A+ Fundamentos – App ToDo List

Listastick

Proyecto: Listastick

Estudiante: Bruno Grajales

DOCUMENTO DE INVESTIGACIONES

Register

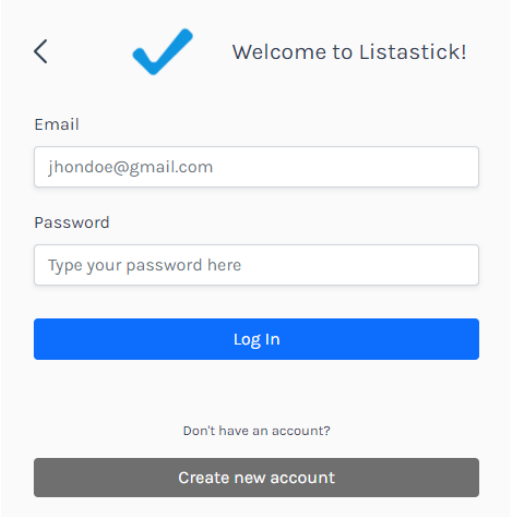
Para el login y el register de la pagina se utilizo localStorage, de manera que se pueda salir y entrar de la pagina y obtener una experiencia mas “real”

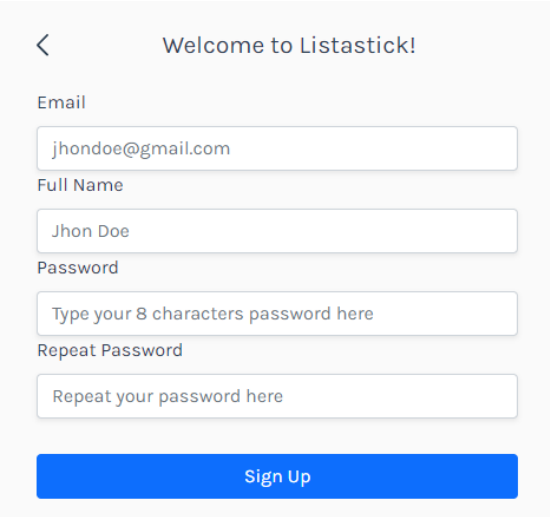
Utilizando las siguientes fuentes:

Link 1: https://www.w3resource.com/javascript/form/javascript-sample-registration-form-validation.php

Link 2: https://medium.com/swlh/how-to-create-your-first-login-page-with-html-css-and-javascript-602dd71144f1

Se crearon dos forms





Cada una con sus propios checks de validez, primero un usuario se registra, utilizando elementos de el Link 1 corroboramos con las funciones check*campo*Validity() que:

- Email sea un string, seguido de un @ seguido de otro string, seguido de “.com”
- Full Name sean dos string separadas por espacio
- Password sea mas larga o igual a 8 caracteres
- Y Repeat Password sea igual a Password

Si todos estos requerimientos se cumplen, se guardara el nuevo usuario en el array userDatabase en localStorage, usuario el cual podrá ser reutilizado para entrar a la web

```
class user {
    constructor(email, fullName, pa
        this.email = email
        this.fullName = fullName
        this.password = password
        this.pfp = pfp
        this.theme = theme
        this.tasks = tasks
        this.listgroup = listgroup
    }
}
```

Esta base de usuarios tiene objetos “user” los cuales tienen Email, Fullname, password (del registro), pfp (default es pfp.png genérico), theme (default), tasks (array de tareas del usuario), listgroup (listas del usuario).

Login

El login funciona de manera similar al register, pero en cambio chequea que el Email corresponda a un usuario dentro del array userDatabase y de ser asi luego pasa a comprobar si la contraseña ingresada por el usuario matchea la contraseña guardada para ese usuario.

isLoggedIn

Luego de que un usuario se registra o se logea en el sitio, este usuario (objeto de la userDatabase) se guarda en la variable del localStorage isLoggedIn, lo cual guarda que usuario en ese dispositivo esta logeado, cual es su tema de preferencia, e idioma.

Al entrar al index, si isLoggedIn tiene algo guardado redirecciona automáticamente hacia la home page.

Y viceversa, si se entra al home directamente e isLoggedIn no posee ningún valor se redirecciona a la landing page.

