



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL**

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

Tema:

**APLICACIÓN FIREBASE ANDROID PARA LA GESTIÓN DE TAREAS
ACADÉMICAS A LOS ESTUDIANTES DE EDUCACIÓN BÁSICA EN LA
UNIDAD EDUCATIVA EMANUEL.**

Trabajo de Integración Curricular, Modalidad: Proyecto de Investigación, presentado
previo la obtención del título de Ingeniero en Tecnologías de la Información.

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de software

Autor: Jonathan Adrian De la Cruz Llundo

Tutor: Ing. Oscar Fernando Ibarra Torres.

Ambato - Ecuador

septiembre - 2022

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Integración Curricular con el tema: APLICACIÓN FIREBASE ANDROID PARA LA GESTIÓN DE TAREAS ACADÉMICAS A LOS ESTUDIANTES DE EDUCACIÓN BÁSICA EN LA UNIDAD EDUCATIVA EMANUEL, desarrollado bajo la modalidad Proyecto de Investigación realizado por el señor Jonathan Adrian De la Cruz Llundo estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la ejecución de la Unidad de Integración Curricular y la obtención del título de tercer nivel, de grado en la Universidad Técnica de Ambato y sus reformas y el numeral 7.4 del respectivo instructivo.

Ambato, septiembre 2022

Ing. Oscar Fernando Ibarra Torres

TUTOR

AUTORÍA

El presente trabajo de Integración Curricular titulado: APLICACIÓN FIREBASE ANDROID PARA LA GESTIÓN DE TAREAS ACADÉMICAS A LOS ESTUDIANTES DE EDUCACIÓN BÁSICA EN LA UNIDAD EDUCATIVA EMANUEL es absolutamente original, auténtico y personal. En virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, septiembre 2022



Jonathan Adrian De la Cruz Llundo

C.C. 1805275201

AUTOR

APROBACIÓN TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Integración Curricular presentado por el señor Jonathan Adrian De la Cruz Llundo, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado APLICACIÓN FIREBASE ANDROID PARA LA GESTIÓN DE TAREAS ACADÉMICAS A LOS ESTUDIANTES DE EDUCACIÓN BÁSICA EN LA UNIDAD EDUCATIVA EMANUEL, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la ejecución de la Unidad de Integración Curricular y la obtención del título de tercer nivel, de grado en la Universidad Técnica de Ambato y sus reformas y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, septiembre 2022.

Ing. Pilar Urrutia, Mg.

PRESIDENTE DEL TRIBUNAL

Ing. Sandra Lucrecia Carrillo Ríos

PROFESOR CALIFICADOR

Ing. Edison Homero Álvarez Mayorga

PROFESOR CALIFICADOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Integración Curricular como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Integración Curricular en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, septiembre 2022.

A handwritten signature in black ink, appearing to read 'Jonathan', is written over a horizontal dashed line.

Jonathan Adrian De la Cruz Llundo

C.C. 1805275201

AUTOR

DEDICATORIA

El presente proyecto se lo dedico a Dios en primer lugar por darme la sabiduría y la vida para cumplir una más de mis metas.

A mis padres Alfredo y Marisol por ser las personas más importantes en mi vida que siempre han buscado mi felicidad y la de mis hermanos aun por encima de la suya propia, por cada uno de sus consejos, por el amor que me han brindado cada uno de los días de mi vida, por su apoyo moral y económico, por sus palabras de aliento, por ser mis ejemplos a seguir, por su gran valentía con la que me han enseñado a afrontar la vida espero algún día ser tan buen padre como ustedes lo han sido para mí, no pude ser más afortunado al tenerlos como padres los mejores de este mundo, gracias por todo lo que han hecho por mí.

A mis hermanos Diego y Carolina, por ser los mejores hermanos que alguien pudiera pedir, por estar siempre a mi lado, por su compañía, gracias por siempre sacarme una sonrisa aún en los momentos más difíciles, por su ayuda incondicional y por ser las personas que siempre me ha alentado a seguir adelante avanzando hacia mis sueños y luchar hasta lograrlos, gracias porque no importa que dificultad se me presente sé que siempre podré contar con ustedes y ustedes conmigo.

A mis tíos, primos y abuelitos, por siempre darme su amor y apoyo, por sus consejos y palabras de aliento que me han impulsado a alcanzar esta meta.

AGRADECIMIENTO

Quiero agradecer a Dios por la oportunidad que me dio y haberme guiado en cada uno de mis pasos hacia esta nueva meta.

A mi tutor Ing. Fernando Ibarra por guiarme y ayudarme con su conocimiento en el desarrollo de cada una de las fases del proyecto.

A la Unidad Educativa Emanuel, profesores, alumnos y en especial a su rectora Lic. Pamela Izurieta, por darme las facilidades y acceso a su institución para realizar el presente trabajo investigativo.

A la Universidad Técnica de Ambato, profesores y compañeros por acompañarme en este proceso y darme la oportunidad de progresar y cumplir uno de mis sueños.

ÍNDICE

APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
APROBACIÓN TRIBUNAL DE GRADO	iv
DERECHOS DE AUTOR.....	v
ÍNDICE.....	viii
ÍNDICE DE FIGURAS	xiii
ÍNDICE DE TABLAS	xx
RESUMEN EJECUTIVO.....	xxiv
ABSTRACT	xxv
CAPÍTULO I.- MARCO TEÓRICO	1
1.1 Tema de investigación	1
1.1.1 Planteamiento del problema	1
1.2 Antecedentes investigativos	2
1.3 Fundamentación teórica.....	6
1.3.1 Desarrollo de software.....	6
1.3.2 Desarrollo móvil.....	7
1.3.3 Aplicaciones Android	8
1.3.4 Aplicación Android con tecnología firebase.....	9
1.3.5 Conocimiento científico.....	10

1.3.6	Proceso de aprendizaje.....	11
1.3.7	Tareas escolares	12
1.3.8	Envío de tareas a los estudiantes	13
1.3.9	Almacenamiento local	14
1.3.10	Almacenamiento en la nube.....	14
1.3.11	Firebase.....	14
1.3.12	Metodologías ágiles.....	15
1.3.13	Mobile – D	16
1.3.14	Android Studio.....	16
1.3.15	Java	17
1.3.16	BlueStacks	17
1.3.17	Balsamiq	17
1.4	Objetivos.....	18
1.4.1	Objetivo general.....	18
1.4.2	Objetivos específicos.....	18
CAPÍTULO II.- METODOLOGÍA		19
2.1	Materiales	19
2.2	Métodos.....	21
2.2.1	Modalidad de la investigación	21
2.2.2	Población y muestra	21

2.2.3	Recolección de información	23
2.2.3.1	Resultados obtenidos de las encuestas aplicadas a los estudiantes de la Unidad Educativa Emanuel sobre la gestión de tareas académicas.	23
2.2.3.2	Resultados obtenidos de las encuestas aplicadas a los profesores de la Unidad Educativa Emanuel.....	32
2.2.4	Procesamiento y análisis de datos	39
CAPÍTULO III.- RESULTADOS Y DISCUSIÓN		41
3.1	Análisis y discusión	41
3.1.1	Proceso para el envío y recibo de tareas escolares	41
3.1.2	Mejoras al proceso de envío y recibo de tareas escolares	43
3.1.3	Almacenamiento local contra almacenamiento en la nube.....	44
3.1.4	Opciones de almacenamiento en la nube	45
3.1.5	Cloud Firestore para el desarrollo de aplicaciones	46
3.1.6	Metodologías de desarrollo	47
3.1.6.1	Metodología Mobile – D	50
3.2	Desarrollo de la propuesta	51
3.2.1	Fase 1: Exploración	51
3.2.1.1	Definición de los stakeholders	51
3.2.1.2	Definición del alcance.....	52
3.2.1.3	Establecimiento del proyecto.....	53
3.2.2	Fase 2: Inicialización	54

3.2.2.1	Configuración del ambiente de desarrollo	54
3.2.2.2	Planificación de fases.....	55
3.2.2.3	Diseño de la base de datos	60
3.2.2.4	Arquitectura de la aplicación firebase Android	62
3.2.2.5	Análisis de los requisitos iniciales.....	62
3.2.2.6	Funciones establecidas	64
3.2.2.7	Diagramas de caso de uso	69
3.2.2.8	Prototipo de la aplicación firebase Android.....	71
3.2.3	Fase 3: Producción.....	82
3.2.3.1	Backend	83
3.2.3.2	Storycards.....	88
3.2.4	Fase 4: Estabilización	127
3.2.5	Fase 5: Pruebas	128
3.2.5.1	Pruebas de las funciones implementadas	128
3.2.5.2	Pruebas de aceptación.....	144
3.2.6	Implantación de la aplicación firebase Android	152
CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES.....		158
4.1	Conclusiones.....	158
4.2	Recomendaciones.....	158
BIBLIOGRAFÍA		160

ANEXOS.....	165
--------------------	------------

ÍNDICE DE FIGURAS

Figura 2.1	Pregunta 1. Encuesta a los estudiantes	23
Figura 2.2	Pregunta 2. Encuesta a los estudiantes.	24
Figura 2.3	Pregunta 3. Encuesta a los estudiantes.	25
Figura 2.4	Pregunta 4. Encuesta a los estudiantes.	26
Figura 2.5	Pregunta 5. Encuesta a los estudiantes.	27
Figura 2.6	Pregunta 6. Encuesta a los estudiantes.	28
Figura 2.7	Pregunta 7. Encuesta a los estudiantes.	29
Figura 2.8	Pregunta 8. Encuesta a los estudiantes.	30
Figura 2.9	Pregunta 9. Encuesta a los estudiantes.	31
Figura 2.10	Pregunta 1. Encuesta a los profesores.....	32
Figura 2.11	Pregunta 2. Encuesta a los profesores.....	33
Figura 2.12	Pregunta 3. Encuesta a los profesores.....	34
Figura 2.13	Pregunta 4. Encuesta a los profesores.....	35
Figura 2.14	Pregunta 5. Encuesta a los profesores.....	36
Figura 2.15	Pregunta 6. Encuesta a los profesores.....	37
Figura 2.16	Pregunta 7. Encuesta a los profesores.....	38
Figura 3.1	Proceso de envío y recibo de tareas escolares	41
Figura 3.2	Mejoras en el proceso de envío y recibo de tareas escolares.	43
Figura 3.3	Fases de Mobile – D.....	50

Figura 3.4	Modelo conceptual de la base de datos.	60
Figura 3.5	Modelo relacional de la base de datos.....	61
Figura 3.6	Arquitectura de la aplicación firebase Android.	62
Figura 3.7	Caso de uso para el administrador.	70
Figura 3.8	Caso de uso para profesores.	70
Figura 3.9	Caso de uso para estudiantes.	71
Figura 3.10	Inicio de sesión para el administrador.	72
Figura 3.11	Menú para el administrador.	72
Figura 3.12	Aceptar o rechazar profesores para el administrador.	73
Figura 3.13	Aceptar o rechazar estudiantes para el administrador.	73
Figura 3.14	Cerrar sesión para el administrador.....	74
Figura 3.15	Inicio de sesión para estudiantes y profesores.....	74
Figura 3.16	Registro de cuenta para estudiantes y profesores.	75
Figura 3.17	Restablecer contraseña para estudiantes y profesores.	75
Figura 3.18	Menú para profesores.....	76
Figura 3.19	Agregar tareas para el profesor.....	77
Figura 3.20	Ver tareas pendientes para el profesor.	77
Figura 3.21	Ver tareas entregadas para el profesor.....	78
Figura 3.22	Recordar tareas para el profesor.....	78
Figura 3.23	Ver o Editar perfil para el profesor.	79

Figura 3.24	Cerrar sesión para el profesor.	79
Figura 3.25	Menú para estudiantes.	80
Figura 3.26	Visualizar tareas pendientes de entrega para el estudiante.	81
Figura 3.27	Entregar tareas para el estudiante.	81
Figura 3.28	Ver o Editar perfil para el estudiante.	82
Figura 3.29	Cerrar sesión para el estudiante.	82
Figura 3.30	Importación de los servicios para comprobar si el usuario ya inició sesión.	87
Figura 3.31	Interfaz para comprobar si el usuario ya inició sesión.	87
Figura 3.32	Clase para comprobar si el usuario ya inició sesión.	88
Figura 3.33	Código del inicio de sesión. Administrador.	90
Figura 3.34	Código de aprobar o rechazar profesores. Administrador.	91
Figura 3.35	Código de aprobar o rechazar estudiantes. Administrador.	92
Figura 3.36	Código para cerrar sesión. Administrador.	93
Figura 3.37	Código del inicio de sesión. Profesores.	94
Figura 3.38	Código para abrir la página de inicio. Profesores.	95
Figura 3.39	Código del registro de cuenta. Profesores.	96
Figura 3.40	Código de restablecer contraseña. Profesores.	98
Figura 3.41	Código de verificar curso o materia. Profesores.	99
Figura 3.42	Código para cargar materias disponibles. Profesores.	99

Figura 3.43	Código para cargar cursos disponibles. Profesores.....	100
Figura 3.44	Código para guardar la tarea en la base de datos. Profesores.....	101
Figura 3.45	Adapter de tareas. Profesores.....	102
Figura 3.46	Código de buscar las tareas por el id. Profesores.	103
Figura 3.47	Código de cargar datos de la tarea en la interfaz gráfica. Profesores.	104
Figura 3.48	Código para ver tareas entregadas. Profesores.....	105
Figura 3.49	Código para recordar tareas. Profesores.....	106
Figura 3.50	Entidad Profesores.....	108
Figura 3.51	Código para cargar opciones de cursos y materias. Profesores....	109
Figura 3.52	Código para mostrar información del perfil. Profesores.	110
Figura 3.53	Código para actualizar información del perfil. Profesores.....	111
Figura 3.54	Código para cerrar sesión. Profesores.	112
Figura 3.55	Código para abrir página de inicio. Profesores.....	112
Figura 3.56	Código del inicio de sesión. Estudiantes.....	114
Figura 3.57	Código para abrir la página de inicio. Estudiantes.	114
Figura 3.58	Código del registro de cuenta. Estudiantes.....	115
Figura 3.59	Código de restablecer contraseña. Estudiantes.....	117
Figura 3.60	Adapter de tareas. Estudiantes.....	118
Figura 3.61	Código de buscar las tareas por el curso. Estudiantes.	118

Figura 3.62	Código para mostrar datos de la tarea en la interfaz gráfica. Estudiantes.	119
Figura 3.63	Código de selecciona el archivo de entrega. Estudiantes.....	120
Figura 3.64	Código de insertar entrega en la base de datos. Estudiantes.	121
Figura 3.65	Entidad Estudiantes.....	122
Figura 3.66	Código para cargar opciones de cursos. Estudiantes.	122
Figura 3.67	Código para mostrar información del perfil. Estudiantes.	123
Figura 3.68	Código para actualizar información del perfil. Estudiantes.....	124
Figura 3.69	Código para cerrar sesión. Estudiantes.	125
Figura 3.70	Código para abrir página de inicio. Estudiantes.....	125
Figura 3.71	Código de la clase para notificaciones. Estudiantes.....	126
Figura 3.72	Código para mostrar la notificación. Estudiantes.	127
Figura 3.73	Inicio de sesión. Administrador.....	128
Figura 3.74	Menú. Administrador.....	129
Figura 3.75	Aprobar o rechazar profesores. Administrador.	129
Figura 3.76	Aprobar o rechazar estudiantes. Administrador.....	130
Figura 3.77	Cerrar sesión. Administrador.....	130
Figura 3.78	Registro para estudiantes y profesores.	131
Figura 3.79	Restablecer contraseña para estudiantes y profesores.	132
Figura 3.80	Inicio de sesión para estudiantes y profesores.....	132

Figura 3.81	Menú. Profesores.....	133
Figura 3.82	Calendario para establecer la fecha de entrega. Profesores.	133
Figura 3.83	Agregar tarea. Profesores.	134
Figura 3.84	Lista de tareas pendientes. Profesores.	134
Figura 3.85	Eliminar tarea. Profesores.	135
Figura 3.86	Editar tarea. Profesores.....	135
Figura 3.87	Revisar entregas. Profesores.	136
Figura 3.88	Ver contenido de la entrega. Profesores.	136
Figura 3.89	Descargar archivo entregado. Profesores.	137
Figura 3.90	Recodar tarea. Profesores.	137
Figura 3.91	Correo enviado al estudiante. Profesores.	137
Figura 3.92	Información del perfil. Profesores.....	138
Figura 3.93	Actualizar información del perfil. Profesores.	139
Figura 3.94	Cerrar sesión. Profesores.	139
Figura 3.95	Menú. Estudiantes.....	140
Figura 3.96	Ver tareas pendientes. Estudiantes.	140
Figura 3.97	Entregar tarea. Estudiantes.	141
Figura 3.98	Información del perfil. Estudiantes.....	141
Figura 3.99	Actualizar información del perfil. Estudiantes.	142
Figura 3.100	Cerrar sesión. Estudiantes.	142

Figura 3.101	Notificación uno. Estudiantes.	143
Figura 3.102	Notificación dos. Estudiantes.	143
Figura 3.103	APK de instalación en la nube.	153
Figura 3.104	Descarga del archivo de instalación.	154
Figura 3.105	Instalación de la aplicación firebase Android.	154
Figura 3.106	Aplicación instalada.....	154
Figura 3.107	Estabilidad de la aplicación.....	156
Figura 3.108	Tiempo de uso promedio.	156
Figura 3.109	Interfaces más visitadas.....	157

ÍNDICE DE TABLAS

Tabla 2.1	Población de estudio	22
Tabla 2.2	Tabulación pregunta 1. Encuesta a los estudiantes.	23
Tabla 2.3	Tabulación pregunta 2. Encuesta a los estudiantes.	24
Tabla 2.4	Tabulación pregunta 3. Encuesta a los estudiantes.	25
Tabla 2.5	Tabulación pregunta 4. Encuesta a los estudiantes.	26
Tabla 2.6	Tabulación pregunta 5. Encuesta a los estudiantes.	27
Tabla 2.7	Tabulación pregunta 6. Encuesta a los estudiantes.	28
Tabla 2.8	Tabulación pregunta 7. Encuesta a los estudiantes.	29
Tabla 2.9	Tabulación pregunta 8. Encuesta a los estudiantes.	30
Tabla 2.10	Tabulación pregunta 9. Encuesta a los estudiantes.	31
Tabla 2.11	Tabulación pregunta 1. Encuesta a los profesores.	32
Tabla 2.12	Tabulación pregunta 2. Encuesta a los profesores.	33
Tabla 2.13	Tabulación pregunta 3. Encuesta a los profesores.	34
Tabla 2.14	Tabulación pregunta 4. Encuesta a los profesores.	35
Tabla 2.15	Tabulación pregunta 5. Encuesta a los profesores.	36
Tabla 2.16	Tabulación pregunta 6. Encuesta a los profesores.	37
Tabla 2.17	Tabulación pregunta 7. Encuesta a los profesores.	38
Tabla 3.1	Comparación entre almacenamiento local y en la nube	45
Tabla 3.2	Comparación entre OneDrive, Dropbox y Cloud Firestore.	46

Tabla 3.3	Características de Cloud Firestore.	47
Tabla 3.4	Comparación de las metodologías de desarrollo.	49
Tabla 3.5	Planificación de las fases de Mobile – D.	60
Tabla 3.6	Funciones para el administrador.	65
Tabla 3.7	Funciones para los profesores.	67
Tabla 3.8	Funciones para los estudiantes.	69
Tabla 3.9	Storycard del inicio de sesión para el administrador.	89
Tabla 3.10	Storycard de aprobar o rechazar profesores para el administrador.	90
Tabla 3.11	Storycard de aprobar o rechazar estudiantes para el administrador.	91
Tabla 3.12	Storycard de cerrar sesión para el administrador.	93
Tabla 3.13	Storycard del inicio de sesión para profesores.	94
Tabla 3.14	Storycard del registro de cuenta para profesores.	96
Tabla 3.15	Storycard de restablecer contraseña para profesores.	97
Tabla 3.16	Storycard de restablecer contraseña para profesores.	98
Tabla 3.17	Storycard de ver tareas pendientes para profesores.	102
Tabla 3.18	Storycard de revisar entregas para profesores.	104
Tabla 3.19	Storycard de recordar tareas para profesores.	106
Tabla 3.20	Storycard de ver perfil para profesores.	107
Tabla 3.21	Storycard de actualizar perfil para profesores.	110

Tabla 3.22	Storycard de cerrar sesión para profesores.	112
Tabla 3.23	Storycard del inicio de sesión para estudiantes.	113
Tabla 3.24	Storycard del registro de cuenta para estudiantes.	115
Tabla 3.25	Storycard de restablecer contraseña para estudiantes.	116
Tabla 3.26	Storycard de ver tareas pendientes para estudiantes.	117
Tabla 3.27	Entregar tareas para estudiantes.	119
Tabla 3.28	Storycard de ver perfil para estudiantes.	121
Tabla 3.29	Storycard de actualizar perfil para estudiantes.	123
Tabla 3.30	Storycard de cerrar sesión para estudiantes.	124
Tabla 3.31	Storycard de notificaciones para estudiantes.	126
Tabla 3.32	Prueba de aceptación 1.	144
Tabla 3.33	Prueba de aceptación 2.	145
Tabla 3.34	Prueba de aceptación 3.	145
Tabla 3.35	Prueba de aceptación 4.	145
Tabla 3.36	Prueba de aceptación 5.	146
Tabla 3.37	Prueba de aceptación 6.	146
Tabla 3.38	Prueba de aceptación 7.	147
Tabla 3.39	Prueba de aceptación 8.	147
Tabla 3.40	Prueba de aceptación 9.	148
Tabla 3.41	Prueba de aceptación 10.	148

Tabla 3.42	Prueba de aceptación 11.....	149
Tabla 3.43	Prueba de aceptación 12.....	149
Tabla 3.44	Prueba de aceptación 13.....	150
Tabla 3.45	Prueba de aceptación 14.....	150
Tabla 3.46	Prueba de aceptación 15.....	151
Tabla 3.47	Prueba de aceptación 16.....	151
Tabla 3.48	Prueba de aceptación 17.....	152
Tabla 3.49	Prueba de aceptación 18.....	152
Tabla 3.50	Cronograma de capacitación.....	155

RESUMEN EJECUTIVO

El uso de la tecnología ayuda a garantizar el cumplimiento de tareas teniendo un impacto positivo en la formación académica estudiantil, ya que dentro del ámbito educativo las tareas escolares cumplen un rol importante, debido a que refuerzan los conocimientos adquiridos por los estudiantes en el aula de clases logrando así un exitoso proceso de enseñanza – aprendizaje.

El presente proyecto tiene como objetivo asistir a los estudiantes y profesores de la Unidad Educativa Emanuel en los procesos referentes a las tareas académicas brindando una herramienta tecnológica para este fin, logrando así reducir el índice de incumplimiento y ayudando a la gestión de nuevos conocimientos.

La aplicación firebase Android se usa dentro del proceso de envío y recibo de tareas, guardando la información de la tarea, sus detalles y sus respectivas instrucciones en la nube para que los estudiantes tenga acceso a ellas en todo momento, además se envían notificaciones recordando realizar las tareas y revisarlas para su respectiva entrega, logrando que el índice de cumplimiento sea mayor.

La aplicación fue desarrollada usando java como lenguaje de programación y Android Studio como IDE de desarrollo, se usó los servicios de firebase para el manejo de la base de datos en la nube, para el registro de cuentas de usuarios y para el envío de notificaciones, todo esto se logró a través de la aplicación de Mobile – D como metodología de desarrollo.

Palabras clave: Aplicación móvil, almacenamiento en la nube, Android, firebase, tareas académicas, Mobile – D.

ABSTRACT

Use of technology helps to ensure the fulfillment of homework having a positive impact on student academic training, because in the educational environment homework plays an important role, because it reinforces the knowledge acquired by students in the classroom thus achieving a successful teaching-learning process.

The objective of this project is to assist students and teachers of the Emanuel Educational Unit in the processes of academic homeworks by providing a technological tool for this purpose, so reducing the rate of non-compliance and helping the management of new knowledge.

The firebase Android application is used in the sending and receiving homeworks process, saving their information, their details, and their respective instructions in the cloud so that students always have access to them, also notifications are sent reminding them to perform the homeworks and review them for their respective submission, achieving that the compliance rate is higher.

The application was developed using java as programming language and Android Studio as development IDE, firebase services were used to manage the database in the cloud, to register user accounts and to send notifications, all this was possible through the Mobile - D application as development methodology.

Keywords: Mobile application, cloud storage, Android, firebase, homeworks, Mobile – D.

CAPÍTULO I.- MARCO TEÓRICO

1.1 Tema de investigación

APLICACIÓN FIREBASE ANDROID PARA LA GESTIÓN DE TAREAS ACADÉMICAS A LOS ESTUDIANTES DE EDUCACIÓN BÁSICA EN LA UNIDAD EDUCATIVA EMANUEL.

1.1.1 Planteamiento del problema

El mundo afronta una situación anormal, el confinamiento por el virus del COVID-19, provocando dificultad en varios ámbitos de nuestra vida diaria, uno de los aspectos más afectados es la educación, que debido a la pandemia se ha tenido que realizar de forma virtual, aunque en la actualidad se está retomando las clases de forma presencial, la pandemia ha acrecentado las dificultades que ya existían para la gestión de las tareas académicas[1]. El cumplimiento de tareas es un punto clave en el proceso de aprendizaje pero existen muchos problemas para realizarlas y entregarlas, también vale recalcar que la pandemia aún no ha desaparecido, y la posibilidad de que esta fuerce nuevamente las clases de forma virtual está presente, es por eso que se debe buscar una solución a los problemas ya evidenciados con el propósito de disminuir el impacto negativo en el proceso de interaprendizaje.

El diario “El Periódico” en Barcelona, España cita lo siguiente, el confinamiento pone a la luz muchos defectos del sistema educativo que es incapaz de dar una respuesta clara y efectiva, que vaya más allá de cada escuela, de cada profesor y de cada alumno, la brecha digital existente imposibilita que muchos niños puedan seguir aprendiendo, siendo esto un indicador implacable y vergonzoso de desigualdad de oportunidades[1]. Dichas desigualdades siempre han existido incluso antes de la pandemia, pero no se les daba la importancia necesaria, es por lo que existe la necesidad de garantizar que los estudiantes tengan un medio que garantice su proceso de aprendizaje.

Ante la situación planteada Muñoz y Lluch dicen, tanto familiares como alumnos se han visto en la obligación de adquirir las herramientas necesarias para seguir con los temas de clase y con el cumplimiento de tareas desde casa. Bajo el contexto de la

suspensión de actividades presenciales en centros educativos, las tareas escolares resultan ser la alternativa óptima para garantizar la continuidad del proceso de enseñanza[2]. A pesar del valor de este hecho, lamentablemente no se le da la importancia necesaria, entregando tareas mal realizadas o simplemente no entregándolas.

Por otra parte, en Ecuador la pandemia ha afectado duramente a un sistema educativo que no estaba preparado para afrontar una situación así. Según un estudio realizado en la ciudad de Ibarra, se resalta que el principal problema en el sistema educativo es el incumplimiento de tareas, por falta de recursos tecnológicos, la mayoría de los estudiantes no disponen de un computador con acceso a internet en sus hogares, gran cantidad de ellos solo disponen de un celular, por lo que se le deben enviar tareas para que lo realicen de manera física, para posteriormente enviar la foto a través de WhatsApp o correo electrónico[3]. Esta forma de entregar tareas es tediosa y poco eficiente dificultando el proceso y por ende incrementado el nivel de incumplimiento.

La ciudad de Ambato no es la excepción, como indica un estudio realizado en la Unidad Educativa Iberoamérica, del total de la muestra tomada, el 38% de los estudiantes no cumple con las tareas enviadas, la mayoría de estos casos se da por que la mayoría de los estudiantes olvida que tareas que debe realizar[4]. El hecho de olvidar las tareas asignadas provoca que estas no puedan ser cumplidas o se cumplan mal, por falta de información y de guías de cómo deben ser realizadas.

1.2 Antecedentes investigativos

Al realizar una investigación bibliográfica sobre el tema en diferentes trabajos de titulación de universidades ecuatorianas, así como en artículos de revistas científicas, se han encontrado trabajos que servirán de apoyo para la presente investigación.

Analizando la importancia que tiene Firebase para el proyecto, así como su trabajo en conjunto con desarrollo móvil en Android, se destaca las principales ventajas, del uso de Android hoy en día y las ventajas que Firebase brinda a la hora del desarrollo.

Según German Cardiel Altemir[5] en su tesis “Desarrollo de una aplicación Fitness en el sistema operativo Android con la plataforma Firebase”, realizada en la Universidad de Lleida. En el año 2019 se puede determinar que:

- Firebase es un servicio de Google, utilizado para el desarrollo móvil en la nube, cuyas condiciones de seguridad y también de procesamiento de información se aplican para todos los servicios que ofrece.
- Firebase tiene certificaciones bajo los principales estándares mundiales de seguridad y privacidad a través del cumplimiento de normas de la Organización Internacional de Normalización (ISO). En el momento en que una aplicación conecta con Firebase, los datos se encuentran encriptados, esto también ocurre en el proceso contrario es decir cuando Firebase intenta conectar con alguna aplicación, porque los servicios de Firebase encriptan la información que está transitando con el Protocolo Seguro de Transferencia de Hipertexto (HTTPS), además de aislar los datos de los clientes de manera lógica.
- Firebase también reduce el acceso hacia los datos, aplicando normas de seguridad altas, manteniendo un acceso mínimo y subiendo el nivel de seguridad.

Según Marina Castellote García[6] en su tesis “Desarrollo de una aplicación Android de apuestas utilizando Firebase para la sincronización de datos”, realizada en la Universidad de Jaume I. En el año 2017 se puede concluir lo siguiente:

- Las aplicaciones móviles permanecen siempre al lado de los usuarios debido, principalmente, a que solo en ciertos casos se separan de su celular. Es por este motivo que son el lugar perfecto para notificar y mantener informado al usuario.
- Firebase proporciona un servicio de backend, permitiendo que, a través de su Interfaz de Programación de Aplicaciones (API), se pueda tanto guardar información como sincronizarla en tiempo real, todo esto sin que el desarrollador programe, por lado del servidor, grandes cantidades de código.
- Para el desarrollo de una aplicación móvil, existen varias opciones: lenguaje nativo, aplicación híbrida e inclusive una aplicación web. Sin embargo, al

realizar una aplicación en lenguaje nativo se obtiene un mejor rendimiento, debido a que una aplicación nativa esta optimizada para un sistema operativo en concreto, adaptándose mejor al hardware y por ende mejorado la experiencia del usuario. Además, posee mayor seguridad que una aplicación web, porque en este caso la seguridad depende principalmente del navegador web y es más fácil de desarrollar que una aplicación híbrida.

- Problemas de estabilidad y de desarrollo están completamente solucionados con Firebase, debido a la gran cantidad de funciones que posee, las cuales son: base de datos en tiempo real, servicio de autenticación, solución de errores, almacenamiento de archivos y medida de estadísticas recogidas desde los usuarios.

Además, se ha analizado el papel clave que juega la tecnología en la educación y el impacto positivo que tiene en el proceso de enseñanza, siempre y cuando se las utilice correctamente, aprovechando todas las bondades que brinda para facilitar el trabajo de docentes y de estudiantes.

Según Martha Jiménez y María Ortega[7] en su artículo científico “El Uso de una Aplicación Móvil en la Enseñanza de la Lectura” publicada en la revista Información Tecnológica. En el año 2017 se puede determinar que:

- Una sociedad que no posee educación está destinada a fracasar en varios aspectos, ya que no posee las herramientas necesarias para afrontar las demandas del mundo moderno.
- El uso de tecnología en el proceso de enseñanza provoca una mejora en la condición anímica, debido al uso de un recurso vanguardista provocando una sensación de inclusión al mundo actual, además los usuarios indican que se encuentran muy satisfechos con el uso de una aplicación móvil que es fácil de manejar.
- El uso de las Tecnologías de la Información y la Comunicación (TIC) en el proceso de enseñanza-aprendizaje, genera una condición de autonomía en el usuario, diluyendo los obstáculos que se oponen a este proceso y también

facilitándolo en gran manera, reduciendo el trabajo de estudiantes y maestros, aumentando el tiempo de estudio y obteniendo mejores resultados.

Se analiza la gran problemática que es el incumplimiento de tareas por parte de los estudiantes y los impactos negativos que tiene en el sistema educativo.

Según Jhury Bermeo Urgiles[8] en su tesis “Análisis de los factores asociados al incumplimiento de tareas escolares, en cuarto año de educación básica, de la escuela Juan León Mera, Parroquia Molleturo”, realizada en la Universidad Politécnica Salesiana Sede Cuenca. En el año 2019 se puede concluir lo siguiente:

- Existe una gran falencia en la metodología de enseñanza actual, además de la desmotivación por parte de los estudiantes que se ve reflejada en un desinterés total por estudiar. Los principales problemas son: desintegración familiar, falta de afecto familiar y carencia de vigilancia paternal debido, principalmente, a que la situación económica obliga a trabajar a ambos padres, ocupando la mayor parte de su tiempo en asuntos laborales y descuidando el control y revisión de tareas escolares de sus hijos.
- Dentro del proceso enseñanza-aprendizaje, la metodología utilizada para impartir clases juega un papel muy importante, tanto así que, una metodología innovadora, divertida y fácil, garantizara que el estudiante se motive por aprender más cada día y cumpla sus tareas a tiempo. Es por lo que el docente debe buscar estrategias basadas en el uso de las TIC, que sean fáciles de usar y a la que todos puedan acceder.
- Las causas y factores asociados para el incumplimiento de tareas son de tipo socioeducativas, como: motivación, conocimientos adquiridos para el desarrollo de tareas, falta de información sobre las actividades pendientes y de tipo socioeconómicos como: falta de recursos, todo esto conlleva a que el estudiante pierda el interés y la motivación para estudiar, descuidando sus obligaciones como estudiante.

Además, se analiza el impacto que tiene el uso de una aplicación móvil para el control de tareas, listando así los puntos más importantes sobre este tema.

Según Rosa Navarrete y Ambar Rodríguez[9] en su tesis “El diario escolar electrónico para la gestión del control de tareas”, realizada en la Universidad de Guayaquil. En el año 2018 se concluye lo siguiente:

- Al implementar el software se mejorará el proceso de envío de tareas que son cotidianas en un día a día de los estudiantes. Facilitando que el padre de familia tenga acceso siempre a las actividades escolares asignadas a su representado a través de una aplicación móvil, lo cual permitirá un mayor control, evitando que las tareas no sean cumplidas por falta de información, ya que muchas veces los estudiantes, debido a varias razones, se olvidan de sus responsabilidades.
- Se busca otras alternativas para gestionar el control de tareas, mejorando la comunicación entre los padres y el docente, promoviendo así que el rendimiento académico mejore, contribuyendo a alcanzar los estándares de calidad requeridos para todas las instituciones educativas a nivel nacional.
- Es de gran importancia, conocer sobre la tecnología en edades tempranas, ya que el proceso cognitivo de los niños les permite aprender con mayor facilidad y adaptarse mejor a los cambios.
- La tecnología ha ayudado a que la tasa de deserción escolar disminuya, mejorando el promedio de asistencia además de que las habilidades de aprendizaje también mejoran significativamente. Además, el trabajo colaborativo se puede hacer mejor, más fácil y rápido, ayudando a que los estudiantes puedan acercarse e intercambiar ideas, mejorando las habilidades de interacción social muy necesarias en el mundo actual.

1.3 Fundamentación teórica

1.3.1 Desarrollo de software

Al hablar sobre desarrollo de software hacemos referencia al estudio de todos los componentes que son necesarios a la hora de crear, gestionar, mantener y testear el software, las personas que cumplen con este proceso se le conoce como desarrolladores, personas que tienen la capacidad de construir sistemas informáticos y una vez construidos son capaces de implementarlos y hacerlos funcionar, utilizado

para este fin uno o más lenguajes de programación[10]. El desarrollo de software es fundamental en la sociedad, avanzada tecnológicamente, en la que vivimos debido a que el uso de software es más común, en muchas actividades cotidianas que tiempo atrás eran impasables realizarlas a través de un programa informático.

El desarrollo de software tiene sus inicios en la década de los 40 debido a la aparición de las primeras computadoras, para esa época el desarrollo de software no contaba con estándares ni parámetros para realizarlo, desde esa fecha hasta el hoy ha tenido un desarrollo evolutivo muy rápido[11]. Al inicio el desarrollo de software era totalmente empírico, sin parámetros que marcaran su ejecución lo cual provoca que la mayoría de los proyectos fracasaran generando una crisis con el software.

Es solo hasta la década de los 70 que se empieza a utilizar el término de “Ingeniería de software”, principalmente debido al uso de modelos y guías básicas, que dividen al proyecto en etapas, que abarcan desde su conceptualización hasta su entrega. En cada una de estas etapas se crean normas y parámetros, garantizando el cumplimiento de cada una de estas etapas y por ende el desarrollo del software [11]. El aplicar modelos al desarrollo ayudo a que este se pudiera cumplir de mejor manera y en menos tiempo, teniendo un incremento masivo y generando nuevas técnicas para el desarrollo, que acortaran los tiempos de entrega y garantizaran que el software funcione correctamente.

Hoy en día el desarrollo de software esta estandarizado lo cual garantiza un producto final de calidad, logrando satisfacer las necesidades del usuario facilitando muchas de las actividades diarias que realiza [10]. El desarrollo de software es muy importante porque el mundo se encuentra automatizado, y es necesario innovar día a día, creando nuevos softwares que faciliten la vida de las personas o mejorando los que ya existen.

1.3.2 Desarrollo móvil

Cuando se hace referencia al concepto de desarrollo móvil, se habla sobre el conjunto de requerimientos involucrados en el desarrollo de software enfocado para su ejecución en dispositivos informáticos inalámbricos y pequeños, tales como smartphones, entre otros dispositivos tecnológicos[12]. Dicho desarrollo es muy

importante porque se pueden programar aplicaciones que puedan ser usadas por los usuarios en cualquier lugar.

El desarrollo de aplicaciones móviles nace a partir del desarrollo de software, pero con una diferencia muy marcada, ya que dicho desarrollo está enfocado en aprovechar al máximo las características que ofrecen los dispositivos móviles[13]. Esto ha facilitado la creación de aplicaciones únicas, que mucho tiempo atrás era impensables, principalmente por el limitante que ofrecía el hardware, hoy en día se pueden encontrar aplicaciones móviles específicas para cada dispositivo, las cuales se encuentran optimizadas, garantizando un óptimo funcionamiento.

Debido al avance de la tecnología, en la actualidad, existen muchos dispositivos que son pequeños y fáciles de llevar, esto ha provocado un gran incremento en el desarrollo móvil, ya que ahora no es necesario que un programa se ejecute en una computadora, sino que los dispositivos móviles pueden tranquilamente llevar acabo un gran número de tareas. Las aplicaciones móviles, son la mejor forma de llegar a los usuarios, tal es su impacto que la mayoría de las aplicaciones hoy en día son multiplataforma, ahí es donde radica la importancia del desarrollo de aplicaciones móviles.

