

**Implementación de una aplicación móvil con Angular para la entrega asistida de medicamentos a adultos mayores en el barrio El Carmen, Tunjuelito – Bogotá**

Johan Sebastián Mejía Carmona

Johanna Milena Muñoz Arciniegas

Juan Esteban Venegas Torres

Director:

Arellys De Jesús Correa Rodríguez

Universidad Nacional Abierta y a Distancia-UNAD

Escuela de Ciencias Básicas Tecnología e Ingeniería.

Ingeniería de Sistemas

2026

## Resumen

En el barrio El Carmen, ubicado en la localidad de Tunjuelito en Bogotá, los adultos mayores y las personas con movilidad reducida enfrentan múltiples barreras para acceder a medicamentos. Las largas filas en los puntos de dispensación, que en algunos casos superan las dos horas, sumadas a riesgos de seguridad, fallas en los sistemas de autorización y condiciones climáticas adversas, generan abandono en los tratamientos y un evidente deterioro en la calidad de vida de los usuarios. Ante esta situación, el presente proyecto propone el desarrollo de una aplicación móvil orientada a optimizar el proceso de entrega de medicamentos a domicilio. El objetivo principal es reducir los tiempos de espera y mejorar la accesibilidad de la población vulnerable, garantizando continuidad terapéutica y bienestar. La propuesta se enmarca en un enfoque de investigación aplicada con diseño mixto, que integra encuestas y entrevistas a usuarios y personal de salud para identificar necesidades reales. La metodología incluye la adopción de la gestión ágil mediante Scrum, organizada en sprints para el diseño, validación y prueba del sistema. El desarrollo técnico, basado en Angular, Node.js y MongoDB, busca asegurar escalabilidad, seguridad e integración con sistemas de las EPS.

**Palabras clave:** Entrega de medicamentos, adultos mayores, movilidad reducida, aplicación móvil, Scrum, Angular, Node.js, MongoDB, integración con EPS, Tunjuelito, Producto Mínimo Viable

## Abstract

In the El Carmen neighborhood, located in the Tunjuelito district of Bogotá, older adults and people with reduced mobility face multiple barriers to accessing medicines. Long lines at dispensing points, which in some cases exceed two hours, combined with security risks, failures in authorization systems, and adverse weather conditions, lead to treatment abandonment and a clear deterioration in users' quality of life. Given this situation, this project proposes the development of a mobile application aimed at optimizing the process of home delivery of medications. The main objective is to reduce waiting times and improve accessibility for vulnerable populations, ensuring therapeutic continuity and well-being. The proposal is part of an applied research approach with a mixed design, which integrates surveys and interviews with users and health personnel to identify real needs. The methodology includes the adoption of agile management using Scrum, organized in sprints for the design, validation, and testing of the system. The technical development, based on Angular, Node.js, and MongoDB, seeks to ensure scalability, security, and integration with EPS systems.

**Keywords:** Medicine delivery, elderly people, reduced mobility, mobile application, Scrum, Angular, Node.js, MongoDB, integration with EPS, Tunjuelito, Minimum Viable Product.

## Tabla de Contenido

Resumen.....	2
Abstract.....	3
Tabla de Contenido.....	4
Lista de Tablas .....	12
Lista de Figuras.....	14
Lista de Apéndices.....	17
Introducción .....	18
Justificación .....	19
Planteamiento del Proyecto.....	21
Objetivos.....	24
Objetivo General .....	24
Objetivos Específicos.....	24
Marco Referencial.....	25
Antecedentes .....	25
Aplicaciones Móviles Orientadas a la Entrega de Medicamentos .....	27
Tecnologías Utilizadas: Angular y Metodologías Ágiles (Scrum) .....	28
Metodología Scrum en Proyectos de Software Aplicado.....	28
Brecha Identificada y Aporte del Proyecto .....	28

Marco Teórico.....	32
Patrones de Diseño y Técnicas de Gestión de Dependencias.....	32
Teorías de Diseño y Arquitectura de Software.....	39
Teorías de Requisitos y Experiencia de Usuario.....	43
Teorías de Procesos y Metodologías de Desarrollo.....	44
Teorías de Calidad de Software.....	48
Teorías de Interoperabilidad en Salud.....	49
Teoría de la optimización de recursos.....	51
Mínimo Producto Viable (MVP).....	51
Otras Teorías.....	53
Marco Conceptual.....	54
Términos Técnicos Fundamentales.....	54
Términos del Dominio de Aplicación.....	57
Relación entre Conceptos.....	61
Marco Legal.....	63
Ley 1581 de 2012 Protección de Datos Personales.....	63
Ley 1438 de 2011 Reforma del Sistema de Salud.....	63
Ley 23 de 1981 Código de Ética Médica.....	64
Ley 1266 de 2008 Habeas Data Financiero y de Información Personal.....	64

Ley 1751 de 2015 Derecho Fundamental a la Salud.....	64
Ley 527 de 1999 Comercio Electrónico .....	65
Regulación de Plataformas Móviles y Datos Personales.....	65
Normativas Internacionales .....	65
Marco Metodológico.....	69
Tipo de estudio .....	69
Enfoque de investigación .....	69
Hipótesis.....	70
Técnicas e instrumentos de recolección de información.....	70
Encuesta semiestructurada.....	71
Metodologías a implementar .....	73
Business Process Management (BPM).....	74
Marco ágil Scrum .....	75
Fases del Desarrollo del Proyecto .....	76
Análisis del proceso actual de entrega de medicamentos en el barrio El Carmen (Tunjuelito – Bogotá).....	79
Actores y roles del proceso de entrega de medicamentos en el barrio el Carmen .....	79
Segmentación de público Objetivo .....	80
Aplicación de técnicas para recolección de información .....	80
Encuesta semiestructurada a adultos mayores y personas con movilidad reducida .....	80

Análisis de Resultados .....	82
Descripción del Proceso Actual de entregas de medicamentos en el barrio El Carmen - Tunjuelito .....	87
Modelo AS-IS del proceso de entregas de medicamentos en el barrio El Carmen – Tunjuelito .....	88
Descripción del Proceso en su Estado Actual .....	88
Entradas del Proceso.....	90
Salidas del Proceso .....	90
Herramientas Utilizadas .....	90
Actores Involucrados.....	90
Puntos Críticos (Cuellos de Botella Identificados) .....	91
Identificación de puntos críticos Dolores (Pains) y necesidades de usuario.....	91
Requisitos del sistema.....	92
Rediseño del proceso de Aplicación para entrega de medicamentos MediExpress.....	93
Propuesta Estado Futuro (TO-BE) .....	93
Definición del MPV .....	95
Épicas del Proyecto .....	96
Historias de Usuario .....	98
Diseño de la arquitectura de la aplicación móvil para la entrega de medicamentos.....	110
Objetivo Arquitectónico.....	112

Principios Arquitectónicos .....	112
API First .....	112
Modularidad .....	113
Bajo acoplamiento y alta cohesión .....	113
Seguridad por defecto (OWASP) .....	113
Cloud Ready .....	113
12-Factor App.....	114
Reutilización de componentes .....	114
Observabilidad integrada.....	114
Drivers Arquitecturales .....	114
Atributos de Calidad (Requisitos No Funcionales).....	116
Vistas Arquitectónicas (Modelo C4).....	117
Diagrama de Contexto (C4-1) .....	118
Diagrama de Contenedores (C4-2) .....	119
Diagrama de Componentes (C4-3).....	121
Vistas adicionales (Rozanski & Woods) .....	125
Vista de Información / Datos.....	127
Vista de Despliegue .....	128
Vista de Seguridad.....	130

Arquitectura Lógica.....	132
Capa de Presentación (Frontend ).....	132
Capa de Aplicación (Servicios / Casos de uso) .....	133
Capa de Dominio (Reglas de negocio) .....	133
Capa de Infraestructura (Persistencia, Comunicación, Frameworks).....	134
Arquitectura Física (Infraestructura).....	135
Servidor VPS / Nube (Hosting Principal) .....	135
Balanceador de Carga (Load Balancer).....	136
CDN (Content Delivery Network) .....	136
Contenedores Docker .....	136
Red Interna (Private Network) .....	137
Modelo de Datos .....	137
Entidades y Atributos .....	137
Relaciones entre Entidades .....	140
Integración y APIs.....	141
Seguridad y Privacidad.....	142
Mecanismo de autenticación y autorización.....	142
Cifrado en tránsito y en reposo.....	142
Registro y auditoría de eventos .....	143

Políticas de copia de seguridad y restauración .....	143
Cumplimiento con normativas de privacidad .....	143
Estrategia de Pruebas .....	144
Decisiones Arquitectónicas (ADR).....	145
Estrategia de Despliegue .....	146
Observabilidad y Mantenimiento .....	146
Métricas (latencia, errores, disponibilidad).....	147
Logs centralizados y trazabilidad de solicitudes .....	147
Alertas automáticas.....	147
Políticas de actualización, backup y restauración .....	148
Desarrollo de la aplicación móvil para la entrega de medicamentos desde la metodología Scrum .....	149
Tipo de Prototipo.....	150
Product Backlog .....	150
Módulo de Autenticación y Usuarios .....	151
Módulo de Solicitudes.....	152
Módulo de Medicamentos .....	152
Módulo de Entregas.....	153
Módulo de Notificaciones .....	153
Desarrollo Iterativo del Proyecto (Sprints) .....	154

Sprint Planning .....	154
Evaluación de la funcionalidad y usabilidad del producto mínimo viable .....	170
Resultados .....	180
Conclusiones y Recomendaciones .....	182
Conclusiones .....	182
Recomendaciones.....	183
Referencias Bibliográficas .....	184
Apéndices.....	201
Apéndice A.....	201
Apéndice B.....	207
Apéndice C.....	208
Apéndice D.....	209

## Lista de Tablas

<b>Tabla 1</b>	Antecedentes .....	26
<b>Tabla 2</b>	Conversiones de la formula estadística.....	73
<b>Tabla 3</b>	Actores y roles del proceso de entrega de medicamentos.....	79
<b>Tabla 4</b>	Segmentación de público Objetivo .....	80
<b>Tabla 5</b>	Problemas y dificultades del proceso actual .....	88
<b>Tabla 6</b>	Entradas del Proceso .....	90
<b>Tabla 7</b>	Salidas del Proceso .....	90
<b>Tabla 8</b>	Actores Involucrados .....	90
<b>Tabla 9</b>	Cuellos de Botella Identificados .....	91
<b>Tabla 10</b>	Dolores (Pains) y necesidades de usuario.....	92
<b>Tabla 11</b>	Requisitos del sistema.....	93
<b>Tabla 12</b>	Propuesta Estado Futuro (TO-BE).....	94
<b>Tabla 13</b>	MoSCoW .....	96
<b>Tabla 14</b>	Técnica MoSCoW.....	96
<b>Tabla 15</b>	Épicas del proyecto .....	97
<b>Tabla 16</b>	Drivers arquitecturales .....	115
<b>Tabla 17</b>	Atributos de Calidad .....	116
<b>Tabla 18</b>	Vista de seguridad.....	132
<b>Tabla 19</b>	Capa de Infraestructura .....	134
<b>Tabla 20</b>	Relaciones entre Entidades .....	140
<b>Tabla 21</b>	Integración y APIs .....	141

<b>Tabla 22</b>	Estrategia de Pruebas .....	144
<b>Tabla 23</b>	Decisiones Arquitectónicas (ADR).....	145
<b>Tabla 24</b>	Estrategia de Despliegue.....	146
<b>Tabla 25</b>	Módulo de Autenticación y Usuarios .....	151
<b>Tabla 26</b>	Módulo de Solicitudes .....	152
<b>Tabla 27</b>	Módulo de Medicamentos.....	152
<b>Tabla 28</b>	Módulo de Entregas .....	153
<b>Tabla 29</b>	Módulo de Notificaciones.....	153
<b>Tabla 30</b>	Sprint 1- Autenticación y usuarios.....	154
<b>Tabla 31</b>	Sprint 1 - Sprint Review .....	158
<b>Tabla 32</b>	Sprint 2-Solicitudes.....	159
<b>Tabla 33</b>	Sprint 2 - Sprint Review .....	161
<b>Tabla 34</b>	Sprint 3-Medicamentos .....	162
<b>Tabla 35</b>	Sprint 3 - Sprint Review .....	162
<b>Tabla 36</b>	Sprint 4- Entregas .....	163
<b>Tabla 37</b>	Sprint 4-Sprint Review .....	164
<b>Tabla 38</b>	Sprint 5- Notificaciones .....	165
<b>Tabla 39</b>	Sprint 5 - Sprint Review .....	166
<b>Tabla 40</b>	Sprint 6 - Auditoria y seguridad.....	167
<b>Tabla 41</b>	Sprint 6-Sprint Review .....	168

## Lista de Figuras

<b>Figura 1</b>	Encuesta semiestructurada .....	81
<b>Figura 2</b>	Rango de edad del usuario.....	82
<b>Figura 3</b>	Rango de Tiempo que toma llegar al punto de reclamación .....	83
<b>Figura 4</b>	¿El tiempo de demora es adecuado?.....	83
<b>Figura 5</b>	Tiempo de espera para recibir los medicamentos .....	84
<b>Figura 6</b>	¿Padece discapacidad física o movilidad reducida?.....	84
<b>Figura 7</b>	¿Solicita con frecuencia sus medicamentos?.....	85
<b>Figura 8</b>	¿Puede algún familiar reclamar los medicamentos? .....	85
<b>Figura 9</b>	¿Está satisfecho con el servicio prestado por el punto de reclamación? .....	86
<b>Figura 10</b>	¿Cree que una aplicación móvil que facilite la entrega de medicamentos a domicilio mejora el servicio que se presta actualmente? .....	86
<b>Figura 11</b>	¿ Puede acceder fácilmente a aplicaciones en dispositivos tecnológicos como celular, Tablet o computador? ”.....	87
<b>Figura 12</b>	Proceso en su Estado Actual .....	89
<b>Figura 13</b>	Flujo del proceso de Aplicación para entrega de medicamentos MediExpress .....	95
<b>Figura 14</b>	Formato tarjeta de Historia de Usuario .....	99
<b>Figura 15</b>	HU-001 Registro de usuario básico.....	100
<b>Figura 16</b>	HU-002 Inicio de sesión seguro .....	100
<b>Figura 17</b>	HU-003 Recuperar contraseña .....	101
<b>Figura 18</b>	HU-004 Editar perfil .....	101
<b>Figura 19</b>	Solicitar medicamento .....	102
<b>Figura 20</b>	Adjuntar foto de fórmula.....	102

<b>Figura 21</b>	Ver estado de solicitudes .....	103
<b>Figura 22</b>	Cancelar solicitud .....	103
<b>Figura 23</b>	Confirmación de EPS. ....	104
<b>Figura 24</b>	Gestión del catálogo de medicamentos .....	104
<b>Figura 25</b>	Ver descripción de medicamento .....	105
<b>Figura 26</b>	Asignar mensajero .....	105
<b>Figura 27</b>	Ver entregas asignadas .....	106
<b>Figura 28</b>	Actualizar estado de entrega.....	106
<b>Figura 29</b>	Firma digital. ....	107
<b>Figura 30</b>	Ver ubicación del usuario.....	107
<b>Figura 31</b>	Notificaciones por cambio de estado.....	108
<b>Figura 32</b>	Alertas masivas.....	108
<b>Figura 33</b>	Recordatorios de medicamentos.....	109
<b>Figura 34</b>	Modelo (Mongo-Express-Angular-Node).....	111
<b>Figura 35</b>	Diagrama de Contexto.....	118
<b>Figura 36</b>	Diagrama de Contenedores .....	120
<b>Figura 37</b>	Diagrama de Componentes .....	122
<b>Figura 38</b>	Diagrama Vistas adicionales (Rozanski & Woods) .....	126
<b>Figura 39</b>	Vista de Información / Datos.....	128
<b>Figura 40</b>	Vista de Despliegue.....	129
<b>Figura 41</b>	Vista de Seguridad.....	131
<b>Figura 42</b>	Arquitectura Física (Infraestructura) .....	135
<b>Figura 43</b>	Diagrama ERD MongoDB .....	140

<b>Figura 44</b> Login usuario.....	155
<b>Figura 45</b> Dashboard Usuario.....	155
<b>Figura 46</b> Login operador.....	156
<b>Figura 47</b> Dashboard Operador.....	156
<b>Figura 48</b> Login Incorrecto usuario.....	157
<b>Figura 49</b> Login Incorrecto operador.....	157
<b>Figura 50</b> Solicitud medicamento.....	160
<b>Figura 51</b> Solicitud enviada.....	160
<b>Figura 52</b> Ingreso medicamentos.....	162
<b>Figura 53</b> Asignación entregas.....	164
<b>Figura 54</b> Panel Notificaciones.....	166
<b>Figura 55</b> Prueba de Funcionalidad - Autenticación.....	170
<b>Figura 56</b> Prueba de Funcionalidad – Solicitudes.....	172
<b>Figura 57</b> Prueba de Funcionalidad -Inventario Medicamentos.....	174
<b>Figura 58</b> Prueba de Funcionalidad – Entregas.....	176
<b>Figura 59</b> Prueba de Funcionalidad – Notificaciones.....	178

## Lista de Apéndices

<b>Apéndice A</b> Formulario Encuesta de análisis sobre el servicio de atención en el punto de reclamación del barrio El Carmen-Tunjuelito .....	201
<b>Apéndice B</b> Visión del proyecto .....	207
<b>Apéndice C</b> Enlace de acceso a aplicación MediExpress .....	208
<b>Apéndice D</b> Manual de Usuario MediExpress .....	209

## Introducción

En el barrio El Carmen de la localidad de Tunjuelito, Bogotá, los adultos mayores y las personas con movilidad reducida enfrentan barreras significativas para acceder oportunamente a sus medicamentos. Las largas filas en los puntos de dispensación, que pueden superar las dos horas, junto con la exposición a riesgos de seguridad, fallas en los sistemas de autorización y condiciones climáticas adversas, generan interrupciones en los tratamientos y un deterioro en la calidad de vida de esta población vulnerable.

Ante esta problemática, el presente proyecto propone el diseño e implementación de una aplicación móvil orientada a optimizar el proceso de solicitud y entrega domiciliaria de medicamentos. La solución busca mejorar la accesibilidad al servicio, reducir las barreras físicas asociadas al desplazamiento y contribuir a la continuidad terapéutica mediante el uso de herramientas digitales adaptadas a las necesidades de los usuarios.

El proyecto se enmarca en un enfoque de investigación aplicada con diseño mixto, que integra encuestas dirigidas a usuarios para identificar requerimientos funcionales y no funcionales del sistema. La gestión del desarrollo se fundamenta en la metodología ágil Scrum, estructurada en sprints iterativos que permitieron construir y validar un Producto Mínimo Viable (MVP) con retroalimentación continua.

El desarrollo técnico se realizó utilizando Angular para el frontend, Node.js para la lógica del servidor y MongoDB como sistema de base de datos, bajo principios de arquitectura escalable y segura. A través de esta propuesta, se busca aportar una solución tecnológica contextualizada que pueda mejorar la experiencia de los usuarios y servir como modelo replicable en entornos comunitarios con características similares.

## Justificación

El acceso oportuno a medicamentos constituye un factor determinante en la calidad de vida de los adultos mayores y de las personas con movilidad reducida, quienes enfrentan múltiples dificultades para desplazarse hasta los puntos de reclamación. Actualmente, el proceso de entrega en estos lugares presenta fallas recurrentes: largas filas que superan varias horas, insuficiencia de personal, fallos en los sistemas de autorización y demoras en la dispensación. Estas limitaciones afectan directamente la adherencia a los tratamientos y generan un impacto negativo en la salud y el bienestar de la población vulnerable.

Ante este panorama, surge la necesidad de implementar una solución tecnológica innovadora que optimice la distribución de medicamentos mediante una aplicación móvil desarrollada en Angular y gestionada bajo la metodología ágil Scrum. Esta herramienta permitirá a los usuarios del barrio El Carmen solicitar medicamentos a domicilio, reduciendo los riesgos asociados al desplazamiento, mejorando la experiencia del paciente y fortaleciendo la gestión de los recursos disponibles, al evitar pérdidas por retrasos o entregas inadecuadas.

Más allá de su utilidad práctica, este proyecto también tiene un aporte académico y científico, ya que valida la aplicación de metodologías ágiles y tecnologías móviles en un entorno de salud comunitaria. La integración de Angular y Scrum en el desarrollo de un Producto Mínimo Viable (MVP) contribuye a generar conocimiento aplicado, que puede ser documentado y replicado en otros contextos con características similares. De este modo, la iniciativa no solo responde a una problemática local, sino que se proyecta como un modelo replicable para comunidades vulnerables en diferentes regiones del país e incluso en entornos internacionales con necesidades análogas.

En síntesis, esta propuesta combina innovación tecnológica, pertinencia social y valor académico, al tiempo que fortalece la calidad de vida de los usuarios y aporta al cuerpo de conocimiento sobre el uso de metodologías ágiles y aplicaciones móviles en salud pública.

## Planteamiento del Proyecto

El barrio El Carmen, ubicado en la localidad de Tunjuelito, Bogotá, enfrenta una problemática recurrente relacionada con la entrega de medicamentos a su población más vulnerable. En particular, los adultos mayores y las personas con movilidad reducida deben desplazarse largas distancias hasta los puntos de dispensación, lo que implica exponerse a condiciones climáticas adversas, riesgos de seguridad en la vía pública y prolongados tiempos de espera (DANE, 2021; Personería de Bogotá, 2025).

Estas dificultades no solo afectan el acceso oportuno a los tratamientos, sino que también inciden en el bienestar físico y emocional de los usuarios, al generar desgaste, ansiedad y, en muchos casos, abandono del tratamiento. A lo anterior se suma la existencia de fallas en las plataformas de autorización y la insuficiencia de personal en las farmacias de las EPS, lo cual intensifica la congestión y disminuye la calidad del servicio (Defensoría del Pueblo, 2023; El Tiempo, 2024).

El escenario descrito evidencia la necesidad de contar con mecanismos alternativos de distribución que prioricen la atención de quienes presentan mayores barreras de acceso. En este marco, se plantea el desarrollo de una solución tecnológica segura y escalable que optimice la logística de entrega de medicamentos, reduzca la dependencia de la asistencia presencial y ofrezca una experiencia de usuario más ágil, inclusiva y humanizada (MinSalud, 2020; OMS, 2020).

La población objetivo de adultos mayores en Tunjuelito y, por ende, barrio El Carmen, es considerable. De acuerdo con el DANE (2014), en la localidad de Tunjuelito, que alberga aproximadamente 200,725 habitantes, cerca del 11,5 % corresponde a personas de 60 años o

más, es decir, unos 22,990 adultos mayores. Además, el índice de envejecimiento en Tunjuelito supera al promedio de la ciudad, con 57 personas mayores de 60 años por cada 100 menores de 14 años, frente al índice general de Bogotá de 56.

En términos de atención, en 2022, la Subdirección para la Vejez de la Secretaría Distrital de Integración Social atendió a 3.917 personas mayores en toda Tunjuelito, lo que confirma la concentración y necesidad de atención en esta población

Según datos suministrados por la página oficial del barrio Tunjuelito, habitan aproximadamente 22.571 personas mayores de 60 años, lo que equivale al 9.1 % de la población local y supera la tasa de envejecimiento promedio de Bogotá (8.8 %). Esta cifra representa una proporción significativa de población vulnerable que enfrenta barreras físicas y sociales para acceder a servicios esenciales.

Simultáneamente, la ciudad atraviesa una crisis de acceso a medicamentos. Las largas filas y el desabastecimiento se han vuelto constantes, particularmente en el sur de Bogotá, donde cientos de adultos mayores llegan desde la madrugada para reclamar tratamientos críticos, muchas veces sin éxito

Actualmente el punto de reclamación de medicamentos del barrio El Carmen enfrenta una dificultad, especialmente para personas de la tercera edad y/o con movilidad reducida, en el cual deben realizar filas de espera bastante agotadoras con tiempos de 2 a 3 horas diariamente. Además del tiempo de espera, los adultos mayores deben salir de casa en la madrugada y recorrer trayectos largos, lo que los expone a riesgos de seguridad en el camino.

Por otro lado, el sistema que se maneja en el punto de reclamación presenta caídas constantes, lo que genera que los pacientes pierdan tiempo esperando hasta que se restablece la atención. Con lo anterior, se debe analizar e implementar una solución tecnológica que permita una entrega de medicamentos viable, segura y cómoda para los usuarios más vulnerables que deben asistir a dicho establecimiento.

La magnitud del problema también queda reflejada en el número de quejas registradas: de acuerdo con datos obtenidos de la Personería de Bogotá, en el primer trimestre de 2025, la entidad recibió 791 solicitudes relacionadas con dificultades en la entrega de medicamentos, lo que representa el 42 % de las quejas en salud.

La urgencia de resolver esta situación radica en que los usuarios vulnerables se ven expuestos a riesgos de seguridad personal, afectaciones en su salud física por las largas esperas y, en algunos casos, a condiciones ambientales adversas como el frío o la lluvia, mientras esperan ser atendidos.

Ante esta problemática, esta propuesta busca responder al siguiente interrogante principal: ¿De qué manera una aplicación móvil puede mejorar el proceso de entrega de medicamentos y salvaguardar el bienestar de personas adultas mayores y con movilidad reducida en el barrio El Carmen, localidad de Tunjuelito – Bogotá?

## **Objetivos**

### **Objetivo General**

Implementar una aplicación móvil con Angular para asistir la entrega de medicamentos a adultos mayores en el barrio El Carmen (Tunjuelito – Bogotá), en un periodo de cinco meses, con el fin de optimizar su acceso oportuno y seguro al tratamiento médico.

### **Objetivos Específicos**

Analizar el proceso actual de entrega de medicamentos a adultos mayores y personas con movilidad reducida en el barrio El Carmen (Tunjuelito – Bogotá), mediante entrevistas y observación, para identificar requisitos funcionales y no funcionales en un periodo de dos semanas.

Diseñar la arquitectura de la aplicación móvil a través de principios de escalabilidad, mantenibilidad y modularidad, con el fin de optimizar el proceso de entrega de medicamentos en el barrio El Carmen, durante la fase inicial de desarrollo del proyecto.

Desarrollar la aplicación móvil utilizando el framework Angular, bajo un ciclo de trabajo ágil con Scrum, para construir un producto mínimo viable que optimice la entrega de medicamentos a adultos mayores en el barrio El Carmen, durante un periodo de ocho semanas.

Evaluar la funcionalidad básica implementada y la usabilidad inicial de la aplicación móvil, mediante pruebas exploratorias y retroalimentación de usuarios objetivo, con el fin de comprobar la pertinencia del producto mínimo viable frente a las necesidades identificadas, durante las dos semanas finales del desarrollo.

## **Marco Referencial**

El presente marco referencial tiene como finalidad establecer los fundamentos conceptuales, teóricos, normativos y contextuales que sustentan el desarrollo del proyecto. Su propósito es brindar una base sólida que permita comprender el problema identificado en la entrega de medicamentos a domicilio y orientar la construcción de una solución tecnológica coherente con las necesidades de la población objetivo. En este sentido, se incluyen antecedentes sobre la problemática en el sector salud, teorías vinculadas a la usabilidad y accesibilidad de aplicaciones móviles, modelos logísticos de distribución y marcos regulatorios relacionados con la protección de datos en el ámbito sanitario.

### **Antecedentes**

Con el fin de identificar tendencias, soluciones existentes y vacíos tecnológicos asociados al problema de la entrega de medicamentos, se realizó un análisis documental de investigaciones y desarrollos relacionados con la salud digital. Esta revisión de la literatura existente recopila experiencias internacionales y nacionales sobre el uso de aplicaciones móviles, telemedicina y logística farmacéutica apoyada en tecnologías digitales, así como las metodologías y framework más utilizados, entre ellos Angular y metodología Scrum. La revisión permitió reconocer avances significativos en otros contextos y, al mismo tiempo evidencia la falta de soluciones integradas en Colombia que respondan a las necesidades de adultos mayores y personas con movilidad reducida, lo que sustenta la pertinencia del proyecto propuesto.

**Tabla 1**  
*Antecedentes*

<b>Título / Autores (Año)</b>	<b>Tecnologías / Metodología</b>	<b>Alcance</b>	<b>Limitaciones</b>	<b>Aprendizajes Relevantes</b>
Implementación de apps para distribución comunitaria de medicamentos (World Health Organization, 2021)	Aplicaciones Android; GPS; bases de datos móviles; sincronización offline.	Pilotos rurales para reducir tiempos de entrega y mejorar trazabilidad.	Conectividad intermitente; dependencia de personal local; escalabilidad limitada.	La sincronización offline y el GPS optimizan rutas en contextos con red inestable.
Integración de e-prescription con logística especializada DocMorris (2023); (Atlassian, 2026)	Plataformas de prescripción electrónica; APIs; trazabilidad y entrega logística.	Implementaciones nacionales que integran receta electrónica con logística y entrega a domicilio.	Modelos regulados con alta inversión; replicabilidad limitada en sistemas fragmentados.	La integración entre e-prescription y logística reduce errores, pero requiere interoperabilidad EPS.
Programas mHealth para coordinación y dispensación WHO (2021).	Plataformas mHealth; SMS/app; monitoreo de cadena de frío.	Proyectos rurales para mejorar cobertura y reducir pérdidas de medicamentos sensibles.	Necesidad de infraestructura y personal local; interoperabilidad limitada.	La coordinación digital y trazabilidad son efectivas si hay resiliencia ante fallas de red.
Entrega domiciliaria por EPS durante pandemia MinSalud (2020).	Canales telefónicos; apps privadas; gestión temporal de domicilios.	Intervenciones locales para garantizar continuidad terapéutica en COVID-19.	Falta de estandarización; baja integración EPS; diseño poco adaptado a adultos mayores.	Evidencia la necesidad de soluciones accesibles e integradas centradas en el usuario.
Implementación de Agile/Scrum en entornos sanitarios (Atlassian, 2026).	Scrum; sprints; validación iterativa con usuarios.	Aplicación de metodologías ágiles en servicios de salud domiciliaria.	Requiere compromiso institucional y adaptación cultural.	Scrum mejora la alineación con necesidades reales y acelera entregas iterativas.
Usabilidad de aplicaciones	Evaluaciones de usabilidad;	Estudios urbanos con enfoque en	Poca atención a adultos mayores;	Adaptar UI/UX (texto grande,

móviles para pedidos a domicilio Interaction Design Foundation, s. f.	métricas SUS; pruebas con usuarios.	apps de entrega y experiencia de usuario.	usuarios con alta alfabetización digital.	botones amplios) mejora la aceptación y usabilidad.
Logística farmacéutica y última milla en Colombia. (Ministerio de Salud y Protección Social, 2020).	Análisis BPM; optimización de rutas; modelos de costos.	Diagnósticos nacionales y propuestas para optimizar distribución urbana.	Poca implementación piloto; falta de articulación con actores del sistema de salud.	Las estrategias BPM son transferibles a apps locales y refuerzan el uso de Scrum.

Nota. Elaboración propia a partir de WHO (2021), MinSalud (2020), Atlassian, 2026, DocMorris (2023)

### **Aplicaciones Móviles Orientadas a la Entrega de Medicamentos**

La literatura evidencia múltiples aplicaciones móviles desarrolladas para optimizar la entrega de medicamentos. En India, proyectos de salud digital han demostrado mejoras en la distribución comunitaria mediante aplicaciones de registro y seguimiento (WHO (2021)). En África, programas apoyados por organizaciones internacionales han utilizado mHealth para coordinar la dispensación en comunidades rurales, reduciendo tiempos de espera y pérdidas en la cadena de frío (WHO, 2021). En Europa, iniciativas como DocMorris en Suiza integran plataformas de e-prescription con servicios logísticos especializados, constituyendo referentes de éxito en la combinación de tecnología y distribución farmacéutica (DocMorris, 2023).

En América Latina, países como Chile y México han impulsado soluciones de telemedicina y farmacias digitales, aunque con coberturas aún limitadas a áreas urbanas. En Colombia, durante la pandemia se implementaron programas de entrega domiciliaria de medicamentos por parte de las EPS, con resultados positivos en continuidad de tratamientos, pero sin un sistema estandarizado de integración tecnológica (MinSalud, 2020). Asimismo, existen aplicaciones privadas de farmacias que permiten realizar pedidos en línea, aunque

carecen de integración con los procesos de autorización de las EPS y de un enfoque adaptado a adultos mayores con movilidad reducida.

### **Tecnologías Utilizadas: Angular y Metodologías Ágiles (Scrum)**

El framework Angular, desarrollado por Google, se ha consolidado como una herramienta robusta para la creación de aplicaciones web y móviles progresivas (PWA). Destaca por su arquitectura modular, uso de TypeScript, escalabilidad, soporte de accesibilidad (WAI-ARIA) y facilidad de integración con APIs y sistemas de información en salud (Angular, 2024; Google Developers, 2023). En proyectos de salud digital, Angular ha sido utilizado para desarrollar soluciones interoperables y accesibles, lo que respalda su viabilidad técnica en contextos de innovación social (NgRx, 2023; W3C, 2023).

### **Metodología Scrum en Proyectos de Software Aplicado**

La metodología ágil Scrum ha sido aplicada en proyectos de salud digital y educación con el fin de promover entregas iterativas, adaptabilidad a cambios y participación activa de los usuarios en el desarrollo de soluciones. Estudios recientes destacan que Scrum mejora la calidad del software y fortalece la alineación con las necesidades reales de los usuarios, lo que resulta particularmente relevante en contextos de intervención social y comunitaria ( Kessler et al., 2022).

### **Brecha Identificada y Aporte del Proyecto**

Si bien existen experiencias internacionales exitosas en Europa y avances en Colombia en la entrega domiciliaria de medicamentos, aún no se ha documentado ni validado en el país una

solución tecnológica integrada que combine autorización EPS, gestión logística y aplicación móvil accesible. Las aplicaciones privadas actuales no están adaptadas a las condiciones de adultos mayores ni a usuarios con movilidad reducida, y la evidencia académica sobre apps desarrolladas con Angular en salud pública en Colombia es prácticamente inexistente.

Este vacío representa una oportunidad de innovación social y tecnológica: la implementación de un Producto Mínimo Viable (MVP) con Angular bajo la metodología Scrum, validado en un entorno comunitario (barrio El Carmen, Bogotá), lo cual puede generar conocimiento aplicado y replicable para el sistema de salud nacional.

Después de realizada la revisión de antecedentes, se identificó que, en el uso de tecnologías móviles y seguimiento geoespacial, la mayoría de los proyectos analizados emplean aplicaciones con funciones de geolocalización para coordinar la entrega y optimizar rutas (DocMorris, WHO). Los estudios revisados evidencian un beneficio directo en la continuidad del tratamiento, ya que las iniciativas de entrega domiciliaria de medicamentos han demostrado reducir el abandono terapéutico, un hallazgo consistente en experiencias desarrolladas en India, África y en los pilotos implementados por las EPS en Colombia. Asimismo, se resalta el valor de aplicar metodologías ágiles e iterativas como Scrum, las cuales permiten una mayor alineación con las necesidades de los usuarios, favorecen la adaptabilidad en entornos clínicos y mejoran la eficiencia en la implementación de soluciones tecnológicas en salud.

No obstante, persisten vacíos y limitaciones recurrentes, entre ellos la falta de integración con los procesos de autorización de las EPS, ya que en Colombia las aplicaciones privadas y los pilotos de entrega domiciliaria rara vez se articulan con estos sistemas, generando fricciones operativas (MinSalud, 2020). También se observa una escasa adaptación de las soluciones

tecnológicas a las necesidades de los adultos mayores, con pocas evidencias de interfaces accesibles o asistencia por voz, lo cual ha sido señalado como una brecha en los estudios de usabilidad.

Adicionalmente, se identifican problemas de interoperabilidad y ausencia de estándares unificados, lo que contrasta con las experiencias europeas donde la integración entre la prescripción electrónica y la logística farmacéutica ha resultado clave para el éxito de las plataformas. En los contextos rurales, la resiliencia ante fallas de conectividad sigue siendo un reto, destacándose la necesidad de mecanismos de sincronización offline que aseguren la continuidad del servicio. Finalmente, aunque existen numerosos pilotos operativos, aún son limitadas las evaluaciones formales y publicaciones académicas que documenten aprendizajes replicables en el contexto colombiano.

Se evidencia que, en Colombia, al no existir soluciones que permitan optimizar la entrega de medicamentos a personas vulnerables y con el objetivo de suplir las necesidades no resueltas identificadas en la revisión, es necesario integrar en una sola solución la autorización EPS, la gestión logística y una aplicación móvil accesible, garantizando un flujo digital completo para la entrega de medicamentos. Asimismo, se propone diseñar la experiencia de usuario (UX) específicamente para adultos mayores, incorporando elementos de accesibilidad como texto ampliado, asistencia por voz, procesos simplificados y opciones de apoyo familiar, aspectos poco abordados en la literatura nacional. Finalmente, se plantea probar un Producto Mínimo Viable en un entorno comunitario, registrando métricas de desempeño como tiempos de entrega, y tasas de error, con el fin de generar evidencia aplicable y replicable en el contexto del sistema de salud colombiano.

Este proyecto se inspira en las siguientes literaturas, las cuales se tomaron como base de

inspiración para el proyecto MediExpress; WHO (2021) recomienda implementar la sincronización offline y geolocalización para rutas eficientes; DocMorris sugiere adoptar un enfoque de integración entre prescripción electrónica y logística, aunque a menor escala y ajustado al contexto EPS; Atlassian (2026) estableció el uso de Scrum para validar iterativamente funcionalidades con usuarios y EPS; Interaction Design Foundation plantearon priorizar pruebas de usabilidad con adultos mayores; el Ministerio de Salud y Protección Social, (2020) con su estudio en logística, propone aplicar BPM para modelar AS-IS/TO-BE y optimizar la “última milla”.

