



POSGRADOS

MAESTRÍA EN

SOFTWARE CON MENCIÓN EN DESARROLLO WEB Y MÓVIL

RPC-SO-34-NO.778-2021

OPCIÓN DE TITULACIÓN:

PROYECTO DE TITULACIÓN CON
COMPONENTES DE INVESTIGACIÓN
APLICADA Y/O DE DESARROLLO

TEMA:

[DISEÑO DE UNA APLICACIÓN MÓVIL
PARA REDES DE COLABORACIÓN SOCIAL]

AUTOR(ES)

LAURA ELIZABETH ALMEIDA CÁRDENAS
STEPHANY PAULETTE CONTRERAS DELGADO

DIRECTOR:

PATSY MALENA PRIETO VELEZ

QUITO – ECUADOR
2025



Autores:***Laura Elizabeth Almeida Cárdenas***

Ingeniera de Sistemas de Computación e Informática
Candidata a Magíster en Software por la Universidad Politécnica Salesiana – Sede Quito
lalmeidac2@est.ups.edu.ec

***Stephany Paulette Contreras Delgado***

Ingeniera de Sistemas
Candidata a Magíster en Software por la Universidad Politécnica Salesiana – Sede Guayaquil
scontrerasd@est.ups.edu.ec

Dirigido por:***Patsy Malena Prieto Vélez***

Ingeniera en Sistemas Informáticos y de Computación
Magister en Gestión Informática Empresarial
pprieto@ups.edu.ec

Todos los derechos reservados.

Queda prohibida, salvo excepción prevista en la Ley, cualquier forma de reproducción, distribución, comunicación pública y transformación de esta obra para fines comerciales, sin contar con autorización de los titulares de propiedad intelectual. La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual. Se permite la libre difusión de este texto con fines académicos e investigativos por cualquier medio, con la debida notificación a los autores.

DERECHOS RESERVADOS

2025 © Universidad Politécnica Salesiana.

QUITO– ECUADOR – SUDAMÉRICA

Laura Elizabeth Almeida Cárdenas

Stephany Paulette Contreras Delgado

Diseño de una aplicación móvil para redes de colaboración social

DEDICATORIA

A Dios primero, porque es la fuente de mi fortaleza, quien conoce todo sobre mí, y me ama más allá de lo que puedo comprender. A El, quien me salvó y me restauró, por su gran misericordia me ha permitido llegar hasta aquí, aun cuando fue un camino difícil de recorrer.

Dedico este trabajo a mis tres hijos, David, Juanito y Sarita, que son la fuente de mi alegría, de mi gozo, por quienes he luchado para verlos crecer fuertes, valientes, decididos, responsables, agradecidos, y sobre todo amados, por Dios y por su madre. A ellos, gracias por creer en mí, y por tener siempre una sonrisa, una palabra de aliento para seguir adelante.

A los miembros de mi amada iglesia IEVI Norte, que fue la inspiración para el tema de este proyecto, a quienes de todo corazón, espero le sea de utilidad. Que Dios siga abriendo caminos para fortalecer los lazos de hermanos en Cristo.

Elizabeth Almeida

A mis padres, quienes desde el inicio me mostraron el valor del conocimiento y la importancia de la perseverancia. A mi compañero de vida, por su paciencia y comprensión durante las largas horas de trabajo. A mis maestros y mentores, que con su guía me han dado las herramientas necesarias para completar este proyecto. Gracias a todos por su presencia y apoyo inquebrantable en este camino.

Stephany Contreras

AGRADECIMIENTO

Expreso mi gratitud a mis hijos y a mi madre, que fueron un apoyo incondicional en este camino recorrido, y quienes siempre tuvieron palabras de aliento y confianza para culminar esta carrera.

Gracias a todo el cuerpo docente y administrativo de la Universidad Politécnica Salesiana, en especial a Rodrigo, Patsy, Lilian, Mónica, Pablo, y Sebastián, quienes impartieron su conocimiento de una forma dedicada y generosa, fomentando en mí un continuo anhelo de aprender y un compromiso para completar con excelencia la carrera.

Gracias a Juan Carlos, Darío, Marco, Stephy, y a todos mis compañeros de Maestría por compartir sus conocimientos, experiencias y amistad, enriqueciendo no solo mi aprendizaje académico, sino también mi crecimiento personal a lo largo de este camino.

Elizabeth Almeida

AGRADECIMIENTO

En primer lugar, agradezco a Dios, quien me ha dado la fortaleza y sabiduría necesarias para llegar hasta aquí, iluminando mi camino y guiándome en cada paso de esta etapa de mi vida. Sin su constante presencia y guía, esta meta no habría sido posible.

A mi amado esposo, mi mayor compañero, quien ha estado a mi lado de manera incondicional, brindándome apoyo constante, ánimo y ayuda en cada momento que lo he necesitado. Tu paciencia, comprensión y amor han sido pilares fundamentales en este proceso, y te agradezco profundamente por caminar conmigo en esta travesía.

A mi compañera de tesis, por su invaluable disposición y apoyo. Gracias por estar siempre lista para ayudarme, por cada consejo y cada instante de colaboración. Juntas hemos logrado superar los desafíos de este trabajo, y estoy profundamente agradecida por haber compartido esta experiencia a tu lado.

A mi familia, mi más grande inspiración y fortaleza. Gracias por ser el motor de mis sueños y los cimientos de la persona en la que me he convertido. Sus enseñanzas, valores y amor incondicional me han formado y me motivan a continuar esforzándome y a perseguir siempre lo mejor.

Por último, extendiendo mi gratitud a la institución, que ha sido un pilar de apoyo durante todo mi recorrido en la maestría. Agradezco no solo por el respaldo brindado, sino también por el conocimiento invaluable que he recibido y que sin duda llevaré conmigo en mi futuro profesional.

A todos, gracias por su apoyo en esta etapa tan importante de mi vida.

Stephany Contreras

TABLA DE CONTENIDO

Resumen	11
Abstract	12
1 Introducción	13
2 Determinación del problema	14
2.1 Formulación del Problema.....	14
2.2 Justificación del Problema	15
2.3 Objetivos	16
3 Marco Teórico Referencial	17
3.1 Ingeniería de software	17
3.2 Experiencia de usuario UX	18
3.3 Valor para el negocio	19
3.4 Metodologías ágiles	19
3.5 Metodología Lean Startup	19
3.6 Etapas de Lean Startup	21
3.7 Pasos de Lean Startup.....	22
3.8 Principios SOLID	23
3.9 Código Limpio	24
4 Materiales y Metodología	25
4.1 Materiales	25
4.2 Metodología Lean Startup	29
4.3 Identificar la idea	30
4.4 Crear una hipótesis	30
4.4.1 Informe de la experiencia de usuario.....	31
4.4.2 Requerimientos Funcionales	31
4.4.3 Requerimientos No Funcionales	33
4.4.4 Diagramas C4	35
4.4.5 Arquitectura.....	38
4.4.6 Mapa del sitio	41
4.4.7 User Flows	42
4.5 Prototipo	45
4.5.1 Prototipo Inicial	45
4.6 Validación.....	47

4.6.1	Lanzar el Prototipo	47
4.6.2	Medir y Recopilar Datos	49
4.7	Iteración	50
4.7.1	Mapa del sitio	51
4.7.2	User Flows	52
4.7.3	Prototipo final.....	55
5	Resultados y discusión.....	59
5.1	Modelo de Base de Datos	60
5.2	FireBase.....	61
5.2.1	FireBase Authentication	62
5.2.2	FireBase Firestore DataBase.....	62
5.2.3	FireBase Storage	63
5.2.4	FireBase App Check	63
5.3	Aplicación Móvil.....	64
5.3.1	Código Fuente en Kotlin	64
5.3.2	Interfaces al Usuario - Requerimientos Funcionales.....	66
5.3.3	Requerimientos no Funcionales	71
5.4	Discusión	73
5.4.1	Metodología	73
5.4.1	Materiales.....	74
6	Conclusiones.....	75
7	Trabajo futuro.....	77
8	Referencias	78
Anexo A:	Informe UX	80
Colaboradores o Equipo de trabajo.....		80
Público		80
Entrevista abierta		80
Card Sorting		82
Mapas de Empatía		83
User Persona.....		84
Journing Map.....		85
Anexo B:	Resultados del test de usabilidad.....	86
Anexo C:	Resultados de la encuesta.....	89

ÍNDICE DE IMÁGENES

Imagen 1: Etapa Startup Construir	21
Imagen 2: Etapa Startup Medir.....	21
Imagen 3: Etapa Startup Aprender	21
Imagen 4: Pasos Startup Idea	22
Imagen 5: Pasos Startup Hipótesis	22
Imagen 6: Pasos Startup Prototipo	22
Imagen 7: Pasos Startup Validación.....	23
Imagen 8: Pasos Startup Iteración	23
Imagen 9: Herramienta Trello.....	25
Imagen 10: Herramienta Figma	25
Imagen 11: Herramienta Maze	26
Imagen 12: Herramienta Structurizr.....	26
Imagen 13: Herramienta Draw.io	27
Imagen 14: Herramienta Android Studio.....	27
Imagen 15: Herramienta Postman.....	28
Imagen 16: Consola de Firebase del Proyecto Red de Colaboración	28
Imagen 17: Repositorio del Proyecto en GitHub	29
Imagen 18: Diagrama C4 de Contexto de la Aplicación Red de Colaboración	36
Imagen 19: Diagrama C4 de Contenedores de la Aplicación Red de Colaboración.....	37
Imagen 20: Diagrama C4 de Componentes de la Aplicación Red de Colaboración.....	37
Imagen 21: Diagrama UML de Componentes de Arquitectura	38
Imagen 22: Arquitectura con enfoque tradicional versus Firebase	39
Imagen 23: Backend de la App Red de Colaboración.....	40
Imagen 24: Mapa del Sitio, primera iteración	41
Imagen 25: User Flow para Registrar Datos Personales, primera iteración.....	42
Imagen 26 User Flow para Registrar la Solicitud de Ayuda, primera iteración	43
Imagen 27: User Flow para Registrar una Ayuda Efectivizada, primera iteración	43
Imagen 28: User Flow para Registrar las Categorías de Ayuda, primera iteración	44
Imagen 29: Prototipo, Pantalla de Inicio de Sesión, primera iteración.....	45
Imagen 30: Prototipo, Pantalla de Solicitud de Ayuda, primera iteración	46
Imagen 31: Prototipo, Pantalla de Ofrecer Ayudas, primera iteración	46
Imagen 32: Prototipo, Pantalla de Perfil y Categorías, primera iteración.....	47
Imagen 33: Mapa del Sitio, segunda iteración.....	51
Imagen 34: User Flow para Registrar Datos Personales, segunda iteración	52
Imagen 35 User Flow para Registrar la Solicitud de Ayuda, segunda iteración	53

Imagen 36: User Flow para Responder a la Solicitud de Ayuda, segunda iteración.....	53
Imagen 37: User Flow para Registrar una Ayuda Efectivizada, segunda iteración.....	54
Imagen 38: Prototipo, Pantalla de Eventos e Inicio de Sesión, segunda iteración.....	55
Imagen 39: Prototipo, Pantalla de Solicitud de Ayuda, segunda iteración	56
Imagen 40: Prototipo, Pantalla de Ofrecer Ayudas, segunda iteración	57
Imagen 41: Prototipo, Pantalla de Perfil y Categorías, segunda iteración	58
Imagen 42: Diagrama de Base de Datos	60
Imagen 43: Backend de la App Red de Colaboración.....	61
Imagen 44: Firebase Authentication.....	62
Imagen 45: Firebase Firestore Database	63
Imagen 46: Firebase Storage	63
Imagen 47: Firebase AppCheck.....	64
Imagen 48: Programación en Lenguaje Kotlin	65
Imagen 49: Pantallas de Bienvenida y Eventos.....	66
Imagen 50: Pantallas de Login y Perfil de Usuario	67
Imagen 51: Pantallas de Solicitud de Ayuda e Historial (Solicitudes).....	68
Imagen 52: Pantallas de Lista de Solicitudes y Ofrecer Ayuda.....	69
Imagen 53: Pantallas de Historial (Ayudas realizadas) y Cierre de Solicitudes	70
Imagen 54: Firebase Performance.....	72

DISEÑO DE UNA APLICACIÓN MÓVIL PARA REDES DE COLABORACIÓN SOCIAL

AUTORES:

LAURA ELIZABETH ALMEIDA CARDENAS
STEPHANY PAULETTE CONTRERAS
DELGADO |

RESUMEN

La aplicación móvil para redes de colaboración social surge de la necesidad de contar con una herramienta que permita canalizar los requerimientos emergentes de personas que pertenecen a la Iglesia IEVI Norte, ubicada en la ciudad de Quito. Existen necesidades que muchas veces por temor, recelo o creer que no habrá una persona que los apoye, los miembros de la iglesia no lo comunican, o si lo comunican, no encuentran a alguien que cubra su necesidad o quien la puede brindar se informa demasiado tarde.

Esta experiencia es nueva para los usuarios, incluso a nivel del país no se ha encontrado una aplicación que permita registrar a los miembros de una comunidad, así como las áreas en las que pueden colaborar a otros, y que las personas en necesidad soliciten ayuda y la reciban en forma efectiva. El grupo es reducido, alrededor de 200 personas, pues la ayuda en varios casos es presencial, con víveres, atención médica, consejería, y el propósito es que incremente el sentido de pertenencia a la iglesia.

El valor de este proyecto no está directamente en el desarrollo de la aplicación, pues se utilizan herramientas ampliamente conocidas como Android Studio y Firebase. El valor se encuentra en crear algo nuevo de la forma más rápida posible con herramientas ágiles que permitan vender la idea de gestionar la colaboración en forma automatizada a los miembros que forman parte de la iglesia, para generar un producto que sea de verdadera utilidad.

Palabras clave:

Colaboración, Ayuda, Móvil, UX, Lean Startup, Firebase

ABSTRACT

The mobile application for social collaboration networks arises from the need to have a tool that allows channeling the emerging requirements of people who belong to the IEVI Norte Church, located in the city of Quito. There are needs that many times, due to fear, suspicion or believing that there will not be a person to support them, the members of the church do not communicate it, or if they communicate it, they do not find anyone to cover their need or who can provide it, is informed too much late.

This experience is new for member users, even at the country level, there is not an application that allows members of a community to register, as well as the areas in which they can collaborate with others, and people in need request help and receive it effectively. The group is small, around 200 people, since the help in several cases is in person, with groceries, medical care, counseling, and the purpose is to increase the sense of belonging to the church.

The value of this project is not directly in the development of the application, since widely known tools such as Android Studio and Firebase are used. The value is found in creating something new as quickly as possible with agile tools that allow to sell the idea of managing collaboration in an automated way to the members who are part of the church, to generate a product that is truly useful.

Key words: |

Collaboration, Help, Mobile, UX, Lean Startup, Firebase

1 INTRODUCCIÓN

En las tiendas de aplicaciones como App Store o Google Play existen muchas aplicaciones orientadas a promover la colaboración desinteresada hacia otras personas, con el fin de aumentar en pequeña o gran medida las oportunidades de los más necesitados. Es el caso de ShareTheMeal, una aplicación de la ONU que permite contribuir al Programa Mundial de Alimentos [1]. Charity Miles es otro ejemplo en el que se recauda dinero para la organización benéfica que se elija, de acuerdo a los metros recorridos al caminar, correr, bailar o montar en bicicleta; los fondos se obtienen de los patrocinadores, que obtienen el beneficio por los anuncios que incluyen dentro la aplicación. G-GET es una aplicación desarrollada para móviles que permite hacer donaciones de alimentos, educación y salud [2].

La aplicación propuesta en el presente proyecto tiene un concepto de colaboración similar a “Be My Eyes - Helping blind see”, en la cual las personas necesitadas son personas con problemas de visión, quienes solicitan ayuda a usuarios voluntarios videntes. Estos reciben las peticiones de ayuda a su celular mediante una notificación; se establece una conexión de audio y video entre el usuario que solicita la ayuda y el voluntario que acepta la petición para que este último pueda asistirle en cosas como saber la fecha de caducidad de un producto, o cruzar una calle. [3]

Sin embargo, no se encuentra una aplicación en la que no sólo se preste ayuda, sino que también se la reciba, entregando en forma voluntaria los recursos o el tiempo para beneficiar a otras personas que pertenezcan a un grupo específico o una comunidad. La IEVI Norte es una institución sin fines de lucro que encaja en un perfil de red social comunitaria. Su misión es servir a Dios y a la comunidad evangelizando y ayudando a los más necesitados del sector del Valle de Calderón-Carapungo. Esta ayuda se realiza tanto para los miembros que pertenecen a la iglesia como para los referidos por los miembros, en áreas como alimentación, salud, consejería, acompañamiento, u oración.

2 DETERMINACIÓN DEL PROBLEMA

En una comunidad como la iglesia IEVI Norte, el principal objetivo es la unidad en un solo cuerpo para evangelizar a las personas. Esta unidad se consigue cuando hay un bienestar y estabilidad en cada uno de sus integrantes. Sin embargo, existen personas que enfrentan situaciones difíciles o conocen de personas que necesitan ayuda. Afortunadamente, también existen personas que quieren y pueden colaborar con ayuda o donaciones en las diferentes áreas de necesidad.

La iglesia requiere primero, contar con una base de información de sus miembros y sus capacidades, y segundo, organizar la distribución de ayudas para canalizarlas en forma adecuada hacia las personas que las necesitan, muchas de las cuales no manifiestan su necesidad por desconocimiento, vergüenza o falta de motivación.

2.1 FORMULACIÓN DEL PROBLEMA

El problema es que los miembros de la iglesia son numerosos, alrededor de doscientos, y aunque están organizados en grupos, no cuentan con la información de los que requieren ayuda ni de los que la pueden ofrecer, tampoco se tiene un registro de las áreas en las que pueden ayudar las personas que están dispuestas a colaborar. Adicionalmente, la manera de canalizar las ayudas para conectar al que quiere ayudar con el que necesita la ayuda es muy engorrosa, todo lo cual vuelve el proceso demasiado lento y poco efectivo.

El proyecto consiste en dar una solución tecnológica para una organización social en la cual sus integrantes puedan dar o recibir bienes o servicios de acuerdo a las habilidades y las necesidades de cada uno. Se va a implementar en la iglesia IEVI Norte.

Esta solución tecnológica es una aplicación móvil en la que se cuenta con una base de datos de los integrantes de la IEVI, que pueden registrar los servicios de ayuda que pueden ofrecer, como asistencia médica, ayuda económica, en víveres, entre

otros. También pueden solicitar ayuda en las mismas áreas. La aplicación empareja a las personas que solicitan y a las que ofrecen la ayuda y emite las alertas correspondientes.

2.2 JUSTIFICACIÓN DEL PROBLEMA

En el mundo actual, existe una tendencia a buscar obtener un beneficio económico personal en todo lo que una persona emprende, y muchos adoptan comportamientos gratificantes con el fin de mejorar su calidad de vida, aunque en ocasiones estos comportamientos puedan conllevar costos significativos. Se está perdiendo la capacidad de dar sin esperar nada a cambio, y está demostrado que el dar proporciona una satisfacción personal que sobrepasa cualquier bien material. De hecho, existen beneficios para la salud asociados con dar a los demás, siendo posible que adoptar un comportamiento pro social tenga más éxito a la hora de reducir las respuestas psicológicas y fisiológicas al estrés [4].

Por esta razón y por la obvia necesidad que existen en muchos lugares, varias organizaciones promueven el uso de aplicaciones que permiten ayudar a las personas que nos rodean, para producir este espíritu de solidaridad con los más necesitados, y aumentar la satisfacción personal o empresarial. Adicionalmente, este tipo de aplicaciones permiten que los miembros que forman parte de una comunidad incrementen el sentido de pertenencia hacia la misma. Es así como el presente proyecto plantea el desarrollo de una aplicación móvil en donde se pueda registrar la forma en que cada integrante puede aportar a un grupo social, y al mismo tiempo recibir ayuda en el caso de necesitarla.

La iglesia IEVI Norte, al ser una organización que evangeliza y ayuda a los más necesitados, requiere este tipo de aplicación, pues le será de gran utilidad para resolver la falta de información y de organización en la gestión de las ayudas. Primero la iglesia va a disponer de una base de datos de todos los miembros de la iglesia y de las posibles ayudas que algunos integrantes pueden ofrecer. Luego, la aplicación conectará al que ofrece la ayuda con el que la necesita para efectivizar la

ayuda. Finalmente, en tiempos de crisis, esta aplicación se utilizará como la principal y primera vía de canalización de ayuda comunitaria.

El éxito del proyecto se podrá evidenciar a través de la cantidad de asistencias o ayudas realizadas en forma efectiva, la cual puede ser medible porque quien la recibe debe cerrar la solicitud de asistencia para poder solicitar otra ayuda nuevamente.

2.3 OBJETIVOS

El objetivo general es dar una solución tecnológica para una organización social en la cual sus integrantes puedan dar o recibir bienes o servicios de acuerdo a las habilidades y las necesidades de cada uno.

Los objetivos específicos son los siguientes:

- Recopilar las impresiones, emociones y expectativas de los usuarios de la IEVI - Iglesia Evangélica de Iñaquito- Norte respecto a esta nueva propuesta de aplicación.
- Determinar los requerimientos funcionales y no funcionales de la red de colaboración de la IEVI Norte.
- Diseñar una solución de una aplicación móvil para canalizar las ayudas entre los miembros de la IEVI y contar con una base de información de sus miembros.
- Diseñar, implementar y probar un MVP- Minimum Viable Product (producto mínimo viable) de la aplicación móvil para redes de colaboración.

3 MARCO TEÓRICO REFERENCIAL

El marco teórico del presente proyecto describe conceptos sobre ingeniería de software, que se han ido aplicando en el transcurso del desarrollo de la aplicación móvil para redes de colaboración social.

3.1 INGENIERÍA DE SOFTWARE

Según la definición de la IEEE 610.12-1990, la ingeniería de software es la aplicación sistemática, ordenada y cuantificable del desarrollo, operación y mantenimiento de software [5]. Los procesos utilizados para desarrollar un software de alta calidad comprenden desde la gestión del proyecto hasta la atención y seguimiento al cliente; en medio de estos, las etapas de definición de requerimientos, diseño, codificación, pruebas, despliegue, gestión de proveedores y de configuración, auditoría, procesos de medición y mejora.

La transición del desarrollo de software por etapas al enfoque ágil surge de la necesidad de adaptarse a la dinámica cambiante de los negocios de enfoque blando. La inflexibilidad de los métodos tradicionales no lograba satisfacer las demandas de un entorno donde los requisitos evolucionan constantemente.

En el contexto de una aplicación para redes de colaboración social, la incertidumbre es aún mayor debido a la falta de precedentes y la imprevisibilidad de la respuesta del usuario ante nuevas experiencias. Esto resalta la importancia de adoptar un enfoque ágil que permita iterar rápidamente sobre el producto y ajustarlo según la retroalimentación recibida.

