



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

**Escuela de bolsillo: diseño e implementación de una
aplicación Android
para acercar a los familiares de alumnos de educación
infantil al centro educativo.**

TRABAJO FIN DE GRADO

Grado en Ingeniería Informática

Autor: Juan José Gonzalez Blesa

Tutor: Andrés Martínez, David de
Ruiz García, Juan Carlos

Curso 2019-2020

Resumen

En el siguiente proyecto se diseña una aplicación Android para usarse en un dispositivo móvil, que aporta a las primeras etapas educativas herramientas para mejorar el desarrollo de sus estudiantes y acercar a los padres, madres o tutores a los centros para que conozcan en un vistazo rápido como va avanzando el desarrollo de sus hijos.

La principal utilidad que ofrece la aplicación Android es un espacio capaz de almacenar cualquier tipo de recurso (información del centro, actividades y fichas de trabajo, informes pedagógicos y/o psicológicos, informes de incidencias,...), medio de comunicación entre padres y profesorado que permitirá atender a cualquier comportamiento anómalo del estudiante y poder tratarlo de la mejor manera posible, medio de realizar clases a distancia enfocado a esas edades tempranas (donde las videollamadas no son un método ágil para los más pequeños, pero si en forma de juegos específicos de cada asignatura para que los profesores puedan evaluar el progreso de sus alumnos).

El estado de alarma a provocado que muchos centros y colegios busquen una forma de continuar realizando las tareas y muchos centros han recurrido a la utilización del correo electrónico, siendo un medio poco ágil para los padres para plantear a sus hijos las tareas a realizar durante la semana. Esta aplicación también ayudara a que esta labor se realice de manera mucho más rápida porque tanto el profesor como los padres, recibirán notificaciones y avisos ante cualquier cambio de estado en la vista de sus estudiantes.

La herramienta de desarrollo utilizada es Android Studio que es el IDE oficial de Android.

Palabras clave: Android, Tablet, Docencia, Plataforma, Recursos, Informes Pedagógicos, Informes Psicológicos

Resum

En el següent projecte es dissenya una aplicació Android per usar-se en un dispositiu mòbil (telèfon mòbil o tablet), que aporta a les primeres etapes educatives eines per millorar el desenvolupament dels estudiants i apropar als pares, mares o tutors als centres per que coneguin en un cop d'ull ràpid com va avançant el desenvolupament dels seus fills.

La principal utilitat que ofereix l'aplicació Android és un espai capaç d'emmagatzemar qualsevol tipus de recurs (informació de centre, activitats i fitxes de treball, informes pedagògics i / o psicològics, informes d'incidències, ...), mitjà de comunicació entre pares i professorat que permetrà atendre qualsevol comportament anòmal de l'estudiant i poder tractar-lo de la millor manera possible, mitjà de realitzar classes a distància enfocat a aquestes edats primerenques (on les trucades de vídeo no són un mètode àgil per als més petits, però si en forma de jocs específics de cada assignatura perquè els professors puguin avaluar el progrés dels seus alumnes).

L'estat d'alarma a provocat que molts centres i col·legis busquin una forma de continuar realitzant les tasques i molts centres han recorregut a la utilització de l'correu electrònic, sent un mitjà poc àgil per als pares per plantejar als seus fills les tasques a realitzar durant la setmana. Aquesta aplicació també ajudés a que aquesta tasca es realitzi de manera molt més ràpida perquè tant el professor com els pares, rebran notificacions i avisos davant de qualsevol canvi d'estat a la vista dels seus estudiants.

L'eina de desenvolupament utilitzada és Android Studio que és l'IDE oficial d'Android.

Paraules clau: Android, Tablet, Docencia, Plataforma educativa, Informe Pedagògic, Informe Psicològic

Abstract

In the following project, an Android application is designed to be used on a mobile device (mobile phone or tablet), which provides tools to improve the development of students in the early educational stages and brings parents or guardians be closer to the centers to let them know how their children's development is progressing.

The main use offered by the Android application is a space capable of storing any type of resource (information about the center, activities and worksheets, pedagogical and / or psychological reports, incident reports, ...), means of communication between parents and teachers that will allow to attend to any anomalous behavior of the student and to be able to treat it in the best possible way, means of conducting distance classes focused on those early ages (where video calls are not an agile method for the little ones, but in the form of specific games of each subject so that teachers can evaluate the progress of their students).

The state of alarm has caused many centers and schools to look for a way to continue carrying out their tasks and many centers have resorted to the use of email, being a not very agile means for parents to propose to their children the tasks to be carried out during the week. This application will also help this work to be done much more quickly because both the teacher and the parents will receive notifications and notices before any change in status in the view of their students.

The development tool used is Android Studio which is the official Android IDE.

Key words: Android, Tablet, Teaching, Platform, Resources, Pedagogical Reports, Psychological Reports

Índice general

Índice general	V
Índice de figuras	VII
Índice de tablas	VII
<hr/>	
1 Introducción	1
1.1 Motivación	1
1.2 Objetivos	2
1.3 Estructura de la memoria	2
2 Estado del arte	3
2.1 Schooltivity	3
2.2 Educamos	4
2.3 Solucion planteada: PocketSchool	5
3 Android y las herramientas utilizadas	7
3.1 Android Studio	8
3.2 Firebase	9
3.2.1 Firebase Authentication	10
3.2.2 Firebase Cloud Firestore	10
3.2.3 Firebase Cloud Storage	10
3.2.4 Firebase Cloud Messaging	10
3.3 Github	10
4 Requisitos del sistema	11
4.1 Actores	11
4.2 Requisitos funcionales	12
4.3 Requisitos no funcionales	16
4.4 Casos de uso	16
4.5 Diagrama de clases	21
4.6 Prototipos de la aplicación	22
4.6.1 Mockup del Login	22
4.6.2 Mockup de restablecer la contraseña	22
4.6.3 Mockup de la ventana principal del administrador	23
4.6.4 Mockup de creacion de colegio	24
5 Aplicación Android: PocketShool	27
5.1 Modelo de datos	27
5.2 Interfaz gráfica	28
5.2.1 Actividad de login	28
5.2.2 Actividades principales	28
5.2.3 Actividades de configuración	29
5.2.4 Actividades de creación	30
5.2.5 Actividad de recursos, incidencias, notas y tutorias	31
5.2.6 Actividad de chat	32
5.3 Implementación	33
5.3.1 Login	33

5.3.2	Crear datos profesor	34
5.3.3	Obtener url de Firebase Storage	34
5.3.4	CardView personalizado	35
5.3.5	Envío de mensaje	35
5.3.6	Permisos del manifest	35
6	Conclusiones	37
	Bibliografía	39

Índice de figuras

2.1	Logo Schooltivity	3
2.2	Logo Educamos de SM	4
2.3	Logo PocketSchool	5
3.1	Cuota Mercado Primer Semestre 2019	7
3.2	Logo Android Studio	8
3.3	Logo Android Studio	9
3.4	Funciones de Firebase	9
3.5	Logo de Github	10
4.1	Casos de uso rol = ADMINISTRADOR	17
4.2	Casos de uso rol = PROFESOR	18
4.3	Casos de uso rol = FAMILIAR	19
4.4	Diagrama de clases	21
4.5	LOGIN mockup	22
4.6	Restaurar contraseña mockup	23
4.7	Activity principal ADMIN mockup	24
4.8	CRUD colegio mockup	25
5.1	Actividad de login	28
5.2	Actividades principales	29
5.3	Actividades de configuración	30
5.4	Actividades de configuración	30
5.5	Actividad de recursos	31
5.6	Actividad creación de incidencias y notas	32
5.7	Actividad de recursos	32
5.8	Código para loguearse	33
5.9	HashMap de profesor	34
5.10	Url descargar imagen	34
5.11	CardView de recursos	35
5.12	Método para enviar mensajes	35
5.13	Permisos Manifest	35

Índice de tablas

4.1	Actores del sistema	11
4.2	Mostrar pantalla inicial de carga	12
4.3	Acceso a datos del servidor de Firebase	12

4.4	Loguearse mediante Firebase Auth	12
4.5	Crear Colegio	12
4.6	Modificar Colegio	12
4.7	Consultar Colegio	13
4.8	Eliminar Colegio	13
4.9	Crear Aula	13
4.10	Modificar Aula	13
4.11	Consultar Aula	13
4.12	Eliminar Aula	13
4.13	Crear Area/Asignatura	14
4.14	Modificar Area/Asignatura	14
4.15	Consultar Area/Asignatura	14
4.16	Eliminar Area/Asignatura	14
4.17	Crear Usuario	14
4.18	Modificar Usuario	15
4.19	Consultar Usuario	15
4.20	Eliminar Usuario	15
4.21	Notificar de nuevos datos	15
4.22	Listar evento	15
4.24	Subir fichero	15
4.23	Notificar de nuevos datos	16
4.25	Descargar fichero	16
4.26	Crear medio de comunicación	16

CAPÍTULO 1

Introducción

Las tecnologías de la información y de la comunicación ya es un hecho que se consideran imprescindibles en la vida cotidiana de las personas en los últimos años. Estamos rodeados de diferentes dispositivos conectados entre si para facilitarnos las cosas. Sin duda, el elemento más representativo de las nuevas tecnologías es Internet, ya que nos ofrece un abanico enorme de posibilidades y de acceso a diferentes herramientas y/o utilidades, que nos ofrece servicios en multitudes de campos, desde trámites administrativos, como cursos de aprendizajes online.

La gran mayoría de centros educativos, están digitalizando sus contenidos para poder ofrecer a sus alumnos y familiares la posibilidad de tener plataformas donde poder obtener información relacionada con el centro y poder disponer de información sobre el desarrollo de sus hijos.

Trás los últimos acontecimientos sucedidos a nivel mundial a causa de la enfermedad COVID-19, todos los centros educativos han tenido que modificar su rutina de trabajo, surgiendo de este cambio unas series de necesidades (algunas incluso básicas como el de disponer de una cuenta de correo electrónico). Hay muchos centros educativos que no están o no estaban preparados para poder seguir realizando sus tareas de enseñanza con total normalidad, sobretodo para las etapas de educación infantil. Las tareas que suele realizar un profesor de infantil, es totalmente diferente si tienes que dar clases a adolescentes, donde con una videollamada puedes darles toda la información que necesitan para que continuen con sus labores. Los niños comprendidos entre 0 y 6 años que son los años que abarca la educación infantil (escuelas infantiles de primer y segundo ciclo), necesitan otro tipo de ejercicios y/o tareas para que no pierdan la concentración y se desarrollen de la mejor manera posible.

