Documentación de pruebas

Pruebas unitarias:

Pruebas unitarias realizadas: 5

**P.U 1 Mostrar las tasas de cambios de las monedas obtenidas por respuesta de la API REST**

Pasos de comprobación

1-iniciar la aplicación

2-dirigirse a inicio

3-presionar botón “ver valores de monedas internacionales”

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamenteCaptura de pantalla de un celular

Descripción generada automáticamente

Código prueba unitaria P.U 1:

Texto

Descripción generada automáticamente

RESULADO PRUEBA UNITARIA P.U 1 CON JASMINE Y KARMA

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente con confianza media

Conclusión P.U 1: no se detectan incidencias

**P.U 2 SE AGREGA CORRECTAMENTE UN PRODUCTO SI TODOS LOS CAMPOS SON VALIDOS**

**Pasos para la comprobación**

1. **Iniciar la aplicación**
2. **Logearse como administrador**
3. **Dirigirse a “productos”**
4. **Presionar “agregar zapatilla”**
5. **Completar el formulario**
6. **Presionar el botón “agregar”**
7. **Comprobar si el producto fue agregado**

Código prueba unitaria P.U 2

Texto

Descripción generada automáticamente

RESULTADO PRUEBA UNITARIA P.U 2 CON JASMINE Y KARMA

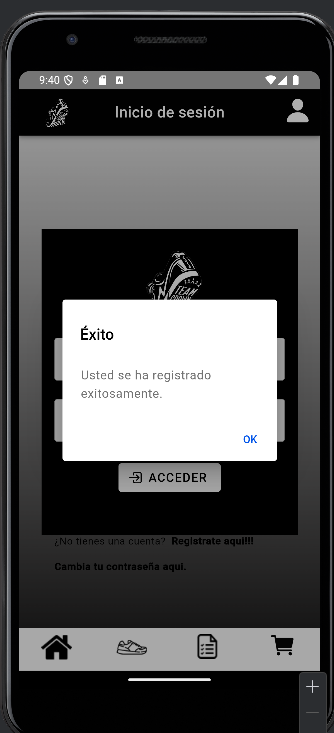
Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

P.U 3 VALIDAR LOS CAMPOS DE USUARIO AL MOMENTO DE REGISTRARSE

PASOS PARA COMPROBACION:

1. Iniciar la aplicación
2. Dirigirse al login
3. Presionar en “regístrate aqui”
4. Completar el formulario de registro
5. Presionar el registrar



Código P.U.3

Texto

Descripción generada automáticamente

RESULTADO PRUEBA UNITARIA P.U 3 CON JASMINE Y KARMA

Texto

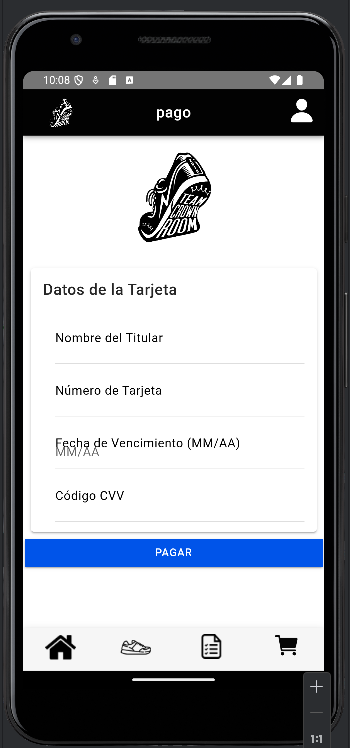
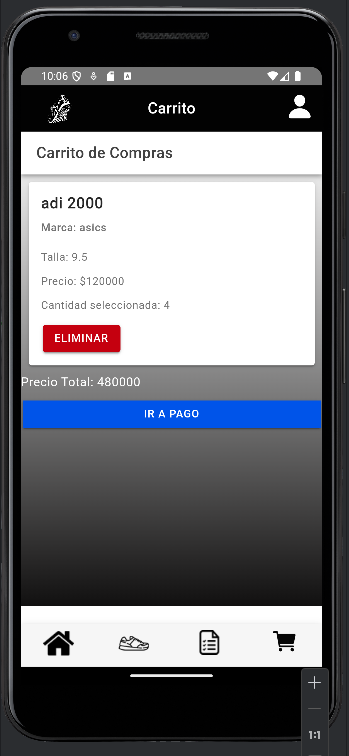
Descripción generada automáticamente

