

SISTEMA DE INFORMACIÓN AMACOM

Sistema de información multiplataforma para el acompañamiento de padres en el proceso de parto.

Mario Hernán Vallejo Huertas, Valentina Escobar Bueno y Jesús Sebastián Vargas Martínez

Trabajo para Optar Título de Ingeniero de Sistemas

Director

Emilio Justiniano Cárcamo Troconis

Maestría en ciencias de la computación

Codirector

Carolina Vargas Porras

Doctora en enfermería

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingeniería de Sistemas e Informática

Bucaramanga

2024

Dedicatoria

Valentina Escobar Bueno: “Estoy profundamente agradecida por todo el camino recorrido, un viaje en el que nunca he estado sola, gracias al constante acompañamiento, amor y apoyo de mi familia; de mis padres maravillosos Marco Tulio y Raquel que siempre han dado lo mejor de ellos para que yo sea feliz; de mi hermana María de los Angeles quien toda la vida fue mi ejemplo a seguir, mi mentora; de mi tío Gerardo quien ha sido como mi otro padre, mi nonita María del Rosario Carvajal de Bueno, quien hasta sus últimos días me enseñó lo fuerte y luchadora que se puede ser; de mis amigos Hecta, Jesús, Filip, Fransito, Sergio y Deivid quienes han impactado en mí de una manera significativa e indescriptible; de todas las personas que están y ya no están en mi vida, ya que cada uno de ellos han aportado gran parte de quien soy ahora y por último me siento muy agradecida del universo y de mi misma por ser quien soy.”

Mario Hernan Vallejo Huertas: “Todo lo que ha llevado al siguiente trabajo de grado no es solo la culminación de mi esfuerzo individual, sino el de todas las personas que lo han hecho posible al apoyarme de muchas formas y tener fe en mí. Especialmente quiero agradecer a mis padres Mario Vallejo y Maria Huertas, por brindarme oportunidades que ellos nunca tuvieron, por darme todo lo que pudieron y hacerme quien soy. A mi hermana Lucy Vallejo, quien ha sido una segunda madre y siempre se ha preocupado y tenido fe en mí, no sé qué habría sido de mí sin ella en estos años en la universidad. Me siento muy orgulloso de ellos y les estaré eternamente en deuda.

Agradezco a mi compañero Sebastián Vargas y compañera Valentina Escobar por su determinación y actitud ante las dificultades que encontramos en el camino. Agradezco también

SISTEMA DE INFORMACIÓN AMACOM

al ing. Alejandro Pérez y la ing. Alejandra Herrera, a quienes considero familia, por haberme enseñado algo muy importante: vivir y amar.

Creo que todos somos lo que las personas han hecho de nosotros. No alcanzo a recordar a todas las personas que han dejado su huella en mi vida. Por eso, a todas a quienes recuerdo y a quienes no, les expreso mi más sincero agradecimiento.”

Jesús Sebastián Vargas Martínez: “Gracias totales e infinitas a mi familia.”

Agradecimientos

Queremos reconocer y agradecer al profesor Emilio Justiniano Carcamo Troconis por su rol esencial durante el desarrollo de nuestro trabajo de grado. Su guía y conocimientos han sido fundamentales en cada fase de este proyecto. A través de su asesoría, aprendimos a enfrentar y superar los desafíos inherentes a un trabajo de esta envergadura. La disponibilidad y apoyo del profesor Emilio nos han permitido no solo avanzar en nuestra investigación, sino también comprender mejor los complejos aspectos de nuestro proyecto. Su contribución no solo a nuestra educación sino a nuestro crecimiento personal ha sido un componente clave en la realización de este trabajo académico. Por ello, le extendemos nuestro más sincero agradecimiento.

Tabla de Contenidos

Introducción	17
1. Generalidades del proyecto.....	18
1.1 Planteamiento y justificación del problema.....	18
1.2 Alcance del proyecto.....	20
2. Objetivos	21
2.1 Objetivo General.....	21
2.2 Objetivos específicos.....	21
3. Marco de Referencia.....	22
3.1 Marco Contextual.....	22
3.1.1 NANDA, NIC y NOC.....	22
3.1.2 Proceso de enfermería.....	23
3.2 Marco Tecnológico.....	24
3.2.1 Tecnologías y Lenguajes.....	24
3.3 Antecedentes.....	27
4. Fase de análisis.....	28
4.1 Análisis de requerimientos.....	28
4.1.1 Requerimientos funcionales para el usuario final.....	29
4.1.2 Requerimientos funcionales para el usuario enfermería.....	30
4.1.3 Requerimientos funcionales para el usuario administrador.....	32
4.1.4 Requerimientos no funcionales.....	35

SISTEMA DE INFORMACIÓN AMACOM

4.2 Casos de uso y escenarios	35
5. Fase de planeación.....	62
5.1 Marco metodológico.....	62
5.2 Planeación de sprints.....	64
5.3 Modelo de datos	67
5.3.1 Diseño del modelo de datos satisfaciendo los requerimientos del proyecto.....	67
6. Fase de desarrollo.....	68
6.1 Desarrollo Back-end.....	69
6.1.1 Conexión con la base de datos y desarrollo de modelos de trabajo.....	70
6.1.2 Implementación de data transfer objects (DTO) y mappers.....	70
6.1.3 Construcción y desarrollo de los modelos de datos	71
6.1.4 Construcción de servicios y repositorios.....	72
6.1.5 Implementación de controladores y validación de tokens de acceso.....	73
6.1.6 Documentación del API REST con Swagger.....	74
6.2 Desarrollo del prototipo web.....	75
6.2.1 Autenticación de usuarios	76
6.2.3 Módulo detalle persona.....	76
6.2.2 Módulos de mantenimiento (administrativos).....	82
6.3 Desarrollo prototipo App Móvil.....	89
6.3.1 Implementación de la capa de presentación.....	90
6.3.2 Implementación de repositorios, entidades y DTO.....	97

SISTEMA DE INFORMACIÓN AMACOM

7. Fase de pruebas.....	98
7.1 Pruebas unitarias.....	98
7.2 Rendimiento y estabilidad en la recepción de peticiones http con PostMan.....	99
7.2.1 Pruebas de funcionales con PostMan.....	100
7.2.2 Pruebas de rendimiento con PostMan.....	101
8. Conclusiones.....	103
9. Trabajo a futuro.....	105
Referencias.....	106
Anexos.....	108

Lista de tablas

Tabla 1 Requerimientos funcionales del usuario final.....	29
Tabla 2 Requerimientos funcionales del usuario de enfermería.....	31
Tabla 3 Requerimientos funcionales del usuario administrador.....	32
Tabla 4 Requerimientos no funcionales.....	35
Tabla 5 CU. N°1.....	35
Tabla 6 CU. N°2.....	36
Tabla 7 CU N°3.....	38
Tabla 8 CU N°4.....	39
Tabla 9 CU N°5.....	41
Tabla 10 CU N°6.....	42
Tabla 11 CU N°7.....	44
Tabla 12 CU N°8.....	45
Tabla 13 CU N° 9.....	47
Tabla 14 CU N°10.....	48
Tabla 15 CU N°11.....	50
Tabla 16 CU N°12.....	51
Tabla 17 CU N°13.....	53
Tabla 18 CU N°14.....	54
Tabla 19 CU N°15.....	56
Tabla 20 CU N°16.....	57
Tabla 21 CU N°17.....	59
Tabla 22 CU N°18.....	60
Tabla 23 CU N°19.....	60
Tabla 24 CU N°20.....	61
Tabla 25 Programación de sprints.....	64

SISTEMA DE INFORMACIÓN AMACOM

Tabla 26 Programación sprint N° 1.....	64
Tabla 27 Programación sprint N° 2.....	64
Tabla 28 Programación sprint N° 3.....	66
Tabla 29 Programación sprint N° 4.....	66

Lista de figuras

Figura 1: Fase del marco metodológico.	48
Figura 2: Flujo diseño del modelo de base de datos relacional	53
Figura 3: Flujo gestión de situaciones usuario final y enfermería.	54
Figura 4: Diagrama de arquitectura.	55
Figura 5: Estructura interna del back-end.	56
Figura 6: Conexión a base de datos y migraciones.	57
Figura 7: Uso de los DTO en la recepción y envío de datos.	58
Figura 8: Índice de modelos de datos de la aplicación.	58
Figura 9: Servicios de la aplicación.	59
Figura 10: Funcionamiento autenticación JWT.	60
Figura 11: Interfaz gráfica Swagger	61
Figura 12: Índice de estructura angular, admin, autenticación y calendario.	62
Figura 13: Vista inicio sesión del software.	62
Figura 14: Vista listado de personas.	63
Figura 15: Vista consultar persona - gestión situaciones.	64
Figura 16: Vista adicionar situación primera parte, detalle situación.	64
Figura 17: Vista adicionar situación segunda parte, signos de alarma.	65
Figura 18: Vista consultar situación primera parte.	65
Figura 19: Vista consultar situación segunda parte, gestión de diagnósticos.	66
Figura 20: Vista adicionar diagnóstico primera parte.	66
Figura 21: Vista adicionar diagnóstico segunda parte.	66
Figura 22: Vista gestionar bitácoras.	67
Figura 23: Vista calendario.	67
Figura 24: Vista listado de usuarios.	68
	10

SISTEMA DE INFORMACIÓN AMACOM

Figura 25: Vista adicionar usuario.	69
Figura 26: Vista modal consulta de usuario.	69
Figura 27: Vista eliminar usuario.	69
Figura 28: Vista CRUD de géneros.	70
Figura 29: Vista CRUD tipos de documento.	71
Figura 30: Vista CRUD signos de alarma.	71
Figura 31: Vista CRUD signos de temas y subtemas.	72
Figura 32: Vista CRUD de logros.	72
Figura 33: Vista CRUD material de apoyo.	73
Figura 34: Vista CRUD tipos de institución.	73
Figura 35: Vista CRUD de servicios.	74
Figura 36: Vista CRUD de instituciones.	75
Figura 37: Arquitectura hexagonal aplicada al prototipo.	76
Figura 38: Estructura interna prototipo App Móvil	76
Figura 39: Flujo de funcionamiento del prototipo móvil	77
Figura 40: Vistas del módulo de autenticación del prototipo móvil	78
Figura 41: Vistas del módulo de datos de usuario y otros del prototipo móvil	79
Figura 42: Vistas del módulo de situaciones del prototipo móvil	80
Figura 43: Vistas del módulo de bitácora del prototipo móvil	80
Figura 44: Vistas del módulo de agenda del prototipo móvil	81
Figura 45: Vistas del módulo de material de apoyo del prototipo móvil	82
Figura 46: Vistas del módulo de logros del prototipo móvil	83
Figura 47: Envío y recepción de peticiones HTTP.	84
Figura 48: Petición HTTP GET usando PostMan.	85

SISTEMA DE INFORMACIÓN AMACOM

Figura 49: Criterios de prueba generales para la colección de endpoints y edición de 86 variables de colección

Figura 50: Resultados de la prueba funcional 87

Figura 51: Configuración de la prueba de estabilidad. 87

Figura 52: Resumen de la prueba de rendimiento y estabilidad. 88

Glosario

Acompañamiento de enfermería: se define como la acción de ofrecer asistencia en todas aquellas actividades que un paciente no puede realizar por sí mismo, con el objetivo de contribuir a su salud y bienestar.

Arquitectura hexagonal: Es un patrón en el cual se aísla la lógica de la aplicación de las tecnologías subyacentes, ofreciendo ventajas como migrabilidad, flexibilidad y mantenibilidad.

Base de datos: es una recopilación de datos de acuerdo a una lógica y lenguaje determinado que permite el acceso de forma rápida y segura.

Control de versión: es un método de desarrollo distribuido en diferentes ramas principal y secundarias brindando un registro de los cambios realizados al código fuente de una manera ordenada y estructurada.

Data Transfer Objects (DTO): son un patrón de diseño para el transporte de datos entre procesos con el fin de reducir el llamado a métodos, ofreciendo un mejor rendimiento y pudiendo a su vez centralizar labores de serialización en el mismo objeto.

Json web token (JWT): es un estándar el cual permite la identificación de los agentes y asignación de privilegios.

Modelo vista controlador (MVC): es una arquitectura que facilita el mantenimiento y la evolución de una aplicación, dado que funciona a través de un patrón en el diseño del software, que es manejado para implementar interfaces de usuarios, lógica de control y datos.

SISTEMA DE INFORMACIÓN AMACOM

NANDA: sus siglas significan “Asociación Norteamericana de Diagnóstico de Enfermería”, denominada como una organización constituida por enfermera, que está enfocada en la terminología de enfermería estandarizada, a su vez investiga, difunde y perfecciona la nomenclatura, los criterios y la taxonomía de los diagnósticos de enfermería.

NIC: sus siglas traducen “Nursing Interventions Classification”, hace referencia a una clasificación estandarizada de las intervenciones realizadas en enfermería en los pacientes.

NOC: sus siglas traducen “Nursing Outcomes Classification” y hace relación a una clasificación que describe los objetivos a obtener y evaluar en el cuidado del paciente.

Object-Relational mapping (ORM): Es una técnica que permite la transformación de los objetos del sistema en una forma posible para ser almacenados en una base de datos.

Puerperio: etapa de la mujer que comienza inmediatamente después del parto, en este periodo retroceden los cambios que se produjeron durante el embarazo.

Scrum: es una metodología basada en una iteración continua, donde se realizan pequeñas actividades para construir un producto o proyecto de forma incremental.

Sistema de información: se denomina como un conjunto de componentes que interactúan entre sí para recolectar, almacenar, procesar y transmitir información de manera efectiva.

Sprint: se definen como los ciclos de tiempo incluidos dentro de un proyecto Scrum, los cuales permiten tener un ritmo de trabajo con un tiempo establecido.

UML: Lenguaje Unificado Modelado. Medio de representación visual de los diversos conceptos del desarrollo del software mediante un lenguaje sencillo, intuitivo y completo.

UX: La experiencia de usuario (UX) es el proceso encargado de garantizar que la interacción entre el usuario y un producto sea satisfactoria, útil y fácil de usar. Basándose en el contexto de cada usuario.

Resumen

Título: Sistema de información multiplataforma para el acompañamiento de padres en el proceso de posparto.

Autor: Valentina Escobar Bueno, Mario Hernán Vallejo Huertas, Jesús Sebastián Vargas Martínez

Palabras Clave: Sistema de información, puerperio, apoyo, interacción.

Descripción: Se presenta en este documento el prototipo de Sistema de Información que sirva como apoyo y facilitador de interacción entre padres en su proceso de maternidad y el personal de salud. Teniendo como objetivo brindar una herramienta que facilite el acceso al acompañamiento de enfermería a madres primerizas que a falta de experiencia y preparación llegan a desarrollar diversas problemáticas. Esto en pro de minimizar el número e impacto de las mismas.

El ámbito de desarrollo del sistema de información consiste en adecuarse a los requerimientos de cuatro agentes: el súper administrador, el administrador, personal de enfermería y personas en etapa de maternidad, todo esto bajo criterios de análisis, adaptación, rendimiento y seguridad. Los agentes son quienes evidencian de primera mano el funcionamiento del sistema ya que son quienes interactúan con él directamente. La consecución del proceso mencionado se realiza en base al conocimiento adquirido en el desarrollo de la carrera y la información suministrada por los agentes involucrados junto con el uso de herramientas tecnológicas.

* Trabajo de Grado

** Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería de Sistemas e Informática.
Director: Emilio Justiniano Cárcamo Troconis Codirector: Carolina Vargas Porras

Abstract

Title: Multiplatform information system for the accompaniment of parents in the postpartum process.

Author: Valentina Escobar Bueno, Mario Hernán Vallejo Huertas, Jesús Sebastián Vargas Martínez.

Keywords: Information system, postpartum, support, interaction.

Description: This paper presents the prototype of an information system that serves as a support and facilitator of interaction between parents in their maternity process and health personnel. The objective is to provide a tool that facilitates access to nursing support for new mothers who, due to lack of experience and preparation, develop various problems. This in order to minimize the number and impact of these problems.

The scope of development of the information system consists of adapting it to the requirements of four agents: the super administrator, the administrator, nursing staff and people in the maternity stage, all this under the criteria of analysis, adaptation, performance and security. The agents are the ones who have first-hand evidence of how the system works, since they are the ones who interact with it directly. The achievement of the aforementioned process is based on the knowledge acquired in the development of the career and the information provided by the agents involved together with the use of technological tools.

* Trabajo de Grado

** Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería de Sistemas e Informática.
Director: Emilio Justiniano Cárcamo Troconis Codirector: Carolina Vargas Porras

Introducción

Una de las etapas más importantes de la vida de muchas personas es la paternidad, donde se prueban como seres humanos en la crianza de una nueva vida. Como en toda actividad, la falta de apoyo, conocimientos y experiencia pueden conllevar a consecuencias no deseadas que afectan directamente la vida de padres e hijos por igual. Para evitar esto, diferentes entidades proveen asistencia médica y psicológica que buscan que la etapa transcurra sin mayores complicaciones. Sin embargo, el acceso a dichos servicios se enfrenta a diferentes obstáculos como el desconocimiento, la distancia o la disponibilidad de estos.

Este proyecto propone el desarrollo de un prototipo de sistema de información. El cual daría acceso al usuario a material de apoyo que brinde una base de conocimiento sobre el tiempo del puerperio, el cual está definido como la etapa después del parto hasta que se regula su periodo menstrual, dada esta información cabe resaltar que es un periodo de gran importancia en la vida de la mujer, dado que se generan cambios fisiológicos y psicológicos en la búsqueda de la adaptación de su rol de madre.

Adicionalmente, se busca que sirva como medio para una interacción más ágil entre las personas en su etapa de paternidad y los servicios de salud mediante el registro, seguimiento y diagnóstico de situaciones que puedan poner en riesgo la integridad de padres y bebés, y por ende minimizar las complicaciones en el periodo del postparto y facilitar el acceso a la información mediante una plataforma en la que sea fácil orientarse en caso de desconocimiento o dudas sobre temáticas relacionadas a la paternidad.

1. Generalidades del proyecto

1.1 Planteamiento y justificación del problema

La maternidad y la paternidad son una etapa que muchas personas pasan a lo largo de sus vidas, pero no siempre se sabe cómo actuar frente a ello; cómo poder llevar un buen proceso o simplemente sentir un apoyo que facilite este nuevo rol, pues son muchos retos y barreras que hay que enfrentar. A su vez, existe una gran presión en los padres sobre cómo actuar y qué hacer ante situaciones nuevas, como el embarazo, el parto y el cuidado del recién nacido, lo que puede llegar a ocasionar sobrecarga en aspectos sociales, familiares, de pareja y del rol de padres, puesto que como lo menciona Fernández (2013), la maternidad ocasiona cambios en la vida, en los sentimientos y en las percepciones, determinando como estresantes las nuevas funciones que esto requiere, lo anterior argumenta y hace evidente la necesidad de recurrir a un apoyo externo que brinde seguridad, confianza, compañía y conocimiento a enfrentar esta nueva etapa.

En Colombia, como lo menciona Sagar (2012) es común ver que se aplica la teoría de Leininger, que aborda el conocimiento cultural relacionado con el puerperio, dando lugar a creencias y prácticas arraigadas sin evaluar su eficacia real. Tras la hospitalización, se reduce el contacto con profesionales de enfermería, lo que implica una falta de seguimiento y planificación del cuidado. Dada la fragilidad de este periodo, es esencial contar con un apoyo claro para afrontar situaciones que, según Vargas (2020), pueden poner en riesgo la vida de la madre o del recién nacido debido a la falta de conocimiento o a circunstancias secundarias.

En la sociedad de la información, la integración de las Tecnologías de la Información y Comunicación (TIC) ha simplificado numerosas tareas, volviéndose más eficientes y accesibles. Este avance motiva la implementación de un sistema de información que facilite la comunicación pos-hospitalaria entre padres y profesionales de enfermería. La premisa es que

un contacto cercano entre ambas partes podría reducir la incidencia de situaciones no reportadas al personal de salud, disminuyendo así las complicaciones durante el puerperio.

Actualmente, las aplicaciones diseñadas para respaldar a las madres en el posparto, como BabyCenter, Pregnantis, entre otras, se centran predominantemente en abordar los cambios físicos y las necesidades tangibles del bebé, así como las rutinas asociadas con esta nueva etapa. Sin embargo, al revisar sus descripciones, se evidencia la falta de atención a los cambios en la salud mental de los padres y a los desafíos emocionales que implica este proceso. Es crucial abordar estos aspectos a tiempo. Por lo tanto, surge la necesidad de crear una herramienta que abarque el cuidado físico y mental, proporcionando orientación integral para la salud de los padres primerizos. Este proyecto se concibe como un complemento que integra todas estas dimensiones.

El proyecto propuesto tiene como objetivo proporcionar un apoyo eficiente y rápido a los padres primerizos, afrontando las situaciones diarias de este nuevo rol. La aplicación propuesta brindaría una base de conocimientos fácilmente accesible para buscar información que prevenga, apoye o solucione posibles problemas. Además, mediante reportes de situaciones, mejoraría el acompañamiento de los servicios de salud, agilizando diagnósticos y tratamientos. Esta herramienta se diseñaría pensando en la accesibilidad para los nuevos padres que, debido a las obligaciones cotidianas, encuentran dificultades para acceder a apoyo profesional en persona.

Además, se busca proporcionar información precisa y respaldada por profesionales de la salud para contrarrestar el acceso a información incorrecta que podría tener efectos adversos a los objetivos perseguidos por esta herramienta. En este contexto, el análisis del estado del arte revela la ausencia de herramientas que aborden integralmente las necesidades específicas de los padres primerizos, lo que justifica la realización de este proyecto y el desarrollo de un nuevo software que colme estas carencias en las soluciones actuales.

SISTEMA DE INFORMACIÓN AMACOM

1.2 Alcance del proyecto

La finalidad deseada con este proyecto es el de facilitar la interacción entre madres y padres primerizos y el personal de salud en pro de mejorar la documentación y atención temprana de posibles problemáticas surgidas en este periodo.