1.3.3 Aplicaciones Android

Cuando se va a desarrollar una aplicación móvil, la mejor opción es desarrollarla de forma nativa, es decir en su propio lenguaje, ya que este tipo de desarrollo ofrece ventajas muy relevantes en contra de aplicaciones híbridas o de tipo web.

A parte de eso el desarrollo de aplicaciones Android son, dentro del entorno de aplicaciones móviles nativas, una de las mejores opciones, ya que se posee una gran cantidad de guías de diseño, como ejemplo más relevante se ubica el “Material Design”, una guía de diseño desarrollada por Google, la cual abarca normativas de diseño para que las aplicaciones desarrolladas tengan un buen aspecto visual ayudando a que el usuario disfrute de esta experiencia en cualquier dispositivo móvil ajustándose a las mediadas del mismo [6]. Android al ser el sistema operativo que maneja Google posee una gran documentación, por lo tanto, su desarrollo es relativamente más fácil de realizar.

Otra de las principales ventajas del desarrollo Android, es que este sistema operativo es el más utilizado en dispositivos móviles, por lo tanto, las aplicaciones realizadas para este entorno tendrán mayor impacto en el mundo ya que podrán ser usadas por un mayor número de personas. A parte de eso, está el hecho de que Android es un sistema operativo completamente libre, por lo tanto, se lo puede adquirir sin ningún costo ya sea para instalarlo en algún dispositivo o como para desarrollar aplicaciones con el[14]. El hecho de que sea libre ha provocado que el fabricar celulares con este sistema operativo sea más barato, por ende, los precios para el público también bajan, aumentando la demanda y ampliando más el uso de Android, además que el desarrollo de aplicaciones se puede hacer de manera más eficiente, ya que puede probarse en un mayor número de usuarios, en poco tiempo, detectar fallos y corregirlos.

Hoy en día la programación en dispositivos móviles es fundamental, y dentro de este ámbito, el desarrollo de aplicaciones para el sistema operativo es crucial, debido a que este sistema operativo es el más utilizado a nivel mundial, por lo tanto, el desarrollo de estas aplicaciones tendrá un mayor impacto global, ya que pueden llegar a un mayor número de usuarios. Además de esta gran ventaja, también se incluye que existe una gran documentación sobre el desarrollo de aplicaciones Android, lo cual facilita la programación.

1.3.4 Aplicación Android con tecnología firebase

Firebase es un servicio de Google, que ayuda al desarrollo de aplicaciones móviles, puede ser utilizado para el desarrollo web, desarrollo aplicaciones móviles tanto para Android como para IOS. Brinda un servidor de backend para que las aplicaciones puedan acceder a el, con un gran índice de seguridad, sin el uso de grandes líneas de código, además brinda servicios para autenticación, base de datos en tiempo real, solución de errores, funciones de testeo, almacenamiento de archivos en la nube y medición de estadísticas en base a los datos de los usuarios [6]. Al usar Firebase como el backend de aplicaciones Android generamos un sistema fuerte con alto índice de seguridad y con una alto grado de accesibilidad a los datos.

Una aplicación firebase Android, es una gran opción para el desarrollo de aplicaciones móviles, debido a que el sistema operativo Android es el sistema operativo que Google maneja, por otro lado, el servicio de Firebase también fue diseñado por Google, lo cual genera un alto índice de compatibilidad, además de la gran documentación que existe para su desarrollo.

Al generar una aplicación Android logramos que la mayoría de las personas en el mundo puedan acceder a ella, ya que hoy en día casi todas las personas cuentan con dispositivos móviles, y al combinar este desarrollo con firebase, usando la función de guardar archivos en la nube, garantiza el acceso a la información en cualquier momento.

1.3.5 Conocimiento científico

Se conoce como conocimiento científico al conjunto de hechos que son verificables y sustentados en evidencia basados en teorías científicas, todo esto siguiendo el método científico garantizando la validez de la información[15]. El conocimiento científico sigue una metodología para llevarse a cabo, se diferencia mucho del conocimiento empírico que se basa en la experiencia y no sigue ningún método para ser adquirido.

También el conocimiento científico es coherente, preciso se base en la objetividad, dejando de lado la subjetividad, para así llegar a ser universal, todo esto siguiendo un conjunto de pasos ordenados. Además, es verificable y está disponible para ser sometido a pruebas que comprueben o no su validez, a través de este conocimiento se puede comprender y explicar de mejor manera los fenómenos naturales y la realidad[16]. Al ser objetivo deja de lado las creencias de las personas, no se basa en supuestos, todo lo que se llame conocimiento científico debe haber sido construido siguiendo un método y puede ser verificable, además de servir de base para la construcción de nuevo conocimiento, es por lo que el método científico es la base para construir este conocimiento.

El conocimiento científico tiene como objetivos: comprender y explicar, basado en la objetividad y la precisión, porque ocurren las cosas; establecer leyes, principios y relaciones para los fenómenos; analizar las leyes que se rigen de forma natural e

instaurar conclusiones que sean validas universalmente[17]. El conocimiento científico es la base para el avance tecnológico de la sociedad, que se ha ido acumulando a lo largo de generaciones, dando solución a problemas críticos de la sociedad, este conocimiento mientras las personas existan no se detendrá.

1.3.6 Proceso de aprendizaje

Al hablar sobre el proceso de aprendizaje, no se puede dejar de lado el comportamiento humano, es por lo que se define al proceso de aprendizaje al conjunto de métodos utilizados para adquirir nuevas habilidades, conductas y conocimiento[18]. El aprendizaje debe ser constante en la vida de las personas es por lo que se habla de un proceso que se está ejecutando día a día, ayudando a que las personas aprendan nuevas cosas.

El proceso de aprendizaje tiene sus bases en el estudio , en la experiencia, en el razonamiento y en la observación. En el influyen una gran serie de factores como la sociedad en la cual la persona habita, los valores y principios adquiridos en su familia, cambios de conducta, entre otros factores propios del comportamiento humano [18]. Todo lo que aprendemos ya sea de manera empírica o científica, sigue un proceso para que pueda ser parte de nosotros y poder aplicarlo en algún momento, donde este conocimiento adquirido, a través de este proceso de aprendizaje, pueda sernos de utilidad.

Por otra parte, el aprendizaje humano tiene una íntima relación con la educación ayudando al desarrollo personal, es por eso la importancia de que sea orientado correctamente y es más efectivo cuando la persona tiene un deseo de aprender. Este proceso para aprender es de suma importancia para ciencias como la educación, la psicología educacional, la pedagogía y la neuropsicología[19]. El aprendizaje es fundamental para la educación y para el desarrollo personal de cada individuo, potenciado sus habilidades.

El proceso de aprendizaje es la base de la educación, por lo que este proceso debe ser garantizado a través de metodologías eficientes, que ayude a los estudiantes a aprender, aplicar y recordar todo el conocimiento adquirido.

1.3.7 Tareas escolares

Se define como tarea escolar al trabajo asignado por parte de los profesores hacia sus estudiantes, dichas actividades deben ser cumplidas fuera del aula, es decir en la casa[20]. Las tareas escolares son trabajos adicionales necesarios, que ayudan a reforzar todo lo visto dentro del aula, para así garantizar un eficiente proceso de aprendizaje.

Dentro del proceso de enseñanza-aprendizaje en unidades educativas, las tareas escolares son fundamentales, ya que ayudan a reforzar el conocimiento adquirido, planteando problemas y actividades que deben ser resueltas con lo aprendido en el aula[21]. Dada la importancia que tienen las tareas escolares, deben ser cumplidas para así garantizar el aprendizaje por parte de los estudiantes.

El papel que juega los padres es fundamental, ya que ayudan a que las tareas sean cumplidas de manera correcta, principalmente en alumnos de corta edad que recién están entrando en el sistema educativo. Alrededor de esto existe un debate ya que algunos pedagogos afirman que las actividades escolares fuera del aula son excesivas y no siguen una metodología que garantice su eficacia a la hora de aprender, por otra parte, también están quienes argumentan que si no se envían estas actividades extras el estudiante no será capaz de reforzar su conocimiento[22]. El apoyo de los padres, una buena metodología y la cantidad correcta de tareas escolares garantiza un óptimo aprendizaje.

Las tareas escolares son de suma importancia a la hora de aprender, porque no basta solo con las horas de clases en las aulas, el estudiante necesita de actividades extras que ayuden a reforzar, evaluar, aplicar y recordar el conocimiento adquirido en las aulas de clases, pero una cantidad excesiva de tareas y sin una metodología que las haga sencillas y divertidas, provocará estrés en los estudiantes, dificultado el proceso de aprendizaje. La influencia de los padres es de suma importancia, ya que, principalmente en la niñez, los estudiantes necesitan de alguien que los guíe y ayude a ser responsables.

1.3.8 Envío de tareas a los estudiantes

Las tareas escolares siempre han sido enviadas de la misma forma durante mucho tiempo, el docente dice la actividad que se debe realizar, el alumno copia las instrucciones y una vez estando en su casa se propone a realizarlas, después a la siguiente clase el profesor recoge las tareas y procede a calificarlas [20]. Todo este proceso tenía dificultades si el alumno no copio o entendió mal la instrucción, la tarea realizada también estaba mal, o a veces el alumno simplemente no realizaba la tarea, lo cual provoca que el aprendizaje no sea completo.

Para controlar y recordar el envío de tareas escolares de los docentes hacia sus estudiantes se utilizan plataformas dedicadas para otros fines, como WhatsApp, dificultado este proceso [3]. Al no contar con una aplicación que sea específica para el envío de tareas, los docentes optan por usar otras plataformas, que son difíciles de usar, provocando que las instrucciones para la realización de tareas no lleguen a todos los estudiantes, lo cual provoca que las actividades asignadas no sean cumplidas por todos los estudiantes dificultando el proceso de aprendizaje.

El proceso de envío de tareas presenta muchas dificultades, debido a la ausencia de padres a la hora de clases, porque la mayoría de ellos tiene que salir a trabajar, los estudiantes se quedan solos a la hora de recibir clase y la mayoría de ellos no presta atención a las clases y no conoce las actividades que el docente les ha asignado [23]. Al no estar presentes los padres cuando las tareas son asignadas, provoca que se desconozca la actividad que se debe realizar por ende existe un incumplimiento de esta.

El envío y entrega de tareas siempre ha tenido dificultades, pero hoy en día, debido a la situación actual, estas dificultades han ido en aumento, aumentando el índice de incumplimiento, dificultando más el proceso de enseñanza-aprendizaje, todo esto dentro de un sistema educativo que no estaba listo para afrontar esta situación. Es por lo que el envío de tareas debe utilizar herramientas tecnológicas que faciliten su cumplimiento y garanticen que la información sobre las actividades a realizar este disponible tanto para padres, como para alumnos en cualquier momento.

1.3.9 Almacenamiento local

El almacenamiento local se puede definir como el almacenamiento de los datos en dispositivos físicos, como discos duros, pen drives, entre otros[24]. Este tipo de almacenamiento hoy en día es muy importante y no se debe dejar de lado.

También se puede decir que el almacenamiento local, es el almacenamiento básico de cualquier dispositivo tecnológico que requiera algún tipo de almacenamiento de datos para poder funcionar, y es el que más se usa porque es el más fácil de implementar.

1.3.10 Almacenamiento en la nube

El almacenamiento en la nube se define como un servicio mediante el cual se puede almacenar una gran cantidad de datos, dichos datos se los transfiere a través de internet o de alguna otra red, hacia un sistema que los almacena externamente y que está a cargo de un tercero[25]. Esto permite guardar diferentes tipos de datos como fotos, videos y documentos, de manera más segura que un almacenamiento local.

La mayoría de los sistemas de almacenamiento en la nube son escalables, esto permite adaptarse a las diferentes necesidades de los usuarios a la hora de almacenar datos, permitiendo acceder a los mismo desde cualquier lugar o dispositivo[26]. Todo esto garantiza una mayor disponibilidad ya que se puede acceder a los datos desde cualquier punto del planeta en donde se tenga una conexión a internet.

El almacenamiento en la nube hoy en día es una gran herramienta que la mayoría de las empresas usa para garantizar que su información no se pierda y tenga una alta disponibilidad para ser usada cuando se requiera, además que esto permite respaldar dicha información sin preocuparse por la seguridad ya que de esto se encarga la empresa que ofrece el servicio de almacenamiento.

1.3.11 Firebase

Firebase es una plataforma creada por Google que inició en el año 2014, esta plataforma ayuda al desarrollo de aplicaciones de mayor calidad de manera rápida y eficiente, se pueden desarrollar aplicaciones tanto en Android, iOS y web[27].

Cuando se trata de desarrollo firebase ayuda a la creación de mejores aplicaciones, ya que permite minimizar el tiempo de optimización, así como el tiempo de desarrollo, gracias a la gran cantidad de funciones que posee entre las cuales destacan el testeado y la detección de errores. Además de eso permite almacenar todos los datos en la nube, testar la aplicación o la configuración remota son funciones destacables que ofrece[28]. Debido a la gran cantidad de funciones que ofrece firebase es una de las plataformas más utilizadas por los desarrolladores para la creación de aplicaciones.

Firestore posee una gran cantidad de servicios entre los cuales destacan:

- **Cloud firestore:** es una base de datos NoSQL, que permite almacenar los datos de manera escalable, así como sincronizarlos desde las aplicaciones móviles desarrolladas [29]. Además, se garantiza la disponibilidad y seguridad de los datos.
- **Firestore Authentication:** este servicio ayuda a la gestión de usuarios mediante la creación de un sistema de autenticación rápido de usar, fácil de gestionar y con una seguridad muy elevada[29]. Esto garantiza una gestión de usuarios muy eficiente.

1.3.12 Metodologías ágiles

Las metodologías ágiles nacieron en el 2001 a cargo de un grupo de programadores, son aquellas que ayudan adaptar la forma de desarrollo de un proyecto , logrando que este tenga mayor flexibilidad y una capacidad alta, a la hora de adaptarse a las nuevas circunstancias del entorno en el que se encuentra [30]. Estas permiten obtener un mejor producto, logrando que el cliente quede satisfecho con el producto final.

Estas metodologías están formadas por una serie de principios, conocidos como el manifiesto ágil, dichos principios son las bases que deben seguir cualquier metodología que se haga llamar ágil, entre los cuales destacan[31]:

- El cliente es una parte fundamental y debe estar involucrado en todas las fases de entrega , tanto en etapas tempranas como en continuas.

- Los cambios siempre deben ser aceptados, incluso en etapas finales del proyecto.
- Trabajo colaborativo flexible, repartiendo tareas, con tiempos específicos de entrega.
- La simplicidad es clave, evitando las cosas complicadas al realizar las tareas.

Las metodologías ágiles, reúnen características de las metodologías tradicionales permitiendo hacerlas más flexibles, ayudando al desarrollo de proyectos, existen un gran número de metodologías ágiles, entre las cuales destacan: Scrum, XP, Kanban, Lean y Mobile – D.

1.3.13 Mobile – D

Es una metodología para el desarrollo de software, fue creada en el año 2005 en Finlandia, su principal característica es que se basa en una serie de metodologías ya creadas, las cuales son; Extreme Programming(XP), Crystal Methodologies, entre otras[30]. Al basarse en otras metodologías obtiene lo mejor de cada una de ellas.

Uno de los principales objetivos de la metodología es conseguir la ejecución de proyectos en ciclos de desarrollo extremadamente rápidos con grupos de trabajo pequeños, basándose en cinco fases las cuales son: exploración, inicialización, producción, estabilización y pruebas [30]. Lo cual la convierte en una de las metodologías para el desarrollo de software más utilizadas actualmente.

1.3.14 Android Studio

Android Studio es un entorno de desarrollo integrado(IDE) creado en el año 2014, es el IDE de desarrollo oficial para la creación de aplicaciones Android, está basado en IntelliJ IDEA, un entorno de desarrollo para programas informáticos[32]. Ofrece una gran cantidad de funciones que ayudan al desarrollo de manera rápida y eficiente.

Al ser el IDE de desarrollo oficial para aplicaciones Android, permite realizar un gran número de funciones de manera rápida, ayudando a la conexión con Firebase y sus servicios como la base de datos.

1.3.15 Java

Java es un lenguaje de programación orientado a objetos creado en 1995, permite al programador escribir el código una sola vez y ejecutarlo en cualquier dispositivo. Es un lenguaje simple y potente está basado en lenguajes como C y C++, pero omitiendo las funcionalidades obsoletas y potenciando sus ventajas[33]. Es uno de los lenguajes más fáciles de utilizar y uno de los más usados a nivel mundial.

Java es uno de los lenguajes más simples, pero potentes y sigue siendo uno de los más usados en el mundo a pesar de su antigüedad, con una gran cantidad de librerías que facilita la programación para el desarrollo de aplicaciones móviles, además al ser multiplataforma garantiza la compatibilidad con dispositivos Android.

1.3.16 BlueStacks

Es un emulador de Android lanzado en el año 2011 para sistemas operativos Windows, Linux y MacOS, la aplicación sirve para poder usar cualquier aplicación Android en un computador, posee la opción de personalizar botones y un gran rendimiento lo cual la hace uno de los emuladores más utilizados, además se puede enlazar con Android Studio para ejecutar las aplicaciones creadas[34]. Debido a todas las ventajas que ofrece por encima de otras aplicaciones similares es una de las utilizadas y la facilidad que tiene para poder enlazarse con Android Studio lo hace uno de los emuladores favoritos para los desarrolladores de aplicaciones móviles.

1.3.17 Balsamiq

Balsamiq es una herramienta de diseño de prototipos creada en el año 2008, nos permite realizar diseños para las aplicaciones ya sean de tipo web, móviles o incluso de escritorio, una de las principales ventajas es que permite ser usada en línea sin necesidad de instalar nada, además posee una gran variedad de plantillas fáciles de usar lo cual ayuda a realizar el trabajo de manera rápida y eficiente[35]. Debido a estas características permite crear prototipos de interfaces simples y fáciles de entender, ayudando al desarrollo de proyectos.

1.4 Objetivos

1.4.1 Objetivo general

Implantar una aplicación móvil para la gestión de tareas académicas a los estudiantes de educación básica en la Unidad Educativa Emanuel.

1.4.2 Objetivos específicos

- Definir las dificultades a las que se enfrentan los estudiantes y docentes en el proceso de envío y recibo de tareas.
- Analizar la tecnología de almacenamiento en la nube con la plataforma firebase de Google.
- Desarrollar una aplicación Android con tecnología firebase de Google para el envío y recibo de tareas escolares.

CAPÍTULO II.- METODOLOGÍA

2.1 Materiales

Los materiales utilizados para la recolección de datos fueron dos encuestas aplicadas a los estudiantes y profesores de la Unidad Educativa Emanuel, con el objetivo de recolectar información sobre los problemas y dificultades a la hora de la gestión de tareas académicas.

Cuestionario de preguntas sobre la gestión de tareas académicas aplicadas a los estudiantes de la Unidad Educativa Emanuel.

1. ¿Considera usted que la información brindada por el docente sobre la tarea a realizar es clara?

Siempre() Casi siempre() A veces() Casi nunca() Nunca()

2. ¿Considera usted que recuerda todas las tareas académicas que debe presentar?

Siempre() Casi siempre() A veces() Casi nunca() Nunca()

3. Con qué frecuencia, usted realiza una tarea, pero olvida entregarla.

Siempre() Casi siempre() A veces() Casi nunca() Nunca()

4. ¿Considera usted que las tareas ayudan a reforzar los conocimientos adquiridos en clases?

Siempre() Casi siempre() A veces() Casi nunca() Nunca()

5. ¿Usted lleva un registro de las tareas académicas que debe presentar?

Siempre() Casi siempre() A veces() Casi nunca() Nunca()

6. Escoja el principal motivo por el cual cree usted que no se podría presentar una tarea académica.

Olvido() Falta de tiempo() Instrucciones no claras()

7. ¿Cree usted que el uso de la tecnología ayudaría a la presentación a tiempo de tareas escolares?

Siempre() Casi siempre() A veces() Casi nunca() Nunca()

8. ¿Usaría una aplicación informática que le ayude a recordar las tareas escolares?

Siempre() Casi siempre() A veces() Casi nunca() Nunca()

9. ¿Qué tipo de aplicación considera que se debería desarrollar para que ayude a la presentación de tareas escolares?

App() Web() Escritorio()

Cuestionario de preguntas sobre la gestión de tareas académicas aplicadas a los profesores de la Unidad Educativa Emanuel.

1. De acuerdo con su criterio, ¿cuáles son los problemas más habituales para no presentar las tareas?

2. ¿Con que frecuencia los estudiantes no presentan las tareas?

Siempre() Casi siempre() A veces() Casi nunca() Nunca()

3. De acuerdo con su criterio, ¿influye el hecho de no presentar tareas con el rendimiento en evaluaciones?

Siempre() Casi siempre() A veces() Casi nunca() Nunca()

4. ¿De la última tarea que envió a sus estudiantes, cuantos no la presentaron?

Muchos() Algunos() Pocos() Ningunos()

5. ¿Y de los que presentaron la última tarea, cuantos la realizaron correctamente?

Muchos() Algunos() Pocos() Ningunos()

6. ¿Cree usted que una aplicación que recuerde a los estudiantes las tareas a realizar, y además permita entregarlas por ese medio ayudaría a que los estudiantes cumplan eficientemente?

Siempre() Casi siempre() A veces() Casi nunca() Nunca()

7. ¿Qué características debería tener el aplicativo para que permita el recordatorio y la entrega de tareas?

2.2 Métodos

2.2.1 Modalidad de la investigación

En el presente proyecto se aplicaron dos modalidades de investigación las cuales son: bibliográfica-documental y de campo.

- **Investigación bibliográfica-documental:** debido a que se recolectó información de distintas fuentes como tesis, artículos científicos, libros y revistas científicas, para poder tener el sustento científico en el cual basar el proyecto, mediante esta investigación se obtuvo diferentes conceptos sobre el problema, además de guías de cómo llegar a la solución de la problemática.
- **Investigación de campo:** campo debido a que la información que se obtuvo, de la variable dependiente: gestión de tareas académicas a los estudiantes y la variable independiente: aplicación firebase Android, fue en el mismo lugar en donde está el objeto de estudio.

2.2.2 Población y muestra

Población: La población en la cual se va a trabajar son los estudiantes de educación básica y sus respectivos docentes de la Unidad Educativa Emanuel.

Población	Número	Porcentaje
Estudiantes	245	95,70%
Profesores	11	4,30%
Total	256	100%

Tabla 2.1 Población de estudio

Elaborado por: El investigador.

Muestra de estudiantes: Los estudiantes son 245, al superar los 100 se deberá tomar una muestra. Para el cálculo del tamaño de la muestra se considera un nivel de confiabilidad del 95%, y con un error muestral del 5%, con estos datos se aplica la siguiente fórmula:

$$n = \frac{N Z^2 P Q}{N e^2 + Z^2 P Q}$$

Siendo:

N = 245 estudiantes de educación básica

Z = nivel de confiabilidad del 95% o 1,96

P = probabilidad de que ocurra 0,5

Q = probabilidad que no ocurra 0,5

e = error de muestreo del 5% o 0,05

Aplicando la fórmula tenemos:

$$n = \frac{245 * 1,96^2 * 0,5 * 0,5}{245 * 0,05^2 + 1,96^2 * 0,5 * 0,5} = 108,88$$

Obteniendo un resultado de 108,88 siendo el tamaño significativo de la muestra de 109 estudiantes.

Muestra de profesores: En el caso de los profesores, son un total de 11 profesores, los cuales pertenecen a educación básica de la Unidad Educativa Emanuel, al no pasar el tamaño de la población de 100 individuos se trabajará con el total de los profesores.

2.2.3 Recolección de información

Una vez aplicadas las encuestas tanto a estudiantes como a profesores de la Unidad Educativa Emanuel, se obtuvieron los siguientes resultados.

2.2.3.1 Resultados obtenidos de las encuestas aplicadas a los estudiantes de la Unidad Educativa Emanuel sobre la gestión de tareas académicas.

Pregunta 1: ¿Considera usted que la información brindada por el docente sobre la tarea a realizar es clara?

Siempre	Casi siempre	A veces	Casi nunca	Nunca	Total
56	29	19	5	0	109

Tabla 2.2 Tabulación pregunta 1. Encuesta a los estudiantes.

Elaborado por: El investigador.

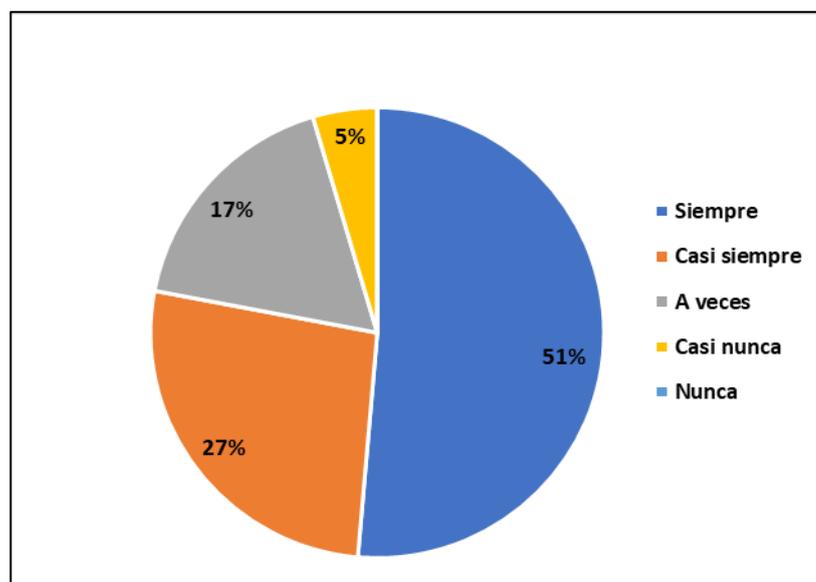


Figura 2.1 Pregunta 1. Encuesta a los estudiantes

Elaborado por: El investigador.

Análisis e interpretación de resultados

Según los datos obtenidos y evidenciados en la **Figura 2.1**, el 51% de los estudiantes dicen que siempre la información brindada por el docente sobre las tareas a realizar es clara, el 27% dice que casi siempre, el 17% a veces y el 5% casi nunca, por otra parte, el indicador de nunca no obtuvo respuestas. Según los datos obtenidos se puede concluir que solo la mitad de los estudiantes entiende completamente las instrucciones sobre las tareas, lo cual provoca que las tareas no sean realizadas o que sean realizadas de manera incorrecta.

Pregunta 2: ¿Considera usted que recuerda todas las tareas académicas que debe presentar?

Siempre	Casi siempre	A veces	Casi nunca	Nunca	Total
25	44	22	13	5	109

Tabla 2.3 Tabulación pregunta 2. Encuesta a los estudiantes.

Elaborado por: El investigador.

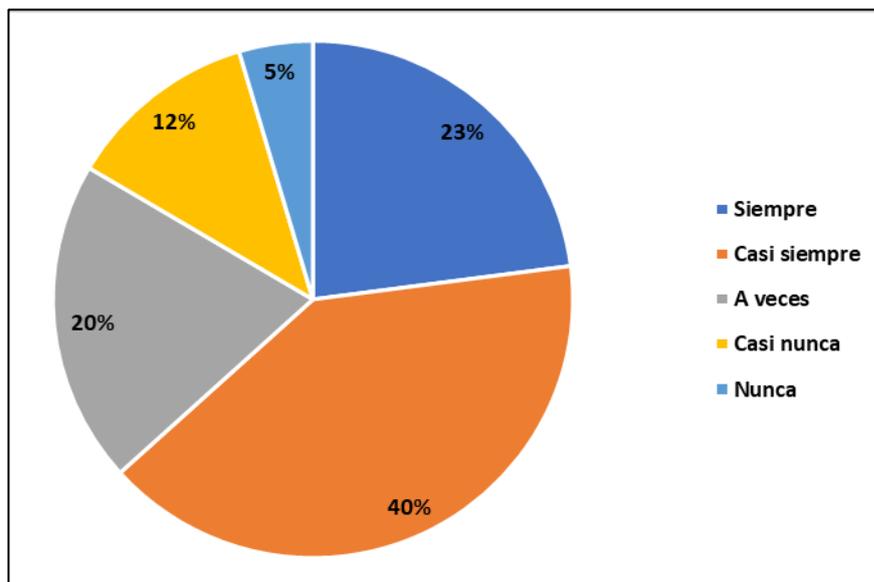


Figura 2.2 Pregunta 2. Encuesta a los estudiantes.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.2**, el 40% de los estudiantes dicen que casi siempre recuerdan las tareas académicas que deben realizar, el 23% dice que siempre, el 20% a veces, el 12% casi nunca y el 5% nunca. Por lo tanto, se puede concluir que la mayoría de los estudiantes no siempre recuerdan todas las tareas académicas que les son asignadas, y al no recordarlas no las realizan.

Pregunta 3: ¿Con qué frecuencia, usted realiza una tarea, pero olvida entregarla?

Siempre	Casi siempre	A veces	Casi nunca	Nunca	Total
7	23	32	30	17	109

Tabla 2.4 Tabulación pregunta 3. Encuesta a los estudiantes.

Elaborado por: El investigador.

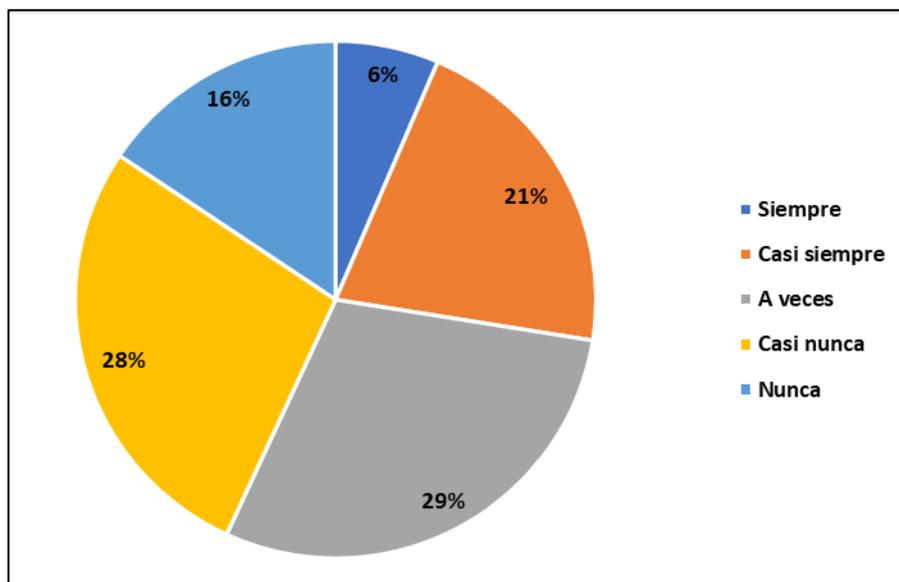


Figura 2.3 Pregunta 3. Encuesta a los estudiantes.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.3**, el 29% de los estudiantes encuestados afirman que a veces realizan las tareas, pero se olvidan entregarlas, el 28% dice que casi nunca, el 21% casi siempre, el 16% nunca y el 6% siempre. Mediante los datos obtenidos se puede afirmar que la mayoría de los estudiantes dice que, si realizan las tareas, pero se olvidan entregarlas, lo cual es un aspecto muy importante al tomar en cuenta, para el desarrollo del proyecto.

Pregunta 4: ¿Considera usted que las tareas ayudan a reforzar los conocimientos adquiridos en clases?

Siempre	Casi siempre	A veces	Casi nunca	Nunca	Total
69	23	11	5	1	109

Tabla 2.5 Tabulación pregunta 4. Encuesta a los estudiantes.

Elaborado por: El investigador.

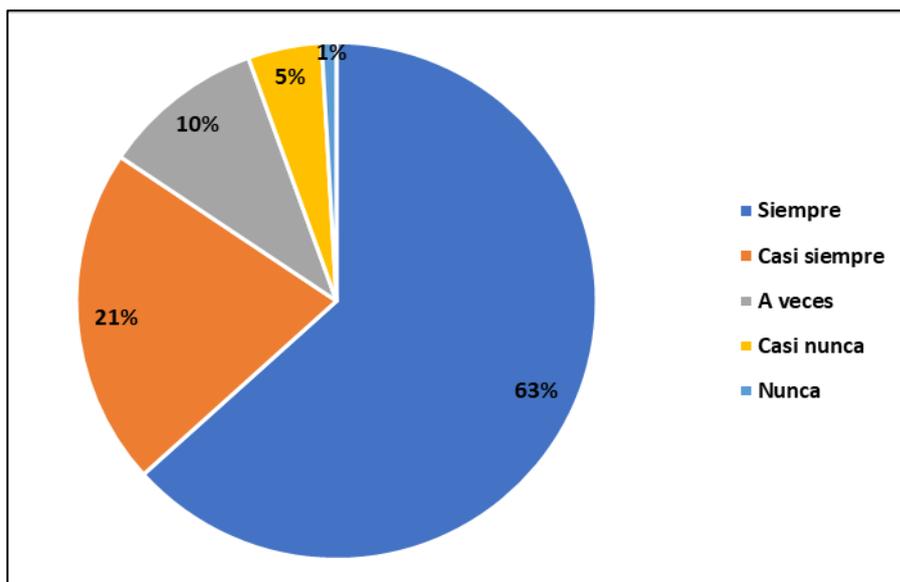


Figura 2.4 Pregunta 4. Encuesta a los estudiantes.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.4**, el 63% de los estudiantes encuestados afirman que siempre las tareas académicas enviadas por los profesores ayudan a reforzar los conocimientos adquiridos en clases, el 21% afirma que casi siempre, el 10% a veces, el 5% casi nunca y el 1% nunca. Se puede evidenciar la importancia de las tareas académicas en el proceso de enseñanza aprendizaje, ya que la mayoría afirma que estas ayudan a reforzar los conocimientos, es por eso la importancia de que sean realizadas de manera correcta.

Pregunta 5: ¿Usted lleva un registro de las tareas académicas que debe presentar?

Siempre	Casi siempre	A veces	Casi nunca	Nunca	Total
34	24	25	14	12	109

Tabla 2.6 Tabulación pregunta 5. Encuesta a los estudiantes.

Elaborado por: El investigador.

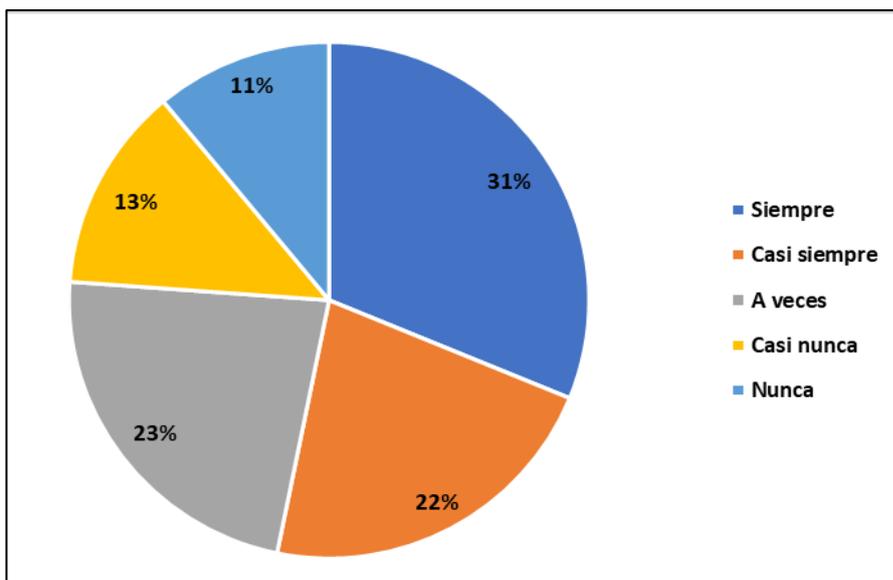


Figura 2.5 Pregunta 5. Encuesta a los estudiantes.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.5**, el 31% de los estudiantes encuestados afirman que siempre llevan un registro de las tareas que deben realizar, el 23% afirma que a veces, el 22% casi siempre, el 13% casi nunca y el 5% nunca. Se puede concluir que solo pocos estudiantes llevan un registro de todas las tareas que deben realizar, en consecuencia, la mayoría de ellos olvida las actividades que deben realizar y no las presentan.

Pregunta 6: Escoja el principal motivo por el cual cree usted que no se podría presentar una tarea académica.

Olvido	Falta de tiempo	Instrucciones no claras	Total
58	33	18	109

Tabla 2.7 Tabulación pregunta 6. Encuesta a los estudiantes.

Elaborado por: El investigador.

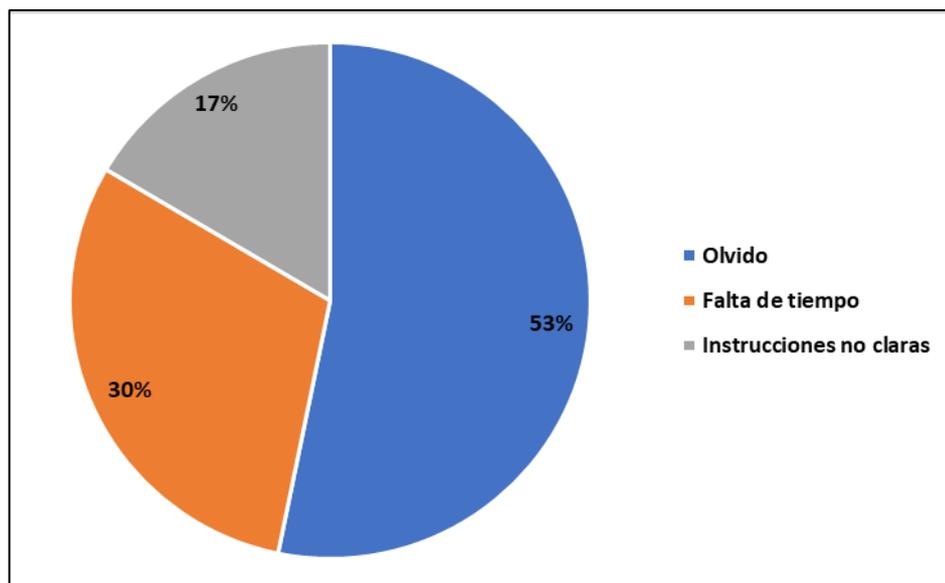


Figura 2.6 Pregunta 6. Encuesta a los estudiantes.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.6**, el 53% de los estudiantes dicen que el principal motivo por el cual no presentar una tarea es porque que se olvidan realizarla, el 30% dice que es por falta de tiempo y el 17% dice que es porque las instrucciones sobre las tareas no son claras. Mediante los datos obtenidos se puede concluir que el principal motivo por que los estudiantes no presentan las tareas es porque se olvidan de que tienen que realizarla afectando así a su rendimiento académico ya que si no realizan tareas no refuerzan el conocimiento adquirido en clases.

Pregunta 7: Cree usted que el uso de la tecnología ayudaría a la presentación a tiempo de tareas escolares.

