

**ESCUELA POLITÉCNICA NACIONAL**

**FACULTAD DE INGENIERÍA EN SISTEMAS**

**INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN**

**PERÍODO ACADÉMICO:** 2021-A

**ASIGNATURA:** Aplicaciones Móviles

**FECHA DE ENTREGA:** 07/12/2020

**NOMBRE ESTUDIANTE:** Boris Caiza

**Ensayo Sobre los Servicios de Firebase**

* ***Introducción:***

El presente ensayo tiene como objetivo, describir de manera breve y concisa todos los servicios que nos ofrece firebase para poder aplicarlo a una aplicación web o móvil, adicionalmente se incluirá un párrafo corto en dónde se explique como el servicio puede implementarse en una aplicación móvil.

* ***Desarrollo:***

Como se mencionó en la introducción se tomara una aplicación móvil de calificación de películas como lo es el caso de IMDb o rotten tomatoes, en caso de que el servicio no pueda ser aplicado a la aplicación con esta temática se optara por describir una breve aplicación móvil con otra temática.

**Autenticación:**

Nos permite autenticar a un usuario, ya sea mediante, contraseñas, número de teléfono, correo electrónico o proveedores de terceros como lo son google, facebook, github o una base de datos propia de nosotros. Por otra parte la mayoría de apps necesita identificar a los usuarios, para que la app pueda guardar sus datos y así tener una experiencia más personalizada para cada usuario, luego de la autenticación viene la fase de autorización, que este servicio también nos ayuda a gestionar, con esto doy acceso a los recursos que cada usuario tiene. También mantiene los datos de cada usuario seguros. Se puede optimizar el login, al hacerlo con el login de google. Y contiene todo lo necesario para un correcto login, como lo es recuperación de contraseña, bloqueos, etc.

En cuanto a su implementación con nuestra aplicación base, se necesita autenticar al usuario para así poder guardar información del mismo y posteriormente mediante aprendizaje de máquina poder brindar un contenido más personalizado, como puede ser películas preferenciales que el usuario prefiera, o categorías de películas que más atraen al usuario. También para posibles olvidos de contraseña. Básicamente todo lo que conforma un login normal.

**Cloud Firestore**

Nos permite almacenar los datos de nuestra aplicación en la nube, así los usuarios puede acceder desde cualquier dispositivo a la aplicación y su contenido siempre estará disponible. Por otra parte no nos preocupamos por los servidores, ya que estos puedes estar en funcionalmente todo el tiempo necesario, por lo cual el desarrollador no tiene que preocuparse por fallas de servidor. El espacio de almacenamiento siempre será ilimitado, es decir que si la aplicación empieza a tener una demanda creciente de usuarios no será un problema. También posee librerías para el correcto manejo de la aplicación sin internet, es decir que la aplicación construida no necesariamente necesita conexión a internet, los datos se actualizarán una vez que se tenga conexión nuevamente. Entonces el desarrollador no tiene que preocuparse por la infraestructura de su aplicación, por tanto puede dedicar más tiempo al desarrollo de la misma. Se puede conectar con varios servicios que ofrece firebase. Los datos almacenados en una cloud firestore, están totalmente seguros, de igual forma se pueden crear políticas de acceso a los recursos.

En cuanto a su implementación con nuestra aplicación base, se necesita guardar en una base de datos información como usuarios, películas, premios por actor y listas de favoritos, básicamente toda la información que contendrá la aplicación.

**Database Realtime**

Los datos guardados en una database realtime se actualizan el tiempo real, lo que quiere decir que mientras el usuario usa la aplicación, el aplicativo mostrara los nuevos cambios que se efectuaron en la base de datos, lo cual facilita el acceso desde diferentes dispositivos a una misma aplicación, en otras palabras tiene una notificación inmediata, si el usuario no tiene internet, los cambios se guardaran en la memoria cache y se actualizarán cuando el usuario vuelva a tener internet. El desarrollador no tiene que preocuparse por los servidores únicamente por el desarrollo. La diferencia con cloud firestore, es que no se organizan los datos de la misma forma. Los datos almacenados en una base de datos real time están totalmente seguros, de igual forma se pueden crear políticas de acceso a los recursos.

En cuanto a su implementación con nuestra aplicación base, toda la información de la aplicación se guardaría en una database realtime, pero dado que CloudFirestores es más utilizado por su rendimiento y eficacia se prefiere utilizar CloudFirestore.

**Cloud Storage**

Es similar a una base de datos, pero este es exclusivo para desarrolladores que fabrican aplicaciones que necesitan almacenar, compartir o entregar contenido generado netamente por el usuario como puede ser fotos, videos o gifts. Se pueden crear reglas específicas para compartir contendido, como todos los demás servicios de firebase estos datos se encuentran almacenados de forma segura. Se maneja de forma óptima la carga y descarga de archivos.

