**Pruebas de Caja Negra Intforlan – Store V1.0**

Freddy Páez B., Dennis Parra G., Jorge Ramos G.

Departamento de Ciencias de la Computación,

Universidad de las Fuerzas Armadas ESPE – Ecuador

Análisis y Diseño de Software – 8311

Ing. Jenny Ruiz R.

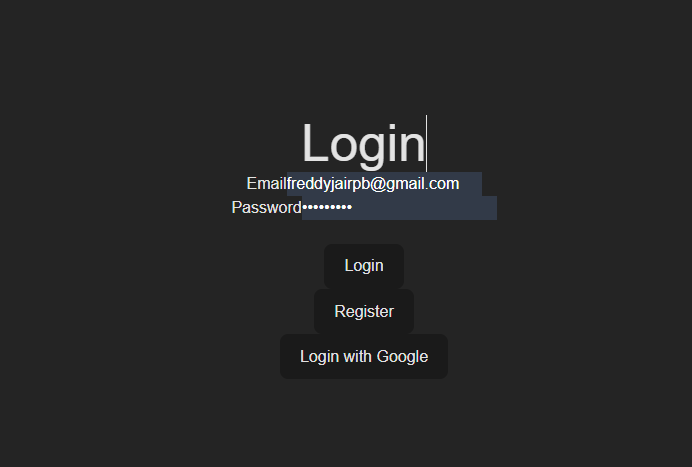
**Sangolquí, 25 de enero de 2023**

**Validación login usuario**

| **VARIABLE** | **CLASE DE EQUIVALENCIA** | **ESTADO** | **REPRESENTANTE** |
| --- | --- | --- | --- |
| **Administrador – Login** | Const login= (email, password) {signInWithEmailAndPassword(auth, email, password; } ; | Válido | Ingresa como administrador |
|  | const login  != (email, password) {signInWithEmailAndPassword(auth, email, password; } ; | No Válido | "Usuario incorrecto, revise los datos" |

**Campos Completos**

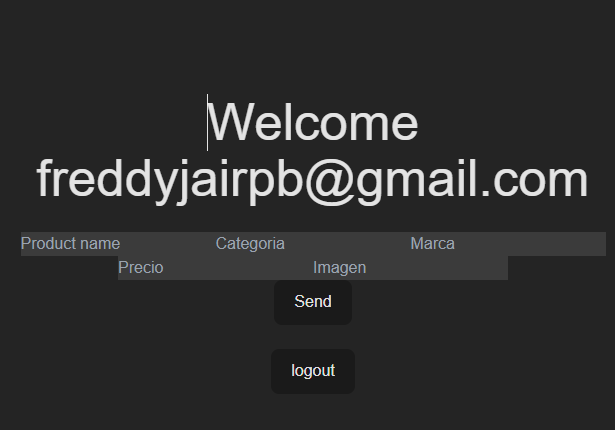
Se ingresa como administrador y si las credenciales ingresadas son correctas, se le permite el acceso al administrador.



En el caso que las credenciales ingresadas por el usuario, sean incorrectas, se emitirá un mensaje en consola de “datos incorrectos” y no se permitirá el acceso a la cuenta.



Cuando las credenciales sean las correctas se le permite al usuario acceder a su cuenta:



**VALIDACIÓN DE REGISTRAR UN NUEVO ADMINISTRADOR**

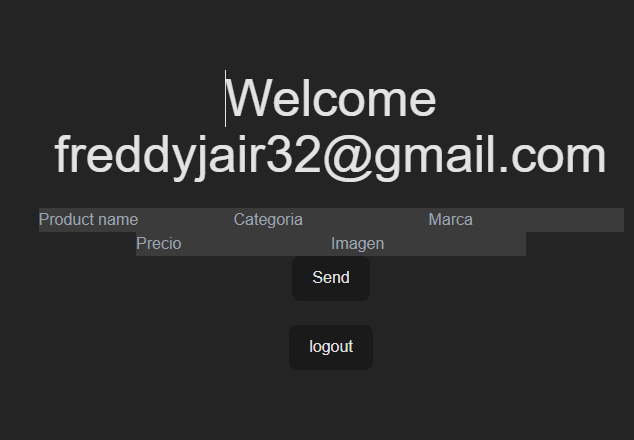
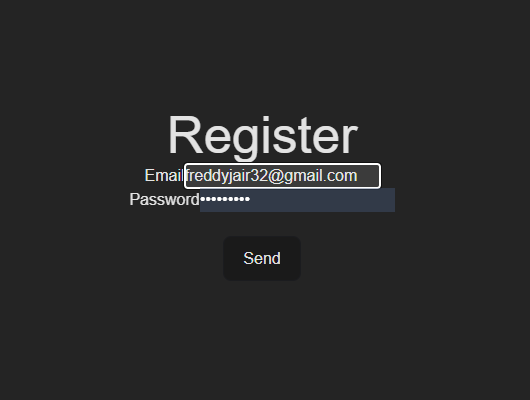
| **VARIABLE** | **CLASE DE EQUIVALENCIA** | **ESTADO** | **REPRESENTANTE** |
| --- | --- | --- | --- |
| **Administrador – Registrar** | EC1: const signUp  = (email, password) {createUserWi thEmai1AndPassword (auth, email, password)); | Válido | "Administrador registrado exitosamente” |
| EC2: const signUp  != (email, password) {createUserWi thEmai1AndPassword (auth, email, password)); | No válido | "auth/email already in use”  No permite el registro de un administrador nuevo con el mismo correo. |
| **Registro** | If(empty(email) | Válido | “Registro con éxito” |
| If(empty(email) | No válido | “ ” |

**Prueba registro nuevo administrador**

El administrador principal, podrá registrar uno nuevo siempre y cuando el correo a registrar no se encuentre previamente creado. En el caso que existiese dicho error, se presentará el mensaje en consola de la siguiente manera:



**Ejemplo de un nuevo administrador:**



Como podemos ver, el registro de un nuevo adminsitrador se cumplió con éxito.

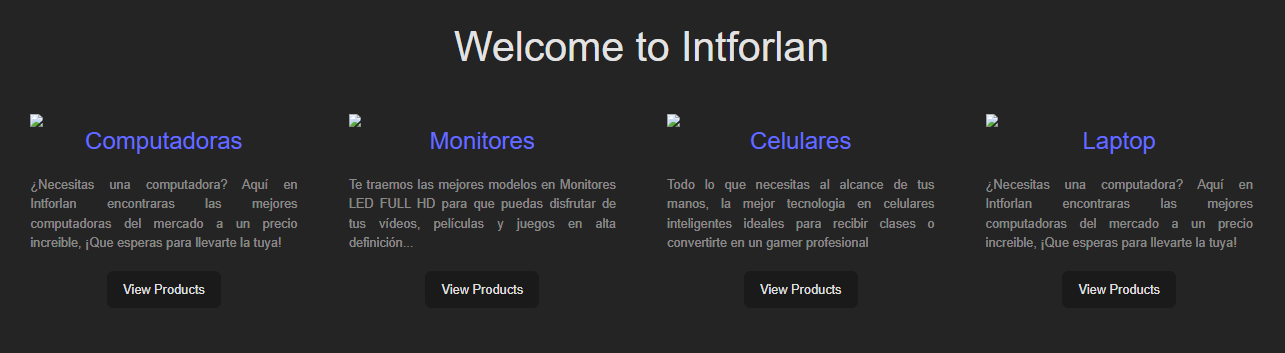
En el caso, que se envíen datos vacíos de igual forma la página web nos arrojará un mensaje en consola:

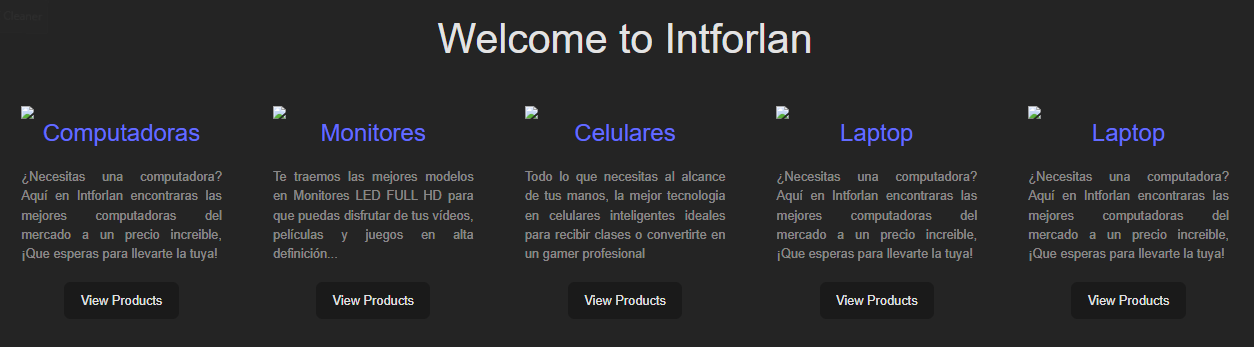


**PRUEBAS DE CAJA NEGA DE PRESENTAR CATEGORIAS**

|  |  |  |  |
| --- | --- | --- | --- |
| **VARIABLE** | **CLASE DE EQUIVALENCIA** | **ESTADO** | **REPRESENTANTE** |
| **Numero de categorías** | EC3: listCategories.map((category) => {//pesentacion de la categoria} | Válido | Ingresa como usuario |

En esta prueba consiste en mostrar cada una de las categorías disponibles dentro de la base de datos, y las características de cada una de estas (id, nombre de la categoría y su descripción)

La aplicación está diseñada para agregar el número de categorías que deseemos



**PRUEBAS DE CAJA NEGRA DE PRESENTAR PRODUCTOS**

|  |  |  |  |
| --- | --- | --- | --- |
| **VARIABLE** | **CLASE DE EQUIVALENCIA** | **ESTADO** | **REPRESENTANTE** |
| **Numero de productos por categoria** | EC3: products.map((product) => {//presentacion de productos} | Válido | Ingresa como usuario |
| **Categorización de los productos según la categoria** | if (product.categoria === categoryProduct) {//presentacion de productos dependiendo de su categoria} | Válido | Ingresa como usuario |

**PRUEBAS DE CAJA NEGRA DE REGISTRAR PRODUCTOS**

|  |  |  |  |
| --- | --- | --- | --- |
| **VARIABLE** | **CLASE DE EQUIVALENCIA** | **ESTADO** | **REPRESENTANTE** |
| **Producto** | EC2: const handleCreate = (e) => {e.preventDefault(); createProduct(); }; | Válido | Ingresa como administrador |
| **Llenar** | EC3: <form onSubmit={handleCreate}> | Válido | Registro del producto |
| **Limpiar** | EC4: <button type="Reset">Clear</button> | Válido | Limpiar registro del producto |

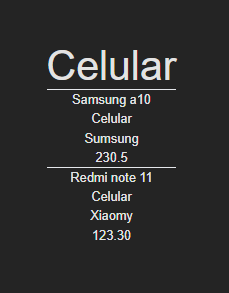
**PRUEBAS DE CAJA NEGRA DE ELIMINAR PRODUCTOS**

|  |  |  |  |
| --- | --- | --- | --- |
| **VARIABLE** | **CLASE DE EQUIVALENCIA** | **ESTADO** | **REPRESENTANTE** |
| **Producto** | EC1: const deleteProduct = async (id) => {path: "products". id} | Válido | Eliminar producto |
| **Llenar** | EC2: confirmDelete ? Await deleteDoc(ProductDoc) : null; }; | Válido | Ver productos |

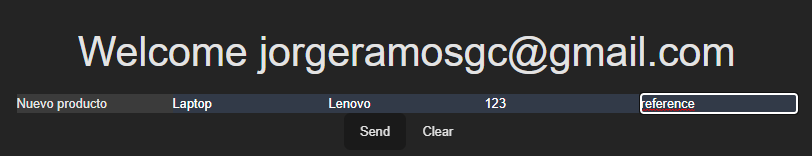
La presentacion de productos tiene que ser solo de la categoría seleccionada

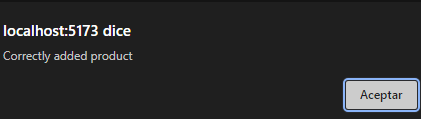


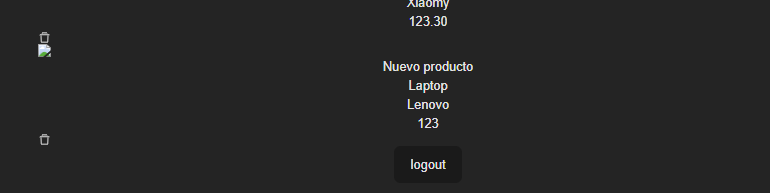
En caso de elegir otra categoría mostrará los productos disponibles de esta



* Se visualiza el registro del nuevo producto.







* Se visualiza la eliminación de un producto ya agregado.