Siempre	Casi siempre	A veces	Casi nunca	Nunca	Total
49	34	22	4	0	109

Tabla 2.8 Tabulación pregunta 7. Encuesta a los estudiantes.

Elaborado por: El investigador.

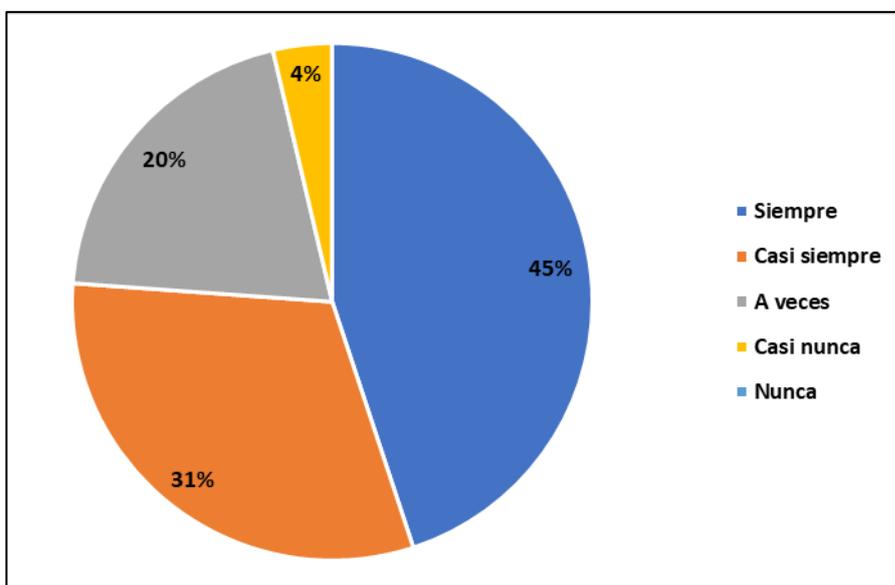


Figura 2.7 Pregunta 7. Encuesta a los estudiantes.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.7**, el 45% de los encuestados dicen que siempre el uso de la tecnología ayudaría a la presentación de tareas escolares a tiempo, el 31% dice que casi siempre, el 20% a veces y el 4% casi nunca, el indicador nunca no obtuvo respuestas. Así que se puede concluir que la mayoría de los estudiantes apoyan la idea que la tecnología ayudará a resolver los problemas que se dan para no presentar las tareas.

Pregunta 8: Usaría una aplicación informática que le ayude a recordar las tareas escolares.

Siempre	Casi siempre	A veces	Casi nunca	Nunca	Total
62	22	17	2	6	109

Tabla 2.9 Tabulación pregunta 8. Encuesta a los estudiantes.

Elaborado por: El investigador.

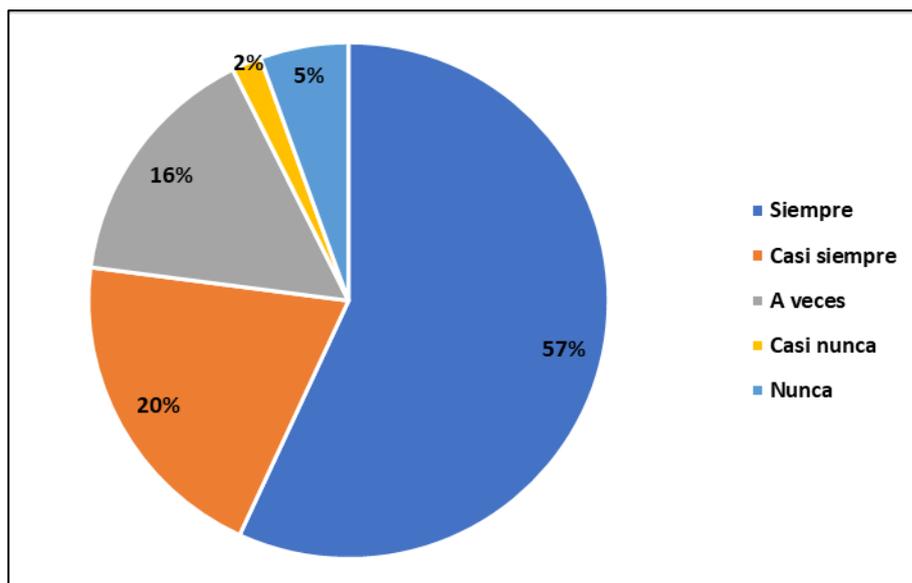


Figura 2.8 Pregunta 8. Encuesta a los estudiantes.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.8**, el 57% de los encuestados dicen que siempre usarían una aplicación informática que le ayude a recordar las tareas escolares que deben presentar, el 20% dice que casi siempre, el 16% a veces, el 5% nunca y el 2% casi nunca. Por lo tanto, se puede concluir que la mayoría de los estudiantes encuestados están a favor de usar una aplicación tecnológica que ayude a la gestión de las tareas académicas, recordándoles que tareas deben presentar para que puedan ser presentadas a tiempo.

Pregunta 9: Qué tipo de aplicación considera que se debería desarrollar para que ayude a la presentación de tareas escolares.

App	Web	Escritorio	Total
60	38	11	109

Tabla 2.10 Tabulación pregunta 9. Encuesta a los estudiantes.

Elaborado por: El investigador.

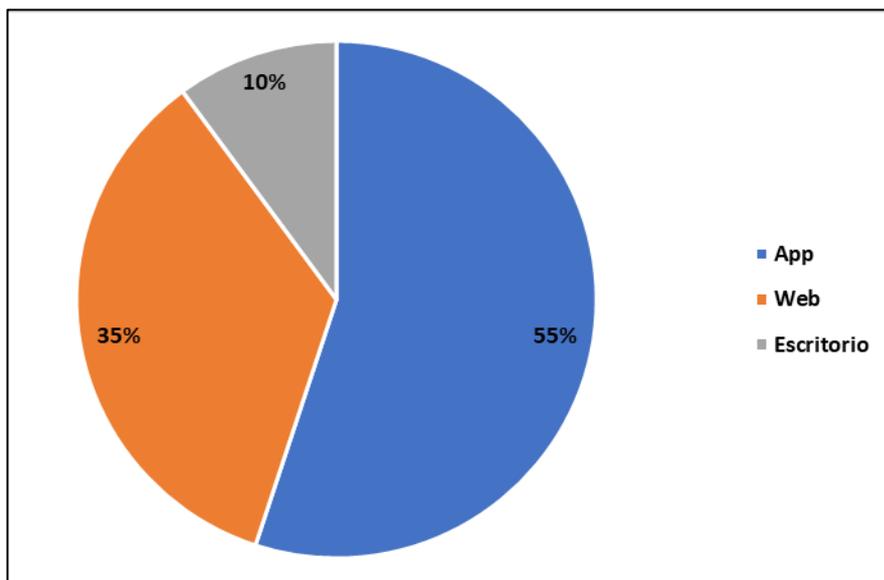


Figura 2.9 Pregunta 9. Encuesta a los estudiantes.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.9**, el 55% de los encuestados dicen que el tipo de aplicación que se debe desarrollar para que ayude a la presentación de tareas escolares es una app, el 35% dice que una web y el 10% de escritorio. Se puede concluir que la mayoría de los estudiantes dicen que la aplicación que prefieren para la gestión de tareas escolares es una aplicación móvil, debido a la facilidad de uso y de disponibilidad.

2.2.3.2 Resultados obtenidos de las encuestas aplicadas a los profesores de la Unidad Educativa Emanuel

Pregunta 1: De acuerdo con su criterio, ¿cuáles son los problemas más habituales para no presentar las tareas?

Olvido	Instrucciones no claras	Falta de atención	Problemas familiares	Falta de supervisión paterna	Total
3	1	1	2	4	11

Tabla 2.11 Tabulación pregunta 1. Encuesta a los profesores.

Elaborado por: El investigador.

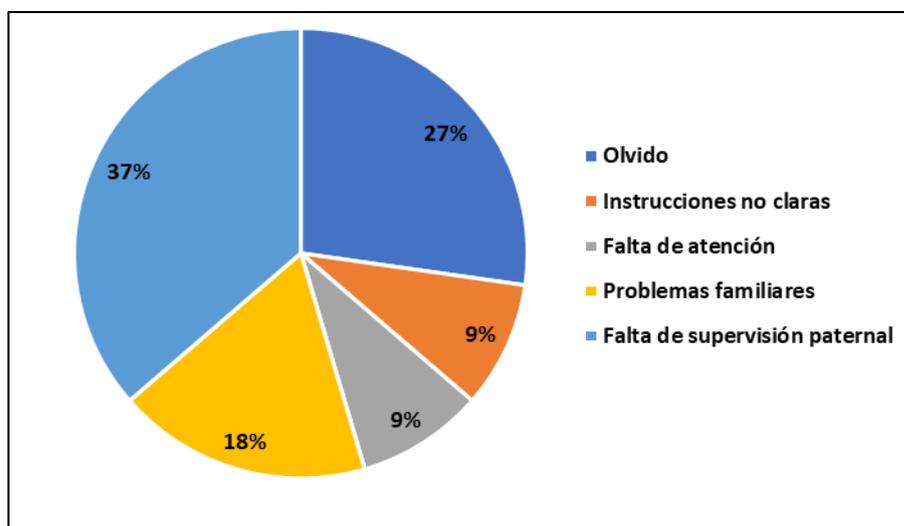


Figura 2.10 Pregunta 1. Encuesta a los profesores.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.10**, el 37% de los profesores encuestados dicen que el problema más habitual para no presentar tareas es la falta de supervisión paterna, el 27% dicen que es el olvido, el 18% por problemas familiares, el 9% por falta de atención y el 9% por que las instrucciones no son claras. Se puede concluir que los dos principales motivos por el cual no se presentan tareas según los profesores son la falta de supervisión paterna y el olvido por parte de los estudiantes.

Pregunta 2: ¿Con que frecuencia los estudiantes no presentan las tareas?

Siempre	Casi siempre	A veces	Casi nunca	Nunca	Total
0	2	7	2	0	11

Tabla 2.12 Tabulación pregunta 2. Encuesta a los profesores.

Elaborado por: El investigador.

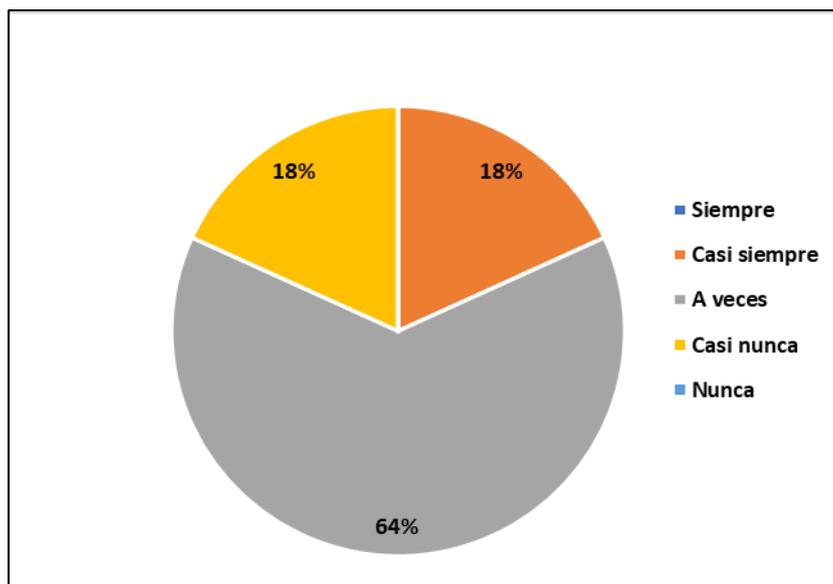


Figura 2.11 Pregunta 2. Encuesta a los profesores.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.11**, el 64% de los profesores encuestados afirman que a veces los alumnos no presentan tareas, el 18% dicen que casi nunca y el 18% casi siempre, los indicadores de siempre y nunca no presentaron ningún resultado. Según los datos obtenidos se puede afirmar que existe un gran índice de alumnos que no presentan tareas de manera habitual.

Pregunta 3: De acuerdo con su criterio ¿influye el hecho de no presentar tareas con el rendimiento en evaluaciones?

Siempre	Casi siempre	A veces	Casi nunca	Nunca	Total
6	4	1	0	0	11

Tabla 2.13 Tabulación pregunta 3. Encuesta a los profesores.

Elaborado por: El investigador.

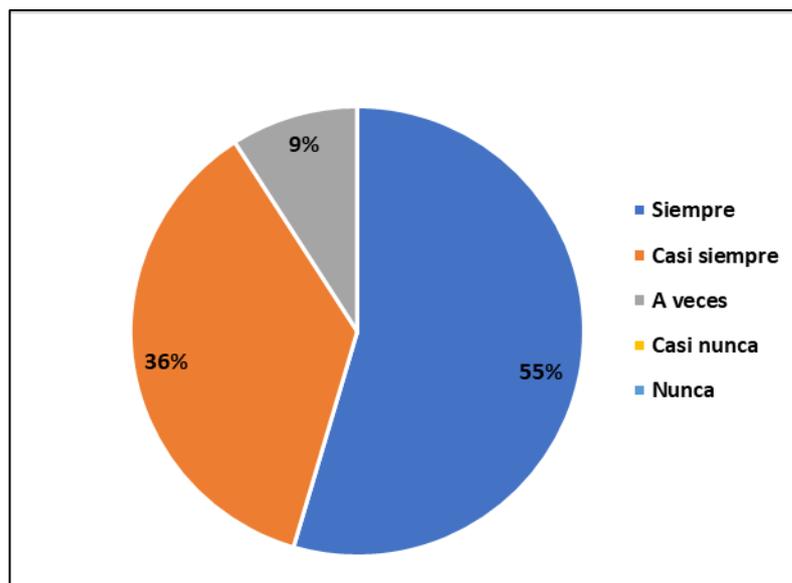


Figura 2.12 Pregunta 3. Encuesta a los profesores.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.12**, el 55% de los profesores encuestados afirman que siempre el hecho de no presentar tareas influye en el rendimiento escolar de los alumnos, el 36% dicen que casi siempre, el 9% a veces, por otro lado, los indicadores de casi nunca y nunca no presentan resultados. Por lo tanto, se puede concluir que, las tareas escolares son muy importantes e influyen en el rendimiento escolar, es por eso por lo que estas deben ser realizadas a tiempo y de manera correcta.

Pregunta 4: ¿De la última tarea que envió a sus estudiantes, cuantos no la presentaron?

Muchos	Algunos	Pocos	Ninguno	Total
0	5	6	0	11

Tabla 2.14 Tabulación pregunta 4. Encuesta a los profesores.

Elaborado por: El investigador.

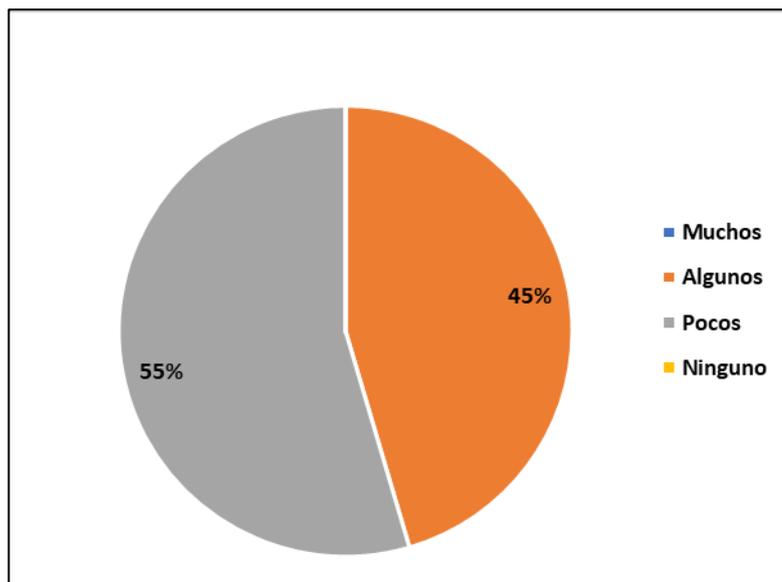


Figura 2.13 Pregunta 4. Encuesta a los profesores.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la *Figura 2.13*, el 55% de los profesores encuestados afirman que de la última tarea enviada pocos de los estudiantes no la presentaron y el 45% dice que algunos, mientras que los indicadores de muchos y ninguno no presentaron respuestas. Por lo que se puede concluir que existen un número significativo de estudiantes que no presentaron la última tarea enviada, lo que genera muchos problemas educativos.

Pregunta 5: ¿Y de los que presentaron la última tarea, cuantos la realizaron correctamente?

Muchos	Algunos	Pocos	Ninguno	Total
4	6	1	0	11

Tabla 2.15 Tabulación pregunta 5. Encuesta a los profesores.

Elaborado por: El investigador.

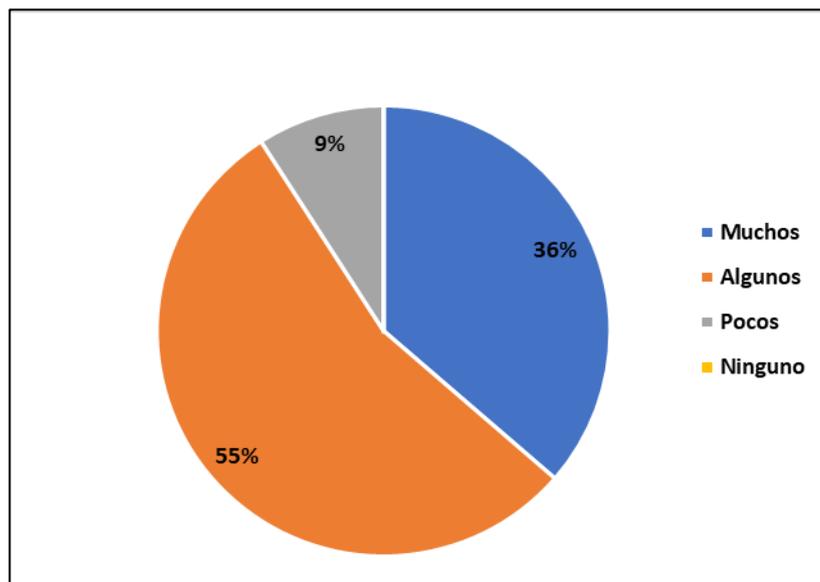


Figura 2.14 Pregunta 5. Encuesta a los profesores.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.14**, el 55% de los profesores encuestados afirman que la última tarea enviada solo algunos de los estudiantes la realizaron correctamente, el 36% dice que muchos y el 9% pocos, mientras que el indicador de ninguno no presenta ningún resultado. Así que se puede concluir que la mitad de los profesores dicen que solo algunos de los estudiantes que realizan la tarea, la realizan de manera correcta, existiendo un alto índice de estudiantes que no realizan las tareas como deben, lo cual genera inconvenientes académicos.

Pregunta 6: ¿Cree usted que una aplicación que recuerde a los estudiantes las tareas a realizar, y además permita entregarlas por ese medio ayudaría a que los estudiantes cumplan eficientemente?

Siempre	Casi siempre	A veces	Casi nunca	Nunca	Total
3	6	2	0	0	11

Tabla 2.16 Tabulación pregunta 6. Encuesta a los profesores.

Elaborado por: El investigador.

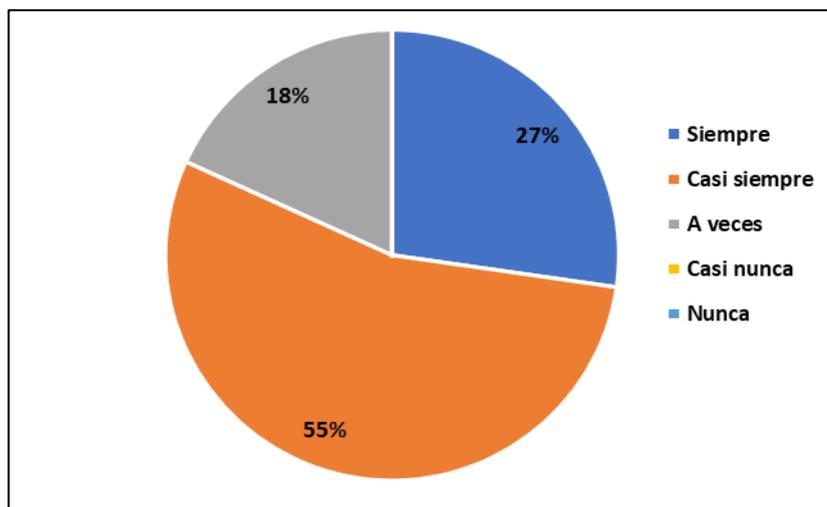


Figura 2.15 Pregunta 6. Encuesta a los profesores.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.15**, el 55% de los profesores encuestados dicen que siempre el uso de una aplicación ayudaría al cumplimiento de tareas escolares, el 27% dicen que siempre, el 18% a veces, por otro lado, los indicadores de casi nunca y nunca no presentan resultados. Se puede concluir que los profesores están a favor del uso de la tecnología, como una herramienta que ayude a los estudiantes a presentar las tareas escolares a tiempo y de manera correcta.

Pregunta 7: Que características debería tener el aplicativo para que permita el recordatorio y la entrega de tareas

Correo electrónico	Aplicación móvil	Notificaciones a los padres	Página web	Total
4	5	1	1	11

Tabla 2.17 Tabulación pregunta 7. Encuesta a los profesores.

Elaborado por: El investigador.

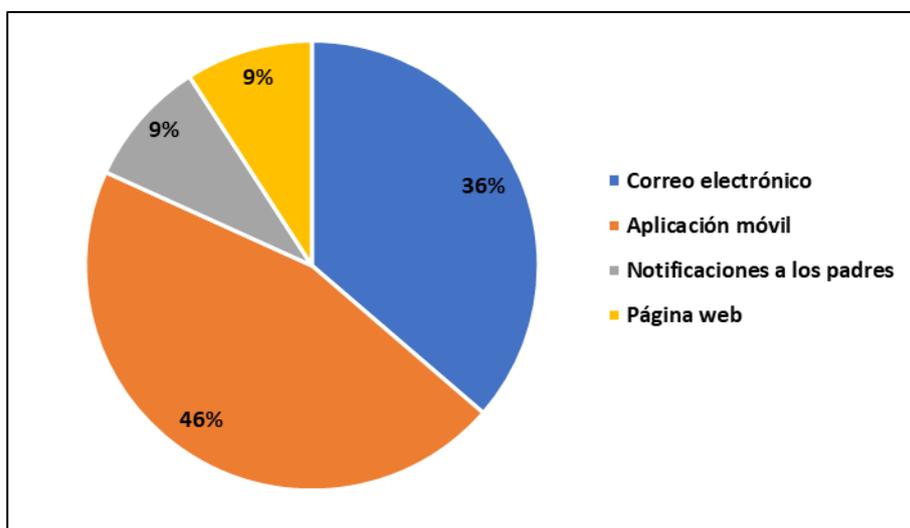


Figura 2.16 Pregunta 7. Encuesta a los profesores.

Elaborado por: El investigador.

Análisis e interpretación de los resultados:

De acuerdo con los datos obtenidos mediante la encuesta aplicada y evidenciados en la **Figura 2.16**, el 46% de los profesores encuestados dicen que una de las principales características de una herramienta tecnológica para la gestión de tareas escolares sería una aplicación móvil, el 36% dicen que lo mejor sería por correo electrónico, el 9% dice que lo mejor sería una herramienta que notifique a los padres y el 9% dicen que lo mejor sería una página web. Mediante los datos obtenidos, se puede afirmar que la mejor opción tecnológica, según el punto de vista de los profesores encuestados, sería una aplicación móvil ya que los estudiantes siempre están usando el celular y esto ayudaría a el cumplimiento de las tareas.

2.2.4 Procesamiento y análisis de datos

De acuerdo con los datos obtenidos en las encuestas aplicadas tanto a estudiantes, como a profesores de la Unidad Educativa Emanuel, se puede determinar lo siguiente:

- El principal problema a la hora de entregar tareas académicas es que los estudiantes no recuerdan todas las tareas que deben realizar, además, un gran porcentaje, de los alumnos que, si entregan las tareas, no las realizan correctamente, debido a que las instrucciones no son claras o simplemente el alumno no las recuerda.
- Solo un pequeño porcentaje de alumnos lleva un registro de todas las tareas que debe cumplir, siendo esta una causa del alto índice de incumplimiento.
- Las tareas académicas tienen un aspecto clave en la educación, ya que a través de ellas se ayuda a reforzar los conocimientos adquiridos en clases y también son claves a la hora de rendir evaluaciones, es por eso la importancia de que estas sean cumplidas de manera correcta.
- Existe un alto índice de aceptación, tanto de estudiantes, como de maestros, cuando se habla de una herramienta tecnológica que ayude a la gestión de tareas escolares.

- Una aplicación móvil sería la solución más factible a la hora de gestionar tareas, debido a que los estudiantes la mayor parte del tiempo pasan en sus celulares, y sería más fácil notificar sobre las tareas pendientes.

Los resultados que se obtuvieron de las encuestas aplicadas a estudiantes y profesores de la Unidad Educativa Emanuel son claves a la hora del desarrollo del proyecto, porque permiten establecer los problemas que tiene el usuario y las posibles soluciones.

CAPÍTULO III.- RESULTADOS Y DISCUSIÓN

3.1 Análisis y discusión

3.1.1 Proceso para el envío y recibo de tareas escolares

Tanto estudiantes como docentes, de la Unidad Educativa Emanuel, son parte del mismo proceso para el envío y recibo de tareas escolares.

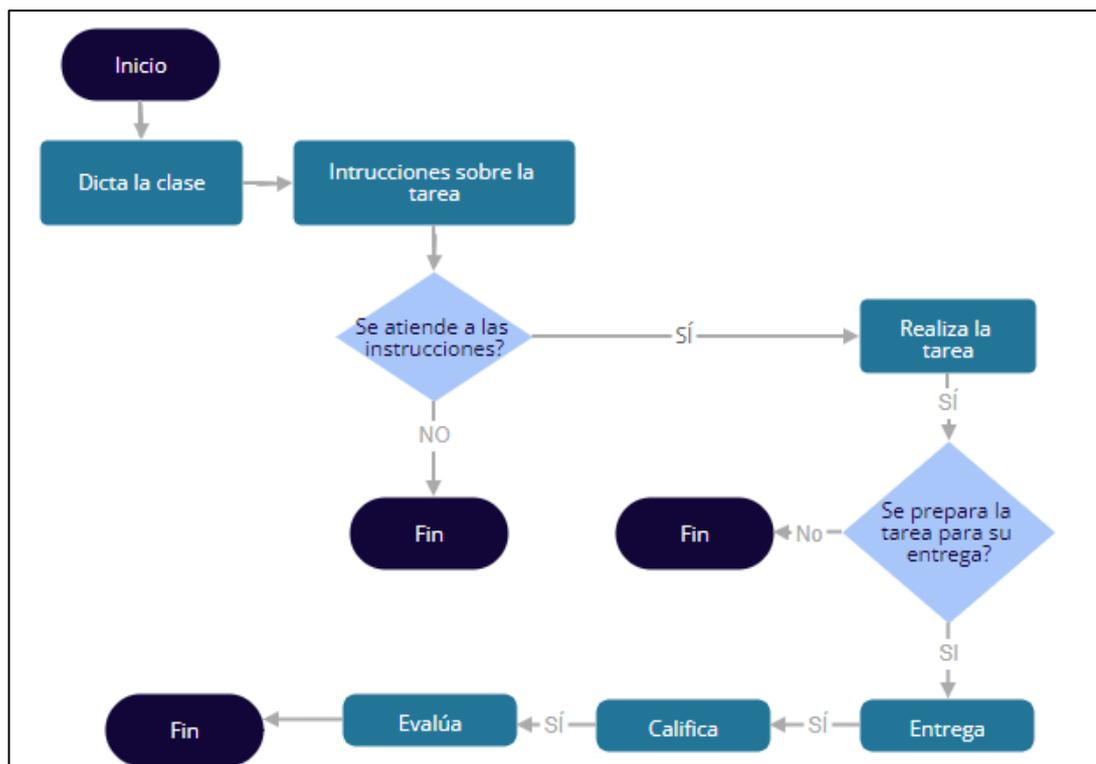


Figura 3.1 Proceso de envío y recibo de tareas escolares

Elaborado por: El investigador

En la **Figura 3.1**, se puede evidenciar el proceso actual para la gestión de tareas escolares, como primer paso, el profesor dicta la clase, sobre el tema que haya preparado.

Después al terminar la clase o el tema, el profesor dará las instrucciones sobre la tarea que será asignada a los estudiantes, dicha tarea sirve como un refuerzo a los conocimientos adquiridos en el aula de clase, ayudando así a la adquisición de nuevos conocimientos por parte de los estudiantes.

Cuando el profesor dicta la clase se ha podido evidenciar los principales inconvenientes que afectan a este proceso los cuales son: los estudiantes no prestan atención a las instrucciones sobre la tarea, algunos si ponen atención pero no las copian y al momento de querer realizar la tarea no las recuerdan, lo cual provoca que no sepan que actividades hay que realizar y la tarea no sea cumplida lo cual se ve reflejado a la hora de la evaluación ya que el estudiante no está en capacidad de resolver los problemas correctamente porque no hubo un refuerzo en la casa sobre el tema impartido en clase.

Si el estudiante si anotó las instrucciones sobre la tarea a realizar o si las recuerda, procederá a realizar la tarea , logrando así una mayor probabilidad de comprender y aprender el tema impartido en clase.

Después se puede evidenciar otro de los grandes problemas que afectan a este proceso, el estudiante sí realiza la tarea, pero se olvida de llevarla consigo para poder presentarla, esto se da debido a varias razones, como que el estudiante no revisa el horario para el día siguiente, no recuerda la fecha de presentación de la tarea o porque tiene muchas más tareas y es más probable que alguna de ellas no sea presentada, afectando así a sus calificaciones ya que el profesor dirá que la tarea no fue presentada aunque el estudiante si la haya realizado.

Si el estudiante si preparó la tarea para ser presentada, entonces el proceso sigue sin inconvenientes, después de que el estudiante presenta la tarea esta puede ser revisada y calificada según los parámetros que el profesor considere como importantes para asignar una nota a la tarea realizada.

Como último paso de este proceso, se procede a evaluar el conocimiento adquirido en la clase y que fue reforzado con la tarea asignada sobre ese tema en particular, si este proceso fue llevado correctamente se puede decir que el proceso de enseñanza aprendizaje tiene un mayor rango de éxito. Si el proceso falla existe una mayor probabilidad que el estudiante tenga un bajo nivel de aprendizaje, problemas en cursos superiores y bajas calificaciones.

3.1.2 Mejoras al proceso de envío y recibo de tareas escolares

El proceso de entrega y recibo de tareas que anteriormente existía presentaba muchos problemas, es por lo que es necesario realizar una mejora de procesos buscando disminuir los problemas ya evidenciados.

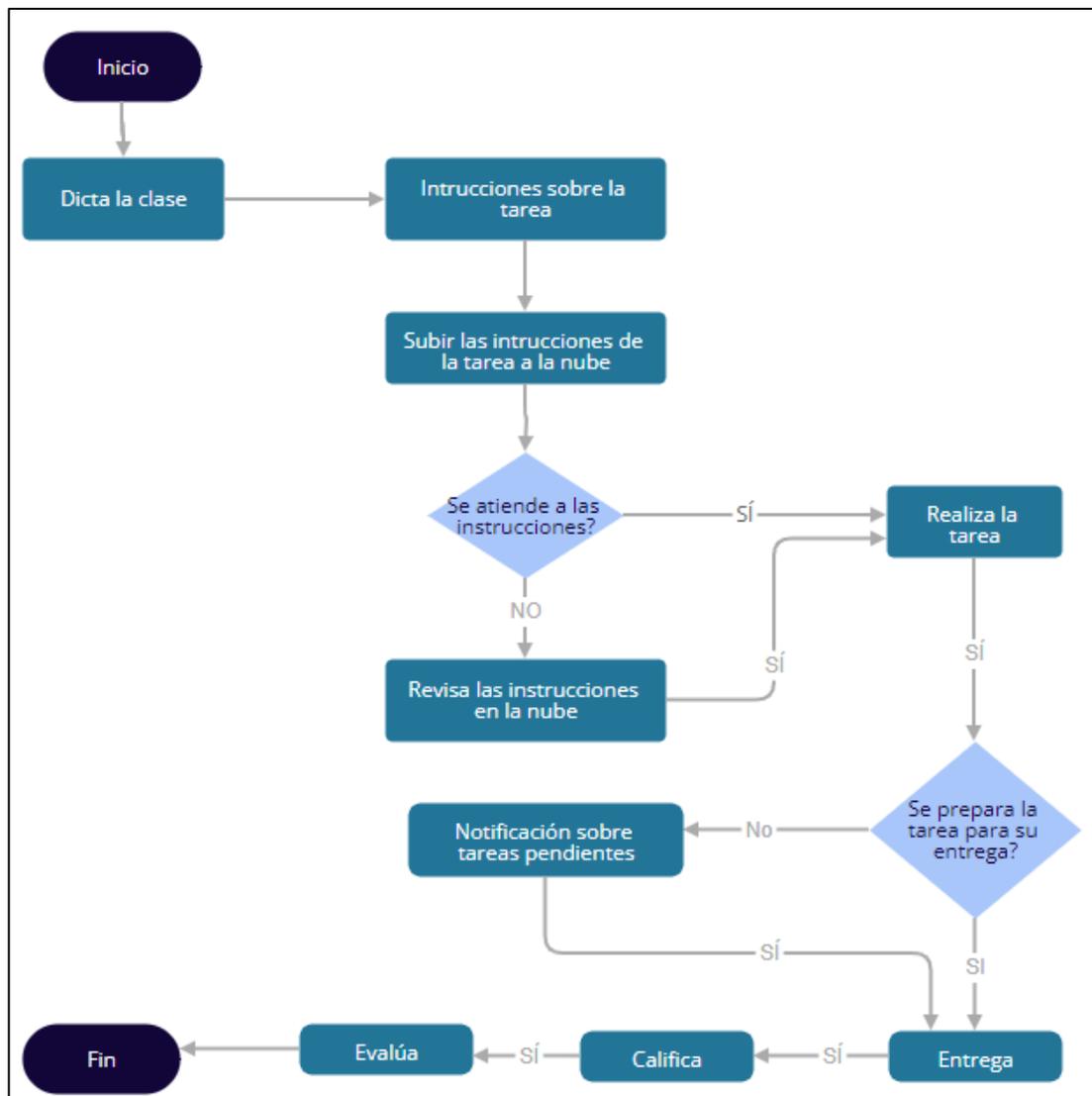


Figura 3.2 Mejoras en el proceso de envío y recibo de tareas escolares.

Elaborado por: El investigador.

Como se puede observar en la **Figura 3.2**, se realizaron cambios al proceso que existía para el envío y recibo de tareas escolares, poniendo énfasis en dos aspectos claves que afectan gravemente a este proceso.

El primer punto clave, está en el momento en el cual el profesor dicta las instrucciones sobre la tarea, algunos estudiantes no prestan atención y no anotan las instrucciones por ende al momento de querer realizar la tarea no pueden hacerla porque no saben cómo, es por lo que el proyecto busca una manera alternativa en que los estudiantes siempre tengan disponible las instrucciones sobre la tarea en la nube y poder acceder a ellas cuando quieran, aunque no las hayan anotado. El profesor deberá subir las instrucciones de que realizar, fecha de entrega y como se debe presentar la tarea.

El segundo aspecto clave que afecta a este proceso se da a la hora cuando el estudiante ya realizó la tarea, pero, debido a varias circunstancias, no recordó llevarla consigo y no pudo presentarla, el proyecto busca solucionar este problema dando al estudiante un aviso sobre todas las tareas que debe presentar con suficiente antelación para que pueda preparar la tarea y presentarla.

3.1.3 Almacenamiento local contra almacenamiento en la nube

El almacenamiento local es un tipo de almacenamiento que permite guardar la información en dispositivos físicos, a los cuales el usuario tiene acceso y los puede manipular. Por otra parte, el almacenamiento en la nube permite almacenar los archivos en un lugar externo y que es manejado por un tercero.

Criterio	Almacenamiento local	Almacenamiento en la nube
Accesibilidad	No necesita una conexión a internet	Necesita una conexión a internet
Almacenamiento	Limitado por la capacidad del dispositivo físico	Ilimitado, se puede ampliar de acuerdo con la necesidad
Velocidad de acceso	Rápida porque se accede directamente al dispositivo	Depende de la velocidad del internet disponible
Seguridad	Alta	Alta
Disponibilidad	Baja, es necesario tener el dispositivo físico	Alta, se puede acceder desde cualquier parte
Costo	Elevado.	Medio

Recuperación	Si se pierde o se daña el dispositivo físico es imposible recuperar los datos	Los datos no se pueden perder ya que los guarda un tercero
Compartir	Difícil de compartir la información con otras personas	Facilidad de compartir la información con otras personas

Tabla 3.1 Comparación entre almacenamiento local y en la nube

Elaborado por: El investigador.

Al analizar la información presente en la **Tabla 3.1**, sobre el análisis comparativo entre almacenamiento local y almacenamiento en la nube se puede concluir que el almacenamiento en la nube tiene características que satisfacen de mejor manera las necesidades del proyecto debido a que es necesario ofrecer una manera en la cual los estudiantes puedan acceder a las tareas pendientes en cualquier lugar en el que se encuentren y muchas veces el almacenamiento local de un dispositivo móvil es muy limitado y al acumular una gran cantidad de tareas el almacenamiento local queda descartado.

3.1.4 Opciones de almacenamiento en la nube

El almacenamiento en la nube permite guardar información de manera flexible, segura y garantizando una disponibilidad a los datos alta, dentro de este mundo existen una gran variedad de opciones, entre las que podemos destacar OneDrive un servicio ofrecido por Microsoft, Dropbox una de las opciones más populares y Cloud Firestore el servicio ofrecido por Firebase.

Criterio	OneDrive	Dropbox	Cloud Firestore
Especialización	Almacenamiento de archivos	Almacenamiento de fotos	Diseñado para el desarrollo de aplicaciones
Almacenamiento gratuito	5GB	2GB	10GB
Seguridad	Alta	Alta	Alta

Trabajo sin conexión	Disponible	No disponible	No disponible
Costo del plan básico mensual	10\$	16\$	5\$
Copias de seguridad	Programables por el usuario	Programables por el usuario	Programables por el usuario

Tabla 3.2 Comparación entre OneDrive, Dropbox y Cloud Firestore.

Elaborado por: El investigador.

En la **Tabla 3.2**, se presenta un análisis comparativo de las principales opciones para almacenamiento en la nube teniendo en cuenta criterios como: costo, seguridad y capacidad. En base a este análisis se puede concluir que la mejor opción para el desarrollo de la propuesta es Cloud Firestore, el servicio de almacenamiento en la nube brindado por Firebase, una de las principales razones es que este servicio está especializado para el desarrollo de aplicaciones ofreciendo un almacenamiento flexible y fácil de utilizar la seguridad en los datos también es alta, además ofrece una mayor cantidad de almacenamiento gratuito y el plan básico mensual es el mejor desde un punto de vista precio – almacenamiento.