A continuación, se mencionan los ítems que se incluyen en el alcance del proyecto:

- Diseño de aplicativo web y sistema de información destinado al personal de salud para administrar y gestionar la interacción con el mismo.
- Diseño de aplicativo móvil y sistema de información, destinado a interactuar directamente con madres y padres.

2. Objetivos

2.1 Objetivo General

Desarrollar un prototipo de sistema de información web y móvil para el acompañamiento de padres en el proceso de posparto.

2.2 Objetivos específicos

- Definir los requerimientos del proceso de enfermería asociados a la gestión de situaciones de salud en la etapa de puerperio y la interacción entre los padres y profesionales de enfermería.
- Realizar el diseño de una arquitectura y prototipado de sistema de información para el acompañamiento a padres en el proceso de puerperio.
- Desarrollar un prototipo de sistema de información en sus componentes backend, frontend y móvil, para apoyar a los padres en el proceso de posparto.
- Realizar un plan de pruebas que verifique el apropiado funcionamiento del prototipo de software.

3. Marco de Referencia

3.1 Marco Contextual

Con el fin de contextualizar el marco contextual en el que se desarrolló este proyecto, definiendo con brevedad conceptos claves para comprender el proceso de enfermería y el contexto de la problemática en el acompañamiento a padres en el proceso de puerperio.

3.1.1 NANDA, NIC y NOC

NANDA, NIC y NOC son sistemas de clasificación empleados en el ámbito de la enfermería con el propósito de estandarizar la terminología y la documentación vinculadas a la atención de los pacientes. A continuación, se abordan de forma breve:

- **NANDA International (NANDA-I):**

La Asociación Internacional de Diagnósticos de Enfermería de América del Norte (NANDA-I). Establece un lenguaje uniforme para describir problemas de salud y situaciones que las enfermeras enfrentan en su práctica. Un diagnóstico NANDA podría ser “Dolor agudo” o “Riesgo de infección”.

- **NIC (Clasificación de Intervenciones de Enfermería):**

La Clasificación de Intervenciones de Enfermería (NIC) ofrece un conjunto estandarizado de acciones de enfermería que abordan los problemas de salud identificados mediante diagnósticos NANDA. Una intervención NIC podría ser “Administración de analgésicos” o “Educación sobre el manejo del dolor”.

- **NOC (Clasificación de Resultados de Enfermería):**

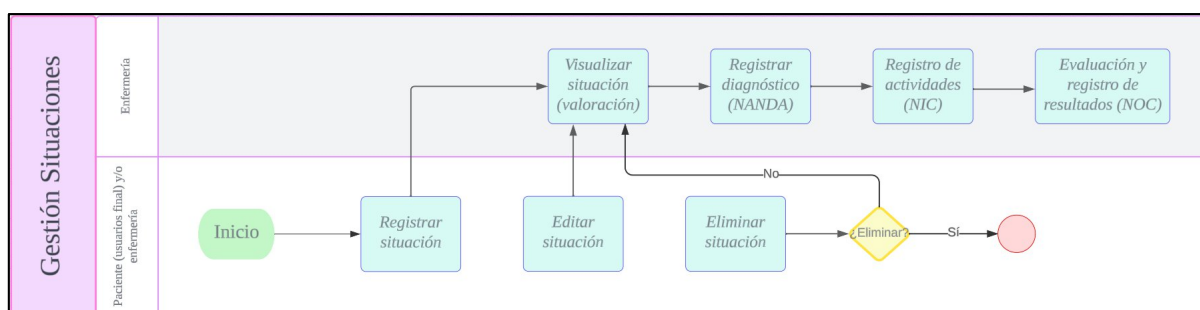
La Clasificación de Resultados de Enfermería (NOC) proporciona un conjunto estandarizado de resultados que se utilizan para evaluar la respuesta del paciente a las intervenciones de enfermería. Un resultado NOC podría ser “Alivio del dolor” o “Mejora de la movilidad”.

3.1.2 Proceso de enfermería

Es crucial destacar que, en este proyecto, la relación entre paciente y personal de enfermería constituye un pilar fundamental, esto se debe a la importancia de un seguimiento continuo y detallado de la madre durante el proceso posparto. Al facilitar este seguimiento, no solo permite al personal de enfermería mantener un control eficiente sobre el progreso de la paciente, sino que también brinda a la madre una plataforma en la que puede expresar y documentar su experiencia durante el puerperio. Este enfoque bidireccional asegura una atención más personalizada y efectiva, resaltando la importancia de una comunicación fluida y empática entre la paciente y el personal de enfermería.

Figura 1

Flujo gestión de situaciones usuario final y enfermería



De esta manera, el módulo que plasma lo anteriormente descrito es el de gestión de situaciones (figura 1), en este el sistema permite que la madre (usuario final) documente diversas situaciones experimentadas, donde tiene la capacidad de detallar la situación específica que enfrenta, identificar el tema relacionado, evaluar el nivel de impacto, y expresar las emociones y percepciones experimentadas. Además, puede registrar su reacción inicial, sus acciones inmediatas, y cualquier señal de alerta observada tanto en su bebé como en sí misma.

Por otro lado, el personal de enfermería, al examinar las situaciones registradas por la paciente, está en posición de elaborar uno o varios diagnósticos conforme al sistema NANDA, este proceso implica una evaluación detallada de los hechos descritos por la madre, permitiendo a enfermería aplicar su juicio clínico y conocimientos para identificar problemas o patrones de

salud y así poder establecer diagnósticos precisos y pertinentes. El siguiente paso implica la planificación e implementación de estrategias y actividades diseñadas para abordar de manera efectiva los diagnósticos identificados utilizando la Clasificación de Intervenciones de Enfermería (NIC), y poder, así como paso final aplicar la Clasificación de Resultados de Enfermería (NOC), la cual se utiliza para evaluar y medir los resultados de estas intervenciones, logrando resultados óptimos y adaptados a las necesidades de cada madre en su proceso de posparto.

3.2 Marco Tecnológico

En el desarrollo e implementación del software se hace uso de tecnologías ligadas al área de desarrollo web y móvil. Con ello se busca entre otras cosas una distribución eficiente de tareas de desarrollo de fácil desarrollo e integración y garantizar una buena calidad.

3.2.1 *Tecnologías y Lenguajes*

A continuación, se habla sobre los lenguajes, tecnologías y librerías a usar en el desarrollo del software.

3.2.1.1 JavaScript. JavaScript (JS) es un lenguaje de programación ligero, siendo el único lenguaje de programación que entienden los navegadores web debido a que no es necesario compilar para ser ejecutado (Engiluz, 2009). Se usa dentro de una plataforma web debido a que dota al lenguaje de hipertexto de la web de funcionalidades manejadas como eventos.

3.2.1.2 NodeJs. Es un entorno de tiempo de ejecución del lenguaje de programación JavaScript orientado a eventos asíncronos que añade características que hacen que pueda manipular el sistema de archivos, crear bases de datos directamente y crear servidores web (Mead, 2018).

3.2.1.3 Angular. Se trata de una framework para la creación de aplicativos frontend web con un estilo SPA (Single Page Application), es decir una aplicación web que está contenida en un solo archivo, en un HTML.

El desarrollo de esta se hace a través del lenguaje de programación Typescript (aunque también se puede con Javascript) que luego es compilado a JavaScript y desplegado.

3.2.1.4 Git. Git es un sistema de control de versiones distribuido, disponible a través de una licencia de software gratuita. Brinda a los desarrolladores una copia de repositorio del código y muchas y útiles maneras de manejar diversas versiones en su contexto. La capacidad de asociar un repositorio local con otros remotos permite desarrollar una gran variedad de flujos de trabajo (Spinellis, 2012).

3.2.1.5 Docker. Docker es una tecnología de virtualización basada en contenedores que son algo así como máquinas virtuales (VM) de bajo peso. Este provee al desarrollador de un flujo de trabajo que ayuda en la construcción y distribución de software haciendo que se construyan aplicaciones en contenedores (Anderson, 2015). Como consecuencia, permite separar las aplicaciones de su infraestructura permitiendo generar y compartir software rápidamente.

3.2.1.6 MySQL. Consiste en un sistema de gestión de bases de datos (DBMS) de tipo relacional, MySQL es de código abierto, respaldado por Oracle. Este DBMS de tipo relacional al igual que todos los de su tipo hablan un lenguaje común SQL con el que es posible definir, manipular y consultar datos.

3.2.1.7 Flutter. Flutter es un framework de código abierto desarrollado en base a Dart y mantenido por Google. Los desarrolladores de front-end y pila completa utilizan Flutter para crear una interfaz de usuario (IU) de aplicación para varias plataformas con un único código base. El framework trabaja haciendo que el UI se componga de widgets - similares a componentes- que describen el cómo deben mirarse en base a su configuración y estado.

3.2.1.9 Java. Java es un lenguaje de programación orientado a objetos y una plataforma informática lanzada por primera vez por Sun Microsystems en 1995. Su uso en internet hizo que se propulsara debido a que simplifican la programación y dio a manejo a unos de los mayores problemas de la web: la portabilidad y seguridad (Schildt, 2022).

3.3 Antecedentes

La era digital en el campo de la salud ha venido cambiando y construyendo nuevos métodos de prestación de este servicio, los cuales tienen el objetivo de optimizar las asistencias médicas y/o de enfermería centradas en brindar una atención más rápida, de mejor calidad, y más accesible. Conviene tomar de ejemplo y guía las plataformas relacionadas existentes en pro de la identificación de factores que beneficien el desarrollo del software.

- **Amacompri**

En un estudio realizado por Vargas, Cárcamo, Ferré y Molina (2021) titulado “*Sistema de información Amacompri para el seguimiento de enfermería posparto*” se desarrolló un software Amacompri el cual brinda seguimiento de la etapa del posparto, que se basa en la teoría de Mercer de convertirse en madre y analiza el entorno vital: La familia y los amigos de la madre primeriza. Se diseñó bajo el marco de trabajo para desarrollo ágil denominado SCRUM, con tecnologías web Java y base de datos PostgreSQL.

- **BabyCenter**

Es una aplicación que brinda acompañamiento, apoyo e información a los padres en las diferentes etapas como el embarazo, el posparto y hasta el crecimiento del bebé. Fue creada en el año 2006 para el sistema operativo IOS y en el 2008 se creó una versión para dispositivos móviles con sistema operativo Android.

- **Mi bebé y yo**

Se trata de un portal web que ofrece a las madres información importante a través de artículos que facilitan el proceso del embarazo y el cuidado del bebé. Funciona como una comunidad donde las personas comentan sus opiniones y así se genera un intercambio de experiencias e información importante. Además, cada mes se encargan de editar una revista, la cual es otorgada a las madres para los primeros meses de vida del bebé.

- **Pregnantis**

Esta aplicación se basa en un consultorio médico en línea enfocado en las etapas de embarazo y posparto, adicionalmente, cuenta con una comunidad en línea para comunicarse entre usuarios. Esta aplicación está disponible para los sistemas IOS y Android, fue creada en el año 2015 por un grupo de desarrolladores en Colombia y se ha ido expandiendo por América Latina.

Como se puede observar, existen varias aplicaciones que brindan acompañamiento a los padres primerizos y en su etapa de embarazo y posparto, sin embargo revisando sus descripciones se puede observar que todas van enfocadas a los cambios físicos y tangibles, a las necesidades del bebe y las diferentes rutinas que esta nueva etapa puede ocasionar, pero no menciona de los cambios o ayudas en la salud mental de los padres, en los diferentes desafíos emocionales que es enfrentarse a este nuevo proceso y lo importante que es tratar este tipo de acontecimientos a tiempo, por ende se hace necesario crear una nueva herramienta que involucre el cuidado físico, y mental y que de orientaciones en todo lo que concierne a la salud de los padres primerizos.

4. Fase de análisis

4.1 Análisis de requerimientos

La aplicación web permite el registro ilimitado de usuarios con un rol específico asignado por el administrador. Para ello, se cuenta con tres roles de usuario, los cuales son:

- Administrador de la plataforma (Administrador)
- Personal de enfermería (Enfermería)
- Padres y madres (Usuario Final)

La aplicación móvil permite el registro ilimitado de usuarios con un formulario de creación de usuario, asignado por defecto el rol de usuario “Usuario Final”. A continuación, se presentan los requerimientos funcionales y no funcionales del sistema de información realizado, este separado por actores, tal y como fueron descritos anteriormente.

4.1.1 Requerimientos funcionales para el usuario final

En la siguiente tabla se enumeran los requerimientos funcionales del sistema destinados a la operación del usuario final. La agrupación de requerimientos se realiza en base a los casos de uso donde se describen de forma breve y se hace referencia a su vista de usuario. Se muestra además la fase en la que se dio origen al requerimiento junto con los ajustes realizados al mismo a lo largo del desarrollo del proyecto.

Tabla 1

Requerimientos funcionales del usuario final.

Nº	Requerimiento	Descripción El software permite
1	Inicio	Visualizar la página principal.
2	Registrar	Registrar usuario con datos personales, correo y contraseña.
3	Acceder a la plataforma	Iniciar sesión en la aplicación móvil con correo y contraseña. (Ver Figura 40)
4	Recuperar contraseña	Realizar la restauración de la contraseña mediante el correo electrónico registrado. (Ver Figura 40)
5	Visualizar y modificar datos personales	Visualizar y modificar nombres, apellidos, número y tipo de documento, dirección, género, estado civil, ocupación, fecha de nacimiento. (Ver Figura 41)

SISTEMA DE INFORMACIÓN AMACOM

6	Gestionar bitácoras de usuario	Visualizar, registrar, editar y eliminar bitácoras relacionadas al usuario, las cuales contienen título, fecha y descripción. (Ver Figuras 22 y 43)
7	Gestionar situaciones usuario	Listar las situaciones parametrizadas por el usuario, así mismo permite registrar, editar y eliminar situaciones con un tema, descripción, primer pensamiento, comportamiento, grado de afectación y sentimientos. (Ver Figuras 15-18 y 42)
8	Consultar diagnósticos	Al consultar cada situación se puede acceder al listado y a la información de cada diagnóstico que han registrado los usuarios enfermería y/o administrador.
9	Listar material de apoyo	Listar el material de apoyo agrupado por temas y subtemas. (Ver Figura 45)
10	Obtener logros	Se presenta al usuario un cuestionario que permite evaluar su nivel en los diferentes logros asociados a temas y subtemas, a partir de las diferentes respuestas insertadas se asigna un nivel y una calificación a cada tema. (Ver Figura 46)
11	Calendario usuario	Visualizar un calendario con los eventos creados por el usuario en el día y hora seleccionados, en él también se registra, edita y elimina eventos con un tipo de evento, título, descripción, fecha inicial y final. (Ver Figuras 23 y 44)

4.1.2 Requerimientos funcionales para el usuario enfermería

En la siguiente tabla se enumeran los requerimientos funcionales del sistema destinados a la operación de enfermería.

Tabla 2

Requerimientos funcionales del usuario de enfermería.

N°	Requerimiento	Descripción El software permite
12	Visualizar y modificar datos personales	Visualizar y modificar nombres, apellidos, número y tipo de documento, dirección, género, estado civil, ocupación, fecha de nacimiento e imagen.
13	Gestionar personas	Visualizar el listado de personas registradas en el sistema, así mismo registrar, editar y eliminar persona con nombre, apellido, tipo de documento, documento, dirección, estado civil, ocupación, firma, imagen. (Ver Figura 14)
14	Gestionar situaciones persona	Al consultar cada persona, se listan las situaciones parametrizadas por el usuario, así mismo permite registrar, editar y eliminar situaciones con un tema, descripción, primer pensamiento, comportamiento, grado de afectación y sentimientos. (Ver Figuras 15-18)
15	Gestionar Diagnósticos de situaciones persona	Visualizar el listado de situaciones registradas por una persona, en donde en su detalle es posible registrar, editar y eliminar diagnósticos. (Ver Figuras 19-21)
16	Gestionar bitácoras persona	Así mismo en el detalle de la persona, se puede visualizar, registrar, editar y eliminar bitácoras relacionadas al usuario, las cuales contienen título, fecha y descripción.
17	Calendario persona	Visualizar calendario con los eventos programados por una

		persona. (Ver Figura 22)
18	Calendario usuario	Visualizar un calendario con los eventos creados por el usuario en el día y hora seleccionados, en él también se registra, edita y elimina eventos con un tipo de evento, título, descripción, fecha inicial y final. (Ver Figura 23)

4.1.3 Requerimientos funcionales para el usuario administrador

En la siguiente tabla se enumeran los requerimientos funcionales del sistema destinados a la operación de administrador, heredando los requerimientos establecidos por la operación de enfermería.

Tabla 3

Requerimientos funcionales del usuario administrador.

Nº	Requerimiento	Descripción
		El software permite
19	Visualizar y modificar datos personales	Visualizar y modificar nombres, apellidos, número y tipo de documento, dirección, género, estado civil, ocupación, fecha de nacimiento e imagen.
20	Gestionar servicios	Visualizar el listado de servicios registrados en el sistema, así mismo permite registrar, editar y eliminar servicios con nombre y descripción. (Ver Figura 35)
21	Gestionar tipos de institución	Visualizar el listado de tipos de institución registradas en el sistema, así mismo permite registrar, editar y eliminar tipos de institución con nombre y descripción. (Ver Figura 34)

SISTEMA DE INFORMACIÓN AMACOM

22	Gestionar instituciones	Visualizar el listado de instituciones registradas en el sistema, así mismo permite registrar, editar y eliminar instituciones con nombre, tipo de institución, descripción, servicios y personal asociados. (Ver Figura 36)
23	Gestionar temas	Visualizar el listado de temas registrados en el sistema, así mismo permite registrar, editar y eliminar tema con nombre, indicador de validez y tema padre asociado. (Ver Figura 31)
24	Gestionar materiales de apoyo	Al entrar en el detalle de temas, se visualiza el listado de materiales de apoyo registrados en el sistema, así mismo permite registrar, editar y eliminar material de apoyo con nombre, url, descripción y tema asociado. (Ver Figura 33)
25	Gestionar logros	Al entrar en el detalle de temas, se visualiza el listado de logros registrados en el sistema, así mismo permite registrar, editar y eliminar logro con nombre. (Ver Figura 32)
26	Gestionar signos de alarma	Visualizar el listado de signos de alarma registrados en el sistema, así mismo permite registrar, editar y eliminar signo de alarma con nombre, tipo y descripción. (Ver Figura 30)
27	Gestionar tipos de documento	Visualizar el listado de tipos de documentos registrados en el sistema, así mismo permite registrar, editar y eliminar tipo de documento con nombre y código. (Ver Figura 29)
28	Gestionar géneros	Visualizar el listado de géneros registrados en el sistema, así mismo permite registrar, editar y eliminar género con nombre. (Ver Figura 28)

SISTEMA DE INFORMACIÓN AMACOM

29	Gestionar usuarios	Visualizar el listado de usuarios registrados en el sistema, así mismo permite registrar, editar y eliminar usuario con nombre, correo electrónico, contraseña, rol y persona asociada. (Ver Figura 27)
30	Gestionar personas	Visualizar el listado de personas registradas en el sistema, así mismo registrar, editar y eliminar persona con nombre, apellido, tipo de documento, documento, dirección, estado civil, ocupación, firma, imagen. (Ver Figura 14)
31	Gestionar situaciones persona	Al consultar cada persona, se listan las situaciones parametrizadas por el usuario, así mismo permite registrar, editar y eliminar situaciones con un tema, descripción, primer pensamiento, comportamiento, grado de afectación y sentimientos. (Ver Figuras 15-18)
32	Gestionar Diagnósticos de situaciones persona	Visualizar el listado de situaciones registradas por una persona, en donde en su detalle es posible registrar, editar y eliminar diagnósticos. (Ver Figuras 19-21)
33	Gestionar bitácoras persona	Así mismo en el detalle de la persona, se puede visualizar, registrar, editar y eliminar bitácoras relacionadas al usuario, las cuales contienen título, fecha y descripción. (Ver Figura 22)
34	Calendario persona	Visualizar calendario con los eventos programados por una persona.
35	Calendario usuario	Visualizar un calendario con los eventos creados por el usuario

		en el día y hora seleccionados, en él también se registra, edita y elimina eventos con un tipo de evento, título, descripción, fecha inicial y final. (Ver Figura 23)
--	--	---

4.1.4 Requerimientos no funcionales

Tabla 4

Requerimientos no funcionales

Requerimiento	Descripción
RNF1 Seguridad	La aplicación debe proteger las contraseñas de los usuarios utilizando el estándar SHA-256
RNF2 Compatibilidad	La aplicación móvil debe ser compatible con los sistemas Android y iOS. La aplicación web debe tener soporte multiplataforma.
RNF3 Rendimiento	La aplicación debe ser capaz de soportar una carga simultánea de 100 usuarios simultáneos sin sufrir interrupciones durante 20 minutos.
RNF4 Software	La aplicación debe ser realizada utilizando la arquitectura de microservicios.
RNF5 Disponibilidad	La aplicación debe ofrecer sus servicios y funcionalidades del 99.9% de las veces que un usuario lo solicite.
RNF6 Seguridad	La aplicación debe garantizar un adecuado tratamiento de los datos personales de los usuarios basado en las regulaciones colombianas.
RNF7 Mantenibilidad	La aplicación debe seguir buenas prácticas de programación para facilitar su mantenibilidad y escalabilidad.

4.2 Casos de uso y escenarios

Tabla 5

CU. N°1

Nombre del CU	Autenticación
Nombre	<i>CU N°1</i>
Requerimiento	1, 2, 3 y 4
Actores	Admin, Enfermería y Usuario Final
Descripción	Los actores pueden registrarse y loguearse en el sistema. (Ver Figuras 13 y 40)
Precondiciones	N.A.
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario presiona en “Registrarse” 2. El prototipo muestra una pantalla con el formulario registrarse. 3. El usuario llena los campos correspondientes y necesarios y pulsa "Registrarse". 6. Si el formulario fue llenado de manera válida se crea exitosamente el registro y redirecciona al inicio. 7. El usuario puede loguearse llenando el formulario con las credenciales ya previamente registradas y presionar en “Ingresar”. 8. Si el formulario fue llenado de manera válida ingresa exitosamente el usuario.

