

# **APLICACIÓN MÓVIL PARA EL ALQUILER DE SALONES DE EVENTOS**

Directores:

Dianalin Neme Prada

Iván Rodrigo Romero Flórez

Jenny Marcela Candelo & Brayan Steven Herreño

Mayo 2021

Universidad Antonio Nariño.

Facultad de Ingeniería de Sistemas.

Especialización en Ingeniería de Software

## CONTENIDO

1. Formulación y descripción del problema .....	8
2. Objetivo General.....	10
3. Objetivos Específicos .....	10
4. Marco de referencia .....	11
4.1. Estado del arte.....	11
4.2. Impacto .....	14
4.3. Componente de Innovación .....	15
4.4. Marco teórico.....	15
5. Metodología.....	28
6. proceso de software .....	30
6.1. Requerimientos funcionales .....	30
6.2. Requerimientos no funcionales .....	34
6.3. Diseño y arquitectura .....	36
6.3.1. Diagrama de despliegue.....	36
6.3.2. Diagrama de uso arquitecturalmente relevante.....	36
6.3.3. Diagrama de secuencia .....	40
6.3.4. Diagrama de clases .....	45
6.3.5. Arquitectura de alto nivel.....	46

	3
7. Construcción.....	48
7.1. Creación de proyecto en Firebase.....	48
7.2. Creación de base de datos con Firestore.....	57
7.3. Creación de arquitectura y desarrollo en Android Studio .....	60
8. Pruebas.....	63
9. Instalación y configuración .....	81
10. Conclusiones.....	84
11. Referencias .....	85

## Listado de tablas

Tabla 1.	Requerimiento Funcional 1 .....	30
Tabla 2.	Requerimiento Funcional 2 .....	31
Tabla 3.	Requerimiento Funcional 3 .....	31
Tabla 4.	Requerimiento Funcional 4 .....	32
Tabla 5.	Requerimiento Funcional 5 .....	33
Tabla 6.	Requerimiento Funcional 6 .....	34
Tabla 7.	Requerimiento no Funcional 1 .....	34
Tabla 8.	Requerimiento no Funcional 2 .....	35
Tabla 9.	Tabla actores del sistema .....	37
Tabla 10.	Escenarios de calidad.....	63
Tabla 11.	Tabla de riesgos de seguridad.....	63
Tabla 12.	Tabla de controles a los riesgos de seguridad.....	65

## Lista de figuras

Figura 1.	Logo Ezpot. ....	11
Figura 2.	Logo TSEventos. ....	11
Figura 3.	Logo Regus.....	12

	5
Figura 4. Logo Banquete.com.co.....	12
Figura 5. Logo HolaPlace. ....	13
Figura 6. Logo VenuesPlace. ....	13
Figura 7. Logo Wimet.....	14
Figura 8. Arquitectura LINT .....	27
Figura 9. Diagrama de despliegue.....	36
Figura 10. Diagrama de casos de uso.....	37
Figura 11. Diagrama de secuencia Iniciar Sesión .....	41
Figura 12. Diagrama de secuencia Realizar búsqueda.....	42
Figura 13. Diagrama de secuencia Añadir a carrito de compras .....	43
Figura 14. Diagrama de secuencia Realizar pago .....	44
Figura 15. Diagrama de clases .....	45
Figura 16. Diagrama de arquitectura.....	46
Figura 17. Creación del proyecto en Firebase.....	48
Figura 18. Creación del proyecto en Firebase.....	49
Figura 19. Habilitar los tipos de login (Facebook y Google).....	50
Figura 20. Configurar login con Google .....	51
Figura 21. Configurar login con Google .....	51
Figura 22. Habilitar y configurar login con Facebook.....	52

	6
Figura 23. Creación de proyecto en modo desarrollador de Facebook.....	53
Figura 24. Configuración del proyecto en Facebook.....	53
Figura 25. Configuración básica para el proyecto en Facebook .....	53
Figura 26. Registro de la llave SHA1 y el Package .....	54
Figura 27. Configuración de permiso y comunicación con Firebase.....	55
Figura 28. Configuración de archivo para permiso y comunicación con Firebase.....	56
Figura 29. Llamado del archivo Gradle del proyecto .....	56
Figura 30. Llamado de las dependencias en el archivo Gradle del proyecto.....	57
Figura 31. Creación de base de datos con Firestore.....	57
Figura 32. Habilitar e implementar bases de dos Firestore .....	58
Figura 33. Configuración de acceso a la base de datos.....	58
Figura 34. Habilitar e implementar store .....	59
Figura 35. Desarrollo en Android Studio.....	60
Figura 36. Desarrollo en Android Studio.....	60
Figura 37. Desarrollo en Android Studio.....	61
Figura 38. Pruebas en LINT.....	67
Figura 39. Prueba de accesibilidad .....	67
Figura 40. Prueba de exactitud.....	69
Figura 41. Pruebas de internacionalización .....	71

	7
Figura 42. Prueba de desempeño .....	73
Figura 43. Pruebas de seguridad .....	74
Figura 44. Prueba de usabilidad .....	75
Figura 45. Pruebas en Inspector para JAVA.....	76
Figura 46. Estructura de clases .....	76
Figura 47. Madurez de código .....	76
Figura 48. declaración de redundancia .....	77
Figura 49. IMPORTS .....	78
Figura 50. Lenguaje JAVA nivel de migración .....	78
Figura 51. Memoria.....	79
Figura 52. Probables Bugs .....	79
Figura 53. Redundancia .....	80
Figura 54. Archivo APK .....	81
Figura 55. Ejecución del APK .....	82
Figura 56. Logo EventApp.....	82
Figura 57. Pantalla de login .....	83

## 1. Formulación y descripción del problema

Día a día en el ámbito empresarial y/o familiar, existen diversas celebraciones que son compartidas por cada una de sus miembros, dichas celebraciones llevan un sinnúmero de diligencias, tales como: alimentación, sonido, decoración y el lugar donde se puedan reunir a los invitados.

En la búsqueda del lugar donde realizar la celebración, hay ciertos factores que hacen de esta actividad una de las más complicadas y en la cual se debe realizar una extensa investigación para llegar al lugar que cumpla con los criterios de tamaño, servicios y ubicación que necesitamos.

Para lograr lo anterior, es necesario dedicar horas a la visita de posibles sitios; esto trae consigo inversiones en transporte y tiempo, sobre todo si entre un lugar y otro existe una gran distancia por recorrer, sin olvidar que necesitamos tener conocimiento de la ciudad para poder desplazarnos sin mayor contratiempo. Pero para lograr visitar un posible sitio, es necesario obtener la información de las características que ofrece y si cumple con las expectativas del usuario; esto se logra a través de una consulta en un buscador de su elección, no obstante, ¿cómo saber si los lugares que me aparecen cumplen con lo requerido?, no existe una respuesta obvia a este interrogante, se debe ingresar a cada página, navegar por el sitio e ir descubriendo las características y servicios que ofrece, constituyéndose en una larga investigación, sin la certeza de obtener lo que deseamos.

Cada uno de estos inconvenientes, están en contra de lo que en principio es una celebración, pasar de disfrutar y divertirse, a momentos de estrés para los interesados y en una carrera



contra el tiempo para llegar al día elegido con todo preparado. Teniendo en cuenta la descripción de la problemática se pretende llegar a la resolución del siguiente interrogante

¿Cómo permitir a los clientes encontrar en un solo lugar la información requerida para la toma de decisiones, respecto al alquiler de salones de eventos y servicios adicionales para sus celebraciones?

## **2. Objetivo General**

Poner a la disposición de las personas una App que les permita realizar la búsqueda y alquiler de un salón de eventos agregando o eliminando servicios adicionales.

## **3. Objetivos Específicos**

- Beneficiar a la comunidad disminuyendo sus tiempos de búsqueda poniendo a su disposición una base de datos que contenga los servicios necesarios para realizar un evento.
- Promocionar los distintos salones de eventos, con el fin de atraer clientes a los mismos.
- Realizar cotizaciones sin desplazarse al lugar donde está ubicado el salón de eventos o la empresa que ofrece los servicios adicionales al alquiler.
- Implementar una arquitectura que soporte una alta demanda de usuarios y garantizar una buena experiencia a los mismos.
- Garantizar el arriendo de salones y compra de servicios de una forma rápida e intuitiva.
- Generar calificación de los servicios prestados.
- Implementar una aplicación que cumpla con los escenarios de seguridad y usabilidad.
- Implementar métodos de autenticación seguros con el fin de salvaguardar la información de los usuarios.

## 4. Marco de referencia

### 4.1. Estado del arte

Al realizar la búsqueda de antecedentes para este proyecto se encontraron algunos sitios web que ofrecen diferentes servicios:



**Figura 1. Logo Ezpot.**

**Fuente:** [www.ezpot.com](http://www.ezpot.com)

Plataforma que permite conectar a dueños de espacios, con personas que quieren hacer uso de estos lugares (espacios para trabajar, terrazas privadas, restaurantes, etc.).

Los dueños de los espacios publican en esta página y estos son convertidos en ezpots para ganar dinero por su alquiler temporal (Ezpot).



**Figura 2. Logo TSEventos.**

**Fuente:** <http://tseventos.co/>

Empresa integral en operación logística y producción de eventos con presencia a nivel nacional ejecutan directamente campañas de posicionamiento de marca, fidelización de clientes, activaciones y lanzamientos de productos entre otros.

Llevan a cabo y desarrollan cada una de las ideas de los clientes. Cuentan con un amplio

portafolio de servicios, en donde cubren las expectativas de principio a fin, sin la tercerización de servicios, son participativos en todos los procesos desde la conceptualización, planeación y ejecución de cada proyecto (TSEventos).



**Figura 3. Logo Regus.**

**Fuente:** <https://www.regus.com.co/>

Red de oficinas, espacios de coworking y salas de juntas opera en 120 países.

Proveedor de soluciones de espacios de trabajo flexible.

Ofrecen a las personas y las empresas la capacidad de trabajar donde quieran, cuando quieran y como quieran (Regus).

### **Proyecto salón de eventos dos quebradas**

Es un estudio de factibilidad que se realizó para ver si era posible crear una empresa que organice eventos sociales.

Creación de la microempresa “SALON DE EVENTOS DOSQUEBRADAS” para prestar un servicio a la comunidad de Dosquebradas, Pereira y Santa Rosa sobre celebraciones sociales y aula de clases o conferencias. Este proyecto consiste en crear una microempresa de prestación de servicios de preparación de fiestas y eventos para generar empleo a la inversionista y otros posibles empleos directos e indirectos (Ana María Obando, 2008, p.13).



**Figura 1. Logo Banquete.com.co.**

**Fuente:** <https://www.banquete.com.co/>

Web de empresas de catering y banquetes que te permite contactar directamente con los mejores profesionales del sector, pedir cotizaciones de catering de forma gratuita y comparar los precios (banquete.com.co).



**Figura 4. Logo HolaPlace.**

**Fuente:** <https://holaplace.com/es/>

Portal de alquiler terrazas y espacios privados para eventos al estilo de Airbnb en España.

Su actividad principal es poner en contacto a personas que quieran disfrutar de una terraza o espacio privado por una tarde, con personas que los tienen y quieran ganar un dinero extra. Se describen como un #airbnb, pero de terrazas y espacios para eventos (HolaPlace).



**Figura 5. Logo VenuesPlace.**

**Fuente:** <https://www.venuesplace.com/es>

Portal web y buscador especializado en lugares de celebración en España, ofrece asesoramiento.

Ofrecen una guía completa de espacios únicos, con información completa y cuidada de cada uno; se puede pedir presupuestos sin compromiso y por hacerlo a través de VenuesPlace dan beneficios de condiciones especiales.

Ofrecen servicio personalizado de búsqueda de espacios y organización del evento, asesoran gratuitamente por teléfono o por chat (VenuesPlace).



**Figura 6. Logo Wimmet.**

**Fuente:** <https://wimmet.co/>

Sitio web en el que se pueden publicar espacios que se tienen disponibles para alquilar se puede hacer la publicación de fotos, disponibilidad y precios.

Cuando los miembros reserven el espacio, envían una notificación con los datos de la persona que haya hecho la reserva, su foto, documento de identidad y la fecha de la reserva (Wimmet).

Después de realizar una amplia revisión en la web se puede determinar que hay múltiples sitios web que ofrecen el servicio de catering, alquiler de salones y organización de eventos, cada uno de estos por separado, a diferencia del planteamiento principal de este proyecto que es agrupar todos estos servicios en una App.

#### **4.2. Impacto**

El desarrollo de un aplicativo móvil para llevar a la mano el alquiler de espacios para eventos, genera un impacto en el mercado de los eventos, ya que permitirá facilitar la relación cliente-vendedor, además de mejorar la experiencia en estos tipos de procesos de compra.

También se intenta cambiar la dinámica de los pasos que se deben seguir para lograr el alquiler de un salón de eventos, beneficiando así a las personas interesadas en estos sitios reduciendo de manera drástica los tiempos de búsqueda, desplazamientos, costos y todo lo concerniente a esta actividad.

Debido a que este proyecto está enfocado a dar servicio en la web es de fácil acceso a la comunidad en general y en parte también se da la oportunidad de inclusión en el mercado a los diferentes actores que participa en este sector aportando así en gran medida a la generación de empleo y crecimiento de la industria de los eventos.

### **4.3. Componente de Innovación**

Los eventos sociales son algo muy popular en la población y la oportunidad de mejorar la experiencia al alquilar espacios sociales nos permite innovar en la forma de interactuar con las ofertas, en los clientes, sectores y ciudades ofreciendo la posibilidad de encontrar paquetes con los servicios adicionales al arrendamiento de salón y recibir cotizaciones sin desplazarse al lugar, con el fin de ahorrar costos y satisfacer todos los requerimientos en la organización del evento.

La aplicación tendrá como componente de innovación la agrupación de la mayor cantidad de servicios que son requeridos para la organización de un evento como lo son el servicio de catering, decoración, iluminación, música y sonido, organización del evento, entre otros; tener acceso a un carrito de compras, crear una lista de favoritos y al finalizar el proceso de contratación de servicio o alquiler el cliente podrá realizar una calificación de atención prestada.

### **4.4. Marco teórico**

**Evento:** hace referencia a un acontecimiento de importancia que reúne a un grupo de personas para la celebración de suceso generalmente programado, en el libro *Eventos, como organizarlos con éxito (2003)*, Rosario Jijena lo describe como:

Un evento es todo acontecimiento previamente organizado que reúne a un determinado número de personas en tiempo y lugar preestablecidos, que desarrollarán y compartirán una serie de actividades afines a un mismo objetivo para estímulo del comercio, la industria, el intercambio social y la cultura general. (p.33).

En la publicación *Manual para la organización de eventos artísticos y actividades de la Oficina de la Unesco en Costa Rica (2009)* plantea que los eventos se pueden clasificar según el tipo de actividad:

Un recurso para clasificar los eventos es su tipología; por ejemplo, si se trata de conferencias, charlas, seminarios, congresos, actos públicos, exposiciones, lanzamientos de marca o productos, aniversarios, festivales, graduaciones y otros. Además, en el amplio espacio de la cultura, podemos hablar también de varios tipos de actividades, entre ellas:

- Culturales:** actividades de formación, difusión, artísticas, lúdicas, recreativas (estas a su vez se subdividen en otras).

- Socioeducativas:** fiestas, reunión de organizaciones, acciones de movilización, de bien social, de información a la sociedad civil.

- Deportivas:** eventos deportivos y de esparcimiento, al aire libre, de naturaleza y medio ambiente (este rango es aún más amplio si el concepto se une a lo lúdico, donde el enfoque es favorecer tanto el desarrollo espiritual como el corporal).

- Empresariales y corporativos:** reuniones, charlas, presentaciones de productos, seminarios, conferencias, encuentros, capacitaciones, congresos, giras y ferias.

### **Salón de eventos**

Según la publicación de Julián Pérez Porto y María Merino *Definición de salón (2014)*:



Salón es el aumentativo de sala. El término se utiliza para nombrar al aposento de grandes dimensiones en una casa que se destina a la recepción de las visitas o para albergar fiestas y reuniones.

Florencia Uche en su publicación plantea que existen diferentes tipos de salones:

### **Exposición en la que se ofrecen y comercian productos**

En el ámbito comercial, de la industria, también nos encontramos con una referencia de la palabra, ya que designa a la exposición o feria comercial en la cual se exhiben y comercializan los productos y servicios de una industria o área de negocios.

### **Aula de una escuela**

También, salón, puede denominar el aula de una institución educativa y la habitación de una empresa en la cual se celebran las juntas de directorio o reuniones importantes.

### **Salón de belleza**

Si bien los salones de belleza son lugares socialmente concurridos por las mujeres, para cortarse el cabello, para cambiar el color del mismo, depilarse, peinarse para una fiesta, entre otros, debemos decir, que también son espacios visitados por los hombres para cortarse el pelo, la barba y bigote.

Teniendo en cuenta que el desarrollo de este sistema va enfocado a las plataformas móviles y una alta demanda de usuario se pretende implementar una arquitectura por microservicios, con el fin de soportar el alto flujo de solicitudes y transacciones de los usuarios, sin embargo, se debe establecer el tipo de desarrollo móvil, en la actualidad existen los siguientes:

### **Aplicaciones Nativas:**

Las aplicaciones nativas se desarrollan de forma específica para el sistema operativo en el que se van a ejecutar. De esta forma, si una App está concebida para ser ejecutada en Android, tendrá un desarrollo sustancialmente distinto que si se desarrollase para ser ejecutada en iOS. Ante esta necesidad, el desarrollador de apps empleará lenguajes de programación específicos y concretos para cada sistema operativo. En términos generales, con los lenguajes de programación Java o Kotlin podremos crear aplicaciones para dispositivos Android, con Objective-C o Swift crearemos aplicaciones específicas para iOS.

Este tipo de desarrollos tiene ventajas e inconvenientes. En cuanto a los beneficios, vemos que son aplicaciones que ofrecen una mejor interfaz a los usuarios, además de ser capaces de aprovechar todo el potencial de los teléfonos inteligentes en los que están instaladas. Este tipo de desarrollos sacan el máximo potencial de todos los recursos del teléfono y ofrecen una muy buena experiencia al usuario.

Sin embargo, en cuanto a las desventajas, este tipo de desarrollo de aplicaciones requiere de mayor tiempo e inversión, por lo que la empresa que decida desarrollar aplicaciones nativas deberá destinar una inversión mayor ya que, como estas apps se desarrollan para el sistema operativo en el que se ejecutarán, requieren la creación de tantas copias como sean necesarias para cada uno de los sistemas operativos en los que se van a ejecutar. El desarrollo no es compatible de un sistema operativo a otro, no se puede reutilizar el mismo código y requiere que el desarrollador sepa utilizar los diferentes lenguajes de programación. De esta forma, el usuario tendrá una buena experiencia de navegación, independientemente del sistema operativo que utilice su Smartphone.

### **Aplicaciones progresivas**

Las aplicaciones web progresivas o progressive web apps son una nueva generación de aplicaciones que está revolucionando el panorama actual, incrementando y avanzando en sus funcionalidades teniendo en cuenta el dispositivo móvil en el que van a ejecutarse para extraer todo su potencial.

Estas apps progresivas aportan al usuario las ventajas de las aplicaciones nativas, pero se desarrollan en base a tecnologías web ampliamente conocidas por los programadores como HTML, CSS o JavaScript. Sería algo así como páginas web muy similares en su funcionamiento a las aplicaciones nativas, pero que no requieren de lenguajes de programación específicos para cada sistema operativo.

Las aplicaciones progresivas, que reciben este nombre por ser capaces de adaptarse a cada dispositivo (de manera progresiva), consiguen una experiencia de usuario Premium, tanto es así que el usuario no suele notar la diferencia respecto a las apps nativas. Además, no requieren que el usuario disponga de una buena conexión a internet, ya que podrá acceder a ellas sin conexión o con mala conexión, y beneficiarse de todas sus funcionalidades, sin embargo, dependen del tipo de navegador que se encuentre instalado el dispositivo donde se va a ejecutar, por tal motivo presenta incompatibilidades con navegadores como Safari que es el navegador oficial del sistema operativo IOS, por tal presenta inconvenientes en la ejecución de dispositivos iPhone.

Teniendo en cuenta lo anteriormente mencionado con respecto a los diferentes tipos de desarrollo para plataformas móviles, para este proyecto se eligió un desarrollo nativo dado que es la solución más estable y escalable en el tiempo teniendo en cuenta los tiempos de

desarrollo son mayores ya que necesitaríamos realizar la implementación en dos plataformas (Android, IOS).

Por la otra parte el desarrollo del backend tiene muchísimas alternativas sin embargo nos vamos a enfocar en Firebase, ya que Es una plataforma digital que se utiliza para facilitar el desarrollo de aplicaciones web o móviles de una forma efectiva, rápida y sencilla, la cual es utilizada por sus diversas funciones como una técnica de Marketing Digital para aumentar la base de usuarios y generar mayores beneficios económicos.

Su principal objetivo, es mejorar el rendimiento de las apps mediante la implementación de diversas funcionalidades que van a hacer de la aplicación en cuestión, mucho más manejable, segura y de fácil acceso para los usuarios.

