

Luis Daniel Hormiga Gonzalez - 20172020026

Miguel Angel Gonzalez Rodriguez - 20172020023

Brenda Natalia Portilla Alonso - 20161020030

Plan de pruebas

Objetivos

General:

Diseñar un plan de pruebas que genere una retroalimentación sobre los errores, defectos o fallas que tiene el proyecto a ser presentado, con el objetivo de realizar las correcciones pertinentes, para asegurar el cumplimiento de los alcances del mismo.

Específicos:

- Hacer un correcto levantamiento de requerimientos, identificando la información y los componentes a incluir.
- Hacer uso de Diagramas de Gantt para delimitar tiempos para realizar pruebas y entregas.
- Hacer un software perdurable en el tiempo a base de la retroalimentación dada previamente

Alcance

Módulo	Pruebas	Descripción
GUI	Facilidad de uso	Que el usuario sepa qué quiere hacer y lo pueda hacer en todo momento
	Experiencia de usuario	Estetica del software para generar una experiencia intuitiva
Lógica del negocio	Funcionalidad	El software debe cumplir con todos los requerimientos funcionales
DAO	Persistencia	El sistema debe ser capaz de guardar la información necesaria en una base de datos, para poder tener acceso a ellos sin tener ningún problema de consistencia e integridad.
No Funcionales	No Funcionales	El sistema debe cumplir con los requerimientos no funcionales planteados previamente

Técnicas

Pruebas Visuales:

V1. Comprobar que un usuario no autorizado recibe un mensaje de no autorizado y no puede acceder

V2. Comprobar que las imágenes de las camisetas sean consistentes a la hora de personalizarlas y no generen bugs

V3. Probar que la página sea responsiva

V4. Los objetos y características de las ventanas , menús , tamaño, posición , estado y layout se comportan según los estándares.

Pruebas Funcionales:

Prueba de Acceso al aplicativo web.

Descripción: Probar el acceso de los usuarios a la página web

Objetivo: comprobar que se haga un correcto Log-In

Pruebas a Realizar:

F1. Comprobar que la creación de usuarios sea satisfactoria

F2. Comprobar que un usuario creado previamente no acceda con un password incorrecto

Pruebas Generales:

Descripción: Consulta de aplicaciones

Objetivo: Comprobar la recuperación de información asociada a un registro del catálogo o pedido.

Pruebas a Realizar:

G1. Seleccionando un usuario en la lista , se obtienen todos sus datos

G2. Comprobar que estén todos los datos registrados por el usuario

Cronograma

Cronograma de actividades				
Actividad	Miembro	Tiempo de ejecución	Entrega	Porcentaje completado
V1	Lider de soporte	1 día	Semana 10	100%
V2	Lider de soporte	1 día	Semana 10	0%
V3	Lider de soporte	1 día	Semana 10	100%
V4	Lider de soporte	1 día	Semana 10	100%
F1	Lider de soporte	1 día	Semana 11	100%
F2	Lider de soporte	1 día	Semana 11	100%
G1	Lider de soporte	1 día	Semana 11	100%
G2	Lider de soporte	1 día	Semana 11	100%

Para la actividad V1 se realizó la siguiente prueba (sin interfaz gráfica, sólo comprobando funcionalidad):