1.1 Motivación

El principal motivo de este proyecto surgió a raíz del confinamiento y se detecto que en los primeros niveles de la educación de los niños y niñas, no disponen de herramientas ni medios para seguir realizando las labores con normalidad que realizarían en una clase.

La educación infantil, es decir, aquella que abarca desde los 0 hasta los 6 años, se considera la más importante puesto que en estas edades tan tempranas es donde se asentará la base de la futura educación que recibirán. Es aquí donde adquirirán hábitos, valores y habilidades que seguirán perfeccionando en los siguientes cursos académicos.

Debido a esto, existe la posibilidad de crear una aplicación para que desde un dispositivo móvil, tanto los padres como los profesores pueden ir actualizando la información de sus alumnos, y generar un feedback entre colegio y familia. Esto es muy importante

para los más pequeños, porque podrán aprender teniendo al educador y sus padres en una misma dirección.

1.2 Objetivos

El objetivo del trabajo final de grado es desarrollar una aplicación móvil en Android para crear un feedback entre padres y profesorado de educación infantil poniendo a su disposición una herramienta útil y sencilla, capaz de realizar esta serie de funcionalidades:

- Los usuarios puedes disponer de información desde sú dispositivo móvil.
- Existiran diferentes roles donde se definirá una serie de funciones especificas y adaptadas a cada tipo de usuario.
- Se recibiran notificaciones ante cualquier novedad que aparezca en la aplicación.
- Se establecerá un medio de comunicación directa entre familiares y profesores.
- Acceso a diferentes materiales o recursos suministrados por el profesorado.

Todos estos objetivos, nos ha sido posible incorporar gracias a la plataforma digital Firebase de Google. Analizaremos sus principales funcionalidades en los próximos capitulos.

1.3 Estructura de la memoria

La memoria esta formada por X capítulos. En el primer capítulo habla de la importancia de las TIC en la vida cotidiana y de la necesidad de disponer de una herramienta que genere un feedback entre escuela-familia en las primeras etapas de la educación (educación infantil). Es muy importante incluso necesario que en los primeros años de la educación de los niños, los padres y profesores se impliquen y se cree un vinculo de comunicación para facilitarles las cosas.

El segundo capítulo analiza dos aplicaciones similares que existen en el mercado, indicando que beneficios y funcionalidades ofrecen cada una de ellas. También menciona que va a aportar nuestra aplicación a los usuarios que vayan a utilizarla, fusionando las mejores características de ambas aplicaciones.

El tercer capítulo habla sobre el sistema operativo Android, el cual se considera el SO más empleado a nivel mundial y dispone de un grán abanico de compatibilidades con multitud de dispositivos. Se habla de las herramientas utilizadas para la realización del proyecto entre las que cabe destacar la api oficial para desarrollar aplicaciones Android (Android Studio) y de la plataforma Firebase que nos ofrece Google y nos facilita las tareas de implementación.

El cuarto capítulo habla de los requisitos funcionales que hemos detectado para nuestra aplicación, enumera los casos de uso junto con su diagrama de clases correspondiente y muestra los primero mockups que se desarrollaron para la aplicación cuando todavía era una idea en fase de preparación.

PENDIENTE DE TERMINAR

CAPÍTULO 2

Estado del arte

Dentro del entorno docente podemos encontrar multitud de aplicaciones destinadas a mejorar la relación escuela - familia, tan necesaria si se pretende una educación adecuada y significativa. Después de analizar varias aplicaciones, he considerado las siguientes dos aplicaciones dignas de mencionar por sus similitudes con la aplicación desarrollada en este proyecto.

2.1 Schooltivity



Figura 2.1: Logo Schooltivity

Esta aplicación está diseñada para trabajar a modo de agenda. Es muy intuitiva y los centros de educación infantil de primer ciclo (0 a 3 años) pueden sacarle mucho partido. Como he mencionado anteriormente, puede utilizarse como agenda en la que los educadores pueden informar a las familias diariamente de como han pasado sus hijos el día (comidas, deposiciones, logros,...), algo que muchas familias aprecian ya que se les puede dar un trato muy personal en esta etapa en el que su hijo no les puede informar de lo acontecido en el día.

Por otra parte, desde el centro se pueden enviar fotos, sin la necesidad de utilizar redes sociales y evitar así que muchas familias no quieran firmar la autorización para que sus hijos no salgan en ellas, es decir, elimina el temor de las familias a que su hijo de 2 años salga en Facebook, por ejemplo.

Finalmente, esta aplicación tiene un apartado en el que aparece un chat a través del cual profesorado y padres pueden compartir impresiones, pedir material, etc. en el acto. Dicho de otra forma, crea un feedback inmediato familia - escuela.

Por mi parte, me gustaría destacar que es una app muy nueva en el mercado en la que hay funcionalidades que todavía están en fases de depuración, y también el hecho de que sea muy sencilla, provoca que sea complicada de utilizar en centros con más etapas educativas.

2.2 Educamos



Figura 2.2: Logo Educamos de SM

Es una plataforma educativa utilizada por muchos centros ya que esta diseñada por la editorial SM con la que muchos de ellos trabajan. Para las familias funciona como app sólo disponible para móvil, donde en ella pueden estar informados a través de circulares y avisos puestos por el centro. También es muy intuitiva para las familias a la par de completa, puesto que toda la información académica de sus hijos la tiene aquí plasmada.

En el otro bando, los profesores tienen una app para tablet, la cual necesita implantar muchas mejoras y un manejo más accesible. Pero también pueden acceder a través de la web, cuyo acceso y uso es más sencillo. Aquí los maestros pueden diseñar su horario, añadir tareas en el calendario, elaborar informes de asistencia, generar citas para tutorías, etc.

Esta aplicación se nota que fue diseñada para ser una aplicación web y que están migrándola para disponerla a través de un dispositivo móvil.

Un punto importante a destacar es que se le hecha en falta una comunicación inmediata (chat) que permita seguir más de cerca las necesidades de los alumnos. Lo más parecido que tiene es el acceso a una cuenta de correo de outlook tanto para familiares como profesores. A las familias la propia aplicación sí les avisa de nuevos correos mediante su notificación correspondiente, pero para el profesorado tiene que meterse explícitamente en la plataforma para consultar si hay novedades. Esto retrasa y dificulta una comunicación más fluida.

2.3 Solucion planteada: PocketSchool



Figura 2.3: Logo PocketSchool

La solución desarrollada en este proyecto se llama PocketSchool, y es una mezcla de las dos aplicaciones mencionadas arriba. Se le puede considerar una app intuitiva donde a través de diferentes botones puedes ir visualizando la información que se dispone del alumno/centro y se incorpora un chat entre los familiares y el profesorado para disponer de un canal directo de comunicación. Es una fusión de las dos aplicaciones anteriores, incluyendo el canal de comunicación, y la posibilidad de disponer de mucha información del centro.

En los siguiente apartados de la memoria explicaremos con más detalles las partes de la aplicación y cuales son sus casos de uso.

CAPÍTULO 3

Android y las herramientas utilizadas

Android es un sistema operativo inicialmente pensado para teléfonos móviles, al igual que iOS, Symbian y Blackberry OS. Lo que lo hace diferente es que está basado en Linux, un núcleo de sistema operativo libre, gratuito y multiplataforma.

El sistema permite programar aplicaciones en una variación de Java llamada Dalvik. El sistema operativo proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono (como el GPS, las llamadas, la agenda, etc.) de una forma muy sencilla en un lenguaje de programación muy conocido como es Java.

Como se puede observar en la siguiente figura 3.1, Android es el sistema operativo más utilizado a nivel mundial debido a que es compatible con muchos tipos de dispositivos móviles.

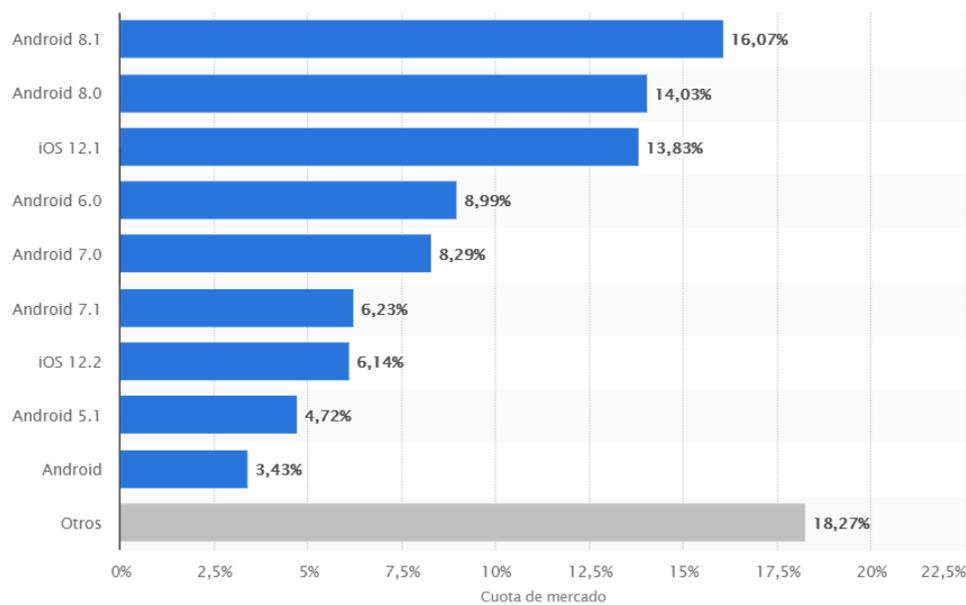


Figura 3.1: Cuota Mercado Primer Semestre 2019

He escogido este SO para desarrollar la aplicación y tener más posibilidades a la hora de distribuir, monetizar (en caso de que quiera poner mi aplicación a la venta a través de los servicios de la Play Store) y tener compatibilidad con un mayor número de dispositivos.