La metodología ágil ofrece la ventaja de desarrollar un producto mínimo viable en un corto período de tiempo. Esto permite poner rápidamente en manos de los usuarios una versión inicial de la aplicación para obtener comentarios tempranos y relevantes. La retroalimentación obtenida del producto mínimo viable sirve como guía para priorizar y realizar ajustes en función de las necesidades cambiantes del

usuario. Esta retroalimentación continua facilita la adaptación del producto a medida que se descubren nuevas perspectivas y requerimientos.

La flexibilidad inherente de la metodología ágil permite responder de manera efectiva a los cambios en los requisitos del usuario, ya que el proceso de desarrollo está diseñado para incorporar modificaciones de manera ágil y eficiente. La iteración continua del desarrollo basada en la retroalimentación del usuario permite reducir el riesgo de fracaso al validar constantemente las decisiones de diseño y funcionalidad del producto.

Al adoptar un enfoque ágil, se fomenta una cultura de colaboración y comunicación constante entre los miembros del equipo de desarrollo y los usuarios. Esta interacción directa ayuda a garantizar que el producto final satisfaga las necesidades y expectativas del usuario de manera óptima.

En resumen, la aplicación de una metodología ágil en el desarrollo de la aplicación para redes de colaboración social ofrece una solución efectiva para mitigar los desafíos asociados con la incertidumbre de los requisitos y la dinámica cambiante del mercado. Esta aproximación permite adaptarse de manera proactiva a las necesidades del usuario y maximizar el valor del producto entregado.

3.2 EXPERIENCIA DE USUARIO UX

La experiencia de usuario se define como la percepción y respuesta que se genera de utilizar o anticipar el uso de un producto, sistema o servicio. Esta experiencia se divide en dos aspectos, por un lado, la generación de ideas y planificación de los factores que influyen en la experiencia; y por otro lado, las emociones, preferencias, comportamientos, confort, confianza, antes y después de usar el producto [6].

La experiencia de usuario en las aplicaciones web o móviles va de la mano con las metodologías ágiles puesto que permite conocer en cada producto mínimo viable, en cada iteración, o en cada modificación, la satisfacción del usuario, de forma que se pueda avanzar en forma rápida hacia el producto final deseado, sin que esto

conleve una gran expectativa para el usuario, sino al contrario, este conozca y apruebe la aplicación que va a utilizar y se sienta motivado y satisfecho de usarla.

3.3 VALOR PARA EL NEGOCIO

Es muy importante identificar el valor para el negocio, y centrar los esfuerzos en proporcionar este valor a través del sistema desarrollado, tomando en cuenta que existen entornos tan turbulentos, que este valor cambia con el tiempo. Es una habilidad del director de proyecto consolidar las “preocupaciones” de los diferentes stakeholders para sintetizar la información recopilada y concentrarse en lo que genera valor.

Con esta idea en mente, se recopilan las historias de usuario, se priorizan, se organizan las tareas, y se desarrolla un prototipo de la aplicación, el mismo que es presentado al usuario en forma inmediata para recibir su correspondiente retroalimentación.

3.4 METODOLOGÍAS ÁGILES

Las metodologías ágiles vienen a resolver el problema del cambio de requisitos que se presenta en forma periódica a lo largo del proyecto, en lugar de intentar definir todos los requisitos al inicio del mismo.

La metodología incluye las formas o procesos para gestionar los cambios en los requisitos, pero es muy importante que exista una buena comunicación entre los integrantes del equipo de trabajo y con los usuarios del sistema, para recibir de ellos una retroalimentación temprana y periódica, que reduzca el desperdicio de tiempo y recursos. Una de las metodologías ágiles más usadas hoy en día es Lean Startup.

3.5 METODOLOGÍA LEAN STARTUP

Lean Startup aplica las mejores prácticas y enfoques ágiles; busca un modelo de negocio viable, sostenible y repetible para crear productos que los clientes usarán

y pagarán; valida el tipo de negocio y las hipótesis de crecimiento, entre otras hipótesis, a través del ciclo de retroalimentación construir (el producto) - medir (la respuesta de los consumidores) - aprender (para la siguiente iteración) [7].

Normalmente, Lean Startup se utiliza como herramienta para ampliar la base de clientes. Es así que este modelo tiene cuatro etapas: identificación del cliente, verificación del consumidor/modelo de negocio, ampliación de la base de clientes y creación de la empresa. La principal diferencia entre este modelo y el tradicional de desarrollo de producto es que cada una de las etapas es iterativa [8]. El método implica el desarrollo rápido de un producto viable, su prueba permite considerar todos los deseos y comentarios de los consumidores. Adicionalmente, Lean considera un desperdicio de recursos a cualquier esfuerzo realizado que no genere valor [9].

Esta es la metodología que se utilizará en el desarrollo de la aplicación del presente proyecto, en vista de que no se conoce en forma concreta la respuesta de los usuarios al producto desarrollado, y estos encuentran que es mucho más fácil proveer una retroalimentación adecuada basada en la manipulación. Por lo tanto, se diseñará un MVP que pueda ser construido de la manera más rápida posible, para que además de obtener una rápida retroalimentación para realizar un producto que genere valor, se ajuste al poco tiempo que se dispone para desarrollar la aplicación. [10].

3.6 ETAPAS DE LEAN STARTUP

Construir

Crear un prototipo

Elaborar un producto mínimo viable (MVP) con funciones básicas para conocer la reacción de los clientes.

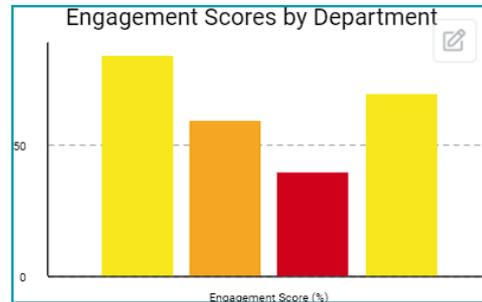


Imagen 1: Etapa Startup Construir
Tomado de-(Contreras-Almeida-2024)

Medir

Obtener retroalimentación

Recopilar información de consumidores, stakeholders y otros involucrados.

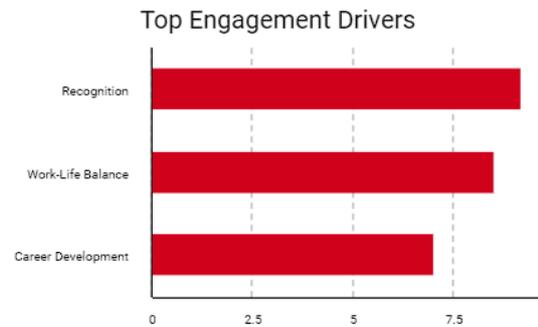


Imagen 2: Etapa Startup Medir
Tomado de-(Contreras-Almeida-2024)

Aprender

Mejorar el producto

Analizar los resultados para implementar nuevas mejoras en el producto inicial.

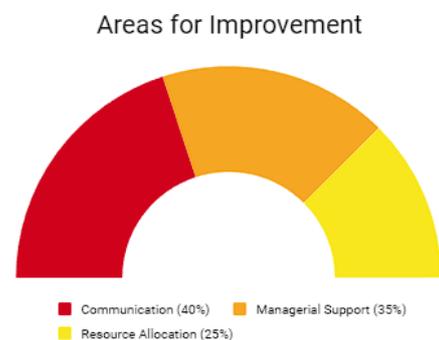


Imagen 3: Etapa Startup Aprender
Tomado de-(Contreras-Almeida-2024)

3.7 PASOS DE LEAN STARTUP



1 Idea

Genera una idea de proyecto innovadora que satisfaga una necesidad del mercado.

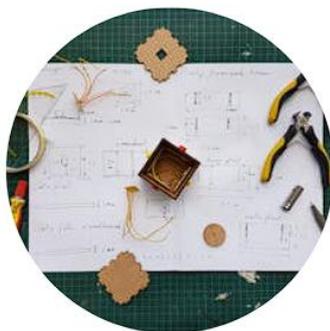
*Imagen 4: Pasos Startup Idea
Tomado de-(Contreras-Almeida-2024)*

2 Hipótesis

Formula una hipótesis sobre cómo tu aplicación móvil abordará el problema o necesidad.



*Imagen 5: Pasos Startup Hipótesis
Tomado de-(Contreras-Almeida-2024)*



3 Prototipo

Desarrolla una versión simplificada de la aplicación que tenga solo las características esenciales para probar tu hipótesis. Esto debe ser lo más rápido y económico posible.

*Imagen 6: Pasos Startup Prototipo
Tomado de-(Contreras-Almeida-2024)*

4 Validación

Recoge feedback de los clientes, valida si tu idea tiene potencial y adapta la aplicación en consecuencia



*Imagen 7: Pasos Startup Validación
Tomado de-(Contreras-Almeida-2024)*



5 Iteración

Mejora tu prototipo y repite el proceso hasta alcanzar el éxito.

*Imagen 8: Pasos Startup Iteración
Tomado de-(Contreras-Almeida-2024)*

3.8 PRINCIPIOS SOLID

Se los considera los cinco principios básicos en el diseño y la programación orientada a objetos. La intención es aplicar estos principios en conjunto para que sea más probable obtener un software fácil de mantener y extender en el tiempo.

S -> Single Responsibility Principle -> Un objeto debería tener una única responsabilidad.

O -> Open/Closed Principle -> Las entidades de software deberían estar abiertas para extensión pero cerradas para modificación.

L -> Liskov Substitution Principle -> Un objeto en un programa podría ser reemplazado con instancias de sus subtipos sin alterar la correctitud del programa.

I -> Interface Segregation Principle -> Muchas interfaces específicas son mejores que interfaces de propósitos generales.

D -> Dependency Inversion Principle -> Deberíamos depender de las abstracciones y no de las concretizaciones.

No en todos los productos de software se pueden aplicar estos principios, posiblemente el de responsabilidad única es uno que siempre se debe tomar en cuenta, para reducir la complejidad.

3.9 CÓDIGO LIMPIO

Código limpio es un código fácil de leer, entender y mantener. Su lógica debe ser directa para prevenir errores ocultos, mantener las dependencias al mínimo para simplificar el mantenimiento, implementar un procesamiento de errores exhaustivo conforme a una estrategia bien definida, y garantizar un rendimiento óptimo para evitar que los usuarios introduzcan optimizaciones innecesarias que puedan perjudicar el código.

Lo mejor es utilizar código limpio desde la primera línea de código, de manera que crezca en forma ordenada y sistemática, para que no solo el mantenimiento sino el desarrollo sea fácil de realizar. Si bien los principios, técnicas y métodos utilizados para escribir código limpio son muy útiles para dar mantenimiento al código realizado por otros desarrolladores, en la aplicación del presente proyecto se va a utilizar código limpio desde el inicio, para que sea fácilmente entendible y extensible.

4 MATERIALES Y METODOLOGÍA

4.1 MATERIALES

En la actualidad existen infinidad de herramientas que permiten pasar por las etapas de desarrollo en forma rápida y óptima. Desde herramientas para conocer la experiencia del usuario al utilizar la aplicación, hasta herramientas para su despliegue. A continuación, se listan las herramientas que se han utilizado en el transcurso de este proyecto.

Trello

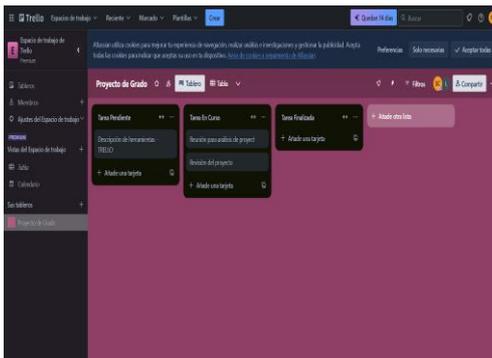


Imagen 9: Herramienta Trello
Tomado de (Atlassian, 2017))

Esta es una herramienta de gestión de proyecto. Sirve para manejar la metodología de Lean Start Up a través de tableros Kanban para gestionar las tareas de manera visual colaborativa. Se puede organizar un tablero por estados y asignar las tareas a los miembros del trabajo.

Figma

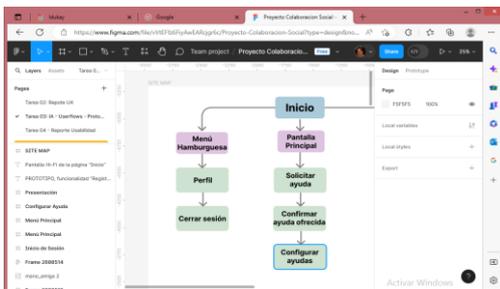


Imagen 10: Herramienta Figma
Tomado de (Adobe, 2022))

Es una herramienta muy potente que permite crear prototipos con presentaciones en tiempo real. También permite toda clase de diseños, importando modelos ya existentes de mapas de empatía, de

user-persona, journeying maps, entre otros.

Permite que varios diseñadores trabajen de forma simultánea en un mismo proyecto, incluye herramientas para crear prototipos interactivos lo que facilita la visualización, ofrece la capacidad de crear y utilizar componentes reutilizables.

Maze

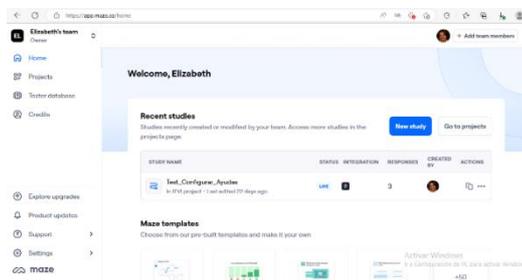


Imagen 11: Herramienta Maze
Tomado de ((Vega, 2018))

Maze es una herramienta para realizar pruebas de usabilidad de aplicaciones, que permite validar ideas rápidamente y recoger datos de las experiencias de usuarios de prueba o reales. Ofrece un enfoque ágil y eficiente para iterar diseños y mejorar la experiencia del usuario mediante pruebas remotas y automáticas.

Structurizr

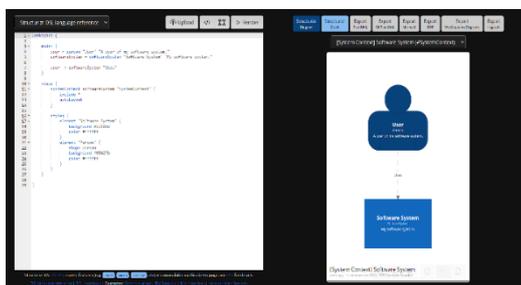


Imagen 12: Herramienta Structurizr
Tomado de ((Tune, 2013))

Es una herramienta que permite visualizar documentos de arquitectura de software. Con esta herramienta se pueden realizar los diagramas C4 de componentes, contenedores y clases.

Este modelo se destaca por el uso de descripciones claras en todos los elementos que se incluyen en los diagramas, lo que elimina la ambigüedad que

existe, por ejemplo, en los diagramas UML. La notación de los diagramas C4 no se restringe a un conjunto de reglas de estandarización, sino que se centra en separar al sistema en los cuatro niveles mencionados y en la creación de diagramas comprensibles, en los que no existen restricciones para adaptar cualquier elemento al dominio de la aplicación [11].

Draw.io

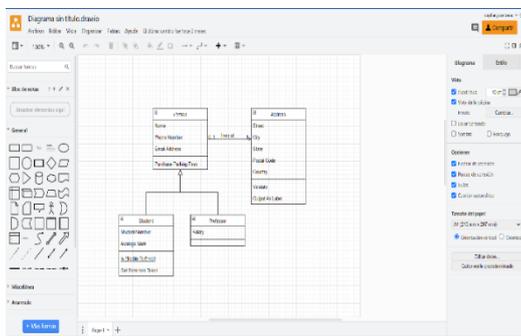


Imagen 13: Herramienta Draw.io
Tomado de (Google LLC, 2024)

Es una aplicación donde se pueden crear diagramas y representaciones visuales; además esta plataforma sirve para poder desarrollar los diseños de las bases de datos, los diagramas de flujo y la estructura que va a tener la aplicación.

Android Studio

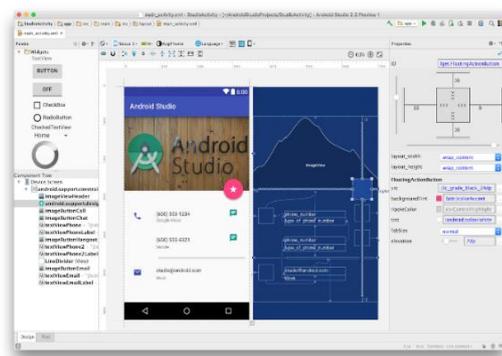


Imagen 14: Herramienta Android Studio
Tomado de (Google, 2013)

Es un entorno de desarrollo que proporciona una interfaz de usuario intuitiva y bien organizada, incluye editor de código autocompletado, refactorización y navegación eficiente.

Este entorno de desarrollo permite dos lenguajes de programación Java y Kotlin. Utiliza un sistema de construcción Gradle para su compilación.

Postman

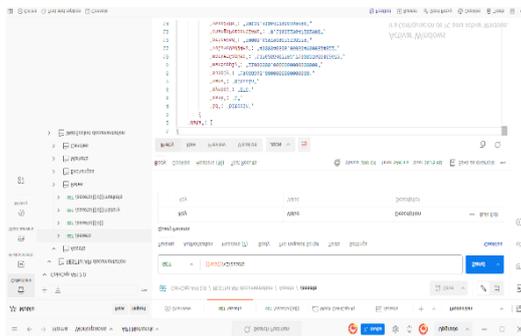


Imagen 15: Herramienta Postman
Tomado de (Abhinav Asthana, 2012)

Es una herramienta para probar, desarrollar y documentar APIs (Interfaz de Programación de Aplicaciones). Es una plataforma completa que permite a los desarrolladores enviar solicitudes HTTP a un servidor y recibir respuestas. Permite organizar solicitudes relacionadas en colecciones, definir entornos como desarrollo, pruebas o producción, crear scripts de pruebas automatizadas, entre otros.

Firebase



Imagen 16: Consola de Firebase del Proyecto Red de Colaboración
Tomado de (Google LLC, 2024)

Es una plataforma en la nube para el desarrollo de aplicaciones web y móvil. Está disponible para distintas plataformas (iOS, Android y web), con lo que es más rápido trabajar en el desarrollo. Dispone de varias herramientas, las que se utilizan en el presente proyecto son:

- **Authentication:** facilita el trabajo de autenticación del usuario al tiempo que le permite interactuar con APIs de Google en forma más nativa.
- **Firestore Database:** para almacenar la base de datos, la cual es documental.

- Storage: para almacenar las imágenes.
- App Check: para la seguridad de la aplicación.

GitHub

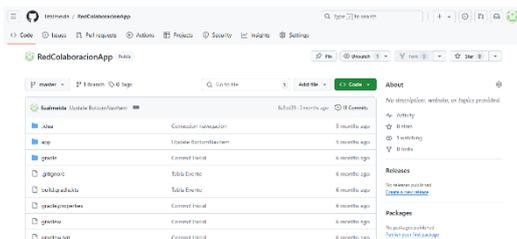


Imagen 17: Repositorio del Proyecto en GitHub
Tomado de (Microsoft Corp., 2024)

Es una plataforma de trabajo colaborativo basado en la nube y sirve para el control de versiones Git.

Permite a los desarrolladores realizar un seguimiento de los cambios de manera efectiva y gestionar versiones de proyectos de software.

Sus repositorios son espacios donde se almacena y se organiza el código fuente y pueden ser públicos o privados.

Facilita el desarrollo de múltiples desarrolladores dentro de un mismo proyecto y cada repositorio cuenta con su propia URL.

4.2 METODOLOGÍA LEAN STARTUP

En el marco teórico se realizó una introducción a la metodología Lean Startup. Esta va a ser aplicada en el desarrollo del proyecto, el cual inicia con una primera etapa de “Construir”, en la que se elabora una hipótesis tomada de la información recopilada de entrevistas, y reacciones de los usuarios a un primer diseño en borrador, y luego a un prototipo desarrollado en Figma. Se procede a la etapa de “Medir”, colocando el prototipo en Maze para obtener la retroalimentación de los usuarios. Con esta información, se pasa a la etapa de “Aprender” para definir los cambios que deben efectuarse en la siguiente iteración.

La metodología se pone en práctica mediante cinco pasos, que van desde la identificación de la idea hasta la definición de los cambios necesarios para la siguiente iteración.

4.3 IDENTIFICAR LA IDEA

La aplicación propuesta permitiría primero contar con una base de datos de todos los miembros y de los tipos de ayudas que algunos integrantes de la organización pueden ofrecer, como asistencia médica, ayuda económica, en víveres, entre otros. Luego, los usuarios solicitan ayuda en una de estas categorías, y la aplicación conecta al que ofrece la ayuda con el que la necesita para efectivizarla mediante mensajes o alertas. Finalmente, una vez que la ayuda se ha realizado, el usuario que la recibe cierra la ayuda con algún comentario para retroalimentar al proceso.

4.4 CREAR UNA HIPÓTESIS

De acuerdo a las entrevistas realizadas, se formula la siguiente hipótesis: "Una aplicación que empareje la necesidad de una persona con otra persona que la puede resolver, ayudará a los integrantes de una comunidad a solucionar el problema de canalizar de forma efectiva las donaciones u ofrendas".

La mejor y más ágil forma de presentar al usuario una primera aproximación de lo que la aplicación va a realizar es a través de un prototipo. Para realizar este prototipo es necesario pasar por las etapas de comunicación interactiva con el usuario para recoger la experiencia del usuario desde un principio e identificar sus expectativas y puntos de dolor.

Por lo tanto, se va a esquematizar la información recopilada a través de las herramientas disponibles en el mercado. En este caso se va a utilizar FIGMA para realizar un informe de la experiencia de usuario, el mapa del sitio, y finalmente un prototipo.

4.4.1 INFORME DE LA EXPERIENCIA DE USUARIO

La principal necesidad del usuario es poder registrar su solicitud de ayuda, que en ocasiones es urgente, para que una persona que esté dispuesta a colaborar con la asistencia le responda, y que efectivamente se materialice esta ayuda.

En cuanto a la usabilidad, la aplicación debe ser muy fácil e intuitiva de aprender, con pocas y opciones concretas, sobre todo para el público de mayor edad.

En el Anexo A se detallan algunas herramientas utilizadas para recoger la experiencia del usuario con respecto a la aplicación, como un card sorting, mapas de empatía, user persona y journey maps.

4.4.2 REQUERIMIENTOS FUNCIONALES

La IEVI Norte necesita un sistema que permita registrar a los miembros de la comunidad, que constituyen el principal usuario de la aplicación móvil. Este usuario es el que podrá registrar las solicitudes de ayuda y será el mismo usuario que también podrá ofrecer ayuda a quien la solicite. El usuario administrador utilizará la aplicación web para gestionar los eventos, monitorear la actividad del sistema, y habilitar o deshabilitar a los miembros de la iglesia. La aplicación web no se desarrollará en el presente proyecto.

