INSTAL·LACIÓ INICIAL

Creació del projecte:

```
alumne@insebre-com:~/vue-project$ npm create vue@latest
Need to install the following packages:
create-vue@3.10.3
Ok to proceed? (y) y

Vue.js - The Progressive JavaScript Framework

✓ Project name: … gmoreno-vue-project
✓ Add TypeScript? … No / Yes
✓ Add JSX Support? … No / Yes
✓ Add Vue Router for Single Page Application development? … No / Yes
✓ Add Pinia for state management? … No / Yes
✓ Add Vitest for Unit Testing? … No / Yes
✓ Add an End-to-End Testing Solution? › No
✓ Add ESLint for code quality? … No / Yes
✓ Add Vue DevTools 7 extension for debugging? (experimental) … No / Yes

Scaffolding project in /home/alumne/vue-project/gmoreno-vue-project...

Done. Now run:

  cd gmoreno-vue-project
  npm install
  npm run dev

npm notice
npm notice New patch version of npm available! 10.5.0 -> 10.5.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.5.2
npm notice Run npm install -g npm@10.5.2 to update!
npm notice
```

Accedim a la carpeta creada i instal·lem paquets inicials

```
alumne@insebre-com:~/vue-project$ cd gmoreno-vue-project/
alumne@insebre-com:~/vue-project/gmoreno-vue-project$ npm install

added 27 packages, and audited 28 packages in 15s

4 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

Amb run dev podem executar el servidor per a visualitzar la pàgina web.

```
alumne@insebre-com:~/vue-project/gmoreno-vue-project$ npm run dev

> gmoreno-vue-project@0.0.0 dev
> vite


  VITE v5.2.8  ready in 1058 ms

  →  Local:   http://localhost:5173/
  →  Network: use --host to expose
  →  press h + enter to show help
```

La llibreria que he triat es Prime Vue, he triat aquesta perquè té moltes valoracions positives i amb una gran quantitat de components.

Instal·lació de la llibreria i alguns components extra

```
alumne@insebre-com:~/vue-project/gmoreno-vue-project$ npm install primevue

added 2 packages, and audited 30 packages in 2s

4 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

```
alumne@insebre-com:~/vue-project/gmoreno-vue-project$ npm install primeicons

added 1 package, and audited 31 packages in 566ms

4 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
alumne@insebre-com:~/vue-project/gmoreno-vue-project$
```

```
alumne@insebre-com:~/vue-project/gmoreno-vue-project$ npm install primeflex

added 1 package, and audited 32 packages in 1s

4 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

ROUTES

Primer, he creat l'arxiu routes.js amb les diferents pagines i la ruta que tindran.

```js
import { createRouter, createWebHistory } from 'vue-router';
import Home from '../views/HomeView.vue';
import users_table from '../views/CrudView.vue';
import Register from '../views/RegisterView.vue';
import Login from '../views/LoginView.vue';
import UserProfile from '../components/UserProfile.vue';

const routes = [ //Per a activar pagines nomes tinguin acces despres de haber iniciat sessió "meta: { requiresAuth: true }"
  { path: '/',  component: Home },
  { path: '/users-table', component: users_table},
  { path: '/sign-up', component: Register },
  { path: '/login', component: Login },
  { path: '/profile', component: UserProfile, meta: { requiresAuth: true } },
];

const router = createRouter({
  history: createWebHistory(),
  routes
});

/*

Logica que utilitza un store per a nomes permetre acces a aquells usuaris que han iniciat sessió

router.beforeEach((to, from, next) => {
  if (to.matched.some(record => record.meta.requiresAuth)) {
    // Comprueba si el usuario está autenticado
    if (!store.state.isAuthenticated) {
      next({ path: '/login', query: { from: to.path} });
    } else {
      next();
    }
  } else {
    next();
  }
});

*/

export default router;
```

El header mostrara les diferents rutes. Per a la creació del header he utilitzat MenuBar de PrimeVue

```html
<template>
    <div>
        <div class="header-container">
            <Menubar :model="items" class="p-menubar"> <!-- Component MenuBar de PrimeVue -->
                <template #item="{ item, props, hasSubmenu }">
                    <router-link v-if="item.route" v-slot="{ href, navigate }" :to="item.route" custom>
                        <a ripple :href="href" v-bind="props.action" @click="navigate" class="nav-link">
                            <span :class="item.icon" />
                            <span class="ml-2">{{ item.label }}</span>
                        </a>
                    </router-link>
                    <a v-else ripple :href="item.url" :target="item.target" v-bind="props.action" class="nav-link">
                        <span :class="item.icon" />
                        <span class="ml-2">{{ item.label }}</span>
                        <span v-if="hasSubmenu" class="pi pi-fw pi-angle-down ml-2" />
                    </a>
                </template>
            </Menubar>
        </div>
    </div>
</template>
```

```
23    <script setup>
24    import { ref } from 'vue';
25    import { useRouter } from 'vue-router';
26
27    const router = useRouter();
28
29    // Ruta a les pagines
30    const items = ref([
31      { label: 'Home', route: '/' },
32      { label: 'Clasificación', route: '/users-table' },
33      { label: 'Registro', route: '/sign-up' },
34      { label: 'Iniciar session', route: '/login' },
35      {
36        label: 'Perfil',
37        items: [
38          { label: 'Perfil', route: '/profile' },
39          { label: 'Logout', route: '/logout' }
40        ]
41      },
42    ]);
43    </script>
44
```

Dins de App.vue, he cridat als diferents components de la següent manera