3.1.5 Cloud Firestore para el desarrollo de aplicaciones

Es un servicio de almacenamiento en la nube ofrecido por la plataforma Firebase, es una base de datos de tipo NoSQL, que permite guardar datos y sincronizarlos con aplicaciones Android, IOS y web de manera directa esto permite ahorrar código y reducir tiempos de desarrollo. Tiene características muy útiles, las cuales se describen en la **Tabla 3.3**.

Característica	Descripción
Consultas	Usa colecciones e índices, permitiendo acceder a los datos con consultas rápidas de manera sencilla.
Estructura de datos	Al almacenar los datos lo hace de manera escalable a través de jerarquías almacenando así datos con

	relaciones, permitiendo guardar de manera ordena toda la información.
No utiliza servidores	Posee sus propias reglas de seguridad para el acceso a los datos, lo cual permite ingresar a la información sin el procesamiento de líneas de código en un servidor, logrando así una conexión directa con la aplicación.
Sincronización de datos	El servicio permite sincronizar datos desde diferentes dispositivos al mismo tiempo, llevando un registro total de todos los cambios realizados para poder tener un correcto control de usuarios.
Seguridad	Basada en usuarios, es decir se pueden establecer reglas de acceso a la información basadas en los permisos que el usuario posea.

Tabla 3.3 Características de Cloud Firestore.

Elaborado por: El investigador.

Cloud Firestore posee una gran flexibilidad a la hora de guardar datos, además ofrece una gran facilidad para acceder a los datos a través de una aplicación Android, sin necesidad de un código de servidor reduciendo líneas de código y tiempo de desarrollo, lo cual hace a este servicio una herramienta indispensable a la hora de desarrollar el proyecto.

3.1.6 Metodologías de desarrollo

Una metodología de desarrollo ofrece una serie de pasos estructurados, los cuales permiten alcanzar un objetivo definido, de manera sistemática, ordenada y rápida, optimizando el uso de los recursos disponibles y estableciendo un inicio y un fin, permitiendo satisfacer las necesidades del cliente de manera eficiente en un tiempo definido y limitado.

Para el desarrollo de aplicaciones móviles existen una gran cantidad de metodologías, con diferentes características y distintas maneras de ejecución, dependiendo de las

características que tenga el proyecto existen metodologías que serán mejores para su aplicación que otras.

La metodología Scrum permite aplicar una serie de pasos ordenados y regularizados, para el trabajo en equipo, siendo este la base de esta metodología, logrando establecer y repartir tareas para obtener un producto final de manera rápida y eficiente. Esta metodología permite la entrega parcial de los avances del proyecto, adaptando y modificando el producto de acuerdo con las especificaciones del cliente.

XP es una metodología que permite el desarrollo de proyectos de manera flexible y eficiente llevando un control continuo sobre los avances del proyecto. Se basa en la comunicación en todo momento con el cliente, logrando así que los desarrolladores se adapten a los cambios continuos que el cliente plantea en el proyecto.

Kanban nació en los años 40, originalmente nació para ser aplicada en los procesos de fabricación de productos, pero con el paso de los años ha ido ganado popularidad en el campo de los programadores. Se basa en la definición de tareas de forma visual en lo que se conoce como el tablero de tareas, en donde se definen y establecen tiempos a las tareas.

Lean es una metodología innovadora la cual busca optimizar los procesos productivos de la empresa logrando optimizar al máximo los recursos, reduciendo tanto la inversión económica, el tiempo y el esfuerzo para la obtención de un resultado específico.

Test Driven Development es una metodología de desarrollo que se basa en la codificación de pruebas, desarrollo y recodificación de manera continua del código ya construido, se basa en el diseño orientado a las necesidades ya que se enfoca en el requisito concreto logrando reducir la redundancia y aumentando la productividad.

Mobile – D reúne características de otras metodologías, obteniendo lo mejor de cada una de ellas y desechando sus principales inconvenientes, permite entregas continuas y se adapta a los cambios que el cliente requiere.

Criterio	XP	Kanban	Scrum	Lean	Test Driven Development	Mobile - D
Base	Comunicación en todo momento entre el cliente y el desarrollador	Visualización de las tareas y del flujo de trabajo	Identificar procesos ineficientes y solucionarlos	Aprovechar los recursos al máximo	Diseño orientado a las necesidades	Metodologías ya conocidas utilizadas de manera estricta
Proyectos	Pequeños, medianos	Grandes	Grandes	Pequeños, medianos	Medianos	Pequeños
Trabajo cooperativo	Parejas	Grupos grandes con tareas específicas	Grupos medianos	Grupos pequeños	Grupos pequeños	Grupos pequeños
Modificaciones	Cambios posibles entre cada iteración	Los cambios se los realiza entre los plazos de entrega	Cambios de manera continua	Después de cada paso se realiza una revisión y los cambios necesarios	Proceso de modificación limitada	Se permiten cambios entre días dedicados al desarrollo
Tiempo de desarrollo	Largos	Largos	Cortos	Largos	Largos	Cortos
Entregables	Al final de cada iteración	Al final de cada paso	Revisión continua de entregables	Al final de cada paso	Pruebas de código continuas	Entrega continua
Documentación	Historias de usuarios Tarjetas CRC	Bitácoras	Definiciones de hecho y listo	Informes de seguimiento	Pruebas de debugging	Storycards

Tabla 3.4 *Comparación de las metodologías de desarrollo.*

Elaborado por: El investigador.

Después de analizar la información presentada en la **Tabla 3.4** se puede concluir que la mejor metodología que se adapta al proyecto es Mobile – D, debido principalmente a que esta metodología está enfocada al desarrollo de proyectos pequeños y en un tiempo de desarrollo corto, características que posee el proyecto. Pero además de esto, la metodología se basa en otras metodologías ya existentes tales como XP, Crystal Methodologies, entre otras, tomando las mejores características de cada una de ellas, eliminando las principales fallas y consiguiendo una metodología con todo lo necesario para el desarrollo de aplicaciones móviles.

3.1.6.1 Metodología Mobile – D

La metodología Mobile – D establece cinco fases, las cuales son utilizadas para el desarrollo de la aplicación firebase Android para la gestión de tareas académicas.

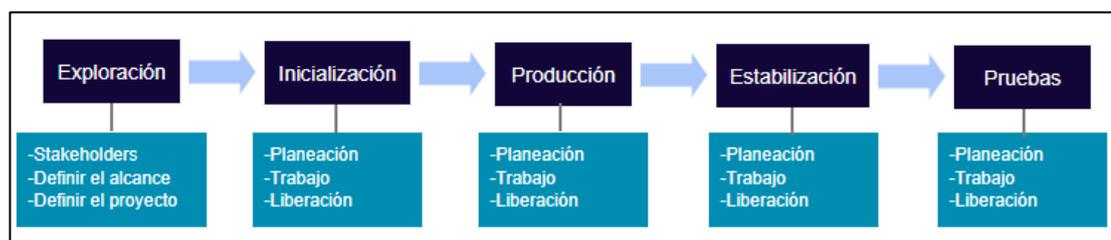


Figura 3.3 Fases de Mobile – D.

Elaborado por: El investigador.

Fase 1: Exploración

- Definir los interesados del proyecto.
- Definir el alcance de la aplicación firebase Android para la gestión de tareas académicas.
- Definir el proyecto.

Fase 2: Inicialización

- Análisis de requerimientos.
- Definir la arquitectura del software
- Determinar las funcionalidades

- Interfaces de los usuarios
- Diagramas de los casos de usos

Fase 3: Producción

- Planificación: realizar el análisis, las mejoras y la priorización de los requisitos.
- Trabajo: implementar las funcionalidades de la aplicación a través de pruebas de software.
- Liberación: realizar pruebas de funcionamiento.

Fase 4: Estabilización

- Realizar las mismas tareas que en la fase de Producción, mejorando la aplicación en caso de que sea requerido.
- Funcionalidad de la aplicación completa.
- Documentar el proyecto.

Fase 5: Pruebas

- Realizar pruebas de funcionamiento de la aplicación, con el fin de comprobar su usabilidad con los usuarios.

3.2 Desarrollo de la propuesta

El presente proyecto se desarrolló utilizando las cinco fases definidas por la metodología Mobile – D.

3.2.1 Fase 1: Exploración

La primera fase permite obtener los interesados en el proyecto, así como definir el alcance y sus funcionalidades básicas.

3.2.1.1 Definición de los stakeholders

- **Empresa:** Unidad Educativa Emanuel

- **Usuarios:** Rectora, profesores y estudiantes de la Unidad Educativa Emanuel
- **Desarrollador:** Adrian De la Cruz. Autor del proyecto.

3.2.1.2 Definición del alcance

Las funciones de la aplicación están definidas de acuerdo con los requisitos ya establecidos, a través de la información recolectada por medio de las encuestas aplicadas a los profesores y estudiantes de la Unidad Educativa Emanuel.

Los datos se almacenarán en Cloud Firestore, y se accederán a ellos a través de las funciones de Firebase, para garantizar la integridad seguridad y disponibilidad de los datos proporcionados por los usuarios.

La aplicación, establece las funcionalidades de acuerdo con los roles que desempeñan cada usuario, a fin de cumplir con el objetivo:

- **Funcionalidades para el administrador:** Iniciar sesión, Aprobar profesores, Denegar profesores, Aprobar estudiantes, Denegar estudiantes, Cerrar sesión.
- **Funcionalidades para los profesores:** Registro, Iniciar sesión, Agregar tareas, Revisar tareas pendientes, Ver tareas entregadas , Enviar recordatorio de tareas, Actualizar perfil, Cerrar sesión.
- **Funcionalidades para los estudiantes:** Registro, Iniciar sesión, Revisar las tareas pendientes, Entregar tareas, Actualizar perfil, Cerrar sesión.

Limitaciones

La aplicación firebase Android posee las siguientes limitaciones:

- Los usuarios deben poseer una conexión a internet para poder hacer uso de la aplicación.
- Los usuarios deben poseer dispositivos con versiones iguales o superiores a Android 7.0.

3.2.1.3 Establecimiento del proyecto

Para el desarrollo del proyecto se utilizaron las siguientes herramientas:

- **Android Studio**

Editor de código fuente para el desarrollo de aplicaciones Android, es el IDE de desarrollo oficial para aplicaciones Android lo cual lo hace uno de los más completos y fáciles de usar, con las mayores ventajas sobre otros IDEs similares.

- **Java**

Lenguaje de programación, que permite la integración de Firebase con Android.

- **Firebase Authentication**

Servicio de Firebase que ayuda a la gestión de usuarios de manera flexible y rápida, además ofrece una seguridad alta que permite mantener la integridad de los datos brindados por los usuarios.

- **Cloud Firestore**

Servicio de base de datos en la nube ofrecido por Firebase, con una gran flexibilidad y seguridad, lo cual garantiza el acceso y la integridad de los datos.

- **Firebase**

Plataforma creada por Google que ofrece una gran cantidad de servicios para ayudar a la construcción de aplicaciones móviles ayudando a reducir código y tiempo de desarrollo.

- **Awesome Validation**

Librería para la validación de campos de manera automática, ayudando a reducir código.

- **BlueStacks**

Emulador Android, que permite ejecutar las aplicaciones creadas en Android Studio directamente.

- **Balsamiq**

Herramienta para el diseño de prototipos de interfaces gráficas, que ofrece una gran cantidad de elementos fáciles de usar lo cual ayuda a diseñar de manera rápida y eficiente.

3.2.2 Fase 2: Inicialización

La segunda fase de la metodología permite preparar e identificar los recursos necesarios para llevar a cabo el proyecto, además de esta fase se obtiene el diseño de la aplicación firebase Android.

3.2.2.1 Configuración del ambiente de desarrollo

Por medio de la configuración se obtuvo el ambiente requerido para el trabajo con los recursos necesarios por medio de la instalación y configuración de herramientas que ayuden al desarrollo del proyecto.

- Tipo de aplicación: Nativa
- Librería de validación: Awesome Validation
- Librería de verificación de conexión a red: Merlin
- Servicios de autenticación: Firebase Authentication
- Sistema Gestor de Base de Datos: Cloud Firestore

Preparación del ambiente

Se instalaron y configuraron las herramientas necesarias, de acuerdo con los requerimientos para el desarrollo del proyecto:

- Awesome Validation
 - Android Studio
 - Lenguaje de programación java
- Merlin
 - Android Studio
 - Lenguaje de programación java

- Firebase Authentication
 - Android Studio
 - Lenguaje de programación Java
 - Configuración en Firebase
- Cloud Firestore
 - Android Studio
 - Lenguaje de programación Java
 - Configuración en Firebase
- BlueStacks
- Balsamiq

3.2.2.2 Planificación de fases

Las fases utilizadas por la metodología Mobile – D, se planificaron de acuerdo con las necesidades del proyecto:

Fase	Iteración	Descripción
Exploración	Iteración 0	- Definir los stakeholders. - Definir el alcance. - Establecimiento del proyecto.
Iniciación	Iteración 1	- Configurar el ambiente de desarrollo.
	Iteración 2	- Diseñar la base de datos. - Analizar los requisitos iniciales.
Producción	Iteración 3	- Analizar e implementar la funcionalidad para el Inicio de sesión del administrador. - Crear y actualizar el storycard

		- Pruebas
	Iteración 4	- Analizar e implementar la funcionalidad para Aprobar o Rechazar profesores para el administrador. - Crear y actualizar el storycard - Pruebas
	Iteración 5	- Analizar e implementar la funcionalidad para Aprobar o Rechazar estudiantes para el administrador. - Crear y actualizar el storycard - Pruebas
	Iteración 6	- Analizar e implementar la funcionalidad para Cerrar sesión del administrador. - Crear y actualizar el storycard - Pruebas
	Iteración 7	- Analizar e implementar la funcionalidad para el Registro de estudiantes y profesores. - Crear y actualizar el storycard - Pruebas
	Iteración 8	- Analizar e implementar la funcionalidad para Restablecer

		<p>la contraseña de estudiantes y profesores.</p> <ul style="list-style-type: none"> - Crear y actualizar el storycard -Pruebas
	Iteración 9	<ul style="list-style-type: none"> - Analizar e implementar la funcionalidad para el Inicio de sesión de estudiantes y profesores. - Crear y actualizar el storycard - Pruebas
	Iteración 10	<ul style="list-style-type: none"> - Analizar e implementar la funcionalidad para Agregar tareas para los profesores. - Crear y actualizar el storycard - Pruebas
	Iteración 11	<ul style="list-style-type: none"> - Analizar e implementar la funcionalidad para Ver tareas pendientes para los profesores. - Crear y actualizar el storycard - Pruebas
	Iteración 12	<ul style="list-style-type: none"> - Analizar e implementar la funcionalidad para Ver los archivos de las tareas entregadas para los profesores. - Crear y actualizar el storycard.

		- Pruebas.
	Iteración 13	- Analizar e implementar la funcionalidad para Recordar la tarea por correo para los profesores. - Crear y actualizar el storycard. - Pruebas.
	Iteración 14	- Analizar e implementar la funcionalidad para Ver y Actualizar el perfil de los profesores. - Crear y actualizar el storycard - Pruebas
	Iteración 15	- Analizar e implementar la funcionalidad para Cerrar sesión de los profesores. - Crear y actualizar el storycard - Pruebas
	Iteración 16	- Analizar e implementar la funcionalidad para Ver tareas pendientes para los estudiantes. - Crear y actualizar el storycard - Pruebas

	Iteración 17	<ul style="list-style-type: none"> - Analizar e implementar la funcionalidad para Entregar tareas para los estudiantes. - Crear y actualizar el storycard - Pruebas
	Iteración 18	<ul style="list-style-type: none"> - Analizar e implementar la funcionalidad para Ver y Actualizar el perfil de los estudiantes. - Crear y actualizar el storycard - Pruebas
	Iteración 19	<ul style="list-style-type: none"> - Analizar e implementar la funcionalidad para Cerrar sesión de los estudiantes. - Crear y actualizar el storycard - Pruebas
	Iteración 20	<ul style="list-style-type: none"> - Analizar e implementar la funcionalidad para las notificaciones de los estudiantes. - Crear y actualizar el storycard - Pruebas
Estabilización	Iteración 21	<ul style="list-style-type: none"> - Ajustes al diseño de la base de datos

		- Ajustes al diseño de las interfaces
Pruebas	Iteración 22	- Pruebas de la aplicación firebase Android para la gestión de tareas académicas de acuerdo con los requerimientos de funcionalidad establecidos.

Tabla 3.5 Planificación de las fases de Mobile – D.

Elaborado por: El investigador.

3.2.2.3 Diseño de la base de datos

La base de datos es una parte fundamental en el desarrollo de la aplicación, es por lo que se definieron los modelos tanto conceptual como relacional de la misma, dichos modelos permiten definir de manera clara como está diseñada la base de datos, sus tablas y relaciones.

Modelo conceptual

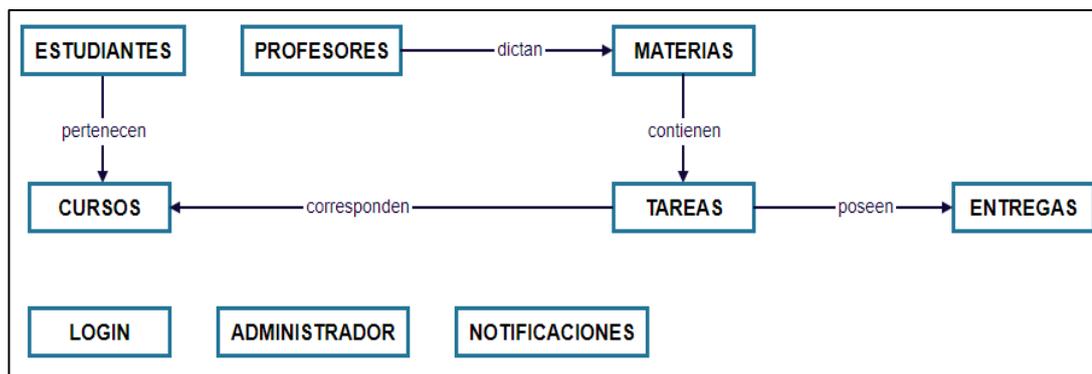


Figura 3.4 Modelo conceptual de la base de datos.

Elaborado por: El investigador

Modelo relacional

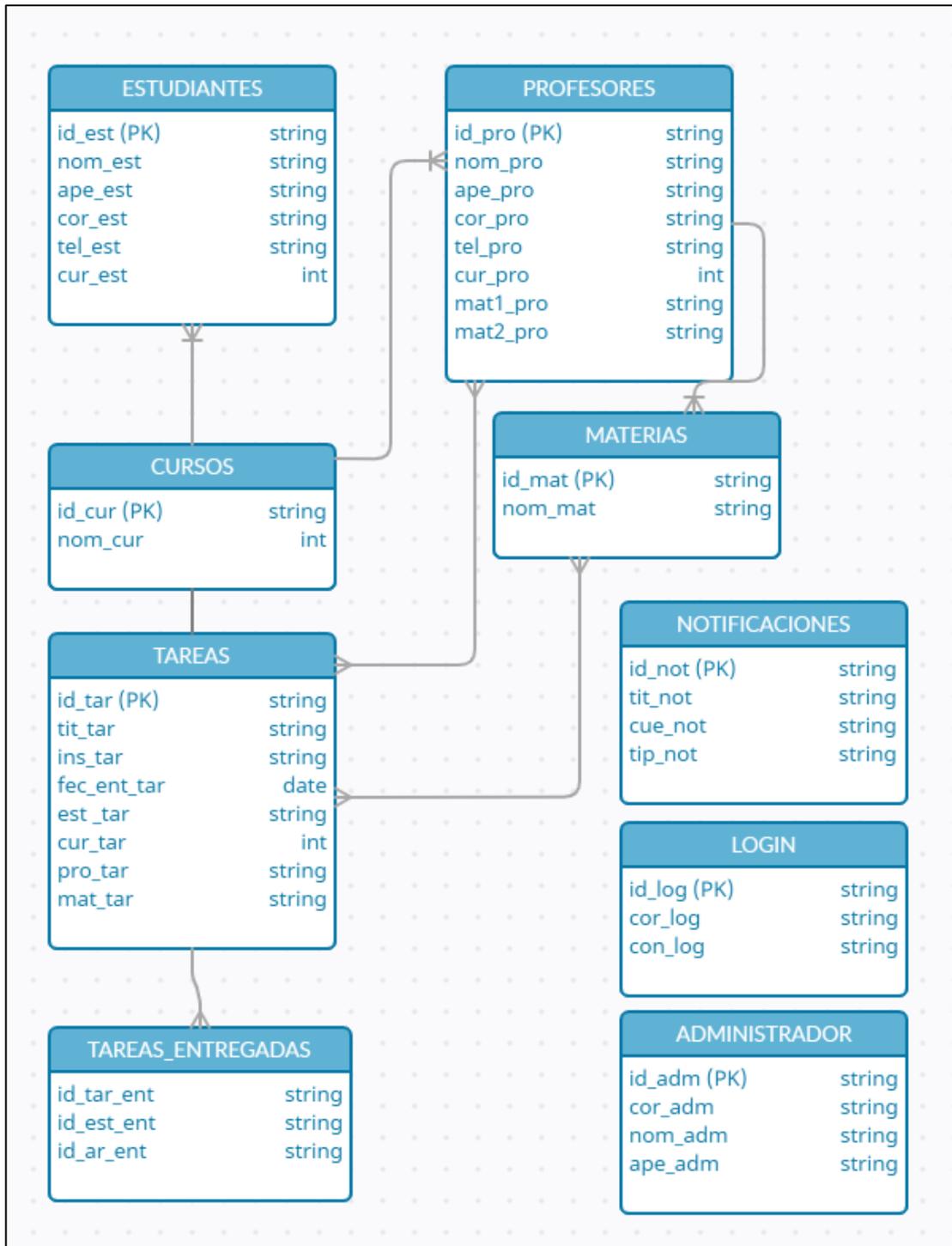


Figura 3.5 Modelo relacional de la base de datos.

Elaborado por: El investigador.

3.2.2.4 Arquitectura de la aplicación firebase Android

La arquitectura de la aplicación firebase Android propuesta, permite a los usuarios(administrador, profesores y estudiantes) acceder a los servicios ofrecidos por la aplicación para la gestión de tareas académicas. Dependiendo del rol del usuario puede acceder a las funciones habilitadas para dicho rol, la petición de las funciones requiere una conexión a internet para poder acceder a los servicios de firebase y retornar el resultado de acuerdo con la petición del usuario.

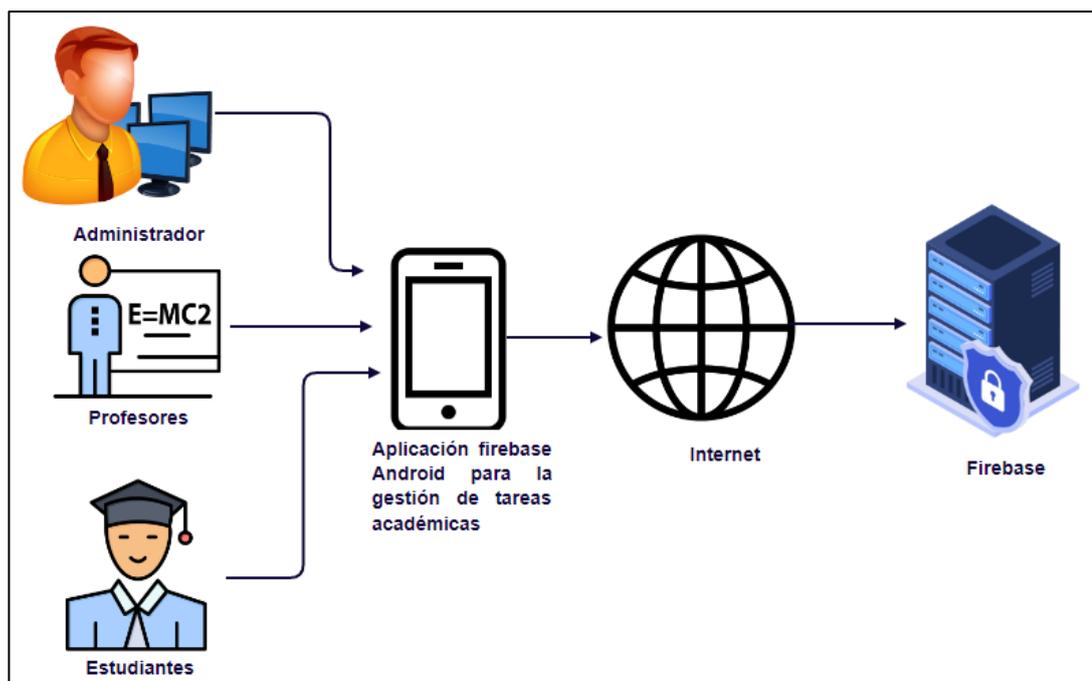


Figura 3.6 Arquitectura de la aplicación firebase Android.

Elaborado por: El investigador.

3.2.2.5 Análisis de los requisitos iniciales

Requerimientos de acuerdo con los roles designados para las funciones de la aplicación firebase Android.

Administrador:

- El administrador podrá iniciar sesión con el correo y contraseña que han sido asignados.

- El administrador podrá aprobar los usuarios que quieran registrarse como profesores.
- El administrador podrá rechazar los usuarios que quieran registrarse como profesores.
- El administrador podrá aprobar los usuarios que quieran registrarse como estudiantes.
- El administrador podrá rechazar los usuarios que quieran registrarse como estudiantes.

Profesores:

- El profesor podrá registrar sus datos de inicio de sesión y personales para acceder al servicio.
- El profesor podrá cambiar su contraseña a través de un enlace de restablecimiento enviado al correo proporcionado.
- El profesor podrá iniciar sesión con el correo y contraseña que haya establecido.
- El profesor podrá agregar una tarea, asignado título, instrucciones, fecha de entrega, curso y materia.
- El profesor podrá revisar las tareas que estén pendientes de entrega.
- El profesor podrá revisar los archivos enviados.
- El profesor podrá enviar un correo a los alumnos recordando las tareas pendientes de entrega.
- El profesor podrá revisar la información de su perfil.
- El profesor podrá actualizar la información de su perfil.
- El profesor podrá cerrar sesión.

Estudiantes

- El estudiante podrá registrar sus datos de inicio de sesión y personales para acceder al servicio.
- El estudiante podrá cambiar su contraseña a través de un enlace de restablecimiento enviado al correo proporcionado.

- El estudiante podrá iniciar sesión con el correo y contraseña que haya establecido.
- El estudiante podrá visualizar el título, instrucciones, materia y fecha de entrega de las tareas pendientes.
- El estudiante podrá subir archivos para entregar las tareas.
- El estudiante podrá revisar la información de su perfil.
- El estudiante podrá actualizar la información de su perfil.
- El estudiante recibirá notificaciones de acuerdo con las funciones que sean requeridas.
- El estudiante podrá cerrar sesión.

3.2.2.6 Funciones establecidas

De acuerdo con los requerimientos iniciales establecidos, además de la información recolectada a través de las encuestas aplicadas a estudiantes y profesores de la Unidad Educativa Emanuel, se pudo determinar las funciones necesarias que debe tener la aplicación firebase Android.

Funciones para que el administrador pueda realizar a través del uso de la aplicación firebase Android para la gestión de tareas académicas.

Función	Opción	Actividad
Inicio de sesión	Correo	El administrador debe escribir su correo y contraseña.
	Contraseña	
Aprobar o Rechazar cuentas de profesores	Lista	El administrador debe dar clic a los botones dentro de la lista.
Observación		
El administrador podrá aceptar o rechazar las cuentas que se quieran registrar con el rol de “Profesor”, al dar clic en una de las opciones mostradas para cada una de las cuentas pendientes de aceptación.		
Aprobar o Rechazar cuentas de estudiantes	Lista	El administrador debe dar clic a los botones dentro de la lista.

Observación		
El administrador podrá aceptar o rechazar las cuentas que se quieran registrar con el rol de “Estudiante”, al dar clic en una de las opciones mostradas para cada una de las cuentas pendientes de aceptación.		
Cerrar sesión	Menú	El administrador debe seleccionar la opción en el menú.

Tabla 3.6 Funciones para el administrador.

Elaborado por: El investigador.

Funciones para que el profesor pueda realizar a través del uso de la aplicación firebase Android para la gestión de tareas académicas.

Función	Opción	Actividad
Registrar datos	Correo	El profesor debe registrar su correo y la contraseña respectiva para el inicio de sesión, sus nombres, apellidos y teléfono como datos personales y definir su rol.
	Contraseña	
	Nombres	
	Apellidos	
	Teléfono	
	Rol	
Restablecer contraseña	Correo	El profesor debe escribir el correo electrónico.
Observación		
El profesor recibirá un enlace al correo electrónico proporcionado en el cual podrá restablecer la contraseña y poder iniciar sesión normalmente con la nueva contraseña establecida.		
Inicio de sesión	Correo	El profesor debe escribir su correo y contraseña.
	Contraseña	
Agregar tareas	Título	El profesor debe escribir el título y las instrucciones, también debe seleccionar el curso y la materia
	Instrucciones	
	Curso	

	Materia	además debe establecer una fecha de entrega.
	Fecha de entrega	
Observación		
El profesor podrá agregar nuevas tareas escribiendo el título, estableciendo las instrucciones y la fecha de entrega. Debido a que el profesor dicta clases en más de un curso y más de una materia deberá establecer el curso al cual se le asignará la tarea y la materia de esta.		
Ver tareas	Menú	El profesor debe seleccionar la opción en el menú.
Observación		
El profesor podrá escoger dentro del menú la opción de ver tareas, en dicha opción podrá ver todas las tareas que ha creado y que aun estén pendientes de entrega, así como los detalles referentes a dicha tarea.		
Revisar entregas	Menú	El profesor debe seleccionar la opción en el menú.
Observación		
El profesor podrá revisar los archivos enviados por los estudiantes de acuerdo con cada una de las tareas.		
Recordar tareas	Menú	El profesor debe seleccionar la opción en el menú.
Observación		
El profesor podrá seleccionar la opción dentro del menú y enviar un correo electrónico a los estudiantes con la información de la tarea para recordar su entrega.		
Ver perfil	Menú	El profesor debe seleccionar la opción en el menú.
Observación		
El profesor podrá revisar todos sus datos personales asignados en el momento de registro de cuenta.		

Actualizar perfil	Nombres	El profesor debe actualizar sus nombres, apellidos y teléfono si fuere necesario, además el curso al cual pertenece y las materias que dicta.
	Apellidos	
	Teléfono	
	Curso	
	Materia 1	
	Materia 2	
Observación		
El profesor podrá revisar su información personal y en el caso que fuere necesario podrá actualizar sus datos personales, además puede actualizar su curso si fuere el caso que dicta clases en un solo curso, por otra parte, también podrá actualizar la materia que dicta, en el caso que dicte dos materias también podrá actualizar la materia adicional.		
Cerrar sesión	Menú	El profesor debe seleccionar la opción en el menú.
Observación		
El profesor podrá cerrar sesión con la opción del menú y regresar a la página de inicio de sesión.		

Tabla 3.7 Funciones para los profesores.

Elaborado por: El investigador.

Funciones para que el estudiante pueda realizar a través del uso de la aplicación firebase Android para la gestión de tareas académicas.

Función	Opción	Actividad
Registrar datos	Correo	El estudiante debe registrar su correo y la contraseña respectiva para el inicio de sesión, sus nombres, apellidos y teléfono como datos personales y definir su rol.
	Contraseña	
	Nombres	
	Apellidos	
	Teléfono	
	Rol	

Restablecer contraseña	Correo	El estudiante debe escribir el correo electrónico.
Observación		
El estudiante recibirá un enlace al correo electrónico proporcionado en el cual podrá restablecer la contraseña y poder iniciar sesión normalmente con la nueva contraseña establecida.		
Inicio de sesión	Correo	El estudiante debe escribir su correo y contraseña.
	Contraseña	
Ver tareas	Menú	El estudiante debe seleccionar la opción en el menú.
Observación		
El estudiante podrá ver todas las tareas que tenga pendientes de entregar, así como detalles importantes de las mismas como fecha de entrega, instrucciones, materia lo cual ayudará a su cumplimiento de manera correcta.		
Entregar tareas	Menú	El estudiante debe seleccionar la opción en el menú.
Observación		
El estudiante podrá seleccionar la opción dentro del menú y escoger un archivo para entregarlo como parte del cumplimiento de la tarea.		
Ver perfil	Menú	El estudiante debe seleccionar la opción en el menú.
El estudiante podrá revisar todos sus datos personales asignados en el momento de registro de cuenta.		
Actualizar perfil	Nombres	El estudiante debe actualizar sus nombres, apellidos y teléfono si fuere necesario además el curso al cual pertenece.
	Apellidos	
	Teléfono	
	Curso	
Observación		

El estudiante podrá revisar su información personal y en el caso que fuere necesario podrá actualizar sus datos personales, además puede actualizar su curso, este dato es muy importante ya que las tareas se asignan por cursos, por lo tanto, el estudiante solo puede ver las tareas del curso al cual pertenece.		
Cerrar sesión	Menú	El estudiante debe seleccionar la opción en el menú.
Observación		
El estudiante podrá cerrar sesión con la opción del menú y regresar a la página de inicio de sesión.		
Notificaciones		El estudiante recibirá notificaciones para recordar revisar las tareas pendientes.
Observación		
El estudiante recibirá en su celular una notificación a una hora específica del día para que recuerde revisar las tareas que tiene pendientes de entrega para así realizarlas y entregarlas.		

Tabla 3.8 Funciones para los estudiantes.

Elaborado por: El investigador.

3.2.2.7 Diagramas de caso de uso

Los diagramas de caso de uso se establecen para los tres roles que existen dentro de la aplicación firebase Android (Administrador, Profesores y Estudiantes), dichos diagramas indican la relación e interacción existente entre las funciones de la aplicación y los roles.

Diagrama de caso de uso para el administrador

La **Figura 3.7** indica las funciones que están habilitadas para el administrador a través del uso de la aplicación firebase Android.

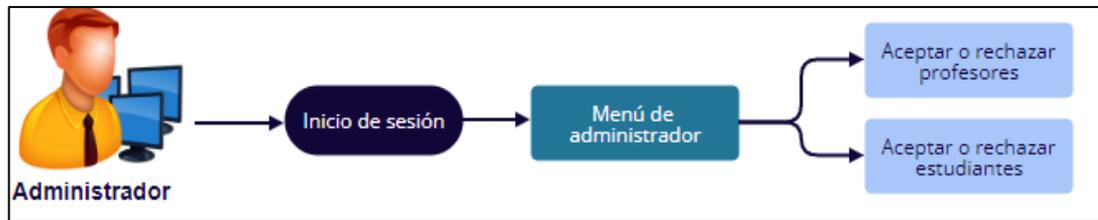


Figura 3.7 Caso de uso para el administrador.

Elaborado por: El investigador.

Diagrama de caso de uso para profesores

La **Figura 3.8** indica las funciones que están habilitadas para los profesores a través del uso de la aplicación firebase Android.

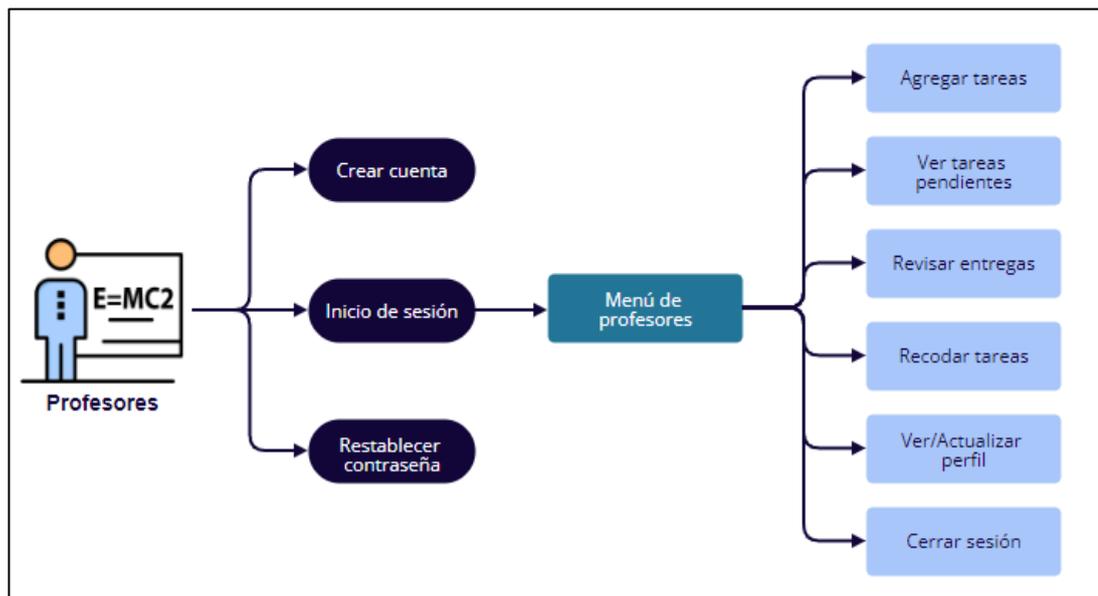


Figura 3.8 Caso de uso para profesores.

Elaborado por: El investigador.

Diagrama de caso de uso para estudiantes

La **Figura 3.9** indica las funciones que están habilitadas para los estudiantes a través del uso de la aplicación firebase Android.

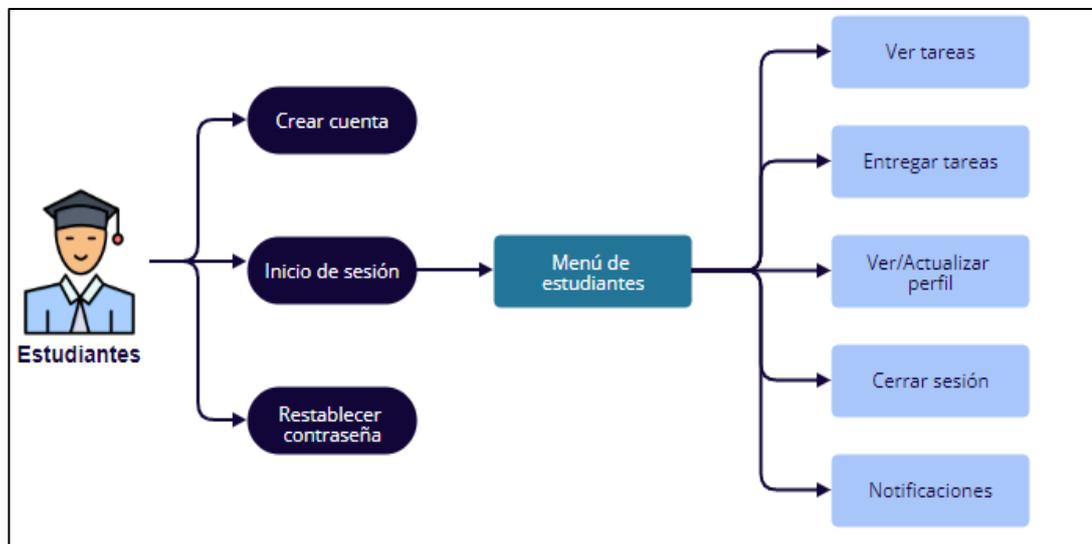


Figura 3.9 Caso de uso para estudiantes.