Los servicios ofrecidos se pueden dividir en 3 categorías:

**En el desarrollo:** Son diferentes y con distintas utilidades:

**Real time data base:** Nos ayudan, dándonos una base de datos en tiempo real.

**Autenticación:** Se ejecutan para identificar a los usuarios mediante el email o las redes sociales.

**Nube de almacenamiento:** Almacenamiento y envío de archivos a la escala de Google.

**Hosting:** Se usa para publicar nuestra página web.

**Remote config:** Se emplea para modificar ciertos aspectos de nuestra App sin la necesidad de actualizar la misma.

**Test lab:** Es para probar la aplicación antes de publicarla.

**Crash reporting:** Se utiliza para reportar errores de la aplicación.

Para hacer crecer nuestras aplicaciones y con esto, nuestro negocio, nos da la posibilidad de:

- Enviar notificaciones a los usuarios, con Notifications.
- Permite mostrar las aplicaciones de una forma bastante adecuada en los resultados de los motores de búsqueda, con el uso de App indexing.
- Proporciona la forma para acceder a la aplicación desde otros links desde otras aplicaciones, mediante Dynamic Links.
- Permite hacer publicidad de nuestra aplicación usando AdWords.
- Ayuda a monetizar nuestra aplicación mediante la publicidad, utilizando AdMob.

Hay que tener en cuenta que las aplicaciones hoy en día representan un mercado que aún se está descubriendo y se mantiene en constante evolución, por ende, adherirse a este se debe hacer utilizando las mejores plataformas.

### **Las Analíticas o Analytics:**

**Analytics:** Muestra los resultados sobre el comportamiento del usuario en su IOS o Android apps. Permitiendo que puedas tomar las mejores decisiones sobre tu producto y una optimización de tu estrategia de marketing.

**Firebase Analytics:** se puede utilizar conjunto a Google AdWords para lograr obtener estadísticas sobre tu aplicación y cómo esta ha llegado a los usuarios y sus reacciones a la misma.

Teniendo en cuenta lo anteriormente mencionado sobre Firebase, se hace un complemento en tiempo, ya que se pretende realizar un desarrollo nativo, tendremos un consumo considerable de tiempo de desarrollo en front-end, Firebase nos permite realizar un desarrollo de Back-end más rápido y eficiente.

### **Metodología Scrum**

En su publicación *Metodología Scrum: qué es y cómo funciona* (2020) Encarna Abellán nos plantea esta metodología de la siguiente manera:

Se trata de una metodología de trabajo ágil que tiene como finalidad la entrega de valor en períodos cortos de tiempo y para ello se basa en tres pilares: la transparencia, inspección y adaptación. Esto permite al cliente, junto con su equipo comercial, insertar el producto en el mercado pronto, rápido y empezar a obtener ventas.

Con el método Scrum todos los implicados tienen conocimiento de qué ocurre en el proyecto y cómo ocurre. Esto hace que haya un entendimiento “común” del proyecto, una visión global.

### **1. Transparencia**

Con el método Scrum todos los implicados tienen conocimiento de qué ocurre en el proyecto y cómo ocurre. Esto hace que haya un entendimiento “común” del proyecto, una visión global.

### **2. Inspección**

Los miembros del equipo Scrum frecuentemente inspeccionan el progreso para detectar posibles problemas. La inspección no es un examen diario, sino una forma de saber que el trabajo fluye y que el equipo funciona de manera auto-organizada.

### **3. Adaptación**

Cuando hay algo que cambiar, el equipo se ajusta para conseguir el objetivo del sprint. Esta es la clave para conseguir el éxito en proyectos complejos, donde los requisitos son cambiantes o poco definidos y en donde la adaptación, la innovación, la complejidad y flexibilidad son fundamentales.

En Scrum existen 3 roles muy importantes: Product Owner, Scrum Master y Equipo de desarrollo.

### **1. Product Owner:**

Es el responsable de maximizar el valor del trabajo del equipo de desarrollo. La maximización del valor del trabajo viene de la mano de una buena gestión del Product Backlog, el cual explicaremos más adelante.

El Product Owner es el único perfil que habla constantemente con el cliente, lo que le obliga a tener muchos conocimientos sobre negocio.

Un equipo Scrum debe tener solo un Product Owner y este puede ser parte del equipo de desarrollo.

### **2. Scrum Master:**

Es el responsable de que las técnicas Scrum sean comprendidas y aplicadas en la organización. Es el manager de Scrum, un líder que se encarga de eliminar impedimentos o inconvenientes que tenga el equipo dentro de un sprint (que ya revisaremos en detalle más adelante), aplicando las mejores técnicas para fortalecer el equipo de marketing digital.

Dentro de la organización, el Scrum Master tiene la labor de ayudar en la adopción de esta metodología en todos los equipos.

### **3. Equipo de desarrollo:**

Son los encargados de realizar las tareas priorizadas por el Product Owner. Es un equipo multifuncional y auto-organizado. Son los únicos que estiman las tareas del product backlog, sin dejarse influenciar por nadie.

El desarrollo iterativo se realiza en un sprint, que contiene los siguientes eventos: sprint planning, daily meeting, sprint review y sprint retrospective.

### **1. Sprint**

El sprint es el corazón de Scrum, es el contenedor de los demás hitos del proceso. Todo lo que ocurre en una iteración para entregar valor está dentro de un sprint. La duración máxima es de un mes, el tiempo se determina en base al nivel de comunicación que el cliente quiere tener con el equipo.

### **2. Sprint planning**

En esta reunión todo el equipo Scrum define qué tareas se van a abordar y cuál será el objetivo del sprint. La primera reunión que se hace en el sprint puede llegar a tener una duración de 8 horas para sprints de un mes.

El equipo se hace las siguientes preguntas:

¿Qué se va a hacer en el sprint? En base a ello, se eligen tareas del Product backlog.

¿Cómo lo vamos a hacer? El equipo de desarrollo define las tareas necesarias para completar cada ítem elegido del Product Backlog.

### **3. Daily meeting**

Es una reunión diaria dentro del sprint que tiene como máximo 15 minutos de duración. En ella debe participar, sí o sí, el equipo de desarrollo y el Scrum Master. El Product Owner no tiene necesidad de estar presente.

En esta reunión diaria el equipo de desarrollo hace las siguientes tres preguntas:

¿Qué hice ayer?

¿Qué voy a hacer hoy?



¿Tengo algún impedimento que necesito que me solucionen?

#### **4. Sprint review**

La review del valor que vamos a entregar al cliente se hace en esta reunión, al final de cada sprint. Su duración es de 4 horas para sprints de un mes, y es la única reunión de Scrum a la que puede asistir el cliente. En ella el Product Owner presenta lo desarrollado al cliente y el equipo de desarrollo muestra su funcionamiento.

#### **5. Sprint retrospective**

La retrospectiva es el último evento de Scrum, tiene una duración de 3 horas para Sprints de un mes, y es la reunión del equipo en la que se hace una evaluación de cómo se ha implementado la metodología Scrum en el último sprint.

Las herramientas principales de Scrum son: product backlog y sprint backlog.

#### **1. Product backlog**

Básicamente, el product backlog es el listado de tareas que engloba todo un proyecto. Cualquier cosa que debamos hacer debe estar en el product backlog y con un tiempo estimado por el equipo de desarrollo.

La responsabilidad exclusiva de ordenar el product backlog es del Product Owner, que se encuentra en constante comunicación con el cliente para asegurarse de que las prioridades están bien establecidas.

#### **2. Sprint backlog**

Es el grupo de tareas del product backlog que el equipo de desarrollo elige en el sprint planning junto con el plan para poder desarrollarlas. Debe ser conocido por todo el equipo, para asegurarse de que el foco debe estar en este grupo de tareas.

### **Ventajas de la metodología Scrum**

Scrum es muy fácil de aprender: los roles, hitos y herramientas son claros y tienen un objetivo por lo que es un método muy relacionado con nuestra manera diaria de trabajar.

El cliente puede comenzar a usar el producto rápidamente.

Se agiliza el proceso, ya que la entrega de valor es muy frecuente.

Menor probabilidad de sorpresas o imprevistos, porque el cliente está viendo frecuentemente el proyecto.

### **Desventajas de la metodología Scrum**

Aunque Scrum sea fácil de aprender, es muy difícil implementarlo. Esto supone una predisposición y un cambio de cultura de la organización que debe ir desde los altos mandos hasta los clientes.

La necesidad de tener equipos multidisciplinares puede ser un problema, ya que es difícil encontrar personas que sean capaces de hacer todo el trabajo de un equipo.

El equipo puede tender a realizar el camino más corto para conseguir el objetivo de un sprint, el cual no siempre ofrece resultados de calidad.

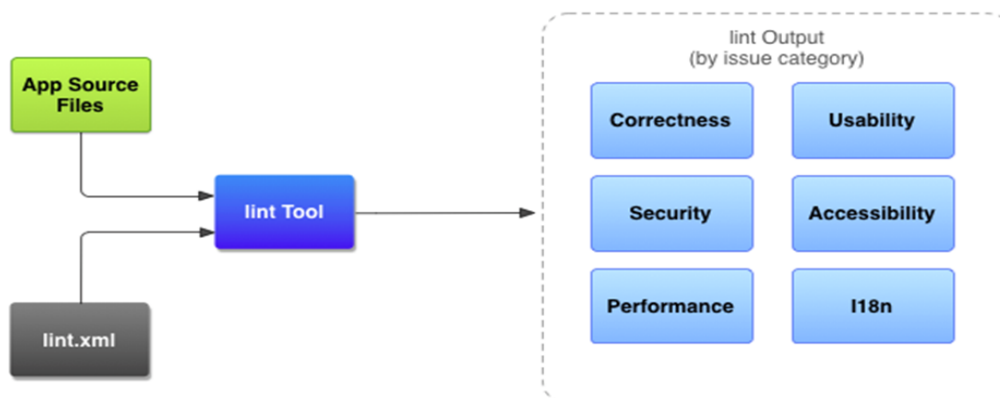
### **LINT**

Además de garantizar que su aplicación cumpla con sus requisitos funcionales mediante la creación de pruebas, es importante que también se asegure de que su código no tenga problemas estructurales ejecutando el código a través de lint. La herramienta lint ayuda a encontrar código mal estructurado que puede afectar la confiabilidad y eficiencia de sus aplicaciones de Android y hacer que su código sea más difícil de mantener.

Por ejemplo, si sus archivos de recursos XML contienen espacios de nombres no utilizados, esto ocupa espacio y genera un procesamiento innecesario. Otros problemas estructurales, como el uso de elementos obsoletos o llamadas a la API que no son compatibles con las versiones de la API de destino, pueden provocar que el código no se ejecute correctamente. Android Studio proporciona una herramienta de escaneo de código llamada lint que puede ayudarlo a identificar y corregir problemas con la calidad estructural de su código sin tener que ejecutar la aplicación o escribir casos de prueba. Cada problema detectado por la herramienta se informa con un mensaje de descripción y un nivel de gravedad, para que pueda priorizar rápidamente las mejoras críticas que deben realizarse. Además, puede reducir el nivel de gravedad de un problema para ignorar los problemas que no son relevantes para su proyecto o aumentar el nivel de gravedad para resaltar problemas específicos.

La herramienta lint comprueba los archivos fuente de su proyecto de Android en busca de posibles errores y mejoras de optimización en cuanto a corrección, seguridad, rendimiento, usabilidad, accesibilidad e internacionalización. Cuando se usa Android Studio, las inspecciones IDE y lint configuradas se ejecutan cada vez que crea su aplicación.

**Figura 7. Arquitectura LINT**



## 5. Metodología

Para este proyecto se utilizó la metodología agile Scrum, principalmente porque se dispuso de un grupo pequeño para todo lo relacionado con este proyecto, además se contaba con poco tiempo para la realización de este, y esta metodología de trabajarlo en pequeñas partes se acomodó mucho a las necesidades del equipo y del proyecto como tal, permitiendo así optimizar y mejorar los procesos de trabajo, y al realizar una revisión continua, se pudo lograr una mayor calidad en el software.

### **Implementación de la metodología**

**Product Owner:** Jenny Candelo

**Scrum Master:** Brayan Herreño

**Equipo de desarrollo:** Brayan Herreño, Jenny Candelo

**Sprint:** viernes de cada semana

**Sprint planning:** lunes. 2 horas

**Daily meeting:** diaria

**Sprint review:** viernes- 1 hora

**Sprint retrospective:** viernes – 30 minutos

**Recolección de información:** Se hizo una búsqueda en la web sobre los posibles lugares que podrían tener una actividad igual o similar a la plateada por nosotros, en el proceso vimos que era más factible dividirlos en categorías según la actividad que realizaban lo cual facilito mucho más la búsqueda al tener presente en debíamos enfocarnos.

**Análisis de información:** con la información recolectada en la búsqueda se pudo hacer una depuración de los sitios web, apps y proyectos más relevantes para el nuestro y con esto

iniciar el planteamiento de los objetivos y las principales características con las cuales debía cumplir el proyecto para ser realmente atractivo y funcional.

**Preparación del proyecto:** para la preparación del proyecto se había decidido que SCRUM se ajustaba mucho a los plazos y al equipo, entonces se procedió a plantear un plan de trabajo y los roles, teniendo en cuenta que estos podrían sufrir cambios a lo largo del proyecto.

**Desarrollo del proyecto:** Para este proyecto se usó la metodología SCRUM, contemplando la capacidad del equipo encargado del tema, se realizó el levantamiento de los requerimientos, las pruebas, la documentación, el desarrollo del aplicativo teniendo siempre en cuenta que cumpliera con los estándares de calidad planteados

## 6. proceso de software

En esta etapa de se describe cada una de las partes más relevantes en proceso del desarrollo del aplicativo resultante de este proyecto, en el que se incluyen aquellas funcionalidades que son indispensables en el correcto funcionamiento del sistema, las interacciones que pueden tener los diferentes usuarios con este y el funcionamiento interno del sistema con los recursos externos.

### 6.1. Requerimientos funcionales

**Tabla 1.      Requerimiento Funcional 1**

<b>RF01.</b>	<b>Permitir al cliente consultar los salones y/o servicios registrados en el sistema.</b>
Especificación	El sistema debe permitir al cliente consultar los salones que estén registrados en el sistema
Justificación	Este requerimiento permite consultar la disponibilidad de un salón en un momento determinado permitiendo la futura contratación del servicio o alquiler.
Alcanzable	Este requerimiento se podrá cumplir siguiendo la disponibilidad de los servicios solicitados.
Medible	Se tendrá en cuenta la solicitud realizada por el usuario vs la disponibilidad del salón.

Fuente: Elaboración propia

**Tabla 2.      Requerimiento Funcional 2**

<b>RF02.</b>	<b>Permitir al cliente alquilar salón</b>
Especificación	El sistema debe permitir al cliente realizar el proceso de alquiler del salón que este haya elegido.
Justificación	Este requerimiento permite al cliente realizar el alquiler de un salón si este se encuentra disponible.
Alcanzable	Se podrá cumplir con el objetivo si el cliente se encuentra registrado en el sistema y si el salón se encuentra disponible.
Medible	Se podrá realizar el alquiler del salón si el cliente y el arrendador están de acuerdo con los términos del contrato.

Fuente: Elaboración propia

**Tabla 3.      Requerimiento Funcional 3**

<b>RF03</b>	<b>Cancelar el proceso de alquiler del salón</b>
Especificación	El sistema debe permitir al cliente cancelar el proceso alquiler.

Justificación	Permite que el cliente pueda desistir de su decisión de tomar en alquiler un salón en caso de necesitarlo.
Alcanzable	Se podrá realizar la cancelación del proceso de alquiler si es con justa razón.
Medible	Se podrá realizar el proceso de cancelación si el cliente y el arrendador están de acuerdo con los términos del contrato.

Fuente: Elaboración propia

**Tabla 4.      Requerimiento Funcional 4**

<b>RF04</b>	<b>Insertar nuevos salones y/ servicios en la base de datos.</b>
Especificación	El sistema debe permitir a los proveedores ingresar nuevos salones y/o servicios al sistema.
Justificación	El proveedor puede insertar nuevos salones y/o servicios en la base de datos, para ello, deberá rellenar un formulario con los datos referentes al salón y/o servicio.



Alcanzable	El salón o servicio se encuentra disponible en el sistema y puede ser consultado por el usuario.
Medible	Se podrá hacer la consulta del salón o servicio en el sistema.

Fuente: Elaboración propia

**Tabla 5. Requerimiento Funcional 5**

FR05	<b>Adicionar a una lista de favoritos salones y/o servicios.</b>
Especificación	El sistema debe permitir adicionar a una lista de favoritos salones y/o servicios.
Justificación	El sistema debe permitir al cliente ingresar nuevos salones y/o servicios a su lista de favoritos.
Alcanzable	El salón o servicio se encuentra disponible en el sistema y puede ser consultado por el usuario.
Medible	Se podrá adicionar salones y/o servicios en la lista de favoritos.

Fuente: Elaboración propia

**Tabla 6.      Requerimiento Funcional 6**

RF06	<b>adicionar a un carro de compras los productos que desea adquirir.</b>
Especificación	El sistema debe permitir adicionar a un carro de compras los productos que desea adquirir.
Justificación	El sistema debe permitir al cliente adicionar salones y/o servicios a su carrito de compras.
Alcanzable	El salón o servicio se encuentra disponible en el sistema y puede ser consultado por el usuario.
Medible	Se podrá adicionar salones y/o servicios en el carro de compras.

Fuente: Elaboración propia

## 6.2. Requerimientos no funcionales

**Tabla 7.      Requerimiento no Funcional 1**

<b>RNF01</b>	<b>El sistema debe garantizar la disponibilidad de la información.</b>
Especificación	La información que se encuentra registrada en el sistema debe estar respaldada en caso que se presente algún evento para garantizar la disponibilidad.

Criterio de aceptación	La información debe ser almacenada en una base de datos adecuada.
Restricciones	Capacidad en Disco de almacenamiento y backup de la información.
Medible	La información debe ser presentada en el momento que se requiere.
Prioridad	Alta.

Fuente: Elaboración propia

**Tabla 8. Requerimiento no Funcional 2**

<b>RNF02.</b>	<b>El sistema debe garantizar la integridad de los datos de usuario de ingreso al sistema.</b>
Justificación	Las credenciales de acceso en la autenticación deben ser protegidas para garantizar la no suplantación de los usuarios y acceso no permitido por terceros.
Criterio de aceptación	La contraseña al ingresar en el formulario de registro o autenticación debe ser ofuscada.
Restricciones	Ninguna.
Medible	Ingreso al sistema.
Prioridad	Alta.

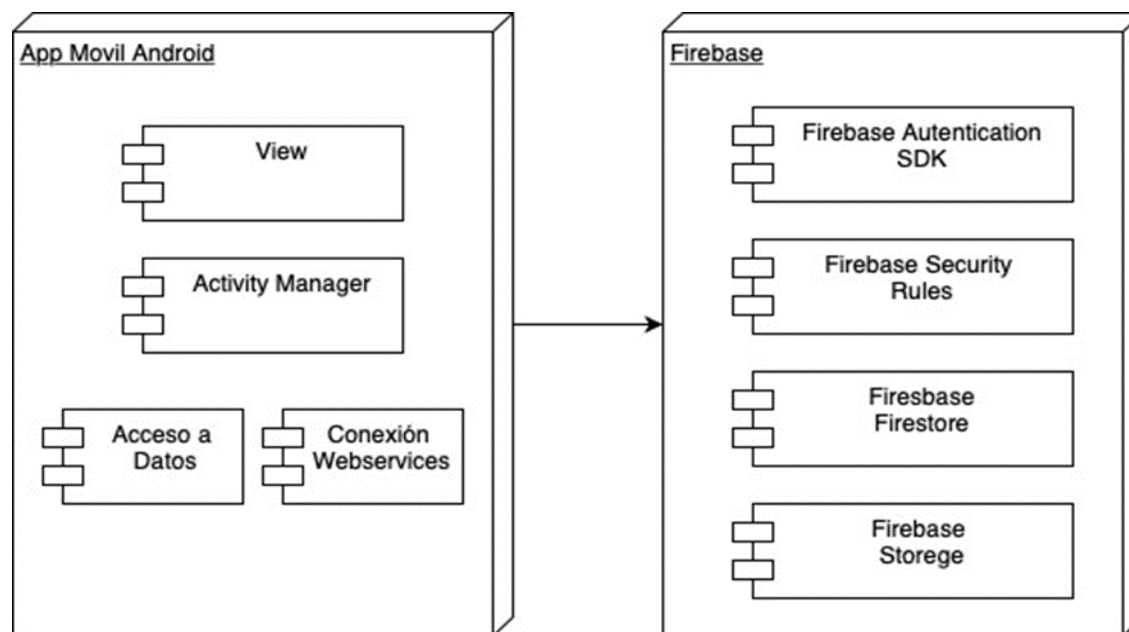
Fuente: Elaboración propia

### 6.3. Diseño y arquitectura

#### 6.3.1. Diagrama de despliegue

El back-end se va orientar a la nube estará apoyado en Firebase el cual ofrece un componente de seguridad (Firebase Security Rules), la autenticación de usuarios se realiza con (Firebase Authentication SDK), gestión de bases de datos no relacionales (Firebase Firestore) y almacenamiento de archivos (Firebase Storage). Por otra parte, se implementarán aplicaciones móviles nativas las cuales se instalarán a partir de archivos .APK para Android y .IPA para sistemas IOS.