```
src > V App.vue > {} style
1    <template>
2      <div id="app">
3        <Header />
4          <main>
5            <router-view />
6          </main>
7        <Footer />
8      </div>
9    </template>
10
11   <script>
12   import Header from './components/header.vue';
13   import Footer from './components/footer.vue';
14
15   export default {
16     name: 'App',
17     components: {
18       Header,
19       Footer,
20     }
21   };
22   </script>
```
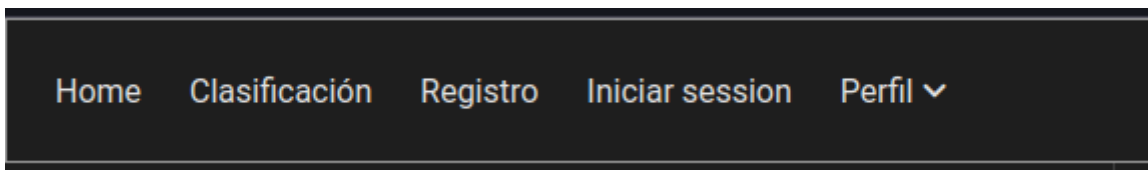
Aixi es el resultat.

Home   Clasificación   Registro   Iniciar session   Perfil ✓

PAGINA INICIAL

```vue
<template>
  <div class="background">
    <div class="overlay"></div>
    <div class="content animate-fade-in">
      <h1 class="title">
        TEEWORLDS QUACK SERVERS
      </h1>
      <Button label="Descargar" rounded raised class="custom-button" />
    </div>
  </div>

  <section class="news-section">
    <div class="container">
      <div class="text-center mb-8">
        <h2 class="section-title">Últimas Noticias</h2>
      </div>
      <div class="news-container">
        <div class="table-wrapper">
          <table class="news-table">
            <thead>
              <tr>
                <th class="news-header">Lanzamiento de los Nuevos Servidores</th>
              </tr>
            </thead>
            <tbody>
              <tr>
                <td class="news-content">
                  <p>Estamos emocionados de anunciar el lanzamiento de nuestros nuevos servidores. Después de meses de desarrollo y pruebas exhaustivas, finalmente están disponibles para que la comunidad los
                  <br>
                  <p>Los nuevos servidores ofrecen una experiencia de juego mejorada, con una mayor estabilidad, rendimiento optimizado y nuevas características emocionantes. Desde nuevas arenas hasta modos d
                  <br>
                  <p>¡Únete a nosotros en este emocionante lanzamiento y sumérgete en la acción de Teeworlds Quack Serves hoy mismo!</p>
                </td>
              </tr>
            </tbody>
            <tfoot>
              <tr>
                <td class="news-footer">26-05-2024</td>
              </tr>
            </tfoot>
          </table>
        </div>
      </div>
    </div>
  </section>

  <hr class="divider">

  <section class="community-section">
    <div class="container">
      <div class="text-center">
        <h2 class="section-title">Únete a Nuestra Comunidad</h2>
      </div>
      <div class="buttons-container">
        <div class="button-row">
          <Button label="Discord" raised class="community-button discord" />
          <Button label="Twitter" raised class="community-button twitter" />
          <Button label="Reddit" raised class="community-button reddit" />
        </div>
        <div class="button-row">
          <Button label="Soporte" raised class="community-button support" />
        </div>
      </div>
    </div>
  </section>

