

## ITEL05 - SERVICIOS Y APLICACIONES PARA IOT

# DOCUMENTACIÓN

### SISTEMA PARA LA GESTIÓN DE PERSONAL Y EVENTOS EN SEMANA DE INGENIERÍA

## TELEASSOCIATION

#### INTEGRANTES:

- |                       |          |
|-----------------------|----------|
| • MIGUEL AHUMADA      | 20190050 |
| • STEFHANIE JARAMILLO | 20190674 |
| • WILLIAM LAVADO      | 20200839 |
| • LEONARDO ABANTO     | 20201696 |

SEMESTRE: 2023-2

## 1. Requerimientos del proyecto:

Flujo	Requerimientos		Detalles	Exigibles/ Deseable
ALUMNO	Registro / Inicio de Sesión	Datos	Para registrarse, el usuario deberá de colocar los siguientes datos: nombre y apellido, código, correo (PUCP), contraseña (8 caracteres, que incluya una mayúscula y número), especificar si es estudiante o egresado. Posteriormente, el delegado general validará esta información.	Exigibles
		Respuesta	Le llegará un correo al usuario donde se indique si ha sido aceptado o denegada su solicitud para registrarse. Si es denegado, el correo incluye el motivo del rechazo, caso contrario solo recibe un correo de confirmación.	Exigibles
		Recuperación de contraseña	El usuario deberá hacer click en la opción "recuperar contraseña", tras lo cual, se procederá a enviar un código de 6 dígitos al correo con el que se registró anteriormente. Deberá colocar el código en la aplicación y colocar su nueva contraseña.	Exigible
	Lista de eventos	Plataforma	El usuario tendrá acceso a una lista de eventos publicada por el delegado de actividad y podrá apoyar el evento inicialmente como barra. Cada evento contará con una pequeña descripción, foto, fecha y lugar (geolocalizado). Se podrá ver la ruta más corta entre su ubicación y el evento. El usuario será capaz de observar los eventos en los que participa en una pestaña aparte que se llame "Mis eventos".	Exigibles
		Sala de Chat	Cada evento tiene su sala de chat, cuando el usuario decide ser parte de una actividad automáticamente se le habilita la sala, estas estarán en la pestaña "Mis chats". Recibirá una notificación por cada mensaje nuevo que reciba.	Exigibles
	Fotos	Fotos	Cada usuario podrá subir las fotos que desee del evento al que ha participado ya sea como barra o participante, se aceptaran fotos en formato jpg, jpeg, png.	Exigible

	Donaciones	Donaciones	Se podrán visualizar los números de cuenta de la AITEL y su QR para donaciones. Si el alumno es egresado tendrá que donar un monto mínimo de 100 soles, pero si es estudiante no existirá esta limitante. El usuario tendrá que subir una captura de pantalla de su donación, por plin o yape (formato jpg, jpeg, png). Tras esto, el delegado verificará el pago y el alumno recibirá una notificación confirmando el acuse de recibo.	Exigibles
DELEGADO DE ACTIVIDAD	CRUD	Detalles del evento	Por cada evento, se podrá ver los detalles registrados por el delegado de actividad, es decir, la descripción, el lugar, el contexto, la fecha y la hora. Podrá crear, leer, editar y borrar cada evento de su actividad asignada, para lo cual necesitará ingresar los datos relacionados al mismo: descripción, foto, fecha y lugar. Además de ello, podrá visualizar los participantes y asignarles un rol :barra o jugador	Exigibles
		Cambio de estado	El delegado podrá ver la cantidad de apoyos y la lista de participantes al evento. Podrá cambiar su estado de barra a participante y viceversa. Si un cambio ocurre se le notificará al alumno por correo electrónico.	Exigibles
		Término de actividad	El delegado deberá indicar la finalización de la actividad y podrá subir fotos de este evento. Se restringe la inscripción de más alumnos cuando una actividad y/o evento ya está finalizado.	Exigibles
DELEGADO GENERAL	CRUD de actividades	Creación y actualización de Actividad	El delegado por crear actividades y designar delegados del mismo siempre y cuando este usuario a designar tenga la condición de alumno. Posteriormente podrá hacer la edición del mismo por si necesita cambiar de delegado de actividad.r	Exigibles
	CRUD de alumnos	CRUD de alumnos	El delegado podrá, validar, y denegar registros alumnos sobre la lista general de alumnos. Además, será el encargado de verificar las cuotas de donaciones que se realicen. Podrá colocar a un alumno en la lista negra (baneo).	Exigibles
	Estadísticas	Estadísticas	Vista en la cual se mostrarán las estadísticas del sistema en gráficos, tales como la relación porcentual de alumnos y egresados, las estadísticas de las recaudaciones por género y cantidad de apoyos por actividad por género, la cantidad de actividades con mayor número de participantes.	Exigibles

