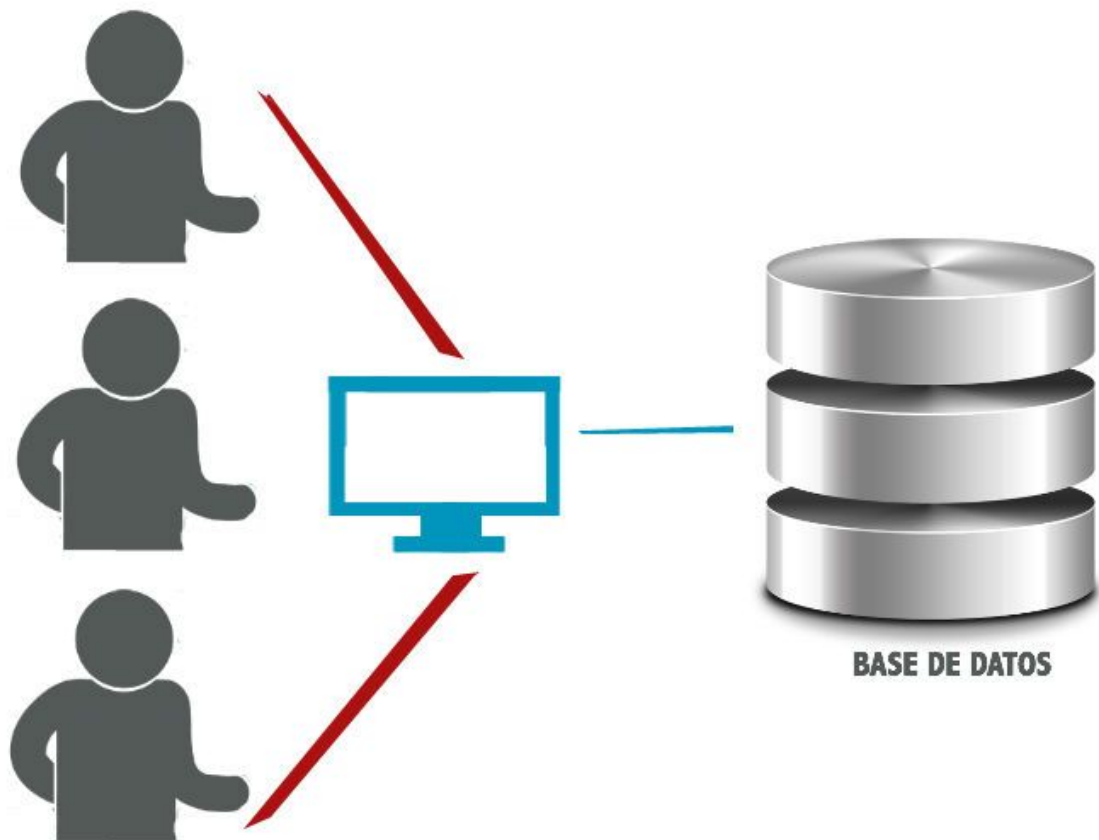


Microproyecto Almacenamiento API WEB Storage



Índice

Índice	2
Introducción	3
Requerimientos	3
Objetivo General:	3
Objetivos específicos:	3
Ideas preliminares o diseño:	3
Descripción de la solución propuesta:	3
Herramientas utilizadas	3
Codificación	4
Interfaz de Usuario:	24
Estimación	28
Conclusiones	28

Introducción

El proyecto a presentar está enfocado en brindar una plataforma que haga las veces de galería multimedia personal y segura.

Requerimientos

Objetivo General:

Desarrollar una plataforma que permite crear usuarios personales con su contenido multimedia protegido y solo a la vista de estos.

Objetivos específicos:

1. Permitir la creación de usuarios con contraseñas protegidas
2. Permitir a los usuarios agregar.
 - a. notas
 - b. audio
 - c. video
 - d. imágenes
3. Permitir a los usuarios la posibilidad de mantener o no su sesión en el navegador

Ideas preliminares o diseño:

1. Obtener contenido multimedia de prueba
2. Establecer el flujo del usuario y las vistas necesarias
3. Establecer el flujo de los datos de entrada y salida en las distintas vistas
4. Agregar estilos por medio de CSS para dar una mejor UI

Descripción de la solución propuesta:

Herramientas utilizadas

Para el desarrollo de la propuesta se han utilizado las siguientes tecnologías:

- Visual Studio Code (IDE)
- HTML
- CSS
- Javascript nativo
- Librería [Materialize](#) (Librería externa para CSS)

Codificación

- Index.html

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!-- Import Google Icon Font -->
  <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet" />
  <!-- Compiled and minified CSS -->
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css" />
  <link rel="stylesheet" href="../styles/styles.css" type="text/css" />
  <title>Mini Proyecto</title>
</head>