El ciclo desde que el usuario solicita la ayuda hasta que esta se efectiviza, es el siguiente:

- El sistema registra las solicitudes de ayuda, especificando el tipo como oración, víveres, medicinas, asistencia en casa, acompañamiento, entre otros. Los usuarios deben ser clasificados por grupos, y los tipos de ayuda deben estar visibles según los grupos a los que pertenece cada persona.
- Cada miembro de la comunidad tiene la opción de registrar un tipo de ayuda que puede brindar a los demás, de forma que cuando alguna persona requiere la asistencia, el o los usuarios que ofrecen ese tipo de ayuda reciban una notificación.

- Cuando un usuario envía una solicitud de ayuda, esta se desplegará a las personas que configuraron ese tipo de ayuda, por ejemplo, de víveres, al tiempo que se les envía una notificación automáticamente.
- Las personas que aceptan brindar la ayuda pueden mirar los datos de la persona solicitante, su contacto y su dirección, para poder concretar la asistencia, y son visibles dependiendo del tipo de ayuda. Existe un máximo de personas que pueden brindar ayuda a una misma solicitud, según el tipo.
- Finalmente, cuando la ayuda se efectiviza, el usuario solicitante confirma la recepción de la asistencia, puede ingresar un comentario, y se cierra la solicitud. En caso de que no se efectivice, puede registrar su solicitud como “no efectivizada”, ingresar un comentario, y se cierra la solicitud.

Por tanto, el usuario podrá al final del proyecto:

- Registrar sus datos personales incluida la dirección y teléfono, junto con las categorías en las que quiere prestar ayuda.

La dirección y el teléfono son necesarios para que la persona que va a brindar la ayuda contacte a la persona que la necesita, y conozca dónde se encuentra para confirmar o no la prestación. Posteriormente podría mejorarse la aplicación incluyendo la ubicación exacta para que canalice la ayuda a la o las personas más cercanas.

- Registrar la solicitud de ayuda o asistencia en una categoría, y visualizar el historial de ayudas solicitadas.

Para registrar una solicitud de ayuda, el usuario debe escoger la categoría en la que desea la ayuda, como por ejemplo víveres, explicar su situación, y escoger la prioridad con la que la necesita.

- El historial permitirá visualizar la lista de todas las ayudas solicitadas, en curso, o finalizadas, incluida la solicitud que acaba de realizar.

- Responder solicitudes de ayuda confirmando que se va a realizar la asistencia, y visualizar el historial de ayudas realizadas.

Al responder el usuario debe escribir un comentario y la fecha y hora a la que va a efectivizar la ayuda. Esta información es notificada a la persona que solicita la ayuda. El historial permitirá visualizar la lista de todas las ayudas respondidas que estén en curso, o finalizadas, incluida la solicitud que acaba de responder.

- Registrar cuando una asistencia se ha efectivizado o no, y el motivo. En el historial se debe incluir una columna que muestre el estado en el que se encuentra una solicitud. Los estados serían los siguientes:
 - o Solicitada: cuando el usuario registra la solicitud de ayuda.
 - o En curso: cuando el usuario que desea, responde a la solicitud confirmando la fecha y hora.
 - o Finalizada: cuando en forma manual el usuario que recibe la ayuda señala que esta fue efectivizada o no lo fue, junto con un comentario, con lo que la ayuda se marca en verde o en rojo respectivamente.

4.4.3 REQUERIMIENTOS NO FUNCIONALES

Los requerimientos no funcionales de una aplicación móvil proporcionan criterios de calidad y restricciones que afectan su rendimiento, seguridad y usabilidad. Incluyen aspectos como la velocidad de respuesta, compatibilidad con dispositivos, y accesibilidad; la escalabilidad y mantenibilidad que garantizan el crecimiento y actualización sin problemas; y las medidas de seguridad, como autenticación y protección de datos. Estos requerimientos son clave para una óptima experiencia de usuario y el éxito del desarrollo.

- Compatibilidad con Android, para que la aplicación llegue a la mayor cantidad de usuarios posibles, en vista de que este sistema operativo es el más ampliamente utilizado en dispositivos móviles. El lenguaje que se utilizará es Kotlin para Android.

La compatibilidad con IOS con el uso de Swift como lenguaje, quedará como trabajo futuro al presente proyecto.

- Usabilidad, esto es, que las interfaces sean sencillas de utilizar, intuitivas, motivadoras, incluso para personas que no están familiarizadas con la tecnología, puesto que hay también personas de la tercera edad que forman parte de la comunidad. Justamente para lograr cumplir con este requerimiento, se cuenta con todas las herramientas de experiencia de usuario ya recopiladas.
- Tiempo de respuesta rápido, para que el usuario pueda tener una experiencia agradable, y que la aplicación genere confianza en su uso, fomentando la participación y uso continuo. Se utilizará la herramienta Firebase Performance, que captura métricas relacionadas con el rendimiento, incluyendo tiempos de respuesta de la UI y el tiempo que tarda la aplicación en completar ciertas tareas importantes.
- Consumo de recursos, limitando la cantidad de información que se envía o se recibe, y el tamaño de las imágenes. De esta forma se garantizará un funcionamiento confiable incluso en momentos de alta demanda, o cuando el internet es limitado. Se va a tener especial precaución con este requerimiento, puesto que puede marcar la diferencia entre que un usuario siga utilizando o no la aplicación.
- Seguridad, para que los datos personales sean visibles solo para los usuarios que lo requieren, al establecer el contacto entre el solicitante y el dador de la ayuda. Esto se va a realizar, por un lado, utilizando los servicios de la nube que ofrecen seguridad en la información, y por otro lado, en la programación, utilizando los estándares OWASP para aplicaciones móviles. La plataforma en la nube de Firebase se encargará de la creación y autenticación de los usuarios, validando las cuentas de correo con las que los usuarios ingresarán al sistema, así como la regla de generación de las contraseñas.

4.4.4 DIAGRAMAS C4

El modelo C4 está diseñado para representar una aplicación de software con un conjunto de diagramas que describen los componentes e interacciones entre ellos, con el propósito de ayudar a los equipos a comprender la arquitectura de la aplicación a través de diferentes niveles de abstracción representados en cada diagrama.

En el mundo del desarrollo de software, el modelo C4 es una herramienta esencial para la comunicación entre los interesados, desarrolladores y otros miembros del equipo, pues proporciona un lenguaje común para describir un sistema de software, lo que causa un entendimiento compartido que es crítico para un resultado exitoso del proyecto.

Existen muchas aplicaciones para graficar los diagramas C4, pero para este proyecto se escogió Structurizr en vista que proporciona una gran facilidad para la generación de los diagramas a través de un código fuente de fácil desarrollo y comprensión. El resultado fueron los siguientes diagramas de “Contexto del Sistema”, de “Contenedores”, y de “Componentes”, que es el paso previo al desarrollo del código fuente de la aplicación.

Diagrama C4 de Contexto

La iglesia IEVI necesita canalizar las ayudas que requieren varios miembros de la iglesia a las personas que pueden ofrecerla, así como comunicar a toda la comunidad sobre las reuniones semanales de propósito específico. Solo existen dos tipos de usuarios, los miembros de la iglesia que utilizan el sistema y el administrador que controla los eventos y monitorea la actividad. Todo el backend lo maneja la plataforma online Firebase.

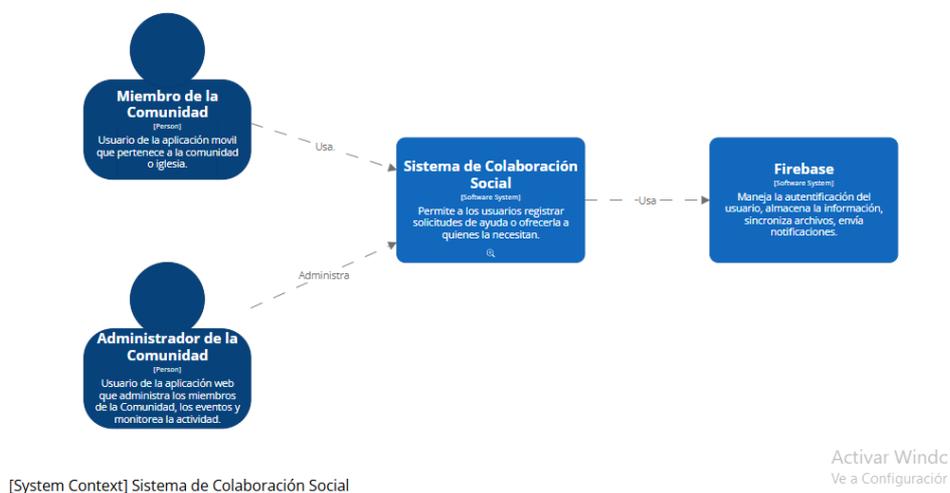
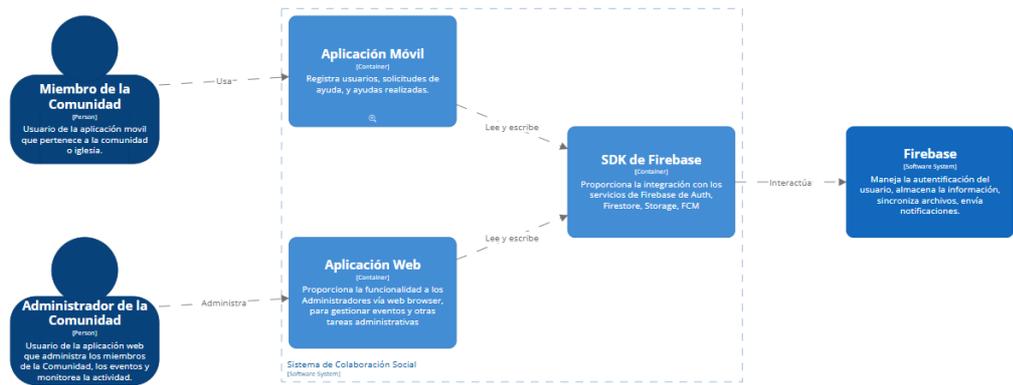


Imagen 18: Diagrama C4 de Contexto de la Aplicación Red de Colaboración Tomado de-(Contreras-Almeida-2024)

Diagrama C4 de Contenedores

El Sistema de Colaboración Social se utilizaría en dos plataformas dependiendo del tipo de usuario, el usuario miembro de la comunidad desde el dispositivo móvil, y el usuario administrador desde la página web. En el alcance definido al inicio del presente proyecto, se especificó que se va a realizar el diseño solo de la aplicación móvil para el usuario miembro de la comunidad. En cualquiera de los dos casos, se debe utilizar un SDK del Firebase para la integración entre la aplicación y las diferentes funcionalidades del Firebase, como la autenticación, base de datos documental, y almacenamiento de archivos.



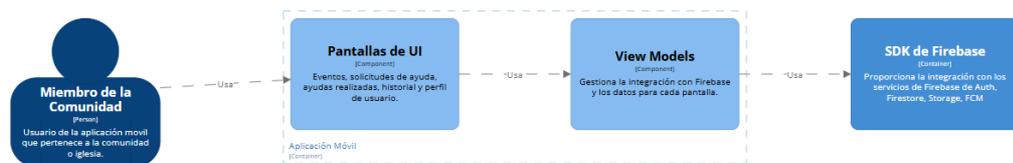
[Container] Sistema de Colaboración Social

Activar Windows
Ve a Configuración para activar Window

Imagen 19: Diagrama C4 de Contenedores de la Aplicación Red de Colaboración Tomado de-(Contreras-Almeida-2024)

Diagrama C4 de Componentes

Dentro de la aplicación móvil, la estructura se descompone en los View Models y las pantallas de UI. En los view models se realiza la conexión al Firebase y el envío y recepción de datos desde y hacia la base de datos documental, utilizando el SDK de Firebase. Las pantallas de UI son las interfaces que el usuario utiliza para interactuar con la aplicación.

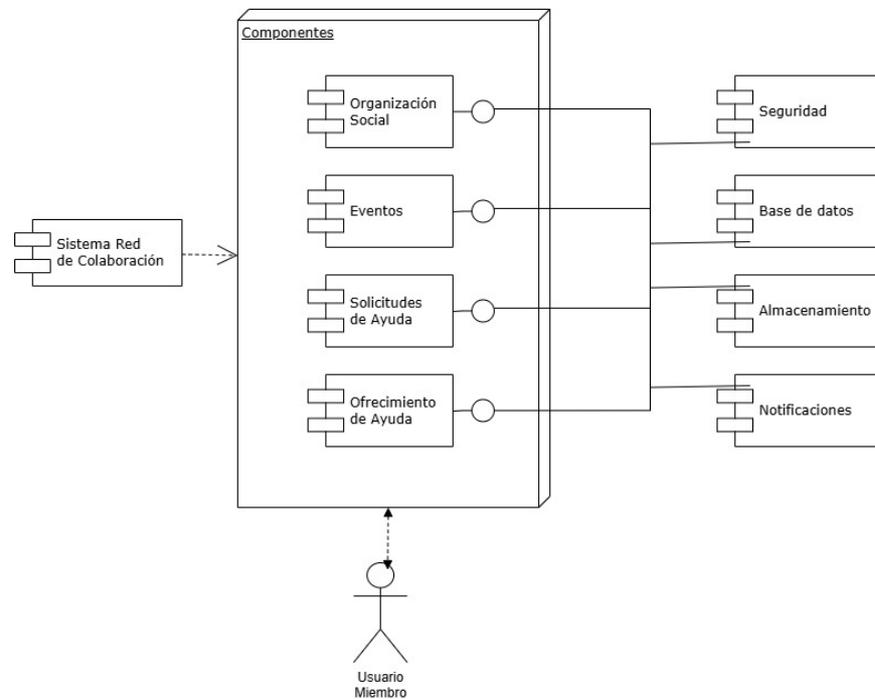


[Component] Sistema de Colaboración Social - Aplicación Móvil

Imagen 20: Diagrama C4 de Componentes de la Aplicación Red de Colaboración Tomado de-(Contreras-Almeida-2024)

4.4.5 ARQUITECTURA

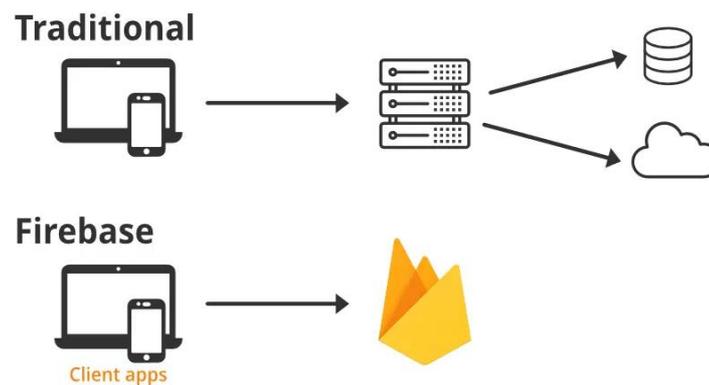
Se complementan los diagramas C4 con el diagrama UML de componentes para representar las relaciones entre los componentes del sistema.



*Imagen 21: Diagrama UML de Componentes de Arquitectura
Tomado de (Arvind Padmanabhan, 2021)*

El usuario miembro de la organización social, puede visualizar los eventos de la entidad y gestionar las solicitudes y ofrecimientos de ayuda. Estos componentes requieren de una base de datos, almacenamiento, notificaciones, así como de la autenticación y seguridades para evitar el abuso de intrusos.

Si se habla de utilizar metodologías ágiles, Firebase es una excelente opción, pues simplifica el trabajo de los desarrolladores en temas de seguridad, base de datos, almacenamiento, crecimiento, monetización y análisis de datos de las aplicaciones, lo cual les permite concentrarse en la experiencia del usuario, acelerar el ciclo de desarrollo y ofrecer productos de alta calidad de manera más rápida y eficiente.



*Imagen 22: Arquitectura con enfoque tradicional versus Firebase
Tomado de (Arvind Padmanabhan, 2021)*

La aplicación se desarrolla en Android Studio, una herramienta de desarrollo para móviles, en lenguaje Kotlin. Interactúa con Firebase, la plataforma en la nube mencionada, para desarrollo de aplicaciones Web y móviles, que simplifica su desarrollo.

El esquema de la Arquitectura es el siguiente:

1. Frontend (Aplicación móvil)
 - UI/UX: Kotlin
2. Autenticación: Firebase Authentication
 - Registro e inicio de sesión de usuarios.
 - Métodos soportados: email/password, OAuth (Google, Facebook), y anónimos.
 - Retorna un ID Token o JWT que identifica al usuario.
3. App Check: Verifica que las solicitudes provienen de una aplicación segura y aprobada (previene abuso automatizado).
 - Utiliza tokens de App Check hacia Firebase y recursos relacionados.
 - Compatible con Firebase Database, Firestore, Storage y más.
4. Base de Datos: Firestore Database
 - Estructura NoSQL (base de datos documental): Colecciones y documentos.
 - Almacena datos estructurados como perfiles de usuario, eventos, solicitudes de ayuda, ayudas realizadas.

- Reglas de seguridad: Permiten restringir el acceso según el estado de autenticación del usuario y los tokens de App Check.
5. Almacenamiento: Firebase Storage
- Puede guardar archivos multimedia (imágenes, videos, documentos, etc.), como las fotografías del perfil de los usuarios.
 - Reglas de seguridad: Controlan quién puede leer/escribir archivos con base en el UID del usuario o App Check.
6. Backend (Cloud Functions)
- Procesamiento adicional como envío de correos, notificaciones, o integración con servicios externos.
 - Puede dispararse en base a eventos de Firestore, Storage o autenticación.
7. Monitoreo y Rendimiento
- Firebase Performance: Captura métricas relacionadas con el rendimiento, incluyendo tiempos de respuesta de la UI.
 - Firebase Analytics: Para seguimiento de uso y comportamiento del usuario.

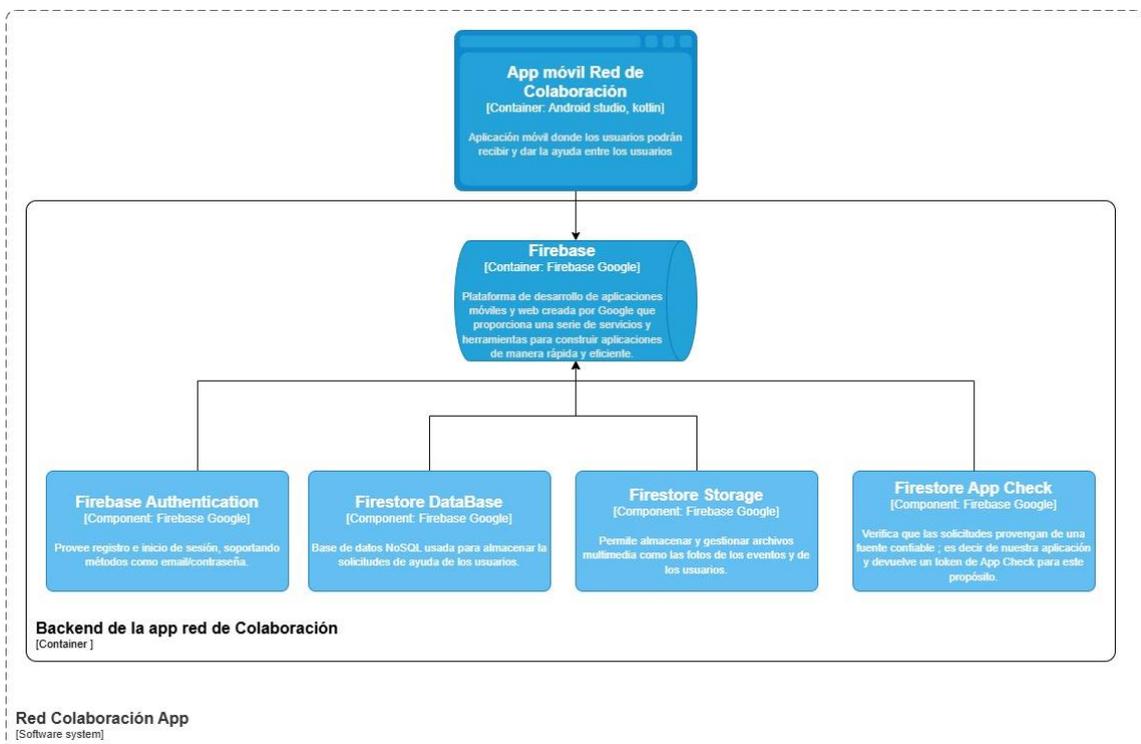


Imagen 23: Backend de la App Red de Colaboración
Tomado de-(Contreras-Almeida-2024)

4.4.6 MAPA DEL SITIO

En base a la recopilación de los requerimientos funcionales, se elabora un mapa de sitio, que es una representación estructurada del contenido de la aplicación móvil. El propósito principal es solicitar y ofrecer ayuda a los miembros de la comunidad, por lo que esto se refleja en la pantalla principal.

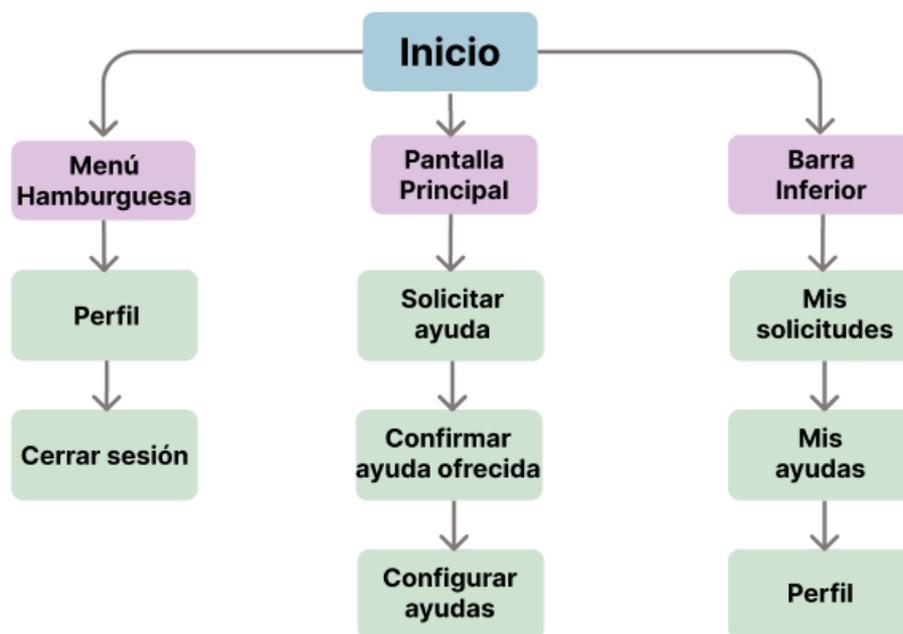


Imagen 24: Mapa del Sitio, primera iteración
Tomado de-(Contreras-Almeida-2024)

4.4.7 USER FLOWS

Los User Flows son diagramas o representaciones visuales que muestran los pasos que un usuario debe seguir para alcanzar un objetivo específico dentro de la aplicación. Describen el camino que toma un usuario, desde su entrada a la plataforma hasta la finalización de una tarea concreta.

