



UNIVERSIDAD TÉCNICA DE AMBATO

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

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

Tema:

**APLICACIÓN MÓVIL CON GEOLOCALIZACIÓN PARA EL REPORTE
CIUDADANO DE DAÑOS EN LA INFRAESTRUCTURA URBANA DE LA
CIUDAD DE AMBATO.**

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la obtención del título de Ingeniero en Tecnologías de la Información.

ÁREA: Base de datos y Sistemas informáticos

LÍNEA DE INVESTIGACIÓN: Tecnología de la información y Sistemas de control

AUTOR: Deybi Paul Rivera Rivera

TUTOR: Ing. Oscar Fernando Ibarra Torres, Mg.

Ambato - Ecuador

febrero – 2024

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: APLICACIÓN MÓVIL CON GEOLOCALIZACIÓN PARA EL REPORTE CIUDADANO DE DAÑOS EN LA INFRAESTRUCTURA URBANA DE LA CIUDAD DE AMBATO, desarrollado bajo la modalidad Proyecto de Investigación por el señor Deybi Paul Rivera Rivera, 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 Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, febrero 2024.

Ing. Oscar Fernando Ibarra Torres, Mg.

TUTOR

AUTORÍA

El presente trabajo de titulación con el tema: APLICACIÓN MÓVIL CON GEOLOCALIZACIÓN PARA EL REPORTE CIUDADANO DE DAÑOS EN LA INFRAESTRUCTURA URBANA DE LA CIUDAD DE AMBATO es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, febrero 2024.



Deybi Paul Rivera Rivera

C.C. 1401235914

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, febrero 2024.



Deybi Paul Rivera Rivera

C.C. 1401235914

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por el señor Deybi Paul Rivera Rivera, 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 MÓVIL CON GEOLOCALIZACIÓN PARA EL REPORTE CIUDADANO DE DAÑOS EN LA INFRAESTRUCTURA URBANA DE LA CIUDAD DE AMBATO**, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, febrero 2024.

Ing. Elsa Pilar Urrutia Urrutia, Mg.
PRESIDENTE DEL TRIBUNAL

Ing. Daniel Jeréz Mayorga, Mg.
PROFESOR CALIFICADOR

Ing. Leonardo Torres Valverde, Mg.
PROFESOR CALIFICADOR

DEDICATORIA

El proyecto presentado está dedicado a mi madre, Claudia Rivera, por haberme forjado como la persona que soy actualmente, su apoyo moral, que ha sabido guiarme con fortaleza y un amor incondicional.

A mis hermanos, Diana y Danny, por su cariño y consejos, que siempre estuvieron pendientes en mis necesidades durante todo este proceso y logramos que este sueño se haga realidad.

A mi abuela, Matilde Atiencia, por sus sabias palabras e inspiración.

AGRADECIMIENTO

Agradezco enormemente a mi familia, por ser el pilar incluso en los momentos más difíciles, quienes supieron brindarme ese tiempo para escucharme y apoyarme.

A mis amigos y compañeros Bryann, Cristian, Roger y Stiven, que estuvieron conmigo en este largo camino. Su cariño, apoyo y confianza han sido invaluable y ser mi equipo de aliento.

A cada uno de los docentes, por su experiencia, comprensión paciencia y exigencia.

ÍNDICE GENERAL DE CONTENIDOS

PORTADA	i
APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
DERECHOS DE AUTOR	iv
APROBACIÓN DEL TRIBUNAL DE GRADO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICE GENERAL DE CONTENIDOS	viii
ÍNDICE DE TABLAS	xii
ÍNDICE DE FIGURAS	xiv
ÍNDICE DE ANEXOS	xviii
RESUMEN EJECUTIVO	xix
ABSTRACT	xx
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	3

1.3.1 Aplicaciones Móviles.....	3
1.3.2 Interfaz y Experiencia de Usuarios	3
1.3.3 Funcionalidades principales	3
1.3.4 Comunicación y conectividad	4
1.3.5 Planificación del mantenimiento.....	5
1.3.6 Mantenimiento preventivo	5
1.3.7 Control de calidad	5
1.3.8 Gestión de mantenimiento.....	6
1.3.9 IDE.....	6
1.3.10 Framework	6
1.3.11 Geolocalización.....	6
1.4 Objetivos	7
1.4.1 Objetivo general	7
1.4.2 Objetivos específicos	7
CAPÍTULO II. METODOLOGÍA.....	8
2.1 Materiales.....	8
2.2 Métodos.....	11

2.2.1 Modalidad de la investigación	11
2.2.2 Población y muestra	12
2.2.3 Recolección de información.....	14
2.2.4 Procesamiento y análisis de datos	28
CAPÍTULO III. RESULTADOS Y DISCUSIÓN.....	29
3.1 Análisis y discusión de resultados.....	29
3.1.1 Identificación de problemas de gestión de mantenimiento	29
3.1.2 Identificación de problemas de infraestructura urbana	30
3.1.3 Herramientas de desarrollo de software móvil.....	31
3.1.4 Sistemas Gestores de Bases de Datos	34
3.1.5 Técnicas y herramientas de geolocalización	38
3.1.6 Metodologías para el desarrollo móvil.....	42
3.2 Desarrollo de la propuesta.....	43
3.2.1 Fase 1: Exploración.....	43
3.2.2 Fase 2: Inicialización.....	46
3.2.3 Fase 3: Producción	66
3.2.4 Fase 4: Estabilización.....	94
3.2.5 Fase 5: Pruebas del sistema.....	96
3.2.6 Implantación la aplicación móvil con geolocalización para el reporte ciudadano de daños en la Infraestructura Urbana de la ciudad de Ambato.....	106
CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES	111

4.1 Conclusiones	111
4.2 Recomendaciones.....	112
REFERENCIAS BIBLIOGRÁFICAS.....	113
ANEXOS.....	116

ÍNDICE DE TABLAS

Tabla 1. Población de estudio.	12
Tabla 2. Simbología del cálculo de la muestra.	13
Tabla 3. Rangos de confiabilidad de alfa de Cronbach.....	15
Tabla 4. Problemas de gestión de mantenimiento de la infraestructura urbana [23].	29
Tabla 5. Problemas de infraestructura urbana frecuentes.	30
Tabla 6. Comparativa entre entornos de desarrollo y frameworks.	32
Tabla 7. Comparativa entre Sistemas de Gestión de Bases de Datos.	35
Tabla 8. Comparativa entre técnicas de geolocalización.	38
Tabla 9. Comparativa entre herramientas de geolocalización.	41
Tabla 10. Comparativa entre metodologías ágiles.	42
Tabla 11. Establecimiento de los interesados.	43
Tabla 12. Planificación de fases de la metodología Mobile-D.	47
Tabla 13. Storycard inicio de sesión, cerrar sesión, registrar usuario.....	69
Tabla 14. Storycard recuperar contraseña.....	72
Tabla 15. Storycard perfil de usuario.....	73
Tabla 16. Storycard editar perfil y contraseña.	74
Tabla 17. Storycard generar reportes.	76
Tabla 18. Storycard visualizar reportes del usuario.	77
Tabla 19. Storycard editar y borrar reportes.	79
Tabla 20. Storycard visualizar reportes de otros usuarios.....	81

Tabla 21. Storycard visualizar detalles del reporte.	82
Tabla 22. Storycard reaccionar y comentar reportes.....	83
Tabla 23. Storycard geolocalización y mapas.....	85
Tabla 24. Storycard aprobación de reportes.....	86
Tabla 25. Storycard registrar administradores.	88
Tabla 26. Storycard visualizar notificaciones.	90
Tabla 27. Storycard información general.....	91

ÍNDICE DE FIGURAS

Figura 1. Tabulación de resultados pregunta 1.	16
Figura 2. Tabulación de resultados pregunta 2.	17
Figura 3. Tabulación de resultados pregunta 3.	18
Figura 4. Tabulación de resultados pregunta 4.	19
Figura 5. Tabulación de resultados pregunta 5.	20
Figura 6. Tabulación de resultados pregunta 6.	21
Figura 7. Tabulación de resultados pregunta 7.	22
Figura 8. Tabulación de resultados pregunta 8.	23
Figura 9. Tabulación de resultados pregunta 9.	24
Figura 10. Tabulación de resultados pregunta 10.	25
Figura 11. Tabulación de resultados pregunta 11.	26
Figura 12. Tabulación de resultados pregunta 12.	27
Figura 13. Arquitectura de la aplicación móvil.....	50
Figura 14. Colecciones del proyecto en Firestore Database.	50
Figura 15. Almacenamiento de imágenes en Firebase Storage.....	52
Figura 16. Esquema de navegabilidad de la aplicación móvil.	53
Figura 17. Diagrama de caso de uso usuario común.....	54
Figura 18. Diagrama de caso de uso administrador.	55
Figura 19. Prototipo inicio de sesión.....	56
Figura 20. Prototipo recuperar contraseña.	57

Figura 21. Prototipo registrar usuario.	58
Figura 22. Prototipo perfil de usuario.	59
Figura 23. Prototipo página principal.....	60
Figura 24. Prototipo mis reportes.....	61
Figura 25. Prototipo aprobaciones.	62
Figura 26. Prototipo información.....	63
Figura 27. Prototipo detalles del reporte.....	64
Figura 28. Prototipo generar reporte.	65
Figura 29. Prototipo notificaciones.....	66
Figura 30. Dependencias del proyecto.....	68
Figura 31. Método registrar usuario.....	70
Figura 32. Método iniciar sesión.....	71
Figura 33. Método recuperar contraseña.....	72
Figura 34. Método recuperar usuario.....	73
Figura 35. Método editar perfil de usuario.....	75
Figura 36. Método cambiar contraseña.....	75
Figura 37. Método generar reporte.....	77
Figura 38. Método para recuperar los reportes del usuario.....	78
Figura 39. Método editar reporte.	79
Figura 40. Método eliminar un reporte.	80
Figura 41. Método recuperar reportes de usuarios.....	81

Figura 42. Método mostrar detalles del reporte.	82
Figura 43. Método reaccionar reporte.	84
Figura 44. Método comentar reportes.	84
Figura 45. Método obtener ubicación del dispositivo.	86
Figura 46. Método aprobar/rechazar reporte.	87
Figura 47. Método registrar administrador.	89
Figura 48. Método recuperar notificaciones.	91
Figura 49. Método obtener reportes más comentados.	92
Figura 50. Método obtener reportes más reaccionados.	92
Figura 51. Método obtener reportes por categoría.	93
Figura 52. Método obtener reportes por sector.	93
Figura 53. Método obtener usuario que más han reportado.	94
Figura 54. Página inicio de sesión.	96
Figura 55. Página registrar usuario.	97
Figura 56. Página recuperar contraseña.	97
Figura 57. Página perfil de usuario.	98
Figura 58. Funcionalidad editar perfil.	98
Figura 59. Funcionalidad editar contraseña.	99
Figura 60. Página generar reporte.	99
Figura 61. Página visualizar mis reportes.	100
Figura 62. Funcionalidad editar reporte.	101

Figura 63. Página visualizar otros reportes.	101
Figura 64. Página detalles del reporte.	102
Figura 65. Funcionalidad geolocalización y mapas.	103
Figura 66. Funcionalidad aprobar/rechazar reporte.	104
Figura 67. Funcionalidad registrar administradores.....	104
Figura 68. Página notificaciones.	105
Figura 69. Página información general.	106
Figura 70. Código QR del repositorio de la aplicación.....	107
Figura 71. Repositorio en GitHub.....	107
Figura 72. Archivo en OneDrive.....	108
Figura 73. Descarga del archivo de instalación.	108
Figura 74. Habilitar permisos de origen desconocido.....	109
Figura 75. Confirmar instalación.	109
Figura 76. Aplicación instalada en el dispositivo móvil.....	110

ÍNDICE DE ANEXOS

Anexo A. Manual de usuario	116
----------------------------------	-----

RESUMEN EJECUTIVO

Las aplicaciones móviles se posicionan como herramientas poderosas para fortalecer la participación ciudadana y optimizar la gestión de infraestructuras urbanas. Este proyecto de investigación se enfoca en mejorar la gestión de mantenimiento de la infraestructura urbana en Ambato mediante la implementación de una aplicación móvil con geolocalización para el reporte ciudadano de daños, desarrollada con el framework Flutter.

La metodología Mobile-D, adoptada en este proyecto, ha facilitado el desarrollo eficiente de la aplicación, permitiendo crear una herramienta altamente efectiva. La elección del framework Flutter ha sido clave para garantizar una experiencia de usuario fluida y una implementación eficaz.

La aplicación móvil permitirá a los ciudadanos reportar de manera eficiente y precisa los daños en la infraestructura urbana, agilizando la identificación y resolución de problemas por parte de las autoridades locales. Este enfoque integral mejorará significativamente la gestión de mantenimiento, asegurando una infraestructura en condiciones óptimas para una ciudad más segura y habitable.

La determinación cuidadosa de las herramientas de desarrollo garantiza una aplicación fácil de usar, simplificando el proceso de reporte de daños para los ciudadanos. La metodología Mobile-D, ha permitido cumplir con los requisitos del proyecto de manera oportuna, asegurando una implementación exitosa.

En resumen, este proyecto propone la implementación de una aplicación móvil para el reporte ciudadano de daños en Ambato, respaldada por la metodología Mobile-D y desarrollada con el framework Flutter. Esta combinación asegura eficiencia en la gestión de mantenimiento, contribuyendo de manera significativa a una ciudad más segura y agradable.

Palabras clave: Infraestructura urbana, participación ciudadana, gestión de mantenimiento, Framework, Flutter, Mobile-D.

ABSTRACT

Mobile applications are positioned as powerful tools to strengthen citizen participation and optimise urban infrastructure management. This research project focuses on improving the management of urban infrastructure maintenance in Ambato through the implementation of a mobile application with geolocation for citizen damage reporting, developed with the Flutter framework.

The Mobile-D methodology, adopted in this project, has facilitated the efficient development of the application, allowing the creation of a highly effective tool. The choice of the Flutter framework has been key to ensure a smooth user experience and efficient implementation.

The mobile application will enable citizens to efficiently and accurately report damage to urban infrastructure, speeding up the identification and resolution of problems by local authorities. This comprehensive approach will significantly improve maintenance management, ensuring an infrastructure in optimal condition for a safer and more liveable city.

The careful determination of the development tools ensures a user-friendly application, simplifying the damage reporting process for citizens. The Mobile-D methodology has enabled the project requirements to be met in a timely manner, ensuring a successful implementation.

In summary, this project proposes the implementation of a mobile application for citizen damage reporting in Ambato, supported by the Mobile-D methodology and developed with the Flutter framework. This combination ensures efficiency in maintenance management, contributing significantly to a safer and more pleasant city.

Keywords: Urban infrastructure, citizen participation, maintenance management, Framework, Flutter, Mobile-D.

CAPÍTULO I. MARCO TEÓRICO

1.1 Tema de investigación

APLICACIÓN MÓVIL CON GEOLOCALIZACIÓN PARA EL REPORTE CIUDADANO DE DAÑOS EN LA INFRAESTRUCTURA URBANA DE LA CIUDAD DE AMBATO.

1.1.1 Planteamiento del problema

A nivel mundial, las Tecnologías de la Información y Comunicación (TIC) han tenido un impacto importante en la gestión de la Infraestructura Urbana, especialmente en la participación ciudadana. Las TIC han permitido una mayor transparencia en la gestión de la infraestructura, lo que ha mejorado la confianza de los ciudadanos en los gobiernos locales y ha fomentado una mayor colaboración entre el sector público y la sociedad civil. La tecnología también ha permitido una mayor eficiencia en la gestión de los recursos y la planificación de proyectos, lo que ha mejorado la calidad de vida de los ciudadanos [1].

En países como Ecuador, las TIC han sido utilizadas para mejorar la gestión de la Infraestructura Urbana y la participación ciudadana. Los gobiernos locales han implementado sistemas de información geográfica, plataformas en línea y aplicaciones móviles para recopilar información sobre los problemas que afectan directamente al desarrollo de la sociedad y permitir a los ciudadanos reportarlos de manera fácil y eficiente. Esto ha mejorado la comunicación entre los ciudadanos y los gobiernos locales, lo que ha permitido una respuesta más rápida y efectiva a los problemas de infraestructura [2].

En la ciudad de Ambato, las aplicaciones móviles y plataformas en línea, como redes sociales, han sido utilizadas para mejorar la participación ciudadana en la gestión de la Infraestructura Urbana. Los ciudadanos pueden reportar fácilmente los problemas de través de estas herramientas [3].

1.2 Antecedentes investigativos

Luego de investigar en los repositorios de la Universidad, se ha podido observar que son escasos los temas similares al que se pretende abordar dentro del país.

En el ámbito de estudio relacionado con la gestión del mantenimiento de la Infraestructura Urbana y el uso de aplicaciones móviles, se han realizado diversas investigaciones que han aportado al conocimiento existente.

Un estudio relevante realizado por Wang, M. [4] se centró en el impacto de una aplicación móvil en la gestión del mantenimiento de Infraestructuras Urbanas. Los resultados de esta investigación resaltaron la importancia de la aplicación móvil para agilizar los procesos de reporte y seguimiento de incidencias, así como para mejorar la planificación y ejecución de tareas de mantenimiento.

Otra investigación realizada por Wang, X [5], se enfocó en el desarrollo de una aplicación móvil para la gestión eficiente del mantenimiento de la infraestructura vial. Este estudio analizó el impacto de la aplicación en la optimización de los procesos de mantenimiento, la reducción de tiempos de respuesta y la mejora en la comunicación entre los actores involucrados.

Además, Chung, K. [6] llevaron a cabo un análisis de aplicaciones móviles utilizadas en la gestión de mantenimiento de Infraestructuras Urbanas en diferentes contextos. Su investigación tuvo como objetivo identificar las funcionalidades más relevantes y evaluar su eficacia en la mejora de los procesos de mantenimiento.

Estos estudios podrían explorar aspectos como la evaluación de la calidad del mantenimiento realizado, la eficacia de las estrategias de mantenimiento implementadas, la optimización de los recursos utilizados y la identificación de áreas de mejora en la gestión del mantenimiento [7]. Investigaciones de este tipo contribuirían a llenar el vacío existente en el conocimiento y permitirían comprender mejor los desafíos y oportunidades relacionados con la gestión del mantenimiento de la Infraestructura Urbana en el contexto de la aplicación móvil.

Estos antecedentes investigativos proporcionan un marco teórico sólido y experiencias previas que respaldan la importancia y los beneficios de utilizar aplicaciones móviles en

la gestión del mantenimiento de la Infraestructura Urbana. A través de estas investigaciones previas, se ha evidenciado el potencial de las aplicaciones móviles para agilizar los procesos, mejorar la comunicación y optimizar la planificación del mantenimiento de Infraestructuras Urbanas [8].

1.3 Fundamentación teórica

1.3.1 Aplicaciones Móviles

Con la aparición de la telefonía celular, surgió el desarrollo de Aplicaciones Móviles que desempeñaban funciones importantes para facilitar la intercomunicación entre los usuarios en la red. Actualmente las aplicaciones móviles desempeñan un rol importante para la adaptación de los usuarios en la era de la digitalización donde gran parte de los procesos cotidianos se los puede realizar a través de un dispositivo móvil inteligente [9].

1.3.2 Interfaz y Experiencia de Usuarios

El estudio de la Experiencia de Usuario (UX), del inglés, User Experience, y la Usabilidad de un sistema software, nos permite generar valor agregado a un producto, el cual afecta directamente en la aceptación de este involucrando factores sociales, culturales y de contexto. La UX, es el conjunto de múltiples factores y elementos relativos, que inciden en el uso de un producto o servicio, cuyo resultado genera una percepción y respuesta por parte del usuario [10]. Los dispositivos móviles cuentan con una capacidad especial de procesamiento, conexión permanente o intermitente de red, memoria limitada, diseñados específicamente para una función, pero tienen la versatilidad de desarrollar otras funciones más genéricas [11].

1.3.3 Funcionalidades principales

Desde la propagación de las aplicaciones móviles, estos han beneficiado a centenares de usuarios, debido al sin número de funciones que estos realizan y demuestran, el alcance de un rol importante dentro de varios sectores como: el político, social, cultural, entre otros. Para que una aplicación tenga mejor rentabilidad y sea comercial, se considera ciertas características:

a. *Interfaz intuitiva*

Esto se refiere a que el entorno o la presentación de la App es amigable y que cualquier usuario la usa sin ningún problema o la necesidad de ser capacitado.

b. *Seguridad*

Las aplicaciones protegen la información y mantener la privacidad de los usuarios, ya que estos son comprometidos con ataques cibernéticos.

c. *Funcionamiento offline (Fuera de línea)*

Las aplicaciones cumplen con su funcionalidad para la que fueron creadas sin la disponibilidad de tener Internet.

d. *Actualizaciones periódicas*

Las aplicaciones son actualizadas cada cierto tiempo, ya sea para corregir errores los cuales tienen mejoras a las versiones actuales permitiendo cumplir con las necesidades y requerimiento de quienes las usan [12].

1.3.4 Comunicación y conectividad

Las comunicaciones móviles pueden construir el camino propuesto, principalmente con la evolución de las redes y dispositivos móviles hacia la era 5G que, en términos de aplicaciones, está surgiendo con una amplia gama de nuevos servicios.

Los juegos en línea en tiempo real, Realidad Virtual y transmisión de video de Ultra Alta Definición, que requieren una alta calidad, velocidad de acceso, eficiencia energética y espectral, baja latencia, escalabilidad y despliegue de infraestructura, para los que se están desarrollando nuevos paradigmas como Mobile Edge Computing, Ultra-Dense Networks y Network Slice Broker.

La transformación digital es el núcleo de la cuarta revolución industrial, y las infraestructuras de red 5G serán activos de apoyo clave. Un escenario de aplicación de impulso es formar una red de fábricas distribuidas geográficamente con una adaptación flexible de las capacidades de producción y el intercambio de recursos y activos para

mejorar el cumplimiento de los pedidos y allanando el camino para nuevas oportunidades comerciales [13].

1.3.5 Planificación del mantenimiento

El Mantenimiento, se constituye en aquellas actividades que se planifican y ejecutan con el objetivo de tener en estado óptimo de la Infraestructura Urbana y además cautelar y fiscalizar el capital que se invierte en obras de rehabilitación y mejora que a su vez incluyen obras conexas y complementarias [14]. Dentro de las finalidades del mantenimiento, es justamente el cuidado riguroso sobre las inversiones que se efectúen en los diferentes proyectos de inversión referidos a construcción o rehabilitación, con lo cual se asegura la transitabilidad continua, en forma cómoda y segura [15].

1.3.6 Mantenimiento preventivo

Se define como la actividad que se contempla para que los gastos sean menores en comparación al monto de la compra para sustituir ese bien, más si se trata de una gran inversión económica para la empresa u organización.

Además, con esto se busca mantener o alargar la vida útil que el activo puede llegar a poseer según el fabricante. Una de las mayores características del mantenimiento preventivo es que busca que las fallas o deterioros que se presentan puedan ser atendidas con la mayor brevedad posible.

Para llevar a cabo un adecuado mantenimiento preventivo, se debe contar con un plan para que el mismo se lleve a cabo acorde con las fechas e instrucciones que el fabricante menciona o persona encargada y la tarea la realice sin posibles eventualidades. En caso de que estas se presenten, el mismo pueda tener opciones para que el activo siga funcionando como lo venía haciendo [16].

1.3.7 Control de calidad

El control de calidad en obra ha sido una herramienta indispensable desde inicios de las civilizaciones. A lo largo de la historia, como parte fundamental y complementaria, surgen reglamentos y normativas las cuales han sido fundamentales para brindar un mejor control y resultados de proyectos.

El desarrollo de un Plan de Control de Calidad surge a partir de la Segunda Guerra Mundial, de la necesidad de la reconstrucción de proyectos en los diferentes países afectados, en los cuales se hacía evidente la falta de calidad y correcta ejecución de los proyectos [7].

1.3.8 Gestión de mantenimiento

Entendemos precisamente la gestión como la labor de sutura que cose la brecha entre la agenda política de un equipo de gobierno y su ejecución; como la construcción de puentes entre un determinado programa, es decir, una voluntad de acción, y su materialización. Precisamente por ello, por su papel mediador que se pone al servicio de las partes y hace que todas ganen [14].

1.3.9 IDE

Un IDE es una herramienta pensada para un lenguaje de programación que permite al desarrollador comprender conceptos iniciales y verificar fácilmente resultados, también se lo conoce como Ambiente de Desarrollo Integrado o Ambiente de Desarrollo Interactivo [18].

1.3.10 Framework

Es un conjunto de archivos y directorios que tiene como objetivo facilitar el desarrollo de aplicaciones. Su característica principal es contener funciones desarrolladas previamente y puesta en marcha, dirigidos para un lenguaje de programación en específico. De esta forma, el desarrollador se puede centrar en el problema principal y no tener que implementar funciones comunes desde cero [19], [20].

1.3.11 Geolocalización

La geolocalización se refiere a la posición geográfica de un objeto, como un usuario o un dispositivo electrónico [21]. La principal función de esta técnica es identificar la ubicación, proporcionando indicaciones, además de integrarse a un software de mapas [22].

1.4 Objetivos

1.4.1 Objetivo general

Implantar una aplicación móvil con geolocalización para el reporte ciudadano de daños en la Infraestructura Urbana de la ciudad de Ambato.

1.4.2 Objetivos específicos

- Identificar los problemas de gestión de mantenimiento de la Infraestructura Urbana en la ciudad de Ambato.
- Determinar las herramientas de desarrollo de software que permita la construcción de la aplicación móvil propuesta.
- Desarrollar una aplicación móvil con geolocalización para el reporte ciudadano de daños en la infraestructura pública urbana de la ciudad de Ambato.

CAPÍTULO II. METODOLOGÍA

2.1 Materiales

En el marco de esta investigación, se realizó una encuesta a los habitantes de la zona urbana de la ciudad de Ambato. El objetivo principal de esta encuesta fue evaluar la satisfacción y percepción de los residentes con respecto a la gestión de mantenimiento de la infraestructura urbana en la localidad.

Los materiales utilizados incluyeron cuestionarios estructurados que permitieron abordar de manera sistemática las cuestiones relativas a la satisfacción y percepción de los ciudadanos acerca de la gestión de mantenimiento de la infraestructura urbana.

Encuesta dirigida a los habitantes de la zona urbana de la ciudad de Ambato

Objetivo: Evaluar la satisfacción y percepción de los residentes de Ambato con respecto a la gestión de mantenimiento de la infraestructura urbana en su localidad.

Indicaciones:

- **Escalas de Respuesta:**
 - Se utiliza una escala de 1 a 5, donde 1 representa un aspecto positivo y 5 negativo.

Preguntas

1. En una escala del 1 al 5, donde 1 es “muy satisfecho”, 2 es “satisfecho”, 3 es “neutro”, 4 es “insatisfecho” y 5 es “muy insatisfecho”, **¿cuán satisfecho está con la gestión de mantenimiento de la infraestructura urbana en la ciudad de Ambato?**

- | | |
|----------------------------|----------------------------|
| <input type="checkbox"/> 1 | <input type="checkbox"/> 4 |
| <input type="checkbox"/> 2 | <input type="checkbox"/> 5 |
| <input type="checkbox"/> 3 | |

2. **¿Con qué frecuencia ha observado problemas en la infraestructura urbana de Ambato en el último año?**

- | | |
|--------------------------------------|---------------------------------------|
| <input type="checkbox"/> Diariamente | <input type="checkbox"/> Semanalmente |
| | <input type="checkbox"/> Mensualmente |

- Anual Nunca

3. En una escala del 1 al 5, donde 1 es “facilitaría mucho”, 2 es “facilitaría”, 3 es “neutro”, 4 es “facilitaría poco” y 5 es “no facilitaría”, **¿en qué medida cree que una aplicación móvil facilitaría el reporte de problemas en la infraestructura urbana en Ambato?**

- 1 4
 2 5
 3

4. En una escala del 1 al 5, donde 1 es “muy eficiente”, 2 es “eficiente”, 3 es “neutro”, 4 es “poco eficiente” y 5 es “no es eficiente”, **¿considera que la gestión actual de mantenimiento de la infraestructura urbana en Ambato es eficiente?**

- 1 4
 2 5
 3

5. **Si alguna vez ha reportado problemas de infraestructura urbana, ¿ha experimentado retrasos en la resolución de estos después de reportarlos?**

- Si
 No

6. **¿Qué tipo de dispositivo electrónico usa con más frecuencia?**

- Computador
 Celular
 Tablet
 Otro _____
 Ninguno

7. **¿Estaría dispuesto a utilizar una aplicación móvil para reportar problemas en la infraestructura urbana de Ambato?**

- Si
 No

8. **¿Cree que una aplicación móvil podría acelerar la respuesta a problemas de infraestructura urbana?**

- Si
 No

9. ¿Cuánto tiempo estaría dispuesto a dedicar a la utilización de una aplicación móvil para reportar problemas en la infraestructura urbana por cada reporte?