Elaborado por: El investigador.

3.2.2.8 Prototipo de la aplicación firebase Android

Para el presente proyecto se usó la herramienta Balsamiq para la creación de los prototipos de las interfaces gráficas tanto para las funciones de los profesores como de los estudiantes, con el fin de tener una vista previa del funcionamiento de la aplicación.

Prototipo para el administrador

De acuerdo con el rol establecido, se diseñó los distintos prototipos por cada vista que el administrador visualizará al ejecutar la aplicación firebase Android.

De acuerdo con cada función que el administrador va a ejecutar se creó una vista previa con todos los elementos necesarios para que se lleve a cabo correctamente la función establecida referente a la gestión de tareas académicas.

Inicio de sesión: en la *Figura 3.10*, se visualiza los campos de para el inicio de sesión del administrador.



Figura 3.10 Inicio de sesión para el administrador.

Elaborado por: El investigador.

Menú: En la *Figura 3.11*, se visualiza el menú para profesores.



Figura 3.11 Menú para el administrador.

Elaborado por: El investigador.

Aceptar o rechazar profesores: En la *Figura 3.12*, se visualiza el diseño de la interfaz para que el administrador pueda aceptar o rechazar las cuentas que se quieran registrar para usar la aplicación con el rol de “Profesor”.



Figura 3.12 Aceptar o rechazar profesores para el administrador.

Elaborado por: El investigador.

Aceptar o rechazar estudiantes: En la *Figura 3.13*, se visualiza el diseño de la interfaz para que el administrador pueda aceptar o rechazar las cuentas que se quieran registrar para usar la aplicación con el rol de “Estudiante”.

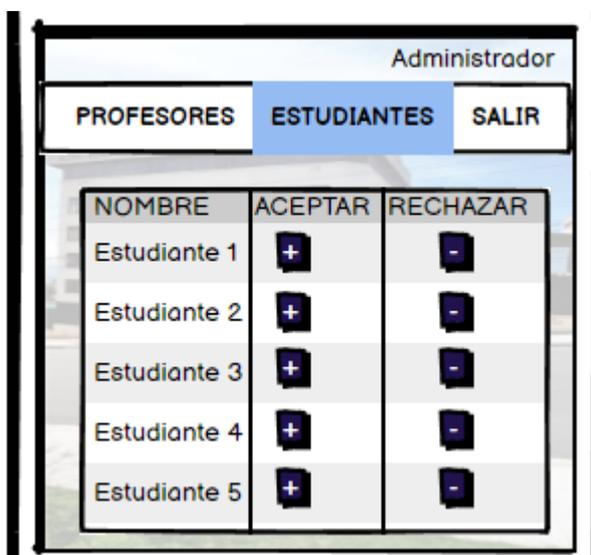


Figura 3.13 Aceptar o rechazar estudiantes para el administrador.

Elaborado por: El investigador.

Cerrar sesión: En la **Figura 3.14**, se visualiza el diseño de la interfaz para que el administrador pueda cerrar sesión y regresar al inicio de la aplicación.

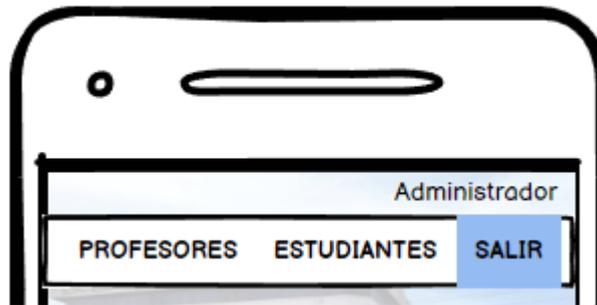


Figura 3.14 Cerrar sesión para el administrador.

Elaborado por: El investigador.

Prototipo de Inicio de sesión para estudiantes y profesores

En el diseño de Inicio de sesión, se requieren los campos de texto para el correo electrónico y la contraseña tal como se muestra en la **Figura 3.15**.



Figura 3.15 Inicio de sesión para estudiantes y profesores.

Elaborado por: El investigador.

Prototipo de Registro de cuenta para estudiantes y profesores

En el diseño de Registro de cuenta, se requieren los campos de texto para el correo electrónico, la contraseña, los nombres, apellidos y el teléfono, además de un combo box para la selección del rol, como se puede ver en la *Figura 3.16*.

The image shows a mobile application prototype for account registration. The screen is titled "REGISTRO DE CUENTA". It contains several input fields: "Correo:" (Email), "Contraseña:" (Password), "Nombres:" (First Name), "Apellidos:" (Last Name), and "Celular:" (Cellular Phone). Below these is a dropdown menu labeled "Rol:" with the text "Selecciona un rol" and a downward arrow. At the bottom of the form is a blue button with the text "REGISTRARSE".

Figura 3.16 Registro de cuenta para estudiantes y profesores.

Elaborado por: El investigador.

Prototipo de Restablecer contraseña para estudiantes y profesores

En el diseño de Restablecer contraseña, se requiere un solo campo de texto para ingresar el correo electrónico del usuario que desee restablecer su contraseña, como se puede evidenciar en la *Figura 3.17*.

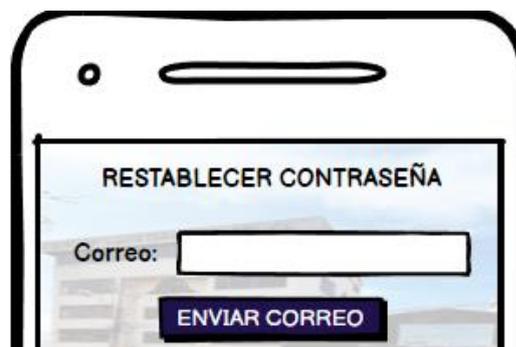
The image shows a mobile application prototype for password recovery. The screen is titled "RESTABLECER CONTRASEÑA". It features a single text input field labeled "Correo:". Below the input field is a blue button with the text "ENVIAR CORREO".

Figura 3.17 Restablecer contraseña para estudiantes y profesores.

Elaborado por: El investigador.

Prototipo para profesores

De acuerdo con el rol establecido, se diseñó los distintos prototipos por cada vista que el profesor visualizará al ejecutar la aplicación firebase Android.

De acuerdo con cada función que el profesor va a ejecutar se creó una vista previa con todos los elementos necesarios para que se lleve a cabo correctamente la función establecida referente a la gestión de tareas académicas.

Menú: En la *Figura 3.18*, se visualiza el menú para profesores.



Figura 3.18 Menú para profesores.

Elaborado por: El investigador.

Agregar tareas: En la *Figura 3.19*, se visualiza el diseño de la interfaz para que el profesor pueda agregar una nueva tarea, llenar los campos con la información requerida y necesaria para que los estudiantes puedan llevar a cabo dicha tarea, además de asignar un curso.

Bienvenido: Profesor

TAREAS PERFIL SALIR

AGREGAR TAREA

Título:

Instrucciones:

Fecha:

Curso:

Materia:

AGREGAR

Figura 3.19 Agregar tareas para el profesor.

Elaborado por: El investigador.

Ver tareas pendientes: En la *Figura 3.20*, se visualiza el diseño de la interfaz para que el profesor pueda ver las tareas que ha creado y que aún están pendientes de entrega, así como los detalles de esta.

Bienvenido: Profesor

TAREAS PERFIL SALIR

TAREAS PENDIENTES

TITULO	ENVÍO	ENTREGA	MATERIA
Tarea 1	10/06/2022	12/06/2022	Ciencias
Tarea 2	11/06/2022	12/06/2022	Sociales
Tarea 3	12/06/2022	13/06/2022	Ciencias
Tarea 4	13/06/2022	14/06/2022	Lenguaje
Tarea 5	14/06/2022	15/06/2022	Matemáticas

Figura 3.20 Ver tareas pendientes para el profesor.

Elaborado por: El investigador.

Ver tareas entregadas: En la *Figura 3.21*, se visualiza el diseño de la interfaz para que el profesor pueda ver los archivos de las tareas entregadas.

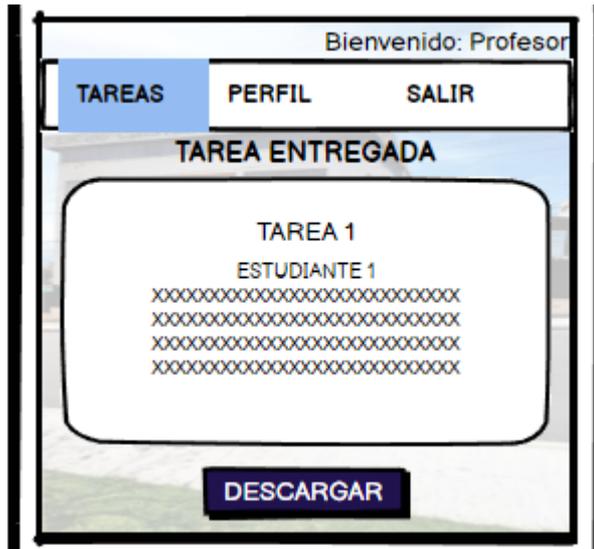


Figura 3.21 Ver tareas entregadas para el profesor.

Elaborado por: El investigador.

Recordar tareas: En la *Figura 3.22*, se visualiza el diseño de la interfaz para que el profesor pueda recordar la tarea a través de un correo electrónico.



Figura 3.22 Recordar tareas para el profesor.

Elaborado por: El investigador.

Ver o Editar perfil: En la *Figura 3.23*, se visualiza el diseño de la interfaz para que el profesor pueda ver la información de su perfil, así como editarla.



Figura 3.23 Ver o Editar perfil para el profesor.

Elaborado por: El investigador.

Cerrar sesión: En la *Figura 3.24*, se visualiza el diseño de la interfaz para que el profesor pueda cerrar sesión y regresar al inicio de la aplicación.

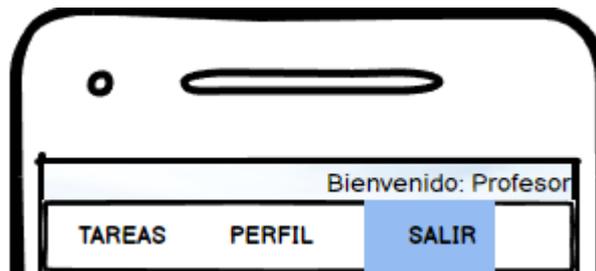


Figura 3.24 Cerrar sesión para el profesor.

Elaborado por: El investigador.

Prototipo para estudiantes

De acuerdo con el rol establecido, se diseñó los distintos prototipos por cada vista que el estudiante visualizará al ejecutar la aplicación firebase Android.

De acuerdo con cada función que el estudiante va a ejecutar se creó una vista previa con todos los elementos necesarios para que se lleve a cabo correctamente la función establecida referente a la gestión de tareas académicas.

Menú: En la *Figura 3.25*, se visualiza cada una de las opciones del menú al cual estudiante tendrá acceso al momento de iniciar sesión. Cada una de las opciones de menú llevan a otra interfaz en la cual se realiza cada una de las acciones que el estudiante requiere para la gestión de tareas académicas.



Figura 3.25 Menú para estudiantes.

Elaborado por: El investigador.

Ver tareas: En la *Figura 3.26*, se visualiza el diseño de la interfaz para que el estudiante pueda visualizar los detalles de todas las tareas que tenga pendientes de entrega, así como también la opción de ver específicamente las tareas que tengan la fecha de entrega para el día de mañana, desde el día en que se esté revisando las tareas desde la aplicación.

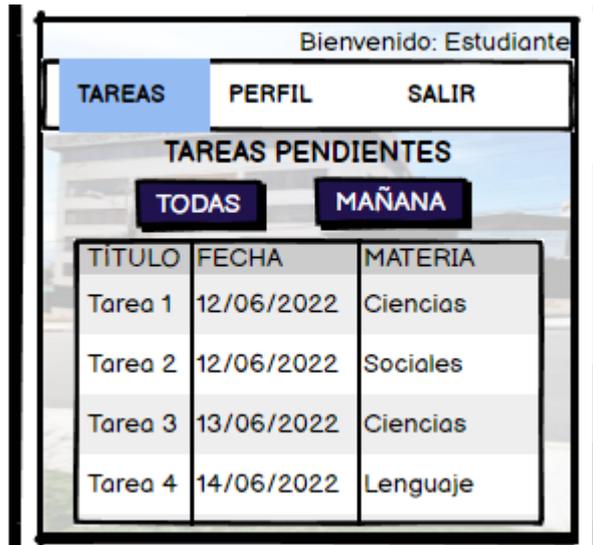


Figura 3.26 Visualizar tareas pendientes de entrega para el estudiante.

Elaborado por: El investigador.

Entregar tareas: En la *Figura 3.27*, se visualiza el diseño de la interfaz para que el estudiante pueda enviar un archivo relacionado con la tarea para así entregarla.



Figura 3.27 Entregar tareas para el estudiante.

Elaborado por: El investigador.

Ver o Editar perfil: En la *Figura 3.28*, se visualiza el diseño de la interfaz para que el estudiante pueda ver la información de su perfil, así como editarla.



Figura 3.28 Ver o Editar perfil para el estudiante.

Elaborado por: El investigador.

Cerrar sesión: En la *Figura 3.29*, se visualiza el diseño de la interfaz para que el estudiante pueda cerrar sesión y regresar al inicio de la aplicación.



Figura 3.29 Cerrar sesión para el estudiante.

Elaborado por: El investigador.

3.2.3 Fase 3: Producción

En la fase de producción de la metodología se desarrolló la aplicación firebase Android, acorde con las funciones establecidas.

3.2.3.1 Backend

Mediante los servicios que ofrece Firebase se realiza la comunicación entre el servidor y las peticiones ejecutadas por los usuarios desde el frontend desarrollado con los widgets propios de Android Studio.

Está conformado por clases, entidades, interfaces y servicios. Firebase permite comunicarse directamente con el servidor sin necesidad de código específico para los servicios, ya que solo es necesario importar las librerías directamente desde Android Studio y así acceder a todos los servicios, lo cual permite ahorrar código y reducir tiempos de desarrollo.

Clases

Las clases permiten hacer referencia a la interfaz gráfica, ejecutar los métodos que el usuario requiera y ejecutar las peticiones hacia los servicios de firebase.

Listado de clases:

- LoginActivity.java
- RegistrarActivity.java
- RecuperarContraseñaActivity.java
- AdministradorActivity.java
- ProfesoresInicioActivity.java
- ProfesoresPerfilFragment.java
- ProfesoresTareasFragment.java
- ProfesorAdapter.java
- ProfesorView.java
- EstudiantesInicioActivity.java
- EstudiantesPerfilFragment.java
- EstudiantesTareasFragment.java
- EstudianteAdapter.java
- EstudianteView.java

- TareaAdapter.java
- TareaAdapterEstudiante.java
- DetallesTareaFragment.java
- EditarTareaFragment.java
- VerEntregasFragment.java
- VisorPdfActivity.java
- NotificacionesPush.java

Entidades

Las entidades son clases con los campos de cada una de las tablas creadas en la base de datos.

Listado de entidades:

- Profesor.java
- Estudiante.java
- Tarea.java
- TareasEntregadas.java

Interfaces

Las interfaces son la representación gráfica que el usuario podrá visualizar e interactuar para ejecutar las diferentes funciones que le corresponden.

Listado de interfaces:

- activity_login.xml
- activity_registrar.xml
- activity_recuperar_contrasena.xml
- activity_administrador.xml
- view_usuarios_admin.xml
- activity_profesores_inicio.xml
- fragment_profesores_perfil.xml

- fragment_profesores_tareas.xml
- activity_estudiantes_inicio.xml
- fragment_estudiantes_perfil.xml
- fragment_estudiantes_tareas.xml
- list_view_tareas.xml
- view_tarea_single.xml
- view_tarea_estudiante.xml
- fragment_ver_entregas.xml
- fragment_editar_tarea.xml
- fragment_detalles_tarea.xml
- activityvisor_pdf.xml
- spinner_item_rol.xml
- ic_launcher_background.xml
- splashscreen.xml
- colors.xml
- strings.xml

Servicios

Los servicios son los que nos permiten comunicarse con el servidor y ejecutar las peticiones del usuario.

Lista de servicios

- com.basgeekball.awesomevalidation.AwesomeValidation
- com.basgeekball.awesomevalidation.ValidationStyle
- com.google.android.gms.tasks.OnCompleteListener
- com.google.android.gms.tasks.OnFailureListener
- com.google.android.gms.tasks.OnSuccessListener
- com.google.android.gms.tasks.Task
- com.google.firebase.auth.AuthResult
- com.google.firebase.auth.FirebaseAuth

- com.google.firebase.auth.FirebaseAuthException
- com.google.firebase.auth.FirebaseUser
- com.google.firebase.firestore.DocumentSnapshot
- com.google.firebase.firestore.FirebaseFirestore
- com.google.firebase.firestore.QueryDocumentSnapshot
- com.google.firebase.firestore.QuerySnapshot
- com.novoda.merlin.Bindable
- com.novoda.merlin.Connectable
- com.novoda.merlin.Disconnectable
- com.novoda.merlin.Merlin
- com.novoda.merlin.NetworkStatus
- java.util.HashMap
- java.util.Map
- java.util.Properties
- javax.mail.Authenticator
- javax.mail.Message
- javax.mail.PasswordAuthentication
- javax.mail.Session
- javax.mail.Transport
- javax.mail.internet.InternetAddress
- javax.mail.internet.MimeMessage;
- com.github.barteksc.pdfviewer.PDFView
- com.google.firebase.storage.FirebaseStorage
- com.google.firebase.storage.StorageReference

Ejemplo de código de los servicios de Firebase

La *Figura 3.30* indica las librerías que deben ser importadas para poder usar los servicios de autenticación de usuarios de firebase y los servicios de Merlin para comprobar si existe o no una conexión a internet.

```

import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseAuthException;
import com.google.firebase.auth.FirebaseUser;
import com.novoda мерlin.Bindable;
import com.novoda мерlin.Connectable;
import com.novoda мерlin.Disconnectable;
import com.novoda мерlin.Merlin;
import com.novoda мерlin.NetworkStatus;

```

Figura 3.30 *Importación de los servicios para comprobar si el usuario ya inició sesión.*

Elaborado por: El investigador.

La **Figura 3.31** indica la interfaz para comprobar si el usuario ya ha iniciado sesión y acceder a la interfaz de inicio directamente.

```

<Space
    android:layout_width="match_parent"
    android:layout_height="15dp"
    android:layout_weight="1" />

<Button
    android:id="@+id/btnIngresar"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:backgroundTint="@color/purple_500"
    android:onClick="ingresar"
    android:text="INGRESAR"
    android:textColor="@color/white" />

```

Figura 3.31 *Interfaz para comprobar si el usuario ya inició sesión.*

Elaborado por: El investigador.

La **Figura 3.32** indica la clase para comprobar si el usuario ya ha iniciado sesión y acceder a la interfaz de inicio directamente.

```

@Override
public void onConnect() {
    //Iniciar directo si el usuario ya se logueo
    FirebaseAuth mAuth = FirebaseAuth.getInstance();
    FirebaseUser user = mAuth.getCurrentUser();
    if (user!=null){
        user.getEmail();
        irInicio(user.getEmail());
    }
    //Habilitar controles
    btnIngresar.setEnabled(true);
    txtRegistrar.setEnabled(true);
    txtCambiarContrasena.setEnabled(true);
}

```

Figura 3.32 Clase para comprobar si el usuario ya inició sesión.

Elaborado por: El investigador.

3.2.3.2 Storycards

El objetivo de las storycards es llevar un registro detallado de las actividades realizadas para poder desarrollar las funciones que la aplicación firebase Android debe ejecutar, teniendo en cuenta los requerimientos ya establecidos de acuerdo con los roles que los usuarios desempeñan, describiendo aspectos importantes como: la dificultad, el tipo y el esfuerzo además se definen las fechas en las que se implementó, se ejecutó y se verificó.

Storycards del Administrador

Las storycards del administrador detallan todas las actividades realizadas para lograr desarrollar las funciones requeridas por este usuario, de acuerdo con los requerimientos ya establecidos.

Storycard – Inicio de sesión para el administrador

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
1	Nuevo	3	3	3	3	Alta
Descripción: Inicio de sesión para el administrador						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
07/06/2022		Implementación			Sin comentario	
07/06/2022		Ejecución			Sin comentario	
07/06/2022		Verificación			Sin comentario	

Tabla 3.9 Storycard del inicio de sesión para el administrador.

Elaborado por: El investigador.

Para la función del inicio de sesión del administrador se creó un método mediante el cual se accede al servicio de firebase para autenticación y con el uso de la librería Awesome Validation se validan que los campos de texto sean correctos, como se observa en la **Figura 3.33**.

```

public void irInicio(String correo){
    //Barra de progreso
    progreso.setMessage("Espere un momento...");
    progreso.setCanceledOnTouchOutside(false);
    progreso.show();
    //Actualizar Tareas
    verificarFecha();
    firestore.collection("Administrador").whereEqualTo("cor_adm", correo).
        get().addOnCompleteListener(new OnCompleteListener<QuerySnapshot>() {
            @Override
            public void onComplete(@NonNull Task<QuerySnapshot> task) {
                if (task.isSuccessful()) {
                    if(task.getResult().isEmpty()){
                        buscarProfesor(correo);
                    }else {
                        for (QueryDocumentSnapshot admin : task.getResult()) {
                            String id = admin.getId();
                            Intent i = new Intent(LoginActivity.this,
                                AdministradorActivity.class);
                            //Evitar que se creen activitys innecesarios
                            i.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP|
                                Intent.FLAG_ACTIVITY_CLEAR_TASK|Intent.FLAG_ACTIVITY_NEW_TASK);
                        }
                    }
                }
            }
        });
}

```

```

//Enviar el correo
i.putExtra("id",id);
//Iniciar nueva actividad
startActivity(i);
    }
}
progreso.dismiss();
}
});
}

```

Figura 3.33 Código del inicio de sesión. Administrador.

Elaborado por: El investigador

Storycard – Aprobar o Rechazar profesores para el administrador

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
2	Nuevo	4	4	4	4	Alta
Descripción: Aprobar o rechazar cuentas de profesores para el administrador						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
07/06/2022		Implementación			Sin comentario	
08/06/2022		Ejecución			Sin comentario	
08/06/2022		Verificación			Sin comentario	

Tabla 3.10 Storycard de aprobar o rechazar profesores para el administrador.

Elaborado por: El investigador.

Para la función de aceptar o rechazar profesores del administrador se creó un método con el cual se obtiene una lista con todas las cuentas que se quieren registrar con el rol de “Profesor”, como se observa en la **Figura 3.34**.

```

public void cargarProfesores(){
    //Inicializar los servicios de firebase
    firestore = FirebaseFirestore.getInstance();

    //Query de búsqueda
    firestore.collection("Profesores").whereEqualTo("ace_pro", "Nuevo").

```

```

        addSnapshotListener(new EventListener<QuerySnapshot>() {
            @Override
            public void onEvent(@Nullable QuerySnapshot value,
                                @Nullable FirebaseFirestoreException error) {
                if(value.isEmpty()){
                    txtListaVacíaPro.setVisibility(View.VISIBLE);
                }else {
                    txtListaVacíaPro.setVisibility(View.INVISIBLE);
                }
            }
        });

        Query query = firestore.collection("Profesores").whereEqualTo("ace_pro", "Nuevo");
        FirestoreRecyclerOptions<ProfesorView> profesores = new FirestoreRecyclerOptions.
            Builder<ProfesorView>().setQuery(query, ProfesorView.class).build();
        profesorAdapter = new ProfesorAdapter(profesores, this);
        //Leer cambios
        profesorAdapter.notifyDataSetChanged();
        //Enviar el adapter al recicler
        listaPro.setAdapter(profesorAdapter);
    }
}

```

Figura 3.34 Código de aprobar o rechazar profesores. Administrador.

Elaborado por: El investigador

Storycard – Aprobar o Rechazar estudiantes para el administrador

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
3	Nuevo	4	4	4	4	Alta
Descripción: Aprobar o rechazar cuentas de estudiantes para el administrador						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
08/06/2022		Implementación			Sin comentario	
08/06/2022		Ejecución			Sin comentario	
08/06/2022		Verificación			Sin comentario	

Tabla 3.11 Storycard de aprobar o rechazar estudiantes para el administrador.

Elaborado por: El investigador.

Para la función de aprobar o rechazar estudiantes del administrador se creó un método con el cual se obtiene una lista con todas las cuentas que se quieren registrar con el rol de “Estudiante”, como se observa en la *Figura 3.35*.

```

public void cargarEstudiantes(){
    //Inicializar los servicios de firebase
    firestore = FirebaseFirestore.getInstance();

    //Query de búsqueda
    firestore.collection("Estudiantes").whereEqualTo("ace_est", "Nuevo").
        addSnapshotListener(new ValueEventListener<QuerySnapshot>() {
            @Override
            public void onEvent(@Nullable QuerySnapshot value,
                               @Nullable FirebaseFirestoreException error) {
                if(value.isEmpty()){
                    txtListaVaciaEst.setVisibility(View.VISIBLE);
                }else {
                    txtListaVaciaEst.setVisibility(View.INVISIBLE);
                }
            }
        });

    Query query = firestore.collection("Estudiantes").whereEqualTo("ace_est", "Nuevo");
    FirestoreRecyclerOptions<EstudianteView> estudiantes = new FirestoreRecyclerOptions.
        Builder<EstudianteView>().setQuery(query, EstudianteView.class).build();
    estudianteAdapter = new EstudianteAdapter(estudiantes, this);
    //Leer cambios
    estudianteAdapter.notifyDataSetChanged();
    //Enviar el adapter al recycler
    listaEst.setAdapter(estudianteAdapter);
}

```

Figura 3.35 Código de aprobar o rechazar estudiantes. Administrador.

Elaborado por: El investigador

Storycard – Cerrar sesión para el administrador

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
4	Nuevo	3	3	3	3	Alta
Descripción: Cerrar sesión para el administrador						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	

08/06/2022	Implementación	Sin comentario
08/06/2022	Ejecución	Sin comentario
08/06/2022	Verificación	Sin comentario

Tabla 3.12 Storycard de cerrar sesión para el administrador.

Elaborado por: El investigador.

Para la función de cerrar sesión del administrador se creó un método con el cual se accede al servicio de autenticación de firebase y se cierra la sesión del usuario, como se observa en la **Figura 3.36**.

```
public void cerrarSesion(View view){
    FirebaseAuth.getInstance().signOut();
    irLogin();
}
```

Figura 3.36 Código para cerrar sesión. Administrador.

Elaborado por: El investigador

Storycards de Profesores

Las storycards de profesores detallan todas las actividades realizadas para lograr desarrollar las funciones requeridas por este usuario, de acuerdo con los requerimientos ya establecidos.

Storycard – Inicio de sesión para profesores

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
5	Nuevo	3	3	3	3	Alta
Descripción: Inicio de sesión para los profesores						
Fecha		Acción		Comentario		
06/06/2022		Definición		Sin comentario		
08/06/2022		Implementación		Sin comentario		
09/06/2022		Ejecución		Sin comentario		

09/06/2022	Verificación	Sin comentario
------------	--------------	----------------

Tabla 3.13 Storycard del inicio de sesión para profesores.

Elaborado por: El investigador.

Para la función del inicio de sesión de profesores se creó un método mediante el cual se accede al servicio de firebase para autenticación y con el uso de la librería Awesome Validation se validan que los campos de texto sean correctos, como se observa en la **Figura 3.37**.

```

public void ingresar(View view){
    //Inicializar los servicios de firebase
    firebaseAuth = firebaseAuth.getInstance();
    //Inicializar los servicios de validacion
    awesomeValidation = new AwesomeValidation(ValidationStyle.BASIC);
    //Cadenas de validacion
    String regexContrasena = "(?=\\w+\\d)(?=\\w+[A-Za-z])\\S{8,}";
    //Reglas de validacion
    awesomeValidation.addValidation(this, R.id.txtEmail,
        Patterns.EMAIL_ADDRESS, "Ingrese un correo válido");
    awesomeValidation.addValidation(this, R.id.txtContrasena,
        regexContrasena, "Contraseña incorrecta");

    if(awesomeValidation.validate()){
        //Asignar variables a los campos
        String correo = txtCorreo.getText().toString();
        String contrasena = txtContrasena.getText().toString();
        //Firebase para iniciar sesion
        firebaseAuth.signInWithEmailAndPassword(correo,contrasena).
            addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    if(task.isSuccessful()){
                        irInicio(correo);
                    }else {
                        String errorCode =((FirebaseAuthException) task.getException()).
                            getErrorCode();
                        mostrarError(errorCode);
                    }
                }
            });
    }
}

```

Figura 3.37 Código del inicio de sesión. Profesores.

Elaborado por: El investigador.

En la **Figura 3.38** se puede ver el método para que el profesor abra la página de inicio que le corresponde, se usó los servicios de firebase para acceder a la base y verificar si el usuario que inicio sesión tiene el rol de “Profesor”.

```

public void irInicio(String correo){
    //Barra de progreso
    progreso.setMessage("Espere un momento...");
    progreso.setCanceledOnTouchOutside(false);
    progreso.show();
    //Actualizar Tareas
    verificarFecha();
    //Inicializar servicios de firebase
    firestore = FirebaseFirestore.getInstance();
    firestore.collection("Profesores").whereEqualTo("cor_pro", correo).get().
        addOnCompleteListener(new OnCompleteListener<QuerySnapshot>() {
            @Override
            public void onComplete(@NonNull Task<QuerySnapshot> task) {
                if (task.isSuccessful()) {
                    if(task.getResult().isEmpty()){
                        buscarEstudiante(correo);
                    }else {
                        for (QueryDocumentSnapshot profesor : task.getResult()) {
                            String id = profesor.getId();
                            Intent i = new Intent(LoginActivity.this, ProfesoresInicioActivity.class);
                            //Evitar que se creen activitiys innecesarios
                            i.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
                                Intent.FLAG_ACTIVITY_CLEAR_TASK|Intent.FLAG_ACTIVITY_NEW_TASK);
                            //Enviar el correo
                            i.putExtra("id",id);
                            //Iniciar nueva actividad
                            startActivity(i);
                        }
                    }
                }
            }
        });
    progreso.dismiss();
}

```

Figura 3.38 Código para abrir la página de inicio. Profesores.

Elaborado por: El investigador.

Storycard – Registro para profesores

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
6	Nuevo	3	3	3	3	Alta
Descripción: Registro de cuenta para los profesores.						
Fecha		Acción			Comentario	

06/06/2022	Definición	Sin comentario
09/06/2022	Implementación	Sin comentario
10/06/2022	Ejecución	Sin comentario
11/06/2022	Verificación	Sin comentario

Tabla 3.14 Storycard del registro de cuenta para profesores.

Elaborado por: El investigador.

Para el registro de cuenta para los profesores, se creó un método mediante el cual se mapea la información del profesor en un objeto de tipo “Map”, el cual será enviado a la base de datos de firebase para su inserción como se muestra en la **Figura 3.39**.

```
private void insertarProfesor(String correo,String nombres, String apellidos,String telefono){
    //Inicializar los servicios de firebase
    firestore = FirebaseFirestore.getInstance();

    //Mapear la informacion a un variable
    Map<String, Object> profesor = new HashMap<>();
    profesor.put("cor_pro",correo);
    profesor.put("nom_pro",nombres);
    profesor.put("ape_pro",apellidos);
    profesor.put("tel_pro",telefono);
    profesor.put("cur_pro","NA");
    profesor.put("mat1_pro","NA");
    profesor.put("mat2_pro","NA");

    //Insertar el usuario en la tabla
    firestore.collection("Profesores").add(profesor).
        addOnSuccessListener(new OnSuccessListener<DocumentReference>() {
            @Override
            public void onSuccess(DocumentReference documentReference) {
                Toast.makeText(RegistrarActivity.this, "Usuario creado con éxito!",
                    Toast.LENGTH_SHORT).show();
            }
        })
}
```

Figura 3.39 Código del registro de cuenta. Profesores.

Elaborado por: El investigador.

Storycard – Restablecer contraseña para profesores

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
7	Nuevo	3	3	3	3	Alta

Descripción: Restablecer contraseña para los profesores.		
Fecha	Acción	Comentario
06/06/2022	Definición	Sin comentario
10/06/2022	Implementación	Sin comentario
10/06/2022	Ejecución	Sin comentario
11/06/2022	Verificación	Sin comentario

Tabla 3.15 Storycard de restablecer contraseña para profesores.

Elaborado por: El investigador.

Para la función de restablecer contraseña se creó un método el cual toma el correo electrónico que el usuario asigne y se envía el enlace correspondiente para que el profesor pueda recuperar la contraseña de su cuenta como indica la figura 3.27.

```

public void restablecerContraseña(View view){
    //Instanciar variable de firebase
    firebaseAuth = FirebaseAuth.getInstance();
    //Establecer el idioma del correo para restablecer
    firebaseAuth.setLanguageCode("es");
    //Inicializar servicios de validacion
    awesomeValidation = new AwesomeValidation(ValidationStyle.BASIC);
    //Cadenas de validacion
    awesomeValidation.addValidation(this, R.id.txtCorreoRecuperar,
        Patterns.EMAIL_ADDRESS, R.string.correo_invalido);
    if(awesomeValidation.validate()){
        //Asignar variables
        String correo = txtCorreo.getText().toString();
        progreso.setMessage("Espere un momento...");
        progreso.setCanceledOnTouchOutside(false);
        progreso.show();
        //Enviar mensaje al correo dado
        firebaseAuth.sendPasswordResetEmail(correo).
            addOnCompleteListener(new OnCompleteListener<Void>() {
                @Override
                public void onComplete(@NonNull Task<Void> task) {
                    if(task.isSuccessful()){
                        Toast.makeText(RecuperarContraseñaActivity.this,
                            "El correo para restablecer contraseña, fue enviado con éxito.",
                            Toast.LENGTH_LONG).show();
                    }else{
                        Toast.makeText(RecuperarContraseñaActivity.this,
                            "No se pudo enviar el correo",Toast.LENGTH_SHORT).show();
                    }
                }
            })
    }
}

```

```

        progreso.dismiss();
        finish();
    }
});

```

Figura 3.40 Código de restablecer contraseña. Profesores.

Elaborado por: El investigador.

Storycard – Agregar tarea para profesores

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
8	Nuevo	4	4	4	4	Alta
Descripción: Agregar una nueva tarea para los profesores.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
11/06/2022		Implementación			Sin comentario	
13/06/2022		Ejecución			Sin comentario	
13/06/2022		Verificación			Sin comentario	

Tabla 3.16 Storycard de restablecer contraseña para profesores.

Elaborado por: El investigador.

Para la función de agregar una tarea de los profesores, se creó un método que compruebe que el profesor tiene un curso o una materia ya asignada, sino es así no se puede crear una nueva tarea como se puede observar en la **Figura 3.41**.

```

public void verificarCurso(){
    ProfesoresInicioActivity referencia = new ProfesoresInicioActivity();
    referencia.ocultarLista();
    profesor = referencia.profesor;
    if(profesor.getCurso().equals("NA") && profesor.getMateria().equals("NA")){
        txtAlerta.setVisibility(getView().VISIBLE);
    }else {
        if(profesor.getCurso().equals("NA")){
            cargarMaterias(profesor);
        }
    }
}

```

```

        if(profesor.getMateria1().equals("NA")){
            cargarCursos(profesor);
        }
        agregar.setVisibility(getView().VISIBLE);
    }
    ver.setVisibility(getView().INVISIBLE);
}
}

```

Figura 3.41 Código de verificar curso o materia. Profesores.

Elaborado por: El investigador.

Como muestra la **Figura 3.42**, se cargan las materias que el profesor tiene disponibles para asignarlas en una nueva tarea.

```

public void cargarMaterias(Profesor profe){
    //Tomar valores
    String mat1 = profe.getMateria1();
    String mat2 = profe.getMateria2();

    //Dar opciones al spinner de cursos
    String[] opcionesCursos = {"Segundo", "Tercero", "Cuarto", "Quinto", "" +
        "Sexto", "Séptimo", "Octavo", "Noveno", "Décimo"};
    ArrayAdapter<String> adapter = new ArrayAdapter<>(getContext(),
        R.layout.spinner_item_roles, opcionesCursos);
    cursos.setAdapter(adapter);

    //Dar opciones al spinner de materias
    if(mat2.equals("NA")){
        String[] opcionesmaterias = {mat1};
        ArrayAdapter<String> adapter1 = new ArrayAdapter<>(getContext(),
            R.layout.spinner_item_roles, opcionesmaterias);
        materias.setAdapter(adapter1);
    }else {
        String[] opcionesmaterias = {mat1,mat2};
        ArrayAdapter<String> adapter1 = new ArrayAdapter<>(getContext(),
            R.layout.spinner_item_roles, opcionesmaterias);
        materias.setAdapter(adapter1);
    }
}
}

```

Figura 3.42 Código para cargar materias disponibles. Profesores.

Elaborado por: El investigador.

En la **Figura 3.43** se observa el método para cargar los cursos disponibles a los cuales el profesor puede asignar una nueva tarea.

```
public void cargarCursos(Profesor profe){
    //Tomar valores
    String cur = profe.getCurso();

    //Dar opciones al spinner de cursos
    String[] opcionesCursos = {cur};
    ArrayAdapter<String> adapter = new ArrayAdapter<~>(getContext(),
        R.layout.spinner_item_roles, opcionesCursos);
    cursos.setAdapter(adapter);

    //Dar opciones al spinner de materias
    String[] opcionesMaterias = {"Lengua y literatura", "Matemática", "" +
        "Ciencias sociales", "Ciencias naturales"};
    ArrayAdapter<String> adapter1 = new ArrayAdapter<~>(getContext(),
        R.layout.spinner_item_roles, opcionesMaterias);
    materias.setAdapter(adapter1);
}
```

Figura 3.43 Código para cargar cursos disponibles. Profesores.

Elaborado por: El investigador.