Tarea: Registrar datos personales

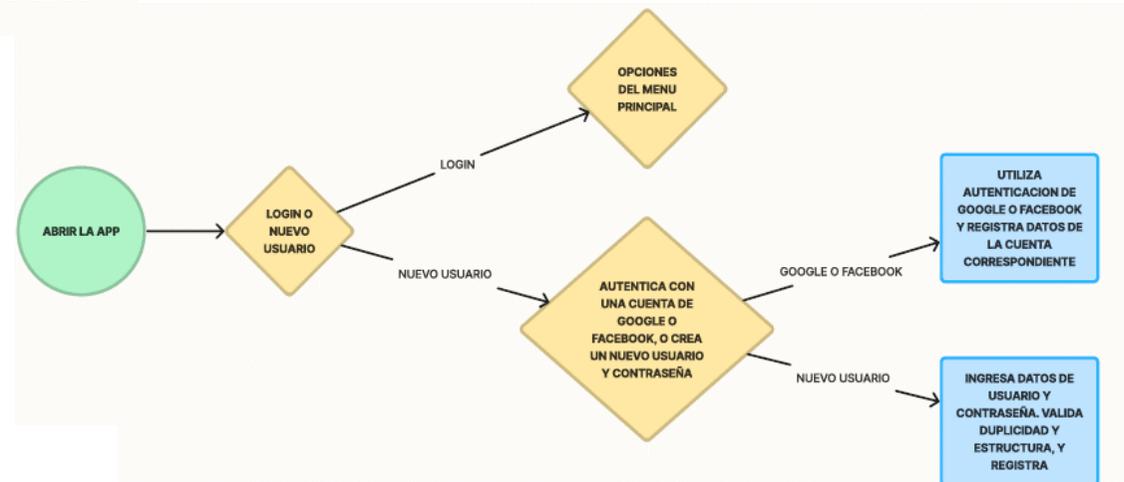


Imagen 25: User Flow para Registrar Datos Personales, primera iteración
Tomado de-(Contreras-Almeida-2024)

Tarea: Registrar la solicitud de ayuda o asistencia en una categoría

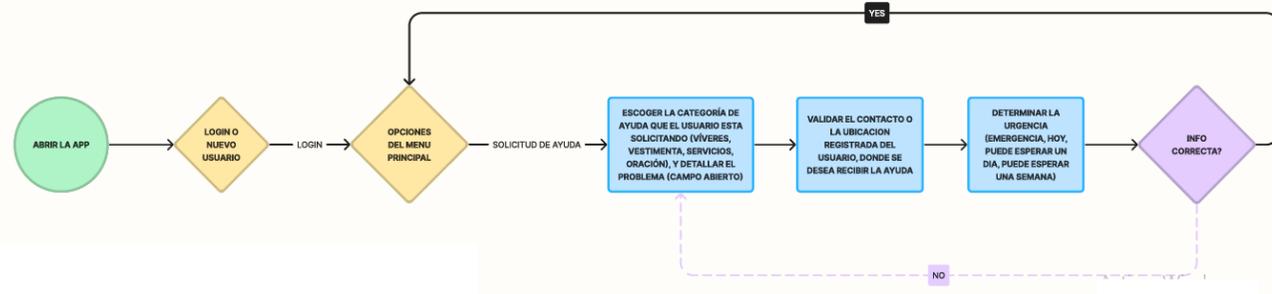


Imagen 26 User Flow para Registrar la Solicitud de Ayuda, primera iteración
Tomado de-(Contreras-Almeida-2024)

Tarea: Registrar cuando una asistencia ha sido efectuada o no, y el motivo

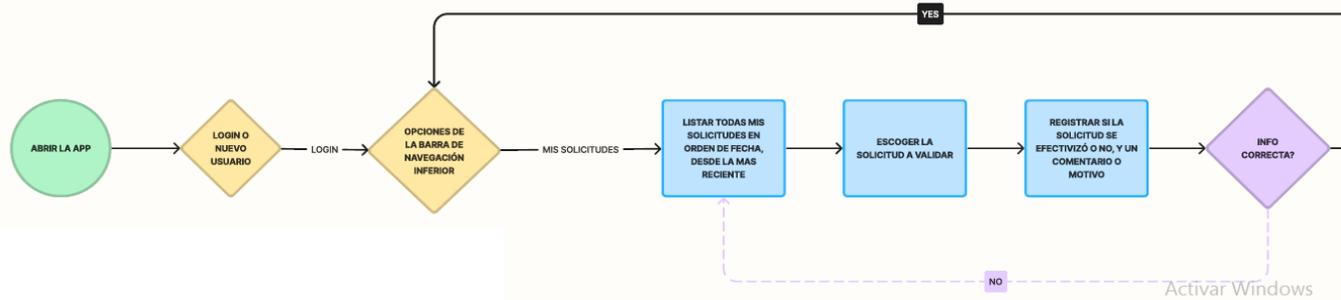


Imagen 27: User Flow para Registrar una Ayuda Efectivizada, primera iteración
Tomado de-(Contreras-Almeida-2024)

Tarea: Registro de las categorías para prestar ayuda

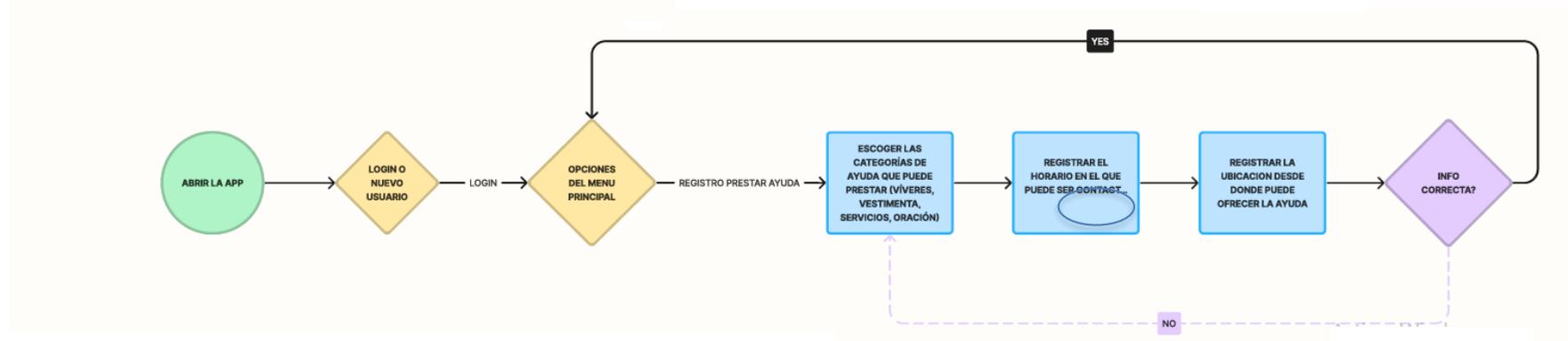


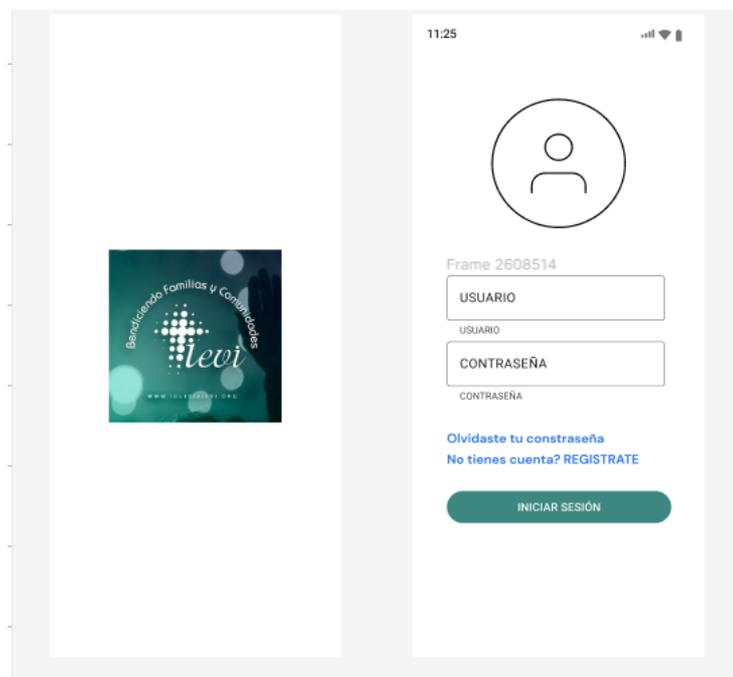
Imagen 28: User Flow para Registrar las Categorías de Ayuda, primera iteración
Tomado de-(Contreras-Almeida-2024)

4.5 PROTOTIPO

Con toda la información recopilada, se desarrolla un primer prototipo que va a ser evaluado por los miembros de la iglesia, así como una encuesta de usabilidad del prototipo.

4.5.1 PROTOTIPO INICIAL

Tomando como base toda la información recopilada, se elabora un primer prototipo en Figma, siguiendo lo establecido en el Mapa de Sitio y los User Flows.



*Imagen 29: Prototipo, Pantalla de Inicio de Sesión, primera iteración
Tomado de-(Contreras-Almeida-2024)*

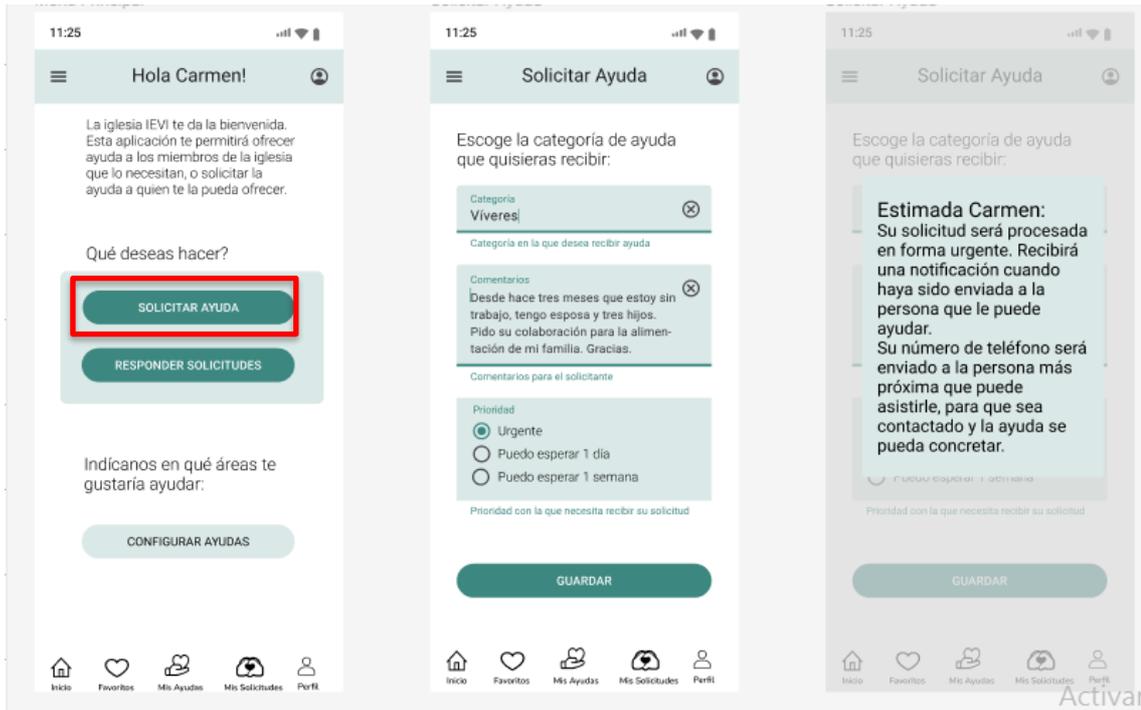


Imagen 30: Prototipo, Pantalla de Solicitud de Ayuda, primera iteración
Tomado de-(Contreras-Almeida-2024)

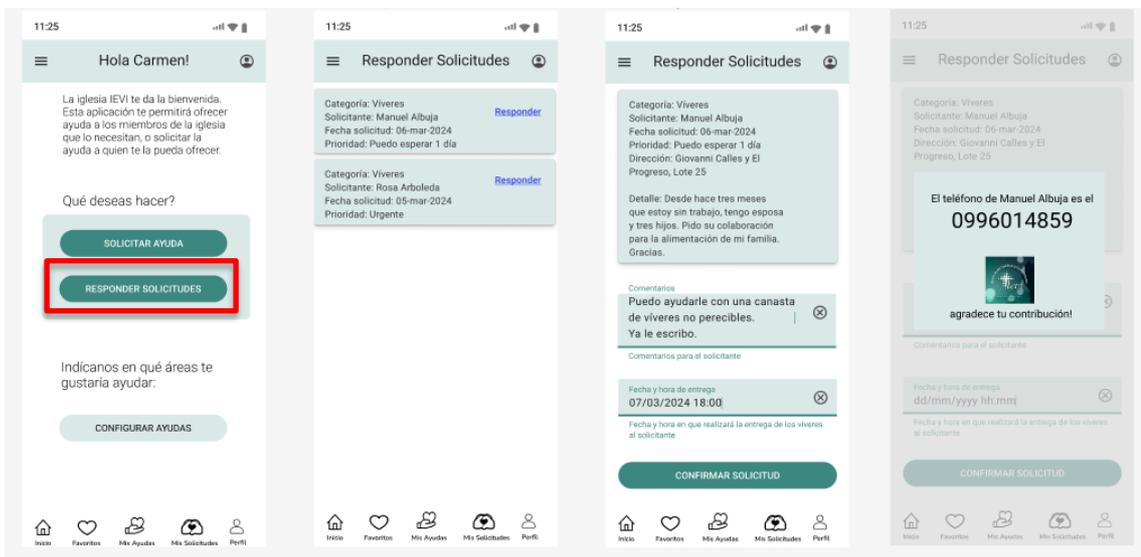


Imagen 31: Prototipo, Pantalla de Ofrecer Ayudas, primera iteración
Tomado de-(Contreras-Almeida-2024)

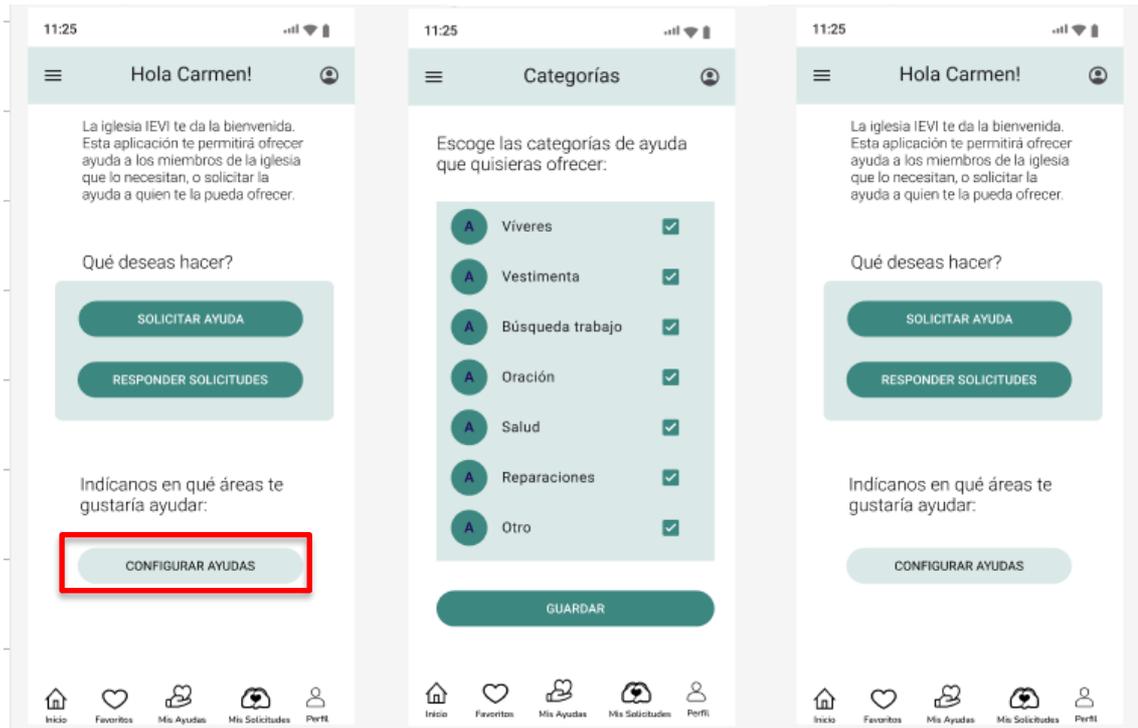


Imagen 32: Prototipo, Pantalla de Perfil y Categorías, primera iteración
Tomado de-(Contreras-Almeida-2024)

4.6 VALIDACIÓN

La validación es el proceso más importante a la hora de definir los cambios que se van a realizar en la aplicación, pues la retroalimentación del usuario es lo que delimita los ajustes en las diferentes pantallas, e incluso en la redefinición de requerimientos.

4.6.1 LANZAR EL PROTOTIPO

Para realizar una validación del prototipo, se elabora un test de usabilidad en MAZE, para lo cual se publica el prototipo y una encuesta, y se distribuye el enlace a doce usuarios de la IEVI Norte. Con el prototipo se pretende que el usuario pueda revisar las funcionalidades de la aplicación y recoger su comportamiento en las diferentes pantallas, y con la encuesta, recoger las impresiones, comentarios y sugerencias del prototipo.

Se escogieron como usuarios participantes para evaluar el proyecto y realizar la encuesta a un pastor, un siervo, dos colaboradores y ocho asistentes a la iglesia en calidad de miembros integrantes que asisten en forma regular, doce personas en total. Por el momento no fue posible observar sus expresiones o reacciones cara a cara, que hubiera sido lo ideal, para validar lo que está diseñado, e identificar los puntos de dolor más fácilmente.

Los criterios para la selección de los participantes de las pruebas fueron los siguientes:

- Representatividad de la muestra: todos los integrantes de la iglesia van a utilizar la aplicación con el mismo propósito, pero cada perfil puede aportar con un punto de vista diferente, por lo que es importante recopilar las impresiones de todos los tipos de usuarios.
- Perfil demográfico: De la misma forma, el criterio de personas con edades, género, y nivel educativo variados va a contribuir en la recopilación de las necesidades, objetivos y expectativas de los usuarios objetivo.
- Experiencia con tecnología: En una primera etapa, la muestra incluye usuarios que tengan familiaridad con aplicaciones en dispositivos móviles.

Las preguntas del cuestionario estuvieron enfocadas en validar las opciones de ayuda que se pueden ofrecer, en la combinación de colores, el nivel de comprensión de la aplicación, el nivel de utilidad, y sugerencias abiertas en el diseño.

El objetivo principal del test es valorar si los usuarios comprenden el funcionamiento de la aplicación y el beneficio que obtendría la organización y sus miembros. También permitirá validar lo que se ha derivado de las entrevistas y encuestas con los usuarios. Y finalmente, detectar los puntos de dolor, frustraciones e inconformidades de los usuarios frente al diseño de la aplicación.

Una vez recopiladas las respuestas tanto del prototipo como de la encuesta, se analiza la información para extraer las conclusiones que permitan mejorar la aplicación y, por tanto, la experiencia del usuario. Los resultados del test ofrecerán

una retroalimentación al diseño de la aplicación para realizar los ajustes o mejoras necesarios.

4.6.2 MEDIR Y RECOPIRAR DATOS

Se recopilan los datos sobre cómo los usuarios interactúan con la aplicación y los resultados se detallan en el Anexo B, los cuales se resumen en siguientes puntos:

- En las pantallas del rendimiento por pantalla y en los mapas de calor se puede observar que existe una deficiente interacción del usuario con el prototipo. Esto se debe a que el usuario quiere ingresar información en lugar de solo observar lo que se despliega; también escoge cualquier botón de la pantalla en lugar de ir hacia el botón que lleva a la siguiente pantalla; finalmente, puede quedarse en una pantalla por mucho tiempo por hacer otra actividad, con lo que el tiempo se incrementa sin un motivo justificado.
- La impresión general del prototipo es de aceptación, aunque en la mayoría de los casos, no pudieron completar el flujo. El usuario está acostumbrado a interactuar con aplicaciones y no con prototipos, por lo que se concluye que el usuario necesita un acompañamiento o guía personalizada para movilizarse en el prototipo.

Se recopilan las respuestas del cuestionario y los resultados se detallan en el Anexo C, los cuales se resumen en los siguientes puntos:

- El principal resultado obtenido es saber que la aplicación es aceptada, y de las sugerencias se extraen algunos cambios en el diseño que pueden mejorar la experiencia del usuario, como colocar los mensajes más concretos, o ponerle un nombre llamativo a la aplicación. Una vez realizados los cambios, ya se podría realizar la programación de un producto mínimo viable, para que el usuario ingrese y consulte información que le permita familiarizarse de mejor manera con las funcionalidades.
- El principal objetivo de este estudio fue mejorar la experiencia del usuario de acuerdo a su comportamiento y respuestas realizadas. Las herramientas utilizadas

fueron muy útiles y hacen una gran diferencia a la hora de vincular al usuario en el proceso de análisis y diseño de la aplicación.

4.7 ITERACIÓN

El proceso de aprender y adaptar define una nueva iteración. Se utilizan los datos recopilados para evaluar si la hipótesis es correcta, y si es necesario realizar cambios en las características, el diseño o el enfoque general.

De los resultados obtenidos en el punto anterior, se desprende que la hipótesis planteada es correcta, pero se requiere un cambio en el enfoque general de la aplicación: esta no solo servirá para canalizar las ayudas sino para cualquier otra funcionalidad futura, por lo que se incorporó una pantalla inicial con una lista de eventos de la iglesia, y solo cuando el usuario quiera pedir y ofrecer ayuda, debe autenticarse. También se deben realizar algunos ajustes pequeños al diseño del prototipo.

Con esto, se replantea y se adapta el mapa del sitio, los user flows y el prototipo.

4.7.1 MAPA DEL SITIO

En vista de que la Iglesia no cuenta con una aplicación móvil oficial, esta servirá no solo para canalizar las ayudas sino para cualquier otra funcionalidad futura. Por ello, se incorpora una pantalla inicial con una lista de eventos de la iglesia. Se debe realizar un ajuste adicional y es que el usuario debe autenticarse solo cuando quiera pedir y ofrecer ayuda.

