)***
this will create necessary file

- b. *npm i express mysql nodemon (only once)*
this will create connection to mysql, if gives error the install mysql2 on VS and run
- c. Create file index.js and run it
node index.js (only once)
- d. start node js
npm start
to kill process use : pkill nodejs or pkillnodejs
- e. start axis and cross
npm I axios
npm start
npm I cors
npm start
- f. Modify **package.json** and add, screen shot below
"type": "module",
"start": "nodemon index.js"
- h. Now check backend data
<http://localhost:8800/>
<http://localhost:8800/vehicle>

T

STEP 2: React JS

Open new terminal on VS

Go to client folder and install react

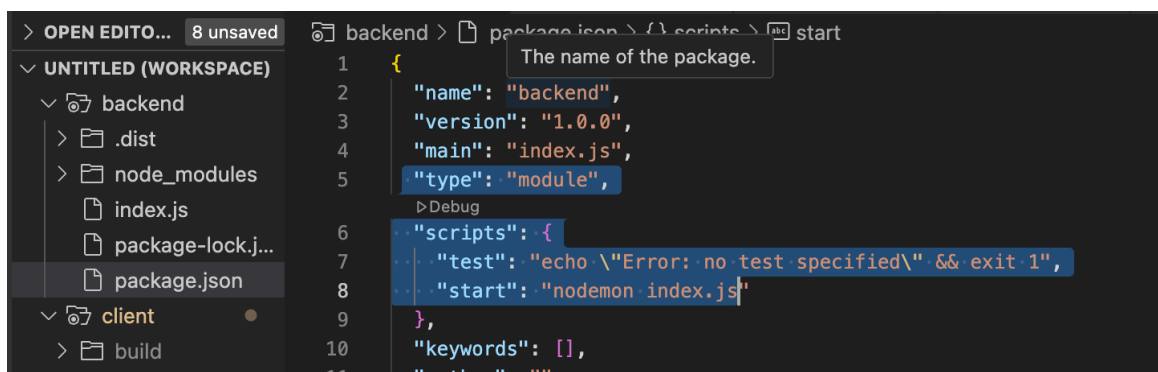
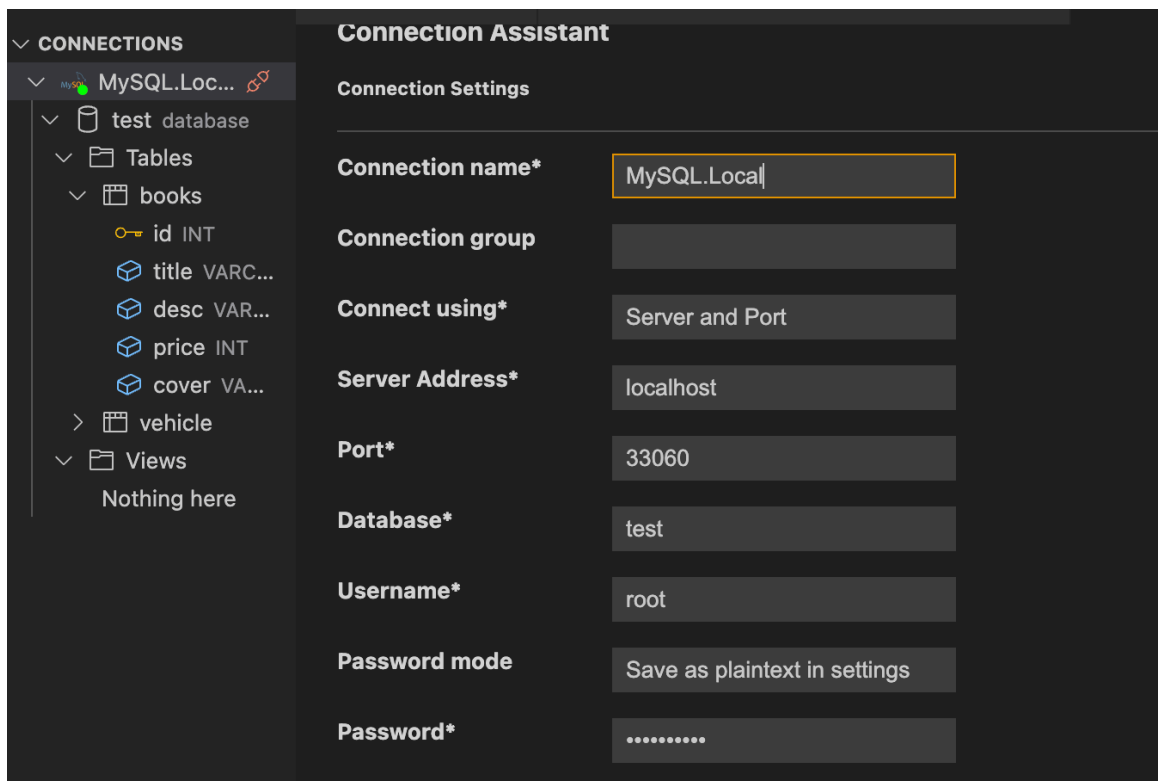
- a. install react, this will create necessary file
npx create-react-app . (Only once)
npm start
npm install react-router-dom —save
npm start
- b. create folders src and pages
in src folder create file app.js and style.css
in pages folder create 3file s , Vehicle (to see data), Add ,Update
- c. start axis and cross and react will open on <http://localhost3000/>