Tabla 6

CU. N°2

Nombre del CU	Gestión de usuarios
Nombre	CUN°2
Requerimiento	29
Actores	Admin
Descripción	El admin puede visualizar los usuarios existentes, también adicionarlos, editarlos y eliminarlos. (Ver Figuras 24-27)
Precondiciones	Tener rol de admin para tener los permisos necesarios
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario presiona en el menú "Usuarios" 2. El prototipo muestra una vista con el listado de usuarios existentes y puede filtrar la información por nombre de usuario, rol o estado. 3. El usuario puede seleccionar la opción "Adicionar" 4. El prototipo muestra una pantalla con el formulario para la creación de un usuario. 5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar". 6. Si el formulario fue llenado de manera válida se crea exitosamente el usuario y redirige a la grilla principal donde se encontrará actualizada. 7. El usuario puede seleccionar la opción "Editar" 8. El prototipo muestra una pantalla con el formulario para la edición del usuario seleccionado. 9. El usuario puede modificar cualquier campo y pulsar "Guardar".

	<p>10. Si el formulario fue llenado de manera válida se edita exitosamente el usuario y redirige a la grilla principal donde se encontrará actualizada.</p> <p>11. El usuario puede seleccionar la opción "Eliminar"</p> <p>12. El prototipo muestra un modal de confirmación de la acción.</p> <p>13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal.</p> <p>14. El usuario puede seleccionar la opción "Consultar"</p> <p>15. El prototipo muestra un modal con la información del usuario seleccionado.</p> <p>16. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.</p>
--	---

Tabla 7

CUN°3

Nombre del CU	Gestión de personas
Nombre	<i>CUN°3</i>
Requerimiento	30
Actores	Admin
Descripción	El admin puede visualizar las personas existentes, también adicionar, editar y eliminar. (Ver Figura 14)
Precondiciones	Tener rol de admin para así tener los permisos necesarios
Poscondiciones	N.A.

<p>Flujo del proceso</p>	<ol style="list-style-type: none"> 1. El usuario presiona en el menú "Personas" 2. El prototipo muestra una vista con el listado de personas existentes y poder filtrar la información por nombre, apellido o estado. 3. El usuario puede seleccionar la opción "Adicionar". 4. El prototipo muestra una pantalla con el formulario para la creación de una persona. 5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar". 6. Si el formulario fue llenado de manera válida se crea exitosamente la persona y redirige a la grilla principal donde se encontrará actualizada. 7. El usuario puede seleccionar la opción "Editar" 8. El prototipo muestra una pantalla con el formulario para la edición de la persona seleccionada. 9. El usuario puede modificar cualquier campo y pulsar "Guardar". 10. Si el formulario fue llenado de manera válida se edita exitosamente la persona y se redirige a la grilla principal donde se encontrará actualizada. 11. El usuario puede seleccionar la opción "Eliminar" 12. El prototipo muestra un modal de confirmación de la acción. 13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal. 15. El usuario puede seleccionar la opción "Consultar" 16. El prototipo muestra una pantalla con la información de la persona
--------------------------	---

	<p>seleccionada.</p> <p>17. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.</p>
--	---

Tabla 8

CUN°4

Nombre del CU	Gestión de géneros
Nombre	CU <i>CUN°4</i>
Requerimiento	28
Actores	Admin
Descripción	El admin puede visualizar los géneros existentes, también adicionarlos, editarlos y eliminarlos. (Ver Figura 28)
Precondiciones	Tener rol de admin para así tener los permisos necesarios
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario presiona en el menú "Géneros" 2. El prototipo muestra una vista con el listado de géneros existentes en el sistema. 3. El usuario puede seleccionar la opción "Adicionar" 4. El prototipo muestra un modal con el formulario para la creación de un género. 5. El usuario llena el campo correspondiente y pulsa "Guardar". 6. Si el formulario fue llenado de manera válida se crea exitosamente el género y se visualizará en la grilla principal donde se encontrará

	<p>actualizada.</p> <p>7. El usuario puede seleccionar la opción "Editar"</p> <p>8. El prototipo muestra un modal con el formulario para la edición del género seleccionado.</p> <p>9. El usuario puede modificar el campo y pulsar "Guardar".</p> <p>10. Si el formulario fue llenado de manera válida se edita exitosamente el género y en la grilla principal se encontrará actualizada.</p> <p>11. El usuario puede seleccionar la opción "Eliminar"</p> <p>12. El prototipo muestra un modal de confirmación de la acción.</p> <p>13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal.</p> <p>14. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.</p>
--	--

Tabla 9

CUN°5

Nombre del CU	Gestión tipos de documento
Nombre	<i>CUN°5</i>
Requerimiento	27
Actores	Admin
Descripción	El admin puede visualizar los tipos de documento existentes, también adicionarlos, editarlos y eliminarlos. (Ver Figura 29)
Precondiciones	Tener rol de admin para así tener los permisos necesarios.

Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario presiona en el menú "Tipos de documento" 2. El prototipo muestra una vista con el listado de tipos de documento existentes en el sistema. 3. El usuario puede seleccionar la opción "Adicionar" 4. El prototipo muestra un modal con el formulario para la creación de un tipo de documento. 5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar". 6. Si el formulario fue llenado de manera válida se crea exitosamente el tipo de documento y se visualizará en la grilla principal donde se encontrará actualizada. 7. El usuario puede seleccionar la opción "Editar" 8. El prototipo muestra un modal con el formulario para la edición del tipo de documento seleccionado. 9. El usuario puede modificar el campo y pulsar "Guardar". 10. Si el formulario fue llenado de manera válida se edita exitosamente el tipo de documento y en la grilla principal se actualizará. 11. El usuario puede seleccionar la opción "Eliminar" 12. El prototipo muestra un modo de confirmación de la acción. 13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal. 14. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla

	principal sin ninguna modificación.
--	-------------------------------------

Tabla 10

CUN°6

Nombre del CU	Gestión signos de alarma
Nombre	<i>CUN°6</i>
Requerimiento	26
Actores	Admin
Descripción	El admin puede visualizar los signos de alarma existentes, también adicionarlos, editarlos y eliminarlos. (Ver Figura 30)
Precondiciones	Tener rol de admin para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario presiona en el menú "Signos de alarma" 2. El prototipo muestra una vista con el listado de los signos de alarma existentes en el sistema. 3. El usuario puede seleccionar la opción "Adicionar" 4. El prototipo muestra un modal con el formulario para la creación de un signo de alarma. 5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar". 6. Si el formulario fue completado de manera válida se crea exitosamente el signo de alarma y se visualizará en la grilla principal donde se encontrará actualizada. 7. El usuario puede seleccionar la opción "Editar"

	<p>8. El prototipo muestra un modal con el formulario para la edición del signo de alarma seleccionado.</p> <p>9. El usuario puede modificar el campo y pulsar "Guardar".</p> <p>10. Si el formulario fue llenado de manera válida se edita exitosamente el signo de alarma y en la grilla principal se actualizará.</p> <p>11. El usuario puede seleccionar la opción "Eliminar"</p> <p>12. El prototipo muestra un modal de confirmación de la acción.</p> <p>13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal.</p> <p>14. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.</p>
--	---

Tabla 11

CUN°7

Nombre del CU	Gestión de temas y subtemas
Nombre	<i>CUN°7</i>
Requerimiento	23
Actores	Admin
Descripción	El admin puede visualizar los temas y subtemas existentes, también adicionarlos, editarlos y eliminarlos. (Ver Figura 31)
Precondiciones	Tener rol de admin para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	1. El usuario presiona en el menú "Temas y subtemas"

	<p>2. El prototipo muestra una vista con el listado de temas existentes en el sistema.</p> <p>3. El usuario puede seleccionar la opción "Adicionar".</p> <p>4. El prototipo muestra una pantalla con el formulario para la creación de un tema.</p> <p>5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar".</p> <p>6. Si el formulario fue llenado de manera válida se crea exitosamente el tema y redirige a la grilla principal donde se encontrará actualizada.</p> <p>7. El usuario puede seleccionar la opción "Editar".</p> <p>8. El prototipo muestra una pantalla con el formulario para la edición del tema seleccionado.</p> <p>9. El usuario puede modificar cualquier campo y pulsar "Guardar".</p> <p>10. Si el formulario fue llenado de manera válida se edita exitosamente el tema y se redirige a la grilla principal donde se encontrará actualizada.</p> <p>11. El usuario puede seleccionar la opción "Eliminar"</p> <p>12. El prototipo muestra un modal de confirmación de la acción.</p> <p>13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal.</p> <p>15. El usuario puede seleccionar la opción "Consultar"</p> <p>16. El prototipo muestra una pantalla con la información del tema seleccionado, además se muestra una tabla con los subtemas asociados, la cual tiene la opción "Editar" en cada registro.</p> <p>17. El usuario puede seleccionar la opción "Editar".</p>
--	--

	<p>18. El prototipo muestra una pantalla con el formulario para la edición del subtema seleccionado.</p> <p>19. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.</p>
--	---

Tabla 12

CUN°8

Nombre del CU	Gestión de logros
Nombre	<i>CUN°8</i>
Requerimiento	25
Actores	Admin
Descripción	El admin puede visualizar los logros existentes por tema, también adicionarlos, editarlos y eliminarlos. (Ver Figura 32)
Precondiciones	Tener rol de admin para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario se dirige a la pantalla "Consultar tema" 2. El prototipo muestra en esa vista una sección con el listado de los logros existentes en el sistema y asociados a dicho tema. 3. El usuario puede seleccionar la opción "Adicionar" 4. El prototipo muestra un modal con el formulario para la creación de logro. 5. El usuario llena el campo correspondiente y necesario y pulsa "Guardar".

	<p>6. Si el formulario fue completado de manera válida se crea exitosamente el logro y se visualizará en la grilla principal donde se encontrará actualizada.</p> <p>7. El usuario puede seleccionar la opción "Editar"</p> <p>8. El prototipo muestra un modal con el formulario para la edición del logro seleccionado.</p> <p>9. El usuario puede modificar el campo y pulsar "Guardar".</p> <p>10. Si el formulario fue llenado de manera válida se edita exitosamente el logro y en la grilla principal se actualizará.</p> <p>11. El usuario puede seleccionar la opción "Eliminar"</p> <p>12. El prototipo muestra un modal de confirmación de la acción.</p> <p>13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal.</p> <p>14. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.</p>
--	---

Tabla 13

CUN° 9

Nombre del CU	Gestión material de apoyo
Nombre	<i>CUN°9</i>
Requerimiento	24
Actores	Admin
Descripción	El admin puede visualizar los materiales de apoyo existentes por tema, también adicionarlos, editarlos y eliminarlos. (Ver Figura 33)

SISTEMA DE INFORMACIÓN AMACOM

Precondiciones	Tener rol de admin para así tener los permisos necesarios.
Poscondiciones	<ol style="list-style-type: none"> 1. El usuario se dirige a la pantalla "Consultar tema" 2. El prototipo muestra en esa vista una sección con el listado de los materiales de apoyo existentes en el sistema y asociados a dicho tema. 3. El usuario puede seleccionar la opción "Adicionar" 4. El prototipo muestra un modal con el formulario para la creación del material de apoyo. 5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar". 6. Si el formulario fue completado de manera válida se crea exitosamente el material de apoyo y se visualizará en la grilla principal donde se encontrará actualizada. 7. El usuario puede seleccionar la opción "Editar" 8. El prototipo muestra un modal con el formulario para la edición del material de apoyo seleccionado. 9. El usuario puede modificar los campos y pulsar "Guardar". 10. Si el formulario fue completado de manera válida se edita exitosamente el material de apoyo y en la grilla principal se actualizará. 11. El usuario puede seleccionar la opción "Eliminar" 12. El prototipo muestra un modal de confirmación de la acción. 13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal. 14. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla

	principal sin ninguna modificación.
Flujo del proceso	

Tabla 14

CUN°10

Nombre del CU	Gestión tipos de institución
Nombre	<i>CUN°10</i>
Requerimiento	21
Actores	Admin
Descripción	El admin puede visualizar los tipos de institución existentes, también adicionarlos, editarlos y eliminarlos. (Ver Figura 34)
Precondiciones	Tener rol de admin para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario se dirige a la pantalla "Tipos de instituciones" 2. El prototipo muestra una vista con el listado de tipos de institución existentes en el sistema. 3. El usuario puede seleccionar la opción "Adicionar" 4. El prototipo muestra un modal con el formulario para la creación del tipo de institución. 5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar". 6. Si el formulario fue llenado de manera válida se crea exitosamente el tipo de institución y se visualizará en la grilla principal donde se

	<p>encontrará actualizada.</p> <p>7. El usuario puede seleccionar la opción "Editar"</p> <p>8. El prototipo muestra un modal con el formulario para la edición del tipo de institución seleccionado.</p> <p>9. El usuario puede modificar los campos y pulsar "Guardar".</p> <p>10. Si el formulario fue completado de manera válida se edita exitosamente el tipo de institución y en la grilla principal se actualizará.</p> <p>11. El usuario puede seleccionar la opción "Eliminar"</p> <p>12. El prototipo muestra un modal de confirmación de la acción.</p> <p>13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal.</p> <p>14. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.</p>
--	---

Tabla 15

CUN°11

Nombre del CU	Gestión de servicios
Nombre	<i>CUN°11</i>
Requerimiento	20
Actores	Admin
Descripción	El admin puede visualizar los servicios existentes, también adicionarlos, editarlos y eliminarlos. (Ver Figura 35)
Precondiciones	Tener rol de admin para así tener los permisos necesarios.

Poscondiciones	
	<ol style="list-style-type: none"> 1. El usuario se dirige a la pantalla "Servicios" 2. El prototipo muestra una vista con el listado de los servicios existentes en el sistema. 3. El usuario puede seleccionar la opción "Adicionar" 4. El prototipo muestra un modal con el formulario para la creación del servicio. 5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar". 6. Si el formulario fue llenado de manera válida se crea exitosamente el servicio y se visualizará en la grilla principal donde se encontrará actualizada. 7. El usuario puede seleccionar la opción "Editar" 8. El prototipo muestra un modal con el formulario para la edición del servicio seleccionado. 9. El usuario puede modificar los campos y pulsar "Guardar". 10. Si el formulario fue completado de manera válida se edita exitosamente el servicio y en la grilla principal se actualizará. 11. El usuario puede seleccionar la opción "Eliminar" 12. El prototipo muestra un modal de confirmación de la acción. 13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal. 14. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla
Flujo del proceso	

	principal sin ninguna modificación.
--	-------------------------------------

Tabla 16*CUN°12*

Nombre del CU	Gestión de instituciones
Nombre	<i>CUN°12</i>
Requerimiento	22
Actores	Admin
Descripción	El admin puede visualizar las instituciones existentes, también adicionar, editar y eliminar. (Ver Figura 36)
Precondiciones	Tener rol de administrador para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario presiona en el menú "Instituciones" 2. El prototipo muestra una vista con el listado de las instituciones existentes y puede filtrar la información por nombre y tipo de institución. 3. El usuario puede seleccionar la opción "Adicionar". 4. El prototipo muestra una pantalla con el formulario para la creación de una institución, en esta pantalla se le puede asociar servicios y personal ya anteriormente parametrizados. 5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar". 6. Si el formulario fue llenado de manera válida se crea exitosamente la institución y redirige a la grilla principal donde se encontrará

	<p>actualizada.</p> <p>7. El usuario puede seleccionar la opción "Editar"</p> <p>8. El prototipo muestra una pantalla con el formulario para la edición de la institución seleccionada.</p> <p>9. El usuario puede modificar cualquier campo y pulsar "Guardar".</p> <p>10. Si el formulario fue llenado de manera válida se edita exitosamente la institución y se redirige a la grilla principal donde se encontrará actualizada.</p> <p>11. El usuario puede seleccionar la opción "Eliminar"</p> <p>12. El prototipo muestra un modal de confirmación de la acción.</p> <p>13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal.</p> <p>15. El usuario puede seleccionar la opción "Consultar"</p> <p>16. El prototipo muestra una pantalla con la información de la institución seleccionada.</p> <p>17. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.</p>
--	--

Tabla 17

CUN°13

Nombre del CU	Gestión de situaciones
Nombre	<i>CUN°13</i>
Requerimiento	7, 14 y 31

SISTEMA DE INFORMACIÓN AMACOM

Actores	Admin, Enfermería y Usuario Final
Descripción	Los actores pueden visualizar las situaciones existentes, también adicionarlas, editarlas y eliminarlas. (Ver Figuras 15-18 y 42)
Precondiciones	Tener rol de admin, enfermería o usuario final para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario presiona en el menú "Situaciones" 2. El prototipo muestra una vista con el listado de situaciones existentes en el sistema. 3. El usuario puede seleccionar la opción "Adicionar". 4. El prototipo muestra una pantalla con el formulario para la creación de una situación, en esta pantalla se le puede asociar servicios los signos de alarma de la madre y el bebe ya anteriormente parametrizados. 5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar". 6. Si el formulario fue llenado de manera válida se crea exitosamente la situación y redirige a la grilla principal donde se encontrará actualizada. 7. El usuario puede seleccionar la opción "Editar". 8. El prototipo muestra una pantalla con el formulario para la edición de la situación seleccionada. 9. El usuario puede modificar cualquier campo y pulsar "Guardar". 10. Si el formulario fue llenado de manera válida se edita exitosamente la situación y se redirige a la grilla principal donde se encontrará

	<p>actualizada.</p> <p>11. El usuario puede seleccionar la opción "Eliminar"</p> <p>12. El prototipo muestra un modal de confirmación de la acción.</p> <p>13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal.</p> <p>15. El usuario puede seleccionar la opción "Consultar"</p> <p>16. El prototipo muestra una pantalla con la información de la situación seleccionada.</p> <p>17. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.</p>
--	--

Tabla 18

CUN°14

Nombre del CU	Gestión de diagnósticos
Nombre	<i>CUN°14</i>
Requerimiento	8, 15 y 32
Actores	Admin y Enfermería
Descripción	Los actores pueden visualizar los diagnósticos existentes por situación, también adicionarlos, editarlos y eliminarlos. (Ver Figuras 19-21)
Precondiciones	Tener rol de admin o enfermería para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	<p>1. El usuario se dirige a la pantalla "Consultar situación"</p> <p>2. El prototipo muestra en esa vista una sección con el listado de los</p>

	<p>diagnósticos existentes en el sistema y asociados a dicha situación.</p> <p>3. El usuario puede seleccionar la opción "Adicionar".</p> <p>4. El prototipo muestra una pantalla con el formulario para la creación de un diagnóstico, en esta pantalla se le puede asociar intervenciones y actividades a realizar a la madre.</p> <p>5. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar".</p> <p>6. Si el formulario fue llenado de manera válida se crea exitosamente el diagnóstico y redirige a la grilla principal donde se encontrará actualizada.</p> <p>7. El usuario puede seleccionar la opción "Editar".</p> <p>8. El prototipo muestra una pantalla con el formulario para la edición del diagnóstico seleccionado.</p> <p>9. El usuario puede modificar cualquier campo y pulsar "Guardar".</p> <p>10. Si el formulario fue completado de manera válida se edita exitosamente el diagnóstico y se redirige a la grilla principal donde se encontrará actualizada.</p> <p>11. El usuario puede seleccionar la opción "Eliminar"</p> <p>12. El prototipo muestra un modal de confirmación de la acción.</p> <p>13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal.</p> <p>15. El usuario puede seleccionar la opción "Consultar"</p> <p>16. El prototipo muestra una pantalla con la información del diagnóstico seleccionado.</p> <p>17. Las anteriores funciones de adicionar, editar y eliminar tienen una</p>
--	--

	opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.
--	---

Tabla 19

CUN°15

Nombre del CU	Gestión de bitácora
Nombre	<i>CUN°15</i>
Requerimiento	6, 16 y 33
Actores	Admin, Enfermería y Usuario Final
Descripción	Los actores pueden visualizar las bitácoras existentes, también adicionarlas, editarlas y eliminarlas. (Ver Figuras 22 y 43)
Precondiciones	Tener rol de admin, enfermería o usuario final para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario presiona en el menú "Bitácora" 2. El prototipo muestra una vista con el listado de las bitácoras existentes en el sistema. 3. El usuario puede seleccionar la opción "Adicionar" 4. El prototipo muestra una pantalla con el formulario para la creación de una bitácora. 5. El usuario llena los campos correspondientes y pulsa "Guardar". 6. Si el formulario fue llenado de manera válida se crea exitosamente la bitácora y se visualizará en la grilla principal donde se encontrará actualizada.

	<p>7. El usuario puede seleccionar la opción "Editar"</p> <p>8. El prototipo muestra un modal con el formulario para la edición de la bitácora seleccionada.</p> <p>9. El usuario puede modificar los campos y pulsar "Guardar".</p> <p>10. Si el formulario fue llenado de manera válida se edita exitosamente la bitácora y en la grilla principal se encontrará actualizada.</p> <p>11. El usuario puede seleccionar la opción "Eliminar"</p> <p>12. El prototipo muestra un modal de confirmación de la acción.</p> <p>13. El usuario puede pulsar "Eliminar", el registro será eliminado y actualizado en la grilla principal.</p> <p>14. Las anteriores funciones de adicionar, editar y eliminar tienen una opción de "cancelar", la cual finaliza el proceso y retorna a la grilla principal sin ninguna modificación.</p>
--	--

Tabla 20

CUN°16

Nombre del CU	Calendario
Nombre	<i>CUN°16</i>
Requerimiento	11, 18 y 35
Actores	Admin, Enfermería y Usuario Final
Descripción	Los actores pueden visualizar la funcionalidad de calendario para la gestión y especificación de eventos. (Ver Figura 23 y 41)
Precondiciones	Tener rol de admin, enfermería o usuario final para así tener los permisos necesarios.

Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario presiona en el menú "Calendario" 2. El prototipo muestra una vista con el calendario. 3. El usuario puede visualizarlo y gestionarlo por mes, semana o día 4. El usuario puede pulsar "Agregar evento". 5. El prototipo muestra un modal con el formulario para la creación del evento. 6. El usuario llena los campos correspondientes y necesarios y pulsa "Guardar". 7. Si el formulario fue llenado de manera válida se crea exitosamente el evento y este será visualizado en el calendario. 8. El usuario puede seleccionar el evento. 9. El prototipo muestra un modal con el formulario para la edición del evento seleccionado. 10. El usuario puede modificar los campos y pulsar "Guardar". 11. Si el formulario fue completado de manera válida se edita exitosamente el evento y este se actualizará en el calendario. 12. En este mismo modal el usuario puede seleccionar la opción "Eliminar" 13. El registro será eliminado y actualizado en el calendario. 14. El calendario incluye un menú seleccionable donde el usuario puede filtrar el evento por tipo de evento.

Tabla 21

CUN°17

Nombre del CU	Gestión de funcionalidades por persona
Nombre	CUN°17
Requerimiento	13-17 y 30-34
Actores	Admin y Enfermería
Descripción	Los actores pueden visualizar las personas existentes, también adicionar, editar y eliminar como en el CU de personas, además pueden gestionar por persona sus respectivas situaciones, bitácoras y calendario. (Ver Figuras 14-23)
Precondiciones	Tener rol de admin o enfermería para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario puede realizar el mismo flujo presentado en el CU N° 3 gestión de personas. 2. Cuando el usuario ingresa a la pantalla de "consultar persona" puede seleccionar el botón "Situación", el cual se especifica en el CU N° 13 gestión de situaciones 3. Cuando el usuario ingresa a la pantalla de "consultar persona" puede seleccionar el botón "Bitácora", el cual se especifica en el CU N° 15 gestión de bitácora. 4. Cuando el usuario ingresa a la pantalla de "consultar persona" puede seleccionar el botón "Calendario", el cual se especifica en el CU N° 16 calendario.

Tabla 22

CUN°18

Nombre del CU	Evaluación logros
Nombre	<i>CUN°18</i>
Requerimiento	10
Actores	Usuario Final
Descripción	El actor puede realizar un cuestionario respecto a situaciones que haya vivido en el posparto y así se le asignará un nivel. (Ver Figura 46)
Precondiciones	Tener rol de usuario final para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario selecciona en el menú “Mis logros”. 2. El usuario puede seleccionar “Valoración inicial”. 3. Si el usuario selecciona “Valoración inicial”, puede seleccionar en un formulario con diferentes situaciones que tanto ha pasado por ellas. 4. Cuando el usuario haya llenado exitosamente el formulario puede seleccionar “Guardar Evaluación”. 5. La información se guarda y se le asigna una categoría y un nivel dependiendo de la información que haya llenado en el cuestionario.

Tabla 23

CUN°19

Nombre del CU	Kit materno
Nombre	<i>CUN°19</i>
Requerimiento	9
Actores	Usuario Final

Descripción	El actor puede visualizar un kit materno en el que se encuentran materiales de apoyo. (Ver Figura 45)
Precondiciones	Tener rol de usuario final para así tener los permisos necesarios.
Poscondiciones	N.A.
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario selecciona en el menú “Kit materno”. 2. El usuario puede visualizar los materiales de apoyo parametrizados por tema.

Tabla 24

CUN°20

Nombre del CU	Perfil y configuraciones
Nombre	<i>CUN°20</i>
Requerimiento	5, 12 y 19
Actores	Admin, Enfermería y Usuario Final
Descripción	Los actores pueden visualizar el perfil con su información básica. (Ver Figura 41)
Precondiciones	Tener rol de admin, enfermería o usuario final para así tener los permisos necesarios.
Poscondiciones	
Flujo del proceso	<ol style="list-style-type: none"> 1. El usuario selecciona en su avatar “Cuenta”. 2. El prototipo muestra una pantalla con la información básica del usuario. 3. El usuario también puede seleccionar en el avatar “Configuraciones”.

	<p>4. El prototipo muestra una pantalla con el formulario para la edición de la información básica de su perfil.</p> <p>5. El usuario edita los campos correspondientes y necesarios y pulsa "Guardar".</p> <p>6. Si el formulario fue llenado de manera válida se edita exitosamente el perfil.</p>
--	--

5. Fase de planeación

5.1 Marco metodológico

Es importante en el desarrollo de este proyecto la identificación de las buenas prácticas para la optimización de procesos, de igual manera el uso del marco de trabajo DevOps, el cual busca la unión del desarrollo del software (Dev) y las operaciones de TI (Ops). Esto se hace para promover la interacción, comunicación y colaboración entre las partes; y en consecuencia mejorar la calidad del software, mejorar la respuesta a las necesidades del sistema y alcanzar los objetivos establecidos en el proyecto.

Se trabajó también con el marco de trabajo ágil Scrum, pues permite una mayor flexibilidad, velocidad y organización en los requerimientos. Gracias a la definición de roles, de objetivos, su alcance, tiempo del cumplimiento de estos (sprints), interacción entre los miembros del proyecto; se genera una mayor comunicación, organización y visibilidad del proyecto, dando resultados más eficientes.

La siguiente figura muestra el orden y las fases del marco metodológico:

Figura 2

Fases del marco metodológico



● Análisis de requerimientos

En esta fase se tuvo en cuenta el análisis y establecimiento de los requerimientos funcionales y no funcionales en base a la lógica de negocio, estos se ejecutarán para el desarrollo del prototipo de sistema de información.

● Planeación y preparación del proyecto

Según (Pressman, 2010), es el lugar en el que los requerimientos de los participantes, las necesidades del negocio y las consideraciones técnicas se combinan para dar forma a un sistema. El diseño crea una representación o modelo del software que proporciona información detallada sobre la arquitectura del software, estructura de datos y la estructuración interna que son necesarias para un correcto desarrollo e implementación del sistema.

● Desarrollo del software

Esta fase trata de traducir el diseño del software a un lenguaje que pueda ser interpretado por una computadora. Este proceso incluye entre otras cosas la creación y configuración de ambientes virtuales. En el desarrollo de este proyecto se usarán Java, JavaScript y Flutter como lenguajes de programación. Para la plataforma web se usará HyperText Markup Language 5 (HTML5) como lenguaje hipertexto y Sassy Cascading Style Sheets (SCSS) como lenguaje de hojas de estilos.

● Verificación y Pruebas

Según (Sommerville, 2006), durante y después de cada etapa del desarrollo e implementación, el software debe ser verificado y validado para asegurar que cumple con las funcionalidades esperadas para sus distintos agentes. Este proceso comienza con revisiones a

requerimientos, diseño y desarrollo hasta la finalización del mismo. El objetivo final de la etapa de pruebas es que para cada componente de programa individual se descubran defectos. Posteriormente, cuando se integran para formar subsistemas o el sistema completo, se realizan pruebas de sistema para verificar el cumplimiento de sus requerimientos.

5.2 Planeación de sprints

Conforme al cronograma establecido, las actividades se organizaron siguiendo los principios del marco ágil SCRUM. Este proceso abarcó desde la evaluación de las historias de usuario hasta la determinación de las cargas de trabajo y la organización de los sprints.

Tabla 25

Programación de sprints

SPRINTS	
CANTIDAD	PERIODO DE DURACIÓN
4	4 SEMANAS

Tabla 26

Programación sprint N° 1

N°	Historias de usuario	Complejidad
1	Análisis y definición de requerimientos funcionales y no funcionales.	5
2	Elaboración del modelo ER y el diagrama de flujo del software.	13
3	Diseño y desarrollo de la base de datos usada por el sistema.	21
4	Establecimiento de APIs, herramientas de desarrollo y documentación sobre el manejo de información.	2
5	Configuración e instalación del ambiente local.	3
6	Creación de wireframes de autenticación.	4
7	Creación de wireframes administrativos del sistema.	5
8	Creación de wireframe evaluación de logros	2
9	Creación de wireframe kit materno	2
10	Creación de wireframe calendario	2
TOTAL	10	59

Tabla 27

Programación sprint N° 2

SISTEMA DE INFORMACIÓN AMACOM

Nº	Historias de usuario	Complejidad
1	Creación de las entidades del sistema.	7
2	Creación e implementación de los DTO necesarios en el sistema.	7
3	Creación e implementación de las interfaces necesarias en el sistema.	7
3	Creación e implementación de las clases mapper necesarias en el sistema.	13
4	Creación de las clases rest-controller necesarias en el sistema.	13
5	Creación de roles del sistema: Admin, enfermería y usuario final.	5
6	Creación e implementación de los servicios: Autenticación.	13
7	Creación e implementación de los servicios: CRUD administrativos.	21
8	Creación e implementación de los servicios: evaluación de logros.	7
9	Creación e implementación de los servicios: kit materno.	7
10	Creación e implementación de los servicios: calendario.	7
TOTAL	10	107

Tabla 28

Programación sprint N° 3

Nº	Historias de usuario	Complejidad
1	Configuración del proyecto en Angular	3
2	Implementación de los roles del sistema.	7
3	Desarrollo e implementación del prototipo para la interfaz web del inicio de sesión del software	13
4	Desarrollo e implementación del prototipo para la interfaz web de los módulos administrativos (CRUD).	13
5	Desarrollo e implementación del prototipo para la interfaz web de las situaciones y diagnósticos del usuario.	13
6	Desarrollo e implementación del prototipo para la interfaz web de las bitácoras del usuario.	7
7	Desarrollo e implementación del prototipo para la interfaz web de la evaluación de logros.	13
8	Desarrollo e implementación del prototipo para la interfaz web del kit materno.	7
9	Desarrollo e implementación del prototipo para la interfaz web del calendario.	13
10	Conexión entre servicios backend y frontend.	21
11	Configuración del proyecto en Flutter	3
12	Desarrollo e implementación del prototipo para la interfaz móvil del inicio de sesión del software	13

SISTEMA DE INFORMACIÓN AMACOM

13	Desarrollo e implementación del prototipo para la interfaz móvil de las situaciones y diagnósticos del usuario.	13
14	Desarrollo e implementación del prototipo para la interfaz móvil de las bitácoras del usuario.	7
15	Desarrollo e implementación del prototipo para la interfaz móvil de la evaluación de logros.	13
16	Desarrollo e implementación del prototipo para la interfaz móvil del kit materno.	7
17	Desarrollo e implementación del prototipo para la interfaz móvil del calendario.	13
TOTAL 10		179

Tabla 29

Programación sprint N° 4

N°	Historias de usuario	Complejidad
1	Pruebas unitarias de desarrollador para evaluar la funcionalidad del software.	7
3	Pruebas unitarias en PostMan para evaluar la funcionalidad del software.	7
4	Arreglos y ajustes finales al frontend y backend	5
5	Documentación del código	2
TOTAL	10	21

5.3 Modelo de datos

En base a los requerimientos detectados para el proyecto y el alcance de las funcionalidades del software se diseñó un modelo de base de datos que pudiese cumplir con las necesidades planteadas siguiendo los fundamentos de la normalización. Con ello se buscó garantizar la seguridad y flexibilidad de los datos al eliminar redundancia al evadir el uso de dependencias inconsistentes.

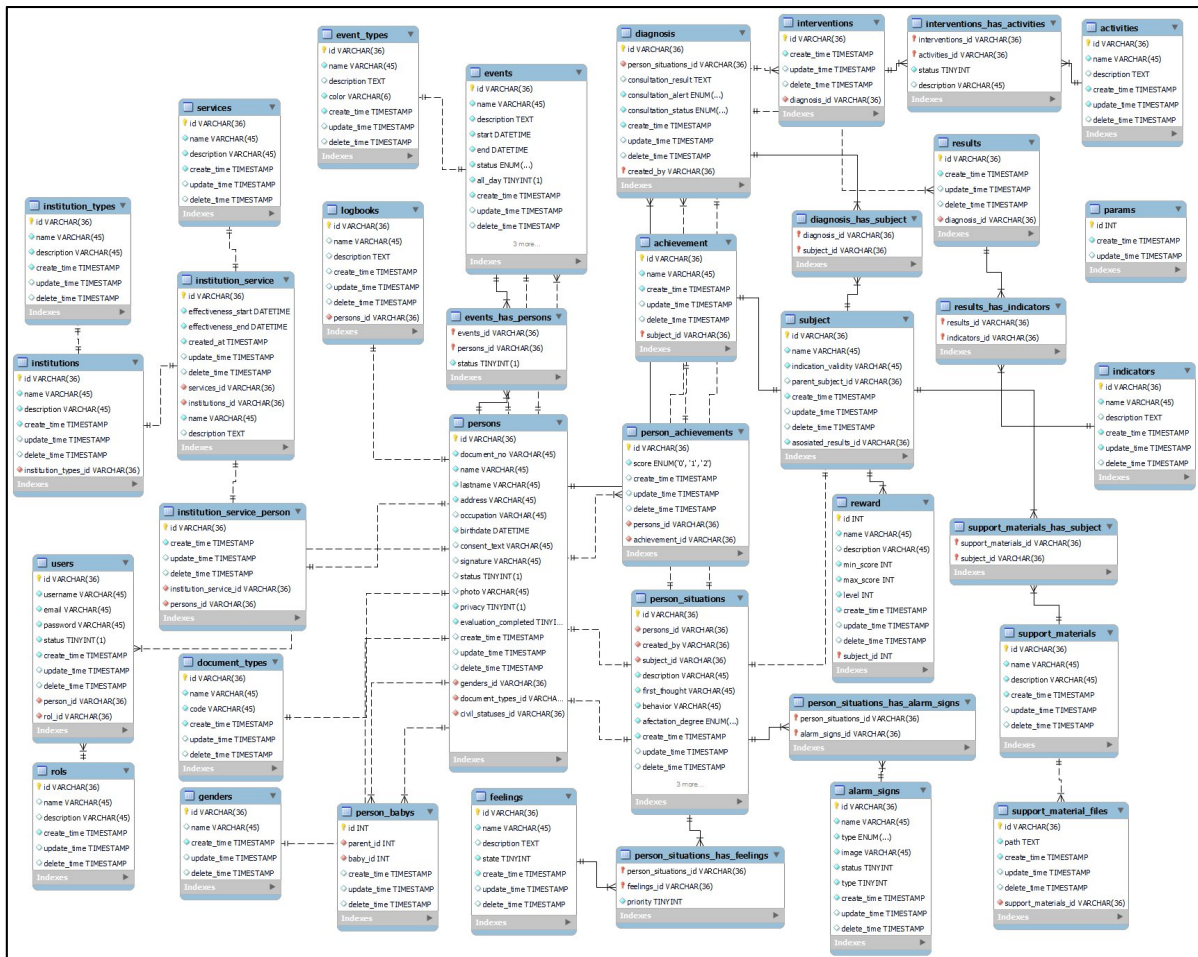
5.3.1 Diseño del modelo de datos satisfaciendo los requerimientos del proyecto

En la fase de planeación propuso un modelo inicial para definir la arquitectura de desarrollo de software que garantice un buen flujo de información para los servicios que se crearían después de diseñar la base de datos, basándose en el análisis de los requerimientos del proyecto.

A partir de ello se tomaron seis aspectos de importancia para su diseño: Usuarios, instituciones, situaciones, bitácoras, eventos, logros y material de apoyo.

Figura 3

Flujo diseño del modelo de base de datos relacional

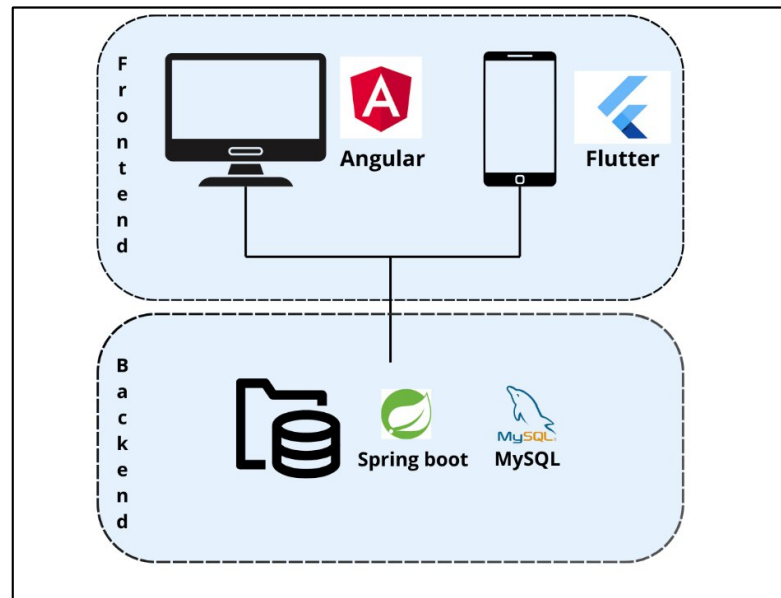


6. Fase de desarrollo

De acuerdo a la revisión realizada en el marco de referencia se estableció un análisis de diferentes arquitecturas y lenguajes enfocados a la necesidad del proyecto se determinó el uso de una arquitectura basada en microservicios la cual está definida en la figura 5.

Figura 4

Diagrama de arquitectura

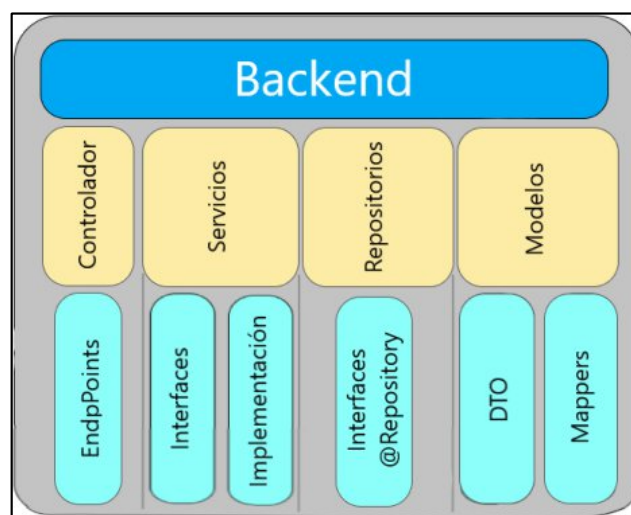


6.1 Desarrollo Back-end

Teniendo en cuenta el requerimiento no funcional expresado por el director de este proyecto, la implementación del backend se realizó en base a Spring Boot, un framework de trabajo basado en Java. Este provee una amplia gama de herramientas que facilitan el desarrollo de aplicaciones y microservicios como el manejo de dependencias, estabilidad, rendimiento y soporte. En base a las herramientas prestadas por Spring Boot se estructuró el proyecto de la siguiente forma:

Figura 5

Estructura interna del back-end

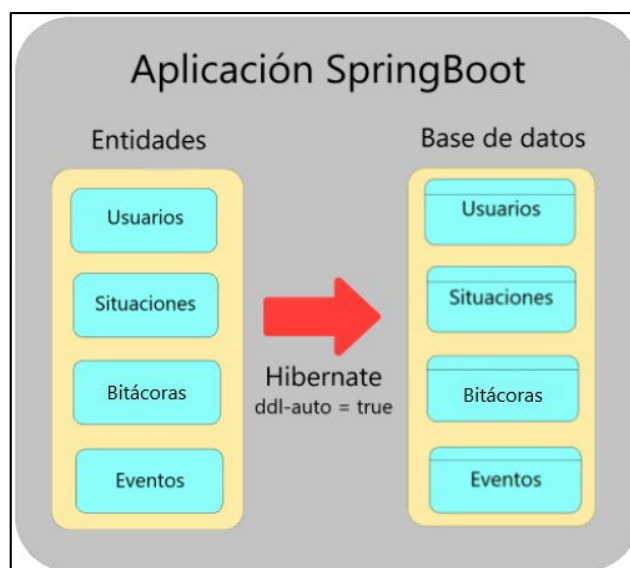


6.1.1 Conexión con la base de datos y desarrollo de modelos de trabajo

Debido a las múltiples características de Spring Boot algunas acciones como el establecer una conexión con la base de datos y realizar migraciones se hacen de forma nativa al configurar parámetros de conexión, modelos y habilitar el ddl-auto. Al habilitar dicha propiedad se le indica al framework que se debe generar el esquema de la base de datos en base a los modelos marcados como entidades en la aplicación. Encargándose de labores de creación, validación, actualización y borrado de las tablas y propiedades.

Figura 6

Conexión a base de datos y migraciones



La configuración de los modelos en el backend se realizó teniendo en cuenta que estos actúan como una representación de cada tabla y sus relaciones en la base de datos. Todo ello considerando que el framework soporta el uso de ORM (Object-Relational mapping).

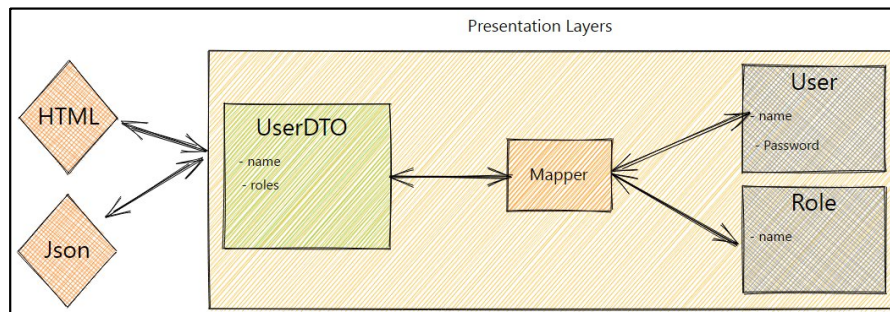
6.1.2 Implementación de data transfer objects (DTO) y mappers

En la recepción y envío de respuesta en peticiones HTTP que responden a labores de búsqueda, creación y modificación de registros. Para realizar operaciones sobre registros en la base de datos se realiza un mapeo de los DTO a modelos de datos, y a su vez cada vez que se

leen registros de la base de datos estos se mapean de forma inversa. El mapeo de objetos a DTO y viceversa se realiza mediante mappers que realizan dicha acción mediante la asignación de propiedades según se requiere en base al nombre o en base a anotaciones del desarrollador.