La **Figura 3.44** muestra el método mediante el cual se accede a la base de datos de firebase, se mapea la información de la tarea en un objeto de tipo “Map” y se guarda dicha tarea en la tabla “Tareas”.

```
public void insertarTareaBase(){
    //Inicializar los servicios de firebase
    firestore = FirebaseFirestore.getInstance();

    //Tomar la fecha de hoy para comprobar
    Date f = new Date();
    String fechaH = formatoFecha(f);

    //Tomar la fecha de mañana para comprobar
    Date mañana = new Date(f.getTime() + (1000 * 60 * 60 * 24));
    String fechaM = formatoFecha(mañana);
}
```

```

String mat = materias.getSelectedItem().toString();
String pro = profesor.getId();

//Servicios de validacion
awesomeValidation = new AwesomeValidation(ValidationStyle.BASIC);
//Cadenas de validacion
String regexVacio = ".{4,}";
//Reglas de validacion
awesomeValidation.addValidation(getActivity(), R.id.txtTituloPro,
    regexVacio, "Información requerida");
awesomeValidation.addValidation(getActivity(), R.id.txtInstruccionesPro,
    regexVacio, "Información requerida");

if(awesomeValidation.validate()){
    //Mapear la infromacion a un variable
    Map<String, Object> map = new HashMap<>();
    map.put("tit_tar",tit);
    map.put("ins_tar",ins);
    map.put("fec_ent_tar",fecha);
    if(fecha.equals(fechaH)){
        map.put("est_tar", "Hoy");
    }else{
        if(fecha.equals(fechaM)){
            map.put("est_tar", "Mañana");
        }else{
            map.put("est_tar", "Nueva");
        }
    }
    map.put("cur_tar",cur);
    map.put("mat_tar",mat);
    map.put("pro_tar",pro);

    //Insertar el tarea en la tabla
    firestore.collection("Tareas").add(map).
        addOnSuccessListener(new OnSuccessListener<DocumentReference>() {
            @Override
            public void onSuccess(DocumentReference documentReference) {
                Toast.makeText(getContext(), "Tarea creada con éxito!",
                    Toast.LENGTH_SHORT).show();
            }
        });
    encerarCamposAgregar();
}

```

Figura 3.44 Código para guardar la tarea en la base de datos. Profesores.

Elaborado por: El investigador.

Storycard – Ver tareas pendientes para profesores

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
9	Nuevo	4	4	4	4	Alta
Descripción: Ver tareas pendientes de revisión para los profesores.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
13/06/2022		Implementación			Sin comentario	
15/06/2022		Ejecución			Sin comentario	
15/06/2022		Verificación			Sin comentario	

Tabla 3.17 Storycard de ver tareas pendientes para profesores.

Elaborado por: El investigador.

Para la función de ver tareas pendientes de los profesores se creó una clase de tipo “Adapter” la cual carga los datos de la tarea en la interfaz gráfica correspondiente, como se muestra en la **Figura 3.45**.

```

public class TareaAdapter extends FirestoreRecyclerAdapter<Tarea,TareaAdapter.ViewHolder> {
    //Servicios de firebase
    private FirebaseFirestore firestore = FirebaseFirestore.getInstance();
    //Declarar activity para el toast
    Activity activity;
    /**
     * Create a new RecyclerView adapter that listens to a Firestore Query. See {@link
     * FirestoreRecyclerOptions} for configuration options.
     *
     * @param options
     */
    public TareaAdapter(@NonNull FirestoreRecyclerOptions<Tarea> options, Activity activity) {
        super(options);

        this.activity = activity;
    }
}

```

Figura 3.45 Adapter de tareas. Profesores.

Elaborado por: El investigador.

En la **Figura 3.46** se observa el método que busca las tareas de acuerdo con el id del profesor para mostrar solo las tareas que hayan sido creadas por él.

```
public void cargarTareas(){
    //Inicializar los servicios de firebase
    firestore = FirebaseFirestore.getInstance();

    //Query de busqueda
    Query query = firestore.collection("Tareas").whereEqualTo("pro_tar",id);

    FirestoreRecyclerOptions<Tarea> tareas = new FirestoreRecyclerOptions.
        Builder<Tarea>().setQuery(query,Tarea.class).build();
    tareaAdapter = new TareaAdapter(tareas,this);
    //Leer cambios
    tareaAdapter.notifyDataSetChanged();
    //Enviar el adapter al recycler
    lista.setAdapter(tareaAdapter);
}
```

Figura 3.46 Código de buscar las tareas por el id. Profesores.

Elaborado por: El investigador.

La **Figura 3.47** indica el método que carga los elementos de la interfaz gráfica y la información dentro de ellos.

```
protected void onBindViewHolder(@NonNull ViewHolder viewHolder,
                                int i, @NonNull Tarea Tarea) {
    //Obtener el id del dato seleccionado
    DocumentSnapshot documentSnapshot = getSnapshots().
        getSnapshot(viewHolder.getAdapterPosition());
    final String id = documentSnapshot.getId();

    //Mostrar los datos en los elementos
    viewHolder.txtTitulo.setText(Tarea.getTit_tar());
    String estado = Tarea.getEst_tar();
    switch (estado){
        case "Nueva":
            viewHolder.txtFecha.setText(Tarea.getFec_ent_tar());
            break;
        case "Hoy":
            viewHolder.txtFecha.setText(estado);
            break;
    }
}
```

```

        case "Mañana":
            viewHolder.txtFecha.setText(estado);
            break;
    }
    viewHolder.txtMateria.setText(Tarea.getMat_tar());
    viewHolder.txtCurso.setText(Tarea.getCur_tar());
    viewHolder.txtInstrucciones.setText(Tarea.getIns_tar());

    //LLamar al boton de eliminar
    viewHolder.btnEliminar.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) { eliminarTarea(id); }
    });
}

```

Figura 3.47 Código de cargar datos de la tarea en la interfaz gráfica. Profesores.

Elaborado por: El investigador.

Storycard – Revisar entregas para profesores

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
10	Nuevo	4	4	4	4	Alta
Descripción: Revisar los archivos entregados de las tareas para los profesores.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
13/06/2022		Implementación			Sin comentario	
14/06/2022		Ejecución			Sin comentario	
15/06/2022		Verificación			Sin comentario	

Tabla 3.18 Storycard de revisar entregas para profesores.

Elaborado por: El investigador.

Para la función de revisar entregas de los profesores se creó un método que busca los archivos entregados de acuerdo con la tarea seleccionada y los muestra en una lista para seleccionar y ver un archivo específico, como se muestra en la **Figura 3.48**.

15/06/2022	Verificación	Sin comentario
------------	--------------	----------------

Tabla 3.19 *Storycard de recordar tareas para profesores.*

Elaborado por: El investigador.

Para la función de recordar tareas a través del correo electrónico se crea un método mediante el cual se accede a la base de datos mediante los servicios de firebase y se obtiene una lista con los correos de los estudiantes a los cuales se les quiere recordar la tarea, después se procede a enviar los correos electrónicos haciendo uso de la librería propia de java, como se muestra en la **Figura 3.49**.

```

public void enviar(String emisor, String contra, String destino, String fecha,
                  String instrucciones, String titulo){
    //Cuerpo de correo
    String fechaF = detalleFecha(fecha);
    String tit = "<p> <b>Tarea: </b>" + titulo + "</p>";
    String ins = "<p> <b>Instrucciones: </b>" + instrucciones + "</p>";
    String fec = "<p> <b>Fecha: </b>" + fechaF + "</p>";
    String alerta = "<b>Para más detalles revisa la aplicación.</b>";
    String mensaje = tit + ins + fec + alerta;

    //Políticas de privacidad
    StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder().permitAll().build();
    StrictMode.setThreadPolicy(policy);
    Properties p = new Properties();
    p.put("mail.smtp.host", "smtp.office365.com");
    p.put("mail.smtp.ssl.trust", "smtp.office365.com");
    p.put("mail.smtp.auth", "true");
    p.put("mail.smtp.port", "587");
    p.put("mail.smtp.starttls.enable", "true");
    try {
        session = Session.getDefaultInstance(p, getPasswordAuthentication() → {
            return new PasswordAuthentication(emisor, contra);
        });
    };
    if(session!=null){
        Message message = new MimeMessage(session);
        message.setFrom(new InternetAddress(emisor));
        message.setSubject("Recordatorio de tarea");
        message.setRecipients(Message.RecipientType.TO, InternetAddress.parse(destino));
        message.setContent(mensaje, "text/html; charset=utf-8");
        Transport transport = session.getTransport("smtp");
        transport.connect("smtp.office365.com", 587, emisor, contra);
        transport.sendMessage(message, message.getAllRecipients());
        transport.close();
    }
}

```

Figura 3.49 *Código para recordar tareas. Profesores.*

Elaborado por: El investigador.

Storycard – Ver perfil para profesores

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
12	Nuevo	4	4	4	4	Alta
Descripción: Ver la información del perfil para los profesores.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
13/06/2022		Implementación			Sin comentario	
14/06/2022		Ejecución			Sin comentario	
14/06/2022		Verificación			Sin comentario	

Tabla 3.20 Storycard de ver perfil para profesores.

Elaborado por: El investigador.

Para la función de ver la información del perfil para profesores, se creó una entidad llamada “Profesor”, la cual almacena y devuelve la información de acuerdo sea requerida, como indica la **Figura 3.50**.

```

public class Profesor {
    String id;
    String Nombres;
    String Apellidos;
    String Telefono;
    String Curso;
    String Materia1;
    String Materia2;

    //Constructor
    public Profesor() {
    }

    //Metodos get de datos
    public String getNombres() { return Nombres; }
    public String getApellidos() { return Apellidos; }
    public String getTelefono() { return Telefono; }
    public String getCurso() { return Curso; }
    public String getMaterial() { return Materia1; }
}

```

```

public void setMateria2(String materia2) { Materia2 = materia2; }
public void setId(String id) { this.id = id; }
public void setNombres(String nombres) { Nombres = nombres; }
public void setApellidos(String apellidos) { Apellidos = apellidos; }
public void setTelefono(String telefono) { Telefono = telefono; }
}

```

Figura 3.50 Entidad Profesores.

Elaborado por: El investigador.

La *Figura 3.51* muestra el método para asignar valores a las opciones de cursos y materias, para que el profesor los pueda seleccionar.

```

public void llenarSpinners(Profesor profe){
    //Asignar variables
    String cur = profe.getCurso();
    String mat1 = profe.getMateria1();
    String mat2 = profe.getMateria2();

    //Dar opciones al spinner de cursos
    String[] opcionesCursos = {"NA", "Segundo", "Tercero", "Cuarto", "Quinto"};
    ArrayAdapter<String> adapter = new ArrayAdapter<>(getContext(),
        R.layout.spinner_item_roles, opcionesCursos);
    cursos.setAdapter(adapter);
    switch (cur){
        case "NA":
            cursos.setSelection(0);
            break;
        case "Segundo":
            cursos.setSelection(1);
            break;
        case "Tercero":
            cursos.setSelection(2);
            break;
        case "Cuarto":
            cursos.setSelection(3);
            break;
        case "Quinto":
            cursos.setSelection(4);
            break;
    }
}

```

```

//Dar opciones al spinner de materia 1
String[] opcionesMaterial = {"NA", "Lengua y literatura", "Matemática",
    "Ciencias sociales", "Ciencias naturales", "Educación artística",
    "Inglés", "Educación cristiana", "Educación física"};
ArrayAdapter<String> adapter2 = new ArrayAdapter<~>(getContext(),
    R.layout.spinner_item_roles, opcionesMaterial);
materias1.setAdapter(adapter2);
switch (mat1){
    case "NA":
        materias1.setSelection(0);
        break;
    case "Lengua y literatura":
        materias1.setSelection(1);
        break;
    case "Matemática":
        materias1.setSelection(2);
        break;
    case "Ciencias sociales":
        materias1.setSelection(3);
        break;
    case "Ciencias naturales":
        materias1.setSelection(4);
        break;
    case "Educación artística":
        materias1.setSelection(5);
        break;
    case "Inglés":
        materias1.setSelection(6);
        break;
    case "Educación cristiana":
        materias1.setSelection(7);
        break;
    case "Educación física":
        materias1.setSelection(8);
        break;
}

```

Figura 3.51 Código para cargar opciones de cursos y materias. Profesores.

Elaborado por: El investigador.

En la **Figura 3.52** se observa el método creado para mostrar la información del perfil, en los campos de texto correspondientes.

```

public void llenarDatos(){
    //Llenar los datos del profesor
    profesor = referencia.profesor;
    informacion.setVisibility(getView().VISIBLE);
    txtNombres.setText(profesor.getNombres());
    txtApellidos.setText(profesor.getApellidos());
    txtTelefono.setText(profesor.getTelefono());

    //Llenar spinners
    llenarSpinners(profesor);
}

```

Figura 3.52 Código para mostrar información del perfil. Profesores.

Elaborado por: El investigador.

Storycard – Actualizar perfil para profesores

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
13	Nuevo	4	4	4	4	Alta
Descripción: Actualizar la información del perfil para los profesores.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
14/06/2022		Implementación			Sin comentario	
15/06/2022		Ejecución			Sin comentario	
15/06/2022		Verificación			Sin comentario	

Tabla 3.21 Storycard de actualizar perfil para profesores.

Elaborado por: El investigador.

Para actualizar la información del perfil de profesores se creó un método el cual capta la información que el usuario actualizó y se asigna a un objeto de tipo “Map” el cual es enviado a la base de datos mediante los servicios de firebase para su actualización, como se puede observar en la **Figura 3.53**.

```

public void guardarCambios(){
    //Asignar variables
    String id = profesor.getId();
    String tel = txtTelefono.getText().toString();
    String cur = cursos.getSelectedItem().toString();
    String mat1 = materias1.getSelectedItem().toString();
    String mat2 = materias2.getSelectedItem().toString();

    //Mapear la informacion
    Map<String, Object> map = new HashMap<>();
    map.put("tel_pro", tel);
    map.put("cur_pro", cur);
    map.put("mat1_pro", mat1);
    map.put("mat2_pro", mat2);

    //Actualizar datos
    firestore.collection("Profesores").document(id).update(map).
        addOnSuccessListener(new OnSuccessListener<Void>() {
            @Override
            public void onSuccess(Void unused) {
                Toast.makeText(getContext(), "Actualizado exitosamente", Toast.LENGTH_SHORT).show();
                informacion.setVisibility(getView().INVISIBLE);
            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Toast.makeText(getContext(), "Error al actualizar", Toast.LENGTH_SHORT).show();
            }
        });
}
}

```

Figura 3.53 Código para actualizar información del perfil. Profesores.

Elaborado por: El investigador.

Storycard – Cerrar sesión para profesores

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
14	Nuevo	3	3	3	3	Alta
Descripción: Cerrar sesión para los profesores.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
16/06/2022		Implementación			Sin comentario	

17/06/2022	Ejecución	Sin comentario
17/06/2022	Verificación	Sin comentario

Tabla 3.22 Storycard de cerrar sesión para profesores.

Elaborado por: El investigador.

Para la función de cerrar sesión se creó un método que cierra la conexión establecida con el servicio de autenticación de firebase, como se observa en la **Figura 3.54**.

```
public void cerrarSesion(View view){
    FirebaseAuth.getInstance().signOut();
    irLogin();
}
```

Figura 3.54 Código para cerrar sesión. Profesores.

Elaborado por: El investigador.

En la **Figura 3.55** se puede observar el método creado para abrir la página de inicio, después que el profesor haya cerrado sesión.

```
public void irLogin(){
    Intent i = new Intent(ProfesoresInicioActivity.this, LoginActivity.class);
    //Evitar que se creen activitys innecesarios
    i.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP|Intent.FLAG_ACTIVITY_CLEAR_TASK|
        Intent.FLAG_ACTIVITY_NEW_TASK);
    //Iniciar nueva actividad
    startActivity(i);
}
```

Figura 3.55 Código para abrir página de inicio. Profesores.

Elaborado por: El investigador.

Storycards de Estudiantes

Las storycards de estudiantes detallan todas las actividades realizadas para lograr desarrollar las funciones requeridas por este usuario, de acuerdo con los requerimientos ya establecidos.

Storycard – Inicio de sesión para estudiantes

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
15	Nuevo	3	3	3	3	Alta
Descripción: Inicio de sesión para los estudiantes						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
20/06/2022		Implementación			Sin comentario	
21/06/2022		Ejecución			Sin comentario	
21/06/2022		Verificación			Sin comentario	

Tabla 3.23 Storycard del inicio de sesión para estudiantes.

Elaborado por: El investigador.

Para la función del inicio de sesión de estudiantes se creó un método mediante el cual se accede al servicio de firebase para autenticación y con el uso de la librería Awesome Validation se validan que los campos de texto sean correctos, como se observa en la **Figura 3.56**.

```

public void ingresar(View view){
    //Inicializar los servicios de firebase
    firebaseAuth = firebaseAuth.getInstance();
    //Inicializar los servicios de validacion
    awesomeValidation = new AwesomeValidation(ValidationStyle.BASIC);
    //Cadenas de validacion
    String regexContrasena = "(?=\w+\d)(?=\w+[A-Za-z])\S{8,}";
    //Reglas de validacion
    awesomeValidation.addValidation(this, R.id.txtEmail, Patterns.EMAIL_ADDRESS, "Ingrese un correo válido");
    awesomeValidation.addValidation(this, R.id.txtContrasena, regexContrasena, "Contraseña incorrecta");

    if(awesomeValidation.validate()){
        //Asignar variables a los campos
        String correo = txtCorreo.getText().toString();
        String contrasena = txtContrasena.getText().toString();
        //Firebase para iniciar sesion
        firebaseAuth.signInWithEmailAndPassword(correo, contrasena).
        addOnCompleteListener(new OnCompleteListener<AuthResult>() {
            @Override
            public void onComplete(@NonNull Task<AuthResult> task) {
                if(task.isSuccessful()){
                    irInicio(correo);
                }else {
                    String errorCode = ((FirebaseAuthException) task.getException()).getErrorCode();
                }
            }
        });
    }
}

```


Descripción: Registro de cuenta para los estudiantes.		
Fecha	Acción	Comentario
06/06/2022	Definición	Sin comentario
21/06/2022	Implementación	Sin comentario
21/06/2022	Ejecución	Sin comentario
21/06/2022	Verificación	Sin comentario

Tabla 3.24 Storycard del registro de cuenta para estudiantes.

Elaborado por: El investigador.

Para el registro de cuenta para los estudiantes, se creó un método mediante el cual se mapea la información del estudiante en un objeto de tipo “Map”, el cual será enviado a la base de datos de firebase para su inserción como se muestra en la **Figura 3.58**.

```
private void insertarProfesor(String correo,String nombres, String apellidos,String telefono){
    //Inicializar los servicios de firebase
    firestore = FirebaseFirestore.getInstance();

    //Mapear la informacion a un variable
    Map<String, Object> profesor = new HashMap<>();
    profesor.put("cor_pro", correo);
    profesor.put("nom_pro", nombres);
    profesor.put("ape_pro", apellidos);
    profesor.put("tel_pro", telefono);
    profesor.put("cur_pro", "NA");
    profesor.put("mat1_pro", "NA");
    profesor.put("mat2_pro", "NA");

    //Insertar el usuario en la tabla
    firestore.collection("Profesores").add(profesor).
        onSuccess(new OnSuccessListener<DocumentReference>() {
            @Override
            public void onSuccess(DocumentReference documentReference) {
                Toast.makeText(RegistrarActivity.this, "Usuario creado con éxito!",
                    Toast.LENGTH_SHORT).show();
            }
        })
}
```

Figura 3.58 Código del registro de cuenta. Estudiantes.

Elaborado por: El investigador.

Storycard – Restablecer contraseña para estudiantes

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
17	Nuevo	3	3	3	3	Alta
Descripción: Restablecer contraseña para los estudiantes.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
22/06/2022		Implementación			Sin comentario	
23/06/2022		Ejecución			Sin comentario	
23/06/2022		Verificación			Sin comentario	

Tabla 3.25 Storycard de restablecer contraseña para estudiantes.

Elaborado por: El investigador.

Para la función de restablecer contraseña se creó un método el cual toma el correo electrónico que el usuario asigne y se envía el enlace correspondiente para que el estudiante pueda recuperar la contraseña de su cuenta como indica la *Figura 3.59*.

```

public void restablecerContraseña(View view){
    //Instanciar variable de firebase
    firebaseAuth = FirebaseAuth.getInstance();
    //Establecer el idioma del correo para restablecer
    firebaseAuth.setLanguageCode("es");
    //Inicializar servicios de validacion
    awesomeValidation = new AwesomeValidation(ValidationStyle.BASIC);
    //Cadenas de validacion
    awesomeValidation.addValidation(this, R.id.txtCorreoRecuperar,
        Patterns.EMAIL_ADDRESS, R.string.correo_invalido);
    if(awesomeValidation.validate()){
        //Asignar variables
        String correo = txtCorreo.getText().toString();
        progreso.setMessage("Espere un momento...");
        progreso.setCanceledOnTouchOutside(false);
        progreso.show();
        //Enviar mensaje al correo dado
        firebaseAuth.sendPasswordResetEmail(correo).
            addOnCompleteListener(new OnCompleteListener<Void>() {

```

```

@Override
public void onComplete(@NonNull Task<Void> task) {
    if(task.isSuccessful()){
        Toast.makeText(RecuperarContrasenaActivity.this,
            "El correo para restablecer contraseña, fue enviado con éxito.",
            Toast.LENGTH_LONG).show();
    }else{
        Toast.makeText(RecuperarContrasenaActivity.this,
            "No se pudo enviar el correo", Toast.LENGTH_SHORT).show();
    }
    progreso.dismiss();
    finish();
}
});

```

Figura 3.59 Código de restablecer contraseña. Estudiantes.

Elaborado por: El investigador.

Storycard – Ver tareas pendientes para estudiantes

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
18	Nuevo	4	4	4	4	Alta
Descripción: Ver tareas pendientes de entrega para los estudiantes.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
23/06/2022		Implementación			Sin comentario	
24/06/2022		Ejecución			Sin comentario	
24/06/2022		Verificación			Sin comentario	

Tabla 3.26 Storycard de ver tareas pendientes para estudiantes.

Elaborado por: El investigador.

Para implementar la funcionalidad de ver tareas pendientes de los estudiantes, se creó una clase de tipo “Adapter” la cual carga los datos en la interfaz gráfica correspondiente como se puede observar en la **Figura 3.60**.

```

public class TareaAdapterEstudiante extends FirestoreRecyclerAdapter<Tarea,
    TareaAdapterEstudiante.ViewHolder> {
    FirebaseFirestore firestore = FirebaseFirestore.getInstance();
    FragmentManager manager;
    public TareaAdapterEstudiante(@NonNull FirestoreRecyclerOptions<Tarea> options) {
        super(options);
    }
}

```

Figura 3.60 Adapter de tareas. Estudiantes.

Elaborado por: El investigador.

En la **Figura 3.61** se puede observar el método creado para buscar las tareas de acuerdo con el curso del estudiante, usando el query de búsqueda de firebase para mostrar solo las tareas del curso al que pertenece el estudiante.

```

public void cargarTareas(String cur){
    lista.setLayoutManager(new LinearLayoutManager(this));
    //Inicializar los servicios de firebase
    firestore = FirebaseFirestore.getInstance();

    //Query de busqueda
    Query query = firestore.collection("Tareas").whereEqualTo("cur_tar", cur);

    FirestoreRecyclerOptions<Tarea> tareas = new FirestoreRecyclerOptions.
        Builder<Tarea>().setQuery(query, Tarea.class).build();
    tareaAdapter = new TareaAdapterEstudiante(tareas, getSupportFragmentManager());
    //Leer cambios
    tareaAdapter.notifyDataSetChanged();
    //Enviar el adapter al recycler
    lista.setAdapter(tareaAdapter);
    tareaAdapter.startListening();
}

```

Figura 3.61 Código de buscar las tareas por el curso. Estudiantes.

Elaborado por: El investigador.

La **Figura 3.62** indica el método creado para mostrar la información de la tarea en la interfaz gráfica.

```

public void leerTarea(){
    firestore.collection("Tareas").document(idTarea).get().
        addOnSuccessListener(new OnSuccessListener<DocumentSnapshot>() {
            @Override
            public void onSuccess(DocumentSnapshot tarea) {

```

```

String tit = tarea.getString("tit_tar");
String det = tarea.getString("ins_tar");
String fec = tarea.getString("fec_ent_tar");
String mat = tarea.getString("mat_tar");
String est = tarea.getString("est_tar");
txtTitulo.setText(tit);
txtInstrucciones.setText(det);
txtMateria.setText(mat);
String fecF = detalleFecha(fec);
if(est.equals("Hoy")){
    fecF = "HOY: " + fecF;
}
if(est.equals("Mañana")){
    fecF = "MAÑANA: " + fecF;
}
txtFecha.setText(fecF);
}
});
}

```

Figura 3.62 Código para mostrar datos de la tarea en la interfaz gráfica.
Estudiantes.

Elaborado por: El investigador.

Storycard – Entregar tareas para estudiantes

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
19	Nuevo	4	4	4	4	Alta
Descripción: Entregar los archivos para cumplir la tarea para los estudiantes.						
Fecha		Acción		Comentario		
06/06/2022		Definición		Sin comentario		
24/06/2022		Implementación		Sin comentario		
24/06/2022		Ejecución		Sin comentario		
24/06/2022		Verificación		Sin comentario		

Tabla 3.27 Entregar tareas para estudiantes.

Elaborado por: El investigador.

Para implementar la funcionalidad de entregar tareas para los estudiantes se creó un método mediante el cual se abre una ventana para que el estudiante seleccione el archivo que quiere subir y almacenarlo en la base de datos de firebase como se puede observar en la **Figura 3.63**.

```

public void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if(data==null){
        Toast.makeText(getApplicationContext(), "No se seleccionó ningún archivo.",
            Toast.LENGTH_SHORT).show();
    }else{
        //Barra de progreso
        progreso.setMessage("Subiendo archivo...");
        progreso.setCanceledOnTouchOutside(false);
        progreso.show();
        //Cargar archivo
        archivo = data.getData();
        StorageReference storageReference = FirebaseStorage.getInstance().getReference();
        final String nombreArc = nomEst + "-" + titTarea;
        //Subir el archivo
        final StorageReference filepath = storageReference.child(nombreArc + "." + "pdf");
        filepath.putFile(archivo).
            addOnCompleteListener(new OnCompleteListener<UploadTask.TaskSnapshot>() {
                @Override
                public void onComplete(@NonNull Task<UploadTask.TaskSnapshot> task) {
                    if(task.isSuccessful()){
                        if(!entrega){
                            agregarEntrega(nombreArc);
                        }
                        progreso.dismiss();
                    }
                }
            });
        getDialog().dismiss();
        Toast.makeText(getApplicationContext(), "Tarea entregada", Toast.LENGTH_SHORT).show();
    }
}

```

Figura 3.63 Código de selecciona el archivo de entrega. Estudiantes.

Elaborado por: El investigador.

En la **Figura 3.64** se puede observar el método para insertar el detalle de la entrega en la tabla “Tareas_Entregadas” en la base de datos Cloud Firestore.

```

public void agregarEntrega(String idArchivo){
    String arc = idArchivo + ".pdf";
    Map<String, Object> map = new HashMap<>();
}

```

```

map.put("id_tar_ent",idTarea);
map.put("id_est_ent",idEst);
map.put("id_arc_ent",arc);

firestore.collection("Tareas_Entregadas").add(map);
}

```

Figura 3.64 Código de insertar entrega en la base de datos. Estudiantes.

Elaborado por: El investigador.

Storycard – Ver perfil para estudiantes

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
20	Nuevo	4	4	4	4	Alta
Descripción: Ver la información del perfil para los estudiantes.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
24/06/2022		Implementación			Sin comentario	
25/06/2022		Ejecución			Sin comentario	
25/06/2022		Verificación			Sin comentario	

Tabla 3.28 Storycard de ver perfil para estudiantes.

Elaborado por: El investigador.

Para la función de ver la información del perfil para estudiantes, se creó una entidad llamada “Estudiante”, la cual almacena y devuelve la información según sea requerida, como indica la *Figura 3.65*.

```

public class Estudiante {
    String id;
    String Nombres;
    String Apellidos;
    String Telefono;
    String Curso;

    //Constructor
    public Estudiante() {

```

```

}

//Metodos get de datos
public String getNombres() { return Nombres; }
public String getApellidos() { return Apellidos; }
public String getTelefono() { return Telefono; }
public String getCurso() { return Curso; }
public String getId() { return id; }

//Metodos set de datos
public void setCurso(String curso) { Curso = curso; }
public void setId(String id) { this.id = id; }
public void setNombres(String nombres) { Nombres = nombres; }
public void setApellidos(String apellidos) { Apellidos = apellidos; }
public void setTelefono(String telefono) { Telefono = telefono; }

```

Figura 3.65 Entidad Estudiantes.

Elaborado por: El investigador.

En la **Figura 3.66** se puede observar el método para asignar valores a la opción de cursos, para que el estudiante pueda seleccionarlo.

```

public void llenarSpinners(Estudiante estu){
    //Asignar variables
    String cur = estu.getCurso();

    //Dar opciones al spinner de cursos
    String[] opcionesCursos = {"Seleccione...", "Segundo", "Tercero", "Cuarto",
        "Quinto", "Sexto", "Séptimo", "Octavo", "Noveno", "Décimo"};
    ArrayAdapter<String> adapter = new ArrayAdapter<>(getContext(),
        R.layout.spinner_item_rolas, opcionesCursos);
    cursos.setAdapter(adapter);
    switch (cur){...}
}

```

Figura 3.66 Código para cargar opciones de cursos. Estudiantes.

Elaborado por: El investigador.

En la **Figura 3.67** se observa el método para mostrar la información del estudiante en los campos de texto correspondientes.

```

public void llenarDatos(){
    //Llenar los datos del profesor
    estudiante = referencia.estudiante;
    informacion.setVisibility(getView().VISIBLE);
    txtNombres.setText(estudiante.getNombres());
    txtApellidos.setText(estudiante.getApellidos());
    txtTelefono.setText(estudiante.getTelefono());

    //Llenar spinners
    llenarSpinners(estudiante);
}

```

Figura 3.67 Código para mostrar información del perfil. Estudiantes.

Elaborado por: El investigador.

Storycard – Actualizar perfil para estudiantes

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
21	Nuevo	4	4	4	4	Alta
Descripción: Actualizar la información del perfil para los estudiantes.						
Fecha	Acción			Comentario		
06/06/2022	Definición			Sin comentario		
26/06/2022	Implementación			Sin comentario		
26/06/2022	Ejecución			Sin comentario		
26/06/2022	Verificación			Sin comentario		

Tabla 3.29 Storycard de actualizar perfil para estudiantes.

Elaborado por: El investigador.

Para implementar la funcionalidad de actualizar la información del perfil para los estudiantes se creó un método el cual guarda la información nueva en un objeto de tipo “Map” y mediante los servicios de firebase se manda el objeto a la base de datos para que sean actualizados como se observa en la *Figura 3.68*.

```

public void guardarCambios(){
    //Asignar variables
    String id = estudiante.getId();
}

```

```

String tel = txtTelefono.getText().toString();
String cur = cursos.getSelectedItem().toString();

//Mapear la informacion
Map<String, Object> map = new HashMap<>();
map.put("tel_est", tel);
map.put("cur_est", cur);

//Actualizar datos
firestore.collection("Estudiantes").document(id).update(map).
    addOnSuccessListener(new OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void unused) {
            Toast.makeText(getContext(), "Actualizado exitosamente",
                Toast.LENGTH_SHORT).show();
            informacion.setVisibility(getView().INVISIBLE);
        }
    }).addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            Toast.makeText(getContext(), "Error al actualizar", Toast.LENGTH_SHORT).show();
        }
    });
}
}

```

Figura 3.68 Código para actualizar información del perfil. Estudiantes.

Elaborado por: El investigador.

Storycard – Cerrar sesión para estudiantes

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
22	Nuevo	3	3	3	3	Alta
Descripción: Cerrar sesión para los estudiantes.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
27/06/2022		Implementación			Sin comentario	
27/06/2022		Ejecución			Sin comentario	
27/06/2022		Verificación			Sin comentario	

Tabla 3.30 Storycard de cerrar sesión para estudiantes.

Elaborado por: El investigador.

Para la función de cerrar sesión de estudiantes se creó un método que cierra la conexión establecida con el servicio de autenticación de firebase como indica la **Figura 3.69**.

```
public void cerrarSesion(View view){
    FirebaseAuth.getInstance().signOut();
    tareaAdapter.stopListening();
    tareaManAdapter.stopListening();
    irLogin();
}
```

Figura 3.69 Código para cerrar sesión. Estudiantes.

Elaborado por: El investigador.

En la **Figura 3.70** se puede observar el método creado para mostrar la página de inicio luego del cierre de sesión.

```
public void irLogin(){
    Intent i = new Intent(EstudiantesInicioActivity.this, LoginActivity.class);
    //Evitar que se creen activitys innecesarios
    i.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP|Intent.FLAG_ACTIVITY_CLEAR_TASK
    |Intent.FLAG_ACTIVITY_NEW_TASK);
    //Iniciar nueva actividad
    startActivity(i);
}
```

Figura 3.70 Código para abrir página de inicio. Estudiantes.

Elaborado por: El investigador.

Storycard – Notificaciones para estudiantes

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
23	Nuevo	4	4	4	4	Alta
Descripción: Notificaciones para los estudiantes.						
Fecha		Acción			Comentario	
06/06/2022		Definición			Sin comentario	
28/06/2022		Implementación			Sin comentario	

01/07/2022	Ejecución	Sin comentario
01/07/2022	Verificación	Sin comentario

Tabla 3.31 Storycard de notificaciones para estudiantes.

Elaborado por: El investigador.

Para las notificaciones de los estudiantes se creó una clase que usa el servicio de notificaciones proporcionado por firebase, como indica la **Figura 3.71**.

```
public class NotificacionesPush extends FirebaseMessagingService {
    final String CHANNEL_ID = "Notificacion";

    @Override
    public void onMessageReceived(@NonNull RemoteMessage message) {
        super.onMessageReceived(message);
        if(message.getData().size() > 0){
            crearNotificacion();
        }
    }
}
```

Figura 3.71 Código de la clase para notificaciones. Estudiantes.

Elaborado por: El investigador.

En la **Figura 3.72** se puede observar el método creado para mostrar la notificación y abrir la aplicación al dar clic sobre la misma.

```
private void crearNotificacion(){
    //Notificaciones
    String con = "Mensaje";

    //Que abrir
    Intent intent = new Intent(this, LoginActivity.class);
    intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_CLEAR_TASK);
    PendingIntent pendingIntent = PendingIntent.getActivity(this, 0, intent, 0);

    //Notificacion
    NotificationCompat.Builder builder = new NotificationCompat.Builder(this, CHANNEL_ID)
        .setSmallIcon(R.drawable.ic_recodar)
        .setContentTitle("Unidad")
        .setContentText(con)
        .setPriority(NotificationCompat.PRIORITY_HIGH)
        .setContentIntent(pendingIntent);
}
```

```

if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
    CharSequence name = "channel_name";
    String description = "channel_description";
    int importance = NotificationManager.IMPORTANCE_DEFAULT;
    NotificationChannel channel = new NotificationChannel(CHANNEL_ID, name, importance);
    channel.setDescription(description);
    // Register the channel with the system; you can't change the importance
    // or other notification behaviors after this
    NotificationManager notificationManager = getSystemService(NotificationManager.class);
    notificationManager.createNotificationChannel(channel);
}

//Mostrar notificación
NotificationManagerCompat notificationManager = NotificationManagerCompat.from(this);
// notificationId is a unique int for each notification that you must define
notificationManager.notify(123, builder.build());
}

```

Figura 3.72 Código para mostrar la notificación. Estudiantes.

Elaborado por: El investigador.

3.2.4 Fase 4: Estabilización

Dentro de la fase de estabilización de la metodología se registran los cambios realizados para el desarrollo de la aplicación firebase Android.

El diseño y modelamiento de datos tuvo cambios debido a que se agregaron o modificaron tablas.

Los cambios registrados dentro del modelo de base de datos fueron los siguientes:

- **Tabla Estudiantes:** se modificó el tipo de dato de int a string, del campo curs_est(Curso del estudiante) y se agregó el campo de tipo string ace_est(Aceptación de cuenta).
- **Tabla Cursos:** se modificó el tipo de dato de int a string, del campo nom_cur(Nombre del curso).
- **Tabla Profesores:** se modificó el tipo de dato de int a string, del campo cur_pro(Curso del profesor) y se agregó el campo de tipo string ace_pro(Aceptación de cuenta).

- **Tabla Tareas:** se modificó el tipo de dato de int a string, del campo cur_tar(Curso al que pertenece la tarea) y se agregó el campo de tipo string tip_tar(Tipo de entrega).
- **Tabla Notificaciones:** se eliminó el campo tip_not(Tipo de notificación).
- Se creó la tabla auxiliar **Actualizar**, la cual guarda información para actualizar el campo est_tar(Estado de la tarea) de la tabla Tareas.

3.2.5 Fase 5: Pruebas

La fase de pruebas permite verificar y evaluar la correcta ejecución de las funciones implementadas para que el usuario interactúe con la aplicación firebase Android, esto con el objetivo de corregir errores y mejorar cada una de las funciones implementadas.

3.2.5.1 Pruebas de las funciones implementadas

Funcionalidades para el administrador

De acuerdo con las funciones establecidas del administrador, se muestran los resultados obtenidos de las pruebas realizadas en la aplicación firebase Android.

Inicio de sesión

Para el inicio de sesión se debe digitar el correo electrónico y la contraseña, como se indica en la *Figura 3.73*.



Figura 3.73 Inicio de sesión. Administrador.

Elaborado por: El investigador.

Menú

En el menú para el administrador se muestran las funciones que puede ejecutar, como se indica en la *Figura 3.74*.



Figura 3.74 Menú. Administrador.

Elaborado por: El investigador.

Aprobar o Rechazar profesores

Para aprobar o rechazar las cuentas de profesores, se visualiza una lista con los nombres de las posibles cuentas a registrarse y se muestran los botones para aprobar o rechazar la cuenta, de acuerdo con el criterio del administrador. El resultado de la prueba se puede observar en la *Figura 3.75*.

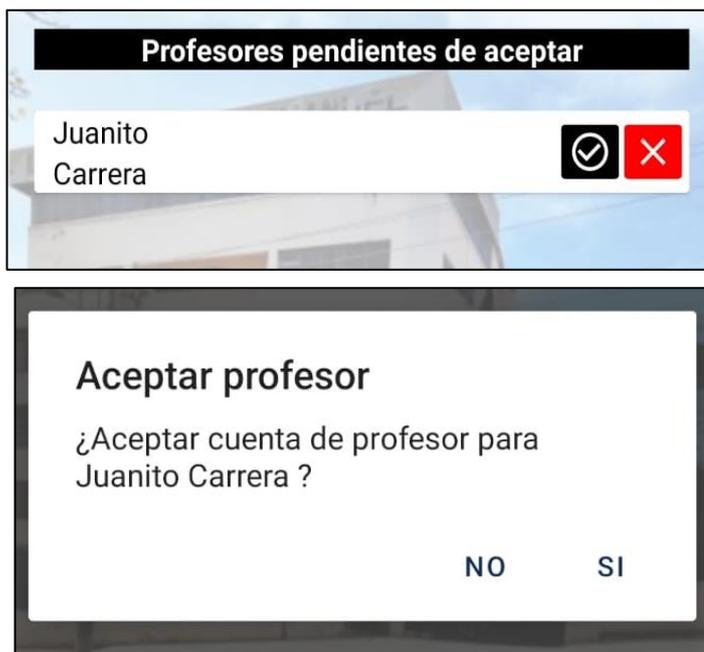


Figura 3.75 Aprobar o rechazar profesores. Administrador.

Elaborado por: El investigador.

Aprobar o Rechazar estudiantes

Para aprobar o rechazar las cuentas de estudiantes, se visualiza una lista con los nombres de las posibles cuentas a registrarse y se muestran los botones para aprobar o rechazar la cuenta, de acuerdo con el criterio del administrador. El resultado de la prueba se puede observar en la *Figura 3.76*.