- | | |
|---|--|
| <input type="checkbox"/> Menos de 5 minutos | <input type="checkbox"/> 10-15 minutos |
| <input type="checkbox"/> 5-10 minutos | <input type="checkbox"/> Más de 15 minutos |

10. ¿Qué tipo de problemas en la infraestructura urbana le gustaría poder reportar a través de una aplicación móvil? (Selección múltiple)

- | | |
|--|---|
| <input type="checkbox"/> Baches | <input type="checkbox"/> Daños en las veredas |
| <input type="checkbox"/> Alumbrado público | <input type="checkbox"/> Otros _____ |

11. ¿Qué características considera esenciales en una aplicación móvil para reportar problemas en la infraestructura urbana? (Selección múltiple)

- | | |
|--|---|
| <input type="checkbox"/> Geolocalización | <input type="checkbox"/> Interacción directa con otros usuarios |
| <input type="checkbox"/> Adjuntar fotos | <input type="checkbox"/> Otros _____ |
| <input type="checkbox"/> Comentarios | |

12. En una escala del 1 al 5, donde 1 es “contribuiría mucho”, 2 es “contribuiría”, 3 es “neutro”, 4 es “contribuiría poco” y 5 es “no contribuiría”, ¿Cree que una aplicación móvil podría contribuir a una mejor gestión de la infraestructura urbana de Ambato?

- 1
- 2
- 3
- 4
- 5

2.2 Métodos

2.2.1 Modalidad de la investigación

En este estudio, se adoptó un enfoque de investigación que abarca los enfoques de investigación de la modalidad de campo, bibliográfica, descriptiva y aplicada.

a. Investigación de campo

La Investigación de campo implicó la recopilación de datos primarios mediante la realización de encuestas en la ciudad de Ambato. Proporcionó una base sólida de conocimientos teóricos y empíricos sobre los temas clave relacionados con la gestión del mantenimiento de la Infraestructura Urbana y la participación ciudadana. Esta información respaldó y enriqueció el desarrollo de la aplicación móvil y las decisiones tomadas durante el proceso de investigación y desarrollo.

b. Investigación bibliográfica

La Investigación bibliográfica se centró en revisar fuentes bibliográficas, estudios previos, informes técnicos y marco normativo relacionados con la gestión del mantenimiento de la Infraestructura Urbana y las aplicaciones móviles para la participación ciudadana. El objetivo es obtener un conocimiento sólido y fundamentado para mejorar la gestión del mantenimiento y promover la participación activa de los ciudadanos en el desarrollo urbano.

c. Investigación descriptiva

Se llevó a cabo un análisis y descripción de la situación actual de la gestión del mantenimiento de la Infraestructura Urbana en la ciudad de Ambato. Se recopiló datos sobre los problemas existentes, las áreas más afectadas, los tipos de mantenimiento requeridos y los procesos actuales de gestión. Además, se identificó las necesidades y las limitaciones tanto de los ciudadanos como de las autoridades municipales en relación con el mantenimiento de la infraestructura. Esta investigación descriptiva proporcionó una comprensión detallada del contexto actual en el que se implementó la aplicación móvil.

d. Investigación aplicada

Se centró en el desarrollo de una solución práctica para mejorar la gestión del mantenimiento de la Infraestructura Urbana mediante la aplicación móvil con geolocalización. Se llevó a cabo un proceso de diseño y desarrollo de la aplicación, incluyendo la creación de la interfaz, la integración de la geolocalización, la funcionalidad de reporte de problemas y el mecanismo de retroalimentación. Esta investigación aplicada también implicó pruebas y evaluaciones de la aplicación en situaciones reales, con el fin de medir su efectividad y recopilar datos sobre la satisfacción de los ciudadanos y la eficiencia en la gestión del mantenimiento.

2.2.2 Población y muestra

La población con la cual se trabajó en este proyecto es la de la zona urbana de la ciudad de Ambato. Se obtuvo una muestra aleatoria simple a fin de que cumpla la normalidad de la información.

La elección de 400 individuos como población para la recolección de información se basa en varios factores clave. En primer lugar, esta muestra se considera representativa de la diversidad demográfica y socioeconómica presente en el entorno urbano de Ambato. Además, el tamaño de la población es viable dentro de los recursos limitados disponibles para este estudio, lo que garantiza la realización efectiva de la encuesta en términos de tiempo y presupuesto. Además, este tamaño de población permite alcanzar un nivel adecuado de precisión estadística y confianza en los resultados, sin imponer restricciones metodológicas específicas. En conjunto, seleccionar 400 habitantes proporciona una base sólida para obtener resultados significativos y representativos que contribuyan a los objetivos de esta investigación.

Tabla 1. Población de estudio.

Población	Número	Porcentaje
Habitantes de la zona urbana de la ciudad de Ambato.	400	100%
Total	400	100%

Dado que la población obtenida cuenta con un total mayor a 100, se realizará el cálculo de la muestra utilizando la fórmula apropiada para muestras finitas con 400 habitantes de la zona urbana de la ciudad de Ambato.

Esto garantizará que el tamaño de la muestra sea representativo y adecuado para obtener conclusiones válidas sobre la población objetivo.

Fórmula:

$$n = \frac{NZ^2PQ}{Ne^2 + Z^2PQ} \quad (1)$$

Tabla 2. Simbología del cálculo de la muestra.

Simbología	Significado	Representación
N	Población	400
e	Error estándar	0.09
Z	Nivel de confianza del 95%	1.96
P	Probabilidad de éxito	0.5
Q	Probabilidad de fracaso	0.5
n	Tamaño de la muestra	91

$$n = \frac{(400)(1,96)^2(0,5)(0,5)}{(400)(0,09)^2 + (1,96)^2(0,5)(0,5)} = 91 \quad (2)$$

El tamaño de la muestra corresponde a 91 habitantes de la zona urbana de la ciudad de Ambato.

2.2.3 Recolección de información

Se encuestó aplicando el instrumento de tipo cuestionario con la herramienta de Formularios en línea y así poder llegar al público objetivo de manera fácil y rápida.

a. Validación del instrumento

Antes de exponer los resultados de un instrumento, como en este caso, el cuestionario, se debe validar las respuestas y así obtener el grado de confiabilidad. Se utilizó el coeficiente Alfa de Cronbach para las 12 preguntas, excepto las preguntas 10 y 11, porque no aplica para este caso.

Para calcular el coeficiente de alfa de Cronbach se aplicó la siguiente fórmula:

$$\alpha = \left(\frac{k}{k-1} \right) \left(1 - \frac{\sum v_i}{v_t} \right) \quad (3)$$

Donde:

k = número de ítems.

v_i = varianza del ítem.

$\sum v_i$ = sumatoria de las varianzas del ítem.

v_t = varianza total.

Con la simbología dada, se procedió a reemplazar los valores:

$$\alpha = \left(\frac{10}{10-1} \right) \left(1 - \frac{13,85}{43,56} \right) = 0,76 \quad (4)$$

Tabla 3. Rangos de confiabilidad de alfa de Cronbach.

Alfa de Cronbach	Valoración de confiabilidad
0,9	Excelente
0,9 - 0,8	Buena
0,8 - 0,7	Aceptable
0,7 - 0,6	Débil
0,6 - 0,5	Pobre
< 0.5	Inaceptable

Basando el resultado en la Tabla 2, se tiene que la confiabilidad de los resultados de la encuesta es aceptable con un coeficiente de 0,76.

b. Resultados de las encuestas

La encuesta se llevó a cabo con la participación de 400 personas, tomando una muestra finita de 91 de la población de la zona urbana de la ciudad de Ambato.

Pregunta 1: ¿Cuán satisfecho está con la gestión de mantenimiento de la infraestructura urbana en la ciudad de Ambato?

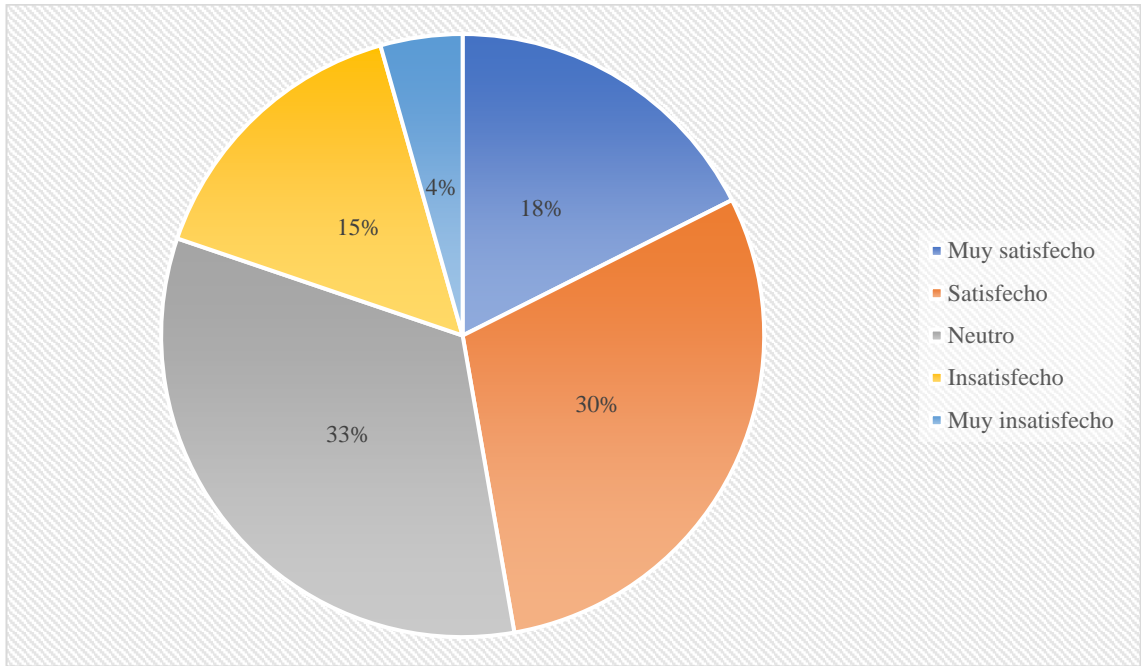


Figura 1. Tabulación de resultados pregunta 1.

Análisis e interpretación de resultados

Al analizar la pregunta 1, respecto a la satisfacción de la gestión de mantenimiento de la infraestructura urbana de la ciudad, se puede observar que el 33% se encuentran en una posición neutral, el 30% está satisfecho, el 18% está muy satisfecho, el 15% está insatisfecho y el 4% está muy insatisfecho. Estos resultados reflejan el grado de satisfacción por parte de los ciudadanos, sobre la gestión de mantenimiento en la ciudad.

Pregunta 2: ¿Con qué frecuencia ha observado problemas en la infraestructura urbana de Ambato en el último año?

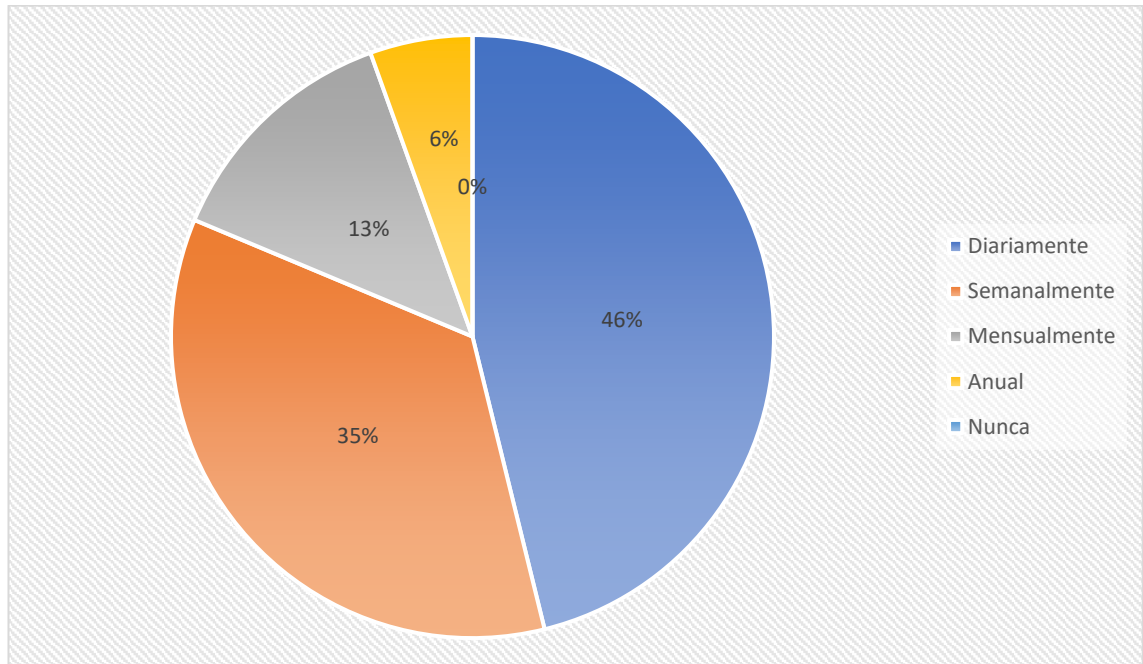


Figura 2. Tabulación de resultados pregunta 2.

Análisis e interpretación de resultados

Al analizar la pregunta 2, respecto a la frecuencia con que los encuestados han observado problemas de infraestructura urbana en la ciudad, se puede observar que el 46% de los encuestados han visto problemas diariamente, el 35% semanalmente, el 13% mensualmente, el 6% anualmente y el 0% nunca. Los resultados muestran que un porcentaje significativo de encuestados ha sido testigo de problemas de infraestructura en la ciudad con una frecuencia considerable.

Pregunta 3: ¿En qué medida cree que una aplicación móvil facilitaría el reporte de problemas en la infraestructura urbana en Ambato?

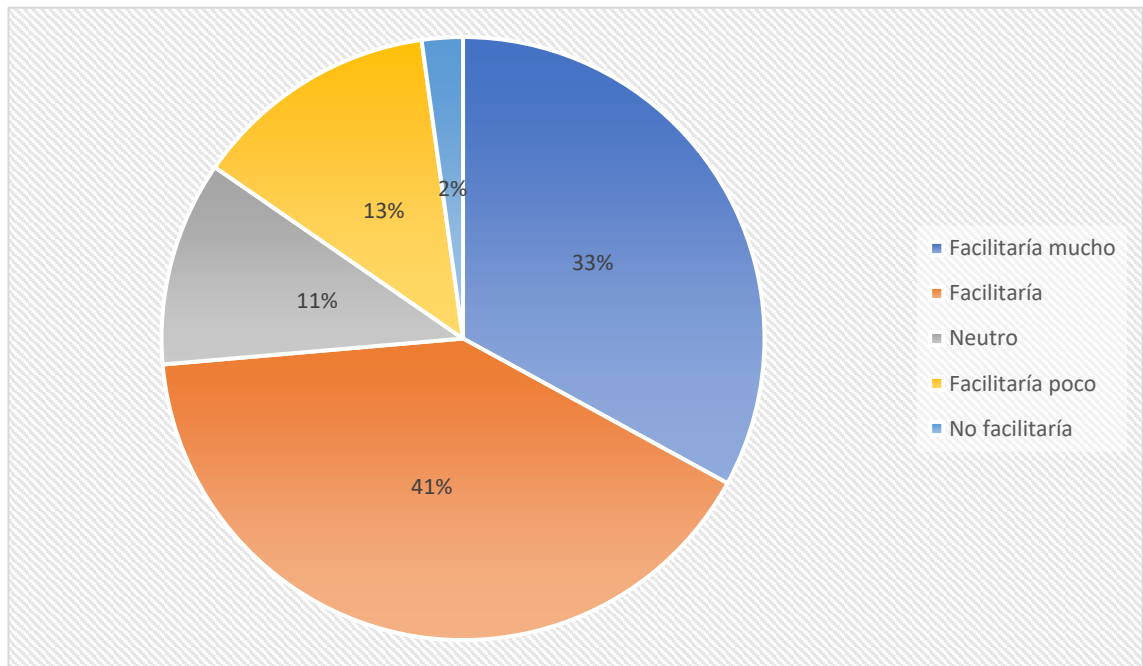


Figura 3. Tabulación de resultados pregunta 3.

Análisis e interpretación de resultados

Al analizar la pregunta 3, respecto a la opinión de en qué medida los encuestados creen que una aplicación móvil facilitaría el reporte de problemas en la infraestructura urbana, se puede observar que el 41% cree que facilitaría el proceso, el 33% opina que facilitaría mucho, el 13% piensa que facilitaría poco, el 11% se mantiene en una posición neutral, y el 2% cree que no facilitaría el proceso. Este hallazgo muestra un alto potencial de adopción y utilidad de esta tecnología para mejorar la eficiencia en el mantenimiento de la infraestructura urbana de la ciudad.

Pregunta 4: ¿Considera que la gestión actual de mantenimiento de la infraestructura urbana en Ambato es eficiente?

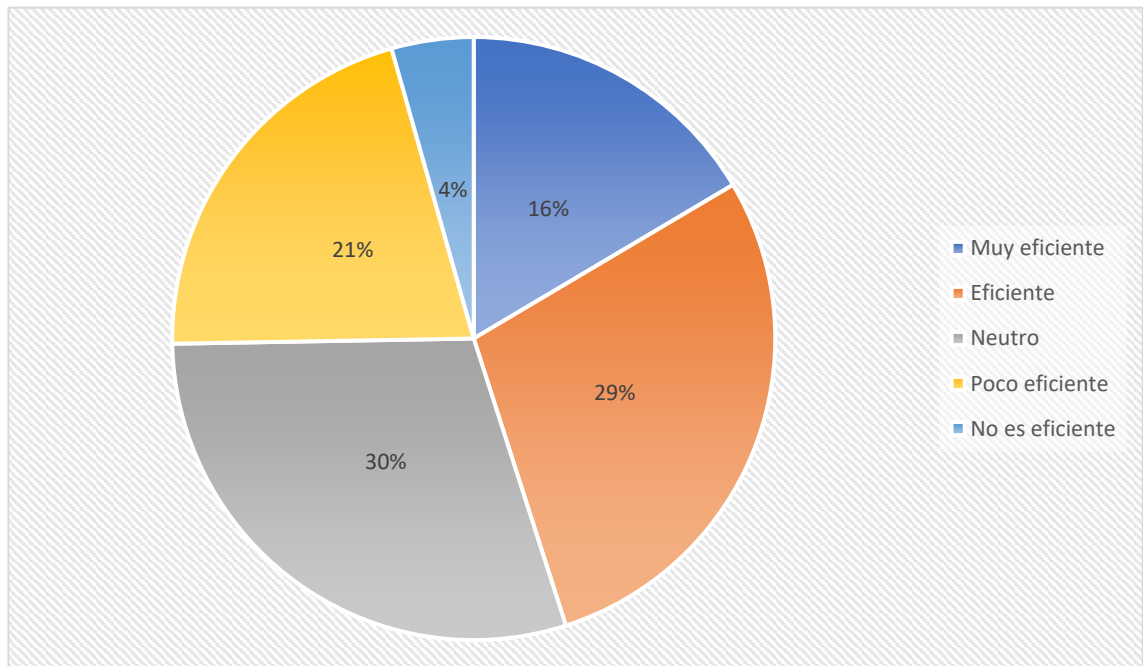


Figura 4. Tabulación de resultados pregunta 4.

Análisis e interpretación de resultados

Al analizar la pregunta 4, respecto a la consideración de los encuestados de si es que la gestión actual de mantenimiento de la infraestructura urbana en Ambato es eficiente, se puede observar que el 30% se encuentra en una posición neutral, el 29% consideran que es eficiente, el 21% piensan que es poco eficiente, el 16% creen que es muy eficiente y el 4% afirman que no es eficiente. Los resultados muestran el grado de eficiencia de la gestión actual del mantenimiento.

Pregunta 5: Si alguna vez ha reportado problemas de infraestructura urbana, ¿ha experimentado retrasos en la resolución de estos después de reportarlos?

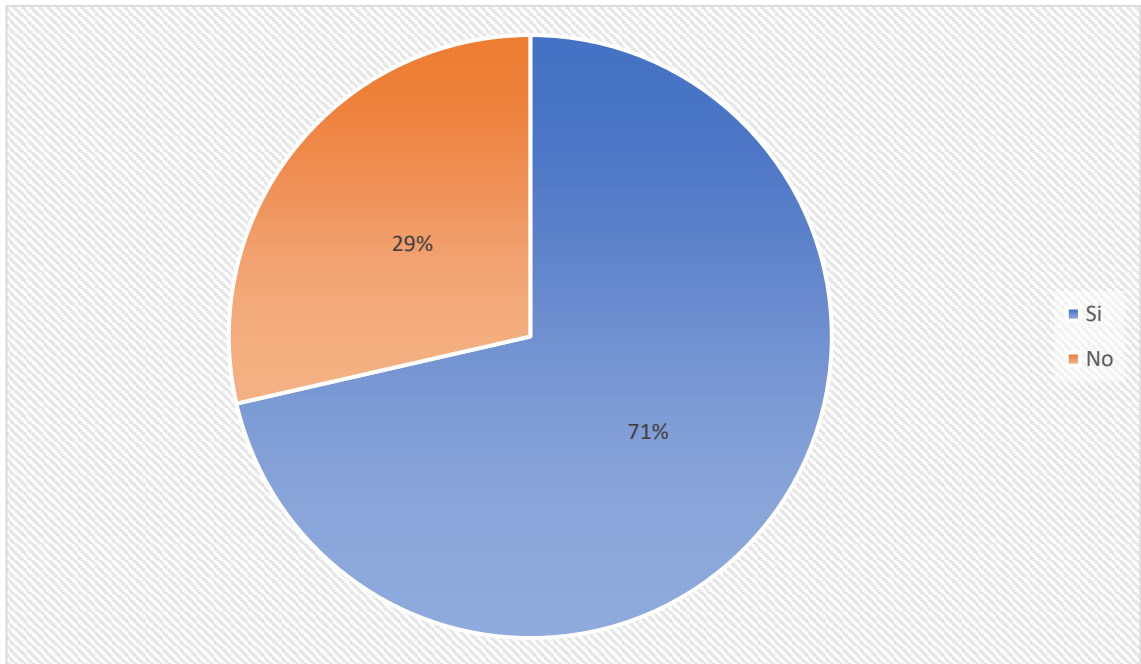


Figura 5. Tabulación de resultados pregunta 5.

Análisis e interpretación de resultados

Al analizar la pregunta 5, respecto a si los encuestados han experimentado retrasos en la resolución de problemas después de haber reportado problemas de infraestructura urbana, se puede observar que el 71% si han sido testigos del retraso en la resolución y el 29% no. Estos resultados muestran que los ciudadanos que han participado en el reporte de problemas, experimentan retrasos en la resolución de estos.

Pregunta 6: ¿Qué tipo de dispositivo electrónico usa con más frecuencia?

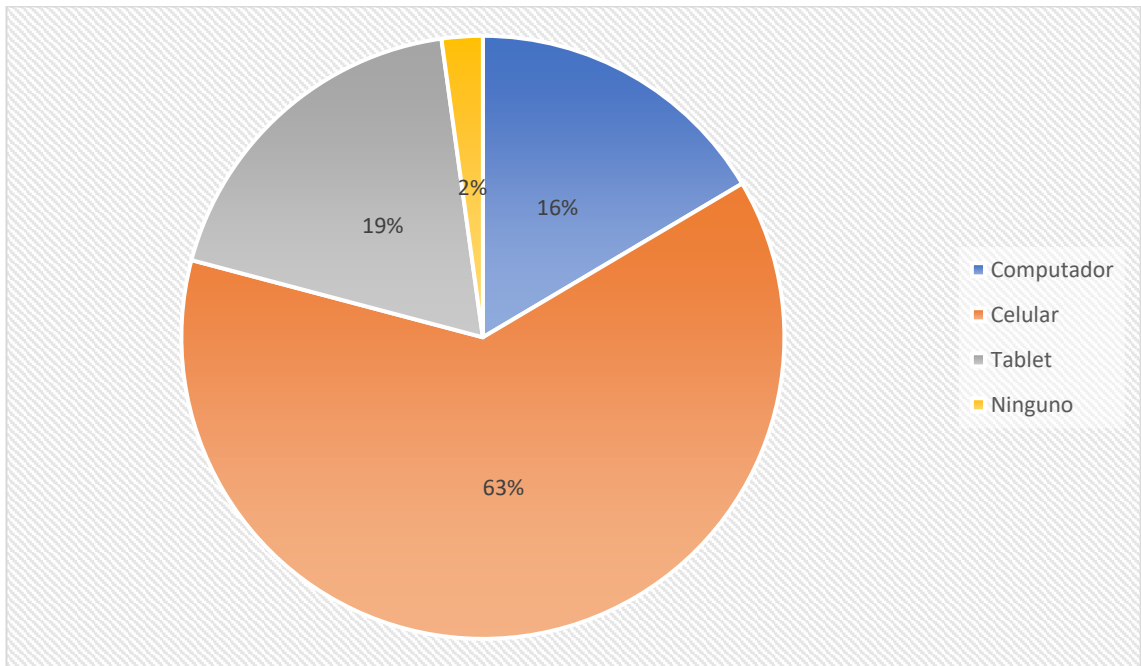


Figura 6. Tabulación de resultados pregunta 6.

Análisis e interpretación de resultados

Al analizar la pregunta 6, respecto al dispositivo que usan los encuestados con más frecuencia, se puede observar que el 63% usan celular, el 19% Tablet, el 16% computador y el 2% no usan ningún dispositivo electrónico de la lista. Estos resultados muestran que la mayoría de usuarios, tienen como dispositivo con mayor frecuencia de uso, un teléfono celular.

Pregunta 7: ¿Estaría dispuesto a utilizar una aplicación móvil para reportar problemas en la infraestructura urbana de Ambato?

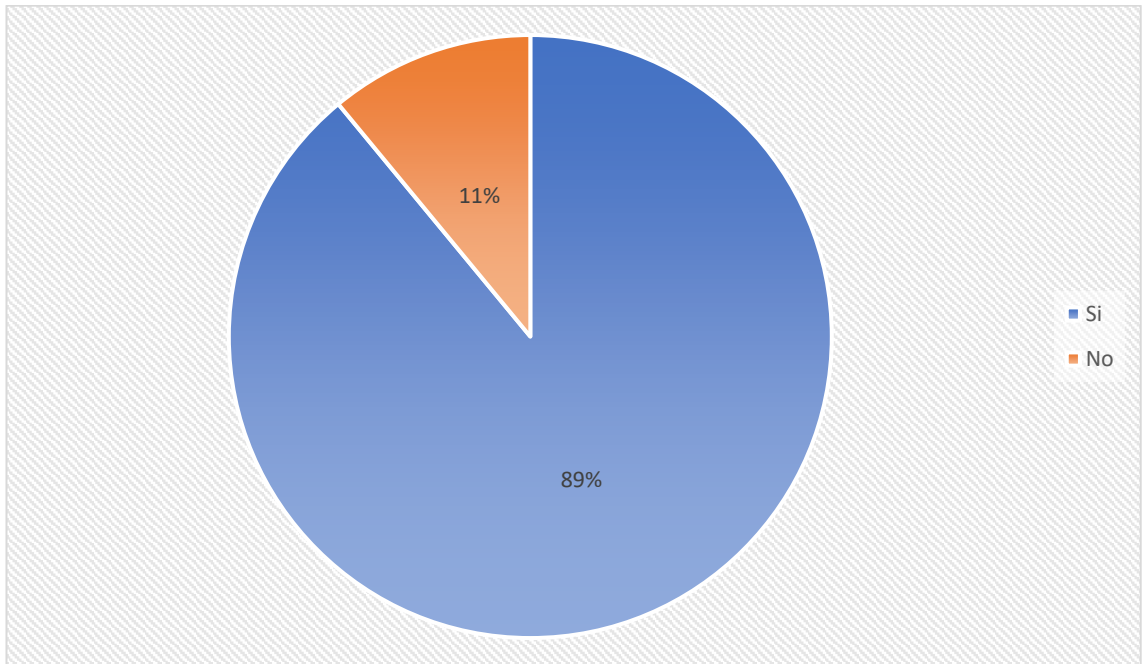


Figura 7. Tabulación de resultados pregunta 7.