Figura 7

Uso de los DTO en la recepción y envío de datos



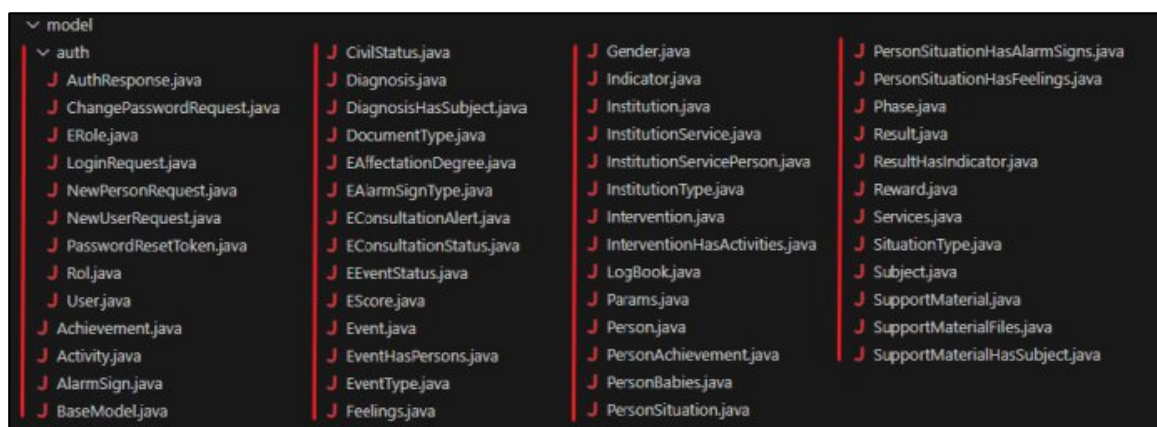
Nota: Recuperado de The DTO Pattern

6.1.3 Construcción y desarrollo de los modelos de datos

En la aplicación backend los modelos de datos constituyen la forma de definir e interactuar con cada tabla en la base de datos. Cada modelo es básicamente una clase marcada como entidad que se relaciona directamente con una tabla en la base de datos junto con sus propiedades.

Figura 8

Índice de modelos de datos de la aplicación



Las propiedades de cada entidad requieren ser señaladas como columnas que pueden a su vez crear relaciones con otras entidades, definir valores por defecto, señalar un tipo especial de propiedad, entre otros.

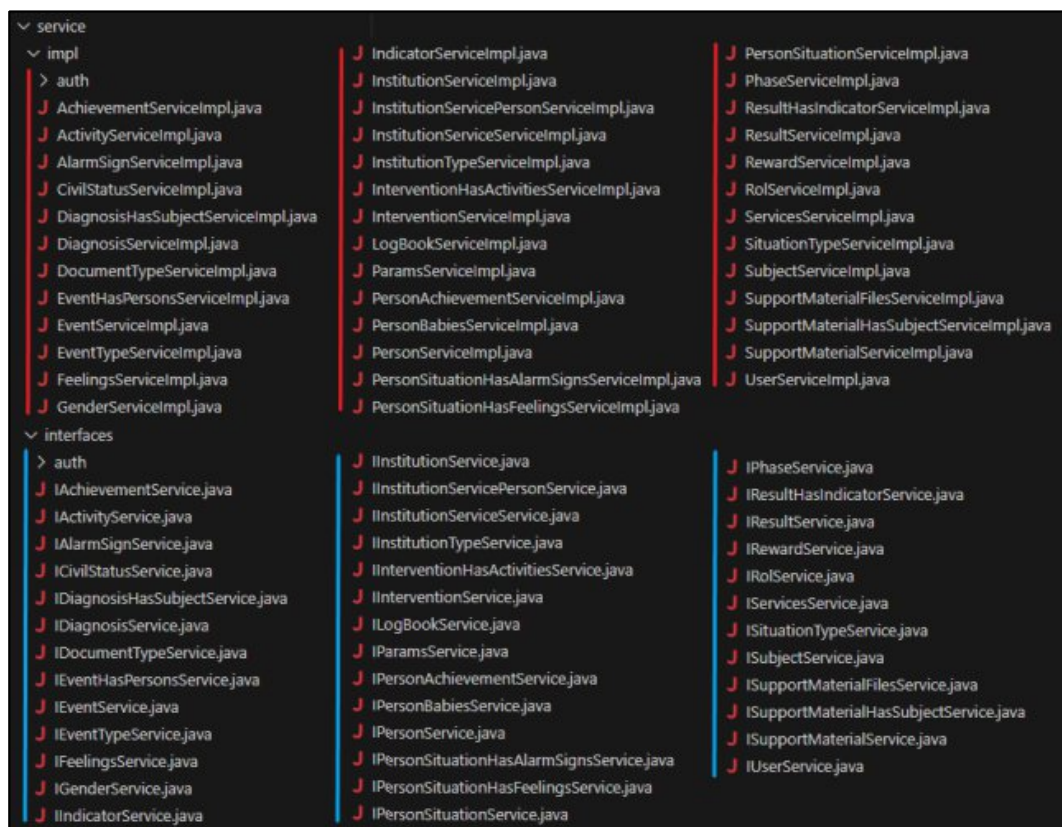
6.1.4 Construcción de servicios y repositorios

Una vez implementados los modelos de datos se realizó la implementación de los servicios y repositorios respectivos para cada modelo. Para cada servicio se construyó una interfaz que permite usar interfaces en la definición de instancias, esto facilita el proceso de pruebas al permitir la creación de mocks.

Los servicios realizan funciones como autenticación, recepción, procesamiento, guardado y envío de registros, usando otros servicios o repositorios que operan en la base de datos. También hacen funciones adicionales como encriptación y envío de correo. Todo esto se hace según los casos de uso y requerimientos del usuario.

Figura 9

Servicios de la aplicación

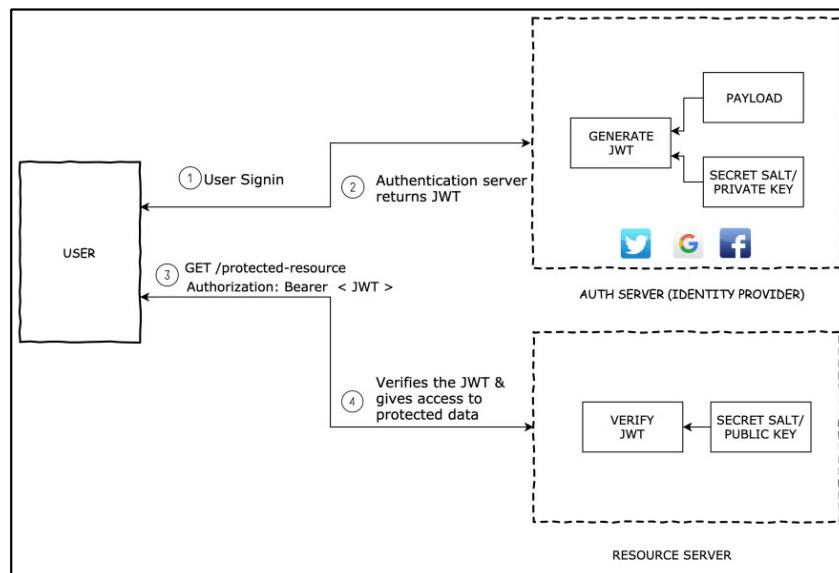


6.1.5 Implementación de controladores y validación de tokens de acceso

La implementación de controladores para la recepción y procesamiento de peticiones HTTP se llevó a cabo usando las herramientas de Spring Boot que permiten el definir una clase como un controlador rest, de esta forma para cada modelo se desarrolló un controlador con distintas rutas para cumplir con las diferentes operaciones de los usuarios en el sistema. Adicionalmente, se configuró un filtro de autenticación basado en JWT para cada endpoint salvo aquellos requeridos para el registro de usuarios finales del sistema y el login.

Figura 10

Funcionamiento autenticación JWT



Nota: Recuperado de How JWT authentication works

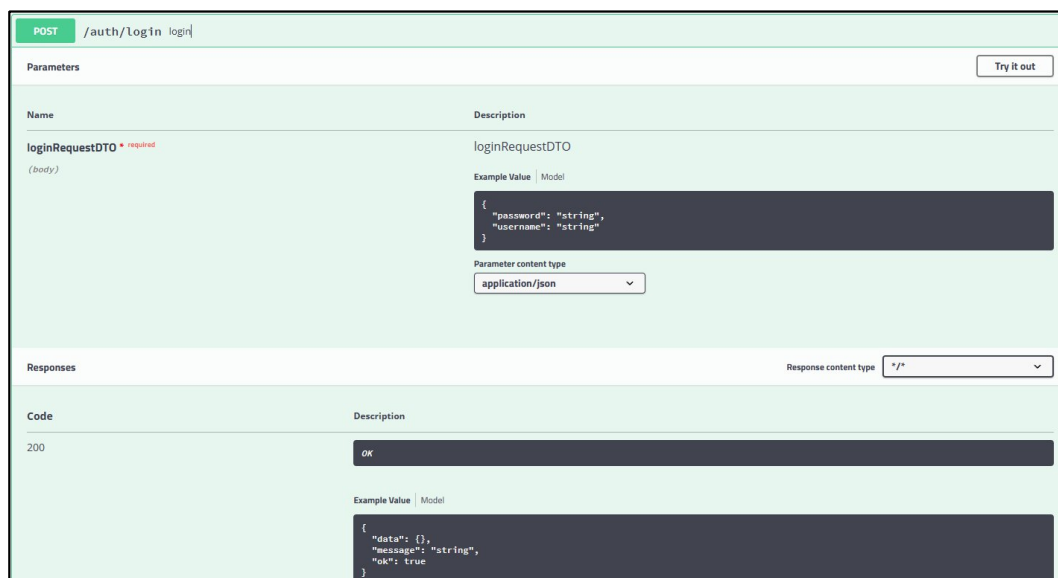
El desarrollo de controladores se realizó en base al patrón de arquitectura MVC (Modelo Vista Controlador), esto en pro de la independencia entre componentes de la aplicación. En este patrón de arquitectura el controlador actúa como mediador entre el modelo y la vista. Se encarga de procesar las solicitudes del usuario y retornar una respuesta.

6.1.6 Documentación del API REST con Swagger.

Swagger en Spring Boot es una herramienta que permite generar dinámicamente documentación en REST API. Una vez configurado en el proyecto, permite la renderización de un apartado web que presenta las diferentes rutas disponibles para la ejecución de peticiones HTTP así como los parámetros y respuestas esperadas.

Figura 11

Interfaz gráfica Swagger

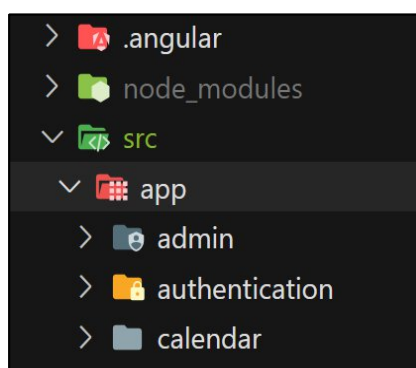


6.2 Desarrollo del prototipo web

Partiendo de las herramientas iniciales establecidas durante el primer sprint en Scrum, entre las cuales se incluyó el framework Angular, se dió inicio al proyecto mediante la creación de la configuración angular. Así, se implementó una estructura de carpetas de manera organizada que además refleja la distinción de roles dentro del sistema, los cuales son: administrador, enfermería y usuario final, y a su vez también ciertas funcionalidades como autenticación y calendario. Esta separación facilitó la creación ordenada de componentes para cada módulo, ayudando a un desarrollo organizado y eficaz.

Figura 12

Índice de estructura angular, admin, autenticación y calendario

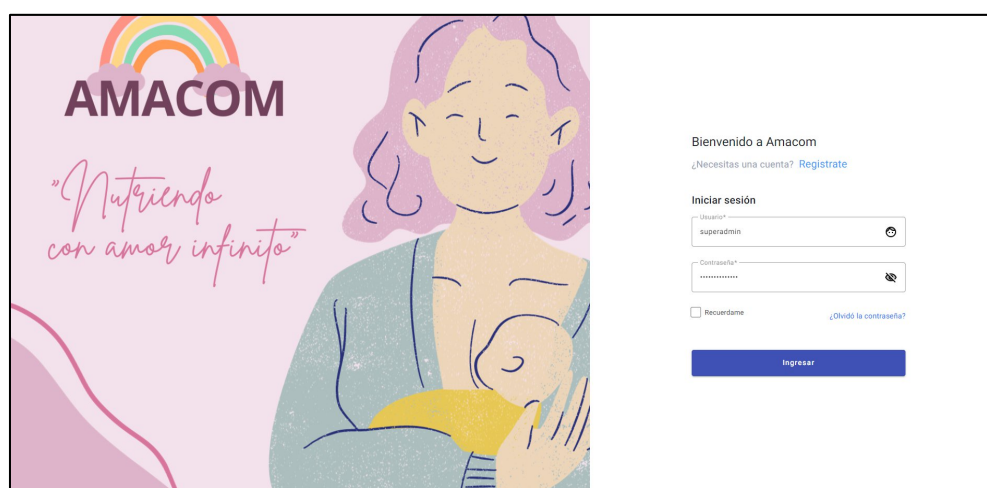


6.2.1 Autenticación de usuarios

La autenticación de usuarios es un módulo relevante en el desarrollo del proyecto Amacom, ya que facilita que los usuarios registrados y correctamente validados puedan ingresar al prototipo web de una manera segura, así permitiendo acceso a las diferentes funcionalidades que el sistema posee dependiendo del rol de cada usuario.

Figura 13

Vista inicio sesión del software



6.2.3 Módulo detalle persona

El módulo más destacado de Amacom es el detalle de la persona. En este, los usuarios con roles de administrador o enfermería disponen de acceso al CRUD de personas. Aquí, pueden gestionar la información de cada individuo mediante la adición, edición, eliminación y consulta de datos. Esta última característica, no solo proporciona información detallada sobre la persona, sino que también ofrece las siguientes funcionalidades:

Figura 14

Vista listado de personas

Lista de personas

En esta pantalla podrás visualizar las personas

Lista de personas

🔍 Escribe para filtrar

No	Nombre Completo	Genero	Documento	última actualización	Acciones
c532ae36	Admin principal	Masculino	Cedula de ciudadanía	9/1/2024	🔍 📄 🗑️
140e9afc	mario Pruebando	Masculino	Cedula de ciudadanía	4/1/2024	🔍 📄 🗑️
7be2de4a	Name LastName	Masculino	Cedula de ciudadanía	18/1/2024	🔍 📄 🗑️
58a4017e	PruebaDos prueba	No binario	Cedula de ciudadanía	19/1/2024	🔍 📄 🗑️
beae07f5	pruebaveeb prueba	No binario	Cedula de ciudadanía	14/1/2024	🔍 📄 🗑️

Items per page: 5 1 - 5 of 6

6.2.3.1 Gestión de situaciones

Este componente es esencial, ya que facilita al personal de enfermería la gestión integral de las situaciones relacionadas con el usuario final. Permite una interacción detallada y personalizada, asegurando que cada caso de cada madre sea manejado de acuerdo con sus necesidades específicas. El flujo detallado de este proceso, incluyendo los pasos y procedimientos que el personal de enfermería sigue, se describe en el apartado 5.4, titulado 'Proceso de enfermería', brindando así una visión más amplia del procedimiento de gestión de situaciones en la etapa de posparto de una paciente.

Figura 15

Vista consultar persona - gestión situaciones

SISTEMA DE INFORMACIÓN AMACOM

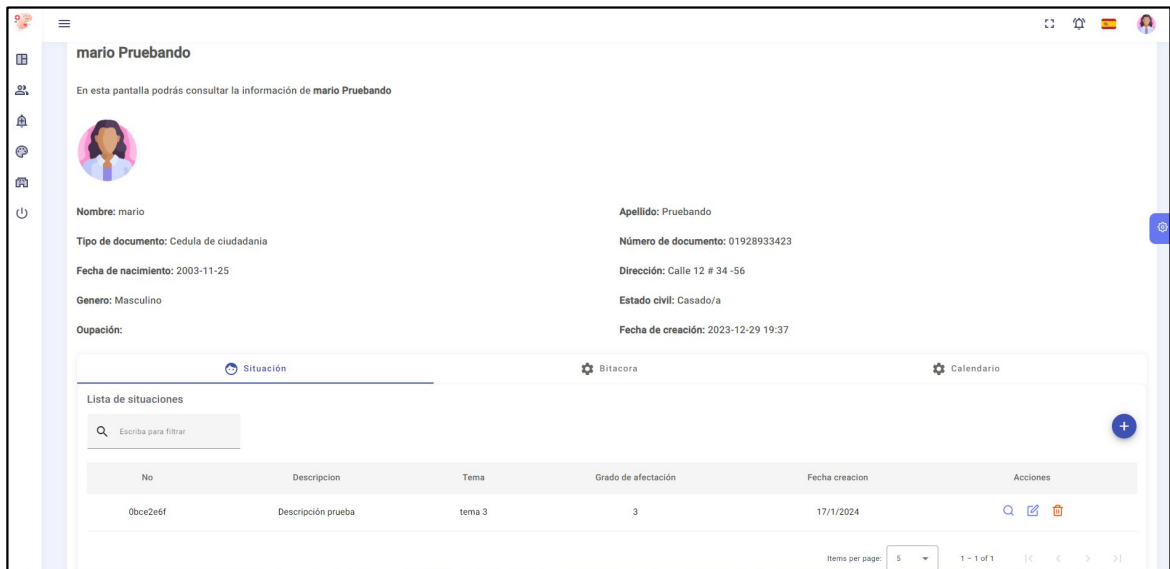


Figura 16

Vista adicionar situación primera parte, detalle situación

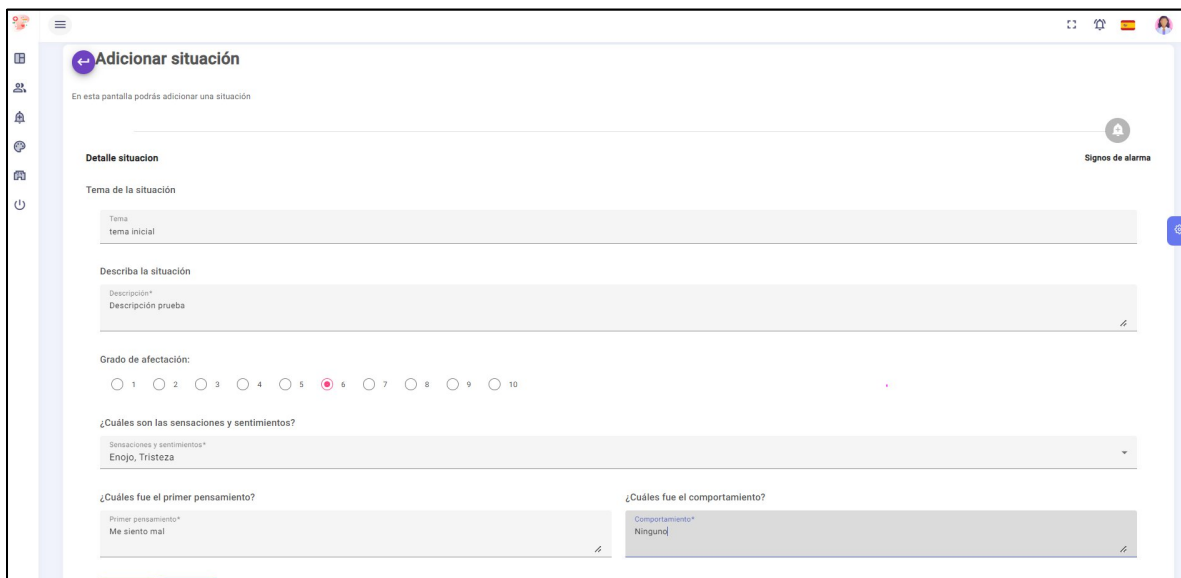


Figura 17

Vista adicionar situación segunda parte, signos de alarma

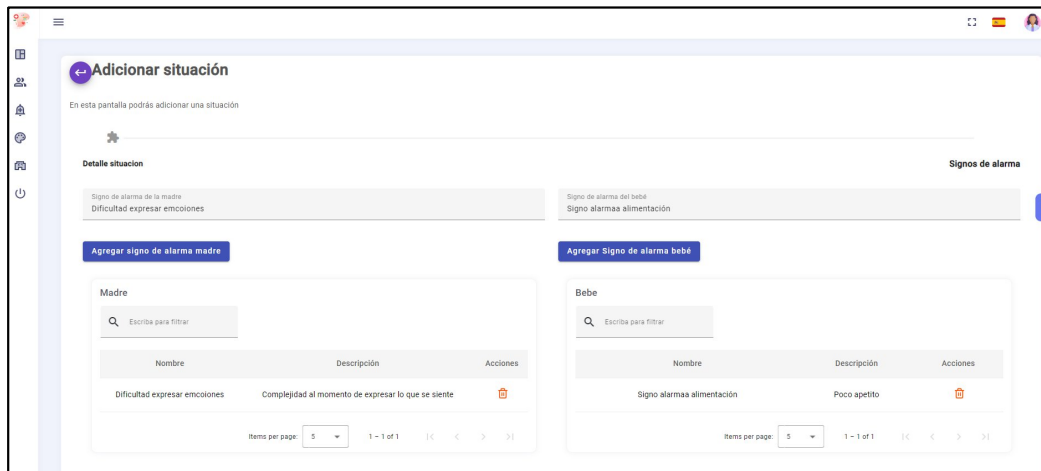
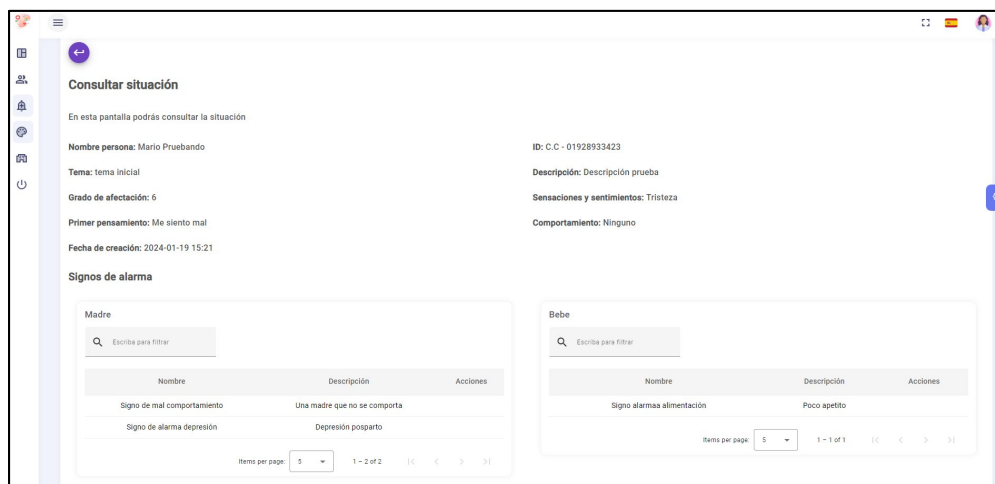


Figura 18

Vista consultar situación primera parte



El personal de enfermería primero examina detenidamente la situación del paciente. Luego de este análisis, pueden identificar y registrar uno o varios diagnósticos. A partir de allí, puede ingresar actividades que velen por el cuidado de la paciente. Finalmente, enfermería revisa cómo se ha desarrollado todo el proceso y registra un resultado, asegurándose de que el paciente reciba la atención más adecuada a su situación.