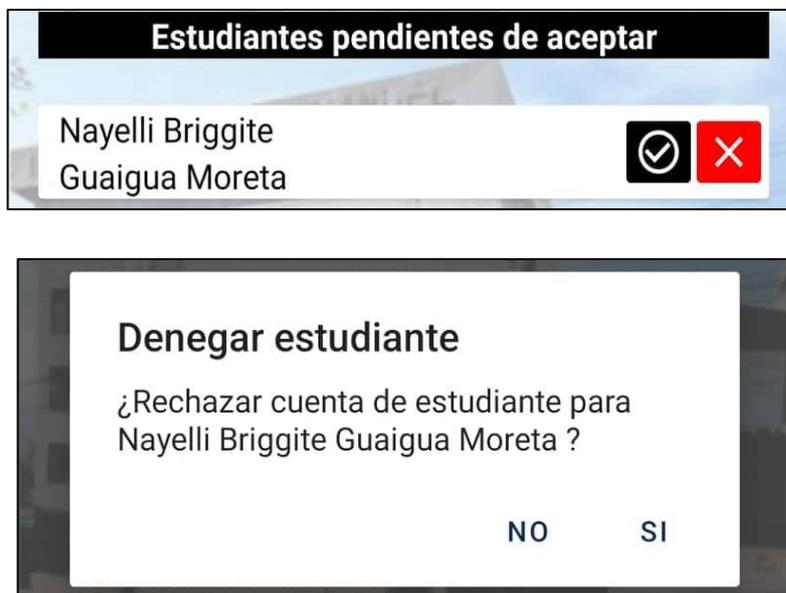


Figura 3.76 Aprobar o rechazar estudiantes. Administrador.

Elaborado por: El investigador.

Cerrar sesión

El administrador puede cerrar sesión al dar clic en la opción del menú y volver a la página de inicio, como se observa en la *Figura 3.77*.



Figura 3.77 Cerrar sesión. Administrador.

Elaborado por: El investigador.

Registro para estudiantes y profesores

Para el registro de cuenta de estudiantes y profesores se debe digitar la información para el inicio de sesión: correo electrónico y contraseña, así como la información personal nombres, apellidos y número de celular, y establecer un rol. El resultado de la prueba se indica en la *Figura 3.78*.



The image shows a mobile application registration form titled "Crear cuenta". At the top, there is a plus sign and a person icon. Below this is a green header with the text "Datos requeridos". The form contains several input fields: "Correo electronico:" with an envelope icon, "Contraseña:" with a password icon and an eye icon, "Nombres:" with a person icon, "Apellidos:" with a person icon, "Celular:" with a mobile phone icon, and "Rol:" with a dropdown menu showing "Selecciona una opción...". At the bottom, there is a dark blue button labeled "REGISTRAR CUENTA".

Figura 3.78 Registro para estudiantes y profesores.

Elaborado por: El investigador.

Restablecer contraseña para estudiantes y profesores

Para el restablecimiento de contraseña para estudiantes y profesores se debe digitar un correo electrónico, como se observa en la *Figura 3.79*.



Figura 3.79 Restablecer contraseña para estudiantes y profesores.

Elaborado por: El investigador.

Inicio de sesión para estudiantes y profesores

Para el inicio de sesión de estudiantes y profesores se debe digitar el correo electrónico y la contraseña, como se observa en la *Figura 3.80*.



Figura 3.80 Inicio de sesión para estudiantes y profesores.

Elaborado por: El investigador

Funcionalidades para profesores

De acuerdo con las funciones establecidas de los profesores, se muestran los resultados obtenidos de las pruebas realizadas en la aplicación firebase Android.

Menú

En el menú para los profesores se muestran las funciones que puede ejecutar, como se observa en la *Figura 3.81*.

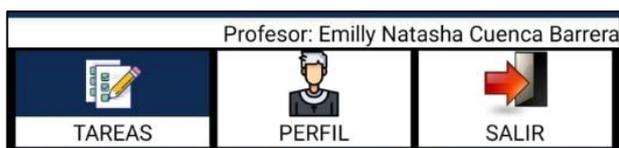


Figura 3.81 Menú. Profesores.

Elaborado por: El investigador.

Agregar nueva tarea

El profesor escribe el título de la tarea y las instrucciones para cumplir la misma, establece una fecha de entrega con el calendario desplegado (*Figura 3.82*) y selecciona el curso, la materia y el tipo de entrega. El resultado de la prueba se observa en la *Figura 3.83*.

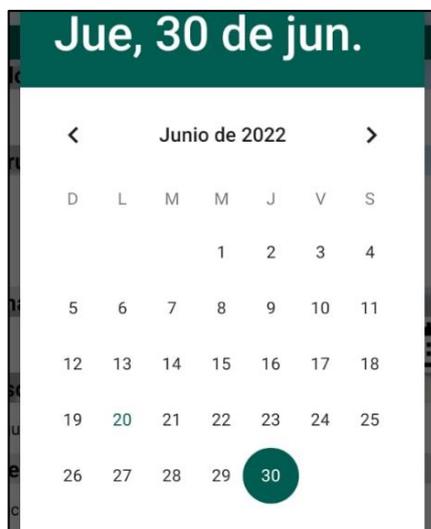


Figura 3.82 Calendario para establecer la fecha de entrega. Profesores.

Elaborado por: El investigador.

Agregar nueva tarea

Título:

Instrucciones:

Fecha de entrega:

Curso:
Segundo

Materia:
Lengua y literatura

Tipo de entrega:
Presencial

GUARDAR

Figura 3.83 Agregar tarea. Profesores.

Elaborado por: El investigador.

Ver lista de tareas pendientes

El profesor puede ver las tareas pendientes de entrega en una lista, así como los detalles específicos de las mismas: título, curso, fecha de entrega, materia, tipo de entrega e instrucciones, también se muestran los botones para ejecutar acciones específicas de la tarea. El profesor solo podrá ver las tareas que el haya creado. El profesor puede eliminar la tarea (*Figura 3.85*) si así lo requiere y editar la información de la tarea (*Figura 3.86*). El resultado de la prueba se puede observar en la *Figura 3.84*.

Lista de tareas pendientes

Leer

Segundo Hoy

Lengua y literatura

Virtual

Leer el libro de la página 5 a 10.

Figura 3.84 Lista de tareas pendientes. Profesores.

Elaborado por: El investigador.

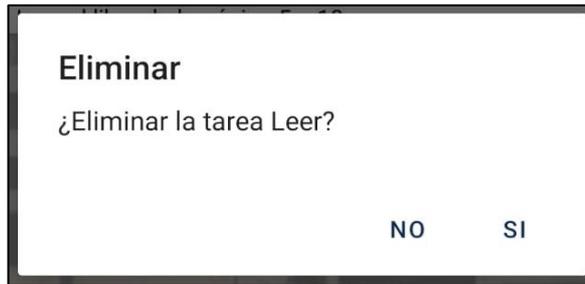


Figura 3.85 *Eliminar tarea. Profesores.*

Elaborado por: El investigador.

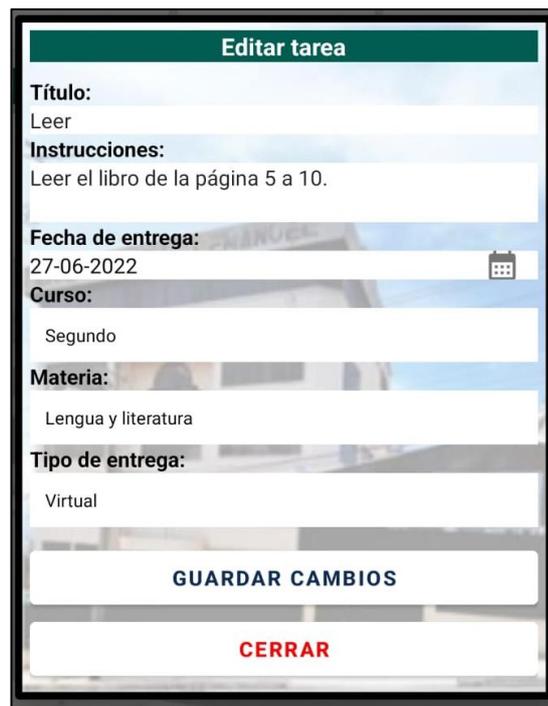


Figura 3.86 *Editar tarea. Profesores.*

Elaborado por: El investigador.

Revisar entregas

El profesor puede ver los archivos que han sido entregados en una lista con el nombre del alumno y la tarea respectiva, también puede visualizar el contenido del archivo(**Figura 3.88**) en una nueva ventana y así mismo descargar el archivo de la tarea(**Figura 3.89**) si así lo requiere. El resultado de la prueba se puede observar en la **Figura 3.87**.

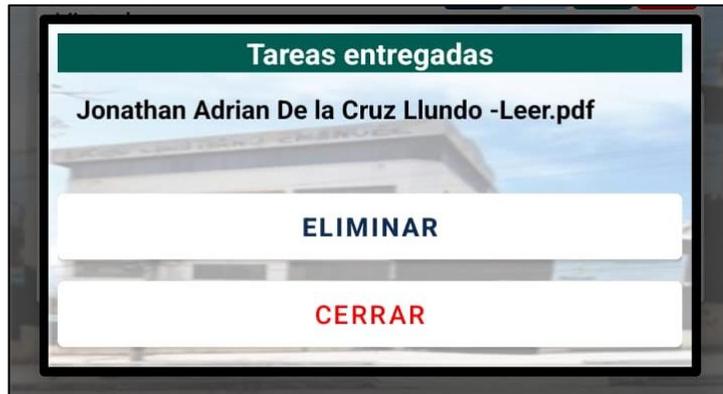


Figura 3.87 Revisar entregas. Profesores.

Elaborado por: El investigador.

Estudiante: Jonathan Adrian De la Cruz Llundo
Tarea: Leer

DESCARGAR SALIR

UNIVERSIDAD TÉCNICA DE AMBATO

Tema: Experimento de Joule-Ciclos de refrigeracion y de motor a combustion 2-3 y 4 tiempos
 Integrantes: Nicolay Cuenca - Diego De La Cruz
 Asignatura: Termodinámica
 Carrera: Ingeniería Industrial Curso: Tercero "A" Fecha: 12/06/2022

OBJETIVOS

- Comprender la importancia y relevancia que tiene el experimento de joule para la termodinámica
- Investigar el funcionamiento mediante ciclos en los motores de 2,3y 4 tiempos y sus características mas importantes
- Consultar el funcionamiento de los refrigeradores y cada componente que tiene este ciclo de refrigeración

Figura 3.88 Ver contenido de la entrega. Profesores.

Elaborado por: El investigador.



Figura 3.89 Descargar archivo entregado. Profesores.

Elaborado por: El investigador.

Recordar tarea

El profesor puede recordar una tarea a través de un correo electrónico(**Figura 3.91**) enviado a todos los estudiantes que pertenezcan al curso al cual se le asigno la tarea que se desea recordar. El resultado de la prueba se observa en la **Figura 3.90**.

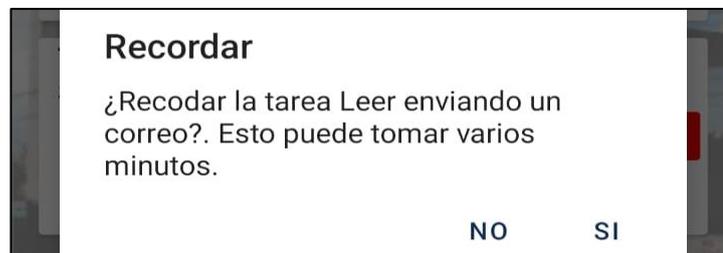


Figura 3.90 Recordar tarea. Profesores.

Elaborado por: El investigador.

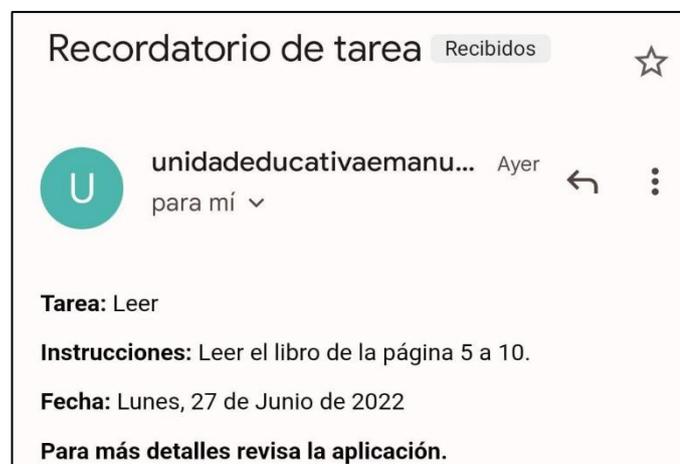
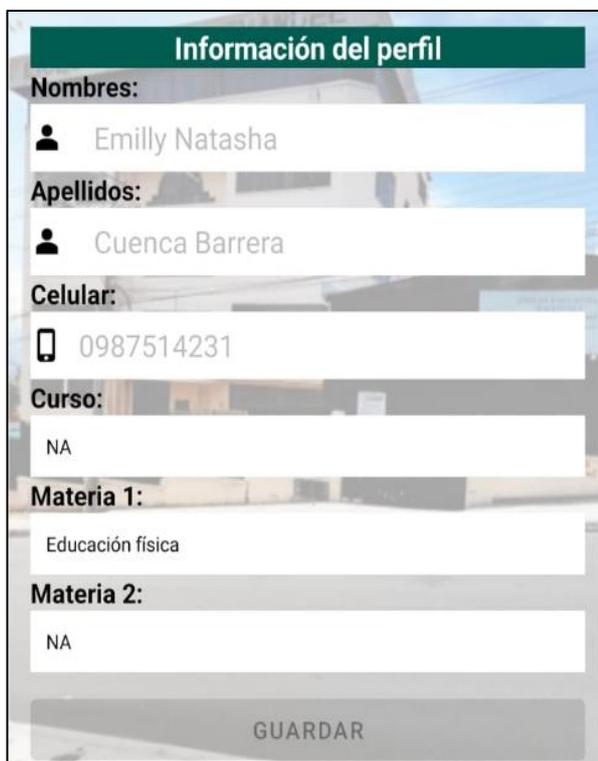


Figura 3.91 Correo enviado al estudiante. Profesores.

Elaborado por: El investigador.

Ver perfil

El profesor puede ver sus nombres, apellidos y número de teléfono datos asignados al momento de crear la cuenta, también puede ver el curso en el que dicta clase si fuere el caso, la o las materias que dicta siempre y cuando ya hayan sido asignadas. El resultado de la prueba se observa en la **Figura 3.92**



Información del perfil	
Nombres:	Emilly Natasha
Apellidos:	Cuenca Barrera
Celular:	0987514231
Curso:	NA
Materia 1:	Educación física
Materia 2:	NA
GUARDAR	

Figura 3.92 Información del perfil. Profesores.

Elaborado por: El investigador.

Actualizar perfil

El profesor puede actualizar su número de teléfono si lo requiere, los datos de nombres y apellidos no están disponibles para ser actualizados. Puede actualizar un nuevo curso en el que dicta clases o establecer uno, sino lo tuviera asignado, igualmente puede hacer lo mismo para la materia, o las materias que dicte y que requieran de actualización. En la **Figura 3.93** se puede observar el resultado de la ejecución.

Información del perfil

Nombres:
Emily Natasha

Apellidos:
Cuenca Barrera

Celular:
0987514231

Curso:
NA

Materia 1:
Educación física

Materia 2:
NA

GUARDAR

Figura 3.93 Actualizar información del perfil. Profesores.

Elaborado por: El investigador.

Cerrar sesión

El profesor puede cerrar sesión dentro de las opciones del menú, lo cual lo lleva a la página de inicio de sesión, como indica la *Figura 3.94*.



Figura 3.94 Cerrar sesión. Profesores.

Elaborado por: El investigador.

Funcionalidades para estudiantes

De acuerdo con las funciones establecidas de los estudiantes, se muestran los resultados obtenidos de las pruebas realizadas en la aplicación firebase Android.

Menú

En el menú para los estudiantes se muestran las funciones que puede ejecutar, como se observa en la *Figura 3.95*.



Figura 3.95 Menú. Estudiantes.

Elaborado por: El investigador.

Ver lista de tareas pendientes

El estudiante puede ver las tareas pendientes de entrega en una lista con datos reducidos pero relevantes, además si así lo requiere tiene la opción de ver todos los detalles: título, instrucciones, fecha de entrega y materia de cada una de las tareas. El estudiante solo puede ver las tareas que estén asignadas para el curso al cual pertenece. El resultado de la prueba se observa en la *Figura 3.96*.

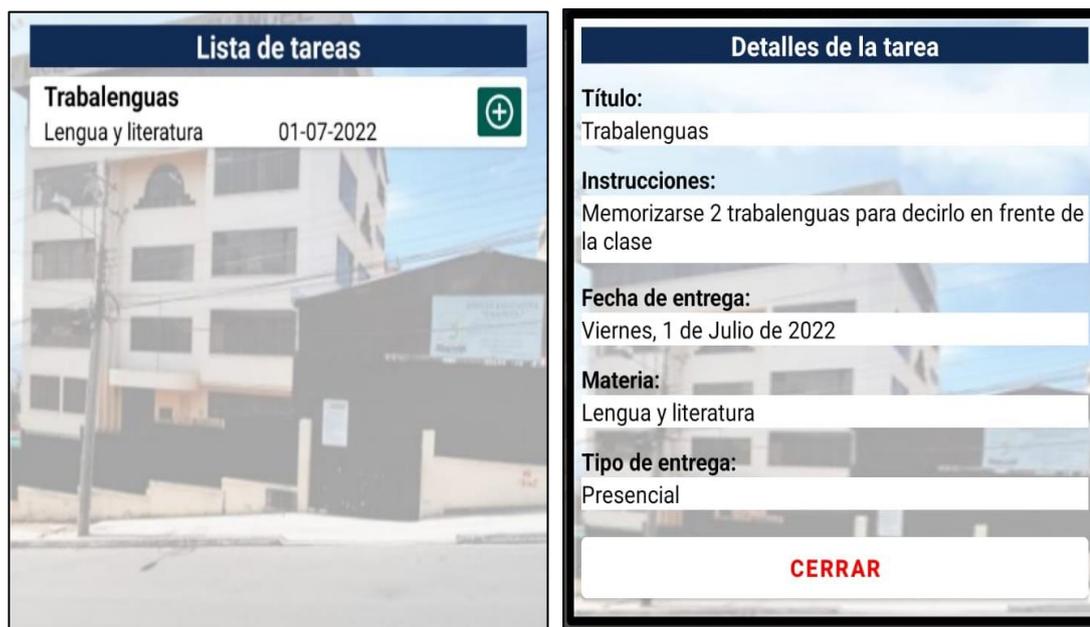
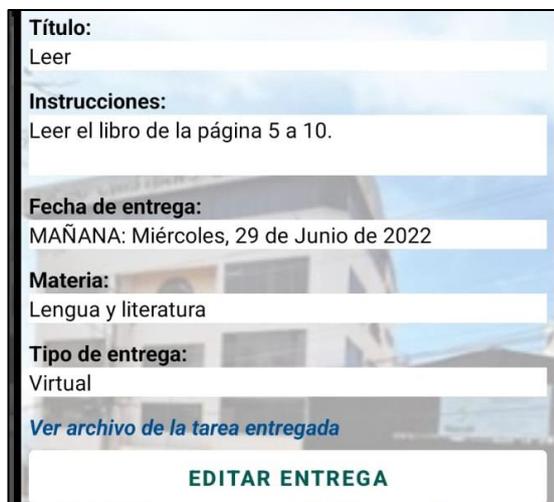


Figura 3.96 Ver tareas pendientes. Estudiantes.

Elaborado por: El investigador.

Entregar tareas

El estudiante puede enviar un archivo si la tarea así lo requiere y si anteriormente ya ha enviado el archivo puede editar la entrega. Como indica la **Figura 3.97**.



The screenshot shows a task submission form with the following fields and values:

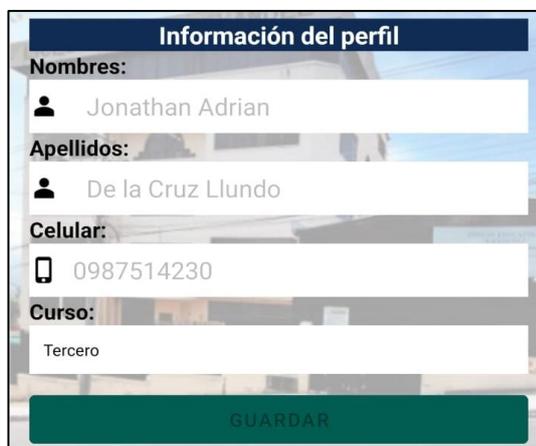
- Título:** Leer
- Instrucciones:** Leer el libro de la página 5 a 10.
- Fecha de entrega:** MAÑANA: Miércoles, 29 de Junio de 2022
- Materia:** Lengua y literatura
- Tipo de entrega:** Virtual
- [Ver archivo de la tarea entregada](#)
- EDITAR ENTREGA** (button)

Figura 3.97 Entregar tarea. Estudiantes.

Elaborado por: El investigador.

Ver perfil

El estudiante puede ver sus nombres, apellidos y número de teléfono datos asignados al momento de crear la cuenta, también puede ver el curso al que pertenece. El resultado de la prueba se observa en la **Figura 3.98**.



The screenshot shows a profile information form with the following fields and values:

- Información del perfil** (header)
- Nombres:** Jonathan Adrian
- Apellidos:** De la Cruz Llundo
- Celular:** 0987514230
- Curso:** Tercero
- GUARDAR** (button)

Figura 3.98 Información del perfil. Estudiantes.

Elaborado por: El investigador.

Actualizar perfil

El estudiante puede actualizar su número de teléfono si lo requiere, los datos de nombres y apellidos no están disponibles para ser actualizados. Puede actualizar el curso al que pertenece. En la **Figura 3.99** se puede observar el resultado de la ejecución.



Información del perfil	
Nombres:	Jonathan Adrian
Apellidos:	De la Cruz Llundo
Celular:	0987514230
Curso:	Tercero
GUARDAR	

Figura 3.99 Actualizar información del perfil. Estudiantes.

Elaborado por: El investigador.

Cerrar sesión

El estudiante puede cerrar sesión dentro de las opciones del menú, lo cual lo lleva a la página de inicio de sesión, como indica la **Figura 3.100**.



Figura 3.100 Cerrar sesión. Estudiantes.

Elaborado por: El investigador.

Notificaciones

El estudiante recibe dos notificaciones a lo largo del día. La primera notificación se recibe en la tarde después de clases, dicha notificación recuerda al estudiante el realizar las tareas que tiene pendientes (**Figura 3.101**). La segunda notificación se envía en la

noche, la cual recuerda al alumno preparar las tareas para su presentación(*Figura 3.102*).

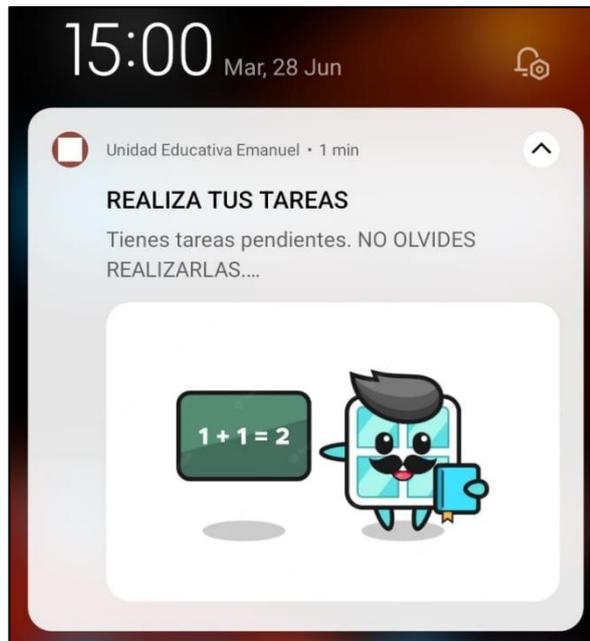


Figura 3.101 Notificación uno. Estudiantes.

Elaborado por: El investigador.



Figura 3.102 Notificación dos. Estudiantes.

Elaborado por: El investigador.

3.2.5.2 Pruebas de aceptación

De acuerdo con las pruebas realizadas sobre las funciones de la aplicación firebase Android, se obtuvo las siguientes pruebas de aceptación.

Prueba de aceptación	Número: 1
N° de storycard: 1	
Nombre: Inicio de sesión del administrador	
Descripción: La función permite al administrador autenticarse e ingresa a las demás funciones de la aplicación.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con los campos de texto para el correo electrónico y la contraseña. El botón permite autenticar al administrador si los datos son correctos.	
Resultado esperado: Si los datos ingresados son correctos, se accede a la página de inicio del administrador para ejecutar las demás funciones.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.32 Prueba de aceptación 1.

Elaborado por: El investigador.

Prueba de aceptación	Número: 2
N° de storycard: 2	
Nombre: Aprobar o rechazar profesores del administrador.	
Descripción: La función permite al administrador aprobar o rechazar las cuentas que se quieran registrar con el rol de “Profesor”.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con una lista la cual contiene los nombres y apellidos de los usuarios. Además de la lista se muestran dos botones por cada uno de los usuarios de la lista mostrada, para que el administrador pueda aprobar o rechazar el usuario.	
Resultado esperado: Al dar clic en el botón “Aprobar” se muestra un cuadro de diálogo para que el administrador confirme que desea aprobar la cuenta seleccionada, al dar clic en el botón “Rechazar” se muestra un cuadro de diálogo para que el administrador confirme que desea rechazar la cuenta seleccionada.	

Resultado de la prueba: Satisfactorio.

Tabla 3.33 Prueba de aceptación 2.

Elaborado por: El investigador.

Prueba de aceptación	Número: 3
N° de storycard: 3	
Nombre: Aprobar o rechazar estudiantes del administrador.	
Descripción: La función permite al administrador aprobar o rechazar las cuentas que se quieran registrar con el rol de “Estudiante”.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con una lista la cual contiene los nombres y apellidos de los usuarios. Los botones permiten aprobar o rechazar el usuario.	
Resultado esperado: Al dar clic en el botón “Aprobar” se muestra un cuadro de diálogo para que el administrador confirme que desea aprobar la cuenta seleccionada, al dar clic en el botón “Rechazar” se muestra un cuadro de diálogo para que el administrador confirme que desea rechazar la cuenta seleccionada.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.34 Prueba de aceptación 3.

Elaborado por: El investigador.

Prueba de aceptación	Número: 4
N° de storycard: 4	
Nombre: Cerrar sesión del administrador.	
Descripción: La función permite al administrador cerrar la sesión de su cuenta.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con el botón para cerrar sesión.	
Resultado esperado: El administrador da clic al botón “Salir” y se cierra la sesión.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.35 Prueba de aceptación 4.

Elaborado por: El investigador.

Prueba de aceptación	Número: 5
N° de storycard: 5	
Nombre: Registro para estudiantes y profesores.	
Descripción: La función permite al usuario ingresar los datos para crear la cuenta.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con los campos de texto respectivos para ingresar la información. El botón permite al usuario crear la cuenta si los datos son correctos.	
Resultado esperado: Si los datos son correctos, se guarda la información en la base de datos.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.36 Prueba de aceptación 5.

Elaborado por: El investigador

Prueba de aceptación	Número: 6
N° de storycard: 6	
Nombre: Restablecer contraseña para estudiantes y profesores.	
Descripción: La función permite al usuario recibir un enlace al correo electrónico mediante el cual se puede restablecer la contraseña de la cuenta.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con el campo de texto para ingresar el correo electrónico. El botón permite enviar el enlace para restablecer la contraseña al correo asignado.	
Resultado esperado: Se envía un enlace al correo electrónico el cual permite restablecer la contraseña.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.37 Prueba de aceptación 6.

Elaborado por: El investigador.

Prueba de aceptación	Número: 7
N° de storycard: 7	
Nombre: Inicio de sesión para estudiantes y profesores.	

Descripción: La función permite al usuario autenticarse e ingresar a las demás funciones de la aplicación.
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con los campos de texto para el correo electrónico y la contraseña. El botón permite autenticar el usuario si los datos son correctos.
Resultado esperado: Si los datos ingresados son correctos, se accede a la página de inicio para ejecutar las demás funciones, de acuerdo con el rol del usuario.
Resultado de la prueba: Satisfactorio.

Tabla 3.38 Prueba de aceptación 7.

Elaborado por: El investigador.

Prueba de aceptación	Número: 8
N° de storycard: 8	
Nombre: Agregar nueva tarea para profesores.	
Descripción: La función permite al profesor ingresar los datos requeridos para una tarea y crearla.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con los campos de texto para título e instrucciones, un calendario para establecer la fecha de entrega y los campos de selección para curso, materia y tipo de entrega. El botón permite guardar la tarea en la base de datos.	
Resultado esperado: El profesor digita el título y las instrucciones, escoge una fecha de entrega en el calendario, escoge un curso, una materia y un tipo de entrega y al dar clic en el botón de “Agregar” la información se guarda en la base de datos.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.39 Prueba de aceptación 8.

Elaborado por: El investigador.

Prueba de aceptación	Número: 9
N° de storycard: 9	
Nombre: Ver tareas pendientes para profesores.	

Descripción: La función permite al profesor ver todas las tareas pendientes de entrega.
Condiciones de ejecución e interfaz: Se visualiza una lista con la información de las tareas pendientes. Los botones permiten eliminar la tarea o editar información de la misma.
Resultado esperado: El profesor visualiza una lista con todos los detalles de la tarea pendiente, al dar clic en el botón “Eliminar” la tarea se elimina y al dar clic en el botón “Editar” se muestra la información de la tarea con la opción de poder editar dicha información y actualizarla.
Resultado de la prueba: Satisfactorio.

Tabla 3.40 Prueba de aceptación 9.

Elaborado por: El investigador.

Prueba de aceptación	Número: 10
Nº de storycard: 10	
Nombre: Revisar entregas para profesores.	
Descripción: La función permite al profesor ver los archivos de las tareas entregadas.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con una lista de los archivos entregados. El botón permite descargar el archivo.	
Resultado esperado: El profesor visualiza una lista con los archivos entregados, al dar clic en el nombre del archivo se abre una ventana en la cual se visualiza el contenido. Al dar clic en el botón “Descargar” se descarga el archivo.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.41 Prueba de aceptación 10.

Elaborado por: El investigador.

Prueba de aceptación	Número: 11
Nº de storycard: 11	
Nombre: Recordar tareas para profesores.	

Descripción: La función permite al profesor recordar una tarea a través del correo electrónico.
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con la lista de tareas pendientes. El botón permite enviar el correo.
Resultado esperado: El profesor visualiza una lista con la información de las tareas pendientes, al dar clic en el botón “Recordar” aparece un cuadro de diálogo donde el profesor confirma que quiere recordar la tarea seleccionada, al dar clic en “Confirmar” se envía un correo con información de la tarea: título, instrucciones y fecha de entrega, a los estudiantes.
Resultado de la prueba: Satisfactorio.

Tabla 3.42 Prueba de aceptación 11.

Elaborado por: El investigador.

Prueba de aceptación	Número: 12
Nº de storycard: 12	
Nombre: Actualizar y Editar perfil para profesores.	
Descripción: La función permite al profesor ver la información de su perfil y actualizarla si fuere necesario.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con los campos de texto para los nombres, apellidos y teléfono, además de los campos de selección para el curso, la materia uno y la materia dos. El botón permite actualizar la información en la base de datos.	
Resultado esperado: El profesor visualiza sus datos en los campos de texto correspondientes, al dar clic en el botón “Actualizar” se habilitan los campos que están disponibles para editar, se ingresa los datos nuevos y al dar clic en el botón “Guardar” se actualizan los campos.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.43 Prueba de aceptación 12.

Elaborado por: El investigador.

Prueba de aceptación	Número: 13
N° de storycard: 13	
Nombre: Cerrar sesión para profesores.	
Descripción: La función permite al profesor cerrar la sesión de su cuenta.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con el botón para cerrar sesión.	
Resultado esperado: El profesor da clic en el botón “Salir”, se cierra la sesión y se muestra la página de inicio de sesión.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.44 Prueba de aceptación 13.

Elaborado por: El investigador.

Prueba de aceptación	Número: 14
N° de storycard: 14	
Nombre: Ver tareas pendientes para estudiantes.	
Descripción: La función permite al estudiante ver todas las tareas pendientes de entrega.	
Condiciones de ejecución e interfaz: Se visualiza una lista con la información de las tareas pendientes. El botón permite visualizar todos los detalles de la tarea.	
Resultado esperado: El estudiante visualiza una lista con información básica de la tarea: título, fecha de entrega y materia. Al dar clic en el botón “Más” se visualiza todos los detalles de la tarea seleccionada.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.45 Prueba de aceptación 14.

Elaborado por: El investigador.

Prueba de aceptación	Número: 15
N° de storycard: 15	
Nombre: Entregar tareas para estudiantes.	

Descripción: La función permite al estudiante entregar los archivos para cumplir con una determinada tarea.
Condiciones de ejecución e interfaz: Se visualiza una lista con la información de las tareas pendientes. Si las condiciones de la tarea son las correctas se visualiza el botón correspondiente para subir un archivo y así cumplir con la entrega de la tarea.
Resultado esperado: El estudiante visualiza una lista con información básica de la tarea: título, fecha de entrega y materia. Si las condiciones de la tarea son las correctas se habilita el botón que al dar clic se abre el explorador de archivos para que el estudiante escoja el archivo correspondiente y suba la tarea a la base de datos. Además, si el estudiante ya entregó la tarea puede editarla y visualizar la información del archivo entregado.
Resultado de la prueba: Satisfactorio.

Tabla 3.46 Prueba de aceptación 15.

Elaborado por: El investigador.

Prueba de aceptación	Número: 16
Nº de storycard: 16	
Nombre: Actualizar y Editar perfil para estudiantes.	
Descripción: La función permite al estudiante ver la información de su perfil y actualizarla si fuere necesario.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con los campos de texto para los nombres, apellidos y teléfono, además del campo de selección para el curso. El botón permite actualizar la información en la base de datos.	
Resultado esperado: El estudiante visualiza sus datos en los campos de texto correspondientes, al dar clic en el botón “Actualizar” se habilitan los campos que están disponibles para editar, se ingresan los datos nuevos y al dar clic en el botón “Guardar” se actualizan los campos.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.47 Prueba de aceptación 16.

Elaborado por: El investigador

Prueba de aceptación	Número: 17
N° de storycard: 17	
Nombre: Cerrar sesión para estudiantes.	
Descripción: La función permite al estudiante cerrar la sesión de su cuenta.	
Condiciones de ejecución e interfaz: Se visualiza la interfaz gráfica con el botón para cerrar sesión.	
Resultado esperado: El estudiante da clic en el botón “Salir”, se cierra la sesión y se muestra la página de inicio de sesión.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.48 Prueba de aceptación 17.

Elaborado por: El investigador.

Prueba de aceptación	Número: 18
N° de storycard: 18	
Nombre: Notificaciones para estudiantes.	
Descripción: La función permite al estudiante recibir notificaciones.	
Condiciones de ejecución e interfaz: Se visualiza la barra principal del dispositivo donde se ven las notificaciones.	
Resultado esperado: El estudiante recibe una primera notificación en la tarde la cual le recuerda que debe realizar sus tareas. La segunda notificación se recibe en la noche la cual recuerda al estudiante preparar las tareas que ha realizado para que las pueda entregar, cumpliendo así con las misma.	
Resultado de la prueba: Satisfactorio.	

Tabla 3.49 Prueba de aceptación 18.

Elaborado por: El investigador.

3.2.6 Implantación de la aplicación firebase Android

Una vez terminadas las cinco fases de la metodología Mobile – D, se procedió a implantar la solución dentro de la Unidad Educativa Emanuel.

Instalación

Para la instalación de la aplicación se procedió a subir el archivo .apk, al almacenamiento en la nube de drive, para que mediante un enlace los usuarios de la aplicación(Administrador, Profesores y Estudiantes) puedan descargarlo e instalarlo en sus dispositivos móviles. Como se observa en la **Figura 3.103**.

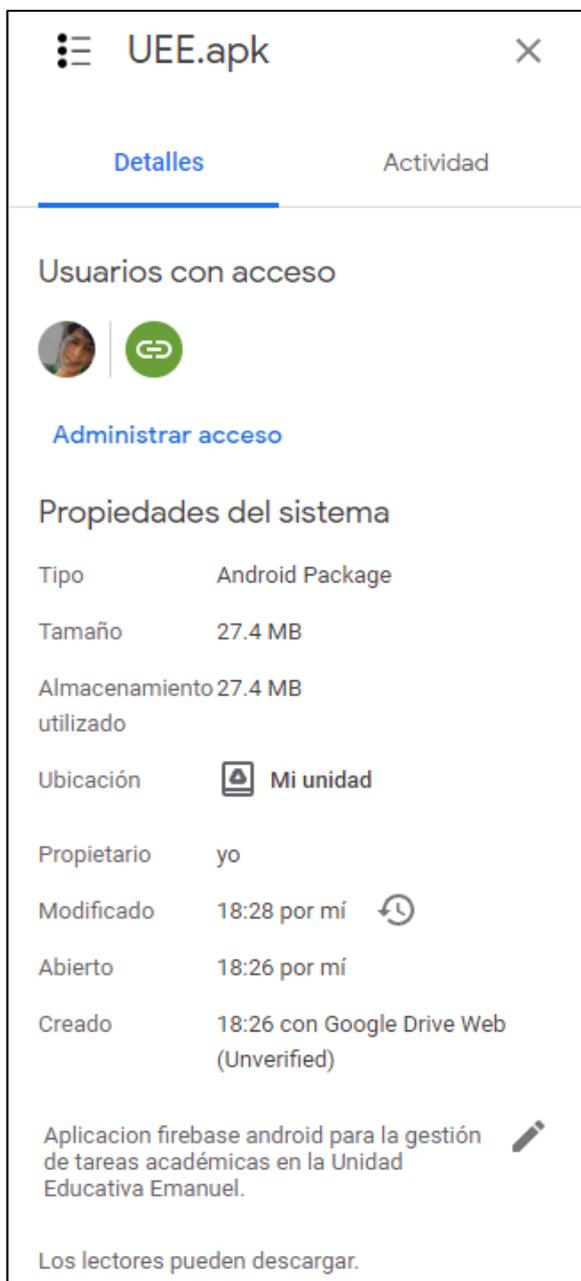


Figura 3.103 APK de instalación en la nube.

Elaborado por: El investigador.

Proceso de descarga e instalación

El usuario accede al enlace y descarga el archivo de instalación de la aplicación firebase Android, como se puede observar en la **Figura 3.104**.



Figura 3.104 Descarga del archivo de instalación.

Elaborado por: El investigador.

Una vez descargado el archivo el usuario debe abrir el archivo descargado para empezar con la instalación, como indica la **Figura 3.105**.