  <hr class="divider">

  <section class="download-section">
    <div class="container">
      <div class="text-center">
        <h2 class="section-title">Descarga el cliente oficial de Quack Serves</h2>
      </div>
      <div class="buttons-container">
        <div class="button-row">
          <Button label="Windows" raised class="download-button windows" />
          <Button label="Mac" raised class="download-button mac" />
          <Button label="Linux" raised class="download-button linux" />
        </div>
      </div>
    </div>
  </section>
</template>
```
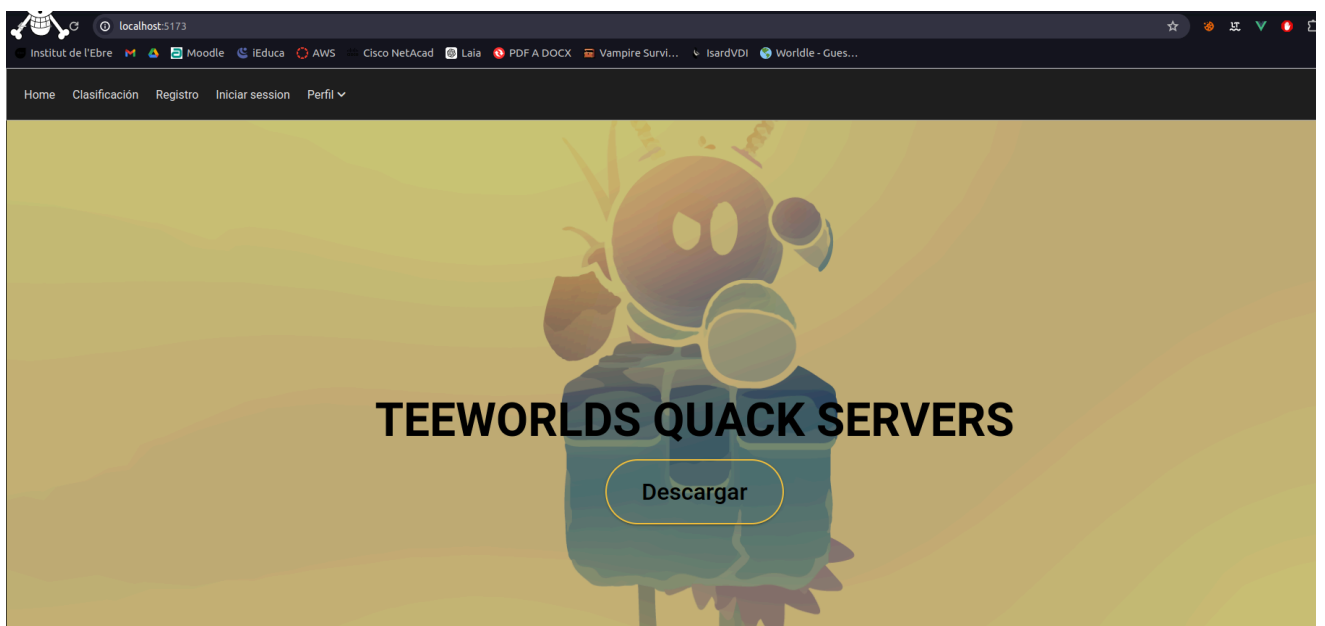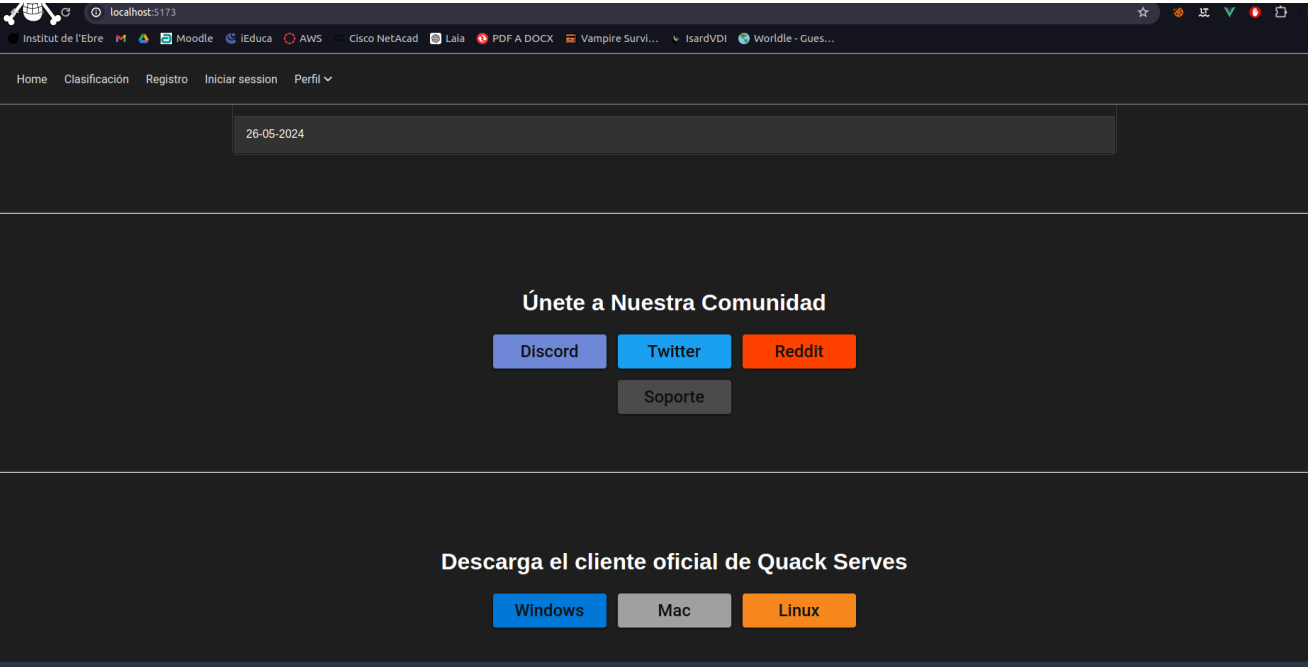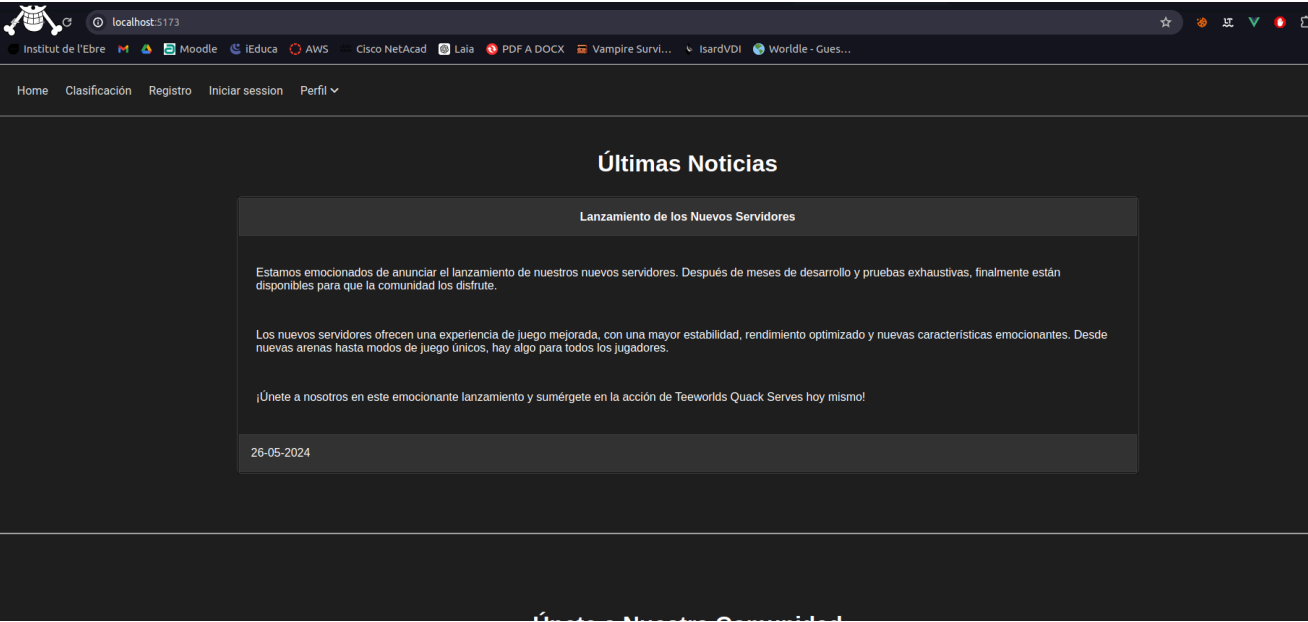
```
85    <script>
86    export default {
87      name: "Home",
88    };
89    </script>
90
91    <style>
92    @keyframes fadeIn {
93      0% {
94        opacity: 0;
95        transform: translateY(20px);
96      }
97      100% {
98        opacity: 1;
99        transform: translateY(0);
100     }
101   }
102
103   .animate-fade-in {
104     animation: fadeIn 2s ease-in-out;
105   }
106
107   .background {
108     position: relative;
109     height: 100vh;
110     background: url('src/assets/teeworlds_wallpaper.png') no-repeat center center;
111     background-size: cover;
112   }
113
114   .overlay {
115     position: absolute;
116     inset: 0;
117     background-color: □rgba(0, 0, 0, 0.2);
118   }
119
120     content {
```

Últimas Noticias

Lanzamiento de los Nuevos Servidores

Estamos emocionados de anunciar el lanzamiento de nuestros nuevos servidores. Después de meses de desarrollo y pruebas exhaustivas, finalmente están disponibles para que la comunidad los disfrute.

Los nuevos servidores ofrecen una experiencia de juego mejorada, con una mayor estabilidad, rendimiento optimizado y nuevas características emocionantes. Desde nuevas arenas hasta modos de juego únicos, hay algo para todos los jugadores.

¡Únete a nosotros en este emocionante lanzamiento y sumérgete en la acción de Teeworlds Quack Serves hoy mismo!

26-05-2024

Únete a Nuestra Comunidad

Únete a Nuestra Comunidad

Discord   Twitter   Reddit

Soporte

Descarga el cliente oficial de Quack Serves

Windows   Mac   Linux

FOOTER