**Figura 8. Diagrama de despliegue.**

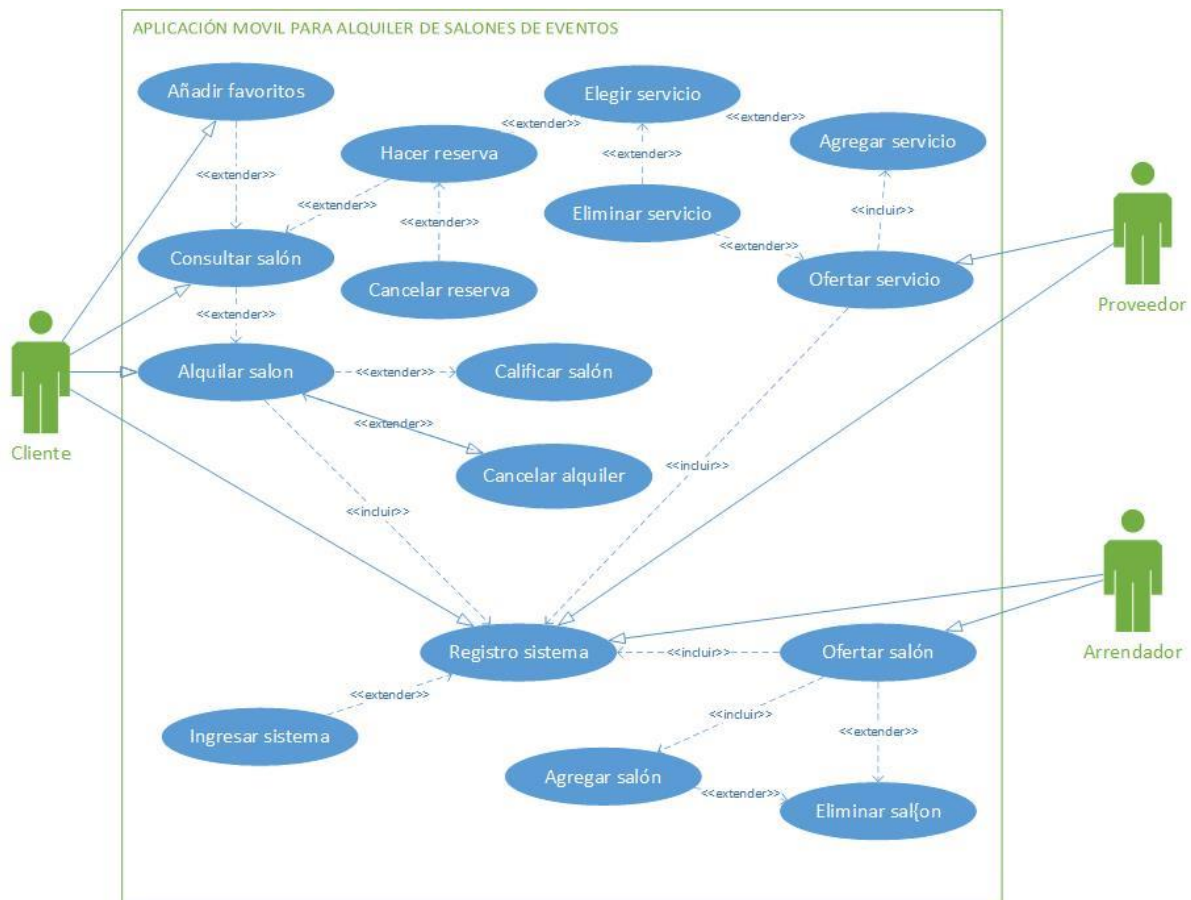


Fuente: Elaboración propia

#### 6.3.2. Diagrama de uso arquitecturalmente relevante

En el diagrama de despliegue se muestran las acciones que los diferentes usuarios pueden realizar en el sistema dependiendo de su rol

**Figura 9. Diagrama de casos de uso.**



Fuente: Elaboración propia.

**Tabla 9. Tabla actores del sistema**

Actor	Descripción
<b>Cliente</b>	Usuario del sistema que tendrá acceso a los productos o servicios ofrecidos por los proveedores, podrá elegir lo que más se ajuste a sus necesidades, tendrá acceso a un carrito de compras para hacer el pago y

	a una lista de deseos para armar su paquete.
<b>Proveedor</b>	Usuario que podrá ofertar sus productos o servicios según las políticas del sistema, también deberán tener actualizado su inventario y/o disponibilidad y costos.
<b>Arrendador</b>	Usuario que podrá ofertar salones disponibles para eventos según las políticas del sistema, también deberán tener actualizada su disponibilidad y costos.

Fuente: Elaboración propia.

**Actor:** Cliente

**Consultar salón:** El cliente podrá consultar los salones de eventos que se encuentren disponibles en el sistema.

**Añadir a favoritos:** Si el cliente se encuentra interesado en algunos de los servicios o salones de eventos tiene la posibilidad de agregarlos a una lista de deseos para consultarlo después.

**Hacer reserva:** Si el cliente encuentra atractivo algún servicio o salón tiene la opción de poder realizar una reserva.

**Alquilar salón:** Si el cliente encuentra que el salón de eventos cumple con sus expectativas podrá realizar el alquiler.

**Cancelar alquiler:** El cliente podrá cancelar el proceso de alquiler del salón si en algún momento lo requiere.

**Calificar salón:** Después de haber recibido el servicio el cliente puede dar una calificación de acuerdo a su experiencia.

**Registro en el sistema:** Si el cliente requiere alquilar un salón de eventos o contrata un servicio debe registrar sus datos en el sistema.

**Ingresar al sistema:** Después de haber registrados sus datos en el sistema el cliente podrá ingresar al sistema utilizando sus credenciales.

**Actor:** Proveedor

**Ofertar servicio:** El proveedor podrá registrar los servicios que desee ofertar a los clientes.

**Agregar servicio:** El proveedor podrá agregar en cualquier momento otros servicios que tenga disponibles.

**Eliminar servicio:** El proveedor podrá eliminar en cualquier momento los servicios que no tenga disponibles o no desee seguir ofertando.

**Registro en el sistema:** Si el proveedor requiere ofertar un servicio debe registrar sus datos en el sistema.

**Elegir servicio:** El proveedor podrá escoger los servicios que tiene en su lista para tomar la acción que desee.

**Ingresar al sistema:** Después de haber registrados sus datos en el sistema el proveedor podrá ingresar al sistema utilizando sus credenciales.

**Actor:** Arrendador

**Registro en el sistema:** Si el arrendador requiere ofertar un salón de eventos debe registrar sus datos en el sistema.

**Ofertar salón:** El arrendador podrá registrar los salones que desee ofertar a los clientes.

**Agregar salón:** El arrendador podrá agregar en cualquier momento otros salones que tenga disponibles.

**Ingresar al sistema:** Después de haber registrados sus datos en el sistema el arrendador podrá ingresar al sistema utilizando sus credenciales.

**Eliminar salón:** El arrendador podrá eliminar en cualquier momento el o los salones que no tenga disponibles o no desee seguir ofertando.

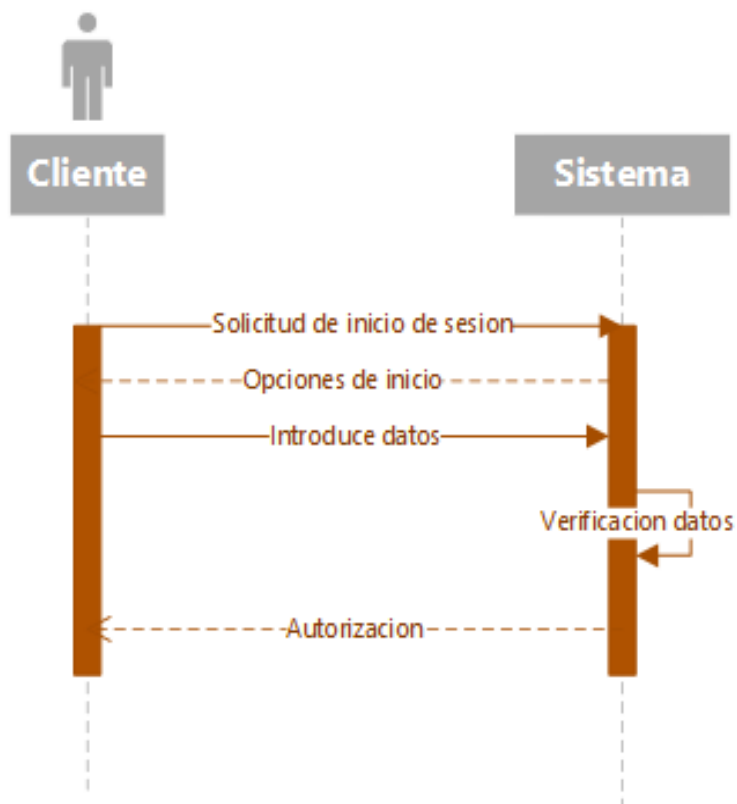
**Registro en el sistema:** Si el arrendador requiere ofertar un salón debe registrar sus datos en el sistema.

### 6.3.3. Diagrama de secuencia

Los diagramas de secuencia muestran en un orden cronológico como suceden los eventos cuando un cliente requiere realizar determinada acción y además muestra las interacciones entre los distintos objetos para lograrlo. Para este proyecto se tuvieron en cuenta los diagramas de secuencia mas relevantes para el planteamiento funcional principal del proyecto que es realizar el alquiler de un salón de eventos.

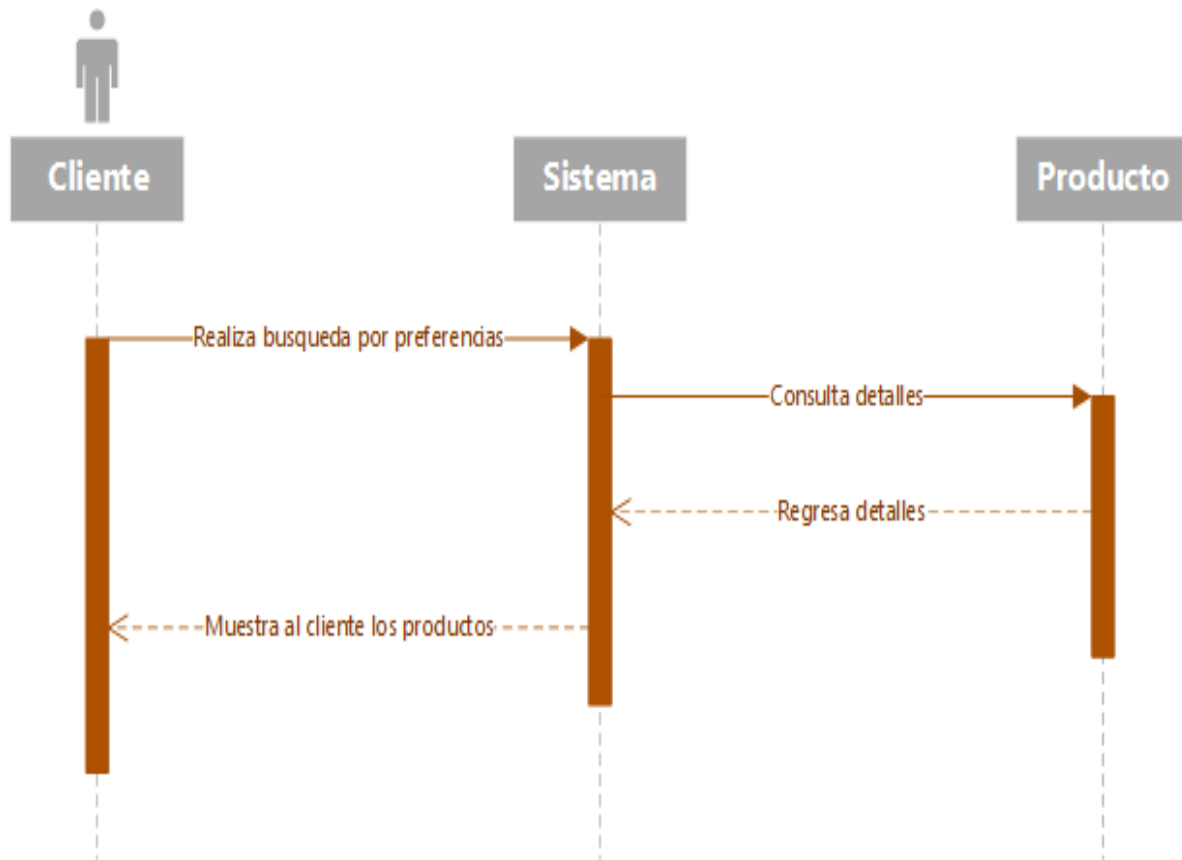
Para dar cumplimiento al objetivo principal de este proyecto se tuvieron cuenta 4 secuencias en las cuales en cada una el resultado final debe ser positivo para poder cumplir con la secuencia. Las secuencias a seguir son las siguientes:



**Figura 10. Diagrama de secuencia Iniciar Sesión**

Fuente: Elaboración propia

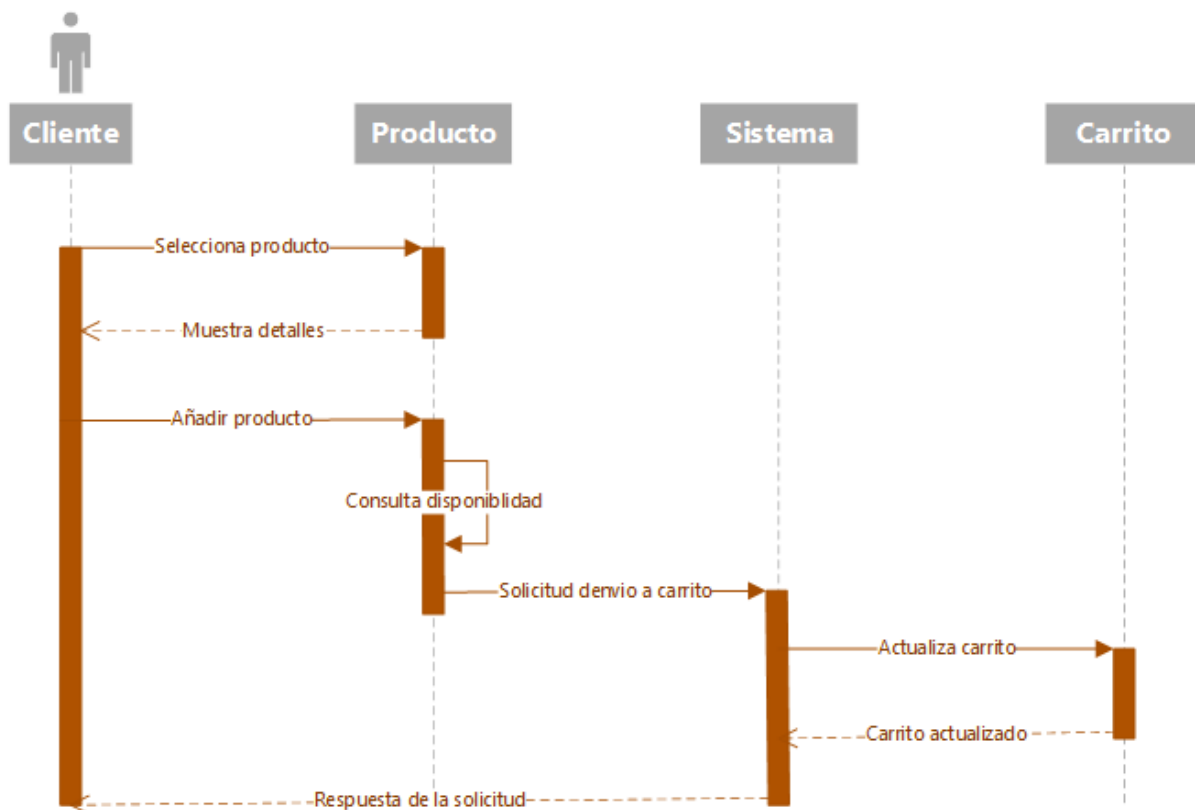
En este primer diagrama se representa la secuencia que realizarán el cliente y el sistema para la verificación y correcto inicio de sesión.

**Figura 11. Diagrama de secuencia Realizar búsqueda**

Fuente: Elaboración propia

Después que el cliente ingresa sus datos de inicio de sesión y el sistema los verifica, el siguiente paso es realizar la búsqueda de los productos de interés por parte del cliente, el sistema filtrara los resultados mostrando los detalles de los productos que cumplen con los criterios planteados inicialmente.

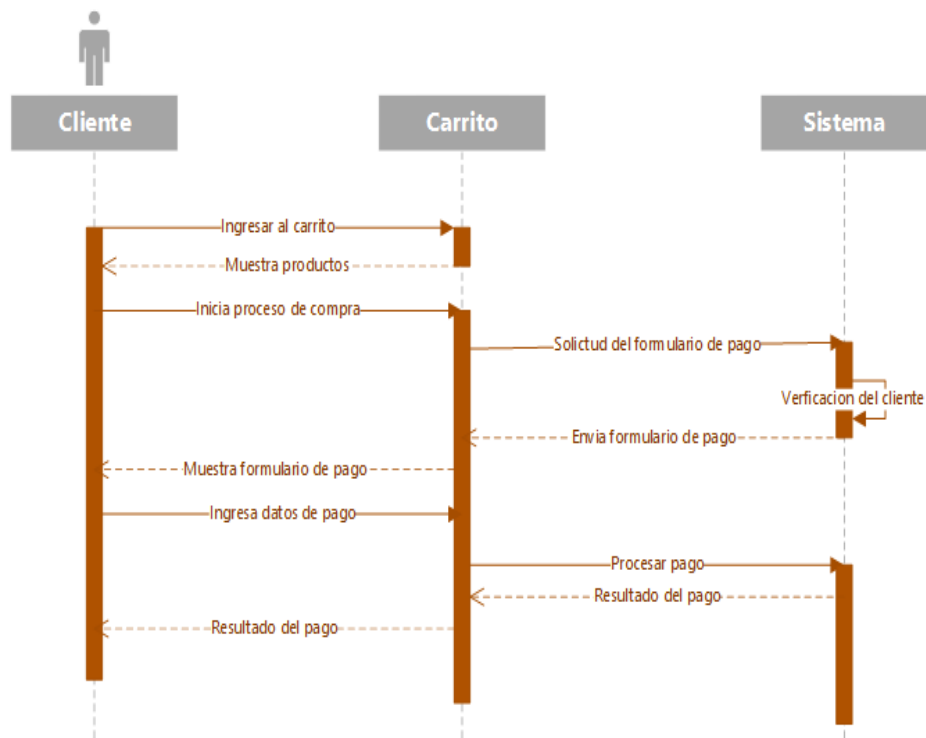
Figura 12. Diagrama de secuencia Añadir a carrito de compras



Fuente: Elaboración propia

El siguiente paso después de que el cliente ha encontrado el/los productos que se ajustan a sus necesidades es ponerlos en el carrito de compras, que serán añadidos según la disponibilidad.