Para este desarrollo vamos a hablar de dos herramientas fundamentales que nos han facilitado el trabajo diario: Android Studio y Firebase.

3.1 Android Studio

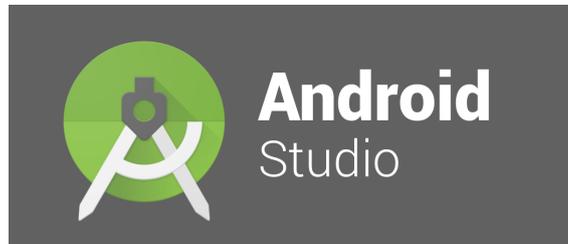


Figura 3.2: Logo Android Studio

La principal herramienta utilizada para este proyecto ha sido Android Studio, puesto que se trata del entorno de desarrollo integrado (IDE) oficial de Android. Además de ofrecer el editor de códigos y las herramientas para desarrolladores de IntelliJ, nos ofrece más funciones que aumentan la productividad a la hora de desarrollar apps. A continuación mencionamos algunas de estas funciones:

- Un sistema de compilación flexible basado en Gradle
- Un emulador rápido y cargado de funciones
- Un entorno unificado donde puedes desarrollar para todos los dispositivos Android
- Aplicación de cambios para insertar cambios de códigos y recursos a la aplicación en ejecución sin reiniciar la aplicación
- Integración con GitHub y plantillas de código para ayudarte a compilar funciones de apps comunes y también importar código de muestra
- Variedad de marcos de trabajo y herramientas de prueba
- Herramientas de Lint para identificar problemas de rendimiento, usabilidad y compatibilidad de la versión, entre otros
- Compatibilidad con C++ y NDK
- Compatibilidad integrada con Google Cloud Platform, que facilita la integración con Google Cloud Messaging y App Engine

Otra funcionalidad que nos ha facilitado el trabajo y que se integra perfectamente a Android Studio ha sido Firebase de Google.

3.2 Firebase



Figura 3.3: Logo Android Studio

Firebase es una plataforma digital que se utiliza para facilitar el desarrollo de aplicaciones web o móviles de una forma rápida, efectiva y sencilla. 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.

De todos los servicios que ofrece esta plataforma mostrados en la figura 3.4 voy a explicar las funcionalidades que hemos utilizado para nuestro proyecto.

Crea mejores aplicaciones

- Cloud Firestore**
Almacene y sincronice datos de aplicaciones a escala global
- Firestore ML** BETA
Aprendizaje automático para desarrolladores móviles
- Funciones en la nube**
Ejecute código de backend móvil sin administrar servidores
- Autenticación**
Autentica a los usuarios de forma sencilla y segura
- Hosting**
Entregue activos de aplicaciones web con velocidad y seguridad
- Almacenamiento en la nube**
Almacene y sirva archivos a escala de Google
- Base de datos en tiempo real**
Almacene y sincronice datos de aplicaciones en milisegundos

Mejorar la calidad de la aplicación

- Crashlytics**
Priorice y solucione problemas con potentes informes de fallos en tiempo real
- Supervisión del rendimiento**
Obtenga información sobre el rendimiento de su aplicación
- Laboratorio de pruebas**
Pruebe su aplicación en dispositivos alojados por Google
- Distribución de aplicaciones** BETA
Distribuya versiones preliminares de su aplicación a sus probadores de confianza

Haga crecer su negocio

- Mensajería en la aplicación** BETA
Involucre a los usuarios activos de la aplicación con mensajes contextuales
- Google analítico**
Obtenga análisis de aplicaciones gratuito e ilimitado
- Predicciones**
Segmentación inteligente de usuarios basada en el comportamiento previsto
- Pruebas A / B** BETA
Optimize la experiencia de su aplicación a través de la experimentación
- Mensajería en la nube**
Envía mensajes y notificaciones dirigidos
- Configuración remota**
Modifique su aplicación sin implementar una nueva versión
- Enlaces dinámicos**
Impulse el crecimiento mediante el uso de enlaces profundos con atribución

Figura 3.4: Funciones de Firebase

3.2.1. Firebase Authentication

Firebase Authentication busca facilitar la creación de sistemas de autenticación seguros, a la vez que mejora la experiencia de integración y acceso para los usuarios finales. Proporciona una solución de identidad de extremo a extremo, compatible con cuentas de correo electrónico y contraseñas, autenticación telefónica, acceso mediante Google, Twitter, Facebook y GitHub, y mucho más.

3.2.2. Firebase Cloud Firestore

Cloud Firestore es una base de datos de documentos NoSQL que permite almacenar, sincronizar y consultar fácilmente datos para tus apps móviles y web a escala global.

3.2.3. Firebase Cloud Storage

Cloud Storage se diseñó para ayudarte a almacenar y procesar con rapidez y facilidad el contenido generado por usuarios, como fotos y videos.

3.2.4. Firebase Cloud Messaging

Firebase Cloud Messaging (FCM) proporciona una conexión confiable y de bajo consumo de batería entre tu servidor y los dispositivos, lo que te permite enviar y recibir mensajes y notificaciones en Android, iOS y la Web sin costo.

3.3 Github

Otra herramienta que hemos utilizado a sido Github que nos ha proporcionado un lugar donde almacenar nuestro código y disponer de un control de versiones para poder saltar a cualquier punto de restauración anterior.



Figura 3.5: Logo de Github

CAPÍTULO 4

Requisitos del sistema

En el siguiente capítulo se va a realizar el proceso de Ingeniería de requisitos. Este análisis nos ayuda a estructurar el proyecto, conocer el comportamiento que realizará cada parte de esa estructura y conocer que relaciones existirán entre ellas. Se incluyen los casos de uso, requisitos funcionales, requisitos no funcionales y diagrama de clases.

4.1 Actores

Lo primero es identificar que usuarios van a utilizar la aplicación y dependiendo del rol que le asignemos podrá disponer de unos opciones u otras. Según el rol asignado, identificamos 3 actores principales que se describen en la siguiente tabla.

Código	Nombre de actor	Descripción
Actor-01	Administrador	Representa a la/s persona/s que realizarán las tareas de gestión (dar de alta un colegio con todos sus componentes, registrar profesor,familiar y alumno,...), puesta en marcha y resolución de incidencias.
Actor-02	Profesor	Representa a la/s persona/s que utilizarán la aplicación (con el rol de profesor) como su principal herramienta de gestión de sus alumnos (realizar seguimiento de sus alumnos, compartir material, informar a los padres,...).
Actor-03	Familiar	Representa a la/s persona/s que utilizarán la aplicación (con el rol de familiar) como su principal herramienta para conocer el estado en todo momento de sus hijos, consultar los informes que han sido generados por los profesionales del centro y descargar todo el material necesario que puedan necesitar.

Tabla 4.1: Actores del sistema

4.2 Requisitos funcionales

RFUN-01	Mostrar pantalla inicial de carga
Version	1.0
Descripcion	Muestra una ventana de carga mientras comprueba si debe hacer auto-login o no.
Prioridad	Baja

Tabla 4.2: Mostrar pantalla inicial de carga

RFUN-02	Acceso a datos del servidor de Firebase
Version	1.0
Descripcion	Configurar la app para que podamos interactuar con las funcionalidades de Firebase conectandolas entre sí.
Prioridad	Esencial

Tabla 4.3: Acceso a datos del servidor de Firebase

RFUN-03	Login de usuario
Version	1.0
Descripcion	Poder loguearse mediante Firebase Auth con un correo electronico y contraseña.
Prioridad	Esencial

Tabla 4.4: Loguearse mediante Firebase Auth

RFUN-04	Crear Colegio
Version	1.0
Descripcion	Dar de alta un colegio y almacenarlo en la base de datos NoSQL que nos ofrece Firebase Cloud Firestore.
Prioridad	Esencial

Tabla 4.5: Crear Colegio

RFUN-05	Modificar Colegio
Version	1.0
Descripcion	Editar los datos del colegio y actualizar la información en la base de datos.
Prioridad	Esencial

Tabla 4.6: Modificar Colegio

RFUN-06	Consultar Colegio
Version	1.0
Descripcion	Poder consultar los datos de todos los colegios que esten almacenados en nuestra base de datos.
Prioridad	Esencial

Tabla 4.7: Consultar Colegio

RFUN-07	Eliminar Colegio
Version	1.0
Descripcion	Dar de baja un colegio y eliminarlo de la base de datos.
Prioridad	Esencial

Tabla 4.8: Eliminar Colegio

RFUN-08	Crear Aula
Version	1.0
Descripcion	Dar de alta una clase y almacenarlo en la base de datos NoSQL que nos ofrece Firebase Cloud Firestore.
Prioridad	Esencial

Tabla 4.9: Crear Aula

RFUN-09	Modificar Aula
Version	1.0
Descripcion	Editar los datos de la clase y actualizar la información en la base de datos.
Prioridad	Esencial

Tabla 4.10: Modificar Aula

RFUN-10	Consultar Aula
Version	1.0
Descripcion	Poder consultar los datos de todas las clases que dispone el centro educativo y que esten almacenados en nuestra base de datos.
Prioridad	Esencial

Tabla 4.11: Consultar Aula

RFUN-11	Eliminar Aula
Version	1.0
Descripcion	Dar de baja una clase y eliminarla de la base de datos.
Prioridad	Esencial

Tabla 4.12: Eliminar Aula

RFUN-12	Crear Area/Asignatura
Version	1.0
Descripcion	Dar de alta un area/asignatura y almacenarla en la base de datos NoSQL que nos ofrece Firebase Cloud Firestore.
Prioridad	Esencial

Tabla 4.13: Crear Area/Asignatura

RFUN-13	Modificar Area/Asignatura
Version	1.0
Descripcion	Editar los datos del area/asignatura y actualizar la información en la base de datos.
Prioridad	Esencial

Tabla 4.14: Modificar Area/Asignatura

RFUN-14	Consultar Area/Asignatura
Version	1.0
Descripcion	Poder consultar los datos de todas las areas/asignaturas que se imparten en el centro educativo y que están almacenadas en nuestra base de datos.
Prioridad	Esencial

Tabla 4.15: Consultar Area/Asignatura

RFUN-15	Eliminar Area/Asignatura
Version	1.0
Descripcion	Dar de baja un area/asignatura y eliminarla de la base de datos.
Prioridad	Esencial