```vue
<template>
  <footer class="footer">
    <div class="footer-container">
      <div class="footer-grid">
        <!-- Columna 1 -->
        <div>
          <p class="footer-text">© 2024 Teeworlds Quack Serves. Fake Web.</p>
        </div>
        <!-- Columna 2 -->
        <div>
          <h4 class="footer-heading">Redes Sociales</h4>
          <ul class="footer-list">
            <li><a href="#" class="footer-link">Discord</a></li>
            <li><a href="#" class="footer-link">Twitter</a></li>
            <li><a href="#" class="footer-link">Reddit</a></li>
          </ul>
        </div>
        <!-- Columna 3 -->
        <div>
          <h4 class="footer-heading">Recursos</h4>
          <ul class="footer-list">
            <li><a href="#" class="footer-link">Noticias</a></li>
          </ul>
        </div>
        <!-- Columna 4 -->
        <div>
          <h4 class="footer-heading">Acerca de</h4>
          <ul class="footer-list">
            <li><a href="#" class="footer-link">Equipo</a></li>
            <li><a href="#" class="footer-link">Términos de Servicio</a></li>
            <li><a href="#" class="footer-link">Política de Privacidad</a></li>
          </ul>
        </div>
      </div>
    </div>
  </footer>
</template>
```

```
39    <style>
40    .footer {
41      background-color: ☐#2d3748; /* Similar a bg-gray-800 */
42      color: ■#d1d5db; /* Similar a text-gray-300 */
43      padding: 2rem 0;
44    }
45
46    .footer-container {
47      max-width: 80rem; /* Similar a max-w-7xl */
48      margin: 0 auto;
49      padding: 0 1rem; /* Similar a px-4 */
50    }
51
52    .footer-grid {
53      display: grid;
54      grid-template-columns: 1fr;
55      gap: 2rem;
56    }
57
58    @media (min-width: 640px) {
59      .footer-grid {
60        grid-template-columns: repeat(2, 1fr);
61      }
62    }
63
64    @media (min-width: 1024px) {
65      .footer-grid {
66        grid-template-columns: repeat(4, 1fr);
67      }
68    }
69
70    .footer-text {
71      padding-top: 1rem;
72      padding-bottom: 1rem;
73      margin-bottom: 0.5rem;
74    }
75
76    .footer-heading {
77      font-size: 1.125rem; /* Similar a text-lg */
78      font-weight: 600; /* Similar a font-semibold */
79      margin-bottom: 1rem;
80    }
81
82    .footer-list {
83      list-style: none;
84      padding: 0;
85    }
86
87    .footer-link {
88      color: ■#d1d5db; /* Similar a text-gray-300 */
89      text-decoration: none;
90      transition: color 0.3s;
91    }
92
93    .footer-link:hover {
94      color: ■#9ca3af; /* Similar a hover:text-gray-400 */
95    }
96    </style>
```

© 2024 Teeworlds Quack Serves. Fake Web.

Redes Sociales
Discord
Twitter
Reddit

Recursos
Noticias

Acerca de
Equipo
Términos de Servicio
Política de Privacidad

CREACIÓ CRUD

DATABASE

El primer es crear la base de dades, jo utilitzaré mysql.

```
gerard@PC001-OF:~/projects/vue-projects/2024-ASIX2-MP09-vue-page$ sudo mysql -u gerard -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 21
Server version: 8.0.36-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql> create database vueproject;
Query OK, 1 row affected (0.01 sec)