## 2. Servicios implementados

Se utilizó como plataforma principal para almacenar la lógica empresarial y manejo de datos del proyecto la plataforma de Firebase la cuál es una base de datos en tiempo real basado en el modelo NoSQL. Esta es utilizada para la implementación de servicios web y móvil como Backend as a Services (BaaS). A continuación se explicarán los servicios de backend que nos ofreció Firebase en nuestro proyecto y el servicio EmailSender implementado en la aplicación:

### 2.1 Firestore Database:

- Fue la base de datos NoSQL en la nube que almacenó la lógica de nuestro proyecto. Para ello cada dato que queremos almacenar lo guardamos en datos dentro de un documento que pertenece a una colección. Esta data se guarda parecida al formato json por lo que su uso es muy versátil al momento de buscar datos.
- Documentación:  
<https://firebase.google.com/docs/firestore?hl=es-419>

### 2.2 Storage:

- Se utilizó para el registro y almacenamiento de imágenes en la nube debido a su capacidad de poder abstraer objetos grandes y posteriormente poder ser mostrados.
- Documentación:  
<https://firebase.google.com/docs/storage?hl=es-419>

### 2.3 Realtime Database

- Es otra base de datos alojada en la nube. Esta cumple con el rol de almacenar y mostrar los mensajes presentes en un chat dentro de un evento. Debido a su comunicación bidireccional, permite notar los cambios repentinamente obteniendo una correcta visualización y sincronismo por parte del usuario al utilizar la aplicación.
- Documentación:  
<https://firebase.google.com/docs/database?hl=es-419>

### 2.4 Authentication

- Es el servicio de autenticación básico de Firebase, este almacena las credenciales de los usuarios y permite generar sesiones dentro de la aplicación, así como también el préstamo del servicio de recuperación de contraseña.
- Documentación:  
<https://firebase.google.com/docs/auth?hl=es-419>

## 2.5 Messaging

- Se empleó el servicio de mensajería en la nube de Google, Firebase Cloud Messaging (FCM), para facilitar la entrega eficiente de notificaciones instantáneas. Estas notificaciones se diseñaron para comunicar información relevante sobre eventos, registros y la transmisión de mensajes entre uno o más usuarios.
- Las ventajas destacadas de este servicio incluyen la capacidad de segmentar de manera efectiva a los usuarios, lo que permitió personalizar la entrega de notificaciones según los requerimientos establecidos. Además, se aprovechó la funcionalidad de envío grupal para distribuir mensajes a múltiples destinatarios simultáneamente, optimizando así la difusión de información respecto a los eventos realizados. La capacidad de enviar mensajes en tiempo real fue fundamental para proporcionar una experiencia de usuario más dinámica y receptiva. La integración con demás servicios de Firebase permitió una implementación más eficiente, ya que posibilita un mejor manejo de información respecto a un usuario con el uso de Firebase Authentication y Firestore Database.
- Documentación:

<https://firebase.google.com/docs/cloud-messaging?hl=es>

## 2.6 Google Analytics for Firebase

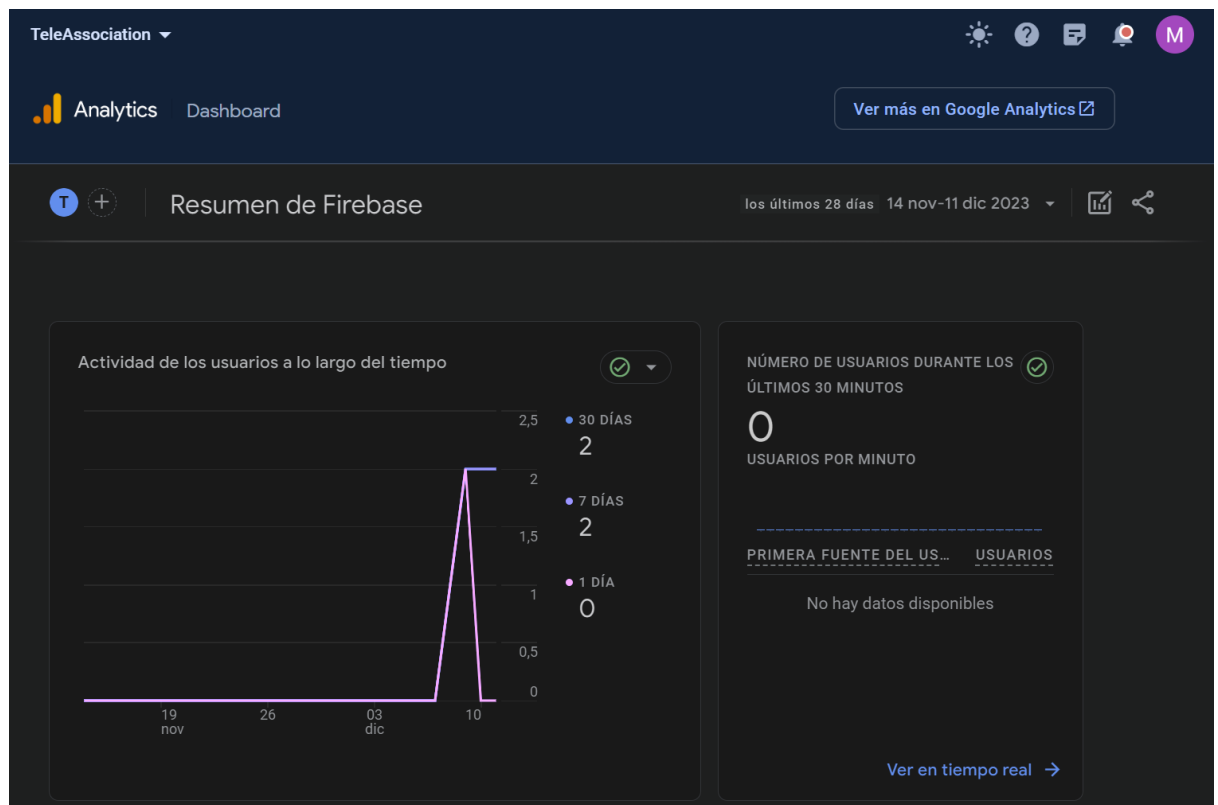
Utilizamos esta solución de Google con el propósito de obtener información valiosa sobre el rendimiento de la aplicación y datos estadísticos sobre cómo los usuarios interactúan con las notificaciones push. Es importante destacar que su implementación es beneficiosa, especialmente cuando se maneja un gran número de usuarios, tanto estudiantes como egresados, ya que permite una gestión más eficiente de los recursos disponibles.

### 1. Realtime Analytics

Brinda información inmediata de los usuarios que interactúan en la aplicación en ese momento, lo cual permite comprender la carga actual en la aplicación y ver las acciones de respuesta entre usuarios respondiendo notificaciones push en tiempo real.

### 2. Analytics Dashboard

Esta sección permite obtener una información resumida de los usuarios activos y sesiones a lo largo del tiempo. Asimismo, es importante esta obtención de información; puesto que, nos permite comprender cuántos usuarios regresan a la aplicación después de su primera interacción y el tiempo de interacción promedio en la aplicación.



## 2.7 EmailSender

Servicio de correo electrónico almacenado dentro del proyecto que nos sirve para el envío de mensajes cuando se utiliza la aplicación.

## 3. Costos Mensuales (31 días)

(Hacer costos)

Para este presupuesto, hemos estimado una cantidad de aproximadamente 2000 personas. Entonces, para estimar los costos, nos basamos en las siguientes funcionalidades que Firebase indica en su página que poseen planes de pago:

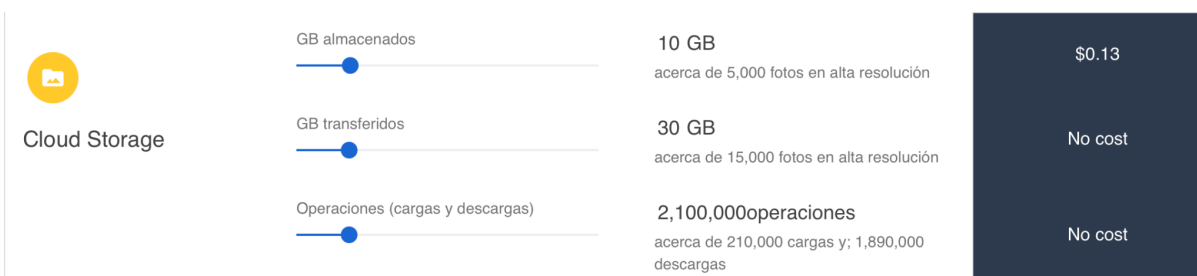
1. Firestore Database:

- ☐ Estimamos un máximo de menos de 600 000 operaciones de escritura para la cantidad de eventos y actividades de semana de ingeniería y sus escrituras, esto sería gratis.
- ☐ Estimamos un máximo de 2 000 000 operaciones de lectura para la cantidad de eventos y actividades de semana de ingeniería, esto costaría \$0.30.
- ☐ Estimamos un máximo de menos de 600 000 operaciones de borrado para la cantidad de eventos y actividades de semana de ingeniería, esto sería gratis.



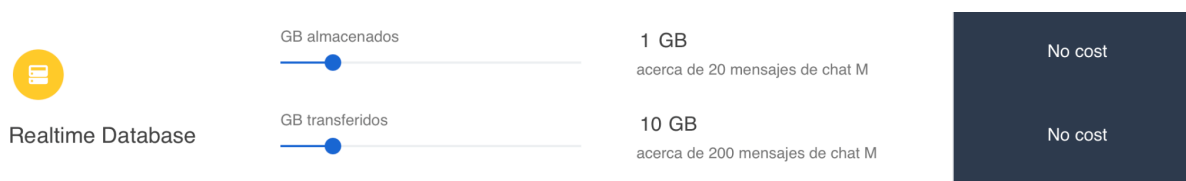
2. Storage:

- ☐ Estimamos un máximo de 5 000 fotos en alta resolución o 10GB de capacidad de almacenamiento para la cantidad de eventos y actividades de semana de ingeniería, lo cual presenta un costo de \$0.13.
- ☐ Estimamos un máximo de menos de 20 000 fotos en alta resolución o 40GB de capacidad para transferencia para la cantidad de eventos y actividades de semana de ingeniería, lo cual presenta un costo de \$1.20.
- ☐ Estimamos un máximo menor a 2 100 000 operaciones (210 000 cargas y 1 890 000 descargas) para la cantidad de eventos y actividades de semana de ingeniería, esto sería gratis.



### 3. Realtime Database:

- ☐ Estimamos un máximo de 21 400 0000 de mensajes de 50 bytes o 1GB para la cantidad de chats de los eventos de semana de ingeniería, esto sería gratis.
- ☐ Estimamos un máximo de 10GB transferidos en chats para la cantidad de chats de los eventos de semana de ingeniería, esto sería gratis.



### 4. Authentication:

- ☐ Estimamos un máximo mucho menor de 50 0000 usuarios activos para semana de ingeniería, esto sería gratis.



Luego, tanto Cloud Messaging como Google Analytics no tienen costo alguno.

Por otra parte, para la geolocalización y ruta óptima se utiliza la Api de Google Maps tal que:

- ☐ Sdk para Android: \$0 por la carga e interacción con los mapas
- ☐ Directions Api: Estimando unas 50 000 solicitudes mensuales, el costo ascendería a \$250 (cada 1000 solicitudes cuesta \$5).

Con todos estos precios mapeados, nuestro costo total mensual para mantener esta aplicación sería de unos \$251.63.



