

# Práctica 3: Pruebas de desarrolladores

**Grupo:** GrupoPequeñoPrácticasDS2\_2

**Alumnos:** Raúl Rodríguez Pérez y Raúl Castro Moreno

Esta práctica consiste en la creación de diferentes pruebas o tests para evaluar nuestro trabajo de la práctica 2. Dichos test van a ser de gran utilidad para detectar errores que no hemos apreciado durante el desarrollo de la práctica anterior. Durante la creación de las pruebas podría ser necesario realizar pequeños cambios en el diseño de las clases del proyecto original, los cuales, serán registrados en este informe.

## 1. Pruebas de unidad

Las pruebas unitarias son útiles para verificar el comportamiento de una única función, método o clase. El paquete de prueba proporciona el marco principal para escribir pruebas unitarias y el paquete flutter test proporciona utilidades adicionales para probar. En primer lugar, comprobamos la funcionalidad de la clase FiltroIVA. Nuestro objetivo es comprobar que el valor del precio (que es la variable que modifica la clase), debe estar inicializada a 5.0 euros. Y por otro lado, también queremos comprobar la operación de aplicarle el IVA al precio, funcione correctamente.

Para la realización de este test, creamos la clase 'FiltroIVA\_test.dart', realizamos los imports necesarios de 'flutter\_test' y simplemente creamos el main con los diferentes test, y a su vez, las diferentes llamadas al método expect(), el cual, es el encargado de comprobar la similitud o diferencia de los argumentos que le pases.

Las capturas de la ejecución de este test son las siguientes:

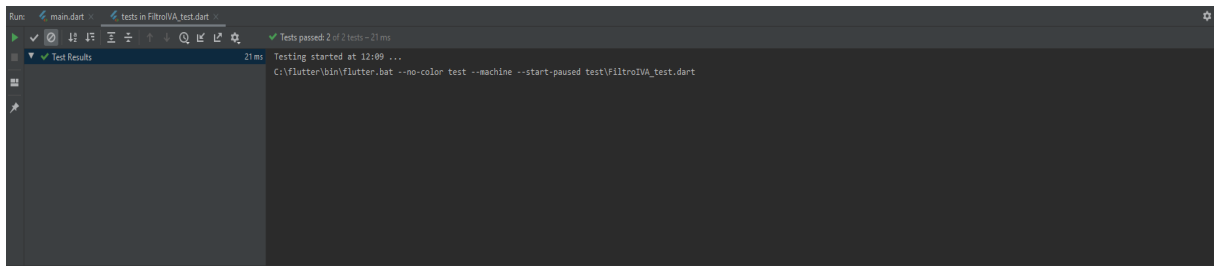
tests in FiltroIVA\_test.dart: 2 total, 2 passed

FiltroIVA\_test.dart

■ FiltroIVA

Value should start at 5.0

Must be multiplied by 0.21



En segundo lugar de las pruebas de unidad, vamos a hacer la comprobación del correcto funcionamiento de la clase FiltroPersonas. En el test comprobaremos en primer lugar, que el número de personas está inicializada a 0.0, y en segundo lugar, comprobar que sí el número de personas es de 10,11 o 12, se aplica perfectamente el descuento.

Las capturas de la ejecución de este test son las siguientes:

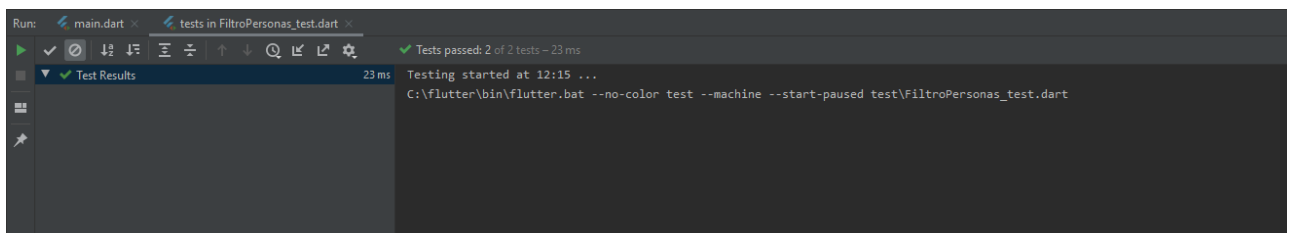
tests in FiltroPersonas\_test.dart: 2 total, 2 passed

FiltroPersonas\_test.dart

FiltroPersonas

Value should start at 0.0

If n\_personas = 10, 11 or 12; discount must be applied



Por último en estos test de unidad, vamos a comprobar que los métodos de calcular el coste y el tiempo medio de la clase objetivo funcionen correctamente. Para ello, inicializamos una serie de alquileres con un tiempo y un coste conocido. Y comparamos el resultado de los cálculos con el valor que devuelven las funciones a testear.

