



**PLATAFORMA DE SOFTWARE WEB BASADA EN TELEMEDICINA COMO
APOYO A LA FUNDACIÓN CASA DE LA MUJER DE BOYACÁ, PARA LA
GESTIÓN DE LA INFORMACIÓN SOBRE EL MALTRATO DE GÉNERO**

PROPONENTE

Santiago Andrés Rodríguez Garcés
CC. 1005650954
Cod. 2251094

DIRECTOR

Iván Fernando Fonseca Barinas

ASESOR

Mery Yolima Uribe Rios

Tunja
20 de noviembre del 2021

CONTENIDO

FICHA TÉCNICA DEL PROYECTO	5
PLANTEAMIENTO DEL PROBLEMA	7
JUSTIFICACIÓN	9
OBJETIVOS	10
OBJETIVO GENERAL	10
OBJETIVOS ESPECÍFICOS	10
MARCO TEÓRICO	11
TELEMEDICINA	11
VENTAJAS DE LA TELEMEDICINA	12
BENEFICIOS PARA EL PACIENTE	12
VENTAJAS PARA EL INSTITUCIÓN DE SALUD	13
CONCEPTO DE VIOLENCIA DE GÉNERO	13
TIPOS DE VIOLENCIA	14
APLICACIÓN WEB	16
TIPOS DE APLICACIONES WEB	16
METODOLOGÍA	18
6.1. FASE 1: PLANIFICACIÓN:	20
ACTIVIDAD F1.1: ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS	20
6.2. FASE 2: DISEÑO	21
ACTIVIDAD F2.1: DISEÑO Y CONFIGURACIÓN ENTORNO DE PRUEBAS	21
6.3. FASE 3: CODIFICACIÓN	21
ACTIVIDAD F3.1: DESARROLLO DE LOS MÓDULOS DE LA PLATAFORMA	21
6.4. FASE 4: PRUEBAS	22
ACTIVIDAD 4.1: PRUEBAS DE ACEPTACIÓN	22
SE REALIZAN LAS PRUEBAS DE ACEPTACIÓN CLIENTE USUARIO.	22
EQUIPO DE TRABAJO	22
CRONOGRAMA DE ACTIVIDADES	23
DESARROLLO DEL PROYECTO	25
8.1. FASE 1. ANÁLISIS DE REQUISITOS	25

8.1.1. ANÁLISIS DE HISTORIAS DE USUARIO	25
8.1.2. CLASIFICACIÓN DE HISTORIAS DE USUARIO	27
8.2. FASE 2. DISEÑO Y ARQUITECTURA	30
8.2.1. DISEÑO DE LA BASE DE DATOS	30
8.2.2. DISEÑO DE LA ARQUITECTURA	32
TECNOLOGÍA BACKEND	32
VELOCIDAD Y SENCILLEZ	32
SEGURIDAD	33
ORM (MAPEADOR RELACIONAL DE OBJETOS)	33
ESCALABLE Y CONFIABLE	33
8.2.3. MÓDULOS	35
8.2.4. INTERFAZ DE USUARIO	36
8.3. FASE 3. IMPLEMENTACIÓN	38
8.3.1. DESARROLLO DEL BACKEND	38
8.3.1.1 BUENAS PRÁCTICAS DE PROGRAMACIÓN	39
8.3.1.2. PRUEBAS	42
8.3.2. DESARROLLO DEL FRONTEND	44
TECNOLOGÍA FRONTEND	44
MODULARIDAD	44
RENDERIZADO RÁPIDO	45
INTERFACES DE USUARIO RICAS Y FÁCILES DE PROBAR.	45
PRESUPUESTO	51
RESUMEN DEL PRESUPUESTO	51
RECURSO HUMANO	51
RECURSO MATERIAL Y/O EQUIPOS	52
CONCLUSIONES	52
REFERENCIAS	54
ANEXOS	58
ANEXO 1. ACTAS DE REUNIONES	58
ANEXO 2. HISTORIAS DE USUARIO	58
ANEXO 3. ARTÍCULO INVESTIGATIVO	58
ANEXO 4. DIAGRAMA ENTIDAD-RELACIÓN BASE DE DATOS	58
ANEXO 5. MOCKUPS DISEÑO INTERFAZ	58
ANEXO 6. MANUAL DE USUARIO	58

ANEXO 7. DESCRIPCIÓN DE SOFTWARE	58
Anexo 8. Diccionario de Datos Base de Datos	58

ÍNDICE DE TABLAS

Tabla 1 Objetivos específicos	10
Tabla 2 Fase 1. Análisis de historias de usuario	20
Tabla 3 Fase 2. Diseño y Arquitectura	24
Tabla 4 Explicación Paleta de Colores	26
Tabla 5 Fase 3. Implementación	33
Tabla 6 Cronograma de actividades	36
Tabla 7 Resumen del presupuesto	38
Tabla 8 Recurso humano	38
Tabla 9 Recurso material y/o equipos	39

ÍNDICE DE ILUSTRACIONES

Ilustración 1 Diagrama de pescado que ilustra las causas y efectos de la situación problema	8
Ilustración 2 Ciclo de vida de la metodología XP	11
Ilustración 3 Historia de Usuarios sin Numeración	17
Ilustración 4 Mockup vista chat	17
Ilustración 5 Historias de Usuario en GitKraken	18
Ilustración 6 Historias de Usuario Numeradas	19
Ilustración 7 Mockup Vista Solicitar cita	19
Ilustración 8 Diseño de la base de Datos	22
Ilustración 9 Arquitectura de Software Propuesta	23
Ilustración 10 Paleta de Colores	26

Ilustración 11 Pantalla Iniciar Sesión	27
Ilustración 12 Aplicación del principio de Single Responsibility en la clase Appointment.	30
Ilustración 13 Aplicación de la herencia de clases de acuerdo con el principio Open-Close	30
Ilustración 14 Ejecución exitosa de pruebas unitarias	32
Ilustración 15 Repositorio en GitHub del código fuente del backend del software.	34
Ilustración 16 Repositorio en GitHub del código fuente del frontend del software.	34
Ilustración 17 Repositorio en GitHub del README del backend del software.	35
Ilustración 18 Repositorio en GitHub del README del frontend del software.	35

1. FICHA TÉCNICA DEL PROYECTO

Título	Plataforma de software web como apoyo a la Fundación Casa de la Mujer de Boyacá, para la gestión de la información sobre el maltrato de género
Nombre Estudiante	Santiago Andrés Rodríguez Garcés
Documento estudiante	1005650954
Correo electrónico	santiago.rodriguezg@usantoto.edu.co
Director	Iván Fernando Fonseca Barinas
Asesor del Proyecto/ Profesor investigador	Mery Yolima Uribe Rios
Lugar de ejecución del proyecto	Universidad Santo Tomás Tunja
Duración	10 meses
Costo	\$27.615.000
Palabras claves	Gestión citas, Violencia de género, comunicación

Los abajo firmantes confirman que todos los datos incluidos en la presente propuesta son correctos y verídicos, que no incumplen ninguna ley o norma vigente (incluir nombres y firmas de estudiantes y director).