**Figura 13. Diagrama de secuencia Realizar pago**

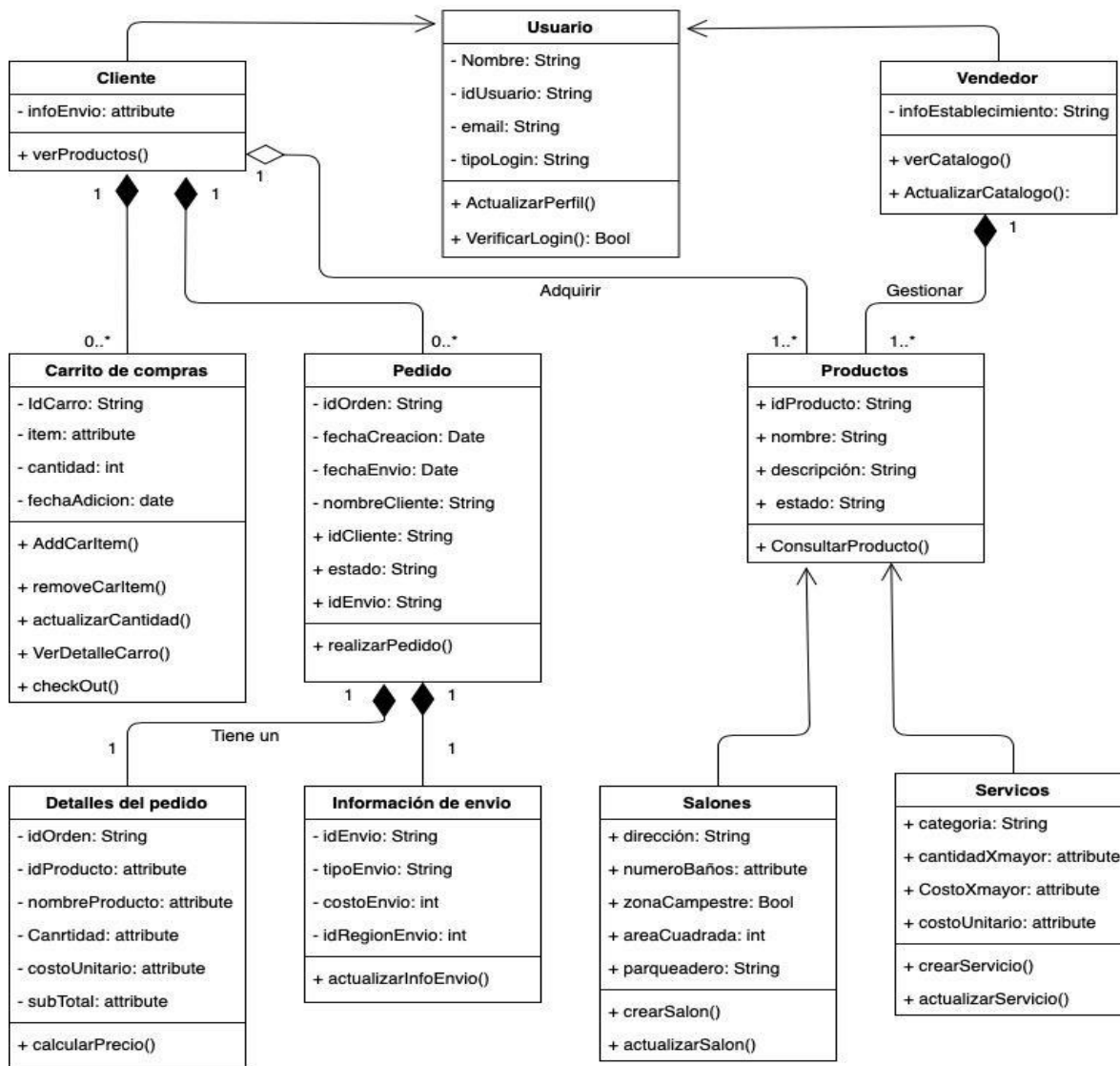


Fuente: Elaboración propia

El ultimo paso de esta serie es realizar el pago, cuando el cliente se encuentra satisfecho con los productos que ha añadido al carrito de compras, procederá a la realización del pago, en la cual el sistema le mostrará en detalle todos los productos del pedido, el precio final y los métodos de pago disponibles, después que el cliente elige como pagar e ingresa los datos el sistema hará la verificación.

### 6.3.4. Diagrama de clases

Figura 14. Diagrama de clases



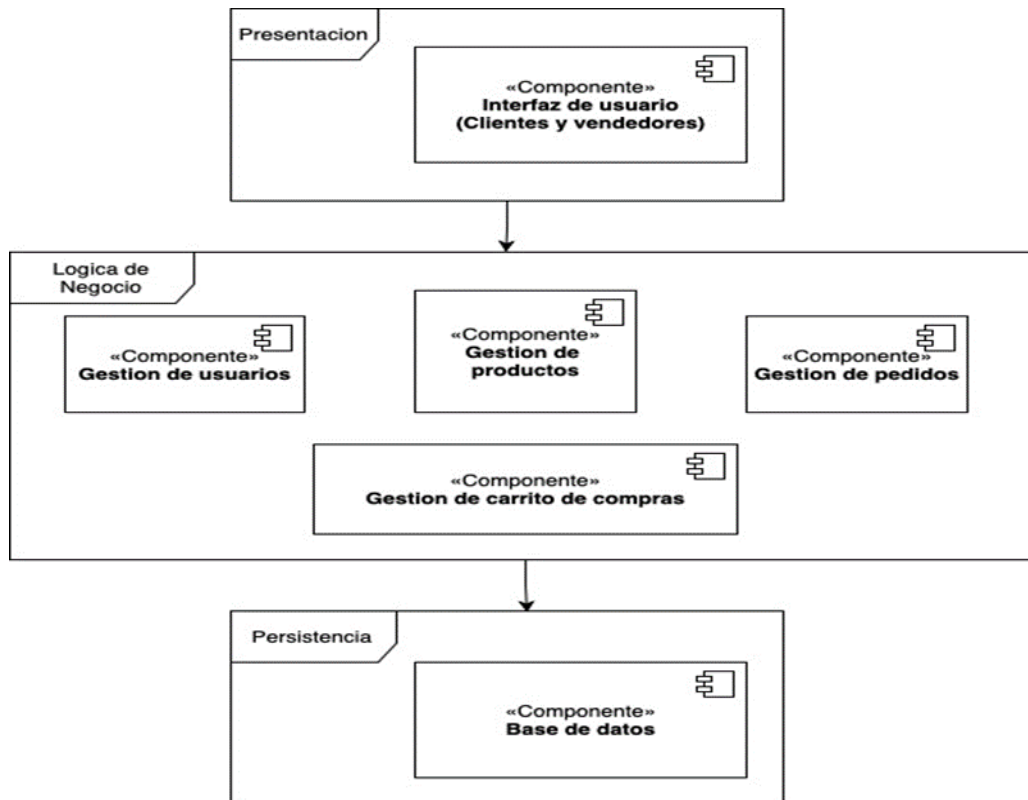
Fuente: Elaboración propia

El diagrama de clases nos muestra las diferentes clases necesarias para el funcionamiento del sistema y su interacción entre ellas, en este diagrama se encuentran las clases

correspondientes a los actores principales y las acciones que se pueden realizar sobre los productos en el sistema.

### 6.3.5. Arquitectura de alto nivel

**Figura 15. Diagrama de arquitectura**



En este diagrama podemos evidenciar como la arquitectura MVC operara en nuestro Sistema donde en la parte de las vistas tenemos las interfaces que, de usuarios tanto para vendedores como para clientes, en los controladores tenemos la lógica de negocio donde se realizara la gestión de usuarios, productos, pedidos y carritos de compra, y finalmente en los modelos tenemos la fachada y los DAO's que administraran la comunicación a la base de datos en la nube.

En un principio se planteó utilizar la arquitectura por microservicios, pero en el proceso se llegó a la conclusión que para disminuir los problemas de seguridad Firebase es la mejor opción para mejorar la comunicación además permite el escalamiento de la App en caso de ser requerido.

Para definir la arquitectura a utilizar en este sistema se utilizaron los patrones DAOS y FACADE con el fin de mantener independiente la comunicación a la base datos.

Las ventajas de implementar la arquitectura por MVC en este sistema son:

- La separación del Modelo de la Vista, es decir, separar los datos de la representación visual de los mismos.
- Es mucho más sencillo agregar múltiples representaciones de los mismos datos o Información.
- Facilita agregar nuevos tipos de datos según sea requerido por la aplicación ya que son independientes del funcionamiento de las otras capas.
- Crea independencia de funcionamiento.
- Facilita el mantenimiento en caso de errores.
- Ofrece maneras más sencillas para probar el correcto funcionamiento del sistema.
- Permite el escalamiento de la aplicación en caso de ser requerido.

## 7. Construcción

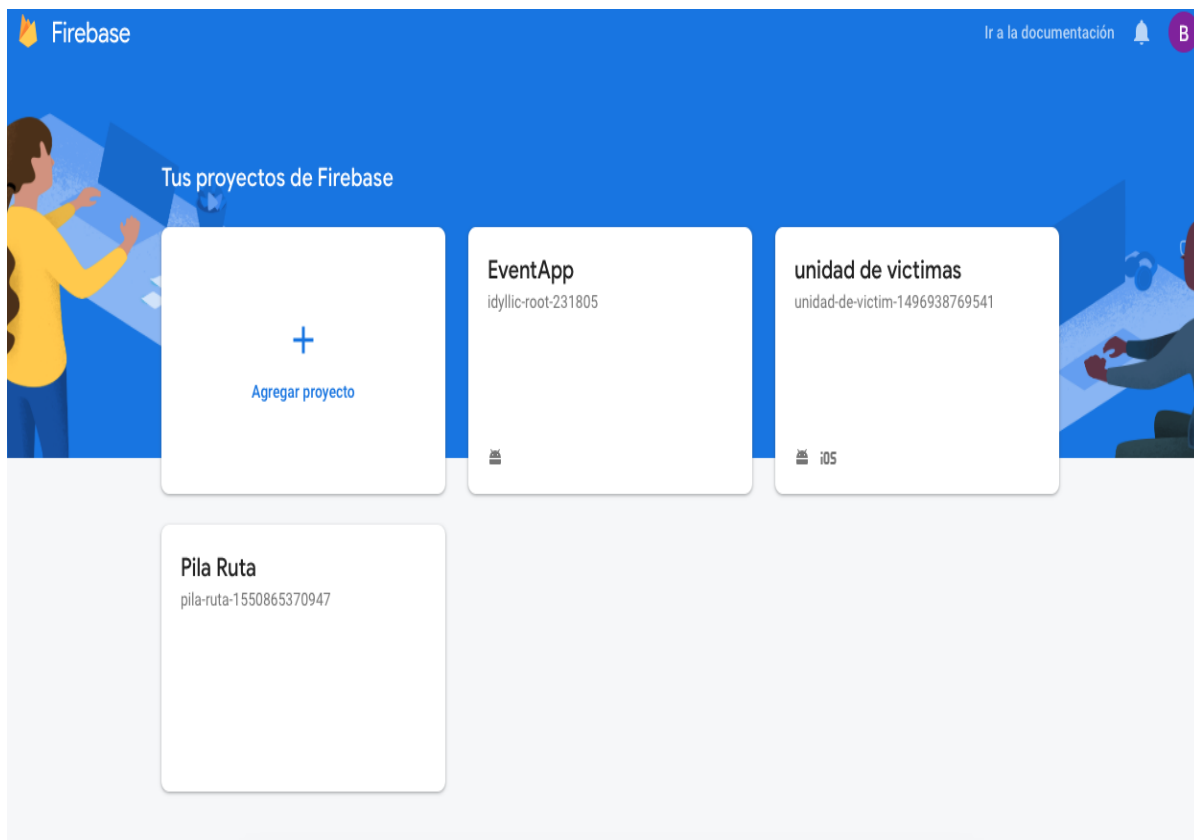
Para realizar la construcción de la aplicación se utilizaron en esencia dos tecnologías Firebase para la gestión de login de los usuarios y manejo de la base de datos y Android Studio Code para la arquitectura y el desarrollo.

En el proceso se siguieron los siguientes pasos:

### 7.1. Creación de proyecto en Firebase

Se realiza la creación del proyecto en Firebase Console con una cuenta de correo de Google.

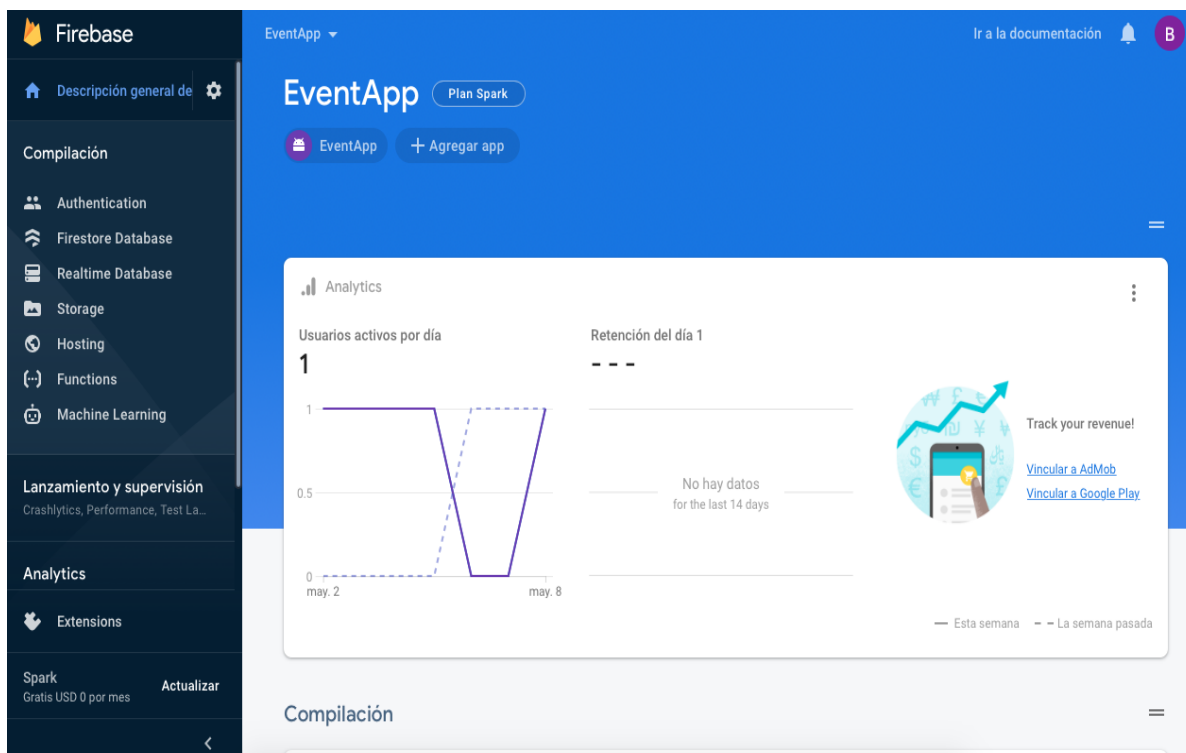
**Figura 16. Creación del proyecto en Firebase**



Fuente: Elaboración propia



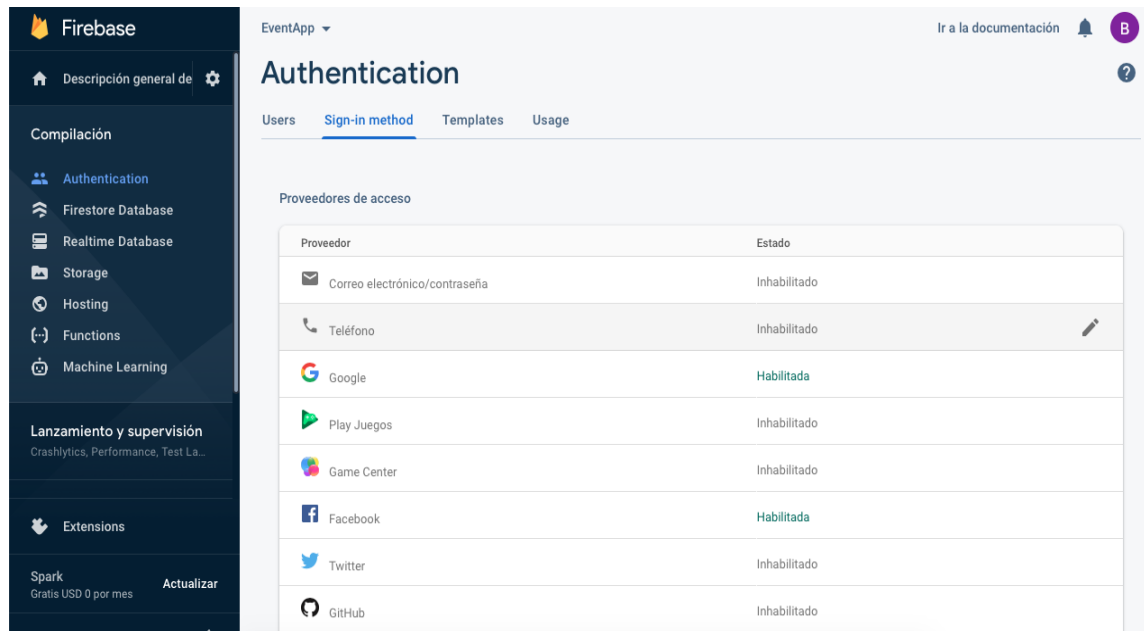
**Figura 17. Creación del proyecto en Firebase**



Fuente: Elaboración propia

## **Habilitar tipos de login con Firebase**

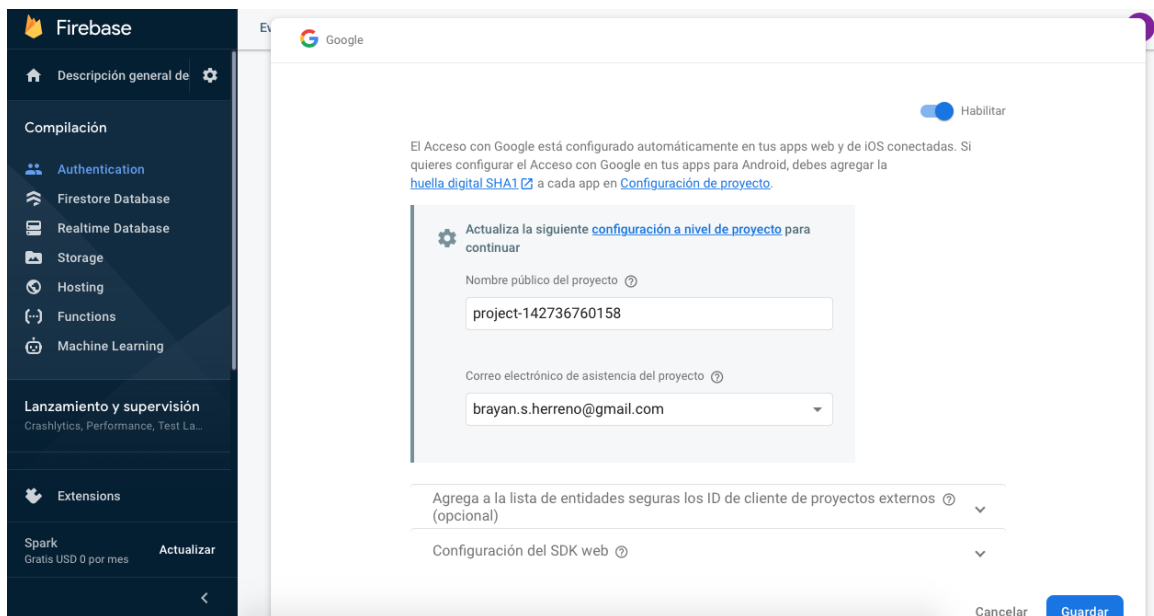
Se realiza la habilitan los tipos de login (Facebook y Google).

**Figura 18. Habilitar los tipos de login (Facebook y Google)**

Fuente: Elaboración propia

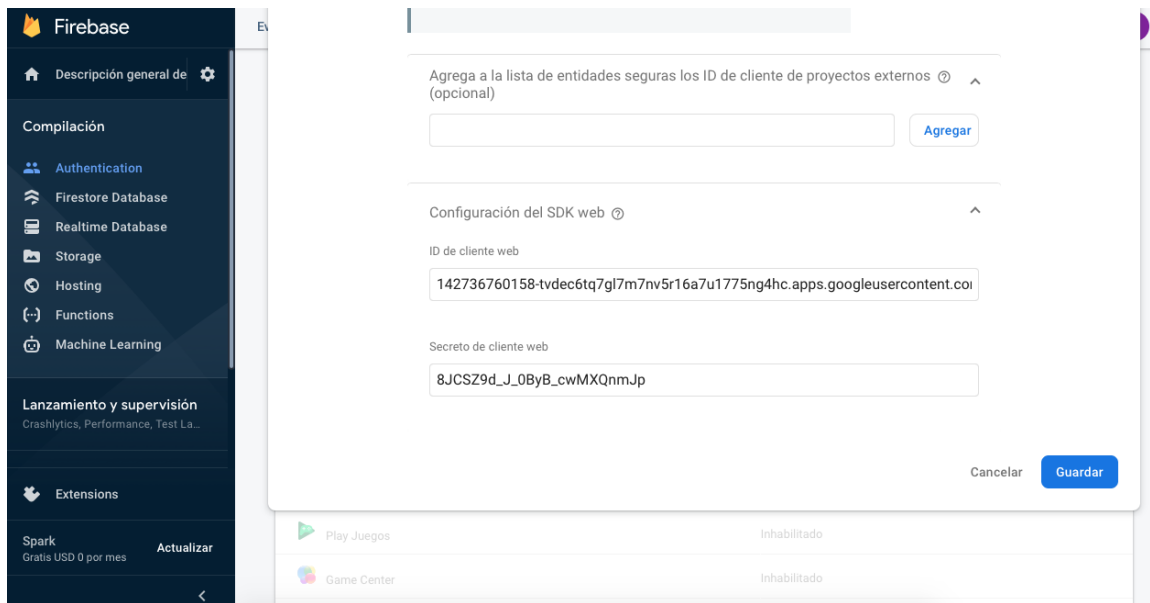
### Configurar login con Google

Se realiza la respectiva configuración para habilitar la llave SHA1 del nuestro aplicativo para permitir el acceso a la autenticación de Firebase con Google.