Tabla 4.16: Eliminar Area/Asignatura

RFUN-16	Crear Usuario
Version	1.0
Descripcion	Dar de alta un usuario dependiendo del rol que se desee y almacenarlo en su correspondiente tabla de la base de datos NoSQL que nos ofrece Firebase Cloud Firestore.
Prioridad	Esencial

Tabla 4.17: Crear Usuario

RFUN-17	Modificar Usuario
Version	1.0
Descripcion	Editar los datos del usuario y actualizar la información en la base de datos.
Prioridad	Esencial

Tabla 4.18: Modificar Usuario

RFUN-18	Consultar Usuario
Version	1.0
Descripcion	Poder consultar los datos de todos los usuarios que están almacenadas en nuestra base de datos.
Prioridad	Esencial

Tabla 4.19: Consultar Usuario

RFUN-19	Eliminar Usuario
Version	1.0
Descripcion	Dar de baja un usuario y eliminarlo de la base de datos.
Prioridad	Esencial

Tabla 4.20: Eliminar Usuario

RFUN-20	Generar evento
Version	1.0
Descripcion	Poder registrar tutoria, incidencia, autorizacion, nota en las BBDD correspondientes.
Prioridad	Esencial

Tabla 4.21: Notificar de nuevos datos

RFUN-21	Listar evento
Version	1.0
Descripcion	Visualizar los diferentes informes generados por el profesor dentro de cada activity.
Prioridad	Esencial

Tabla 4.22: Listar evento

RFUN-23	Subir fichero
Version	1.0
Descripcion	Poder almacenar documentos en Firebase Storage
Prioridad	Media

Tabla 4.24: Subir fichero

RFUN-22	Notificar de nuevos datos
Version	1.0
Descripcion	Mandar notificación a los usuarios cuando se actualiza la información de alguna de las BBDD que tambien utilicen.
Prioridad	Media

Tabla 4.23: Notificar de nuevos datos

RFUN-24	Descargar fichero
Version	1.0
Descripcion	Poder descargar documentos que han sido publicados en Firebase Storage
Prioridad	Media

Tabla 4.25: Descargar fichero

RFUN-25	Crear medio de comunicación
Version	1.0
Descripcion	Generar una sala para que el profesor y el familiar puedan intercambiar mensajes utilizando la funcionalidad de Cloud Messaging de Firebase
Prioridad	Media

Tabla 4.26: Crear medio de comunicación

4.3 Requisitos no funcionales

La aplicación dispone de un sistema de autenticación mediante Firebase Auth, que nos permite/obliga tener una conectividad a Internet para poder verificar el correo y contraseña introducidos.

También debe garantizar la persistencia de los datos, en caso de que algún dispositivo se quede sin conexión, poder seguir trabajando con normalidad y cuando se restablezca la conexión de nuevo, que continúe existiendo una coherencia con los datos que hay en la base de datos.

La aplicación debe ser escalable. De ese modo se puede ampliar con nuevas funcionalidades si detectamos alguna nueva necesidad, y se podrá hacer de manera rápida y cómoda. Otro requisito no funcional es que nuestra aplicación debe ser intuitiva a la hora de utilizarse. Se debe navegar por la aplicación sin cometer errores en cuanto a querer disponer de una información u otra.

4.4 Casos de uso

Para determinar todas las acciones que se pueden realizar en la aplicación (los casos de uso), hemos diseñado tres esquemas correspondientes a cada rol, para hacernos una idea de las tareas y relaciones que nos encontraremos.