Calendario

Para la realización del modal de calendario se utilizó la siguiente fuente:

https://medium.com/@nitinpatel_20236/challenge-of-building-a-calendar-with-pure-javascript-a86f1303267d

En la misma se detalla el paso a paso de crear un calendario de manera dinámica para el mes actual.

- Creando variable para el primer día del mes (indica el número de día de la semana en el que el mes arranco) let firstDay = (new Date(year, month)).getDay();
- Utiliza la función getDaysInMonth para devolvernos si es un mes de 30 o 31 días
- Luego con un For anidado, crea 6 rows (tr) para la tabla en la que insertaremos el calendario y para cada row crea 7 td, incluyendo el número de día actualizado con la variable “date”
- En los mismos For, existen variaciones a los de la fuente, ya que se utilizó un array auxiliar “daysWithTasks” el cual nos dice en que días el usuario tiene tareas asignadas, le incluye al elemento td la clase “notif” para identificar ese día con un punto

Alertas

Se utilizo la biblioteca SweetAlert2 https://sweetalert2.github.io/ para el uso de alertas, disparadas a través de JS y hechas también dentro de las funciones con Swal.fire:

```
Swal.fire({
    icon: 'success',
    title: 'Message Sent',
    showConfirmButton: false,
    timer: 2000
  })
```

- Estas alertas se usaron ya que ofrecen una gran variedad de posibilidades y maneras de customizarlas, e incluir mensajes personalizados

- Dándome la posibilidad de hacer en el momento acciones cuando el usuario confirma o niega una acción con la acción .then(result)

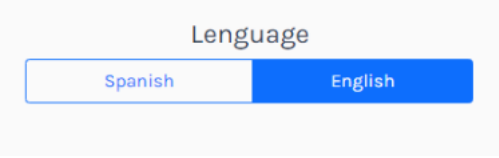
```
Swal.fire({
    title: ((storedLeng == "spanish") ? "Estas seguro que deseas eliminar esta tarea?" : "Are you sure you want to delete this task?"),
    icon: "warning",
    showCancelButton: true,
    confirmButtonText: ((storedLeng == "spanish") ? "Si, eliminar" : "Yes, delete"),
    cancelButtonText: ((storedLeng == "spanish") ? "Cancelar" : "Cancel")
}).then(result => {
    if (result.isConfirmed) {
        for (let index = todosIndex; index < todosArr.length; index++) {

            if (todosArr[index].shouldDisplay == false && todosArr[index-1].shouldDisplay == true) {
                todosArr[index].shouldDisplay = true
                break;
            }

        }

        todosArr.splice(todosIndex, 1)

        renderTodosArr()
    }
})
```

Lenguaje

Se implemento la posibilidad de que el usuario pueda cambiar de idioma la página, luego de buscar se realizo de manera manual una variable “storedLeng” que guarda en la memoria local el lenguaje que el usuario elije, en la carga de cada pagina con el archivo lenguaje.js se chequea en que pagina se esta y si storedLeng es “spanish” o “english”, si es spanish se cambian los elementos que contengan texto a su versión en español y si es english no se cambia nada.

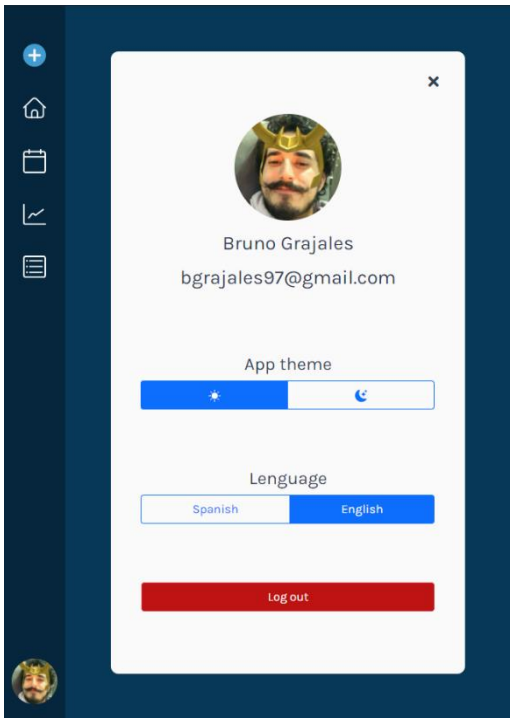
```
document.querySelector("#lenguageBtn").addEventListener("click", function(){
    Swal.fire({
        title: 'Choose your lenguage',
        showDenyButton: true,
        confirmButtonText: `Español`,
        denyButtonText: `English`,
      }).then((result) => {
        if (result.isConfirmed) {···
        } else {
            leng = "english"
            location.reload()
        }

        localStorage.setItem('lenguage', JSON.stringify(leng))

    })
})
```