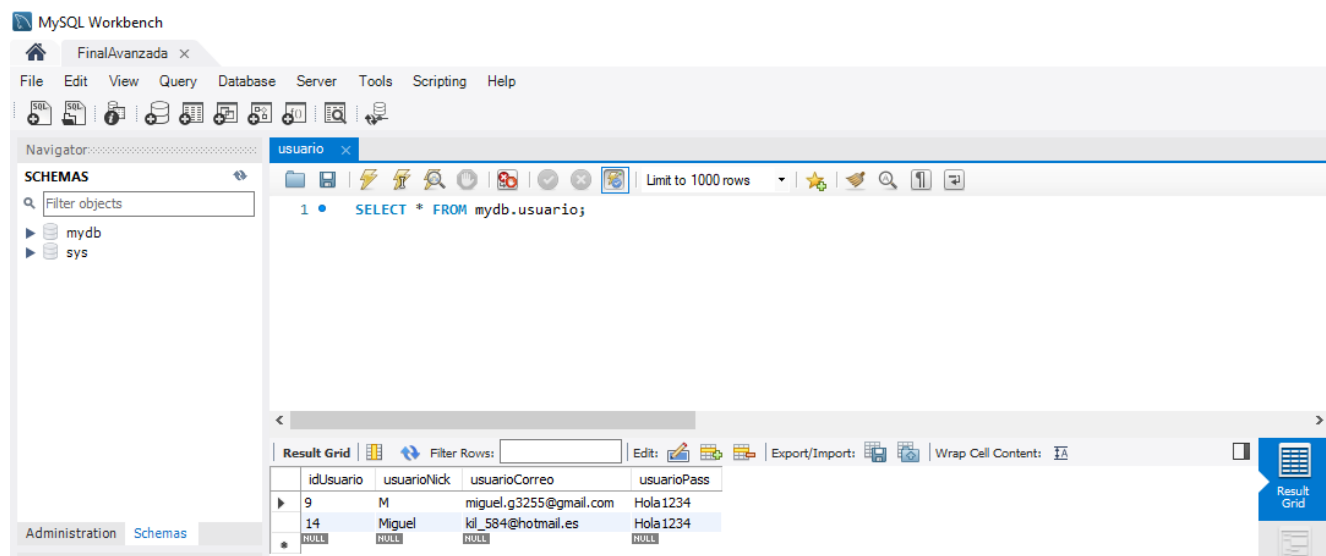
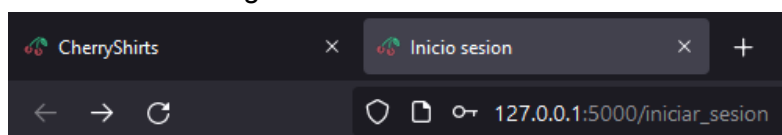


Fig 1. Datos de usuarios en BD

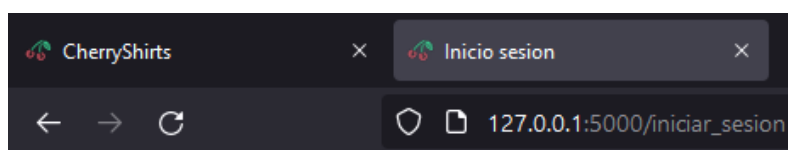
Se intentó acceder con un usuario que no pertenece a la base de datos mostrada anteriormente, el resultado fue el siguiente:



Inicie sesion

¿No se ha registrado? [Registrese](#)

Fig 2. Datos para probar inicio de sesión



correo o contraseña incorrectos, intentelo de nuevo

Inicie sesion

¿No se ha registrado? [Registrese](#)

Fig 3. Resultado obtenido de inicio de sesión malos

Como se puede observar en la figura 3, el acceso fue denegado a quien no tuviese credenciales correctas

La actividad V2 no se pudo realizar ya que la función a la que hace referencia no se alcanzó a implementar en el proyecto.

Para la actividad V3:

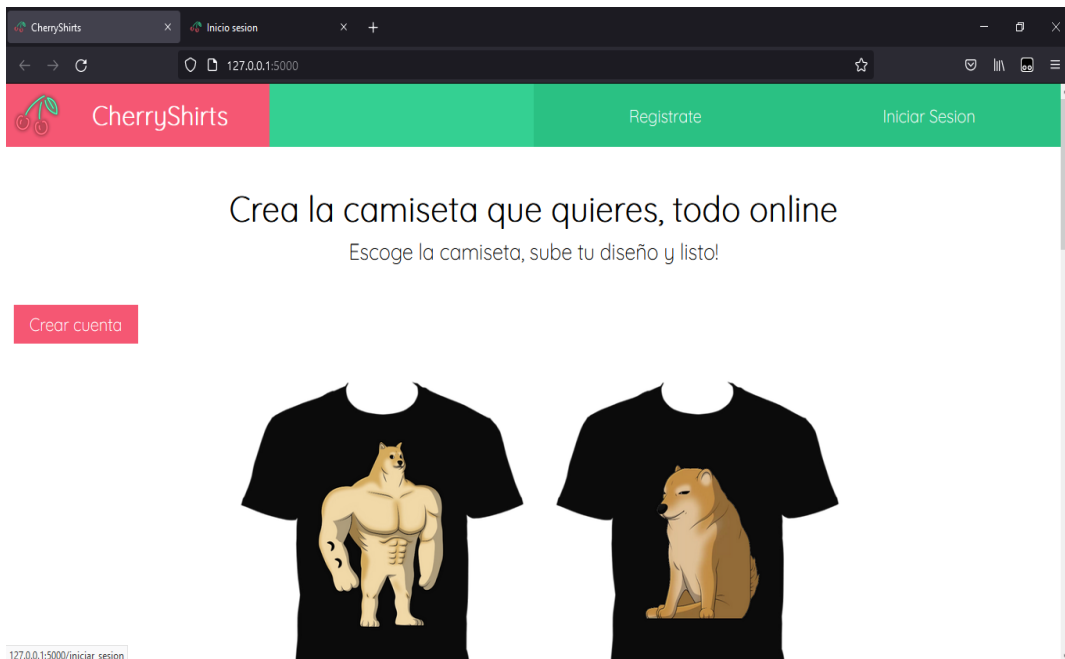


Fig 4. Página principal

En la página principal se probaron los botones a los que se puede acceder que son los de iniciar sesión y registrar usuario. Al pasar el cursor por ellas, se puede notar una respuesta de la página que indica la posibilidad de accionar eventos al pulsarlas.

Para la actividad V4:

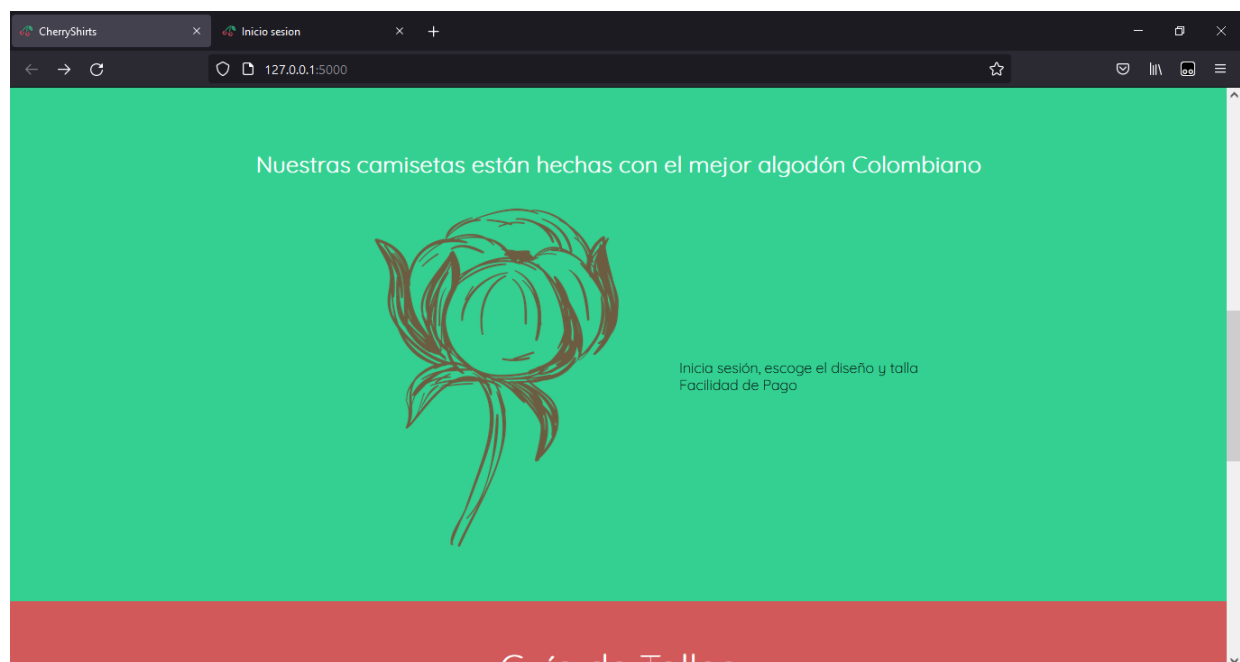


Fig 5. Página principal 2

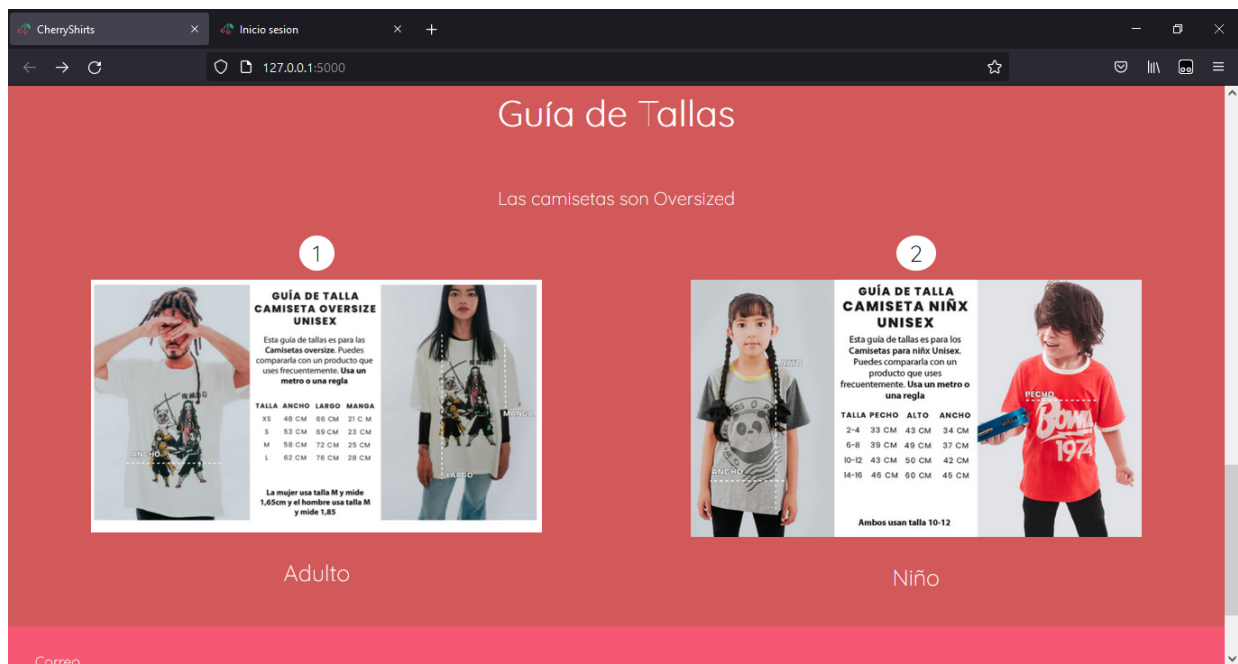


Fig 6. Página principal 3

Con las imágenes anteriores vemos cómo está construída en su totalidad la página, que refleja los estándares de los que se habla

Para la actividad f1 (solo funcionalidad):

¿Quieres registrarte?

