

Documento de pruebas

Proyecto - QuebraGalho
Framework - Flutter
Herramientas de testeo - flutter_test , mockito, build_runner

AgendamentoPageService

Objetivo

Verificar que se puede consultar la lista de horarios no disponibles correctamente desde la API.

Archivo de test

test/agendamento/agendamento_page_service_test.dart

Casos de prueba

ID	Descripción	Entrada	Resultado esperado	Resultado obtenido
AG01	Retorna lista de horarios si Status 200	servicioid = 1	Devuelve lista de DateTime	Correcto
AG02	Lanza excepción si status != 200	servicioid = 1	Lanza Exception	Correcto

Clase: CadastroPage

Objetivo

Validar el comportamiento del formulario de registro de clientes/prestadores.

Archivos de test

test/cadastro/cadastro_page_test.dart

test/cadastro/cadastro_page_submit_test.dart

Casos de prueba

ID	Descripción	Entrada	Resultado esperado	Resultado obtenido
CD01	Renderiza tabs Cliente/Prestador		Se visualizan ambos tabs	Correcto
CD02	Formulario invalido no envia	Campos vacíos	Se muestra error de validación	Correcto
CD03	Envio exitoso con mock HTTP	Datos válidos y mock 201	Se muestra Snackbar o éxito	Correcto

Clase: PerfilPageService
Objetivo
Validar funciones del perfil: buscar, actualizar, eliminar imagen.

Archivo de test
test/perfil/perfil_page_service_test.dart

Casos de prueba

ID	Descripción	Entrada	Resultado esperado	Resultado obtenido
PF01	Buscar perfil con status 200	usuariold = 1	Devuelve Map con perfil	Correcto
PF02	Remover imagen con status 204	usuariold = 1	Retorna True	Correcto
PF03	Actualizar perfil con status 200	Datos actualizados	Devuelve nuevo Map	Correcto

Clase: LoginPage
Objetivo
Verificar que la pantalla de login funcione correctamente y responda al formulario.

Archivo de test
test/login/login_page_test.dart

Casos de prueba

ID	Descripción	Entrada	Resultado esperado	Resultado obtenido
LG01	Renderiza campos con email y contraseña		Se encuentra los 2 campos y botón	Correcto
LG02	Validación con campos vacíos	Presionar con campos vacíos	No debe avanzar ni romper	Correcto

Clase: TelaConfirmacaoSolicitacao
Objetivo
Verificar que la pantalla de confirmación funcione correctamente y ejecute la acción esperada.

Archivo de test

test/confirmacao/tela_confirmacao_solicitacao_test.dart

Casos de prueba

ID	Descripción	Entrada	Resultado esperado	Resultado obtenido
TC01	Renderizado correcto		Se muestra texto "Confirmar" y botón	Correcto
TC02	Accion de confirmar	Presionar botón	se ejecuta onConfirmar()	Correcto

Tests de Usabilidad

Criar Cliente / Prestador:

El app abre en la pantalla inicial y cumple con el primer requisito funcional que es poder hacer un registro ya sea como cliente o prestador - Correcto

Al hacer Login:

El app pregunta si quieres ingresar como Cliente o Prestador, dependiendo cual opción elijas te llevará al apartado de Cliente o Prestador - Correcto

Al elegir la opción Cliente:

Se mostrarán todos los Prestadores registrados, con las tags pertenecientes a cada uno y su evaluación general. - Correcto

Al elegir un TAG o buscar una TAG:

Te mostrará todos los Prestadores con trabajos relacionados a la TAG que ingresaste o buscaste. -Correcto

Al elegir un Prestador:

Te mostrara los servicios que ese prestador ofrece y también habrá un boton de denuncias.- Correcto

Al elegir un Servicio:

se abrirá el calendario, al elegir el dia se mostrará los horarios que el prestador puede realizar el servicio - Correcto

Al Confirmar el Servicio:

Se enviará una solicitud al prestador.- Correcto

Al Hacer denuncia:

Se dejará elegir la opción sobre el motivo de la denuncia y una descripción sobre. Dicha denuncia será analizada por un moderador.- Correcto

Al ingresar como Prestador:

En la Home aparecerán todos los servicios que te solicitaron y un botón para aceptar y otro declinar. - Correcto

Luego de ser aceptado el servicio se habilita el chat entre el cliente y el prestador.- Correcto

Editar datos:

Tanto el cliente como el prestador puede editar sus datos, a diferencia que el prestador tendrá algunas opciones más como por ejemplo editar el horario en que ofrece los servicios.- Correcto

Agregar TAGS

El prestador puede añadir TAGS al momento de crear un servicio, ya sea porque no existe esa TAG o porque no se encaja en ninguna de las existentes.- Correcto

Agregar Servicio

El prestador puede agregar servicios que ofrece, elegir el tiempo que aproximadamente toma y también el valor y descripción del servicio.- Correcto

Apelar denuncias

El prestador puede ser denunciado por el cliente, también se puede abrir un apelo ya sea porque la denuncia fue injusta o un malentendido.- Correcto

Parte moderador:

Aceptar Prestadores

El moderador puede aceptar los prestadores luego de que se registran, aparecerá los datos del prestador y su documento.- Correcto

Ver denuncias

El moderador puede ver todas las denuncias que fueron hechas, en la lista de denuncias se puede ingresar para ver los detalles, ya sea el motivo o id del usuario denunciado, así como otros detalles. - Correcto

Ver apelos

El moderador también puede ver los apelos que hicieron los prestadores luego de ser denunciados, ya sea porque fue un malentendido o una denuncia injusta.

Después de eso el moderador podrá aceptar el apelo o no. - Correcto

Hacer de perfiles otros moderadores

El moderador puede dar acceso a las funciones de moderador a otros usuarios. - Correcto