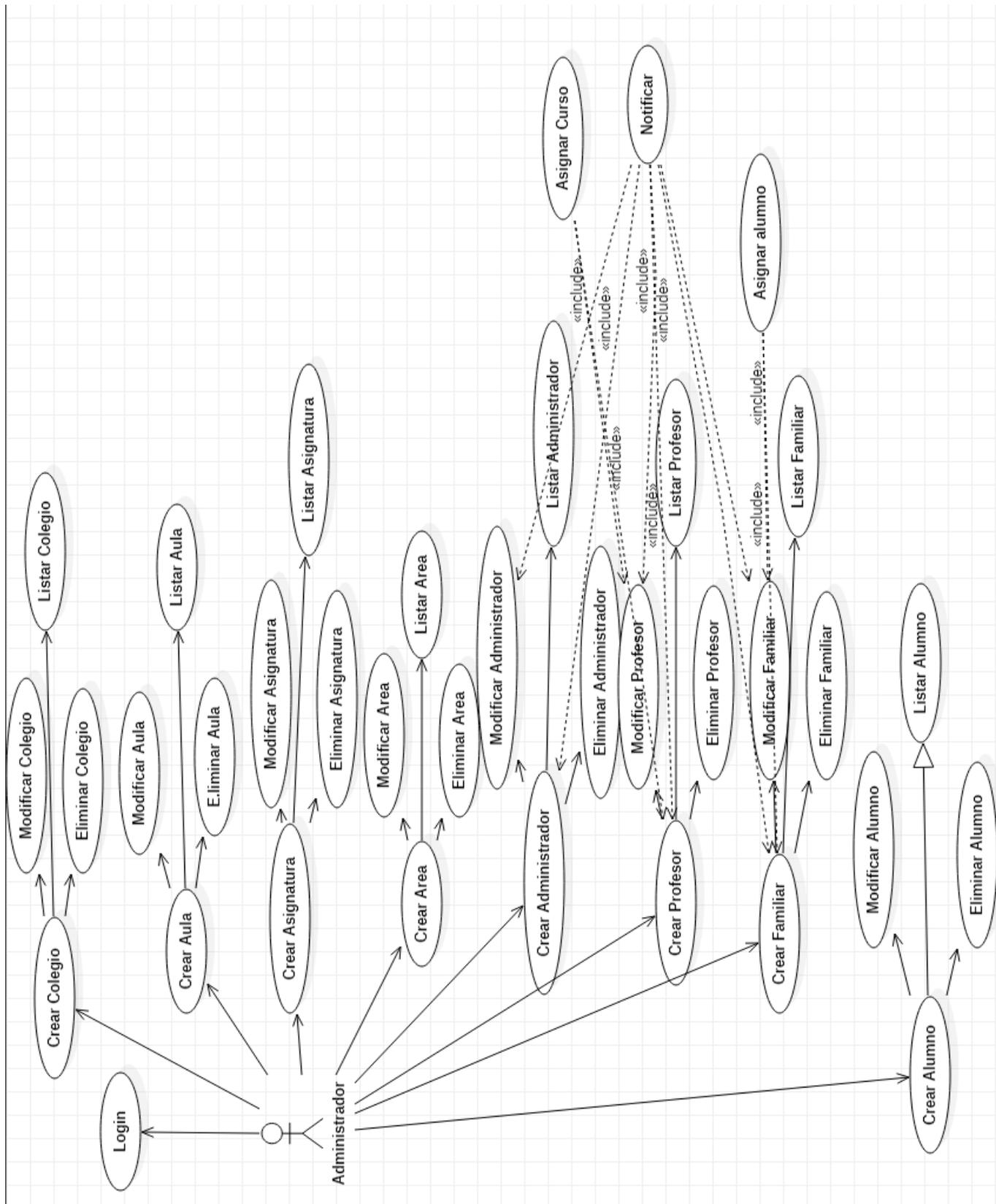


Figura 4.1: Casos de uso rol = ADMINISTRADOR

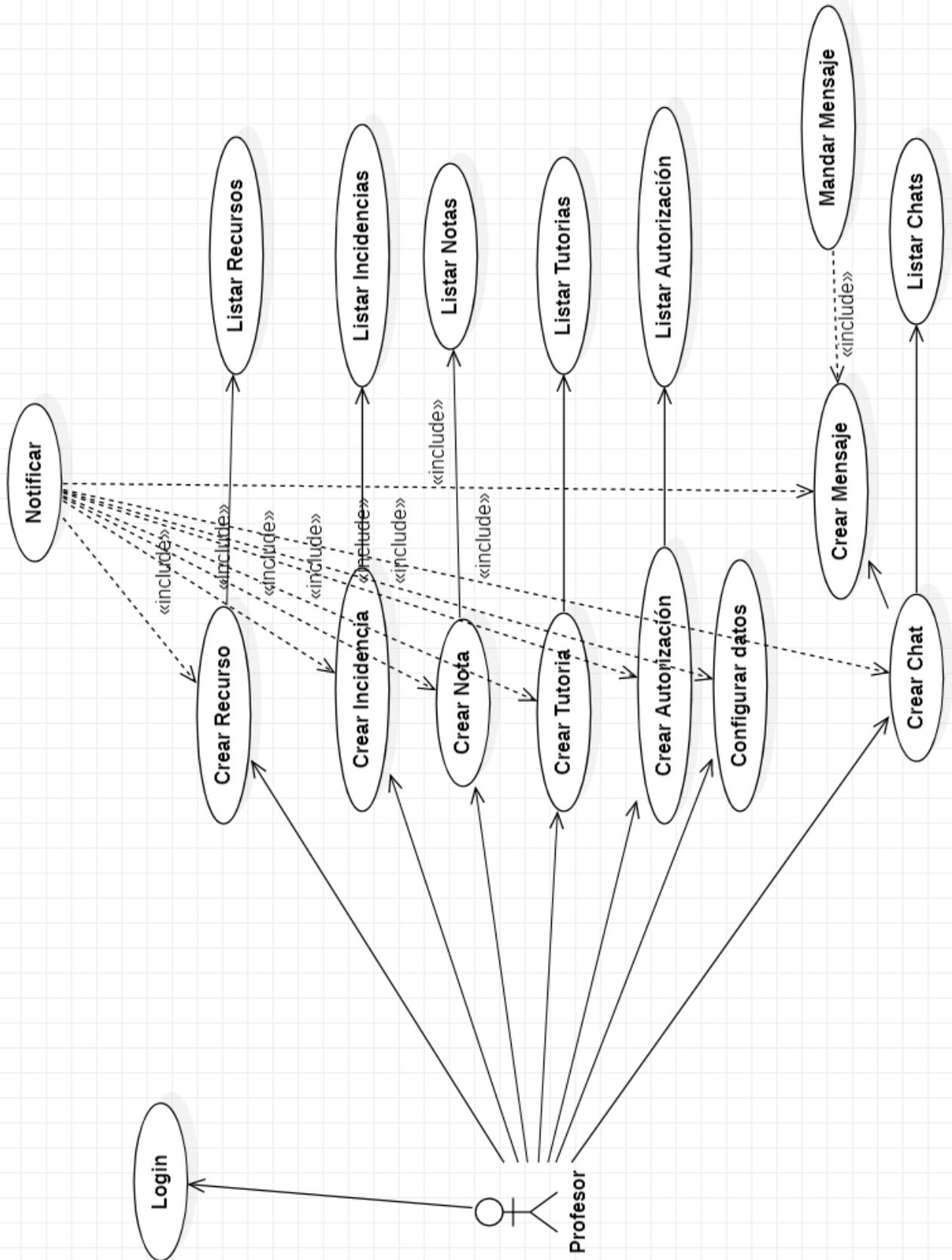


Figura 4.2: Casos de uso rol = PROFESOR

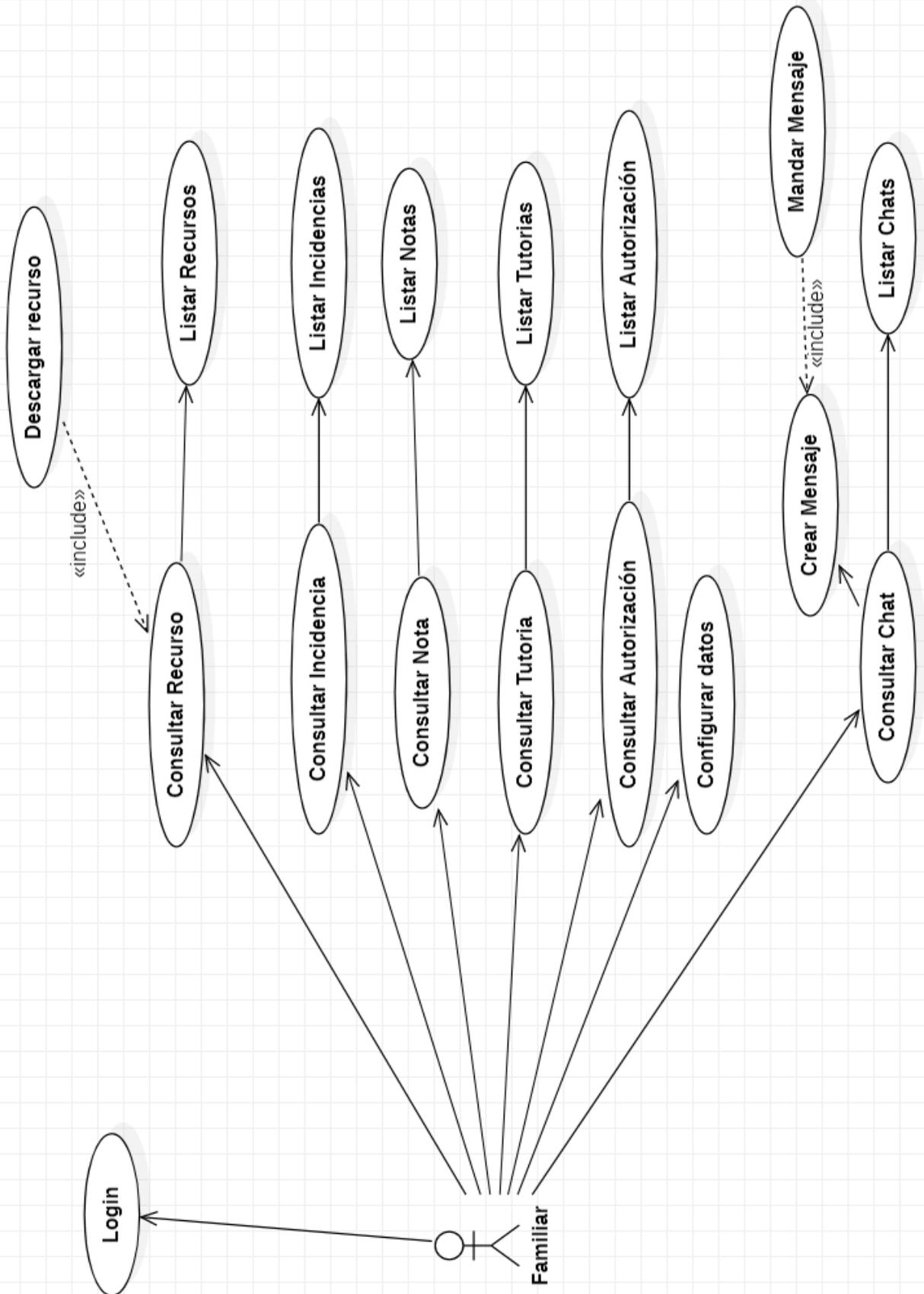


Figura 4.3: Casos de uso rol = FAMILIAR

4.5 Diagrama de clases

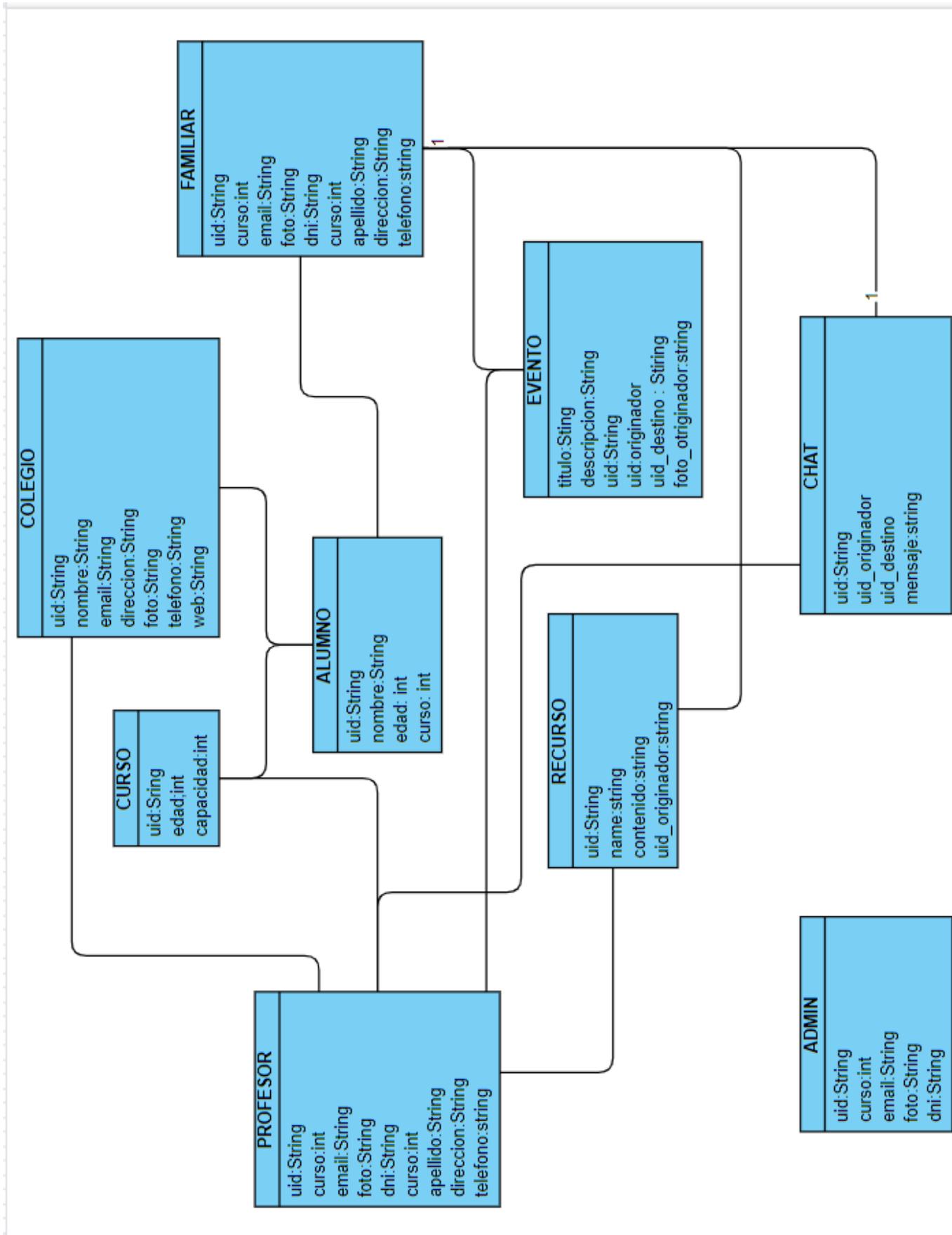


Figura 4.4: Diagrama de clases

4.6 Prototipos de la aplicación

Los primeros bocetos fueron diseñados para estudiar que elementos eran necesarios en cada activity. Aquí muestro unos cuantos bocetos iniciales que fueron tomando forma al comienzo del desarrollo del proyecto.

4.6.1. Mockup del Login

En la primera ventana de nuestra aplicación nos saldrá un menú donde deberemos poner el correo y la contraseña para poder loguearnos y acceder a la actividad principal. Se insertan dos botones, uno para validar el acceso y el otro botón para recibir al correo introducido la forma de restablecer la contraseña en caso de que no nos acordemos de ella.