mysql> use vueproject;
Database changed
```
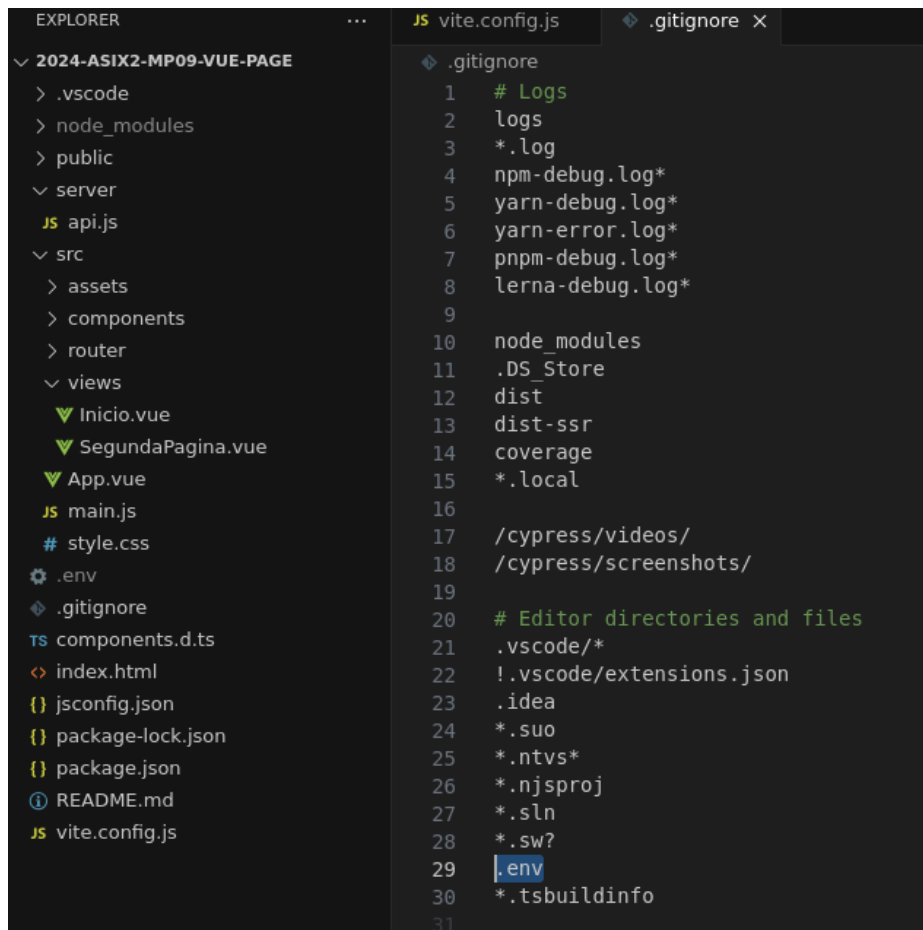
```
mysql> create table users (
    -> id INT AUTO_INCREMENT PRIMARY KEY,
    -> name VARCHAR(255) NOT NULL,
    -> email VARCHAR(255) NOT NULL
    -> );
Query OK, 0 rows affected (0.05 sec)

mysql> exit
```

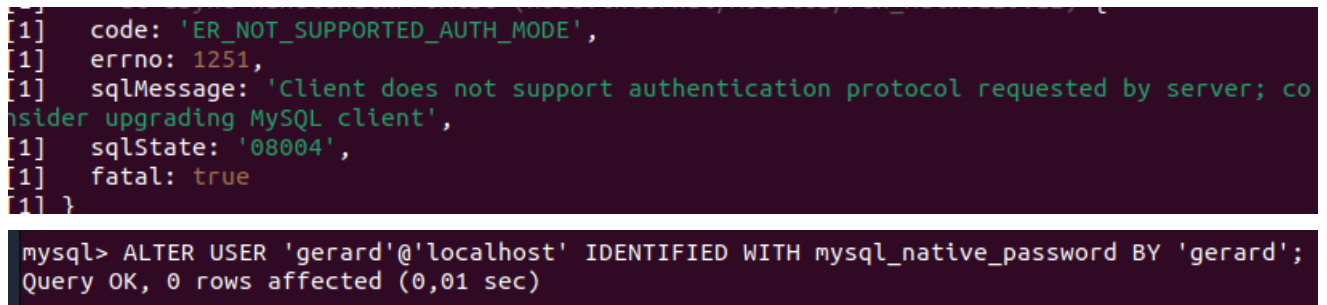Dins del directori del projecte, he creat un arxiu .env amb els següents parametres.

```
1   # API CONF
2   JWT_SECRET=eb47486c46898e7b04aad056dd311f2eeeda4a5ae5fc6d446fb1bf687571a15ae67111694767be337bf658b86afe5a1685d7d2da3f815303dea3480a00fcb35c
3   API_PORT=3657
4
5   #MYSQL CONF
6   DB_HOST=localhost
7   DB_USER=gerard
8   DB_PASSWORD=gerard
9   DB_DATABASE=vueproject
10  DB_TABLE=users
11
```

En projectes reals és important afegir dins de .gitignore l'arxiu .env

Si mostra aquest error, es culpa del format de la contrasenya que utilitza MySQL.

```
[1]     code: 'ER_NOT_SUPPORTED_AUTH_MODE',
[1]     errno: 1251,
[1]     sqlMessage: 'Client does not support authentication protocol requested by server; co
nsider upgrading MySQL client',
[1]     sqlState: '08004',
[1]     fatal: true
[1] }
```

```
mysql> ALTER USER 'gerard'@'localhost' IDENTIFIED WITH mysql_native_password BY 'gerard';
Query OK, 0 rows affected (0,01 sec)
```

API

```
gerard@PC001-OF:~/projects/vue-projects/2024-ASIX2-MP09-vue-page$ npm install mysql

added 11 packages, and audited 88 packages in 3s

18 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

```
gerard@PC001-OF:~/projects/vue-projects/2024-ASIX2-MP09-vue-page$ npm install express

up to date, audited 331 packages in 885ms

61 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

```
gerard@PC001-OF:~/projects/vue-projects/2024-ASIX2-MP09-vue-page$ npm install dotenv

added 1 package, and audited 89 packages in 718ms

19 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

JS api.js   ✕

server > JS api.js > ⊘ connection.connect() callback

```js
 1   import bcrypt from 'bcrypt';
 2   import jwt from 'jsonwebtoken';
 3   import express from 'express';
 4   import mysql from 'mysql';
 5   import dotevn from 'dotenv'
 6   dotevn.config()
 7
 8   const api = express();
 9
10   api.use(express.json());
11
12   // Obtener datos la base de datos usando dotenv
13   const connection = mysql.createConnection({
14     host: process.env.DB_HOST,
15     user: process.env.DB_USER,
16     password: process.env.DB_PASSWORD,
17     database: process.env.DB_DATABASE,
18   });
19
```