Para concluir, la revisión de antecedentes mostró avances claros en la digitalización de la entrega de medicamentos (plataformas móviles, e-prescription y logística), pero se identificaron vacíos críticos en el contexto colombiano: falta de integración técnica con procesos de autorización de las EPS, ausencia de soluciones adaptadas a adultos mayores y carencia de evaluaciones replicables a nivel comunitario (MinSalud, 2020; DocMorris, 2023).

Dado lo anterior, el desarrollo de un Producto Mínimo Viable que combine integración con autorización EPS; gestión logística local (última milla) y una interfaz centrada en adultos mayores (accesibilidad y usabilidad), responde de forma directa a las brechas identificadas. La evidencia revisada respalda la eficacia técnica de los componentes (geolocalización, sincronización offline, prácticas de seguridad y escalabilidad), mientras que la carencia de implementaciones integradas y evaluadas localmente justifica la pertinencia académica y social del proyecto.

En síntesis, la literatura y los casos revisados avalan la viabilidad técnica y el impacto potencial de una solución integrada; simultáneamente, muestran que ningún antecedente documentado en Colombia cumple las tres dimensiones (autorización EPS + logística integrada +

accesibilidad para adultos mayores). Por tanto, el proyecto propuesto llena un vacío tecnológico y social relevante y genera conocimiento aplicable y replicable para el sistema de salud nacional.

## **Marco Teórico**

El marco teórico constituye la base conceptual sobre la cual se sustenta el presente proyecto de desarrollo de software. En este apartado se recopilan, analizan y articulan teorías, enfoques y modelos vinculados a la ingeniería de software, la usabilidad, la accesibilidad, la logística de distribución y las metodologías ágiles de desarrollo. Estos referentes constituyen la base para orientar las decisiones relacionadas con el diseño, desarrollo e implementación, así como con la validación de la solución tecnológica propuesta, asegurando su coherencia con los estándares técnicos, las necesidades de los usuarios y los objetivos del proyecto. De esta forma, se garantiza que el proyecto mantenga alineación con los avances más recientes en salud digital (mHealth) y con las buenas prácticas de la computación aplicada a la atención sanitaria, promoviendo soluciones innovadoras, seguras y centradas en el usuario.

A continuación, se presentan los principales patrones, técnicas y teorías que fundamentan este proyecto:

### ***Patrones de Diseño y Técnicas de Gestión de Dependencias***

Son elementos fundamentales en la arquitectura de software moderna, pues garantizan la modularidad, escalabilidad y mantenibilidad del sistema. Entre ellos destacan la Inyección de Dependencias (Dependency Injection) y los patrones Repository, ampliamente utilizados en aplicaciones empresariales y web.

**Inyección de Dependencias (DI).** Consiste en un principio de diseño que promueve la inversión de control (IoC), permitiendo que los objetos reciban sus dependencias desde el exterior en lugar de crearlas internamente. Esto reduce el acoplamiento y mejora la capacidad de prueba y mantenimiento del software. Según Microsoft Learn (2023), la DI facilita la gestión de servicios y la reutilización de componentes en arquitecturas basadas en microservicios y aplicaciones web modernas.

**Patrón Repository.** Actúa como una capa intermedia entre la lógica de negocio y el acceso a datos, encapsulando las operaciones de persistencia y proporcionando una abstracción del almacenamiento (Evans, 2004). De este modo, la lógica del dominio se mantiene independiente de los detalles del motor de base de datos o del framework de persistencia utilizado (Baeldung, 2023). Este patrón contribuye a mejorar la separación de responsabilidades y favorece el cumplimiento del principio Single Responsibility.

**Principios de interacción y retroalimentación en UX.** Los principios de interacción y retroalimentación en el diseño de experiencia de usuario (UX) son esenciales para garantizar una comunicación efectiva entre el sistema y el usuario. De acuerdo con Nielsen (2020), la visibilidad del estado del sistema es uno de los pilares de la usabilidad, pues los usuarios deben ser informados oportunamente sobre las acciones realizadas, los cambios de estado y los resultados obtenidos. Esto permite mantener la confianza y el control durante la interacción.

Asimismo, Shneiderman (2016) destaca dentro de sus “Ocho reglas de oro” que los sistemas deben proporcionar retroalimentación informativa y consistente, asegurando que cada acción del usuario tenga una respuesta perceptible. Este principio mejora la comprensión del sistema y reduce la incertidumbre durante el uso.

Por su parte, el Interaction Design Foundation (2023) enfatiza que la retroalimentación

no debe limitarse a mensajes de error, sino también incluir confirmaciones visuales, auditivas o animaciones que indiquen progreso o éxito en las tareas. Estas respuestas inmediatas ayudan a crear una sensación de diálogo entre el usuario y la interfaz.

De igual modo, ICS UX Design Team (2022) plantea que la retroalimentación efectiva contribuye a la accesibilidad y al aprendizaje del usuario, ya que le permite comprender las consecuencias de sus acciones, fomentando una interacción más intuitiva.

En conclusión, los principios de interacción y retroalimentación en UX promueven sistemas transparentes, comprensibles y centrados en el usuario, donde cada acción tiene una respuesta clara y significativa que facilita la navegación, refuerza la confianza y mejora la experiencia general.

**Infraestructura y Backend.** La infraestructura de tecnologías de la información (TI) se refiere al conjunto de componentes físicos y lógicos que permiten el funcionamiento, la administración y la seguridad de los sistemas tecnológicos de una organización. Esta incluye servidores, redes, sistemas operativos, plataformas en la nube, almacenamiento y software necesario para soportar las aplicaciones y los datos corporativos (IBM, s. f.; Gartner, 2025). De acuerdo con el Instituto Nacional de Estándares y Tecnología (NIST), la infraestructura puede implementarse localmente o a través de servicios en la nube, bajo modelos como Infraestructura como Servicio (IaaS), con el fin de ofrecer flexibilidad, escalabilidad y disponibilidad de recursos tecnológicos (Mell & Grance, 2011).

Por su parte, el backend constituye la parte del sistema que opera en el lado del servidor, gestionando la lógica de negocio, la base de datos, la autenticación y la comunicación con el frontend. Es responsable del procesamiento y almacenamiento de datos, así como de la conexión

con servicios y APIs externas que garantizan el correcto funcionamiento de una aplicación web o móvil (MDN Web Docs, 2025). En términos generales, el backend actúa como el núcleo funcional que permite que las operaciones de una aplicación se ejecuten de manera eficiente, segura y estructurada.

**Lógica en el desarrollo de software.** La lógica en el desarrollo de software se refiere al conjunto de principios, estructuras y reglas que determinan el comportamiento de un programa y la secuencia de operaciones necesarias para alcanzar un resultado específico. Según IBM (2024), la lógica en programación es el proceso mediante el cual se definen las condiciones, decisiones y flujos de ejecución que permiten que una aplicación funcione correctamente y responda a los requerimientos del usuario. Esta lógica guía la forma en que el software procesa datos, ejecuta instrucciones y produce resultados coherentes.

De acuerdo con Thomas H. Cormen et al. (2022), la lógica de programación constituye la base del pensamiento computacional, ya que implica diseñar algoritmos estructurados y aplicar principios formales para la resolución eficiente de problemas. En la edición más reciente de *Introduction to Algorithms*, los autores resaltan la importancia de la correcta estructuración algorítmica, la optimización y el análisis del comportamiento del software como elementos fundamentales para garantizar eficiencia y escalabilidad.

Asimismo, según IEEE (2023), la calidad del software depende en gran medida de una lógica bien estructurada, ya que esta impacta directamente en la mantenibilidad, confiabilidad y rendimiento del sistema, aspectos esenciales en entornos de desarrollo colaborativo y metodologías ágiles.

En síntesis, la lógica en el desarrollo de software no solo representa la capacidad técnica para escribir código funcional, sino también la aplicación de un pensamiento estructurado y racional orientado a resolver problemas de manera eficiente, segura y escalable.

**Interfaz de usuario (UI) y experiencia de usuario (UX).** La Interfaz de Usuario (UI) y la Experiencia de Usuario (UX) son conceptos complementarios dentro del diseño digital, orientados a garantizar la interacción efectiva, accesible y satisfactoria entre el usuario y un sistema.

Según Interaction Design Foundation (2024), la UI se refiere a los elementos visuales e interactivos que conforman la superficie de un producto digital —como botones, menús, tipografía, colores e iconografía—, cuyo propósito es facilitar la comunicación entre el usuario y la aplicación. Por su parte, la UX comprende el conjunto de percepciones y respuestas emocionales del usuario al interactuar con un producto o servicio, abarcando la usabilidad, eficiencia, accesibilidad y satisfacción general (Nielsen Norman Group, 2023).

De acuerdo con Adobe XD Ideas (2024), mientras que la UI se centra en el diseño estético y la presentación, la UX se enfoca en la funcionalidad, la arquitectura de la información y el flujo de interacción. Ambos enfoques son esenciales en el desarrollo de software, ya que una interfaz bien diseñada contribuye a una experiencia de usuario intuitiva, coherente y significativa.

**Diseño centrado en el usuario (DCU).** El Diseño Centrado en el Usuario (DCU) es un enfoque metodológico del desarrollo de productos y servicios digitales que prioriza las necesidades, características y objetivos de los usuarios en todas las etapas del proceso de diseño. Según la Norma ISO 9241-210:2019, el DCU busca optimizar la experiencia de uso mediante la comprensión profunda del contexto, la participación activa de los usuarios y la evaluación continua de las soluciones propuestas. Este enfoque garantiza que el producto final sea accesible, eficiente y satisfactorio para quien lo utiliza (International Organization for Standardization [ISO], 2019).

De acuerdo con Nielsen Norman Group (2023), el diseño centrado en el usuario implica un proceso iterativo en el que las decisiones se basan en la investigación y la validación constante de la usabilidad, asegurando que el sistema responda efectivamente a las expectativas reales de los usuarios. Por su parte, Interaction Design Foundation (2024) señala que el DCU no se limita al diseño visual, sino que abarca la arquitectura de la información, la interacción y la accesibilidad, convirtiéndose en una práctica esencial para la creación de experiencias digitales inclusivas y efectivas.

**Diseño UI/UX interactivo.** El diseño UI/UX interactivo se entiende como el conjunto de prácticas, principios y técnicas orientadas a concebir interfaces digitales que faciliten interacciones significativas, eficientes y agradables entre las personas y los sistemas. Este enfoque integra tanto la Interfaz de Usuario (UI) —los elementos visuales y de interacción— como la Experiencia de Usuario (UX) —la percepción global, la usabilidad y la satisfacción al usar el producto—, procurando que cada interacción comunique estado, posibilidad de acción y resultado (Norman, 2013; Nielsen Norman Group, 2020).

Un diseño interactivo efectivo se apoya en principios claros: visibilidad del estado del sistema, retroalimentación inmediata, consistencia, control del usuario, prevención de errores y microinteracciones significativas. Estos principios ayudan a que el usuario entienda qué puede hacer, qué está ocurriendo y qué resultado esperar de sus acciones, reduciendo la carga cognitiva y aumentando la confianza en la aplicación (Nielsen Norman Group, 2020; Shneiderman, 2016).

El proceso de diseño UI/UX interactivo sigue un ciclo iterativo centrado en el usuario: investigación (personas, jornadas y contextos de uso), definición (arquitectura de información y flujos), ideación (wireframes y prototipos), evaluación (pruebas de usabilidad) y refinamiento. La norma ISO 9241-210 refuerza la necesidad de involucrar usuarios reales durante todo el proceso para garantizar que las soluciones respondan a necesidades reales y contextos concretos (ISO, 2019; Interaction Design Foundation, 2024).

En cuanto a técnicas y artefactos, el diseño interactivo hace uso de wireframes (baja a media fidelidad) para validar jerarquías y flujos, prototipos interactivos (media y alta fidelidad) para simular interacciones y patrones de diseño (componentes reutilizables) para mantener consistencia. Las pruebas con prototipos (pruebas de tareas, pruebas moderadas y pruebas remotas) permiten medir eficiencia, eficacia y satisfacción mediante métricas como tasa de éxito,

tiempo de tarea y puntuación SUS (System Usability Scale) (Adobe, 2024; Nielsen Norman Group, 2018).

La accesibilidad y la inclusión son requisitos no negociables: un diseño interactivo debe cumplir con recomendaciones de accesibilidad (por ejemplo, WCAG) y adoptar prácticas como contraste suficiente, navegación por teclado, texto alternativo, etiquetas claras y tamaños de objetivo apropiados, lo que mejora la usabilidad para todos los usuarios, incluidas las personas mayores y con discapacidad (W3C, 2018; ISO, 2019).

Las microinteracciones (pequeñas respuestas al usuario como animaciones, confirmaciones o sonidos) y la retroalimentación progresiva (por ejemplo, barras de progreso o estados de carga) elevan la calidad percibida y reducen la incertidumbre en operaciones que tardan más en completarse. Sin embargo, deben implementarse con moderación y consistencia para no distraer ni afectar el rendimiento (Norman, 2013; Google Developers, 2024).

Finalmente, el diseño interactivo debe alinearse con métricas de producto y negocio: KPIs de usabilidad (tasa de éxito, tiempo en tarea, SUS), métricas de adopción (retención, activación) y métricas de rendimiento (tiempos de respuesta). Integrar pruebas continuas (testing A/B, monitoreo de eventos y sesiones) permite iterar basándose en evidencia y priorizar mejoras que impacten a usuarios reales (Nielsen Norman Group, 2018; Google Developers, 2024).

### *Teorías de Diseño y Arquitectura de Software*

**Arquitecturas.** En el ámbito de la ingeniería de software, el término arquitectura hace referencia a la estructura fundamental de un sistema, conformada por sus componentes, sus relaciones entre sí y con el entorno, así como los principios y lineamientos que orientan su diseño y evolución (ISO/IEC/IEEE 42010:2011).

La arquitectura de software establece una visión de alto nivel que permite comprender cómo se organiza una aplicación en términos de capas, módulos y servicios. Según Sommerville (2011), una arquitectura bien definida no solo describe la organización interna del sistema, sino que también facilita la comunicación entre los actores involucrados, apoya la toma de decisiones de diseño y asegura la escalabilidad, mantenibilidad y seguridad de la solución.

Existen diversos enfoques arquitectónicos relevantes en proyectos de software aplicado:

**Arquitectura en capas(Back-end).** La arquitectura en capas es un patrón estructural ampliamente utilizado en el desarrollo de software que organiza el sistema en niveles jerárquicos, donde cada capa tiene responsabilidades específicas y se comunica únicamente con las capas adyacentes. Según Pressman y Maxim (2020), este enfoque permite mejorar la mantenibilidad, escalabilidad y reutilización del software, al separar claramente las funciones de presentación, lógica de negocio y acceso a datos.

De acuerdo con Microsoft Learn (2023), la arquitectura en capas se estructura comúnmente en cuatro niveles: presentación, aplicación, dominio y persistencia. Esta separación facilita el desarrollo modular y el trabajo en equipo, ya que cada capa puede ser desarrollada y probada de forma independiente.

Por su parte, Martin (2017) destaca que la arquitectura en capas promueve el principio de dependencia hacia el interior, donde las capas externas dependen de las internas, pero nunca, al contrario. Este principio es fundamental en arquitecturas limpias y orientadas a la sostenibilidad del código, garantizando que los cambios en la interfaz o en la infraestructura no afecten la lógica central del negocio.

Asimismo, Oracle (2024) explica que este modelo es ideal para sistemas empresariales complejos, ya que facilita la integración con servicios externos, bases de datos y aplicaciones distribuidas, manteniendo un flujo de información ordenado y seguro.

**Arquitectura Modular Monolítica (Front-end).** La arquitectura modular monolítica en el front-end es un enfoque de diseño en el cual la aplicación se construye como una única unidad desplegable (monolito), pero organizada internamente en módulos independientes y bien estructurados. Este modelo permite mantener una base de código centralizada sin sacrificar la separación de responsabilidades ni la escalabilidad interna del sistema.

Según Microsoft (s. f.), un monolito es una aplicación construida y desplegada como una sola unidad, lo que simplifica la gestión inicial del proyecto, aunque puede volverse complejo si no se organiza adecuadamente. En el contexto del front-end, esto significa que toda la interfaz de usuario se ejecuta dentro de una única aplicación, pero puede estructurarse en componentes y módulos desacoplados.

Por su parte, Angular (s. f.) promueve explícitamente la arquitectura modular, donde la aplicación se divide en módulos funcionales que encapsulan componentes, servicios y rutas. Este enfoque facilita la mantenibilidad, la reutilización del código y la escalabilidad progresiva del sistema sin necesidad de fragmentarlo en múltiples aplicaciones independientes.

Asimismo, Martin Fowler (2015) explica que el patrón monolítico puede ser una estrategia adecuada en etapas iniciales del desarrollo, siempre que exista una clara separación interna de responsabilidades. Un monolito bien estructurado puede evolucionar y escalar sin incurrir en la complejidad operativa temprana que implican los micro-frontends o arquitecturas distribuidas.

En síntesis, la arquitectura modular monolítica en el front-end combina la simplicidad de un despliegue único con los beneficios de la modularidad interna, permitiendo un desarrollo organizado, mantenible y escalable, especialmente en proyectos de tamaño pequeño y mediano o en fases iniciales de validación.

**Escalabilidad y Seguridad.** La escalabilidad en el desarrollo de software se refiere a la capacidad de un sistema para mantener o mejorar su rendimiento a medida que aumenta la carga de trabajo o el número de usuarios. Según Bass, Clements y Kazman (2021), un sistema escalable puede crecer de forma vertical (aumentando recursos de hardware en un solo servidor) o horizontal (añadiendo más servidores o instancias), sin comprometer su funcionalidad ni disponibilidad.

De acuerdo con Microsoft Learn (2023), la escalabilidad es un principio clave del diseño de software en la nube, permitiendo adaptar dinámicamente los recursos a las necesidades del sistema mediante servicios como load balancers o auto-scaling. Esto mejora la eficiencia operativa y optimiza los costos de mantenimiento.

Por otro lado, la seguridad en el software se define como el conjunto de mecanismos, prácticas y políticas destinadas a proteger la información, los servicios y los usuarios frente a accesos no autorizados, fallos o ataques cibernéticos. Según Stallings (2020), la seguridad del software abarca aspectos como la confidencialidad, integridad y disponibilidad (modelo CIA), que garantizan la confianza en los sistemas informáticos.

Asimismo, OWASP (2024) enfatiza que la seguridad debe integrarse desde las primeras etapas del ciclo de desarrollo, aplicando estrategias como la autenticación segura, el cifrado de datos, la validación de entradas y las pruebas de penetración. En entornos web y móviles, esto

resulta fundamental para prevenir vulnerabilidades comunes como inyecciones SQL, exposición de datos sensibles o ataques XSS.

### ***Teorías de Requisitos y Experiencia de Usuario***

La interacción humana-computador (HCI) constituye un eje fundamental en el diseño de sistemas orientados a la salud digital, ya que permite garantizar que la aplicación responda a las necesidades cognitivas y comunicativas de los usuarios finales. Bajo este enfoque, el proyecto se apoya en el Diseño Centrado en el Usuario (DCU), que orienta la definición de requisitos considerando las características de la población adulta mayor y usuarios con movilidad reducida, quienes requieren interfaces simples, accesibles y con retroalimentación clara (Norman, 2013; Nielsen, 1994).

De manera complementaria, se aplican principios cuantitativos provenientes de teorías clásicas de la usabilidad:

**Ley de Fitts.** Utilizada para optimizar el tamaño y la disposición de botones, de manera que las acciones más frecuentes requieran menos tiempo y esfuerzo motor.

**Ley de Hick.** Aplicada para limitar el número de opciones presentadas al usuario, reduciendo la carga cognitiva y agilizando la toma de decisiones dentro de la aplicación.

Estos lineamientos permiten diseñar un flujo de interacción que minimice errores, reduzca tiempos de respuesta y mejore la satisfacción del usuario en procesos críticos como la solicitud y confirmación de medicamentos.

### *Teorías de Procesos y Metodologías de Desarrollo*

**Metodología Ágil – Scrum.** Esta metodología, permite que, en la aplicación de entrega de medicamentos, su desarrollo sea iterativo y flexible, adaptándose a cambios rápidos, con ciclos cortos y efectivos. La asignación de roles es fundamental en esta metodología y permite que cada encargado aporte de manera importante en el desarrollo, por otro lado las actividades establecidas en la metodología ágil entregan información vital del avance del desarrollo del producto :

**Product Owner .** Es el responsable de maximizar el valor del producto que desarrolla el equipo de Scrum. Su función principal consiste en gestionar el Product Backlog, priorizando los requerimientos de acuerdo con las necesidades de los usuarios y los objetivos estratégicos del proyecto. Según la Guía Scrum (Scrum.org, 2020), el Product Owner asegura que el equipo desarrolle las funcionalidades de mayor impacto y valor, manteniendo una comunicación constante con los interesados (stakeholders) para alinear expectativas y resultados.

En proyectos de salud digital, como el presente, el Product Owner cumple un papel fundamental al actuar como puente entre la población beneficiaria (adultos mayores y personas con movilidad reducida), las instituciones de salud (EPS/IPS) y el equipo de desarrollo. De este modo, garantiza que la aplicación móvil no solo sea técnicamente viable, sino también pertinente y útil para la comunidad objetivo.

**Scrum Master.** Es el facilitador del marco de trabajo Scrum y actúa como guía para que el equipo de desarrollo y el Product Owner comprendan y apliquen correctamente los principios ágiles. Según la Guía Scrum (Scrum.org, 2020), el Scrum Master no dirige al equipo, sino que promueve la autogestión, elimina impedimentos y asegura que las prácticas de Scrum se implementen de manera efectiva.

Su labor también incluye fomentar la mejora continua mediante la facilitación de las reuniones clave (Daily Scrum, Sprint Review, Sprint Retrospective) y la creación de un entorno colaborativo en el que se priorice la transparencia, la inspección y la adaptación.

En proyectos de salud digital, como el presente, el rol del Scrum Master adquiere especial relevancia al garantizar que las iteraciones se desarrollen de manera eficiente, que los requerimientos de accesibilidad y usabilidad para adultos mayores sean incorporados en los ciclos de trabajo, y que el equipo mantenga el enfoque en los objetivos sociales y tecnológicos del proyecto.

**Equipo de desarrollo.** Está conformado por los profesionales responsables de construir el producto durante cada iteración o Sprint. Según la Guía Scrum (Scrum.org, 2020), este equipo es multifuncional y autoorganizado, lo que significa que posee todas las competencias necesarias para transformar los requerimientos priorizados en el Product Backlog en un incremento funcional del producto.

A diferencia de los modelos tradicionales de desarrollo, el Development Team no depende de jerarquías externas para distribuir tareas, sino que gestiona de manera autónoma la planificación, ejecución y verificación de cada entrega. Esta autonomía favorece la creatividad,

la responsabilidad compartida y la capacidad de adaptación frente a cambios en los requerimientos.

En el contexto del presente proyecto, el Development Team está integrado por desarrolladores de software especializados, gestores de bases de datos en MongoDB y profesionales encargados de la interfaz de usuario. Su rol es garantizar que la aplicación móvil cumpla con los criterios de usabilidad, accesibilidad y eficiencia logística requeridos para optimizar el proceso de entrega de medicamentos a adultos mayores y personas con movilidad reducida en el barrio El Carmen.

**Daily Scrum.** Es una reunión breve y diaria del equipo de desarrollo, con una duración máxima de 15 minutos. De acuerdo con la Guía Scrum (Scrum.org, 2020), su objetivo principal es inspeccionar el progreso hacia el objetivo del Sprint y ajustar el plan de trabajo de las próximas 24 horas. Durante esta reunión, cada integrante del Development Team comparte lo que realizó el día anterior, lo que planea realizar en la jornada actual y los posibles impedimentos que podrían afectar el cumplimiento de los objetivos.

Este evento promueve la transparencia y la comunicación constante dentro del equipo, permite identificar de manera temprana obstáculos técnicos u organizativos y fortalece la autogestión del grupo. A diferencia de reuniones de control tradicionales, el Daily Scrum no busca supervisar, sino coordinar y sincronizar esfuerzos para asegurar el avance continuo del proyecto.

En el contexto de este proyecto, los Daily Scrums resultan fundamentales para coordinar tareas relacionadas con el desarrollo en Angular, la integración de bases de datos en MongoDB y

la validación de requisitos de usabilidad para adultos mayores y personas con movilidad reducida. Además, permiten detectar oportunamente posibles retrasos en la entrega de funcionalidades que impactan directamente en la optimización del proceso logístico de distribución de medicamentos.

**Review.** La Sprint Review es un evento de Scrum que se realiza al finalizar cada Sprint, con el propósito de inspeccionar el incremento desarrollado y adaptar el Product Backlog si es necesario. Según la Guía Scrum (Scrum.org, 2020), en esta reunión participan el Product Owner, el Scrum Master, el Development Team y los principales stakeholders o interesados del proyecto, quienes colaboran para evaluar los avances y planear los próximos pasos.

Durante la Sprint Review, el Development Team presenta las funcionalidades completadas, el Product Owner informa sobre el estado del backlog y los objetivos alcanzados, y los interesados brindan retroalimentación. Esta interacción fomenta la transparencia y asegura que el producto en construcción se mantenga alineado con las necesidades reales de los usuarios y la comunidad.

En el caso de este proyecto, la Sprint Review permite validar con los representantes de la población adulta mayor y con las instituciones de salud (EPS/IPS) que la aplicación móvil cumple con los criterios de accesibilidad, usabilidad y eficiencia logística definidos. Asimismo, garantiza que los ajustes propuestos durante el ciclo de desarrollo se integren oportunamente, fortaleciendo la pertinencia social y tecnológica de la solución.

**Retrospectiva.** La Sprint Retrospective es el último evento de cada Sprint y tiene como propósito reflexionar sobre el proceso de trabajo con el fin de identificar oportunidades de mejora. Según la Guía Scrum (Scrum.org, 2020), en esta reunión participan el Scrum Master, el Product Owner y el Development Team, y se centra en analizar qué aspectos funcionaron bien, cuáles representaron dificultades y qué acciones concretas se implementarán en el siguiente Sprint para aumentar la eficacia y la calidad del producto.

Este evento fomenta la mejora continua, la colaboración interna y el fortalecimiento del equipo como unidad autoorganizada. Además, contribuye a que los equipos Scrum mantengan un ritmo sostenible, ajusten prácticas técnicas y refuercen la comunicación.

En el contexto de este proyecto, la Sprint Retrospective resulta clave para evaluar el trabajo realizado en torno al desarrollo en Angular, la integración con bases de datos MongoDB y la usabilidad de la aplicación móvil para adultos mayores con movilidad reducida. A partir de esta retroalimentación, se pueden introducir mejoras en el diseño, en la accesibilidad de las interfaces o en la coordinación logística, garantizando que cada iteración del producto se acerque más a las necesidades reales de la comunidad.

### ***Teorías de Calidad de Software***

Las teorías de calidad de software proporcionan los fundamentos conceptuales que orientan el diseño, la evaluación y la mejora continua de los sistemas informáticos. Estas teorías buscan garantizar que un producto de software cumpla con los requerimientos funcionales y no funcionales, ofreciendo un desempeño confiable, seguro y orientado al usuario (Pressman & Maxim, 2020).

**Calidad de Software.** La calidad de software se define como el grado en que un producto de software satisface los requerimientos funcionales y no funcionales establecidos, así como las expectativas del usuario (Pressman & Maxim, 2020). Este concepto no solo implica que el software funcione correctamente, sino que sea eficiente, confiable, mantenible, seguro y fácil de usar. De acuerdo con Sommerville (2020), la calidad es un atributo integral que abarca tanto las propiedades internas del código como la percepción del usuario final sobre el desempeño y la utilidad del sistema.

En la actualidad, la calidad de software está estrechamente vinculada con metodologías ágiles, la integración continua y la automatización de pruebas, que permiten detectar errores de forma temprana y mejorar la estabilidad del producto (Larman, 2019). En este contexto, la calidad no se percibe como un resultado final, sino como un proceso continuo de mejora a lo largo de todo el ciclo de vida del desarrollo.

### ***Teorías de Interoperabilidad en Salud***

La interoperabilidad en salud se refiere a la capacidad de los sistemas y aplicaciones sanitarias para intercambiar, interpretar y utilizar información de manera efectiva y segura, con el fin de mejorar la atención al paciente y optimizar los procesos clínicos (ISO/TC 215, 2021). Esta capacidad es fundamental en los ecosistemas de salud digital, donde múltiples instituciones, dispositivos y plataformas deben comunicarse entre sí sin perder integridad ni contexto en los datos.

De acuerdo con HIMSS (2020), la interoperabilidad se clasifica en cuatro niveles principales:

Fundacional, que permite la transmisión básica de datos entre sistemas.

Estructural, que garantiza el formato y la sintaxis coherente de los datos intercambiados.

Semántica, que asegura que la información tenga el mismo significado en diferentes contextos.

Organizacional, que involucra políticas, normas y acuerdos que facilitan la colaboración entre instituciones.

Desde una perspectiva teórica, la interoperabilidad en salud se apoya en estándares internacionales que definen estructuras comunes para el intercambio de información clínica. Entre estos, el más relevante en la actualidad es HL7 FHIR (Fast Healthcare Interoperability Resources), desarrollado por Health Level Seven International (HL7).

Según HL7 International (2023), FHIR combina los principios de las versiones anteriores de HL7 (V2 y V3) con tecnologías web modernas como RESTful APIs, JSON y XML, lo que facilita la integración de sistemas clínicos, aplicaciones móviles y plataformas en la nube. Su arquitectura basada en “recursos” permite representar entidades clínicas (como pacientes, diagnósticos, medicamentos o citas) de manera modular y estandarizada, promoviendo la reutilización y la flexibilidad.

Bender y Sartipi (2013) explican que HL7 FHIR mejora significativamente la interoperabilidad semántica al utilizar ontologías y terminologías clínicas estandarizadas (como SNOMED CT, LOINC y ICD-10), garantizando una interpretación uniforme de los datos médicos en distintos entornos tecnológicos. De esta manera, FHIR no solo permite el intercambio de información, sino que también impulsa la continuidad asistencial y la toma de

decisiones clínicas basadas en datos.

Asimismo, Mandl y Kohane (2022) destacan que FHIR representa una evolución hacia la interoperabilidad abierta, ya que permite a desarrolladores externos crear aplicaciones interoperables con sistemas hospitalarios mediante interfaces seguras, fomentando la innovación en la salud digital.

En síntesis, las teorías de interoperabilidad en salud , apoyadas en estándares como HL7 FHIR, constituyen un pilar esencial en la transformación digital del sector sanitario, ya que garantizan la comunicación efectiva entre sistemas, la integridad de los datos clínicos y la continuidad del cuidado del paciente.

### ***Teoría de la optimización de recursos***

Para maximizar la efectividad de un servicio, es necesario utilizar los recursos disponibles de la manera más eficiente posible, minimizando los tiempos de espera y maximizando la satisfacción del usuario (Pressman, 2014; Sommerville, 2011). Esta teoría es clave para justificar la implementación de una solución móvil que optimice los procesos de entrega de medicamentos.

### ***Mínimo Producto Viable (MVP)***

El Producto Mínimo Viable, por sus siglas en inglés MVP (Minimum Viable Product), consiste en el desarrollo y lanzamiento de una versión inicial y simplificada de la aplicación, cuyo propósito principal es validar hipótesis y recopilar retroalimentación directa de los usuarios (Ries, 2011). En el caso de la aplicación para la entrega de medicamentos, esta fase permitirá evaluar la usabilidad, la comprensión de las funciones, la pertinencia de la navegación y el nivel

de satisfacción de los adultos mayores y personas con movilidad reducida, quienes representan el público objetivo principal.

Implementar un MVP resulta fundamental para detectar puntos débiles, identificar necesidades reales, mejorar la experiencia de usuario y orientar los ajustes necesarios antes de realizar un despliegue a gran escala. A continuación, se presentan las características clave del MVP:

**Funcionalidad esencial.** El MVP se centra en resolver el problema principal previamente identificado.

**Retroalimentación valiosa.** El lanzamiento inicial posibilita la recolección de comentarios y sugerencias de los usuarios, lo que permite identificar qué funcionalidades cumplen con las expectativas y cuáles requieren ajustes, facilitando así la validación del producto y la mejora continua (Ries, 2011)

**Validación de la idea.** Esta fase facilita comprobar la viabilidad técnica y funcional del sistema, reduciendo riesgos asociados a la inversión y al desarrollo completo del producto (Atlassian, 2026).

**Aceleración del lanzamiento.** La implementación de un Producto Mínimo Viable (MVP) permite disponer de una versión funcional en etapas tempranas del desarrollo, facilitando una salida más rápida al mercado. Este enfoque no solo incrementa la competitividad del producto, sino que también posibilita generar impacto inmediato en el barrio El Carmen, al tiempo que abre oportunidades para su escalabilidad y expansión hacia otras localidades y ciudades (Atlassian, 2026).

En síntesis, el MVP constituye una estrategia ágil que asegura un contacto constante con

los usuarios y un proceso de mejora continua. Esto permite que la aplicación evolucione progresivamente, incorporando criterios de accesibilidad y respondiendo de manera flexible a las necesidades de la población adulta mayor y con movilidad reducida.

### *Otras Teorías*

Finalmente, es importante resaltar que, además de las teorías propias de la ingeniería de software, el marco teórico debe incorporar aquellas que se relacionan directamente con el enfoque particular del proyecto. En este caso, al tratarse de una solución orientada al ámbito de la salud digital y la logística de medicamentos, resulta fundamental integrar teorías y marcos de referencia provenientes de distintas disciplinas.

En primer lugar, las teorías de usabilidad y accesibilidad constituyen un eje central, ya que garantizan que la aplicación responda a las necesidades de usuarios adultos mayores y personas con movilidad reducida. Modelos como el Diseño Centrado en el Usuario (DCU) y los lineamientos de accesibilidad definidos por el W3C (2023) ofrecen criterios claros para la construcción de interfaces inclusivas y adaptadas a la población objetivo.

Por otro lado, las teorías de gestión logística aportan elementos para la planificación y optimización de la llamada “última milla” en la distribución de medicamentos. Estas teorías permiten analizar variables como rutas de entrega, costos de transporte, trazabilidad y manejo de inventarios (Boysen, Briskorn & Schwerdfeger, 2021).

Adicionalmente, el proyecto se apoya en teorías vinculadas a la seguridad y protección de datos, como los principios establecidos en la Ley 1581 de 2012 en Colombia y el Reglamento General de Protección de Datos (GDPR) en Europa, los cuales orientan la forma en que deben

manejarse los datos sensibles en entornos tecnológicos de salud.

De esta manera, el marco teórico no solo respalda la construcción técnica de la aplicación, sino que también asegura su pertinencia social, ética y normativa, reforzando la aplicabilidad de la solución en un entorno real de salud.

### **Marco Conceptual**

El marco conceptual de este proyecto integra conceptos relacionados con el diseño centrado en el usuario, la optimización logística y la transformación digital en salud, con un enfoque social orientado a la inclusión. Al integrar usabilidad y accesibilidad con la gestión logística y la autorización EPS, se plantea un modelo tecnológico que no solo mejora la eficiencia operativa del sistema de salud, sino que también responde a las necesidades reales de los adultos mayores y personas con movilidad reducida, promoviendo la equidad en la atención y la reducción de brechas digitales. Este enfoque multidimensional demuestra que la tecnología puede ser un medio para fortalecer la autonomía del paciente y mejorar la calidad de vida, cumpliendo con los lineamientos de innovación y salud digital recomendados por organismos internacionales como la OMS y la ISO.

### ***Términos Técnicos Fundamentales***

Los términos técnicos fundamentales permiten comprender los componentes y principios tecnológicos que sustentan el desarrollo del software del proyecto. A continuación, se definen los conceptos clave aplicados en el contexto de desarrollo ágil, arquitectura limpia y servicios digitales en salud.

**API (Application Programming Interface).** Una API es un conjunto de reglas, protocolos y herramientas que permiten la comunicación entre distintas aplicaciones o sistemas.

Facilita la integración y el intercambio de datos entre servicios, lo que resulta esencial en proyectos que requieren conexión entre plataformas EPS, bases de datos médicas y aplicaciones móviles (Microsoft Learn, 2024). Según Red Hat (2023), las APIs actúan como un puente que permite automatizar procesos, compartir información y extender funcionalidades sin comprometer la seguridad ni la independencia de cada componente.

**Escalabilidad.** La escalabilidad es la capacidad de un sistema para aumentar su rendimiento y soportar un mayor volumen de trabajo sin degradar su funcionamiento (Google Cloud, 2024). En el contexto del proyecto, una arquitectura escalable garantiza que la aplicación pueda atender a más usuarios, procesar mayor cantidad de solicitudes o integrarse con nuevas funciones sin requerir una reconstrucción completa. Pressman y Maxim (2020) señalan que la escalabilidad es un atributo esencial de la calidad del software según los estándares ISO/IEC 25010.

**DevOps.** DevOps es una filosofía y conjunto de prácticas que integran los equipos de desarrollo (Development) y operaciones (Operations) con el objetivo de mejorar la colaboración, la automatización y la entrega continua de software (AWS, 2024). Según Atlassian (2023), DevOps busca reducir el tiempo entre el desarrollo de una funcionalidad y su implementación, mediante herramientas de integración continua (CI) y entrega continua (CD), lo que favorece la estabilidad y agilidad en proyectos de salud digital.

