

# Documentación de la Aplicación Móvil



## UNIVERSIDAD DE GRANADA

Álvaro Vega Romero  
Esther García Gallego  
Sandra Reinoso Ortega  
Alexander Collado Rojas  
Javier Osakar Lozano

## Análisis, Diseño y Resultado del Sistema Móvil

---

### Parte del Análisis

#### 1. Requisitos Funcionales

Los requisitos funcionales del sistema serán los siguientes:

Gestión de Usuarios:

- Dar de alta a usuarios
- Iniciar sesión como usuario
- Cerrar sesión
- Ver datos de usuario

Gestión de Productos:

- Consultar catálogo de productos
- Filtrar catálogo de productos por precio, distancia, tipo y estado de un producto
- Realizar compra de un producto
- Ver datos (detalles) de producto

Detalle	Descripción
RF #	1
Nombre	Dar alta usuarios
Descripción	Permite el alta de un usuario en el sistema
Entrada	Datos del cliente (Nombre, apellidos, email, teléfono, username, contraseña)
Procesamiento	Prerrequisito: acceso no identificado] Registro de la cuenta en la BD externo al sistema Postrequisito: acceso identificado.
Salida	Acceder a la aplicación

Detalle	Descripción
RF #	2
Nombre	Consultar catálogo de productos
Descripción	Permite consultar los productos disponibles
Entrada	-
Procesamiento	Consultar los productos que existen en un catálogo, estos guardados en una BD externa al sistema
Salida	Productos con sus atributos

Detalle	Descripción
RF #	3
Nombre	Filtrar productos por sus características
Descripción	Dentro de un catálogo el usuario podrá filtrar productos por precio, distancia, tipo o estado del producto.
Entrada	Filtro que se desee aplicar
Procesamiento	El usuario usará los filtros que quiera aplicar y se filtrarán los productos, enviando solo los que cumplan con las características deseadas.
Salida	Catálogo con los productos filtrados

Detalle	Descripción
RF #	4
Nombre	Realizar Compra
Descripción	El usuario presionará la tecla de compra y se añadirá este producto a sus pedidos. Para un caso inicial solo con presionar el botón se compra.
Entrada	Producto que se desea comprar
Procesamiento	El producto existente en la BD externa se eliminará de esta y se añadirá a los pedidos de un usuario.
Salida	Confirmación de la compra

Detalle	Descripción
RF #	5
Nombre	Iniciar sesión como Usuario
Descripción	Permite a un usuario iniciar sesión en la aplicación con su usuario y contraseña
Entrada	Nombre de Usuario y Contraseña
Procesamiento	Prerrequisito: El usuario debe estar identificado en el sistema. Se comprobará que exista un usuario relacionado con esa contraseña
Salida	Acceso a la aplicación

Detalle	Descripción
RF#	6
Nombre	Cerrar sesión
Descripción	Dado un usuario loggeado, permitir que salga de su sesión.
Entrada	-
Procesamiento	Prerequisito: estamos loggeados Se libera la variable asociada al usuario actual y se sale a la sección de log-in
Salida	Salir de la página principal y eliminar la variables asociada al usuario

Detalle	Descripción
RF#	7
Nombre	Ver datos de un usuario
Descripción	Dado un usuario loggeado, acceder a una página que permita ver sus datos.
Entrada	-
Procesamiento	Prerequisito: estamos loggeados Se accede a la página asociada a ver datos de usuario, pasándole como parámetro el usuario loggeado
Salida	Salir de la página principal y avanzar a la página de datos

Detalle	Descripción
RF#	8
Nombre	Ver detalles de producto.
Descripción	Dada la lista de productos, se puede hacer click al botón de “ver más” de un producto, y ver los detalles como descripción y más.
Entrada	-
Procesamiento	Se accede a la página de producto, pasándole el id del producto sobre el que se hizo click.
Salida	Salir de la página principal y avanzar a la página de detalles del producto

## 2. Requisitos No Funcionales

Los requisitos no funcionales del sistema serán:

- **Facilidad de uso:**
  - El sistema tendrá opciones fáciles, intuitivas y concisas.
- **Rendimiento:**
  - Corto tiempo de respuesta a los accesos a la base de datos
- **Implementación:**
  - Interfaz interactiva
  - Desarrollado en Dart y Flutter
- **Evolutivo:**
  - El sistema tendrá la capacidad de ampliarse más y añadir más requisitos funcionales según avance.
- **Fiabilidad:** Capacidad de recuperación de fallo de sistema por pérdida de datos

## 3. Listado de criterios de calidad

Para un proyecto más elaborado se realizan unos ciertos criterios de calidad éticos y que sigan las normativas legales.