Connexio a la base de dades i creo un usuari que es diu admin

```
20    connection.connect((err) => {
21      if (err) {
22        console.error('Error conectando a la base de datos:', err);
23        return;
24      }
25      console.log('Conectado a la base de datos.');
26
27      const username = 'admin';
28      const email = 'admin@admin.com';
29      const password = 'admin';
30      const hashedPassword = bcrypt.hashSync(password, 10);
31      const points = 9999999;
32
33      // Comprobar si l'usuari existeix
34      connection.query(`SELECT * FROM ${process.env.DB_DATABASE}.${process.env.DB_TABLE} WHERE username = ? OR email = ?`, [username, email], (error, results) => {
35        if (error) {
36          console.error('Error al verificar el usuario:', error);
37          return;
38        }
39
40        if (results.length > 0) {
41          console.log('Credenciales: admin:admin');
42        } else {
43          // Insertar a l'usuari administrador
44          const user = {
45            username: username,
46            email: email,
47            password: hashedPassword,
48            points: points,
49          };
50          connection.query(`INSERT INTO ${process.env.DB_DATABASE}.${process.env.DB_TABLE} SET ?`, user, (error, results) => {
51            if (error) {
52              console.error('Error al insertar el usuario:', error);
53            } else {
54              console.log(`Credenciales: admin:admin`);
55            }
56          });
57        }
58      });
59    });
```

Logica per obtenir les dades de la API.

```
61    // Query per a obtenir les dades de la taula "users"
62    api.get('/users', (req, res) => {
63      const query = 'SELECT * FROM users';
64      connection.query(query, (err, results) => {
65        if (err) {
66          console.error('Error al obtener usuarios:', err);
67          res.status(500).json({ error: 'Error al obtener usuarios' });
68          return;
69        }
70        res.json(results);
71      });
72    });
73
```

Aquesta part és la lògica per al formulari de registre.

```
74   // Logica per al registre
75   api.post('/register', async (req, res) => {
76     const { username, email, password } = req.body;
77
78     // Comprovar si el correu electronic existeix
79     const userExistsQuery = 'SELECT * FROM users WHERE email = ?';
80     connection.query(userExistsQuery, [email], async (err, results) => {
81       if (err) {
82         console.error('Error al verificar usuario:', err);
83         res.status(500).json({ error: 'Error al verificar usuario' });
84         return;
85       }
86
87       if (results.length > 0) {
88         res.status(400).json({ error: 'El correo electrónico ya existe' });
89         return;
90       }
91
92       // Comprovar si el nom d'usuari existeix
93       const userExistsQuery = 'SELECT * FROM users WHERE username = ?';
94       connection.query(userExistsQuery, [username], async (err, results) => {
95         if (err) {
96           console.error('Error al verificar usuario:', err);
97           res.status(500).json({ error: 'Error al verificar usuario' });
98           return;
99         }
100
101         if (results.length > 0) {
102           res.status(400).json({ error: 'El nombre de usuario ya existe' });
103           return;
104         }
105
106         //Hashear la contrasenya i insertar l'usuari
107         const hashedPassword = await bcrypt.hash(password, 10);
108         const query = 'INSERT INTO users (username, email, password) VALUES (?, ?, ?)';
109         connection.query(query, [username, email, hashedPassword], (err, results) => {
110           if (err) {
111             console.error('Error al registrar usuario:', err);
112             res.status(500).json({ error: 'Error al registrar usuario' });
113             return;
114           }
115           res.json({ message: 'Usuario registrado con éxito' });
116         });
117       });
118     });
119   });
```

Aquesta part, es la logica per al formulari d'inici de sessió.

```
121   // Logica per al login
122   api.post('/login', (req, res) => {
123     const { username, password } = req.body;
124     const query = 'SELECT * FROM users WHERE username = ?';
125     connection.query(query, [username], async (err, results) => {
126       if (err || results.length === 0) {
127         res.status(401).json({ error: 'Usuario o contraseña incorrectos' });
128         return;
129       }
130
131       const user = results[0];
132       // Comprovar la contrasenya es correcta
133       const passwordMatches = await bcrypt.compare(password, user.password);
134       if (!passwordMatches) {
135         res.status(401).json({ error: 'Usuario o contraseña incorrectos' });
136         return;
137       }
138       try {
139         const payload = {
140           id: user.id,
141           email: user.email,
142           created_at: user.created_at,
143           username: user.username
144         };
145         const token = jwt.sign(
146           payload,
147           process.env.JWT_SECRET,
148         );
149         res.json({
150           token,
151           success: true,
152           user: {
153             email: payload.email,
154             username: payload.username,
155           },
156         });
157       }
158       catch (error) {
159         console.log(error);
160         res.status(401).json({ error: 'Usuario o contraseña incorrectos' });
161       }
162     });
163   });
```

Per ultim, la logica per a iniciar el servidor express.

```
202    // Iniciar el servidor
203    api.listen(process.env.API_PORT, () => {
204      console.log(`Servidor de API iniciado en http://localhost:${process.env.API_PORT}`);
205    });
```

Per a que al fer "npm run dev", he utilitzat concurrently i he modificat el package.json

```
JS api.js    M        {} package.json ✕

{} package.json > ...
    1    {
    2      "name": "gmoreno-vue-project",
    3      "version": "0.0.0",
    4      "private": true,
    5      "type": "module",
         ▷ Debug
    6      "scripts": {
    7        "dev": "concurrently \"npm run serve:vite\" \"npm run serve:api\"",
    8        "serve:vite": "vite",
    9        "build": "vite build",
   10        "preview": "vite preview",
   11        "serve:api": "node server/api.js"
   12      },
```

```
←  →  C    ⓘ  localhost:3657/users

Dar formato al texto ☑

[
  {
    "id": 1,
    "username": "admin",
    "email": "admin@example.com",
    "password": "admin",
    "created_at": "2024-04-26T11:14:21.000Z",
    "updated_at": "2024-04-26T11:14:21.000Z"
  },
```

