PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA EN SISTEMAS



Trabajo de Titulación

Tema: Desarrollo de un prototipo de aplicación para el control de gastos personales

AUTOR:

Erick Julián Correa Loachamín

QUITO DM, JUNIO DE 2024

Agradecimiento

Agradezco a mis padres quienes me han dado su apoyo incondicional a lo largo de mi trayectoria universitaria y han sido mis pilares fundamentales en cada etapa de mi vida. Les estoy agradecido por su confianza y amor incondicional y por haberme enseñado a perseguir mis sueños y nunca rendirme.

Agradezco a mis profesores quienes han formado parte de mi trayectoria universitaria y han sido parte fundamental en mi desarrollo profesional.

Tabla de contenido

1. CAPÍTULO 1 7

1.1. JUSTIFICACIÓN 7

1.2. PLANTEAMIENTO DEL PROBLEMA 7

1.3. OBJETIVOS 8

1.3.1. GENERAL 8

1.3.2. ESPECÍFICOS 8

1.4. ALCANCE 8

2. CAPÍTULO 2 9

2.1. ANTECEDENTES 9

2.1.1. APLICACIONES DE GESTIÓN DE FINANZAS 9

2.2. MARCO TEÓRICO 10

2.2.1. METODOLOGÍA DE DESARROLLO 10

2.2.2. FIREBASE 10

2.2.3. IDE DE DESARROLLO ANDROID STUDIO 11

2.2.4. DISEÑO CENTRADO EN EL USUARIO 12

2.2.5. EXPERIENCIA DEL USUARIO 12

3. CAPÍTULO 3 14

3.1. FASE DE PLANIFICACIÓN 14

3.1.1. HISTORIAS DE USUARIO 14

3.1.2. ITERACIONES 16

3.2. FASE DE DISEÑO 18

3.2.1. TARJETAS CRC 18

3.3. FASE DE CODIFICACIÓN 19

3.3.1. BASE DE DATOS FIREBASE 19

3.3.2. FLUJO DE PANTALLAS 19

3.4. PRUEBAS DE LA APLICACIÓN 22

3.4.1. PRUEBAS DE FUNCIONAMIENTO 22

3.4.2. PRUEBAS DE RENDIMIENTO 48

4. CONCLUSIONES Y RECOMENDACIONES 51

4.1. CONCLUSIONES 51

4.2. RECOMENDACIONES 53

5. BIBLIOGRAFÍA 54

6. ANEXOS 56

Índice de ilustraciones

Ilustración 1. Estructura de la base de datos en Firebase 19

Ilustración 2. Pantallas de inicio y menú 19

Ilustración 3. Pantallas de categorías 20

Ilustración 4. Pantallas de pagos 20

Ilustración 5. Pantallas de reportes 21

Ilustración 6. Pantalla para crear un gasto 22

Ilustración 7. Mensaje exitoso al crear un gasto 23

Ilustración 8. Gasto creado en la lista 23

Ilustración 9. Gasto seleccionado para eliminar 24

Ilustración 10. Confirmación de la eliminación del gasto 25

Ilustración 11. Mensaje de eliminación exitosa 25

Ilustración 12. Pantalla para crear un ingreso 26

Ilustración 13. Confirmación de la creación del ingreso 27

Ilustración 14. Ingreso seleccionado para eliminar 28

Ilustración 15. Mensaje para confirmar la eliminación 29

Ilustración 16. Mensaje de eliminación exitosa 29

Ilustración 17. Visualización de ingresos 30

Ilustración 18. Visualización de gastos 31

Ilustración 19. Pantalla de categorías 32

Ilustración 20. Creación de categoría 32

Ilustración 21. Mensaje de confirmación de creación de categoría 33

Ilustración 22. Lista de categorías 34

Ilustración 23. Pantalla para eliminar una categoría 34

Ilustración 24. Mensaje de confirmación para eliminar una categoría 35

Ilustración 25. Mensaje que confirma la eliminación de la categoría 35

Ilustración 26. Lista de categorías de gastos 36
Ilustración 27. Pantalla para modificar una categoría 37
Ilustración 28. Mensaje que confirma la modificación de la categoría 37
Ilustración 29. Pantalla para recordatorios 38
Ilustración 30. Pantalla para crear un recordatorio 39
Ilustración 31. Mensaje que confirma la creación del pago 39
Ilustración 32. Notificación del recordatorio 40
Ilustración 33. Lista de recordatorios 41
Ilustración 34. Pantalla del recordatorio del pago 41
Ilustración 35. Confirmación de la eliminación del recordatorio 42
Ilustración 36. Mensaje que confirma la eliminación del recordatorio 42
Ilustración 37. Pantalla de reportes 43
Ilustración 38. Resumen del reporte en la aplicación 44
Ilustración 39. Notificación de la creación del archivo PDF 44
Ilustración 40. Informe PDF 45
Ilustración 41. Pantalla para crear reportes 46
Ilustración 42. Lista del reporte 46
Ilustración 43. Notificación hoja de cálculo 47
Ilustración 44. Hoja de cálculo generada 47
Ilustración 45. Instalación de firebase-admin 49
Ilustración 46. Script para realizar la prueba de inserción 49
Ilustración 47. Resultados del script 50
Ilustración 48. Respuesta del script 51
Ilustración 49. Script para realizar la prueba de consulta 51

Índice de tablas

Tabla 1. Historia de usuario 1 14

Tabla 2. Historia de usuario 2 14

Tabla 3. Historia de usuario 3 15

Tabla 4. Historia de usuario 4 15

Tabla 5. Historia de usuario 5 15

Tabla 6. Historia de usuario 6 16

Tabla 7. Iteración 1 16

Tabla 8. Iteración 2 16

Tabla 9. Iteración 3 17

Tabla 10. Iteración 4 17

Tabla 11. Tarjeta CRC categoría 18

Tabla 12. Tarjeta CRC transacción 18

Tabla 13. Tarjeta CRC pago 18

1. CAPÍTULO 1

1.1. JUSTIFICACIÓN

Actualmente, gestionar las finanzas personales se ha vuelto de suma importancia, puesto que la buena administración financiera permite un buen bienestar económico y a su vez estabilidad económica. A pesar de esto, existen pocas herramientas tecnológicas para gestionar correctamente los gastos personales, por lo cual, se propone la creación de un aplicativo móvil que permita el control de gastos personales.

En la sociedad actual, la digitalización se ha vuelto fundamental en el día a día de las personas. Por ende, es necesario desarrollar herramientas que funcionen junto con las nuevas plataformas tecnológicas y que sean accesibles para todos. En este contexto, las aplicaciones móviles o web son elementos esenciales para lograr este objetivo, así pues, la aplicación busca proveer de una herramienta tecnológica a los usuarios para llevar un buen manejo de sus finanzas.

1.2. PLANTEAMIENTO DEL PROBLEMA

El problema radica en la dificultad para determinar qué gastos se han hecho y en la planificación financiera a largo plazo, de igual forma parte del problema es que varias aplicaciones carecen de opciones que permitan a los usuarios manejar de una manera correcta sus finanzas, lo que provoca que muchas personas no puedan comprender y controlar sus hábitos de gastos, lo que resulta en una planificación financiera precaria.

Algunos problemas secundarios también son la complejidad de algunos sistemas financieros, pues al no saber cómo usarlos genera desconfianza en los usuarios, y la carencia de herramientas especializadas, lo que no permite realizar un

seguimiento eficiente de los gastos. Por lo que este trabajo busca ayudar con estos problemas a través de una aplicación que permita manejar los gastos personales.

1.3. OBJETIVOS

1.3.1. GENERAL

Desarrollar un prototipo de una aplicación de control de gastos personales que permita a los usuarios gestionar sus finanzas de manera efectiva a través de una aplicación móvil.

1.3.2. ESPECÍFICOS

- Realizar el levantamiento de las expectativas y necesidades del usuario para el sistema de control de gastos personales, con el fin de determinar los requerimientos funcionales y no funcionales.
- Diseñar un prototipo funcional del sistema de gestión de gastos.
- Desarrollar un prototipo de una aplicación de gestión de gastos, que incorpore las características clave identificadas a través del levantamiento de requerimientos.

1.4. ALCANCE

El proyecto tiene como objetivo desarrollar el prototipo para una aplicación de control de gastos, por lo que, se establece el alcance hasta la fase de pruebas, en este caso como la metodología seleccionada es XP o *Extreme Programming*, el proyecto busca llegar hasta la fase de pruebas de XP.

2. CAPÍTULO 2

2.1. ANTECEDENTES

2.1.1. APLICACIONES DE GESTIÓN DE FINANZAS

Actualmente existen diversas aplicaciones de gestión de finanzas dentro de las tiendas de aplicaciones, estas pueden resultar útiles hasta cierto punto, por ejemplo, las bancas virtuales son de gran ayuda para llevar registros de consumos que se realizan a través de los servicios de la banca, sin embargo, muchos de los gastos que se realizan a diario no son a través de los servicios de la banca, lo que complica su seguimiento.

A pesar de la existencia de aplicaciones que buscan ayudar a la gestión de gastos, muchas solo tienen como objetivo principal el registro de gastos y ninguna otra funcionalidad que ayude a los usuarios con su planificación financiera, ocasionando que muchas de estas aplicaciones no sean ocupadas debido a las limitaciones que tienen.