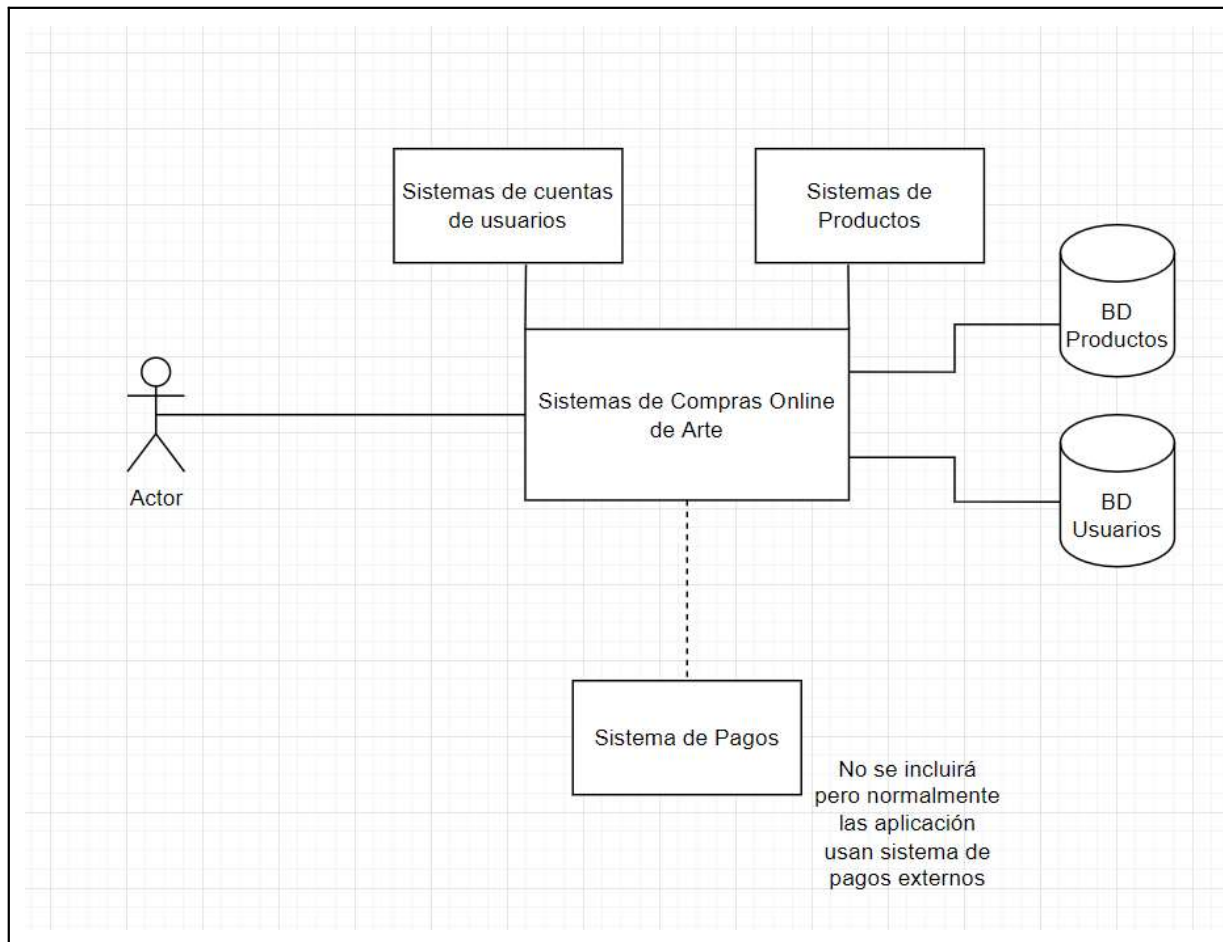
- **Ético:**
  - Transparencia hacia el cliente, la única forma de generar activos será por la venta de productos y no por la venta de información del cliente a otras empresas.

- Comercio Justo: Los artículos que se comercialicen cumplirán los criterios de solidaridad establecidos por la Coordinación Estatal de Comercio Justo.
- **Regulable y Seguro:**
  - La aplicación debe cumplir la Ley Orgánica 3/2018, del 5 de diciembre, de Protección de Datos Personales y la garantía de los derechos digitales. Y la Ley 34/2002, del 11 de julio de servicios de la sociedad de la información y del comercio electrónico.
- **Accesible y fácil de usar:**
  - Por normativa, todas las páginas deben ser accesibles. Buscaremos que nuestra aplicación sea accesible y fácil de usar para todo tipo de personas.
- **Internacionalizable:**
  - La aplicación será fácilmente adaptable para distintas, idiomas, monedas...

#### 4. Lista y descripción de las partes interesadas en el sistema

- Desarrolladores: Nosotros, encargados de elaborar la aplicación siguiendo un modelo arquitectónico y un diagrama de clases.
- Clientes: Aquellos usuarios que utilizarán la aplicación de compra.
- Técnico de pruebas: Al igual que los desarrolladores somos nosotros los encargados de realizar las pruebas necesarias para que todo funcione perfectamente.
- Ingeniero de producción y mantenimiento y webmaster, supervisando el mantenimiento de la aplicación y dando soporte hardware y middleware a esta.
- Administrador del Sistema, también para el mantenimiento de la aplicación.