Firma del autor
Santiago Andrés Rodríguez Garcés



Firma del director
Iván Fernando Fonseca Barinas



Firma del asesor
Mery Yolima Uribe Rios

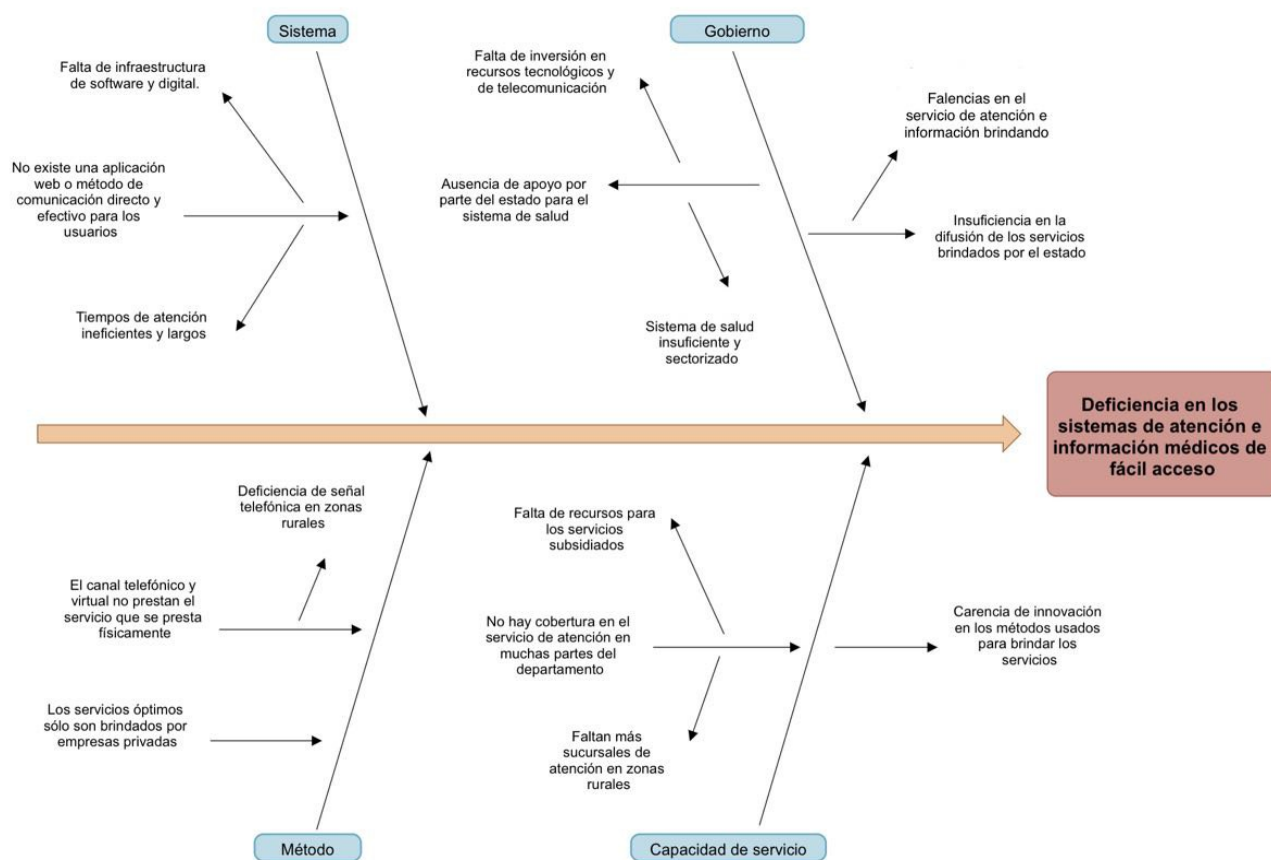
2. PLANTEAMIENTO DEL PROBLEMA

En el departamento de Boyacá se han reportado durante los últimos trimestres de los años 2017, 2018 y 2019, un total de 57 notificaciones sobre casos de maltrato hacia la mujer por semana (Plan de Desarrollo territorial Boyacá, 2020). Esto evidencia que el comportamiento de denuncias ha incrementado, ya sea por violencia sexual, psicológica, física o verbal, donde Tunja se posiciona como la zona con más recurrencia de estos conflictos. La Fundación Casa de la Mujer es el ente encargado de recibir estos casos para guiar a las mujeres en el reconocimiento de sus derechos y servir como mecanismo de apoyo por medio del diálogo. Sin embargo, esta misma carece de una herramienta tecnológica que permita tener un control y seguimiento adecuado a los casos de violencia.

En la actualidad existe una deficiencia en los sistemas de atención e información médicos de fácil acceso, dado que no hay alguno que permita dar respuesta ni atención a posibles inquietudes o casos relacionados con problemas de maltrato físico, familiar y psicológico, que se pueda usar de manera gratuita y sencilla. Ya que la mayoría de los sistemas de telemedicina o atención que ofrecen un servicio oportuno y de calidad en el país son de empresas privadas y se requiere pagar para adquirirlos [1]. Los servicios subsidiados por el estado o brindados por fundaciones sin fines de lucro (espacios físicos de información, asesoría y atención de calidad para las mujeres en toda su diversidad, con el fin de promover el acceso a apoyo social y de salud, generalmente desde una perspectiva de derechos y género) aunque hacen una gran labor referente a crear planes estratégicos y abarcar algunas ciudades principales, a menudo se enfrentan a una distribución inequitativa de recursos humanos entre las regiones, poca capacidad institucional en los territorios, escasa autonomía de prestadores de servicios de salud y exclusión del Plan Obligatorio de Salud (Suarez Roso, 2017). Esto genera problemas como la falta de recursos para llegar a áreas de difícil acceso o para brindar atención temprana y oportuna. Asimismo abre la puerta a la inseguridad en la vida de muchas mujeres que ni siquiera tienen la oportunidad de hacerse oír, de denunciar o recibir atención.

A continuación, se muestra el diagrama de espina de pescado que muestra las causas y efectos de la situación problema.

Ilustración 1 Diagrama de pescado que ilustra las causas y efectos de la situación problema



Fuente: Autor

3. JUSTIFICACIÓN

Según información aportada por la Fundación Casa de la Mujer [2], no se puede tener una cobertura total sobre las diversas denuncias que se presentan a diario en el departamento y en la ciudad de Tunja, es así como la Fundación Casa de la Mujer con colaboración de la UPTC, crean un espacio físico en donde las mujeres con diferentes problemáticas llegan para recibir apoyo y orientación, recopilando de esta manera denuncias sobre agresiones y buscando un profesional adecuado e idóneo para que les colabore en seguir un tratamiento adecuado y óptimo, pero es en este punto donde no se tiene una cobertura total, debido a que no se cuenta con una herramienta actualizada que haga la captura de la información y así permita tenerla disponible haciendo que sea ágil la respuesta de las denuncias recibidas.

El software basado en telemedicina podrá ayudar a cubrir esta problemática, permitiendo que la mujer logre realizar su solicitud disminuyendo el tiempo y procedimientos [3], o si desea contactarse con un profesional que la pueda atender en tiempo real evitando el desplazamiento a las oficinas, así mismo facilita que los profesionales de la casa de la mujer puedan tener mayor acceso a las constantes inquietudes que presentan las mujeres.

La implementación de este software le permitirá a la fundación Casa de la Mujer mejorar las condiciones del servicio creando un canal de comunicación efectivo[4], ampliando la cobertura total de usuarias, facilitando el acceso en la prestación de servicios a la población que presenta limitaciones geográficas, llegando a prestar el servicio no solo en la ciudad de Tunja sino también a otros municipios, aumentando la capacidad de consultas diarias y brindando una respuesta en un tiempo más oportuno. Los

profesionales de la casa de la mujer podrán obtener un registro de información de casos digitalizado para llevar un adecuado seguimiento y control[5].

4. OBJETIVOS

4.1. Objetivo General

Desarrollar una plataforma orientada a la web para la Fundación de la Casa de la Mujer que optimice la gestión de citas de los pacientes de forma remota incluyendo chat de comunicación.

4.2. Objetivos específicos

Tabla 1 Objetivos específicos

Nro.	Objetivo específico
1	Realizar el diseño de la base de datos para definir la estructura adecuada de la información a través de Power Designer.
2	Desarrollar la plataforma web cumpliendo con requerimientos dados por la Fundación Casa de la Mujer utilizando las buenas prácticas de programación orientada a objetos.
3	Aplicar las pruebas de calidad para identificar posibles fallos utilizando pruebas unitarias por cada funcionalidad.

Fuente: Autor

5. MARCO TEÓRICO

En este capítulo se presentan los fundamentos teóricos, en los cuales se basó el desarrollo de este trabajo, tales como los asociados a telemedicina, violencia de género y herramientas usadas en la elaboración de la plataforma.

5.1. Telemedicina

Básicamente, según la Asociación y Telemedicina Estadounidenses (ATA), la telemedicina se define como el intercambio de información médica entre dos ubicaciones diferentes utilizando la comunicación electrónica para la salud y educación del paciente o proveedor de atención médica y con el objetivo de mejorar la atención al paciente (Vilorio, 2009, pág.335).

En este sentido, para González (2013), la telemedicina corresponde a los servicios de información a disposición de los profesionales y pacientes en el tratamiento de emergencias o cuando sea necesario. En concreto, en cuanto a los servicios de tratamiento, se mencionan las siguientes alternativas:

Servicios relacionados con la atención al paciente y relacionados con el diagnóstico:

- Telemonitorización: se caracteriza por el seguimiento o seguimiento de los pacientes, cuya información se recoge en centros remotos para su análisis permanente (Segrelles, 2013).
- Telediagnóstico: corresponde al diagnóstico realizado por un especialista en base a información (imágenes e historial médico) recibida a través de Internet. El propósito es aclarar los desacuerdos sobre el estado de salud del paciente.

- Teleconferencia: corresponde a videoconferencias llevadas a cabo entre profesionales médicos, quienes tienen la responsabilidad de hacer un seguimiento y cuidado a los pacientes, pero sin la presencia de este (Soriano & Martín, 2010).

5.1.1 Ventajas de la telemedicina

La telemedicina aporta múltiples y diferenciados beneficios no solo a los pacientes y médicos, que generalmente son los principales actores de la telemedicina, sino también a los médicos remitentes, las instituciones de salud y la sociedad en general.