Figura 19

Vista consultar situación segunda parte, gestión de diagnósticos

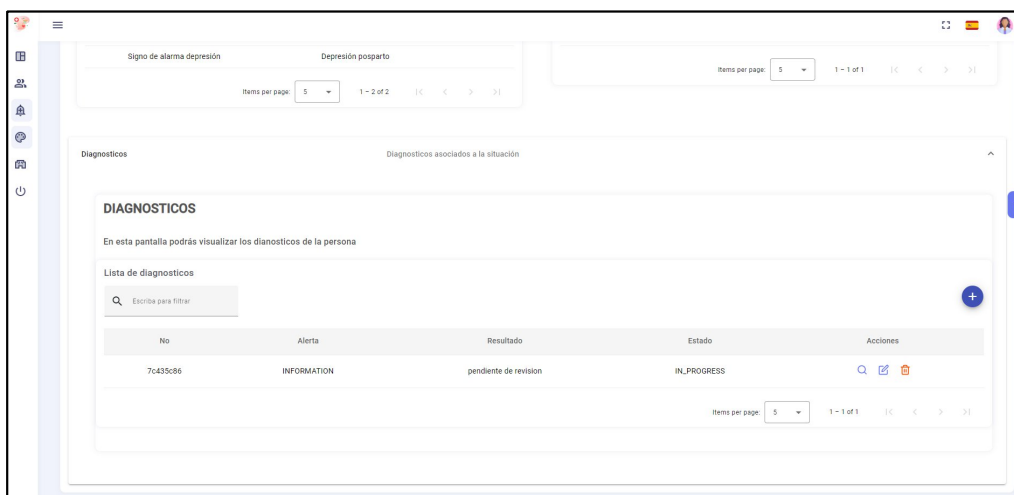


Figura 20

Vista adicional diagnóstico primera parte

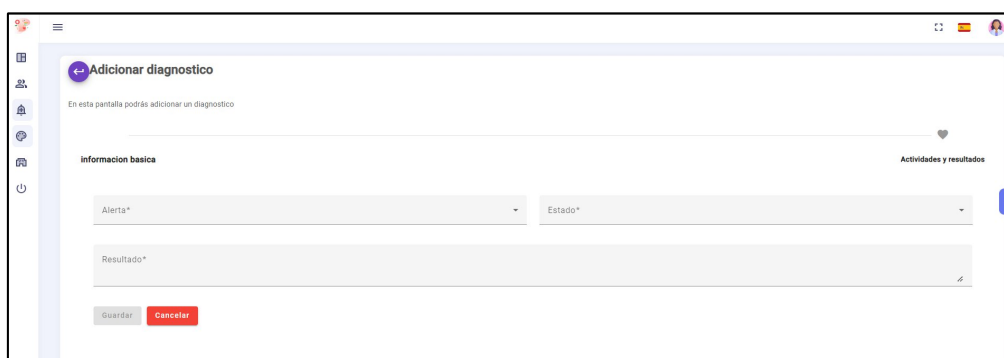
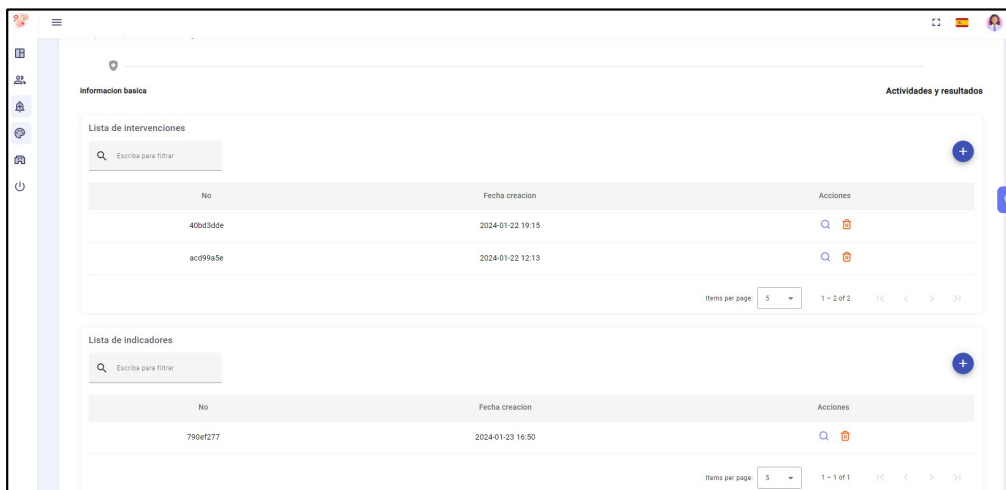


Figura 21

Vista adicional diagnóstico segunda parte

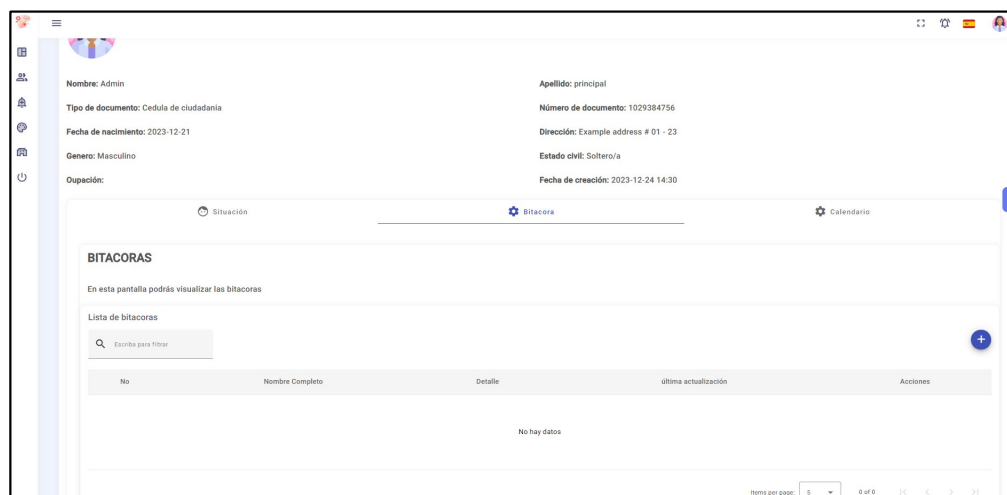


6.2.3.2 Gestión de bitácoras

Este componente lleva la misma estructura que los demás gestionar, de esta manera se puede llevar un registro del diario del usuario.

Figura 22

Vista gestionar bitácoras

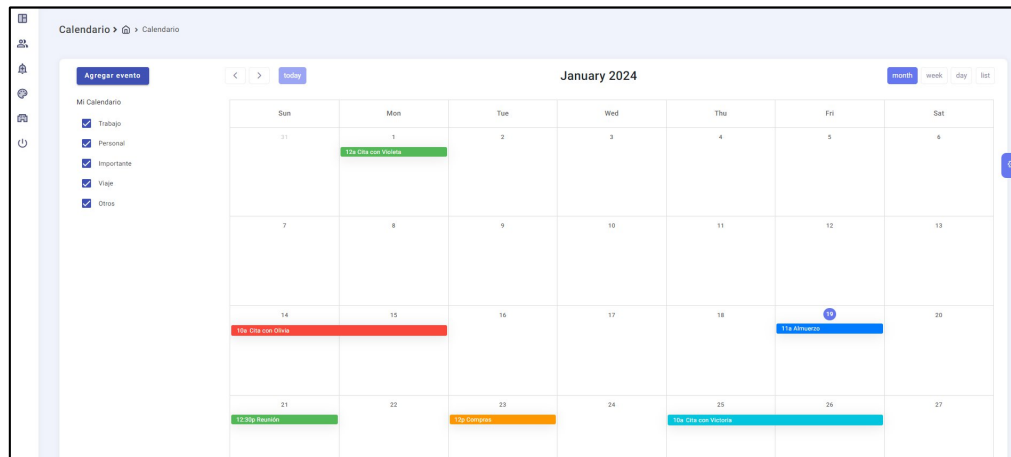


6.2.3.3 Calendario

Este módulo como su nombre lo indica presenta un calendario que permite al usuario agregar eventos y administrarlos, ofreciendo así una herramienta para organizar de manera efectiva su rutina diaria.

Figura 23

Vista calendario



6.2.2 Módulos de mantenimiento (administrativos)

Uno de los componentes fundamentales en este sistema son los módulos de mantenimiento, a los cuales solo tiene acceso el rol de administrador. Estos se encargan de llevar a cabo operaciones CRUD para administrar y mantener la información parametrizada en el sistema. Esta información ajustada juega un papel crucial al facilitar los datos necesarios en el proceso de otros módulos del sistema.

6.2.3.1 Gestión de usuarios

Tomando como ejemplo el módulo CRUD de usuarios, el cual desempeña un papel crucial dentro del sistema, ya que este módulo se encarga de recopilar información detallada de cada usuario, posibilitando así el ingreso y gestión efectiva de estos usuarios en la plataforma. De esta manera el flujo que sigue es: primero se ingresa al menú contentivo de las diferentes funcionalidades del sistema, allí yace el módulo de usuarios, en el que el admin no solo va a poder visualizar los diferentes usuarios existentes con su información, sino también tiene los permisos necesarios para agregarlos, editarlos y eliminarlos. Todo este proceso de adición y edición culmina satisfactoriamente completando el formulario y validando los campos con la información de cada usuario ingresado.

Figura 24

Vista listado de usuarios

SISTEMA DE INFORMACIÓN AMACOM

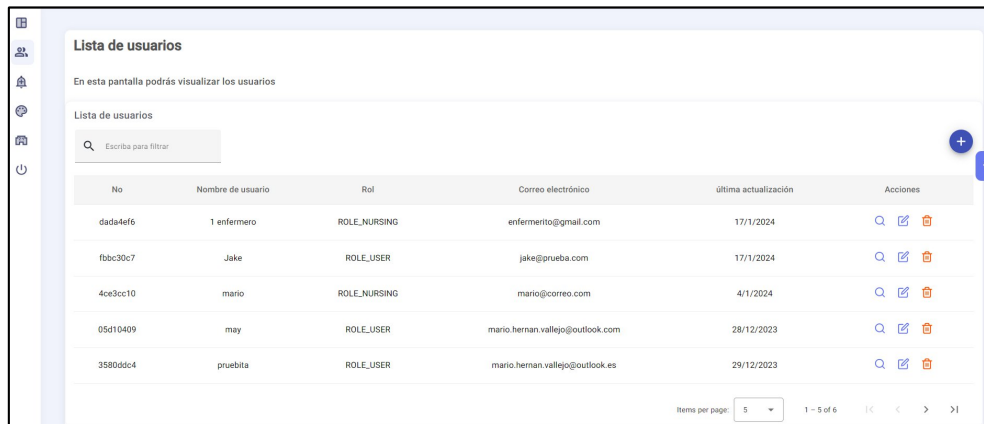


Figura 25

Vista adicionar usuario

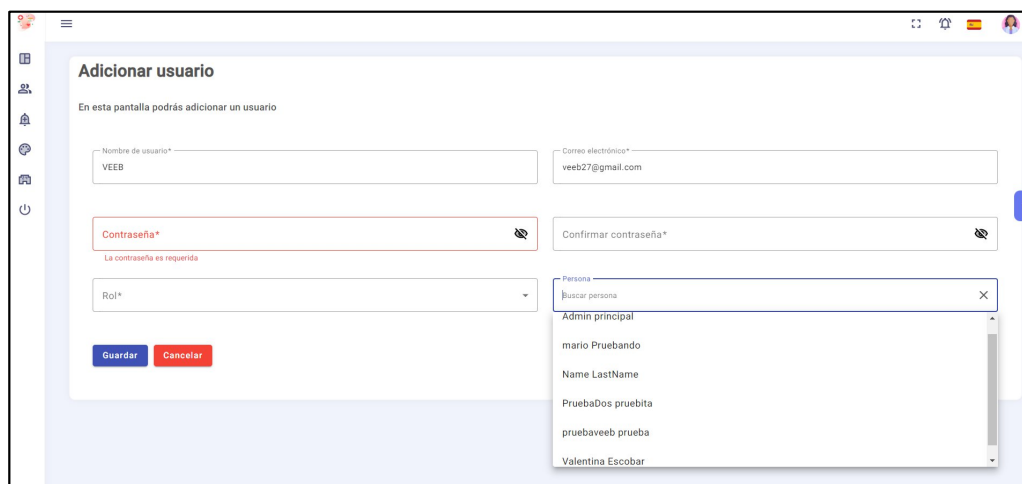


Figura 26

Vista modal consulta de usuario

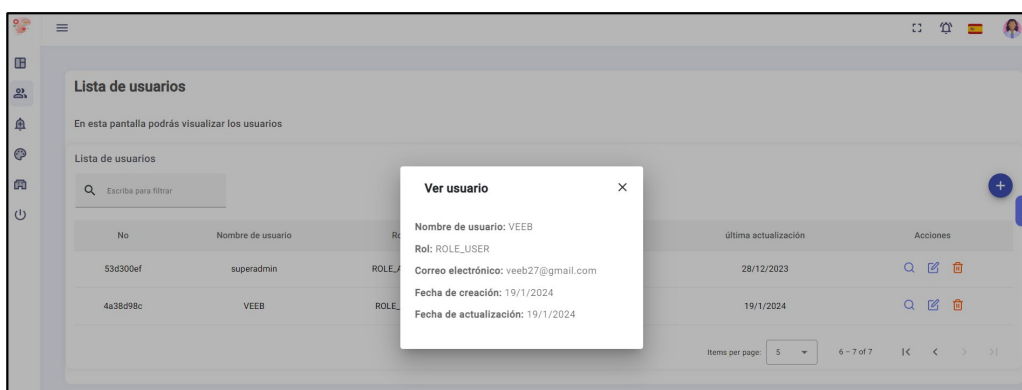
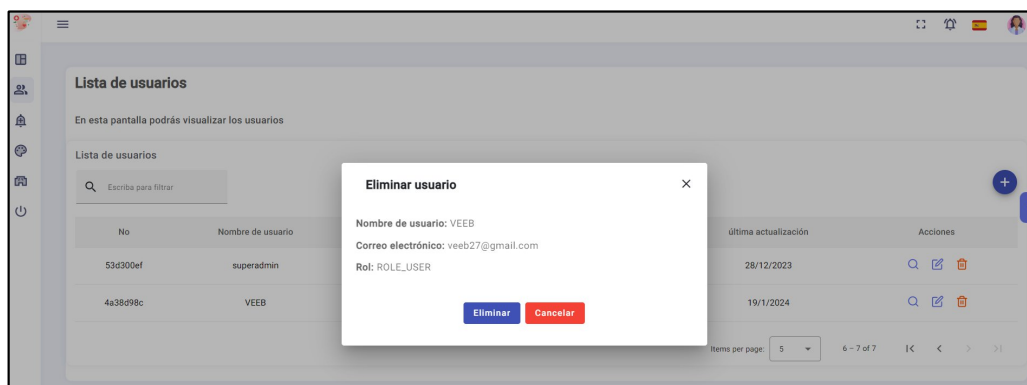


Figura 27

Vista eliminar usuario



Así, como el CRUD mencionado anteriormente se realizó, de la misma manera se configuró y estructuraron los demás CRUDs dentro del módulo de mantenimiento. Entre estos se incluyen: gestión de personas, géneros, tipos de documento, signos de alarma de la madre y el bebé, temas, logros, material de apoyo, tipos de instituciones, servicios e instituciones.

6.2.3.2 Gestión de géneros

En la gestión de usuarios, personas y personal es necesario y de vital importancia tener su información base, ya que esto facilita el control y el seguimiento de las personas y los servicios que requieren, por lo tanto, tener una gestión del género es relevante no solo para poder identificar esta información en cada usuario sino también permite que no haya limitaciones al momento de querer gestionar los géneros, de esta manera en este módulo el admin puede visualizar, adicionar, editar y eliminar esta información.

Figura 28

Vista CRUD de géneros

No	Genero	última actualización	Acciones
02e447c2	Otro	14/1/2024	[Search] [Edit] [Delete]
9ee184ff	Femenino	24/12/2023	[Search] [Edit] [Delete]
ad5ea406	Masculino	24/12/2023	[Search] [Edit] [Delete]
af26df28	No binario	9/1/2024	[Search] [Edit] [Delete]

6.2.3.3 Gestión tipos de documento

De la misma manera que se gestiona la funcionalidad de los géneros, también se desarrolló la gestión de tipos de documentos para así poder tener un mejor control de la información básica de cada usuario

Figura 29

Vista CRUD tipos de documento

No	Nombre	Codigo	última actualización	Acciones
49c37435	Cédula de ciudadanía	C.C	24/12/2023	[Search] [Edit] [Delete]
ed609291	Pasaporte	pas	17/1/2024	[Search] [Edit] [Delete]
f1f73020	Tarjeta de identidad	Ti	9/1/2024	[Search] [Edit] [Delete]

6.2.3.4 Gestión signos de alarma

El proceso que se lleva a cabo en este módulo se basa en el mismo procedimiento de los gestionar, en este caso es fundamental debido a que estos signos de alarma representan los focos rojos que pueden tener una madre y su hijo, por tal motivo se hace una asignación a cada signo para así poder identificar si son de la mamá o de su bebe.

Figura 30

Vista CRUD signos de alarma

Signos de alarma

En esta pantalla podrás visualizar los signos de alarma existentes

Lista de signos de alarma

Busca para filtrar

No	Nombre	Descripción	Tipo	Estado	última actualización	Acciones
e4302f10	Dificultad expresar emociones	Complejidad al momento de expresar lo que se siente	MOTHER	Activo	23/1/2024	[Iconos de acción]
1bf00f54	Dolor de barriga	Léle panisha	MOTHER	Inactivo	12/1/2024	[Iconos de acción]
79aa5509	Diarrea	Pues tiene diarrea :)	BABY	Inactivo	12/1/2024	[Iconos de acción]
cd38c900	Signo alarmas alimentación	Poco apetito	BABY	Activo	9/1/2024	[Iconos de acción]
2e5756d5	Signo de alarma depresión	Depresión posparto	MOTHER	Activo	9/1/2024	[Iconos de acción]

Items per page: 5 | 1 - 5 of 6

6.2.3.5 Gestión temas y subtemas

Es importante llevar a cabo un control de los temas del proyecto, este módulo resulta crucial ya que constituye un eje fundamental en numerosos procesos. La relevancia de este módulo se debe a la importancia de asociar adecuadamente la mayoría de los procesos del sistema a un tema correspondiente ya parametrizado.

Figura 31

Vista CRUD signos de temas y subtemas

TEMAS Y SUBTEMAS

En esta pantalla podrás visualizar los temas y subtemas existentes

Lista de temas y subtemas

Busca para filtrar

No	Nombre	validéz	última actualización	Acciones
2032eb3	tema de amacom	validado	2024-01-22 14:15	[Iconos de acción]
c26f504d	Tema de prueba 1	Validity	2024-01-12 06:26	[Iconos de acción]
9f232095	Tema de prueba 2	Validity	2024-01-12 06:27	[Iconos de acción]
c915212c	tema inicial	aaa	2024-01-07 21:28	[Iconos de acción]

Items per page: 5 | 1 - 4 of 4

6.2.3.6 Gestión de logros

Este componente es fundamental, la importancia de la gestión de los logros del sistema radica en que permite una evaluación exhaustiva del usuario final en diversos aspectos del proceso posparto, asegurando así una comprensión integral y efectiva de su progreso y necesidades

Figura 32

Vista CRUD de logros

No	Nombre	última actualización	Acciones
1	Logro 1	22/09/2023	[Edit] [Delete]
2	Logro 2	22/09/2023	[Edit] [Delete]
3	Logro 3	22/09/2023	[Edit] [Delete]

6.2.3.7 Gestión de material de apoyo

Este módulo, dedicado al material de apoyo, sigue el mismo protocolo de gestión que los demás CRUD. Su funcionalidad clave reside en facilitar la presentación de este material a las pacientes, proporcionando así información valiosa y asistencia durante su proceso de posparto.