**Figura 19. Configurar login con Google**

Fuente: Elaboración propia

**Figura 20. Configurar login con Google**

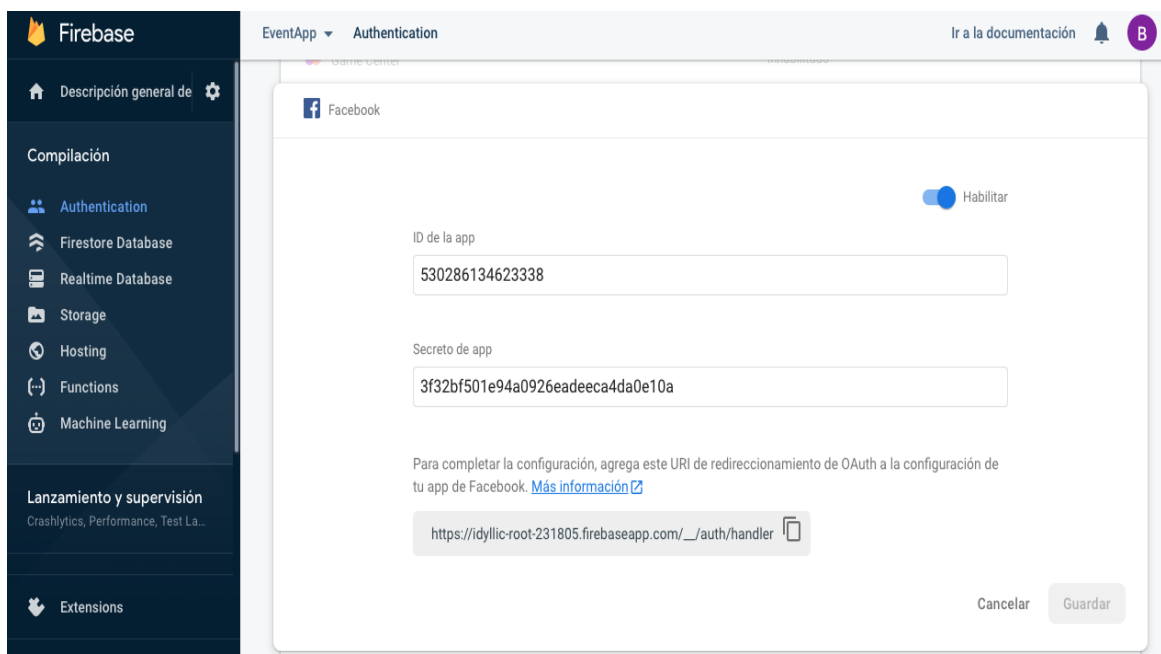


Fuente: Elaboración propia

## Habilitar y configurar login con Facebook

Se realiza la respectiva configuración para habilitar la llave SHA1 del nuestro aplicativo para permitir el acceso a la autenticación de Firebase con Facebook.

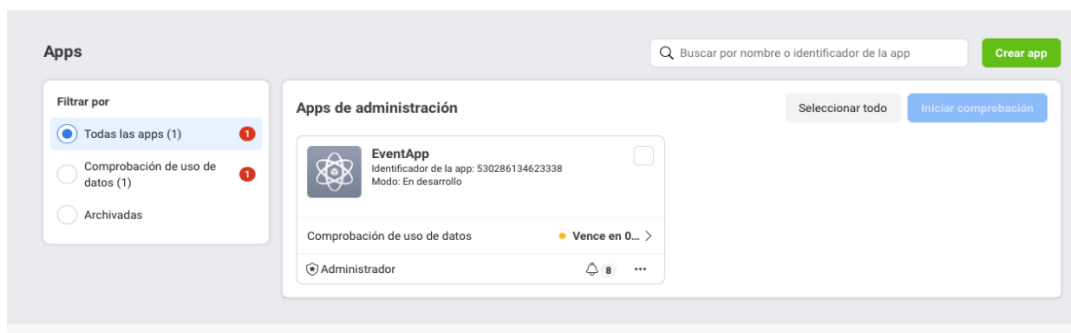
### Figura 21. Habilitar y configurar login con Facebook



Fuente: Elaboración propia

Se realiza la creación de proyecto en modo desarrollador de Facebook

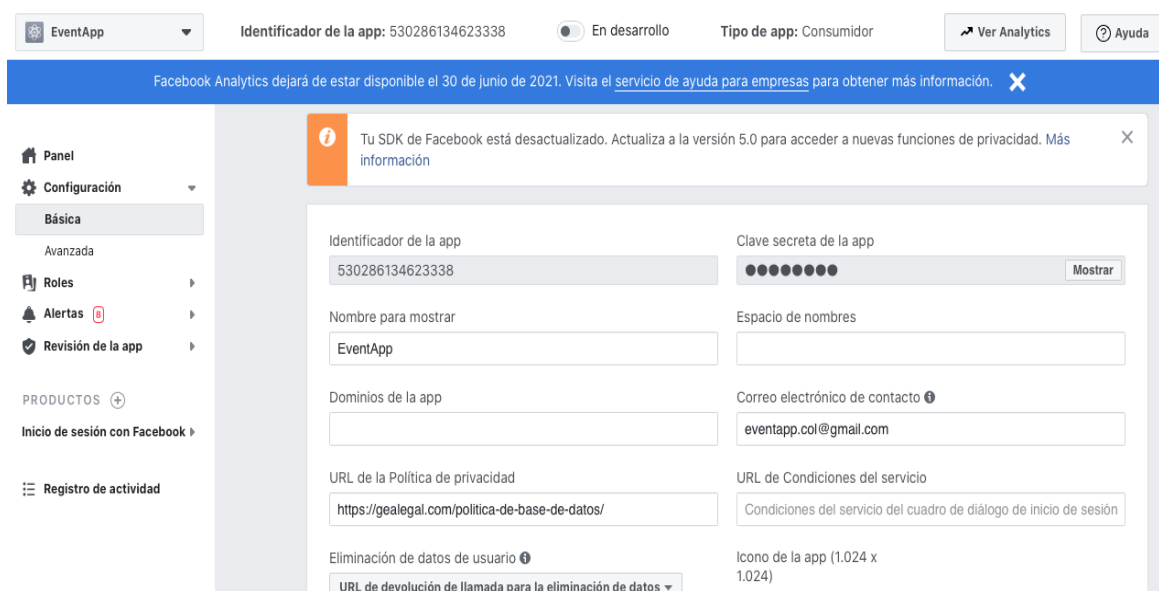
**Figura 22. Creación de proyecto en modo desarrollador de Facebook**



Fuente: Elaboración propia

Se realiza la configuración básica para el proyecto en Facebook

**Figura 23. Configuración del proyecto en Facebook**



Fuente: Elaboración propia

**Figura 24. Configuración básica para el proyecto en Facebook**

The screenshot shows the Facebook Developers app configuration interface. At the top, it says "FACEBOOK for Developers" and includes navigation links for "Documentos", "Herramientas", "Ayuda", and "Mis apps". A search bar is present with the text "Buscar documentación para desarrolladores". Below this, the app name "EventApp" is shown, along with the app ID "Identificador de la app: 530286134623338" and the status "En desarrollo". The app type is "Consumidor". There are buttons for "Ver Analytics" and "Ayuda".

A blue banner at the top of the main content area states: "Facebook Analytics dejará de estar disponible el 30 de junio de 2021. Visita el [servicio de ayuda para empresas](#) para obtener más información." followed by a close button (X).

The left sidebar contains a navigation menu with items: "Panel", "Configuración" (expanded to show "Básica" and "Avanzada"), "Roles", "Alertas" (with a notification icon), "Revisión de la app", "PRODUCTOS", "Inicio de sesión con Facebook", and "Registro de actividad".

The main content area is titled "Verificación". It contains the text: "Algunos permisos y funciones requieren verificación individual o del negocio para permitirte acceder a ciertos tipos de datos. Puedes iniciar el proceso de verificación en cualquier momento." and a button labeled "Iniciar verificación".

Below this is a section titled "Información de contacto del delegado de protección de datos". It explains that the GDPR requires companies to designate a data protection officer. It includes a text input field for "Nombre (opcional)".

Fuente: Elaboración propia

Se registra la llave SHA1 y el Package de nuestra aplicación.

### Figura 25. Registro de la llave SHA1 y el Package

This screenshot shows the "Android" configuration step in the Facebook Developers app configuration interface. The top navigation and app information are identical to the previous screenshot.

The main content area is titled "Android" and includes a "Inicio rápido" button with a close icon (X). It contains several form fields:

- "Nombre del paquete de Google Play": A text input field containing "com.example.eventapp".
- "Nombre de la clase": A text input field containing "com.example.eventapp.LoginActivity".
- "Hashes de clave": A text input field containing two SHA-1 hashes: "lqnlGxfA5i5T10d7NGoitZpEpo=" and "8fNdsKCSM6c2lFRya6urjdZb07E=".
- "URL de Amazon Appstore (opcional)": A text input field with the example "http://www.amazon.com/dp/B004GJDQT8".

At the bottom, there are two toggle switches:

- "Inicio de sesión único": A toggle switch set to "No". Below it, the text reads: "Se iniciará desde las notificaciones de Android".
- "Registro automático de eventos en la app (recomendado)": A toggle switch set to "Si". Below it, the text reads: "Si activas esta opción, se registrarán automáticamente los eventos que se produzcan en la app (compras, inicios de períodos de prueba y suscripciones) procesados a través de Google Play Store. Para registrar automáticamente eventos de compra, usa el SDK de Facebook para Android v.4.36 o superior. Si quieres registrar las suscripciones y los períodos de prueba, necesitas el SDK de Facebook para Android v5.1. Nota: Una vez que actives esta opción, deberás dejar de registrar manualmente estas acciones en Android para evitar que aparezcan duplicadas en los informes. Más información".

At the bottom right, there are two buttons: "Descartar" and "Guardar cambios".

Fuente: Elaboración propia

## Creación de archivo de permisos con Firebase

Se realiza la configuración de archivo para permiso y comunicación con Firebase

**Figura 26. Configuración de permiso y comunicación con Firebase**

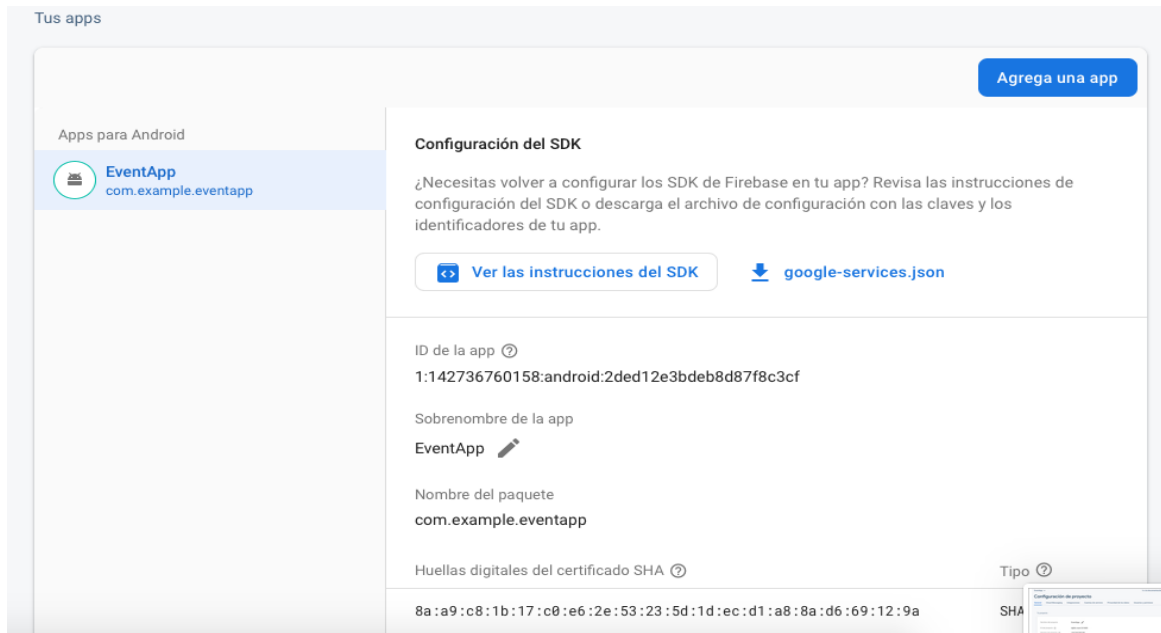
The screenshot shows the 'Configuración de proyecto' (Project Configuration) page in the Firebase console. The page is titled 'EventApp' and includes a navigation menu with options: General, Cloud Messaging, Integraciones, Cuentas de servicio, Privacidad de los datos, and Usuarios y permisos. The 'General' tab is selected. Under the heading 'Tu proyecto', there is a table of project details:

Nombre del proyecto	EventApp
ID del proyecto	idyllic-root-231805
Número de proyecto	142736760158
Ubicación predeterminada de los recursos de GCP	southamerica-east1
Clave de API web	AlzaSyDcgUCe4xu7zEzDYpU5NAh80Se4BbosxOA

Below this table is the 'Configuración pública' (Public Configuration) section, which controls the instances of the project shown to the public. It includes a 'Nombre público' (Public Name) field set to 'project-142736760158' and a 'Correo electrónico de asistencia' (Support Email) field set to 'brayan.s.herreno@gmail.com'.

Fuente: Elaboración propia

**Figura 27. Configuración de archivo para permiso y comunicación con Firebase**

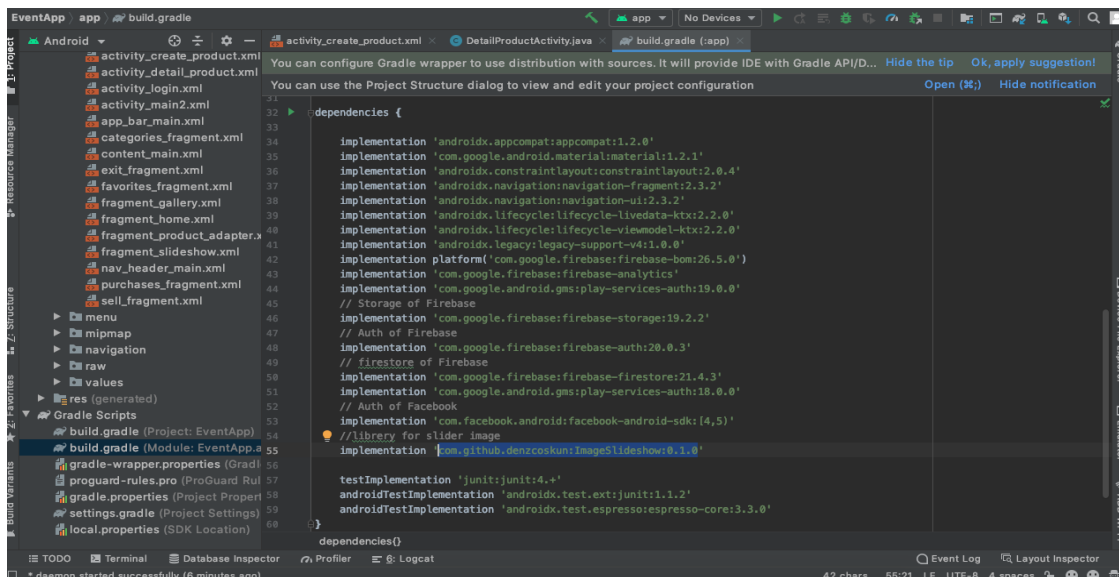


Fuente: Elaboración propia

## Implementación de dependencias con Firebase

Realizamos el llamado de las dependencias en el archivo Gradle del proyecto

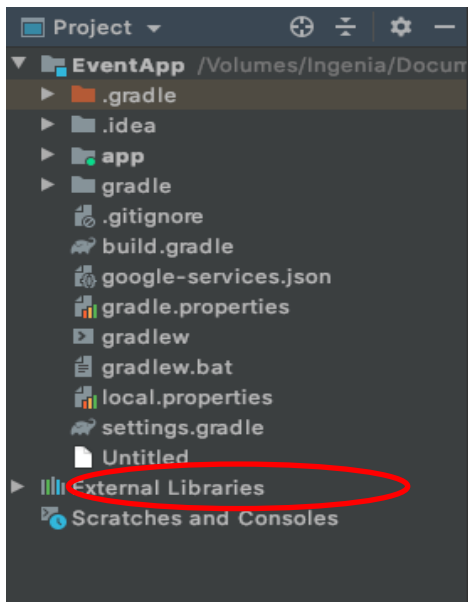
**Figura 28. Llamado del archivo Gradle del proyecto**





Fuente: Elaboración propia

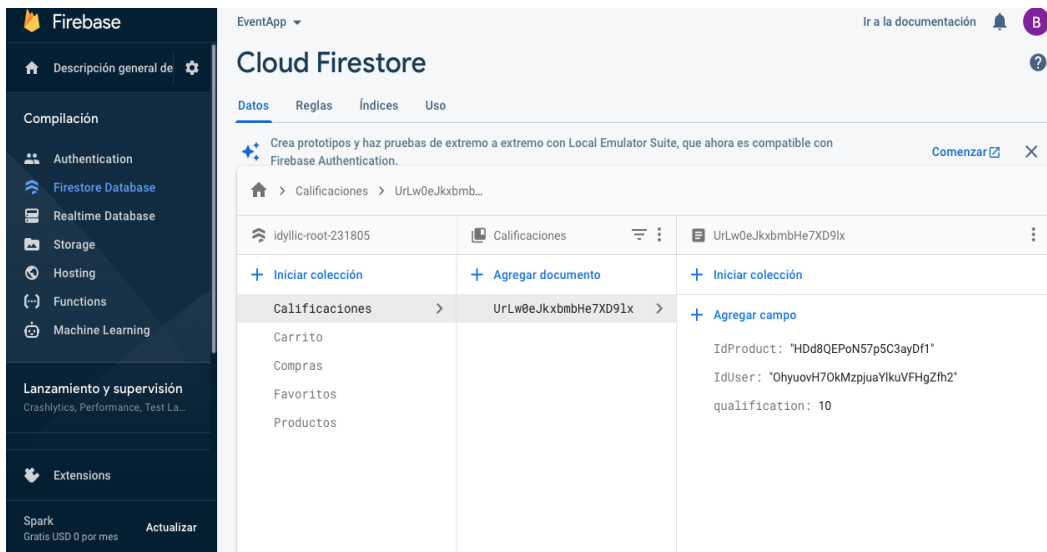
**Figura 29. Llamado de las dependencias en el archivo Gradle del proyecto**



Fuente: Elaboración propia

## 7.2. Creación de base de datos con Firestore

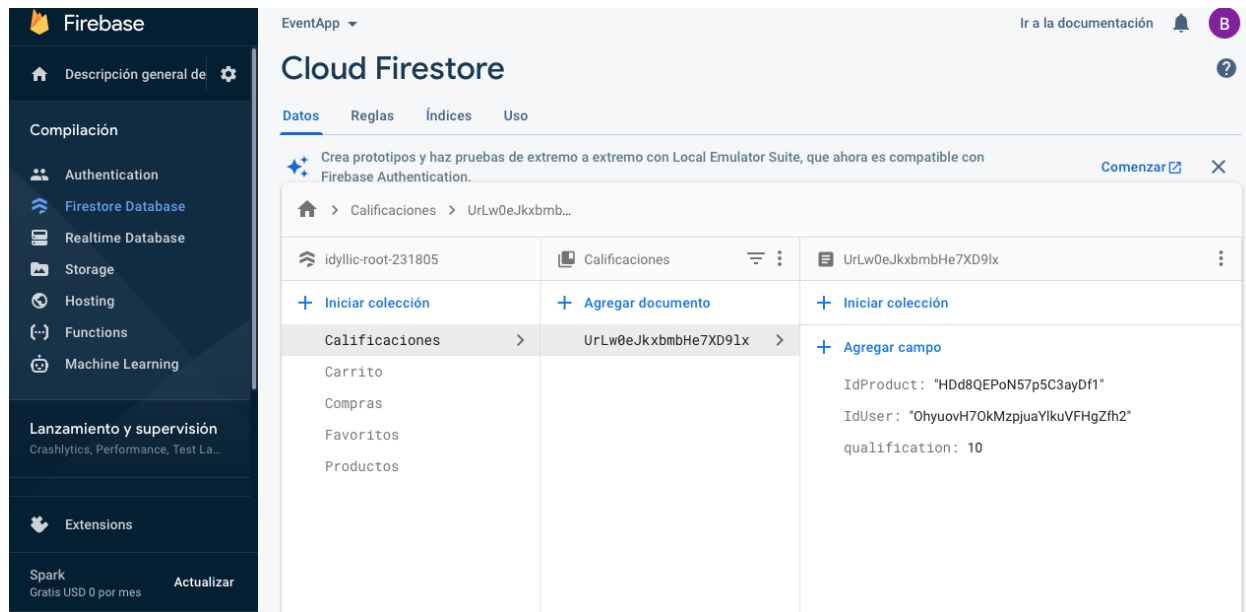
**Figura 30. Creación de base de datos con Firestore**