## 5. Diagrama de Modelo de Contexto



## 6. Diseño de Pruebas de Sistema Integración

Planificación (decidir a grandes rasgos)		
¿Qué se quiere probar y cómo?	Se quieren probar los requisitos funcionales del sistema, cualquier interacción posible en busca de posibles errores. Se introducirán datos tanto correctos como erróneos esperando que el sistema detecte lo incorrecto y le pida al usuario cambiar los datos por correctos	
¿Quién lo hará y cuándo?	Normalmente sería un técnico de pruebas, pero en este caso lo haremos nosotros como grupo las pruebas pertinentes.	
¿Qué criterios se usarán para decidir cuándo terminar las pruebas?	Que los requisitos funcionales cumplan con sus funcionalidades básicas y puedan ejecutarse sin errores propios del sistema o errores de introducción de datos.	
Análisis (especificar las distintas condiciones de prueba)		
A partir de la especificación de requisitos funcionales/no funcionales de tu sistema, elige los elementos específicos cuyo comportamiento haya que probar, identificando las condiciones y datos requeridos para probarlos		
Elemento a probar	Condiciones	Datos Requeridos
Dar alta usuario	Introducción de datos de usuario de manera correcta e incorrecta.	Nombre, apellidos, email, teléfono, username, contraseña
Iniciar sesión usuario	Introducción de usuario y contraseña. No permitirá que inicie sesión un usuario con una contraseña que no le corresponde	Username y contraseña
Consultar catálogo producto	Permite visualizar todos los productos al entrar en la aplicación	
Filtrar productos por sus características	Permite visualizar productos que cumplan con cualquier combinación de filtros que se haga	Asignar los filtros requeridos y sus datos.
Realizar Compras	Permite escoger un producto y darle a comprar	Selección de un producto, con idProducto

La prueba de integración consistirá en la simulación de la compra de un producto. Para ello, el usuario deberá introducir su nombre de usuario y su clave para iniciar sesión. Posteriormente pulsará el botón de “iniciar sesión” y se le llevará a la página principal de los productos.