Figura 33

Vista CRUD material de apoyo

No	Nombre	Descripción	última actualización	Acciones
1	Material 1	material de apoyo 1	21/08/2023	[Edit] [Delete]
2	Material 2	material de apoyo 2	22/08/2023	[Edit] [Delete]
3	Material 3	material de apoyo 3	22/08/2023	[Edit] [Delete]

6.2.3.8 Gestión de tipos de institución

Este gestor se basa en un crud donde se pueden parametrizar diferentes tipos de instituciones, cumple un papel importante debido a que en el gestor de instituciones (6.2.3.10), requiere asociar un tipo de institución al momento de parametrizar una institución.

Figura 34

Vista CRUD tipos de institución

No	Nombre	Descripción	última actualización	Acciones
9ad05f28	Privada	descripción entidad pruebaa	9/1/2024	[Iconos de acciones]
54150223	Pública	Es un tipo entidad	9/1/2024	[Iconos de acciones]

6.2.3.9 Gestión de servicios

En el módulo de servicios se lleva el mismo proceso de los gestionar, donde se presenta un listado de los servicios presentes en el sistema, pueden ser adicionados, editados o eliminados. Es de vital importancia debido a que estos servicios están asociados a las instituciones parametrizadas.

Figura 35

Vista CRUD de servicios

No	Nombre	Descripción	última actualización	Acciones
f099f459	pruebaa		17/1/2024	[Iconos de acciones]
6934b8a6	servicio 3		7/1/2024	[Iconos de acciones]
75e601f7	servicio 1		5/1/2024	[Iconos de acciones]
e12f9568	servicio 2		7/1/2024	[Iconos de acciones]

6.2.3.10 Gestión de instituciones

Para el gestionar instituciones, se muestra un listado de las instituciones registradas en el sistema, así mismo se puede crear, editar y eliminarlas. Es importante mencionar que al momento de parametrizar una institución se le asocian servicios, tipo de institución y personal que ya anteriormente han sido parametrizados.

Figura 36

Vista CRUD de instituciones

INSTITUCIONES

En esta pantalla podrás visualizar las instituciones existentes

Lista de instituciones

Escríbela para filtrar

No	Nombre	Descripción	Tipo	última actualización	Acciones
904fe061	institución de educación superior	descripción prueba	Pública	16/1/2024	[Iconos de acciones]
e9c15aa1	instituto de confesionalco	Una empresa muy util	Privada	16/1/2024	[Iconos de acciones]
4d50c17	prueba	prueba	Privada	9/1/2024	[Iconos de acciones]
22679ab3	prueba institución	prueba	Pública	17/1/2024	[Iconos de acciones]
b2b5f027	pruebaa	prueba	Privada	17/1/2024	[Iconos de acciones]

Items per page: 5 1 - 5 of 6 [Paging icons]

6.3 Desarrollo prototipo App Móvil

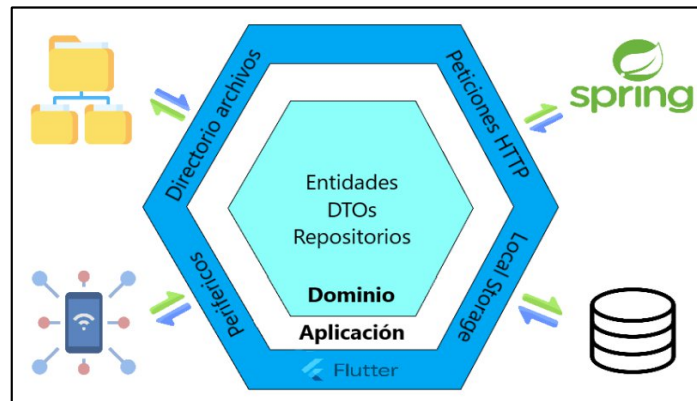
El prototipo se diseñó y desarrolló teniendo en cuenta diversos aspectos relacionados con el perfil del usuario final al que se dirige. Debido a la notable diferencia en el espacio disponible para mostrar información, se requirió una propuesta distinta a la de la web, adaptada al reducido tamaño de la pantalla. Para ello, se llevaron a cabo diversas actividades relacionadas con la arquitectura, las librerías y los estilos a emplear.

Flutter usa como base un motor gráfico que realiza la renderización en dos dimensiones de los elementos gráficos o Widgets. El framework cuenta con el amplio respaldo de la comunidad y dispone de una variada oferta de librerías de elementos gráficos, gestores de estado, etc. Esto permitió reducir considerablemente el tiempo de desarrollo al evitar la implementación desde cero.

El prototipo se elaboró siguiendo el patrón de arquitectura hexagonal, que logra separar la lógica de negocio del sistema de la tecnología subyacente, ofreciendo ventajas evidentes como migrabilidad, flexibilidad y mantenibilidad.

Figura 37

Arquitectura hexagonal aplicada al prototipo



El uso del patrón de diseño hexagonal se evidencia en el uso de interfaces para definir la lógica para los diferentes proveedores de datos y repositorios, así como el uso de modelos y DTOs para la comunicación con el API REST. Esto se evidencia en la estructura del directorio de la app.

Figura 38

Estructura interna prototipo App Móvil

```

    ✓ lib
      > l10n
    ✓ src
      > config
    ✓ data
      > dataSources
      > repositories
    ✓ domain
      > dtos
      > entities
      > repositories
    ✓ presentation
      > state
      > views
      > widgets
      > utils
    main.dart
  
```

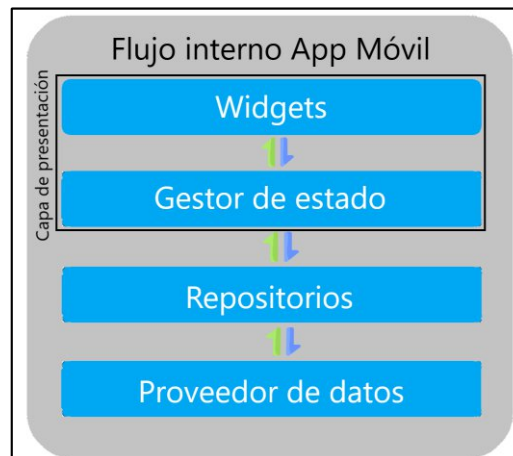
6.3.1 Implementación de la capa de presentación

Juntos los componentes gráficos (Widgets) y el gestor de estado conforman la capa de presentación o UI. Quien se encarga de presentar y capturar información en base a las necesidades de los requerimientos de la aplicación. De forma general, cuando el usuario solicita o ingresa información, el componente gráfico notifica al gestor de estado quien se encarga de

hacer el llamado a los repositorios para enviar o solicitar los datos necesarios. Posteriormente, al obtener una respuesta, este notifica a los elementos gráficos para que actualicen su estado en caso de ser necesario.

Figura 39

Flujo de funcionamiento del prototipo móvil

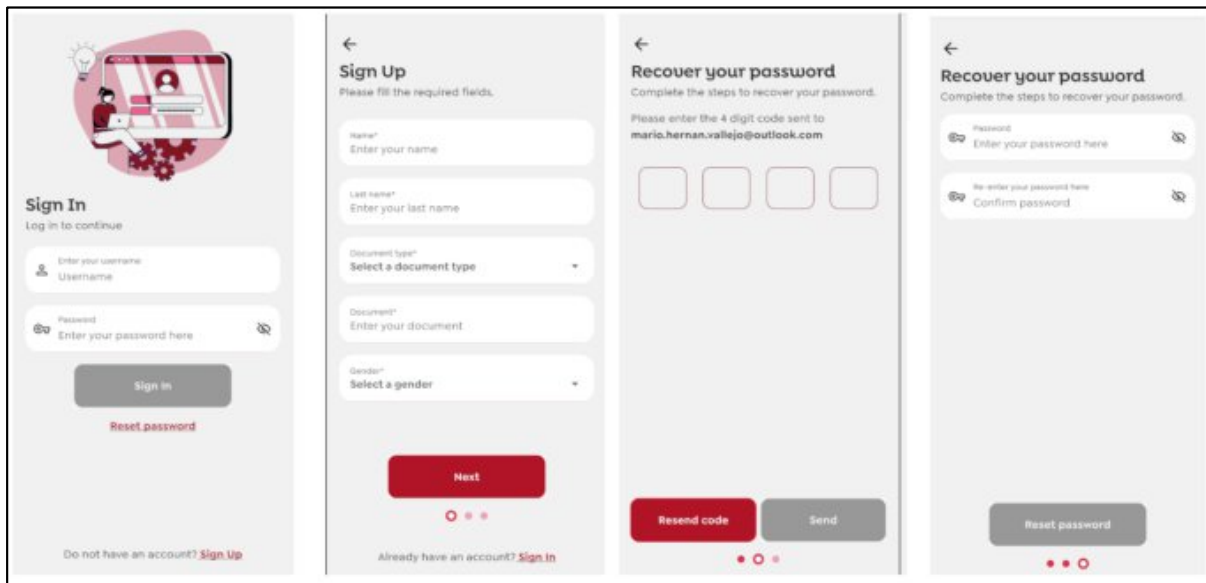


6.3.1.1 Desarrollo del UI para el módulo de autenticación

Teniendo en cuenta que el módulo de autenticación es la puerta de entrada a las funcionalidades del prototipo y la primera interacción con el usuario, se desarrolló pensando en darle al usuario un acceso seguro, intuitivo y eficiente. Para ello se tomó la disposición de elementos presente en los diferentes antecedentes presentados en este proyecto. Como primera actividad del módulo se realizó una maquetación inicial de las diferentes vistas, separándolas en elementos que juntos las conforman. Posteriormente se añadieron diferentes instancias del gestor de estado para el manejo del estado de los elementos gráficos que los requieren y la captación de información de los diferentes formularios.

Figura 40

Vistas del módulo de autenticación del prototipo móvil

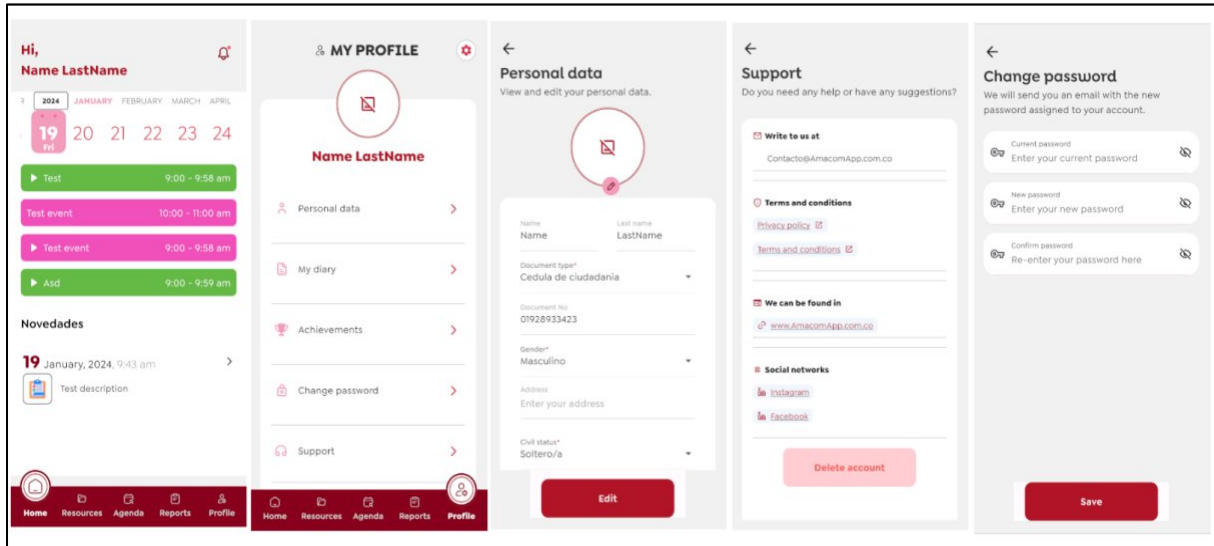


6.3.1.2 Desarrollo del UI para el módulo de datos de usuario y otros

Posterior al desarrollo del módulo de autenticación, se desarrolló una barra de navegación para habilitar al usuario la navegación entre las diferentes funcionalidades del prototipo de forma fácil e intuitiva. Con ello realizado, se procedió a complementar el módulo de autenticación añadiendo las vistas necesarias para actualizar datos de usuario y obtener acceso a módulos y vistas como logros, bitácora y soporte. Adicionalmente, se añadió una pantalla inicial que muestra los eventos cercanos y las novedades en cuanto a las situaciones registradas por el usuario.

Figura 41

Vistas del módulo de datos de usuario y otros del prototipo móvil

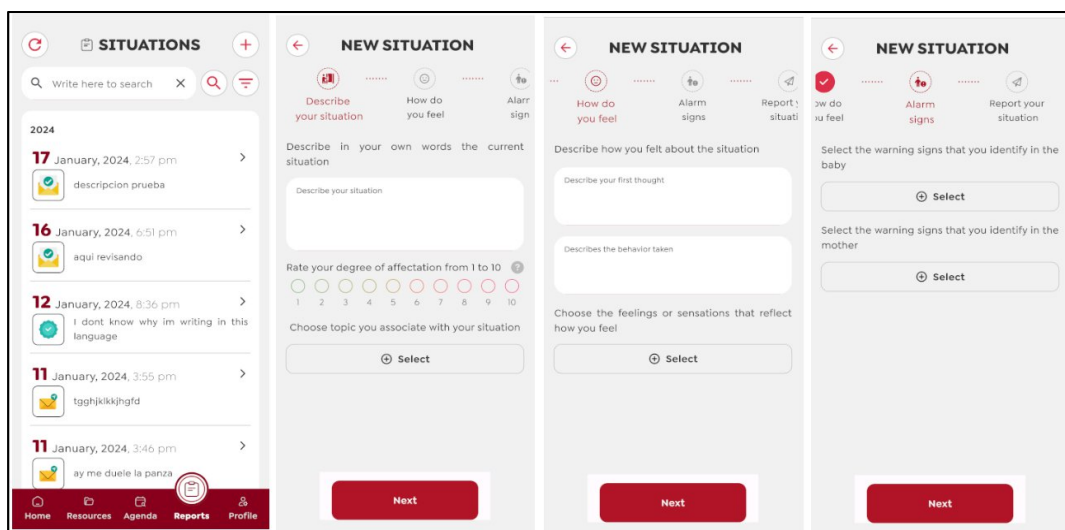


6.3.1.3 .Desarrollo del UI para el módulo de situaciones

Una vez establecida una forma efectiva e intuitiva de navegación entre las funcionalidades del prototipo, se procedió a desarrollar el módulo de situaciones donde el usuario puede consultar el estado de las situaciones registradas y registrar nuevas situaciones en el sistema. Pudiendo filtrar por diferentes parámetros de búsqueda como palabras clave y el estado de las mismas. En el caso del formulario de registro, debido a su extensión se dividió en diferentes páginas que favorecen la experiencia del usuario en el llenado del mismo.

Figura 42

Vistas del módulo de situaciones del prototipo móvil

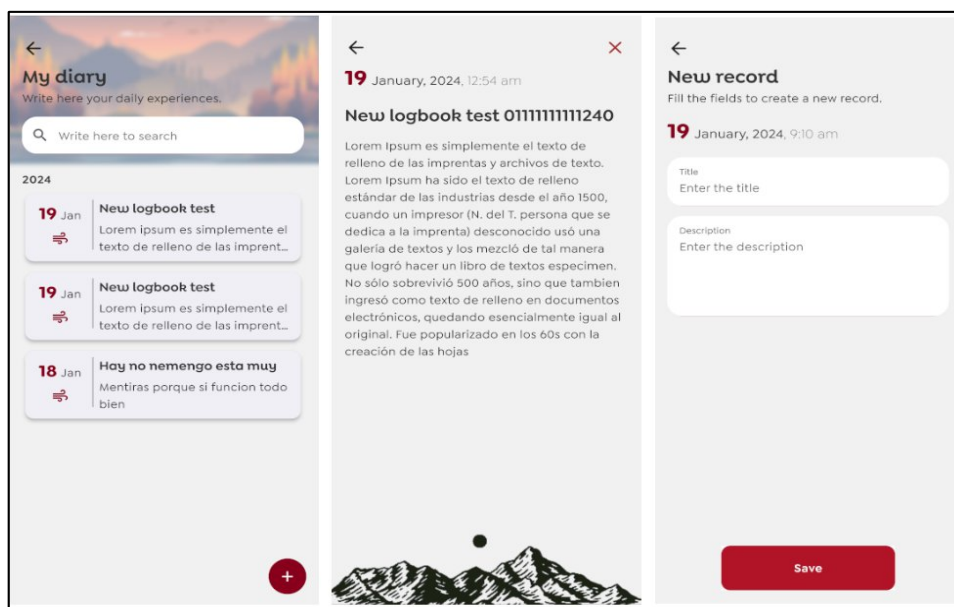


6.3.1.4 Desarrollo del UI para el módulo de bitácora

Para la gestión de bitácora del usuario se desarrolló una interfaz que presenta el listado de bitácoras creadas por el usuario, permitiendo filtrar mediante la búsqueda de palabras clave y la gestión de cada registro. Adicionalmente se presenta el formulario de creación donde se solicitan los datos necesarios en una sola página.

Figura 43

Vistas del módulo de bitácora del prototipo móvil

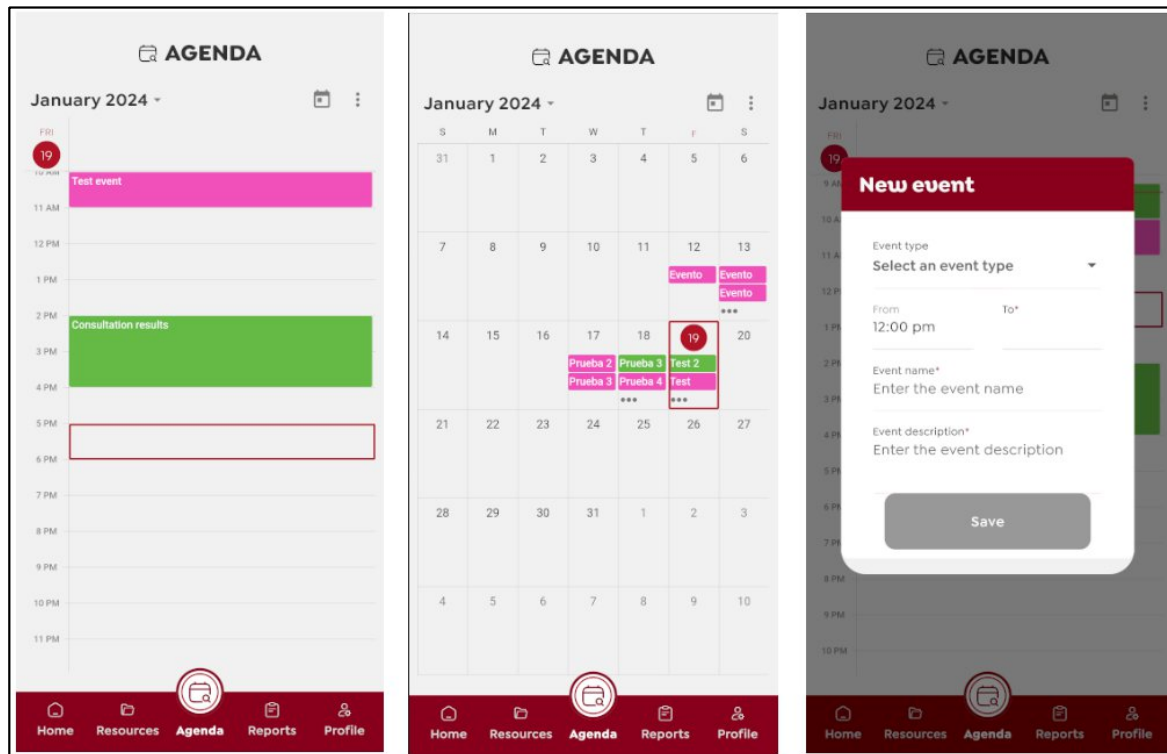


6.3.1.5 Desarrollo del UI para el módulo de agenda

En la interfaz destinada a la gestión de agenda del usuario se desarrolló una vista que presenta el listado de eventos en donde el usuario es partícipe, permitiendo filtrar mediante la selección del rango de fechas seleccionadas en la vista. Adicionalmente se presenta el formulario de creación donde se solicitan los datos necesarios en una ventana emergente.

Figura 44

Vistas del módulo de agenda del prototipo móvil

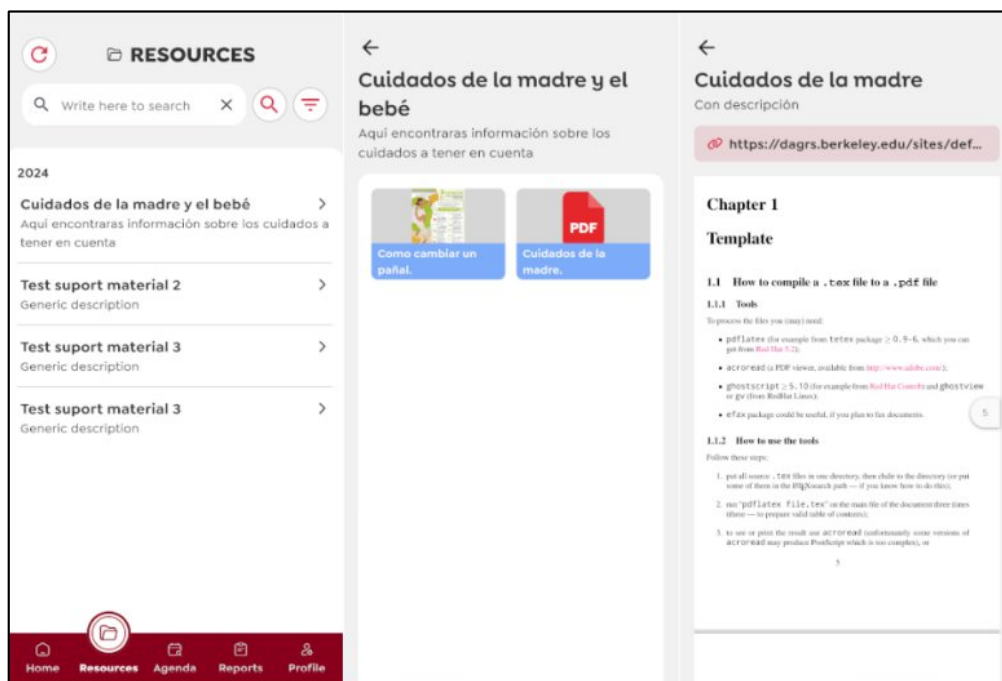


6.3.1.6 Desarrollo del UI para el módulo de material de apoyo

En la interfaz destinada a la exploración del módulo del material de apoyo se desarrolló una vista que presenta el listado de registros, permitiendo filtrar mediante entre palabras clave y la selección de un tema en concreto. Como cada material de apoyo posee un conjunto de archivos asociado, en la vista del detalle se muestra el listado de archivos mostrando el tipo de archivo en cuestión. Al seleccionar un ítem, se muestra su detalle permitiendo de forma dinámica la visualización de imágenes, documentos y videos.