Fuente: Elaboración propia

## Habilitar e implementar bases de dos Firestore

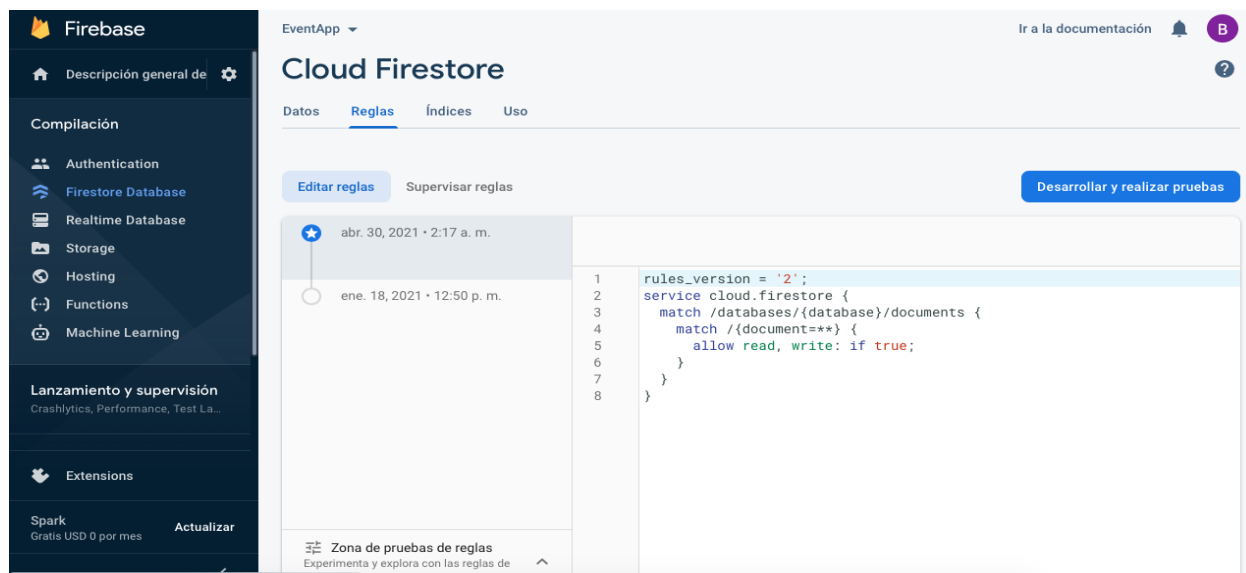
**Figura 31. Habilitar e implementar bases de dos Firestore**



Fuente: Elaboración propia

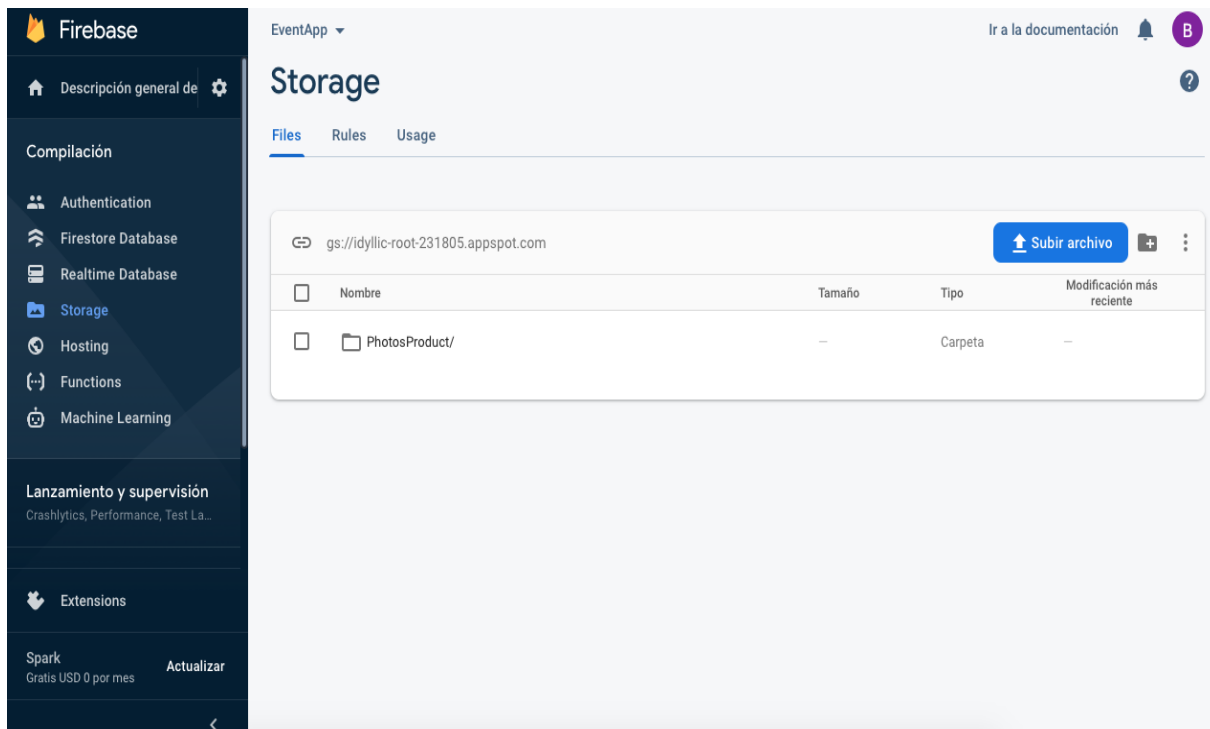
Se realiza la configuración de acceso a la base de datos

**Figura 32. Configuración de acceso a la base de datos**



Fuente: Elaboración propia

**Habilitar e implementar store de archivos con Firebase**

**Figura 33. Habilitar e implementar store**

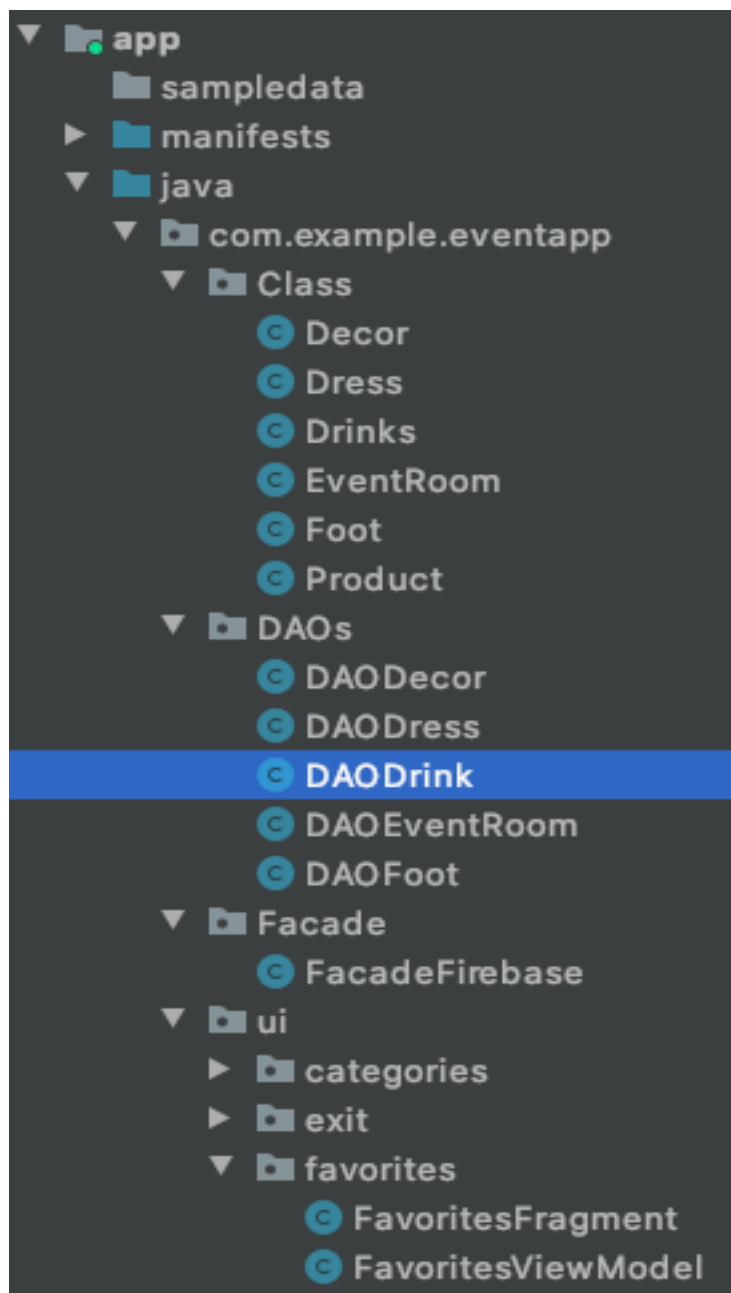
The screenshot displays the Firebase Storage console interface. On the left is a dark sidebar with the 'Storage' option highlighted. The main content area shows the storage bucket 'gs://idyllic-root-231805.appspot.com' with a 'Subir archivo' button. Below this is a table listing the contents of the bucket.

Nombre	Tamaño	Tipo	Modificación más reciente
PhotosProduct/	-	Carpeta	-

Fuente: Elaboración propia

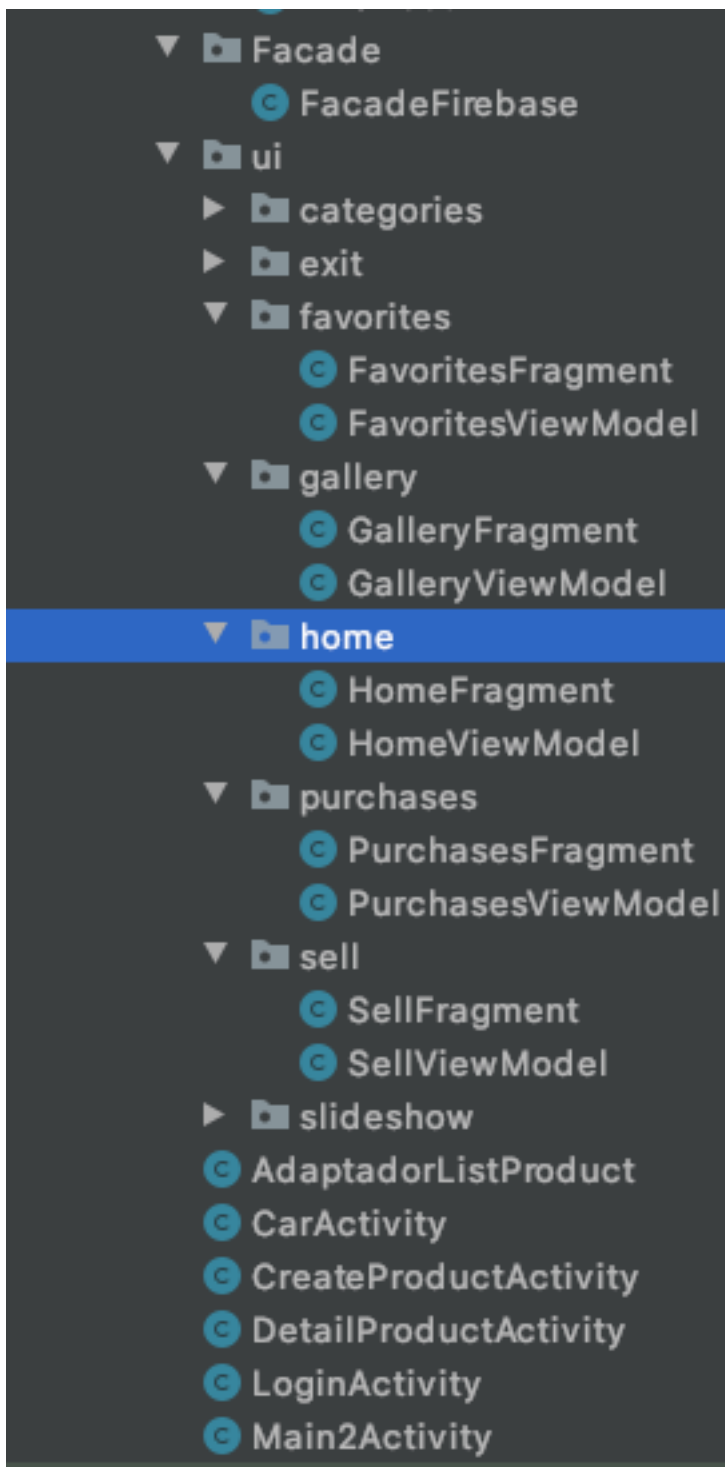
### 7.3. Creación de arquitectura y desarrollo en Android Studio

Figura 34. Desarrollo en Android Studio



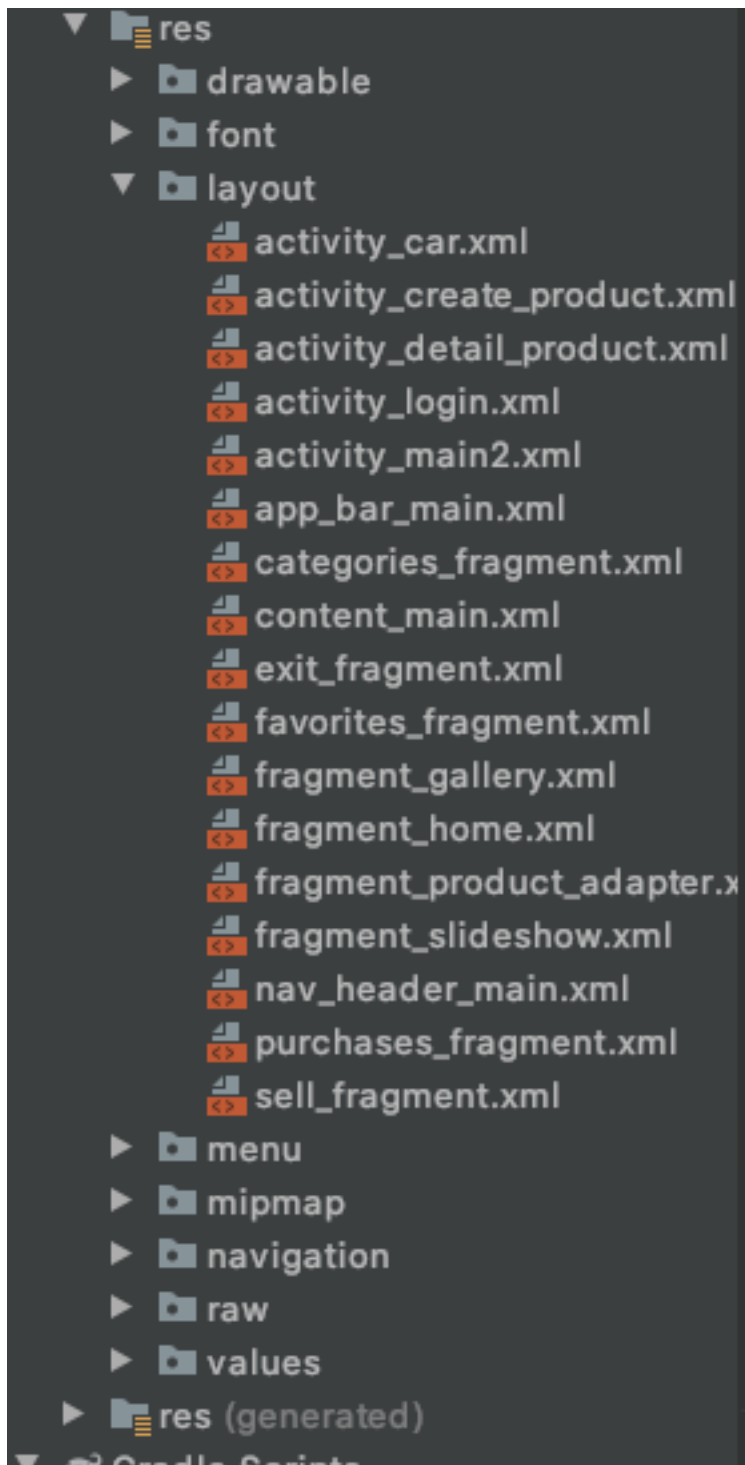
Fuente: Elaboración propia

Figura 35. Desarrollo en Android Studio



Fuente: Elaboración propia

**Figura 36. Desarrollo en Android Studio**



Fuente: Elaboración propia

## 8. Pruebas

**Tabla 10. Escenarios de calidad**

ESCENARIO	DESCRIPCION	PRIORIDAD
Seguridad	Los usuarios y contraseñas en la autenticación deben ser protegidos para garantizar la no suplantación de los usuarios y acceso no permitido por terceros.	Alta
Disponibilidad	La información del sistema debe estar respaldada ante cualquier evento de tal manera que se garantice su disponibilidad y manejo de históricos.	Alta
Desempeño	El sistema debe estar en capacidad de atender las solicitudes recurrentes que se puedan presentar.	Alta
Latencia	El sistema debe mostrar la respuesta a cada acción en tiempos cortos para mantener la atención del usuario centrado.	Alta
Usabilidad	La navegación por el sistema debe ser intuitiva para mejorar la experiencia del usuario.	Media

Fuente: Elaboración propia

Para la realización de este proyecto se plantearon los anteriores escenarios de calidad sobre los cuales serán realizadas las pruebas.

**Tabla 11. Tabla de riesgos de seguridad**

Código	Riesgo	Activo de Información	Causas	Consecuencias	Frecuencia	Impacto	Riesgo Inherente
1	Falta de disponibilidad de la información alojada en servidor en la nube	Datos del aplicativo	Mala manipulación del administrador del servidor	Interrupción de la operación	Poco Frecuente (1)	5	Alto

2	Pérdida, fuga, borrado intencional y/o accidental de datos de los servidores o unidades de almacenamiento	Base de datos del aplicativo	Virus informáticos y/o Ataques Informáticos.	Reprocesos Afectación de la Imagen Hallazgos de entes regulatorios Interrupción a las operaciones	Poco Frecuente (1)	5	<b>Alto</b>
3	Detrimiento en el desempeño y capacidad de los sistemas, afectando la disponibilidad	Servidores	No se cuenta con una gestión de la capacidad No existen políticas para la gestión de la capacidad	Interrupción en las operaciones	Moderado (3)	4	<b>Alto</b>
4	falsa identidad de los usuarios del aplicativo		No tener una validación adecuada para creación de perfiles en el aplicativo	Oferta de servicios y espacios inexistentes	Moderado (3)	3	<b>Moderado</b>

Fuente: Elaboración propia

Para la realización de este proyecto se tuvieron en cuenta algunos posibles riesgos que se pueden presentar y estos fueron tenidos en cuenta a la hora de la ejecución de las pruebas.



**Tabla 12. Tabla de controles a los riesgos de seguridad**

No.	Nombre de Control	Descripción del Control	Clase	Tipo	Valor control	Riesgo Residual
1.1	Backup Interno	El área de infraestructura construirá y será el responsable del proceso o procesos relacionado con la ejecución de los controles	Correctivo	Semiautomático	3	<b>Excelente</b>
1.2	Mantenimientos preventivos	El proveedor, antes del inicio de los trabajos deberá presentar un plan de trabajo, en la que detalle los cronogramas de las actividades, personal técnico especialista que realizará los trabajos, recursos y facilidades necesarias, este plan de trabajo será aprobado previamente por el área de informática.	Preventivo	Semiautomático	5	
2.1	Controles de acceso (ADDS).	El administrador debe garantizar que los usuarios prueben ser quienes dicen que son, a través de credenciales o utilización de uno o varios métodos de autenticación.	Preventivo	Semiautomático	5	

2.2	Gestión de privilegios. (ADDS, y Sistemas de Información)	Garantizar el acceso a los usuarios autorizados y evitar el acceso no autorizado a los sistemas, información y servicios de ESAP	Preventivo	Semiautomático	5	
3.1	Establecer procedimientos de gestión de capacidad.	Deberá asegurar que los servicios y recursos de TI se vean respaldados por una capacidad de procesamiento y almacenamiento suficiente y correctamente dimensionada, que garantice que los clientes y usuarios de los servicios puedan desempeñar de una manera eficiente sus tareas.	Correctivo	Manual	2	Regular
3.2	Realizar el monitoreo permanente de la capacidad de la infraestructura a través de umbrales.	Implementar herramientas de monitoreo de infraestructura que vigilen el comportamiento de la misma, basado en niveles máximo y mínimo de uso de los recursos asociados a la capacidad de los componentes. Cantidad de uso de procesador, de memoria, de disco, de red, temperatura de los componentes, etc.	Correctivo	Automático	4	

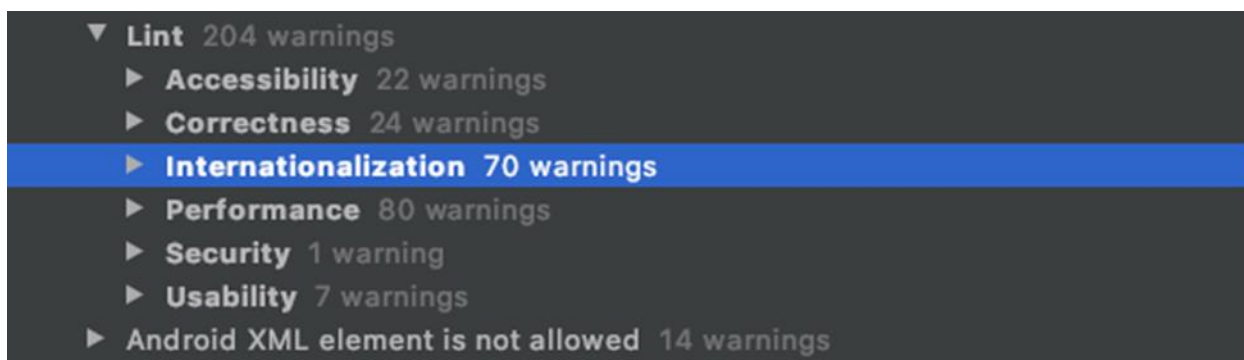
4.1	Autenticación en el aplicativo con cuentas externas	Implementar autenticación con cuentas como Google, Facebook, Twitter, etc. con el fin de dar un filtro a la autenticidad de la identidad de los usuarios.	Preventivo	Automático	6	<b>Moderado</b>
-----	---	---	------------	------------	---	-----------------

Fuente: Elaboración propia

En caso de presentarse los riesgos que se describen en la tabla 11 se plantearon las soluciones a estos inconvenientes.