<body>
  <div id="page-header">
    <h1>Web Storage</h1>
  </div>

  <br />
  <br />
  <br />

  <div class="row">
    <div class="col" id="form">
      <div class="row">
        <div class="input-field col s12">
          <i class="material-icons prefix">account_circle</i>
          <input id="username" type="text" class="validate" />
          <label for="username">Username</label>
        </div>
      </div>
      <div class="row">
        <div class="input-field col s12">
          <i class="material-icons prefix">password</i>
          <input id="password" type="password" class="validate" />
          <label for="password">Password</label>
        </div>
      </div>
      <div class="row">
        <div class="col s6">
          <label>
            <input type="checkbox" class="filled-in" id="remember-me" />
            <span>Remember me</span>
          </label>
        </div>
        <div class="col s6"></div>
      </div>
      <div class="row">
        <button class="waves-effect waves-light btn primary-btn" id="login-btn">
          Login
        </button>
      </div>
      <div class="row">
        <div class="col s6">
          <a href="/view-register.html">Register user</a>
        </div>
        <div class="col s6"></div>
      </div>
    </div>
  </div>

  <!-- Compiled and minified JavaScript -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/js/materialize.min.js">
</script>
  <!-- Crypto JS import -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/crypto-js/3.1.9-1/crypto-js.min.js"></script>
  <script src="../scripts/user-actions-control.js"></script>
  <script src="../scripts/lib.js"></script>
</body>

</html>
```

- view-audios.html

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!-- Import Google Icon Font -->
  <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet" />
  <!-- Compiled and minified CSS -->
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css" />
  <link rel="stylesheet" href="../../styles/styles.css" type="text/css" />
  <title>Add audio</title>
</head>

<body>
  <br />
  <br />
  <br />
  <br />
  <br />

  <div id="page-header">
    <h1>Add audio</h1>
  </div>
  <br />
  <br />
  <br />

  <div class="row">
    <div class="col" id="form">
      <div class="row">
        <div class="input-field col s12">
          <i class="material-icons prefix">headphones</i>
          <input id="audio" type="file" accept="audio/*" style="margin-top: 12px;" class="validate" />
        </div>
      </div>
      <div class="row">
        <button class="waves-effect waves-light btn primary-btn" id="create-audio-btn">
          Add
        </button>
      </div>
    </div>
  </div>

  <!-- Compiled and minified JavaScript -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/js/materialize.min.js">
</script>
  <script src="../../scripts/user-actions-control.js"></script>
  <script src="../../scripts/lib.js"></script>
</body>

</html>

```

- view-images.html

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!--Import Google Icon Font-->
  <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet" />
  <!-- Compiled and minified CSS -->
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css" />
  <link rel="stylesheet" href="../../styles/styles.css" type="text/css" />
  <title>Add images</title>
</head>

<body>
  <br />
  <br />
  <br />
  <br />

  <div id="page-header">
    <h1>Add image</h1>
  </div>
  <br />
  <br />
  <br />

  <div class="row">
    <div class="col" id="form">
      <div class="row">
        <div class="input-field col s12">
          <i class="material-icons prefix">image</i>
          <input id="image" type="file" accept="image/x-png,image/gif,image/jpeg" style="margin-top:
12px;" class="validate" />
        </div>
      </div>
      <div class="row">
        <button class="waves-effect waves-light btn primary-btn" id="create-image-btn">
          Add
        </button>
      </div>
    </div>
  </div>

  <!-- Compiled and minified JavaScript -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/js/materialize.min.js">
</script>
  <script src="../../scripts/user-actions-control.js"></script>
  <script src="../../scripts/lib.js"></script>
</body>

</html>

```

- view-notes.html

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!--Import Google Icon Font-->
  <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet" />
  <!-- Compiled and minified CSS -->
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css" />
  <link rel="stylesheet" href="../../styles/styles.css" type="text/css" />
  <title>Add Note</title>
</head>

<body>
  <br />
  <br />
  <br />

  <div id="page-header">
    <h1>Create note</h1>
  </div>
  <br />
  <br />

  <div class="row">
    <div class="col" id="form">
      <div class="row">
        <div class="input-field col s12">
          <i class="material-icons prefix">title</i>
          <textarea id="title" style="resize: none" class="validate"></textarea>
          <label for="title">Title</label>
        </div>
      </div>
      <div class="row">
        <div class="input-field col s12">
          <i class="material-icons prefix">note</i>
          <textarea id="content" style="resize: none" class="validate"></textarea>
          <label for="content">Note content</label>
        </div>
      </div>
      <div class="row">
        <button class="waves-effect waves-light btn primary-btn" id="create-note-btn">
          Create
        </button>
      </div>
    </div>
  </div>

  <!-- Compiled and minified JavaScript -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/js/materialize.min.js">
</script>
  <script src="../../scripts/user-actions-control.js"></script>
  <script src="../../scripts/lib.js"></script>
</body>

</html>