5.1.1.1 Beneficios para el paciente

Un informe de 2010 de los estados de la Organización de Estados Americanos (OEA). Los beneficios que el paciente puede obtener de un sistema de telemedicina son los siguientes:

- Los tiempos de respuesta se minimizan: esto permite un diagnóstico y tratamiento más eficientes, lo que puede conducir a la detección temprana de una enfermedad catastrófica, por ejemplo.
- La capacidad de obtener una segunda opinión. Los pacientes pueden ser examinados por diferentes médicos si no están completamente seguros de un diagnóstico inicial o desean confirmarlo.
- Atención continua y más personalizada. Esto hace posible brindar atención repetida a pacientes en áreas remotas.
- Se consigue una mejora en la calidad del servicio asegurando noticias y diagnósticos científicos.
- Se reduce el número de exámenes adicionales. Es conveniente un diagnóstico adicional, ya que no requiere investigaciones innecesarias.
- Reducción del estrés. La capacidad de tratar a los pacientes en el lugar los libera del estrés de tener que trasladarse.

- Acceso a especialistas. El paciente puede ser atendido por un especialista (OEA, 2010).

5.1.1.2 Ventajas para el Institución de Salud

Los beneficios de la telemedicina para las entidades de salud se pueden resumir de la siguiente manera:

- Mayor alcance. Hay más oportunidades para ampliar la cartera de servicios, así como el nivel de cobertura de los pacientes.
- Incremento de la productividad. Este aumento se produce a través de una mayor atención a las horas de servicio y la posibilidad de que los médicos estén disponibles las 24 horas del día como especialistas.
- Este es un factor que lo diferencia de la competencia. Con la telemedicina, tiene una ventaja competitiva sobre otras instalaciones porque puede brindar atención integral al paciente.
- Evita o verifica el traslado de pacientes. La telemedicina permite que la instalación mejore el seguimiento de los pacientes, incluida la priorización de ciertos casos complejos que requieren transferencia para proteger la vida.
- Integración de información con fines administrativos, de investigación y científicos.
- La sistematización de la información a través de las TIC es un elemento fundamental de la telemedicina, ya que los datos de pacientes y especialistas están actualizados.
- Reducción de costos de transporte. El tratamiento oportuno por parte de un especialista en áreas remotas o de difícil acceso no solo mejora la calidad de vida del paciente, sino que también contribuye a una reducción significativa de los costos de transporte (González, 2015).

5.2. Concepto de Violencia de Género

Según el artículo 1 de la "Declaración sobre la eliminación de la violencia contra la mujer" de la Conferencia de las Naciones Unidas en Viena de 1993, la violencia de género se define como "cualquier acto de violencia basado en la asociación con el sexo femenino que cause o pueda causar daño o sufrimiento físico, mental o sexual a las mujeres, incluidas amenazas de tales actos y coacción o privación arbitraria de la libertad, ya sea en la vida pública o en la vida privada".

Y la violencia contra la mujer incluye " la violencia física, sexual y psicológica en la familia, incluidas las palizas, el abuso sexual de las niñas en el hogar, la violencia relacionada con la dote, la violación por parte del marido, la mutilación genital y otras prácticas de violencia tradicional contra la mujer, la violencia perpetrada por personas distintas del marido y la violencia relacionada con la explotación. Física , violencia sexual y psicológica a nivel de la comunidad en general, incluida la violación, el abuso sexual, el acoso y la intimidación sexual en el lugar de trabajo, instituciones profesionales y otros entornos, trata de mujeres y prostitución forzada; y la violencia física, sexual y perpetrada o tolerada por el Estado dondequiera que ocurra.

Con esta definición, la violencia de género se conceptualiza y se refiere como un atentado a la integridad, dignidad y libertad de las mujeres, independientemente del entorno en el que se produzca. Hace relativamente poco tiempo, la violencia sexista se vio como un problema familiar y se consideró un problema de entrada en el interior. Las mujeres sufrieron violencia en silencio y familiares, amigos o vecinos ignorados, a pesar de que sabían lo que estaba pasando.

5.2.1. Tipos de Violencia

La violencia contra la mujer puede manifestarse de diversas formas. Los diversos estudios e informes han distinguido tres tipos de violencia de género: violencia física, violencia

psicológica y violencia sexual. Sin embargo, el informe del Grupo de especialistas en violencia contra las mujeres del Consejo de Europa (1997) distingue hasta seis tipos de violencia. Estos son:

- Violencia física: este tipo de violencia es el más evidente e incluye cualquier acto intencional y agresivo que cause o pueda causar daño al cuerpo de la mujer, con o sin consecuencias visibles. Este comportamiento incluye golpes, bofetadas, empujones, tirones de pelo, patadas, quemaduras, mordiscos, asfixia, asfixia, mutilación genital, uso de armas, inmovilizaciones, abortos, torturas.
- Violencia sexual: todo atentado a la libertad sexual de una mujer en el que se ve obligada a tolerar o realizar actos sexuales en contra de su voluntad, aprovechando una posición de poder mediante el engaño, la coacción, las amenazas o la fuerza. Esta actitud va desde el uso de bromas y bromas sexuales, comentarios desagradables, exhibicionismo, proposiciones sexuales no deseadas, toques no deseados para violación, incesto, embarazo forzado, trata y explotación en la industria del sexo.
- Violencia psicológica o emocional: actos, generalmente verbales o económicos, que causan o pueden causar daños psicológicos a las mujeres en su capacidad de toma de decisiones. Incluye el uso de mecanismos de control y comunicaciones que atenten contra su integridad psicológica, bienestar, autoestima o estima en público y en privado ante los demás. Se trata de actos u omisiones que abarcan un amplio abanico de situaciones, que van desde el desprecio, agresión verbal sostenida, humillación, gritos, humillaciones, amenazas, coacciones, insultos .
- Violencia económica: acceso desigual a recursos compartidos, incluye destrucción o privación de manutención y propiedad, control de recursos económicos, negación de manutención en caso de separación o divorcio, negación de derechos de propiedad, prevención de acceso al trabajo, educación o salud .
- Violencia estructural: está relacionada con la violencia económica, pero También incluye barreras intangibles e invisibles que impiden el acceso de las mujeres a los derechos básicos. Incluye la negación de la información inherente a los derechos

fundamentales y las relaciones de poder que los mantienen subordinados en los centros educativos, de decisión o laborales. Se refiere a una forma de violencia no explícita, sin golpizas, insultos o amenazas. Pero aquello que pone a la mujer en una situación en la que tiene un acceso desigual a los recursos o que le imposibilita su desarrollo personal.

- Violencia espiritual: destrucción de las creencias culturales o religiosas de una mujer al castigarla, ridiculizarla u obligarla a adoptar un sistema de creencias ajeno al suyo. Incluye la subyugación e invisibilización de las creencias culturales o religiosas de las mujeres o el análisis desde una perspectiva etnocéntrica.

5.3. Aplicación web

Para (López, 2015), una aplicación web es "una herramienta que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o una intranet mediante un navegador".

Por otro lado, (Silva, 2018) define la aplicación web como "un programa que se puede almacenar en un servidor remoto y ejecutar en un navegador". Una aplicación web es un sitio web que contiene páginas indeterminadas, parciales o completas. El contenido final de la página no se menciona hasta que el usuario solicita una página al servidor web.

5.3.1. Tipos de aplicaciones web

- Aplicaciones web para el comercio electrónico: Este tipo de aplicaciones web es una tienda online, similar a las aplicaciones de comercio móvil o un sitio web de comercio electrónico. Cualquier tipo de aplicación o sitio web que se ocupe de transacciones puede causar problemas porque involucra muchos métodos de pago como PayPal o tarjetas de crédito, además de otras funcionalidades.
- Aplicaciones web estáticas: Las aplicaciones web estáticas suelen ser las más simples, ya que tienen un contenido limitado y no son lo suficientemente flexibles.

Las carteras personales y los currículums digitales son dos ejemplos de aplicaciones web estáticas. Como de costumbre, estos se crean en CSS y HTML; sin embargo, existen otras plataformas que se pueden utilizar para desarrollarlos, como jQuery y Ajax.