Análisis e interpretación de resultados

Al analizar la pregunta 7, respecto a si los encuestados estarían dispuestos a utilizar una aplicación móvil para reportar problemas de infraestructura urbana, se puede observar que el 89% si estarían dispuestos y el 11% no lo están. Los resultados muestran un alto potencial de aceptación y utilidad de esta tecnología para mejorar la participación ciudadana en la resolución de problemas en infraestructura urbana.

Pregunta 8: ¿Cree que una aplicación móvil podría acelerar la respuesta a problemas de infraestructura urbana?

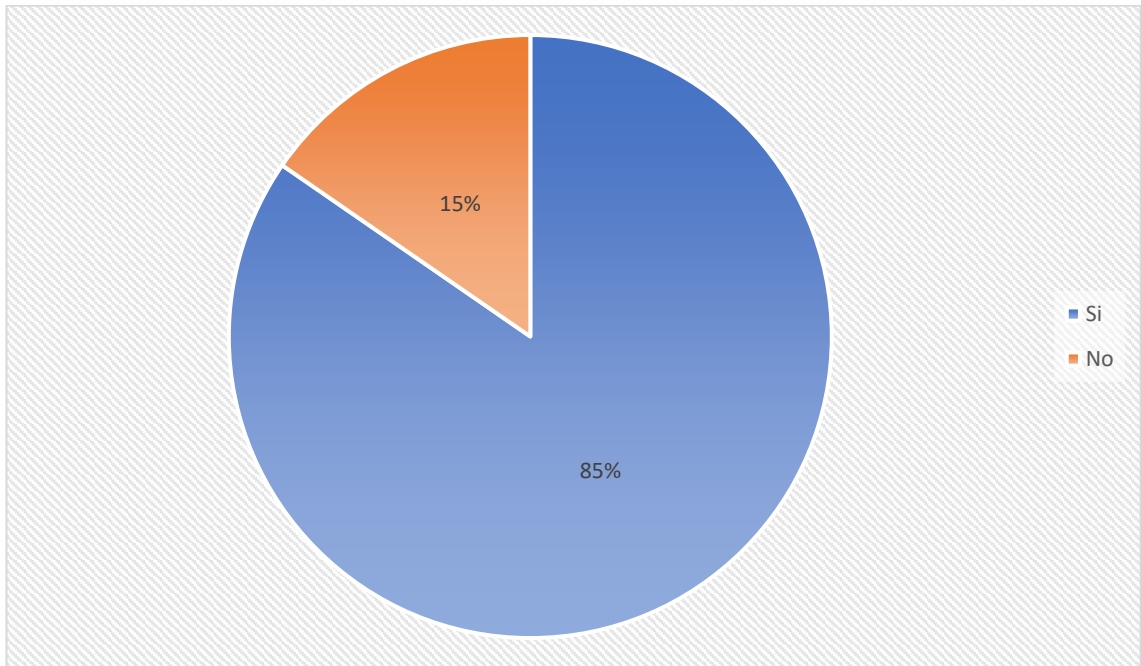


Figura 8. Tabulación de resultados pregunta 8.

Análisis e interpretación de resultados

Al analizar la pregunta 8, respecto a si los encuestados creen que una aplicación móvil podría acelerar la respuesta a problemas de infraestructura urbana, se puede observar que el 85% si creen que podría acelerar la respuesta y el 15% creen que no. Estos resultados resaltan la percepción positiva de la eficacia de la tecnología en la mejora de la gestión de incidencias urbanas.

Pregunta 9: ¿Cuánto tiempo estaría dispuesto a dedicar a la utilización de una aplicación móvil para reportar problemas en la infraestructura urbana por cada reporte?

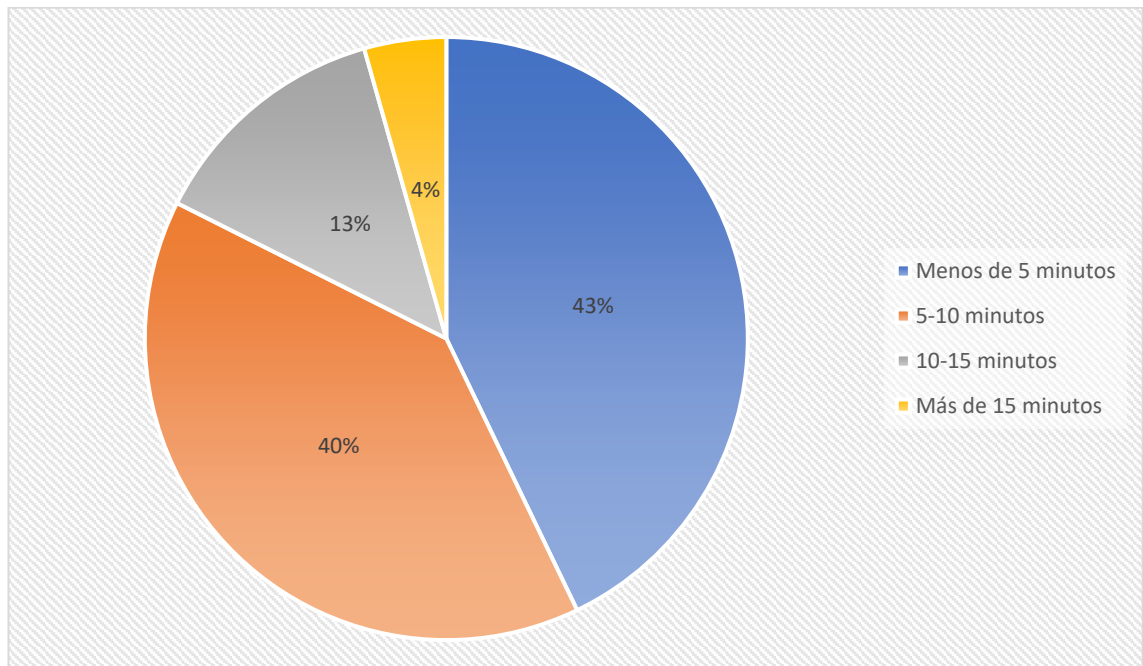


Figura 9. Tabulación de resultados pregunta 9.

Análisis e interpretación de resultados

Al analizar la pregunta 9, respecto al tiempo que los encuestados estarían dispuestos a dedicar a realizar un reporte a través de una aplicación móvil para reportar problemas de infraestructura urbana, se puede observar que el 43% estaría dispuesto a dedicar menos de 5 minutos a reportar, el 40% entre 5 a 10 minutos, el 13% entre 10 y 15 minutos y el 4%, más de 15 minutos. Los resultados a esta pregunta muestran que los usuarios están dispuestos a realizar reportes en lapsos de tiempos cortos.

Pregunta 10: ¿Qué tipo de problemas en la infraestructura urbana le gustaría poder reportar a través de una aplicación móvil?

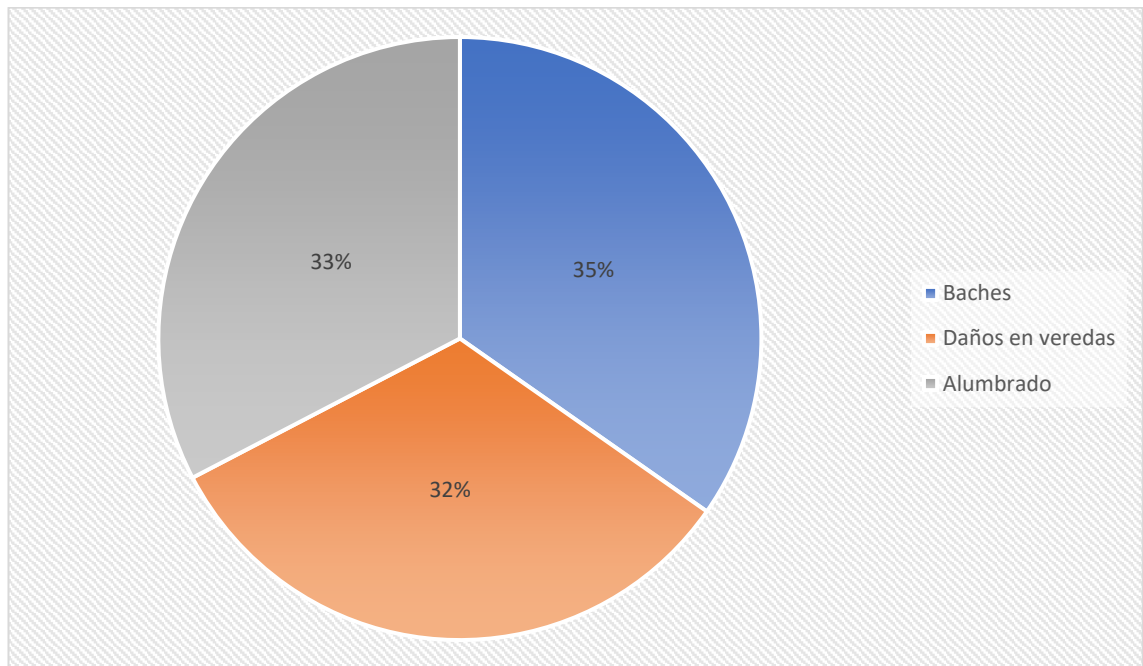


Figura 10. Tabulación de resultados pregunta 10.

Análisis e interpretación de resultados

Al analizar la pregunta 10, respecto a los problemas de infraestructura urbana que les gustaría a los encuestados poder reportar a través de la aplicación móvil, se puede observar que al 35% le gustaría poder reportar baches, al 33% alumbrado y al 32% daños en veredas. Los resultados muestran una inclinación equitativa en cuanto a los tipos de problemas que las personas les gustaría reportar a través de una aplicación móvil.

Pregunta 11: ¿Qué características considera esenciales en una aplicación móvil para reportar problemas en la infraestructura urbana?

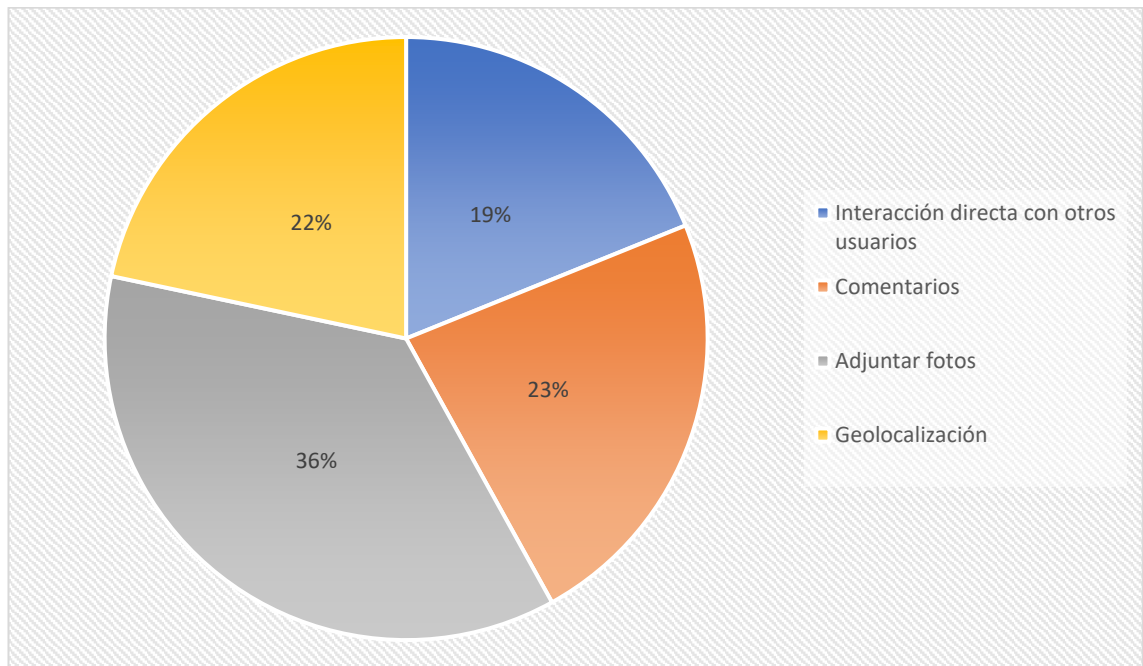


Figura 11. Tabulación de resultados pregunta 11.

Análisis e interpretación de resultados

Al analizar la pregunta 11, respecto a las características esenciales en una aplicación móvil para reportar problemas en la infraestructura urbana de la ciudad, se puede observar que el 36% considera que se debería poder adjuntar fotos, el 23% cree que se deba poder añadir comentarios, el 22% considera que debe tener geolocalización y el 19%, interacción directa con otros usuarios. Los encuestados demuestran interés sobre las características principales que debería tener una aplicación móvil para reportar problemas de infraestructura.

Pregunta 12: ¿Cree que una aplicación móvil podría contribuir a una mejor gestión de la infraestructura urbana de Ambato?

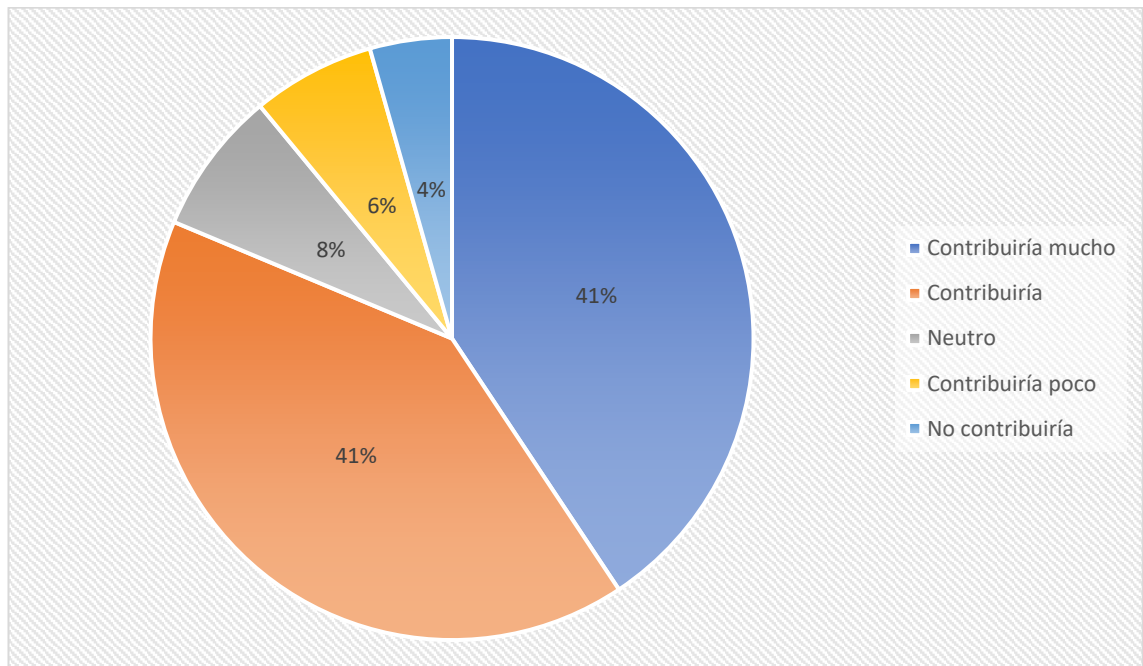


Figura 12. Tabulación de resultados pregunta 12.

Análisis e interpretación de resultados

Al analizar la pregunta 12, respecto a si los encuestados creen que una aplicación móvil podría contribuir a una mejor gestión de la infraestructura urbana de la ciudad, se puede observar que un 41% considera la aplicación móvil contribuiría mucho, otro 41% piensa que contribuiría, el 8% se mantiene en una posición neutral, el 6% cree que contribuiría poco y el 4% afirma que no contribuiría. La percepción general de los encuestados muestra una fuerte confianza en el potencial de esta tecnología.

2.2.4 Procesamiento y análisis de datos

A partir de los datos recopilados durante la recolección de información, se identificaron los siguientes hallazgos:

- La disposición de la mayoría de los encuestados a utilizar una aplicación móvil subraya la importancia de la tecnología como una herramienta efectiva para fomentar la participación activa de la comunidad en la mejora de la infraestructura urbana. Esto sugiere que las autoridades competentes deberían considerar seriamente la implementación de soluciones tecnológicas que promuevan la interacción ciudadana.
- Las diversas opiniones expresadas en las respuestas sobre la satisfacción con la gestión de la infraestructura urbana destacan la necesidad de reconocer y abordar las expectativas y necesidades variadas de la comunidad. Este enfoque podría requerir una adaptación más personalizada en la gestión de la infraestructura para garantizar la satisfacción de una gama más amplia de ciudadanos.
- La percepción de que una aplicación móvil podría acelerar la respuesta a problemas de infraestructura urbana enfatiza la importancia de la eficiencia en la gestión de incidentes. Para satisfacer las expectativas de la comunidad, es necesario optimizar los procesos internos y garantizar una respuesta rápida a los problemas reportados.
- La preferencia por reportar problemas específicos, como baches, alumbrado y daños en veredas, refleja el deseo de los ciudadanos de ser partícipes activos en la mejora de su entorno. Esto indica la necesidad de empoderar a la comunidad y brindarles las herramientas adecuadas para ser una parte activa de la solución, lo que podría mejorar la colaboración entre ciudadanos y autoridades.
- La importancia atribuida a características específicas sugiere la necesidad de desarrollar soluciones que ofrezcan una experiencia completa y fácil de usar. Una aplicación exitosa debe permitir a los usuarios comunicar de manera efectiva los problemas de infraestructura y proporcionar información detallada para su resolución.

CAPÍTULO III. RESULTADOS Y DISCUSIÓN

3.1 Análisis y discusión de resultados

3.1.1 Identificación de problemas de gestión de mantenimiento

Con el objetivo de identificar los problemas de gestión de mantenimiento de la infraestructura urbana, se accedió a fuentes directas proporcionadas por el Gobierno Autónomo Descentralizado (GAD) de Ambato. Estas fuentes, provenientes de la entidad encargada de la administración y supervisión de la infraestructura urbana y de la percepción ciudadana sobre la gestión y resolución de problemas, ofrecieron una perspectiva detallada sobre los desafíos actuales en el mantenimiento de la ciudad.

Tabla 4. Problemas de gestión de mantenimiento de la infraestructura urbana [23].

Problema	Descripción	Consecuencias
Escasez de Recursos	Limitación en presupuestos asignados para mantenimiento.	Dificultad para abordar problemas de manera efectiva.
Períodos Largos de Resolución	Para que se realice un mantenimiento, debe pasar por un largo proceso de informes antes de ir a la solución.	Más tiempo para abordar y resolver los problemas.
Falta de un Canal de Comunicación Directo	Ausencia de un canal eficaz para reportar problemas.	Retrasos en la identificación y resolución de problemas.

3.1.2 Identificación de problemas de infraestructura urbana

Mediante la recolección de información dirigida a los residentes de la zona urbana, se logró identificar problemas en la infraestructura urbana de Ambato. La retroalimentación directa de los ciudadanos proporcionó una comprensión completa de la magnitud y complejidad de estos problemas, facilitando la formulación de estrategias para mejorar la infraestructura urbana.

Tabla 5. Problemas de infraestructura urbana frecuentes.

Problema	Descripción	Frecuencia	Consecuencias
Baches	Superficies con depresiones en el pavimento.	Alta incidencia	Riesgo de daños vehiculares, accidentes y peligro peatonal.
Calles Dañadas	Pavimento agrietado, hundido o levantado.	Puntos críticos identificados	Dificultad en la circulación, aumento de costos de mantenimiento y riesgo de accidentes.
Veredas Deterioradas	Aceras con desgaste, agrietadas o mal conservadas.	Problemas recurrentes	Riesgo de tropiezos, afectación estética y deterioro de la calidad de vida.
Postes Caídos	Postes de servicios públicos en colapso.	Incidentes reportados	Peligro para la seguridad pública y posibles interrupciones en los servicios.

Iluminación Deficiente	Falta de iluminación en áreas urbanas.	Identificada en varias zonas	Aumento de la inseguridad, riesgo de accidentes y disminución de la calidad de vida.
Sistemas de Drenaje Ineficientes	Problemas de inundaciones y acumulación de agua.	Épocas de lluvia	Daños a la infraestructura, riesgo de inundaciones y afectación a la movilidad.
Señalización Insuficiente	Ausencia de señalización adecuada en vías.	Observada en puntos críticos	Riesgo de accidentes de tráfico y dificultad en la navegación vehicular y peatonal.

3.1.3 Herramientas de desarrollo de software móvil

Para establecer las herramientas de desarrollo de software móvil adecuadas para este proyecto, se consideraron varios aspectos que mejor se adapten, como la plataforma e integración de servicios que demanda la aplicación, escalabilidad, rendimiento, seguridad, entre otros factores importantes. La comparativa de entornos de desarrollo y frameworks define las herramientas que se acoplan a la construcción del proyecto.

Android Studio

Respaldado por Google, es un entorno de desarrollo integrado (IDE) especializado para la creación de aplicaciones Android. Basado en IntelliJ IDEA, utiliza Java y Kotlin como lenguajes de programación principales. Proporciona un conjunto completo de herramientas que incluye emuladores de dispositivos Android y una interfaz gráfica para el diseño de interfaces de usuario.

React Native

Desarrollado por Facebook, es un marco de desarrollo de código abierto que utiliza JavaScript y React. Su principal ventaja radica en la creación eficiente de aplicaciones multiplataforma, permitiendo el uso de un mismo código base para iOS y Android. React Native emplea componentes nativos para la interfaz de usuario y facilita la integración de código nativo cuando es necesario.

Flutter

Creado por Google, es un SDK de código abierto para el desarrollo de aplicaciones móviles en iOS y Android desde un solo código base. Utiliza el lenguaje de programación Dart y se destaca por su rendimiento rápido y la creación de interfaces de usuario atractivas mediante un conjunto de widgets personalizables [24].

Xamarin

Respaldado por Microsoft, permite la escritura de aplicaciones nativas compartiendo gran parte del código entre plataformas como iOS y Android. Proporciona acceso total a las API nativas y ofrece flexibilidad con Xamarin Forms para interfaces compartidas o Xamarin Native para un control más detallado de la interfaz de usuario.

Tabla 6. Comparativa entre entornos de desarrollo y frameworks.

Criterios de Evaluación	Android Studio	React Native	Flutter	Xamarin
Facilidad de Uso	Fácil	Neutro	Fácil	Neutro
Compatibilidad de Plataforma	Android/iOS	Android/iOS	Android/iOS	Android/iOS

Integración de Mapas	Google Maps	Mapbox	Mapbox/Google Maps	Google Maps
Lenguaje de Programación	Java/Kotlin	JavaScript	Dart	C#
Tiempo de Desarrollo	Medio	Medio	Corto	Medio
Comunidad y Soporte	Buena	Buena	Buena	Buena
Documentación	Buena	Buena	Excelente	Buena
Rendimiento	Bueno	Bueno	Excelente	Bueno
Escalabilidad	Escalable	Neutro	Muy escalable	Neutro
Librerías Disponibles	Amplias	Amplias	Amplias	Amplias

Seguridad	Buena	Buena	Buena	Buena
Licencia	Libre	Libre	Libre	Libre (limitado)

Al analizar la Tabla 6, que detalla características importantes de cada tecnología, se optó por Flutter para el desarrollo de la aplicación móvil tras evaluar criterios importantes. La facilidad de uso y la compatibilidad multiplataforma de Flutter aseguran una experiencia de desarrollo accesible y una amplia cobertura de audiencia.

3.1.4 Sistemas Gestores de Bases de Datos

Para determinar los Sistemas Gestores de Bases de Datos (SGBD) más apropiados para este proyecto, es esencial tener en cuenta diversos aspectos que se ajusten de manera óptima a las necesidades de la aplicación y su integración con servicios específicos. La evaluación comparativa de SGBD y entornos de desarrollo establece los sistemas que mejor se adaptan a la implementación del proyecto.

Firestore

Es una plataforma de desarrollo móvil y web respaldada por Google. Ofrece un conjunto integral de servicios, incluyendo una base de datos en tiempo real, autenticación, hosting, y funciones en la nube. La base de datos en tiempo real de Firestore utiliza un modelo NoSQL basado en documentos y está diseñada para proporcionar actualizaciones en tiempo real, lo que la hace ideal para aplicaciones que requieren sincronización instantánea de datos.

PostgreSQL

Es un sistema de gestión de bases de datos relacional de código abierto conocido por su confiabilidad y capacidad para manejar grandes volúmenes de datos. Soporta

características avanzadas como transacciones ACID, consultas complejas, y es extensible mediante la creación de funciones y procedimientos almacenados. PostgreSQL es una opción sólida para aplicaciones que requieren un modelo de datos relacional y escalabilidad.

SQLite

Es un motor de base de datos relacional ligero y sin servidor que se implementa como una biblioteca en la aplicación. Destaca por su simplicidad y eficiencia, siendo ideal para aplicaciones móviles y embebidas. Aunque es más adecuado para casos de uso con cargas de trabajo más ligeras, ofrece transacciones ACID y soporte para consultas SQL estándar.

MySQL

Es un sistema de gestión de bases de datos relacional ampliamente utilizado. Es conocido por su rendimiento, escalabilidad y compatibilidad con estándares SQL. Ofrece características como transacciones ACID, replicación, y almacenamiento en caché, lo que lo hace apto para aplicaciones que requieren un manejo eficiente de grandes cantidades de datos y una rápida respuesta a consultas complejas.

Tabla 7. Comparativa entre Sistemas de Gestión de Bases de Datos.

Criterios de Evaluación	Firebase	PostgreSQL	SQLite	MySQL
Tipo de Base de Datos	NoSQL	SQL	SQL	SQL
Escalabilidad	Excelente	Buena	Buena	Buena

Facilidad de Uso	Fácil	Neutro	Fácil	Neutro
Integración con la Aplicación	Buena	Buena	Buena	Buena
Rendimiento	Bueno	Excelente	Bueno	Bueno
Seguridad	Buena	Buena	Buena	Buena
Modelo de Datos	Documentos JSON	Tablas Relacionales	Tablas Relacionales	Tablas Relacionales
Lenguaje de Consulta	NoSQL Query Language	SQL	SQL	SQL
Replicación de Datos	Sí	Sí	No	Sí

Escritura en Tiempo Real	Sí	No	No	No
Costo	Libre (limitado)	Libre	Libre	Libre
Comunidad y Soporte	Excelente	Buena	Buena	Buena
Documentación	Excelente	Buena	Buena	Buena
Escalabilidad Horizontal	Sí	Sí	No	Sí
Uso en Aplicaciones Móviles	Muy común	Común	Común	Común

Al analizar la Tabla 7, se define la adopción de Firebase como sistema de gestión de bases de datos (SGBD) para el presente proyecto que se fundamenta en una evaluación de criterios esenciales, delineando una elección informada que optimiza el rendimiento y la eficiencia del desarrollo. La naturaleza NoSQL de Firebase, respaldada por su capacidad para almacenar y recuperar datos en documentos JSON, brinda flexibilidad

y agilidad en la manipulación de información, alineándose perfectamente con la diversidad de datos inherente a las aplicaciones modernas. Su escalabilidad sobresaliente es vital para garantizar la adaptabilidad del sistema a futuros crecimientos en la demanda de datos, asegurando una experiencia del usuario consistente y sin inconvenientes. Además, Firebase se integra de manera excelente con Flutter, permitiendo así tiempos de respuesta rápidos en la comunicación de datos.