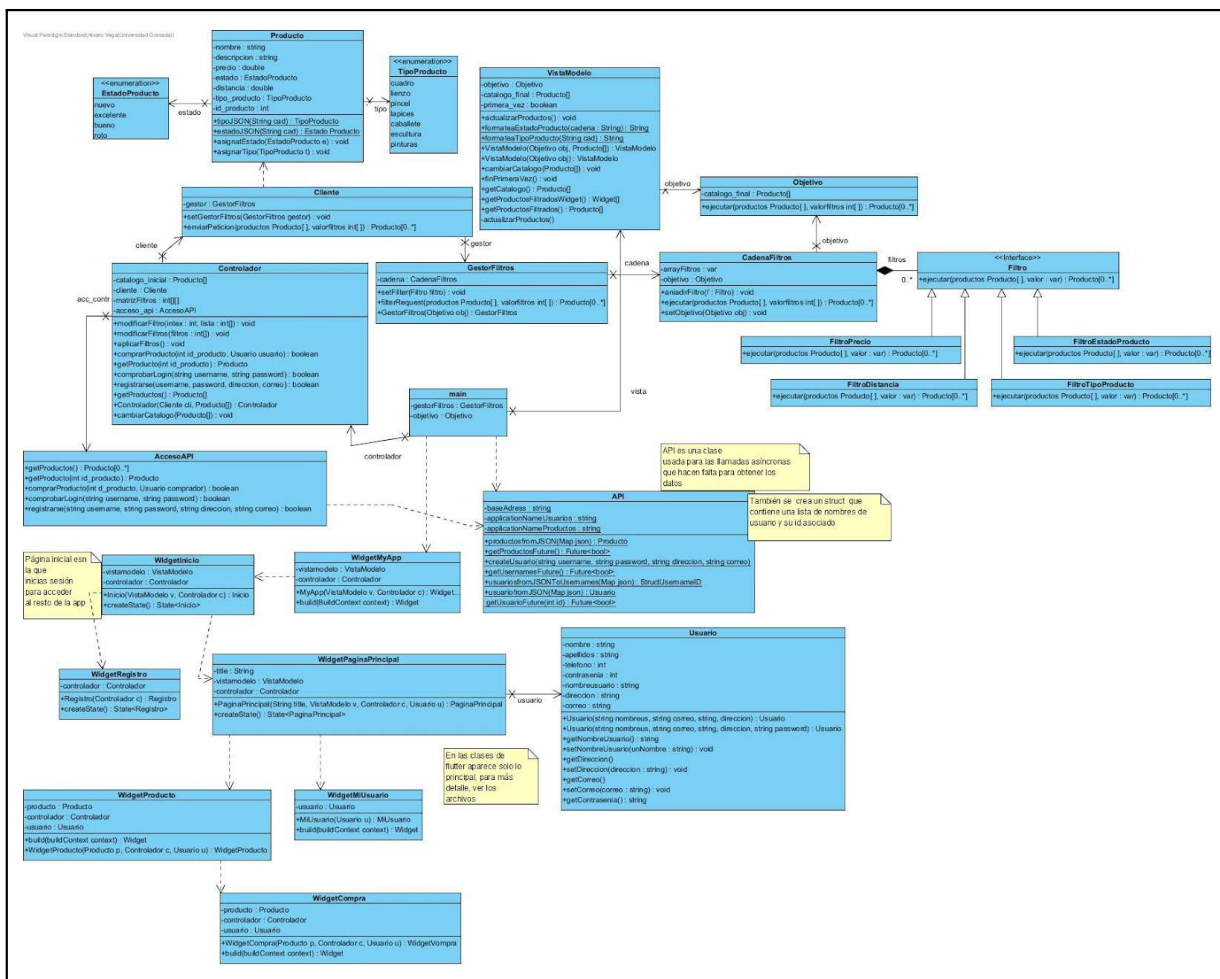


Éste elegirá uno para ver en detalle y pulsará el botón correspondiente, que le llevará a la página de compra. En dicha página podrá comprar el producto deseado al pulsar el botón “comprar”.

## Parte de Diseño

## 7. Diagrama de Clases

Nuestro sistema es bastante sencillo y complejo a la vez.



(Para visualizar la imagen con mayor tamaño, mire el archivo DiagramaClases Movil)

## 8. Diseño de pruebas: condiciones de prueba y casos de prueba

### a. Pruebas de unidad

- **Prueba 1.** Se modificará el atributo 'tipo producto' asociado a un producto específico y se comprobará que el cambio se ha realizado correctamente. Para ello, se utilizará el método *asignarTipo* que permite alterar el tipo del producto, como su propio nombre indica.
- **Prueba 2.** Se modificará el estado del producto de un producto determinado mediante el método *asignarEstado* y se comprobará que se ha realizado correctamente mediante la impresión por pantalla de la información de dicho producto.
- **Prueba 3.** Se comprobará que el atributo del catálogo inicial verdaderamente contiene todos los productos que se han introducido en el programa.
- **Prueba 4.** Se simulará la modificación y aplicación de los filtros. Por ejemplo, se añadirá un nuevo filtro que disminuya el precio de unos artículos concretos y se probará su aplicación modificando la lista de filtros. Para ello se utilizará el método *modificarFiltro* de la clase Controlador.
- **Prueba 5.** Se simulará la compra de un producto por parte de un cliente y se añadirá a su pedido mediante el método *añadirProductoPedido* de la clase cliente.
- **Prueba 6.** Se comprobará que el atributo que contiene la instancia del gestor de filtros en la clase cliente se haya creado correctamente.

### b. Pruebas de widgets

- **Prueba 1.** Se simulará la compra de un producto por parte de un usuario. Para ello, el usuario deberá elegir un producto del cual quiere saber más detalles y, tras pulsar el botón "ver más", llegar a la página de compra. Allí podrá comprar el producto al pulsar el botón "comprar".

- **Prueba 2.** Se simulará la aplicación de los filtros de precio y distancia mediante la modificación de dichos sliders. Se comprobará que la lista de productos mostrados se ha reducido y que tienen las características seleccionadas.
- **Prueba 3.** Se simulará la aplicación del filtro de estado de un producto mediante la selección de un botón con el estado deseado (en este caso 'roto'). Se comprobará que la lista de productos mostrados se ha reducido y que tienen las características seleccionadas.