Una de las aplicaciones que se puede mencionar es Gestor de Gastos, que se encuentra en la Play Store, su principal funcionalidad es el registro de gastos y de ingresos de dinero, también tiene como funcionalidad la creación de gráficos con los gastos e ingresos diarios, semanales, mensuales y anuales y un informe con todos los gastos, ingresos y transferencias.

A pesar de incluir todas estas funcionalidades, el informe que proporciona no ofrece información relevante que pueda ayudar al usuario a tomar mejores decisiones financieras, por lo que el uso de la aplicación puede quedar limitado.

2.2.MARCO TEÓRICO

2.2.1. METODOLOGÍA DE DESARROLLO

Una metodología de desarrollo son los métodos y técnicas que permiten manejar el ciclo de vida de software, son los procesos que se deben seguir para crear un producto de desarrollo de software.

La metodología de desarrollo *Extreme Programming*, está centrada en proyectos de pequeño y mediano tamaño, donde la participación de los usuarios es esencial para el proceso, en esta se hace el uso de iteraciones, lo que permite una planificación iterativa, brindando una ventaja al proceso de desarrollo, que es el retorno en el ciclo de vida, permitiendo hacer cambios de manera más fácil y ágil.

Su ciclo de vida está estructurado por la planificación, el diseño, la codificación, las pruebas y la integración, todas estas etapas permiten a quienes usan esta metodología, gestionar proyectos de forma simple y veloz, y a su vez entregar proyectos de alto valor.

La metodología seleccionada contribuirá a cumplir con el plazo de tiempo establecido, pues permite una planificación ágil y adaptable, de modo que a su vez permitirá la creación de un proyecto de alto valor.

2.2.2. FIREBASE

Firebase es una plataforma en línea que sirve para el desarrollo web y aplicaciones móviles, está destinada a su uso dentro de varios sistemas operativos como *Android* y *iOS* así también a diversos *IDEs* como lo son

REST y *Unity*, de modo que brinda diversos complementos que la hacen una excelente herramienta para desarrollo.

Entre estos complementos, los que se pueden destacar son, la autenticación de usuarios a través de diversos métodos ya sea solo usando un correo y una contraseña o una cuenta de *Google* y otras plataformas, de igual forma brinda el servicio en línea de base de datos *NOSQL* en la que se almacenan los datos en formato *JSON* y permite que estén disponibles en todo momento y en cualquier lugar.

Esta herramienta ha sido seleccionada para el desarrollo de esta aplicación, pues las ventajas que posee ayudan a la creación de proyectos más completos y con grandes funcionalidades.

2.2.3. IDE DE DESARROLLO ANDROID STUDIO

Android Studio es un IDE utilizado en la creación de aplicaciones *Android*, este está basado en *IntelliJ IDEA*, cuyos lenguajes de programación principales son *Kotlin* y *Java*.

Este IDE de desarrollo ofrece varias ventajas a la hora de desarrollar aplicaciones, entre estas, posee un emulador que permite observar cómo se verá la aplicación en los dispositivos móviles, un sistema de complicación basado en *Gradle*, un menú que permite navegar de forma eficaz sobre el proyecto y sobre todo un entorno que permite desarrollar aplicaciones para todos los dispositivos Android.

Todas estas ventajas vuelven a *Android Studio,* una potente herramienta que ayuda en el desarrollo de aplicaciones móviles. Por esta

razón ha sido seleccionada como la herramienta principal de desarrollo de este proyecto.

2.2.4. DISEÑO CENTRADO EN EL USUARIO

En este tipo de diseño el usuario debe estar involucrado en el proceso de creación de productos, lo que obliga a hacer uso de diversos conceptos y disciplinas que ayudan en el proceso, algunas de estas disciplinas o conceptos son: la ergonomía, la interacción humano-computador, la experiencia del usuario, el diseño de interacción, la usabilidad, entre otros, que ayudan a dar un mejor entendimiento a las preferencias y requisitos de los usuarios.

Así pues, el objetivo principal del diseño es la creación de aplicaciones o productos que sean usables y útiles para satisfacer las expectativas de los usuarios de manera correcta y a su vez ayude a mejorar la experiencia que tienen, pues al involucrar a estos en todas las etapas se logra un diseño más efectivo y satisfactorio.

2.2.5. EXPERIENCIA DEL USUARIO

Según la norma ISO 9241-220:2019, la experiencia de usuario se refiere a cómo los usuarios perciben y responden al uso de un sistema, producto o servicio (ISO 9241-220:2019), como tal son las sensaciones y valoraciones que se tienen después de usar un sistema o producto, que pueden ayudar a determinar si el sistema o producto satisface las necesidades y deseos de la persona que lo utiliza.

A partir de eso, la experiencia del usuario debe tener ciertos elementos para que esta pueda ser satisfactoria, según Peter Morville (2005)

para que sea de calidad, el producto, sistema o servicio debe ser útil, usable, localizable, creíble, accesible, deseable y sobre todo valioso, esto tomándolo desde un punto de vista organizacional y centrado en la información, lo que permite que el producto o sistema sea de gran utilidad para quien lo está usando.

De modo que una buena experiencia de usuario, no solo se basa en la funcionalidad del sistema o producto, sino también en cómo este siente y percibe el sistema al interactuar con este, ya que, si se siente capaz de usar el sistema y feliz de hacerlo lo usará con frecuencia, caso contrario si siente frustración y enojo hacia el sistema hará lo posible para no interactuar con este.

3. CAPÍTULO 3

3.1. FASE DE PLANIFICACIÓN

3.1.1. HISTORIAS DE USUARIO

Las historias de usuario se basaron en la encuesta, que se encuentra en el Anexo 1, realizada a 29 personas, a través de la encuesta se pudo obtener la información necesaria para definir los requisitos, las respuestas de la encuesta se las puede observar en los anexos 2, 3, 4 y 5, respectivamente.

Tabla 1. Historia de usuario 1

Título: Registro de gastos	Prioridad: Alta
Historia de usuario: Como usuario, deseo poder registrar cada uno de mis gastos, para poder llevar un control de estos.	
Criterios de Aceptación: • El usuario puede ingresar la cantidad del gasto. • El sistema permite seleccionar la categoría del gasto. • El usuario puede escoger una fecha. • El usuario puede ingresar una descripción del gasto. • El sistema confirma el registro del gasto.	

Fuente. Elaboración propia

Tabla 2. Historia de usuario 2

Título: Registro de ingresos	Prioridad: Alta
Historia de usuario: Como usuario, quiero poder registrar cada uno de mis ingresos, para poder ver el flujo de dinero.	
Criterios de Aceptación: • El usuario puede ingresar la cantidad del ingreso. • El sistema permite seleccionar la categoría del ingreso. • El usuario puede escoger una fecha. • El usuario puede ingresar una breve descripción del ingreso. • El sistema confirma el registro del ingreso.	

Fuente. Elaboración propia

Tabla 3. Historia de usuario 3

Título: Visualización de gastos e ingresos	Prioridad: Media
Historia de usuario: Como usuario de la aplicación, quiero visualizar todos los gastos e ingresos que he hecho, para poder observar de forma gráfica todos mis movimientos.	
Criterios de Aceptación: • El sistema muestra un resumen gráfico de los gastos e ingresos.	

Fuente. Elaboración propia

Tabla 4. Historia de usuario 4

Título: Categorías	Prioridad: Alta
Historia de usuario: Como usuario de la aplicación, quiero poder crear categorías, así como eliminarlas o editarlas	
Criterios de Aceptación: • El usuario puede crear una categoría • El usuario puede eliminar una categoría • El sistema confirma la creación o eliminación de una categoría. • El usuario puede modificar la categoría. • El sistema confirma al usuario la modificación.	

Fuente. Elaboración propia

Tabla 5. Historia de usuario 5

Título: Recordatorios de pagos	Prioridad: Alta
Historia de usuario: Como usuario de la aplicación, quiero recibir recordatorios, para poder gestionar mejor mi presupuesto.	
Criterios de Aceptación: • El usuario puede seleccionar el período de tiempo en el que quiere que el recordatorio llegue. • El usuario puede ingresar una breve descripción del pago. • El sistema confirma el registro del recordatorio. • El sistema muestra el recordatorio como una notificación.	

Fuente. Elaboración propia

Tabla 6. Historia de usuario 6

Título: Creación de reportes	Prioridad: Alta
Historia de usuario: Como usuario de la aplicación, quiero generar reportes de mis ingresos y gastos, para poder ayudarme en la planificación financiera.	
Criterios de Aceptación: • El usuario puede seleccionar el período del que desea el reporte. • El usuario puede seleccionar que categorías el reporte va a poseer. • El usuario puede descargar el reporte para verlo como PDF y hoja de cálculo.	

Fuente. Elaboración propia

3.1.2. ITERACIONES

Tabla 7. Iteración 1

Registro de datos
Objetivos: Crear la base de datos. Desarrollar las funcionalidades de registro de ingresos y gastos. Permitir la categorización de gastos e ingresos.
Tareas: Diseñar e implementar los modelos de datos para gastos e ingresos. Crear interfaces para la entrada de datos y recuperación de datos.

Fuente. Elaboración propia

Tabla 8. Iteración 2

Visualización
Objetivos: Desarrollar una visualización básica de los gastos e ingresos.
Tareas: Crear vista para mostrar los gastos e ingresos registrados. Añadir filtros básicos para visualización por fecha.