**Seguridad (Security).** La seguridad del software comprende los mecanismos y prácticas destinados a proteger los sistemas y datos contra accesos no autorizados, fallas o ataques (OWASP, 2024). En entornos de salud, la seguridad adquiere un papel crítico debido al manejo de información sensible del paciente. Según la Organización Internacional de Normalización (ISO/IEC, 2022), los sistemas deben implementar autenticación robusta, cifrado de datos, control

de accesos y políticas de respaldo para garantizar confidencialidad, integridad y disponibilidad de la información.

**Usabilidad según Nielsen y normas internacionales.** La usabilidad se define como la medida en que un sistema, producto o servicio puede ser utilizado por usuarios específicos para alcanzar objetivos concretos con efectividad, eficiencia y satisfacción en un contexto de uso determinado (ISO 9241-11:2018). Jakob Nielsen, uno de los principales referentes en este campo, plantea cinco atributos clave que determinan la usabilidad de una interfaz: facilidad de aprendizaje, eficiencia, memorabilidad, bajos niveles de error y satisfacción del usuario (Nielsen, 2012).

En el desarrollo de aplicaciones móviles, garantizar la usabilidad implica diseñar interfaces intuitivas, consistentes y adaptadas a las necesidades del público objetivo. En el caso de proyectos orientados a salud digital, estos principios resultan fundamentales, ya que la claridad y la facilidad de uso impactan directamente en la adherencia de los pacientes al sistema y en la efectividad de la solución tecnológica.

**Accesibilidad digital: WCAG 2.1 y diseño inclusivo.** La accesibilidad digital hace referencia a la capacidad de los sistemas y plataformas tecnológicas de ser utilizados por la mayor cantidad posible de personas, incluidas aquellas con discapacidad o limitaciones funcionales. Para ello, la norma más reconocida a nivel internacional son las Pautas de Accesibilidad al Contenido Web (WCAG 2.1), publicadas por el World Wide Web Consortium (W3C), que establecen principios de diseño inclusivo bajo cuatro ejes: perceptible, operable, comprensible y robusto (W3C, 2018).

En el contexto de la salud digital, aplicar estas pautas cobra especial relevancia al considerar a poblaciones como adultos mayores o personas con discapacidad, quienes suelen enfrentar

barreras de acceso tecnológico. El diseño inclusivo contempla elementos como contraste visual adecuado, tipografías legibles, compatibilidad con lectores de pantalla, navegación sencilla y uso de lenguaje claro. Estas prácticas no solo cumplen un estándar técnico, sino que promueven la equidad en el acceso a la información y a los servicios de salud.

**Business Process Management (BPM).** El Business Process Management (BPM), o Gestión de Procesos de Negocio, es un enfoque sistemático que busca identificar, diseñar, ejecutar, monitorear y optimizar los procesos organizacionales con el objetivo de mejorar la eficiencia, la calidad y la capacidad de adaptación al cambio. Según Dumas, La Rosa, Mendling y Reijers (2018), el BPM combina modelos de negocio, métodos de gestión y herramientas tecnológicas para alinear los procesos con los objetivos estratégicos de una organización, garantizando su mejora continua.

### *Términos del Dominio de Aplicación*

A continuación, se definen los términos más relevantes para el desarrollo de la aplicación móvil destinada a mejorar la entrega de medicamentos en el barrio El Carmen:

**Optimización de Servicios.** La optimización de servicios hace referencia al proceso de mejorar la eficiencia y efectividad de los servicios ofrecidos, utilizando herramientas tecnológicas, procesos ágiles y la correcta asignación de recursos. Según estudios en administración de servicios (Bernal, 2010), una adecuada optimización no solo mejora el rendimiento, sino también la experiencia del usuario. En este caso, la aplicación móvil es una herramienta tecnológica que permite reducir tiempos de espera, mejorar la distribución de recursos (como personal y medicamentos), y garantizar que los usuarios reciban un servicio ágil y sin interrupciones.

**Telemedicina y Salud Digital.** La salud digital es un campo interdisciplinario que integra las Tecnologías de la Información y la Comunicación (TIC) con la atención sanitaria, con el propósito de mejorar la accesibilidad, eficiencia y calidad en los servicios de salud. Incluye herramientas como aplicaciones móviles, historias clínicas electrónicas, inteligencia artificial y sistemas de monitoreo remoto (World Health Organization, 2021). Dentro de este marco, la telemedicina se refiere al uso de tecnologías digitales para brindar atención clínica a distancia, permitiendo consultas, diagnósticos, seguimiento de pacientes y prescripción de tratamientos sin necesidad de un contacto físico directo.

Los principales beneficios de estas tecnologías incluyen la reducción de barreras geográficas, la optimización del tiempo de atención, la continuidad en los tratamientos, el empoderamiento del paciente y la disminución de costos tanto para usuarios como para instituciones. Su alcance es especialmente significativo en comunidades con dificultades de acceso, donde representan una herramienta clave para garantizar la equidad en salud.

**EPS (Entidades Promotoras de Salud).** Las Entidades Promotoras de Salud (EPS) son las instituciones responsables de afiliar a la población al Sistema General de Seguridad Social en Salud (SGSSS) en Colombia y garantizar el acceso a los servicios incluidos en el Plan de Beneficios en Salud (Ministerio de Salud y Protección Social, 2022). Su rol central es organizar la prestación de los servicios de salud, gestionar los recursos financieros mediante el recaudo de cotizaciones y coordinar con las Instituciones Prestadoras de Salud (IPS) la atención efectiva a los afiliados.

Las EPS también son responsables de los procesos de autorización de medicamentos, tratamientos y procedimientos, razón por la cual constituyen un actor fundamental en la

problemática del acceso a fármacos. Una gestión eficiente por parte de estas entidades repercute directamente en la calidad de vida de los usuarios, mientras que las deficiencias en sus procesos generan barreras de acceso, descontento social y sobrecarga institucional.

**Logística y Gestión de la Distribución.** La logística de distribución de medicamentos comprende el conjunto de procesos orientados a garantizar que los productos farmacéuticos lleguen a los usuarios en las condiciones adecuadas de calidad, seguridad y tiempo. Este proceso incluye la planeación de la demanda, la gestión de inventarios, el diseño de rutas de distribución y el control de tiempos de entrega (Organización Panamericana de la Salud, 2014).

En el contexto de la salud comunitaria, la logística debe contemplar factores como la trazabilidad de los medicamentos, la cadena de frío en el caso de biológicos, la planificación de rutas óptimas para cubrir sectores urbanos o rurales y la reducción de tiempos de espera. Una logística deficiente puede ocasionar retrasos en tratamientos, aumento de costos operativos y un impacto negativo en la salud del paciente, especialmente en poblaciones vulnerables.

**Optimización de procesos de logística: Tecnología aplicada para eficiencia operativa.** La optimización de los servicios logísticos se refiere a la aplicación de herramientas tecnológicas y modelos de gestión que buscan mejorar la eficiencia en cada etapa de la cadena de suministro. Tecnologías como los sistemas de información geográfica (GIS), aplicaciones móviles, inteligencia artificial y análisis de datos permiten optimizar la planeación de rutas, reducir costos de transporte, anticipar fallas en la cadena de suministro y mejorar la trazabilidad de los medicamentos (European Medicines Agency, 2021; García & Torres, 2020).

En proyectos de salud, estas herramientas favorecen la equidad en el acceso a tratamientos al asegurar que los medicamentos lleguen a tiempo, evitando interrupciones y reduciendo riesgos

en la atención médica. El uso de soluciones digitales también permite un control más transparente, eficiente y sostenible en la operación logística de las instituciones de salud.

**Usuarios y Condiciones de Vulnerabilidad -Movilidad Reducida.** La movilidad reducida hace referencia a la limitación permanente o temporal en la capacidad de una persona para desplazarse de manera autónoma, ya sea por causas físicas, sensoriales, cognitivas, de edad avanzada o derivadas de enfermedades crónicas (Organización Mundial de la Salud, 2011). Desde el punto de vista poblacional, este grupo incluye a adultos mayores, personas con discapacidad motora, usuarios en procesos post quirúrgicos y pacientes con enfermedades degenerativas.

En el contexto de la salud comunitaria, la movilidad reducida representa una barrera de acceso a los servicios sanitarios, pues dificulta el desplazamiento a centros de atención, trámites presenciales y reclamación de medicamentos. Por ello, la implementación de soluciones digitales y logísticas adaptadas a esta población contribuye significativamente a garantizar la equidad en el acceso a la atención médica.

**Usuarios vulnerables.** Los usuarios vulnerables son aquellos que, debido a condiciones sociales, médicas o tecnológicas, enfrentan mayores riesgos de exclusión en el acceso a servicios básicos como la salud. Entre ellos se incluyen adultos mayores, personas con discapacidad, pacientes con bajos recursos económicos, y comunidades en entornos rurales o de difícil acceso (CEPAL, 2020).

Desde la dimensión social, la vulnerabilidad se asocia a desigualdades socioeconómicas, educativas y de conectividad digital. En la dimensión médica, se relaciona con enfermedades crónicas, dependencia de tratamientos continuos y fragilidad física. Finalmente, desde la dimensión tecnológica, la brecha digital limita la capacidad de estas poblaciones para acceder a

soluciones de salud digital, ya sea por falta de dispositivos, conectividad o competencias en el uso de aplicaciones móviles.

En consecuencia, los proyectos de salud digital deben diseñarse bajo un enfoque de inclusión y accesibilidad, garantizando que estas poblaciones no solo sean consideradas en el diseño, sino que también se beneficien directamente de las soluciones implementadas.

**Adulto mayor.** El término adulto mayor hace referencia a las personas de 60 años o más, etapa de la vida en la que suelen presentarse cambios fisiológicos, cognitivos y sociales que pueden afectar la autonomía y la capacidad para realizar actividades cotidianas. Según la Organización Mundial de la Salud (OMS, 2015), el envejecimiento saludable implica mantener la funcionalidad y la calidad de vida, garantizando condiciones de accesibilidad y atención integral. En el contexto de los servicios de salud, los adultos mayores constituyen un grupo prioritario debido a la alta prevalencia de enfermedades crónicas, la dependencia de tratamientos continuos y las limitaciones de movilidad que dificultan el acceso oportuno a medicamentos y servicios médicos. Por tanto, las soluciones tecnológicas deben diseñarse con criterios de usabilidad, accesibilidad y simplicidad, considerando aspectos como el tamaño del texto, la claridad visual, el apoyo familiar y la asistencia por voz, con el fin de promover una experiencia inclusiva y segura.

### ***Relación entre Conceptos***

Los conceptos de usabilidad, accesibilidad, logística y autorización EPS se integran de forma interdependiente en el diseño de soluciones tecnológicas aplicadas al sector salud. La usabilidad, entendida como la facilidad con la que un usuario puede aprender y utilizar un sistema, es clave para garantizar la eficacia y satisfacción de los adultos mayores que presentan limitaciones cognitivas o motrices (Nielsen Norman Group, 2020). Por su parte, la accesibilidad

permite que la aplicación sea inclusiva y adaptable a distintas condiciones sensoriales o físicas, asegurando el acceso equitativo a los servicios de salud digital (W3C, 2018; ISO, 2019).

La logística actúa como el componente operativo que materializa la experiencia de usuario, al coordinar la entrega efectiva de medicamentos y optimizar los tiempos de respuesta. Según la Organización Mundial de la Salud (WHO, 2021), una gestión logística eficiente en el sistema de salud reduce riesgos y garantiza continuidad terapéutica, especialmente en poblaciones vulnerables. En este contexto, la autorización EPS representa el vínculo administrativo que habilita la entrega oportuna del medicamento, integrando los procesos clínicos y logísticos en una sola plataforma. Cuando la aplicación facilita dicha autorización de manera digital y simplificada, se eliminan las barreras que históricamente han limitado el acceso de los adultos mayores a sus tratamientos. De esta forma, la convergencia entre usabilidad, accesibilidad, logística y autorización EPS configura un sistema inclusivo que responde al perfil de vulnerabilidad de los usuarios, al tiempo que promueve eficiencia, autonomía y equidad en el acceso a los servicios de salud.

En síntesis, el marco conceptual de este proyecto articula los principios del diseño centrado en el usuario, la optimización logística y la transformación digital en salud, con un enfoque social orientado a la inclusión. Al integrar usabilidad y accesibilidad con la gestión logística y la autorización EPS, se plantea un modelo tecnológico que no solo mejora la eficiencia operativa del sistema de salud, sino que también responde a las necesidades reales de los adultos mayores y personas con movilidad reducida, promoviendo la equidad en la atención y la reducción de brechas digitales. Este enfoque multidimensional demuestra que la tecnología puede ser un medio para fortalecer la autonomía del paciente y mejorar la calidad de vida, cumpliendo con los lineamientos de innovación y salud digital recomendados por organismos

internacionales como la OMS y la ISO.

## **Marco Legal**

El desarrollo y la implementación de la aplicación móvil para optimizar la entrega de medicamentos a los usuarios del barrio El Carmen deben regirse por diversas normativas nacionales e internacionales, con el fin de garantizar la protección de los derechos de los usuarios, la seguridad de la información y la eficiencia en el funcionamiento del sistema de salud (MinTIC, 2023). A continuación, se describen los principales marcos normativos que sustentan legalmente el proyecto.

### ***Ley 1581 de 2012 Protección de Datos Personales***

La Ley 1581 de 2012 establece las disposiciones generales para la protección de los datos personales en Colombia. Dado que la aplicación manejará información sensible, como datos médicos y personales, se deben aplicar estrictamente los principios de consentimiento informado, finalidad y seguridad (Congreso de la República de Colombia, 2012).

Entre las obligaciones más relevantes se encuentran:

Obtener la autorización explícita del titular antes del tratamiento de los datos.

Implementar medidas de seguridad que garanticen la confidencialidad de los datos sensibles.

Asegurar la transparencia en el uso, almacenamiento y transmisión de la información (SIC, 2024).

### ***Ley 1438 de 2011 Reforma del Sistema de Salud***

La Ley 1438 de 2011 reforma el Sistema General de Seguridad Social en Salud e

introduce estrategias para mejorar la calidad, eficiencia y accesibilidad del sistema (MinSalud, 2011).

Este marco legal respalda el proyecto al promover el uso de tecnologías que faciliten la atención médica y la entrega de medicamentos a poblaciones vulnerables. La norma también enfatiza la importancia de agilizar los procesos administrativos mediante herramientas digitales, alineándose con el propósito del sistema propuesto.

### ***Ley 23 de 1981 Código de Ética Médica***

La Ley 23 de 1981 regula el ejercicio ético de la medicina y la responsabilidad de los profesionales en la prestación del servicio. Dentro de su alcance, establece que los prestadores deben garantizar la entrega oportuna y segura de los medicamentos (Congreso de la República de Colombia, 1981). La aplicación contribuye al cumplimiento de esta disposición al ofrecer un sistema trazable y seguro que minimiza errores y retrasos en la distribución.

### ***Ley 1266 de 2008 Habeas Data Financiero y de Información Personal***

La Ley 1266 de 2008 regula el manejo de la información personal en bases de datos, definiendo principios de veracidad, confidencialidad y finalidad (Congreso de la República de Colombia, 2008). Aunque originalmente se enfoca en el ámbito financiero, es relevante para este proyecto por establecer los criterios para el tratamiento de datos personales y la protección de información sensible, lo que refuerza el cumplimiento de la Ley 1581 de 2012.

### ***Ley 1751 de 2015 Derecho Fundamental a la Salud***

La Ley Estatutaria 1751 de 2015 reconoce la salud como un derecho fundamental autónomo, garantizando la universalidad y la equidad en el acceso a los servicios (MinSalud, 2015). Esta norma destaca la necesidad de incorporar la tecnología en la atención sanitaria, asegurando que los ciudadanos puedan recibir sus tratamientos de forma continua y eficiente. En

ese sentido, la aplicación móvil propuesta contribuye a la materialización del derecho a la salud, especialmente para personas mayores o con movilidad reducida.

### ***Ley 527 de 1999 Comercio Electrónico***

La Ley 527 de 1999 regula el uso de mensajes de datos, firmas digitales y medios electrónicos, garantizando la validez jurídica de las transacciones realizadas por medios digitales (MinTIC, 2023). Esto es fundamental para el componente de comercio electrónico de la aplicación, que permite solicitar medicamentos en línea y realizar gestiones de forma segura y verificable.

### ***Regulación de Plataformas Móviles y Datos Personales***

Las aplicaciones que recopilan o procesan datos personales deben cumplir con las disposiciones de la Ley 1581 de 2012 y la Ley 1266 de 2008, aplicando principios de legalidad, transparencia y seguridad en todo el ciclo de vida de los datos (SIC, 2024). Asimismo, las plataformas móviles deben garantizar que sus mecanismos de autenticación, almacenamiento y comunicación cumplan con las políticas de protección de información definidas por el Decreto 1377 de 2013 y la Política de Seguridad Digital de Colombia (MinTIC, 2022).

### ***Normativas Internacionales***

En caso de que el sistema escale o requiera interoperabilidad internacional, deben considerarse marcos regulatorios globales como:

El Reglamento General de Protección de Datos (GDPR) de la Unión Europea, que establece directrices sobre recopilación, almacenamiento y tratamiento responsable de datos personales (European Union, 2016).

Las Normas de la Organización Mundial de la Salud (OMS) sobre servicios digitales de salud, que promueven la protección del paciente y la ética en el uso de tecnologías sanitarias

(WHO, 2021).

El cumplimiento de los marcos legales descritos no solo garantiza la conformidad del proyecto con las normas nacionales e internacionales, sino que también constituye un mecanismo preventivo frente a riesgos legales, reputacionales y sanitarios. La observancia de la Ley 1581 de 2012 y la Ley 1266 de 2008 sobre protección de datos personales asegura que la aplicación cumpla con los principios de confidencialidad, transparencia y consentimiento informado, minimizando el riesgo de sanciones por vulneración de la privacidad y fortaleciendo la confianza de los usuarios en el sistema.

Asimismo, el cumplimiento de la Ley 1751 de 2015, que reconoce la salud como un derecho fundamental, y de la Ley 1438 de 2011, orientada a la modernización del sistema sanitario, refuerza la legitimidad del proyecto dentro del marco de la salud pública, garantizando que la entrega de medicamentos sea segura, trazable y accesible. De igual forma, la Ley 23 de 1981 asegura la responsabilidad ética en la atención y suministro farmacéutico, evitando prácticas que comprometan la seguridad del paciente o la integridad profesional.

En el ámbito tecnológico, la Ley 527 de 1999 sobre comercio electrónico y las normas del Reglamento General de Protección de Datos (GDPR) aportan directrices esenciales para la validez jurídica de las transacciones digitales, la protección transfronteriza de la información y la interoperabilidad técnica, permitiendo que el sistema pueda escalar hacia otros contextos o integrarse con nuevas plataformas de manera segura y legalmente sustentada.

En síntesis, la observancia de este marco jurídico fortalece la gobernanza del proyecto, al asegurar el cumplimiento normativo en todos los niveles administrativo, tecnológico y sanitario. De esta manera, se mitigan riesgos legales y reputacionales, se protege la información sensible de los usuarios y se sientan las bases para una expansión escalable y sostenible del modelo,

garantizando su replicabilidad en otros territorios o entidades del sistema de salud.

En conclusión, el marco referencial del proyecto integra los fundamentos conceptuales, técnicos, normativos y metodológicos que sustentan la propuesta de desarrollo de una aplicación móvil para la optimización de la entrega de medicamentos en el barrio El Carmen. Los referentes revisados en materia de usabilidad, accesibilidad, arquitectura limpia, metodologías ágiles y salud digital permiten establecer una base sólida para el diseño de una solución centrada en el usuario, técnicamente viable y alineada con las políticas de salud pública del país.

El enfoque metodológico basado en Scrum y los principios de diseño centrado en el usuario (DCU) garantizan la iteración continua con los actores involucrados, lo que disminuye los riesgos de baja adopción y mejora la alineación del producto con las necesidades reales de la población objetivo. De igual modo, la incorporación de marcos de calidad del software (ISO/IEC 25010) y los estándares de interoperabilidad en salud (HL7 FHIR) refuerzan la validez técnica y la compatibilidad del sistema con otros servicios institucionales, aumentando su potencial de escalabilidad y replicabilidad.

La aplicación de principios de seguridad digital y cumplimiento de la Ley 1581 de 2012 sobre protección de datos personales mitiga riesgos asociados al manejo de información sensible, garantizando la confidencialidad, integridad y disponibilidad de los datos. Asimismo, la integración de la autorización EPS dentro del flujo logístico contribuye a la trazabilidad y control administrativo, fortaleciendo la confianza de los usuarios y prestadores.

Finalmente, la articulación entre los componentes teóricos y tecnológicos descritos en el marco referencial evidencia que el proyecto no solo responde a una necesidad operativa del sistema de salud, sino también a un reto social: mejorar la calidad de vida de adultos mayores y personas con movilidad reducida mediante el uso responsable e inclusivo de la tecnología. Este

marco, por tanto, orienta la implementación hacia una solución innovadora que equilibra eficiencia técnica, cumplimiento normativo y sostenibilidad social.

## **Marco Metodológico**

El presente marco metodológico define la ruta científica, técnica y procedimental que orienta el desarrollo del proyecto MediExpress, cuyo propósito es diseñar y validar una aplicación móvil que optimice el proceso de entrega de medicamentos a adultos mayores y personas con movilidad reducida en el barrio El Carmen, localidad de Tunjuelito. Para ello, se establece un enfoque de investigación aplicada con alcance descriptivo, un enfoque mixto de recolección y análisis de datos, el uso de técnicas cualitativas y cuantitativas, y la integración de metodologías de análisis de procesos y desarrollo ágil de software. Estos elementos se articulan de manera secuencial a través de fases que permiten analizar el proceso actual, diseñar la arquitectura del sistema, desarrollar la solución tecnológica y validar su funcionalidad y usabilidad con usuarios reales, garantizando coherencia metodológica y pertinencia social.

### **Tipo de estudio**

El proyecto se enmarca en un tipo de estudio aplicado y descriptivo, dado que se orienta hacia la resolución de problemas inmediatos en contextos específicos, diferenciándose de la investigación básica, que busca generar conocimiento teórico. (Creswell y Plano Clark ,2018),

El carácter descriptivo del estudio permite caracterizar el proceso actual de entrega de medicamentos, identificando sus principales limitaciones en términos de accesibilidad, tiempos de espera y experiencia del usuario, sin manipular deliberadamente las variables analizadas.

### **Enfoque de investigación**

Se adopta un enfoque mixto (cualitativo–cuantitativo), el cual permite integrar datos

numéricos con información cualitativa derivada de las percepciones, experiencias y valoraciones de los participantes. Este enfoque facilita una comprensión más completa del fenómeno estudiado al combinar métricas objetivas, como tiempos de espera o niveles de satisfacción, con apreciaciones subjetivas relacionadas con la usabilidad y accesibilidad del sistema (Hernández-Sampieri et al., 2022; Creswell & Plano Clark, 2018).

En el contexto del desarrollo de software, el enfoque mixto resulta especialmente pertinente, ya que posibilita evaluar tanto el desempeño funcional de la aplicación como la experiencia del usuario final, fortaleciendo la validez interna y externa de los resultados obtenidos.

### **Hipótesis**

Si se desarrolla una aplicación móvil accesible para solicitar y coordinar la entrega de medicamentos, entonces los adultos mayores y personas con movilidad reducida del barrio El Carmen en la localidad de Tunjuelito podrán recibir sus tratamientos en menor tiempo y con menos desplazamientos, porque la digitalización del proceso reduce barreras físicas y agiliza la autorización y distribución desde la EPS.

### **Técnicas e instrumentos de recolección de información**

Para dar cumplimiento a los objetivos del proyecto, se emplean técnicas de recolección de información coherentes con el enfoque mixto de la investigación, las cuales permiten obtener datos tanto cualitativos como cuantitativos relevantes para el análisis y validación de la solución tecnológica Hernández-Sampieri, Fernández-Collado y Baptista (2022).

### ***Encuesta semiestructurada***

La encuesta semiestructurada es un instrumento de recolección de información que permite integrar preguntas cerradas, que facilitan el análisis cuantitativo, con preguntas abiertas que posibilitan obtener información cualitativa más detallada. Según Hernández-Sampieri, Fernández-Collado y Baptista (2022), este tipo de encuesta permite estructurar un conjunto básico de preguntas estandarizadas, sin restringir la posibilidad de que el participante amplíe sus respuestas, lo que aporta profundidad y contexto a los datos.

De acuerdo con Creswell y Creswell (2018), las encuestas semiestructuradas resultan especialmente útiles en estudios aplicados, ya que combinan mediciones numéricas sobre tendencias o percepciones con la exploración interpretativa de experiencias subjetivas. Esto contribuye a obtener una visión más completa del fenómeno estudiado, al permitir analizar tanto la frecuencia de ciertos comportamientos como las razones y significados que los participantes atribuyen a sus acciones.

En el contexto del proyecto MediExpress, la encuesta semiestructurada fue aplicada a los usuarios de la aplicación móvil con el propósito de evaluar aspectos como la facilidad de uso, la accesibilidad, la pertinencia funcional y el nivel de satisfacción con el sistema. Este instrumento permitió obtener datos cuantitativos sobre la frecuencia de uso y la percepción general del servicio, así como comentarios cualitativos sobre dificultades, recomendaciones y oportunidades de mejora. De esta forma, la encuesta aportó evidencia empírica para la validación del sistema, asegurando que el diseño responda a las necesidades reales de los usuarios beneficiarios.

**Población.** Según Hernández-Sampieri, Fernández-Collado y Baptista (2022), la población “comprende el universo o grupo completo del cual el investigador desea obtener información o al que pretende generalizar los resultados del estudio”.

En el marco del proyecto MediExpress, la población objetivo está conformada por usuarios mayores de 60 años y personas con movilidad reducida que reclaman medicamentos en el punto de atención del barrio El Carmen.

De acuerdo con datos del (Dane, 2020), el total de la localidad de la localidad de Tunjuelito es: 183.067 personas, de la cual se encuentra que:

- Población total aproximada de mayores de 60 años de la localidad de Tunjuelito: 22.571 personas
- Población aproximada de personas (mayores y menores de 60 años ) con discapacidad en la localidad de Tunjuelito es de 4.733 personas aprox.
- Del total de personas con discapacidad, la población total de personas con discapacidad: movilidad reducida menor de 60 años (estimación provisional): 4.117 personas aprox.
- Del total de personas con discapacidad, la población total de personas con discapacidad: movilidad reducida mayor de 60 años (estimación provisional): 583 personas aprox.

Nota: estas cifras son provisionales y aplicadas a escala de barrio como proxy. No existe, en fuentes públicas consultadas, una estadística DANE divulgada por barrio (El Carmen). Por eso se usa la proporción de la localidad como referencia y se proporciona un rango para discapacidad dada la variación entre fuentes.

**Muestra.** La muestra corresponde al número representativo de usuarios mayores de 60 años y personas con movilidad reducida que reclaman medicamentos en el punto de atención del barrio El Carmen.

Para ello se aplicará la fórmula estadística para poblaciones finitas:

$$n = \frac{N * Z^2 * p * q}{e^2(N - 1) + Z^2 * p * q}$$

Donde:

**Tabla 2**

*Conversiones de la formula estadística*

<b>Símbolo</b>	<b>Descripción</b>
<b>N</b>	Tamaño de la muestra
<b>N</b>	Tamaño de la población total (usuarios del punto de atención)
<b>Z</b>	Nivel de confianza (valor Z según la distribución normal; 1.96 para 95%)
<b>P</b>	Proporción esperada de éxito (usualmente 0.5 si no se conoce)
<b>Q</b>	Proporción esperada de fracaso (1 - p)
<b>E</b>	Margen de error permitido (generalmente 0.05 = 5%)

Si se considera que en el barrio El Carmen hay aproximadamente 583 usuarios mayores de 60 años y/o con movilidad reducida que reclaman medicamentos:

$$n = \frac{583 * (1.96)^2 * 0.5 * 0.5}{(0.05)^2(583 - 1) + (1.96)^2 * 0.5 * 0.5}$$

$$n \approx 232$$

Por lo tanto, la muestra representativa sería de 232 personas, garantizando un nivel de confianza del 95 % y un margen de error del 5 %.

### **Metodologías a implementar**

Este proyecto hace uso de dos metodologías como son: Business Process Management

(BPM) para la comprensión del proceso de entrega de medicamentos y su rediseño mediante la incorporación de una app móvil y el Marco Ágil Scrum, para el desarrollo del sistema de entrega de medicamentos.

### ***Business Process Management (BPM)***

De acuerdo con Weske (2019), el BPM permite representar los procesos de manera formal mediante modelos gráficos o diagramas (como BPMN, Business Process Model and Notation), facilitando la comunicación entre los equipos técnicos y operativos. Además, impulsa la automatización e integración de flujos de trabajo, mejorando la trazabilidad y reduciendo errores humanos.

En el contexto del proyecto MediExpress, la aplicación de BPM resulta clave para estructurar y optimizar el proceso de entrega de medicamentos, desde la autorización EPS hasta la distribución al usuario final, garantizando la eficiencia logística y la trazabilidad de cada operación dentro del sistema.

**Sistema de autorización de medicamentos.** El sistema de autorización de medicamentos en Colombia está regulado por las Entidades Promotoras de Salud (EPS) y consiste en la validación previa que los usuarios deben realizar para obtener ciertos medicamentos incluidos en el Plan de Beneficios en Salud (PBS) o aquellos que requieren un trámite adicional por su costo o características. Este proceso, aunque necesario para garantizar la sostenibilidad financiera del sistema, suele presentar fallas operativas que afectan directamente al paciente.

Actualmente, los usuarios deben asistir presencialmente a las sedes de las EPS o sus

operadores logísticos, enfrentando largas filas, caídas frecuentes de los sistemas, falta de personal y tiempos de espera superiores a los recomendados. Esto genera retrasos en la entrega de tratamientos, abandono de los mismos y mayor riesgo de complicaciones en la salud de los pacientes, especialmente en adultos mayores y personas con movilidad reducida.

### ***Marco ágil Scrum***

El marco ágil Scrum es una metodología de desarrollo iterativo e incremental que se utiliza para gestionar proyectos complejos de software mediante la colaboración continua, la entrega temprana de valor y la adaptación al cambio. Según Schwaber y Sutherland (2020), creadores del marco, Scrum se fundamenta en los principios del Manifiesto Ágil y busca optimizar la productividad de los equipos mediante ciclos cortos llamados sprints, en los que se planifican, desarrollan y revisan incrementos funcionales del producto.

En el contexto del proyecto MediExpress, la adopción del marco ágil Scrum permite coordinar el desarrollo del sistema móvil y web de forma organizada, flexible y colaborativa, favoreciendo la comunicación entre desarrolladores, diseñadores y actores del sistema de salud. Asimismo, contribuye a reducir riesgos, optimizar tiempos de entrega y asegurar que la solución responda a las necesidades reales de los usuarios finales.

Finalmente, la integración de Business Process Management (BPM) y el marco ágil Scrum en la metodología del proyecto MediExpress permite articular la optimización de procesos organizacionales con el desarrollo iterativo de software centrado en el usuario. Mientras BPM proporciona una visión estructurada y analítica de los flujos de trabajo, desde la autorización EPS hasta la entrega final del medicamento, asegurando su eficiencia, trazabilidad y mejora continua (Dumas et al., 2018; Weske, 2019), Scrum facilita la gestión ágil de los equipos

de desarrollo, fomentando la adaptabilidad, la colaboración y la entrega constante de valor funcional (Schwaber & Sutherland, 2020). Esta combinación metodológica garantiza que la solución tecnológica propuesta sea escalable, eficiente y alineada con los procesos reales del sistema de salud, al tiempo que permite responder de forma dinámica a las necesidades de los usuarios y asegurar una mejora continua del producto.

### ***Fases del Desarrollo del Proyecto***

A continuación, se presentan las fases del proyecto MediExpress, organizadas para asegurar un desarrollo claro y coherente. Estas fases orientan la construcción de una solución tecnológica que optimice la entrega de medicamentos en el barrio El Carmen (Tunjuelito – Bogotá), este enfoque integra análisis del proceso actual, diseño del sistema y validación con los usuarios objetivo.

**Análisis del proceso actual de entrega de en el barrio El Carmen (Tunjuelito – Bogotá).** El análisis del proceso actual de entrega de medicamentos a adultos mayores y personas con movilidad reducida en el barrio El Carmen (Tunjuelito – Bogotá) consiste en el estudio detallado de las etapas, actores, recursos y dificultades presentes en el flujo operativo mediante el cual los usuarios acceden a sus tratamientos médicos. Este análisis busca identificar barreras logísticas, tecnológicas y administrativas, como demoras en autorizaciones EPS, desplazamientos complejos o falta de trazabilidad, que afectan la continuidad del tratamiento y la calidad del servicio. A partir de esta evaluación se establecen oportunidades de mejora orientadas al diseño de una solución tecnológica más eficiente, inclusiva y adaptada a las condiciones de los adultos mayores y personas con movilidad reducida.

### **Diseño de la arquitectura de la aplicación móvil para la entrega de medicamentos.**

El diseño de la arquitectura de la aplicación móvil basada en principios de escalabilidad, mantenibilidad y modularidad consiste en estructurar el sistema de forma que pueda crecer, adaptarse y mantenerse fácilmente a lo largo del tiempo. Según Martin (2017), una arquitectura limpia y modular permite separar responsabilidades en capas bien definidas, facilitando la comprensión, prueba y actualización del software sin afectar otras partes del sistema. La escalabilidad asegura que la aplicación pueda soportar un aumento en usuarios o funcionalidades; la mantenibilidad garantiza que el código sea comprensible y modificable con bajo costo; y la modularidad promueve el desarrollo independiente de componentes reutilizables, mejorando la eficiencia del equipo de desarrollo y la calidad del producto final.

**Desarrollo de la aplicación móvil para la entrega de medicamentos en el barrio El Carmen.** El desarrollo de una aplicación móvil con Angular y metodología Scrum consiste en la creación iterativa e incremental de una herramienta digital que permita optimizar la entrega de medicamentos a adultos mayores en el barrio El Carmen, garantizando accesibilidad, usabilidad y eficiencia logística. Angular, como framework de desarrollo frontend, facilita la construcción de interfaces modulares, escalables y mantenibles (Google Developers, 2024), mientras que Scrum, como marco ágil, promueve la colaboración continua, la retroalimentación temprana y la mejora progresiva del producto (Schwaber & Sutherland, 2020). La combinación de ambas metodologías asegura un desarrollo adaptable a las necesidades reales de los usuarios, reduciendo riesgos y mejorando la calidad del producto final.

**Evaluación de la funcionalidad y usabilidad inicial del producto mínimo viable.** La evaluación de la funcionalidad y usabilidad inicial del producto mínimo viable consiste en aplicar pruebas exploratorias y sesiones de retroalimentación con usuarios representativos del

público objetivo , en este caso, adultos mayores y personas con movilidad reducida del barrio El Carmen, para verificar el correcto funcionamiento de las principales características de la aplicación móvil. Este proceso permite identificar fallos técnicos, medir la facilidad de uso y obtener información cualitativa sobre la experiencia del usuario, asegurando que el diseño y las funcionalidades del sistema respondan efectivamente a las necesidades reales antes de su implementación final (Brooke, 1996; Nielsen, 1994; ISO 9241-210:2019).

## **Análisis del proceso actual de entrega de medicamentos en el barrio El Carmen (Tunjuelito – Bogotá)**

Esta fase tiene como propósito comprender de manera estructurada el proceso actual de entrega de medicamentos a adultos mayores y personas con movilidad reducida, utilizando el enfoque de Modelado de Procesos de Negocio (BPM) para representar el flujo real de actividades, actores, puntos críticos y oportunidades de mejora. Paralelamente, se emplean técnicas centradas en el usuario como encuestas semiestructuradas y observación directa con el fin de identificar necesidades, expectativas y condiciones de uso que orienten los requisitos funcionales y no funcionales del sistema. La combinación de BPM y metodologías centradas en el usuario permite alinear la solución tecnológica tanto con la lógica operativa del negocio como con las experiencias y limitaciones reales de las personas involucradas.

### **Actores y roles del proceso de entrega de medicamentos en el barrio el Carmen**

En este proceso, intervienen los siguientes actores:

**Tabla 3**

*Actores y roles del proceso de entrega de medicamentos*

<b>Tipo de actor</b>	<b>Actor</b>	<b>Descripción</b>
Interno	Administrativo de la EPS o IPS	Encargado de la verificación de autorizaciones y datos.
Interno	Auxiliares de farmacia	Responsables de la entrega física del medicamento.
Interno	Sistema de autorización EPS.	Plataforma digital que valida si el medicamento está aprobado para su entrega.
Externo	Usuarios	Adultos mayores, personas con movilidad reducida y pacientes con tratamientos de continuidad.
Externo	Familiares o acompañantes	Quienes acompañan al usuario cuando no puede moverse por sí mismo.