3.1.5 Técnicas y herramientas de geolocalización

En la elección de las técnicas y herramientas de geolocalización para este proyecto, se buscó una solución eficiente y precisa. La selección de las opciones más adecuadas se llevó a cabo considerando factores importantes, asegurando así una implementación que se alinee de manera adecuada con las necesidades específicas de la aplicación.

GPS

Utiliza señales de satélite para determinar la ubicación precisa del dispositivo.

Geolocalización por IP

Determina la ubicación basándose en la dirección IP del dispositivo.

Wi-Fi y Red Móvil

Utiliza información de torres de señal y puntos de acceso Wi-Fi para estimar la ubicación.

Bluetooth

Utiliza señales Bluetooth para determinar la proximidad a dispositivos específicos.

Tabla 8. Comparativa entre técnicas de geolocalización.

Técnica	Precisión	Requerimientos Especiales	Uso en Flutter
GPS	Alta	Hardware compatible	Paquete location

Geolocalización por IP	Baja-Media	Acceso a servicios externos	Integración de APIs
Wi-Fi y Red Móvil	Media	Acceso a información de red	Paquete connectivity
Bluetooth	Media	Dispositivos compatibles	Paquete flutter_blue

En la estrategia de geolocalización para este proyecto, se ha optado por la técnica GPS debido a su alta precisión, requisito fundamental para cumplir con las necesidades específicas de la aplicación. Esta elección asegura la obtención confiable de datos geospaciales, contribuyendo a una implementación sólida y alineada con los estándares de calidad del proyecto.

A partir de la elección de la técnica GPS para la geolocalización en este proyecto, se procedió a considerar diversas opciones de herramientas. La evaluación detallada de estas herramientas se basa en criterios específicos, incluyendo la capacidad de integración con Flutter, características fundamentales, posibilidades de personalización y costos asociados. Este proceso es fundamental para la selección de la herramienta más idónea, proporcionando así una base sólida para la posterior tabla comparativa.

Google Maps

Ofrece un completo SDK para Flutter que facilita la integración de mapas interactivos en aplicaciones. Proporciona funciones avanzadas de geolocalización, rutas y visualización de mapas, con amplias opciones de personalización. Google Maps es conocido por su cobertura global y proporciona servicios como geo codificación y navegación.

Mapbox

Ofrece un SDK para Flutter que permite integrar mapas altamente personalizables. Es conocido por su diseño atractivo y ofrece funciones avanzadas como mapas vectoriales

y capacidad de diseño personalizado. Mapbox es utilizado en aplicaciones que requieren un alto grado de personalización y visualización de mapas innovadora.

Leaflet

Es una biblioteca de JavaScript ligera y eficiente que se utiliza con Flutter mediante el paquete flutter_leaflet. Aunque es ampliamente conocido en entornos web, la integración con Flutter permite su uso en aplicaciones móviles. Leaflet se destaca por su simplicidad, siendo una opción adecuada para proyectos que buscan una solución ligera y fácil de implementar.

Tabla 9. Comparativa entre herramientas de geolocalización.

Criterios de Evaluación	Google Maps Flutter SDK	Mapbox	Leaflet
Integración con Flutter	Fácil	Moderadamente sencilla	Sencilla
Funcionalidades Clave	Completa	Amplias funciones	Biblioteca de mapas
Documentación y Soporte	Extensa	Detallada, comunidad activa	Sólida, comunidad amplia
Personalización	Limitada	Altamente personalizable	Totalmente personalizable
Costo	Planes según uso	Planes según uso	Libre

Se seleccionó Google Maps para este proyecto debido a su extensa cobertura global, fiabilidad demostrada en la integración con Flutter y su capacidad para ofrecer funciones avanzadas de geolocalización y rutas. La elección se basó en la necesidad de contar con una solución que combine eficacia y versatilidad para cumplir con los requerimientos específicos de la aplicación, asegurando así una implementación exitosa.

3.1.6 Metodologías para el desarrollo móvil

La siguiente tabla comparativa presenta las particularidades más significativas de cada metodología ágil, con el objetivo de resaltar la entrega constante y rápida de resultados.

Tabla 10. Comparativa entre metodologías ágiles.

Criterios de Evaluación	XP (Programación Extrema)	Scrumban	XP con RAD	RAD (Desarrollo Rápido de Aplicaciones)	Mobile D
Tamaño de los Proyectos	Pequeños a Medianos	Todos los tamaños	Pequeños a Medianos	Pequeños a Medianos	Todos los tamaños
Tamaño de Equipo	Pequeño	Variable	Pequeño a Mediano	Variable	Variable
Estilo de Desarrollo	Ágil	Híbrido (Scrum y Kanban)	Ágil con énfasis en desarrollo rápido	Ágil con énfasis en desarrollo rápido	Variable
Estilo de Código	Mantenimiento continuo y refactorización constante	Enfoque combinado de Scrum y Kanban	Enfoque rápido y adaptativo	Iterativo y flexible	Iterativo y adaptativo
Fases o Etapas	Iterativo	Iterativo	Iterativo	Iterativo y Adaptativo	Iterativo y Adaptativo
Tiempo de desarrollo	Mediano	Mediano	Mediano	Mediano	Corto
Complejidad	Baja a Moderada	Variable	Baja a Moderada	Baja a Moderada	Variable

Mobile D fue seleccionado para este proyecto debido a su enfoque ágil y adaptativo, que se alinea con la naturaleza dinámica de las aplicaciones móviles. Su capacidad

para ajustarse a cambios rápidos y su énfasis en la entrega continua fueron factores clave en la elección, teniendo como limitación, el tiempo de desarrollo del proyecto. Además, la capacidad de Mobile D para adaptarse a las características específicas del proyecto, priorizando la eficiencia en el desarrollo y la satisfacción del usuario final, fue determinante en su elección.

3.2 Desarrollo de la propuesta

Para desarrollar la propuesta del presente proyecto, se tiene conocimiento de las 5 fases de la metodología Mobile-D, las cuales son exploración, iniciación, producción, estabilización y prueba del sistema.

3.2.1 Fase 1: Exploración

En esta fase, el o los desarrolladores deben crear un plan y definir las características que va a tener el proyecto. Esto se realiza en tres etapas, como lo son establecer interesados, definir el alcance y establecer proyectos.

a. *Establecimiento de los interesados (stakeholders)*

Tabla 11. Establecimiento de los interesados.

Interesados	Descripción
Desarrollador	Autor y desarrollador de la solución informática del presente proyecto.
Usuarios	Habitantes de la zona urbana de la ciudad de Ambato.

b. *Definición del alcance*

• *Limitaciones*

Las limitaciones que tiene la aplicación son:

- Acceso a internet para tener las funcionalidades completas de la aplicación.
- Dispositivo con la versión de Android 6.0 o superior.

- ***Establecimiento de dependencias***

Para este proyecto, existe la siguiente dependencia:

- La plataforma móvil está destinada exclusivamente para realizar reportes de problemas de infraestructura urbana dentro de la ciudad de Ambato.

- ***Requisitos funcionales para los usuarios comunes***

- Iniciar sesión
- Recuperar contraseña
- Registrar
- Editar información personal
- Cambiar contraseña
- Acceso a reportes de otros usuarios
- Reaccionar a reportes
- Comentar reportes
- Trazar ruta desde la ubicación actual hasta el reporte
- Visualizar sus reportes
- Editar sus reportes
- Eliminar sus reportes
- Información general sobre reportes
- Generar reporte
- Visualizar notificaciones sobre sus reportes
- Cerrar Sesión

- ***Requisitos funcionales para administradores***

- Iniciar sesión
- Recuperar contraseña
- Registrar
- Editar información personal
- Cambiar contraseña
- Acceso a reportes de otros usuarios
- Reaccionar a reportes

- Comentar reportes
- Trazar ruta desde la ubicación actual hasta el reporte
- Visualizar sus reportes
- Editar sus reportes
- Eliminar sus reportes
- Información general sobre reportes
- Aprobar/Rechazar reportes de usuarios comunes
- Registrar a otros administradores
- Generar reporte
- Visualizar notificaciones sobre sus reportes
- Cerrar Sesión

c. Establecimiento del proyecto

En este proyecto se hará uso de las siguientes herramientas:

- *Visual Studio Code*

Es un editor de código fuente ligero y potente desarrollado por Microsoft. Destaca por su amplia gama de extensiones, integración con Git y herramientas avanzadas de depuración, lo que lo convierte en una elección popular entre los desarrolladores.

- *Flutter*

Un framework de código abierto desarrollado por Google para crear aplicaciones nativas con una única base de código. Utiliza el lenguaje de programación Dart y destaca por su rendimiento y facilidad de personalización de la interfaz de usuario.

- *Dart*

Lenguaje de programación desarrollado por Google, utilizado principalmente con el framework Flutter para construir aplicaciones móviles. Es fuertemente tipado y orientado a objetos, diseñado para ser rápido y fácil de aprender.

- ***Firestore Database***

Una base de datos NoSQL en tiempo real basada en la nube ofrecida por Firebase, que facilita el almacenamiento y sincronización de datos entre aplicaciones web y móviles en tiempo real.

- ***Authentication***

Un servicio de Firebase que proporciona una fácil integración de autenticación en aplicaciones mediante la gestión de usuarios, permitiendo la autenticación mediante correos electrónicos, contraseñas, redes sociales y más.

- ***Storage***

Ofrece almacenamiento en la nube para archivos de aplicaciones, permitiendo el fácil manejo de imágenes, videos y otros recursos. Es parte del ecosistema Firebase y se integra bien con otras herramientas de Firebase.

- ***Google Maps***

Una plataforma de servicios de mapas ofrecida por Google que permite la integración de mapas interactivos en aplicaciones web y móviles. Proporciona diversas API para funciones como geolocalización, rutas y visualización de mapas.

3.2.2 Fase 2: Inicialización

En esta fase se debe preparar y reconocer los recursos que serán necesarios para el desarrollo.

a. Configuración del ambiente de desarrollo

La instalación y configuración de las siguientes herramientas fueron necesarias para facilitar el desarrollo del proyecto.

- **Tipo de aplicación:** Multiplataforma
- **Framework:** Flutter
- **Servicios de autenticación:** Firebase Authentication
- **Sistema gestor de base de datos:** Firestore Database

- **Preparación del ambiente**

Las herramientas instaladas para el desarrollo de la aplicación móvil son las siguientes:

- Visual Studio Code
- Framework Flutter
- Lenguaje Dart
- Firebase

b. Planificación de fases

Tabla 12. Planificación de fases de la metodología Mobile-D.

Fase	Iteración	Descripción
Exploración	Iteración 0	<ul style="list-style-type: none"> • Establecer stakeholders • Definir el alcance • Establecer el proyecto
Iniciación	Iteración 1	<ul style="list-style-type: none"> • Configuración del ambiente de desarrollo • Preparación del ambiente
Producción	Iteración 2	<ul style="list-style-type: none"> • Desarrollar la funcionalidad de inicio de sesión, cerrar sesión y registrar usuarios • Creación de storycard • Pruebas de funcionalidad
	Iteración 3	<ul style="list-style-type: none"> • Desarrollar la funcionalidad de recuperar contraseña • Creación de storycard • Pruebas de funcionalidad
	Iteración 4	<ul style="list-style-type: none"> • Desarrollar el módulo de perfil de usuario • Creación de storycard • Pruebas de funcionalidad

	Iteración 5	<ul style="list-style-type: none"> • Desarrollar la funcionalidad de editar perfil y contraseña de usuario • Creación de storycard • Pruebas de funcionalidad
	Iteración 6	<ul style="list-style-type: none"> • Desarrollar la funcionalidad de generar reportes • Creación de storycard • Pruebas de funcionalidad
	Iteración 7	<ul style="list-style-type: none"> • Desarrollar el módulo de visualizar los reportes del usuario • Creación de storycard • Pruebas de funcionalidad
	Iteración 8	<ul style="list-style-type: none"> • Desarrollar la funcionalidad de editar y borrar los reportes del usuario • Creación de storycard • Pruebas de funcionalidad
	Iteración 9	<ul style="list-style-type: none"> • Desarrollar el módulo para visualizar reportes de otros usuarios • Creación de storycard • Pruebas de funcionalidad
	Iteración 10	<ul style="list-style-type: none"> • Desarrollar el módulo para visualizar los detalles de cada reporte • Creación de storycard • Pruebas de funcionalidad
	Iteración 11	<ul style="list-style-type: none"> • Desarrollar la funcionalidad para reaccionar y comentar reportes • Creación de storycard

		<ul style="list-style-type: none"> • Pruebas de funcionalidad
	Iteración 12	<ul style="list-style-type: none"> • Desarrollar la funcionalidad de geolocalización y mapas • Creación de storycard • Pruebas de funcionalidad
	Iteración 13	<ul style="list-style-type: none"> • Desarrollar la funcionalidad de aprobación de reportes • Creación de storycard • Pruebas de funcionalidad
	Iteración 14	<ul style="list-style-type: none"> • Desarrollar la funcionalidad de registrar administradores • Creación de storycard • Pruebas de funcionalidad
	Iteración 15	<ul style="list-style-type: none"> • Desarrollar el módulo de visualizar notificaciones sobre reportes • Creación de storycard • Pruebas de funcionalidad
	Iteración 16	<ul style="list-style-type: none"> • Desarrollar el módulo de información general sobre reportes • Creación de storycard • Pruebas de funcionalidad
Estabilización	Iteración 17	<ul style="list-style-type: none"> • Mejoras en el diseño de la interfaz de usuario • Corrección de errores • Cambios en la base de datos
Pruebas del sistema	Iteración 18	<ul style="list-style-type: none"> • Pruebas acordes a las funciones necesarias de la plataforma móvil

c. Arquitectura de la aplicación móvil

La arquitectura de la plataforma móvil tiene un diseño intuitivo para el usuario final para que este pueda utilizarlo con la menor cantidad de pasos, dando las funcionalidades necesarias para que sea un sistema usable. Los roles de usuario limitarán o añadirán funciones las cuales requieren conexión a internet para poder hacer uso de ellas.

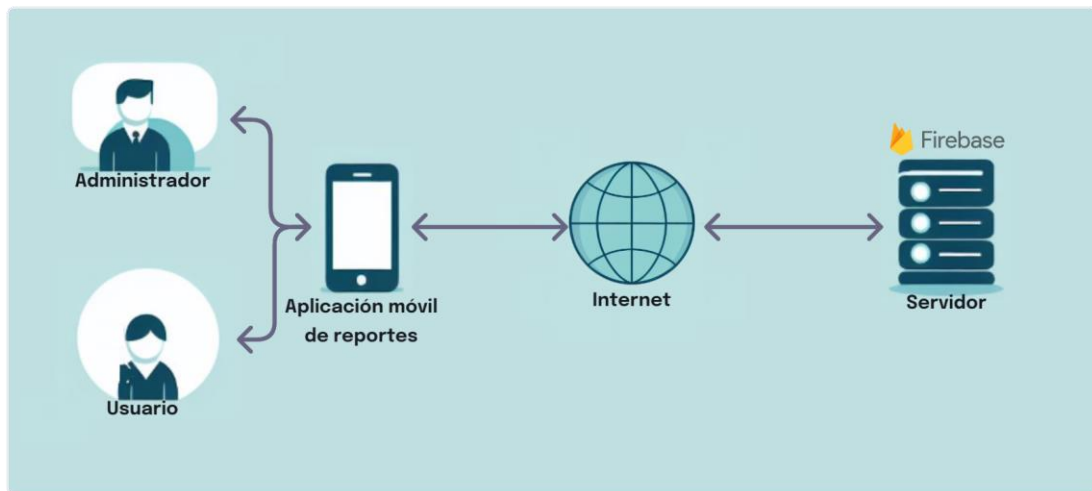


Figura 13. Arquitectura de la aplicación móvil.

d. *Diseño de la base de datos*

Para este proyecto, se utiliza el sistema gestor de base de datos de Firebase, Firestore Database, que es de carácter no relacional, y tiene como características principales las colecciones y documentos.

- *Colecciones*

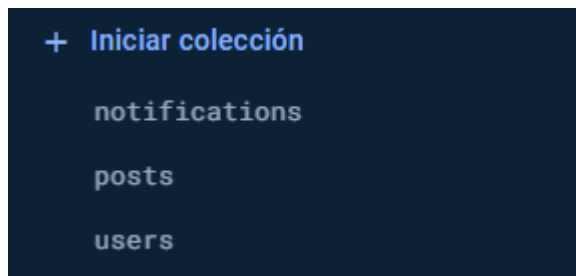


Figura 14. Colecciones del proyecto en Firestore Database.

En la Figura 14 se describen las colecciones que se implementaron para procesar la información necesaria.

- **Colección notifications**

Campo	Tipo de dato
description	String
postId	String
read	String
status	String
timestamp	DateTime
title	String
username	String

- **Colección posts**

Campo	Tipo de dato
category	String
commentCount	Number
coordinates	Map
dateTime	DateTime
description	String
imageUrl	List
location	String
reactions	Number
status	Number
title	String
username	String

- **Subcolección comments**

Campo	Tipo de dato
text	String
timestamp	DateTime
username	String

- **Colección users**

Campo	Tipo de dato
apellido	String
cedula	String

countPosts	Number
email	String
imageUrl	String
nombre	String
password	String
rol	String
username	String

- **Firestore Storage**

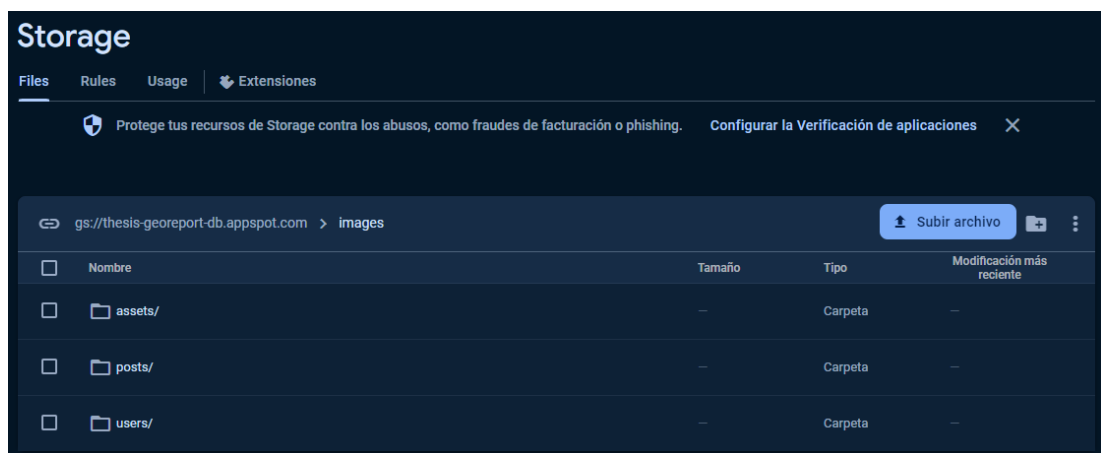


Figura 15. Almacenamiento de imágenes en Firebase Storage.

En la Figura 15 se muestran las rutas principales que gestionan el almacenamiento de imágenes de la aplicación.

e. Esquema de navegabilidad

Se muestra la representación del flujo de navegación en la arquitectura de la aplicación móvil, dando una mejor comprensión del funcionamiento al usuario final.

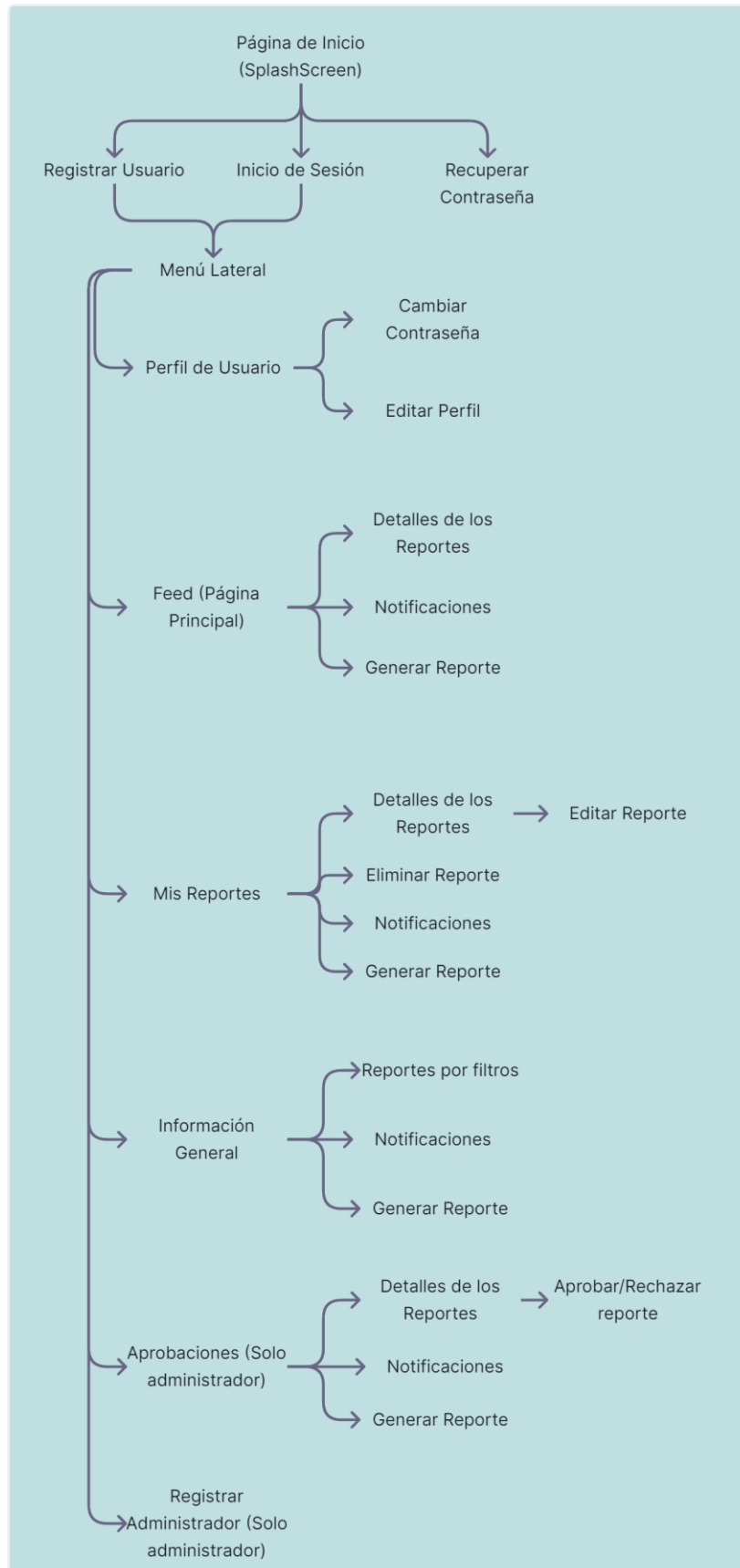


Figura 16. Esquema de navegabilidad de la aplicación móvil.

f. Diagramas de casos de uso

- *Diagrama de caso de uso para usuarios comunes*

El diagrama describe las funciones que los usuarios comunes pueden realizar dentro de la aplicación.



Figura 17. Diagrama de caso de uso usuario común.

- *Diagrama de caso de uso para administradores*

El diagrama describe las funciones adicionales que los administradores pueden realizar dentro de la aplicación, respecto a los usuarios comunes.

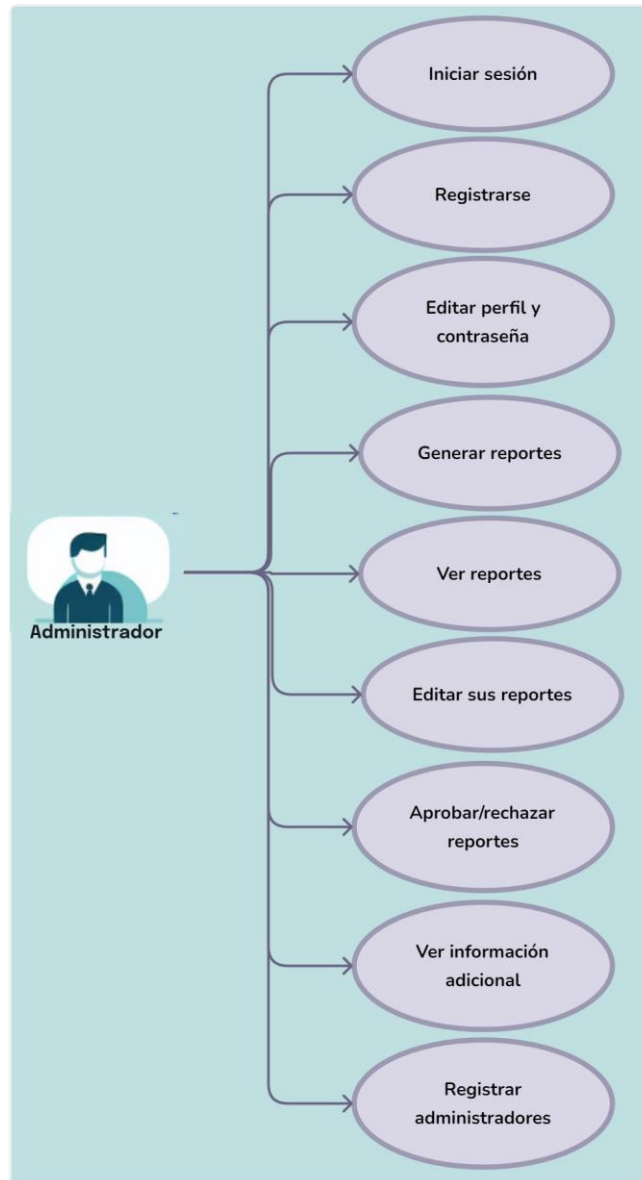


Figura 18. Diagrama de caso de uso administrador.

g. Prototipo de la aplicación

Antes de empezar con el desarrollo inicial de la aplicación móvil, teniendo en cuenta que la interfaz es muy importante en la experiencia de usuario, se ha diseñado un prototipo con la herramienta web Visily y así tener una previsualización de cómo se verá la aplicación al final del desarrollo.

- *Página de inicio de sesión*

Logo

Usuario

Contraseña

👁

Recordarme [Recuperar contraseña](#)

Iniciar sesión

o regístrate

Registrarse

Figura 19. Prototipo inicio de sesión.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá el inicio de sesión dentro de la aplicación.

- *Página de recuperar contraseña*



Figura 20. Prototipo recuperar contraseña.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá la página de recuperar contraseña dentro de la aplicación.

- *Página de registrar usuario*

El prototipo muestra una interfaz de usuario para el registro. En la parte superior, un encabezado negro contiene un símbolo de retroceso y el título "Registarse".

Debajo del encabezado, se encuentran los siguientes elementos:

- Foto de perfil:** Un botón redondeado negro con el texto "Seleccionar imagen".
- Nombre:** Un campo de entrada con el placeholder "Ingrese su nombre".
- Apellido:** Un campo de entrada con el placeholder "Ingrese su apellido".
- Cedula:** Un campo de entrada con el placeholder "Ingrese su cedula".
- Correo:** Un campo de entrada con el placeholder "Ingrese su correo".
- Usuario:** Un campo de entrada con el placeholder "Ingrese su usuario".
- Contraseña:** Un campo de entrada con el placeholder "Ingrese su contraseña" y un ícono de ojo para alternar la visibilidad.
- Confirmar contraseña:** Un campo de entrada con el placeholder "Confirme su contraseña" y un ícono de ojo para alternar la visibilidad.

En la parte inferior del formulario, hay un botón redondeado negro "Registarse". Debajo de este botón, se encuentra un enlace de texto "o inicia sesión" y un botón redondeado negro "Iniciar sesión".

Figura 21. Prototipo registrar usuario.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá el inicio de sesión dentro de la aplicación.

- *Página de perfil de usuario*



Figura 22. Prototipo perfil de usuario.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá la página de perfil de usuario dentro de la aplicación.