Fuente. Elaboración propia

Tabla 9. Iteración 3

Reportes diarios y por categoría
Objetivos: Implementar la generación de reportes diarios y por categoría. Permitir la generación de reportes por categoría de gastos o ingresos.
Tareas: Diseñar e implementar la funcionalidad de reportes diarios. Desarrollo de generación de reportes basados en categorías específicas. Desarrollar la descarga de reportes.

Fuente. Elaboración propia

Tabla 10. Iteración 4

Pruebas
Objetivos: Realizar pruebas exhaustivas para asegurar la fiabilidad y uso de la aplicación.
Tareas: Realizar pruebas unitarias y de integración. Recoger *feedback* de usuarios

Fuente. Elaboración propia

3.2.FASE DE DISEÑO

3.2.1. TARJETAS CRC

Las tarjetas CRC cuyas siglas significa clase, responsabilidad y colaboración, permiten la identificación de clases y sus funciones que tendrán en el sistema, así como las relaciones que tienen entre estas.

Tabla 11. Tarjeta CRC categoría

Categoría	
Responsabilidades	Colaboradores
Permitir la clasificación de transacciones Registrar información de las distintas categorías	Transacción

Fuente. Elaboración propia

Tabla 12. Tarjeta CRC transacción

Transacción	
Responsabilidades	Colaboradores
Registrar los detalles de la transacción Modificar el saldo de acuerdo con cada transacción	Categoría

Fuente. Elaboración propia

Tabla 13. Tarjeta CRC pago

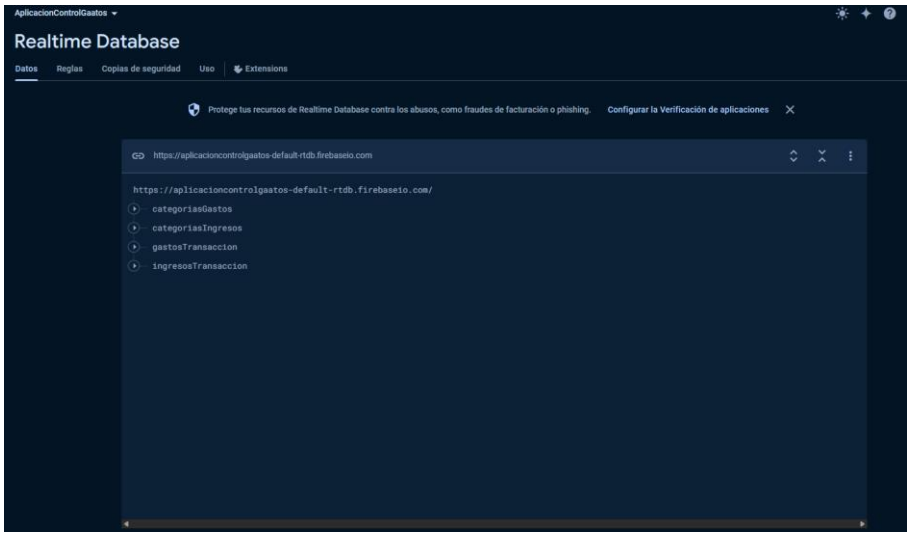
Pago	
Responsabilidades	Colaboradores
Registrar información de los pagos Notificar al usuario cuando se acerca el pago	Usuario

Fuente. Elaboración propia

3.3.FASE DE CODIFICACIÓN

3.3.1. BASE DE DATOS FIREBASE

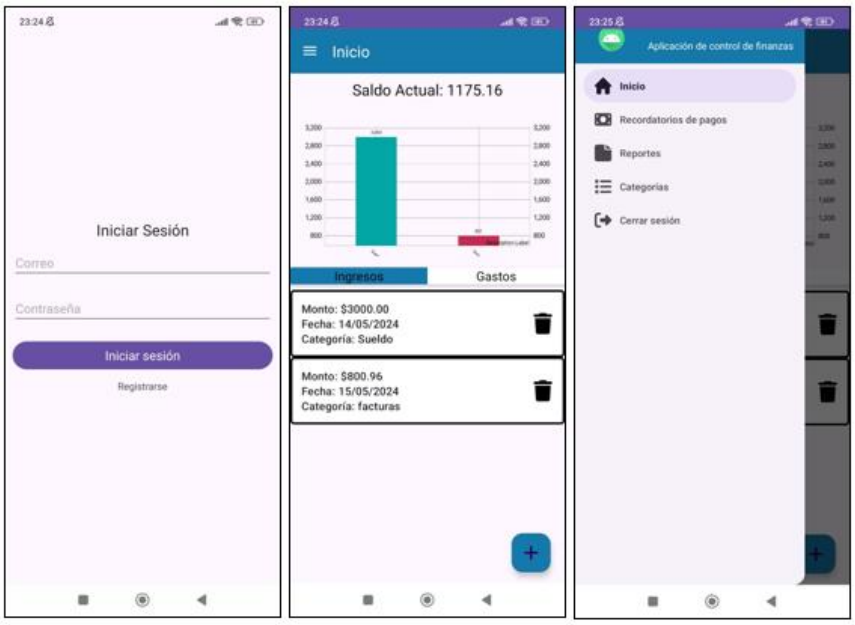
Ilustración 1. Estructura de la base de datos en la consola de Firebase