```

- view-register.html


```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!--Import Google Icon Font-->
  <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet" />
  <!-- Compiled and minified CSS -->
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css" />
  <link rel="stylesheet" href="../../styles/styles.css" type="text/css" />
  <title>Register User</title>
</head>

<body>
  <br />
  <br />
  <div id="page-header">
    <h1>Register</h1>
  </div>
  <br />
  <br />
  <br />
  <div class="row">
    <div class="col" id="form">
      <div class="row">
        <div class="input-field col s12">
          <i class="material-icons prefix">account_circle</i>
          <input id="username" type="text" class="validate" />
          <label for="username">Username</label>
        </div>
      </div>
      <div class="row">
        <div class="input-field col s12">
          <i class="material-icons prefix">password</i>
          <input id="password" type="password" class="validate" />
          <label for="password">Password</label>
        </div>
      </div>
      <div class="row">
        <button class="waves-effect waves-light btn primary-btn" id="register-btn">
          Register
        </button>
      </div>
    </div>
  </div>
  <!-- Compiled and minified JavaScript -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/js/materialize.min.js">
</script>
  <!-- Crypto JS import -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/crypto-js/3.1.9-1/crypto-js.min.js"></script>
  <script src="../../scripts/user-actions-control.js"></script>
  <script src="../../scripts/lib.js"></script>
</body>

</html>

```

- view-user.html


```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!--Import Google Icon Font-->
  <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet" />
  <!-- Compiled and minified CSS -->
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css" />
  <link rel="stylesheet" href="../../styles/styles.css" type="text/css" />
  <title>User Account</title>
</head>

<body>
  <ul id="slide-out" class="sidenav">
    <li>
      <div class="user-view">
        <a<img class="circle" src="" id="user-image" /></a>
        <h3 id="user-username"></h3>
      </div>
    </li>
    <li>
      <div class="divider"></div>
    </li>
    <li>
      <a id="profilepic-btn" class="render-btn"><i class="material-icons">add_a_photo
        <input type="file" id="profilepic-input" hidden accept="image/x-png,image/jpeg" /> </i>Update
Profile Picture</a>
    </li>
    <li>
      <a id="notes-btn" class="render-btn"><i class="material-icons">notes</i>Notes</a>
    </li>
    <li>
      <a id="images-btn" class="render-btn"><i class="material-icons">image</i>Images</a>
    </li>
    <li>
      <a id="videos-btn" class="render-btn"><i class="material-icons">play_arrow</i>Videos</a>
    </li>
    <li>
      <a id="audios-btn" class="render-btn"><i class="material-icons">headphones</i>Audios</a>
    </li>
    <li>
      <a id="logout" class="render-btn"><i class="material-icons">logout</i>logout</a>
    </li>
    <li>
      <a id="delete-user" class="render-btn"><i class="material-icons">delete</i>Delete User</a>
    </li>
  </ul>
  <a href="" data-target="slide-out" class="sidenav-trigger"><i class="material-icons" id="menu-
btn">menu</i></a>
  <div id="container">
    <h1 style="text-align: center;">Render content here :) </h1>
  </div>

  <div class="fixed-action-btn">
    <a class="btn-floating btn-large red">
      <i class="large material-icons">add</i>
    </a>
    <ul>
      <li id="add-note-btn">
        <a class="btn-floating red"><i class="material-icons">notes</i></a>
      </li>
      <li id="add-image-btn">
        <a class="btn-floating yellow darken-1"><i class="material-icons">image</i></a>
      </li>
      <li id="add-video-btn">
        <a class="btn-floating green"><i class="material-icons">play_arrow</i></a>
      </li>
      <li id="add-audio-btn">
        <a class="btn-floating blue"><i class="material-icons">headphones</i></a>
      </li>
    </ul>
  </div>
</body>
<script src="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/js/materialize.min.js"></script>
<script src="../../scripts/user-actions-control.js"></script>
<script src="../../scripts/render-control.js"></script>

</html>

```

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <!--Import Google Icon Font-->
  <link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet" />
  <!-- Compiled and minified CSS -->
  <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css" />
  <link rel="stylesheet" href="../../styles/styles.css" type="text/css" />
  <title>Add video</title>
</head>

<body>
  <br />
  <br />
  <br />
  <br />

  <div id="page-header">
    <h1>Add video</h1>
  </div>
  <br />
  <br />
  <br />

  <div class="row">
    <div class="col" id="form">
      <div class="row">
        <div class="input-field col s12">
          <i class="material-icons prefix">movie</i>
          <input id="video" type="file" accept="video/*" style="margin-top: 12px;" class="validate" />
        </div>
      </div>
      <div class="row">
        <button class="waves-effect waves-light btn primary-btn" id="create-video-btn">
          Add
        </button>
      </div>
    </div>
  </div>

  <!-- Compiled and minified JavaScript -->
  <script src="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/js/materialize.min.js">
</script>
  <script src="../../scripts/user-actions-control.js"></script>
  <script src="../../scripts/lib.js"></script>
</body>

</html>
```

```

body {
  background-color: rgb(241, 247, 235);
}

#page-header {
  display: flex;
  justify-content: center;
}

#form {
  background-color: rgb(224, 236, 253);
  border-radius: 8px;
  width: 22rem;
  box-shadow: 0.2rem 0.2rem 1rem rgb(182, 186, 190);
}