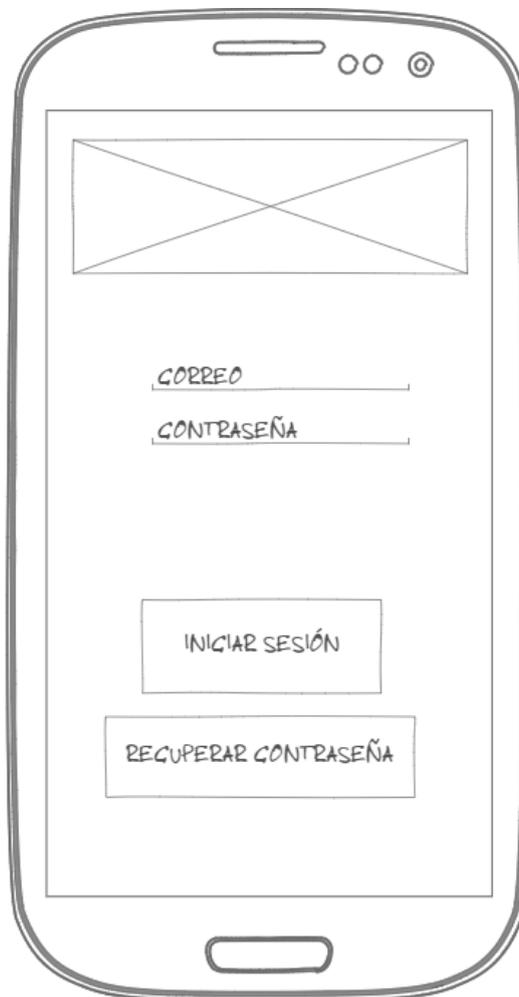


Figura 4.5: LOGIN mockup

4.6.2. Mockup de restablecer la contraseña

En esta ventana pondremos nuestra correo y nuestro dni y se verificara consultando a la base de datos que existe esa tupla en un usuario y se le notificará mediante un correo electrónico para que restablezca la contraseña.



Figura 4.6: Restaurar contraseña mockup

4.6.3. Mockup de la ventana principal del administrador

La actividad principal del administrador consiste en disponer de varios botones para acceder a los menus personalizados de creación, modificación, consulta y borrado para todos los componentes que forman parte de la aplicación.



Figura 4.7: Activity principal ADMIN mockup

4.6.4. Mockup de creacion de colegio

Esta actividad es un formulario donde nos solicita los datos necesarios para registrar un colegio en nuestra base de datos.

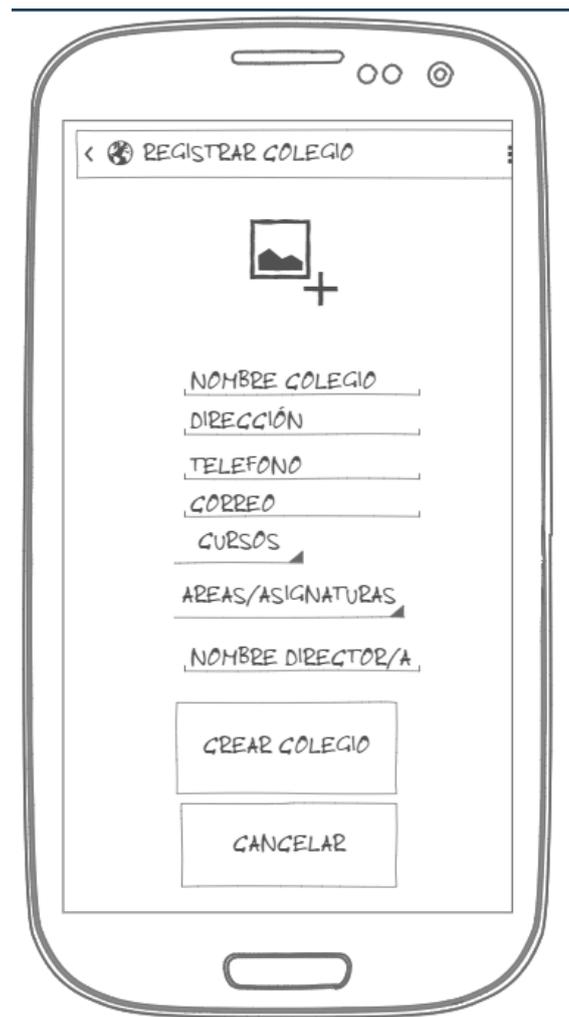


Figura 4.8: CRUD colegio mockup

CAPÍTULO 5

Aplicación Android: PocketShool

Después de analizar todos los requisitos funcionales y de realizar varias muestras de bocetos para las diferentes actividades que compondrá nuestra aplicación, vamos a realizar la definición final que se le ha otorgado.

5.1 Modelo de datos

Para deHe utilizado la funcionalidad de Firebase Firestore para almacenar los datos de la aplicación. Es una base de datos NoSQL y esta compuesta por las siguientes tablas:

- ADMIN: Tabla donde se almacenan todos los usuarios con el rol de Administrador.
- TEACHER: Tabla donde se almacenan todos los usuarios con el rol de Profesor.
- PARENT: Tabla donde se almacenan todos los usuarios con el rol de Familiar.
- STUDENT: Tabla donde se almacenan todos los usuarios con el rol de Alumno.
- SCHOOL: Tabla donde se registran todos los colegios (pensado para una futura ampliación del proyecto, poder gestionar varios centros desde la misma app).
- AULA: Tabla donde se registran las clases que habrá en un colegio (nombre, edad, capacidad).
- ASIGNATURA: Tabla donde se almacena las asignaturas impartidas en el centro.
- AREA: Tabla donde se almacena las areas de trabajo que se califican en la enseñanza infantil.
- RECURSOS: Tabla donde se almacena los datos de los recursos subidos por los profesores.
- TOKENS: Tabla donde se almacena un identificador usado para el envío de notificaciones.
- AUTORIZACIONES: Tabla donde se almacena los datos de las autorizaciones.
- INCIDENCIAS: Tabla donde se almacena los datos de las incidencias.
- NOTAS: Tabla donde se almacena los datos de las notas.
- TUTORIA: Tabla donde se almacena los datos de las tutorías.

5.2 Interfaz gráfica

5.2.1. Actividad de login

El diseño final de la actividad de login es muy similar a la que hicimos de boceto, aparecen dos cuadros de texto para introducir correo y contraseña, añadimos un splitter donde debemos seleccionar con que rol queremos acceder, se incorpora un switch para que si queremos que la aplicación nos loguee automáticamente la próxima vez que accedamos a ella y por último esta el botón de acceder que verificará la autenticación del usuario y verificará que existe en la tabla del rol que se ha seleccionado. Abajo se observa un texto que podemos pulsar sobre él y automáticamente nos mandará al mail que hemos puesto en el cuadro de correo para restablecer la contraseña.

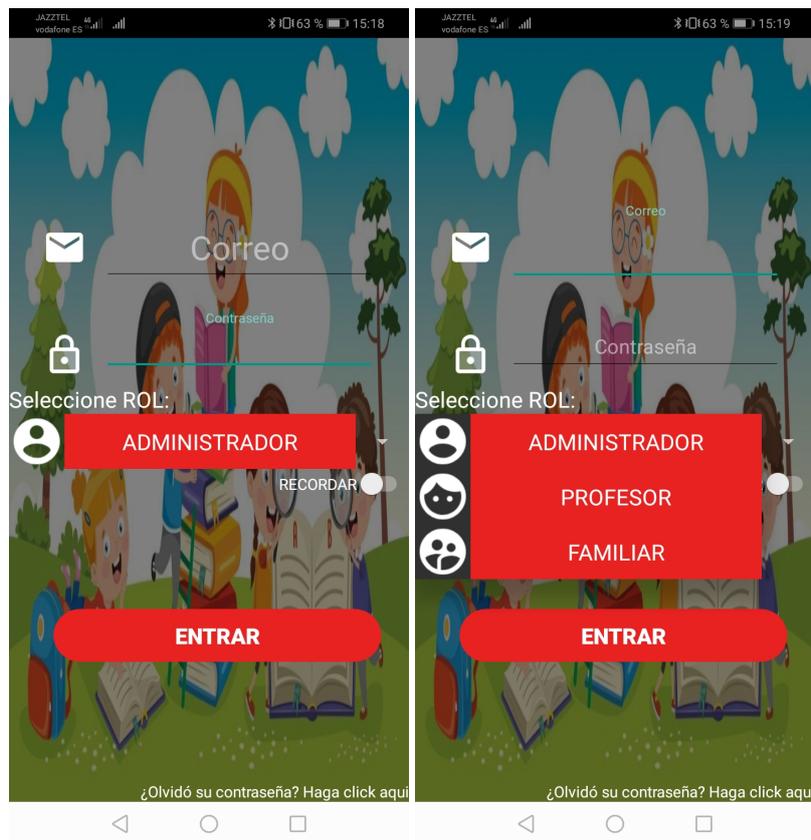


Figura 5.1: Actividad de login

5.2.2. Actividades principales

Las actividades principales de cada rol se asemejan en cuanto a diseño. Todas ellas disponen de una imagen circular que a su vez actúa de botón y nos permite entrar en el modo de configuración del usuario actual. En la esquina superior derecha hay un botón para salir de la aplicación y volver a la actividad de login. El resto de la actividad es muy intuitiva, está formada por botones con un texto en medio que indica a qué actividad se desea ir. Tengo que mencionar que tanto en la actividad principal del profesor como la de los familiares, aparece un botón para mostrar un horario, que todavía no está implementado en esta primera versión de la aplicación y no he visto necesario hacerlo por el momento. La idea de este botón es unificar el resto de eventos en esta aplicación para

poder planificarse la jornada de manera más eficiente viendo todos los datos desde un calendario.