- *Página principal (Feed)*

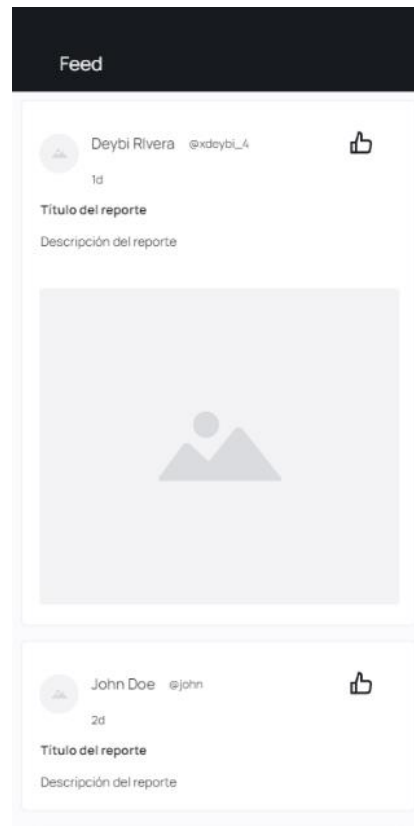


Figura 23. Prototipo página principal.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá la página principal, llamada Feed, dentro de la aplicación.

- *Página de mis reportes*

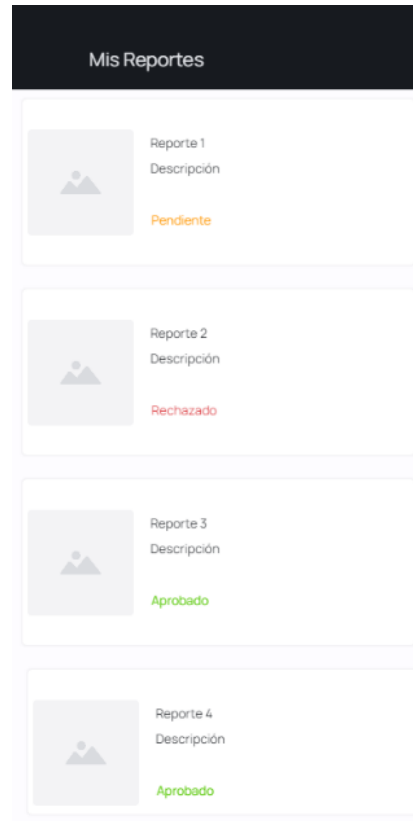


Figura 24. Prototipo mis reportes.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá la página de reportes de usuario, llamada Mis Reportes, dentro de la aplicación.

- *Página de aprobaciones (administrador)*



Figura 25. Prototipo aprobaciones.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá la página de aprobaciones de reportes dentro de la aplicación.

- *Página de información*

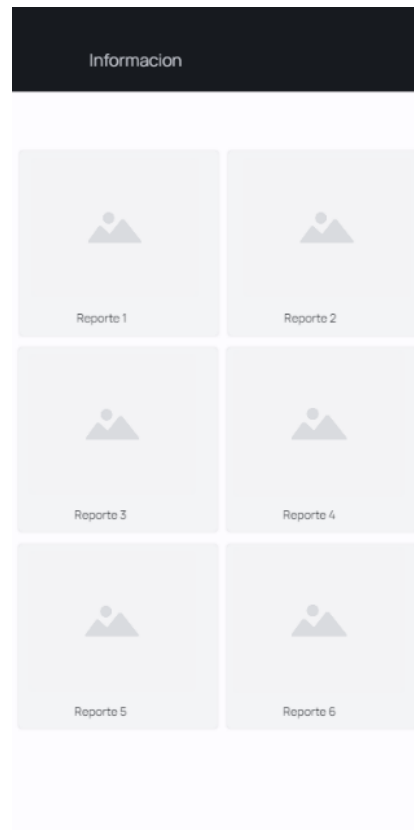


Figura 26. Prototipo información.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá la página de información general de reportes dentro de la aplicación.

- *Página de detalles del reporte*



Figura 27. Prototipo detalles del reporte.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá la página de detalles del reporte dentro de la aplicación.

- *Página de generar reporte*



Formulario de Reporte

Tomar una foto

Categoría

Seleccionar categoría ↓

Título

Escribe un título

Descripción

Escribe una descripción

Ubicación

MAPA

El prototipo muestra una interfaz de usuario para generar un reporte. En la parte superior, hay un encabezado con un botón de retroceso y el título "Formulario de Reporte". Debajo, se encuentran tres botones para "Tomar una foto". A continuación, hay un menú desplegable para "Categoría" con el texto "Seleccionar categoría" y un ícono de flecha hacia abajo. Luego, hay un campo de texto para "Título" con el placeholder "Escribe un título". Después, hay un campo de texto para "Descripción" con el placeholder "Escribe una descripción". Finalmente, hay un campo para "Ubicación" que muestra un ícono de un mapa con el texto "MAPA" y un ícono de una flecha en la esquina inferior derecha.

Figura 28. Prototipo generar reporte.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá la página de generar reporte dentro de la aplicación.

- *Página de notificaciones*



Figura 29. Prototipo notificaciones.

Se muestra el prototipo inicial que describe la idea inicial de cómo se verá la página de notificaciones dentro de la aplicación.

3.2.3 Fase 3: Producción

En esta fase definida por la metodología, se desarrolló la aplicación alineándose a las iteraciones definidas.

a. *Backend*

Firestore permite la implementación de sus servicios, y la comunicación del servidor a través del cliente, es decir, directamente desde el frontend, en este caso, Flutter, y así integrar de forma sencilla los datos de las solicitudes realizadas, con los widgets del framework.

- ***Modelos***
 - comment.dart
 - itemFeed.dart
 - itemMenu.dart
 - post.dart
 - user.dart

- ***Clases***
 - feed_page.dart
 - forgot_password_page.dart
 - generate_report_page.dart
 - information_page.dart
 - login_page.dart
 - menu_page.dart
 - my_reports_page.dart
 - notifications_page.dart
 - pending_post_page.dart
 - post_details_page.dart
 - register_page.dart
 - report_floating_button.dart
 - user_profile_page.dart

- ***Servicios***
 - api_feed.dart
 - api_post.dart
 - api_user.dart

- *Dependencias*



```
! pubspec.yaml M x
! pubspec.yaml
30 dependencies:
31   flutter:
32     sdk: flutter
33   http: ^1.1.0
34   image_picker: ^1.0.4
35   flutter_spinkit: ^5.2.0
36   google_maps_flutter: ^2.5.0
37   cloud_firestore: ^4.12.2
38   permission_handler: ^11.0.1
39   geolocator: ^10.1.0
40   geocoding: ^2.1.1
41   shared_preferences: ^2.2.2
42   path_provider: ^2.1.1
43   intl: ^0.18.1
44   firebase_storage: ^11.4.1
45   image: ^4.1.3
46   location: ^5.0.3
47   flutter_slidable: ^3.0.1
48   photo_view: ^0.14.0
49   flutter_launcher_icons: ^0.13.1
50   cupertino_icons: ^1.0.2
51   firebase_core: ^2.21.0
52   bcrypt: ^1.1.3
53   firebase_auth: ^4.15.0
54   carousel_slider: ^4.2.1
55   shimmer: ^3.0.0
56
```

Figura 30. Dependencias del proyecto.

En la Figura 30 se detallan las dependencias usadas dentro del desarrollo de la aplicación móvil.

b. Storycards

Las Storycards son herramientas descriptivas usadas en la gestión de proyectos de software para representar y controlar las funciones o requisitos. Su objetivo principal es seguir las tareas durante el desarrollo de código para cumplir con los requisitos de las historias de usuario. Estas tarjetas se emplean para establecer prioridades, estimar

tiempos, describir la complejidad y esfuerzo de cada actividad, además de incluir comentarios relevantes de las historias.

- *Iteración 2*

- **Storycard 1**

Tabla 13. Storycard inicio de sesión, cerrar sesión, registrar usuario.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
1	Nuevo	3	3	3	3	Alta
Descripción: Inicio de sesión, cerrar sesión y registrar usuario						
Fecha		Acción		Comentario		
15/10/2023		Definido		Sin comentario		
25/10/2023		Implementado		Sin comentario		
26/10/2023		Ejecutado		Sin comentario		
27/10/2023		Verificado		Sin comentario		

Con Firebase Authentication se procedió a implementar toda la lógica necesaria en la cual se verifica que el correo ni el nombre de usuario estén ya registrados, además de la cédula que permite que la información del usuario sea fidedigna y así que las cuentas no sean creadas con información falsa, las validaciones de los campos se realizan previamente para que los datos sean ingresados correctamente en la base de datos.

```

Future<bool> registerUser() async {
  final ApiUser apiUser = ApiUser(); // Crea una instancia de ApiUser

  String imageUrl = '';
  if (image != null) {
    String filePath = await saveImageLocally(image!);
    imageUrl = await uploadImageToFirebaseStorage(filePath);
  }
  newUser.imageUrl = imageUrl;
  await apiUser.addUser(newUser);

  userAuth.createUserWithEmailAndPassword(
    email: newUser.email, password: password);

  SharedPreferences prefs;
  try {
    prefs = await SharedPreferences.getInstance();
    prefs.setString('username', newUser.username);
  } catch (e) {}
}

return true;
}

Future<String> saveImageLocally(File image) async {
  final directory = await getApplicationDocumentsDirectory();
  fileName = '${newUser.username}-profilePic';
  final filePath = '${directory.path}/${newUser.username}-profilePic.png';
  print('Ruta del archivo: $filePath');
  final File file = File(image.path);
  await file.copy(filePath);
  return filePath;
}

```

Figura 31. Método registrar usuario.

Para el inicio de sesión, también se utilizó Firebase Authentication, donde primero se valida que el usuario exista, si el usuario existe, se procede a verificar la contraseña, una vez todo esté correcto, se procede a iniciar sesión.

```

Future<Map<String, dynamic>>> loginUser(
  BuildContext context, String username) async {
  final ApiUser apiUser = ApiUser();

  final user = await apiUser.getUserByUsername(username);
  if (user != null) {
    name = user['nombre'];

    try {
      UserCredential userCredential =
        await authUser.signInWithEmailAndPassword(
          email: user['email'], password: passwordController.text);
      if (userCredential != null) {
        return user;
      } else {
        return null;
      }
    } catch (e) {
      showErrorSnackBar(context, "Contraseña incorrecta.");
      return null;
    }
  } else {
    showErrorSnackBar(context, "Usuario no encontrado.");
    return null;
  }
}

```

Figura 32. Método iniciar sesión.

Para cerrar sesión, se borra toda la información del usuario de la base de datos local y se redirige a la pantalla de inicio de sesión.

```

Future<void> signOut() async {
  SharedPreferences prefs = await SharedPreferences.getInstance();
  await prefs.remove('username');
}

```

- *Iteración 3*
 - **Storycard 2**

Tabla 14. Storycard recuperar contraseña.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
2	Nuevo	3	3	3	3	Alta
Descripción: Recuperar contraseña						
Fecha		Acción		Comentario		
28/10/2023		Definido		Sin comentario		
30/10/2023		Implementado		Sin comentario		
31/10/2023		Ejecutado		Sin comentario		
31/10/2023		Verificado		Sin comentario		

Para la funcionalidad de recuperar contraseña, se utilizó Firebase Authentication, el cual provee un método para la recuperación de la contraseña de la cuenta mediante correo electrónico.

```

void _sendResetEmail() async {
  if (_formKey.currentState!.validate()) {
    String email = _emailController.text;

    try {
      await FirebaseAuth.instance.sendPasswordResetEmail(email: email);
      _showSuccessDialog();
    } catch (e) {
      _showErrorDialog("Error al enviar el correo de recuperación.");
    }
  }
}

```

Figura 33. Método recuperar contraseña.

- *Iteración 4*
 - **Storycard 3**

Tabla 15. Storycard perfil de usuario.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
3	Nuevo	3	3	3	3	Alta
Descripción: Módulo de perfil de usuario						
Fecha		Acción		Comentario		
01/11/2023		Definido		Sin comentario		
03/11/2023		Implementado		Sin comentario		
04/11/2023		Ejecutado		Sin comentario		
04/11/2023		Verificado		Sin comentario		

Para poder crear el módulo de perfil de usuario, se recupera la información del usuario de Firestore Database y esta se plasma en la interfaz de usuario.

```

Future<Map<String, dynamic>> getUserByUsername(String username) async {
  try {
    final QuerySnapshot usersSnapshot =
      await usersCollection.where('username', isEqualTo: username).get();

    if (usersSnapshot.docs.isNotEmpty) {
      final userData =
        usersSnapshot.docs.first.data() as Map<String, dynamic>;
      return userData;
    } else {
      return null;
    }
  } catch (e) {
    return null;
  }
}

```

Figura 34. Método recuperar usuario.

- *Iteración 5*
- **Storycard 4**

Tabla 16. Storycard editar perfil y contraseña.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
4	Nuevo	4	4	4	4	Alta
Descripción: Editar perfil y contraseña de usuario						
Fecha		Acción		Comentario		
05/11/2023		Definido		Sin comentario		
08/11/2023		Implementado		Sin comentario		
09/11/2023		Ejecutado		Sin comentario		
09/11/2023		Verificado		Sin comentario		

Para editar el perfil, se hacen validaciones de que el usuario ha modificado la información, y se permite cambiar el nombre, apellido y foto de perfil del usuario, accediendo al documento perteneciente al usuario y actualizando la información de este en la base de datos.

```

Future<void> saveEditingChanges(userData) async {
  if (userData['nombre'] != nameController.text ||
      userData['apellido'] != lastNameController.text ||
      _image != null) {
    final userQuery = await FirebaseFirestore.instance
      .collection('users')
      .where('username', isEqualTo: userData['username'])
      .get();
    var userDoc = userQuery.docs.first;
    if (_image == null) {
      userDoc.reference.update({
        'apellido': lastNameController.text,
        'nombre': nameController.text
      });
    } else {
      // Subir foto de perfil
      if (_image != null) {
        String filePath = await saveImageLocally(_image!);
        imageUrl = await uploadImageToFirebaseStorage(filePath);

        userDoc.reference.update({
          'apellido': lastNameController.text,
          'nombre': nameController.text,
          'imageUrl': imageUrl,
        });
      }
    }
  }
}

```

Figura 35. Método editar perfil de usuario.

Para cambiar la contraseña de la cuenta del usuario, se utilizar un método proporcionado por Firebase Authentication, previamente se hacen todas las validaciones correspondientes.

```

//Actualizar contraseña
autUser.currentUser?.updatePassword(passwordController.text);

ScaffoldMessenger.of(context).showSnackBar(
  const SnackBar(
    content: Text('Ha cambiado su contraseña con éxito.'),
    backgroundColor: Colors.green,
  ), // SnackBar
);

```

Figura 36. Método cambiar contraseña.

- *Iteración 6*
- **Storycard 5**

Tabla 17. Storycard generar reportes.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
5	Nuevo	4	4	4	4	Alta
Descripción: Generar reportes						
Fecha		Acción		Comentario		
10/11/2023		Definido		Sin comentario		
14/11/2023		Implementado		Sin comentario		
15/11/2023		Ejecutado		Sin comentario		
15/11/2023		Verificado		Sin comentario		

Para generar un nuevo reporte, se muestra un formulario en el que el usuario debe agregar imágenes relacionadas al reporte, un título y una descripción, la ubicación se obtiene automáticamente mediante el GPS del dispositivo móvil, se valida la información y se ingresa en la base de datos de Firestore Database.


```

Future<bool> postInformation() async {
  List<String> filePath = await saveImageLocally(images);
  imageUrl = await uploadImageToFirebaseStorage(filePath);
  final ApiPost apiPost = ApiPost(); // Crea una instancia de ApiPost

  Post finalPost = Post(
    category: newPost.category,
    coordinates: newPost.coordinates,
    dateTime: newPost.dateTime,
    description: newPost.description,
    imageUrl: imageUrl!,
    location: newPost.location,
    reactions: newPost.reactions,
    status: newPost.status,
    title: newPost.title,
    username: newPost.username);

  final postId = await apiPost.addPost(finalPost);

  try {
    // Obtener la referencia del contador del usuario
    DocumentReference userCountRef =
      FirebaseFirestore.instance.collection('users').doc(username);

    // Incrementar el contador
    await userCountRef.set(
      {'countPosts': FieldValue.increment(1)}, SetOptions(merge: true));
  } catch (e) {
    print('Error al actualizar el contador: $e');
  }

  return true;
}

```

Figura 37. Método generar reporte.

- *Iteración 7*
- **Storycard 6**

Tabla 18. Storycard visualizar reportes del usuario.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
6	Nuevo	3	3	3	3	Alta
Descripción: Visualizar reportes del usuario						
Fecha		Acción		Comentario		
16/11/2023		Definido		Sin comentario		
18/11/2023		Implementado		Sin comentario		
19/11/2023		Ejecutado		Sin comentario		
19/11/2023		Verificado		Sin comentario		

Para poder visualizar los reportes hechos por el usuario, se realiza una petición a la base de datos de Firestore Database de los reportes por usuario y se los ordena de forma descendente por fecha.

```
Stream<List<ItemMenu>> loadPosts() {
  ApiUser apiUser = ApiUser();
  var controller = StreamController<List<ItemMenu>>();
  FirebaseFirestore.instance
    .collection('posts')
    .where('username', isEqualTo: data?['username'])
    .orderBy('dateTime', descending: true)
    .snapshots()
    .listen((querySnapshot) async {
      final realItems =
        await Future.wait(querySnapshot.docs.map((document) async {
          final data = document.data() as Map<String, dynamic>;
          int reactions = data['reactions'] ?? 0;
          final isLiked = data['likes'] != null &&
            data['likes'][this.data?['username']] != null;
          var userData = await apiUser.getUserByUsername(data['username']);

          return ItemMenu(
            userData: userData!,
            title: data['title'],
            description: data['description'],
            username: data['username'],
            dateTime: data['dateTime'],
            location: data['location'],
            coordinates: data['coordinates'],
            imageUrl: (data['imageUrl'] as List).cast<String>(),
            reactions: reactions,
            postId: document.id,
            isLiked: isLiked,
            status: data['status'],
            category: data['category'],
          );
        })).toList();

      controller.add(realItems);
    });
  return controller.stream;
}
```

Figura 38. Método para recuperar los reportes del usuario.

- *Iteración 8*
- **Storycard 7**

Tabla 19. Storycard editar y borrar reportes.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
7	Nuevo	4	4	4	4	Alta
Descripción: Editar y borrar los reportes del usuario						
Fecha		Acción		Comentario		
20/11/2023		Definido		Sin comentario		
24/11/2023		Implementado		Sin comentario		
25/11/2023		Ejecutado		Sin comentario		
25/11/2023		Verificado		Sin comentario		

Para poder editar un reporte, primero se verifica que el reporte todavía no ha sido aprobado por un administrador, entonces se procede a validar de que el formulario ha sido modificado para que se pueda actualizar la información, se permite actualizar categoría, título y descripción del reporte. Una vez hechas las respectivas validaciones se accede al documento del reporte y se actualiza la información de este.

```

Future<void> saveEditingChanges(ItemMenu item) async {
  if (item.title != _titleController.text ||
      item.description != _descriptionController.text ||
      item.category != _dropDownValueCategory) {
    final postsCollection = FirebaseFirestore.instance.collection('posts');
    final postDoc = postsCollection.doc(item.postId);

    postDoc.update({
      'title': _titleController.text,
      'description': _descriptionController.text,
      'category': _dropDownValueCategory
    });
  }
}

```

Figura 39. Método editar reporte.

Para borrar un reporte, al usuario se le pide una confirmación y se procede a eliminar documento perteneciente al reporte y toda la información relacionada al mismo en la base de datos Firestore Database.

```
Future<void> deletePost(item, index) async {
  bool? result = await showDeleteDialog();
  if (result!) {
    try {
      await FirebaseFirestore.instance
        .collection('posts')
        .doc(item.postId)
        .delete();

      QuerySnapshot commentsSnapshot = await FirebaseFirestore.instance
        .collection('posts')
        .doc(item.postId)
        .collection('comments')
        .get();

      for (QueryDocumentSnapshot commentSnapshot in commentsSnapshot.docs) {
        await commentSnapshot.reference.delete();
      }

      QuerySnapshot querySnapshot = await FirebaseFirestore.instance
        .collection('notifications')
        .where('postId', isEqualTo: item.postId)
        .get();

      if (querySnapshot.docs.isNotEmpty) {
        await querySnapshot.docs.first.reference.delete();
      } else {}
    } catch (e) {}
    try {
      DocumentReference userCountRef = FirebaseFirestore.instance
        .collection('users')
        .doc(data!['username']);

      await userCountRef.set({'countPosts': FieldValue.increment(-1)},
        SetOptions(merge: true));
    } catch (e) {}
  } catch (error) {}
  ScaffoldMessenger.of(context).showSnackBar(
    const SnackBar(
      content: Text('Reporte eliminado'),
    ), // SnackBar
  );
}
```

Figura 40. Método eliminar un reporte.

- *Iteración 9*
- **Storycard 8**

Tabla 20. Storycard visualizar reportes de otros usuarios.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
8	Nuevo	4	4	4	4	Alta
Descripción: Visualizar reportes de otros usuarios						
Fecha		Acción		Comentario		
26/11/2023		Definido		Sin comentario		
28/11/2023		Implementado		Sin comentario		
29/11/2023		Ejecutado		Sin comentario		
29/11/2023		Verificado		Sin comentario		

Para poder visualizar los reportes de otros usuarios, se hace una petición a la base de datos de Firestore Database para recuperar los reportes que han sido aprobados, y estos se los ordenan de forma descendente por fecha.

```

Future<void> loadPosts() async {
  ApiUser apiUser = ApiUser();
  final postsCollection = FirebaseFirestore.instance
    .collection('posts')
    .limit(postsNumber)
    .where('status', isEqualTo: 1)
    .orderBy('dateTime', descending: true);
  final querySnapshot = await postsCollection.get();

  final realItems = await _getRealItems(apiUser, querySnapshot);

  setState(() {
    items = realItems;
  });
}

```

Figura 41. Método recuperar reportes de usuarios.

- *Iteración 10*
- **Storycard 9**

Tabla 21. Storycard visualizar detalles del reporte.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
9	Nuevo	4	4	4	4	Alta
Descripción: Visualizar detalles del reporte						
Fecha		Acción		Comentario		
30/11/2023		Definido		Sin comentario		
01/12/2023		Implementado		Sin comentario		
02/12/2023		Ejecutado		Sin comentario		
02/12/2023		Verificado		Sin comentario		

Para poder visualizar los detalles, desde donde se abre el reporte, se envía la información obtenida previamente de este a la siguiente página para poder visualizar los datos de forma detallada.

```

void _onPostItemTap(ItemMenu item) {
  Navigator.push<bool>(
    context,
    MaterialPageRoute(
      builder: (context) => PostDetailsPage(
        post: item,
        username: data?['username'],
        myReportsState: false,
      ), // PostDetailsPage
    ), // MaterialPageRoute
  ).then((result) {
    if (result != null && result) {
      loadPosts();
    }
  });
}

```

Figura 42. Método mostrar detalles del reporte.

- *Iteración 11*
- **Storycard 10**

Tabla 22. Storycard reaccionar y comentar reportes.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
10	Nuevo	4	4	4	4	Alta
Descripción: Reaccionar y comentar reportes.						
Fecha		Acción		Comentario		
03/12/2023		Definido		Sin comentario		
07/12/2023		Implementado		Sin comentario		
08/12/2023		Ejecutado		Sin comentario		
08/12/2023		Verificado		Sin comentario		

Para poder reaccionar a cualquier reporte, el usuario debe presionar un botón, el cual verifica el nombre de usuario para que este pueda reaccionar o quitar la reacción, y así modificar el contador de reacciones del reporte.

```

void toggleLike() {
  final postsCollection = FirebaseFirestore.instance.collection('posts');
  final postDoc = postsCollection.doc(widget.post.postId);

  if (widget.post.isLiked) {
    postDoc.update({
      'reactions': FieldValue.increment(-1),
      'likes.${getUserId()}': FieldValue.delete(),
    });

    setState(() {
      widget.post.isLiked = false;
      widget.post.reactions--;
    });
  } else {
    postDoc.update({
      'reactions': FieldValue.increment(1),
      'likes.${getUserId()}': FieldValue.serverTimestamp(),
    });

    setState(() {
      widget.post.isLiked = true;
      widget.post.reactions++;
    });
  }
}

```

Figura 43. Método reaccionar reporte.

Para poder comentar reportes, se verifica el nombre de usuario, y se ingresa el comentario a la subcolección del documento perteneciente al reporte en la base de datos de Firestore Database.

```

Future<void> uploadComment(String postId, Comment comment) async {
  final postRef = FirebaseFirestore.instance.collection('posts').doc(postId);

  await postRef.collection('comments').add({
    'username': comment.username,
    'text': comment.text,
    'timestamp': comment.timestamp,
  });

  try {
    // Obtener la referencia del contador del usuario
    DocumentReference postCountRef =
      FirebaseFirestore.instance.collection('posts').doc(postId);

    // Incrementar el contador
    await postCountRef.set(
      {'commentCount': FieldValue.increment(1)}, SetOptions(merge: true));
  } catch (e) {
    print('Error al actualizar el contador: $e');
  }
}

```

Figura 44. Método comentar reportes.

- *Iteración 12*
- **Storycard 11**

Tabla 23. Storycard geolocalización y mapas.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
11	Nuevo	4	4	4	4	Alta
Descripción: Geolocalización y mapas						
Fecha		Acción		Comentario		
09/12/2023		Definido		Sin comentario		
10/12/2023		Implementado		Sin comentario		
11/12/2023		Ejecutado		Sin comentario		
11/12/2023		Verificado		Sin comentario		

Antes de que el usuario pueda generar un reporte, primero se accede a permisos de ubicación del dispositivo y se obtiene el nombre de esta con un algoritmo de decodificación.

```

Future<String?> getLocationName() async {
  try {
    // Obtén la ubicación actual
    Position position = await Geolocator.getCurrentPosition(
      desiredAccuracy: LocationAccuracy.high,
    );
    // Obtiene el nombre de la ubicación a partir de las coordenadas
    List<Placemark> placemarks = await placemarkFromCoordinates(
      position.latitude,
      position.longitude,
    );

    if (placemarks.isNotEmpty) {
      Placemark? place = placemarks[0];
      String? finalName = '${place.street!}';
      setState(() {
        coor = [position.latitude, position.longitude];
        locationName = finalName;
        _center = LatLng(coor![0], coor![1]);
      });
      return finalName; // Devuelve el nombre de la ubicación
    } else {
      return "Ubicación desconocida";
    }
  } catch (e) {
    return "Error al obtener la ubicación";
  }
}

```

Figura 45. Método obtener ubicación del dispositivo.

- **Iteración 13**
- **Storycard 12**

Tabla 24. Storycard aprobación de reportes.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
12	Nuevo	4	4	4	4	Alta
Descripción: Aprobación de reportes						
Fecha		Acción		Comentario		
12/12/2023		Definido		Sin comentario		
16/12/2023		Implementado		Sin comentario		
17/12/2023		Ejecutado		Sin comentario		
17/12/2023		Verificado		Sin comentario		

Para poder aprobar o rechazar un reporte realizado, se verifica que el usuario que está yendo a realizar la acción, sea un administrador, se pide una verificación de si se quiere aprobar o rechazar, y si se rechaza, se debe especificar un motivo, y se procede a actualizar el estado del documento perteneciente al reporte en la base de datos de Firestore Database.

```
void changePostStatus(ItemMenu item, status, title, message) async {
  int? res = await _showDialog(
    context,
    title,
    message,
  );

  if (res == 1) {
    final postsCollection = FirebaseFirestore.instance.collection('posts');
    final postDoc = postsCollection.doc(item.postId);

    final CollectionReference notificationsCollection =
      FirebaseFirestore.instance.collection('notifications');

    if (status == 2) {
      postDoc.update({'status': status, 'denyReason': _dropDownValueReason});
      notificationsCollection.add({
        'description': 'Ha sido rechazado.',
        'status': 'Rechazado',
        'denyReason': 'Motivo: $_dropDownValueReason',
        'timestamp': DateTime.now(),
        'title': post.title,
        'username': post.username,
        'postId': post.postId,
        'read': 'unread'
      });
    } else {
      postDoc.update({'status': status});

      notificationsCollection.add({
        'description': 'Ha sido aprobado.',
        'status': 'Aprobado',
        'timestamp': DateTime.now(),
        'title': post.title,
        'username': post.username,
        'postId': post.postId,
        'read': 'unread'
      });
    }

    Navigator.pop(context, true);
  }
}
```

Figura 46. Método aprobar/rechazar reporte.