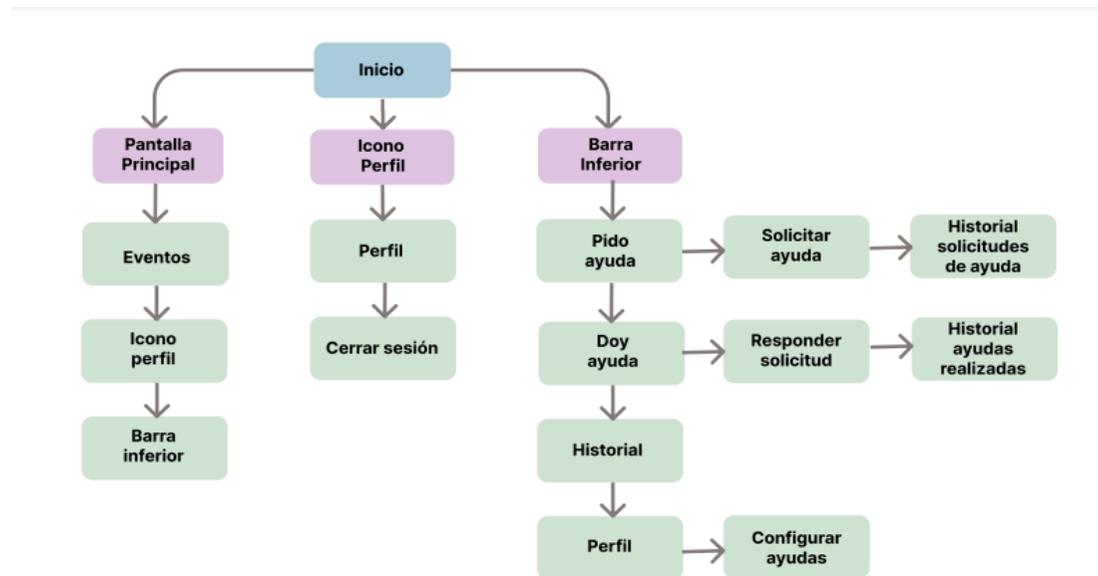


Imagen 33: Mapa del Sitio, segunda iteración
Tomado de-(Contreras-Almeida-2024)

4.7.2 USER FLOWS

En la segunda iteración, se realizar tres cambios. El primero es la inclusión de una pantalla de eventos que se despliega al ingresar a la aplicación que no necesita autenticación. El segundo cambio es la eliminación del flujo para registrar las categorías de ayuda que el usuario puede ofrecer, y se coloca en una pantalla contigua a la de “Datos Personales”. El último cambio es la inclusión de la pantalla de “Responder a la Solicitud de Ayuda”.

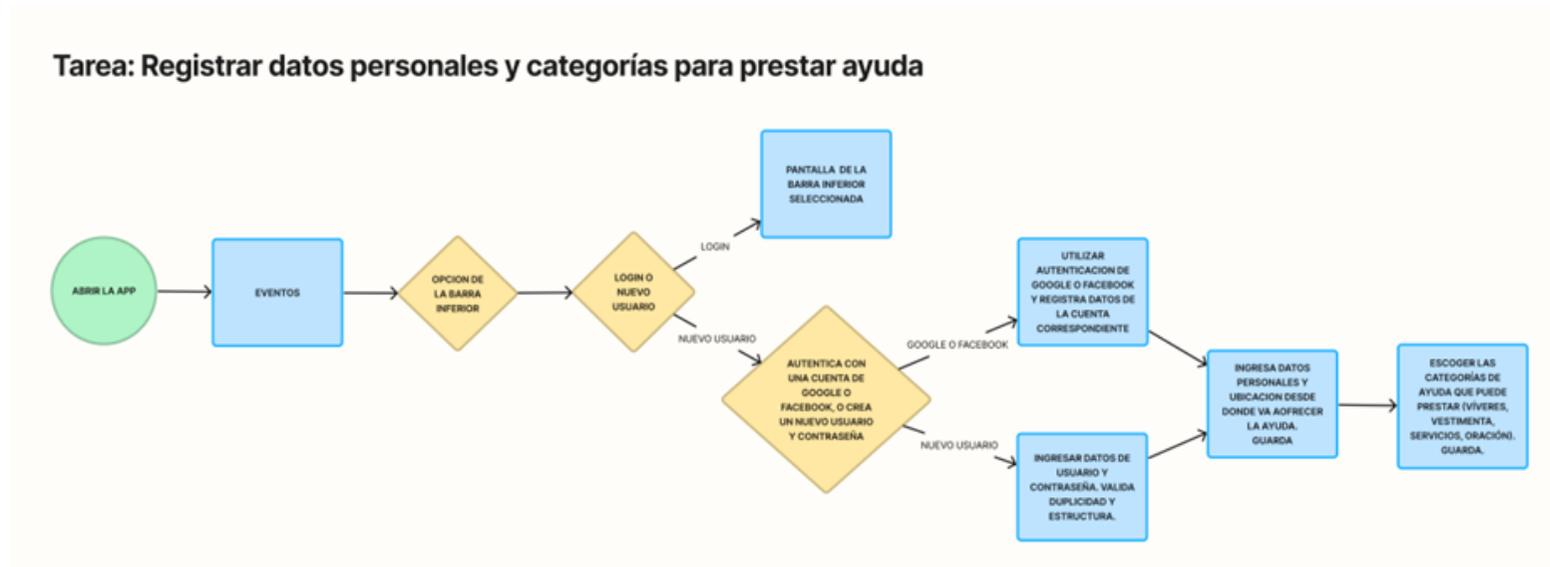


Imagen 34: User Flow para Registrar Datos Personales, segunda iteración
Tomado de-(Contreras-Almeida-2024)

Tarea: Registrar la solicitud de ayuda o asistencia en una categoría

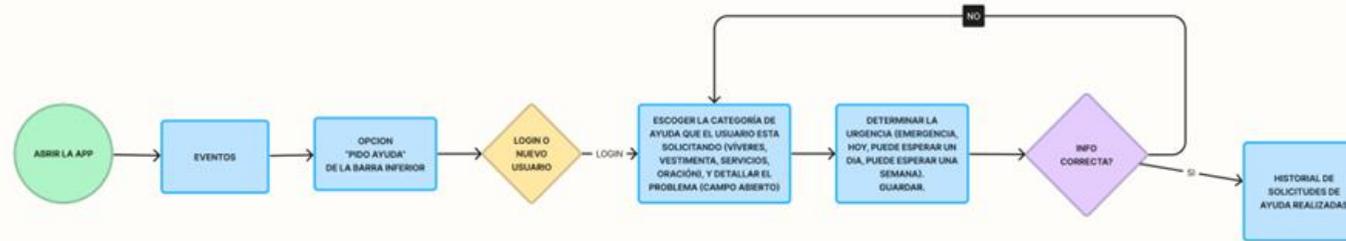


Imagen 35 User Flow para Registrar la Solicitud de Ayuda, segunda iteración
Tomado de-(Contreras-Almeida-2024)

Tarea: Responder a la solicitud de ayuda o asistencia en una categoría

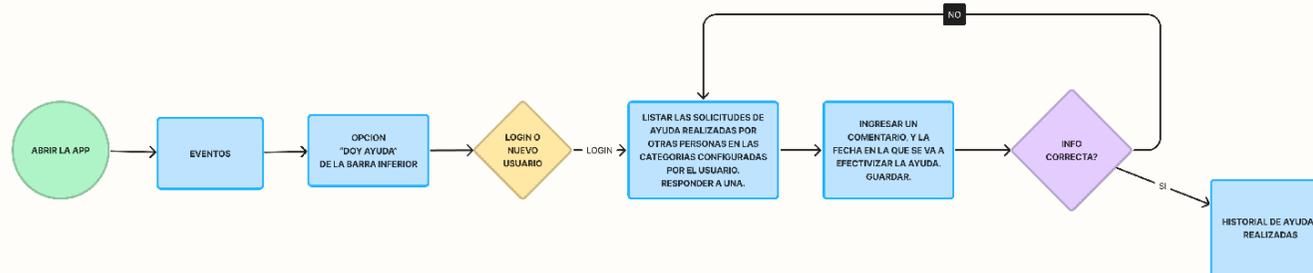


Imagen 36: User Flow para Responder a la Solicitud de Ayuda, segunda iteración
Tomado de-(Contreras-Almeida-2024)

Tarea: Registrar cuando una asistencia ha sido efectuada o no, y el motivo

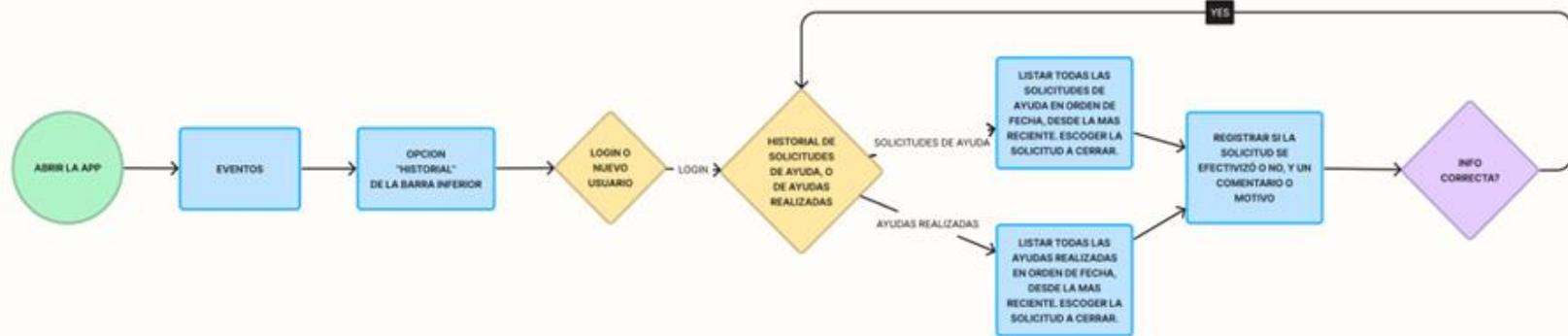


Imagen 37: User Flow para Registrar una Ayuda Efectivizada, segunda iteración
Tomado de-(Contreras-Almeida-2024)

4.7.3 PROTOTIPO FINAL

Con los cambios realizados en el mapa del sitio y los user flows ajustados, se elabora un segundo prototipo en Figma.

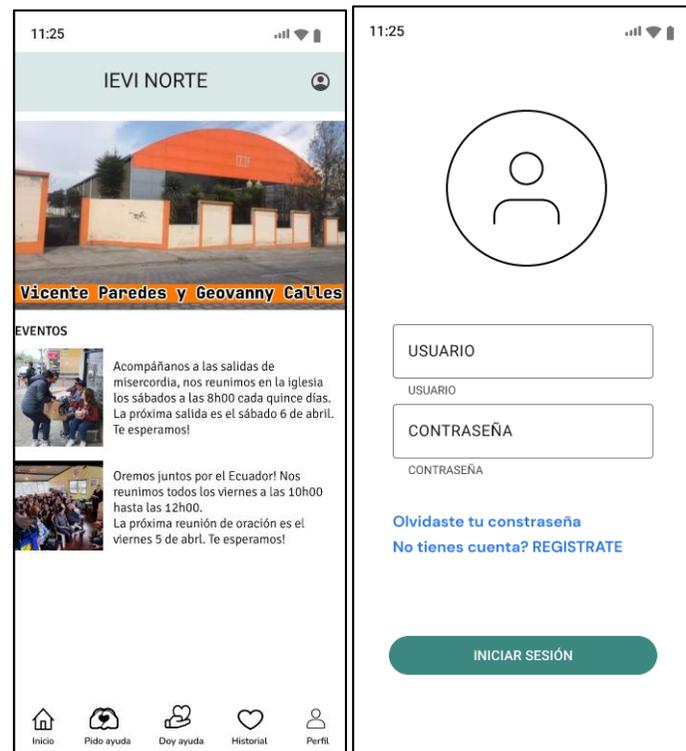


Imagen 38: Prototipo, Pantalla de Eventos e Inicio de Sesión, segunda iteración
Tomado de-(Contreras-Almeida-2024)

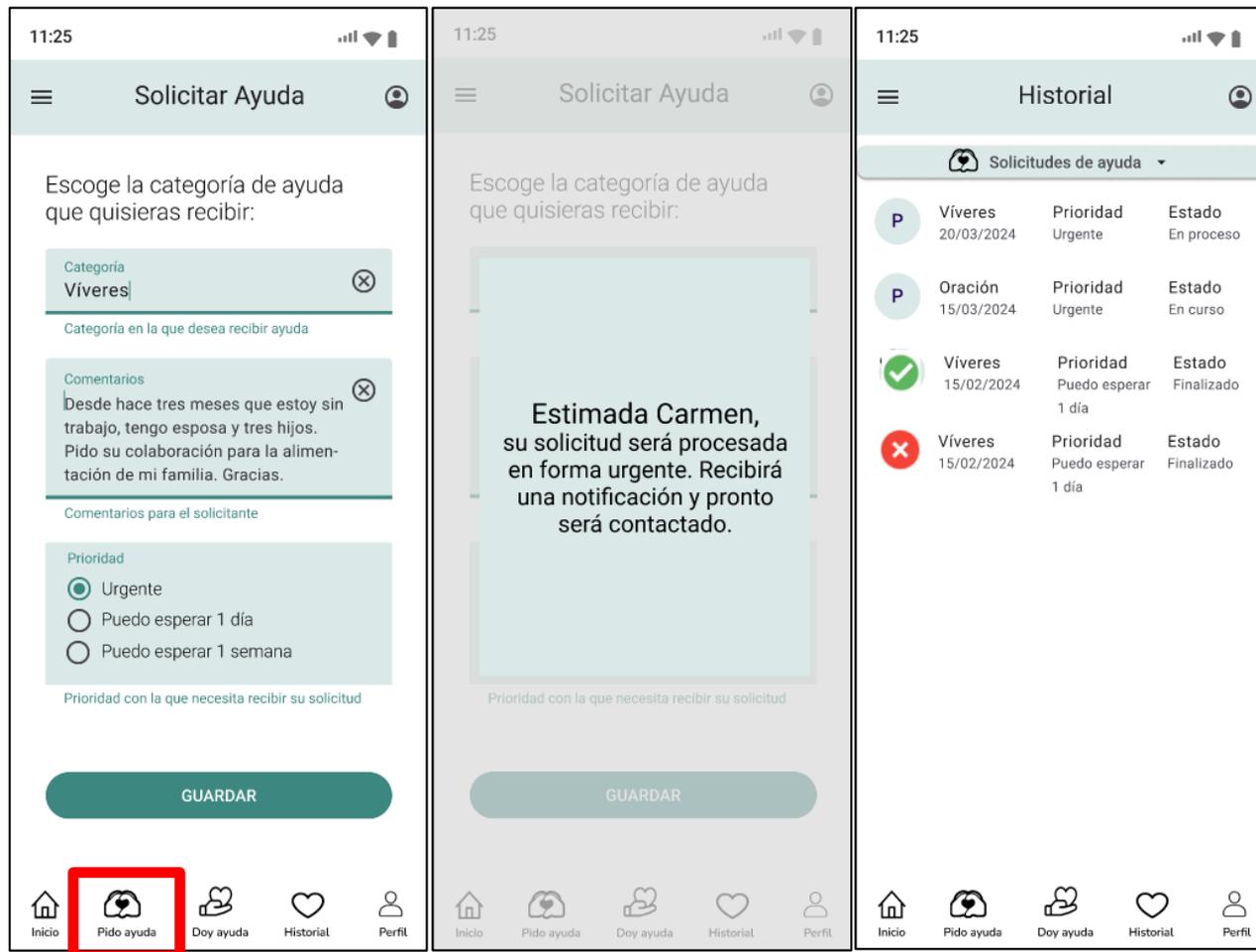


Imagen 39: Prototipo, Pantalla de Solicitud de Ayuda, segunda iteración
Tomado de-(Contreras-Almeida-2024)

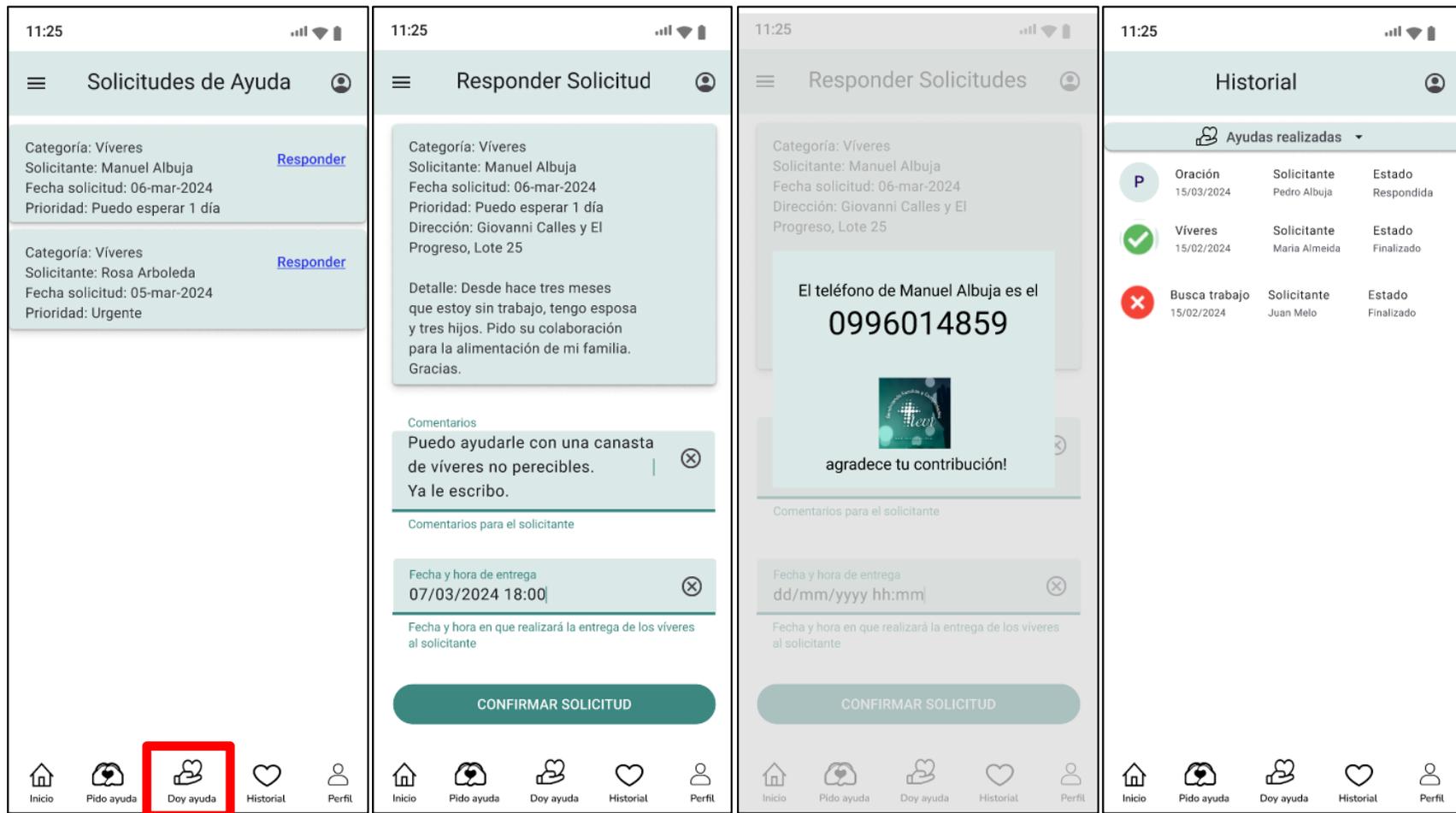


Imagen 40: Prototipo, Pantalla de Ofrecer Ayudas, segunda iteración
 Tomado de-(Contreras-Almeida-2024)

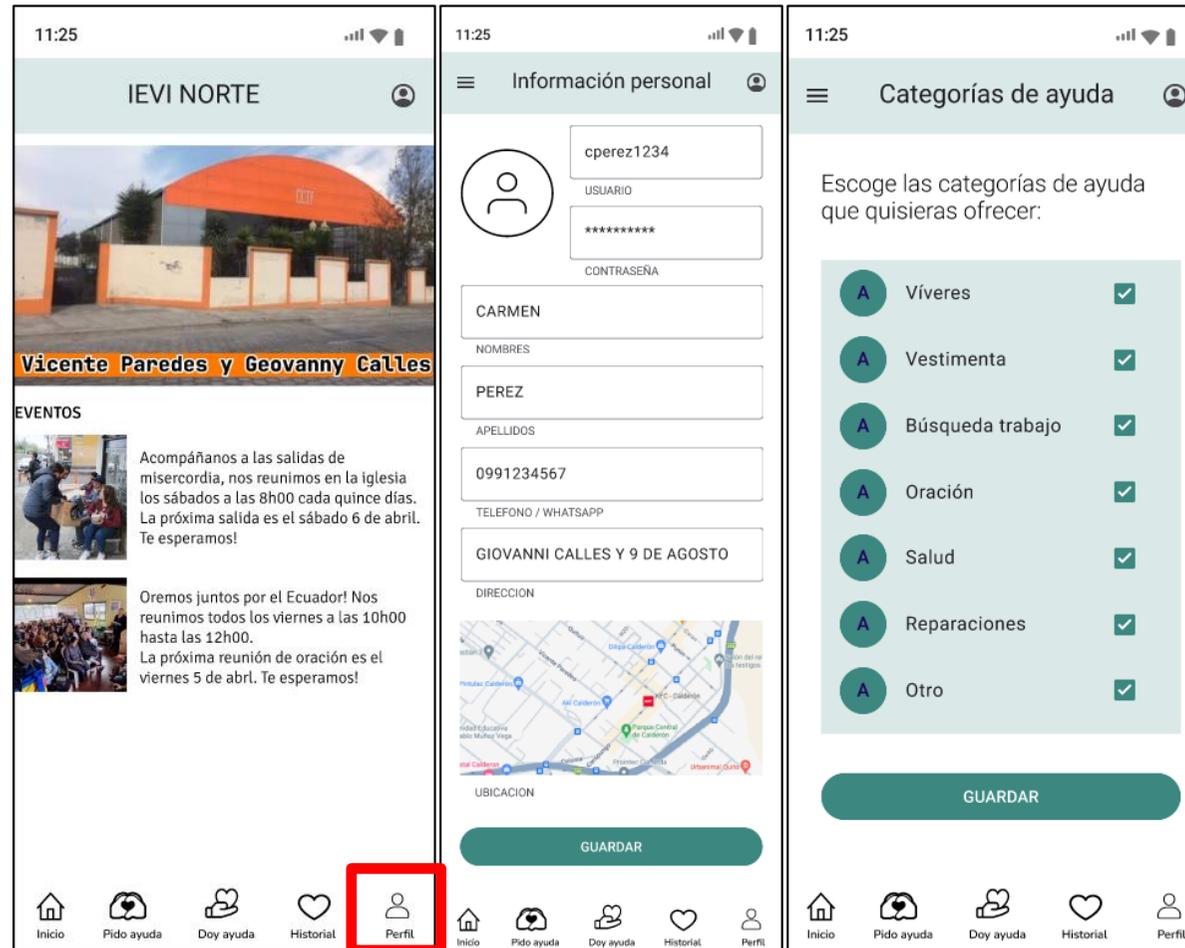


Imagen 41: Prototipo, Pantalla de Perfil y Categorías, segunda iteración
Tomado de-(Contreras-Almeida-2024)

5 RESULTADOS Y DISCUSIÓN

Una vez acordado el prototipo final, el cual se centra más en el diseño, se procede a crear un MVP, que será la primera versión de un producto funcional. Contiene las características principales y se debe lanzar a un público limitado.

Este MVP incluye los cambios al diseño derivados de las pruebas con los usuarios, e incluyen:

- La incorporación de una pantalla inicial con una lista de eventos de la iglesia, y solo cuando el usuario quiera pedir y ofrecer ayuda, debe autenticarse.
- El flujo para registrar las categorías de ayuda que el usuario puede ofrecer, se coloca en una pantalla contigua a la de “Datos Personales”.
- Inclusión de la pantalla de “Responder a la Solicitud de Ayuda”.
- Mensajes de éxito o error cuando se guardan los diferentes registros.