Fuente. Consola de *firebase*

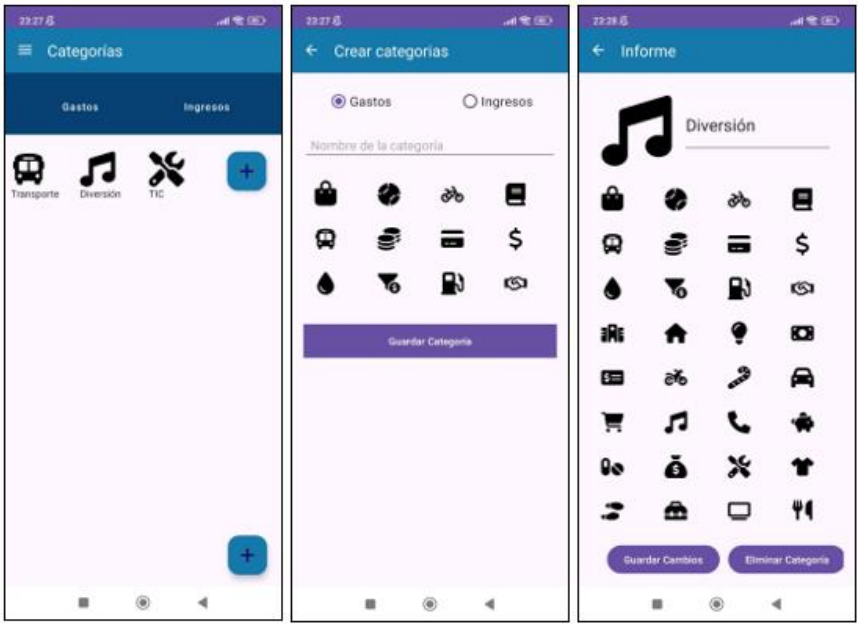
3.3.2. FLUJO DE PANTALLAS

Ilustración 2. Pantallas de inicio y menú



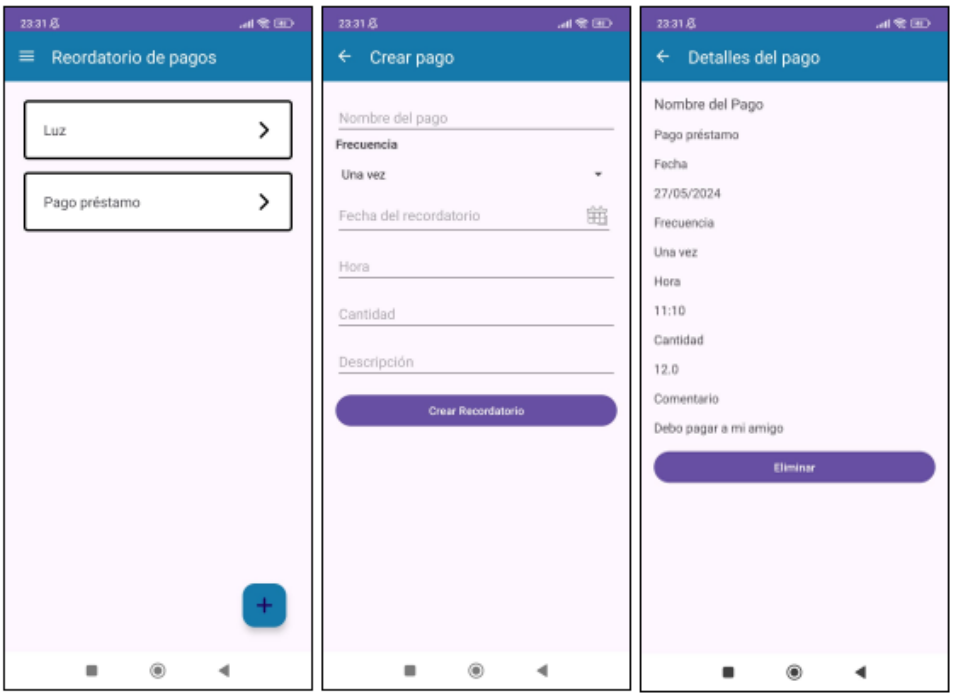
Fuente. Elaboración propia

Ilustración 3. Pantallas de categorías



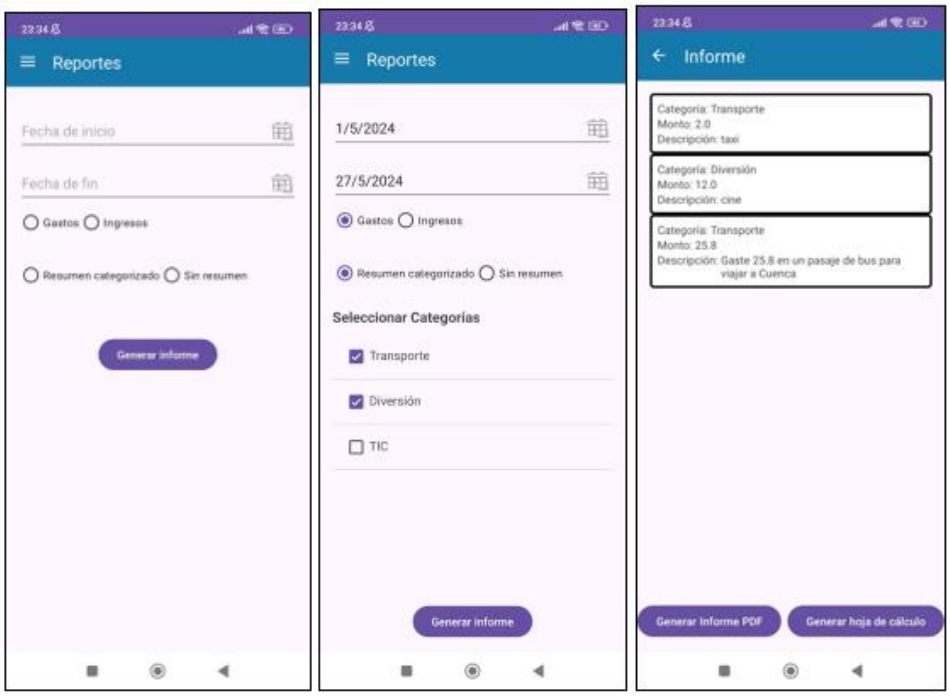
Fuente. Elaboración propia

Ilustración 4. Pantallas de pagos



Fuente. Elaboración propia

Ilustración 5. Pantallas de reportes



Fuente. Elaboración propia

3.4. PRUEBAS DE LA APLICACIÓN

3.4.1. PRUEBAS DE FUNCIONAMIENTO

Pruebas en base a la historia de usuario 1: Registro de gastos

- Prueba para registrar un gasto

  Descripción

  Verificar la creación correcta de gastos en la aplicación y que se almacenen correctamente en la base de datos.

  Expectativas

  Se espera que el gasto se guarde correctamente y aparezca en la lista de gastos.

  Resultados

  Ilustración 6. Pantalla para crear un gasto



  Nota. Pantalla de registro de gastos, el usuario ingresa los datos deseados y presiona el botón guardar.

  Fuente. Elaboración propia

Ilustración 7. Mensaje exitoso al crear un gasto



Nota. Mensaje de creación exitoso del gasto.

Fuente. Elaboración propia

Ilustración 8. Gasto creado en la lista



Nota. Verificación de la creación del gasto en la lista de gastos.

Fuente. Elaboración propia

- Prueba para la eliminación de un gasto

Descripción

Verificar la eliminación correcta de gastos en la aplicación.

Expectativas

Se espera que el gasto se elimine correctamente y ya no aparezca en la base de datos.

Resultados

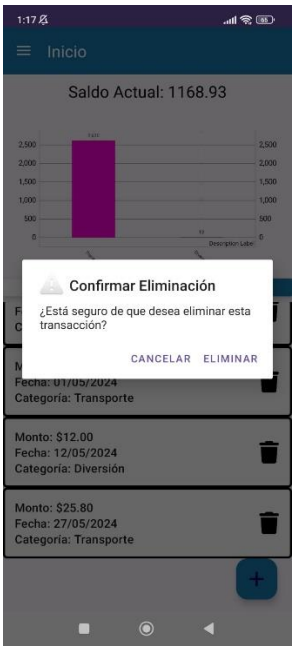
Ilustración 9. Gasto seleccionado para eliminar



Nota. Gasto seleccionado para verificar la eliminación de gastos

Fuente. Elaboración propia

Ilustración 10. Confirmación de la eliminación del gasto



Nota. Se muestra un mensaje de confirmación para eliminar un gasto.

Fuente. Elaboración propia

Ilustración 11. Mensaje de eliminación exitosa



Nota. Mensaje de confirmación de que se ha eliminado el gasto y no se encuentra en la lista el gasto eliminado.

Fuente. Elaboración propia

Pruebas en base a la historia de usuario 2: Registro de ingresos

- Prueba para registrar un ingreso

  Descripción

  Verificar la creación correcta de ingresos en la aplicación y que se almacenen correctamente en la base de datos.

  Expectativas

  Se espera que el ingreso se guarde correctamente y aparezca en la lista de ingresos.

  Resultados

  Ilustración 12. Pantalla para crear un ingreso



  Nota. Pantalla de registro de ingresos, el usuario ingresa los datos deseados y presiona el botón guardar.

  Fuente. Elaboración propia

Ilustración 13. Confirmación de la creación del ingreso



Nota. Mensaje que confirma la creación del ingreso y verificación que existe en la lista.

Fuente. Elaboración propia

- Prueba para la eliminación de un ingreso

  Descripción

  Verificar la eliminación correcta de ingresos en la aplicación.

  Expectativas

  Se espera que el ingreso se elimine correctamente y aparezca en la lista de ingresos.

  Resultados

Ilustración 14. Ingreso seleccionado para eliminar



Nota. Ingreso seleccionado para eliminar.

Fuente. Elaboración propia

Ilustración 15. Mensaje para confirmar la eliminación



Nota. Al pulsar el botón para eliminar aparece un diálogo de confirmación para realizar la acción.

Fuente. Elaboración propia

Ilustración 16. Mensaje de eliminación exitosa



Nota. Se puede observar el mensaje de eliminación exitosa y que el ingreso ya no está en la lista.

Fuente. Elaboración propia

Pruebas en base a la historia de usuario 3: Visualización de gastos e ingresos

Descripción

El usuario puede observar un gráfico de barras de los gastos o ingresos de acuerdo con las categorías.

Expectativas

Se espera que el usuario observe el gráfico donde se muestran los gastos e ingresos.

Resultados

Ilustración 17. Visualización de ingresos



Nota. Gráfico de los ingresos.

Fuente. Elaboración propia

Ilustración 18. Visualización de gastos



Nota. Gráfico de gastos.

Fuente. Elaboración propia

Pruebas en base a la historia de usuario 4: Categorías

- Prueba de creación de categoría

  Descripción

  El usuario puede crear una categoría.

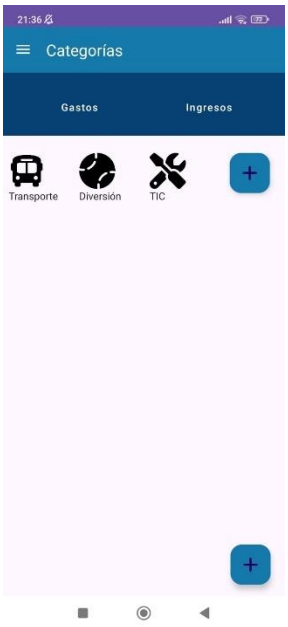
  Expectativas

  La categoría se crea exitosamente y se muestra en la lista de categorías.

  El sistema muestra un mensaje confirmando la creación de la categoría.

  Resultados

Ilustración 19. Pantalla de categorías



Nota. El usuario selecciona el botón con un más para crear una nueva categoría

Fuente. Elaboración propia

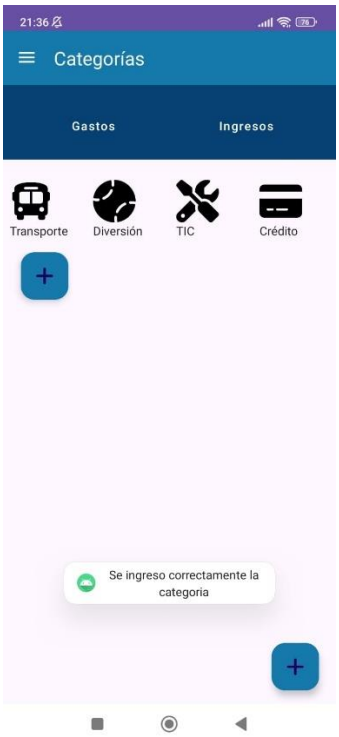
Ilustración 20. Creación de categoría



Nota. El usuario ingresa los datos necesarios y oprime el botón guardar categoría

Fuente. Elaboración propia

Ilustración 21. Mensaje de confirmación de creación de categoría



Nota. El sistema muestra un mensaje de confirmación y se elimina de la lista

Fuente. Elaboración propia

- Prueba de eliminación de categoría

Descripción

El usuario puede eliminar una categoría que no esté asociada a un gasto o ingreso.

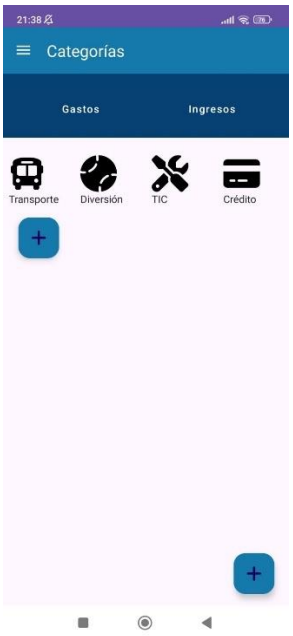
Expectativas

La categoría se elimina exitosamente y se deja de mostrar en la lista de categorías.