*Nota.* Esta tabla muestra los actores y los roles involucrados en la entrega de medicamentos

## Segmentación de público Objetivo

A continuación, se presenta la segmentación del público objetivo del proyecto MediExpress correspondiente a adultos mayores y personas con movilidad reducida:

**Tabla 4**  
*Segmentación de público Objetivo*

<b>Variable</b>	<b>Descripción / Características Relevantes</b>
<b>Demográfica</b>	Adultos mayores de 60 años o más, en su mayoría con enfermedades crónicas que requieren medicación continua (hipertensión, diabetes, artritis). Personas de cualquier rango de edad, pero con movilidad reducida, permanente o temporal, Cuidadores familiares responsables de gestionar trámites y reclamación de medicamentos a los pacientes antes mencionados.
<b>Geográfica</b>	Residentes del barrio El Carmen, localidad de Tunjuelito (Bogotá D.C.), zona predominantemente estratos 1 y 2, con acceso limitado a centros de salud y rutas de transporte cercanas. La distancia a puntos de dispensación representa una barrera frecuente.
<b>Psicográfica</b>	Usuarios con alta necesidad de acompañamiento, orientación y claridad en procesos de salud. Presentan baja alfabetización digital y preferencia por interfaces simples, mensajes claros y soporte humano. Valoran la confianza y la seguridad en el manejo de su información y medicamentos.
<b>Conductual</b>	Frecuencia recurrente de reclamación de medicamentos (mensual o quincenal). Dificultad para asistir a trámites presenciales por limitaciones físicas o de transporte. Disposición a usar una aplicación móvil accesible siempre que reduzca tiempo, desplazamientos y esfuerzo personal.

*Nota.* Esta tabla muestra la segmentación del público objetivo

## Aplicación de técnicas para recolección de información

Con base en la muestra identificada se aplicaron encuestas semiestructuradas para recolectar la información necesaria para el desarrollo asertivo de la aplicación.

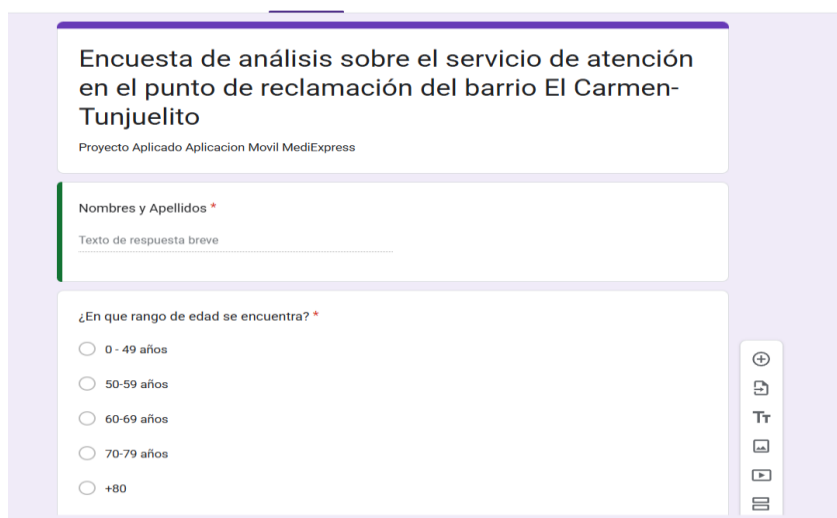
### *Encuesta semiestructurada a adultos mayores y personas con movilidad reducida*

Se aplicó una encuesta semiestructurada a adultos mayores de 60 años, personas con

movilidad reducida y cuidadores, para el análisis de la muestra seleccionada, en una serie de preguntas sobre temas como, el servicio de atención, su trayecto y frecuencia de solicitud de medicamentos, apoyo familiar, entre otros. En el apéndice A se puede observar la encuesta aplicada.

### **Figura 1**

#### *Encuesta semiestructurada*



Encuesta de análisis sobre el servicio de atención en el punto de reclamación del barrio El Carmen-Tunjuelito

Proyecto Aplicado Aplicacion Movil MediExpress

Nombres y Apellidos \*

Texto de respuesta breve

¿En que rango de edad se encuentra? \*

0 - 49 años

50-59 años

60-69 años

70-79 años

+80

*Fuente.* Captura de pantalla de Google Forms, adaptada por los autores (2025)

*Nota.* El enlace de la encuesta desarrollada para los usuarios puede verse en el anexo.

Este instrumento permitió recopilar información relevante sobre los usuarios, incluyendo su rango de edad, disponibilidad y uso de dispositivos tecnológicos (propios o de familiares), tiempos de espera en el punto de reclamación, niveles de satisfacción con el servicio actual y otros aspectos relacionados con su experiencia. A partir de estos datos, fue posible identificar qué proporción de la población estaría en condiciones de adoptar la solución tecnológica propuesta y qué segmentos presentaron barreras de acceso. Estos resultados servirán para orientar estrategias de implementación inclusiva, promoviendo una aplicación móvil de uso sencillo y accesible, así como la articulación con la Alcaldía Local para fortalecer el

acompañamiento y cobertura en los casos que lo requieran.

## Análisis de Resultados

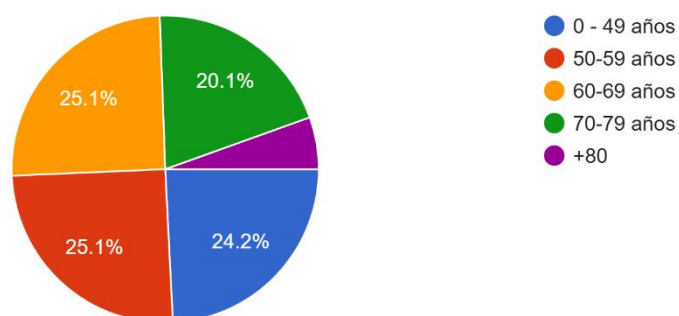
Después de aplicar la encuesta a los adultos mayores y personas con movilidad reducida del barrio El Carmen, a continuación, se presentan los siguientes resultados.

### Figura 2

*Rango de edad del usuario*

¿En que rango de edad se encuentra?

219 respuestas

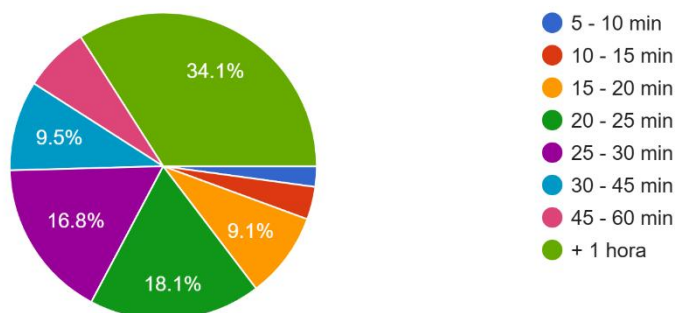


*Nota.* Se puede observar los rangos de edad de los usuarios encuestados, entre los cuales el mayor porcentaje lo tienen las personas entre 50 y 59 años y 60 y 69 años con el 25.1%

**Figura 3***Rango de Tiempo que toma llegar al punto de reclamación*

¿Que rango de tiempo toma para llegar al punto de reclamación?

232 respuestas

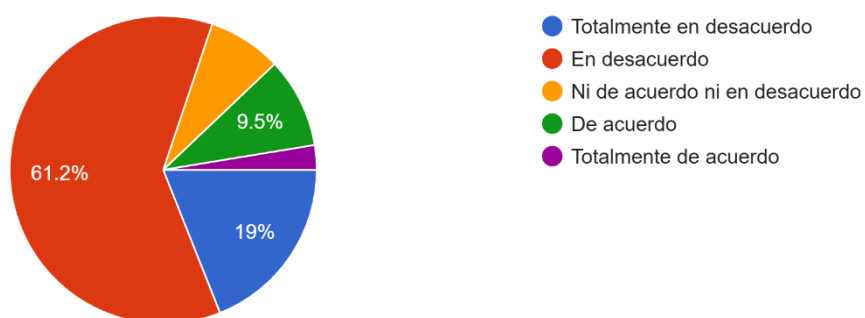


*Nota.* En el grafico anterior se muestra que, en su mayoría (34.1%), los pacientes demoran 1 hora para llegar al punto de reclamación de medicamentos.

**Figura 4***¿El tiempo de demora es adecuado?*

¿El tiempo que demora para llegar al punto de reclamacion de medicamentos es el adecuado?

232 respuestas

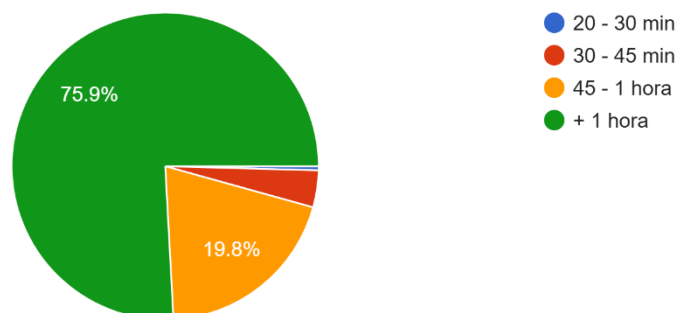


*Nota.* En la imagen anterior se observa que la mayoría de encuestados (61.2%) está en desacuerdo con que el tiempo de demora en llegar al punto de reclamación, sea el adecuado.

**Figura 5***Tiempo de espera para recibir los medicamentos*

¿Cuánto tiempo debe esperar normalmente para ser atendido hasta recibir sus medicamentos?

232 respuestas

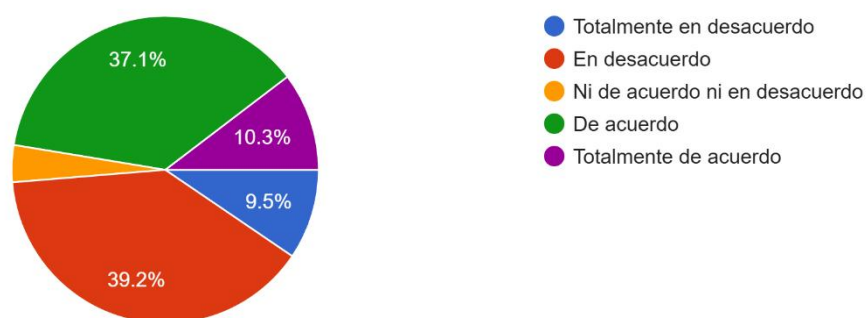


*Nota.* Se observa en la imagen que el 75.9% de encuestados debe esperar por más de 1 hora para recibir sus medicamentos en el punto de reclamación

**Figura 6***¿Padece discapacidad física o movilidad reducida?*

¿Padece alguna discapacidad física o movilidad reducida?

232 respuestas

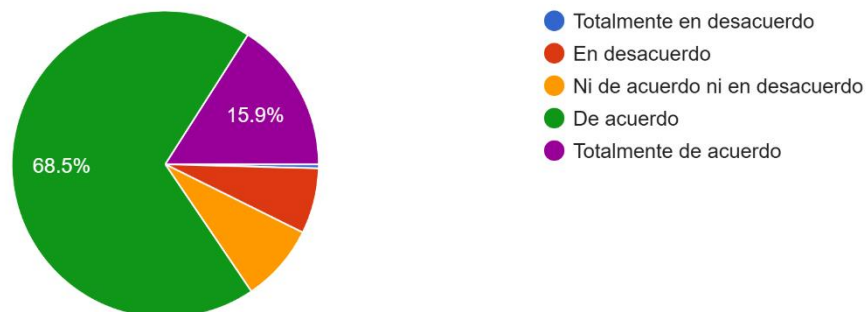


*Nota.* Se observa que la mayoría de encuestados (39.2%) no tienen una discapacidad o movilidad reducida

**Figura 7***¿Solicita con frecuencia sus medicamentos?*

¿Solicita frecuentemente sus medicamentos?

232 respuestas

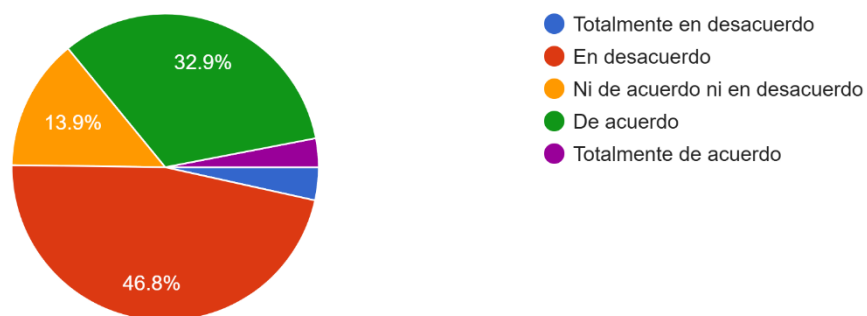


*Nota.* En la imagen se puede observar que el 68.5% de los que fueron encuestados, siendo la mayoría, solicita frecuentemente sus medicamentos

**Figura 8***¿Puede algún familiar reclamar los medicamentos?*

¿ En caso de no poder movilizarse, puede algún familiar reclamar sus medicamentos?

231 respuestas



*Nota.* Se puede observar que de los encuestados, el 46.8% no tiene un familiar que pueda reclamar los medicamentos de los pacientes, en este caso la persona debe por obligación

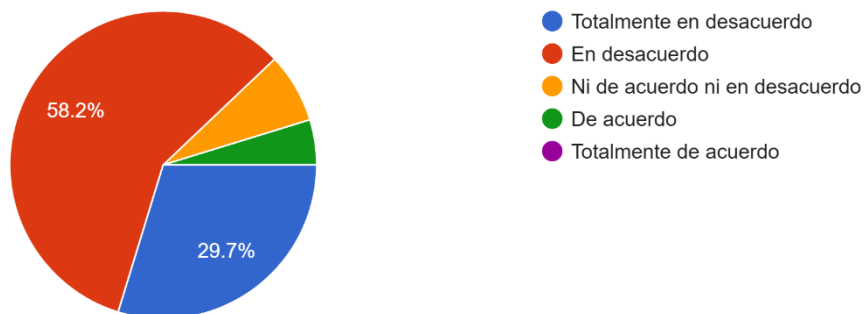
acercarse al punto de reclamación

### Figura 9

*¿Está satisfecho con el servicio prestado por el punto de reclamación?*

¿ Está satisfecho con el servicio prestado en el punto de reclamación de medicamentos?

232 respuestas



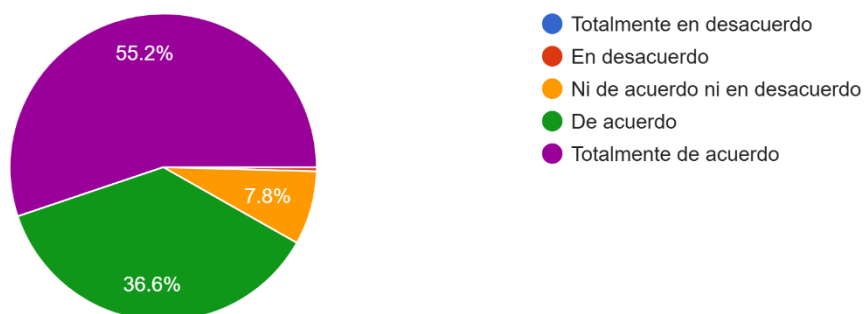
*Nota.* De acuerdo con la imagen anterior, el 58.2% de las personas encuestadas se encuentran insatisfechas con el servicio prestado por el punto de reclamación

### Figura 10

*¿Cree que una aplicación móvil que facilite la entrega de medicamentos a domicilio mejora el servicio que se presta actualmente?*

¿ Cree que una aplicación móvil que facilite la entrega de medicamentos a domicilio, podría mejorar el servicio que se presta actualmente?

232 respuestas



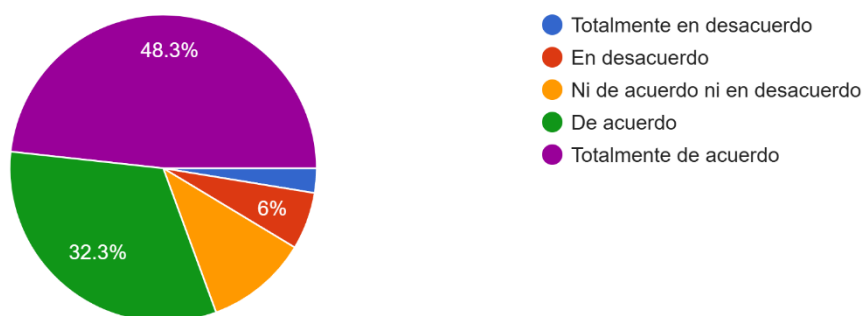
*Nota.* Se puede observar que, de acuerdo con las encuestas realizadas, la mayoría de pacientes

están totalmente de acuerdo con que exista una aplicación móvil que facilite la entrega de medicamentos a domicilio.

### Figura 11

¿Puede acceder fácilmente a aplicaciones en dispositivos tecnológicos como celular, Tablet o computador? ”.

¿Puede acceder fácilmente a aplicaciones en dispositivos tecnológicos como celular, tablet o computador?  
232 respuestas



*Nota.* Se puede observar que el 48.3% de los encuestados tienen acceso fácil a una aplicación en un dispositivo tecnológico

### Descripción del Proceso Actual de entregas de medicamentos en el barrio El Carmen - Tunjuelito

Actualmente, el proceso de entrega de medicamentos en el barrio El Carmen se realiza de manera presencial en el punto de reclamación asignado por la EPS. Las personas, en su mayoría adultos mayores y usuarios con movilidad reducida, deben desplazarse desde sus viviendas hasta el establecimiento, generalmente caminando o utilizando transporte informal. Este desplazamiento puede tomar entre 30 minutos y 1 hora, dependiendo de la distancia y de las condiciones de salud del usuario.

Una vez llegan al punto de dispensación, los usuarios deben realizar una fila de espera, que puede superar las dos horas, especialmente en horarios de mayor afluencia.

**Tabla 5**

*Problemas y dificultades del proceso actual*

<b>Categoría</b>	<b>Descripción</b>
Causas de esperas prolongadas	Alto volumen de usuarios diarios. Poca capacidad operativa de atención. Fallas intermitentes en el sistema EPS. Entrega del medicamento condicionada a disponibilidad en inventario. Retorno del usuario a su vivienda por sus propios medios.
Herramientas utilizadas actualmente	Sistema interno de autorizaciones de la EPS. Plataforma de verificación de historias clínicas. Registros manuales (control de filas, confirmación de entrega).
Dificultades del proceso actual	Sobrecarga del punto de atención que genera filas extensas. Fallos frecuentes en los sistemas de autorización EPS. Falta de personal suficiente para atender la demanda. Limitada accesibilidad para usuarios con movilidad reducida. Ausencia de alternativas digitales, obligando a la presencia física.

Como consecuencia, el proceso actual no garantiza un acceso oportuno y seguro a los medicamentos, afectando directamente la continuidad de los tratamientos y la calidad de vida de la población vulnerable del barrio El Carmen.

## **Modelo AS-IS del proceso de entregas de medicamentos en el barrio El Carmen –**

### **Tunjuelito**

#### *Descripción del Proceso en su Estado Actual*

Inicio

Desplazamiento del usuario desde su vivienda hacia el punto de reclamación de medicamentos.

Registro y toma de turno en el establecimiento de entrega.

Verificación de la orden médica y la autorización en el sistema de la EPS.

Espera en fila hasta ser llamado por el auxiliar de farmacia.

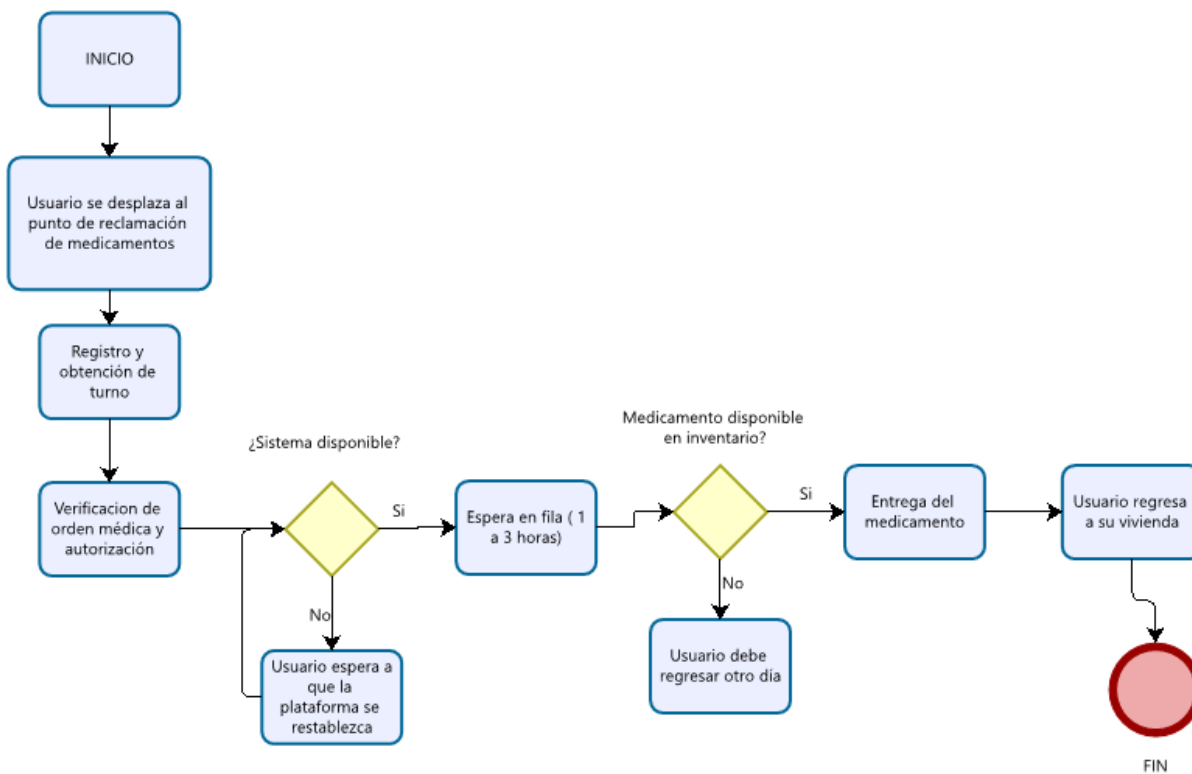
Confirmación de disponibilidad en inventario del medicamento solicitado.

Entrega física del medicamento al usuario o familiar encargado.

Retorno del usuario a su vivienda.

Fin

**Figura 12**  
*Proceso en su Estado Actual*



***Entradas del Proceso*****Tabla 6***Entradas del Proceso*

<b>Entrada</b>	<b>Descripción</b>
Orden médica	Emitida por médico tratante
Autorización EPS	Estado de aprobación para entrega
Documento del usuario	Identificación para registro
Presencia física del usuario o familiar	Requisito obligatorio para reclamar

***Salidas del Proceso*****Tabla 7***Salidas del Proceso*

<b>Salida</b>	<b>Descripción</b>
Medicamento entregado	Usuario recibe tratamiento
Registro de entrega	Firma o confirmación en el sistema
Nueva fecha de reclamación (si aplica)	En caso de medicamento no disponible

***Herramientas Utilizadas***

Plataforma de autorizaciones EPS  
 Sistema de dispensación de farmacia  
 Registro manual / listas de turnos  
 Infraestructura del punto de atención  
 Carnet / cédula del usuario

***Actores Involucrados*****Tabla 8***Actores Involucrados*

<b>Actor</b>	<b>Rol</b>
Usuario / Adulto mayor Familiar / Acompañante	Reclamante Apoyo en caso de movilidad reducida
Personal administrativo	Verificación de datos y autorizaciones

Auxiliar de farmacia	Entrega del medicamento
Sistema EPS	Plataforma de validación y registro

### ***Puntos Críticos (Cuellos de Botella Identificados)***

**Tabla 9**

*Cuellos de Botella Identificados*

<b>Problema</b>	<b>Consecuencia</b>
Largas distancias hasta el punto de entrega	Fatiga, riesgo de caídas y deserción del tratamiento
Filas prolongadas (1–3 horas)	Malestar, descontento y estrés en usuarios vulnerables
Caídas frecuentes del sistema EPS	Retrasos, repetición de visitas y acumulación de usuarios
Inventario limitado o desabastecimiento	Necesidad de regresar otro día, incumplimiento del tratamiento
Falta de acompañamiento familiar en muchos casos	Mayores dificultades para desplazarse y gestionar trámites

### **Identificación de puntos críticos Dolores (Pains) y necesidades de usuario**

En el proceso actual de entrega de medicamentos en el barrio El Carmen se han identificado diversas dificultades que afectan tanto a los usuarios como al personal que participa en la gestión del servicio. Estas molestias se relacionan principalmente con los tiempos de espera, el desplazamiento hasta el punto de entrega, la disponibilidad del medicamento y la capacidad de acompañamiento familiar. Reconocer estos puntos críticos permite comprender la experiencia real de los usuarios y fundamentar la necesidad de una solución más eficiente y accesible, como la implementación de una aplicación móvil que facilite y optimice el proceso. A continuación, se presenta la siguiente tabla donde se listan dichos problemas:

**Tabla 10**  
*Dolores (Pains) y necesidades de usuario*

<b>Actor/Grupo de interés</b>	<b>Dolor identificado</b>	<b>Implicación</b>	<b>Frecuencia/Severidad</b>
Adultos mayores usuarios del servicio	Deben desplazarse largas distancias hasta el punto de entrega.	Cansancio físico, riesgo de caídas y empeoramiento de condiciones de salud.	Alta: afecta a la mayoría debido a la edad y condiciones de movilidad.
Personas con movilidad reducida o discapacidad	Dificultad para permanecer en filas prolongadas y para llegar sin apoyo.	Mayor dependencia de terceros o imposibilidad de reclamar medicamentos.	Muy alta: representa barrera crítica para el acceso al tratamiento.
Personal de farmacia / punto de entrega	Sobrecarga de usuarios y atención lenta debido a fallas del sistema y alta demanda.	Estrés laboral, tiempos de respuesta lentos y atención menos personalizada.	Alta: ocurre diariamente en horarios de mayor afluencia.

### **Requisitos del sistema**

A partir del análisis del proceso actual de entrega de medicamentos en el barrio El Carmen y de los puntos críticos identificados, se definieron los requisitos necesarios para el diseño de la solución propuesta. Estos requisitos permiten establecer las funcionalidades que debe ofrecer el sistema, así como las condiciones de calidad, rendimiento, seguridad y accesibilidad que deben garantizarse para asegurar una experiencia adecuada para los usuarios. En este sentido, se clasifican en requisitos funcionales, que describen las acciones que el sistema debe realizar, y requisitos no funcionales, que especifican las características y restricciones bajo las cuales el sistema debe operar.

**Tabla 11**  
*Requisitos del sistema*

<b>Tipo de requisito</b>	<b>Código</b>	<b>Descripción</b>
Funcional	RF-01	El sistema debe permitir al usuario registrarse con datos básicos (nombre, documento, dirección, contacto).
Funcional	RF-02	El sistema debe permitir solicitar la entrega de medicamentos seleccionando fecha y orden médica.
Funcional	RF-03	El sistema debe notificar al usuario el estado de su solicitud (en trámite, aprobado, entregado).
Funcional	RF-04	El sistema debe permitir consultar el historial de solicitudes y entregas realizadas.
Funcional	RF-05	El sistema debe permitir registrar acompañantes o familiares autorizados para recoger el medicamento.
No Funcional	RNF-01	La aplicación debe ser fácil de usar, con interfaz intuitiva para adultos mayores.
No Funcional	RNF-02	El sistema debe garantizar seguridad y protección de datos personales conforme a la Ley de Habeas Data.
No Funcional	RNF-03	La aplicación debe ser accesible desde dispositivos móviles con conexión básica a internet.
No Funcional	RNF-04	El sistema debe responder en menos de 3 segundos a las solicitudes del usuario.

## **Rediseño del proceso de Aplicación para entrega de medicamentos MediExpress**

### ***Propuesta Estado Futuro (TO-BE)***

A partir del análisis del proceso actual (AS-IS), se identificaron diversos problemas que afectan la eficiencia y accesibilidad del sistema de entrega de medicamentos en el barrio El Carmen, tales como largos tiempos de espera, dificultades de desplazamiento, falta de

acompañamiento y fallas en el sistema de verificación. Con base en estos hallazgos, se propone un rediseño del proceso apoyado en una aplicación móvil, que permita optimizar la gestión de solicitudes, reducir tiempos de atención, mejorar la experiencia del usuario y facilitar el acceso a la información en tiempo real. Esta propuesta plantea un proceso más ágil, automatizado y accesible para los usuarios, especialmente aquellos que presentan condiciones de vulnerabilidad.

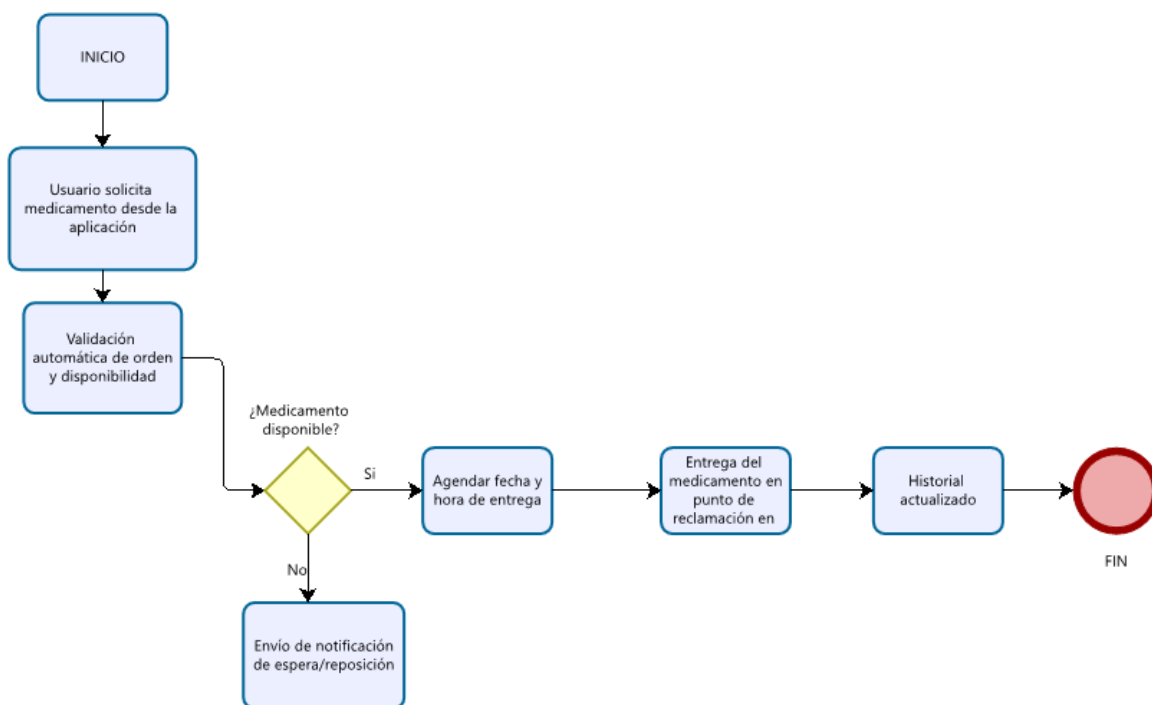
A continuación, se muestran estas características y mejoras propuestas:

**Tabla 12**  
*Propuesta Estado Futuro (TO-BE)*

<b>Mejora Propuesta</b>	<b>Descripción del Cambio</b>	<b>Beneficio Esperado</b>
Solicitud de medicamentos desde la aplicación	El usuario podrá solicitar la entrega desde su hogar.	Reduce desplazamientos y filas en el punto de atención.
Validación digital de la orden médica	La EPS validará automáticamente las órdenes registradas.	Disminuye tiempos administrativos y caídas de sistema.
Notificaciones en tiempo real	El usuario recibe alertas sobre estado de solicitud y disponibilidad.	Genera confianza y mejora la planificación del usuario.
Asignación de citas de recogida	La app permite elegir día y hora de entrega disponibles.	Evita aglomeraciones y reduce tiempos de espera.
Registro de acompañante autorizado	Posibilidad de designar un familiar para recoger el medicamento.	Facilita el proceso para usuarios con movilidad limitada.
Historial digital de medicamentos	Consulta de entregas previas y próximas fechas.	Mejora el control del tratamiento del paciente.

**Figura 13**

*Flujo del proceso de Aplicación para entrega de medicamentos MediExpress*



### ***Definición del MPV***

A partir del análisis de los requerimientos y de las necesidades identificadas en los usuarios, se establece la construcción de un Producto Mínimo Viable (MVP) que permita validar la propuesta de solución de manera funcional y eficiente. El MVP se enfoca en incluir únicamente las funcionalidades esenciales que aseguren el cumplimiento del objetivo principal: facilitar y optimizar el proceso de solicitud y entrega de medicamentos, especialmente para usuarios adultos mayores y personas con movilidad reducida.

La definición del MVP permite priorizar las características críticas del sistema, evitando sobrecargar el desarrollo con elementos que pueden ser incorporados en fases posteriores. Para ello, se utiliza la técnica de priorización MoSCoW, la cual clasifica las funcionalidades en términos de importancia y necesidad, garantizando que la primera versión del sistema sea útil,

accesible y operativa para los usuarios finales.

**Tabla 13**  
*MoSCoW*

<b>Categoría</b>	<b>Significado</b>
M (Must Have)	Funcionalidad esencial, sin ella el sistema no cumple su objetivo.
S (Should Have)	Importante, mejora el servicio, pero no es crítica para el funcionamiento básico.
C (Could Have)	Deseable, aporta valor, pero puede implementarse después.
W (Won't Have por ahora)	No se implementará en esta versión, puede quedar para versiones futuras.

**Tabla 14**  
*Técnica MoSCoW*

<b>Funcionalidad</b>	<b>Categoría (MoSCoW)</b>	<b>Justificación</b>
Registro de usuario en la aplicación	M (Must Have)	Es indispensable para identificar al usuario y permitir acceso al sistema.
Solicitud de medicamentos desde la aplicación	M (Must Have)	Es la funcionalidad central que reemplaza el desplazamiento y la fila presencial.
Validación digital de orden médica	M (Must Have)	Permite automatizar el proceso y evitar caídas o demoras en el punto de entrega.
Notificaciones del estado de la solicitud	S (Should Have)	Mejora la comunicación con el usuario y reduce la incertidumbre.
Agendamiento de fecha y hora para recoger el medicamento	S (Should Have)	Permite evitar aglomeraciones y reducir espera en el punto.
Registro de acompañante autorizado	C (Could Have)	Facilita el apoyo familiar, especialmente para usuarios con movilidad reducida.
Historial de entregas y solicitudes previas	C (Could Have)	Aporta organización y seguimiento del tratamiento del usuario.
Chat o canal directo con la EPS	W (Won't Have por ahora)	No es necesario para el funcionamiento inicial y puede implementarse en versiones futuras.

### ***Épicas del Proyecto***

Las épicas representan iniciativas estratégicas que requieren coordinación a gran escala, impactan múltiples procesos y aportan valor significativo al negocio y a los usuarios. Cada épica

descrita a continuación se encuentra orientada a mejorar el proceso de solicitud y entrega de medicamentos mediante la implementación de una aplicación móvil accesible para la comunidad del barrio El Carmen.

**Tabla 15**  
*Épicas del proyecto*

<b>ID</b>	<b>Épica</b>	<b>Descripción</b>
EP01	Digitalización del proceso de solicitud de medicamentos	Implementar un sistema digital que permita a los usuarios realizar la solicitud de medicamentos desde sus dispositivos móviles, evitando desplazamientos innecesarios y reduciendo la dependencia de la atención física en el punto de entrega. Objetivo de Negocio: Disminuir la saturación del punto de atención y asegurar un acceso más rápido y seguro a los tratamientos.
EP02	Automatización de la validación y autorización médica	Integrar el sistema con la plataforma de la EPS para validar automáticamente las órdenes y autorizaciones médicas, evitando interrupciones y tiempos de espera relacionados con fallas administrativas o del sistema.  Objetivo de Negocio: Optimizar el flujo de verificación, disminuir errores operativos y mejorar la continuidad de tratamiento para los usuarios.
EP03	Implementación de sistema de seguimiento y notificaciones en tiempo real	Incorporar un módulo de comunicación que informe al usuario el estado de su solicitud, disponibilidad del medicamento y recordatorios de retiro o renovación.  Objetivo de Negocio: Fortalecer la experiencia del usuario y aumentar la confianza en el servicio mediante

		información clara y oportuna.
EP04	Gestión de acompañantes y accesibilidad para usuarios vulnerables	<p>Crear una funcionalidad que permita registrar acompañantes autorizados y adaptar la interfaz para facilitar su uso por parte de adultos mayores y personas con movilidad reducida.</p> <p>Objetivo de Negocio: Garantizar que la solución sea inclusiva, accesible y orientada a la población vulnerable del barrio.</p>

---

### ***Historias de Usuario***

Las historias de usuario constituyen un elemento fundamental dentro de la metodología ágil Scrum, ya que permiten describir de manera sencilla y comprensible las funcionalidades que el sistema debe ofrecer desde la perspectiva del usuario final. Cada historia captura una necesidad, expectativa o requisito expresado en un lenguaje natural que facilita la comunicación entre el Product Owner, el equipo de desarrollo y los demás interesados en el proyecto. Para garantizar que estas historias sean claras, viables y aporten valor real al producto, se emplea el formato INVEST, un conjunto de criterios que asegura que cada historia esté adecuadamente estructurada. El acrónimo INVEST significa:

I – Independent (Independiente): la historia debe poder desarrollarse de manera autónoma, sin depender estrictamente de otras.

N – Negotiable (Negociable): debe ser flexible y permitir discusión y refinamiento entre el equipo y el Product Owner.

V – Valuable (Valiosa): tiene que ofrecer un beneficio directo al usuario o al negocio.

E – Estimable (Estimable): debe poder evaluarse en términos de esfuerzo o complejidad.

S – Small (Pequeña): debe ser lo suficientemente corta para completarse dentro de un sprint.

T – Testable (Comprobable): debe contar con criterios claros que permitan verificar su cumplimiento.