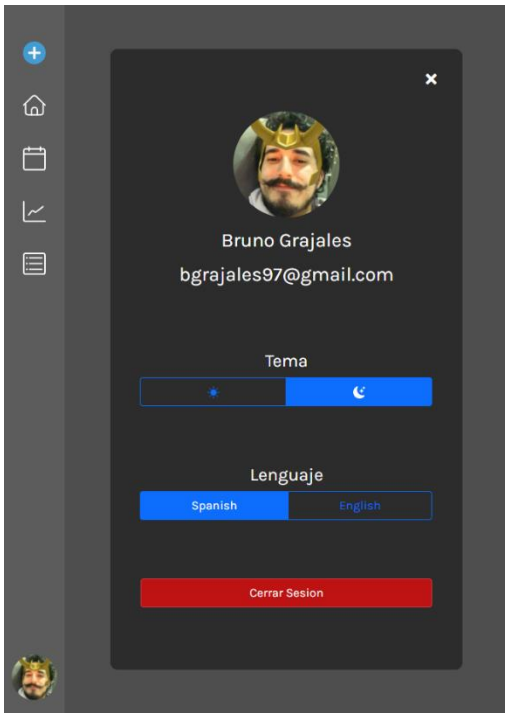
También dando opción a cambiar de idioma en el apratado de perfil guardando la opción preferida en storedLeng



Perfil

Para el modal de perfil se decidió implementar varias opciones para la personalización de la web y que se sienta más propia del usuario.





Lenguaje:

- Como mencionado anteriormente se puede elegir la pagina en español o ingles

Tema:

- Se implemento la opción de utilizar la pagina en tema light o dark utilizando como fuente https://simpleweblearning.com/how-to-create-light-dark-mode-toggle-with-css-and-javascript/
- Donde se declaran 2 sets de variables de colores

```
:root, [data-theme="default"] {
    --primary-color: #1095e0;
    --primary-light-color: #b7dff6;
    --primary-dark-color: #0978d3;

    --accent-color: #ededed;
    --accent-light-color: #ffffff;
    --accent-dark-color: #e5e5e5;

    --theme-text-color: #2E4052;
    --theme-background-color: #fafafa;

    --navBar-color: #05263c;
    --navExtension-color: #083858;

    --card-color: #fff;
    --secondaryBtn-color: #1095e0;

    --sameText-color: #2c2c2c;

    --animate-duration: 300ms;
}
```

```
[data-theme="dark"] {
    --primary-color: #1095e0;
    --primary-light-color: #b7dff6;
    --primary-dark-color: #0978d3;

    --accent-color: #797979;
    --accent-light-color: #d7d7d7;
    --accent-dark-color: #5c5c5c;

    --theme-text-color: #ffffff;
    --theme-background-color: #2c2c2c;

    --navBar-color: #5c5c5c;
    --navExtension-color: #4e4e4e;

    --card-color: #3b3b3b;
    --secondaryBtn-color: #dadada;

    --sameText-color: #2c2c2c;

    --animate-duration: 300ms;
}
```

Luego de ambos “esquemas” declarados se asignan los colores a todos los elementos respectivamente.

Y mediante una función simple themeSelection() se guarda en localStorage lo que el usuario prefiere, si el tema “default” (light) o el tema “dark”

Cuando se inicia la pagina se chequea esta variable guardada para asignarle :root a uno u otro de los data-theme

```
if (isLoggedIn[0].theme == "dark") {
    document.querySelector("#darkCheck").click()
    document.documentElement.setAttribute("data-theme", "dark")
}
```

También dentro del perfil existe la posibilidad de cambiar la foto de perfil utilizando como referencia https://www.youtube.com/watch?v=pmKC5nbiuXo utilizando el objeto fileReader() para seleccionar foto, y luego de que carga cambiar las imágenes que tengan pfp.png como src y también guardar la imagen seleccionada en el objeto usuario en el array usersDatabase.