El sistema muestra un mensaje confirmando la eliminación de la categoría.

Resultados

Ilustración 22. Lista de categorías



Nota. El usuario selecciona la categoría

Fuente. Elaboración propia

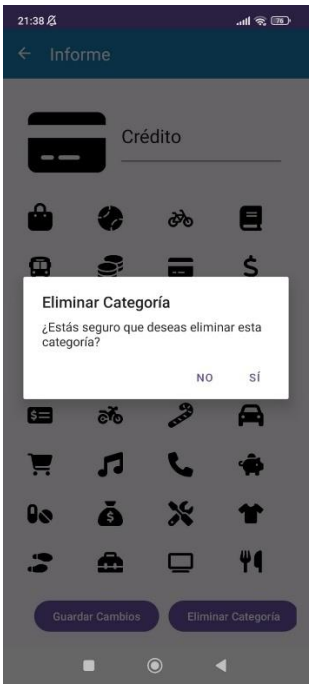
Ilustración 23. Pantalla para eliminar una categoría



Nota. El sistema abre una pantalla donde se puede modificar o eliminar la categoría

Fuente. Elaboración propia

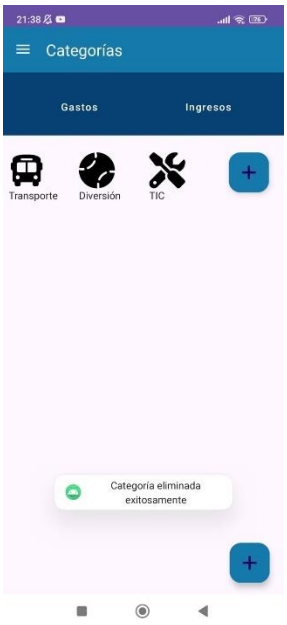
Ilustración 24. Mensaje de confirmación para eliminar una categoría



Nota. Al pulsar eliminar se abre un dialogo para confirmar la acción

Fuente. Elaboración propia

Ilustración 25. Mensaje que confirma la eliminación de la categoría



Nota. Se confirma la eliminación con un mensaje y se elimina de la lista

Fuente. Elaboración propia

- Prueba de modificación de categoría

  Descripción

  El usuario puede eliminar una categoría que no esté asociada a un gasto o ingreso.

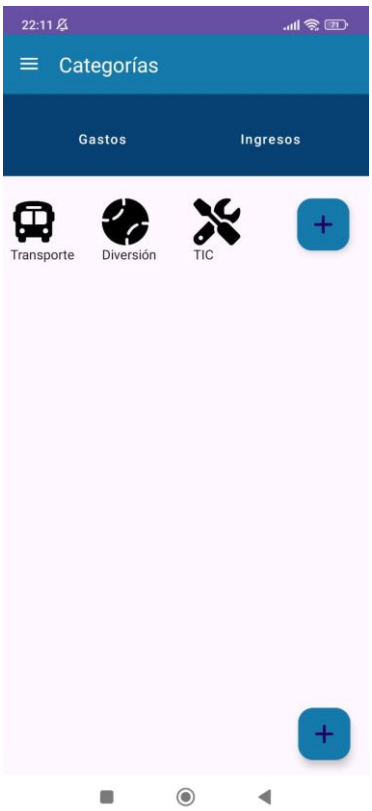
  Expectativas

  La categoría se elimina exitosamente y se deja de mostrar en la lista de categorías.

  El sistema muestra un mensaje confirmando la eliminación de la categoría.

  Resultados

  Ilustración 26. Lista de categorías de gastos



  Nota. El usuario escoge la categoría a modificar

  Fuente. Elaboración propia

Ilustración 27. Pantalla para modificar una categoría



Nota. El usuario cambia lo que desea de la categoría y guarda los cambios

Fuente. Elaboración propia

Ilustración 28. Mensaje que confirma la modificación de la categoría



Nota. Se muestra un mensaje de confirmación y se cambia en la lista

Fuente. Elaboración propia

Pruebas en base a la historia de usuario 5: Recordatorios de pagos

- Prueba de registro de recordatorio

  Descripción

  El usuario puede crear un recordatorio de pago a través de la aplicación.

  Expectativas

  El recordatorio se crea exitosamente y el sistema muestra una notificación según el recordatorio creado.

  Resultados

  Ilustración 29. Pantalla para recordatorios



  Nota. Pantalla de los pagos aquí el usuario selecciona el botón con un más para crear un nuevo pago.

  Fuente. Elaboración propia

Ilustración 30. Pantalla para crear un recordatorio



Nota. El usuario ingresa sus datos y pulsa el botón de guardar.

Fuente. Elaboración propia

Ilustración 31. Mensaje que confirma la creación del pago



Nota. Se confirma la creación del recordatorio y se puede observar en la lista de recordatorios.

Fuente. Elaboración propia

Ilustración 32. Notificación del recordatorio



Nota. Notificación del recordatorio en el dispositivo.

Fuente. Elaboración propia

- Prueba de eliminación de recordatorio

  Descripción

  El usuario puede eliminar un recordatorio de pago a través de la aplicación.

  Expectativas

  El recordatorio se elimina exitosamente y el sistema se lo confirma al usuario.

  Resultados

Ilustración 33. Lista de recordatorios



Nota. Se selecciona el pago que desea eliminar.

Fuente. Elaboración propia

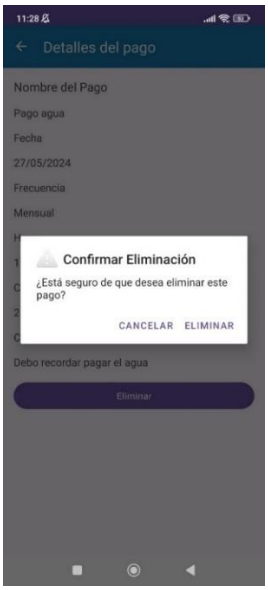
Ilustración 34. Pantalla del recordatorio del pago



Nota. La aplicación abre el detalle del pago y se selecciona el botón eliminar.

Fuente. Elaboración propia

Ilustración 35. Confirmación de la eliminación del recordatorio



Nota. Mensaje de confirmación para eliminar.

Fuente. Elaboración propia

Ilustración 36. Mensaje que confirma la eliminación del recordatorio



Nota. Se muestra el mensaje de pago eliminado y se puede observar que ya no se encuentra en la lista.

Fuente. Elaboración propia

Pruebas en base a la historia de usuario 6: Creación de reportes

- Prueba de generación y descarga del reporte en PDF

  Descripción

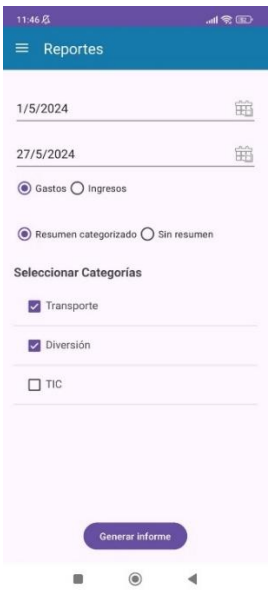
  El usuario puede seleccionar un período de tiempo, si desea un reporte de gastos o ingresos, si quiere un reporte categorizado o un resumen de todas las categorías y seleccionar las categorías que desea, generar el reporte y descargarlo en formato PDF.

  Expectativas

  El usuario puede generar un reporte según los criterios que este desea y lo puede descargar en formato PDF.

  Resultados

  Ilustración 37. Pantalla de reportes



  Nota. Se ingresan los detalles del reporte y se presiona generar informe.

  Fuente. Elaboración propia

Ilustración 38. Resumen del reporte en la aplicación



Nota. Al generar el informe saldrá una pestaña en la que se puede observar la lista de transacciones

Fuente. Elaboración propia

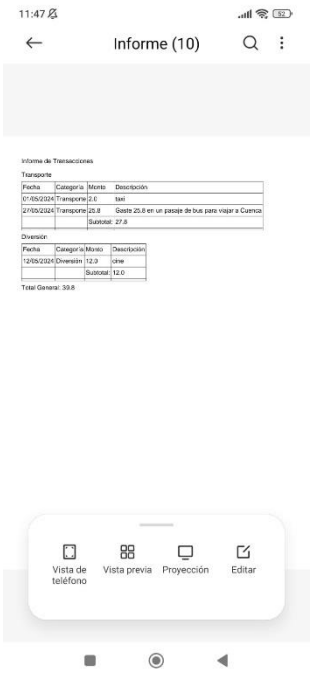
Ilustración 39. Notificación de la creación del archivo PDF



Nota. Después de pulsar el botón generar informe PDF se muestra la notificación del informe.

Fuente. Elaboración propia

Ilustración 40. Informe PDF



Nota. Al dar clic en la notificación se abrirá el informe PDF.

Fuente. Elaboración propia

- Prueba de generación y descarga del reporte en hoja de cálculo

Descripción

El usuario puede seleccionar un período de tiempo, si desea un reporte de gastos o ingresos, si quiere un reporte categorizado o un resumen de todas las categorías y seleccionar las categorías que desea, generar el reporte y descargarlo en formato hoja de cálculo.

Expectativas

El usuario puede generar un reporte según los criterios que este desea y lo puede descargar en formato hoja de cálculo.