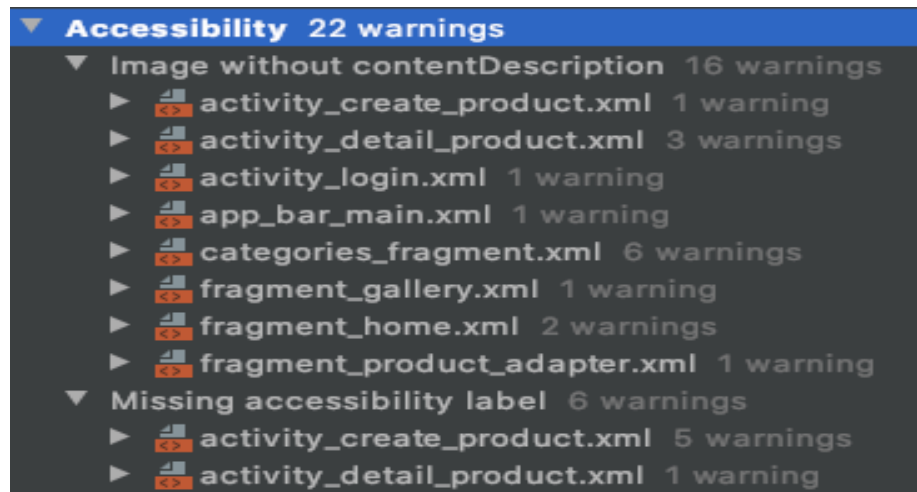
### Pruebas en LINT

**Figura 37. Pruebas en LINT**



Fuente: Elaboración propia

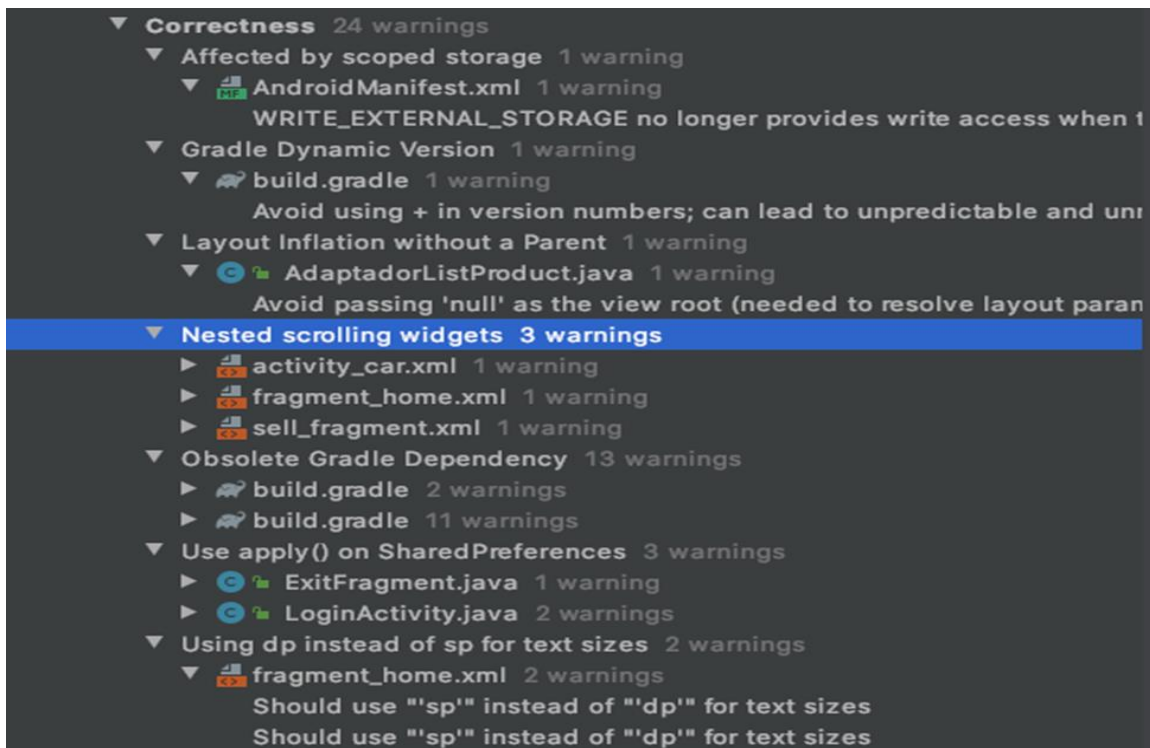
**Figura 38. Prueba de accesibilidad**



Fuente: Elaboración propia

- a. Los widgets no textuales como `ImageViews` e `ImageButtons` deben usar el atributo `contentDescription` para especificar una descripción textual del widget de modo que los lectores de pantalla y otras herramientas de accesibilidad puedan describir adecuadamente la interfaz de usuario. Teniendo en cuenta que los elementos en las pantallas de la aplicación que son puramente decorativos y no proporcionan ningún contenido o permiten una acción del usuario no deben tener descripciones de contenido de accesibilidad. En este caso, simplemente suprima la advertencia con un atributo `tools: ignore = "ContentDescription"`. Tenga en cuenta que, para los campos de texto, no debe establecer tanto la sugerencia como los atributos `contentDescription`, ya que la sugerencia nunca se mostrará. Solo establece la pista.
- b. Falta la etiqueta de accesibilidad Los campos de texto editables deben proporcionar un `android: hint` o, siempre que su `minSdkVersion` sea al menos 17, pueden ser referenciados por una vista con un atributo `android: labelFor`. Cuando use `android: labelFor`, asegúrese de proporcionar un `android: text` o un `android: contentDescription`. Si su vista está etiquetada, pero con una etiqueta en un diseño diferente que incluye esta.

**Figura 39. Prueba de exactitud**



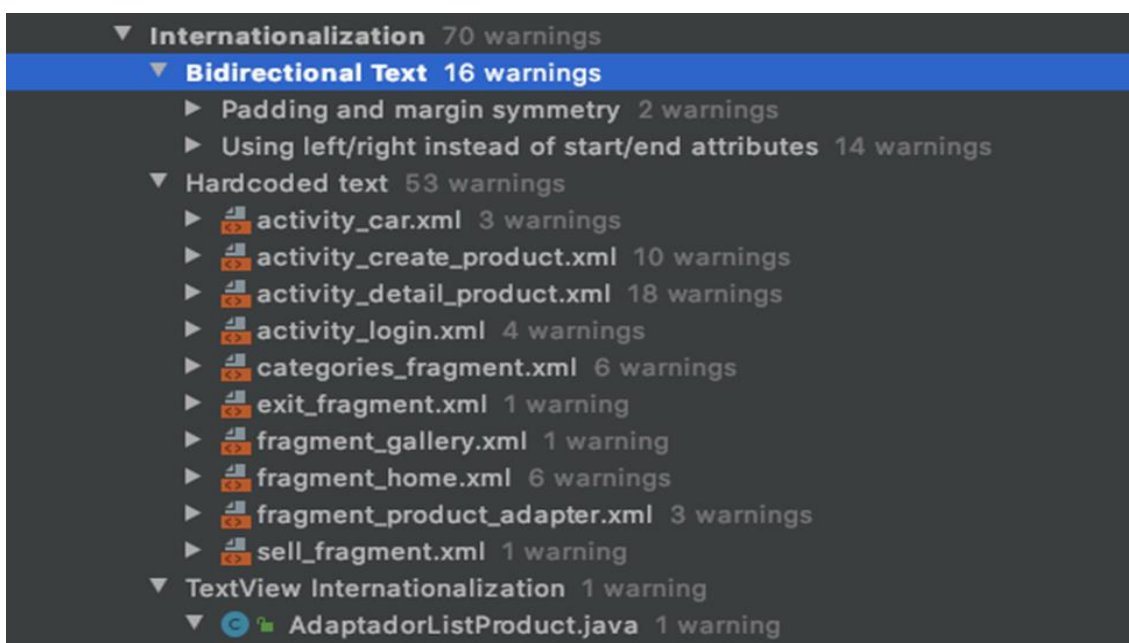
Fuente: Elaboración propia

a. El almacenamiento de alcance se aplica en Android 10+ (o Android 11+ si se usa `requestLegacyExternalStorage`). En particular, `WRITE_EXTERNAL_STORAGE` ya no proporcionará acceso de escritura a todos los archivos; en su lugar, proporcionará el equivalente de `READ_EXTERNAL_STORAGE`. El permiso `MANAGE_EXTERNAL_STORAGE` se puede usar para administrar todos los archivos, pero rara vez es necesario y la mayoría de las aplicaciones en Google Play no pueden usarlo. En su lugar, la mayoría de las aplicaciones deberían migrar para utilizar el almacenamiento de ámbito. Para modificar o eliminar archivos, las aplicaciones deben solicitar acceso de escritura al usuario.

- b. El uso de + en dependencias le permite elegir automáticamente la última versión disponible en lugar de una versión específica con nombre. Como sea, esto no es recomendable; tus construcciones no son repetibles; es posible que haya probado con una versión ligeramente diferente a la que utilizó el servidor de compilación. (Usar una versión dinámica como número de versión principal es más problemático que usarla en la posición de versión secundaria).
- c. Al inflar un diseño, evite pasar nulo como vista principal, ya que de lo contrario se ignorarán los parámetros de diseño en la raíz del diseño inflado.
- d. Un widget de desplazamiento, como ScrollView, no debe contener ningún widget de desplazamiento anidado, ya que tiene varios problemas de usabilidad.
- e. Este detector busca usos de bibliotecas donde la versión que está utilizando no es la versión estable actual. El uso de versiones anteriores está bien, y hay casos en los que deliberadamente desea seguir con una versión anterior. Sin embargo, es posible que simplemente no sepa que hay una versión más reciente disponible, y eso es lo que ayuda a encontrar esta comprobación.
- f. Considere utilizar apply () en lugar de confirmar las preferencias compartidas. Mientras que commit bloquea y escribe sus datos en el almacenamiento persistente inmediatamente, apply lo manejará en segundo plano.
- g. Al configurar los tamaños de texto, normalmente debería utilizar sp, o "píxeles independientes de la escala". Es como la unidad dp, pero también se escala según la preferencia de tamaño de fuente del usuario. Se recomienda que utilice esta unidad al especificar tamaños de fuente, para que se ajusten tanto a la densidad de la pantalla como a

las preferencias del usuario. Hay casos en los que es posible que deba utilizar dp; normalmente, esto sucede cuando el texto está en un contenedor con un tamaño de dp específico. Esto evitará que el texto se derrame fuera del contenedor. Sin embargo, tenga en cuenta que esto significa que la configuración del tamaño de fuente del usuario no se respeta, así que considere ajustar el diseño para que sea más flexible.

**Figura 40. Pruebas de internacionalización**



Fuente: Elaboración propia

- a. Si especifica relleno o margen en el lado izquierdo de un diseño, probablemente también debería especificar relleno en el lado derecho (y viceversa) para la simetría del diseño de derecha a izquierda.
- b. El uso de Gravity # LEFT y Gravity # RIGHT puede generar problemas cuando se renderiza un diseño en lugares donde el texto fluye de derecha a izquierda. Utilice Gravity # START y Gravity # END en su lugar. De manera similar, en los atributos XML gravity y

layout\_gravity, use start en lugar de left. Para atributos XML como padding Left y layout\_marginLeft, use padding Start y layout\_marginStart. NOTA: Si su minSdkVersion es menor que 17, debe agregar tanto los atributos de izquierda / derecha más antiguos como los nuevos atributos de inicio / fin. En plataformas más antiguas, donde no se admite RTL y los atributos de inicio / fin son desconocidos y, por lo tanto, se ignoran, necesita los atributos de izquierda / derecha más antiguos. Hay una verificación de pelusa separada que detecta ese tipo de error.

c. Codificar los atributos de texto directamente en los archivos de diseño es malo por varias razones: \* Al crear variaciones de configuración (por ejemplo, horizontal o vertical), debe repetir el texto real (y mantenerlo actualizado al realizar cambios) \* La aplicación no se puede traducir a otros idiomas simplemente agregando nuevas traducciones para los recursos de cadenas existentes. Hay arreglos rápidos para extraer automáticamente esta cadena codificada en una búsqueda de recursos.

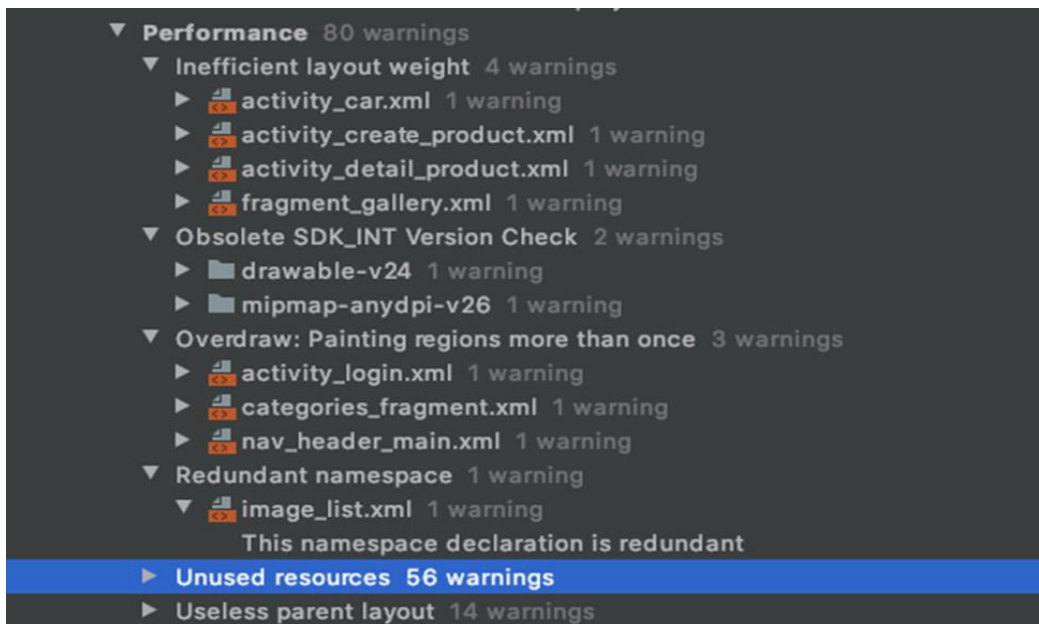
d. Al llamar a TextView # setText nunca llame a Number # toString () para formatear números; no manejará correctamente los separadores de fracciones y los dígitos específicos de la configuración regional. En su lugar, considere usar el formato String # con las especificaciones de formato adecuadas (% d o % f).

\* No pase una cadena literal (por ejemplo, "Hola") para mostrar texto. El texto codificado no se puede traducir correctamente a otros idiomas. En su lugar, considere usar cadenas de recursos de Android.

\* No cree mensajes concatenando fragmentos de texto. Estos mensajes no se pueden traducir correctamente.



**Figura 41. Prueba de desempeño**

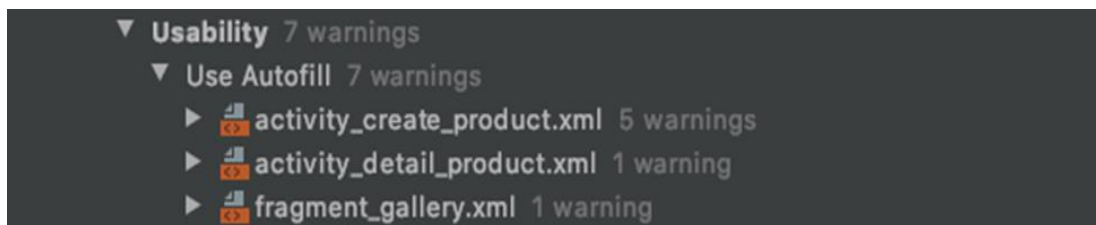


Fuente: Elaboración propia

- a. Cuando solo un widget en un LinearLayout define un peso, es más eficiente asignarle un ancho / alto de 0dp ya que absorberá todo el espacio restante de todos modos. Con un ancho / alto declarado de 0dp, no tiene que medir primero su propio tamaño.
- b. Esta verificación marca las verificaciones de versión que no son necesarias, porque minSdkVersion (o el nivel de API conocido circundante) ya es al menos tan alto como la versión verificada. De manera similar, también busca recursos en carpetas -vNN, como valores-v14 donde el calificador de versión es menor o igual que minSdkVersion, donde el contenido debe fusionarse en la mejor carpeta.
- c. Si configura un diseño de fondo en una vista raíz, entonces debe usar un tema personalizado donde el fondo del tema sea nulo. De lo contrario, el fondo del tema se pintará primero, solo para que su fondo personalizado lo cubra por completo; esto se llama "sobregiro".

- d. En documentos XML de Android, solo especifique el espacio de nombres en el elemento raíz / documento. Las declaraciones de espacio de nombres en otras partes del documento suelen ser restos accidentales de copiar / pegar XML de otros archivos o documentación.
- e. Los recursos no utilizados hacen que las aplicaciones sean más grandes y ralentizan las compilaciones. La verificación de recursos no utilizados puede ignorar las pruebas. Si desea incluir recursos a los que solo se hace referencia desde las pruebas, considere empaquetarlos en un conjunto de fuentes de prueba.
- f. Un diseño con hijos que no tiene hermanos, no es una vista de desplazamiento ni un diseño raíz, y no tiene fondo, se puede eliminar y sus hijos se mueven directamente al padre para una jerarquía de diseño más plana y eficiente.

#### Figura 42. Pruebas de seguridad



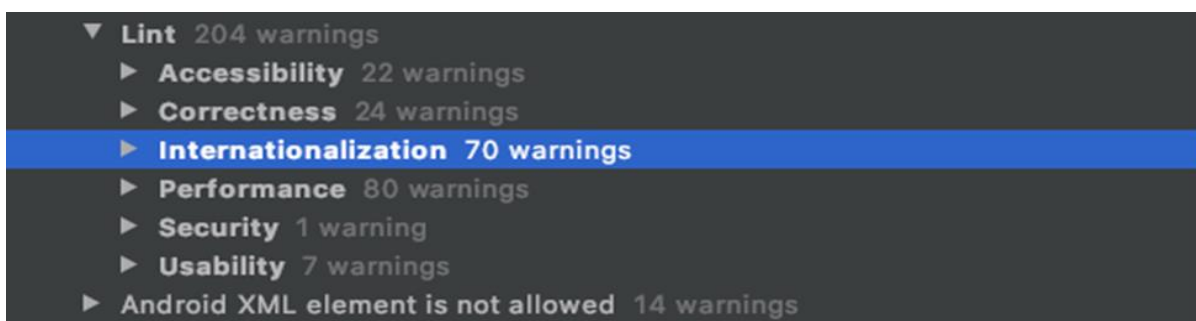
Fuente: Elaboración propia

- a. De forma predeterminada, este indicador se establece en verdadero, lo que significa que el sistema operativo puede hacer una copia de seguridad y restaurar los datos de la aplicación. La configuración de `allowBackup = "false"` impide que se realice una copia de seguridad de la aplicación y, por lo tanto, los usuarios no pueden restaurar los datos relacionados con ella cuando pasan por el asistente de configuración del dispositivo. Permitir copias de seguridad puede tener consecuencias de seguridad para una aplicación.

Actualmente, la copia de seguridad de adb permite a los usuarios que han habilitado la depuración USB copiar los datos de la aplicación fuera del dispositivo. Una vez realizada la copia de seguridad, el usuario puede leer todos los datos de la aplicación. adb restore permite la creación de datos de la aplicación a partir de una fuente especificada por el usuario.

Después de una restauración, las aplicaciones no deben asumir que los datos, los permisos de archivos y los permisos de directorio fueron creados por la propia aplicación. Para corregir esta advertencia, decida si su aplicación debe admitir la copia de seguridad y establezca explícitamente `android:allowBackup = (true | false) "`. Si no se establece en falso, y si se dirige a API 23 o posterior, lint también le advertirá que debe configurar `android:fullBackupContent` para configurar la copia de seguridad automática.