Cada historia se redacta siguiendo la fórmula:

“Como [tipo de usuario], quiero [funcionalidad], para [objetivo o beneficio]”.

Esta estructura facilita comprender quién solicita la función, qué requiere y por qué es importante para la solución. A continuación, se muestra el formato utilizado para este fin:

#### Figura 14

##### Formato tarjeta de Historia de Usuario

Historias de usuario:						Título:	
Como ,							
Quiero ,							
Para poder .							
Criterios de aceptación:							
1.							
2.							
3.							
4.							
5.							
I N V E S T						Valor:	
						Estimación:	
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>						¿Cumple con la definición de listo del equipo? <input type="checkbox"/>	

En este proyecto, las historias de usuario se elaboraron aplicando rigurosamente el formato INVEST, asegurando que cada una de ellas aporte valor al desarrollo del sistema MediExpress y permita una planificación efectiva de los sprints. Asimismo, cada historia se complementa con criterios de aceptación que definen el comportamiento esperado y permiten validar su cumplimiento mediante pruebas funcionales.

De acuerdo con lo expuesto previamente, a continuación, se presentan las diferentes Historias de Usuario empleadas para el desarrollo del proyecto MediExpress. Su propósito principal es definir con precisión los requisitos funcionales y no funcionales del software, tomando como referencia las necesidades identificadas en el planteamiento del problema y la solución propuesta.

Cada historia de usuario describe de manera detallada las funcionalidades esperadas, los actores involucrados en el proceso y los criterios necesarios para aceptar correctamente cada funcionalidad. Esto garantiza una adecuada planificación, priorización y ejecución del desarrollo, asegurando que el producto final responda a las expectativas y necesidades del usuario.

**Figura 15**

*HU-001 Registro de usuario básico*

<b>Historia de Usuario</b>	<b>ID:</b> HU-001	<b>Título:</b> Registro de usuario básico.
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero</b> registrarme con mis datos básicos, <b>Para</b> poder crear una cuenta y acceder a las funciones de la aplicación.	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El sistema debe registrar nombre, documento, teléfono y dirección</li> <li>2. No debe permitir cuentas duplicadas.</li> <li>3. El sistema debe validar campos obligatorios.</li> <li>4. El usuario debe recibir confirmación del registro.</li> <li>5. El registro debe quedar almacenado correctamente.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>5 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 16**

*HU-002 Inicio de sesión seguro*

<b>Historia de Usuario</b>	<b>ID:</b> HU-002	<b>Título:</b> Inicio de sesión seguro
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero</b> registrarme con mis datos básicos, <b>Para</b> poder crear una cuenta y acceder a las funciones de la aplicación.	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El sistema debe validar credenciales</li> <li>2. Debe generar un token válido.</li> <li>3. Debe mostrar error si los datos no coinciden.</li> <li>4. El usuario debe poder cerrar sesión.</li> <li>5. El token debe expirar de forma segura.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>3 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 17**  
*HU-003 Recuperar contraseña*

<b>Historia de Usuario</b>	<b>ID:</b> HU-003	<b>Título:</b> Recuperar contraseña
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero recuperar mi contraseña</b> <b>Para poder restablecer mi acceso en caso de olvido.</b>	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. Debe enviarse un enlace o código por correo/SMS.</li> <li>2. El enlace debe permitir crear una nueva contraseña</li> <li>3. El enlace debe tener expiración.</li> <li>4. Debe validarse identidad del usuario.</li> <li>5. Debe confirmarse el restablecimiento.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>3 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 18**  
*HU-004 Editar perfil*

<b>Historia de Usuario</b>	<b>ID:</b> HU-004	<b>Título:</b> Editar perfil
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero editar mi perfil,</b> <b>Para poder mantener actualizada mi información personal.</b>	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El sistema debe registrar nombre, documento, teléfono y dirección</li> <li>2. No debe permitir cuentas duplicadas.</li> <li>3. El sistema debe validar campos obligatorios.</li> <li>4. El usuario debe recibir confirmación del registro.</li> <li>5. El registro debe quedar almacenado correctamente.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>5 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 19**  
*Solicitar medicamento*

<b>Historia de Usuario</b>	<b>ID:</b> HU-005	<b>Título:</b> Solicitar medicamento
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero</b> solicitar un medicamento, <b>Para</b> poder iniciar el proceso de entrega.	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El formulario debe validar campos obligatorios</li> <li>2. La solicitud debe guardarse correctamente</li> <li>3. Debe generarse un ID de solicitud</li> <li>4. Debe permitir seleccionar medicamento.</li> <li>5. Debe mostrar confirmación al usuario</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>8 horas</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 20**  
*Adjuntar foto de fórmula*

<b>Historia de Usuario</b>	<b>ID:</b> HU-006	<b>Título:</b> Adjuntar foto de fórmula
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero</b> registrarme con mis datos básicos, <b>Para</b> poder crear una cuenta y acceder a las funciones de la aplicación.	
<ol style="list-style-type: none"> <li>1. Debe permitir subir imagen en formatos estándar (jpeg/png).</li> <li>2. Debe controlar tamaño máximo permitido.</li> <li>3. Debe mostrar vista previa</li> <li>4. Debe almacenar la imagen asociada a la solicitud.</li> <li>5. Debe notificar error si la carga falla.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>5 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 21**  
*Ver estado de solicitudes*

<b>Historia de Usuario</b>	<b>ID:</b> HU-007	<b>Título:</b> Ver estado de solicitudes
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero</b> ver el estado de mis solicitudes, <b>Para</b> poder hacer seguimiento al proceso.	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. Debe mostrar lista de solicitudes</li> <li>2. Debe mostrar estados: pendiente, validada, entregada.</li> <li>3. Debe permitir ver detalle de la solicitud</li> <li>4. Debe actualizar estados en tiempo real</li> <li>5. Debe filtrar por estado de solicitud</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>5 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 22**  
*Cancelar solicitud*

<b>Historia de Usuario</b>	<b>ID:</b> HU-008	<b>Título:</b> Cancelar solicitud
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero</b> cancelar una solicitud de medicamento, <b>Para</b> poder corregir errores o detener un trámite.	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. Solo debe permitir cancelación en estado “pendiente”.</li> <li>2. Debe registrar motivo (opcional).</li> <li>3. Debe cambiar estado a “cancelada”.</li> <li>4. Debe notificar al usuario la cancelación realizada exitosamente.</li> <li>5. No debe permitir revertir cancelación</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>3 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 23**  
*Confirmación de EPS.*

<b>Historia de Usuario</b>	<b>ID:</b> HU-009	<b>Título:</b> Confirmación de EPS.
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero</b> recibir confirmación cuando la EPS valide mi autorización, <b>Para</b> poder saber si puedo recibir el medicamento.	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El sistema debe integrarse con sistema EPS.</li> <li>2. El estado debe cambiar automáticamente.</li> <li>3. Debe registrar fecha/hora de validación.</li> <li>4. Debe notificar al usuario la confirmación de la EPS</li> <li>5. Debe actualizarse en la lista de solicitudes.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>8 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 24**  
*Gestión del catálogo de medicamentos*

<b>Historia de Usuario</b>	<b>ID:</b> HU-010	<b>Título:</b> Gestión del catálogo de medicamentos
<b>Descripción</b>	<b>Como administrador,</b> <b>Quiero</b> gestionar el catálogo de medicamentos, <b>Para</b> mantener la información actualizada y disponible.	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El sistema debe crear el medicamento para registrarlo en el inventario</li> <li>2. El sistema debe permitir la edición del medicamento para actualizarlo en el inventario</li> <li>3. El sistema debe permitir la visualización del detalle del medicamento según sea la necesidad del usuario(administrador/dispensador)</li> <li>4. El sistema debe permitir la eliminación del medicamento y que no aparezca en el inventario.</li> <li>5. El sistema debe permitir detectar duplicados del medicamento para corregirlo</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>5 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 25***Ver descripción de medicamento*

<b>Historia de Usuario</b>	<b>ID:</b> HU-011	<b>Título:</b> Ver descripción de medicamento
<b>Descripción</b>	Como usuario, <b>Quiero</b> ver la descripción de un medicamento, <b>Para</b> informarme antes de solicitarlo.	
<b>Criterios de aceptación:</b>		
<ol style="list-style-type: none"> <li>1. El sistema debe mostrar el nombre del medicamento.</li> <li>2. El sistema debe permitir la visualización del detalle de la dosis del medicamento</li> <li>3. El sistema debe permitir la visualización de la descripción del medicamento</li> <li>4. El sistema debe permitir la visualización de las advertencias del medicamento</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>3 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 26***Asignar mensajero*

<b>Historia de Usuario</b>	<b>ID:</b> HU-012	<b>Título:</b> Asignar mensajero
<b>Descripción</b>	Como administrador, <b>Quiero</b> asignar un mensajero a una solicitud, <b>Para</b> garantizar que la entrega sea realizada correctamente.	
<b>Criterios de aceptación:</b>		
<ol style="list-style-type: none"> <li>1. El administrador debe poder seleccionar un mensajero disponible.</li> <li>2. La solicitud debe quedar asociada correctamente al mensajero asignado.</li> <li>3. El sistema debe validar campos obligatorios.</li> <li>4. La entrega debe quedar registrada como ligada a un mensajero.</li> <li>5. El sistema debe permitir registrar fecha de asignación para cada mensajero y entrega.</li> <li>6. El sistema debe permitir actualizar estado de entrega.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>5 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 27**  
*Ver entregas asignadas*

<b>Historia de Usuario</b>	<b>ID:</b> HU-013	<b>Título:</b> Ver entregas asignadas.
<b>Descripción</b>	Como mensajero, <b>Quiero</b> ver mis entregas asignadas, <b>Para</b> organizar mis rutas y tiempo.	
<b>Criterios de aceptación:</b>		
<ol style="list-style-type: none"> <li>1. El mensajero solo debe visualizar entregas que le han sido asignadas.</li> <li>2. El sistema debe permitir filtrar entregas según su estado.</li> <li>3. El sistema debe mostrar la lista de entregas asignadas de manera clara.</li> <li>4. El sistema debe permitir filtrar las entregas por estado.</li> <li>5. El sistema debe permitir ver detalle de la entrega.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>5 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 28**  
*Actualizar estado de entrega*

<b>Historia de Usuario</b>	<b>ID:</b> HU-014	<b>Título:</b> Actualizar estado de entrega.
<b>Descripción</b>	Como mensajero, <b>Quiero</b> actualizar el estado de una entrega, <b>Para</b> reportar su avance en tiempo real.	
<b>Criterios de aceptación:</b>		
<ol style="list-style-type: none"> <li>1. El mensajero debe poder actualizar el estado de la entrega a “en ruta”.</li> <li>2. El mensajero debe poder actualizar el estado de la entrega a “entregada”.</li> <li>3. El sistema debe registrar la hora exacta del cambio de estado.</li> <li>4. El sistema debe permitir cambiar estado a “entregada”.</li> <li>5. El sistema debe permitir registrar hora del cambio.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>5 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 29**  
*Firma digital.*

<b>Historia de Usuario</b>	<b>ID:</b> HU-015	<b>Título:</b> Firma digital.
<b>Descripción</b>	<b>Como mensajero,</b> <b>Quiero registrar la firma digital del usuario</b> <b>Para confirmar la entrega oficialmente.</b>	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El sistema debe permitir capturar la firma digital del usuario receptor.</li> <li>2. El sistema debe permitir adjuntar una fotografía como evidencia de entrega.</li> <li>3. La confirmación de entrega debe quedar registrada correctamente en el sistema.</li> <li>4. El sistema debe permitir adjuntar foto como evidencia</li> <li>5. El sistema debe permitir guardar registro asociado a la entrega.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>8 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 30**  
*Ver ubicación del usuario*

<b>Historia de Usuario</b>	<b>ID:</b> HU-016	<b>Título:</b> Ver ubicación del usuario
<b>Descripción</b>	<b>Como mensajero,</b> <b>Quiero ver la ubicación y dirección del usuario,</b> <b>Para poder optimizar la ruta de entrega.</b>	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El sistema debe mostrar la ubicación del usuario</li> <li>2. Debe visualizarse un mapa con la dirección del usuario.</li> <li>3. Debe presentarse la ruta a seguir para completar la entrega.</li> <li>4. El usuario debe poder ver el mapa de ubicación.</li> <li>5. El sistema debe permitir integración con Google Maps o similar.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>8 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 31**  
*Notificaciones por cambio de estado.*

<b>Historia de Usuario</b>	<b>ID:</b> HU-017	<b>Título:</b> Notificaciones por cambio de estado.
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero</b> recibir notificaciones cuando mi solicitud cambie de estado, <b>Para</b> poder estar informado del proceso.	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El sistema debe enviar automáticamente una notificación cuando cambie el estado de la solicitud.</li> <li>2. El sistema debe registrar el envío de la notificación.</li> <li>3. El usuario debe poder configurar si desea recibir estas notificaciones.</li> <li>4. El sistema debe permitir el registro del envío.</li> <li>5. El registro debe quedar almacenado correctamente.</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>5 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 32**  
*Alertas masivas.*

<b>Historia de Usuario</b>	<b>ID:</b> HU-018	<b>Título:</b> Alertas masivas.
<b>Descripción</b>	<b>Como administrador,</b> <b>Quiero</b> enviar alertas masivas en caso de fallos del sistema, <b>Para</b> poder mantener informados a los usuarios.	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El sistema debe permitir enviar una alerta a todos los usuarios.</li> <li>2. El administrador debe poder redactar y enviar mensajes masivos.</li> <li>3. La alerta debe enviarse inmediatamente a todos los destinatarios.</li> <li>4. El sistema debe permitir notificación inmediata en caso de fallo del sistema</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>5 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

**Figura 33**  
*Recordatorios de medicamentos*

<b>Historia de Usuario</b>	<b>ID:</b> HU-019	<b>Título:</b> Recordatorios de medicamentos.
<b>Descripción</b>	<b>Como usuario,</b> <b>Quiero</b> recibir recordatorios de toma de medicamentos, <b>Para</b> poder cumplir adecuadamente mi tratamiento.	
<b>Criterios de aceptación:</b> <ol style="list-style-type: none"> <li>1. El usuario debe poder programar recordatorios de medicamentos</li> <li>2. El sistema debe enviar automáticamente los recordatorios programados.</li> <li>3. El usuario debe poder configurar los horarios de sus recordatorios.</li> <li>4. El usuario debe poder recibir envío automático de los recordatorios</li> </ol>		
<b>I N V E S T</b>	<b>Estimación</b>	<b>3 puntos</b>
<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	¿Cumple con la definición de listo del equipo?	<input type="checkbox"/>

## **Diseño de la arquitectura de la aplicación móvil para la entrega de medicamentos**

La arquitectura de aplicaciones, o arquitectura de software, constituye la estructura fundamental de un sistema y orienta la organización de sus componentes, su comportamiento y las relaciones entre ellos. Su propósito es garantizar que la solución tecnológica responda adecuadamente tanto a los requisitos técnicos como a las necesidades del negocio, minimizando los riesgos asociados al proceso de construcción del software y asegurando su evolución a largo plazo (Bass, Clements & Kazman, 2013).

La aplicación MediExpress se desarrolla con el propósito de mejorar el proceso de solicitud y entrega de medicamentos para adultos mayores y personas con movilidad reducida, facilitando el acceso a tratamientos médicos sin necesidad de desplazamientos prolongados. Para lograrlo, se emplean tecnologías modernas que garantizan un desarrollo ágil, seguro y mantenible, así como una arquitectura alineada con los requisitos funcionales y no funcionales definidos en la fase de Análisis e Identificación de Requisitos.

En el frontend, MediExpress utiliza Angular, un framework basado en TypeScript que permite construir interfaces modulares, accesibles y optimizadas para dispositivos móviles. Este framework es ideal para implementar una PWA (Progressive Web App), lo que permite que la aplicación funcione incluso en condiciones de conectividad limitada, un aspecto esencial para la población objetivo.

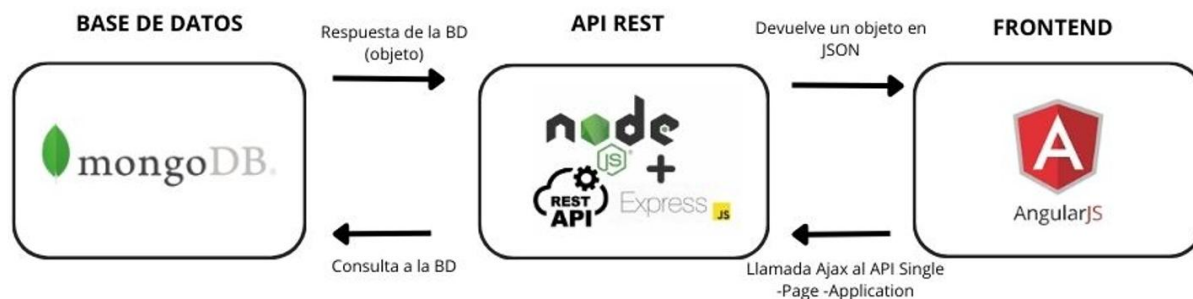
En el backend, se emplea Node.js con Express , lo que garantiza una lógica de negocio consistente, reutilizable y con mecanismos de seguridad modernos. La base de datos utilizada es

MongoDB, una solución NoSQL flexible que se adapta a la naturaleza dinámica de los datos del sistema, como solicitudes, usuarios, entregas y estados del proceso.

Respecto a la estructura del sistema, MediExpress implementa una arquitectura monolítica en capas (Presentación – Aplicación – Dominio – Datos). Este patrón permite integrar todas las funcionalidades en una única aplicación organizada de forma modular, reduciendo complejidad, facilitando la implementación inicial y garantizando mantenibilidad. La separación por capas asegura un bajo acoplamiento interno, mejora la claridad del código y facilita la evolución futura del proyecto.

Con esta combinación de lenguajes, frameworks y una arquitectura monolítica bien estructurada, MediExpress ofrece una solución tecnológica robusta, accesible y alineada con las necesidades reales de los usuarios y los requisitos establecidos para el sistema. A continuación, se presenta el modelo MEAN usado para el proyecto MediExpress:

**Figura 34**  
*Modelo (Mongo-Express-Angular-Node)*



Fuente. Tomado de <https://www.mongodb.com/resources/languages/mean-stack>

## **Objetivo Arquitectónico**

El objetivo arquitectónico de MediExpress es definir la estructura tecnológica que sustentará el desarrollo y operación del sistema, asegurando que su diseño cumpla con los requisitos funcionales y no funcionales identificados en la fase de análisis. Esta arquitectura debe garantizar escalabilidad, permitiendo la evolución futura del sistema; seguridad, mediante mecanismos de protección de datos y control de acceso; mantenibilidad, gracias a la organización modular del código; y rendimiento, optimizando tiempos de respuesta y uso de recursos. Con ello, se busca construir una solución sólida, confiable y alineada con las necesidades reales de los usuarios finales y las condiciones operativas del proceso de entrega de medicamentos.

## **Principios Arquitectónicos**

La arquitectura propuesta para MediExpress se fundamenta en un conjunto de principios que garantizan la calidad técnica del sistema, su mantenibilidad y su capacidad para evolucionar conforme a las necesidades del proceso de entrega de medicamentos. Estos principios orientan el diseño y la implementación del software para asegurar un producto robusto, seguro y adaptable.

### ***API First***

El diseño del sistema prioriza la definición de interfaces y contratos de comunicación antes del desarrollo de las funcionalidades internas. Esto asegura coherencia entre frontend y backend, facilita la integración con servicios externos (EPS, notificaciones) y reduce errores de compatibilidad.

### ***Modularidad***

Aunque la arquitectura es monolítica, el código se organiza en módulos bien delimitados que permiten separar funcionalidades (usuarios, solicitudes, entregas, notificaciones). Esto simplifica el mantenimiento, mejora la claridad y facilita futuras ampliaciones del sistema.

### ***Bajo acoplamiento y alta cohesión***

Los componentes del sistema se diseñan para depender mínimamente entre sí, a la vez que concentran dentro de cada módulo responsabilidades claras y específicas. Esto reduce el impacto de cambios y favorece la evolución controlada del software.

### ***Seguridad por defecto (OWASP)***

El sistema adopta medidas de seguridad desde su concepción, siguiendo lineamientos OWASP como validación de entradas, protección contra ataques comunes (XSS, CSRF, inyección), cifrado de datos y correcto manejo de autenticación y permisos mediante JWT y RBAC.

### ***Cloud Ready***

Aunque el despliegue inicial se realiza en un VPS, la arquitectura está preparada para migrar fácilmente hacia entornos en la nube gracias al uso de contenedores Docker, automatización, configuración externa y diseño independiente de la infraestructura.

### ***12-Factor App***

El sistema se desarrolla siguiendo los principios del modelo 12-Factor, como separación de configuración, portabilidad, ejecución en entornos replicables y uso de servicios externos gestionados, lo que facilita su despliegue, escalado y mantenimiento.

### ***Reutilización de componentes***

Se prioriza la creación de componentes reutilizables tanto en el frontend (Angular) como en el backend (servicios, controladores y modelos), reduciendo duplicidad de código y acelerando el desarrollo.

### ***Observabilidad integrada***

Se incorporan mecanismos de monitoreo, métricas, logs estructurados y trazabilidad desde las primeras etapas. Esto permite detectar fallos, analizar comportamientos y mejorar la confiabilidad del sistema durante operación y mantenimiento.

### **Drivers Arquitecturales**

Los drivers arquitecturales son los factores clave que influyen directamente en el diseño de la arquitectura de software y determinan las decisiones técnicas necesarias para construir un sistema estable, seguro y alineado con los objetivos del negocio. Estos drivers incluyen tanto requisitos funcionales como atributos de calidad, restricciones tecnológicas, necesidades del usuario y consideraciones del entorno operativo (Bass, Clements & Kazman, 2013).

En el proyecto MediExpress, los drivers arquitecturales permiten orientar la solución tecnológica hacia una plataforma confiable para la solicitud y entrega de medicamentos, especialmente dirigida a adultos mayores y personas con movilidad reducida. Debido a la naturaleza sensible del dominio —servicios de salud, manejo de datos personales y procesos logísticos—, estos drivers adquieren un papel central para garantizar que la arquitectura cumpla con expectativas de seguridad, disponibilidad, desempeño, accesibilidad y facilidad de uso.

Identificar y documentar los drivers arquitecturales permite justificar las decisiones adoptadas, asegurar coherencia entre los requisitos y la estructura del sistema, y facilitar la evolución futura de MediExpress. Además, estos drivers sirven como guía para evaluar alternativas tecnológicas, detectar riesgos tempranos y mantener la alineación entre el producto y las necesidades reales de sus usuarios y actores involucrados.

**Tabla 16**  
*Drivers arquitecturales*

<b>Driver</b>	<b>Tipo (Negocio / Técnico / Entorno)</b>	<b>Implicación Arquitectónica</b>
Adultos mayores con baja alfabetización digital	Negocio / Usuario	Interfaz simplificada, botones grandes, flujos cortos y diseño accesible.
Necesidad de respuesta rápida en solicitudes	Técnico	Usar indexación en BD, minimizar latencia y uso de colas si es necesario.
Seguridad de datos sensibles (información médica)	Técnico / Negocio	Aplicar OWASP, validación estricta
Escalabilidad futura del sistema	Técnico	Arquitectura monolítica modular,
Dependencia de servicios externos (EPS, notificaciones)	Técnico	Diseñar módulos desacoplados, manejo de errores, timeouts y reintentos controlados.

### **Atributos de Calidad (Requisitos No Funcionales)**

Los atributos de calidad, también conocidos como requisitos no funcionales, representan las características esenciales que determinan el comportamiento global de un sistema más allá de sus funcionalidades. Estos atributos definen qué tan seguro, confiable, accesible, eficiente y mantenible es un software, influyendo directamente en la experiencia del usuario y en la capacidad del sistema para operar adecuadamente en ambientes reales (Bass, Clements & Kazman, 2013).

En el contexto del proyecto MediExpress, una aplicación móvil orientada a facilitar la solicitud y entrega de medicamentos a adultos mayores y personas con movilidad reducida, los atributos de calidad adquieren un papel fundamental. Dada la sensibilidad de los datos manejados, la necesidad de una operación continua y la importancia de ofrecer una plataforma accesible para usuarios con limitaciones físicas o tecnológicas, los requisitos no funcionales se convierten en elementos críticos para garantizar la seguridad, el rendimiento, la disponibilidad y la usabilidad del sistema.

Definir estos atributos permite establecer criterios que guiarán las decisiones arquitectónicas, la selección de tecnologías, las estrategias de seguridad, el diseño de la interfaz y los mecanismos de despliegue y mantenimiento. Además, proporcionan una base para la evaluación de la calidad del producto final, asegurando que MediExpress cumpla con las expectativas del negocio y con las necesidades reales de sus usuarios finales.

### **Tabla 17**

*Atributos de Calidad*

<b>Atributo</b>	<b>Escenario (Estímulo → Respuesta)</b>	<b>Criterio de Aceptación / Métrica</b>
Rendimiento	El usuario abre la aplicación y solicita cargar sus solicitudes activas, el sistema responde mostrando la información	Tiempo de carga < 3 segundos
Disponibilidad	El usuario intenta acceder al sistema durante horas pico, el sistema debe seguir funcionando sin interrupciones	Disponibilidad $\geq$ 99 % mensual
Seguridad	Un usuario inicia sesión y realiza una solicitud el sistema valida token, permisos y datos	Autenticación mediante JWT, , cifrado TLS
Usabilidad	Un adulto mayor navega por la app para solicitar un medicamento, la interfaz debe ser clara y accesible	Cumplimiento de pautas WCAG, lectura fácil, botones grandes
Escalabilidad	Incremento del 50% en solicitudes diarias, el sistema debe responder sin degradación significativa	Uso de recursos estable (< 70% CPU), respuesta < 4 s
Confiabilidad	Fallo temporal en internet, la app debe continuar operando	Funcionalidad offline con sincronización diferida sin pérdida de datos

**Vistas Arquitectónicas (Modelo C4)**

La documentación arquitectónica de un sistema requiere una representación clara, estructurada y comprensible de sus elementos principales. El Modelo C4, propuesto por Simón Brown, ofrece una forma sencilla y estandarizada de describir la arquitectura mediante vistas jerárquicas que permiten comunicar el sistema desde distintos niveles de detalle, facilitando el entendimiento tanto para perfiles técnicos como para stakeholders no expertos.

En el proyecto MediExpress, el uso del Modelo C4 resulta especialmente adecuado, debido a la necesidad de representar cómo la aplicación móvil interactúa con su ecosistema tecnológico para soportar procesos sensibles como la solicitud, validación y entrega de medicamentos. Las vistas C4 permiten modelar el sistema desde una perspectiva amplia

usuarios, sistemas externos y contexto operativo, hasta su descomposición en contenedores y componentes específicos, haciendo evidente la estructura interna del software y sus responsabilidades.

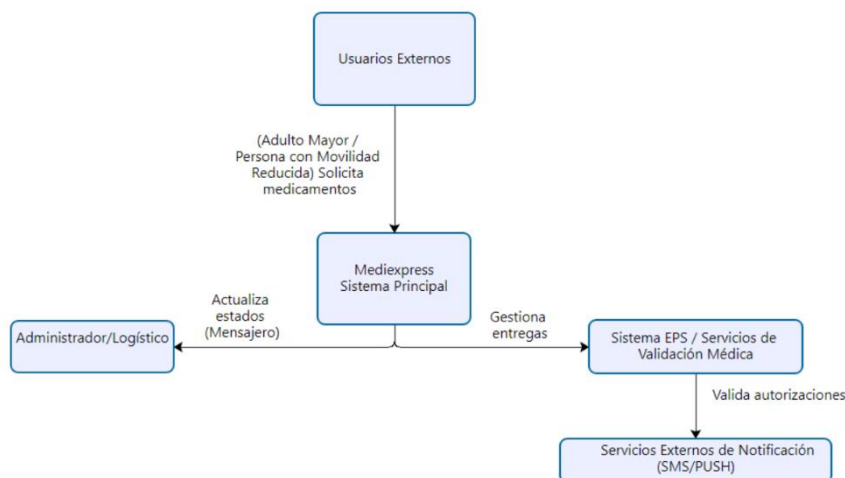
Al emplear este modelo, se documenta la arquitectura de MediExpress de forma clara, coherente y alineada con los requisitos funcionales y no funcionales definidos. Esto facilita la comunicación entre desarrolladores, diseñadores, administradores y otros actores del proyecto, reduce ambigüedades en la implementación y proporciona una referencia sólida para la evolución futura de la plataforma.

#### *Diagrama de Contexto (C4-1)*

El sistema MediExpress es una plataforma digital que permite a adultos mayores y personas con movilidad reducida solicitar medicamentos y coordinar su entrega desde una aplicación móvil tipo PWA. Interactúa con actores humanos y sistemas externos que participan en el proceso logístico, médico y operativo.

#### **Figura 35**

### Diagrama de Contexto



### Diagrama de Contenedores (C4-2)

El Diagrama de Contenedores (C4-2) representa el segundo nivel del Modelo C4 y permite describir la estructura del sistema desde una perspectiva de alto nivel, mostrando sus principales contenedores, aplicaciones, servicios, bases de datos y componentes ejecutables, así como las interacciones que existen entre ellos. Este nivel es fundamental para comprender cómo se distribuyen las responsabilidades dentro del sistema y cómo los elementos tecnológicos colaboran para ofrecer las funcionalidades del producto.

En el caso de MediExpress, el diagrama de contenedores facilita visualizar la arquitectura de la solución tecnológica propuesta, identificando elementos como la aplicación móvil, el backend monolítico, la base de datos, los servicios externos de notificaciones y los sistemas de validación de EPS. Esta representación permite evidenciar cómo fluye la información, qué tecnologías intervienen en cada contenedor y cómo se mantiene la separación lógica entre la interfaz de usuario, la lógica de negocio y la infraestructura.

Además, el diagrama C4-2 proporciona una vista esencial para el diseño, la implementación y la toma de decisiones arquitectónicas, ya que ayuda a asegurar que los contenedores cumplan con los atributos de calidad definidos, tales como seguridad, rendimiento, escalabilidad y mantenibilidad. De esta forma, el modelo contribuye a que MediExpress sea una solución coherente, integrada y preparada para su evolución futura.

El sistema está diseñado con una arquitectura monolítica modular, desplegada en contenedores Docker. Los principales contenedores son:

Frontend Móvil PWA (Angular)

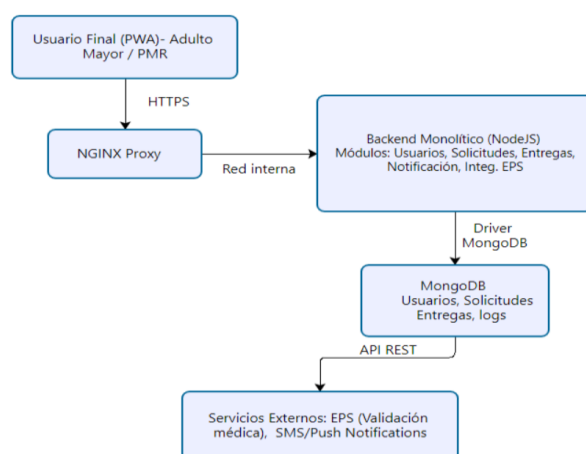
Backend Monolítico (Node.js + Express)

Base de Datos (MongoDB)

Nginx (Reverse Proxy)

Servicios externos: EPS, SMS/Push

**Figura 36**  
*Diagrama de Contenedores*



### ***Diagrama de Componentes (C4-3)***

El Diagrama de Componentes (C4-3) constituye el tercer nivel del Modelo C4 y proporciona una representación detallada de cómo se organiza internamente cada uno de los contenedores definidos en la arquitectura del sistema. Mientras que el Diagrama de Contenedores muestra las aplicaciones, servicios y bases de datos principales, el nivel C4-3 profundiza en la estructura interna del backend, la aplicación móvil y otros elementos clave, desglosándolos en componentes lógicos con responsabilidades claras y bien definidas.

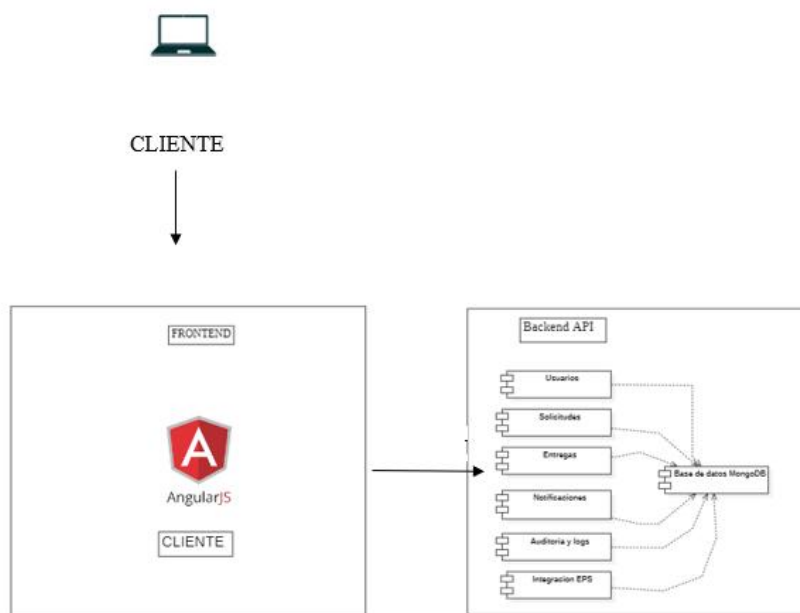
En el caso de MediExpress, el diagrama de componentes resulta fundamental para ilustrar la distribución interna de la lógica de negocio que soporta procesos como la solicitud de medicamentos, el seguimiento de entregas, las notificaciones, la gestión de usuarios y la auditoría. Esta vista permite identificar módulos específicos, como controladores, servicios, repositorios, manejadores de seguridad y componentes de integración y entender cómo colaboran entre sí para ofrecer las funcionalidades del sistema.

Asimismo, el Diagrama de Componentes facilita la comunicación entre desarrolladores, arquitectos y stakeholders técnicos, ya que permite visualizar las dependencias internas, las interfaces expuestas, el flujo de datos y la separación de responsabilidades dentro del monolito modular. Además, contribuye a asegurar que la implementación se mantenga alineada con los principios arquitectónicos definidos, como modularidad, bajo acoplamiento, alta cohesión y reutilización de componentes.

En conjunto, esta vista del Modelo C4 proporciona un nivel de detalle indispensable para garantizar que MediExpress sea una plataforma mantenible, escalable y organizada, capaz de evolucionar con nuevas funcionalidades sin comprometer la estabilidad del sistema.

A continuación, se describen los módulos o componentes principales que conforman la arquitectura interna del sistema MediExpress dentro de cada contenedor (Frontend, Backend y Base de Datos). Este nivel del modelo C4 permite detallar la organización interna de la aplicación y las responsabilidades funcionales de cada componente.

**Figura 37**  
*Diagrama de Componentes*



**Contenedor: Aplicación Móvil (Frontend – Angular).** La aplicación móvil representa la capa de presentación y se compone de los siguientes módulos:

#### **Módulo de Autenticación**

Manejo de inicio de sesión y registro.

Gestión de tokens JWT.

Recuperación de contraseña.

### **Módulo de Solicitudes de Medicamentos**

Crear solicitudes nuevas.

Cargar fórmulas o autorizaciones.

Consultar el estado de solicitudes.

### **Módulo de Perfil del Usuario**

Información básica del usuario.

Actualización de datos personales.

Configuración de notificaciones.

### **Módulo de Entregas**

Visualización de la ruta o estado de entrega.

Actualización en tiempo real mediante notificaciones push.

### **Módulo de Notificaciones**

Recepción de alertas (aprobación, envío, entrega).

Integración con FCM/SMS.

**Contenedor: Backend Monolítico (Node.js- Express).** Este contenedor concentra toda la lógica del negocio mediante componentes modulares y separados:

### **Módulo de Autenticación (Auth Module)**

Autenticación basada en JWT.

Autorización RBAC para personal logístico y usuarios.

Middleware de validación.

**Módulo de Usuarios (Users Module)**

Gestión de usuarios y sus roles.

Actualización y consulta de datos.

Validaciones de integridad.

**Módulo de Solicitudes (Requests Module)**

Registro y administración de solicitudes de medicamentos.

Validación documental.

Gestión de estados del proceso.

**Módulo de Entregas (Deliveries Module)**

Asignación de mensajeros.

Actualización del estado de entrega.

Control de tiempos y evidencias.

**Módulo de Notificaciones (Notifications Module)**

Envío de SMS, emails o notificaciones push.

Integración con servicios externos

**Módulo de Auditoría y Registros (Audit & Logs)**

Trazabilidad de acciones.

Registro de accesos sensibles.

Integración con sistema de monitoreo.

**Contenedor: Base de Datos (MongoDB).** MongoDB almacena datos en colecciones, cada una ligada a un módulo del backend:

**Colección: Users**

Datos personales.

Roles y permisos.

**Colección: Requests**

Peticiones de medicamentos.

Documentos adjuntos.

Estados del flujo.

**Colección: Deliveries**

Rutas asignadas.

Responsable del envío.

Tiempos de entrega y confirmaciones.

**Colección: Notifications**

Historial de alertas enviadas.

Preferencias de usuario.

**Colección: AuditLogs**

Eventos del sistema.