En cuanto a su implementación con nuestra aplicación base, no se ha considerado compartir información creada por cada usuario por lo tanto se opta por otra aplicación la cual los usuarios comparten su contenido como fotos o videos son las redes sociales como Facebook, Instagram e incluso las redes sociales de videojuegos como lo es PlayStation App.

**Hosting**

Firebase brinda un hosting seguro para aplicación web, estas pueden ser de contenido dinámica o estática. Permite el alojamiento de archivos CSS, HTML, JS y el uso de varias APIs, el contenido cargado el hosting es inmediato por lo cual el sitio web estará en todo el mundo en cuestión de segundos. Consta con certificados de seguridad que valida que el sitio sea seguro.

En cuanto a su implementación con nuestra aplicación base no se requiere ningún archivo de extensión html o css, por otro lado en caso de querer migrar nuestra aplicación móvil a una aplicación hibrida Firebase Hosting, sería una excelente opción, pero por el momento el aplicativo está siendo desarrollado para dispositivos móviles con sistema operativo android.

**Functions**

Nos da acceso a cualquier de los módulo de NodeJs para poder usarlos de manera eficiente, este servicio es pagado, pero es sumamente útil. Básicamente con functions podemos ejecutar de forma automática el código de un back end en respuestas de solicitudes de clientes. Me permite crear funcionalidades para mi aplicación de manera más sencilla y eficaz. Nos ayuda a poner toda nuestra lógica de programación en la aplicación. No necesitamos escalar ni mantener los servidores, básicamente solo pagamos por los servicios que deseamos.

En cuanto a su implementación con nuestra aplicación base, se lo puede usar al momento de que un usuario se registre por primera vez y mostrar un mensaje de Welcome. Se puede agregar más funcionalidades en cada una de las actividades, pero aún no se ha decidido por completo que funcionalidades puedan tener, por lo cual se da únicamente este ejemplo.

**Machine learning**

Dada las necesidades del mercado actuales, el aprendizaje automático es parte del mundo de las aplicaciones móviles. Firebase Machine Learning es un paquete potente pero sencillo de usar, tanto para usuarios conocedores del tema, como para novatos, para los usuarios no experimentados no es necesario conocer a profundo el tema de machine learning, para usuarios ya expertos el paquete ofrece una API que los ayuda a usar modelos personalizados de TensorFlow. Por lo cual se puede crear nueva Apis, para las necesidades de cada desarrollador. Las Apis actuales permiten el reconocimiento de imágenes, predicción de mercado, etc.

En cuanto a su implementación con nuestra aplicación base, servirá para poder entregar una experiencia más personalizada al usuario, en otras palabras si el usuario ha seleccionado películas de su gusto, pues la aplicación le recomendará películas parecidas.

**Crashlytics**

Por lo general una aplicación contiene errores que puede descontentar a sus usuarios, para los desarrolladores es difícil encontrar estos errores, por lo tanto crashlytics permite encontrar estos errores y organizarlos en forma de prioridad, para saber cuál resolver primero, de esta manera se mantiene felices a los usuarios. También nos ofrece un dashbord en tiempo real con los problemas presentes en la aplicación. Permite configurar mensajes de alerta en caso de que el error persista y se quiera mantener a los usuarios al tanto del problema.

En cuanto a su implementación con nuestra aplicación base servirá para poder encontrar los errores, cualquier error que a los usuarios les fastidie, para así poder resolverlo de la manera más rápida posible y no tender descontentos a los usuarios.

**Performance**

Performance nos permite obtener información sobre el rendimiento de nuestras aplicaciones, por ejemplo puede ser pantallas de carga de un juego, o cosas similares. Esta información es mostrada en un dashbord de manera clara y ordenada, de tal manera que nos ayude a comprender en tiempo real donde se puede mejorar el rendimiento de la aplicación y así poder ofrecer mejores escenarios a los usuarios. Inclusive me permite conocer la conexión de internet de un área específica.

En cuanto a su implementación con nuestra aplicación base, servirá para ver cómo están las medidas de rendimientos en ciertos países, para así poder tomar medidas al respecto y mejorar la experiencia de usuario, ya sea en cuestión de internet, o cosas similares. A pesar de esto, no solo nos ayudaría con la medida de rendimiento internet, si no todas las medidas que Firebase Performance nos ofrezca.

**Test Lab**

Test lab nos ofrece la posibilidad de probar aplicaciones en diferentes dispositivos, estos pueden ser Smartphone, tablets, IPads, etc. Lo cual nos ayuda a tener una vista de cómo será la experiencia de usuario, también nos ayuda al momento de actualizar un aplicativo, ya que con Test Lab podemos probar la nueva actualización y analizar la experiencia de usuario para así poder llegar a una conclusión de si la actualización es eficiente o no.