.row {
  justify-content: center;
  display: flex;
}

button {
  border-radius: 6px !important;
}

.primary-btn {
  background-color: rgb(143, 143, 143) !important;
}

.secondary-btn {
  background-color: rgb(143, 143, 143) !important;
}

#form .material-icons {
  color: rgb(102, 102, 102) !important;
}

#menu-btn {
  font-size: 36px !important;
  margin-left: 20px;
  margin-top: 20px;
  color: rgba(0, 0, 0, 0.671);
}

.render-btn:hover {
  cursor: pointer !important;
}

```

- lib.js



```
function clearForm() {  
  document.getElementById("username").value = "";  
  document.getElementById("password").value = "";  
}  
  
function crypt(string) {  
  let cipher = CryptoJS.AES.encrypt(string, "CIPHERKEY");  
  cipher = cipher.toString();  
  console.log(cipher);  
  
  return cipher;  
}  
  
function decrypt(string) {  
  return CryptoJS.AES.decrypt(string || "", "CIPHERKEY").toString(CryptoJS.enc.Utf8);  
}
```

- render-control.js

```

window.addEventListener("load", () => {
  /**
   * Creating DB
   */
  let db = null;
  const req = window.indexedDB.open("DB", 1);

  req.onerror = (event) => {
    console.log("error: " + event);
  };

  /**
   * user in session
   */
  const userInSession = localStorage.getItem("remember") === "true" ?
  localStorage.getItem("userInSession") : sessionStorage.getItem("userInSession");

  /**
   * render user data
   */
  req.onsuccess = (event) => {
    db = req.result;
    const userData = db
      .transaction("users")
      .objectStore("users")
      .get(userInSession);
    userData.onsuccess = (event) => {
      document.getElementById("user-username").innerHTML =
        userData.result.username;
      if (userData.result.profilePic !== null) {
        document
          .getElementById("user-image")
          .setAttribute("src", userData.result.profilePic);
      }
    };
  };

  /**
   * Left nav bar
   */
  const navBarInstance = M.Sidenav.init(document.querySelector(".sidenav"));

  /**
   * floating icon
   */
  M.FloatingActionButton.init(document.querySelector(".fixed-action-btn"));

  /**
   * render notes
   */
  document.getElementById("notes-btn")?.addEventListener("click", () => {
    document.getElementById("container").innerHTML = "";
    const req = db
      .transaction("users")
      .objectStore("users")
      .get(userInSession);
    req.onsuccess = (event) => {
      let container = document.getElementById("container");
      container.innerHTML = `<div id="page-header"><h1>Notes:</h1></div><br/>`;
      req.result.notes.forEach((element) => {
        container.innerHTML += `<div class='row'><div class='col'><div class='card blue-grey'><div
class='card-content white-text'><span class='card-title'>${element.title}</span><p>${element.content}
</p></div></div></div></div>`;
      });
      navBarInstance.close();
    };
  });
});

```

```

/**
 * render images
 */
document.getElementById("images-btn")?.addEventListener("click", () => {
  document.getElementById("container").innerHTML = "";
  const req = db
    .transaction("users")
    .objectStore("users")
    .get(userInSession);
  req.onsuccess = (event) => {
    let container = document.getElementById("container");
    container.innerHTML = `<div id="page-header"><h1>Images:</h1></div><br/>`;
    req.result.images.forEach((element) => {
      container.innerHTML += `<div class='row'><div class='col'><div class='card blue-grey'><div
class='card-content white-text'><img src=${element}></div></div></div><br>`;
    });
    navBarInstance.close();
  };
});

/**
 * render videos
 */
document.getElementById("videos-btn")?.addEventListener("click", () => {
  document.getElementById("container").innerHTML = "";
  const req = db
    .transaction("users")
    .objectStore("users")
    .get(userInSession);
  req.onsuccess = (event) => {
    let container = document.getElementById("container");
    container.innerHTML = `<div id="page-header"><h1>Videos:</h1></div><br/>`;
    req.result.videos.forEach((element) => {
      container.innerHTML += `<div class='row'><div class='col'><div class='card blue-grey'><div
class='card-content white-text'><video controls width="500"><source src=${element}></video></div></div>
</div><br>`;
    });
    navBarInstance.close();
  };
});

/**
 * render audios
 */
document.getElementById("audios-btn")?.addEventListener("click", () => {
  document.getElementById("container").innerHTML = "";
  const req = db
    .transaction("users")
    .objectStore("users")
    .get(userInSession);
  req.onsuccess = (event) => {
    let container = document.getElementById("container");
    container.innerHTML = `<div id="page-header"><h1>Audios:</h1></div><br/>`;
    req.result.audios.forEach((element) => {
      container.innerHTML += `<div class='row'><div class='col'><div class='card blue-grey'><div
class='card-content white-text'><audio controls><source src=${element}></audio></div></div></div></div>
<br>`;
    });
    navBarInstance.close();
  };
});

```


- user-actions-control.js