Estos cambios se reflejan más adelante, en las capturas de las pantallas de la Aplicación Móvil, en el apartado “Interfaces al Usuario” – Requerimientos Funcionales”.

5.1 MODELO DE BASE DE DATOS

La imagen muestra un modelo de datos orientado a objetos para la gestión de usuarios, solicitudes de ayuda y ayuda realizadas, categorías de ayuda, y eventos. A continuación, se muestra la imagen del modelo:

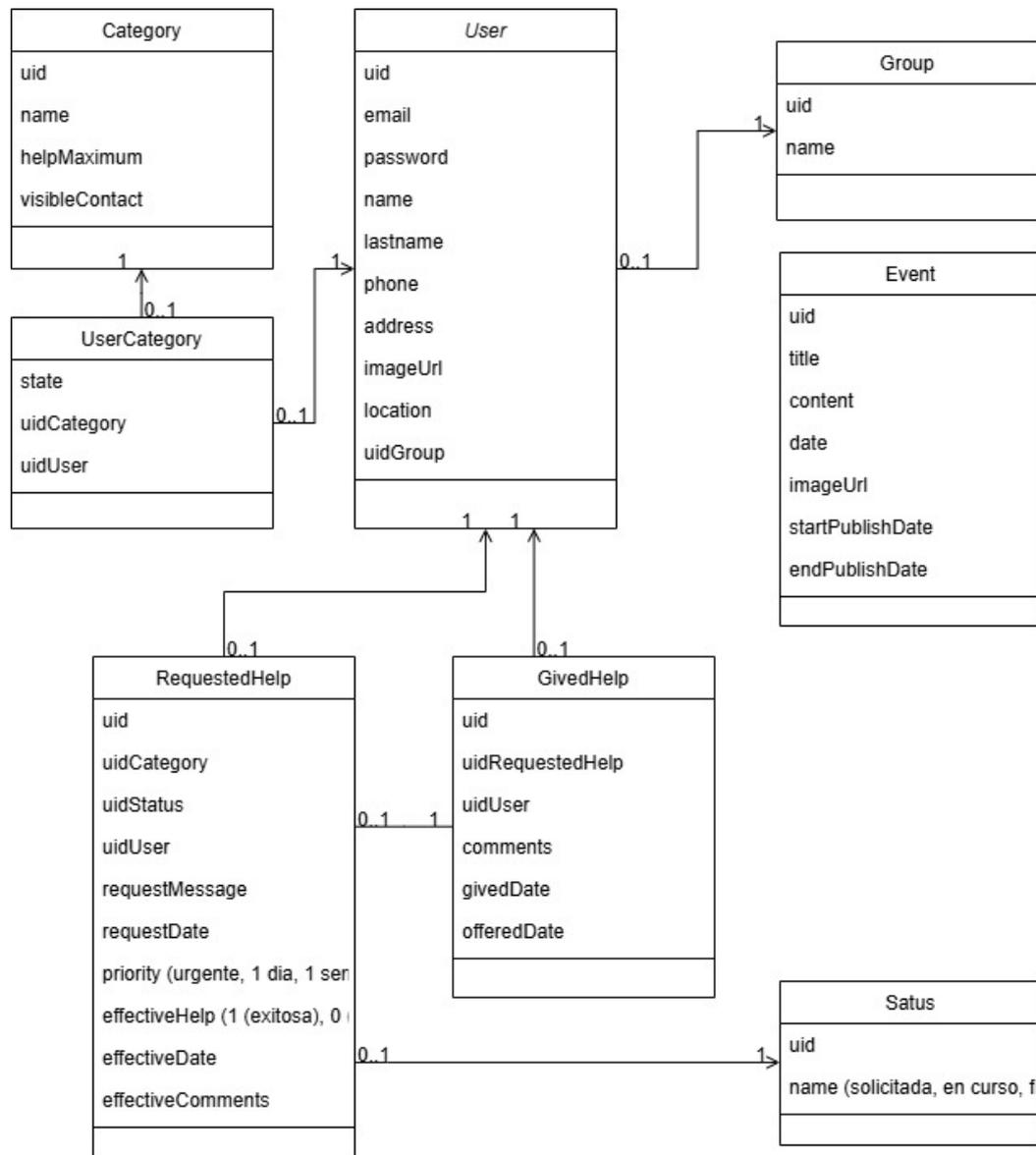


Imagen 42: Diagrama de Base de Datos
Tomado de-(Contreras-Almeida-2024)

5.2 FIREBASE

Firebase es una plataforma de desarrollo de aplicaciones proporcionada por Google, que está pensada para hacer más fácil la creación, administración y escalabilidad de aplicaciones móviles y web. Ofrece una serie de herramientas y servicios backend que facilitan las tareas comunes en el proceso de desarrollo.

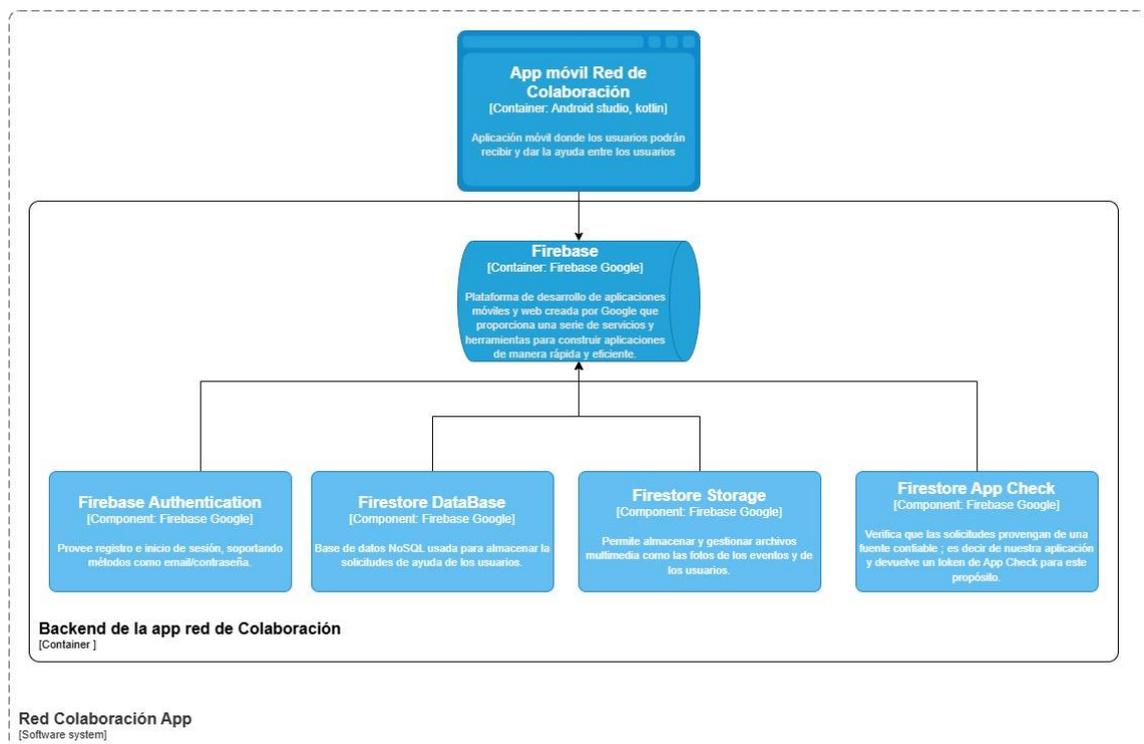
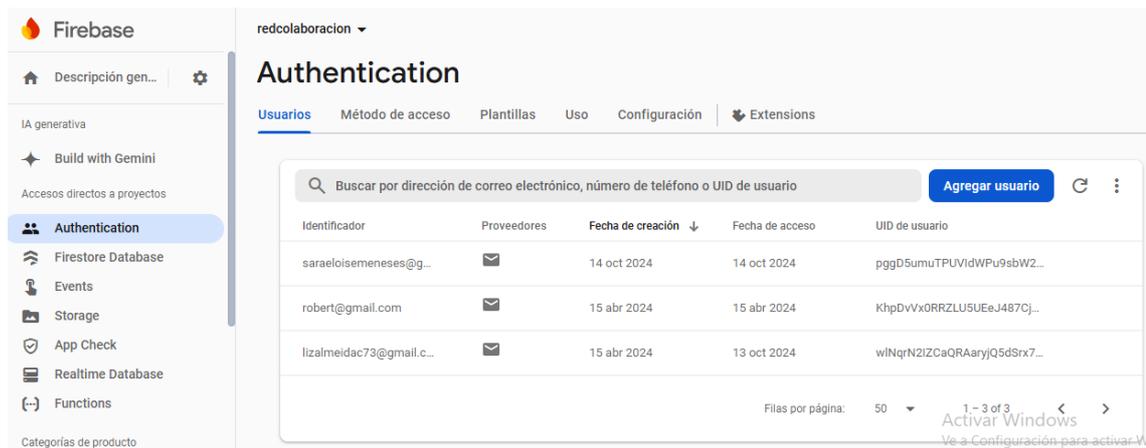


Imagen 43: Backend de la App Red de Colaboración
Tomado de-(Contreras-Almeida-2024)

5.2.1 FIREBASE AUTHENTICATION

Esta sección permite al administrador gestionar los usuarios, ver cuándo se registraron y accedieron, y observar el proveedor de autenticación que usaron para registrarse.



*Imagen 44: Firebase Authentication
Tomado de-(Contreras-Almeida-2024)*

5.2.2 FIREBASE FIRESTORE DATABASE

Esta base de datos es documental, de forma que cada documento representa una tabla del modelo de datos. Cada documento tiene un identificador que representa la clave primaria de cada tabla. La estructura que se muestra en la imagen tiene una colección llamada "requestedHelp", que está diseñada para almacenar solicitudes de ayuda de usuarios.

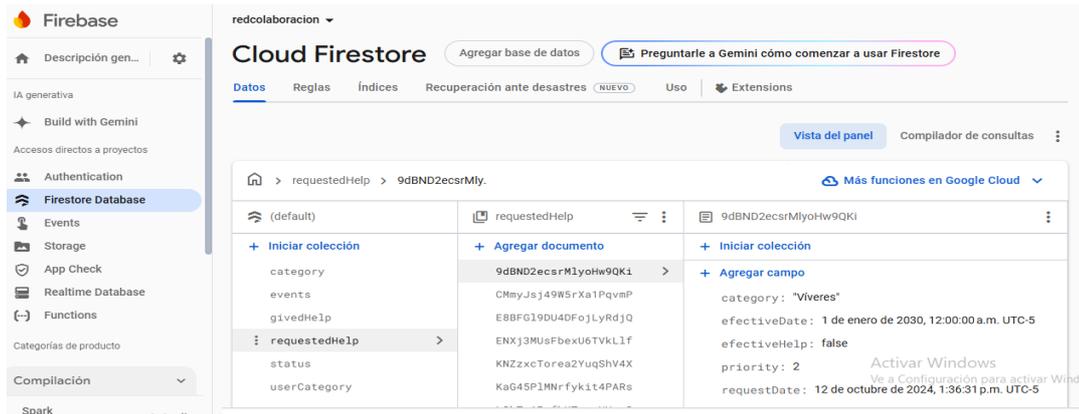


Imagen 45: Firebase Firestore Database
Tomado de-(Contreras-Almeida-2024)

5.2.3 FIREBASE STORAGE

Se almacena los archivos de imágenes de los eventos y de los usuarios en la carpeta *images* del Storage de Firebase, con la opción de subir nuevos archivos.

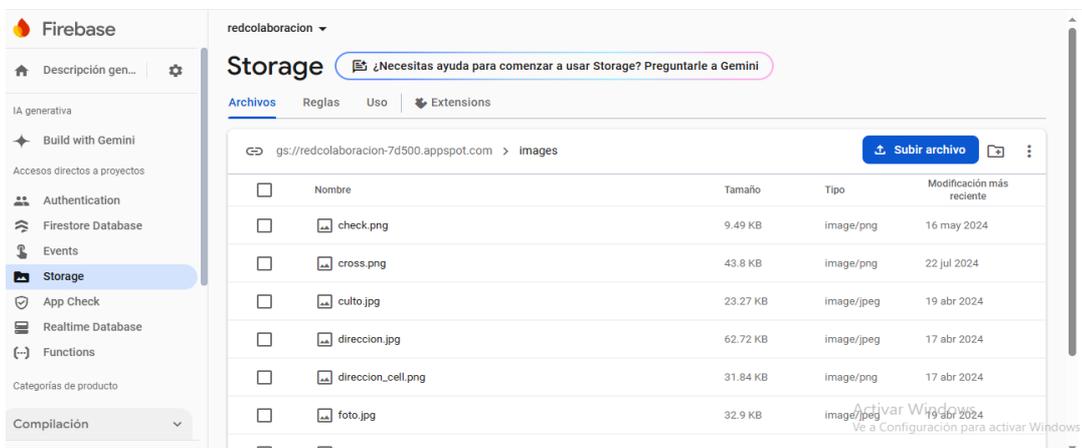
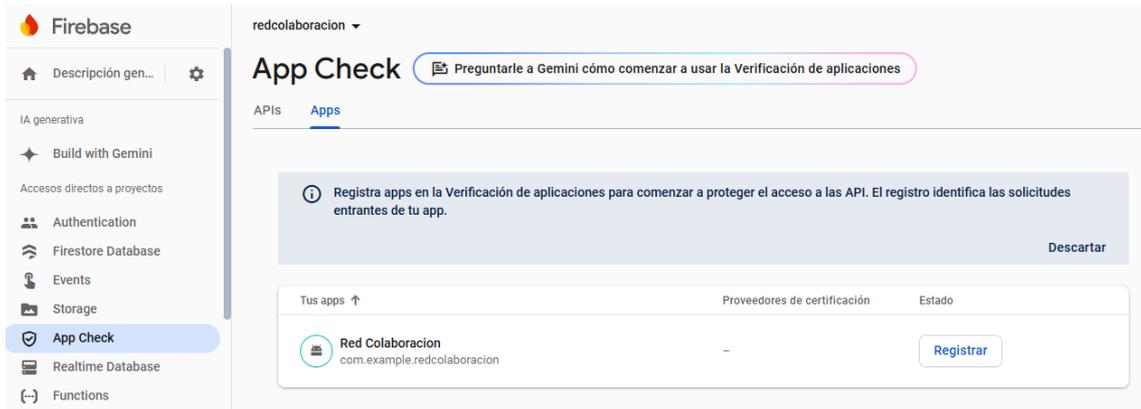


Imagen 46: Firebase Storage
Tomado de-(Contreras-Almeida-2024)

5.2.4 FIREBASE APP CHECK

El administrador puede configurar la protección de las API utilizando Firebase App Check para evitar accesos malintencionados o no autorizados.



*Imagen 47: Firebase AppCheck
Tomado de-(Contreras-Almeida-2024)*

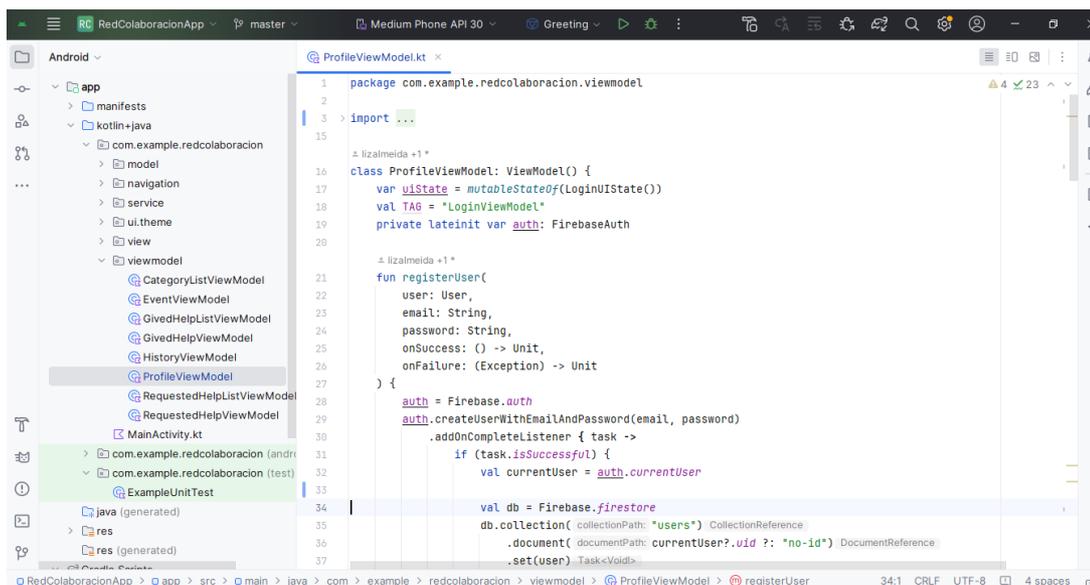
5.3 APLICACIÓN MÓVIL

Se procede a desarrollar la aplicación en base al prototipo final, en Android Studio, con lenguaje Kotlin, generando así el primer MVP con el que el usuario puede interactuar en forma real.

5.3.1 CÓDIGO FUENTE EN KOTLIN

La aplicación desarrollada en Android Studio sigue la arquitectura MVVM (Model-View-ViewModel), con algunas capas adicionales para servicios y navegación. Cada carpeta cumple con la siguiente función:

- Model: Contiene las clases de datos que representan los objetos de negocio.
- Service: Contiene la lógica de acceso a servicios. Podría separarse el contenido de los Viewmodel para colocar las conexiones con firebase en esta sección.
- Navigation: Define cómo navegar entre diferentes pantallas de la aplicación, usando NavController.
- Viewmodel: Contiene los modelos responsables de conectar la lógica de negocio con la UI.
- View: Incluye las clases de UI (interfaces al usuario), en su mayoría composables.



*Imagen 48: Programación en Lenguaje Kotlin
Tomado de-(Contreras-Almeida-2024)*

La figura muestra el IDE de Android Studio donde se reflejan las carpetas mencionadas, entre ellas la de Viewmodel que contiene la lógica del negocio. El archivo que se muestra es “ProfileViewModel.kt”, que contiene la función “registerUser” para crear el usuario y contraseña en el Firebase Authentication, así como para crear el usuario relacionado en la base de datos Firebase Firestore. Cabe mencionar que Firebase Authentication cuenta con funciones propias que permiten realizar actividades específicas en el Firebase, como por ejemplo “auth.createUserWithEmailAndPassword(email, password)” que crea el registro del usuario (correo, contraseña, fechas e identificador único UID) en Firebase Authentication.

El código fuente de la aplicación se encuentra en el link:

<https://github.com/lizalmeida/RedColaboracionApp.git>

5.3.2 INTERFACES AL USUARIO - REQUERIMIENTOS FUNCIONALES

Los requerimientos funcionales se evidencian a través de las pantallas de la aplicación móvil. Estas se encuentran en la carpeta View, y reflejan las pantallas diseñadas en el prototipo, excepto el mapa de ubicación del usuario que será útil cuando se implemente una funcionalidad para canalizar la ayuda a las personas más cercanas.



Pantalla de presentación, como bienvenida al usuario que está ingresando.



Pantalla de eventos, para que cualquier usuario los pueda observar, aunque no esté autenticado.

Imagen 49: Pantallas de Bienvenida y Eventos
Tomado de-(Contreras-Almeida-2024)

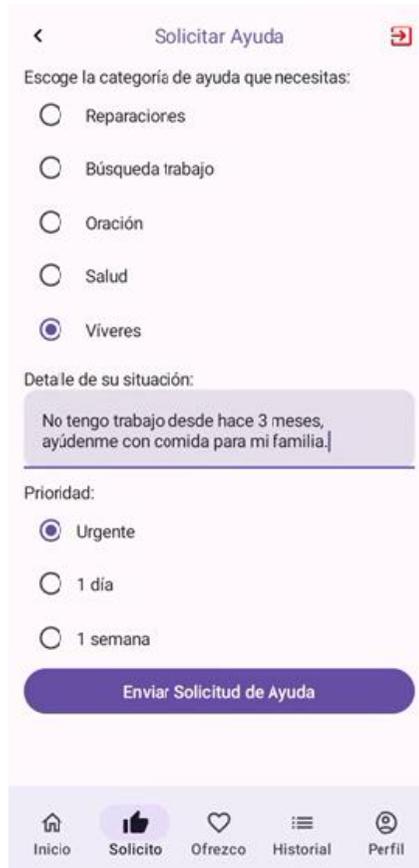


Para ingresar a cualquier opción de la barra inferior, el usuario debe autenticarse, y si no tiene un usuario, tiene la opción de crear una cuenta.



Si el usuario escoge crear una cuenta, o si presiona en el botón de “Perfil” de la barra inferior, aparecerá esta pantalla que le permite ingresar sus datos personales. Cuando el usuario está autenticado, la pantalla le permitirá actualizar sus datos o restablecer su contraseña si la ha olvidado.

Imagen 50: Pantallas de Login y Perfil de Usuario
Tomado de-(Contreras-Almeida-2024)



El usuario que necesita la ayuda, especifica en qué categoría la necesita, el detalle de su situación, y la prioridad.

Internamente se almacena la fecha de solicitud, y el estado se coloca “Solicitada”.

Al presionar el botón de “Enviar Solicitud de Ayuda”, el sistema redirige a la pantalla de “Historial de Ayudas”.



La pantalla de “Historial de Ayudas” permite al usuario escoger sus solicitudes de ayuda o sus ayudas realizadas.

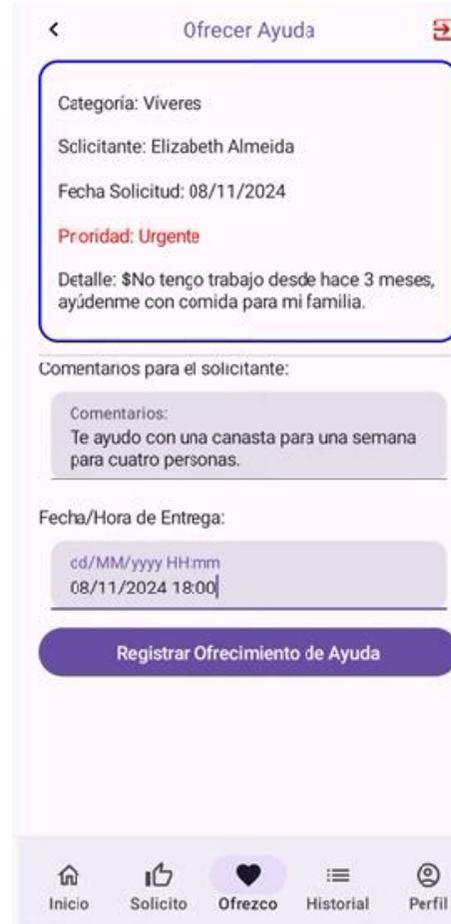
Las solicitudes de ayuda tienen una primera columna que refleja si la ayuda fue efectivizada o no. Permanece en cruz roja hasta que el usuario cierra la solicitud a conformidad (cambia a visto verde) o permanece roja si nunca recibió la ayuda.