- *Iteración 14*
- **Storycard 13**

Tabla 25. Storycard registrar administradores.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
13	Nuevo	3	3	3	3	Alta
Descripción: Registrar administradores						
Fecha		Acción		Comentario		
18/12/2023		Definido		Sin comentario		
19/12/2023		Implementado		Sin comentario		
20/12/2023		Ejecutado		Sin comentario		
20/12/2023		Verificado		Sin comentario		

Para el módulo de registrar administradores, se usa el mismo módulo de registrar usuarios, pero antes de eso se verifica si es que el usuario ha iniciado sesión y que este sea de tipo administrador, y se procede con la inserción de los datos en la base de datos de Firestore Database.

```

// Crear un usuario
final newUser = User(
  imageUrl: '',
  nombre: nameController.text,
  apellido: lastnameController.text,
  cedula: cedulaController.text,
  email: emailController.text,
  password: await hashPassword(passwordController.text),
  rol: widget.isAdmin ? 'admin' : 'user',
  username: usernameController.text.toLowerCase(),
  nroInicios: 0);

if (widget.isAdmin) {
  if (_image != null) {
    String filePath = await saveImageLocally(_image!);
    imageUrl = await uploadImageToFirebaseStorage(filePath);
  }

  newUser.imageUrl = imageUrl!;

  await apiUser.addUser(newUser);

  userAuth.createUserWithEmailAndPassword(
    email: newUser.email, password: passwordController.text);

  setState(() {
    isRegister = true;
  });

  successDialog();
}

```

Figura 47. Método registrar administrador.

- *Iteración 15*
- **Storycard 14**

Tabla 26. Storycard visualizar notificaciones.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
14	Nuevo	4	4	4	4	Alta
Descripción: Visualizar notificaciones sobre reportes						
Fecha		Acción		Comentario		
21/12/2023		Definido		Sin comentario		
23/12/2023		Implementado		Sin comentario		
24/12/2023		Ejecutado		Sin comentario		
24/12/2023		Verificado		Sin comentario		

Para visualizar las notificaciones sobre los reportes de cada usuario, primero se recupera los documentos de la base de datos de Firestore Database pertenecientes a los reportes de cada usuario que se crean a partir de la aprobación o rechazo de un administrador, estas notificaciones se ordenan en forma descendente a partir de la fecha y hora.

```

stream: FirebaseFirestore.instance
  .collection('notifications')
  .where('username', isEqualTo: widget.username)
  .limit(numberNotifications)
  .orderBy('timestamp', descending: true)
  .snapshots(),
builder: (context, AsyncSnapshot<QuerySnapshot> snapshot) {
  if (snapshot.hasError) {
    return Center(child: Text('Error al cargar las notificaciones'));
  }

  if (snapshot.connectionState == ConnectionState.waiting) {
    return Center(child: CircularProgressIndicator());
  }

  var notifications = snapshot.data!.docs;

  if (notifications.isEmpty) {
    return Center(child: Text('No tienes notificaciones recientes.'));
  }
}

```

Figura 48. Método recuperar notificaciones.

- *Iteración 16*
- **Storycard 15**

Tabla 27. Storycard información general.

N°	Tipo	Dificultad		Esfuerzo		Prioridad
		Antes	Después	Antes	Después	
15	Nuevo	4	4	4	4	Alta
Descripción: Módulo de información general sobre reportes						
Fecha		Acción		Comentario		
26/12/2023		Definido		Sin comentario		
29/12/2023		Implementado		Sin comentario		
30/12/2023		Ejecutado		Sin comentario		
30/12/2023		Verificado		Sin comentario		

Para el módulo de información general sobre reportes, se hacen varias peticiones en la base de datos de Firestore Database dependiendo de la sección que se seleccione, una vez recuperada la información, esta se muestra de una forma intuitiva para el usuario final.

```

Future<List<Map<String, dynamic>>> _getMostCommentedPosts() async {
  try {
    CollectionReference postsCollection =
      FirebaseFirestore.instance.collection('posts');
    QuerySnapshot postsSnapshot = await postsCollection
      .orderBy('commentCount', descending: true)
      .limit(10)
      .get();

    List<Map<String, dynamic>> postDetails = [];
    ApiUser apiUser = ApiUser();

    for (QueryDocumentSnapshot postDoc in postsSnapshot.docs) {
      Map<String, dynamic> postData = postDoc.data() as Map<String, dynamic>;

      var userData = await apiUser.getUserByUsername(postData['username']);
      if (userData != null) {
        postDetails.add({
          'postId': postDoc.id,
          'post': postData,
          'user': userData,
          'commentCount': postData['commentCount']
        });
      }
    }
    return postDetails;
  } catch (e) {
    print("Error al obtener los posts más comentados: $e");
    return [];
  }
}

```

Figura 49. Método obtener reportes más comentados.

```

Future<List<ItemMenu>> loadPosts() async {
  ApiUser apiUser = ApiUser();

  final postsCollection = FirebaseFirestore.instance
    .collection('posts')
    .limit(10)
    .where('status', isEqualTo: 1).where('reactions', isGreaterThan: 0)
    .orderBy('reactions', descending: true);
  final querySnapshot = await postsCollection.get();

  final realItems = querySnapshot.docs.map((document) async {
    final data = document.data() as Map<String, dynamic>;

    final isLiked = data['likes'] != null &&
      data['likes'][this.data?['username']] != null;

    var userData = await apiUser.getUserByUsername(data['username']);

    return ItemMenu(
      userData: userData!,
      title: data['title'],
      description: data['description'],
      username: data['username'],
      dateTime: data['dateTime'],
      location: data['location'],
      coordinates: data['coordinates'],
      imageUrl: (data['imageUrl'] as List).cast<String>(),
      reactions: data['reactions'],
      postId: document.id,
      isLiked: isLiked,
      status: data['status'],
      category: data['category']);
  }).toList();
  final List<ItemMenu> futures = await Future.wait(realItems);

  items = futures;

  return items;
}

```

Figura 50. Método obtener reportes más reaccionados.


```

Future<List<ItemMenu>> loadPosts() async {
  ApiUser apiUser = ApiUser();

  final postsCollection = FirebaseFirestore.instance
    .collection('posts')
    .where('category', isEqualTo: category)
    .where('status', isEqualTo: 1)
    .orderBy('dateTime', descending: true);
  final querySnapshot = await postsCollection.get();

  final realItems = querySnapshot.docs.map((document) async {
    final data = document.data() as Map<String, dynamic>;

    final isLiked = data['likes'] != null &&
      data['likes'][this.data['username']] != null;

    var userData = await apiUser.getUserByUsername(data['username']);

    return ItemMenu(
      userData: userData!,
      title: data['title'],
      description: data['description'],
      username: data['username'],
      dateTime: data['dateTime'],
      location: data['location'],
      coordinates: data['coordinates'],
      imageUrl: (data['imageUrl'] as List).cast<String>(),
      reactions: data['reactions'],
      postId: document.id,
      isLiked: isLiked,
      status: data['status'],
      category: data['category']);
  }).toList();
  final List<ItemMenu> futures = await Future.wait(realItems);

  items = futures;
  return items;
}

```

Figura 51. Método obtener reportes por categoría.

```

void _onMarkerTapped(
  LatLng position, BuildContext context, String locationName) async {
  try {
    CollectionReference postsCollection =
      FirebaseFirestore.instance.collection('posts');

    QuerySnapshot postsSnapshot = await postsCollection
      .where(
        'coordinates',
        isEqualTo: {
          'latitude': position.latitude,
          'longitude': position.longitude,
        },
      )
      .where('status', isEqualTo: 1)
      .get();

    List<String> postIds = postsSnapshot.docs.map((doc) => doc.id).toList();

    Navigator.push(
      context,
      MaterialPageRoute(
        builder: (context) => SectorsPage(
          postIds: postIds,
          locationName: locationName,
          username: username,
        ), // SectorsPage
      ), // MaterialPageRoute
    );
  } catch (e) {}
}

```

Figura 52. Método obtener reportes por sector.

```

Future<List<Map<String, dynamic>>> getTopUsers() async {
  try {
    CollectionReference usersCollection =
      FirebaseFirestore.instance.collection('users');

    QuerySnapshot topUsersSnapshot = await usersCollection.where('countPosts', isGreaterThan: 0)
      .orderBy('countPosts', descending: true)
      .limit(10)
      .get();

    List<Map<String, dynamic>> topUsers = topUsersSnapshot.docs
      .map((doc) => {'id': doc.id, ...doc.data() as Map<String, dynamic>})
      .toList();
    return topUsers;
  } catch (e) {
    print('Error al obtener los usuarios: $e');
    return [];
  }
}

```

Figura 53. Método obtener usuario que más han reportado.

3.2.4 Fase 4: Estabilización

En este punto del desarrollo, la fase indica que se debe realizar ajustes, corrección de errores y mejorar la experiencia de usuario.

a. Cambios en la base de datos

- Se añadió un nuevo campo en la colección “posts” llamado “commentCount”, para contabilizar los comentarios que tiene el reporte y así poder mostrar información general.
- Se añadió un nuevo campo en la colección “users” llamado “countPosts”, para contabilizar la cantidad de reportes realizados por el usuario y así poder mostrar información general.

b. Cambios en el diseño

- Se modificó la página de generar reporte, permitiendo así al usuario añadir más de una imagen a los reportes.
- En el perfil de usuario, se añadió una insignia que indica que el usuario es administrador.

c. Cambios en el código

- Se optimizó el código para optimizar la apertura de las páginas cruciales y que estas sean instantáneas.

3.2.5 Fase 5: Pruebas del sistema

a. *Pruebas de funcionalidad*

- **Página de inicio de sesión y registrar usuario**

En la página de inicio de sesión, el usuario debe ingresar sus credenciales, nombre de usuario y contraseña para realizar las respectivas validaciones e ingresar al sistema.

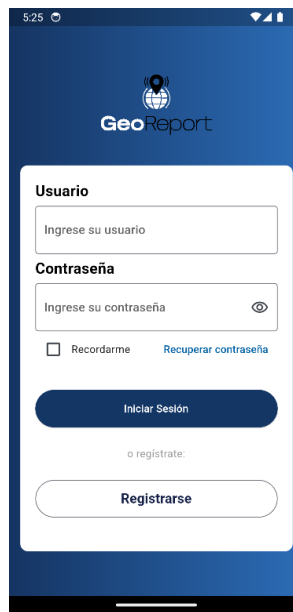


Figura 54. Página inicio de sesión.

Para el registro, el usuario tiene que ingresar información válida, y los campos de correo, cédula y nombre de usuario, no puede repetirse con ningún otro usuario.

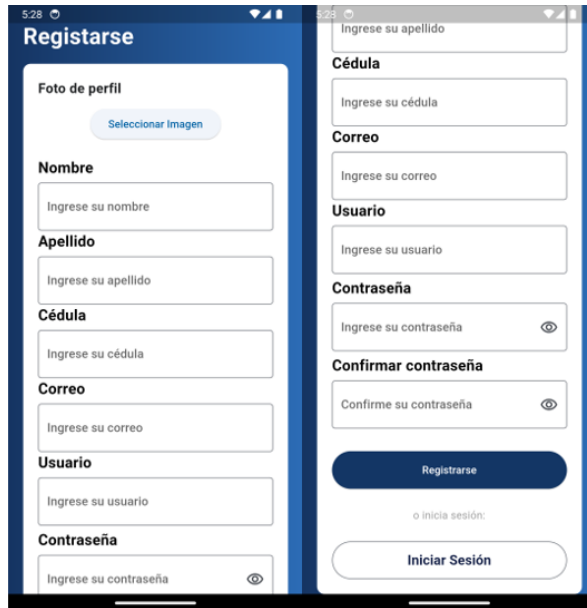


Figura 55. Página registrar usuario.

- **Página de recuperar contraseña**

Para poder obtener un enlace en el correo electrónico, previamente el usuario debe estar registrado en el sistema, y colocar en el formulario su correo electrónico.

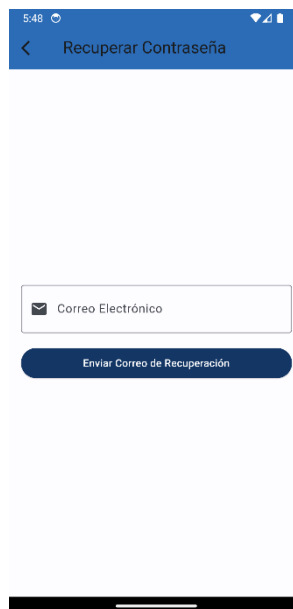


Figura 56. Página recuperar contraseña.

- **Página de perfil de usuario**

En esta página, el usuario puede visualizar su información personal y realizar acciones extras.



Figura 57. Página perfil de usuario.

- **Funcionalidad editar perfil y contraseña**

El usuario puede editar su información personal, como la imagen de perfil, nombre y apellido.



Figura 58. Funcionalidad editar perfil.

Además, el usuario puede editar su contraseña, completando el formulario con información validada.

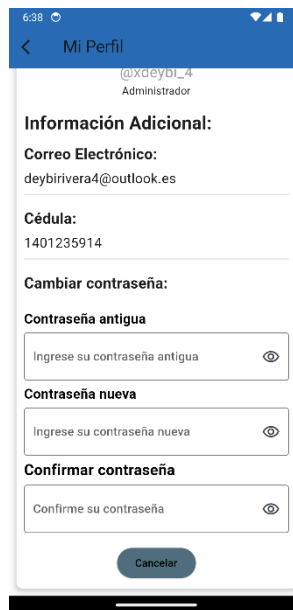


Figura 59. Funcionalidad editar contraseña.

- **Página generar reportes**

Para poder generar un nuevo reporte sobre algún problema de infraestructura, el usuario debe llenar el formulario en el que todos los campos son requeridos, mínimo una imagen, categoría, título y descripción, la ubicación se recopilará automáticamente al entrar a esta página.

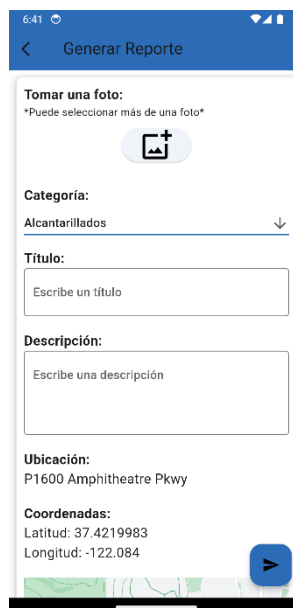


Figura 60. Página generar reporte.

- **Página visualizar reportes de usuario (Mis reportes)**

En esta página, el usuario puede ver sus reportes realizados, así como el estado de estos, como lo son pendiente, aprobado y rechazado, además de poder ingresar al detalle del reporte. Para poder eliminar un reporte, el usuario puede deslizar el cuadro hacia la izquierda y aparecerá la opción.

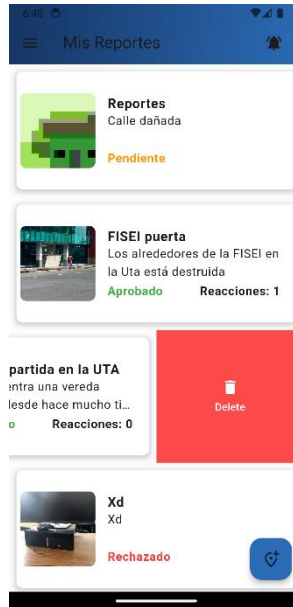


Figura 61. Página visualizar mis reportes.

- **Funcionalidad editar reporte**

Para que el usuario pueda editar la información del reporte, debe ingresar al mismo desde la página de visualizar sus reportes, y debe cumplir con la

condición de que el reporte tenga el estado “Pendiente”, y así proceder con la edición de este, se permite cambiar la categoría, el título y la descripción.

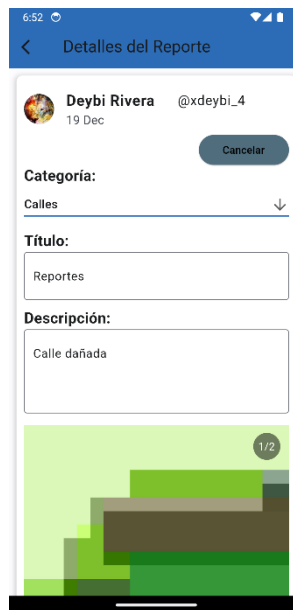


Figura 62. Funcionalidad editar reporte.

- **Página visualizar reportes de otros usuarios (Feed)**

En esta página, el usuario puede visualizar los reportes de otros usuarios que ya han sido aprobados, así como reaccionar a estos y poder entrar a los detalles.



Figura 63. Página visualizar otros reportes.

- **Página visualizar detalles del reporte**

Desde cualquier página, el usuario puede ingresar al detalle de ese reporte con solo dar un toque, en esta página, se puede visualizar la información detallada, como el usuario que ha reportado, la fecha, categoría, título, descripción, imágenes, un mapa que indica la ubicación, dar o quitar una reacción, y una sección para leer y escribir comentarios.

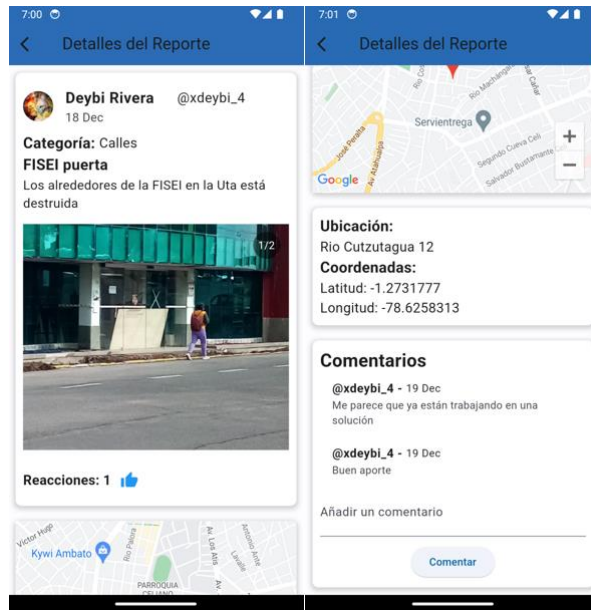


Figura 64. Página detalles del reporte.

- **Funcionalidad geolocalización y mapas**

En el mapa que se encuentra en los detalles del reporte, se puede interactuar, dando un toque en el marcador, se muestra el título y descripción del reporte, además se habilita un botón para poder trazar una ruta desde la ubicación actual hasta el punto del reporte, esto con los mapas de Google.

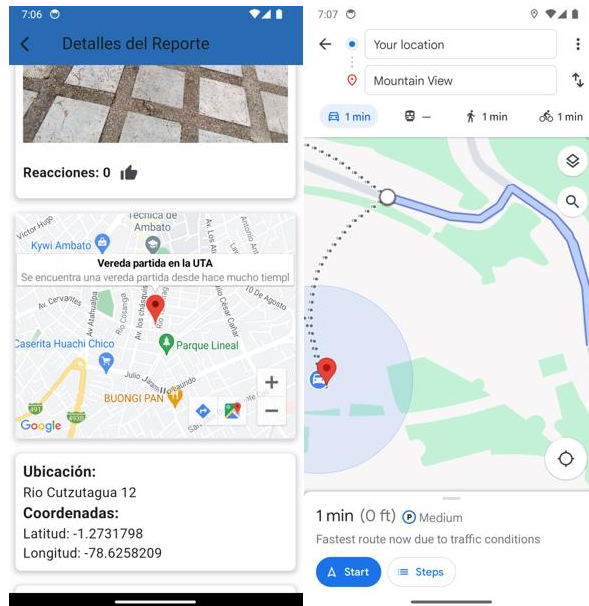


Figura 65. Funcionalidad geolocalización y mapas.

- **Funcionalidad aprobar/rechazar reportes**

Cuando un usuario realice un reporte, a los administradores les llegará en la página de “Aprobaciones” aquellos reportes que estén pendientes a revisión, dentro de los detalles, se puede decidir entre aprobar o rechazar el reporte con

un motivo, y así tener una moderación del contenido dentro de la aplicación móvil.

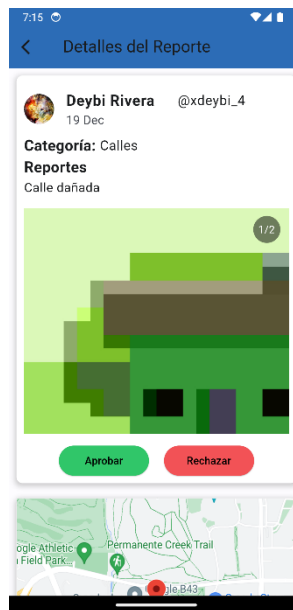


Figura 66. Funcionalidad aprobar/rechazar reporte.

- **Funcionalidad registrar administradores**

Para registrar administradores, se debe iniciar sesión desde un perfil que tenga el rol de administrador ya que solo ellos pueden realizar esta acción, el formulario es igual al de registrar usuario.

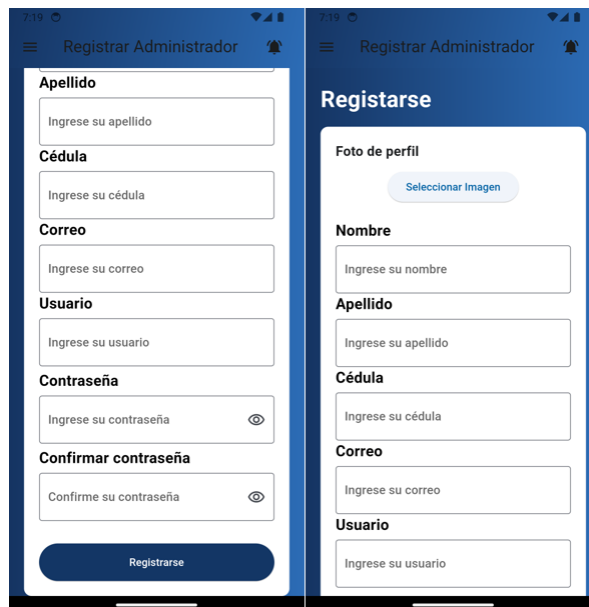


Figura 67. Funcionalidad registrar administradores.

- **Página visualizar notificaciones sobre reportes**

En esta página, el usuario puede ver las notificaciones sobre el estado de sus reportes realizados, ya sean aprobados o rechazados, y ver el motivo del rechazo, además, dando un toque a la notificación, se dirigirá a la página de los detalles del reporte.

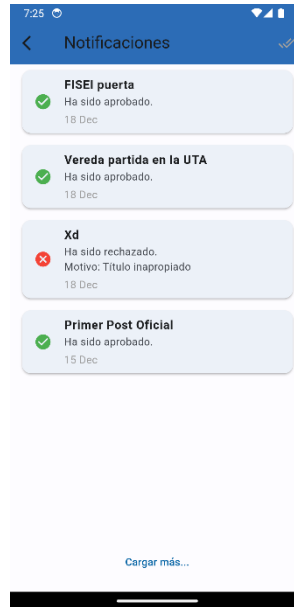


Figura 68. Página notificaciones.

- **Página información general sobre reportes**

En esta página, el usuario puede visualizar información recopilada sobre todos los reportes realizados en la aplicación móvil, separado por diferentes categorías e intereses.



Figura 69. Página información general.

3.2.6 Implantación la aplicación móvil con geolocalización para el reporte ciudadano de daños en la Infraestructura Urbana de la ciudad de Ambato

Al finalizar todas las fases de la metodología Mobile-D, se requirió implantar la aplicación desarrollada.

Con el fin de poder llegar a una amplia cantidad de usuarios, se creó un repositorio en GitHub con información sobre la aplicación móvil, como la política de privacidad, el título, una descripción general, los requisitos y el enlace de descarga del archivo en formato .apk.

Los usuarios pueden acceder al siguiente enlace:
<https://github.com/xDeybi4/GeoReport-Info>

O escanear el código QR desde sus dispositivos móviles:



Figura 70. Código QR del repositorio de la aplicación.

Una vez ingresado al repositorio, se podrá encontrar un enlace que redirige a la descarga del archivo en OneDrive.

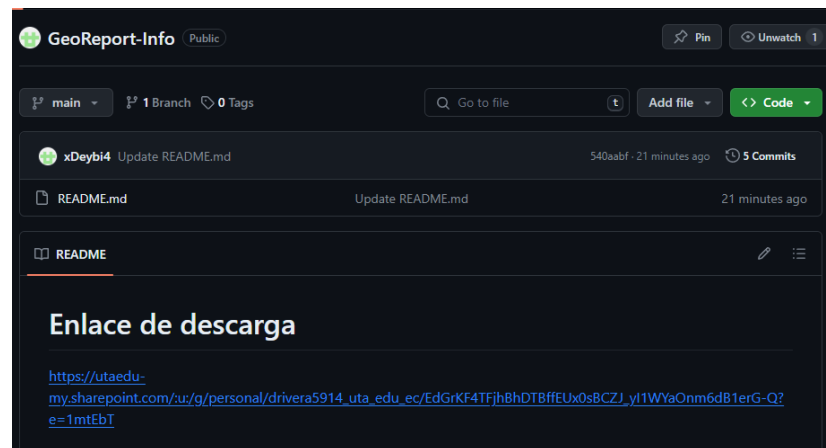


Figura 71. Repositorio en GitHub.

A continuación, se debe presionar el botón “Descargar” para comenzar la descarga.

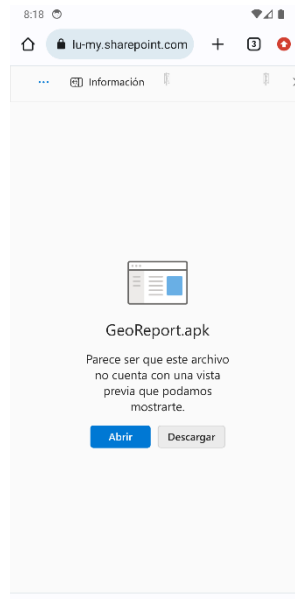


Figura 72. Archivo en OneDrive.

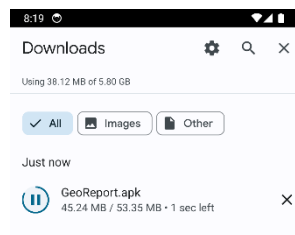


Figura 73. Descarga del archivo de instalación.

Al no ser un archivo proveniente de la tienda oficial de aplicaciones, se debe conceder permisos para instalar desde orígenes desconocidos.

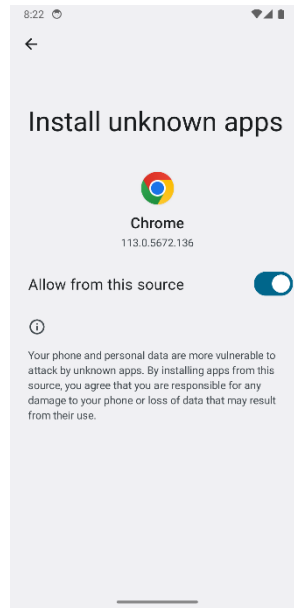


Figura 74. Habilitar permisos de origen desconocido.

Al conceder este permiso, aparecerá este cuadro de diálogo que pedirá la confirmación de la instalación.

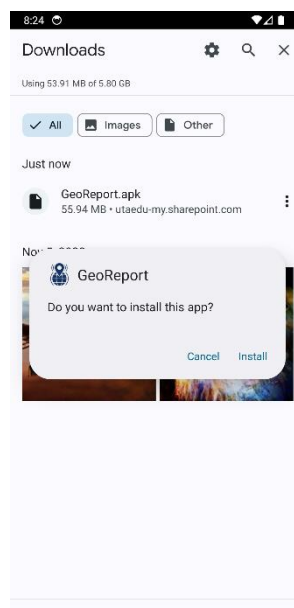


Figura 75. Confirmar instalación.

Finalmente, la aplicación se instalará y se ubicará en la biblioteca de aplicaciones del dispositivo lista para su uso.