P.U 4 registrar la compra si todos los campos son validos

Pasos para comprobación

1. Iniciar la aplicación
2. Logearse como administrador o como usuario
3. Seleccionar el producto a comprar
4. Elegir la cantidad a comprar
5. Añadir el producto al carrito
6. Dirigirse al carrito
7. Presionar “ir a pago”
8. Completar el formulario de compra
9. Presionar “pagar”
10. Comprobar que la compra quede registrada en “mis compras”

Captura de pantalla de un celular

Descripción generada automáticamente



Código P.U 4

Texto

Descripción generada automáticamente

RESULTADO PRUEBA UNITARIA P.U 4 CON JASMINE Y KARMA

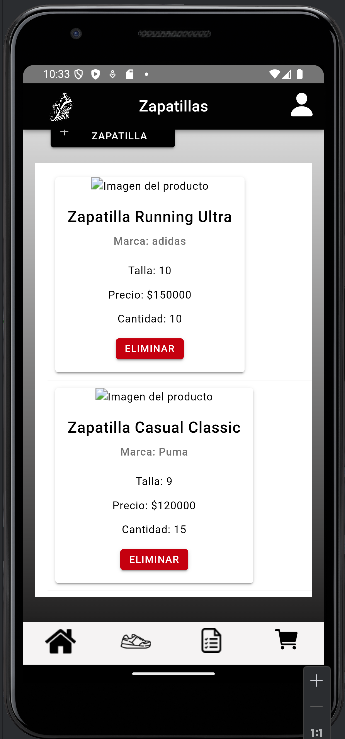
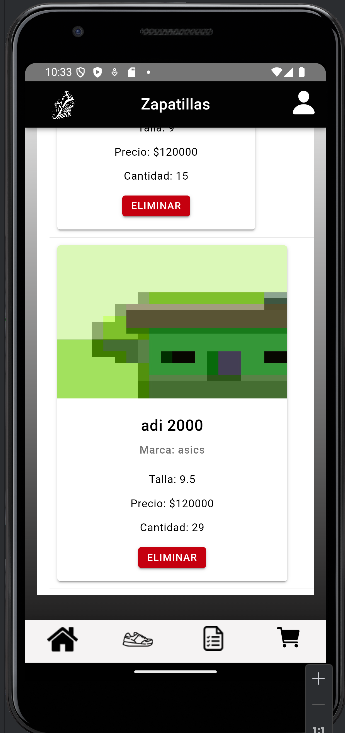
Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

P.U 5 ELIMINAR UN PRODUCTO

Pasos para la comprobación

1. Iniciar la aplicación
2. Logearse como administrador
3. Dirigirse a productos
4. Presionar el botón “eliminar” del producto que quieres eliminar
5. Revisar si el producto se ha eliminado