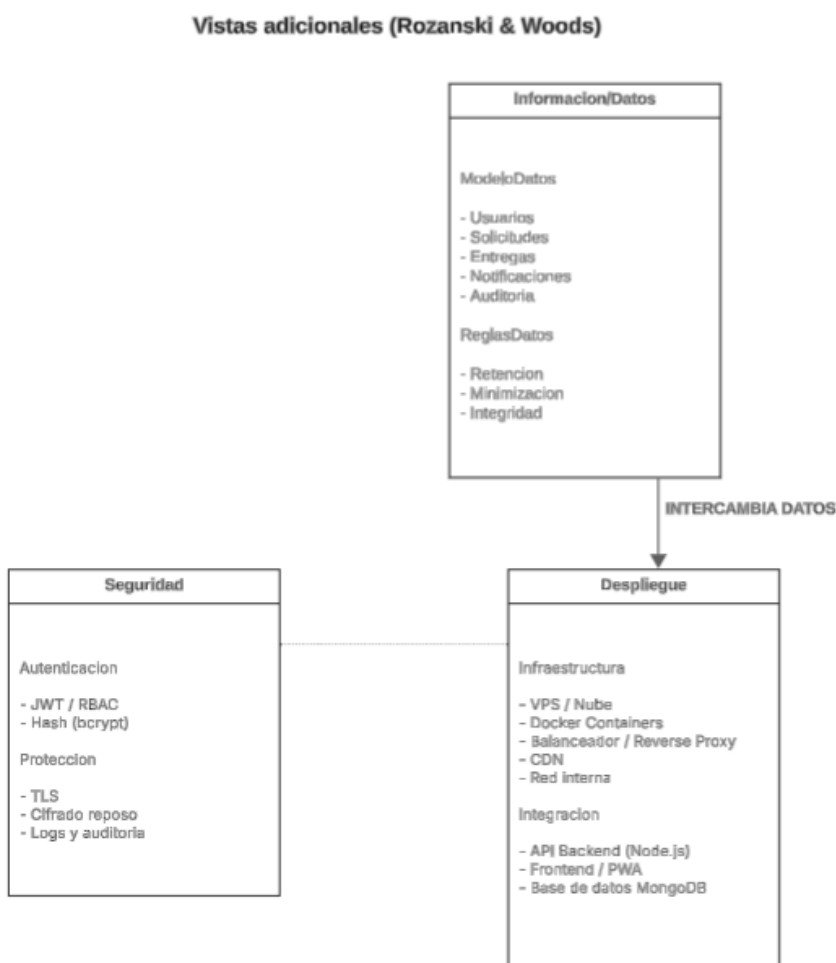
Cambios críticos.

***Vistas adicionales (Rozanski & Woods)***

Las vistas adicionales propuestas por Rozanski y Woods permiten analizar el sistema desde perspectivas específicas que complementan la arquitectura estructural principal. Estas vistas abordan aspectos clave como el manejo de datos, la infraestructura de despliegue y los mecanismos de seguridad, elementos esenciales para garantizar que el sistema sea confiable, seguro y operativo en un entorno real. Para MediExpress, estas vistas son especialmente relevantes debido al manejo de información sensible en salud, la necesidad de disponibilidad continua y la integración con servicios externos.

**Figura 38**

*Diagrama Vistas adicionales (Rozanski & Woods)*



***Vista de Información / Datos***

**Propósito.** Describe el modelo de información que utiliza el sistema, incluyendo las entidades principales, sus atributos, relaciones y el flujo general de datos. Garantiza consistencia, integridad y gobernanza de la información. A continuación, se presentan elementos y/o notas relevantes:

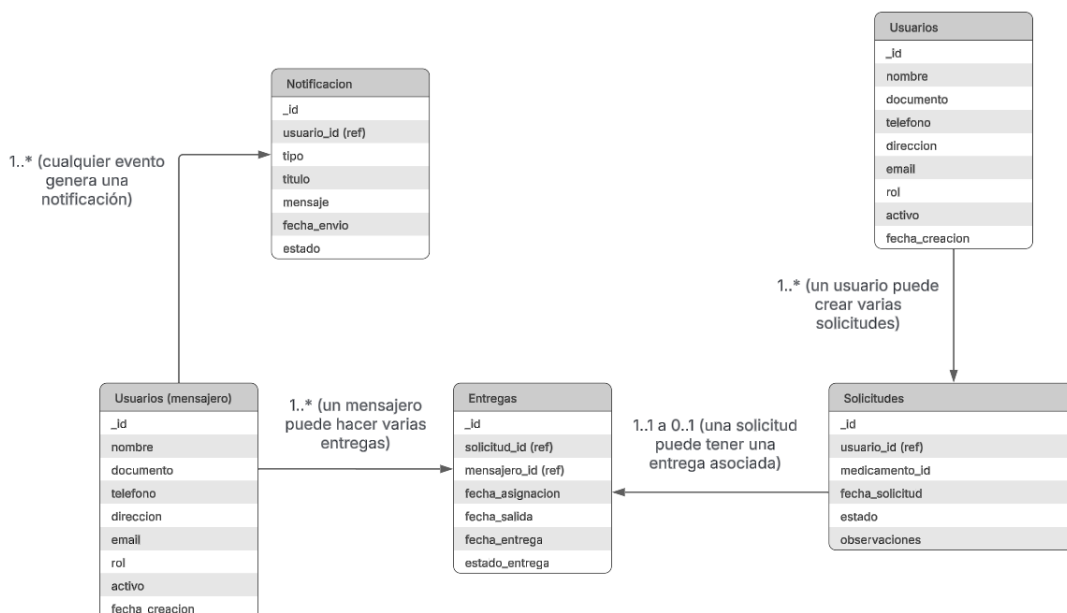
Definición de colecciones y documentos (MongoDB).

Relaciones entre entidades clave: usuarios, solicitudes, entregas, notificaciones.

ERD lógico del dominio.

Políticas de retención y minimización de datos sensibles.

**Figura 39**  
*Vista de Información / Datos*



**Alineación con MediExpress.** Permite estructurar adecuadamente datos críticos como solicitudes de medicamentos, estados de entrega y validaciones EPS, garantizando trazabilidad y consistencia.

### *Vista de Despliegue*

**Propósito.** La Vista de Despliegue describe la infraestructura física y lógica necesaria para ejecutar MediExpress en un entorno real. Define cómo se distribuyen los componentes de software sobre los recursos de hardware y red, garantizando disponibilidad, rendimiento, escalabilidad y seguridad (Rozanski & Woods, 2012). Esta vista es clave para asegurar que la aplicación móvil y sus servicios backend funcionen correctamente ante carga de usuarios, integración con servicios externos y operación continua.

En MediExpress, esta vista articula los elementos que permiten que la plataforma funcione de manera estable en producción, incluyendo servidores, contenedores, red,

almacenamiento y servicios de comunicación. A continuación, se presentan elementos y/o notas relevantes:

VPS o nube como infraestructura base.

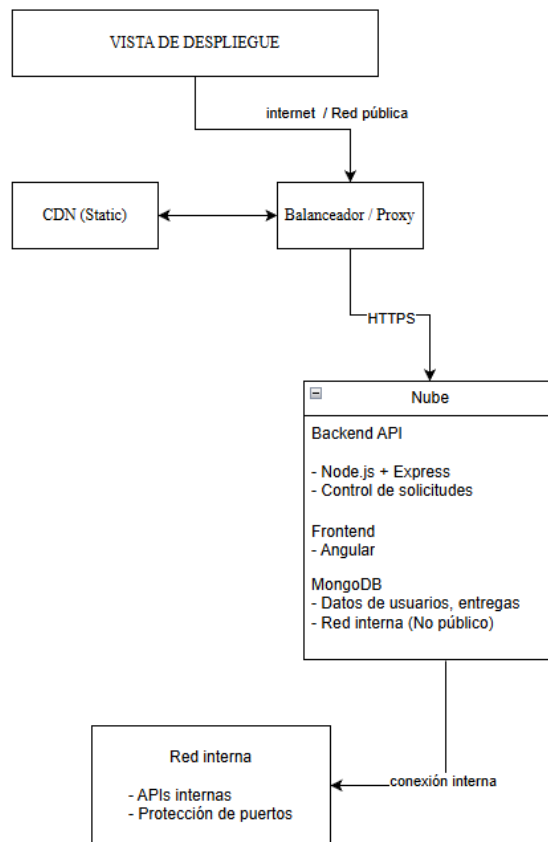
Contenedores Docker para backend y frontend.

Posible orquestación con Kubernetes (futuras fases).

CDN para el contenido estático de la aplicación móvil

Separación de capas: red interna, API, base de datos.

**Figura 40**  
*Vista de Despliegue*



**Alineación con MediExpress.** Asegura un despliegue escalable y confiable, adecuado para operar una aplicación de uso frecuente y con integración a servicios externos.

***Vista de Seguridad***

**Propósito.** Describe los mecanismos de protección del sistema: autenticación, autorización, cifrado, auditoría y cumplimiento normativo. Es clave para salvaguardar datos personales y clínicos. A continuación, se presentan elementos y/o notas relevantes:

Autenticación con JWT y autorización basada en roles (RBAC).

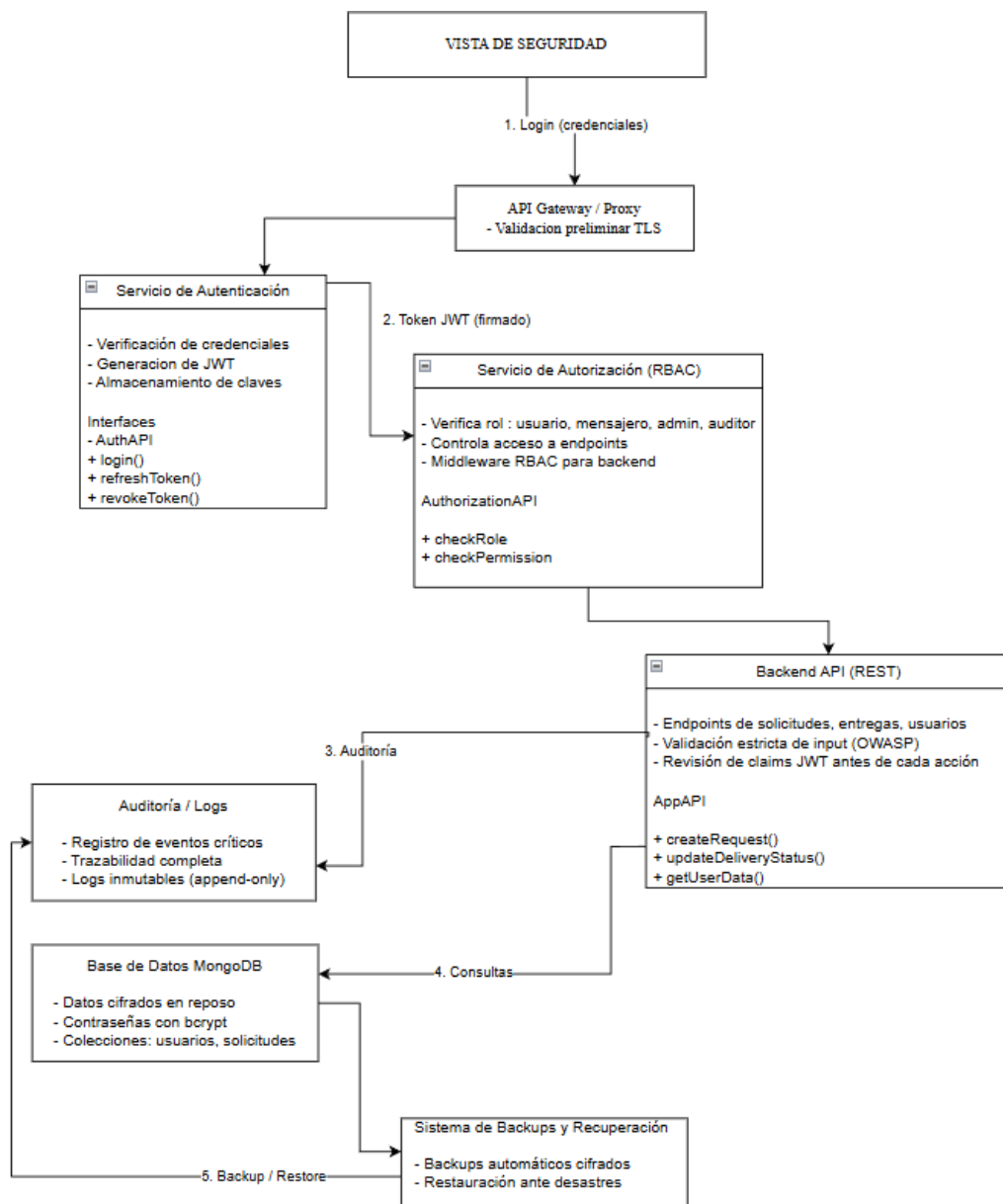
Cifrado TLS para tránsito y hash seguro para contraseñas.

Auditoría de eventos críticos y logs centralizados.

Backups automáticos y recuperación ante desastres.

Cumplimiento con normativas: consentimiento, tratamiento de datos, minimización.

**Figura 41**  
*Vista de Seguridad*



**Alineación con MediExpress.** Protege información sensible de adultos mayores y asegura integridad en procesos como entrega de medicamentos y validación EPS.

**Tabla 18**  
*Vista de seguridad*

<b>Vista</b>	<b>Propósito</b>	<b>Elementos / Notas Relevantes</b>
Información / Datos	Definir el modelo de datos y las relaciones clave.	Tablas, entidades, ERD, metadatos, políticas de datos.
Despliegue	Describir la infraestructura y ambientes.	VPS, Docker, Kubernetes (futuro), CDN, redes internas.
Seguridad	Garantizar protección, autenticación y auditoría.	RBAC, TLS, logs centralizados, copias de seguridad.

### **Arquitectura Lógica**

La arquitectura lógica de MediExpress se organiza en un modelo multicapa que permite separar responsabilidades, mejorar la mantenibilidad del software y facilitar su evolución. Cada capa cumple un rol específico dentro del sistema y agrupa los módulos que implementan su funcionalidad. A continuación, se describen las capas de la solución:

#### ***Capa de Presentación (Frontend )***

Esta capa contiene la interfaz la que interactúan los usuarios finales. Su función principal es ofrecer una experiencia clara, accesible y sencilla para adultos mayores y personas con movilidad reducida. A continuación, se presentan las responsabilidades principales de esta capa:

- Interfaz de usuario móvil/PWA desarrollada en Angular.
- Formularios para solicitudes de medicamentos.
- Visualización del estado de entregas.
- Gestión de autenticación (integración con backend vía JWT).
- Manejo de notificaciones en tiempo real.
- Implementación de accesibilidad (tamaño de letra, contraste, navegación simplificada).

Los siguientes son los módulos/micromódulos en esta capa:

- Módulo de Autenticación
- Módulo de Solicitudes

Módulo de Perfil

Módulo de Entregas

Módulo de Notificaciones

### ***Capa de Aplicación (Servicios / Casos de uso)***

Esta capa actúa como orquestadora de la lógica de negocio, coordinando las operaciones del dominio y gestionando el flujo entre la presentación y las entidades. A continuación, se presentan las responsabilidades principales de esta capa:

Implementación de casos de uso: creación, consulta y actualización de solicitudes.

Validación de reglas básicas antes de delegar al dominio.

Manejo del flujo entre componentes del backend.

Control de autorizaciones de usuario (roles).

Integración con servicios externos (EPS, notificaciones).

Los siguientes son los ejemplos de servicios / casos de uso:

CrearSolicitudService

AsignarEntregaService

NotificarEventoService

VerificarAutorizaciónEPSService

### ***Capa de Dominio (Reglas de negocio)***

Es el núcleo de la aplicación. Contiene las reglas de negocio independientes del framework o de la infraestructura, siguiendo principios DDD (Domain-Driven Design). A continuación, se presentan las responsabilidades principales de esta capa:

Definir entidades, agregados y objetos de valor.

Encapsular la lógica crítica del negocio.

Garantizar coherencia en estados y transiciones.

Definir interfaces de repositorio (sin implementación).

Las siguientes son las entidades principales:

Usuario (datos personales, rol, preferencias)

Solicitud (estado, documentos, fechas)

Entrega (mensajero, ruta, confirmación)

Notificación (tipo, estado, canal)

Los principales beneficios en esta capa son:

Aísla la lógica central del resto del sistema, permitiendo cambios en el backend o la base de datos sin afectar las reglas del negocio.

### ***Capa de Infraestructura (Persistencia, Comunicación, Frameworks)***

Esta capa implementa los detalles técnicos necesarios para ejecutar el sistema.

Proporciona soporte a las demás capas mediante adaptadores concretos. A continuación, se presentan las responsabilidades principales de esta capa:

Implementar repositorios conectados a MongoDB.

Exponer la API REST

Integrar módulos técnicos (JWT, envío de SMS/FCM).

Implementar logging, auditoría y manejo de errores.

Configuración del servidor, contenedores y conexión a servicios externos.

Los principales componentes en esta capa son:

Adaptadores de repositorio (MongoRepository)

Controladores REST

Integración con API de la EPS

Proveedor de notificaciones (SMS / Push)

Módulo de seguridad

**Tabla 19**  
*Capa de Infraestructura*

<b>Capa</b>	<b>Enfoque</b>	<b>Incluye</b>
<b>Presentación</b>	Interfaz y experiencia del usuario	PWA Angular, formularios, vistas, accesibilidad
<b>Aplicación</b>	Casos de uso y orquestación	Servicios, validaciones, integraciones externas
<b>Dominio</b>	Lógica central del negocio	Entidades, reglas, interfaces de repositorio

Capa	Enfoque	Incluye
<b>Infraestructura</b>	Recursos técnicos y persistencia	MongoDB, controladores, repositorios, seguridad

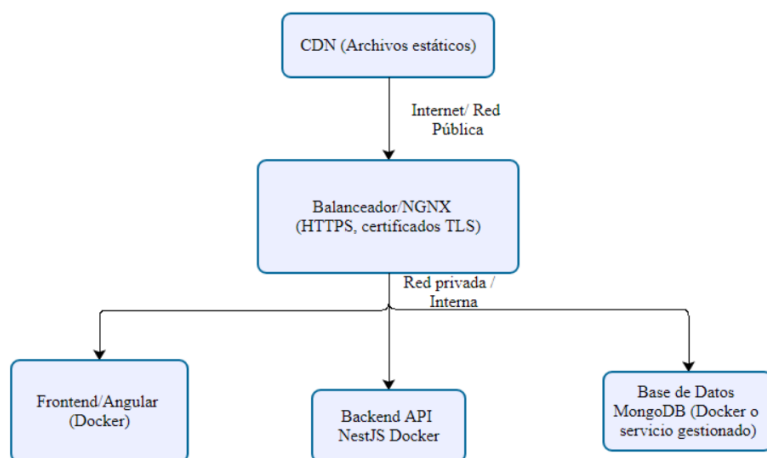
### Arquitectura Física (Infraestructura)

La arquitectura física define la infraestructura tecnológica necesaria para desplegar y operar el sistema MediExpress en un entorno seguro, escalable y confiable. Esta vista describe los componentes físicos, los servicios de red y las plataformas de ejecución involucradas en la operación del sistema.

A continuación, se presenta el diagrama físico sugerido, junto con una descripción detallada de cada elemento:

#### Figura 42

##### Arquitectura Física (Infraestructura)



A continuación, se muestran los componentes de la infraestructura:

#### *Servidor VPS / Nube (Hosting Principal)*

El backend monolítico y la base de datos se despliegan en un servidor VPS o instancia en

la nube. Las siguientes son las responsabilidades del componente:

- Alojar la API REST (Node.js).
- Alojar MongoDB o conectarse a un clúster gestionado.
- Permitir escalado vertical en fases iniciales.
- Proveer recursos dedicados (CPU, RAM y almacenamiento).

### ***Balanceador de Carga (Load Balancer)***

Aunque no es obligatorio en la primera fase, se recomienda para versiones posteriores.

Las siguientes son las funciones del componente:

- Distribuir tráfico entre instancias del backend (cuando existan).
- Añadir HTTPS con certificados TLS
- Proteger contra picos de carga.

### ***CDN (Content Delivery Network)***

El CDN distribuye archivos estáticos del frontend (Angular) e incluye:

- HTML, CSS, JS del frontend.
- Imágenes e íconos de la aplicación móvil.
- Service Worker y recursos PWA.

Los siguientes son los beneficios del componente:

- Reducción de latencia.
- Mayor disponibilidad.
- Mejor experiencia para adultos mayores con conexión limitada.

### ***Contenedores Docker***

Toda la aplicación se despliega sobre contenedores para facilitar portabilidad y consistencia. Los siguientes son componentes dockerizados:

- Backend monolítico
- Frontend (Angular, servido con NGINX).
- Base de datos MongoDB (opcional si se usa servicio externo).

Las siguientes son las ventajas del componente:

Entornos reproducibles.

Aislamiento de dependencias.

Fácil despliegue vía Docker Compose o Kubernetes.

### ***Red Interna (Private Network)***

La comunicación entre componentes se realiza en una red interna aislada.

Usos:

Comunicación backend ↔ base de datos sin exponer puertos públicos.

Acceso del contenedor API al servicio de logs o mensajería.

Seguridad reforzada y menor superficie de ataque.

### **Modelo de Datos**

El modelo de datos está diseñado para soportar el proceso de solicitud, validación, entrega y notificación de medicamentos a adultos mayores y personas con movilidad reducida.

Se emplea un esquema NoSQL (MongoDB), estructurado mediante colecciones relacionadas de forma lógica.

### ***Entidades y Atributos***

**Usuarios.** Representa al usuario final, administrador o mensajero. Tiene los siguientes atributos:

`_id: ObjectId`

`nombre: string`

`documento: string`

`telefono: string`

`direccion: string`

`email: string`

rol: beneficiario | mensajero | admin

activo: boolean

fecha\_creacion: ISODate

**Solicitudes.** Registra la petición de un medicamento por parte del usuario. Tiene los siguientes atributos:

\_id : ObjectId

usuario\_Id : ObjectId (referencia a tabla usuarios)

medicamento\_id: ObjectId (referencia a tabla medicamentos)

fecha\_solicitud: ISODate

estado: string

observaciones: string

**Medicamentos.** Contiene la información del medicamento solicitado. Tiene los siguientes atributos:

\_Id (ObjectId)

Nombre: string

Descripcion: string

Presentacion: string

Codigo: string

stock\_local: number

requiere\_autorizacion: boolean

**Entregas.** Representa el proceso físico de entrega. Tiene los siguientes atributos:

\_id: ObjectId

solicitud\_id : ObjectId (referencia a tabla solicitudes)

Mensajero\_Id: ObjectId (referencia a tabla usuarios)

Fecha\_Asignacion: ISODate

Fecha\_salida: ISODate

fecha\_entrega: ISODate

estado\_entrega: string

**Notificacion.** Contiene los mensajes enviados al usuario. Tiene los siguientes atributos:

\_id (ObjectId)

usuario\_Id : ObjectId (referencia a tabla usuarios)

tipo: string

titulo: string

mensaje: string

fecha\_envio: ISODate

estado: string

**Auditoria.** Registra eventos críticos del sistema. Tiene los siguientes atributos:

\_id : ObjectId

usuario\_Id : ObjectId (referencia a tabla usuarios)

accion: string

fecha\_adicion: ISODate

ip: string

detalle: string

## Relaciones entre Entidades

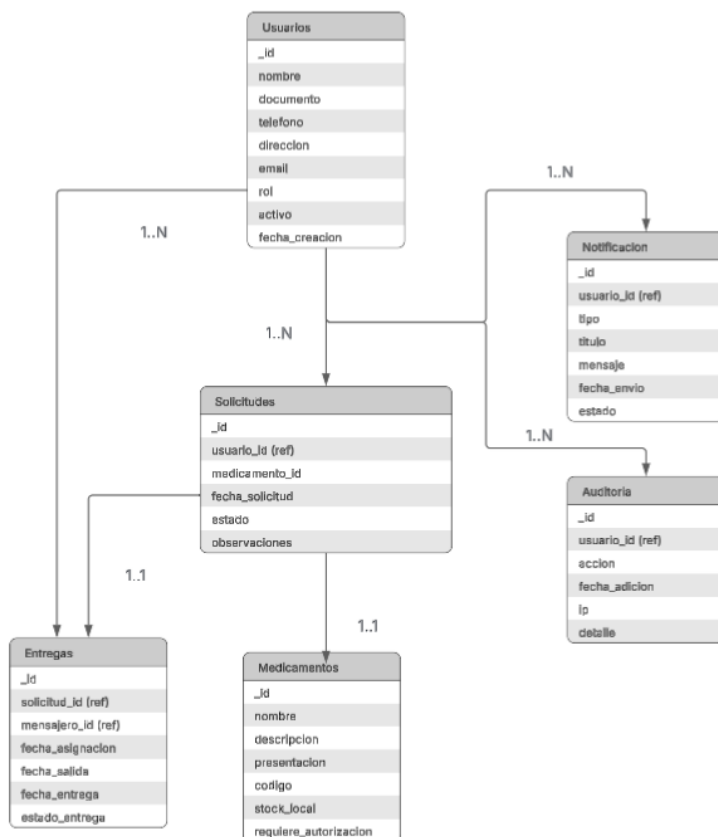
**Tabla 20**

Relaciones entre Entidades

Entidad origen	Entidad destino	Tipo de relación	Implementación
USUARIO	SOLICITUD	1 – N	solicitud.usuario_id
SOLICITUD	ENTREGA	1 – 1/N	entrega.solicitud_id
SOLICITUD	MEDICAMENTOS	1 – N	solicitud.medicamento_id
USUARIO (mensajero)	ENTREGA	1 – N	entrega.mensajero_id
USUARIO	AUDITORIA	1 – N	auditoria.usuario_id
USUARIO	NOTIFICACION	1 – N	auditoria.usuario_id notificacion.usuario_id

**Figura 43**

Diagrama ERD MongoDB



## Integración y APIs

La comunicación entre componentes dentro de una arquitectura moderna depende de la definición clara de servicios y APIs que permitan el intercambio seguro y eficiente de información. Según Microsoft Azure Architecture Center (2023), las APIs REST constituyen el mecanismo estándar para la integración entre sistemas debido a su simplicidad, escalabilidad y compatibilidad con múltiples plataformas.

En el caso de MediExpress, la exposición de APIs facilita la interacción entre el frontend, el backend y los servicios externos, permitiendo gestionar procesos como autenticación, solicitudes de medicamentos, entregas y notificaciones. Además, el uso de mecanismos de seguridad como JWT y OAuth2 garantiza la protección de datos sensibles y el control de acceso basado en roles, en concordancia con las recomendaciones de la Open Web Application Security Project (OWASP, 2024). De esta manera, la arquitectura de MediExpress asegura un flujo confiable, seguro y bien estructurado de información entre todos los actores del sistema.

**Tabla 21**  
*Integración y APIs*

<b>Servicio / API</b>	<b>Propósito</b>	<b>Tipo de interfaz (REST)</b>	<b>Seguridad / Autenticación</b>
Autenticación	Gestión de usuarios, login, roles y sesiones.	REST	JWT / OAuth2
Usuarios	CRUD de usuarios, actualización de datos, perfiles.	REST	JWT / RBAC
Solicitudes de Medicamentos	Registrar solicitudes, consultar estado, adjuntar documentos.	REST	JWT / RBAC
Entregas	Asignación de mensajeros, actualización de estado, confirmación de entrega.	REST	JWT / RBAC

<b>Servicio / API</b>	<b>Propósito</b>	<b>Tipo de interfaz (REST)</b>	<b>Seguridad / Autenticación</b>
Notificaciones	Envío de SMS, correos o push al usuario.	REST	API Key + JWT
Auditoría	Registro de eventos del sistema y trazabilidad.	REST	JWT + control interno

## **Seguridad y Privacidad**

La arquitectura de MediExpress incorpora un conjunto de mecanismos de seguridad y privacidad orientados a garantizar la protección de los datos personales y clínicos de los usuarios, así como la integridad y disponibilidad del sistema. Estos controles se alinean con buenas prácticas internacionales de seguridad en aplicaciones web y móviles.

### ***Mecanismo de autenticación y autorización***

El sistema implementa un modelo de autenticación basado en JSON Web Tokens (JWT) y un esquema de autorización por roles (Role-Based Access Control — RBAC).

Permite diferenciar perfiles como usuario final, mensajero y administrador.

Controla el acceso a recursos sensibles del backend.

Previene accesos no autorizados y reduce la superficie de ataque.

Según Auth0 (2023), el uso de JWT combinado con RBAC asegura un control granular y seguro sobre los permisos de cada tipo de usuario.

### ***Cifrado en tránsito y en reposo***

Para la protección de datos personales y médicos:

TLS 1.2+ cifra toda comunicación entre cliente y servidor, evitando su interceptación.

Las contraseñas se almacenan con hashing seguro, siguiendo recomendaciones

OWASP.

Se aplican políticas de protección del almacenamiento y del acceso a la BD.

OWASP (2024) señala que el cifrado correcto en tránsito y en reposo es fundamental para proteger datos sensibles frente a ataques de intermediarios y accesos indebidos.

### ***Registro y auditoría de eventos***

El sistema incorpora un módulo de auditoría para registrar eventos relevantes, tales como:

Accesos al sistema.

Cambios en estados de solicitudes o entregas.

Acciones realizadas por administradores o mensajeros.

Estos logs permiten:

Identificación de incidentes de seguridad.

Trazabilidad completa del flujo de medicamentos.

Cumplimiento regulatorio y transparencia operativa.

### ***Políticas de copia de seguridad y restauración***

Las copias de seguridad periódicas (backups) garantizan la continuidad del servicio.

Incluyen:

Backups automáticos de la base de datos (diarios o semanales según carga).

Pruebas regulares de restauración.

Almacenamiento seguro y encriptado en la nube.

Esto permite recuperar información ante fallos, pérdida de datos o incidentes cibernéticos.

### ***Cumplimiento con normativas de privacidad***

Debido al manejo de información sensible, MediExpress aplica controles para cumplir

con normativas de protección de datos como la Ley 1581 de 2012 (Colombia) y buenas prácticas internacionales:

Consentimiento informado para tratamiento de datos.

Aviso de privacidad y finalidades del tratamiento.

Minimización de datos: solo se recolecta lo estrictamente necesario.

Limitación de acceso y retención controlada.

Medidas de seguridad organizativas y técnicas.

Google Cloud (2024) destaca que el principio de minimización es esencial para evitar riesgos innecesarios relacionados con datos personales.

## Estrategia de Pruebas

La estrategia de pruebas definida para MediExpress garantiza que la aplicación cumpla con los requisitos funcionales y no funcionales establecidos durante el proceso de análisis, asegurando su calidad, confiabilidad y correcto funcionamiento en escenarios reales. Siguiendo buenas prácticas de ingeniería de software, se incluyen pruebas en distintos niveles, unitarias, integración, end-to-end, rendimiento y seguridad con el fin de cubrir tanto la lógica interna del sistema como la interacción completa del usuario final.

**Tabla 22**  
*Estrategia de Pruebas*

<b>Tipo de Prueba</b>	<b>Herramienta / Framework</b>	<b>Propósito</b>
Unitaria	JUnit	Validar funciones individuales, servicios y lógica aislada.
Integración	Postman	Comprobar la interacción entre módulos, API y base de datos real en contenedores.
End-to-End (E2E)	Selenium	Simular el flujo completo del usuario final en la aplicación.

<b>Tipo de Prueba</b>	<b>Herramienta / Framework</b>	<b>Propósito</b>
Rendimiento / Carga	k6	Medir latencia, capacidad concurrente y estabilidad del sistema bajo carga.
Seguridad	OWASP ZAP	Detectar vulnerabilidades en APIs, autenticación y exposición de datos.
Usabilidad	Pruebas con usuarios	Validar accesibilidad, comprensión y facilidad de uso por adultos mayores.
Regresión	BrowserStack	Garantizar que los nuevos cambios no rompan funcionalidades existentes.

### **Decisiones Arquitectónicas (ADR)**

Las Architecture Decision Records (ADR) son documentos breves que registran decisiones técnicas relevantes tomadas durante el diseño del sistema. Su propósito es preservar el razonamiento detrás de cada elección, facilitar la trazabilidad y permitir que futuros equipos comprendan por qué se adoptó una solución y no otra (Nygard, 2011).

En el contexto de MediExpress, estas decisiones aseguran coherencia arquitectónica, alineación con requisitos funcionales/no funcionales y reducción de riesgos durante la construcción del sistema.

#### **Tabla 23**

*Decisiones Arquitectónicas (ADR)*

<b>ID</b>	<b>Título / Tema</b>	<b>Decisión</b>	<b>Razonamiento / Consecuencia</b>
ADR-001	Framework Backend	Node.js + Express	Framework ligero y flexible que permite modularidad, fácil mantenimiento y rápida construcción del backend monolítico. Favorece simplicidad en la primera versión y amplia disponibilidad de librerías.
ADR-002	Arquitectura General	Arquitectura Monolítica Modular	Reduce complejidad en la Fase 1. Fácil despliegue, menor costo operativo. Se puede migrar a microservicios en fases futuras.
ADR-003	Base de Datos	MongoDB (NoSQL)	Permite manejar estructuras variables (solicitudes, entregas), evita rigidez de esquemas y mejora velocidad de desarrollo.

ID	Título / Tema	Decisión	Razonamiento / Consecuencia
ADR-004	Frontend / Aplicación Móvil	Angular	Ofrece componentes reutilizables, estructura fuerte y soporte para accesibilidad. Permite convertir el frontend en app móvil sin App Store.
ADR-005	Seguridad	Autenticación JWT + RBAC	JWT facilita autenticación en aplicaciones móviles; RBAC controla permisos por rol (usuario, mensajero, admin). Buena práctica OWASP.
ADR-006	Contenerización	Docker para backend, frontend y base de datos	Garantiza entornos consistentes, facilita despliegue y escalabilidad inicial en un VPS.
ADR-007	Notificaciones	Firebase Cloud Messaging (FCM)	Envía alertas de estado sin infraestructura propia. Ideal para adultos mayores: notificaciones simples y directas.
ADR-008	Manejo de Logs y Auditoría	Logging Centralizado	Permite rastrear solicitudes, entregas y actividad del sistema. Mejora trazabilidad y transparencia.
ADR-009	Despliegue	VPS + NGINX como reverse proxy	Económico, estable y adecuado para un monolito. Permite HTTPS y mejora rendimiento del frontend.

## Estrategia de Despliegue

**Tabla 24**

*Estrategia de Despliegue*

Fase	Entorno / Infraestructura	Tecnologías / Herramientas	Objetivo
Fase 1 – Desarrollo	Entorno local de los desarrolladores	Node.js, Express, Angular, Docker Desktop, Git	Permitir desarrollo ágil, pruebas unitarias e integración básica entre módulos.

## Observabilidad y Mantenimiento

La observabilidad y el mantenimiento garantizan la operación continua, confiable y segura de MediExpress una vez desplegado en producción. Estos mecanismos permiten detectar fallos tempranos, monitorear el estado del sistema y asegurar la recuperación ante incidentes, especialmente relevantes debido al carácter sensible de la información y la criticidad del proceso de entrega de medicamentos.

### ***Métricas (latencia, errores, disponibilidad)***

El sistema debe recopilar métricas clave que permitan evaluar su rendimiento y comportamiento en tiempo real, tales como:

- Latencia en las peticiones a la API.

- Tasa de errores (4xx, 5xx).

- Disponibilidad del servicio (uptime).

- Consumo de CPU, memoria y disco.

Estas métricas permiten identificar cuellos de botella, prevenir degradaciones en el servicio y tomar decisiones de optimización basadas en datos.

### ***Logs centralizados y trazabilidad de solicitudes***

Para garantizar seguimiento completo del ciclo de vida de cada solicitud:

- Se implementa log centralizado, consolidando los registros en un único repositorio.

- Cada petición se marca con un ID de correlación, permitiendo rastrear todo el flujo desde la creación hasta la entrega del medicamento.

- Los logs incluyen información como:

  - Accesos

  - Cambios en estados de solicitud

  - Actividad de mensajeros

  - Errores inesperados

Esto mejora auditoría, diagnóstico y cumplimiento normativo.

### ***Alertas automáticas***

El sistema debe contar con alertas basadas en reglas y umbrales predefinidos, tales como:

- Aumento de errores 500

Caídas en disponibilidad

Latencia superior al nivel aceptado

Fallas en la conexión con la base de datos

Interrupción en la integración con la EPS

Estas alertas se envían por correo, SMS o aplicaciones de monitoreo, permitiendo reacción inmediata del equipo técnico.

### ***Políticas de actualización, backup y restauración***

Para asegurar continuidad y resiliencia:

Se programan actualizaciones controladas del software con despliegues sin interrupciones.

Se implementan copias de seguridad periódicas de la base de datos (diarias/semanales).

Los backups se almacenan en repositorios encriptados.

Se realizan pruebas regulares de restauración para verificar la integridad de las copias.

Se contempla un plan básico de recuperación ante desastres.

Estas prácticas garantizan que MediExpress pueda recuperarse rápidamente ante fallos, pérdida de datos o incidentes de seguridad.

## **Desarrollo de la aplicación móvil para la entrega de medicamentos desde la metodología Scrum**

Esta sección describe cómo se desarrolló el prototipo funcional de MediExpress aplicando la metodología ágil Scrum, adaptada al proceso de creación de una aplicación móvil orientada a la gestión y entrega de medicamentos. A diferencia de los enfoques tradicionales, en Scrum no se construye un prototipo aislado o desechable; por el contrario, cada iteración produce un incremento real del producto que es funcional, validable y potencialmente desplegable desde el primer Sprint.