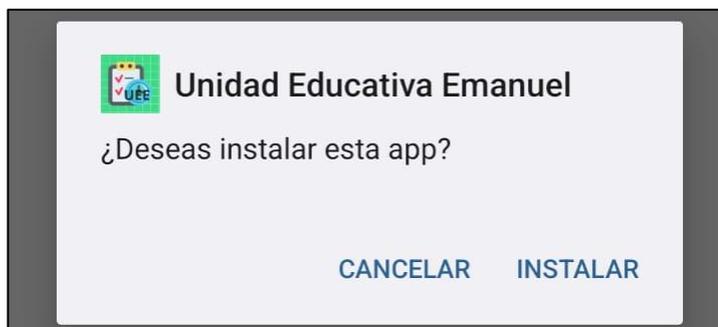


Figura 3.105 Instalación de la aplicación firebase Android.

Elaborado por: El investigador.

La aplicación ha sido instalada y se puede usar, como indica la **Figura 3.16**.



Figura 3.106 Aplicación instalada.

Elaborado por: El investigador.

Capacitación

Se desarrollaron las jornadas de capacitación a los usuarios de la aplicación; donde se explicó las funciones que pueden realizar al hacer uso de la aplicación firebase Android para la gestión de las tareas académicas, el cronograma de capacitación se muestra en la *Tabla 3.50*.

Tema	Instructor	Usuarios	Fecha
Inicio de sesión	Adrian De la Cruz	Profesores, Estudiantes	12/07/2022
Registro de cuenta	Adrian De la Cruz	Profesores, Estudiantes	12/07/2022
Restablecer contraseña	Adrian De la Cruz	Profesores, Estudiantes	12/07/2022
Cerrar sesión	Adrian De la Cruz	Profesores, Estudiantes, Administrador	12/07/2022
Aprobar o Rechazar cuentas	Adrian De la Cruz	Administrador	11/07/2022
Ver tareas pendientes	Adrian De la Cruz	Profesores	12/07/2022
Agregar tareas	Adrian De la Cruz	Profesores	12/07/2022
Ver entregas	Adrian De la Cruz	Profesores	12/07/2022
Recordar tareas	Adrian De la Cruz	Profesores	12/07/2022
Ver tareas pendientes	Adrian De la Cruz	Estudiantes	13/07/2022
Entregar tareas	Adrian De la Cruz	Estudiantes	13/07/2022
Editar entregas	Adrian De la Cruz	Estudiantes	13/07/2022

Tabla 3.50 Cronograma de capacitación.

Elaborado por: El investigador.

Además de la capacitación a los usuarios se entregó un manual con la explicación de todas las funciones (Anexo A).

Pruebas de rendimiento

Después de la implantación de la aplicación, haciendo uso del servicio de análisis de Firebase, se realizaron las pruebas de rendimiento.

En la **Figura 3.107** se puede observar la estabilidad de la aplicación y la ejecución sin fallos por parte de los usuarios, se puede evidenciar que la aplicación tiene una estabilidad del 100%.



Figura 3.107 Estabilidad de la aplicación.

Elaborado por: El investigador.

En la **Figura 3.108** se indica que el tiempo promedio que los usuarios usan la aplicación es de 24 minutos y 33 segundos.



Figura 3.108 Tiempo de uso promedio.

Elaborado por: El investigador.

En la **Figura 3.109** se puede observar las visitas que tienen cada una de las interfaces de la aplicación, siendo la más visitada el Login.

Visitas por Título de página y clase de pantalla

TÍTULO DE PÁGINA Y CL...		VISITAS
LoginActivity	1,2 mil	↑ 296,6 %
ProfesoresInicioActivity	516	-
EstudiantesInicioActivity	221	-
RegistrarActivity	64	↓ 34,7 %
InicioActivity	67	↑ 131,0 %
RecuperarContrasenaActivity	51	↑ 75,9 %
AdministradorActivity	50	-

[Ver páginas y pantallas →](#)

Figura 3.109 Interfaces más visitadas.

Elaborado por: El investigador.

CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- La recolección de datos y su posterior análisis permitió definir de manera clara las dificultades a la que se enfrentan los profesores y alumnos en el proceso de envío y recibo de tareas, existiendo dos puntos clave que dificultan este proceso: el primero se da cuando el alumno no lleva un registro de las tareas que debe cumplir, por ende, no las realiza; el segundo se da cuando el alumno ya realizó la tarea, pero se olvida entregarla.
- La plataforma firebase fue una herramienta clave a la hora de desarrollar el proyecto porque el uso de sus servicios de autenticación, almacenamiento y notificaciones se lo realizó de manera rápida y sencilla, ya que no hay necesidad de un código específico de servidor ayudando a reducir código y tiempos de desarrollo.
- El almacenamiento en la nube utilizando firebase garantizó que la información para los usuarios de la aplicación este siempre disponible y segura, firebase al ser propiedad de Google trabaja bajo sus mismos estándares, lo que convierte a esta base de datos en una de las más flexibles y seguras a nivel mundial.
- El desarrollo e implementación de la aplicación firebase Android ayuda a reducir el índice de incumplimiento de tareas escolares en la Unidad Educativa Emanuel, en vista que los profesores tienen un mejor control de las actividades académicas y siendo una herramienta válida para mejorar la interacción entre el docente y el alumno.

4.2 Recomendaciones

- Se recomienda actualizar los servicios que la plataforma firebase ofrece para así poder adaptarlos a la aplicación teniendo mejoras continuas en la misma.
- Para que la información dentro de la aplicación se actualice de manera instantánea es recomendable usar el método `addSnapshotListener()` en lugar del método `get()`, ya que esto permite actualizar la información sin necesidad

de volver a ejecutar el método para obtener los datos y cargándolos de nuevo en la interfaz gráfica.

- Al mostrar los datos en la interfaz gráfica se recomienda usar una clase de tipo “Adapter” exclusiva para este fin, ya que mediante el uso de esta clase se tiene un mejor manejo de la información con código reducido, de manera fácil y rápida.
- Se recomienda a los profesores y alumnos usar la aplicación para recodar las tareas y sus instrucciones, ayudando a su cumplimiento de manera correcta garantizando la adquisición de nuevos conocimientos.

BIBLIOGRAFÍA

- [1] X. Bonal and S. González, “Confinamiento y efecto escuela.” <https://www.elperiodico.com/es/opinion/20200406/efecto-coronavirus-desigualdad-escuelas-xavier-bonal-sheila-gonzalez-7919442>.
- [2] J. L. Muñoz and L. Lluch, “Educación y Covid-19: Colaboración de las Familias y Tareas Escolares”, [Online]. Available: www.rinace.net/riejs/revistas.uam.es/riejs
- [3] M. Pantoja, N. Lucero, S. Álvarez, and J. Enríquez, “Educación y pandemia: desafío para los docentes de educación básica superior y bachillerato de la ciudad de Ibarra, Ecuador.” http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1990-86442021000400307&lang=es .
- [4] M. Tonato, “Tareas extraescolares en la actitud de aprendizaje de los niños del nivel preparatoria de la Unidad Educativa ‘Iberoamérica,’” Ambato, 2020. [Online]. Available: <https://repositorio.uta.edu.ec/jspui/handle/123456789/31053>
- [5] G. Cardiel Altemir, “Desarrollo de una aplicación Fitness en el sistema operativo Android con la plataforma Firebase,” Lérida, 2019.
- [6] M. Castellote García, “Desarrollo de una aplicación Android de apuestas utilizando Firebase para la sincronización de datos,” Universitat Jaume I, Castello, 2017. [Online]. Available: <http://repositori.uji.es/xmlui/handle/10234/174192>
- [7] M. Jiménez García and M. de los Á. Martínez Ortega, “El Uso de una Aplicación Móvil en la Enseñanza de la Lectura,” *Información tecnológica*, vol. 28, no. 1, pp. 151–160, Feb. 2017, doi: 10.4067/S0718-07642017000100015.
- [8] J. Bermeo Urgiles, “Análisis de los factores asociados al incumplimiento de tareas escolares, en cuarto año de educación básica, de la escuela ‘Juan León Mera’, Parroquia Molleturo,” Cuenca, 2018.
- [9] R. Navarrete and A. Rodríguez, “El diario escolar electrónico para la gestión del control de tareas,” Guayaquil, 2018.

- [10] D. Ramos Cardozzo, *Desarrollo de Software: Requisitos, Estimaciones y Análisis.*, Segunda. 2016. [Online]. Available: <https://books.google.es/books?hl=es&lr=&id=tBaYCwAAQBAJ&oi=fnd&pg=PA1&dq=%22desarrollo+de+software%22&ots=uGsZuf7CI2&sig=Z8qpk4six65yCDO5LJpUeqQORlg#v=onepage&q=%22desarrollo%20de%20software%22&f=false>
- [11] J. P. Zumba Gamboa and C. A. León Arreaga, “Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software.,” *INNOVA Research Journal*, vol. 3, no. 10, pp. 20–33, 2018.
- [12] L. M. Delía, “Desarrollo de aplicaciones móviles multiplataforma,” La Plata, 2017. [Online]. Available: <https://digital.cic.gba.gob.ar/handle/11746/6601>
- [13] R. Ramírez Vique, *Métodos para el desarrollo de aplicaciones móviles*. 2019.
- [14] J. D. Luján Castillo, *Desarrollo de aplicaciones Android con Android Studio: Conoce Android Studio - José Dimas Luján Castillo - Google Libros*. 2019. [Online]. Available: <https://books.google.es/books?hl=es&lr=&id=i96LDwAAQBAJ&oi=fnd&pg=PA14&dq=desarrollo+android&ots=kW3lTxZSNC&sig=ZsBip6Hfab3x2S6IA-11XmTU6Eo#v=onepage&q=desarrollo%20android&f=false>
- [15] E. González Suárez, “Conocimiento científico e información científica,” *ACIMED*, vol. 14, no. 6, pp. 0–0, 2006. [Online]. Available: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1024-94352006000600003&lng=es&nrm=iso&tlng=es
- [16] A. Márquez Jiménez, “Educación y desarrollo en la sociedad del conocimiento.,” *Perfiles Educativos*, vol. 39, no. 158, pp. 3–17, 2017. [Online]. Available: <http://www.scielo.org.mx/pdf/peredu/v39n158/0185-2698-peredu-39-158-00003.pdf>
- [17] A. R. Jiménez and A. O. P. Jacinto, “Métodos científicos de indagación y de construcción del conocimiento,” *Revista Escuela de Administración de Negocios*, no. 82, pp. 175–195, Jul. 2017, doi: 10.21158/01208160.N82.2017.1647.

- [18] M. R. Belando-Montoro, “Aprendizaje a lo largo de la vida. Concepto y componentes,” *Revista Iberoamericana de Educación*, vol. 75, pp. 219–234, 2017.
- [19] Y. Abreu Alvarado, A. D. Barrera Jiménez, W. Taymí Breijo, and I. Bonilla Vichot, “El proceso de enseñanza-aprendizaje de los Estudios Lingüísticos: su impacto en la motivación hacia el estudio de la lengua,” *Mendive. Revista de Educación*, vol. 16, no. 4, pp. 610–623, Oct. 2018. [Online]. Available: <https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/1462/html>
- [20] L. Freire, B. Rodríguez, and R. Martínez, “Padres y madres ante las tareas escolares: La visión del profesorado,” *Aula Abierta*, vol. 48, no. 1, pp. 77–84, 2019.
- [21] E. G. Pilco Pillajo, “La tecnología multimedia y su relación con el desarrollo psicomotriz de niños de educación inicial de la Unidad Educativa ‘Luis A. Martínez’ del cantón Ambato,,” Universidad Técnica de Ambato, Ambato, 2020.
- [22] K. E. Moreno Daza, “La educación virtual en el desarrollo del aprendizaje significativo de los estudiantes de octavo año en Lengua y Literatura en la Unidad Básica General Vicente Rocafuerte,,” Universidad de Guayaquil, Guayaquil, 2019. [Online]. Available: <http://repositorio.ug.edu.ec/handle/redug/43436>
- [23] G. M. Correa Guagua, “Las TICS como estrategias didácticas tecnológicas para el cumplimiento de tareas académicas en la asignatura de historia de los estudiantes de primer año de bachillerato general unificado, jornada nocturna de la Unidad Educativa Fiscal ‘Eloy Alfaro’ del cantón Guayaquil,,” Universidad de Guayaquil, Guayaquil, 2018.
- [24] F. Pérez and P. Sánchez, “Introducción al uso de sistemas de cálculo con OPENHPC y sistemas de almacenamiento masivo de datos LUSTRE | DIGITAL.CSIC,” *Digital.CSIC*, Sep. 2020. [Online]. Available: <https://digital.csic.es/handle/10261/214058>

- [25] K. V. Sotomayor Vergara, “Desarrollo de una aplicación móvil de gestión de servicios de almacenamiento en la nube,” Oct. 2018. [Online]. Available: <https://riunet.upv.es:443/handle/10251/110017>
- [26] S. Vives Parra, “Desarrollo de una aplicación IOT con almacenamiento y procesado de datos en la nube mediante el protocolo MQTT,” Universitat Politècnica de València, Valencia, 2020. Accessed: May 19, 2022. [Online]. Available: www.etsit.upv.es
- [27] J. A. Baldrés González, “Desarrollo de una aplicación multiplataforma mediante el framework Flutter e implementación de servicios de autenticación y base de datos mediante Firebase,” May 2020, Accessed: May 25, 2022. [Online]. Available: <https://riunet.upv.es:443/handle/10251/143049>
- [28] D. Ruipéres, “Desarrollo de aplicaciones y servicios personalizados para industria y de acuerdo con las preferencias de los usuarios,” Universidad Politécnica de Madrid, Madrid, 2020. Accessed: May 25, 2022. [Online]. Available: <https://oa.upm.es/64366/>
- [29] E. Agüero, “Firebase en el desarrollo de aplicaciones móviles,” Universidad Politécnica de Sinaloa, Sinaloa, 2018. Accessed: May 25, 2022. [Online]. Available: <http://repositorio.upsin.edu.mx/formatos/TesinaEdgarUlisesAgueroAguiar6855.pdf>
- [30] P. E. Bautista Salazar, “Aplicación móvil usando el framework Flutter para el control de las tutorías académicas del sistema integrado de la Universidad Técnica de Ambato,” Universidad Técnica de Ambato, Ambato, 2022. [Online]. Available: <https://repositorio.uta.edu.ec/bitstream/123456789/34855/1/t1968ti.pdf>
- [31] E. Gómez, M. Marcillo, and N. Ramírez, “Metodologías ágiles para el desarrollo de proyectos,” *Fundación Universitaria Católica Lunem Gentium*, 2020.
- [32] B. Hohensee, *Introducción a Android Studio*. Babelcube Inc., 2014. Accessed: Jun. 06, 2022. [Online]. Available: https://books.google.com/books/about/Introducci%C3%B3n_A_Android_Studio_Incluye_P.html?hl=es&id=4dkuBQAAQBAJ

- [33] O. Paredes and F. Paredes, *Metodología de programación. Aplicaciones con Java*, Primera. Ambato: Edicumbre Editorial Cooperativa, 2018.
- [34] K. Abd. Latif, R. Hammad, T. T. Sujaka, K. Marzuki, and A. S. Anas, “Forensic Whatsapp Investigation Analysis on Bluestack Simulator Device Using Live Forensic Method With ACPO Standard,” *IJISTECH (International Journal of Information System and Technology)*, vol. 5, no. 3, pp. 331–338, Oct. 2021, doi: 10.30645/IJISTECH.V5I3.148.
- [35] D. Cárdenas, V. Guadalupe, P. Salazar, and B. Vergara, “Sistema de control de tickets para la empresa Cliente Strategik,” *REVISTA ODIGOS*, vol. 2, no. 1, pp. 41–53, Feb. 2021, doi: 10.35290/RO.V2N1.2021.387.

ANEXOS

ANEXO A

<p>Aplicación firebase Android para la gestión de tareas académicas en la Unidad Educativa Emanuel</p>

Manual de usuario

El presente manual de usuario tiene como objetivo dar a conocer a los usuarios(Administrador, Profesores y Estudiantes) las funciones que tiene la aplicación para la gestión de tareas académicas y guiarlos para el uso de las mismas.

ADMINISTRADOR

Administrador

Inicio de sesión



The image shows a login form for an administrator. At the top, it says "BIENVENIDOS" (Welcome) with a large black arrow pointing to the right. Below this is a green button labeled "Iniciar sesión" (Log in). Underneath are two input fields: the first is for the email address, marked with an envelope icon, and the second is for the password, marked with an eye icon. Below the password field is a link that says "¿Ovidaste tu contraseña?" (Forgot your password?). At the bottom of the form is a dark blue button labeled "INGRESAR" (Log in). Below the button is the text "o Regístrate." (or Register.).

El administrador debe digitar su correo electrónico y contraseña, dar clic al botón para ingresar a la aplicación si los datos son correctos.

Correo electrónico

Contraseña

Botón para ingresar

Aprobar o rechazar profesores



Se muestra una lista con los nombres y apellidos de las cuentas que se quieren registrar con el rol de "Profesor", el administrador decide si aprobar o rechazar la cuenta al dar clic en algunos de los botones.

Lista de profesores

Botón para rechazar

Botón para aprobar

Administrador

Aprobar o rechazar profesores

Botón para cerrar sesión



Se muestra una lista con los nombres y apellidos de las cuentas que se quieran registrar con el rol de "Estudiante", el administrador decide si aprobar o rechazar la cuenta al dar clic en algunos de los botones.

Lista de estudiantes

Botón para rechazar

Botón para aprobar

PROFESORES

Profesores

Inicio de sesión



El profesor debe digitar su correo electrónico y contraseña, dar clic al botón para ingresar a la aplicación si los datos son correctos.

Correo electrónico

Contraseña

Botón para ingresar

Registrar cuenta

The image shows a mobile application interface for creating a user account. At the top, the title 'Crear cuenta' is displayed in bold black text, accompanied by a black icon of a person with a plus sign. Below this is a green header bar with the text 'Datos requeridos'. The form consists of several input fields: 'Correo electronico:' with an envelope icon, 'Contraseña:' with a password field and an eye icon, 'Nombres:' with a person icon, 'Apellidos:' with a person icon, 'Celular:' with a mobile phone icon, and 'Rol:' with a dropdown menu showing 'Selecciona una opción...'. At the bottom of the form is a dark blue button with the text 'REGISTRAR CUENTA' in white capital letters. The background of the form is a blurred image of a building.

El profesor debe digitar sus datos personales, al dar clic en el botón guardar la cuenta ha sido creada. Pero no se puede usar hasta que el administrador lo apruebe.

Correo electrónico

Contraseña

Nombres

Apellidos

Numero de celular

Rol de usuario

Botón para crear cuenta

Restablecer contraseña

Restablecer contraseña

Se enviará un enlace a su correo en el cual podrá establecer una nueva contraseña.

Correo electrónico

✉

RESTABLECER CONTRASEÑA

El profesor debe digitar un correo electrónico al cual será enviado un enlace para que pueda establecer una nueva contraseña de acceso para su cuenta, al dar clic al botón se enviará el mensaje.

Correo electrónico

Botón para enviar el correo

Agregar nueva tarea

Profesor: Emilly Natasha Cuenca Barrera

TAREAS PERFIL SALIR

AGREGAR PENDIENTES ACABADAS

Agregar nueva tarea

Título:

Instrucciones:

Fecha de entrega:

Curso:
Segundo

Materia:
Lengua y literatura

Tipo de entrega:
Presencial

GUARDAR

El botón “AGREGAR” muestra la interfaz gráfica para agregar una tarea. El profesor digita los datos de la tarea que desea crear, al dar clic en el botón la tarea se guarda.

Título

Instrucciones

Abrir calendario para seleccionar la fecha

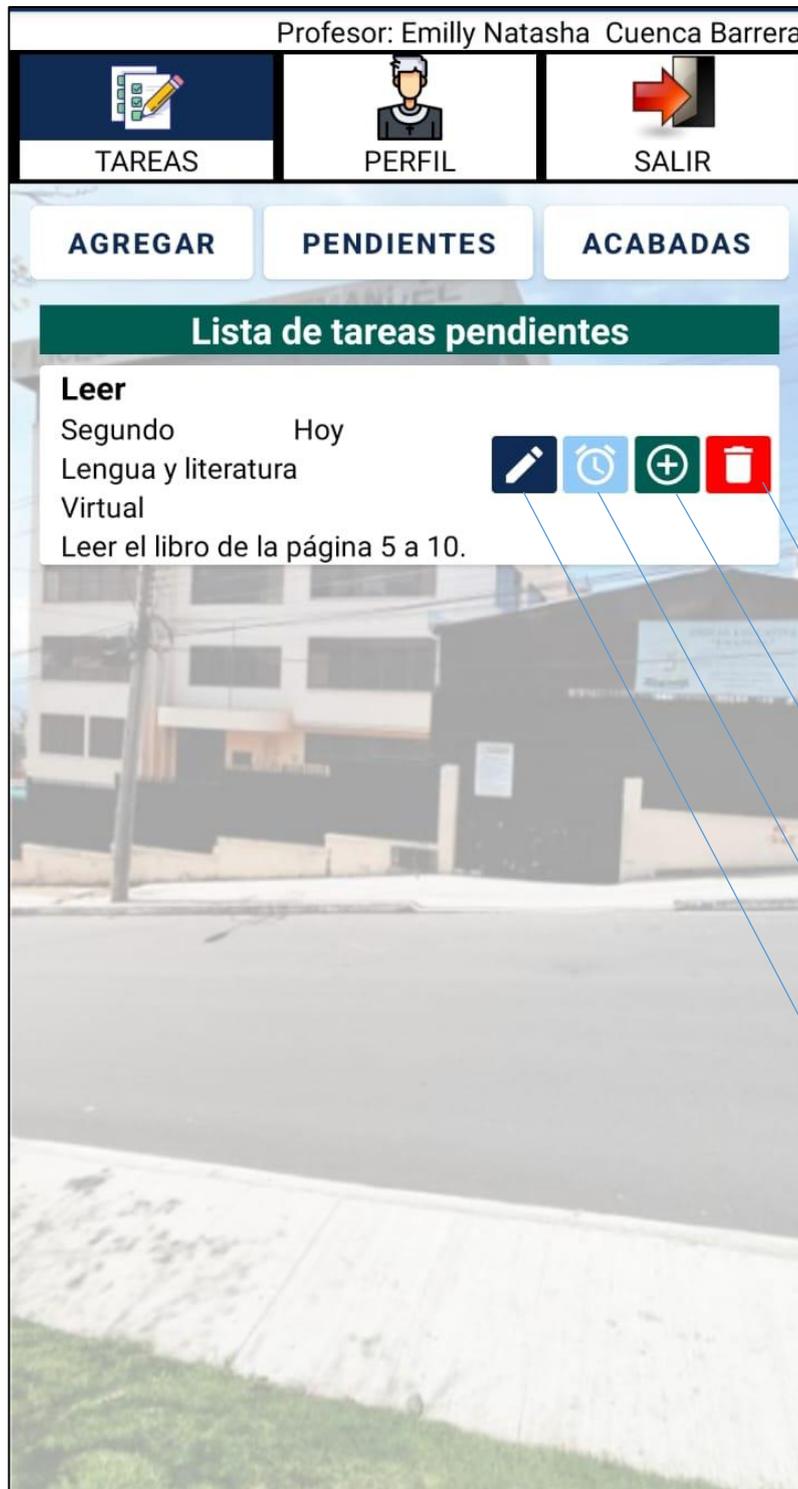
Selección del curso

Selección de la materia

Selección del tipo de entrega

Botón para guardar la tarea

Ver tareas pendientes



El botón "PENDIENTES" muestra la interfaz gráfica para ver las tareas pendientes. El profesor mira la lista de tareas pendientes. El botón "ACABADAS" permite ver las tareas ya finalizadas.

Botón para eliminar la tarea

Botón para revisar entregas

Botón para recordar la tarea

Botón para editar la tarea

Editar tarea

El profesor puede ver la información de la tarea y editarla.

TAREAS PERFIL SALIR

Editar tarea

Título:
Leer

Instrucciones:
Leer el libro de la página 5 a 10.

Fecha de entrega:
27-06-2022

Curso:
Segundo

Materia:
Lengua y literatura

Tipo de entrega:
Virtual

GUARDAR CAMBIOS

CERRAR

Título

Instrucciones

Abrir calendario

Seleccionar curso

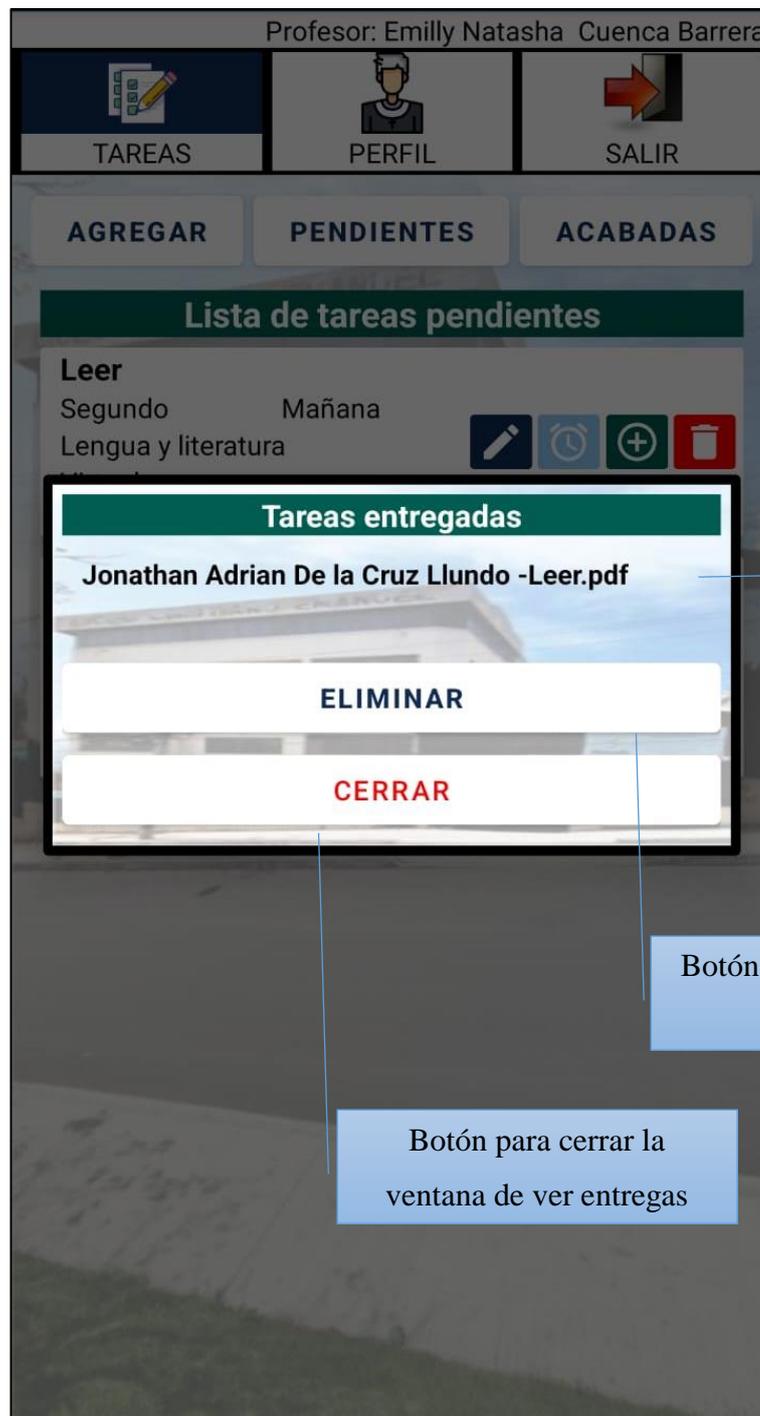
Seleccionar materia

Seleccionar tipo de entrega

Botón para guardar cambios

Botón para cerrar la ventana para editar

Ver entregas



El profesor puede ver una lista con los archivos entregados y ver su contenido al dar clic en alguno de ellos

Lista de archivos entregados

Botón para eliminar la tarea

Botón para cerrar la ventana de ver entregas

Ver contenido del archivo

Estudiante: Jonathan Adrian De la Cruz Llundo
Tarea: Leer

DESCARGAR **SALIR**

UNIVERSIDAD TÉCNICA

Tema: Experimento de Joule-Ciclos de refrigeracion y de motor a combustion 2-3 y 4 tiempos
Integrantes: Nicolay Cuenca - Diego De La Cruz **Fecha:** 12/06/2022
Asignatura: Termodinámica
Carrera: Ingeniería Industrial **Curso:** Tercero "A"

OBJETIVOS

- Comprender la importancia y relevancia que tiene el experimento de joule para la termodinámica
- Investigar el funcionamiento mediante ciclos en los motores de 2,3y 4 tiempos y sus características mas importantes
- Consultar el funcionamiento de los refrigeradores y cada componente que tiene este ciclo de refrigeración

EXPERIMENTO DE JOULE

El profesor visualiza el contenido del archivo referente a la tarea para su revisión.

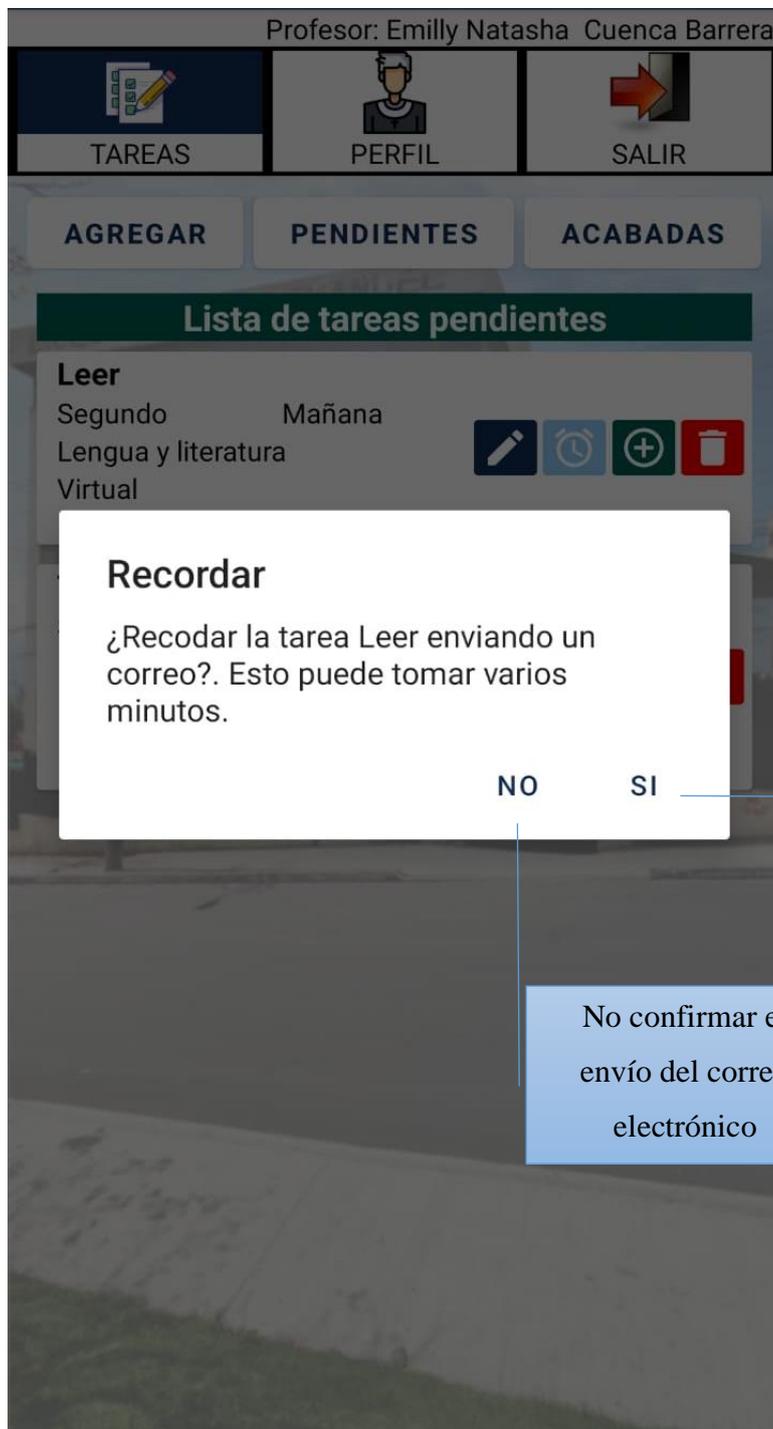
Botón salir del visualizador de archivos

Botón para descargar el archivo

Contenido del archivo

Profesores

Recordar tarea



El profesor puede recordar la información de la tarea a los alumnos mediante un correo electrónico.

Confirmar el envío del correo electrónico

No confirmar el envío del correo electrónico

Profesores

Ver o Editar perfil

Botón para cerrar sesión

Profesor: Emily Natasha Cuenca Barrera

 TAREAS	 PERFIL	 SALIR
---	---	--

VER **ACTUALIZAR**

Información del perfil

Nombres:
 Emily Natasha

Apellidos:
 Cuenca Barrera

Celular:
 0987514231

Curso:
NA

Materia 1:
Educación física

Materia 2:
NA

GUARDAR

El botón “VER” muestra la información, el botón “ACTUALIZAR” habilita los campos para editar la información. El profesor mira su información y la edita si así lo requiere.

Nombres

Apellidos

Número de celular

Seleccionar el curso

Seleccionar la materia 1

Seleccionar la materia 2

Botón para actualizar la información

ESTUDIANTES

Estudiantes

Inicio de sesión

BIENVENIDOS

Iniciar sesión

✉

👁

¿Ovidaste tu contraseña?

INGRESAR

o
Regístrate.

El estudiante debe digitar su correo electrónico y contraseña, dar clic al botón para ingresar a la aplicación si los datos son correctos.

Correo electrónico

Contraseña

Botón para ingresar

Registrar cuenta

The image shows a mobile application interface for creating a student account. At the top, the title 'Crear cuenta' is displayed in bold black text, accompanied by a black icon of a person with a plus sign. Below this is a dark green header with the text 'Datos requeridos'. The form consists of several input fields: 'Correo electronico:' with an envelope icon, 'Contraseña:' with a password icon and a visibility toggle, 'Nombres:' with a person icon, 'Apellidos:' with a person icon, 'Celular:' with a mobile phone icon, and 'Rol:' with a dropdown menu showing 'Selecciona una opción...'. At the bottom of the form is a dark blue button with the text 'REGISTRAR CUENTA' in white capital letters.

El estudiante debe digitar sus datos personales, al dar clic en el botón guardar la cuenta ha sido creada. Pero no se puede usar hasta que el administrador lo apruebe.

Correo electrónico

Contraseña

Nombres

Apellidos

Numero de celular

Rol de usuario

Botón para crear cuenta

Restablecer contraseña



El estudiante debe digitar un correo electrónico al cual será enviado un enlace para que pueda establecer una nueva contraseña de acceso para su cuenta, al dar clic al botón se enviará el mensaje.

Correo electrónico

Botón para enviar el correo

Ver tareas pendientes



El botón “PENDIENTES” muestra la interfaz gráfica para ver las tareas pendientes. El botón “MAÑANA” permite ver las tareas para mañana. El estudiante mira la lista de tareas.

Botón para ver detalles de la tarea

Lista de tareas pendientes

Ver detalles de la tarea

Estudiante: Jonathan Adrian De la Cruz Llundo

TAREAS PERFIL SALIR

MAÑANA PENDIENTES

Lista de tareas

Detalles de la tarea

Título:
Trabalenguas

Instrucciones:
Memorizarse 2 trabalenguas para decirlo en frente de la clase

Fecha de entrega:
Viernes, 1 de Julio de 2022

Materia:
Lengua y literatura

Tipo de entrega:
Presencial

CERRAR

El estudiante puede ver la información de la tarea.

Título

Instrucciones

Fecha

Materia

Tipo de entrega

Botón para cerrar la ventana de detalles

Entregar tarea

The screenshot shows a mobile application interface for a student named Jonathan Adrian De la Cruz Llundo. At the top, there are three navigation buttons: 'TAREAS' (Tasks), 'PERFIL' (Profile), and 'SALIR' (Logout). Below these are two tabs: 'MAÑANA' (Tomorrow) and 'PENDIENTES' (Pending). The main content area is titled 'Detalles de la tarea' (Task details) and contains the following information:

- Título:** Leer
- Instrucciones:** Leer el libro de la página 5 a 10.
- Fecha de entrega:** MAÑANA: Miércoles, 29 de Junio de 2022
- Materia:** Lengua y literatura
- Tipo de entrega:** Virtual
- [Ver archivo de la tarea entregada](#)
- EDITAR ENTREGA** button
- CERRAR** button

Callouts on the right side of the interface provide additional context:

- A large green box at the top right explains: "Si el tipo de entrega es virtual el estudiante visualiza la interfaz gráfica para entregar la tarea. Si la tarea ya fue entregada se puede editar dicha entrega." (If the type of delivery is virtual, the student sees the graphical interface to deliver the task. If the task has already been delivered, it can be edited.)
- Individual green boxes label the fields: 'Título', 'Instrucciones', 'Fecha', 'Materia', 'Tipo de entrega', and 'Ver el contenido de la entrega'.
- At the bottom, two green boxes label the buttons: 'Botón para editar o entregar la tarea' (pointing to 'EDITAR ENTREGA') and 'Botón para cerrar la ventana detalles' (pointing to 'CERRAR').

Ver contenido del archivo

Estudiante: Jonathan Adrian De la Cruz Llundo
Tarea: Leer

DESCARGAR **SALIR**

Botón para descargar el archivo

Botón para salir del visualizador de archivos

Tema: Experimento de Joule-Ciclos de refrigeracion y de motor a combustion 2-3 y 4 tiempos
Integrantes: Nicolay Cuenca - Diego De La Cruz **Fecha:** 12/06/2022
Asignatura: Termodinámica
Carrera: Ingeniería Industrial **Curso:** Tercero "A"

OBJETIVOS

- Comprender la importancia y relevancia que tiene el experimento de joule para la termodinámica
- Investigar el funcionamiento mediante ciclos en los motores de 2,3y 4 tiempos y sus características mas importantes
- Consultar el funcionamiento de los refrigeradores y cada componente que tiene este ciclo de refrigeración

EXPERIMENTO DE JOULE

El estudiante visualiza el contenido del archivo que entregó

Botón para salir del visualizador de archivos

Contenido del archivo

Ver o Editar perfil

Botón para cerrar sesión

Estudiante: Jonathan Adrian De la Cruz Llundo

 TAREAS	 PERFIL	 SALIR
---	---	--

VER **ACTUALIZAR**

Información del perfil

Nombres:
 Jonathan Adrian

Apellidos:
 De la Cruz Llundo

Celular:
 0987514230

Curso:
Tercero

GUARDAR

El botón “VER” muestra la información, el botón “ACTUALIZAR” habilita los campos para editar la información. El estudiante mira su información y la edita si así lo requiere.

Nombres

Apellidos

Número de celular

Seleccionar el curso

Botón para actualizar la información