CODIGO P.U 5

Texto

Descripción generada automáticamente

RESULTADO PRUEBA UNITARIA P.U 5 CON JASMINE Y KARMA

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

PRUEBAS UNITARIAS EN GENERAL

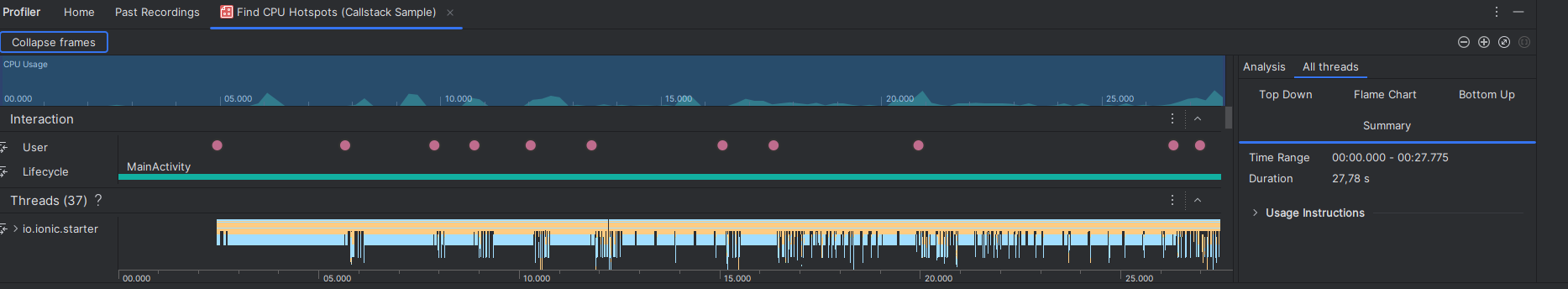
Texto

Descripción generada automáticamente con confianza baja

**PRUEBAS DE CARGA**

**Utilizando Android profiler**

Find CPU hotspot : el gráfico y las secciones que se pueden observar en la imagen representan el comportamiento del uso de la **CPU** y las interacciones de tu aplicación durante la ejecución de ciertas acciones como la que realice que fue la de iniciar sesión con un usuario y cambiar de paginas dentro de mi aplicación



**User (Puntos rosados)**:

* Estos puntos indican momentos en los que hubo **interacciones del usuario**, como toques en botones o desplazamientos.
* En tu caso, un punto puede coincidir con la acción de presionar el botón de "Iniciar sesión".

**Lifecycle (Barra verde agua)**:

* Representa los cambios en el ciclo de vida de la actividad (MainActivity en este caso).
* Una barra continua indica que la actividad estuvo activa y ejecutándose.

**Threads (Hilos)**

* Esta sección muestra el uso de la CPU por cada **hilo** que ejecuta tu aplicación.
* **Barra con colores (como azul, amarillo)**:
  + **Azul claro**: El hilo está activo, pero esperando eventos o recursos.
  + **Naranja**: El hilo está ejecutando tareas activamente en la CPU.
  + **Blanco**: El hilo está inactivo.
* **Hilo principal (io.ionic.starter)**:
  + Es el **UI thread**, donde se procesan las acciones de la interfaz de usuario.
  + Si el hilo principal está constantemente en **naranja**, podría haber tareas pesadas que están bloqueando el rendimiento de la interfaz.

**Picos en el uso de CPU**:

* Si observas un pico en el gráfico de uso de CPU después de presionar "Iniciar sesión", indica que la aplicación ejecutó una tarea intensiva, como una solicitud de red o verificación en la base de datos.
* Esto es normal, pero si el pico es muy alto o se prolonga demasiado, podrías optimizar esa tarea.

**Hilos bloqueados**:

* Si el hilo principal (io.ionic.starter) está en **naranja** durante mucho tiempo, significa que tareas intensivas están ejecutándose en el **UI thread**.
* **Solución**: Mueve esas tareas al **hilo de fondo** (background thread) usando algo como AsyncTask o Coroutines.

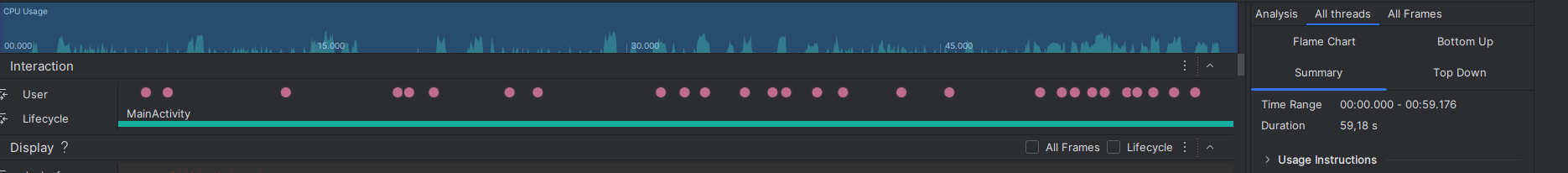
**Interacciones del usuario**:

* Los puntos rosados verifican que las acciones del usuario se registraron correctamente. Si el gráfico de uso de CPU no muestra actividad significativa después de un punto, es posible que la acción no esté bien implementada o que haya un problema en la lógica.

**Duración total**:

* El tiempo total en el **Analysis Panel** (27.78 segundos en tu caso) indica cuánto tiempo estuvo analizando la CPU. Puedes usar este rango para identificar problemas de rendimiento.