```
window.addEventListener("load", function () {
    if (
        window.location.pathname.split("/").pop() !== "index.html" &&
        window.location.pathname.split("/").pop() !== "view-register.html"
    ) {
        if (localStorage.getItem("remember") === "true") {
            if (
                localStorage.getItem("userInSession") === "" ||
                localStorage.getItem("userInSession") === null
            ) {
                window.location = "../views/index.html";
            }
        } else {
            if (
                sessionStorage.getItem("userInSession") === "" ||
                sessionStorage.getItem("userInSession") === null
            ) {
                window.location = "../views/index.html";
            }
        }
    } else {
        if (localStorage.getItem("remember") === "true") {
            if (
                localStorage.getItem("userInSession") !== "" &&
                localStorage.getItem("userInSession") !== null
            ) {
                window.location = "../views/view-user.html";
            }
        } else {
            if (
                sessionStorage.getItem("userInSession") !== "" &&
                sessionStorage.getItem("userInSession") !== null
            ) {
                window.location = "../views/view-user.html";
            }
        }
    }
}

/**
 * Checks indexedDB compatibility
 */
```



```

window.indexedDB =
  window.indexedDB ||
  window.mozIndexedDB ||
  window.webkitIndexedDB ||
  window.msIndexedDB;

window.IDBTransaction =
  window.IDBTransaction ||
  window.webkitIDBTransaction ||
  window.msIDBTransaction;
window.IDBKeyRange =
  window.IDBKeyRange || window.webkitIDBKeyRange ||
window.msIDBKeyRange;

if (!window.indexedDB) {
  window.alert("Your browser doesn't support a stable version of
IndexedDB.");
}

/**
 * Creating DB
 */
let db = null;
const req = window.indexedDB.open("DB", 1);

req.onerror = (event) => {
  console.log("error: " + event);
};

req.onsuccess = (event) => {
  db = req.result;
};

/**
 * user in session
 */
const userInSession =
  localStorage.getItem("remember") === "true"
    ? localStorage.getItem("userInSession")
    : sessionStorage.getItem("userInSession");

/**
 * Creating users "table"

```

```

    */
    req.onupgradeneeded = (event) => {
        event.target.result.createObjectStore("users", { keyPath:
"username" });
    };

    /**
     * Register users function
     */
    document.getElementById("register-btn")?.addEventListener("click", ()
=> {
        const transaction = db
            .transaction("users", "readwrite")
            .objectStore("users")
            .add({
                username: document.getElementById("username").value,
                password: crypt(document.getElementById("password").value),
                profilePic: null,
                notes: [],
                images: [],
                videos: [],
                audios: [],
            });

        transaction.onsuccess = (event) => {
            alert(event.target.result + " succesfully added!");
        };

        transaction.onerror = (event) => {
            alert(event.srcElement.error.message);
        };

        clearForm();
    });

    /**
     * Login user
     */
    document.getElementById("login-btn")?.addEventListener("click", () =>
{
        if (document.getElementById("remember-me").checked) {
            localStorage.setItem("remember", true);
            const req = db

```

```
.transaction("users")
.objectStore("users")
.get(document.getElementById("username").value);

req.onerror = (event) => {
  alert("Unable to retrieve data from database!");
};

req.onsuccess = (event) => {
  if (
    decrypt(req.result?.password) ===
    document.getElementById("password").value
  ) {
    localStorage.setItem(
      "userInSession",
      document.getElementById("username").value
    );
    window.location = "../views/view-user.html";
  } else {
    alert("Incorrect credentials!");
  }
};
} else {
  localStorage.setItem("remember", false);
  const req = db
    .transaction("users")
    .objectStore("users")
    .get(document.getElementById("username").value);

  req.onerror = (event) => {
    alert("Unable to retrieve data from database!");
  };

  req.onsuccess = (event) => {
    if (
      decrypt(req.result?.password) ===
      document.getElementById("password").value
    ) {
      sessionStorage.setItem(
        "userInSession",
        document.getElementById("username").value
      );
      window.location = "../views/view-user.html";
    }
  };
}
```

```

        } else {
            alert("Incorrect credentials!");
        }
    };
}
});

/**
 * update profile picture
 */
document
.getElementById("profilepic-input")
?.addEventListener("change", (event) => {
    const reader = new FileReader();
    reader.addEventListener("load", () => {
        let req = db
            .transaction("users")
            .objectStore("users")
            .get(userInSession);

        req.onerror = (event) => {
            alert("Unable to retrieve data from database!");
        };

        req.onsuccess = (event) => {
            const updatedData = req.result;
            updatedData.profilePic = reader.result;
            req = db
                .transaction("users", "readwrite")
                .objectStore("users")
                .put(updatedData);
            req.onsuccess = () => {
                alert("Profile pic updated!");
                setTimeout(() => {
                    location.reload();
                }, 1000);
            };
        };
    });

    reader.readAsDataURL(event.target.files[0]);
});

document.getElementById("profilepic-btn").addEventListener("click",
() => {

```

```

        document.getElementById("profilepic-input").click();
    });