### **c. Prueba de integración**

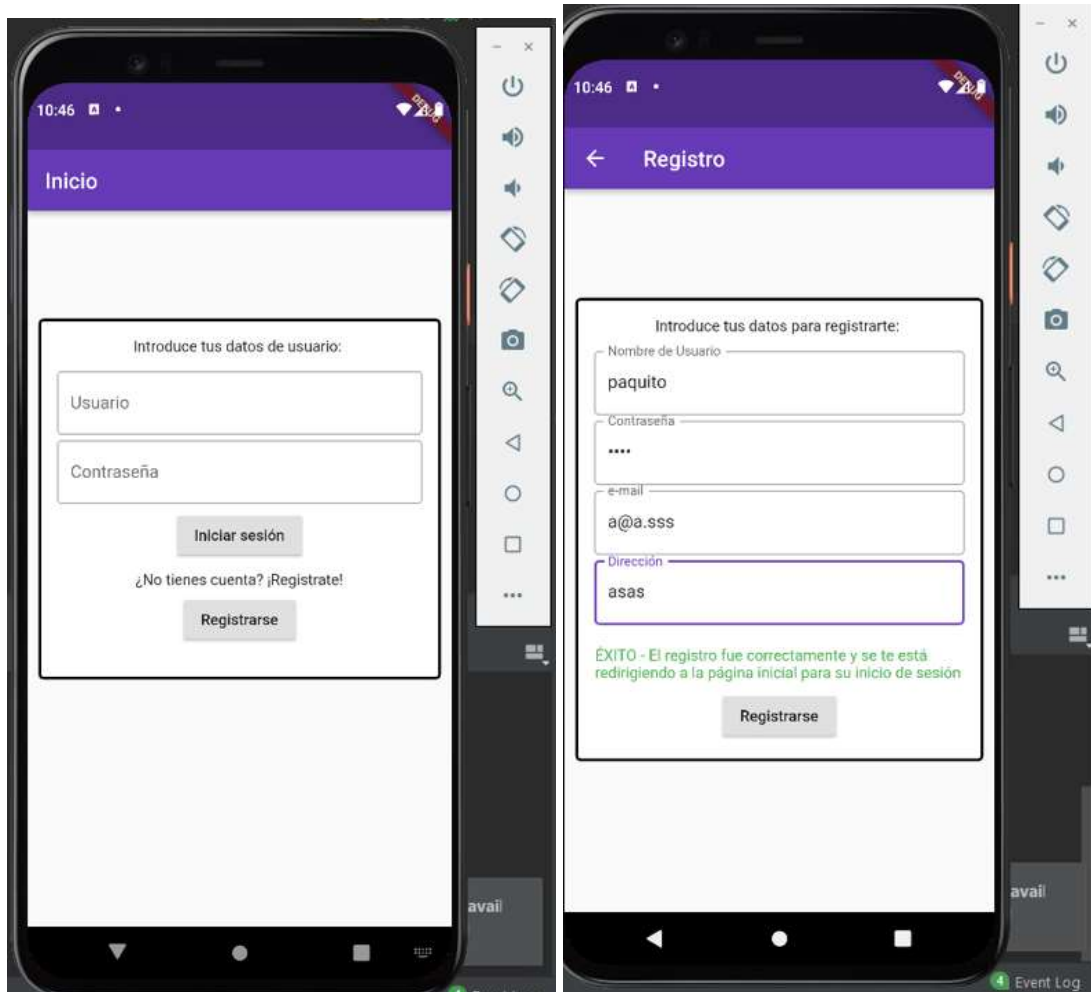
- **Prueba 1.** Se simulará la compra de un producto. Para ello, el usuario deberá introducir su nombre de usuario y su clave para iniciar sesión. Posteriormente pulsará el botón de "iniciar sesión" y se le llevará a la página principal de los productos, donde podrá ver detalles del producto que elija. Pulsará el botón correspondiente, que le llevará a la página de compra. En dicha página podrá comprar el producto deseado al pulsar el botón "comprar".

## Parte del Resultado

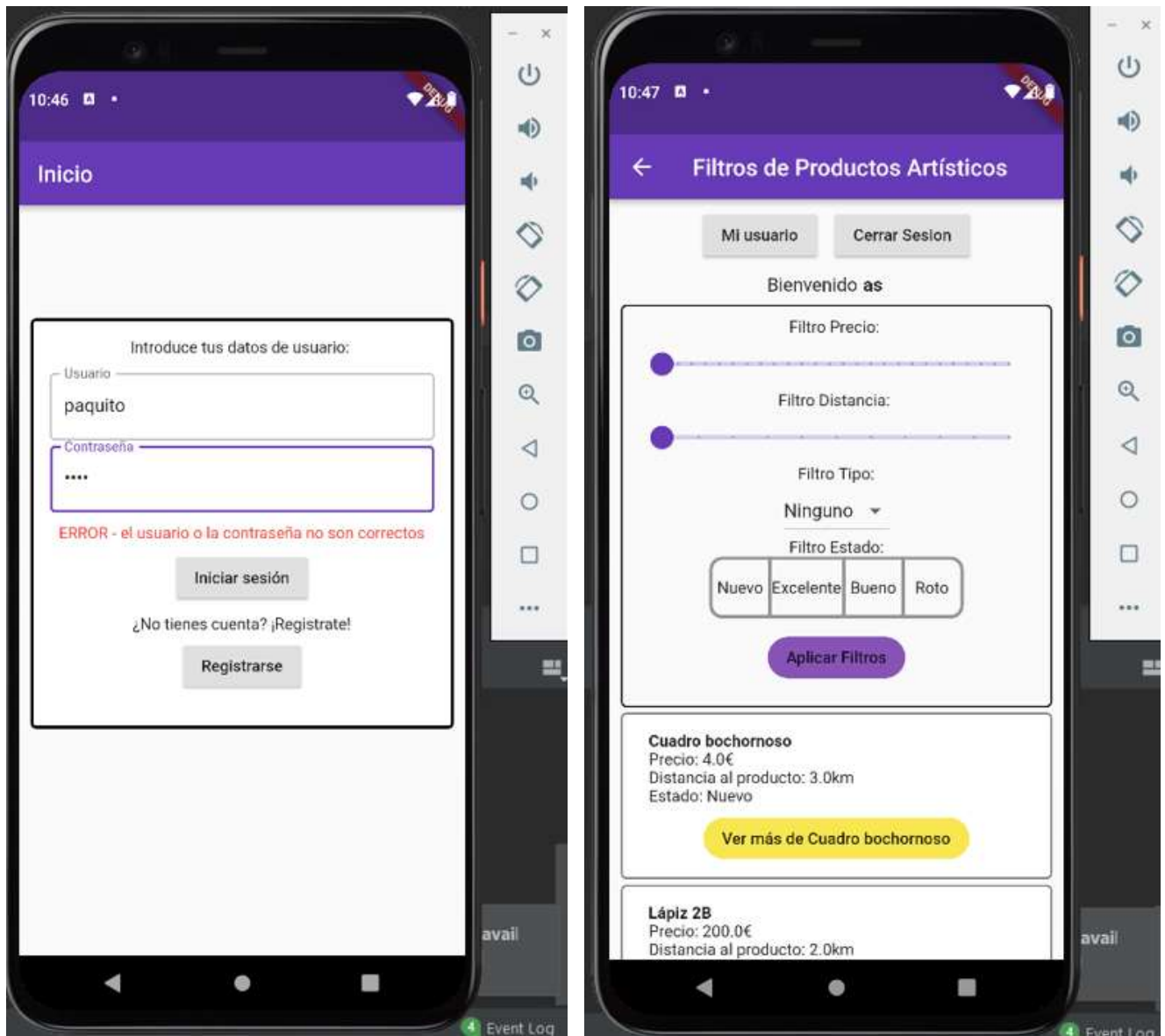
### 9. Capturas de Pantalla del uso de la Aplicación:

(Se pueden ver las capturas en su directorio)

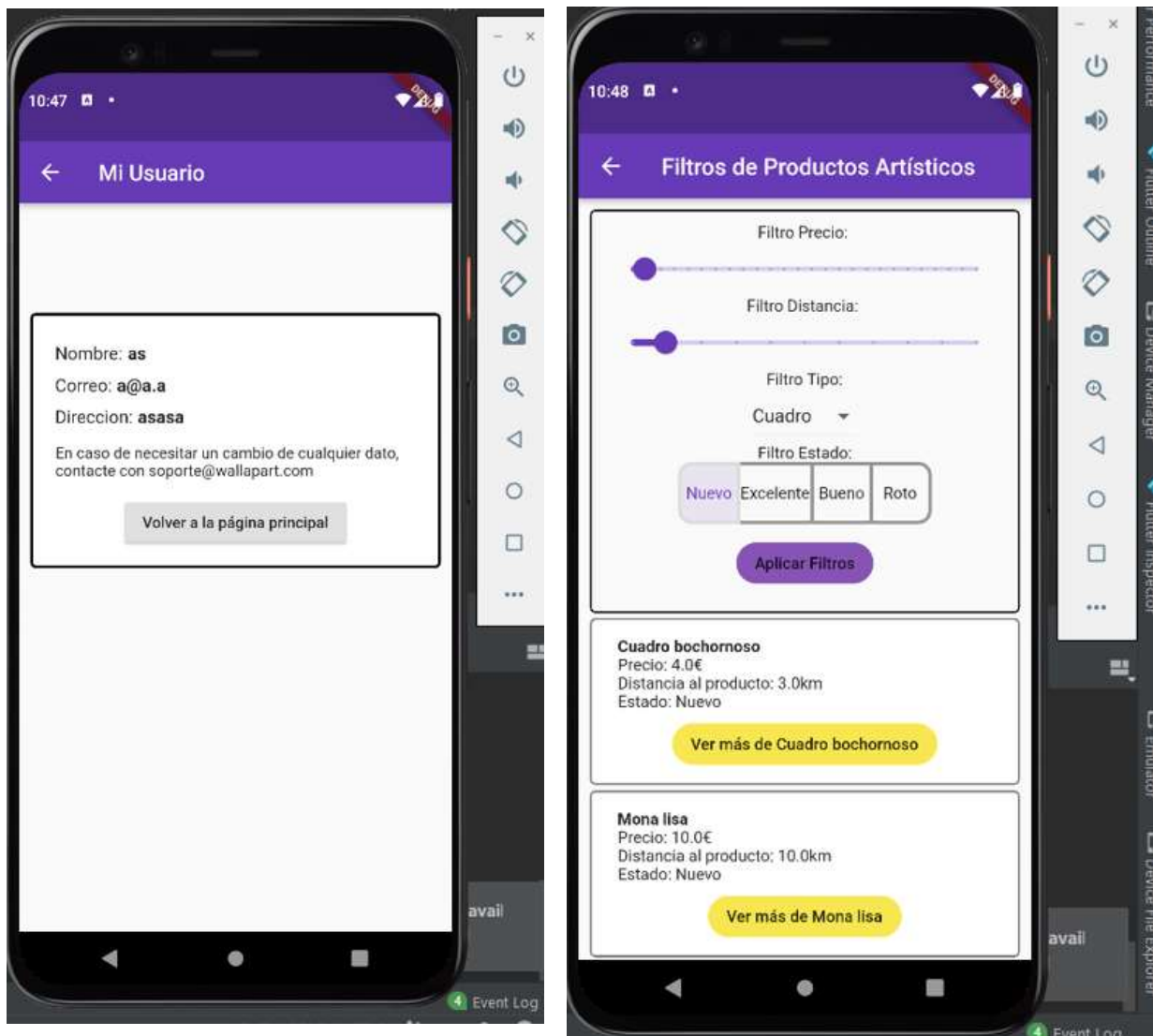
#### 1) Inicio de la aplicación y Registro de un nuevo usuario:



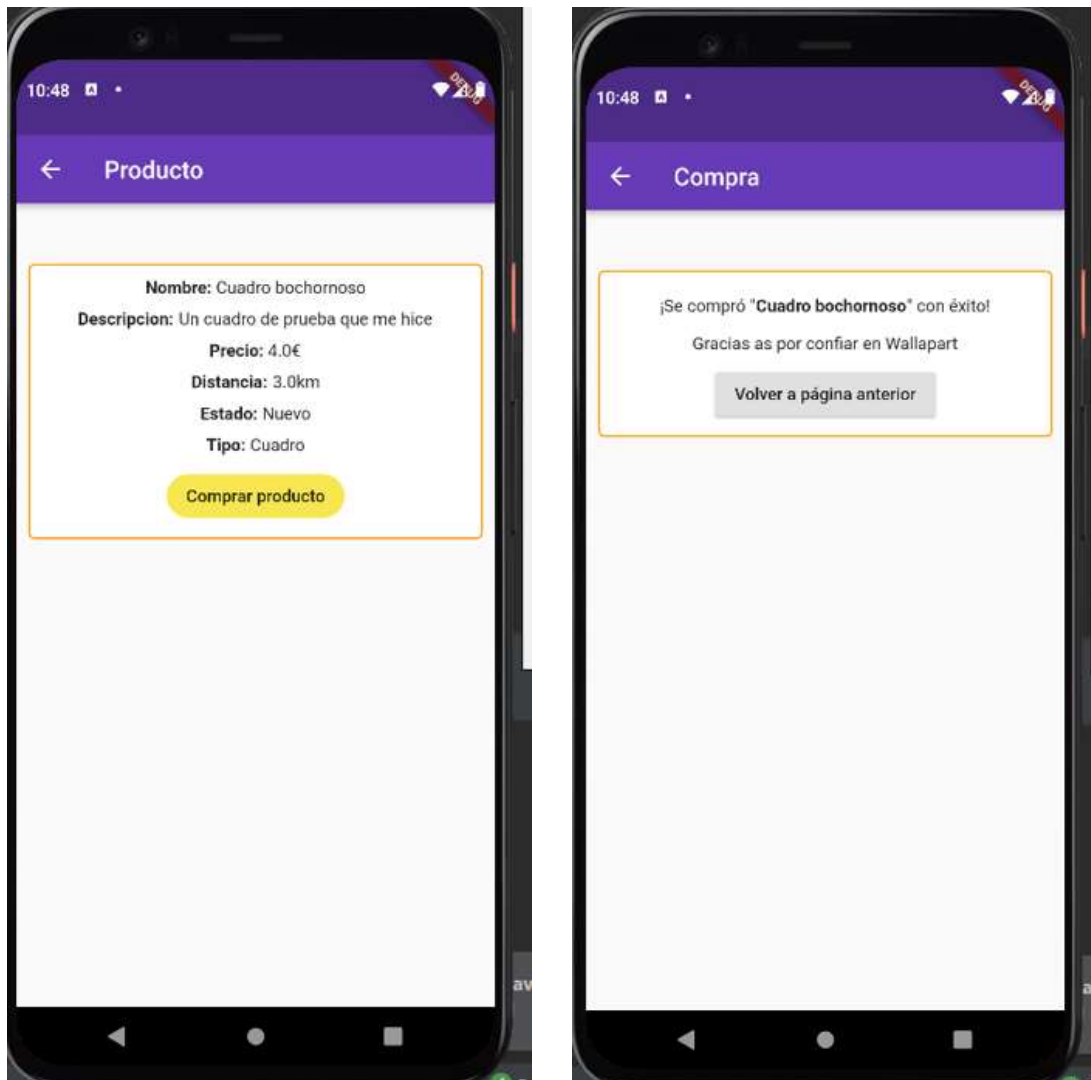
2) Prueba de inicio sesión errónea y acceso con otro usuario registrado