- Aplicaciones web dinámicas: Las aplicaciones web dinámicas son un poco más complejas técnicamente en comparación con las aplicaciones estáticas. Utilizan una base de datos para cargar la información y el contenido se actualiza constantemente cada vez que un usuario accede a él. Los sistemas de gestión de contenido, que vienen con las veintiuna aplicaciones dinámicas, son responsables de este proceso. De esta manera, los usuarios administrativos pueden editar fácilmente el contenido y agregar características adicionales como foros y bases de datos.
- Aplicaciones web animadas: Esta es una de las tecnologías más utilizadas por diseñadores y creativos. La desventaja de desarrollar aplicaciones web animadas es que no es la más adecuada para problemas de posicionamiento web y optimización SEO, ya que los motores de búsqueda no pueden leer los datos correctamente.
- Aplicaciones web con un sistema de gestión de contenido: Para mantener el contenido actualizado, debe instalar un Administrador de contenido (CSM) para que el administrador pueda realizar cambios y actualizaciones él mismo.

A continuación, presentamos los gestores más intuitivos y fáciles de usar:

- WordPress: es sin duda el más utilizado entre los gestores de contenido. Hay mucha información en la web, tutoriales y guías para personalizarla, entenderla y además es gratis.
- Joomla: Es el segundo mejor CMS después de WordPress. Pero no tiene tantos usuarios pero una comunidad sólida.
- Drupal: Es un CMS de software libre. Es muy personalizable y está especialmente recomendado para crear comunidades (Silva, 2018).

6. METODOLOGÍA

Para el desarrollo de la plataforma propuesta se ha elegido la metodología de desarrollo ágil programación extrema (Extreme Programming - XP) la cual promueve el trabajo en equipo, el aprendizaje de los desarrolladores, y propicia un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios.[6]

La metodología XP se complementará con la metodología SCRUM de la que se tomarán algunos aspectos que ayudarán a tener un mayor control sobre el proyecto (cambios de requerimientos, entregas parciales y regulares), y una implementación más eficiente [7]. A continuación, se muestra el ciclo de vida de la metodología XP

Ilustración 2 Ciclo de vida de la metodología XP



Fuente: <https://images.app.goo.gl/fCZUdwUzqAXVooBH8>

Se tomaron en cuenta los siguientes aspectos para la selección respectiva de la metodología:

- **Tiempo para el desarrollo:** El tiempo es limitado por lo que se necesita desarrollar la plataforma con rapidez.
- **Planificación flexible y abierta:** Con el objetivo de un desarrollo ágil y gestión del proyecto con eficacia y control.
- **Comunicación con el cliente:** Se mantiene una comunicación constante entre el cliente y el equipo de desarrollo garantizando que el cliente participe de los avances del proyecto y se cumplan con los requisitos.

La planificación del proyecto se realizó de la siguiente manera:

- **6.1. FASE 1: Planificación:**

Se realizó el análisis de los documentos generados por el equipo de profesionales de la Fundación Casa de la Mujer, en especial lo relacionado con las variables y matrices de riesgo, base fundamental de la plataforma para la clasificación de pacientes y posterior toma de decisiones, en esta fase se llevarían a cabo las siguientes actividades:

- **Actividad F1.1: Análisis y Especificación de Requisitos**

Actividad en la cual, el equipo de analistas e Ingenieros de software, propone la documentación base para el desarrollo de la plataforma, compuesta por:

- Documento de Especificación de Requisitos.
- Definir el alcance que tendrá la plataforma a ser desarrollada.
- Documento de historias de usuario que describen las características y funcionalidad del software a realizar. Estas historias de usuario serán clasificadas y ordenadas mediante un valor de prioridad y costo medio de desarrollo que a su vez permitirán elaborar y actualizar el plan de ejecución y desarrollo por iteraciones. Por cada iteración se establece una fecha de entrega para ayudar a estimar una fecha de entrega total del proyecto. Se debe tener en cuenta que durante el desarrollo se pueden agregar o modificar las historias de usuario por lo que se modificará el cuadro de planificación.
- Usar herramientas como Trello para organizar y priorizar las historias de usuario que estarán en el Sprint (SCRUM).

Entregables:

- Documento de Especificación de Requisitos e historias de usuario. ([Ver Anexo 2](#))
- **6.2. FASE 2: Diseño**
 - **Actividad F2.1: Diseño y configuración entorno de pruebas**
 - Diagrama de Base de Datos (Entidad / Relación).
 - Prototipos del sistema.
 - Elección de lenguajes de programación.
 - Configuración de servidor para entorno de pruebas.

Entregables:

- Diagrama de Base de Datos (Entidad / Relación). ([Ver Anexo 4](#))
- Prototipos del sistema. ([Ver Anexo 5](#))
- **6.3. FASE 3: Codificación**
 - **Actividad F3.1: Desarrollo de los módulos de la plataforma**

De acuerdo a la documentación generada de la actividad F1.1, se llevó a cabo el desarrollo de cada uno de los módulos planteados, teniendo en cuenta las siguientes sub actividades:

 - Desarrollo de módulo de acuerdo a lineamientos del documento de especificación de requisitos.
 - Definición y ejecución de pruebas unitarias y de validación.
 - Especificación de excepciones encontradas/solucionadas (Retroalimentación).

Entregables:

- Software con el módulo de funcionalidades definidas para la entrega.

- **6.4. Fase 4: Pruebas**

- **Actividad 4.1: Pruebas de aceptación**

- Se realizan las pruebas de aceptación cliente usuario.

Por cada iteración se realizan las fases anteriores hasta que se complete el desarrollo de la plataforma.

- **Equipo de trabajo**

A continuación, se indica los roles de los miembros del equipo que desarrollaran el presente proyecto teniendo en cuenta los roles de equipo de la metodología XP:

- Un **desarrollador/tester** quién es el encargado de definir los documentos de requisitos, historias de usuario basadas en la documentación aportada por la Fundación Casa de la Mujer, desarrollar la plataforma web y aplicar las pruebas unitarias.
- Un **director** encargado del asesoramiento y orientación continuo tanto para el equipo de trabajo como para el cliente, facilita recursos para las reuniones sobre avances del proyecto, hacer cumplir las fechas planteadas para las entregas, realizar pruebas de aceptación final.

[illegible]

[illegible]

8. DESARROLLO DEL PROYECTO

A continuación, se detalla cada una de las actividades que tienen cada fase del proyecto.

- **8.1. Fase 1. Análisis de requisitos**

El desarrollo de la fase de análisis de requisitos estuvo compuesto por dos actividades con el objetivo de comprender los requisitos que debía cumplir el software.

- **8.1.1. Análisis de historias de usuario**

El primer acercamiento fue a través de una serie de entrevistas con la directora de la fundación, se realizaron preguntas sobre el funcionamiento actual de la fundación, los actores involucrados y los diferentes procesos que se llevan a cabo en los servicios que presta. Se analizaron los documentos elaborados por los profesionales de la Fundación Casa de la Mujer, especialmente los relacionados con las variables y matrices de riesgo que forman la base fundamental de la plataforma para la clasificación de pacientes y posterior toma de decisiones.

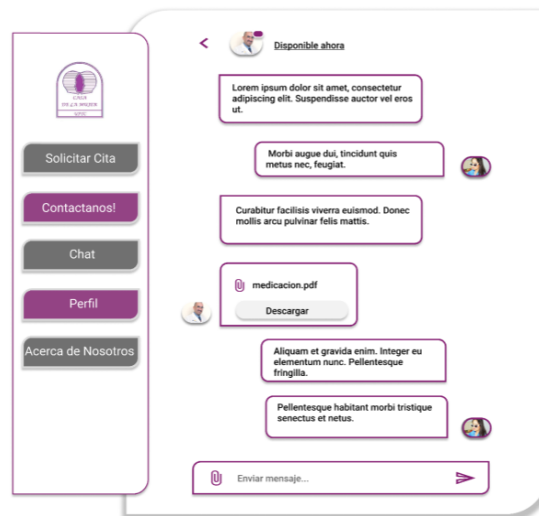
Luego de reunir la información, se realizó y socializó el documento de historias de usuario dispuesto en un documento de Google Docs. En el documento se listan las historias de usuario y adjunto se realizó el primer diseño de mockups (diseños de interfaz de usuario de baja fidelidad) que el desarrollador frontend debía dar cumplimiento. [Ilustración 2, Ilustración 4]

Ilustración 3 Historia de Usuarios sin Numeración

Gestión de Chat

- Como usuario quiero ver la lista de chats a los que pertenezco para poder acceder a ellos.
 - El usuario sólo puede ver los chats de los que es parte
 - Cómo usuario quiero poder interactuar en los chats que tengo para poder comunicarme con los profesionales.
 - El mensaje debe tener una hora de envío.
 - Como profesional de la salud quiero buscar chats por nombre para poder ubicar un usuario rápidamente
 - El nombre del usuario debe coincidir con un usuario registrado en la base de datos y tener un chat previo, de no ser así, se mostrará un mensaje indicando que el usuario existe o no tiene chat previo.
 - Se deben listar los chats cuyo nombre contengan la palabra de búsqueda ingresada por el usuario
- Si el nombre de algún chat concuerda con la palabra ingresada por el profesional de la salud, se mostrará en la pantalla.