    /**
     * add note
     */
    document.getElementById("add-note-btn").addEventListener("click", ()
=> {
        window.location = "../views/view-notes.html";
    });

    document.getElementById("create-note-btn").addEventListener("click",
() => {
        let req =
db.transaction("users").objectStore("users").get(userInSession);

        req.onerror = (event) => {
            alert("Unable to retrieve data from database!");
        };

        req.onsuccess = (event) => {
            req.result.notes.push({
                title: document.getElementById("title").value,
                content: document.getElementById("content").value,
            });
            req = db
                .transaction("users", "readwrite")
                .objectStore("users")
                .put(req.result);

            document.getElementById("title").value = "";
            document.getElementById("content").value = "";
            alert("Note added!");
        };
    });

    /**
     * add image
     */
    document.getElementById("add-image-btn").addEventListener("click",
() => {
        window.location = "../views/view-images.html";
    });

```

```

    document.getElementById("image")?.addEventListener("change", (event)
=> {
    const reader = new FileReader();
    reader.addEventListener("load", () => {
        document
            .getElementById("create-image-btn")
            ?.addEventListener("click", () => {
                let req = db
                    .transaction("users")
                    .objectStore("users")
                    .get(userInSession);

                req.onerror = (event) => {
                    alert("Unable to retrieve data from database!");
                };

                req.onsuccess = (event) => {
                    req.result.images.push(reader.result);
                    req = db
                        .transaction("users", "readwrite")
                        .objectStore("users")
                        .put(req.result);
                    alert("Image added!");
                };
            });
        reader.readAsDataURL(event.target.files[0]);
    });

    /**
     * add video
     */
    document.getElementById("add-video-btn")?.addEventListener("click",
() => {
        window.location = "../views/view-videos.html";
    });
    document.getElementById("video")?.addEventListener("change", (event)
=> {
    const reader = new FileReader();
    reader.addEventListener("load", () => {
        document
            .getElementById("create-video-btn")
            ?.addEventListener("click", () => {

```

```

        let req = db
            .transaction("users")
            .objectStore("users")
            .get(userInSession);

        req.onerror = (event) => {
            alert("Unable to retrieve data from database!");
        };

        req.onsuccess = (event) => {
            req.result.videos.push(reader.result);
            req = db
                .transaction("users", "readwrite")
                .objectStore("users")
                .put(req.result);
            alert("Video added!");
        };
    });
    reader.readAsDataURL(event.target.files[0]);
});

/**
 * add audio
 */
document.getElementById("add-audio-btn")?.addEventListener("click",
() => {
    window.location = "../views/view-audios.html";
});
document.getElementById("audio")?.addEventListener("change", (event)
=> {
    const reader = new FileReader();
    reader.addEventListener("load", () => {
        document
            .getElementById("create-audio-btn")
            ?.addEventListener("click", () => {
                let req = db
                    .transaction("users")
                    .objectStore("users")
                    .get(userInSession);

                req.onerror = (event) => {
                    alert("Unable to retrieve data from database!");
                };
            });
    });
    reader.readAsDataURL(event.target.files[0]);
});

```



```

    };

    req.onsuccess = (event) => {
        req.result.audios.push(reader.result);
        req = db
            .transaction("users", "readwrite")
            .objectStore("users")
            .put(req.result);
        alert("Audio added!");
    };
});

reader.readAsDataURL(event.target.files[0]);
});

/**
 * delete user
 */
document.getElementById("delete-user")?.addEventListener("click", ()
=> {
    let req = db
        .transaction("users", "readwrite")
        .objectStore("users")
        .delete(userInSession);
    req.onsuccess = () => {
        alert("User deleted!");
        sessionStorage.removeItem("userInSession");
        localStorage.removeItem("userInSession");
        localStorage.setItem("remember", false);
        window.location.reload();
    };
});