Figura 45

Vistas del módulo de material de apoyo del prototipo móvil

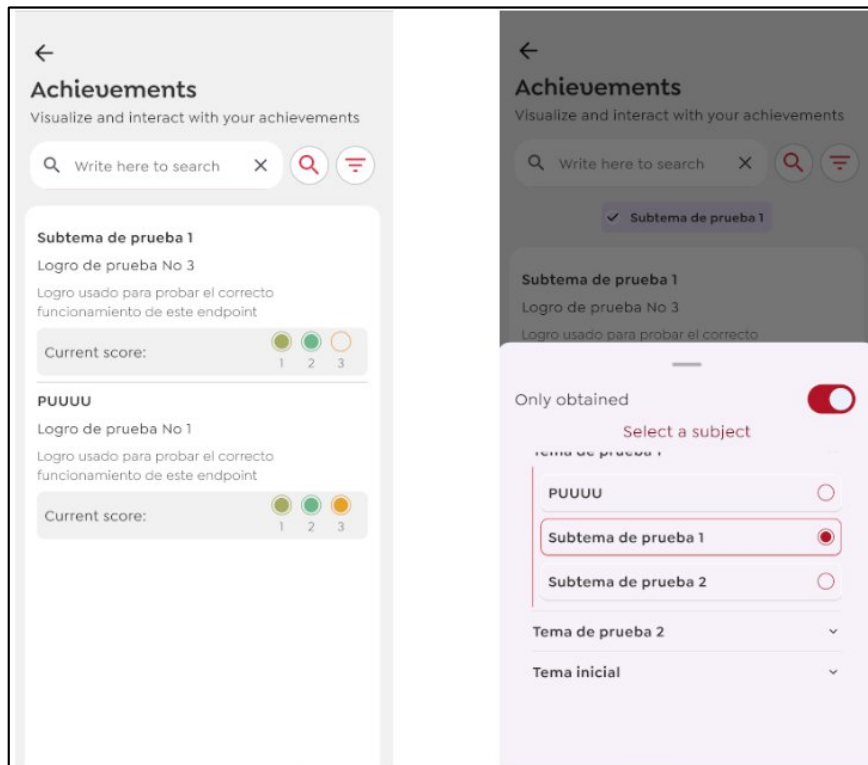


6.3.1.7 Desarrollo del UI para el módulo de logros

En la interfaz destinada a la visualización de logros se desarrolló una vista que presenta el listado de temas que contienen los logros disponibles y un elemento que refleja el rango actual del usuario. Al explorar cada tema se redirige a una vista donde se tienen los logros adjuntos al tema, señalando el avance en cada uno. En dicho listado se puede actualizar el avance del usuario en cualquiera de los ítems.

Figura 46

Vistas del módulo de logros del prototipo móvil



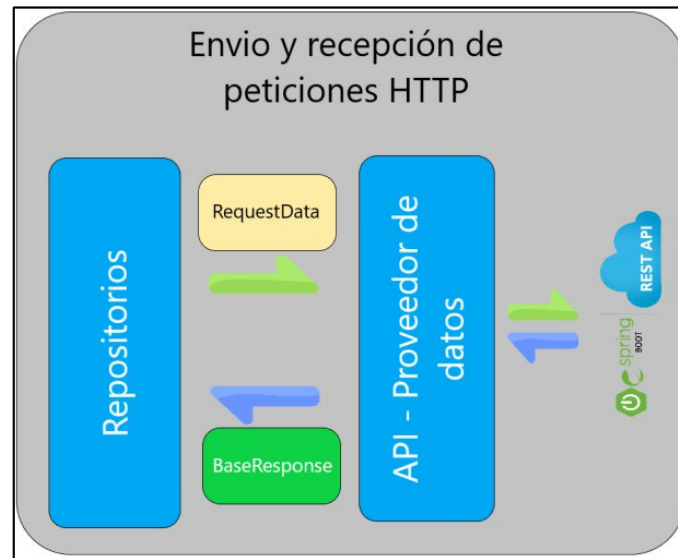
6.3.2 Implementación de repositorios, entidades y DTO

En base a las funcionalidades que se implementaron en el prototipo inicialmente se crearon entidades y DTO que tienen el papel de recibir y enviar información al servidor. También se definieron las interfaces de los distintos repositorios relacionados a cada entidad, con ello se definieron las diferentes funciones, parámetros y respuestas esperadas. Finalmente, se realizó la implementación de cada repositorio. Generalmente se realizan peticiones HTTP al API REST para realizar operaciones CRUD o se guardan y consultan datos de forma local como es el caso de los JWT.

Para facilitar la tarea de envío y recepción de peticiones HTTP se crearon diferentes clases que permiten la construcción de peticiones y la recepción de información de forma simple como se muestra en la figura.

Figura 47

Envío y recepción de peticiones HTTP



Una vez la implementación estuvo completa se procedió a realizar la integración con la capa de presentación, donde gracias a la arquitectura seleccionada solo fue necesario el crear una instancia de la implementación de las interfaces de los repositorios a los proveedores de estado para hacer que todo funcionara con datos del servidor.

7. Fase de pruebas

Para las pruebas de funcionalidad, rendimiento y estabilidad se hizo uso de un servidor Azure Standard B1s, el cual cuenta con 1vcpu y 1GiB de memoria, con Linux (Ubuntu 20.04). Sobre el cual se alojó el backend y permitió la realización de pruebas en un entorno real.

7.1 Pruebas unitarias

En esta etapa del ciclo de vida del software es esencial abordar la fase de pruebas del mismo, donde las unitarias fueron cruciales para la evaluación exhaustiva de las funcionalidades, tanto a nivel del back-end como del front-end del sistema. Es importante mencionar que dichas pruebas fueron aplicadas a lo largo del desarrollo de este se estuvo evaluando cada componente mientras se iba realizando, ya que esto facilita la detección temprana de errores y así soluciones más eficaces y efectivas.

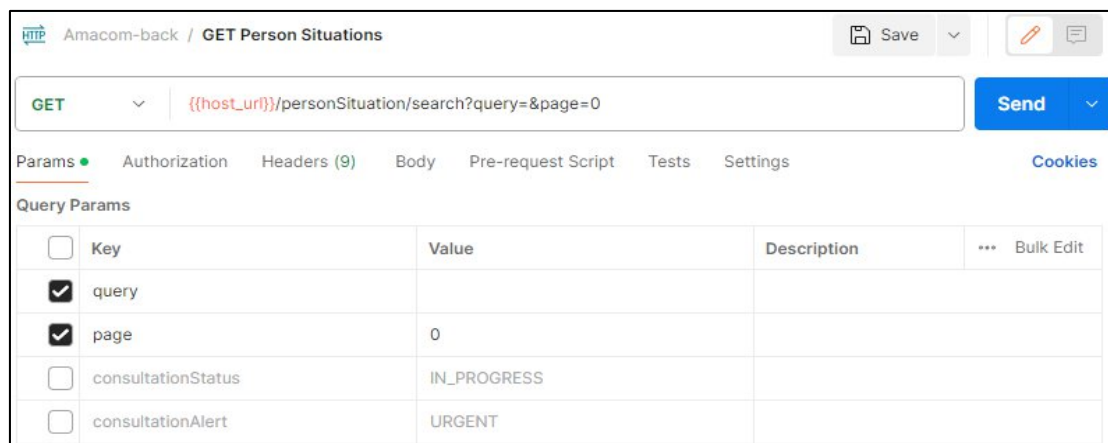
Para poder visualizar y entender de una mejor manera este proceso de pruebas unitarias, se adjunta un video en el que se observa de manera más clara los componentes y funcionalidades del prototipo web y móvil en el acompañamiento de posparto de una madre: [Prototipo sistema de información multiplataforma para el acompañamiento de padres en el proceso de posparto - GoogleDrive](#)

7.2 Rendimiento y estabilidad en la recepción de peticiones http con PostMan

Mediante el uso de la herramienta PostMan que nos permite la realización de pruebas y documentación de REST APIs, se diseñó una colección de peticiones HTTP a diferentes endpoints del Backend con los métodos GET, POST, PUT y DELETE con los diferentes parámetros requeridos.

Figura 48

Petición HTTP GET usando PostMan



PostMan habilita el establecer variables de colección que pueden ser editadas en la fase de pruebas, posibilitando editar dinámicamente datos de autenticación como los JWT y variables como índices. Esto resulta clave para la construcción de peticiones HTTP donde se requiere modificar el contenido de forma dinámica para evitar errores relacionados a la unicidad de los registros en la base de datos. En la colección de endpoints se establecieron dos pruebas principales para todos los endpoints de la misma. Las pruebas se escribieron en

JavaScript bajo las siguientes condiciones: Un tiempo de espera menor a 300 ms y la evaluación del código de respuesta de la petición en el rango 200-204, que son aquellos usados para peticiones exitosas.

Figura 49

Criterios de prueba generales para la colección de endpoints y edición de variables de colección

```
These tests will execute after every request in this collection. Learn more about Postman's execution order.  
1 pm.test("Status code is 200-204", function () {  
2   pm.expect(pm.response.code).to.be.oneOf([200, 201, 202, 203, 204])  
3 });  
4 pm.test("Response time is less than 300ms", function () {  
5   pm.expect(pm.response.responseTime).to.be.below(300);  
6 });  
7  
8  
9 var numeric_value = Number(pm.collectionVariables.get("numeric_value"));  
10  
11 pm.collectionVariables.set("numeric_value", numeric_value + 1);  
12
```

7.2.1 Pruebas de funcionales con PostMan

Con el uso de pruebas funcionales se evaluó la funcionalidad en los diferentes endpoints de la conexión teniendo como criterio las pruebas establecidas en la colección, quedando pendiente el fijar el número de iteraciones y el delay entre cada una. Para la prueba en cuestión se realizaron 100 iteraciones con 0 ms de delay entre cada una arrojando como resultado un tiempo de respuesta promedio de 101 ms en un total de 6600 peticiones, donde solo 8 de ellas no pasaron las pruebas de tiempo de respuesta por un margen muy corto, estableciendo un porcentaje de error de solo el 0.11%.

Figura 50

Resultados de la prueba funcional

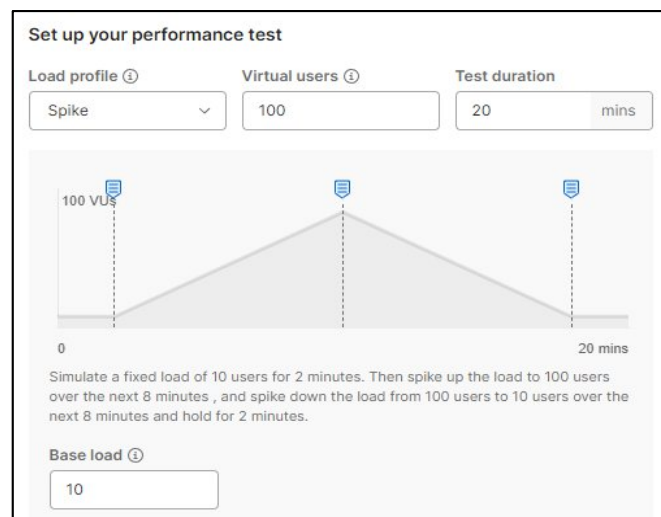
Amacom-back - Run results					
Ran today at 04:09:56 · View all runs					
Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	localhost	100	9m 49s	6600	101 ms
All Tests Passed (6592) Failed (8) Skipped (0)					
Iteration 6					
GET Search persons					
http://amacom.eastus2.cloudapp.azure.com/api/person/query?query=&page=0					
FAIL Response time is less than 300ms AssertionError: expected 301 to be below 300					

7.2.2 Pruebas de rendimiento con PostMan

Para analizar el comportamiento del software bajo condiciones de estrés se realizó una prueba del sistema donde se tuvo un pico de carga de 100 usuarios, donde al empezar y finalizar se tendrían 20 usuarios simultáneos. La prueba tuvo una duración de 20 minutos en los que se realizaron peticiones HTTP a diferentes endpoints del sistema sin establecer un límite.

Figura 51

Configuración de la prueba de estabilidad



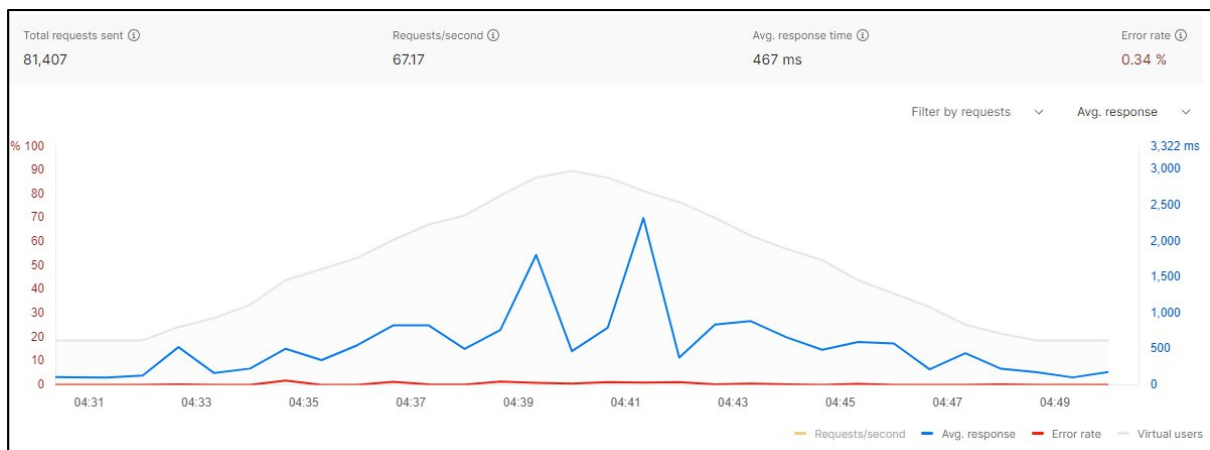
Como se puede observar en la siguiente imagen, se realizaron un total de 81407 peticiones HTTP durante su duración a un promedio de 67.17 peticiones por segundo. Como

SISTEMA DE INFORMACIÓN AMACOM

resultado se obtuvo un margen de error del 0.34% y un tiempo de respuesta promedio de 467 ms (milisegundos).

Figura 52

Resumen de la prueba de rendimiento y estabilidad



8.Conclusiones

Al finalizar este proyecto es posible evaluar el cumplimiento del objetivo de desarrollar un prototipo de sistema de información web y móvil para el acompañamiento de padres en el proceso de posparto. El desarrollo de la aplicación y el sitio web ejecutado en este proyecto se planeó en varias situaciones, etapas y módulos que cubren varias necesidades de tal manera que el acceso a la información logre un apoyo de calidad, oportuno y accesible en cualquier horario, a su vez son plataformas atractivas y fáciles de usar dado que era uno de los fines de este desarrollo.

El uso de sistemas de información que automatizan procesos que se realizaban de forma tradicional, plantea ventajas claras que están ligados a la digitalización de la información como eficiencia y calidad en el manejo y análisis de datos. En el contexto del proyecto, las posibles entidades que usarían este sistema verían mejoras claras en la recopilación y gestión de información en el desarrollo de su operación en la asistencia a padres en el proceso de posparto.

El desarrollo del proyecto planteó grandes desafíos para los integrantes del proyecto, donde se pusieron a prueba los conocimientos de ingeniería adquiridos durante lo largo de la carrera. Si bien la utilización de los conceptos sobre el ciclo de vida del software y su desarrollo fueron claves para lograr su culminación, se hizo indispensable el desarrollo de habilidades de análisis, investigación y programación para la solución de los diferentes desafíos que se encontraron. La verificación de la efectividad en el desarrollo se dió mediante la aplicación e implementación de diferentes pruebas que validaron el cumplimiento de requerimientos en los diferentes módulos del software. Cabe aclarar que fue un proceso arduo de retroalimentación que concluyó en un prototipo que cumple con los requerimientos planteados en la fase de diseño.

SISTEMA DE INFORMACIÓN AMACOM

La oportunidad de crecimiento que nos dió el desarrollo de este proyecto fue enriquecedora en el ámbito profesional y personal. Se aprendieron y pusieron en práctica conocimientos sobre el diseño y desarrollo de software mediante el uso de diferentes tecnologías, herramientas y prácticas de trabajo que facilitan en gran manera esta labor. Sin embargo, a nuestro juicio el mayor aprendizaje está relacionado a las habilidades blandas necesarias para el trabajo en equipo. La cooperación y coordinación que se necesita para afrontar los diversos desafíos que plantea el trabajar con otras personas no es para nada despreciable y es la diferencia entre el éxito o el fracaso de cualquier proyecto.

9.Trabajo a futuro

Como trabajo a futuro, se contempla la expansión del software para incorporar un módulo de citas. Este nuevo componente permitirá a las pacientes solicitar consultas, al mismo tiempo que brinda a enfermería la capacidad de asignarlas de manera eficiente.

Adicionalmente, se proyecta la implementación de un módulo de comunidad para madres, con el objetivo de facilitar la creación de grupos para charlas, actividades, convivencias y compartir experiencias. Esta iniciativa tiene como propósito mejorar el proceso postparto al proporcionar un espacio en el que las madres puedan conectar y apoyarse mutuamente durante esta etapa.

Asimismo, se llevará a cabo la expansión de la aplicación móvil mediante la incorporación de las funcionalidades mencionadas anteriormente y además la integración del módulo de enfermería donde permitirá al personal realizar acciones de manera efectiva a través de la app móvil.

Referencias

- Anderson. (2015). Docker [Software engineering]. IEEE Software, 32(3), 102–c3. [Doi.org](#)
- Eguíluz, J. (2009). Introducción a JavaScript. Obtenido de [N9](#)
- Fernández S, Moreno M, & Cañaveras R. (2013). Percepción de la transición a la maternidad: estudio fenomenológico en la provincia de Barcelona. Aten Primaria. 2013 [Revisado 03-01-2024] Disponible en: [Science Direct](#).
- Fowler, M. (2004). Refactoring: improving the design of existing code. Addison-Wesley Professional.
- Hartson, R. and Pyla, P.S. (2018) Libro: The UX Book: Agile UX Design for a Quality User Experience.
- Mead, A. (2018). Learning node.js development: Learn the fundamentals of node.js, and deploy and test node.js applications on the web. Packt Publishing, Limited.
- Pressman, R. (2010). Ingeniería del software Ingeniería del software: un enfoque práctico (3ed.ed.) McGraw-Hill.
- Process. ISBN 978-013704329, Primera Edición, 26 de julio del 2012.
- Sagar P. (2012). In: Transcultural nursing theory and models education, practice and administration. 1ª Ed. New York: springer; 2012. p. 2-13.
- Schildt, L. (2022). Java (9th ed.). McGraw-Hill Education.
- Schwaber. K (2012) Libro: Essential Scrum: A Practical Guide to the Most Popular Agile
- Sommervill, I. (2006). Ingeniería del software (7ed. ed.). Pearson Educación.
- Spinellis. (2012). Git. IEEE Software, 29(3), 100–101. [Doi.org](#)
- Sutherland J. & Schwaber. K (2020) The Scrum Guide Revisado el 8 de enero del 2024 [Scrum Guides](#)

SISTEMA DE INFORMACIÓN AMACOM

Vargas C, Hernandez L & Molina M. (2020) Aspectos percibidos por las madres primerizas como favorecedores en la adopción de su nuevo rol [Revisado 03-01-2024] [Scielosp](#)

Vargas. C., Cárcamo. E, Ferré. C. Molina M. (2021). Amacompri information system for postpartum nursing monitoring. Ciencia E Innovación En Salud. [Doi.org](#)

Apéndices.

Apéndice A

En este anexo se tiene la ruta al repositorio del backend del proyecto, el cual contiene todo el desarrollo realizado durante el proyecto y documentación. [Repositorio Backend en GitHub](#)

Apéndice B

En este anexo se tiene la ruta al repositorio prototipo web, el cual contiene todo el desarrollo realizado durante el proyecto y documentación. [Repositorio prototipo Web en GitHub](#)

Apéndice C

En este anexo se tiene la ruta al repositorio del prototipo móvil, el cual contiene todo el desarrollo realizado durante el proyecto y documentación. [Repositorio prototipo App Móvil en GitHub](#)

Apéndice D

En este anexo se tiene una ruta al archivo relacionado a la base de datos para una inspección más detallada de la misma. [Archivo diseño de base de datos del proyecto](#)

Apéndice E

En este anexo se encuentra una ruta a la colección de endpoints y pruebas realizadas en PostMan. Adicionalmente cada endpoint posee una documentación generada por la aplicación que puede servir de guía para su comprensión. [Colección de endpoints y pruebas usando PostMan](#)

Apéndice F

En este anexo se encuentra una ruta a el manual de instalación para los diferentes apartados del software. [Manual de instalación del prototipo](#)

Nota: Los apéndices están adjuntos y puede visualizarlos en la base de datos de la biblioteca UIS.