Ilustración 4 Mockup vista chat



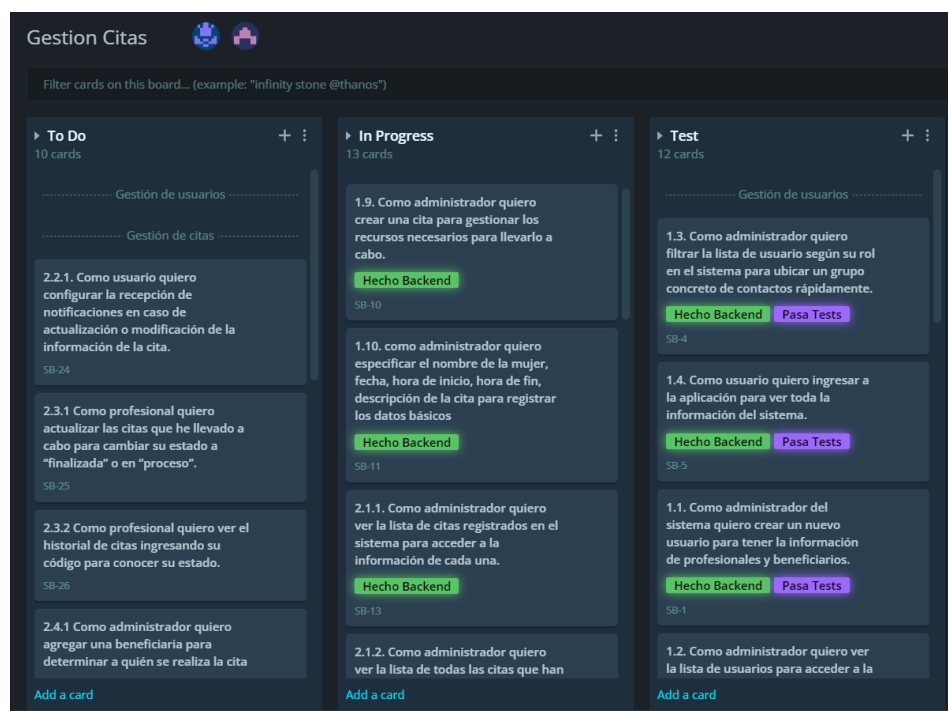
Como compromiso se determinó realizar la lectura y verificación de cada una de las historias contenidas en el documento para comprender los requisitos funcionales que

el software debía cumplir para luego en la siguiente fase (diseño y arquitectura) ser verificadas en el diseño de la base de datos.

■ 8.1.2. Clasificación de historias de usuario

Se dispuso un tablero compartido en la aplicación GitKraken donde se organizaron las historias de usuario en diferentes columnas para facilitar su gestión [Ilustración 5] bajo la metodología SCRUM.

Ilustración 5 Historias de Usuario en GitKraken



Se realizó la identificación y organización de cada una de las historias de usuario dividiéndolas en tres principales módulos para gestionar de manera eficiente la verificación de documento inicial que contenía mayor detalle de cada una por lo que se enumeraron todas.[Ilustración 6, Ilustración 7]

Ilustración 6 Historias de Usuario Numeradas

←

- Historias de Usuario Fundaci...
- Gestión de Usuarios
- 2. Gestión de citas**
- 2.1. Creación de citas, listado...
- 2.2 Seguimiento de citas
- 2.3. Historial de citas
- 2.4. Integrantes citas, respon...
- 3. Gestión de Chat

2. Gestión de citas

2.1. Creación de citas, listado, búsqueda

2.1.1. Como administrador quiero ver la lista de citas registradas en el sistema para acceder a la información de cada una.

2.1.2. Como administrador quiero ver la lista de todas las citas que han sido programadas para poder llevar el control de ellas.

2.1.3. Como beneficiario quiero solicitar una cita para poder ser atendido.

- El usuario debe llenar correctamente el formulario correspondiente
- Debe escoger una fecha disponible
- Debe escribir una pequeña descripción del caso o situación
- Debe escoger que tipo de atención necesita

Ilustración 7 Mockup Vista Solicitar cita



Solicitar Cita

Contactanos!

Chat

Perfil

Acerca de Nosotros



Solicitar Cita

Nombre
Nombre Apellido

Apellidos
Primer, Segundo, Tercero

Identificación
1234567890

Ejemplo
Primer, Segundo, Tercero, Cuarto, Quinto, Sexto

Datos Personales
Primer, Segundo, Tercero, Cuarto, Quinto, Sexto

Descripción
Algunos ejemplos de problemas: dolor de estómago, fiebre, tos, etc.

Tipo de Cita
☒ Jurídica ☐ Psicosocial

Audio
medicacion.mp3

Explorar

Solicitar Cita

A continuación, se muestran las entradas, instrumentos/herramientas, entregables y tiempo de la fase de análisis de historias de usuario.

Tabla 2 Fase 1. Análisis de historias de usuario

Tipo	Elementos
Entradas	• Documentos de historias de usuario
Instrumentos y/o herramientas	• Computador • Conexión a internet • Navegador Web • Cuenta de Google (acceso a la carpeta compartida)
Entregables	Documento de historias de usuario
Tiempo	1 semana

Fuente: Autor

- **8.2. Fase 2. Diseño y Arquitectura**

El desarrollo de la fase de diseño y arquitectura estuvo compuesto por dos actividades: el diseño de la base de datos y el diseño de la arquitectura.

- **8.2.1. Diseño de la base de datos**

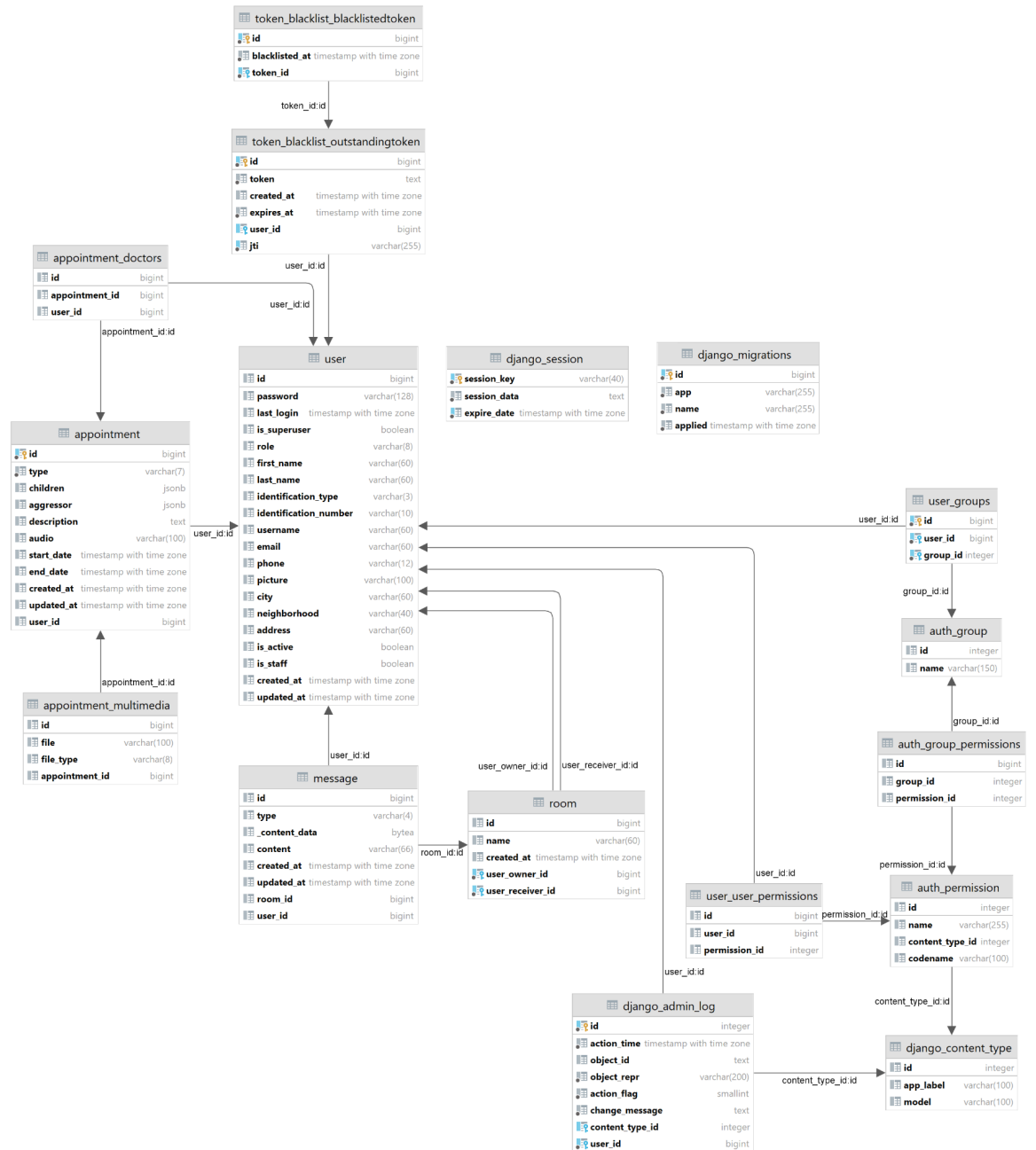
A continuación se presenta el diseño entidad-relación de la base de datos, está compuesto por las siguientes tablas:

- user
- room
- token_blacklist_oustandingtoken
- token_blacklist_blacklistedtoken
- message
- appointment
- appointment_doctors
- appointment_multimedia
- user_groups

Finalmente, se listan las tablas creadas automáticamente por el framework Django:

- Gestión de usuarios, grupos y permisos de estos:
 - auth_group
 - auth_permission
 - auth_group_permissions
 - django_content_type
 - django_session

Ilustración 8 Diseño de la base de Datos



Fuente: Autor

Respecto al motor de base de datos se eligió PostgreSQL. Esta decisión se realizó debido a que este motor de base de datos es relacional, gratuito, seguro, ampliamente usado en el mundo empresarial, además cuenta con buena documentación y soporte. Además permite almacenar datos grandes y sofisticados de forma segura.

■ 8.2.2. Diseño de la arquitectura

Se determinó emplear un lenguaje de programación y framework que facilita la implementación y desarrollo ágil de este tipo de APIs (API REST) por lo que se usó el lenguaje de programación Python y el framework Django.

En la página oficial de Django este se define como:

Django es un framework web de alto nivel en Python que fomenta el desarrollo rápido y el diseño limpio y pragmático. Construido por desarrolladores experimentados, se encarga de gran parte de las molestias del desarrollo web, para poder centrarte en escribir la aplicación sin necesidad de reinventar la rueda. Es gratuito y de código abierto (Django Framework, 2021).

8.2.2.1. Tecnología Backend

Los factores que se tuvieron en cuenta para elegir este framework son:

- **Velocidad y Sencillez**

Uno de los principales objetivos de Django es simplificar el trabajo a la hora de desarrollarlo. Para hacer eso, el marco de Django usa:

- Los principios de desarrollo rápido, lo que significa que cuando se desarrolla se pueden hacer más de una iteración a la vez sin comenzar todo el programa desde cero.
- Filosofía DRY: (don't repeat yourself) , lo que significa que su arquitectura nos permite y nos invita a reutilizar el código existente y centrarse en el único.

- **Seguridad**

Una de las mejores cosas de Django es que puede construir la aplicación a una velocidad más rápida y entregarla sin comprometer la seguridad de la aplicación. Las funciones de seguridad están habilitadas de forma predeterminada en este marco. Tiene protección incorporada para algunos problemas de seguridad comunes, como secuencias de comandos entre sitios, falsificación de solicitudes, clickjacking e inyección SQL. Django lanza nuevos parches de seguridad con bastante frecuencia y responde inmediatamente a las vulnerabilidades de seguridad y alerta a otros marcos.



- **ORM (mapeador relacional de objetos)**

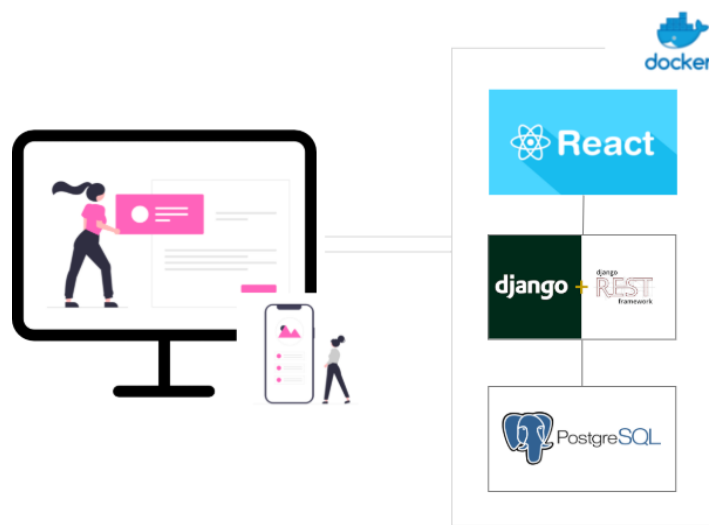
Django ofrece un ORM completamente funcional, compatible con varias bases de datos. Básicamente, ORM es una biblioteca que ayuda a los desarrolladores a interactuar con las bases de datos y les permite transferir los datos automáticamente desde las bases de datos (como MySQL y PostgreSQL) a los objetos. Puede trabajar con varias bases de datos a la vez.

- **Escalable y confiable**

Django viene con una serie de componentes cableados, listos para funcionar por defecto. Estos componentes están desacoplados, por lo que, según el requisito o las soluciones específicas de una aplicación, el desarrollo se puede ampliar o reducir reemplazando o desconectando los componentes.

La Ilustración 9 muestra las principales tecnologías de software a utilizar en términos de infraestructura.

Ilustración 9 Arquitectura de Software Propuesta



La plataforma está empaquetada en contenedores Docker: un contenedor para el motor de base de datos de Postgres, un contenedor para la aplicación backend que ejecuta Python, Django y daphne, y finalmente un contenedor para la aplicación React que impulsa la experiencia visual de los usuarios.

Este tipo de arquitectura tiene las siguientes ventajas:

- Generación de una aplicación portátil que se puede implementar fácilmente en entidades que necesitan el servicio.

- Evitar errores causados por dependencias de aplicaciones y mejorar la seguridad utilizando solo puertos y servicios obligatorios.

Tabla 3 Fase 2. Diseño y Arquitectura

Tipo	Elementos
Entradas	• Documentos de historias de usuario
Instrumentos y/o herramientas	• Computador • Conexión a internet • Navegador Web • Cuenta de Google (acceso a la carpeta compartida) • Power Designer
Entregables	Documento de historias de usuario
Tiempo	3 semanas

Fuente: Autor

■ 8.2.3. Módulos

Los módulos de esta plataforma corresponden a su funcionalidad en base con la arquitectura descrita. Los módulos son:

Módulo de registro de usuarios: hay tres tipos de usuarios en la plataforma:

- **Administrador:** tiene la capacidad de registrar la información del personal de La Casa de la Mujer y verificar la información de las citas de los usuarios.
 - **Empleado:** un empleado profesional de La Casa de la Mujer tiene la capacidad de actualizar su información personal y cambiar su contraseña. También puede concertar una cita o chatear con un usuario.
 - **Usuario:** utiliza los servicios de La Casa de la Mujer, solicita asesoría y cita en la plataforma. Una vez registrado, el usuario puede acceder y editar su información personal.
- **Módulo de solicitud de cita:** un usuario registrado tiene la opción en la plataforma de solicitar una cita mediante un formulario donde se especifique el

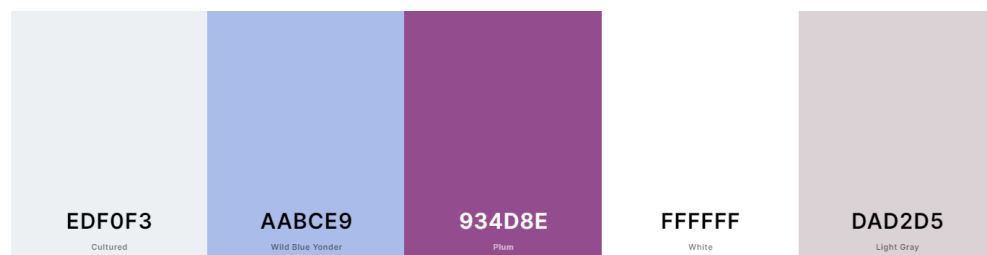
tipo de asesoría (jurídica, psicosocial o ambas) para obtener ayuda con un caso específico.

- **Módulo de chat:** El objetivo de este módulo es brindar a los usuarios una respuesta inmediata y asesoramiento sobre temas relacionados con la violencia de género. Un usuario registrado puede ingresar a una sala de chat y desde allí un profesional de La Casa de la Mujer puede responder a las inquietudes de los usuarios de manera oportuna.