Resultados

Ilustración 41. Pantalla para crear reportes



Nota. Se ingresan los detalles del reporte y se presiona generar informe.

Fuente. Elaboración propia

Ilustración 42. Lista del reporte



Nota. Se puede observar la lista de los ingresos que cumplen con los requisitos.

Fuente. Elaboración propia

Ilustración 43. Notificación hoja de cálculo



Nota. Se muestra la notificación de que el archivo ha sido creado.

Fuente. Elaboración propia

Ilustración 44. Hoja de cálculo generada



Nota. Se puede observar la hoja de cálculo generada.

Fuente. Elaboración propia

3.4.2. PRUEBAS DE RENDIMIENTO

Prueba para evaluar el rendimiento de la inserción masiva de datos en *Firebase Realtime Database*

Entorno de prueba

- Base de datos: *Firebase Realtime Database*
- Herramienta: Script de Node.js
- Cantidad de registros: 10000

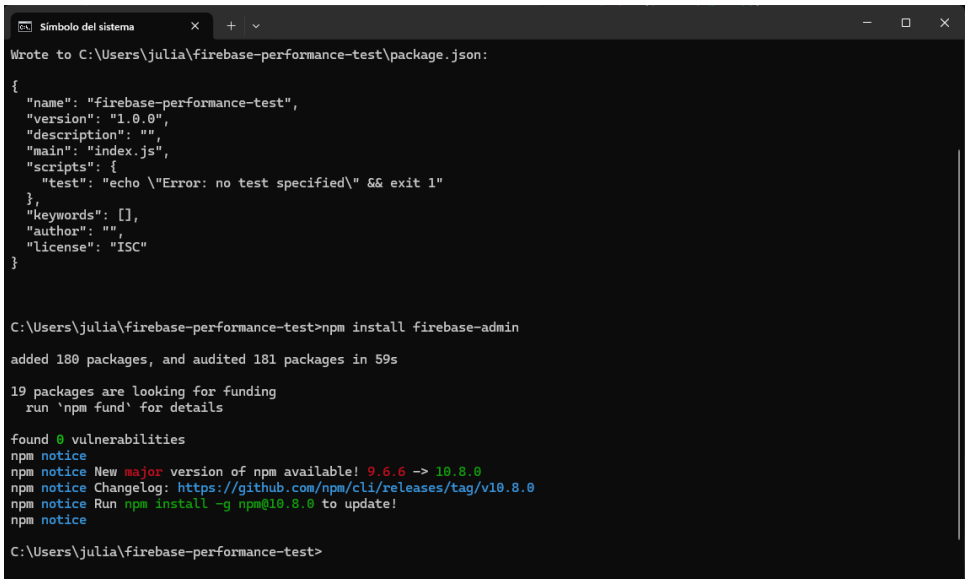
Resultados

El tiempo total de la inserción de los 10000 registros fue de 16 minutos y la latencia que existe entre cada inserción es de 99 a 150 milisegundos, haciendo este un tiempo de respuesta aceptable.

La consistencia de las respuestas del servidor se puede observar que es bastante buena, ya que la latencia de cada inserción está entre 99 y 150 milisegundos, lo que indica una respuesta por parte de la base de datos consistente.

Con respecto al tiempo de inserción, se considera alto, sin embargo, el trabajo que hace la aplicación no necesita de carga masiva de datos, por lo que el tiempo de ejecución al insertar un registro puede considerarse bueno.

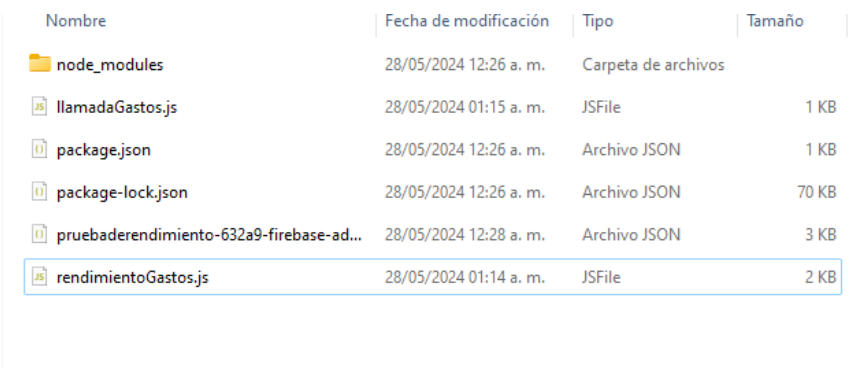
Ilustración 45. Instalación de firebase-admin



Nota. Instalación de *firebase-admin* para realizar las pruebas de rendimiento.

Fuente. Elaboración propia

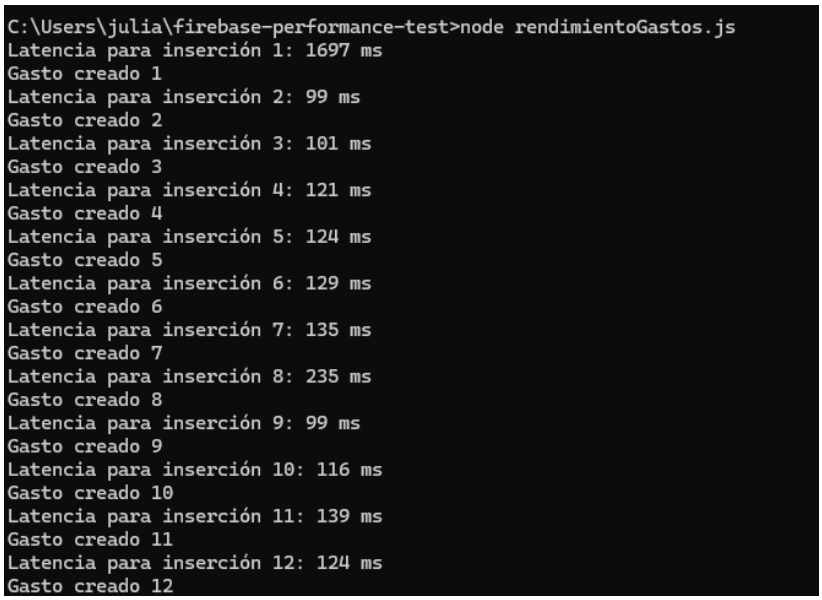
Ilustración 46. Script para realizar la prueba de inserción



Nota. Archivo de *javascript* para realizar la prueba, el código se encuentra ubicado en el anexo 6.

Fuente. Elaboración propia

Ilustración 47. Resultados del script



Nota. Primeros 12 registros creados.

Fuente. Elaboración propia

Prueba para evaluar el rendimiento al hacer consultas de datos en *Firebase Realtime Database*

Entorno de prueba

- Base de datos: Firebase Realtime Database
- Herramienta: Script de Node.js

Resultados

El tiempo total de la consulta fue de 1.939 segundos, por lo que el rendimiento de lectura se podría decir que es el adecuado con respecto a la cantidad de registros que existen, sin embargo, el tiempo también depende del tráfico de la red existente pues al ser un servicio en la nube este depende de la conexión a internet que se posee, por lo que sería recomendable que la conexión a internet sea la mejor posible para el funcionamiento correcto de la aplicación.

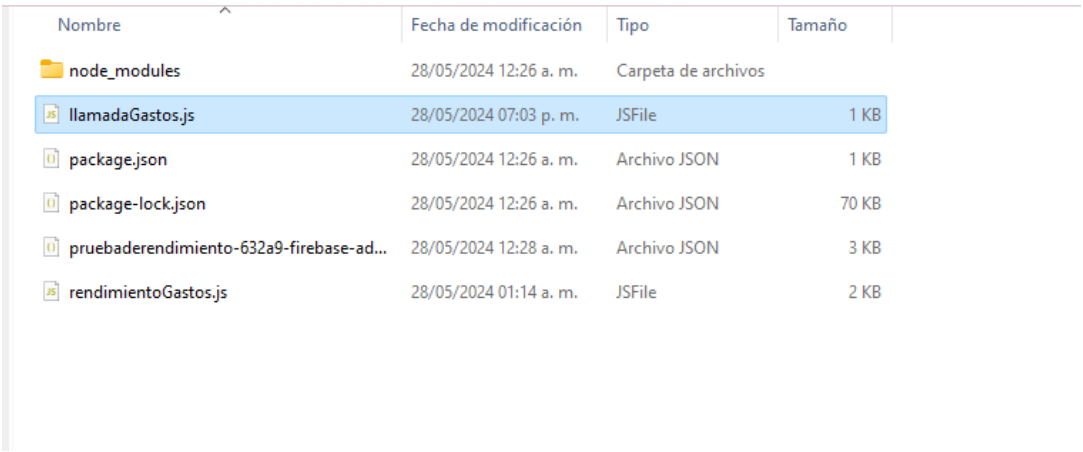
Ilustración 48. Respuesta del script

```
C:\Users\julia\firebase-performance-test>node llamadaGastos.js
generarReporte: 1.939s
Total: 5095184
```

Nota. Se muestra el tiempo en que tarda la consulta

Fuente. Elaboración propia

Ilustración 49. Script para realizar la prueba de consulta



Nota. Este es el script en *javascript* para realizar la prueba, el código se encuentra ubicado en el anexo 7.