CRUD

```
V CrudView.vue M ✕

src > views > V CrudView.vue > {} template
    1    <template>
    2      <div>
    3        <div id="page-container">
    4          <br>
    5          <h1>Clasificacion</h1>
    6          <InputText v-model="searchText" placeholder="Buscar jugador"/>
    7          <DataTable :value="filteredUsers" stripedRows tableStyle="min-width: 50rem">
    8            <Column field="id" header="ID"></Column>
    9            <Column field="username" header="Nombre"></Column>
   10            <Column field="points" sortable header="Puntos"></Column>
   11          </DataTable>
   12        </div>
   13      </div>
   14    </template>
```

```
16  <script>
17  import axios from 'axios';
18
19  export default {
20    name: 'SegundaPagina',
21
22    data() {
23      return {
24        users: [],
25        searchText: ''
26      };
27    },
28    computed: {
29      filteredUsers() {
30        if (!this.searchText) {
31          return this.users;
32        } else {
33          const searchTextLower = this.searchText.toLowerCase();
34          return this.users.filter(user =>
35            user.username.toLowerCase().includes(searchTextLower)
36          );
37        }
38      }
39    },
40    mounted() {
41      this.fetchUsers();
42    },
43    methods: {
44      async fetchUsers() {
45        try {
46          const response = await axios.get('/api/users');
47          this.users = response.data;
48        } catch (error) {
49          console.error('Error al obtener usuarios:', error);
50        }
51      }
52    }
53  };
54  </script>
```

```
56  <style>
57  #page-container {
58    display: flex;
59    flex-direction: column;
60    margin: 0rem 2rem 2rem 2rem
61  }
62  </style>
```


localhost:5173/users-table

Institut de l'Ebre   M   △   Moodle   iEduca   AWS   Cisco NetAcad   Laia   PDF A DOCX   Vampire Survi...   IsardVDI   Worldle - Gues...

Home   Clasificación   Registro   Iniciar session   Perfil ⌄

Clasificacion

Buscar jugador

ID	Nombre	Puntos ↑↓
1	martintimothy	14
2	millerjustin	845
3	collinsvincent	938
4	gregory37	820
5	nataliebaxter	272
6	gonzaleseric	585
7	ryankara	251

LOGIN I REGISTRE

```vue
1   <template>
2     <form @submit.prevent="submitForm">
3       <div>
4         <label for="username">Nombre de usuario:</label>
5         <input type="text" id="username" v-model="username" required autocomplete="username">
6       </div>
7       <div>
8         <label for="password">Contraseña:</label>
9         <input type="password" id="password" v-model="password" required>
10      </div>
11      <div v-if="error" class="error-message">{{ error }}</div><br>
12      <Button type="submit">Iniciar sessión</Button>
13    </form>
14  </template>
15
16  <script>
17  import axios from 'axios';
18
19  export default {
20    data() {
21      return {
22        username: '',
23        password: '',
24        error: ''
25      };
26    },
27    // Validar dades form
28    methods: {
29      async submitForm() {
30        try {
31          const response = await axios.post('/api/login', {
32            username: this.username,
33            password: this.password
34          })
35          // Afegir cookie
36          document.cookie = `jwt=${response.data.token}`
37          localStorage.setItem('localjwt',response.data.token);
38          this.$store.commit('setAuthentication', true);
39          router.push('/');
40          // Moure a la pagina solicitada en cas de utilitzar "requierauth" this.$router.replace(this.$route.query.from);
41        } catch (error) {
42          this.error = error.response.data.error
43          console.log(error)
44        }
45      }
46    }
47  };
48  </script>
```

Home   Clasificación   Registro   Iniciar session   Perfil ⌄

Login

Nombre de usuario:

Contraseña:

Iniciar sessión
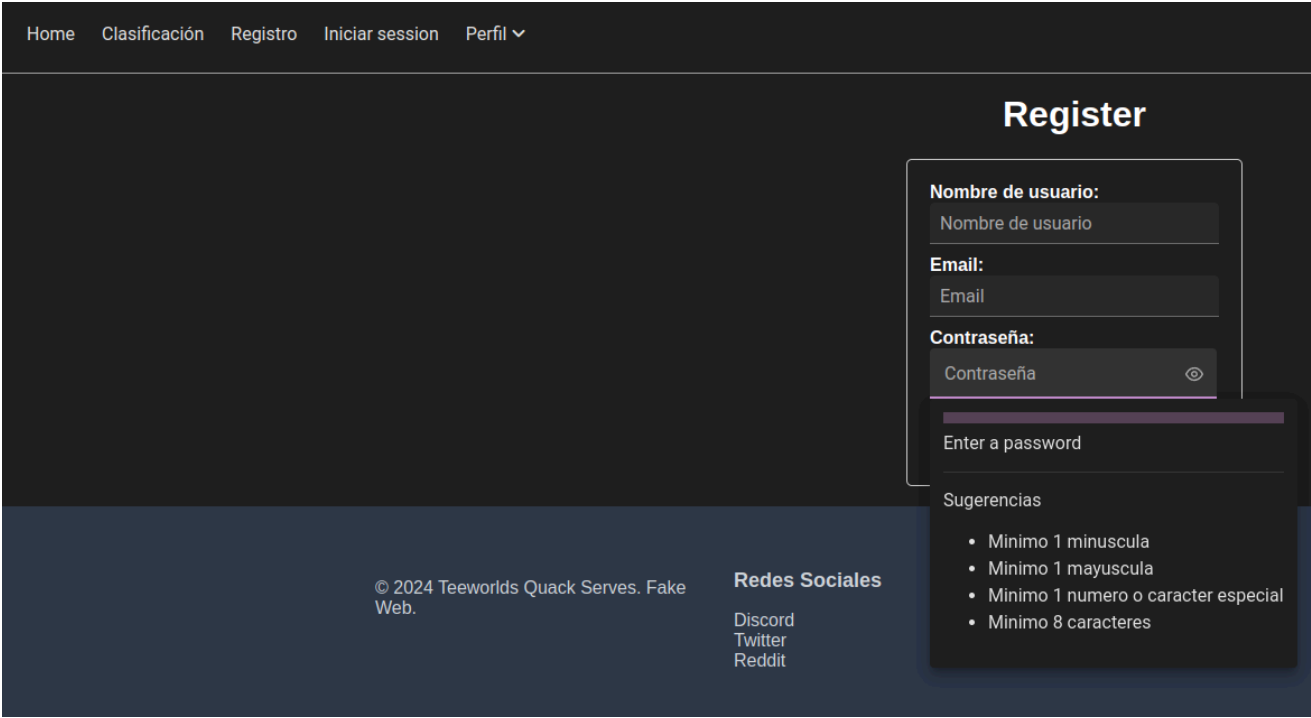
```vue
<!-- components/RegisterForm.vue -->
<template>
  <form @submit.prevent="submitForm">
    <div>
      <label for="username">Nombre de usuario:</label>
      <InputText type="username" id="username" v-model="username" required autocomplete="username" placeholder="Nombre de usuario"/>
    </div>
    <div>
      <label for="email">Email:</label>
      <InputText type="email" id="email" v-model="email" required autocomplete="email" placeholder="Email"/>
    </div>
    <div>
      <label for="password">Contraseña:</label>
      <Password type="password" id="password" v-model="password" required toggle-mask placeholder="Contraseña">
        <template #footer>
          <Divider />
          <p class="mt-2">Sugerencias</p>
          <ul class="pl-2 ml-2 mt-0" style="line-height: 1.5">
            <li>Minimo 1 minuscula</li>
            <li>Minimo 1 mayuscula</li>
            <li>Minimo 1 numero o caracter especial</li>
            <li>Minimo 8 caracteres</li>
          </ul>
        </template>
      </Password>
    </div>
    <div v-if="error" class="error-message">{{ error }}</div><br>
    <Button type="submit" label="Register"/>
  </form>
</template>
```

```vue
<script>
import Password from 'primevue/password';
import router from '../router/routes'
import axios from 'axios';
import InputText from 'primevue/inputtext';

// Exportar datos importantes
export default {
  data() {
    return {
      username: '',
      email: '',
      password: '',
      error: ''
    };
  },

  // Logica para enviar la informacion a la api
  methods: {
    async submitForm() {
      try {
        const response = await axios.post('/api/register', {
          username: this.username,
          email: this.email,
          password: this.password
        });
        console.log(response.data.message);
        router.push('/');
      }
      catch (error) {
        this.error = error.response.data.error
      }
    }
  },
  components: { Password, InputText }
};
</script>
```