■ 8.2.4. Interfaz de usuario

Partiendo de la arquitectura de la Ilustración 9 y basándose en algunas recomendaciones del Laboratorio de Diseño de Políticas Tecnológicas: Violencia y Abuso de Género en Línea [9], la interfaz de usuario de la plataforma fue adaptada por Ant Design de React [10]. La paleta de colores utilizada (ver Ilustración 10) se eligió en base con el logo de La Casa de la Mujer. Este logo tiene dos colores principales: morado y blanco. Para complementarlos, se especificó un gris claro, un azul suave y un azul suave claro.

Ilustración 10 Paleta de Colores



Esos colores se definieron por el significado y las sensaciones que evoca el color en su uso [11] - [13]:

Tabla 4 Explicación Paleta de Colores

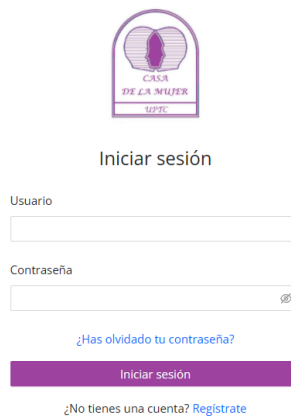
Color	Función
Morado	Evoca creatividad. Las mujeres prefieren el morado, el verde, el rojo y el azul.
Blanco	Evoca pureza, inocencia y positividad. Ayuda a limpiar el diseño
Azul	Evoca calma, confianza, honestidad.
Gris	Puede ser conservador y sofisticado. Al alterar su tonalidad, el gris puede adquirir las características del negro o del blanco.

Los colores se utilizaron para:

- Morado: botones, acciones actuales, títulos principales
- Fondo blanco
- Gris: resaltar selecciones, opción de acciones y menús
- Azul: enlaces importantes
- Azul claro: para resaltar los datos que faltan en un formulario o los errores

Un ejemplo del uso de los colores se expone en la Ilustración 11.

Ilustración 11 Pantalla Iniciar Sesión



The screenshot shows a login interface. At the top is a logo for 'CASA DE LA MUJER' with 'MPTC' below it. Below the logo is the text 'Iniciar sesión'. There are two input fields: 'Usuario' and 'Contraseña'. Below the 'Contraseña' field is a link that says '¿Has olvidado tu contraseña?'. At the bottom is a purple button labeled 'Iniciar sesión'. Below the button is a link that says '¿No tienes una cuenta? [Regístrate](#)'.

• 8.3. Fase 3. Implementación

Esta arquitectura se compone de dos frentes: Backend y Frontend. El desarrollo de la fase de implementación estuvo compuesto de las siguientes actividades:

■ 8.3.1. Desarrollo del Backend

Se modularizó el software en aplicaciones de Django que es la arquitectura propuesta por este framework. En la documentación oficial de Django (Django Framework, 2021), una aplicación describe un paquete de Python que proporciona

un conjunto de características y que estas aplicaciones se pueden reutilizar en varios proyectos.

La arquitectura de backend para el sitio web (contenedores Django y Postgres en la Ilustración 9) consta de los siguientes elementos:

- Los modelos: gestionan las relaciones entre las tablas de la base de datos y Python.
- Los serializadores: validan y serializan los datos entrantes y salientes.
- Conjuntos de consultas: crear, consultar y almacenar los resultados de las consultas de la base de datos como instancias de modelo.
- Vistas: son las clases que incluyen un modelo, un serializador y un conjunto de consultas para cada punto final.
- URL: especifica cuándo llamar a la vista.

Uno de los puntos más importantes en el backend de este sistema de salud es proteger los datos personales de los usuarios, por lo que esta información está encriptada en la base de datos. Los datos como la dirección, el número de teléfono, el apellido y el correo electrónico solo pueden ser vistos por los profesionales que trabajan en un caso.

Al comunicarse a través del chat, las conversaciones se cifran utilizando el estándar AES-256 con modo GCM a través de la biblioteca Pycryptodome[8]. Este protocolo evita que los intrusos accedan al canal de comunicación y comprendan los mensajes.

■ 8.3.1.1 Buenas prácticas de programación

Durante el desarrollo se aplicaron buenas prácticas de programación orientada a objetos, específicamente aplicando los principios SOLID. SOLID es un acrónimo inventado por Robert C. Martin para establecer los cinco principios básicos de la programación orientada a objetos y diseño. Este acrónimo tiene bastante relación con los patrones de diseño, en especial, con la alta cohesión y el bajo acoplamiento (Rubira, 2021). Los principios aplicados fueron (Ilustración 12, Ilustración 13):

- **S – Single Responsibility:** consiste en que cada clase debe tener una única responsabilidad, trabajo o propósito. Por ejemplo, la clase Appointment[Ilustración 10] tiene la única responsabilidad de guardar un objeto con las variables de un formulario.
- **O – Open-Close:** establece que una entidad de software (clase, módulo, función, etc.) debe quedar abierta para su extensión, pero cerrada para su modificación. Es decir, se debe poder extender el comportamiento de tal entidad, pero sin modificar su código fuente (Álvarez et al., 2021). Por ejemplo, la clase User[¡Error! No se encuentra el origen de la referencia.] extiende de las clases AbstractBaseUser y PermissionsMixin de Django.
- **L – Liskov Substitution:** las clases derivadas o secundarias deben ser sustituibles por sus clases base o primarias. Este principio garantiza que cualquier clase que sea hija de una clase principal se pueda utilizar en lugar de su principal sin ningún comportamiento inesperado (GeeksforGeeks, 2021). En este caso se cumplió con este principio por defecto sin presentarse comportamientos inesperados al extender clases de Django como es la clase Model.
- **I – Interface Segregation:** Es similar al principio de responsabilidad única. Afirma que no debemos obligar a ningún cliente a implementar una interfaz que sea irrelevante para ellos. El objetivo principal de este concepto es centrarse en evitar las interfaces pesadas y dar preferencia a muchas interfaces pequeñas específicas del cliente (GeeksforGeeks, 2021). No fue necesario la segregación de las clases debido al amplio uso del principio de responsabilidad única.
- **D – Dependency Inversion:** los módulos/clases de alto nivel no deberían depender de módulos/clases de bajo nivel, sino que deberían depender de abstracciones. También se debe asegurar de que la abstracción no debe depender de los detalles, sino que los detalles deben depender de las abstracciones (GeeksforGeeks, 2021). No hay ejemplo porque se refiere al uso de interfaces y en Python no existen las interfaces, solo la herencia simple y múltiple.

Ilustración 12 Aplicación del principio de Single Responsibility en la clase Appointment.

```
class Appointment(models.Model):
    """Appointment model. Represents a chat room"""

    APPOINTMENT_TYPE_CHOICES = [
        ('PSY', _('Psicosocial')),
        ('LEG', _('Jurídica')),
    ]

    user = models.ForeignKey('accounts.User', verbose_name=_('user'), related_name='patient', on_delete=models.CASCADE)
    doctors = models.ManyToManyField('accounts.User', verbose_name=_('doctores'), related_name='doctors')
    type = models.CharField(verbose_name=_('tipo de cita'), max_length=7)
    children = models.JSONField(verbose_name=_('datos de los hijos'), null=True, blank=True)
    aggressor = models.JSONField(verbose_name=_('datos del agresor'), null=True, blank=True)
    description = models.TextField(verbose_name=_('descripción'), null=True, blank=True)
    audio = models.FileField(verbose_name=_('audio'), upload_to='appointments/audio', validators=[
        FileExtensionValidator(allowed_extensions=['mp3', 'mp4', 'ogg', 'm4a'])
    ])
    start_date = models.DateTimeField(verbose_name=_('fecha de inicio'), null=True, blank=True)
    end_date = models.DateTimeField(verbose_name=_('fecha de finalización'), null=True, blank=True)
    created_at = models.DateTimeField(verbose_name=_('fecha de registro'), auto_now_add=True)
    updated_at = models.DateTimeField(verbose_name=_('fecha de actualización'), auto_now=True)

    class Meta:
        db_table = 'appointment'
        verbose_name = _('cita')
        verbose_name_plural = _('citas')

        permissions = [
            ('add_appointment_from_me', 'Can add my appointments'),
            ('change_appointment_from_me', 'Can change my appointments'),
            ('delete_appointment_from_me', 'Can delete my appointments'),
```

Ilustración 13 Aplicación de la herencia de clases de acuerdo con el principio Open-Close

```
class User(AbstractBaseUser, PermissionsMixin):
    """User model"""

    class Type(models.TextChoices):
        """User types"""

        ADMIN = 'ADMIN', _('Administrador')
        DOCTOR = 'DOC', _('Doctor')
        USER = 'USR', _('Usuario')

    class IdentificationType(models.TextChoices):
        CC = 'CC', _('Cédula de Ciudadanía')
        CE = 'CE', _('Cédula de Extranjería')
        NIT = 'NIT', _('Nit')
```

Además de aplicar los principios SOLID se usaron otras buenas prácticas de programación como el uso de nombres significativos en clases y métodos; métodos con una funcionalidad específica; reducción de declaraciones condicionales para disminuir la

complejidad del código mejorando la comprensión y mantenibilidad; documentación, reutilización y optimización de código; uso de la convención de sintaxis del lenguaje Python y pruebas unitarias.