## 4. Arquitectura

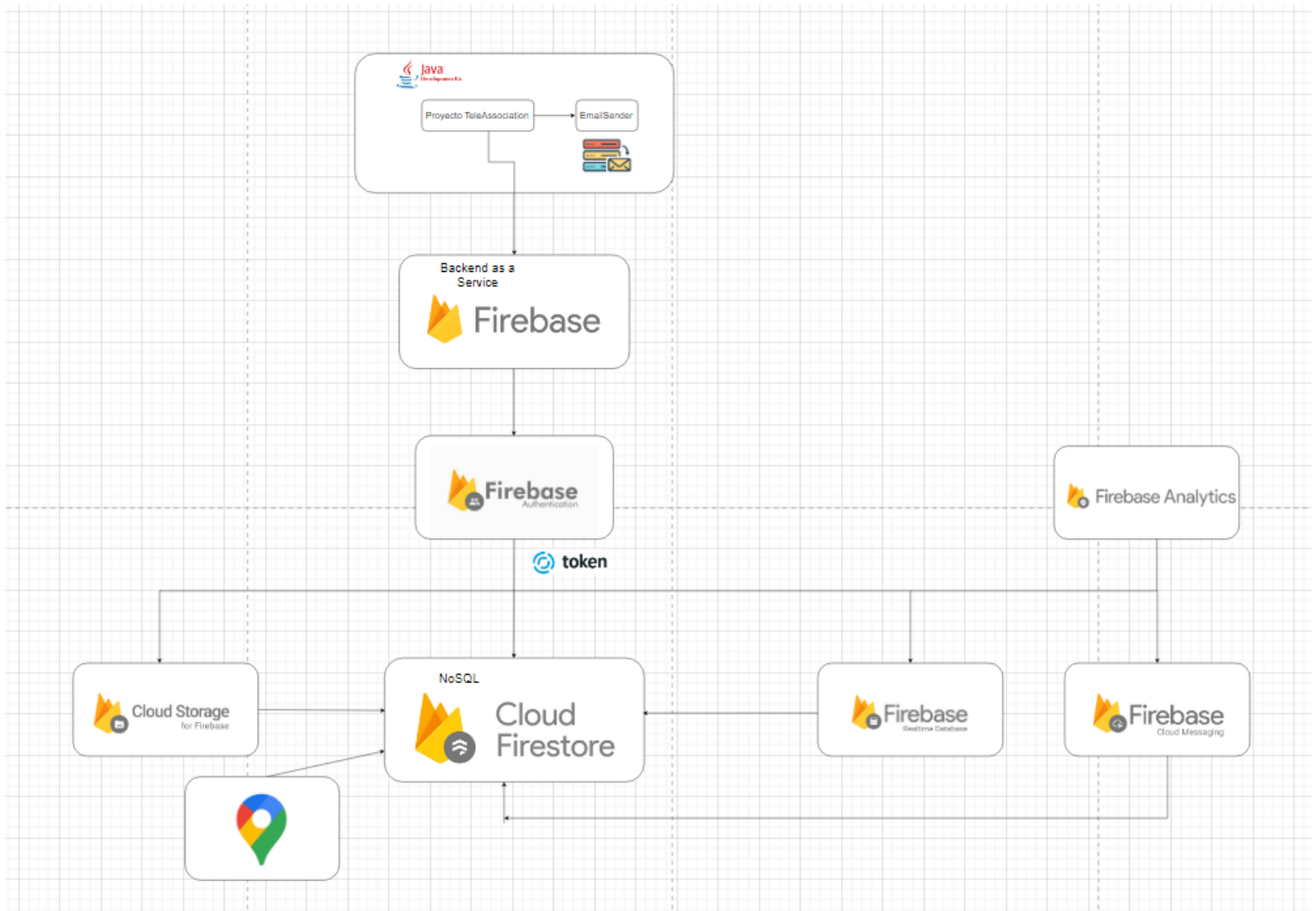


Foto: Arquitectura del proyecto. Elaboración propia.

## 5. Manual de implementación

- Requisitos:

IDE: Android Studio

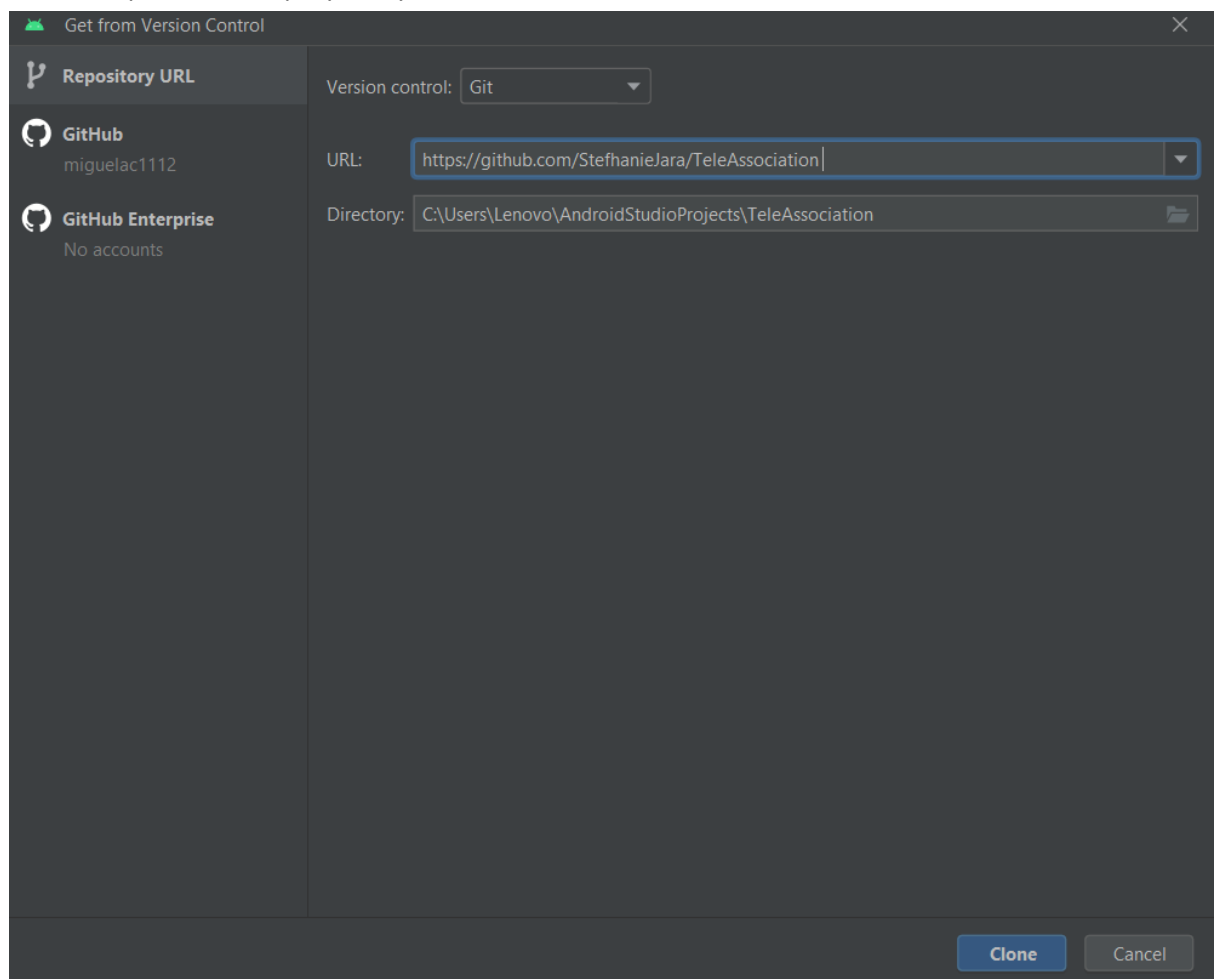
Java Development Kit (JDK)

Java Version: 17/18

Salida a Internet

- Pasos:

- 1) Para desplegar el proyecto primero necesitamos clonar el repositorio del proyecto: <https://github.com/StefhanieJara/TeleAssociation>
- 2) Abrimos nuestro entorno de desarrollo Android Studio, previamente nuestra cuenta de github vinculada, y damos a New -> Project from version control y ponemos el link de nuestro repositorio del proyecto y le damos a clonar.



- 3) Esperamos que el proyecto cargue y ya podremos utilizarlo. Para poder ejecutarlo le damos click al botón de "play" y el proyecto comenzará a cargar.
- 4) Ya cargada la aplicación, esta podrá ser utilizada con normalidad con los requerimientos establecidos.



Foto: Emulador de Android Studio

## 6. Extra

- Credenciales de usuarios existentes:
  - Rol usuario (normal):
    - Usuario: [w.lavado@pucp.edu.pe](mailto:w.lavado@pucp.edu.pe)
    - Contraseña: 12345678!
  - Rol delegado de actividad:
    - Usuario: [a20191010@pucp.edu.pe](mailto:a20191010@pucp.edu.pe)
    - Contraseña: 12345678!
  - Rol delegado general:
    - Usuario: [ahumadac.m@pucp.edu.pe](mailto:ahumadac.m@pucp.edu.pe)
    - Contraseña: 12345678!