#### Figura 43. Prueba de usabilidad



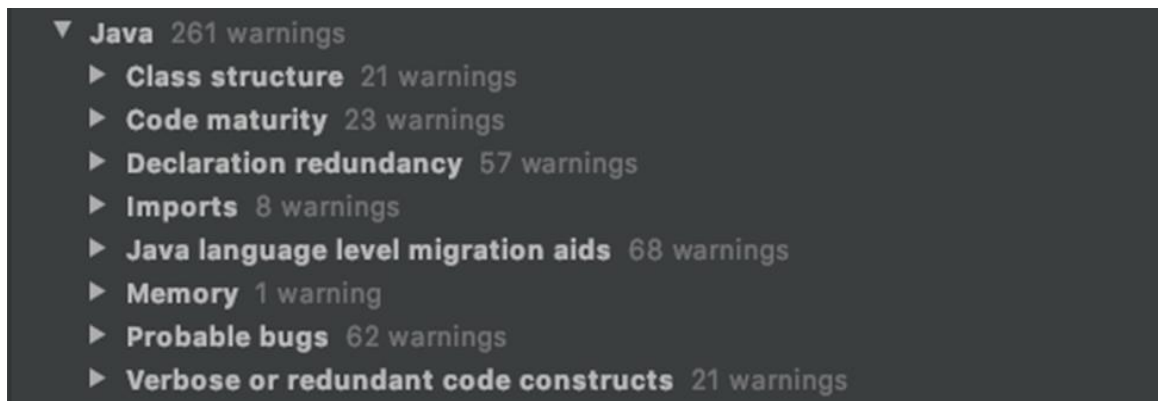
Fuente: Elaboración propia

a. Especifique un atributo `autofillHints` cuando tenga como destino la versión 26 o superior del SDK o especifique explícitamente que la vista no es importante para el autocompletado. Su aplicación puede ayudar a un servicio de autocompletar a clasificar los datos correctamente al proporcionar el significado de cada vista que podría ser autocompletable, como vistas que representan nombres de usuario, contraseñas, campos de tarjetas de crédito, direcciones de correo electrónico, etc. Las sugerencias pueden tener

cualquier valor, pero es se recomienda utilizar valores predefinidos como 'nombre de usuario' para un nombre de usuario o 'creditCardNumber'

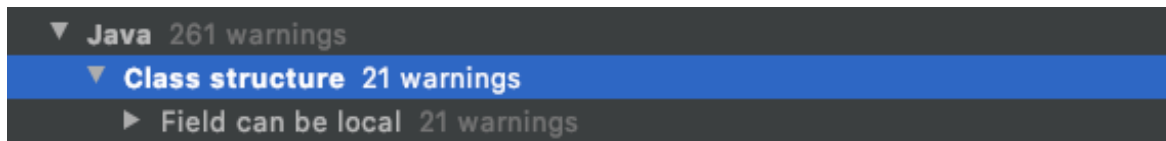
## INSPECTOR (JAVA)

**Figura 44. Pruebas en Inspector para JAVA**



Fuente: Elaboración propia

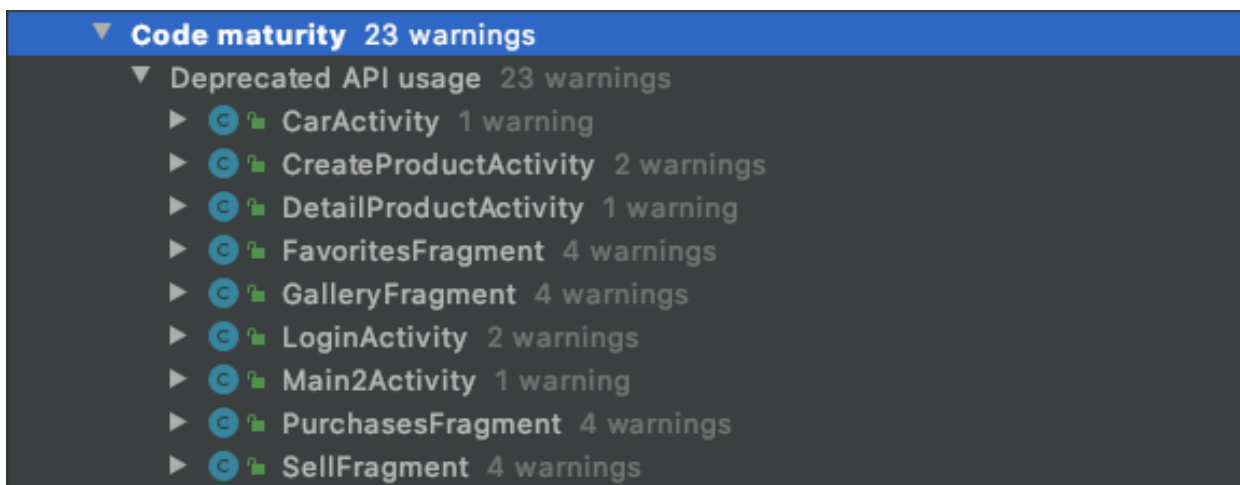
**Figura 45. Estructura de clases**



Fuente: Elaboración propia

Esta inspección busca campos de clase redundantes que se puedan reemplazar con variables locales. Si todos los usos locales de un campo están precedidos por asignaciones a ese campo, el campo puede eliminarse y sus usos reemplazados por variables locales.

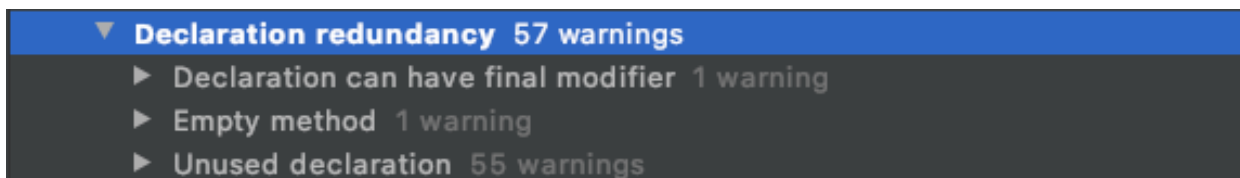
**Figura 46. Madurez de código**



Fuente: Elaboración propia

Informes donde se usa código obsoleto en el alcance de inspección especificado.

#### Figura 47. declaración de redundancia



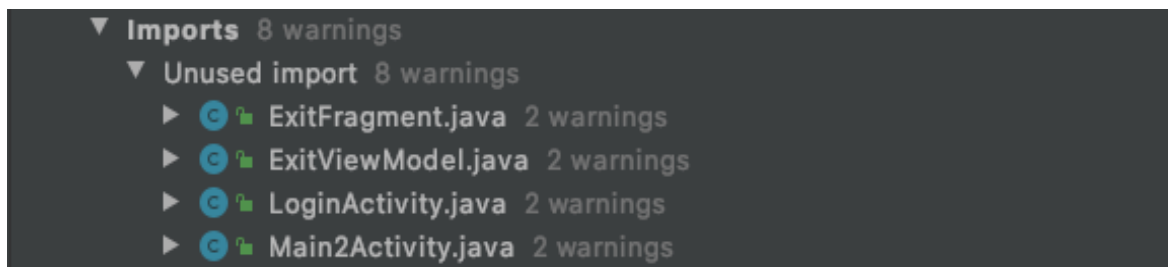
Fuente: Elaboración propia

1. Informa todos los campos, métodos o clases, que se encuentran en el alcance de inspección especificado, que pueden tener un modificador final agregado a sus declaraciones.
2. Informa métodos donde:
  - a. - el método está vacío O
  - b. - todas las implementaciones del método de interfaz están vacías O
  - c. - el método está vacío en sí mismo y solo se reemplaza por métodos vacíos Tenga en cuenta que un método que contiene solo la llamada super () y pasa su propio parámetro también se considera vacío.

3. Informa sobre clases, métodos o campos en el ámbito de inspección especificado que no se utilizan o no se pueden alcanzar desde los puntos de entrada. También informa de los parámetros que no utilizan sus métodos y de todas las implementaciones / sustituciones de métodos y las variables locales que se declaran, pero no se utilizan. Es posible que no se informe de algunos miembros no utilizados durante el resaltado en el editor. Por razones de rendimiento, un miembro no privado se verifica solo cuando su nombre rara vez aparece en el proyecto. Para ver todos los resultados, ejecute la inspección utilizando Analizar | Inspeccione el código ... o analice | Ejecutar inspección por nombre ...

## IMPORTS

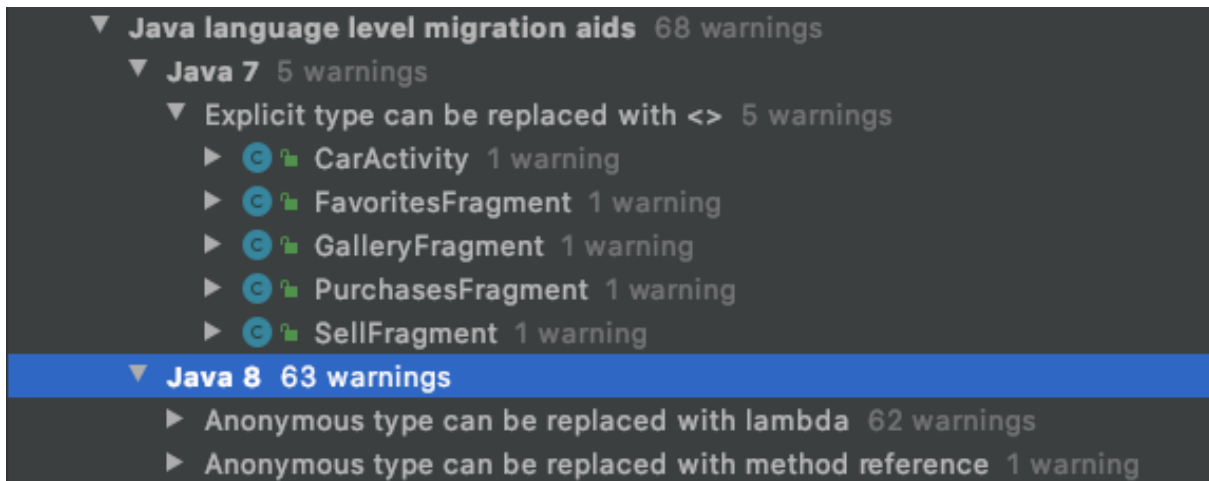
**Figura 48. IMPORTS**



Fuente: Elaboración propia

Reporta declaraciones de importación redundantes.

**Figura 49. Lenguaje JAVA nivel de migración**



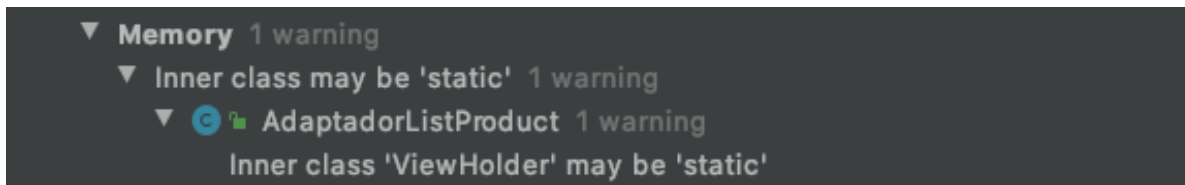
Fuente: Elaboración propia

Informa todas las expresiones nuevas con argumentos de tipo que se pueden reemplazar con el tipo de diamante <>

Informa todas las clases anónimas que se pueden reemplazar con expresiones lambda.

Reporta tipos anónimos que pueden ser reemplazados por referencias a métodos.

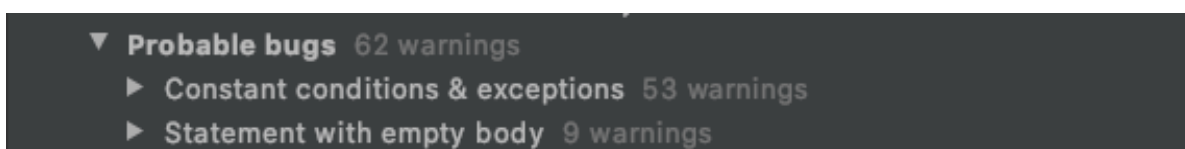
### Figura 50. Memoria



Fuente: Elaboración propia

Informa de las clases internas que se pueden convertir en estáticas de forma segura. Una clase interna puede ser estática si no hace referencia a su instancia adjunta.

### Figura 51. Probables Bugs

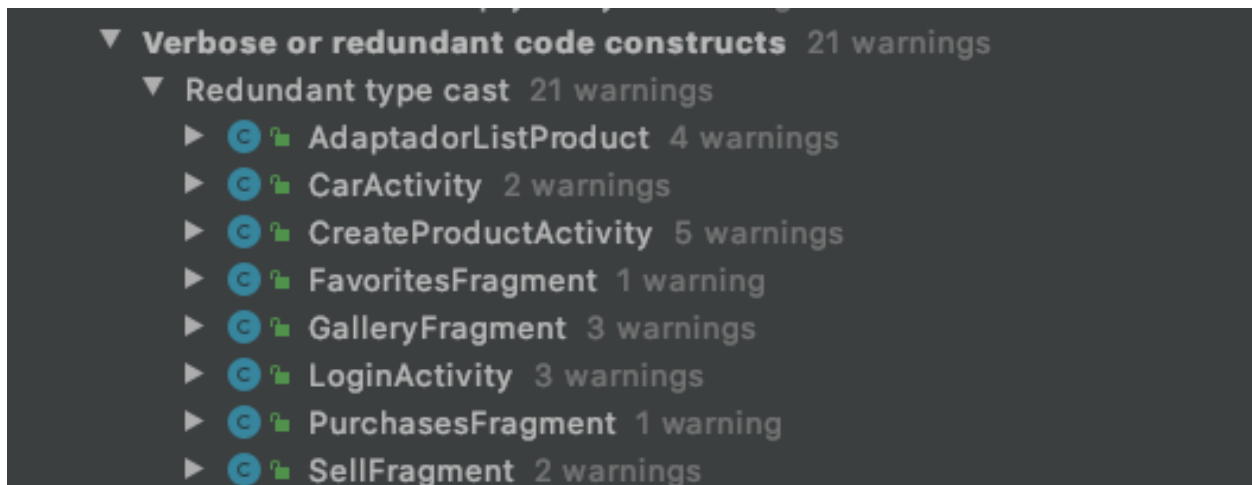


Fuente: Elaboración propia

Esta inspección analiza el control del método y el flujo de datos para reportar posibles condiciones que siempre son verdaderas o falsas, expresiones cuyo valor se comprueba estáticamente como constante y situaciones que pueden conducir a violaciones de contratos de nulabilidad.

Informa si, while, do, for y switch declaraciones que tienen cuerpos vacíos. Aunque en ocasiones se pretende, esta construcción es confusa y, a menudo, es el resultado de un error tipográfico.

### Figura 52. Redundancia



Fuente: Elaboración propia

Informa expresiones de reparto innecesarias.

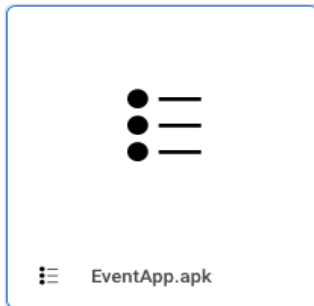


## 9. Instalación y configuración

Para realizar la instalación del aplicativo es necesario descargar el archivo APK desde el dispositivo móvil de la siguiente ruta:

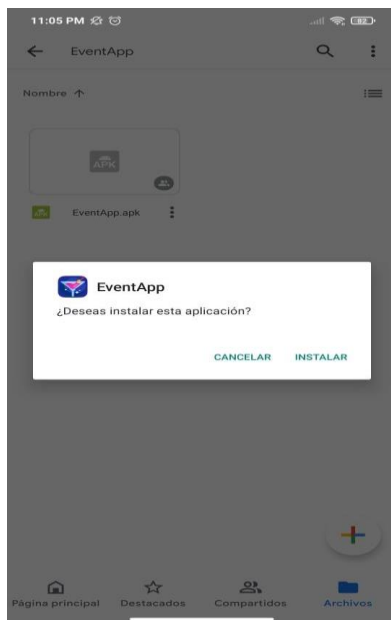
<https://drive.google.com/drive/folders/1n4rX2fRbA-c34OWuIDfbtWFIZBC4aIhQ?usp=sharing>

**Figura 53. Archivo APK**



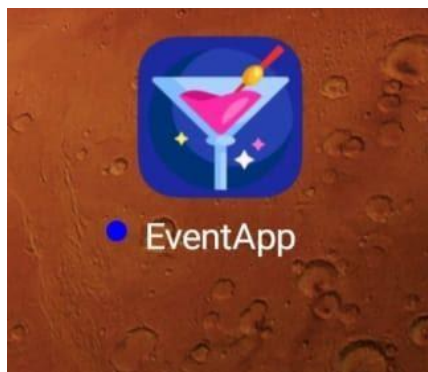
Fuente: Elaboración propia

Ejecutar el APK desde el dispositivo

**Figura 54. Ejecución del APK**

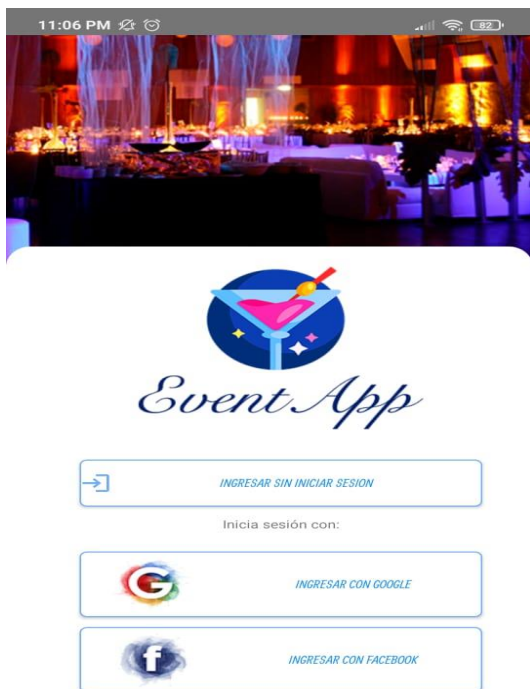
Fuente: Elaboración propia

Ejecutar desde el menú de aplicaciones EventApp

**Figura 55. Logo EventApp**

Fuente: Elaboración propia

Ingresar a la pantalla de login

**Figura 56. Pantalla de login**

Fuente: Elaboración propia

## 10. Conclusiones

En el planteamiento del proyecto inicialmente se pensó en implementar una arquitectura por microservicios, pero al transcurrir el proceso y la verificación de los requerimientos necesarios se llegó a la conclusión que una arquitectura en capas se adaptaba mejor a las necesidades del producto final.

En un principio se pensó crear la aplicación en Android y iOS, al ser una aplicación nativa requirió de mucho trabajo.

Se logró una app la cual facilita el proceso de compra y cotización de salones y/o servicios relacionados con la organización de eventos.

Se implemento una arquitectura orientada a la nube con el fin de soportar la alta demanda de usuarios y garantizar una buena experiencia de usuario.

Se implemento un método de calificación con el fin de poder clasificar los servicios según la experiencia de los usuarios, y que también sirve como referente para los futuros clientes que quieran adquirir el salón o servicio.

Se implementaron pruebas con INSPECTOR LINK e INSPECTOR JAVA del IDE de Android Studio con el fin de satisfacer los escenarios de calidad de seguridad y usabilidad.

Se implemento autenticación con GOOGLE y FACEBOOK por medio de FIREBASE con el fin de salvaguardar la información del usuario y minimizar la probabilidad de usuarios falsos en el sistema.

Los empresarios involucrados en la industria de los eventos encuentran una ventaja competitiva en este aplicativo ya que les permite llegar a un rango más amplio de personas y/o empresas.

## 11. Referencias

- Abellán, E. (2020, March 5). Metodología Scrum: Qué es y cómo funciona. Retrieved September 19, 2020, from <https://www.wearemarketing.com/es/blog/metodologia-scrum-que-es-y-como-funciona.html>
- Acuña, A. (2009). Manual para la organización de eventos artísticos y actividades. Retrieved September 19, 2020, from <https://unesdoc.unesco.org/ark:/48223/pf0000227588>
- Benedet, M. (2020, February 29). Innovación de procesos: Definición y ejemplos. Retrieved September 22, 2020, from <https://retos-operaciones-logistica.eae.es/innovacion-de-procesos-definicion-y-ejemplos/>
- Gonzales, S. (2014, March 14). Ejemplos de innovaciones de producto, de proceso, de marketing. Retrieved September 22, 2020, from <https://prezi.com/zjdld4kbu-ec/ejemplos-de-innovaciones-de-producto-de-proceso-de-marketi/>
- IMPACTO ESPERADO DEL PROYECTO - Torrens Mazzei. (n.d.). Retrieved September 22, 2020, from <http://www.torrens-mazzei.com/pdf/monica/IMPACTO.pdf>
- Obando, A. (n.d.). PROYECTO SALON DE EVENTOS DOSQUEBRADAS ANA MARIA OBANDO GARCIA COD. Retrieved September 18, 2020, from <https://docplayer.es/5395807-Proyecto-salon-de-eventos-dosquebradas-ana-maria-obando-garcia-cod-42162043.html>
- Protocolo IMEP. (2019, May 21). ¿Qué es un evento y cómo se puede clasificar? Retrieved September 18, 2020, from <https://www.protocoloimep.com/articulos/que-es-un-evento-y-clasificacion/>

Santomé, I. (2019, May 27). Ejemplos de productos innovadores: Evolución del packaging.

Retrieved September 19, 2020, from <https://summa.es/blog/ejemplos-de-productos-innovadores-packaging/>

TSEventos/ eventos empresariales, sociales y publicitarios. (n.d.). Retrieved September 18, 2020,

from <http://tseventos.co>

Ucha, F. (2012, May). Definición de Salón. Retrieved September 19, 2020, from

<https://www.definicionabc.com/general/salon.php>