Juan

correo@ej.com

•••••

Enviar

¿Ya está registrado? [Inicie Sesión.](#)

Fig 7. Formulario de registro

Se enviaron los datos de la figura anterior hacia la base de datos. al comprobar el registro en la misma se obtuvo lo siguiente:

idUsuario	usuarioNick	usuarioCorreo	usuarioPass
9	M	miguel.g3255@gmail.com	Hola1234
14	Miguel	kl_584@hotmail.es	Hola1234
15	Juan	correo@ej.com	Contraseña1234

Fig 8. Respuesta al registro

Como se puede observar, la inserción del registro fue exitosa

Para la actividad f2:

Como se vió en la actividad V1, no se puede acceder a la aplicación con un correo no registrado. Al intentarlo con el correo que acabamos de registrar pero una contraseña que no coincide con la de la base de datos, el resultado que obtenemos es el mismo de la figura 3.

Para las actividades G1 y G2:

Para esta podemos observar el código de inicio de sesión, en el cual se piden los datos del usuario que va a ingresar al sistema:

```

43 |         return render_template("registro.html", errores =error)
44 |
45 | @app.route("/iniciar_sesion", methods = ["GET", "POST"])
46 | def iniciar_sesion():
47 |     try:
48 |         if request.method=="POST":
49 |             correo=request.form["email"]
50 |             password=request.form["password"]
51 |             user = mod.comprobar_usuario(correo,password)
52 |             if user!=[]:
53 |                 mod.creaCarrito(user[0])
54 |                 session['id_usuario']=user[0]
55 |                 print(session.get('id_usuario'))
56 |                 return redirect("/")
57 |             else:
58 |                 err=["correo o contraseña incorrectos, intentelo de nuevo"]
59 |                 print(err)
60 |                 return render_template("inicioSesion.html", errores=err)
61 |
62 |         return render_template("inicioSesion.html")
63 |     except:
64 |         err=["correo o contraseña incorrectos, intentelo de nuevo"]
65 |         print(err)

```

Fig 9. Código de inicio de sesión

Al iniciar sesión en el sistema ocurre lo siguiente:

```

127.0.0.1 - - [01/Oct/2021 16:12:54] "GET /static/css/normalize.css HTTP/1.1" 304 -
127.0.0.1 - - [01/Oct/2021 16:12:54] "GET /static/css/style-index.css HTTP/1.1" 304 -
127.0.0.1 - - [01/Oct/2021 16:12:54] "GET /static/assets/sises-adult.png HTTP/1.1" 304 -
127.0.0.1 - - [01/Oct/2021 16:12:55] "GET /static/assets/sizes-child.png HTTP/1.1" 304 -
127.0.0.1 - - [01/Oct/2021 16:12:55] "GET /static/assets/facebook.svg HTTP/1.1" 304 -
127.0.0.1 - - [01/Oct/2021 16:12:55] "GET /static/assets/twitter.svg HTTP/1.1" 304 -
127.0.0.1 - - [01/Oct/2021 16:12:55] "GET /static/assets/flor.png HTTP/1.1" 304 -
127.0.0.1 - - [01/Oct/2021 16:12:55] "GET /static/assets/shirts2.png HTTP/1.1" 304 -
127.0.0.1 - - [01/Oct/2021 16:12:55] "GET /static/assets/cherryShirts-logo.png HTTP/1.1" 304 -
127.0.0.1 - - [01/Oct/2021 16:12:56] "GET /iniciar_sesion HTTP/1.1" 200 -
15
127.0.0.1 - - [01/Oct/2021 16:13:05] "POST /iniciar_sesion HTTP/1.1" 302 -
(15, 'Juan', 'correo@ej.com', 'Contraseña1234')
127.0.0.1 - - [01/Oct/2021 16:13:05] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [01/Oct/2021 16:13:05] "GET /static/images/Perro-mamado%201.png HTTP/1.1" 200 -

```

Fig 10. Traer usuario de la base de datos

Como se puede observar, el programa carga satisfactoriamente los datos que ingresamos para realizar el registro en la figura 7.