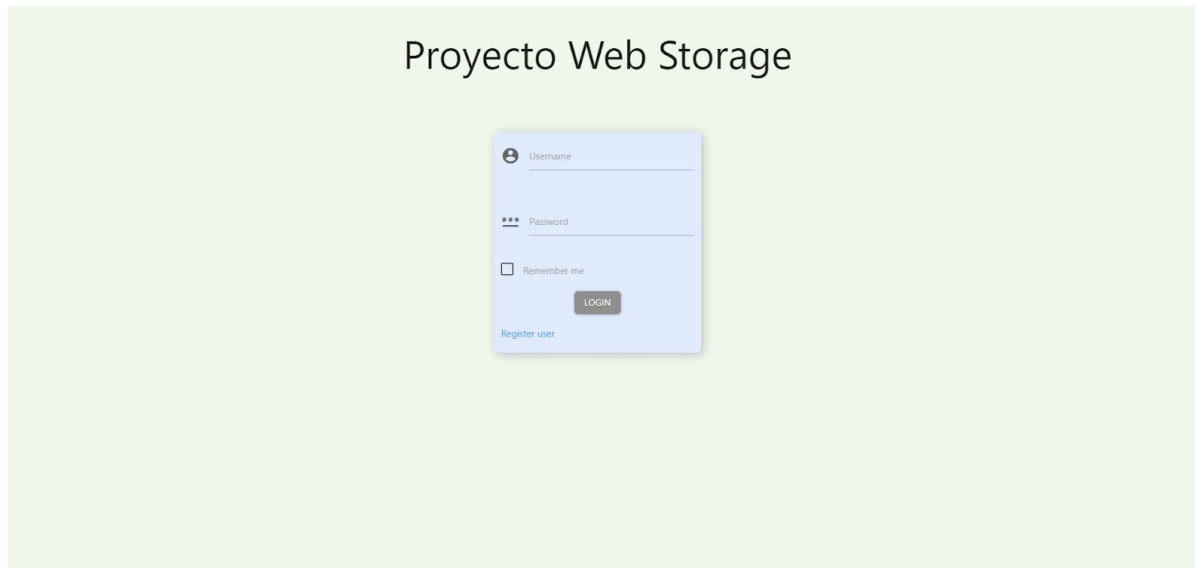
/**
 * logout
 */
document.getElementById("logout")?.addEventListener("click", () => {
    sessionStorage.removeItem("userInSession");
    localStorage.removeItem("userInSession");
    localStorage.setItem("remember", false);
    window.location = "../views/index.html";
});

```

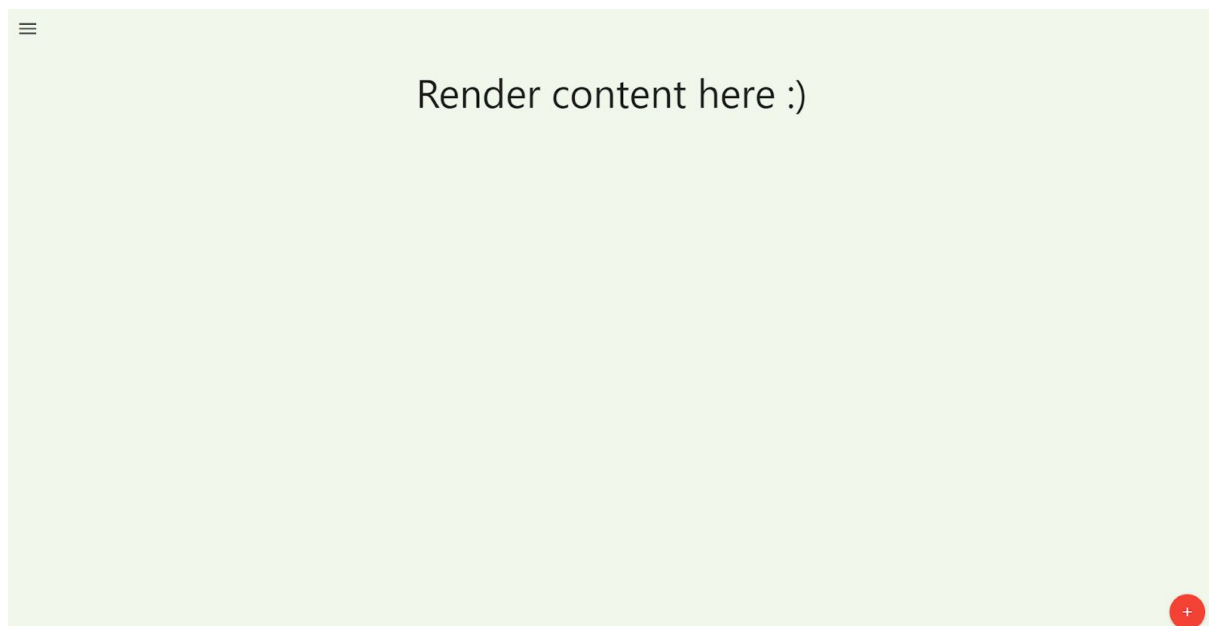
Interfaz de Usuario:

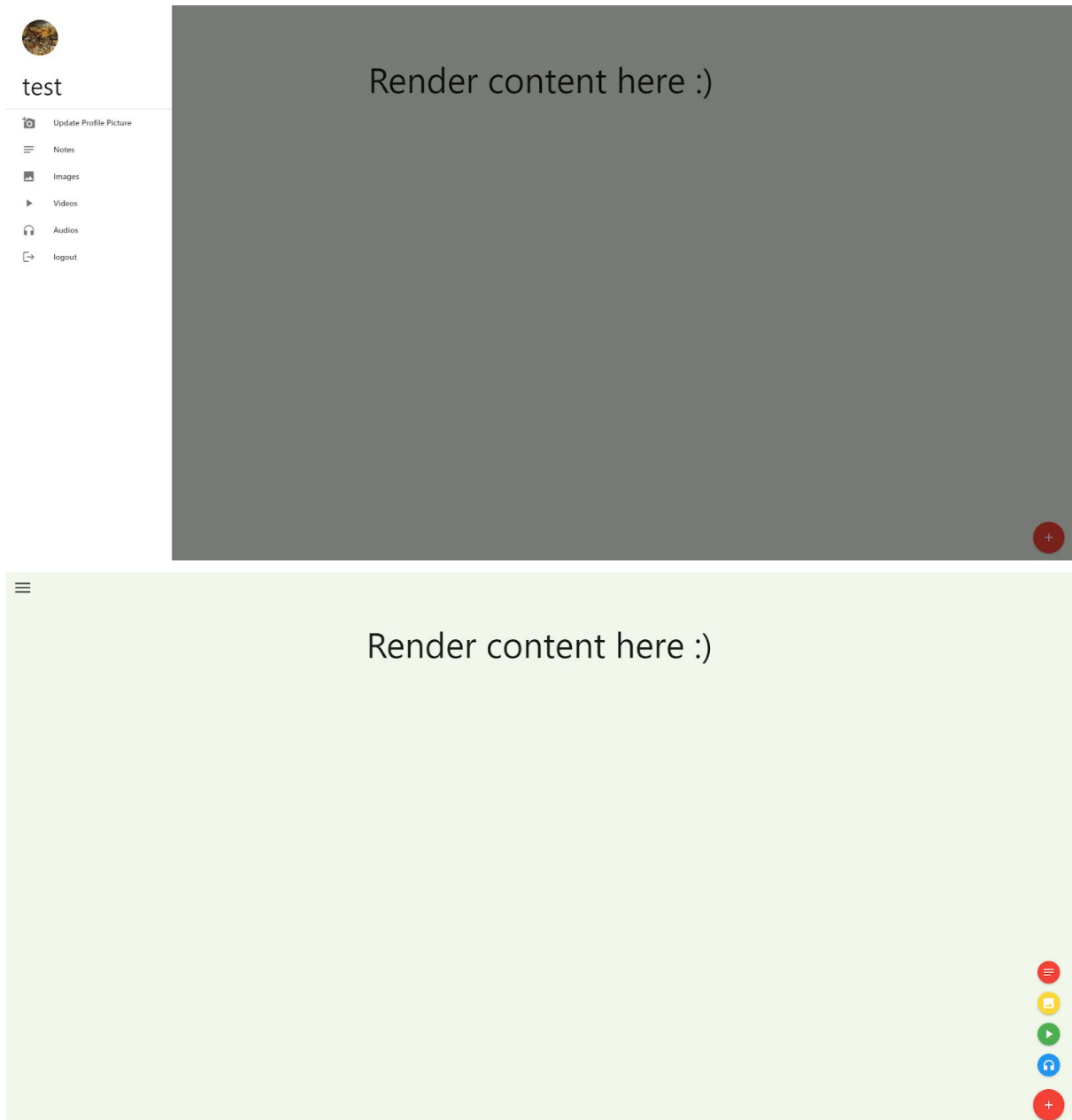
Una vez finalizado el desarrollo se obtienen 7 vistas:

1. index.html
 - a. Landing page para el inicio de sesión de los usuarios



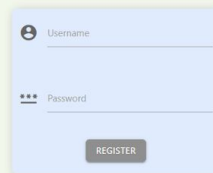
2. view-user.html
 - a. Vista en donde se renderiza el contenido multimedia del usuario, cuenta con un sidebar que en donde se muestra el usuario en cuestión con su foto de perfil personal, además de que se muestran los botones que le permiten al usuario visualizar el contenido que quiera. Por otra parte, en la esquina inferior derecha se muestra un botón flotante con acciones para agregar contenido.





3. view-register.html
 - a. Vista que permite registrar usuarios

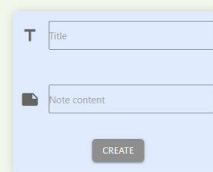
Register



