

Plan de Pruebas del Portafolio Móvil

Nombres: Sarita Marinao, Juan Cornejo

Grupo: 9

Fecha: 21/10/2025

1. Introducción

Este plan tiene como propósito verificar el correcto funcionamiento de las pantallas críticas del proyecto móvil “Portafolio móvil”, desarrollado con Python, Kivy y KivyMD.

Se centra en dos módulos funcionales:

- **LoginScreen** → Validación de credenciales, navegación a pantalla principal.
- **FondoWidget** → Carga, visualización y actualización de datos del usuario.

Las demás pantallas (SobreMi, Habilidades y Proyectos) quedan fuera del alcance al no contar aún con funcionalidades completas.

2. Objetivos de Prueba

Los objetivos de este plan de pruebas son los siguientes:

- **Validar** el correcto acceso del usuario al sistema.
- **Confirmar** la comunicación entre la interfaz de inicio de sesión (LoginScreen), la pantalla principal y la base de datos (gestor.py).
- **Verificar** que los datos del usuario se carguen y actualicen correctamente en la pantalla principal (FondoWidget).
- **Asegurar** que se manejen errores de forma controlada (mensajes y validaciones).
- **Comprobar** que el sistema mantenga su integridad ante entradas inválidas.

3. Plan de Pruebas (Tabla)

ID	Módulo	Descripción	Datos de entrada	Resultado esperado	Tipo de prueba
TST01	LoginScreen	Validar ingreso con credenciales correctas	email="admin@gmail.com", clave="1234"	Navega a pantalla "home"	Funcional
TST02	LoginScreen	Validar error al ingresar credenciales incorrectas	email="fake@mail.com", clave="0000"	Muestra diálogo "Correo o contraseña incorrectos."	Funcional
TST03	LoginScreen	Validar error por campos vacíos	email="", clave=""	Muestra mensaje "Por favor ingrese correo y contraseña."	Unitaria
TST04	FondoWidget	Verificar carga correcta del usuario desde BD	N/A (usuario por defecto en BD)	Propiedades nombre, profesión e intro cargadas correctamente	Integración
TST05	FondoWidget	Actualizar nombre de usuario	nuevo_nombre="Juan Pérez"	Actualiza nombre en UI y en la BD	Funcional
TST06	FondoWidget	Actualizar profesión del usuario	nueva_profesion="Ingeniero de Software"	Actualiza nombre en UI y en la BD	Funcional
TST07	FondoWidget	Actualizar mensaje de bienvenida	nueva_intro="Bienvenido a mi portafolio profesional"	Texto reflejado y guardado en BD	Funcional
TST08	FondoWidget	Intentar actualizar con texto vacío	campo vacío	No se modifica el valor anterior	Unitaria

4. Ejecución de pruebas

Se ejecutaron un total de 8 pruebas correspondientes a los módulos “FondoWidget” (pantalla de inicio) y “Login”. La ejecución detallada se realizó con el comando **pytest -v**

Los resultados de dicho test fueron los siguientes:

```
===== test session starts =====
platform win32 -- Python 3.13.4, pytest-8.4.2, pluggy-1.6.0 -- C:\Users\juanc\AppData\Local\Programs\Python\Python313\python.exe
cachedir: .pytest_cache
rootdir: C:\Users\juanc\OneDrive\Documentos\Kivy
collected 8 items

Portafolio/tests/test_fondowidget.py::test_carga_usuario PASSED [ 12%]
Portafolio/tests/test_fondowidget.py::test_actualizar_nombre PASSED [ 25%]
Portafolio/tests/test_fondowidget.py::test_actualizar_profesion PASSED [ 37%]
Portafolio/tests/test_fondowidget.py::test_actualizar_intro PASSED [ 50%]
Portafolio/tests/test_fondowidget.py::test_no_actualiza_intro_vacia PASSED [ 62%]
Portafolio/tests/test_login.py::test_login_valido PASSED [ 75%]
Portafolio/tests/test_login.py::test_login_invalido PASSED [ 87%]
Portafolio/tests/test_login.py::test_login_campos_vacios PASSED [100%]

===== 8 passed in 0.94s =====
```

Interpretación: Todos los tests pasaron correctamente, indicando que las funcionalidades evaluadas se comportan según lo esperado.