Imagen 51: Pantallas de Solicitud de Ayuda e Historial (Solicitudes)
Tomado de-(Contreras-Almeida-2024)



Para poder ofrecer ayuda, el primer paso es escoger a quién ayudar. En esta pantalla se listan los requerimientos de ayudas de las categorías que el usuario ha configurado en su perfil, en las que desea colaborar.

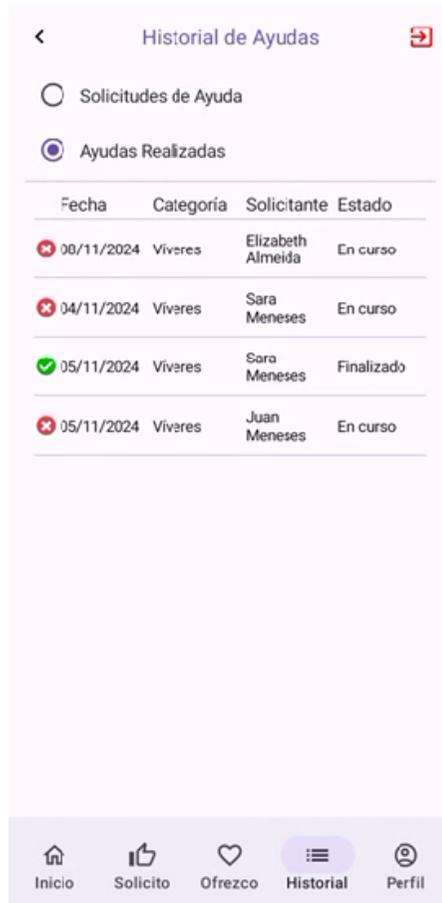
Al escoger la ayuda que desea brindar, se despliega la pantalla para confirmar la asistencia.



Se despliega en la cabecera los comentarios que puso el solicitante de la ayuda. El que la ofrece, debe ingresar algún comentario para el solicitante, y la fecha y hora estimada de entrega de la ayuda.

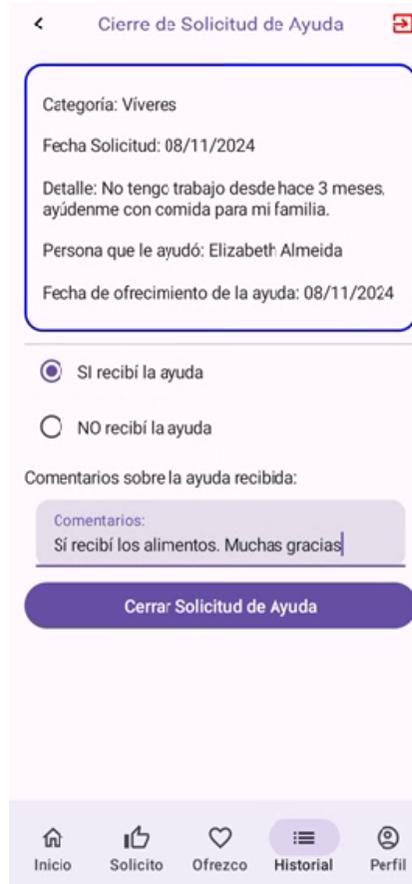
Al presionar el botón de “Registrar Ofrecimiento de Ayuda”, la solicitud cambia a estado “En curso”, y el sistema redirige a la pantalla de “Historial de Ayudas”.

Imagen 52: Pantallas de Lista de Solicitudes y Ofrecer Ayuda
Tomado de-(Contreras-Almeida-2024)



La pantalla de “Historial de Ayudas” permite al usuario escoger sus solicitudes de ayuda o sus ayudas realizadas.

Al presionar en una de las ayudas realizadas, se despliegan los datos de la persona a la que el usuario ofreció ayudar, para que pueda contactarlo.



Una vez que el usuario que ofrece la ayuda, contacta al solicitante y físicamente se concreta la asistencia, el usuario la escoge de la lista de sus solicitudes para proceder a cerrarla. De esta forma termina el ciclo.

La solicitud de ayuda cambia a estado “Finalizada”, con un visto verde si el usuario recibió la ayuda, o con una cruz roja si no la recibió.

Imagen 53: Pantallas de Historial (Ayudas realizadas) y Cierre de Solicitudes
Tomado de-(Contreras-Almeida-2024)

El software desarrollado cumple con los requerimientos funcionales, esto es con el ciclo desde que el usuario registra sus datos personales y solicita la ayuda hasta que registra si esta ayuda se realizó en forma efectiva y en conformidad para el usuario solicitante.

5.3.3 REQUERIMIENTOS NO FUNCIONALES

En cuanto a los requerimientos no funcionales, se cumplieron de acuerdo con lo planteado inicialmente en la hipótesis:

- Compatibilidad con Android

La aplicación fue desarrollada en Android Studio, por lo cual es compatible con todos los dispositivos Android, que es el más ampliamente utilizado en dispositivos móviles en Ecuador.

- Usabilidad

En los resultados del test de usabilidad del prototipo y sobre todo de la encuesta, los usuarios muestran su conformidad con la apariencia y la comprensión del producto. Además, la sola idea de contar una herramienta de este tipo, ilusiona a los usuarios al pensar que van a poder pedir ayuda emergente, imaginar que no van a estar solos, sino que hay toda una comunidad que está pendiente de ellos, al tiempo que van a poder sentirse útiles ayudando a los demás. Esto es muy importante, aun cuando la aplicación tenga algunos puntos por mejorar.

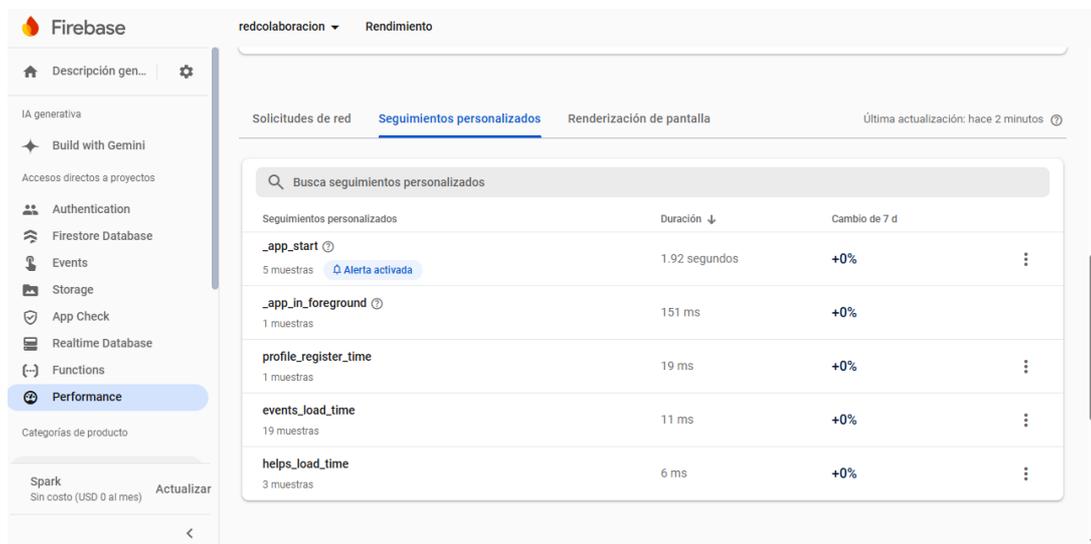
- Tiempo de respuesta rápido

Al ser una plataforma robusta en la nube, Firebase garantiza un tiempo de respuesta rápido, y facilita el control de acceso a los datos, y su monitoreo. Para ello, Firebase cuenta con la herramienta Firebase Performance, que captura métricas relacionadas con el rendimiento, incluyendo tiempos de respuesta de la UI y el tiempo que tarda la aplicación en completar ciertas tareas importantes.

Se pueden agregar métricas que se visualizan en el panel principal del Firebase Performance, y se visualizan en el transcurso del tiempo y el uso de la aplicación.

También se pueden filtrar las métricas para las diferentes versiones de la aplicación, y agregar alertas cuando el tiempo de respuesta supera un valor especificado.

El tiempo de respuesta de la carga de la aplicación es de 1.92 segundos, que se encuentra dentro de un valor aceptable (menor a 2 segundos). Se agregaron algunas métricas para medir el tiempo de respuesta para completar la carga de la pantalla de eventos, de la de ayudas, y el registro del perfil de un usuario. Los resultados fueron los que muestra el siguiente gráfico, que también están dentro de un valor aceptable. Con el uso del sistema por parte de los usuarios, estos valores deben ser supervisados en forma frecuente.



*Imagen 54: Firebase Performance
Tomado de-(Contreras-Almeida-2024)*

- Consumo de recursos:

La cantidad de información que se envía o se recibe es pequeña, lo más grande es el tamaño de las imágenes, específicamente de las fotografías del perfil del usuario, que se ha controlado ajustando a una resolución aceptable. Firebase Performance también proporciona información detallada del uso de CPU y memoria, que se puede habilitar para cada métrica o seguimiento personalizado, en cada sesión, cuando la aplicación tenga una mayor demanda.

- Seguridad: Al igual que el tiempo de respuesta, Firebase se encarga de la creación y autenticación de los usuarios, validando las cuentas de correo con las que los usuarios ingresarán al sistema, así como la regla de generación de las contraseñas. Adicionalmente, en la aplicación se controla que los datos personales sean visibles solo para los usuarios que lo necesitan para efectivizar las ayudas.

5.4 DISCUSIÓN

Adicional al beneficio obtenido por la aplicación móvil desarrollada, el presente proyecto ha sido de gran utilidad para percibir los puntos más importantes a ser tomados en cuenta en el desarrollo de un proyecto de una aplicación móvil utilizando una metodología ágil.

5.4.1 METODOLOGÍA

En esta ocasión, el uso de la metodología Lean Startup no fue tan ágil por la falta de experiencia en el uso de metodologías ágiles para el desarrollo de aplicaciones. Sin embargo, se pudo percibir que es de gran ayuda para concentrarse en el valor del “negocio”, más que en los detalles que restan tiempo para obtener un producto funcional en el menor tiempo posible.

Gracias a Lean Startup, el MVP desarrollado fue una aplicación completa y funcional, lista para ser probada y utilizada por un conjunto de usuarios, quienes permitirán en una próxima iteración, obtener una retroalimentación adecuada para realizar los cambios finales a la aplicación, antes de su lanzamiento a toda la comunidad.

Este MVP cumple en gran medida con los objetivos planteados, con los requerimientos funcionales y no funcionales, pero que probablemente necesite nuevos ciclos de desarrollo ágil para ir adaptándolo en respuesta al uso del sistema y a la experiencia del usuario.

Del MVP resultante se desprende que este modelo puede ser replicable para otras iglesias y organizaciones sociales similares, pues favorece el fortalecimiento de redes de apoyo y solidaridad, a través del intercambio de recursos y servicios que promueve la colaboración en momentos de necesidad.

5.4.1 MATERIALES

En cuanto a las herramientas, el uso de Android Studio como entorno de desarrollo integrado (IDE) y el lenguaje Kotlin han permitido aprovechar al máximo las capacidades de la plataforma Android, asegurando una experiencia de usuario optimizada y una interfaz intuitiva. Sin embargo, tiene el limitante de ser una herramienta que solo funciona en dispositivos Android; para iOS se requiere desarrollar una aplicación completa en lenguaje Swift. La programación en Swift es bastante similar a Kotlin, por lo que este trabajo no representaría una complejidad en tiempo y recursos demasiado alta.

En la gestión de datos, Firebase Firestore Database ha permitido la sincronización de datos en tiempo real, garantizando un acceso rápido y seguro a la información. La base de datos documental es muy adecuada para prototipos y MVP's, incluso en aplicaciones a gran escala o de alta concurrencia. Aunque su administración conlleva más trabajo, se la considera una mejor alternativa por el ahorro de toda una infraestructura en el backend.

La integración de Firebase no solo facilita la implementación de características como autenticación de usuarios y almacenamiento de datos, sino que también mejora la escalabilidad y el rendimiento general de la aplicación. Esto permite que se pueda adaptar a las necesidades cambiantes de los usuarios y responder a un mayor volumen de solicitudes sin comprometer la calidad del servicio.

6 CONCLUSIONES

- La aplicación Red de Colaboración Social es una aplicación móvil que permite a los usuarios de una Comunidad solicitar ayuda y ofrecerla a quien la necesita, por lo que cumple con el objetivo general de ser una solución tecnológica en la cual los integrantes de una organización social puedan dar o recibir bienes o servicios de acuerdo a las habilidades y las necesidades de cada uno.
- También se han cumplido los objetivos específicos del proyecto, al pasar por las etapas de su desarrollo dando la importancia a la experiencia del usuario, estableciendo en su diseño los requerimientos funcionales y no funcionales e implementando un MVP que reúne la mayoría de las expectativas de los usuarios.
- Las bases teóricas han sido útiles para mantener un esquema ágil en todas las etapas de la metodología Lean Startup, concentrándose en mantener el enfoque del valor del “negocio”, pues al ser una entidad sin fines de lucro, el valor es el beneficio obtenido por las ayudas canalizadas entre los miembros de la iglesia.
- El uso de herramientas tecnológicas versátiles e intuitivas, que permitan un flujo dinámico de las diferentes etapas del desarrollo, es clave para un desarrollo ágil, en particular en entornos de incertidumbre, como el de la iglesia, donde existe imprevisibilidad de la respuesta del usuario ante nuevas experiencias. Esto se ha logrado con el desarrollo ágil de un prototipo que permitió una rápida retroalimentación, y con el desarrollo ágil de una aplicación utilizando Firebase para casi todos los servicios requeridos por una arquitectura de aplicaciones móviles, de forma que se pudo dar mayor atención a la experiencia del usuario.
- La aplicación desarrollada se proyecta como una herramienta muy útil para facilitar el intercambio de recursos y servicios, fomentar la solidaridad, fortalecer el sentido de comunidad, y promover la colaboración en momentos de

necesidad, de forma que los miembros de la comunidad se sientan apoyados y valorados. El reto estará en equilibrar la tecnología con la empatía y la participación activa de todos los miembros, garantizando que el beneficio colectivo prevalezca sobre cualquier limitación o riesgo inherente al uso de esta plataforma digital.

7 TRABAJO FUTURO

- El próximo paso es ejecutar la fase de despliegue de la aplicación, para un número limitado de usuarios de forma que se obtenga el feedback para realizar los últimos ajustes antes de un lanzamiento público oficial. Se propone realizar una prueba cerrada (beta testing) de la aplicación móvil. Para garantizar una distribución controlada, se deben realizar los siguientes pasos:
 - Generar una versión de lanzamiento (release build), generando un archivo APK desde el IDE de Android Studio.
 - Seleccionar una plataforma de distribución. Aunque lo usual es utilizar Google Play Console, se sugiere utilizar Firebase App Distribution, para invitar a los testers y recoger el feedback.
 - Recopilar datos para detectar errores y mejorar la aplicación utilizando Firebase Crashlytics. También se puede utilizar herramientas como Google Forms para enviar encuestas a los testers después de un periodo de uso.
 - Analizar los resultados y realizar mejoras, identificando errores frecuentes y cuellos de botella en la experiencia del usuario. Se puede agrupar los problemas por severidad y frecuencia.
 - Planificar el lanzamiento público, generando una versión final y volver a distribuir para validación. Actualizar la versión lista para producción en el Firebase App Distribution. Finalmente, crear materiales promocionales si es necesario (descripciones, capturas de pantalla, videos).
- Existen dos funcionalidades que no fueron necesarias para consolidar el primer MVP, y que deben ser desarrolladas en la versión final de la aplicación:
 - Que el usuario cuente con la posibilidad de autenticarse utilizando su cuenta de Google predefinida en su dispositivo, o con su cuenta de Facebook.
 - Que en el perfil del usuario pueda almacenarse la ubicación exacta utilizando la API de Google Maps, con el objetivo de que el sistema canalice la ayuda a las personas más cercanas.

8 REFERENCIAS

- [1]. Logan Kugler. Smartphone apps for social good. *Communications of the ACM*, 59:18–20, 7 2016.
- [2]. J Ranjani, V Meenakshi Sundaram, and K Karthik. G-get: All in one donation app. pages 645–650. *IEEE*, 11 2022.
- [3]. Hans Jørgen Wiberg. *The story about be my eyes*, 2023.
- [4]. Lee Lazar and Naomi I. Eisenberger. The benefits of giving: Effects of prosocial behavior on recovery from stress. *Psychophysiology*, 59, 2 2022.
- [5]. IEEE Computer Society Standards Coordinating Committee. *IEEE Standard Glossary of Software Engineering Terminology*. 1990.
- [6]. Murein Miksa Mardhia and Ela Dwi Anggraini. Implement a lean UX model: Integrating students’ academic monitoring through a mobile app. page 1–5. *IEEE*, 9 2019.
- [7]. *Lean Mobile App Development: Apply Lean startup methodologies to develop successful iOS and Android apps*. Lean Mobile App Development: Apply Lean startup methodologies to develop successful iOS and Android apps. Packt Publishing Ltd, 2017.
- [8]. Nataliia Veretennikova and Roman Vaskiv. Application of the lean startup methodology in project management at launching new innovative products. Pages 169–172. *IEEE*, 9 2018.
- [9]. Mohammad Abdur Razzak. An empirical study on lean and agile methods in global software development. Pages 61–64. *IEEE*, 8 2016.
- [10]. EG Maida and J. Pacienza. *Metodologías de desarrollo de software*. tesis de licenciatura en sistemas y computación. 2015.
- [11]. Andrea Vazquez-Ingelmo, Alicia Garcia-Holgado, and Francisco J. Garcia-Penalvo. C4 model in a software engineering subject to ease the comprehension of UML and the software. Pages 919–924. *IEEE*, 4 2020

REFERENCIAS DE IMÁGENES

- Abhinav Asthana, A. S. (2012). *Postman*.
- Adobe. (2022). *Figma*. Obtenido de <https://www.figma.com/es-es/>
- Arvind Padmanabhan, P. K. (2021). *Devopedia*. Obtenido de <https://devopedia.org/firebase>
- Atlassian. (2017). *Trello*. Obtenido de <https://trello.com/b/6Cp4c5EU/proyecto-de-grado>
- Google. (2013). *Android Studio*. Obtenido de <https://developer.android.com/studio?hl=es-419>
- Google LLC. (2024). *Draw.io*. Obtenido de Draw.io: https://app.diagrams.net/#G1ddDa3eVtaqJej_P5Ftvo8gOS1pq3csvn#%7B%22pageId%22%3A%22C5RBs43oDa-KdzZeNtuy%22%7D
- Google LLC. (2024). *Firebase*. Obtenido de Firebase: <https://console.firebase.google.com/project/redcolaboracion-7d500/overview?hl=es-419>
- Melanie Perkins, C. O. (2012). *Canva*. Obtenido de <https://www.canva.com/projects>
- Merino, Á. M. (18 de 09 de 2024). *SNGULAR*. Obtenido de SNGULAR: <https://www.sngular.com/es/insights/313/firebase>
- Microsoft Corp. (2024). *GitHub*. Obtenido de GitHub: <https://github.com/lizalmeida/RedColaboracionApp>
- Tune, N. (2013). *Structurizr*. Obtenido de <https://structurizr.com/>
- Vega, J. A. (2018). *Maze*. Obtenido de <https://maze.co/>

ANEXO A: INFORME UX

COLABORADORES O EQUIPO DE TRABAJO

Los colaboradores de este proyecto son el Diseñador de Experiencia de Usuario, dos usuarios miembros de la iglesia, comprometidos, que van a utilizar, evaluar y retroalimentar a la aplicación

PÚBLICO

El público consta de las personas que forman parte de una organización de colaboración social, que en este caso es una iglesia. Son personas de todo tipo de edad, estrato social, educación, habilidades y preferencias.

Este público variado comprende desde jóvenes que les resulta fácil e intuitivo el manejo de aplicaciones en el celular, hasta personas adultas mayores para quienes la aplicación debe constar de dos o tres opciones. Esto va a representar un problema a la hora de realizar un diseño personalizado centrado en el usuario, por lo que se puede pensar en interfaces personalizadas, por ejemplo, según la edad.

ENTREVISTA ABIERTA

Entrevistado: Pastor Eduardo Vega, Coordinador de la Iglesia Evangélica de Iñaquito Norte

Fecha: 22/12/2022

Objetivo: conocer qué es lo que espera de la aplicación, y tener una primera aproximación de la funcionalidad a diseñar.

¿Cuál es el problema que está teniendo la iglesia para asistir a las personas que forman parte de ella?

Cuando una persona necesita ayuda de cualquier tipo, puede ser de salud, medicinas, comida, trabajo, incluso oración, por el momento se está manejando a través de los diferentes grupos pequeños, de discipulado, gedeones, oración, entre otros, pero no se cuenta con una base de información de quiénes realmente vienen en forma frecuente, ni cuentan con una forma de solicitar esta ayuda; a veces ni siquiera la piden por temor o vergüenza. Y por el otro lado, existen personas que les gusta ayudar mucho, a veces con las capacidades que cada uno tiene, puede ser un médico, un obrero, una ama de casa, o intercediendo a través de la oración.

Cómo imagina usted que puede funcionar la aplicación?

Lo primero es cada persona pueda registrarse con su nombre, en lo posible una dirección y un teléfono donde ubicarles. Luego presentarles la posibilidad de ofrecer ayuda en alguna área, como los que mencioné, salud, medicinas, comida, oración, con la opción de poner “Otros” para ir incorporando otras áreas que se puedan clasificar. Finalmente, cada vez que una persona necesite ayuda, la solicita de acuerdo a estas áreas definidas. La aplicación debería buscar quiénes pueden ayudarle con su solicitud y contactar hasta que alguno responda afirmativamente. Si la ayuda se concreta, se debería registrar en el sistema para poder evaluar si el sistema está funcionando de esta manera. Si no, para buscar alternativas de mejora.

Qué problemas cree que se puedan presentar?