En este marco, el desarrollo del sistema se estructuró en ciclos cortos denominados Sprints, dentro de los cuales el equipo planificó, construyó, revisó y mejoró funcionalidades basadas en las necesidades del usuario. Cada incremento incorporó características completas y utilizables, permitiendo evaluar de forma continua la experiencia de los adultos mayores y personas con movilidad reducida, quienes representan el público objetivo del proyecto. Asimismo, Scrum enfatizó la colaboración constante, la inspección continua y la adaptación del producto. Esto permitió que la aplicación móvil evolucione de acuerdo con los requisitos funcionales y no funcionales definidos, incorporando mejoras relacionadas con accesibilidad, fluidez del proceso de solicitud, validación de datos y seguimiento de entregas.

En conjunto, esta metodología garantizó que el desarrollo del prototipo de MediExpress no solo avanzara de manera ordenada y transparente, sino que también mantuviera un enfoque centrado en el usuario, asegurando que cada versión incremental mejore la usabilidad, confiabilidad y efectividad de la solución tecnológica propuesta.

## **Tipo de Prototipo**

En el marco de la metodología Scrum, el desarrollo del producto se orienta desde el inicio hacia la construcción de un prototipo funcional de alta fidelidad, el cual evoluciona de manera incremental a lo largo de los Sprints. A diferencia de los enfoques tradicionales donde los prototipos suelen ser artefactos preliminares o desechables, en Scrum cada incremento constituye una versión real, operativa y utilizable del sistema.

Por ello, cada Sprint debe producir un incremento del producto potencialmente desplegable, que pueda ser mostrado al Product Owner, probado por los usuarios y evaluado en términos de funcionalidad, usabilidad y desempeño. Este enfoque permite validar de forma temprana los requisitos, reducir riesgos y garantizar que la aplicación móvil en este caso, MediExpress, evolucione de acuerdo con las necesidades reales del público objetivo, especialmente adultos mayores y personas con movilidad reducida.

En resumen, el prototipo en Scrum no se concibe como un modelo parcial o estático, sino como una versión funcional en constante mejora, que avanza gradualmente hacia el producto final mediante entregas iterativas, incrementales y con valor tangible para los usuarios.

## **Product Backlog**

El Product Backlog representa el corazón de la planificación en Scrum. Es un listado priorizado y dinámico que contiene todas las funcionalidades, mejoras, requisitos técnicos y correcciones pendientes del proyecto. Cada elemento del backlog, estructurado como Historia de Usuario, describe una necesidad del usuario final o del negocio y sirve como guía para el desarrollo incremental del producto.

Dentro de MediExpress, el Product Backlog se organiza de manera que refleje las

funcionalidades clave asociadas a la solicitud, entrega y seguimiento de medicamentos, manteniendo siempre como eje principal la accesibilidad y facilidad de uso para adultos mayores y personas con movilidad reducida.

Cada ítem del Product Backlog debe contener información suficiente para su comprensión y planificación, incluyendo identificación, descripción, criterios de aceptación, estimación y prioridad. Estos elementos permiten al equipo evaluar el esfuerzo necesario y garantizar que cada incremento entregue valor real en cada Sprint. A continuación, se muestra el backlog previsto para el proyecto MediExpress:

### *Módulo de Autenticación y Usuarios*

**Tabla 25**

*Módulo de Autenticación y Usuarios*

<b>ID</b>	<b>Historia de Usuario</b>	<b>Descripción</b>	<b>Estimación</b>	<b>Prioridad</b>	<b>Criterios de Aceptación</b>	<b>Sprint</b>	<b>Estado</b>
HU-001	Como usuario, quiero registrarme con mis datos básicos	Crear cuenta con validaciones y captura mínima	5	Alta	Registra usuario, evita duplicados, envía confirmación	S1	Pendiente
HU-002	Como usuario, quiero iniciar sesión de forma segura	Login con JWT	3	Alta	JWT válido, error si credenciales no coinciden	S1	Pendiente
HU-003	Como usuario, quiero recuperar mi contraseña	Flujo de recuperación vía correo o SMS	3	Media	Enlace de recuperación funcional	S2	Pendiente
HU-004	Como usuario, quiero editar mi perfil	Permitir actualizar teléfono, dirección y datos personales	5	Media	Cambios persistidos correctamente	S2	Pendiente

**Módulo de Solicitudes****Tabla 26***Módulo de Solicitudes*

ID	Historia de Usuario	Descripción	Est.	Pri.	Criterios de Aceptación	Sprint	Estado
HU-005	Como usuario, quiero solicitar un medicamento	Formulario de solicitud	8	Alta	Guarda solicitud, valida campos	S2	Pendiente
HU-006	Como usuario, quiero adjuntar foto de mi fórmula médica	Adjuntar imágenes	5	Alta	Sube foto, tamaño controlado, vista previa	S2	Pendiente
HU-007	Como usuario, quiero ver el estado de mis solicitudes	Lista y detalle	5	Alta	Muestra estados: pendiente, validada, entregada	S3	Pendiente
HU-008	Como usuario, quiero cancelar una solicitud	Cancelación autorizada	3	Baja	Cancela solo si está en estado “pendiente”	S3	Pendiente
HU-009	Como usuario, quiero recibir confirmación cuando la EPS valide mi autorización	Integración EPS	8	Alta	Estado cambia automáticamente	S4	Pendiente

**Módulo de Medicamentos****Tabla 27***Módulo de Medicamentos*

ID	Historia de Usuario	Descripción	Est.	Pri.	Criterios de Aceptación	Sprint	Estado
HU-010	Como administrador, quiero gestionar el catálogo de medicamentos	CRUD medicamentos	5	Media	Crear, editar, ver y eliminar	S4	Pendiente
HU-011	Como usuario, quiero ver la descripción de un medicamento	Mostrar ficha del medicamento	3	Baja	Nombre, dosis, descripción y advertencias	S4	Pendiente

**Módulo de Entregas****Tabla 28***Módulo de Entregas*

<b>ID</b>	<b>Historia de Usuario</b>	<b>Descripción</b>	<b>Est.</b>	<b>Pri.</b>	<b>Criterios de Aceptación</b>
HU-012	Como administrador, quiero asignar un mensajero a una solicitud	Gestión de entregas	5	Alta	Entrega queda ligada a mensajero
HU-013	Como mensajero, quiero ver mis entregas asignadas	Lista de entregas por usuario	5	Alta	Solo entregas propias, filtro por estado
HU-014	Como mensajero, quiero actualizar el estado de una entrega	Cambiar estado: en ruta, entregada	5	Alta	Cambios registrados con hora
HU-015	Como mensajero, quiero registrar la firma digital del usuario	Confirmación de entrega	8	Alta	Captura firma o foto de evidencia
HU-016	Como mensajero, quiero ver la ubicación y dirección del usuario	Mapa y ruta	8	Media	Google Maps o similar

**Módulo de Notificaciones****Tabla 29***Módulo de Notificaciones*

<b>ID</b>	<b>Historia de Usuario</b>	<b>Est.</b>	<b>Pri.</b>
HU-017	Como usuario, quiero recibir notificaciones cuando mi solicitud cambie de estado	5	Alta
HU-018	Como administrador, quiero enviar alertas masivas en caso de fallos del sistema	5	Media
HU-019	Como usuario, quiero recibir recordatorios de toma de medicamentos	3	Baja

## Desarrollo Iterativo del Proyecto (Sprints)

El desarrollo del proyecto MediExpress se llevó a cabo mediante un enfoque iterativo e incremental basado en la metodología ágil Scrum. Bajo este marco, el avance del producto se organiza en Sprints, ciclos de trabajo de duración fija (normalmente de 1 a 3 semanas) en los cuales el equipo entrega incrementos funcionales del sistema.

Cada Sprint permitió construir, probar y validar partes del producto de forma progresiva, garantizando una evolución continua, retroalimentación constante y reducción de riesgos técnicos o de requerimientos. En cada iteración se realizó planificación, desarrollo, pruebas y revisión del incremento obtenido, lo cual permitió que la aplicación móvil avanzara siempre con funcionalidades verificables y potencialmente desplegadas.

### *Sprint Planning*

**Tabla 30**

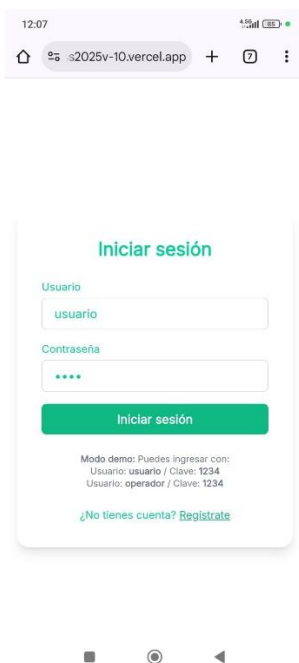
*Sprint 1- Autenticación y usuarios*

<b>Sprint #</b>	<b>Duración</b>	<b>Sprint Goal (Meta del Sprint)</b>	<b>Historias seleccionadas</b>	<b>Tareas técnicas derivadas</b>	<b>Estimación total</b>
Sprint 1	18/08/2025 – 05/09/2025	Habilitar el acceso seguro al sistema mediante registro, inicio de sesión, recuperación de claves y edición básica de perfil.	HU-001, HU-002, HU-003, HU-004	<ul style="list-style-type: none"> <li>• Crear endpoints de registro</li> <li>• Validación de duplicados</li> <li>• Generación JWT</li> <li>• Endpoint de recuperación de contraseña</li> <li>• CRUD perfil</li> <li>• Diseño de interfaz de login/registro</li> <li>• Pruebas de seguridad</li> </ul>	16 puntos

## Sprint Execution

### Figura 44

#### *Login usuario*



### Figura 45

#### *Dashboard Usuario*



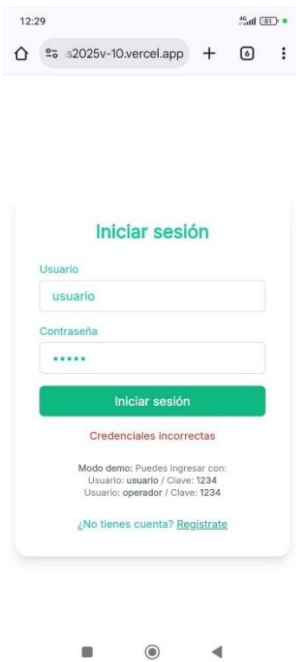
**Figura 46**  
*Login operador*



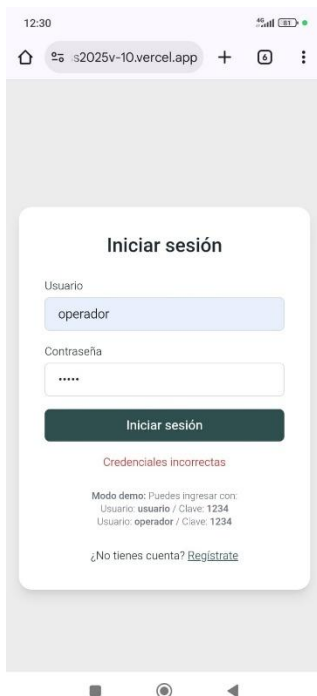
**Figura 47**  
*Dashboard Operador*



**Figura 48**  
*Login Incorrecto usuario*



**Figura 49**  
*Login Incorrecto operador*



**Tabla 31**  
*Sprint 1 - Sprint Review*

<b>Sprint #</b>	<b>Historias completadas</b>	<b>Demostración</b>	<b>Feedback recibido</b>	<b>Decisiones</b>
Sprint 1	HU-001, HU-002, HU-003, HU-004	Durante la revisión se presentó el funcionamiento completo del módulo de autenticación. Se mostró el registro de usuarios, validación de duplicados, inicio y cierre de sesión con autenticación segura. También se enseñó la edición del perfil y el flujo de recuperación de contraseña, evidenciando la necesidad de corrección en este último.	Solicitan agregar validación visual de campos vacíos	Se ajustará UI en Sprint 2 para mejorar mensajes de error

## **Retrospectiva**

### **¿Qué funciono bien?**

Login operativo, registro exitoso de usuarios, validación de duplicados y edición estable del perfil.

### **¿Qué se puede mejorar?**

Evitar cambios de alcance durante el sprint, especialmente en requisitos de validación.

### ¿Qué bloqueos se presentaron

Fallas temporales en el módulo de recuperación de contraseña impidieron su funcionamiento completo.

**Tabla 32**  
*Sprint 2-Solicitudes*

<b>Sprint #</b>	<b>Duración</b>	<b>Sprint Goal (Meta del Sprint)</b>	<b>Historias seleccionadas</b>	<b>Tareas técnicas derivadas</b>	<b>Estimación total</b>
Sprint 2	06/09/2025 – 20/09/2025	Implementar módulo de solicitudes	HU-005, HU- 006, HU-007, HU-008, HU- 009	<ul style="list-style-type: none"> <li>• Formularios solicitudes</li> <li>• Gestión de archivos (fórmula)</li> <li>• Lista de solicitudes</li> <li>• Flujo de cancelación</li> <li>• Integración con EPS</li> <li>•Notificaciones básicas</li> </ul>	29 puntos

### Sprint Execution

**Figura 50**  
*Solicitud medicamento*

12:11 s2025v-10.vercel.app

### Nueva Solicitud de Medicamento

Medicamento \*  
Amoxicilina 500mg

Cantidad (unidades) \*  
1

Frecuencia de consumo \*  
Diaria

Duración del tratamiento (días) \*  
30

Observaciones (opcional)  
Primera dosis

Enviar Solicitud Limpiar

**Figura 51**  
*Solicitud enviada*

12:11 s2025v-10.vercel.app

### Nueva Solicitud de Medicamento

Medicamento \*  
Amoxicilina 500mg

Cantidad (unidades) \*  
1

Frecuencia de consumo \*  
Diaria

Duración del tratamiento (días) \*  
30

Observaciones (opcional)  
Primera dosis

Solicitud enviada exitosamente

**Tabla 33***Sprint 2 - Sprint Review*

<b>Sprint #</b>	<b>Historias completadas</b>	<b>Demostración</b>	<b>Feedback recibido</b>	<b>Decisiones</b>
Sprint 2	HU-005, HU-006, HU-007, HU-008, HU-009	Se demostró la creación de solicitudes mediante formulario validado, la carga de la fórmula médica con vista previa y control de tamaño, la visualización del listado de solicitudes y el cambio automático de estados tras la validación de la EPS. También se mostró el proceso de cancelación para solicitudes pendientes.	Se solicita mejorar filtros y visibilidad del estado de solicitudes; ajustar peso permitido en imágenes.	Optimizar carga de imágenes y añadir filtros adicionales en Sprint 3.

**Retrospectiva****¿Qué funcionó bien?**

Creación exitosa de solicitudes, carga de imágenes estable, listado de estados funcional y flujo de cancelación correcto.

**¿Qué se puede mejorar?**

Controlar cambios de alcance relacionados con integración EPS y validaciones adicionales.

**¿Qué bloqueos se presentaron?**

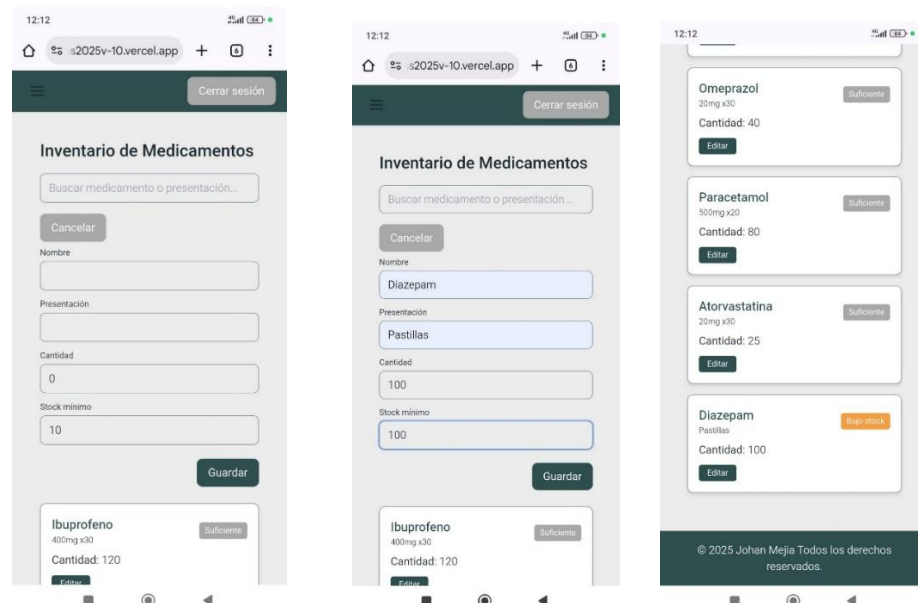
Intermitencia en la validación de autorizaciones con la EPS y errores al cargar imágenes grandes.

**Tabla 34**  
*Sprint 3-Medicamentos*

Sprint #	Duración	Sprint Goal (Meta del Sprint)	Historias seleccionadas	Tareas técnicas derivadas	Estimación total
Sprint 3	21/09/2025 – 05/10/2025	Catálogo de medicamentos funcional	HU-010, HU-011	<ul style="list-style-type: none"> <li>• CRUD medicamentos</li> <li>• Validación de datos</li> <li>• Diseño de fichas</li> <li>• Optimización de consultas DB</li> </ul>	8 puntos

## Sprint Execution

**Figura 52**  
*Ingreso medicamentos*



**Tabla 35**  
*Sprint 3 - Sprint Review*

Sprint #	Historias completadas	Demostración	Feedback recibido	Decisiones
Sprint 3	HU-010, HU-011	Se presentó el módulo de medicamentos, mostrando el CRUD completo (crear,	Solicitan la incorporación de un	Incluir funcionalidad de búsqueda

---

editar, eliminar, consultar) y la visualización de fichas detalladas con nombre, dosis, descripción y advertencias. Se mostró también la validación para evitar duplicados.	buscador por nombre o palabra clave.	en Sprint 4.
---	--------------------------------------	--------------

---

## Retrospectiva

### ¿Qué funcionó bien?

CRUD del catálogo funcionando sin errores, visualización clara de fichas de medicamento.

### ¿Qué se puede mejorar?

Reducir cambios de alcance relacionados con el diseño de interfaz y campos adicionales solicitados.

### ¿Qué bloqueos se presentaron?

Demoras en la carga inicial del catálogo por problemas de conexión con la base de datos.

## Tabla 36

### *Sprint 4- Entregas*

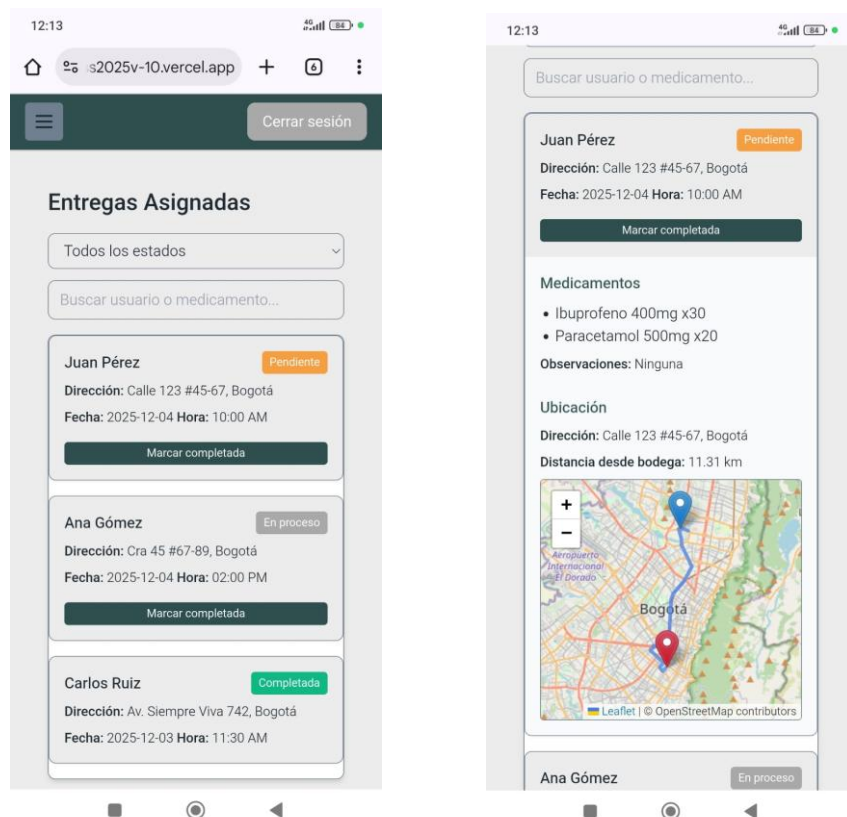
<b>Sprint #</b>	<b>Duración</b>	<b>Sprint Goal (Meta del Sprint)</b>	<b>Historias seleccionadas</b>	<b>Tareas técnicas derivadas</b>	<b>Estimación total</b>
Sprint 4	06/10/2025 – 20/10/2025	Sistema de entregas operable	HU-012, HU-013, HU-014, HU-015, HU-016	<ul style="list-style-type: none"> <li>•Asignación backend</li> <li>• Panel del mensajero</li> <li>• Estados de entrega</li> <li>• Firma digital</li> </ul>	31 puntos

---

- Mapa de ruta

## Sprint Execution

**Figura 53**  
*Asignación entregas*



**Tabla 37**  
*Sprint 4-Sprint Review*

Sprint #	Historias completadas	Demostración	Feedback recibido	Decisiones
Sprint 4	HU-012, HU-013, HU-014, HU-015, HU-016	Se mostró el flujo completo de entregas: asignación del mensajero, panel con entregas asignadas, actualización de estados (“en ruta”, “entregada”) con registro horario, captura de firma digital y evidencia	Solicitan mejorar el tamaño del mapa y contraste visual para mensajeros.	Optimizar interfaz del mapa en Sprint 5.

---

fotográfica, además de la visualización del mapa con la ubicación del usuario.

---

## Retrospectiva

### ¿Qué funcionó bien?

Asignación correcta de mensajeros, panel estable para repartidores, actualización de estados fluida.

### ¿Qué se puede mejorar?

Evitar modificaciones tardías en requisitos del módulo de mapas y firma digital.

### ¿Qué bloqueos se presentaron?

Fallas de GPS y dificultades en la integración con el componente de mapa.

**Tabla 38**

*Sprint 5- Notificaciones*

<b>Sprint #</b>	<b>Duración</b>	<b>Sprint Goal (Meta del Sprint)</b>	<b>Historias seleccionadas</b>	<b>Tareas técnicas derivadas</b>	<b>Estimación total</b>
Sprint 5	21/10/2025 – 02/11/2025	Notificaciones funcionales	HU-017, HU-018, HU-019	<ul style="list-style-type: none"> <li>• Configuración del servicio push</li> <li>• Envío masivo</li> <li>• Recordatorios programados</li> </ul>	13 puntos

## Sprint Execution

**Figura 54**  
*Panel Notificaciones*



**Tabla 39**  
*Sprint 5 - Sprint Review*

<b>Sprint #</b>	<b>Historias completadas</b>	<b>Demostración</b>	<b>Feedback recibido</b>	<b>Decisiones</b>
Sprint 5	HU-017, HU-018, HU-019	Se demostró el sistema de notificaciones automáticas y manuales. Se mostraron alertas por cambio de estado, validación EPS, entrega completada, envío masivo desde el administrador y recordatorios programados de medicamentos.	Comentarios Solicitan permitir elegir horarios personalizados para recordatorios.	Añadir configuración horaria en Sprint 6.

## Retrospectiva

### ¿Qué funcionó bien?

Notificaciones automáticas y manuales funcionando correctamente, recordatorios entregados según programación.

### ¿Qué se puede mejorar?

Mejorar la definición previa del contenido de mensajes para evitar cambios de alcance.

### ¿Qué bloqueos se presentaron?

Problemas en la configuración del servicio push que demoraron el envío inicial.

## Tabla 40

*Sprint 6 - Auditoria y seguridad*

<b>Sprint #</b>	<b>Duración</b>	<b>Sprint Goal (Meta del Sprint)</b>	<b>Historias seleccionadas</b>	<b>Tareas técnicas derivadas</b>	<b>Estimación total</b>
Sprint 6	03/11/2025 – 18/11/2025	Implementar auditoría y seguridad integral	HU-020, HU-021, HU-022, HU-023	<ul style="list-style-type: none"> <li>• Log de acciones</li> <li>• Registro de sesiones</li> <li>• Bloqueo inteligente</li> <li>• Cifrado y privacidad</li> </ul>	20 puntos

**Tabla 41**  
*Sprint 6-Sprint Review*

<b>Sprint #</b>	<b>Historias completadas</b>	<b>Demostración</b>	<b>Feedback recibido</b>	<b>Decisiones</b>
Sprint 6	HU-020, HU-021, HU-022, HU-023	Se presentó el registro de auditoría del sistema, mostrando eventos de actividad. Se revisaron los logs de autenticación (usuario, fecha, IP, resultado), el sistema de bloqueo automático por intentos fallidos y la aplicación de mecanismos de cifrado y privacidad. Se demostró también el cumplimiento básico de Ley 1581 y lineamientos OWASP.	Usuarios sugieren reforzar los mecanismos de respaldo periódico	Implementar backup automatizado tras Sprint 6.

## **Retrospectiva**

### **¿Qué funcionó bien?**

Registro de actividad del sistema estable, log de autenticaciones funcionando y bloqueo automático activado.

### **¿Qué se puede mejorar?**

Reducir cambios de alcance en normas de privacidad y parámetros de cifrado solicitados después del inicio.

### **¿Qué bloqueos se presentaron?**

Inconsistencias en el almacenamiento de registros de auditoría y fallas en pruebas OWASP iniciales.

Link de acceso a repositorio de código fuente:

**<https://github.com/baxterMejia/MediExpress2025v.10>**

## Evaluación de la funcionalidad y usabilidad del producto mínimo viable

**Figura 55**

*Prueba de Funcionalidad - Autenticación*

<b>Nombre del proyecto: MediExpress</b>						
<b>ID del caso de prueba: UAT-001</b>			<b>Escenario de prueba: Ingreso al sistema con credenciales válidas</b>			
<b>Funcionalidad Evaluada: Login</b>			<b>Sprint de Origen: Sprint 1</b>			
<b>Prueba diseñada por: Equipo UAT MediExpress</b>			<b>Prioridad de prueba (Bajo, medio, alto): Alta</b>			
<b>Fecha de ejecución de la prueba: 04/09/2025</b>			<b>Lugar de ejecución de la prueba: Entorno de pruebas UAT MediExpress</b>			
<b>Prueba ejecutada por: Usuario final</b>			<b>Tipo de usuario: Paciente</b>			
<b>Descripción:</b> Verificar que el usuario puede ingresar al sistema con credenciales correctas.						
<b>Condiciones previas:</b> Usuario registrado con usuario y contraseña válidos. <b>Dependencias:</b> Base de datos de usuarios activa.						
<b>Prueba Funcional Ejecutada por el Usuario</b>						
<b>Pasos</b>	<b>Pasos de prueba</b>	<b>Datos de prueba</b>	<b>Resultado esperado</b>	<b>Resultado actual</b>	<b>Estado (falla o pasa)</b>	<b>Nota</b>
Paso 1	Ingresar usuario	usuario	Campo acepta formato válido	OK	Pasa	
Paso 2	Ingresar contraseña	1234	Campo acepta contraseña válida	OK	Pasa	
Paso 3	Seleccionar "Iniciar sesión"	-	Usuario accede al sistema	OK	Pasa	
Paso 4	Ver dashboard	-	Se muestra pantalla inicial	OK	Pasa	
<b>Evaluación del Usuario (Percepción Real)</b>						
<b>Criterio</b>		<b>Calificación (1-5)</b>	<b>Observación</b>			
Facilidad de uso		5	Muy intuitivo			
Claridad de la información		4	Podría resaltar más errores			
Velocidad del sistema		5	Carga rápida			
Utilidad de la función		5	Es indispensable			

Velocidad del sistema	5	Carga rápida	
Utilidad de la función	5	Es indispensable	
Confianza al usarla	5	Sistema seguro	
<b>Incidencias Detectadas por el Usuario</b>			
No	Descripción del Problema	Requiere ajustes (si/no)	Impacto
1	Validación visual débil en campos vacíos	Si	Bajo
<b>Nivel de Aceptación del Usuario</b>			
No	Pregunta	Respuesta (si/no)	
1	¿La funcionalidad resuelve su necesidad?	Si	
2	¿Usaría esta herramienta en su vida real?	Si	
3	¿La recomendaría a otros?	Si	
<b>Conclusión de la Validación con Usuario</b>			
No	Ítem	Resultado (si/no)	
1	Funcionalidad es aceptada	Si	
2	Requiere mejoras	Si	
3	Requiere rediseño	Si	

### Firmas de Validación

\_\_\_\_\_  
Firma del Usuario  
Nombre de Usuario

\_\_\_\_\_  
Firma del Evaluador  
Nombre Evaluador

**Figura 56**  
*Prueba de Funcionalidad – Solicitudes*

<b>Nombre del proyecto: MediExpress</b>						
<b>ID del caso de prueba: UAT-002</b>				<b>Escenario de prueba: Registrar solicitud de medicamento</b>		
<b>Funcionalidad Evaluada: Solicitudes de entrega</b>				<b>Sprint de Origen: Sprint 2</b>		
<b>Prueba diseñada por: Equipo UAT MediExpress</b>				<b>Prioridad de prueba (Bajo, medio, alto): Alta</b>		
<b>Fecha de ejecución de la prueba: 18/09/2025</b>				<b>Lugar de ejecución de la prueba: Entorno de pruebas UAT MediExpress</b>		
<b>Prueba ejecutada por: Usuario final</b>				<b>Tipo de usuario: Paciente</b>		
<b>Descripción: Verificar que el usuario puede registrar solicitud de entrega de medicamento correctamente.</b>						
<b>Condiciones previas: Usuario registrado con usuario y contraseña válidos.</b>						
<b>Dependencias: Base de datos de usuarios activa.</b>						
<b>Prueba Funcional Ejecutada por el Usuario</b>						
<b>Pasos</b>	<b>Pasos de prueba</b>	<b>Datos de prueba</b>	<b>Resultado esperado</b>	<b>Resultado actual</b>	<b>Estado (falla o pasa)</b>	<b>Nota</b>
Paso 1	Abrir formulario	-	Formulario visible	OK	Pasa	
Paso 2	Completar datos	Nombre del medicamento, cantidad, frecuencia de consumo, duración del tratamiento	Datos válidos	OK	Pasa	
Paso 3	Enviar solicitud	-	Solicitud registrada	OK	Pasa	
<b>Evaluación del Usuario (Percepción Real)</b>						
<b>Criterio</b>		<b>Calificación (1–5)</b>	<b>Observación</b>			
Facilidad de uso		5	Muy intuitivo			
Claridad de la información		5	Formularios claros			
Velocidad del sistema		4	Mejorable			
Utilidad de la función		5	Muy útil			
Confianza al usarla		5	Interfaz confiable			

<b>Incidencias Detectadas por el Usuario</b>			
<b>No</b>	<b>Descripción del Problema</b>	<b>Requiere ajustes (si/no)</b>	<b>Impacto</b>
1	Validación velocidad de proceso	Si	Medio
<b>Nivel de Aceptación del Usuario</b>			
<b>No</b>	<b>Pregunta</b>	<b>Respuesta (si/no)</b>	
1	¿La funcionalidad resuelve su necesidad?	Si	
2	¿Usaría esta herramienta en su vida real?	Si	
3	¿La recomendaría a otros?	Si	
<b>Conclusión de la Validación con Usuario</b>			
<b>No</b>	<b>Item</b>	<b>Resultado (si/no)</b>	
1	Funcionalidad es aceptada	Si	
2	Requiere mejoras	Si	
3	Requiere rediseño	Si	

### **Firmas de Validación**

\_\_\_\_\_  
 Firma del Usuario  
 Nombre de Usuario

\_\_\_\_\_  
 Firma del Evaluador  
 Nombre Evaluador

**Figura 57**  
*Prueba de Funcionalidad -Inventario Medicamentos*

<b>Nombre del proyecto: MediExpress</b>						
<b>ID del caso de prueba: UAT-003</b>			<b>Escenario de prueba: Crear, visualizar y editar medicamentos</b>			
<b>Funcionalidad Evaluada: Inventario</b>			<b>Sprint de Origen: Sprint 3</b>			
<b>Prueba diseñada por: Equipo UAT MediExpress</b>			<b>Prioridad de prueba (Bajo, medio, alto): Media</b>			
<b>Fecha de ejecución de la prueba: 03/10/2025</b>			<b>Lugar de ejecución de la prueba: Entorno de pruebas UAT MediExpress</b>			
<b>Prueba ejecutada por: Usuario final</b>			<b>Tipo de usuario: Operador entregas</b>			
<b>Descripción:</b> Verificar que el operador puede crear, visualizar, editar y eliminar medicamentos						
<b>Condiciones previas:</b> Usuario registrado con usuario y contraseña válidos. <b>Dependencias:</b> Base de datos de usuarios activa.						
<b>Prueba Funcional Ejecutada por el Usuario</b>						
<b>Pasos</b>	<b>Pasos de prueba</b>	<b>Datos de prueba</b>	<b>Resultado esperado</b>	<b>Resultado actual</b>	<b>Estado (falla o pasa)</b>	<b>Nota</b>
Paso 1	Crear medicamento	“Acetaminofén 500mg”, “capsulas”, 100, 200	Registro exitoso	OK	Pasa	
Paso 2	Ver ficha	-	Información detallada	OK	Pasa	
Paso 3	Editar medicamento	Actualizar stock	Cambio aplicado	OK	Pasa	
<b>Evaluación del Usuario (Percepción Real)</b>						
<b>Criterio</b>		<b>Calificación (1-5)</b>	<b>Observación</b>			
Facilidad de uso		5	Muy intuitivo			
Claridad de la información		5	Formularios claros			
Velocidad del sistema		5	Mejorable			
Utilidad de la función		5	Muy útil			
Confianza al usarla		5	Interfaz confiable			
<b>Incidencias Detectadas por el Usuario</b>						

No	Descripción del Problema	Requiere ajustes (si/no)	Impacto
<b>Nivel de Aceptación del Usuario</b>			
No	Pregunta	Respuesta (si/no)	
1	¿La funcionalidad resuelve su necesidad?	Si	
2	¿Usaría esta herramienta en su vida real?	Si	
3	¿La recomendaría a otros?	Si	
<b>Conclusión de la Validación con Usuario</b>			
No	Item	Resultado (si/no)	
1	Funcionalidad es aceptada	Si	
2	Requiere mejoras	Si	
3	Requiere rediseño	Si	

### Firmas de Validación

\_\_\_\_\_  
 Firma del Usuario  
 Nombre de Usuario

\_\_\_\_\_  
 Firma del Evaluador  
 Nombre Evaluador

**Figura 58**  
*Prueba de Funcionalidad – Entregas*

<b>Nombre del proyecto: MediExpress</b>						
<b>ID del caso de prueba: UAT-004</b>				<b>Escenario de prueba: Flujo de entrega de medicamento</b>		
<b>Funcionalidad Evaluada: Asignación - En proceso- Pendiente- Completada</b>				<b>Sprint de Origen: Sprint 4</b>		
<b>Prueba diseñada por: Equipo UAT MediExpress</b>				<b>Prioridad de prueba (Bajo, medio, alto): Alta</b>		
<b>Fecha de ejecución de la prueba: 18/10/2025</b>				<b>Lugar de ejecución de la prueba: Entorno de pruebas UAT MediExpress</b>		
<b>Prueba ejecutada por: Usuario final</b>				<b>Tipo de usuario: Operador entrega</b>		
<b>Descripción:</b> Verificar que la asignación de la entrega al operador se realice correctamente						
<b>Condiciones previas:</b> Usuario registrado con usuario y contraseña válidos. <b>Dependencias:</b> Base de datos de usuarios activa.						
<b>Prueba Funcional Ejecutada por el Usuario</b>						
<b>Pasos</b>	<b>Pasos de prueba</b>	<b>Datos de prueba</b>	<b>Resultado esperado</b>	<b>Resultado actual</b>	<b>Estado (falla o pasa)</b>	<b>Nota</b>
Paso 1	Asignar mensajero	Juan Pérez	Solicitud asignada	OK	Pasa	
Paso 2	Ver entregas asignadas	-	Panel visible	OK	Pasa	
Paso 3	Cambiar estado	Pendiente - Completada	Cambio registrado con hora y fecha	OK	Pasa	
<b>Evaluación del Usuario (Percepción Real)</b>						
<b>Criterio</b>		<b>Calificación (1–5)</b>		<b>Observación</b>		
Facilidad de uso		5		Muy intuitivo		
Claridad de la información		5		Información clara		
Velocidad del sistema		5		Velocidad aceptable		
Utilidad de la función		5		Muy útil		

Confianza al usarla	5	Interfaz confiable	
<b>Incidenias Detectadas por el Usuario</b>			
No	Descripción del Problema	Requiere ajustes (si/no)	Impacto
<b>Nivel de Aceptación del Usuario</b>			
No	Pregunta	Respuesta (si/no)	
1	¿La funcionalidad resuelve su necesidad?	Si	
2	¿Usaría esta herramienta en su vida real?	Si	
3	¿La recomendaría a otros?	Si	
<b>Conclusión de la Validación con Usuario</b>			
No	Item	Resultado (si/no)	
1	Funcionalidad es aceptada	Si	
2	Requiere mejoras	Si	
3	Requiere rediseño	Si	

### Firmas de Validación

\_\_\_\_\_  
Firma del Usuario  
Nombre de Usuario

\_\_\_\_\_  
Firma del Evaluador  
Nombre Evaluador

**Figura 59**  
*Prueba de Funcionalidad – Notificaciones*

<b>Nombre del proyecto: MediExpress</b>						
<b>ID del caso de prueba: UAT-005</b>			<b>Escenario de prueba: Notificaciones automáticas y manuales</b>			
<b>Funcionalidad Evaluada: Asignación - En proceso- Pendiente- Completada</b>			<b>Sprint de Origen: Sprint 4</b>			
<b>Prueba diseñada por: Equipo UAT MediExpress</b>			<b>Prioridad de prueba (Bajo, medio, alto): Alta</b>			
<b>Fecha de ejecución de la prueba: 31/10/2025</b>			<b>Lugar de ejecución de la prueba: Entorno de pruebas UAT MediExpress</b>			
<b>Prueba ejecutada por: Usuario final</b>			<b>Tipo de usuario: Paciente</b>			
<b>Descripción: Verificar que lleguen correctamente las notificaciones</b>						
<b>Condiciones previas: Usuario registrado con usuario y contraseña válidos. Dependencias: Base de datos de usuarios activa.</b>						
<b>Prueba Funcional Ejecutada por el Usuario</b>						
<b>Pasos</b>	<b>Pasos de prueba</b>	<b>Datos de prueba</b>	<b>Resultado esperado</b>	<b>Resultado actual</b>	<b>Estado (falla o pasa)</b>	<b>Nota</b>
Paso 1	Realizar solicitud	Datos solicitud	Solicitud realizada	OK	Pasa	
Paso 2	Realizar asignación	Datos asignación	Asignación realizada	OK	Pasa	
Paso 3	Validar panel de notificaciones	-	Notificaciones visibles	OK	Pasa	
<b>Evaluación del Usuario (Percepción Real)</b>						
<b>Criterio</b>		<b>Calificación (1–5)</b>	<b>Observación</b>			
Facilidad de uso		5	Muy intuitivo			
Claridad de la información		5	Información clara			
Velocidad del sistema		5	Velocidad aceptable			
Utilidad de la función		5	Muy útil			
Confianza al usarla		5	Interfaz confiable			

<b>Incidencias Detectadas por el Usuario</b>			
<b>No</b>	<b>Descripción del Problema</b>	<b>Requiere ajustes (si/no)</b>	<b>Impacto</b>
1	Solicitan horarios personalizados (ajuste solicitado).	Si	Alto
<b>Nivel de Aceptación del Usuario</b>			
<b>No</b>	<b>Pregunta</b>	<b>Respuesta (si/no)</b>	
1	¿La funcionalidad resuelve su necesidad?	Si	
2	¿Usaría esta herramienta en su vida real?	Si	
3	¿La recomendaría a otros?	Si	
<b>Conclusión de la Validación con Usuario</b>			
<b>No</b>	<b>Item</b>	<b>Resultado (si/no)</b>	
1	Funcionalidad es aceptada	Si	
2	Requiere mejoras	Si	
3	Requiere rediseño	Si	

### **Firmas de Validación**

\_\_\_\_\_  
 Firma del Usuario  
 Nombre de Usuario

\_\_\_\_\_  
 Firma del Evaluador  
 Nombre Evaluador

## Resultados

El análisis del proceso actual de entrega de medicamentos en el barrio El Carmen permitió identificar fallas estructurales que afectaban directamente la experiencia de los usuarios, especialmente adultos mayores y personas con movilidad reducida. Entre los principales hallazgos se encontraron tiempos excesivos de espera, dificultades en los desplazamientos hacia los puntos de entrega, caídas recurrentes del sistema de validación y ausencia de trazabilidad del estado de los medicamentos. La recolección de información mediante la encuesta semiestructurada, permitió documentar estas barreras y caracterizar las necesidades reales de la población, lo cual sirvió como base para establecer los requisitos funcionales y no funcionales del sistema.