4.1 Análisis por módulo

tests/test_login.py

Test	Resultado	Observaciones
test_login_valido	PASSED	Se permite el acceso con credenciales correctas.
test_login_invalido	PASSED	El sistema rechaza credenciales incorrectas.
test_login_campos_vacios	PASSED	Se valida correctamente la ausencia de datos, evitando errores.

Reflexión: El módulo de login cubre correctamente la autenticación y las validaciones de seguridad básicas.

tests/test_fondowidget.py

Test	Resultado	Observaciones
test_carga_usuario	PASSED	La función de carga de usuario se ejecuta correctamente.
test_actualizar_nombre	PASSED	La actualización del nombre funciona como se espera.
test_actualizar_profesion	PASSED	La actualización de profesión es exitosa.
test_actualizar_intro	PASSED	La introducción del usuario se actualiza correctamente.
test_no_actualiza_intro_vacia	PASSED	Se evita actualizar la introducción si el contenido está vacío.

Reflexión: El módulo “FondoWidget” (pantalla de inicio) muestra un comportamiento estable y seguro, manejando tanto los casos normales como los casos límite.

4.2 Análisis general y reflexivo

- **Cobertura funcional:** Aunque todos los tests pasaron aún no se cuenta con métricas de cobertura de código, Incorporar **pytest-cov** podría identificar rutas de código no evaluadas.
- **Robustez:** Los tests cubren casos normales y algunos límites (campos vacíos, introducción vacía), mostrando que el sistema maneja entradas válidas e inválidas correctamente.
- **Eficiencia:** La ejecución fue rápida (0.94 s), lo que indica que los tests son eficientes y no dependen de operaciones costosas.
- **Mantenibilidad:** La separación clara por módulos facilita localizar posibles fallos y agregar nuevas pruebas sin afectar la estructura existente.
- **Recomendaciones:** Ampliar pruebas para escenarios más complejos (múltiples usuarios, concurrencia) y evaluar excepciones inesperadas y rendimiento si el proyecto escala.

5. Inteligencia artificial utilizada

La inteligencia artificial que se utilizó para desarrollar este plan de pruebas fue ChatGPT.

Uno de los prompts utilizados decía lo siguiente: *“Necesito implementar un plan de pruebas para un proyecto de un portafolio móvil desarrollado con el framework de kivy kivymd, el orden de las carpetas dentro del proyecto es Kivy/Portafolio y dentro de portafolio estan las carpetas app y tests, dentro de la carpeta app hay 3 archivos, main.py, portafolio.kv y gestor.py y dentro de la carpeta tests están los 2 archivos de test_login.py y test_fondowidget.py, necesito que realices un plan de pruebas solamente para la funcionalidad de login y de las funcionalidades de la pantalla de inicio ya que las demás pantallas aun no tienen las funcionalidades listas, hago envío de los 2 archivos main.py y gestor.py para que tengas contexto de como hacer los tests (se envia el codigo de los 2 archivos).”*

Debido a este prompt se pudo tener una base clara de cómo aplicar las pruebas al proyecto, hubieron unos errores pero que gracias a la inteligencia artificial se pudieron solucionar.

Conclusión

El proceso de testing realizado permitió confirmar que las funcionalidades principales del portafolio móvil operan de manera correcta, garantizando la estabilidad, la seguridad y la confiabilidad del sistema, Este trabajo demuestra la importancia del testing dentro del ciclo de desarrollo de software ya que permite detectar errores tempranamente, mejorar la calidad del producto y asegurar una buena experiencia de usuario.

Asimismo las pruebas constituyen una base sólida para futuras iteraciones del proyecto, Como mejora futura se propone ampliar la cobertura de pruebas a todos los módulos pendientes, incorporar métricas automáticas de cobertura con herramientas como *pytest-cov*, y realizar pruebas de rendimiento y estrés para evaluar el comportamiento del sistema ante cargas mayores o escenarios concurrentes, Con estos pasos el sistema podrá evolucionar hacia un producto más robusto, escalable y mantenible.