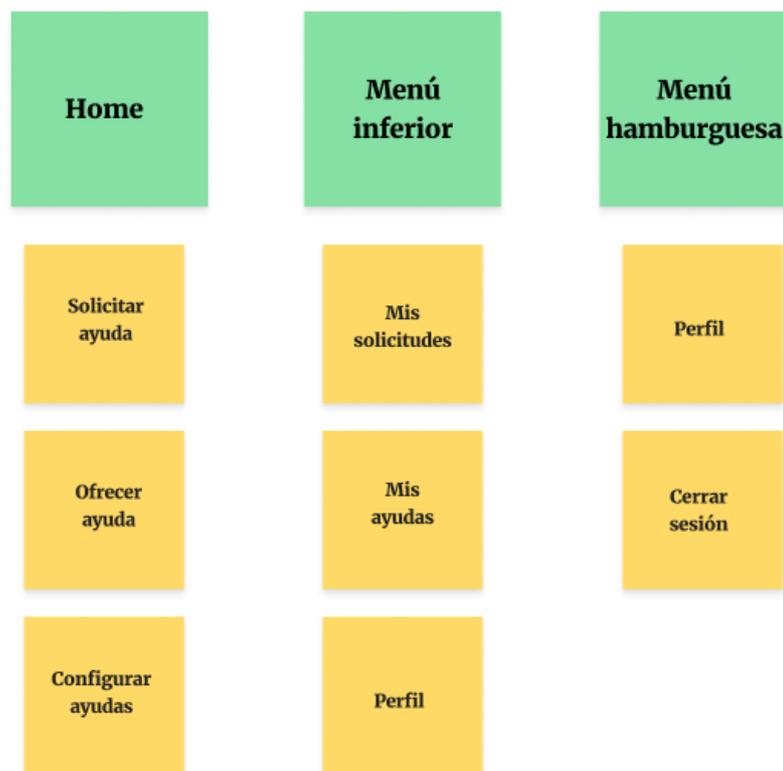
Que las personas, sobre todo las mayores, no cuenten con celular que soporte la aplicación, o no la utilicen por miedo o falta de conocimiento. Pienso que esto se puede ir resolviendo con el tiempo, hasta que se vayan acostumbrando, pero sí puede representar un problema porque muchas personas son de escasos recursos.

Adicionalmente, otro problema puede ser que la gente, sobre todo por estas zonas, es muy exigente, y puede creer que es una obligación de la iglesia canalizar la ayuda que están solicitando, en lugar de ver que es una colaboración individual y voluntaria de cada persona. Esto es un problema porque a la final quien queda mal ante cualquier ayuda no concretada es la iglesia. Pienso que, al principio, hasta ver

cómo marcha este tema, se puede clasificar a las personas por grupos, y se indique de alguna forma que cada categoría de ayuda sea solo para tales grupos, por ejemplo, pueden pedir ayuda de víveres solo los que pertenecen al grupo de discipulado. De esa forma, se puede ir controlando en forma paulatina.

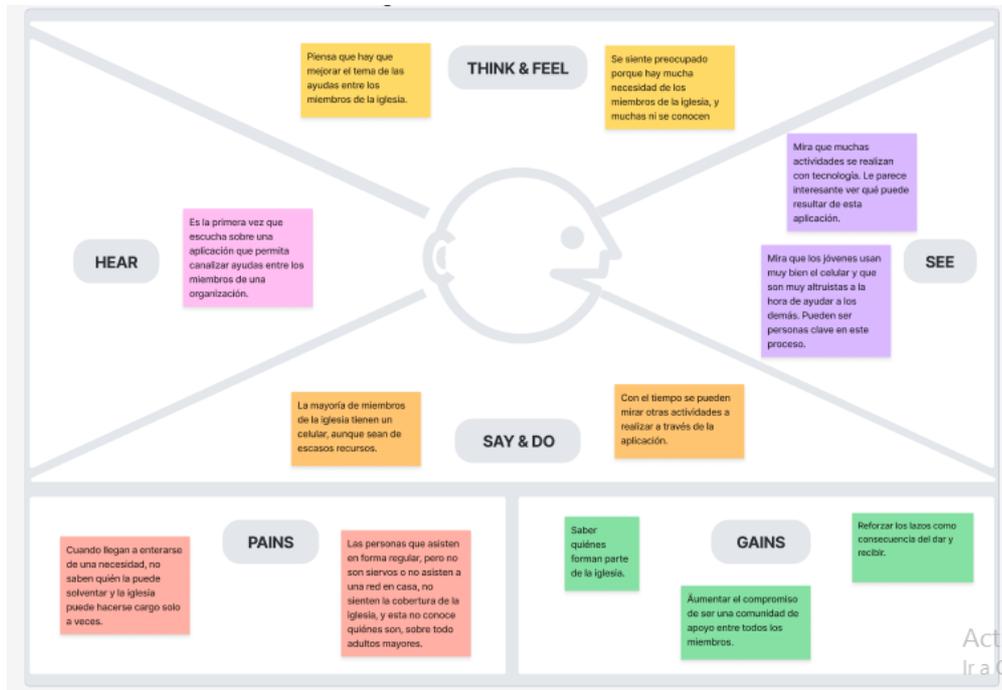
CARD SORTING

De las principales categorías y contenidos de la aplicación.

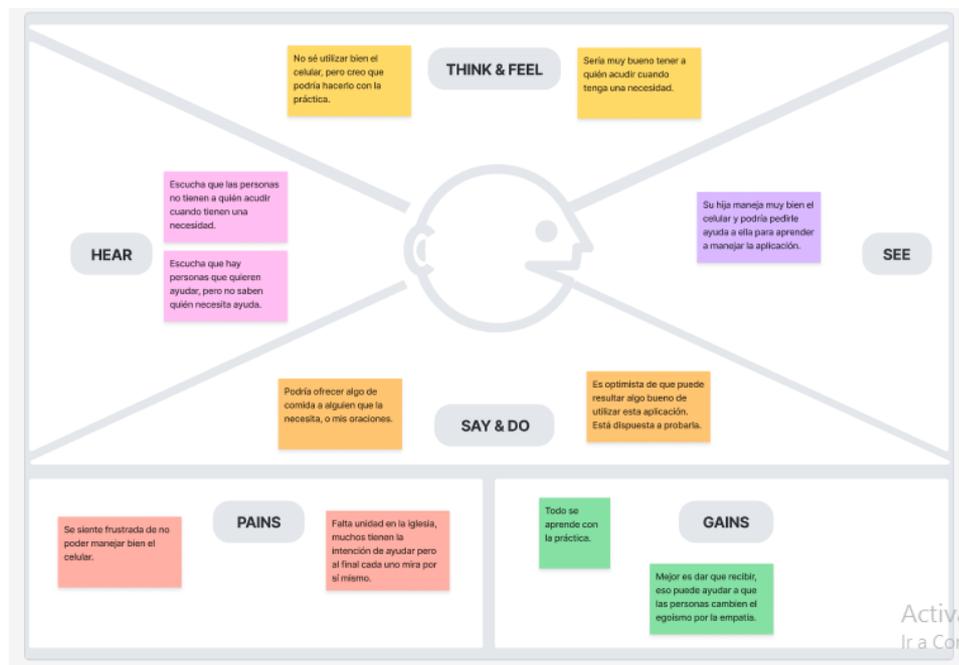


MAPAS DE EMPATÍA

Pastor Eduardo Vega



Carmen Pérez



USER PERSONA

Pastor Eduardo Vega

NOIN DESIGN

Project: User Persona

Client: Eduardo Vega

Date: February 20th, 2024

Persona
The persona canvas can be used to give a customer segment a face and name.



Eduardo Vega

"Soy una persona alegre, optimista, con esperanza, que tiene su mirada puesta en Dios, y el propósito que El tiene para mi vida. Soy un instrumento de Dios para llevar su amor y su promesa de salvación al mundo. Tengo una maravillosa esposa Karina, con la que tenemos tres preciosos hijos, que también son la razón de nuestra vida. Cuando tengo algo de tiempo para mí, me gusta caminar en el bosque o las montañas con mi esposa, y mirar toda la hermosa creación que Dios nos ha dado."

Job Title: Leda, en Teología
Status: Happy
Location: Quito - Ecuador

GOALS

- Fortalecer la unidad de la iglesia que Dios ha puesto a mi cargo como Pastor.
- Especializarme en la carrera de Teología.
- Formar mi propia iglesia.
- Mantener un balance entre las actividades de la iglesia y el tiempo con mi esposa e hijos.
- Viajar de vacaciones con mi esposa y mis hijos una vez al año dentro y fuera del país.

NEEDS

- Recursos para cumplir los objetivos de la iglesia
- Recursos para estudios de mis hijos
- Más colaboración de los miembros de la iglesia
- Un auto nuevo

PERSONALITY

Introvert ————— Extrovert

Analytical ————— Creative

Busy ————— Time rich

Messy ————— Organized

Independent ————— Team player

INFLUENCERS
What products or services are influencing

f in s

BEHAVIOR TRAITS

PASSIONATE EMPATHETIC

CURIOUS ADVENTUROUS

PAIN POINTS

- Cambio de ubicación de la iglesia en un año, a una infraestructura propia.
- La organización de las personas es complicada, cada uno quiere hacer valer sus puntos de vista.
- Generar ingresos para la manutención de los hijos, educación, alimentación, vestimenta, entretenimiento.
- Tiempo insuficiente para todas las actividades planificadas en el día.

INTERESTS

- Evangelización.
- Fortalecer la unidad de la iglesia.

Carmen Pérez

NOIN DESIGN

Project: User Persona

Client: Carmen Pérez

Date: February 20th, 2024

Persona
The persona canvas can be used to give a customer segment a face and name.



Carmen Pérez

"Soy una persona alegre, dinámica, que quiere compartir con los demás. Madre de dos hijos, a los que amo mucho. Me encantan los niños. Me apasiona la costura y la música. En mi tiempo libre, me gusta conocer lugares nuevos, compartiendo con mi familia."

Job Title: Tec. Costura
Status: Happy
Location: Quito - Ecuador

GOALS

- Trabajar con World Vision como misionera.
- Aprender a cantar en forma profesional.
- Terminar de estudiar el bachillerato y ser chef.
- Apoyar a mi hija para que sea profesional e independiente.

NEEDS

- Motivación para empezar a estudiar.
- Tiempo y dinero para cumplir mis objetivos.

PERSONALITY

Introvert ————— Extrovert

Analytical ————— Creative

Busy ————— Time rich

Messy ————— Organized

Independent ————— Team player

INFLUENCERS
What products or services are influencing

f in s

BEHAVIOR TRAITS

PASSIONATE EMPATHETIC

CURIOUS ADVENTUROUS

PAIN POINTS

- Falta de recursos para cumplir mis objetivos
- Problemas de salud, no graves, pero afectan

INTERESTS

- Tener mi propio negocio local para artesanías y bazar.
- Disfrutar de la compañía de mi hija y apoyarla en sus sueños de ser profesional y viajar.

JOURNING MAP

Para registrar las categorías para prestar ayuda.



Eduardo
"Sales Executive"

- 54
- Pastor
- Experienced of 30

Background:
Eduardo Vega, Pastor de la Iglesia Evangélica de Iñaquito. Dedicado a la evangelización de las personas del sector de Calderón, y aumentar el compromiso y la unidad entre los miembros de la iglesia.

Challenges:
Incentivar a los miembros de la iglesia para participar dando y recibiendo ayuda a quien la necesite, y aumentando así el sentido de pertenencia.

Goals:
Registrar los datos personales y las ayudas que puede ofrecer a los demás miembros de la iglesia.

Phases	Discovery	Onboarding	Personalisation	Information Seeking	Continued Engagement
User actions	Se registra el inicio de la actividad de un usuario. <i>Inicio de sesión</i>	La información es bienvenida a Eduardo y se le muestra los primeros pasos para comenzar a usar el sistema. <i>Inicio de sesión</i>	Eduardo ha completado los pasos para comenzar a usar el sistema. <i>Inicio de sesión</i>	Tras registrar los datos personales, Eduardo puede recibir información sobre los servicios que ofrece el sistema. <i>Inicio de sesión</i>	Tras registrar los datos personales, Eduardo puede recibir información sobre los servicios que ofrece el sistema. <i>Inicio de sesión</i>
Touchpoints	El usuario interactúa con el sistema a través de un navegador web.	El sistema muestra una guía de bienvenida para ayudar al usuario a comenzar a usar el sistema.	El sistema muestra una guía de bienvenida para ayudar al usuario a comenzar a usar el sistema.	El sistema muestra una guía de bienvenida para ayudar al usuario a comenzar a usar el sistema.	El sistema muestra una guía de bienvenida para ayudar al usuario a comenzar a usar el sistema.
Feelings and thoughts	 Frustración por no poder registrar los datos personales.	 Satisfacción por poder registrar los datos personales.	 Satisfacción por poder registrar los datos personales.	 Satisfacción por poder registrar los datos personales.	 Satisfacción por poder registrar los datos personales.
Pain points	¿Cómo registrar los datos personales?	¿Cómo registrar los datos personales?	¿Cómo registrar los datos personales?	¿Cómo registrar los datos personales?	¿Cómo registrar los datos personales?
Possible steps solutions	Registrar los datos personales.	Registrar los datos personales.	Registrar los datos personales.	Registrar los datos personales.	Registrar los datos personales.
Storyboarding					

ANEXO B: RESULTADOS DEL TEST DE USABILIDAD



Objetivo del Test

El objetivo principal del test es valorar si los usuarios comprenden el funcionamiento de la aplicación y el beneficio que obtendría la organización y sus miembros a través de él. También permitirá validar lo que se ha derivado de las entrevistas y encuestas con los usuarios. Y finalmente, detectar los puntos de dolor, frustraciones e inconformidades de los usuarios frente al diseño de la aplicación.

Los resultados del test ofrecerán una retroalimentación al diseño de la aplicación para realizar los ajustes o mejoras necesarios.



Método

Una vez que se contruye el prototipo de la aplicación en base a las entrevistas y encuestas a los usuarios, se lo levanta en una herramienta donde los usuarios puedan mirar las diferentes funcionalidades, en este caso en Maze. Con esta misma herramienta se diseña una encuesta que permita obtener las opiniones de los usuarios respecto al prototipo.

Una vez recopiladas las respuestas tanto del prototipo como de la encuesta, se analiza la información para extraer las conclusiones que permitan mejorar la aplicación y por tanto, la experiencia del usuario.



Participantes

Se escogieron como usuarios participantes para evaluar el proyecto y realizar la encuesta a un pastor, un siervo, dos colaboradores y ocho asistentes a la iglesia en calidad de miembros integrantes que asisten en forma regular, doce personas en total.

Por el momento no fue posible observar sus expresiones o reacciones, que hubiera sido lo ideal, para validar lo que está acertadamente diseñado, e identificar los puntos de dolor más fácilmente.



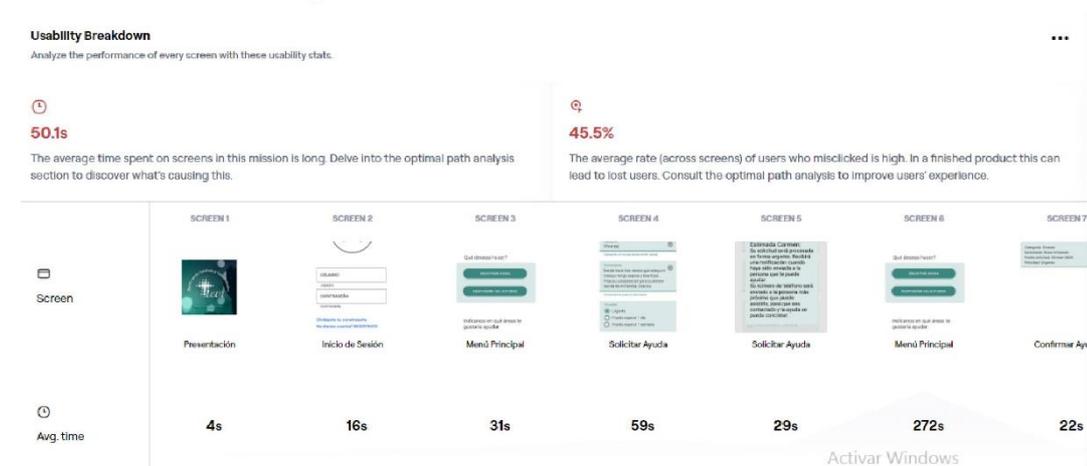
Criterios de selección de los participantes

- Representatividad de la muestra: todos los integrantes de la iglesia van a utilizar la aplicación con el mismo propósito, pero cada perfil puede aportar con un punto de vista diferente, por lo que es importante recopilar las impresiones de todos los tipos de usuarios.
- Perfil demográfico: De la misma forma, el criterio de personas con edades, género, y nivel educativo variados va a contribuir en la recopilación de las necesidades, objetivos y expectativas de los usuarios objetivo.
- Experiencia con tecnología: En una primera etapa, la muestra incluye usuarios que tengan familiaridad con aplicaciones en dispositivos móviles.

Reporte de Usabilidad del Prototipo



Rendimiento por Pantalla



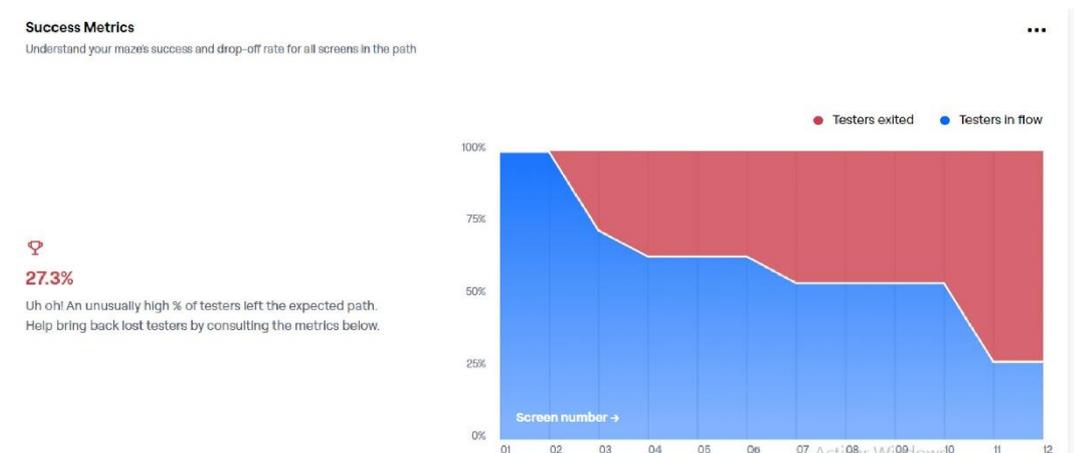
Esta gráfica nos muestra que otra razón para acompañar al usuario en la revisión del prototipo es que puede quedarse en una pantalla por mucho tiempo por hacer otra actividad, con lo que el tiempo se incrementa sin un motivo justificado.

Mapas de Calor



Los mapas de calor nos muestran que el usuario necesita una guía personalizada para movilizarse en el prototipo, puesto que la mayoría espera poder ingresar información o escoger cualquier botón de la pantalla.

Métricas de Éxito



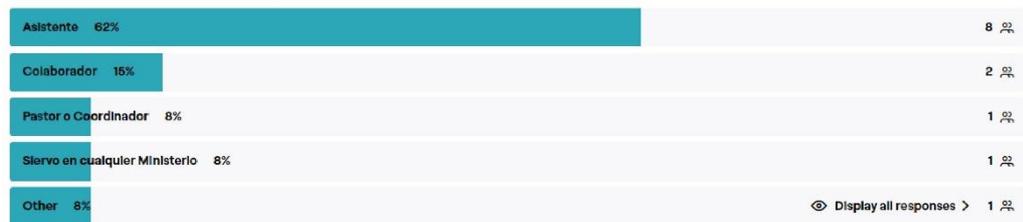
Esta gráfica nos muestra que los usuarios se fueron perdiendo conforme avanzaban en el flujo del prototipo.

ANEXO C: RESULTADOS DE LA ENCUESTA

Resultados de la Encuesta

Escoja la función que desempeña en la iglesia.
Multiple Choice

13
Responses

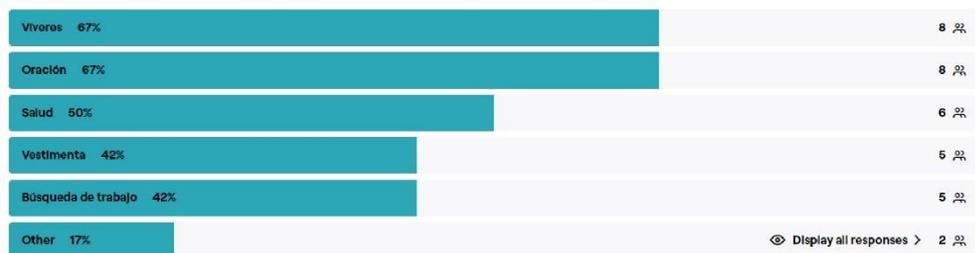


Existen usuarios de todas las categorías, sobre todo de quienes van a utilizar la aplicación (Asistentes).

Resultados de la Encuesta

En qué áreas cree que se puede ayudar a los miembros de la iglesia?
Multiple Choice

12
Responses



Las respuestas corroboran las opciones de ayuda que se pueden dar y recibir en la iglesia.

Resultados de la Encuesta

¿La combinación de colores en la aplicación le pareció agradable?

Yes/No

12 Responses

YES/NO

100%
12 testers

0%
0 testers

Todos están de acuerdo con la combinación de colores utilizada en la aplicación.

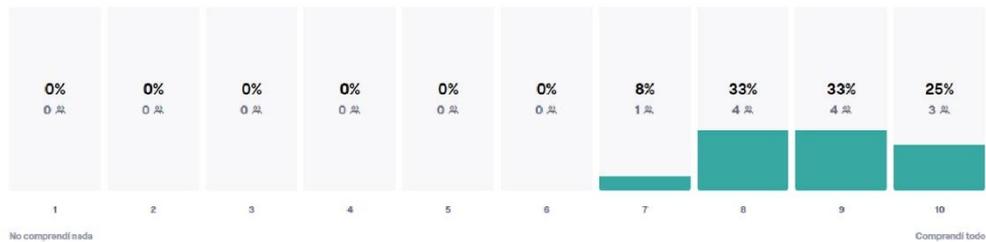
Resultados de la Encuesta

¿Cuál fue el nivel de comprensión de lo que hace la aplicación?

Opinion Scale

12 Responses

8.8 Average



El nivel de comprensión de la aplicación es bueno, a pesar de que el prototipo no presta la facilidad de que el usuario sepa por dónde va el flujo en forma secuencial.

Resultados de la Encuesta

¿Qué nivel de utilidad considera que tendría la aplicación para su iglesia?
Opinion Scale

12 Responses
9 Average



Un nueve es un excelente promedio para tener una buena expectativa de uso de la aplicación, sobre todo considerando que es una aplicación nueva.

Resultados de la Encuesta

¿Qué sugerencias realizaría para mejorar la aplicación?
Open Question [Read about AI data privacy](#)

11 Responses

- Todo está perfecto
- Que se necesite gente de verdad comprometida en ayudar en este proyecto
- todo bien
- La aplicación experimenta pequeños bloqueos
- Se necesitaría un proceso de verificación de datos y situación de los solicitantes de ayuda. Para evitar el uso indiscriminado de la aplicación.
- Los mensajes podrían ser más concretos. El botón de responder solicitud, cambiar por "Ofrecer ayuda. Podría ser el nombre de la Aplicación "911 IEVI"
- Si lo van a gestionar en toda la iglesia lograría ser muy útil
- En la página inicial donde indica cargos de la iglesia. Sería mejor indicar: voluntario activo e inactivo. Ya que sería más compromiso espiritual bíblico que por emociones momentáneas.
- Poder dejar datos personales
- Colocar la ubicación de la persona a la que se va a ayudar para contactarla más rápido.

De las sugerencias se extraen algunos cambios en el diseño que pueden mejorar la experiencia del usuario, como colocar los mensajes más concretos, o ponerle un nombre llamativo a la aplicación.