Fuente. Elaboración propia

4. CONCLUSIONES Y RECOMENDACIONES

4.1. CONCLUSIONES

- El uso de una encuesta para el levantamiento de requerimientos fue esencial para el desarrollo de la aplicación de control de gastos personales. Este enfoque permitió identificar tanto los requerimientos funcionales como no funcionales para crear la aplicación, proporcionando así una guía para la creación de historias de usuarios, que ayudaron a orientar el diseño y

funcionalidad del sistema para poder cumplir con las expectativas que los usuarios tenían en mente.

- La implementación de Firebase para la autenticación de usuarios y manejo de base de datos permitió una mejor gestión de la aplicación en lo que se refiere a tiempos de respuesta y sincronización, pues al ser una herramienta en la nube, provee servicios veloces de almacenamiento y recuperación de datos. Además, se preparó el sistema para manejar grandes volúmenes de usuarios y de datos a futuro, asegurando así la escalabilidad y funcionamiento de la aplicación a largo plazo.
- La identificación y corrección de errores relacionados con permisos para escribir y leer archivos y el envío adecuado de notificaciones representan componentes cruciales para la optimización del desarrollo de una aplicación. Pues mejoran la seguridad y la funcionalidad de la aplicación, garantizando así que funcione de manera correcta.
- El sistema de arrastrar y soltar componentes y las previsualizaciones en tiempo real de la interfaz gráfica de Android Studio, han sido unas herramientas indispensables para el desarrollo de aplicaciones, permitiendo así un proceso más rápido, eficiente y visualmente intuitivo para la creación de pantallas dentro de la aplicación.
- El uso de librerías externas como ItextPDF y EasyExcel es esencial para la implementación de funcionalidades como la generación de reportes en PDF y hojas de cálculo, pues facilitan el desarrollo y brindan flexibilidad al crear informes en diversos formatos, que permiten aportar un valor significativo a la aplicación.

- La realización de pruebas de funcionalidad fue de suma importancia en la creación de la aplicación, pues estas permitieron identificar y corregir errores que podrían haber comprometido la funcionalidad del sistema. Así pues, de esta forma se aseguró que la aplicación cumpliera con los requisitos establecidos en las historias de usuario y la creación de una aplicación más confiable y segura.

4.2. RECOMENDACIONES

- Realizar pruebas con una gran cantidad de usuarios sería lo ideal para obtener una retroalimentación correcta sobre la aplicación, ya sea con respecto al diseño o a la funcionalidad, pues esta retroalimentación permitirá realizar cambios a la aplicación con el fin de que los usuarios se sientan más atraídos al uso de la aplicación.
- Se recomienda realizar una investigación bastante completa sobre las librerías que se pueden usar para implementar ciertas funcionalidades, pues esto permitirá determinar cuáles se pueden usar y son compatibles con las configuraciones de la aplicación y ayudarían a agilizar el proceso de desarrollo.
- Ampliar la encuesta para el levantamiento de requerimientos sería de gran ayuda para obtener más información, pues se pueden hacer preguntas más específicas de acuerdo con ciertas funcionalidades, permitiendo así identificar de mejor manera la expectativa que el usuario tiene respecto a cierta funcionalidad.
- Seguir cursos o leer la documentación del IDE de desarrollo, en este caso Android Studio, puede ser de gran importancia para saber cómo aprovechar sus características al máximo, pues con el conocimiento adquirido se podrán

crear aplicaciones innovadoras y competitivas, con grandes funcionalidades que atraigan a diversos usuarios.

5. BIBLIOGRAFÍA

Domingo, M. G., & Pera, E. M. (2010). Diseño centrado en el usuario. *Universitat Oberta de Catalunya*, 9-12.

Firebase Authentication. (s. f.). Firebase. https://firebase.google.com/docs/auth?hl=es

Firebase expands to become a unified app platform. (s. f.). The Firebase Blog. https://firebase.blog/posts/2016/05/firebase-expands-to-become-unified-app-platform

Firebase Realtime database. (s. f.). Firebase. https://firebase.google.com/docs/database?hl=es

International Organization for Santadarization. (2019). *Ergonomics of human-system interaction — Part 220: Processes for enabling, executing and assessing human-centred design within organizations* (ISO 9241-220:2019). https://www.iso.org/obp/ui/#iso:std:iso:9241:-220:ed-1:v1:en

Introducción a Android Studio. (s. f.). Android Developers. https://developer.android.com/studio/intro?hl=es-419

Joskowicz, J. (2008). Reglas y prácticas en eXtreme Programming. *Universidad de Vigo*, *22*.

Maida, E. G., & Pacienzia, J. (2015). Metodologías de desarrollo de software.

Morville, P. (2005). *Ambient findability: What we find changes who we become*. " O'Reilly Media, Inc.".

Tomás, J., Carbonell, V., Bataller, J., & Lloret, J. (2019). *Firebase: trabajar en la nube*. Alpha Editorial.

6. ANEXOS

Anexo 1

Encuesta

El presente cuestionario tiene fines académicos, la información recopilada en este formulario es de carácter anónimo y será utilizada únicamente para la elaboración de mi proyecto de trabajo de integración curricular, agradezco la sinceridad en las respuestas que me provea.

- ¿Cuáles son sus gastos diarios?
- ¿Qué problemas tiene al administrar sus gastos?
- ¿Lleva algún registro de sus gastos?
- ¿Ha usado aplicaciones que le permiten administrar tu dinero?
    - ¿Qué le gusta de la aplicación?
    - ¿Qué hace la aplicación para ayudar a manejar sus finanzas?

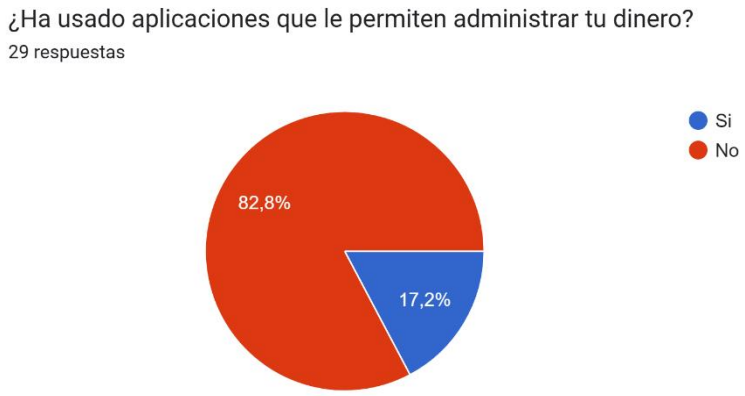
El proyecto tiene como finalidad la realización de una aplicación para el control de gastos, las siguientes preguntas tienen como finalidad saber su opinión sobre las funciones que esta aplicación podría tener

- De los siguientes aspectos, ¿cuál le gustaría registrar en la aplicación? (Puede seleccionar varias respuestas)
    - Gastos diarios (Gasto todos los días en bus 0.35 ctvs)
    - Gastos ocasionales (Hoy comí una salchipapa)
    - Deudas (Debo pagar a la tarjeta $15)
    - Ingresos (Mi mesada es de $10)
- Que le gustaría que la aplicación realizara (Puede seleccionar varias respuestas)
    - Registro de gastos
    - Registro de ingresos
    - Categorización de gastos (salud, ocio)
    - Visualización de gastos (gráficos, informes)
    - Recordatorio de pagos

- En el informe que la aplicación genera, ¿qué información crees que es relevante? (Puede seleccionar varias respuestas)
    - Los gastos categorizados al día a día, mensual o anual
    - Reportes diarios
    - Reportes por categoría
    - Otro tipo de reporte... (especifique cuál)
- ¿Hay algún otro aspecto o funcionalidad que consideres importante para incluir en esta aplicación?