Pruebas utilizando capture system activities



**. CPU Usage (Gráfico superior)**

* **Qué significa**:
  + Este gráfico muestra la carga de la CPU a lo largo del tiempo mientras interactuabas con tu aplicación.
  + Los picos más altos corresponden a tareas que demandaron más procesamiento (como cargar datos del servidor o procesar el formulario de compra).
* **Qué buscar**:
  + Si hay picos excesivamente altos o prolongados, podrías tener tareas que están sobrecargando la CPU.
  + Si la CPU está constantemente ocupada (sin descansos), revisa el código para identificar bucles o tareas innecesarias.

**Interaction (User y Lifecycle)**

**a. User (Puntos rosados):**

* **Qué significa**:
  + Cada punto rosado representa una interacción del usuario, como un toque en un botón, un campo de texto, o una acción en la interfaz.
  + Puedes ver que estas interacciones están distribuidas regularmente, lo que coincide con los pasos realizados en tu prueba (inicio de sesión, selección de producto, etc.).

**b. Lifecycle (Barra verde):**

* **Qué significa**:
  + La barra verde representa el ciclo de vida de la actividad principal (MainActivity).
  + Una barra continua indica que la actividad estuvo activa durante toda la prueba.
  + Si esta barra se interrumpiera, podría indicar pausas o reinicios no esperados de la actividad.

**Interpretación específica de la prueba**

* **Inicio de sesión**:
  + Seguramente el gráfico mostró un pico al validar credenciales (posiblemente con una solicitud de red o acceso a la base de datos).
* **Selección de producto y cantidad**:
  + Estas acciones probablemente involucran interacciones de interfaz de usuario, que deberían ser rápidas y con poca carga.
* **Agregar al carrito**:
  + Podría haber un pequeño pico si esto implica guardar datos localmente o en el backend.
* **Compra y verificación**:
  + El pico más alto en la CPU probablemente corresponde a la confirmación de la compra, especialmente si involucra comunicación con el servidor y actualizaciones en la base de datos.

**Prueba de stress con firebase lab robotest**

**Problemas de las pruebas**

**Captura de pantalla de un celular

Descripción generada automáticamente**

Rendimiento

Gráfico, Histograma

Descripción generada automáticamente

Analizando el grafico y sus resultados:

**CPU (Porcentaje)**

* **Qué significa**:
  + Representa el porcentaje de uso de la CPU a lo largo del tiempo.
  + Las variaciones en el gráfico reflejan las operaciones realizadas por mi aplicación.
* **Interpretación**:
  + Al inicio, hay picos frecuentes pero controlados (alrededor del 10%-15%), probablemente por tareas iniciales como cargar pantallas o interactuar con la interfaz.
  + A medida que la prueba avanza, se observa un comportamiento consistente con picos ocasionales más altos (hasta el 20%), lo que indica procesos intensivos (posiblemente consultas a la red o procesos de backend).
  + Cerca del final, el uso de CPU disminuye, lo cual es normal si la aplicación no está ejecutando tareas pesadas.

**Memoria (KiB)**

* **Qué significa**:
  + Muestra el uso total de memoria (RAM) por parte de mi aplicación a lo largo de la prueba.
* **Interpretación**:
  + Al inicio, hay un aumento rápido de memoria (aproximadamente 200 KiB), lo cual es normal porque la app está cargando recursos iniciales (imágenes, datos, etc.).
  + Después, el uso de memoria se estabiliza, lo que indica que no se están acumulando objetos innecesarios en memoria.
  + Cerca del final, la memoria vuelve a caer, probablemente debido a la finalización de la prueba y la liberación de recursos.

**Red (Bytes/s)**

* **Qué significa**:
  + Muestra la cantidad de datos enviados (naranja) y recibidos (azul) por la red durante la prueba.
* **Interpretación**:
  + En este gráfico, el uso de red es mínimo o nulo durante la prueba, lo que indica que la app no está realizando operaciones intensivas de red o que las solicitudes son ligeras.
  + Si tu aplicación depende de una API, este comportamiento puede significar que las llamadas son esporádicas o bien optimizadas.

**Conclusión general**

**Comportamiento observado:**

* **CPU**: Estable, con picos moderados. Indica un uso eficiente de los recursos.
* **Memoria**: Estabilizada después de la carga inicial, lo que es ideal.
* **Red**: Actividad mínima o inexistente, lo cual es normal si tu app no hace uso intensivo de la red.

Video de firebase robo test manipulando la aplicación, resultando en el grafico anterior de rendimiento



Descripción general

Captura de pantalla de un celular

Descripción generada automáticamente