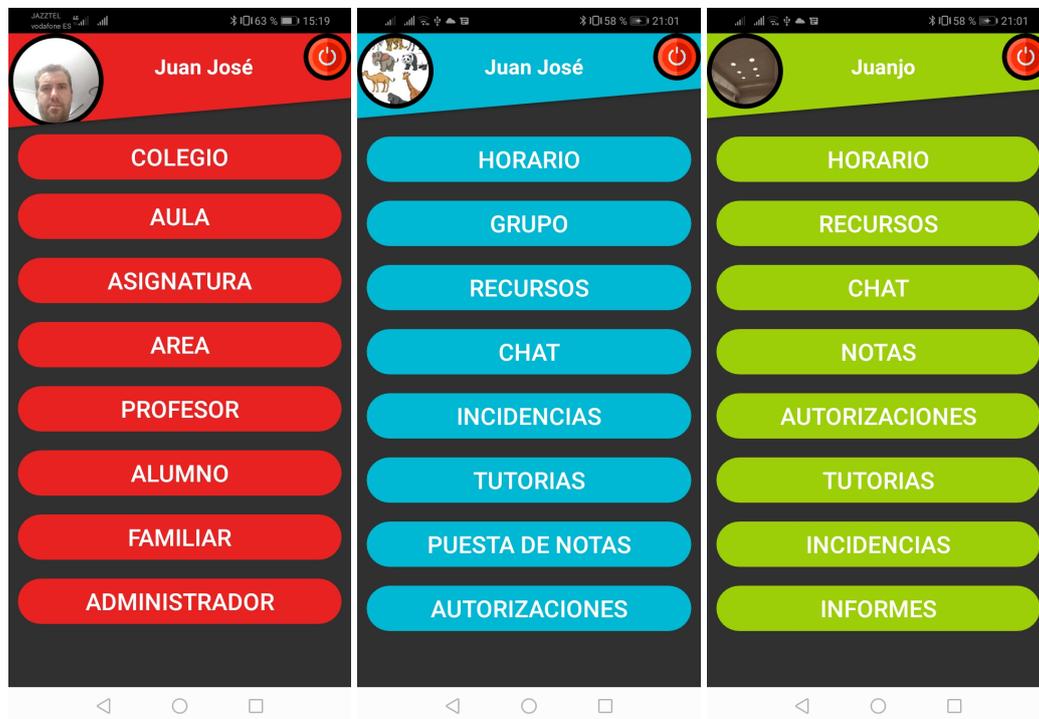


Figura 5.2: Actividades principales

5.2.3. Actividades de configuración

Los botones con la imagen circular que aparecen en la actividades principales nos llevan a otras ventanas donde aparece los datos del usuario actual, permitiendo poder modificar algún dato e incluso cambiarnos la foto de perfil.

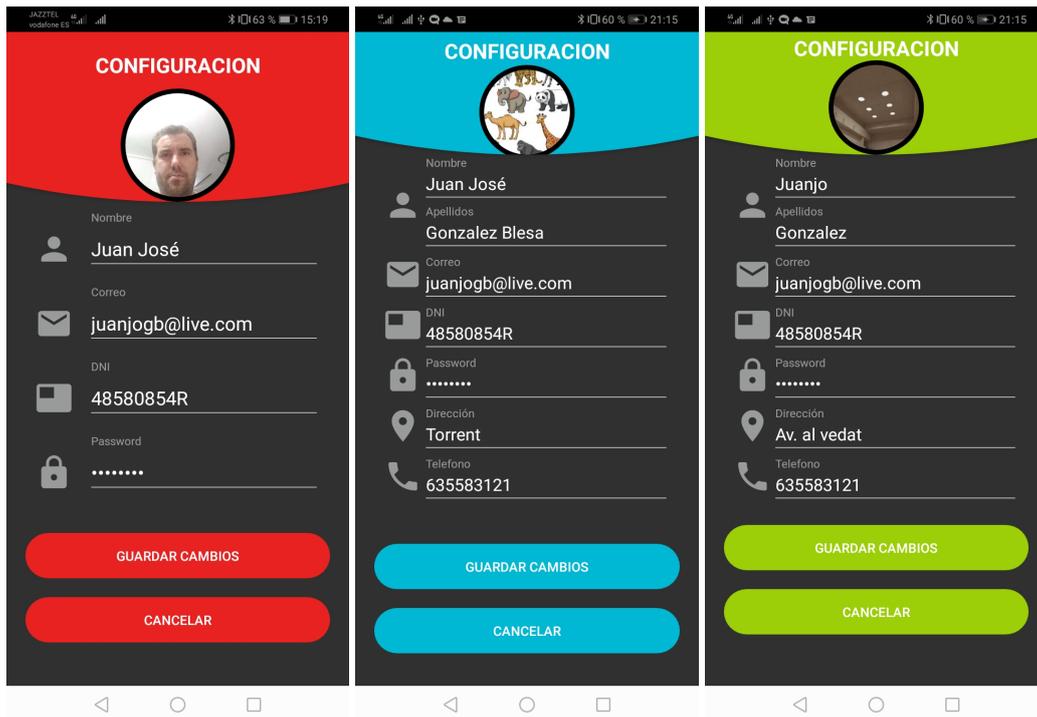


Figura 5.3: Actividades de configuración

5.2.4. Actividades de creación

El administrador se encarga de crear toda el contenido y estructura de las base de datos. Mediante los botones que dispone su actividad principal, puede ir añadiendo, consultando, modificando y/o eliminando cada objeto que se iran almacenando en cada tabla de nuestra base de datos.

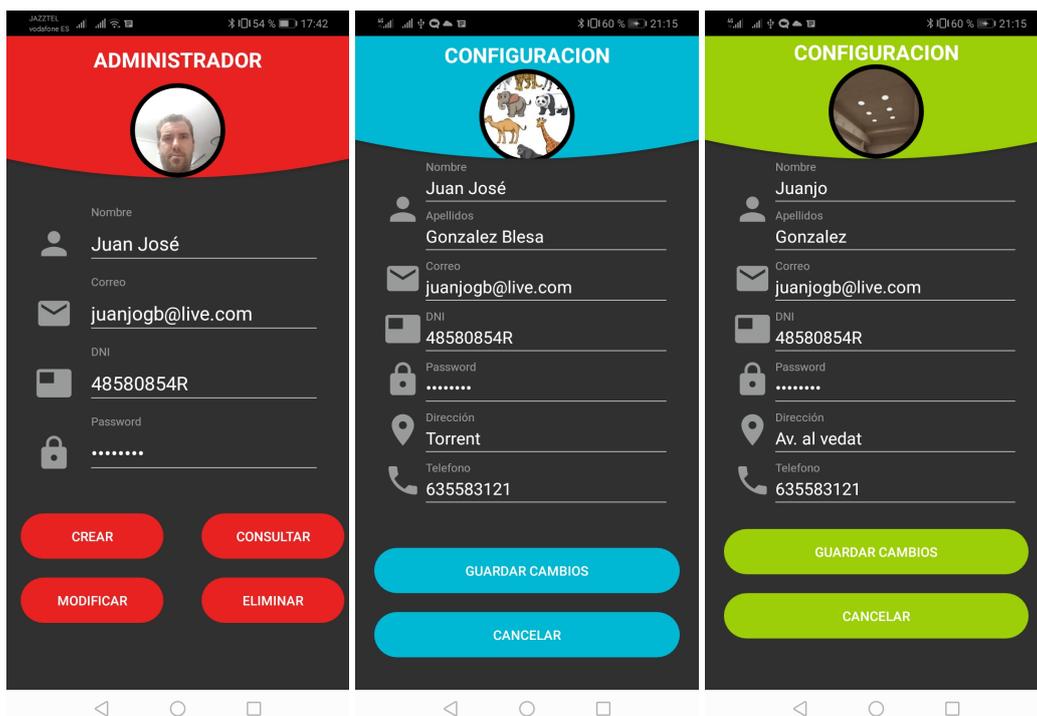


Figura 5.4: Actividades de configuración

5.2.5. Actividad de recursos, incidencias, notas y tutorías

Las actividades de recursos, incidencias, notas y tutorías funcionan de la misma forma. Se establece una actividad con un RecyclerView donde dentro mostramos mediante adapters personalizados toda la información que deseamos.