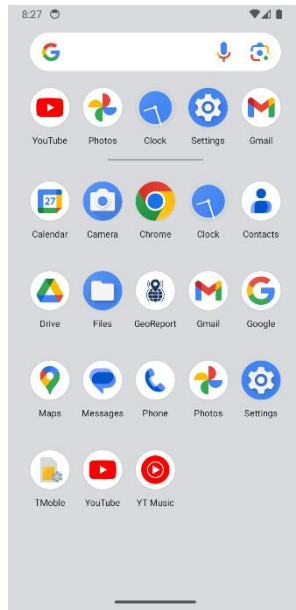


Figura 76. Aplicación instalada en el dispositivo móvil.

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- En conclusión, a través del proceso de recolección de información, el proyecto destaca la importancia de la participación ciudadana en la gestión de la infraestructura. Al permitir que los ciudadanos reporten los daños, se puede garantizar que los problemas se identifiquen y solucionen de manera más rápida y eficiente.
- En resumen, Flutter, como framework de desarrollo de aplicaciones móviles utilizado en este proyecto, permitió crear una plataforma eficiente y efectiva para el reporte ciudadano de daños en la infraestructura urbana. Esto se logró gracias a una amplia gama de herramientas y recursos, así como a una curva de aprendizaje baja, cumpliendo con los requisitos del proyecto.
- Dentro del análisis, la metodología Mobile-D, permitió en el desarrollo de la aplicación móvil, una eficiente gestión de las tareas, fraccionando el proyecto en iteraciones claras y simplificando las entregas requeridas.
- Como conclusión final, la aplicación móvil propuesta permitirá a los ciudadanos reportar de manera más eficiente y precisa los daños en la infraestructura pública urbana, mediante la adopción del sistema, facilitaría a las autoridades a identificar y solucionar los problemas de manera más rápida y eficiente.

4.2 Recomendaciones

- Establecer un sistema de retroalimentación con los usuarios para recopilar comentarios sobre usabilidad de la aplicación y así introducir actualizaciones periódicas con mejoras basadas en la retroalimentación de los usuarios, asegurando la mejora constante del sistema.
- Capacitar previamente al equipo de desarrollo del proyecto sobre las herramientas y características de Flutter, así como versiones, esto permitirá un flujo de trabajo eficiente y menos retrasos en la construcción del proyecto, para mejorar continuamente la experiencia de usuario.
- Documentar de forma detallada las iteraciones del proyecto, incluyendo información adicional, como lecciones aprendidas, propuestas de mejora, implementar revisiones regulares del proceso para identificar posibles ajustes y optimizaciones en las próximas fases del desarrollo del proyecto.
- Se recomienda a las instituciones públicas a promover el desarrollo de aplicaciones móviles que faciliten la comunicación de problemas ciudadanos y mejoren la accesibilidad a los servicios públicos, como resultado aumentará la eficiencia administrativa y fortalecerá la interacción entre la comunidad y las entidades gubernamentales.

REFERENCIAS BIBLIOGRÁFICAS

- [1] D. Bastos, A. Fernández-Caballero, A. Pereira, and N. P. Rocha, "Smart City Applications to Promote Citizen Participation in City Management and Governance: A Systematic Review," *Informatics*, vol. 9, no. 4. MDPI, Dec. 01, 2022. doi: 10.3390/informatics9040089.
- [2] S. J. E. Ordoñez, D. A. D. Toledo, and M. I. A. Campoverde, "Reflections on e-government and citizen participation in Ecuador," *Revista Eurolatinoamericana de Derecho Administrativo*, vol. 8, no. 1, pp. 77–98, Jan. 2021, doi: 10.14409/REDOEDA.V8I1.9562.
- [3] C. P. Chasi, "APLICACIÓN MÓVIL DE APOYO A LA SEGURIDAD BARRIAL PARA ENVÍO Y LOCALIZACIÓN DE ALERTAS DE AUXILIO MEDIANTE NOTIFICACIONES PUSH EN LA PARROQUIA SANTA ROSA DE LA CIUDAD DE AMBATO," 2022.
- [4] M. Wang and X. Yin, "Construction and maintenance of urban underground infrastructure with digital technologies," *Automation in Construction*, vol. 141. Elsevier B.V., Sep. 01, 2022. doi: 10.1016/j.autcon.2022.104464.
- [5] X. Wang, Y. Chen, Z. Han, X. Yao, P. Gu, and Y. Jiang, "Evaluation of mobile-based public participation in China's urban planning: Case study of the PinStreet platform," *Cities*, vol. 109, Feb. 2021, doi: 10.1016/j.cities.2020.102993.
- [6] K. Chung, S. Kim, E. Lee, and J. Y. Park, "Mobile app use for insomnia self-management in urban community-dwelling older Korean adults: Retrospective intervention study," *JMIR Mhealth Uhealth*, vol. 8, no. 8, Aug. 2020, doi: 10.2196/17755.
- [7] V. Miranda Torres, "PROPUESTA DE PLAN DE CONTROL DE CALIDAD DE OBRA DE INFRAESTRUCTURA.," 2021.
- [8] D. Georges, B. Précieux, S. Jadix, D.-Y. Casilda, D.-V. Raymond, and P. Béhanzin, "Participatory public policies The experience of urban development Decentralization Support Project in Benin Politiques publiques participatives Expérience des projets d'appui au développement urbain et à la décentralisation au Bénin," *African Journal on Land Policy and Geospatial Sciences*, vol. 4, no. 3, pp. 2657–2664, 2021, doi: 10.48346/IMIST.PRSM/ajlp-gs.v4i3.23698.
- [9] J. R. Molina Ríos, J. A. Honores Tapia, N. Pedreira-Souto, and H. P. Pardo León, "Comparativa de metodologías de desarrollo de aplicaciones móviles," *3C Tecnología_Glosas de innovación aplicadas a la pyme*, vol. 10, no. 2, pp. 73–93, Jun. 2021, doi: 10.17993/3ctecno/2021.v10n2e38.73-93.
- [10] B. Richardson, M. Campbell-Yeo, and M. Smit, "Mobile Application User Experience Checklist: A Tool to Assess Attention to Core UX Principles," *Int J*

- Hum Comput Interact*, vol. 37, no. 13, pp. 1283–1290, Aug. 2021, doi: 10.1080/10447318.2021.1876361.
- [11] I. Castro Balmaceda, M. Fernandez, S. Rodriguez, C. Vera, L. Fuentes, and J. Magaquian, “Interfaz y Experiencia de Usuario,” 2020.
- [12] G. A. Tubón Cando, “Aplicación móvil con Georreferenciación para gestión de pedidos a domicilio de un local de comida.,” 2020.
- [13] G. D. Corzo and E. L. Alvarez-Aros, “Technological competitiveness strategies in mobile connectivity and industry 4.0 communications in Latin America,” *Informacion Tecnologica*, vol. 31, no. 6, pp. 183–192, 2021, doi: 10.4067/S0718-07642020000600183.
- [14] M. Ogryzek, W. Krupowicz, and N. Sajnog, “Public participation as a tool for solving socio-spatial conflicts of smart cities and smart villages in the sustainable transport system,” *Remote Sens (Basel)*, vol. 13, no. 23, Dec. 2021, doi: 10.3390/rs13234821.
- [15] M. Rojas, A. F. Asesor, M. Villanueva Calderón, and J. Amilcar, “PROPUESTA DE UN PLAN DE INVERSIONES PARA MEJORAR LA INFRAESTRUCTURA VIAL EN LA CIUDAD DE CHICLAYO,” 2020.
- [16] B. A. Atencio Fuentes, “Propuesta de plan para el mantenimiento preventivo y correctivo de la infraestructura civil de Nacazcol Hotel y Villas,” Jul. 2022.
- [17] M. E. Gutiérrez-Mozo, J. Parra-Martínez, and A. Gilsanz-Díaz, “Mainstreaming a gender perspective into the management of infrastructures. The case of alicantine university campus,” *Ciudad y Territorio Estudios Territoriales*, vol. 52, no. 203, pp. 103–120, 2020, doi: 10.37230/CyTET.2020.203.09.
- [18] I. Picie-Alcaraz, B. A. Olivares-Zepahua, I. Lopez-Martinez, C. Romero-Torres, and L. A. Reyes-Hernandez, “Design of an IDE for Teaching Programming using Graphic Elements,” in *Applications in Software Engineering - Proceedings of the 9th International Conference on Software Process Improvement, CIMPS 2020*, Institute of Electrical and Electronics Engineers Inc., Oct. 2020, pp. 78–86. doi: 10.1109/CIMPS52057.2020.9390149.
- [19] AcensTechnologies, “Framework para el desarrollo ágil de aplicaciones,” 2020.
- [20] D. Jaramillo, K. Romero G, and C. Ramos, “Framework de seguridad para desarrollo de aplicaciones móviles y su aporte a la CiberResiliencia,” 2020.
- [21] P. Hernández-Lamas, B. Cabau-Anchuelo, Ó. de Castro-Cuartero, and J. Bernabéu-Larena, “Mobile Applications, Geolocation and Information Technologies for the Study and Communication of the Heritage Value of Public Works,” *Sustainability*, vol. 13, no. 4, p. 2083, Feb. 2021, doi: 10.3390/su13042083.
- [22] Ntrepid, “What Is Geolocation OSINT?,” 2022.

- [23] GADMA, "Obras de mantenimiento en las vías de Ambato." 2020. [Online]. Available: <https://www.ambato.gob.ec/obras-de-mantenimiento-en-las-vias-de-ambato/>
- [24] Z. Mat, "Flutter vs. React Native in 2022." 2022. [Online]. Available: <https://www.amplifyre.com/articles/flutter-vs-react-native-vs-xamarin-best-in-2022>

ANEXOS

Anexo A. Manual de usuario

En la Figura A1 se muestra la portada del manual de usuario de la aplicación móvil.



Figura A1. Portada manual de usuario.

En la figura A2 se muestra indicaciones sobre como iniciar sesión en la aplicación.

Inicio de Sesión



Esta es la pantalla de inicio en la que el usuario puede iniciar sesión, registrarse o recuperar su contraseña.

* Todos los campos son requeridos

1. Campo de nombre de usuario
2. Campo de contraseña
3. Botón mostrar/ocultar contraseña
4. Checkbox para mantener la sesión del usuario
5. Botón para ir a la página de recuperación de contraseña
6. Botón para iniciar sesión en la aplicación
7. Botón para ir a la página de registrar nuevo usuario

The screenshot shows a mobile application interface for 'GeoReport'. At the top, there's a status bar with the time '3:04' and a signal strength indicator. Below that is the 'GeoReport' logo, which consists of a globe icon and the text 'GeoReport'. The main content area is a white card with a dark blue header. It contains two input fields: 'Usuario' with the text 'xdeybi_4' and 'Contraseña' with masked characters. Below the password field is a checkbox labeled 'Recordarme' and a link 'Recuperar contraseña'. There are two buttons: a dark blue 'Iniciar Sesión' button and a white 'Registrarse' button. A small red arrow points to the top right corner of the screen. Numbered callouts (1-7) are placed over the input fields, checkbox, and buttons to identify them.

Figura A2. Manual inicio de sesión

En la figura A3 se muestra indicaciones para recuperar la contraseña de la cuenta.

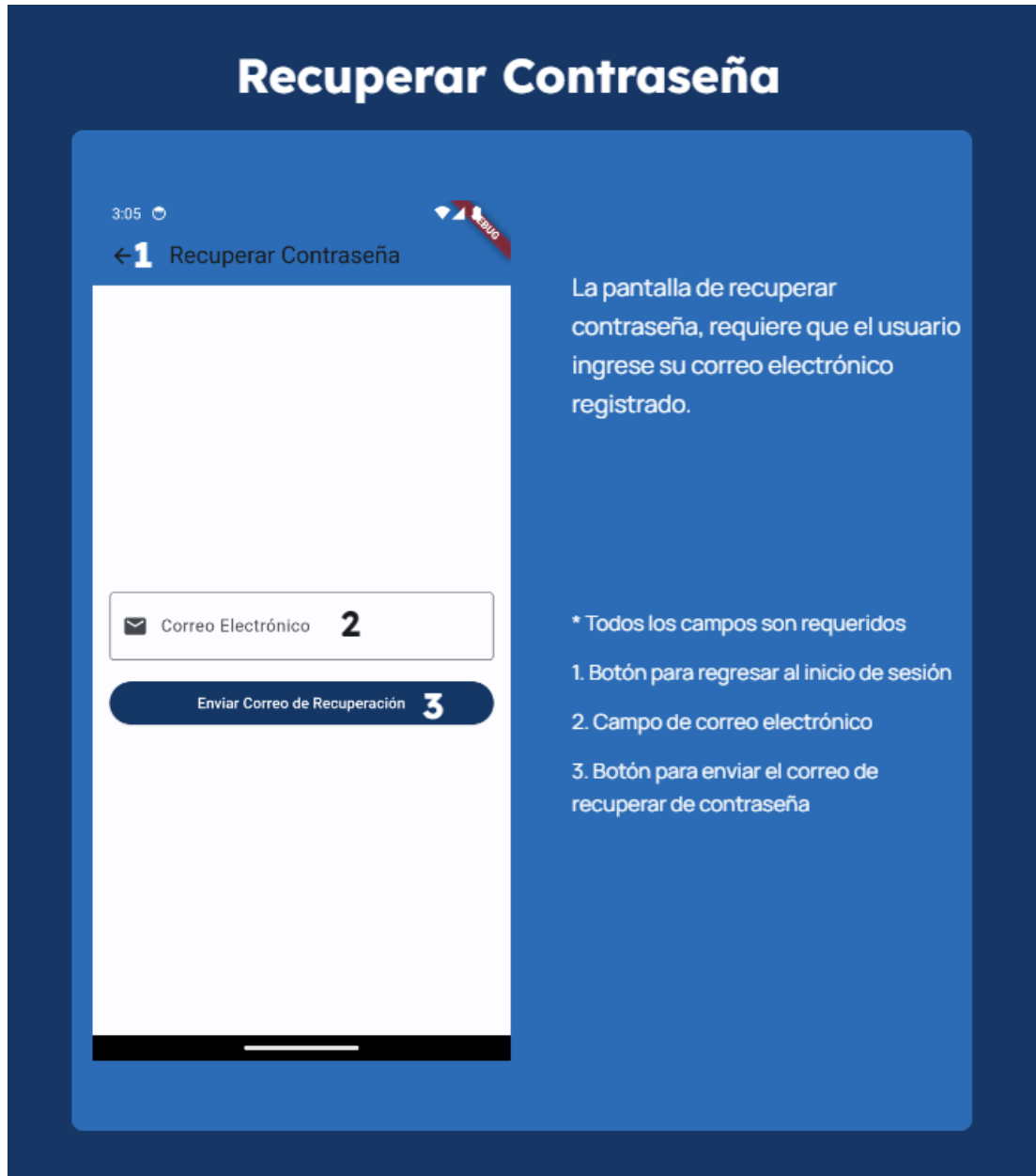


Figura A3. Manual recuperar contraseña.

En la figura A4 se muestra indicaciones sobre como registrar un usuario

Registrar Usuario

En la pantalla de registro, se requiere que el usuario ingrese información real que es validada, como un número de cédula válido y un correo que no haya sido registrado antes



Registarse

Foto de perfil
Seleccionar Imagen **1**

Nombre
Ingrese su nombre **2**

Apellido
Ingrese su apellido **3**

Cédula
Ingrese su cédula **4**

Correo
Ingrese su correo **5**

Usuario
Ingrese su usuario **6**

Contraseña
Ingrese su contraseña **7** 👁️ **8**

Confirmar contraseña
Confirme su contraseña **9** 👁️ **10**

Registarse **11**

o inicia sesión:

Iniciar Sesión **12**

* Todos los campos son requeridos

1. Botón para seleccionar una imagen de perfil de usuario
2. Campo de Nombre
3. Campo de Apellido
4. Campo de Cédula ecuatoriana
5. Campo de Correo electrónico
6. Campo de nombre de Usuario
7. Campo de contraseña
8. Botón mostrar/ocultar contraseña
9. Campo Confirmar contraseña
10. Botón mostrar/ocultar contraseña
11. Botón para Registrar nuevo usuario
12. Botón para volver al Inicio de sesión
13. Botón para Tomar una foto
14. Botón para Seleccionar una foto de la galería
15. Botón para Quitar foto seleccionada
16. Botón para Cancelar acción



Tomar Foto **13** 📷

Seleccionar Foto **14** 🖼️

Quitar foto **15** 🗑️

Cancelar **16**

Figura A4. Manual registrar usuario.

En la figura A5 se muestra el menú lateral en un perfil de usuario común.

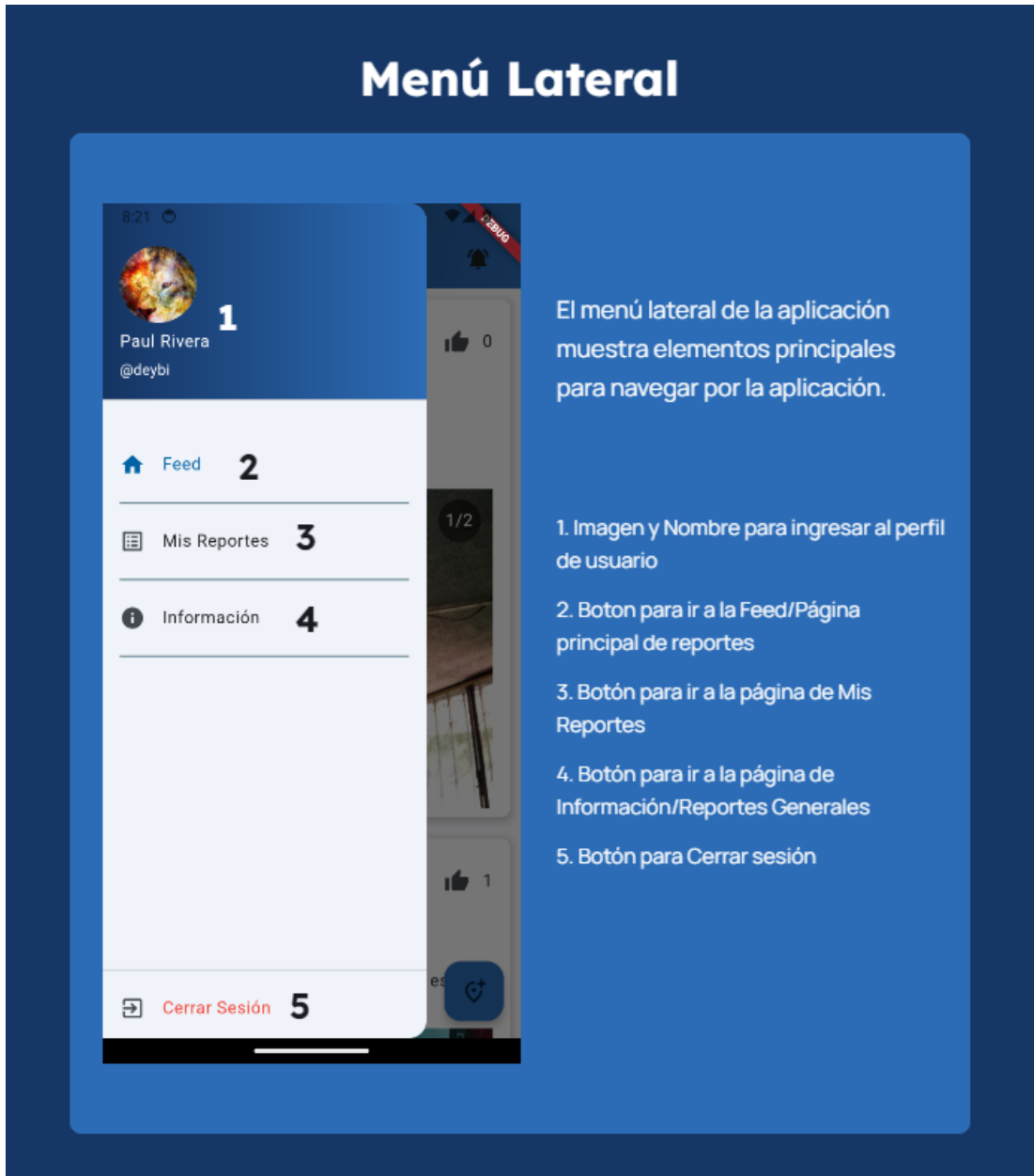


Figura A5. Manual menú lateral de usuario común.

En la figura A6 se muestra la página de perfil de usuario.

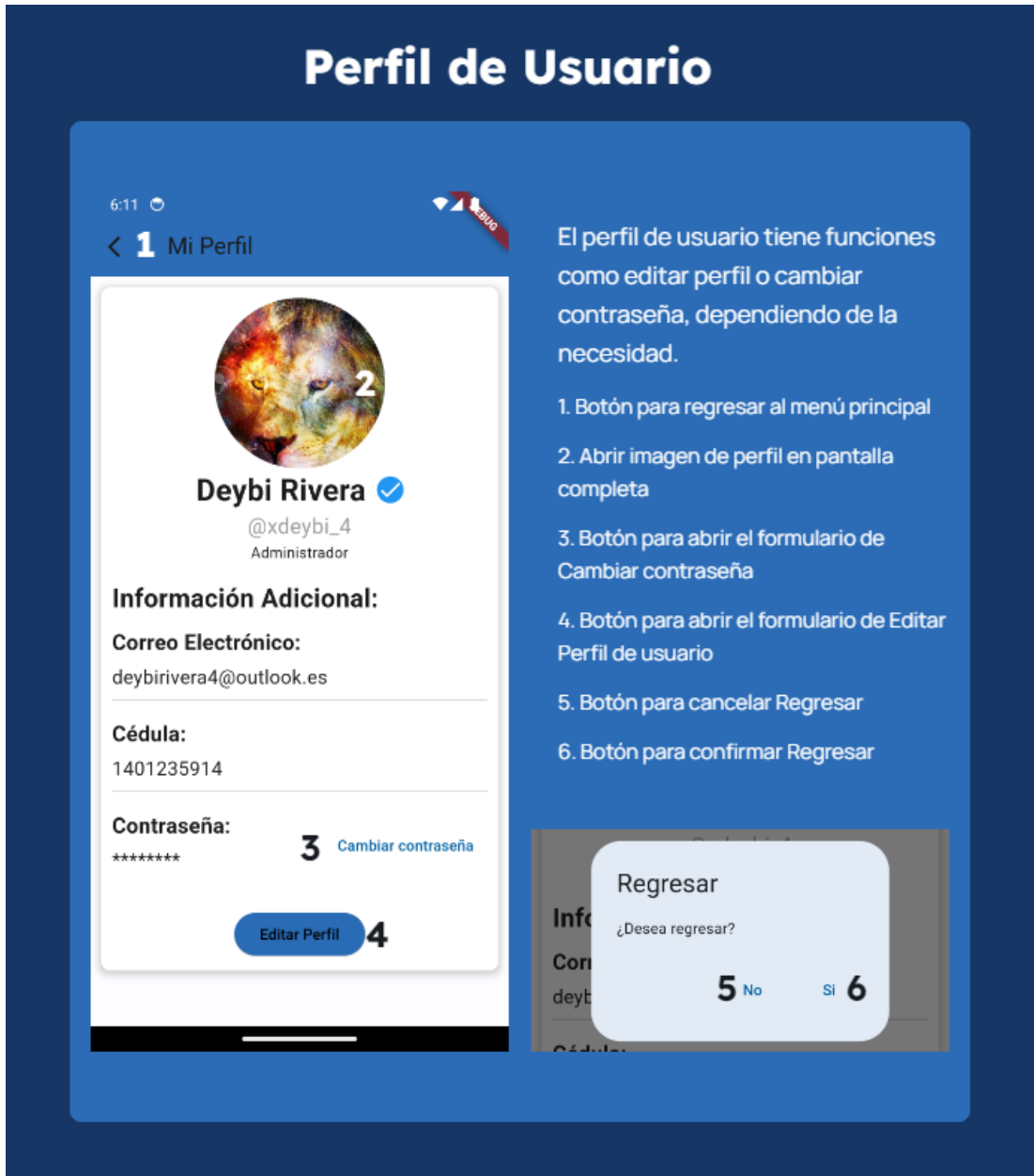


Figura A6. Manual perfil de usuario.

En la figura A7 se muestra indicaciones sobre como cambiar la contraseña.

Cambiar Contraseña

Para cambiar la contraseña, el usuario debe poseer su información de contraseña y completar los campos requeridos.

* Todos los campos son requeridos

1. Campo de Contraseña anterior
2. Campo de Nueva contraseña
3. Botón mostrar/ocultar contraseña
4. Campo de Confirmar nueva contraseña
5. Botón mostrar/ocultar contraseña
6. Botón para Guardar los cambios
7. Botón para Descartar los cambios
8. Botón para Confirmar los cambios

Figura A7. Manual cambiar contraseña.

En la figura A8 se muestra indicaciones sobre como editar el perfil de usuario.

Cambiar Información Personal

1:40

< Mi Perfil

 **1**
@xdeybi_4

Nombre
Deyb **2**

Apellido
Rivera **3**

Información Adicional:
Correo Electrónico:
deybirivera4@outlook.es
Cédula:
1401235914
Contraseña:

Guardar **4**

Para editar el perfil de usuario, existen campos de texto para cambiar el nombre y apellido, además cuenta con un botón para cambiar la foto de perfil.

* Todos los campos son requeridos

1. Botón para seleccionar nueva foto
2. Campo de Nombre
3. Campo de apellido
4. Botón para Guardar los cambios
5. Botón para Descartar los cambios
6. Botón para Confirmar los cambios
7. Botón para Tomar una foto
8. Botón para Seleccionar una foto de la galería
9. Botón para Quitar foto seleccionada
10. Botón para Cancelar acción

Guardar
¿Desea guardar los cambios?
5 Descartar **6** Confirmar

Tomar Foto **7**

Seleccionar Foto **8**

Quitar foto **9**

Cancelar **10**

Figura A8. Manual editar perfil de usuario.

En la figura A9 se muestra la página principal de reportes.

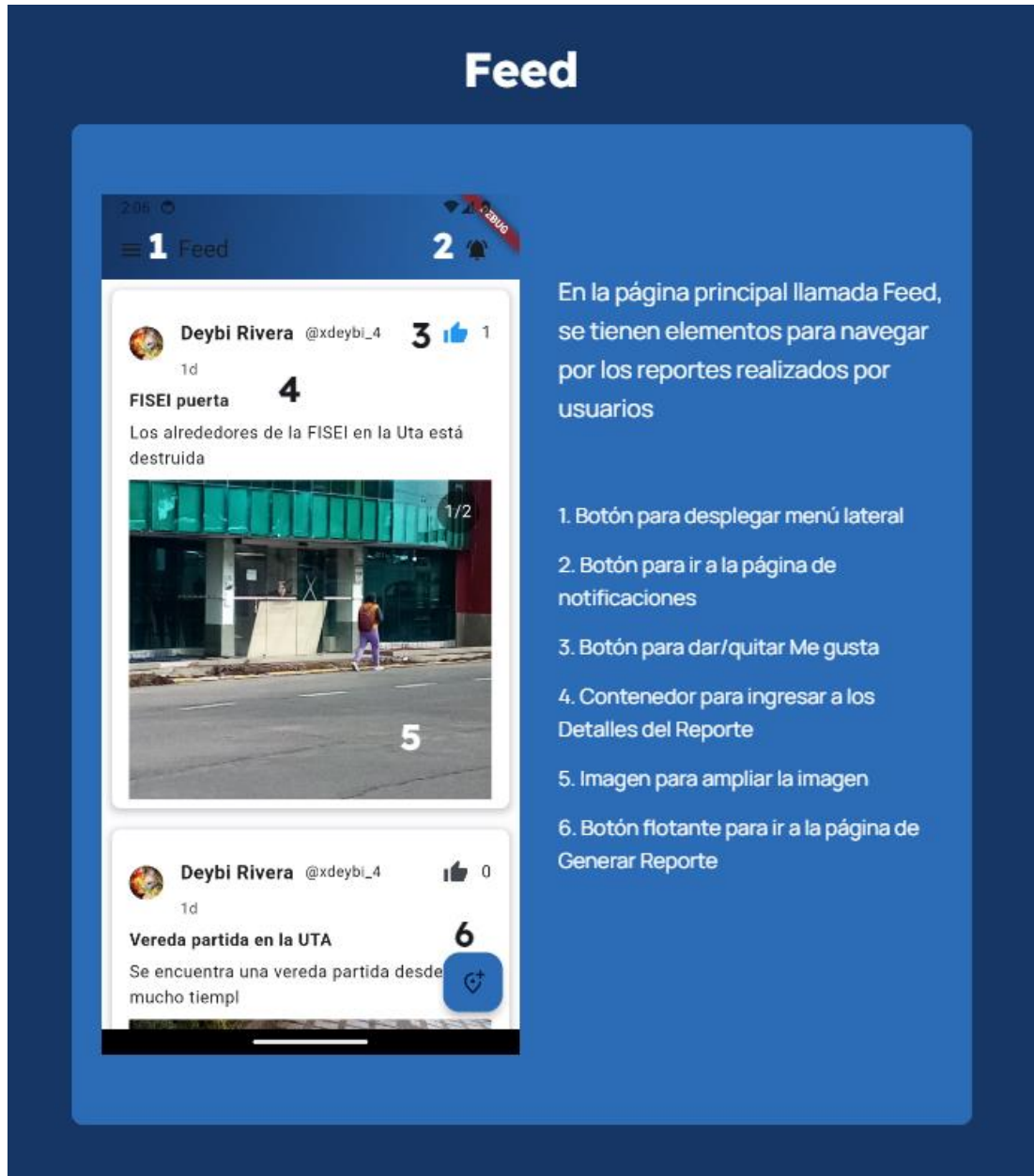


Figura A9. Manual página principal.