Las capturas de la ejecución de este test son las siguientes:

tests in Objetivo\_test.dart: 2 total, 2 passed

22 ms

[Collapse](#) | [Expand](#)

Objetivo\_test.dart

FiltroIVA

Average cost should be the sum of the costs between the number of rentals

passed

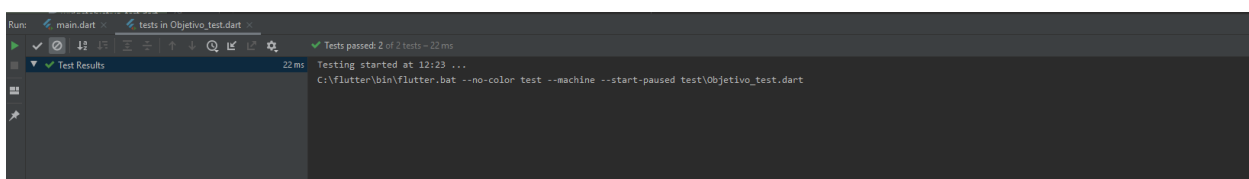
20 ms

Average time should be the sum of the times between the number of rentals

passed

2 ms

Generated by Android Studio on 30/04/21 12:23



## 2. Pruebas de Widget

Las pruebas de Widget nos permiten construir e interactuar con widgets en un entorno de prueba. Esto con la ayuda de funciones como `testWidgets()`, que crea automáticamente un nuevo `WidgetTester` para cada caso de prueba, las funciones de `Finder`, que permiten buscar widgets en el entorno de prueba, y muchas más, nos permiten comprobar que la construcción y el funcionamiento de los widget que hemos creado, sea correcto.

El primer test de widgets que hemos hecho es 'WidgetPaypal\_test'. En este nuevo archivo agrupamos dos test, el primero de ellos comprueba que todos los textos que hemos puesto en el widget, sean los correctos. Y el segundo, comprueba el correcto funcionamiento de los texfield que hemos añadido al widget. Para esto último, lo que hacemos es que primero añadimos texto al texfield, recargamos el widget y comprobamos que lo que hemos escrito anteriormente sigue en el campo del texfield. Estas dos comprobaciones las hemos realizado también para el método de pago con tarjeta, y para nuestro widget objetivo.

Las capturas de la ejecución de 'WidgetPaypal\_test' son las siguientes:

**Comprobar widget pago Paypal: 2 total, 2 passed**691 ms

[Collapse](#) | [Expand](#)

widgetpaypal\_test.dart739 ms

Comprobar widget pago Paypal691 ms

Comprobar text en pago Paypal493 ms

Comprobar textfield de pago Paypal198 ms

Precio del pago: 5.00

Run: main.dart | tests in widgetpaypal\_test.dart

Tests passed: 2 of 2 tests - 739 ms

Testing started at 12:45 ...

C:\flutter\bin\flutter.bat --no-color test --machine --start-paused test/widgetpaypal\_test.dart

Precio del pago: 5.00

Las capturas de la ejecución de 'WidgetTarjeta\_test' son las siguientes:

**tests in widgettarjeta\_test.dart: 2 total, 2 passed**732 ms

[Collapse](#) | [Expand](#)

widgettarjeta\_test.dart732 ms

Comprobar método de pago tarjeta732 ms

Comprobar text en pago tarjeta506 ms

Comprobar textfield de pago Tarjeta226 ms

Precio del pago: 5.00

Run: main.dart | tests in widgettarjeta\_test.dart

Tests passed: 2 of 2 tests - 725 ms

Testing started at 12:49 ...

C:\flutter\bin\flutter.bat --no-color test --machine --start-paused test/widgettarjeta\_test.dart

Precio del pago: 5.00

Las capturas de la ejecución de 'WidgetTarjeta\_test' son las siguientes:

tests in widgetobjetivo_test.dart: 2 total, 2 passed		
widgetobjetivo_test.dart		757 ms
Comprobar objetivo		757 ms
Comprobar text en objetivo	passed	568 ms
Comprobar textfield de objetivo	passed	189 ms

### 3. Prueba de integración

La prueba de integración consiste en realizar un test simulando una interacción normal de la aplicación que hemos creado. Es decir, en nuestro caso, una interacción en la que un cliente entra en la app, rellena los datos del alquiler, y realiza el pago. Siguiendo los pasos de la página oficial de flutter, hemos realizado la creación de dos carpetas las cuales, son las encargadas de ejecutar la prueba de integración 'test\_driver' y 'integration\_test'. Además, tal y como aparece en la página oficial hemos ejecutado la prueba de integración desde la terminal. Apareciendo en la pantalla del móvil que va realizando todos los pasos que le hemos indicado en el test. Y superando el test sin problemas.