En cuanto a su implementación con nuestra aplicación base, se utilizara este servicio cuando se necesite actualizar la aplicación, la aplicación puede tener nuevos cambios en la interfaz, en las imágenes de las películas y como se muestra su crítica, entonces con Test Lab podemos probar si el nuevo cambio visual se visualiza de manera correcta en un Smartphone, tablets o navegadores de esta manera se puede concluir si es efectivo o no implementar la actualización.

**App Distribution**

Permite enviar versiones preliminares de la aplicación desarrollada, y así a través de un panel de control observar la información valiosa sobre estas versiones. Los testers pueden colocar opciones sobre la usabilidad y la experiencia que han tenido. No es necesario instalar SDK, llenar formularios ni revisar procesos para avanzar. Simplemente integrar tu app en Firebase, se obtiene automáticamente métricas de estabilidad de todas las compilaciones de la aplicación, de manera que puedes saber cuándo estás listo para ser puesta a público.

En cuanto a su implementación con nuestra aplicación base, servirá para poder probar si la aplicación ya se encuentra completa y así poder publicarla de manera oficial, y así poder ahorrarnos quejas en la caja de comentarios y el descontento de los usuarios.

**Analytics**

Provee todos los datos necesario que un desarrollador de aplicaciones necesita, provee varios reportes como de login, donde están los usuarios, cuanto tiempo pasan en ella, etc. Nos brinda la posibilidad de crear grupos de usuarios, por ejemplo clasificarlos por países, para así conocer las estadísticas de mi aplicación para un determinado grupo de usuarios. Si se requiere datos más específicos se puede realizar una consulta SQL a la base de datos directamente. Con esta herramienta podemos generar informes sobre un total de hasta 27 eventos distintos, con esto podemos definir cómo se comportan nuestros usuarios para tomar decisiones en base a ese comportamiento.

En cuanto a su implementación con nuestra aplicación base se implementará Google Analytics, básicamente para poder generar reportes con todas las medidas que nos ofrece el servicio, y en base a estos reportes poder implementar una nueva funcionalidad, o generar grupos de usuarios para implementar determinadas funcionalidades a cada uno de los grupos de usuarios. Cambiar aspectos de las interfaces que les parece molesto a los usuarios, etc.

**Predictions**

Con este servicio, podemos predecir lo que los usuarios harán con nuestra aplicación, esto implica acciones como desinstalarla, usarlas más a menudo, no usarla, etc. El servicio de predictions usa el comportamiento pasado de los usuarios y construye un modelo de inteligencia artificial para poder saber que harán en el futuro. Un ejemplo es cuando una app vende cualquier tipo de cosa, con las predicciones podemos saber que usuarios están de acuerdo en comprarlas, cuales no y cuales quieren un descuento para poder hacer la compra, con esto podemos crear una aplicación más personalizada para cada tipo de usuario.

En cuanto a su implementación con nuestra aplicación base, se implementará este servicio, para poder lanzar ofertas de clasificaciones de películas, como puede ser por director, producción, presupuesto, etc. Ya que únicamente la aplicación está pensada para clasificar películas según su categoría es decir si son del género error, comedia, acción, etc. Observar la predicción que nos brinda el servicio y de acuerdo a eso saber si implementarla o no.

**A/B testing**

Permite probar cambios en la aplicación, como puede ser cambios de interfaz, o añadir nuevos elementos, cuando los cambios son implementado A/B testing nos permite saber el impacto que han tenido estos cambios sobre las métricas comerciales más importantes. También proporciona flexibilidad al momento de realizar los cambios en la aplicación para que se ajuste a las necesidades del desarrollador. Con el reporte brindado por A/B testing podemos determinar qué cambio tiene mayor relevancia estadística.

En cuanto a su implementación con nuestra aplicación base, se puede realizar varios cambios en la app, y luego implementar el servicio A/B testing para poder saber si estos cambios van por buen camino o necesitan correcciones o simplemente descartarlos.

**Cloud Messaging**

Cloud Mesagging nos sirve para poder enviar y recibir mensajes o notificaciones en Android, iOS y la web. Con esto podemos enviar mensajes personalizados a un grupo de usuarios, los mensajes pueden contener cualquier cosa, como puede ser un nuevo cambio en la aplicación, o un mensaje al usuario que diga “hace mucho que no usas la app”.

En cuanto a su implementación con nuestra aplicación base, este servicio es de mucha ayuda, pues en todo momento se estrenan nuevas películas con su crítica respectiva, por lo cual un mensaje a la app por cada película nueva es una gran idea.