npm I axios
npm start
npm I cors
npm start

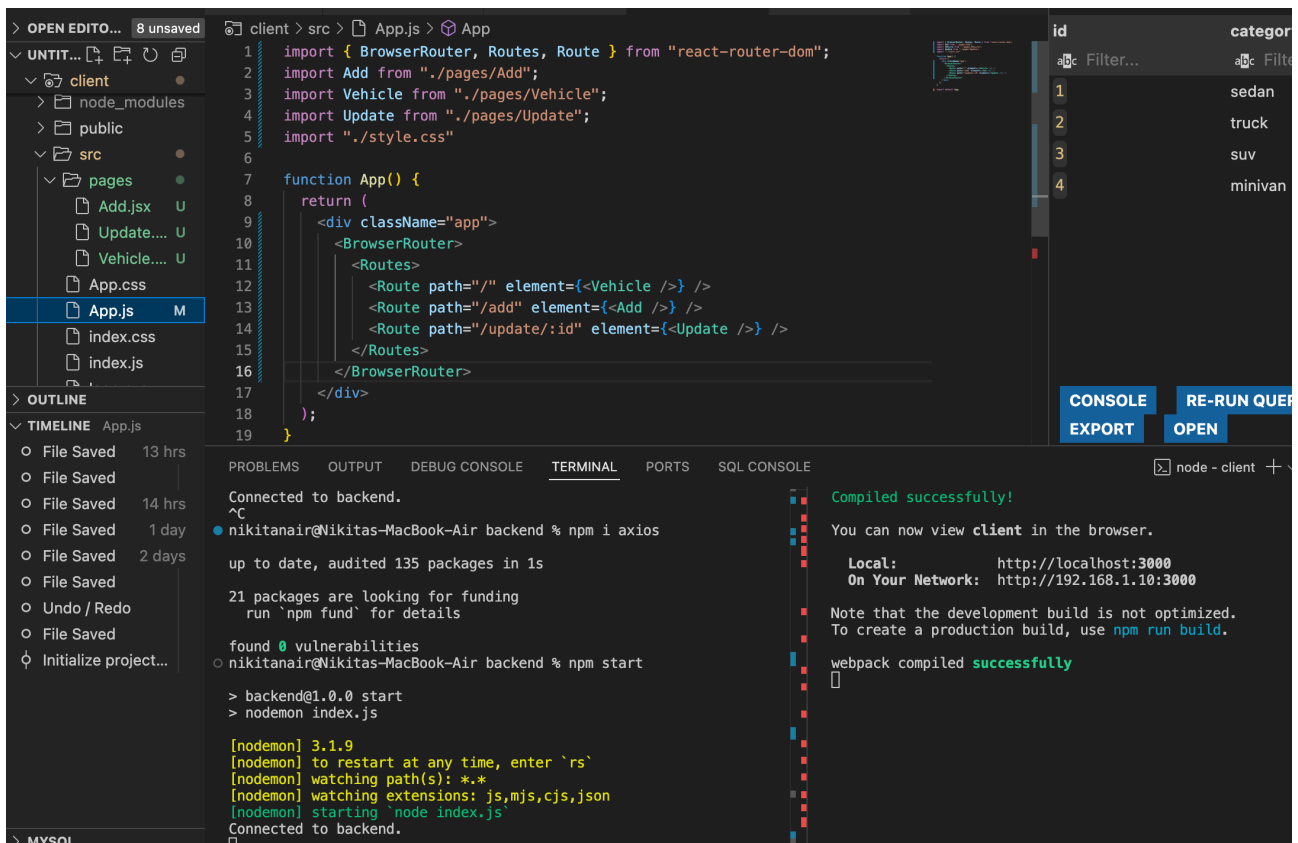
STEP 2: Testing DB, Node JS and React

Vehicle Inventory Management System will open.
Add new vehicle details, update and delete records.
See if data is reflected in DB table.

DB connection from VS Code and package.json file



Two terminals running on VS, one for backend and one for client



Vehicle Management System

Add New Vehicle

[See all vehicle](#)

Vehicle Management System

sedan	truck	suv	minivan
chevrolet	ford	toyota	chryster
yellow	red	blue	red
30	40	50	20
2019	2018	2019	2017
Delete	Delete	Delete	Delete
Update	Update	Update	Update
Add new vehicle			

SQL Scripts:

Create Schema test;

```
CREATE TABLE `books` (  
  `id` int NOT NULL AUTO_INCREMENT,  
  `title` varchar(45) DEFAULT NULL,  
  `desc` varchar(100) DEFAULT NULL,  
  `price` int DEFAULT NULL,  
  `cover` varchar(100) DEFAULT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO_INCREMENT=13 DEFAULT  
CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

```
INSERT INTO `vehicle`  
(`id`, `category`, `make`, `colour`, `mileage`, `year`)  
VALUES  
(1, 'sedan', 'chevrolet', 'yellow', '30', '2019'),  
(2, 'truck', 'ford', 'red', '40', '2018'),  
(3, 'suv', 'toyota', 'blue', '50', '2019'),  
(4, 'minivan', 'chryster', 'red', '20', '2017');
```

Links

<http://localhost:3000>

<http://localhost:8800/vehicle>

<http://localhost:3002/>

<http://localhost:3002/add>

<http://localhost:3002/graph>

<http://localhost:3002/graphpie>

<http://localhost:3002/Vehiclereport>

<http://localhost:3002/Vehiclesearch>