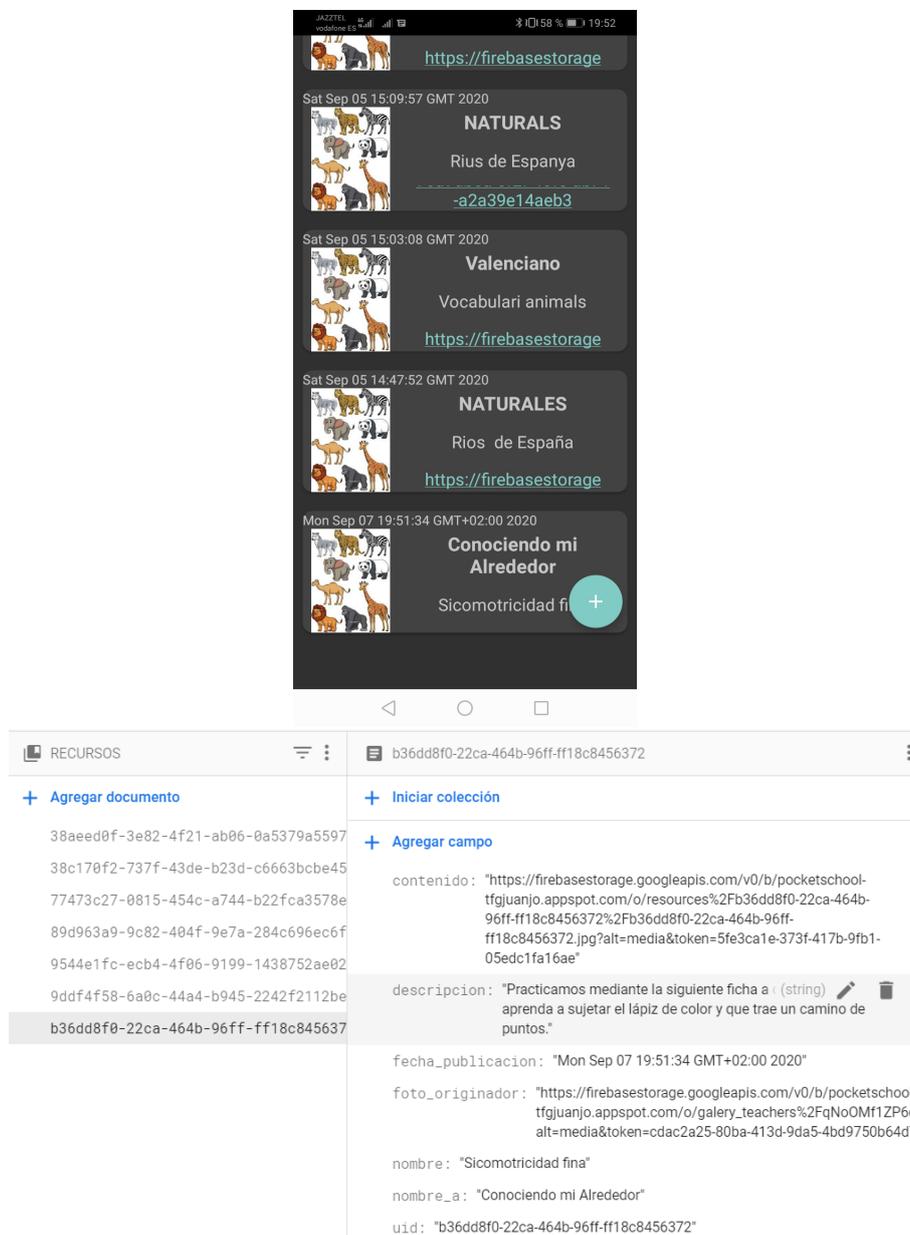


Figura 5.5: Actividad de recursos

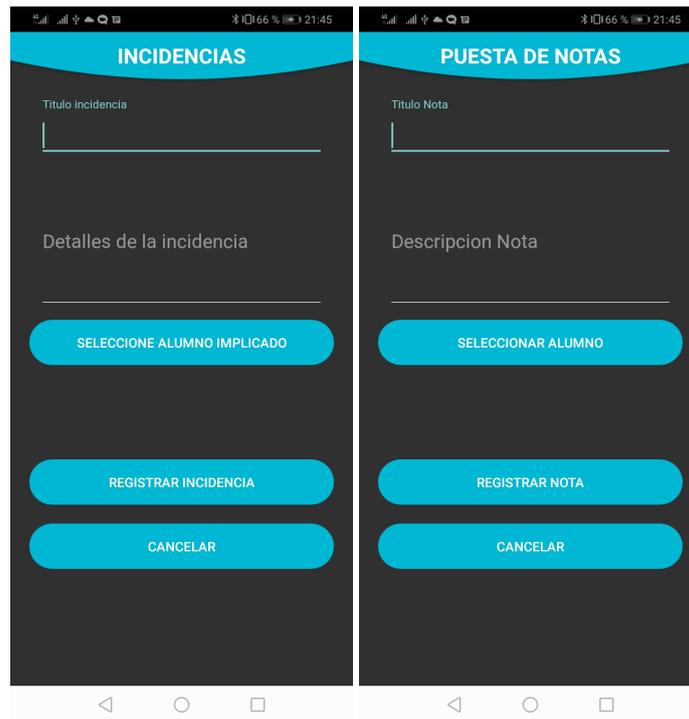


Figura 5.6: Actividad creación de incidencias y notas

5.2.6. Actividad de chat

La actividad de chat, nos ofrece la posibilidad de mandar mensajes entre varios dispositivos, almacenando los mensajes en la base de datos.

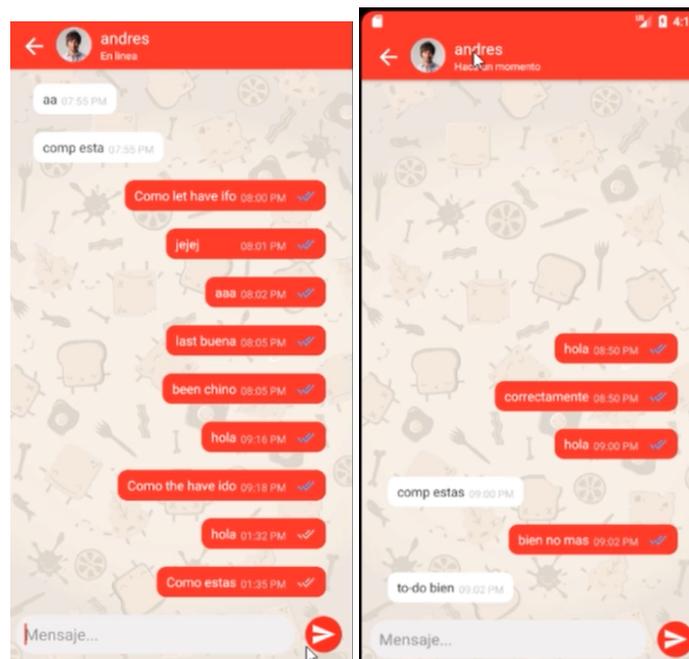


Figura 5.7: Actividad de recursos

5.3 Implementación

Durante el desarrollo de la aplicación, me he encontrado con algunas dificultades al principio para leer datos de la base de datos, mostrar una imagen almacenado en Firebase Storage, o como personalizar los adapter para que muestren los datos como deseaba. Voy a mencionar varios detalles de la implementación que he utilizado con alguna captura de ellas.

5.3.1. Login

```
private void login(final String email, final String password, final String rol) {
    if (!isValidPassword(password)) {
        showToastError( mensaje: "PASSWORD NO VALIDO. (LONGITUD MINIMA = 8)");
    } else if (!isValidEmail(email)) {
        showToastError( mensaje: "CORREO NO VALIDO");
    } else {
        mAuthProvider.login(email,password).addOnCompleteListener( activity: this, (task) → {
            alertDialog.dismiss();
            if (task.isSuccessful()) {
                if (mAuthProvider.getCurrentUser().isEmailVerified()) {
                    getIntentLogin(email,rol);
                } else {
                    showToastError( mensaje: "CORREO NO VERIFICADO");
                    editTextEmail.setError("Verifique su cuenta");
                    editTextPass.setError("Verifique su cuenta");
                }
            } else {
                showToastError( mensaje: "FALLO DE AUTENTICACIÓN");
            }
        });
    }
}
```

Figura 5.8: Código para loguearse

5.3.2. Crear datos profesor

```

//Guardar datos en bdd
Map<String, Object> profesormap = new HashMap<>();
assert profesor != null;
profesormap.put( k: "uid", profesor.getUid());
profesormap.put( k: "nombre", name);
profesormap.put( k: "apellido", apellido);
profesormap.put( k: "email", email);
profesormap.put( k: "curso", profesor.getCurso());
profesormap.put( k: "dni", dni);
profesormap.put( k: "direccion", direccion);
profesormap.put( k: "telefono", telefono);
editor.putString( s: "name",name);
editor.apply();

//mandamos correo de restablecimiento de pass al usuario

```

Figura 5.9: HashMap de profesor

5.3.3. Obtener url de Firebase Storage

```

if (mImageFile != null || mPhotoFile != null) { //La foto se va a cambiar
    Log.d( tag: "Datos", msg: "modificando usuario CAMBIANDO la foto");
    assert profesor != null;
    final StorageReference image_path = imageProvider.getStorage().child("gallery_teachers").child(profesor.getUid()).child(profesor.getUid() +
    UploadTask uploadTask;
    if (mImageFile != null) {
        byte[] imageByte = CompressorBitmapImage.getImage( ctx: ConfigActivity_teacher.this, mImageFile.getPath(), width: 200, height: 200);
        uploadTask = image_path.putBytes(imageByte); //foto de galeria
    } else {
        byte[] imageByte = CompressorBitmapImage.getImage( ctx: ConfigActivity_teacher.this, mPhotoFile.getPath(), width: 200, height: 200);
        uploadTask = image_path.putBytes(imageByte); //foto de camara
    }
    uploadTask.continueWithTask((task) -> {
        if (!task.isSuccessful()) {
            throw Objects.requireNonNull(task.getException());
        }
        return image_path.getDownloadUrl();
    }).addOnCompleteListener((task) -> {
        if (task.isSuccessful()) {
            Uri downloadUrl = task.getResult();
            assert downloadUrl != null;
            Log.i( tag: "The URL : ", downloadUrl.toString());
            //Guardar datos en bdd
            Map<String, Object> profesormap = new HashMap<>();

```

Figura 5.10: Url descargar imagen

5.3.4. CardView personalizado



Figura 5.11: CardView de recursos

5.3.5. Envío de mensaje

```
adapterMensaje = new AdapterMensaje( c: this);
LinearLayoutManager l = new LinearLayoutManager( context: this);
rvMensajes.setLayoutManager(l);
rvMensajes.setAdapter(adapterMensaje);

btnEnviar.setOnClickListener((view) -> {
    databaseReference.push().setValue(new MensajeEnviar(textMensaje.getText().toString(),
        nombrePerfil.getText().toString(), fotoPerfilCadena, type_mensaje: "1", ServerValue.TIMESTAMP));
    textMensaje.setText("");
});
```

Figura 5.12: Método para enviar mensajes

5.3.6. Permisos del manifest

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

Figura 5.13: Permisos Manifest

CAPÍTULO 6

Conclusiones

En el mundo de la educación, nunca nos podemos parar y siempre saldrán nuevas metodologías y modos de enseñanza que podemos seguir aplicando a nuestra aplicación. Además de que me ha resultado muy beneficioso trabajar con Android Studio porque me ha permitido experimentar de primera mano, todas las funcionalidades que se le pueden acoplar, la cantidad de plugins que podemos obtener para tener diseños únicos, y que ante cualquier problema que te encuentres en la implementación, siempre dispones de una enorme cantidad de recursos en Internet que te ayudan a solventar y afianzar lo aprendido. A esta aplicación todavía se le puede sacar mucho partido, como incluir notas de voz al chat, desarrollar la vista del horario en un calendario con todos los eventos centralizados, poder cambiar la configuración de los diseños desde la misma aplicación.

Por otra parte, la integración que hace con la plataforma de Firebase, hace que aún te resulte mucho más atractivo trabajar con Android. La cantidad de funciones que ofrece sin necesidad de comprar nada, es más que suficiente para tener una aplicación decente en el mercado.

Para el desarrollo de las opciones seleccionadas, me han aconsejado varios familiares maestros de escuelas infantiles y compañeros de oficio. Me han trasladado las necesidades que les gustaría que tuvieran las plataformas que usan habitualmente, y las he configurado como a ellos les hubiera gustado tener.

Por último, y como último punto de ampliación que me gustaría hacerle, es subir la aplicación al Play Store, para seguir obteniendo inputs de los usuarios y poder perfeccionar la aplicación hasta conseguir una app completa y estable.

Bibliografía

- [1] Joan Ribas Lequerica. *Desarrollo de aplicaciones para Android*. Edición 2018. ANAYA.
- [2] Documentación de Firebase. Consultado en <https://firebase.google.com/docs>.
- [3] Curso Udemy generación de Chats. Consultado en <https://www.udemy.com/course/crea-una-app-red-social-con-chat-estilo-whatsapp-con-android>.
- [4] Curso Udemy Android desde 0. Consultado en <https://www.udemy.com/course/programacion-de-android-desde-cero>.