Por otro lado, el diseño de la arquitectura de la aplicación se estructuró a partir de los hallazgos del análisis previo, priorizando seguridad, escalabilidad y facilidad de mantenimiento. Se definió una arquitectura modular, lo que permitió crear una separación clara entre las capas de presentación, dominio y datos. Asimismo, se incorporaron lineamientos de interoperabilidad y estándares de seguridad orientados a garantizar la protección de información sensible. El resultado fue una arquitectura sólida y flexible, capaz de soportar los flujos principales del sistema y de adaptarse a futuras ampliaciones.

De acuerdo con lo anterior, el desarrollo de la aplicación se ejecutó siguiendo la metodología Scrum, mediante la planificación de seis sprints que permitieron construir el sistema de manera iterativa e incremental. Durante cada sprint se completaron funcionalidades clave como autenticación, solicitudes de medicamentos, catálogo de fármacos, módulo de entregas y notificaciones. La revisión constante con los interesados permitió validar avances y ajustar elementos técnicos y visuales según las necesidades emergentes. Este enfoque ágil permitió

mantener una evolución controlada del producto, garantizar entregas funcionales al final de cada iteración y sostener un proceso de desarrollo alineado con los objetivos del proyecto.

Finalmente, la evaluación del producto mínimo viable permitió verificar el correcto funcionamiento de los módulos desarrollados y medir la experiencia de los usuarios en términos de usabilidad. Las pruebas funcionales confirmaron el desempeño adecuado del sistema en los flujos esenciales y la retroalimentación directa de los participantes evidenció una interacción fluida, comprensión adecuada de la interfaz y una reducción de fricciones en tareas clave. Los resultados de esta evaluación inicial permitieron identificar fortalezas del diseño, así como aspectos susceptibles de mejora en futuras iteraciones.

Prototipo Funcional aplicación MEDIEXPRESS: <https://youtu.be/gBjF5x82YII>

## Conclusiones y Recomendaciones

### Conclusiones

El desarrollo del presente proyecto permitió demostrar la viabilidad técnica y metodológica de implementar una solución tecnológica orientada a optimizar el proceso de entrega de medicamentos en un entorno comunitario vulnerable. La aplicación de un enfoque de investigación aplicada con diseño mixto facilitó la identificación precisa de necesidades funcionales y operativas del contexto estudiado.

La adopción de la metodología ágil Scrum favoreció la construcción iterativa del Producto Mínimo Viable, permitiendo validar progresivamente los requerimientos definidos y ajustar el sistema de acuerdo con la retroalimentación obtenida. Este enfoque confirmó la pertinencia del uso de metodologías ágiles en proyectos de intervención tecnológica en salud comunitaria.

Desde el punto de vista técnico, la integración de Angular, Node.js y MongoDB evidenció ser adecuada para el desarrollo de una arquitectura modular, escalable y alineada con principios de seguridad e interoperabilidad. Esto demuestra que tecnologías web modernas pueden aplicarse de manera efectiva en soluciones orientadas a la gestión de servicios de salud.

Asimismo, el proyecto permitió evidenciar que la incorporación de criterios de accesibilidad resulta fundamental cuando la población objetivo está conformada por adultos mayores y personas con movilidad reducida. La adaptación de la interfaz a estas condiciones constituye un factor clave para la apropiación tecnológica.

En términos generales, el trabajo aporta un modelo replicable de implementación tecnológica que integra autorización, gestión logística y experiencia de usuario accesible,

contribuyendo al fortalecimiento de iniciativas de salud digital en contextos locales.

## **Recomendaciones**

Con el fin de fortalecer el funcionamiento, escalabilidad y sostenibilidad del sistema MediExpress, se plantean las siguientes recomendaciones orientadas a su evolución futura. En primer lugar, se sugiere avanzar hacia la automatización del proceso logístico, integrando algoritmos de optimización de rutas y asignación inteligente de mensajeros, lo que reduciría tiempos de entrega y aumentaría la eficiencia operativa. Asimismo, es recomendable ampliar las capacidades del módulo de notificaciones, permitiendo configuración avanzada de horarios, recordatorios personalizados y comunicación bidireccional para mejorar la interacción con los usuarios.

De igual manera, se aconseja fortalecer los mecanismos de seguridad y protección de datos, incorporando auditorías continuas, respaldo automático de información y monitoreo proactivo de accesos según lineamientos OWASP más recientes. También sería pertinente integrar un sistema de analítica de datos, que permita obtener métricas clave sobre solicitudes, tiempos de respuesta, zonas críticas y perfiles de uso, facilitando la toma de decisiones basadas en evidencia.

Finalmente, se recomienda evaluar la posibilidad de escalar el sistema hacia una infraestructura en la nube más robusta, habilitar interoperabilidad con otras entidades de salud mediante APIs estandarizadas y considerar el desarrollo de una versión de la aplicación orientada a cuidadores o familiares. Estas mejoras, en conjunto, permitirán potenciar el impacto social del producto, aumentar su confiabilidad y garantizar su continuidad en escenarios de mayor demanda.

## Referencias Bibliográficas

Administradora de los Recursos del Sistema General de Seguridad Social en Salud (ADRES).

(2025). *No hay desabastecimiento generalizado de medicamentos en el país*. <https://www.adres.gov.co/sala-de-prensa/noticias/no-hay-desabastecimiento-generalizado-de-medicamentos-en-el-pais>

Adobe XD Ideas. (2024). *UI vs. UX design: What's the difference?*

<https://xd.adobe.com/ideas/principles/web-design/ui-vs-ux-design/>

Amazon Web Services. (2024). *What is DevOps?*

<https://aws.amazon.com/devops/what-is-devops/>

Angular Team. (2024). *Angular documentation*. Google LLC. <https://angular.dev>

Angular. (s. f.). *Angular architecture overview*. <https://angular.io/guide/architecture>

Arias, J. F. (2021). *Logística farmacéutica y distribución de última milla en Colombia*. Revista Ingeniería Industrial, 15(2), 33–45.

<https://revistas.udistrital.edu.co/index.php/ingenieria/article/view/17045>

Atlassian. (2023). *What is DevOps?* <https://www.atlassian.com/devops>

Atlassian. (2026). *Minimum viable product (MVP): What it is & how to start*.

<https://www.atlassian.com/agile/product-management/minimum-viable-product>

Auth0. (2023). *JSON Web Tokens (JWT) and role-based access control (RBAC)*.

<https://auth0.com/docs/secure/tokens/json-web-tokens>

Baeldung. (2023). *The repository pattern*. <https://www.baeldung.com/java-repository-pattern>

Bass, L., Clements, P., & Kazman, R. (2013). *Software architecture in practice* (3rd ed.).

Addison-Wesley. <https://www.pearson.com/en-us/subject-catalog/p/software-architecture-in-practice/P200000003295>

Bass, L., Clements, P., & Kazman, R. (2021). *Software architecture in practice*.

<https://www.sei.cmu.edu/publications/books/software-architecture-in-practice/>

Bernal, A. (2010). *Administración de servicios*. McGraw-Hill Education.

Bernal, C. A. (2010). *Metodología de la investigación*. [https://abacoenred.com/wp-](https://abacoenred.com/wp-content/uploads/2019/02/Metodologia-de-la-investigacion-Carlos-Bernal.pdf)

[content/uploads/2019/02/Metodologia-de-la-investigacion-Carlos-Bernal.pdf](https://abacoenred.com/wp-content/uploads/2019/02/Metodologia-de-la-investigacion-Carlos-Bernal.pdf)

Boysen, N., Briskorn, D., & Schwerdfeger, S. (2021). *Last-mile delivery concepts: A survey from an operational research perspective*.

<https://www.sciencedirect.com/science/article/pii/S0377221720307708>

Brooke, J. (1996). *SUS: A quick and dirty usability scale*.

<https://hell.meiert.org/core/pdf/sus.pdf>

Brown, S. (s. f.). *The C4 model for visualising software architecture*. <https://c4model.com>

Caracol Radio. (2025, 4 de abril). *42 % de quejas en los servicios de salud en Bogotá son por el difícil acceso a medicamentos.*

<https://caracol.com.co/2025/04/04/42-de-quejas-en-los-servicios-de-salud-en-bogota-son-por-el-dificil-acceso-a-medicamentos/>

CEPAL. (2020). *Personas vulnerables y desigualdad social en América Latina.*

<https://www.cepal.org/es/publicaciones>

Colmenares, D. (2024). *Optimización de la cadena logística farmacéutica urbana en*

*Colombia.* <https://repository.unimilitar.edu.co/handle/10654/43121>

Congreso de la República de Colombia. (1981). *Ley 23 de 1981: Código de ética médica.*

<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=6528>

Congreso de la República de Colombia. (2008). *Ley 1266 de 2008: Habeas data financiero.*

<https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=34488>

Congreso de la República de Colombia. (2012). *Ley 1581 de 2012 – Protección de datos*

*personales.* <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=49981>

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to Algorithms*

(4th ed.). MIT Press. <https://mitpress.mit.edu/9780262046305/introduction-to-algorithms/>

Creswell, J. W., & Creswell, J. D. (2018). *Research design: Qualitative, quantitative, and mixed methods approaches* (5th ed.). SAGE Publications.

<https://us.sagepub.com/en-us/nam/research-design/book255675>

Creswell, J. W., & Plano Clark, V. L. (2018). *Designing and conducting mixed methods research* (3rd ed.). SAGE Publications. <https://us.sagepub.com/en-us/nam/designing-and-conducting-mixed-methods-research/book241842>

Defensoría del Pueblo. (2023). *Defensoría del Pueblo alerta sobre la crisis en el acceso a medicamentos en Colombia*. <https://www.defensoria.gov.co/es/web/guest/-/defensoria-alerta-crisis-acceso-medicamentos-colombia>

Defensoría del Pueblo. (2025). *Medicamentos inaccesibles: derechos vulnerados*. <https://www.defensoria.gov.co/documents/20123/3449669/Informe-audiencia-Defensorial-Salud-2025.pdf>

Departamento Administrativo Nacional de Estadística (DANE). (2014). *Proyecciones de población por sexo y grupos de edad. Bogotá D. C. y localidades*. <https://www.dane.gov.co>

Departamento Administrativo Nacional de Estadística (DANE). (2015). *Encuesta Multipropósito 2014 – resultados para la ciudad de Bogotá*. <https://www.dane.gov.co/index.php/estadisticas-por-tema/pobreza-y-condiciones-de-vida/encuesta-multiproposito/encuesta-multiproposito-2014>

Departamento Administrativo Nacional de Estadística (DANE). (2020). *Censo Nacional de Población y Vivienda 2018*. <https://www.dane.gov.co>

Departamento Administrativo Nacional de Estadística (DANE). (2020). *Resultados del Censo Nacional de Población y Vivienda 2018 – Proyecciones y caracterización poblacional*.

<https://www.dane.gov.co/index.php/estadisticas-por-tema/demografia-y-poblacion/censo-nacional-de-poblacion-y-vivenda-2018>

Departamento Administrativo Nacional de Estadística. (2021). *Indicadores de salud y condiciones sociodemográficas para Bogotá D. C.*

<https://saludata.saludcapital.gov.co>

Docker, Inc. (2024). *Docker documentation.*

<https://docs.docker.com>

DocMorris AG. (2023). *E-Rezept – receta electrónica y entrega de medicamentos online con la app DocMorris.* <https://www.docmorris.de/rezefpte/rezept-einloesen/e-rezept>

Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2018). *Fundamentals of business process management* (2nd ed.). Springer.

<https://link.springer.com/book/10.1007/978-3-662-56509-4>

El Espectador. (2025). *Largas filas durante la madrugada para acceder a medicamentos en el sur de Bogotá.* <https://www.elespectador.com/bogota/largas-filas-durante-la-madrugada-para-acceder-a-medicamentos-en-el-sur-de-bogota/>

El Tiempo. (2024). *Largas filas, demoras y fallas: el drama de los usuarios para recibir medicamentos en las EPS.* <https://www.eltiempo.com/salud/otro-drama-para-usuarios-de-nueva-eps-largas-filas-y-falta-de-entrega-de-medicamentos-3522511>

- European Medicines Agency. (2021). *Digital transformation in medicines regulation*.  
<https://www.ema.europa.eu/en/human-regulatory/overview/digital-transformation>
- European Union. (2016). *General Data Protection Regulation (GDPR)*. <https://eur-lex.europa.eu/eli/reg/2016/679/oj>
- European Union. (2016). *Regulation (EU) 2016/679 (General Data Protection Regulation)*.  
<https://eur-lex.europa.eu/>
- Evans, E. (2004). *Domain-driven design: Tackling complexity in the heart of software*.  
<https://www.domainlanguage.com/ddd/>
- Fielding, R. T. (2000). *Architectural styles and the design of network-based software architectures* (Doctoral dissertation).  
[https://www.ics.uci.edu/~fielding/pubs/dissertation/rest\\_arch\\_style.htm](https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm)
- Fowler, M. (2020). *Presentation domain data layering*.  
<https://martinfowler.com/bliki/PresentationDomainDataLayering.html>
- Fowler, M. (2015). *Monolith first*. <https://martinfowler.com/bliki/MonolithFirst.html>
- García, J., & Torres, M. (2020). *Tecnologías aplicadas a la logística y la cadena de suministro*.  
<https://www.researchgate.net/publication/343567890>
- Gartner. (2025). *IT infrastructure*. <https://www.gartner.com/en/information-technology/glossary/it-infrastructure>

GeeksforGeeks. (2023). *Introduction to programming logic*.

<https://www.geeksforgeeks.org/introduction-to-programming-logic/>

Google Cloud. (2024). *Data minimization and privacy principles*.

<https://cloud.google.com/privacy>

Google Cloud. (2024). *Data minimization best practices*. Google Cloud Architecture

Framework.<https://cloud.google.com/architecture/framework/security/data-minimization>

Google Cloud. (2024). *Scalability explained*. <https://cloud.google.com/architecture/scalability>

Google Cloud. (2024). *What is scalability?* <https://cloud.google.com/architecture/scalability>

Google Developers. (2023). *Progressive Web Apps (PWA): Web capabilities on mobile*.

<https://developers.google.com/web/progressive-web-apps>

Google Developers. (2024). *Angular documentation*. <https://angular.dev>

Google Developers. (2024). *Microinteractions and feedback*.

<https://developers.google.com/design/ui/interaction>

Hernández-Sampieri, R., Fernández-Collado, C., & Baptista, M. del P. (2022). *Metodología de la investigación* (7.<sup>a</sup> ed.). McGraw-Hill Education.

<https://www.mheducation.com.mx/metodologia-de-la-investigacion-hernandez-sampieri.html>

HL7 International. (2019). *FHIR Release 4*.<https://www.hl7.org/fhir/>

IBM. (2024). *What is programming logic?* <https://www.ibm.com/topics/programming-logic>

IBM. (s. f.). *IT infrastructure*. <https://www.ibm.com/topics/it-infrastructure>

IBM. (s. f.). *Software and systems architecture*.

<https://www.ibm.com/cloud/architecture/architectures>

ICS UX Design Team. (2022). *User feedback in UX design*. <https://www.ics.com/blog/user-feedback-ux-design>

Interaction Design Foundation. (s. f.). *Mobile usability*. <https://www.interaction-design.org/literature/topics/mobile-usability>

Interaction Design Foundation. (2023). *Feedback in user interface design*.

<https://www.interaction-design.org/literature/topics/feedback>

Interaction Design Foundation. (2024). *User interface (UI) design*. <https://www.interaction-design.org/literature/topics/user-interface-design>

Interaction Design Foundation. (2024). *User-centered design: Definition and fundamentals*.

<https://www.interaction-design.org/literature/topics/user-centered-design>

International Organization for Standardization. (2018). *ISO 9241-11:2018 Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts*.

<https://www.iso.org/standard/63500.html>

International Organization for Standardization. (2019). *ISO 9241-210:2019 Ergonomics of human-system interaction*. <https://www.iso.org/standard/77520.html>

International Organization for Standardization. (2019). *ISO 9241-11: Ergonomics of human-system interaction — Usability*. <https://www.iso.org/standard/63500.html>

Institute of Electrical and Electronics Engineers. (2023). Software engineering standards and best practices. IEEE. <https://www.ieee.org>

ISO/IEC. (2011). *ISO/IEC 25010: Systems and software quality models*. ISO.

ISO/IEC. (2011). *ISO/IEC 25010:2011 systems and software engineering—Systems and software quality requirements and evaluation (SQuaRE)*. <https://www.iso.org/standard/35733.html>

ISO/IEC. (2022). *Information security management systems — Overview*. <https://www.iso.org/isoiec-27001-information-security.html>

Kessler, R., van der Linden, J., & Petersen, S. (2022). *Agile development and user participation in social and educational technology projects*. *Computers in Human Behavior Reports*, 5, 100145. <https://www.sciencedirect.com/science/article/pii/S2451958822000123>

Larman, C. (2019). *Agile and iterative development: A manager's guide*. <https://www.agilealliance.org/resources/books/agile-and-iterative-development/>

Liskov, B. (1987). *Keynote address—Data abstraction and hierarchy*. <https://www.cs.cmu.edu/~wing/publications/Liskov87.pdf>

MDN Web Docs. (2025). *Server-side website programming*. <https://developer.mozilla.org/en-US/docs/Learn/Server-side>

Mell, P., & Grance, T. (2011). *The NIST definition of cloud computing (Special Publication 800-145)*. National Institute of Standards and Technology.

<https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>

Microsoft. (s. f.). Arquitecturas de aplicaciones web comunes — ¿Qué es una aplicación monolítica? Microsoft Learn. <https://learn.microsoft.com/es-es/dotnet/architecture/modern-web-apps-azure/common-web-application-architectures#what-is-a-monolithic-application>

Microsoft Azure Architecture Center. (2023). *RESTful web API design*. Microsoft Learn. <https://learn.microsoft.com/en-us/azure/architecture/best-practices/api-design>

Microsoft Learn. (2023). *Dependency injection in .NET*. <https://learn.microsoft.com/en-us/dotnet/core/extensions/dependency-injection>

Microsoft Learn. (2023). *DevOps and application lifecycle management*. <https://learn.microsoft.com/en-us/devops/>

Microsoft Learn. (2023). *Layered architecture*. <https://learn.microsoft.com/en-us/azure/architecture/guide/architecture-styles/layered>

Microsoft Learn. (2023). *Scalability and performance design*. <https://learn.microsoft.com/en-us/azure/architecture/best-practices/scalability>

Microsoft Learn. (2024). *Introduction to APIs*. <https://learn.microsoft.com/en-us/training/modules/introduction-to-apis/>

Ministerio de Salud y Protección Social. (2011). *Ley 1438 de 2011: Reforma del sistema general de seguridad social en salud*.

[https://www.minsalud.gov.co/Normatividad\\_Nuevo/Ley%201438%20de%202011.pdf](https://www.minsalud.gov.co/Normatividad_Nuevo/Ley%201438%20de%202011.pdf)

Ministerio de Salud y Protección Social. (2015). *Ley estatutaria 1751 de 2015: Derecho fundamental a la salud*.

[https://www.minsalud.gov.co/Normatividad\\_Nuevo/Ley%201751%20de%202015.pdf](https://www.minsalud.gov.co/Normatividad_Nuevo/Ley%201751%20de%202015.pdf)

Ministerio de Salud y Protección Social. (2020). Colombia mejoró en la entrega domiciliaria de medicamentos en los dos últimos meses.

<https://www.minsalud.gov.co/Paginas/Colombia-mejoro-en-la-entrega-domiciliaria-de-medicamentos-en-los-dos-ultimos-meses-.aspx>

Ministerio de Salud y Protección Social. (2020). *EPS e IPS garantizarán atención a poblaciones vulnerables y entrega domiciliaria de medicamentos durante la emergencia sanitaria*.

<https://www.minsalud.gov.co/Paginas/EPS-e-IPS-garantizaran-atencion-a-poblaciones-vulnerables-.aspx>

Ministerio de Salud y Protección Social. (2020). *Lineamientos para la entrega domiciliaria de medicamentos y tecnologías en salud durante la emergencia sanitaria por COVID-*

*19*. <https://www.minsalud.gov.co/salud/publica/PET/Paginas/Covid-19.aspx>

Ministerio de Salud y Protección Social. (2022). *Entidades promotoras de salud (EPS)*.

<https://www.minsalud.gov.co/salud/PServicios/Paginas/eps.aspx>

Ministerio de Tecnologías de la Información y las Comunicaciones. (2022). *Política de*

*seguridad digital*. [https://www.mintic.gov.co/portal/inicio/PoliticadeGobierno-](https://www.mintic.gov.co/portal/inicio/PoliticadeGobierno-Digital/Seguridad-Digital/)

[Digital/Seguridad-Digital/](https://www.mintic.gov.co/portal/inicio/PoliticadeGobierno-Digital/Seguridad-Digital/)

Ministerio de Tecnologías de la Información y las Comunicaciones. (2023). *Marco normativo*

*TIC y comercio electrónico*. <https://www.mintic.gov.co/portal/inicio/Normativa/>

MongoDB, Inc. (2024). *The MEAN stack*.

<https://www.mongodb.com/resources/languages/mean-stack>

NgRx Team. (2023). *NgRx documentation: Reactive state management for Angular*.

<https://ngrx.io/docs>

Nielsen, J. (2012). *Usability 101: Introduction to usability*. Nielsen Norman Group.

<https://www.nngroup.com/articles/usability-101-introduction-to-usability/>

Nielsen Norman Group. (2018). *Usability metrics*. [https://www.nngroup.com/articles/usability-](https://www.nngroup.com/articles/usability-metrics/)

[metrics/](https://www.nngroup.com/articles/usability-metrics/)

Nielsen Norman Group. (2020). *10 usability heuristics for user interface design*.

<https://www.nngroup.com/articles/ten-usability-heuristics/>

Nielsen Norman Group. (2020). *Usability 101: Introduction to usability*.

<https://www.nngroup.com/articles/usability-101-introduction-to-usability/>

Nielsen Norman Group. (2020). *Visibility of system status*.

<https://www.nngroup.com/articles/visibility-system-status/>

Nielsen Norman Group. (2023). *The definition of user experience (UX)*.

<https://www.nngroup.com/articles/definition-user-experience/>

Nielsen Norman Group. (2023). *User-centered design*. <https://www.nngroup.com/articles/user-centered-design/>

Nielsen, J. (1994). *Usability engineering*. Nielsen Norman Group.

<https://www.nngroup.com/articles/usability-engineering/>

Nielsen, J. (2020). *Visibility of system status*. Nielsen Norman Group.

<https://www.nngroup.com/articles/visibility-system-status/>

Norman, D. A. (2013). *The design of everyday things: Revised and expanded edition*. Nielsen

Norman Group. <https://www.nngroup.com/books/design-everyday-things-revised/>

Nygaard, M. (2011). *Documenting architecture decisions*.

<https://cognitect.com/blog/2011/11/15/documenting-architecture-decisions>

Open Web Application Security Project (OWASP). (2024). *Cryptographic storage cheat*

*sheet*. [https://cheatsheetseries.owasp.org/cheatsheets/Cryptographic\\_Storage\\_Cheat\\_Sheet.html](https://cheatsheetseries.owasp.org/cheatsheets/Cryptographic_Storage_Cheat_Sheet.html)

Open Web Application Security Project (OWASP). (2024). *OWASP API security top 10 – 2023*.

<https://owasp.org/www-project-api-security/>

Oracle. (2024). *Client-server architecture*. <https://www.oracle.com/database/what-is-client-server-architecture/>

Oracle. (2024). *Layered architecture*. <https://www.oracle.com/architecture/layered-architecture/>

Organización Mundial de la Salud. (2011). *World report on disability*.

<https://www.who.int/publications/i/item/9789241564182>

Organización Mundial de la Salud. (2015). *World report on ageing and health*.

<https://www.who.int/publications/i/item/9789241565042>

Organización Mundial de la Salud. (2020). *Access to Medicines and Health Products*.

<https://www.who.int/our-work/access-to-medicines-and-health-products>

Organización Mundial de la Salud. (2020). Medicamentos esenciales: disponibilidad,

asequibilidad, calidad y uso (fact sheet). <https://www.who.int/es/news-room/fact-sheets/detail/essential-medicines>

Organización Mundial de la Salud. (2021). *Global strategy on digital health 2020–2025*.

<https://www.who.int/publications/i/item/9789240020924>

Organización Mundial de la Salud. (2021). *WHO guideline: Recommendations on digital health*

*interventions*. <https://www.who.int/publications/i/item/WHO-RHR-18.06>

Organización Panamericana de la Salud. (2014). *Logística y distribución de medicamentos*.

<https://www.paho.org/es/documentos>

OWASP Foundation. (2024). *OWASP Top 10*. <https://owasp.org/www-project-top10/>

OWASP. (2024). *OWASP top ten web application security risks*. <https://owasp.org/www-project-top-ten/>

Oyarvide, L., & Herrera, M. (2021). *Usability evaluation of mobile applications for home delivery services*. *IEEE Access*, 9, 134221–134231.  
<https://ieeexplore.ieee.org/document/9523147>

Personería de Bogotá, D. C. (2025, 4 de abril). 42 % de las quejas de servicios de salud son por dificultad en acceso a medicamentos. Nota de prensa.  
<https://www.personeriabogota.gov.co/sala-de-prensa/notas-de-prensa/item/1311-42-de-las-quejas-de-servicios-de-salud-son-por-dificultad-en-acceso-a-medicamentos-personeria-distrital>

Pressman, R. S. (2014). *Software engineering concepts*.  
<https://www.mheducation.com/highered/product/software-engineering-pressman/M9780078022128.html>

Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: A practitioner's approach* (9th ed.). <https://www.mheducation.com/highered/product/software-engineering-practitioner-s-approach-pressman/M9781260548006.html>

Red Hat. (2023). *What is an API?* <https://www.redhat.com/en/topics/api/what-are-apis>

Ries, E. (2011). *Minimum viable product (MVP)*.

<https://www.startuplessonslearned.com/2009/08/minimum-viable-product-guide.html>

Rozanski, N., & Woods, E. (2012). *Software systems architecture: Working with stakeholders using viewpoints and perspectives* (2nd ed.). Addison-Wesley.

<https://www.pearson.com/en-us/subject-catalog/p/software-systems-architecture/P200000003302>

Schwaber, K., & Sutherland, J. (2020). *The Scrum guide: The definitive guide to Scrum*.

<https://scrumguides.org/scrum-guide.html>

Scrum.org. (2020). *The Scrum guide: The definitive guide to Scrum*.

<https://www.scrumguides.org/scrum-guide.html>

Shneiderman, B. (2016). *The eight golden rules of interface design*.

<https://www.cs.umd.edu/~ben/goldenrules.html>

Sommerville, I. (2011). *Software engineering concepts*. <https://iansommerville.com/software-engineering-book/>

Sommerville, I. (2020). *Software engineering* (10th ed.). <https://iansommerville.com/software-engineering-book/>

Stallings, W. (2020). *Computer security: Principles and practice*. <https://www.pearson.com/en-us/subject-catalog/p/computer-security-principles-and-practice/P200000003294>

Superintendencia de Industria y Comercio. (2024). *Protección de datos personales*.

<https://www.sic.gov.co/proteccion-de-datos-personales>

Weske, M. (2019). *Business process management: Concepts, languages, architectures* (3rd ed.).

Springer. <https://link.springer.com/book/10.1007/978-3-030-26619-6>

World Health Organization. (2021). *Global strategy on digital health 2020–2025*.

<https://www.who.int/publications/i/item/9789240020924>

World Health Organization. (2021). *mHealth: Use of mobile wireless technologies for public*

*health*. <https://www.who.int/publications/i/item/WHO-RHR-21.04>

World Health Organization. (2021). *Monitoring the cold chain and supply systems using digital*

*health tools*. <https://www.who.int/teams/digital-health-and-innovation>

World Wide Web Consortium. (2018). *Web content accessibility guidelines (WCAG) 2.1*.

<https://www.w3.org/TR/WCAG21/>

World Wide Web Consortium. (2023). *Accessibility principles*.

<https://www.w3.org/WAI/fundamentals/accessibility-principles/>

World Wide Web Consortium. (2023). *WAI-ARIA Authoring Practices 1.2*.

<https://www.w3.org/WAI/ARIA/apg/>

## Apéndices

### Apéndice A

*Formulario Encuesta de análisis sobre el servicio de atención en el punto de reclamación del barrio El Carmen-Tunjuelito*

El formulario completo utilizado para las encuestas de análisis sobre el servicio en el punto de reclamación de medicamentos de El Carmen puede ser consultado en el siguiente enlace:

**Anexo 1 - Encuesta de análisis sobre el servicio de atención en el punto de reclamación del barrio El Carmen-Tunjuelito**

Este instrumento tiene como propósito recolectar información valiosa para conocer la percepción de los pacientes sobre la entrega de medicamentos en el punto de reclamación del barrio El Carmen en Tunjuelito, en el marco del proyecto de trabajo de grado titulado.

1. Nombres y Apellidos \*
2. ¿En qué rango de edad se encuentra? \* Marca solo un óvalo.

0 - 49 años

50-59 años

60-69 años

70-79 años

+80

3. ¿Qué rango de tiempo toma para llegar al punto de reclamación? \*. Marca solo un óvalo.

5 - 10 min

10 - 15 min

15 - 20 min

20 - 25 min

25 - 30 min

30 - 45 min

45 - 60 min

+ 1 hora

4. ¿El tiempo que demora para llegar al punto de reclamación de medicamentos es el adecuado? Marca solo un óvalo. \*

Totalmente en desacuerdo

En desacuerdo

Ni de acuerdo ni en desacuerdo

De acuerdo

Totalmente de acuerdo

5. ¿Cuánto tiempo debe esperar normalmente para ser atendido hasta recibir sus medicamentos? \* Marca solo un óvalo.

20 - 30 min

30 - 45 min

45 - 1 hora

+ 1 hora

6. ¿Padece alguna discapacidad física o movilidad reducida? \* Marca solo un óvalo.

Totalmente en desacuerdo

En desacuerdo

Ni de acuerdo ni en desacuerdo

De acuerdo

Totalmente de acuerdo

7. ¿Solicita frecuentemente sus medicamentos? \* Marca solo un óvalo.

Totalmente en desacuerdo

En desacuerdo

Ni de acuerdo ni en desacuerdo

De acuerdo

Totalmente de acuerdo

8. ¿ En caso de no poder movilizarse, puede algún familiar reclamar sus medicamentos? Marca solo un óvalo. \*

Totalmente en desacuerdo

En desacuerdo

Ni de acuerdo ni en desacuerdo

De acuerdo

Totalmente de acuerdo

9. ¿ Está satisfecho con el servicio prestado en el punto de reclamación de medicamentos? \*

Marca solo un óvalo.

Totalmente en desacuerdo

En desacuerdo

Ni de acuerdo ni en desacuerdo

De acuerdo

Totalmente de acuerdo

10. ¿ Cree que una aplicación móvil que facilite la entrega de medicamentos a \*  
domicilio, podría mejorar el servicio que se presta actualmente? Marca solo un  
óvalo.

Totalmente en  
desacuerdo

En desacuerdo

Ni de acuerdo ni en  
desacuerdo

De acuerdo

Totalmente de acuerdo

11. ¿Puede acceder fácilmente a aplicaciones en dispositivos tecnológicos como \*  
celular, Tablet o computador? Marca solo un óvalo.

Totalmente en  
desacuerdo

En desacuerdo

Ni de acuerdo ni en  
desacuerdo

De acuerdo

Totalmente de acuerdo

### **Autorización para el tratamiento de datos personales**

Autorizo de manera voluntaria, previa, explícita, informada e inequívoca al Proyecto Aplicado MediExpress para que recolecte, almacene y use mis datos personales únicamente con fines académicos e investigativos relacionados con el desarrollo del Proyecto "MediExpress". Los

datos suministrados serán tratados de forma confidencial y serán eliminados una vez transcurrido un período de seis (6) meses después del desarrollo de la investigación. ¿Está de acuerdo con el tratamiento de sus datos personales en los términos descritos? Sí está de acuerdo proceda a enviar este formulario.

## Apéndice B

### Visión del proyecto

A continuación se presenta la visión del proyecto :

<b>Nombre del proyecto</b>	Implementación de una aplicación móvil con Angular para la entrega asistida de medicamentos a adultos mayores en el barrio El Carmen, Tunjuelito – Bogotá
<b>Tipo de proyecto</b>	Prototipo Aplicación Móvil

#### Visión del proyecto:

MediExpress busca convertirse en una solución tecnológica confiable, accesible y centrada en el usuario para optimizar el proceso de entrega de medicamentos en el barrio El Carmen y, posteriormente, en otras zonas de Bogotá. La visión del proyecto es ofrecer una aplicación móvil que permita a los usuarios, especialmente adultos mayores y personas con movilidad reducida, solicitar, rastrear y recibir sus medicamentos de manera segura, rápida y transparente. A través de una plataforma intuitiva, con integración a entidades de salud, mecanismos de auditoría, notificaciones en tiempo real y funcionalidades de apoyo al mensajero, MediExpress aspira a mejorar la eficiencia del servicio, reducir desplazamientos innecesarios y fortalecer la calidad de atención. Con un enfoque escalable y basado en principios de arquitectura moderna y desarrollo ágil, el proyecto proyecta evolucionar hacia un sistema robusto que incremente la seguridad del paciente y la confiabilidad del proceso logístico en el sector salud

**Apéndice C***Enlace de acceso a aplicación MediExpress*

El acceso al demo de la aplicación móvil del proyecto MediExpress está en el siguiente enlace: <https://medi-express2025v-10.vercel.app/login>

## Apéndice D

### *Manual de Usuario MediExpress*

El acceso al manual de la aplicación móvil del proyecto MediExpress está en el siguiente enlace:

[https://drive.google.com/file/d/1pVmoX\\_1-2WliNTIh3RuYzJ7wzS8RH2Fx/view?usp=sharing](https://drive.google.com/file/d/1pVmoX_1-2WliNTIh3RuYzJ7wzS8RH2Fx/view?usp=sharing)

