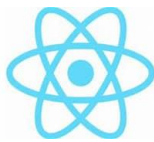


Documentation Technique : Tata Kevin

Sommaire :

- Différent Langage utilisés
- Outil Utilisé
- Comment les utilisés

1. Différent Langage utilisés



: React Js



: Node JS



: CSS

2. Outil Utilisé :



: Visual Studio Code (éditeur de code).



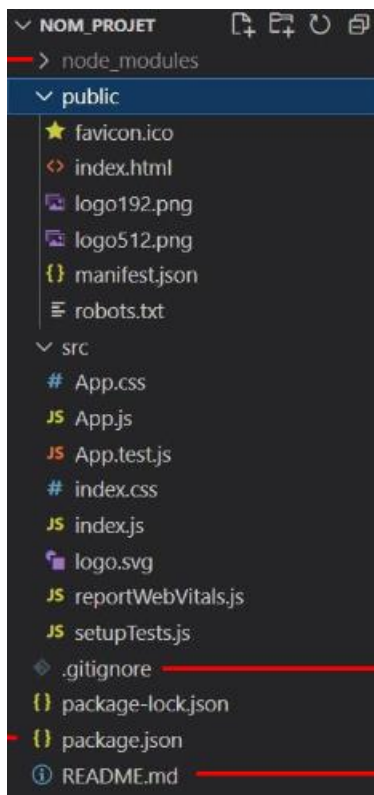
: HeidiSQL (outil d'administration de base de données possédant un éditeur SQL et un constructeur de requête).



: Postman (permet des testers des API).

3. Comment les utilisés :

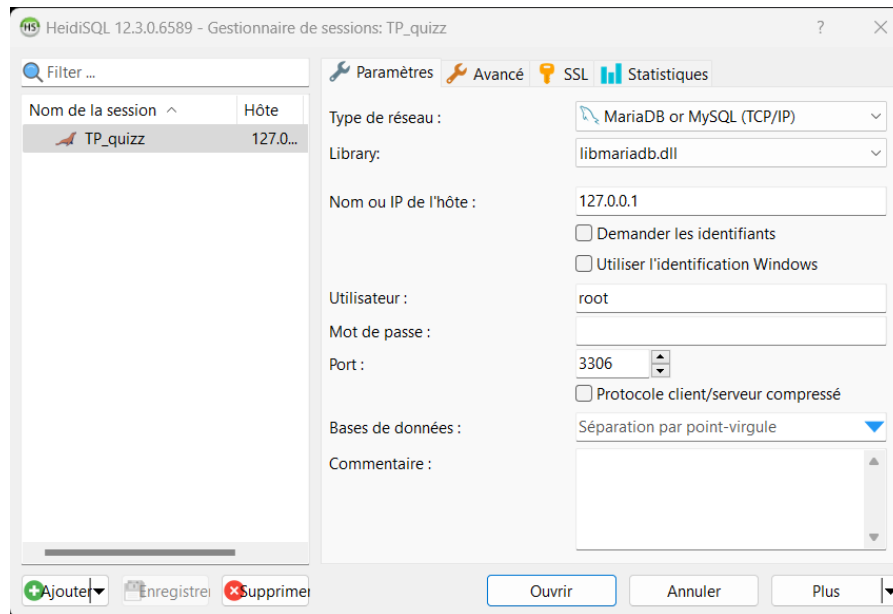
Dans un premier temps, nous ouvrons le terminal et on insère la commande suivant pour créer un projet React. « `npx create-react-app nom_projet` ». Cette commande va créer le projet en lui ajoutant différent dossiers.



Ensuite on se déplace dans le projet avec « `cd nom_projet` » Et enfin nous lançons le projet avec « `npm start` ».

Ensuite nous téléchargeons Node.js, puis nous lançons la commande `node -v` pour vérifier que Node.js est bien téléchargé sur le système

De plus nous commençons par créer la base de données sur HeidiSQL.



Nous créons ensuite les différentes tables.

#	Nom	Type de données	Taille/Ensem...	Non sig...	NULL aut...	ZERO...	Par défaut
1	id	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREME...
2	email	VARCHAR	255	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Pas de défaut
3	mdp	VARCHAR	255	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Pas de défaut

Ces tables sont utilisées pour récupérer l'email ainsi que le mot de passe créé dans l'onglet inscription sur le frontend du site.

Ajouter un compte

Email Mot de Passe

Sur Visual Studio Code, nous configurons le fichier serveur.js.

```

1  const mariadb = require('mariadb')
2  const express = require('express')
3  const app = express()
4  let cors = require('cors')
5
6  require('dotenv').config()
7
8  const pool = mariadb.createPool({
9      host: "127.0.0.1",
10     database: "ap3",
11     user: "root",
12     password: "kevin",
13     connectionLimit: 100,
14 })
15
16 app.use(express.json())
17 app.use(cors())

```

Il faut ensuite lancer le back-end dans le terminal sur Visual studio code, pour lancer le serveur.

```


PS C:\Users\kevin\Downloads\Site-e-commerce-master\Site-e-commerce-master> cd back
PS C:\Users\kevin\Downloads\Site-e-commerce-master\Site-e-commerce-master\back> npm start

> back@1.0.0 start
> node server.js

Serveur à l'écoute

```

Une fois l'adresse email et le mot de passe créer, les données s'affiche directement dans la base de données.

id	 email	mdp
1	kevin@gmail.com	kevin1234
2	gfdgdf@gmail.com	kevin1234
3	kevin@gmail.Com	gfdhfghgf
5	kevintest@gmail.com	kevin1234