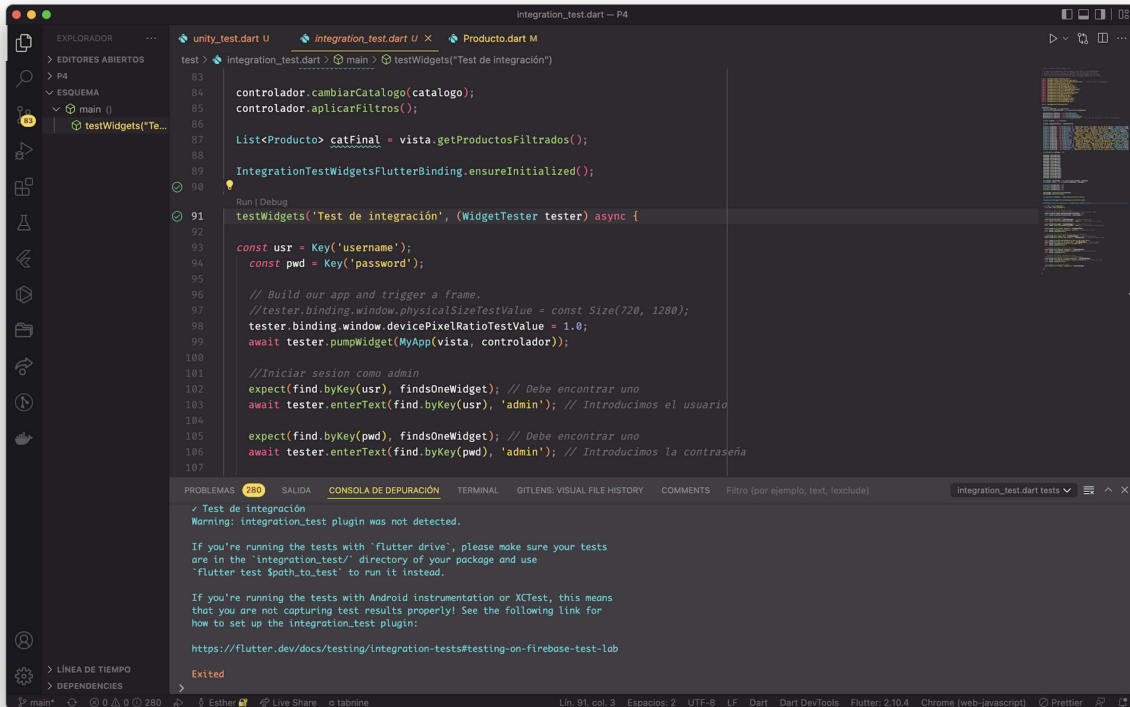
3) Prueba ver datos del usuario y filtrado de productos



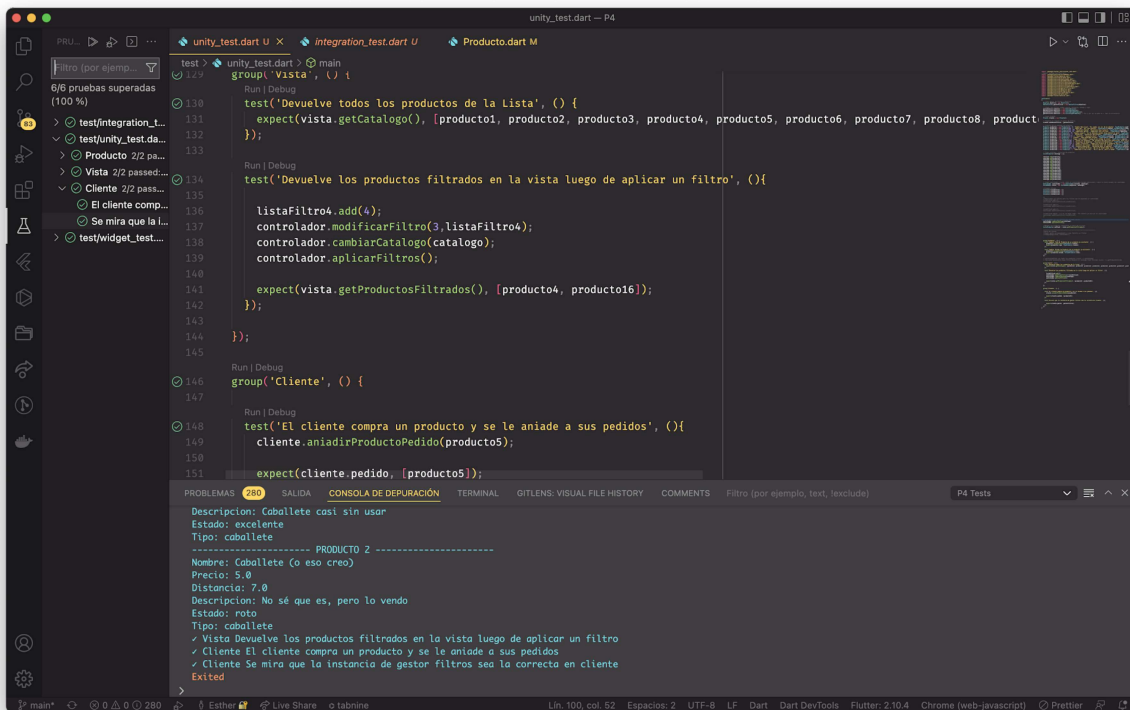
4) Ver detalles y comprar producto



## 10. Ejecución de Tests







## 11. Posibles futuras Ampliaciones/Mejoras

Ante todo, queremos puntualizar que, debido a la falta de tiempo y a problemas de algunos de los miembros del equipo (ya comentados al profesor), hay muchas cosas por hacer y mejorar.

- Más test que permitan probar la funcionalidad de la API, del controlador, etc. Además de las últimas funcionalidades.
- Arreglar *bugs* debidos a la inclusión de la API y sus métodos asíncronos.
- Refinar el código y algunos “apaños” de última hora. Debido a nuestro estado físico, no hemos tenido suficiente tiempo y energía para arreglar estos aspectos, refinarlos y hacer un código de tanta calidad como nos hubiera gustado.
- Encriptar la contraseña.
- Añadir fotos a los productos
- API con maps para distancia real.
- ...