Register

Nombre de usuario:

Nombre de usuario

Email:

Email

Contraseña:

Contraseña   👁

Enter a password

Sugerencias

- Minimo 1 minuscula
- Minimo 1 mayuscula
- Minimo 1 numero o caracter especial
- Minimo 8 caracteres

© 2024 Teeworlds Quack Serves. Fake Web.

Redes Sociales

Discord
Twitter
Reddit

PERFIL DE L'USUARI

```
     1    <template>
     2      <div>
     3        <br>
     4        <h1>En construccion</h1>
     5      </div>
     6    </template>
     7
     8    <script>
     9      import axios from 'axios';
    10
    11      export default {
    12        name: 'Perfil',
    13        data() {
    14          return {
    15            users: []
    16          };
    17        },
    18        mounted() {
    19          this.fetchUsers();
    20        },
    21        methods: {
    22          async fetchUsers() {
    23            try {
    24              const response = await axios.get('/api/users'); // Utilizar axios para obtener datos de la API
    25              this.users = response.data; // Los datos se encuentran en response.data con axios
    26            } catch (error) {
    27              console.error('Error al obtener usuarios:', error);
    28            }
    29          }
    30        }
    31      };
    32    </script>
    33
    34    <style scoped>
    35      .user-profile {
    36        max-width: 400px;
    37        margin: 0 auto;
    38      }
    39
    40      .profile-header {
    41        display: flex;
    42        align-items: center;
    43        gap: 1rem;
    44      }
    45
    46      .profile-content {
    47        margin-top: 1rem;
    48      }
    49    </style>
    50
```

SCRIPT

He creat un script per a iniciar npm, crear la base de dades i configurar alguns parametres.

```bash
#!/bin/bash
sudo apt-get install faker -y
pip install --upgrade faker
sudo service mysql start

read -p  "Vols crear la base de dades? (s/N)" dbcreation

case "$dbcreation" in
        [yYsSsiyes] )
                echo "ok, creare la base de dades de nou"
                echo "Contrasenya de l'usuari root de MySQL"
                sudo mysql -u root -proot -vv <<EOF || { echo ""; exit 1; }

                ## Comprobar si existeix la base de dades vueproject y eliminar-la
                DROP DATABASE IF EXISTS vueproject;

                ## Crear la base de dades vueproject
                CREATE DATABASE vueproject;

                ## Utilitzar la base de dades vueproject
                USE vueproject;

                ## Comprobar si existeix l'usuari gerard y eliminar-lo
                DROP USER IF EXISTS 'gerard'@'localhost';

                ## Afegir l'usuari gerard amb contrasenya gerard y tots els permisos a la base de dades vueproject
                CREATE USER 'gerard'@'localhost' IDENTIFIED WITH mysql_native_password BY 'gerard';

                GRANT ALL PRIVILEGES ON vueproject.* TO 'gerard'@'localhost';
                flush privileges;

                ## Afegir la taula users i totes les columnes corresponents
                CREATE TABLE users (
                    id INT NOT NULL AUTO_INCREMENT,
                    username VARCHAR(255) NOT NULL,
                    email VARCHAR(255) NOT NULL,
                    password VARCHAR(255) NOT NULL,
                    points INT DEFAULT 0,
                    created_at TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
                    updated_at TIMESTAMP NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
                    PRIMARY KEY (id)
                );

                INSERT INTO users (username, email, password, points, created_at, updated_at) VALUES
                $(for i in {1..15}; do
                        echo "('"$(faker user_name)"', '"$(faker email)"', '"$(faker password)"', $((RANDOM % 1001)), NOW(), NOW())"
                done | paste -sd ',' -);
EOF

        ;;
        [nNnoNO] )
                echo "ok, no creare la base de dades"
        ;;
esac

# Iniciar el servidor vite i api
sudo npm install
sudo npm run dev
```