A light blue rectangular form with rounded corners. It contains two input fields: the first is labeled 'Username' with a person icon to its left; the second is labeled 'Password' with three dots to its left. Below these fields is a dark grey button with the word 'REGISTER' in white capital letters.

4. view-notes.html
 - a. Vista para crear notas

Create note



A light blue rectangular form with rounded corners. It contains two input fields: the first is labeled 'Title' with a 'T' icon to its left; the second is labeled 'Note content' with a notepad icon to its left. Below these fields is a dark grey button with the word 'CREATE' in white capital letters.

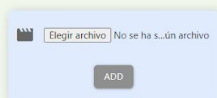
5. view-images.html
 - a. Vista para agregar imagenes

Add image



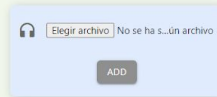
6. view-videos.html
 - a. Vista para agregar videos

Add video



7. view-audios.html
 - a. Vista para agregar audios

Add audio



Estimación

La distribución del tiempo en el proyecto ha sido la siguiente:

- 1) Generar la idea: 2 horas.
- 2) Programar la aplicación: 8 horas.
- 3) Escribir el documento: 2 horas

Conclusiones

Con todo lo anterior se expone que ha sido posible el desarrollo del proyecto de manera óptima y eficaz cumpliendo los objetivos planteados inicialmente.