En la figura A10 se muestra la página de detalles del reporte.

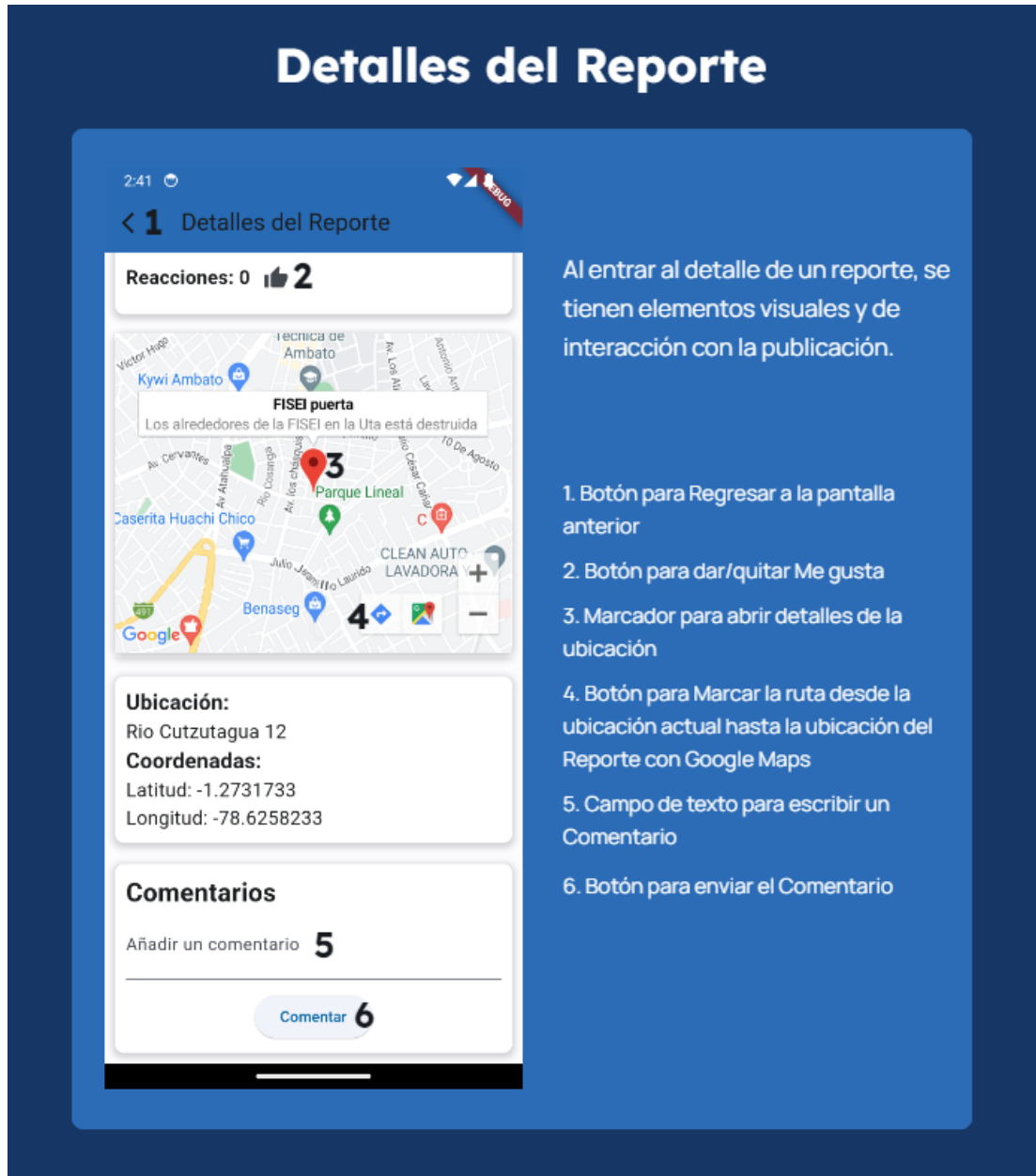


Figura A10. Manual detalles del reporte.

En la figura A11 se muestra la página de reportes del usuario.

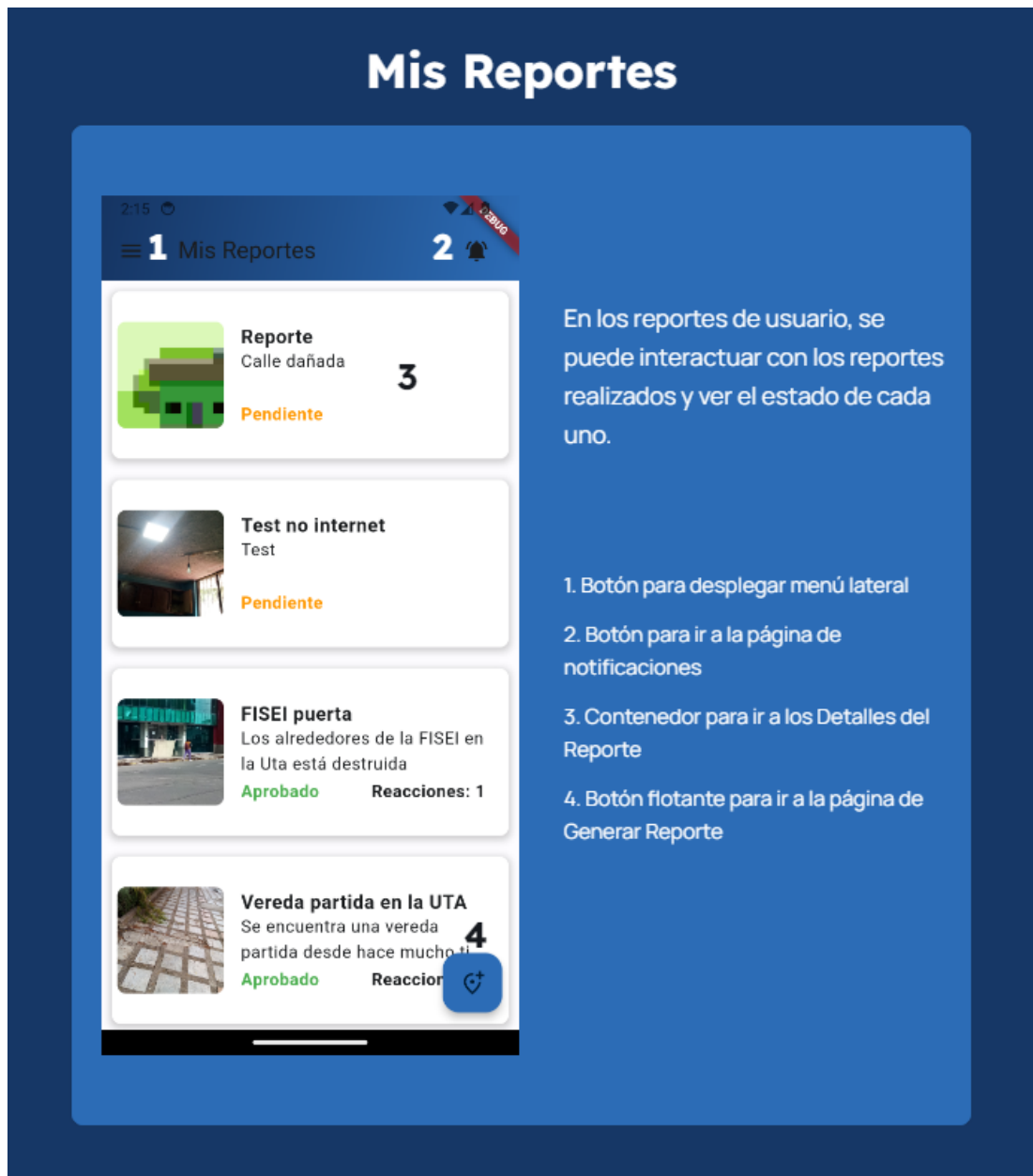


Figura A11. Manual reportes de usuario.

En la figura A12 se muestra indicaciones sobre como editar un reporte.

Editar Reporte

2:49


< 1 Detalles del Reporte

Deybi Rivera @xdeybi_4 1d

Reporte

Categoría: Calles

Calle dañada



2 Editar

* Todos los campos son requeridos

1. Botón para Regresar a la pantalla anterior
2. Botón para Editar un reporte en estado Pendiente.
3. Imagen para ampliar la imagen
4. Botón para Cancelar la edición
5. Menú desplegable para seleccionar la Categoría
6. Campo de Título
7. Campo de Descripción
8. Botón para Finalizar la edición
9. Botón para cancelar Regresar
10. Botón para confirmar Regresar
11. Botón para Descartar los cambios
12. Botón para Confirmar los cambios

Regresar

¿Desea regresar?

9 No **10** Si

Guardar

Guardar cambios.

11 Descartar **12** Guardar

Deybi Rivera @xdeybi_4 1d

4 Cancelar

Categoría:

Calles **5** ↓

Título:

Reporte **6**

Descripción:

Calle dañada **7**

Deybi Rivera @xdeybi_4 1d

8 Finalizar

Categoría:

Calles ↓

Título:

Reportes

Descripción:

Calle dañada

Figura A12. Manual editar reporte.

En la figura A13 se muestra la página de información general de reportes.

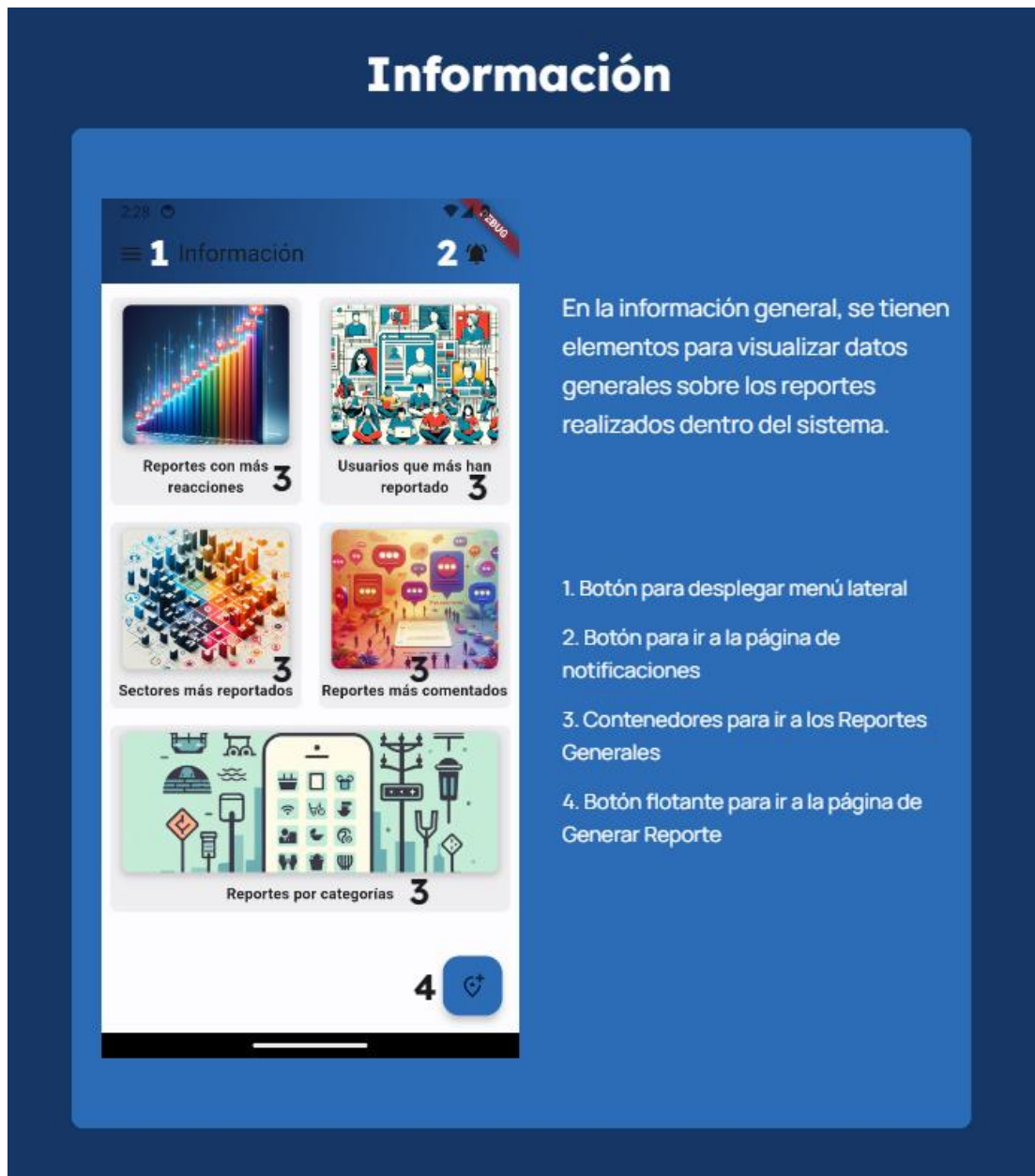


Figura A13. Manual información general de reportes.

En la figura A14 se muestra la página de notificaciones.

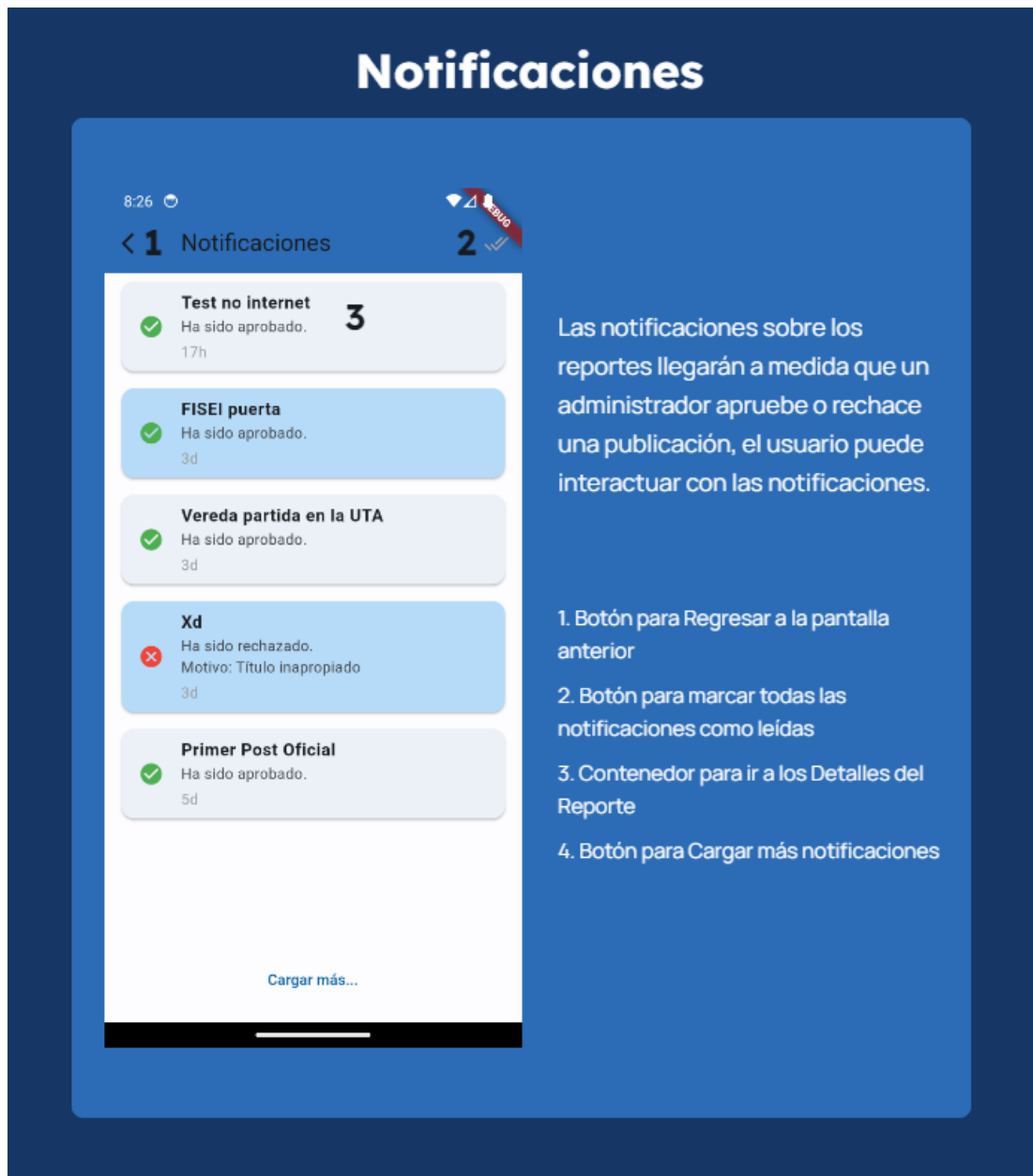


Figura A14. Manual notificaciones.

En la figura A15 se muestra indicaciones sobre como generar un nuevo reporte.

Generar Reporte



Tomar una foto:
Puede seleccionar más de una foto

2

3x

Categoría:
Alcantarillados 4 ↓

Título:
Escribe un título 5

Descripción:
Escribe una descripción 6

7 >

Ubicación:

Para generar un reporte, se debe completar todos los campos, con información real para que un administrador apruebe la publicación.

* Todos los campos son requeridos

1. Botón para Regresar a la pantalla anterior
2. Botón para seleccionar una foto
3. Botón para quitar la foto seleccionada
4. Menú desplegable para seleccionar la Categoría
5. Campo de Título
6. Campo de Descripción
7. Botón flotante para ir a la página de Generar Reporte
8. Botón para cancelar Regresar
9. Botón para confirmar Regresar
10. Botón para Tomar una foto
11. Botón para Seleccionar una foto de la galería
12. Botón para Cancelar acción
13. Botón para Cancelar Enviar Reporte
14. Botón para Confirmar Enviar Reporte



Salir
¿Desea cancelar el reporte?
8 No 9



Tomar Foto 10
Esc
Seleccionar Foto 11
Desc
Esc
Cancelar 12



Enviar Reporte
Su reporte será enviado a revisión para su aprobación.
¿Desea enviar el reporte?
13 No 14

Figura A15. Manual generar reporte.

En la figura A16 se muestra el menú lateral en un perfil de administrador.

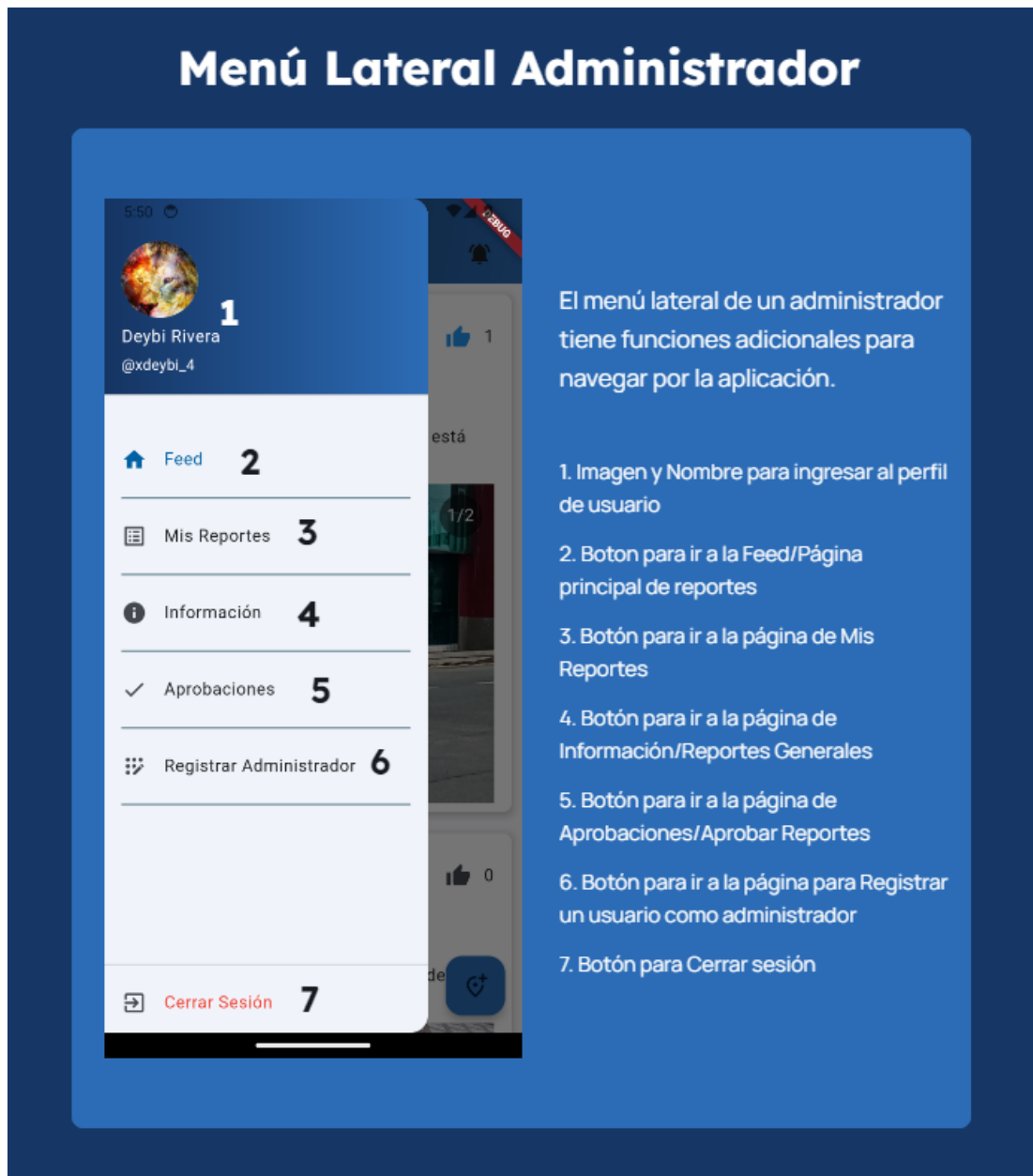


Figura A16. Manual menú lateral de administrador.

En la figura A17 se muestra la página de aprobación de reportes para administradores.

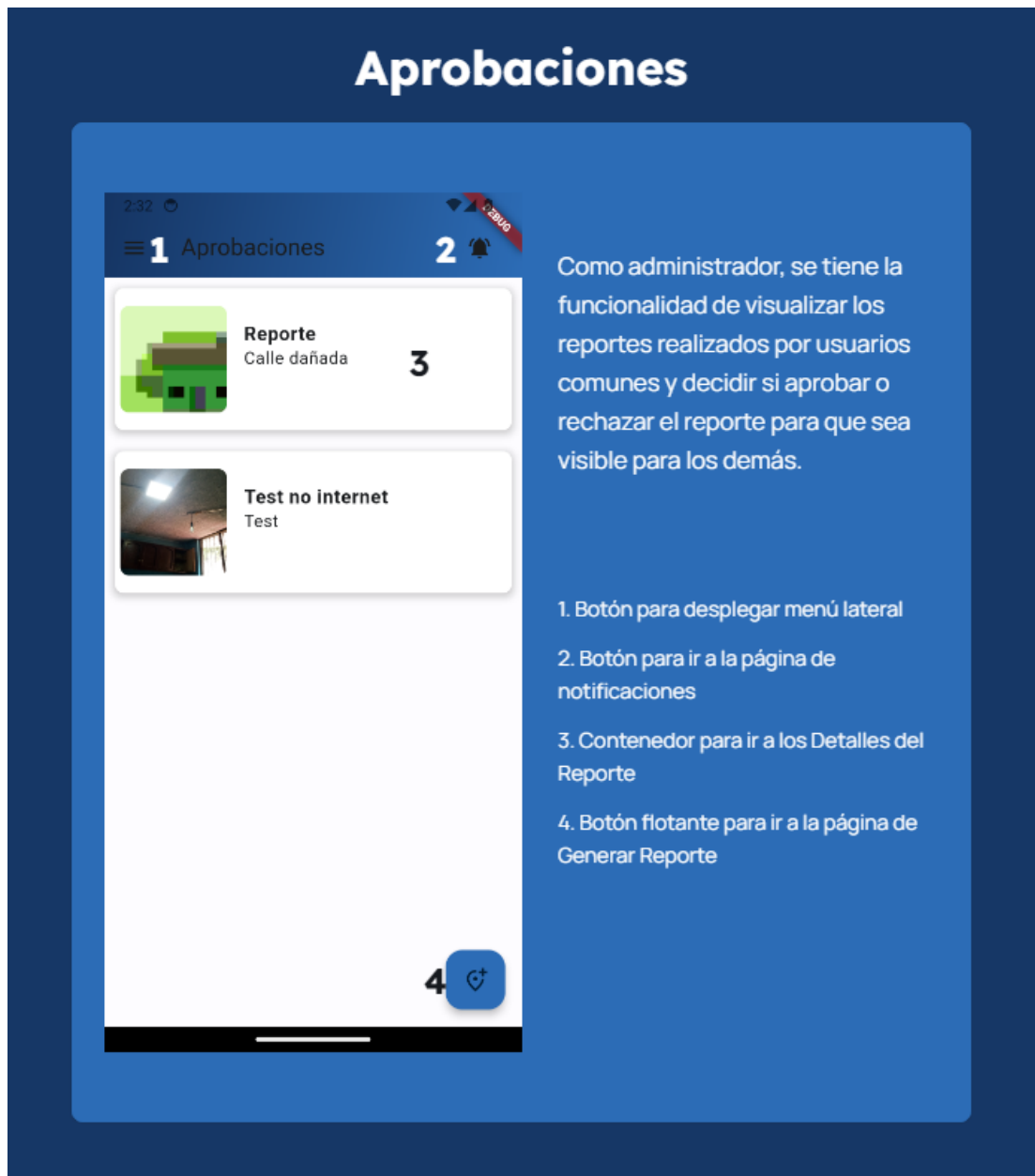


Figura A17. Manual aprobaciones.

En la figura A18 se muestra indicaciones sobre como aprobar o rechazar un reporte.

Aprobar/Rechazar Reporte



Al entrar a un reporte pendiente de revisión, el administrador decide aprobar o rechazar el reporte con los botones correspondientes.

1. Botón para Regresar a la pantalla anterior
2. Imagen para ampliar la imagen
3. Botón para Aprobar un reporte
4. Botón para Rechazar un reporte
5. Botón para cancelar Aprobar
6. Botón para confirmar Aprobar
7. Menú desplegable para seleccionar el Motivo de rechazo
8. Botón para cancelar Rechazar
9. Botón para confirmar Rechazar

Figura A18. Manual aprobar/rechazar reporte como administrador.