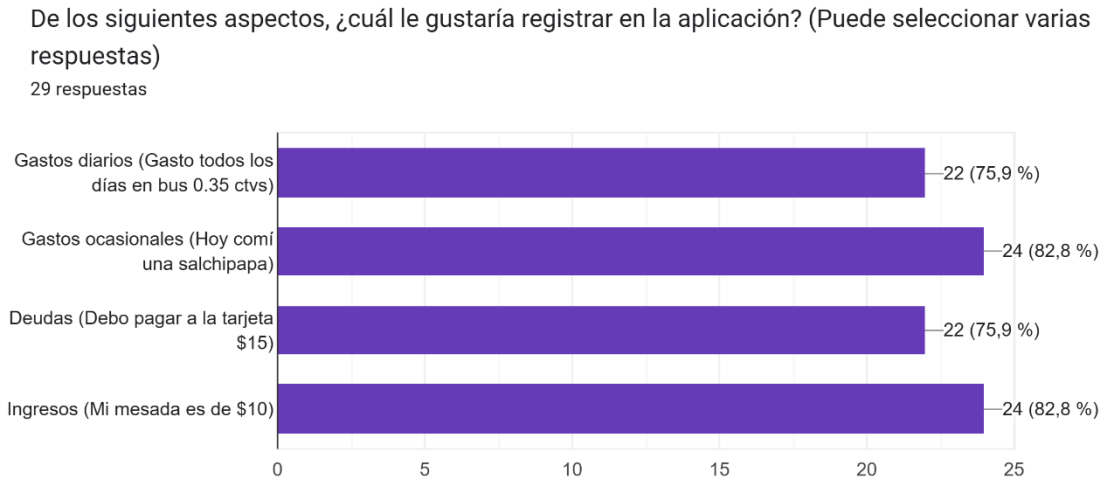
Anexo 2

Respuestas de la encuesta pregunta 4



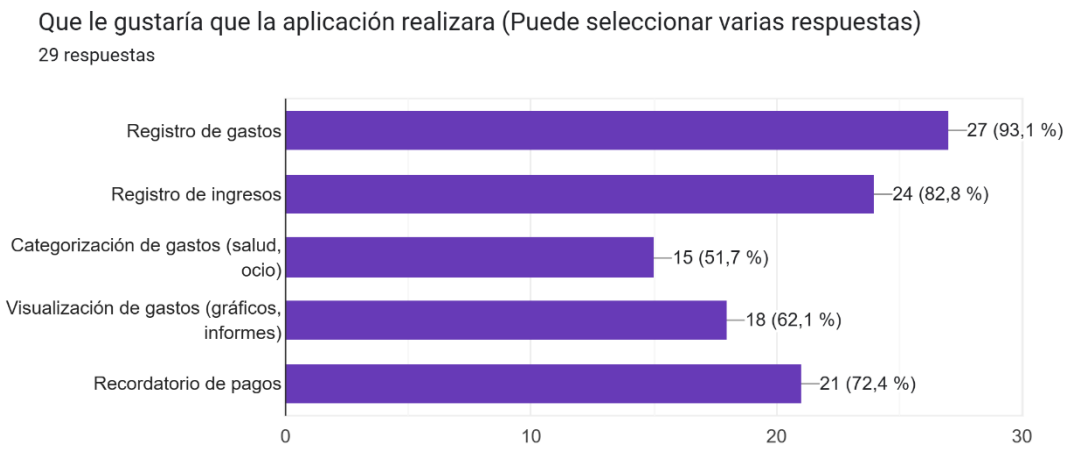
Anexo 3

Respuestas de la encuesta pregunta 7



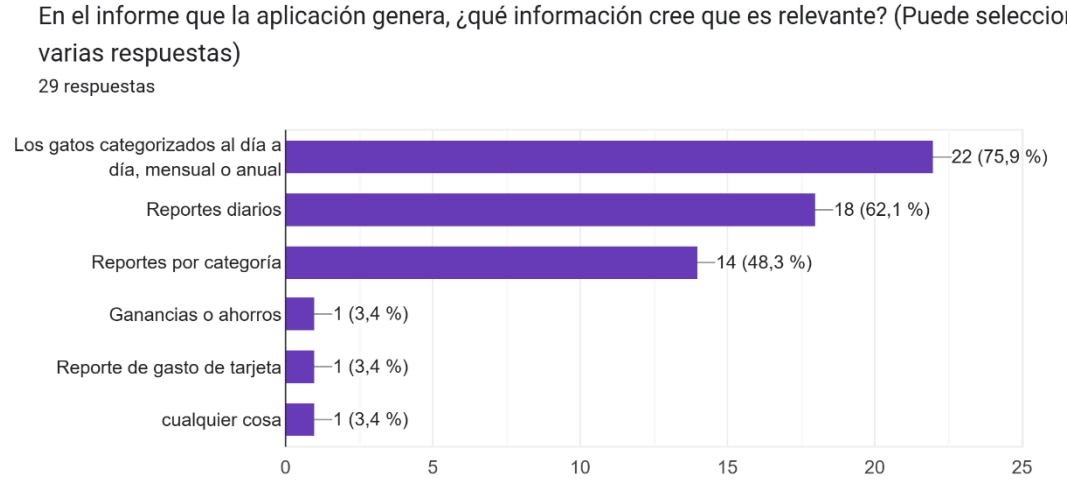
Anexo 4

Respuestas de la encuesta pregunta 8



Anexo 5

Respuestas de la encuesta pregunta 9



Anexo 6

Script para pruebas de inserción

rendimientoGastos.js

C: > Users > julia > firebase-performance-test > rendimientoGastos.js > ...

```
const admin = require('firebase-admin');
const serviceAccount = require('./pruebaderendimiento-632a9-firebase-adminsdk-2r2ac-b60790b7cb.json');

admin.initializeApp({
  credential: admin.credential.cert(serviceAccount),
  databaseURL: 'https://pruebaderendimiento-632a9-default-rtdb.firebaseio.com'
});

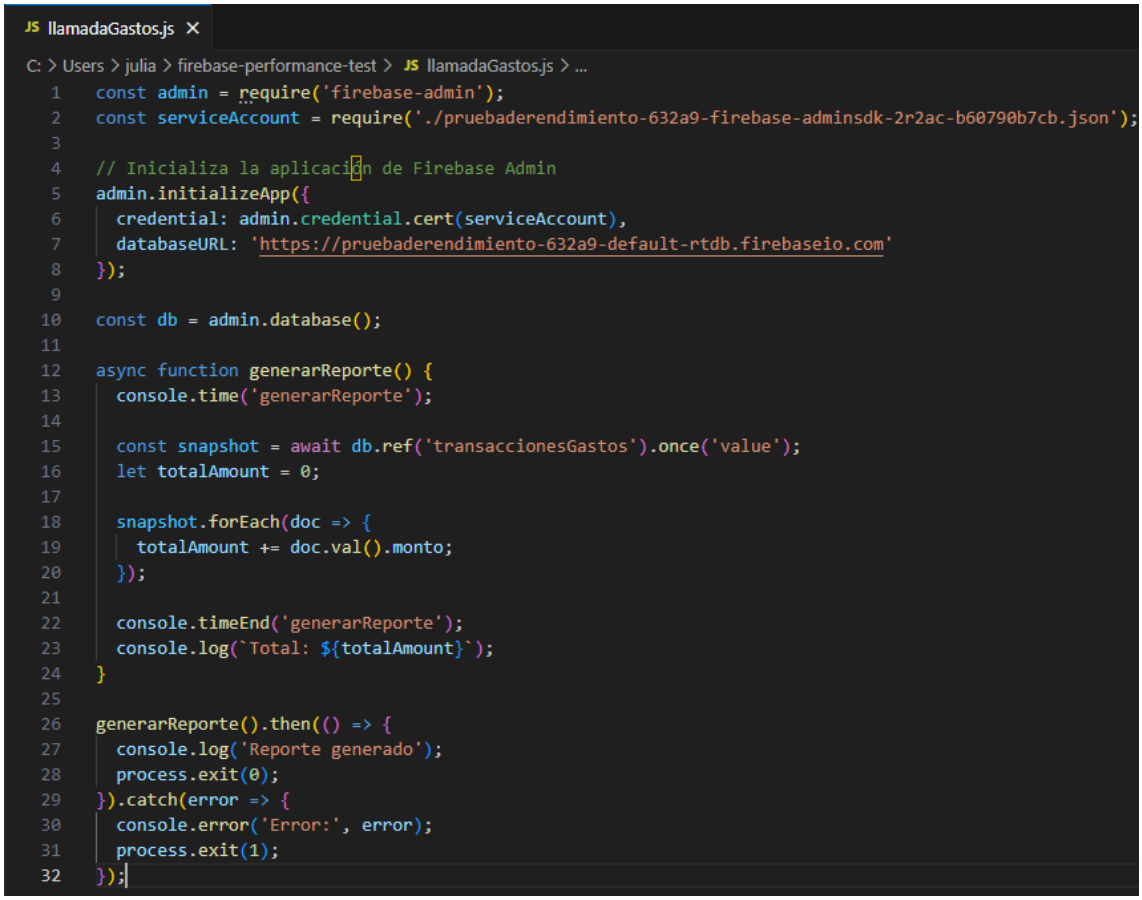
const db = admin.database();

async function crearGastos(num) {
  for (let i = 0; i < num; i++) {
    const start = Date.now();
    await db.ref('transaccionesGastos').push({
      monto: Math.floor(Math.random() * 1000),
      categoria: 'Alimentación',
      fecha: new Date().toISOString(),
      comentario: 'Cena en restaurante',
      tipo:'gasto'
    });
    const end = Date.now();
    const latency = end - start;
    console.log(`Latencia para inserción ${i + 1}: ${latency} ms`);
    console.log(`Gasto creado ${i + 1}`);
  }
  console.log('Se acabaron de registrar los gastos');
}


// Ejecución de las funciones de prueba
crearGastos(10000).then(() => {
    console.log('Se acabo de insertar la data');
    process.exit(0);
  }).catch(error => {
    console.error('Error:', error);
    process.exit(1);
  });
```

Anexo 7

Script para la consulta de datos



```
JS llamadaGastos.js
C: > Users > julia > firebase-performance-test > JS llamadaGastos.js > ...
1   const admin = require('firebase-admin');
2   const serviceAccount = require('./pruebaderendimiento-632a9-firebase-adminsdk-2r2ac-b60790b7cb.json');
3
4   // Inicializa la aplicación de Firebase Admin
5   admin.initializeApp({
6     credential: admin.credential.cert(serviceAccount),
7     databaseURL: 'https://pruebaderendimiento-632a9-default-rtdb.firebaseio.com'
8   });
9
10  const db = admin.database();
11
12  async function generarReporte() {
13    console.time('generarReporte');
14
15    const snapshot = await db.ref('transaccionesGastos').once('value');
16    let totalAmount = 0;
17
18    snapshot.forEach(doc => {
19      totalAmount += doc.val().monto;
20    });
21
22    console.timeEnd('generarReporte');
23    console.log(`Total: ${totalAmount}`);
24  }
25
26  generarReporte().then(() => {
27    console.log('Reporte generado');
28    process.exit(0);
29  }).catch(error => {
30    console.error('Error:', error);
31    process.exit(1);
32  });
```