**In-App Messaging**

Este servicio nos permite interactuar con los usuarios que usen nuestra app de manera activa, con esto nosotros podemos mandar mensajes personalizados, es decir su tipo de letra el fondo, el color de la letra, etc. Los objetivos de estos mensajes es completar acciones claves en la app, como puede ser “si pasas este nivel, obtendrás el siguiente bono”.

En cuanto a su implementación con nuestra aplicación base, servirá para poder enviar mensajes personalizados a los usuarios que tengan gustos por una categoría de películas en especial, por ejemplo a un usuario le puede gustar más las películas del tipo suspenso, entonces cada vez que se estrene una película de suspenso le llegara un nuevo mensaje diciendo “Se estrenó la película xxxx con la siguiente crítica.”

**Remote Config**

Remote Config nos brinda visibilidad y control detallado sobre el comportamiento y la apariencia de una aplicación para que puedas realizar cambios simplemente actualizando su configuración desde Firebase console. Un ejemplo puede ser una aplicación de juegos, un nivel se encuentra demasiado difícil, por lo cual podemos bajar la dificultad, sin embargo hay varios usuarios que les gusta la dificultad actual, entonces podemos aplicar A/B testing para una correcta decisión.

En cuanto a su implementación con nuestra aplicación base, este servicio no se ha considerado por el momento y no se lo considera práctico encontrar una funcionalidad para su uso, el servicio es sumamente útil para aplicaciones móviles de videojuegos como pueden ser Candy crush, ajedrez, dama chinas, etc. Básicamente cualquier aplicación de videojuegos móviles que se maneje por niveles, ya que estos por lo general necesitan actualizaciones de dificultad, y este servicio es ideal para este tipo de problemas.

**Dynamic Links**

Dynamic Links permite implementar vínculos que funcionan tal y como el desarrollador desee, en otras palabras son URLs inteligentes, no importa desde donde se abra el link, si es el iOS o en adroid, en link funciona de manera correcta. Además, los Dynamic Links funcionan incluso cuando hay que instalar la app antes de ver el contenido: si un usuario abre un Dynamic Link en iOS o Android, pero no tiene la app instalada, se le puede sugerir que la instale; cuando la app se abre, accede al vínculo. Básicamente los dynamic links funcionan para cualquier tipo de usuario que se encuentre en cualquier tipo de plataforma.

En cuanto a su implementación con nuestra aplicación base, se puede implementar el servicio para poder ingresar a la aplicación desde una publicación en una página web, en el cual se ofrezca la aplicación o contenidos de oferta relacionados a la aplicación, ya que es un Dynamic Link, servirá tanto par a usuarios de android, iOS e incluso desde navegadores.

**AdMob**

AdMob permite monetizar a través de publicidad, AdMob selecciona automáticamente cual es la publicidad que más dinero da al desarrollador por lo cual escoger la publicidad no es un problema, funciona excelente si se le combina con Google Analytics, para así poder tener estadísticas de la monetización y de la publicidad.

En cuanto a su implementación con nuestra aplicación base, se la puede usar para poder monetizar, de hecho toda aplicación móvil que hace uso de firebase debería implementar este servicio para poder tener publicidad y así generar ingresos extra.

**Extensions**

Nos permite construir una aplicación móvil de manera más sencilla y rápida, ya que implementar algunas funciones desde cero es muy demoroso, ya que hay que codificar, probar y depurar código, extensions no ahorra todo este trabajo, existen extensiones para modificar el tamaño de la imagen, correos electrónicos automáticos, acortar urls, etc.

En cuanto a su implementación con nuestra aplicación base, servirá para implementar todas las funcionalidades que deseemos y que disponga extensions, para así construir la aplicación de manera más rápida, como puede ser el redimensionamiento de imágenes para mostrar fotos de actores, o películas, ya que extensión posee un gran catálogo de funcionalidad, todavía no se ha pensado a fondo como quiere que funcione la aplicación por lo cual solo se menciona el ejemplo dicho.

* ***Conclusiones***

Con los distintos servicios que nos ofrece firebase, podemos llevar todo el ciclo de desarrollo de una aplicación de manera adecuada y correcta, en la fase de desarrollo podemos usar servicios como los de autenticaciones o extensiones, que ya facilitan en gran medida la programación, y nos evita la prueba y corrección de errores, por otro lado en la fase de mantenimiento podemos usar servicios como predictions, cloud messaging, A/B testing para poder realizar actualizaciones de mejor manera, inclusive con el panel de control que nos ofrece el servicio de analytics, podemos mejorar rápidamente lo necesario y eliminar lo innecesario.

Básicamente firebase es una gran opción para el desarrollo de aplicaciones móviles o web, por lo cual se recomienda ampliamente comprender cada uno de los servicios y practicar en cómo implementar cada uno de ellos.