■ 8.3.1.2. Pruebas

La prueba unitaria es un tipo de prueba de software en la que se testean unidades o componentes individuales de un software. El propósito es verificar que cada unidad de código de software funcione como se espera. Las pruebas unitarias aíslan una sección de código y verifican que funciona correctamente. Una unidad puede ser una sola función, método, procedimiento, módulo u objeto.

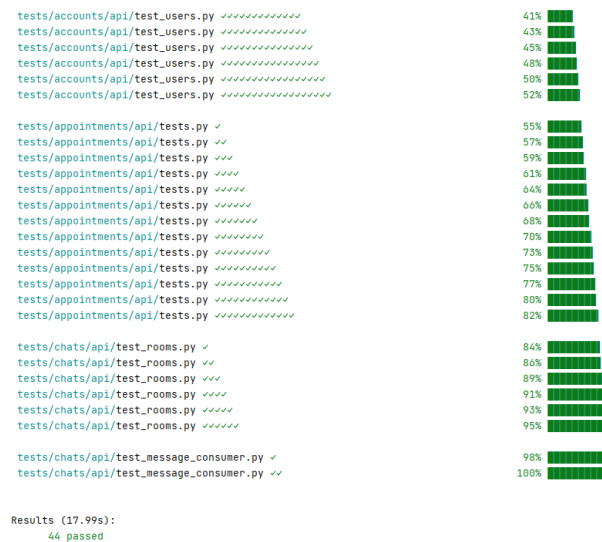
Las pruebas unitarias son otro proceso que se realiza como parte de una metodología de trabajo ágil. Debido a la naturaleza modular de las pruebas unitarias, podemos probar partes del proyecto sin esperar a que se completen otras. Este tipo de pruebas permite la detección temprana de errores que sin su aplicación se descubrirían sólo en etapas más avanzadas como el sistema, la integración e incluso las pruebas beta. Las pruebas unitarias generalmente ahorran tiempo y dinero (Surah, 2020)

Con el marco de prueba que proporcionan Python y Django, hay una gran cantidad de código repetitivo, los problemas de mantenimiento y duplicación aumentan a medida que crecen sus proyectos.

La herramienta de testeo que se escogió para la aplicación de pruebas unitarias es Pytest ya que proporciona una forma simple y elegante de escribir pruebas. Además brinda la capacidad de escribir pruebas como funciones, lo que significa que se ha eliminado una gran cantidad de código repetitivo, lo que hace que el código sea más legible y más fácil de mantener. Pytest también proporciona funciones para reconocer pruebas, definir y usar accesorios.

Se elaboraron pruebas unitarias de las funcionalidades de cada aplicación a medida que se iban desarrollando. En la siguiente Ilustración [Ilustración 14] se muestran los resultados finales de la ejecución de las pruebas obteniéndose un resultado exitoso del 100% de 44 pruebas existentes.

Ilustración 14 Ejecución exitosa de pruebas unitarias



■ 8.3.2. Desarrollo del Frontend

8.3.2.1. Tecnología frontend

La librería usada para el desarrollo del frontend fue React.js. Los factores que se tuvieron en cuenta fueron:

- **Modularidad**

La arquitectura de React permite agrupar código de front-end en componentes personalizados, proporcionando así una base sólida para la optimización general del proceso de desarrollo. La posición de React en la escala de obstinación es estable: la biblioteca no bloquea ni guía en su forma de hacer las cosas. Por un lado, dicta cómo debe crear componentes reutilizables. Por otro lado, da libertad para decidir cómo manejar el enrutamiento y las pruebas, organizar las dependencias. React.js permite escribir código modular limpio, dividiendo el proyecto en componentes separados. Como resultado, cuando el código es modular, se vuelve fácil de mantener y escalar.

En segundo lugar, React es flexible en el sentido de que puede integrarse con otras bibliotecas de terceros y marcos MVC. Puede optar por crear un proyecto de React desde cero o introducir gradualmente React en su base de código heredada existente. Crear una nueva aplicación React no es demasiado difícil. Incrustar componentes de React uno por uno en su código escrito usando otras tecnologías tampoco es un problema. Y viceversa: es relativamente fácil integrar bibliotecas de terceros en aplicaciones React.

- **Renderizado rápido**

React es una biblioteca de JavaScript del lado del cliente que utiliza la representación del lado del servidor. De esta manera brinda el soporte mejorado para la representación del lado del servidor y convierte a React en una solución robusta orientada al rendimiento, perfecta para crear aplicaciones centradas en el contenido, ya sea una aplicación web o aplicaciones nativas para Android e iOS.

- **Interfaces de usuario ricas y fáciles de probar.**

El desarrollo de front-end con React da como resultado no solo aplicaciones de alto rendimiento sino también altamente probables. React en sí mismo facilita la creación de una arquitectura limpia que sea fácil de probar.

La arquitectura frontend para el sitio web (contenedor React en la Ilustración 9) fue diseñada siguiendo buenas prácticas de programación y de acuerdo con los siguientes elementos:

- Config: contiene la configuración de la aplicación, como el envío de solicitudes al backend y funciones que se pueden llamar en cualquier parte de la aplicación.
- Componentes: elementos visuales como botones, formularios, tarjetas que se pueden utilizar en cualquier lugar de la aplicación.
- Páginas: toma varios componentes para crear páginas web, como la página de inicio, el inicio de sesión, el administrador de usuarios y el administrador de citas.
- Hooks personalizados: son un mecanismo para reutilizar la lógica de estado (como establecer una suscripción y guardar el valor actual).
- Servicios: una capa para recuperar o enviar información desde la aplicación back-end.
- Bibliotecas: una capa con una lógica para almacenar la información en la base de datos y manejar las fechas del sistema para un caso.
- Estilos: estilos CSS globales personalizados que se utilizan en componentes y páginas.
- Rutas: permiten configurar la navegación de las páginas de la aplicación, inicio, chat y solicitudes de citas.

A continuación, se muestran las entradas, instrumentos/herramientas, entregables y tiempo de la fase de Implementación.

Tabla 5 Fase 3. Implementación

Tipo	Elementos
------	-----------

Entradas	● Archivo .ldm del diseño entidad-relación de la base de datos
Instrumentos y/o herramientas	● Computador ● Conexión a internet ● Navegador Web ● Cuenta De Google (acceso a la carpeta compartida) ● Power Designer ● Git ● Postman
Entregables	Software el cual cumple con las necesidades y requerimientos especificados por la empresa
Tiempo	9 semanas

Fuente: Autor

Esta fase tiene como resultado el código fuente dispuesto en el repositorio remoto (GitHub).(Ilustración 15, Ilustración 16).

Ilustración 15 Repositorio en GitHub del código fuente del backend del software.

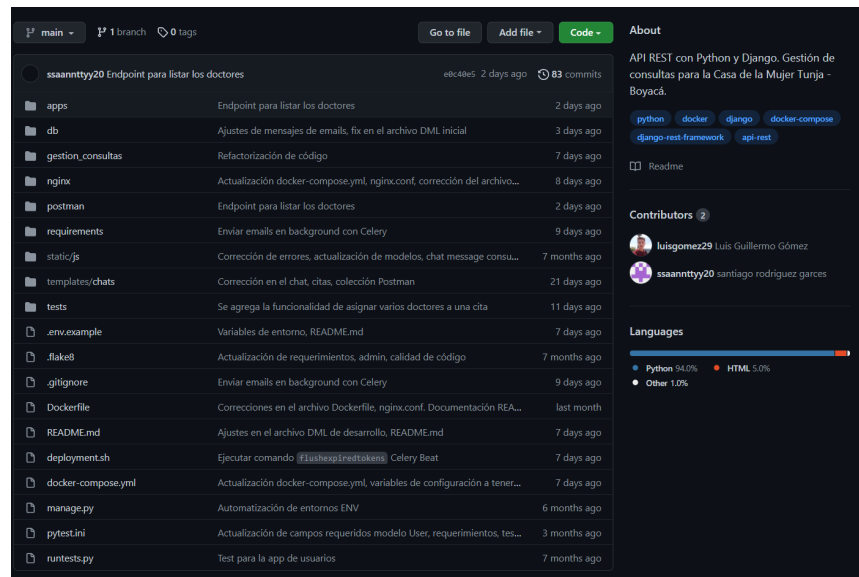
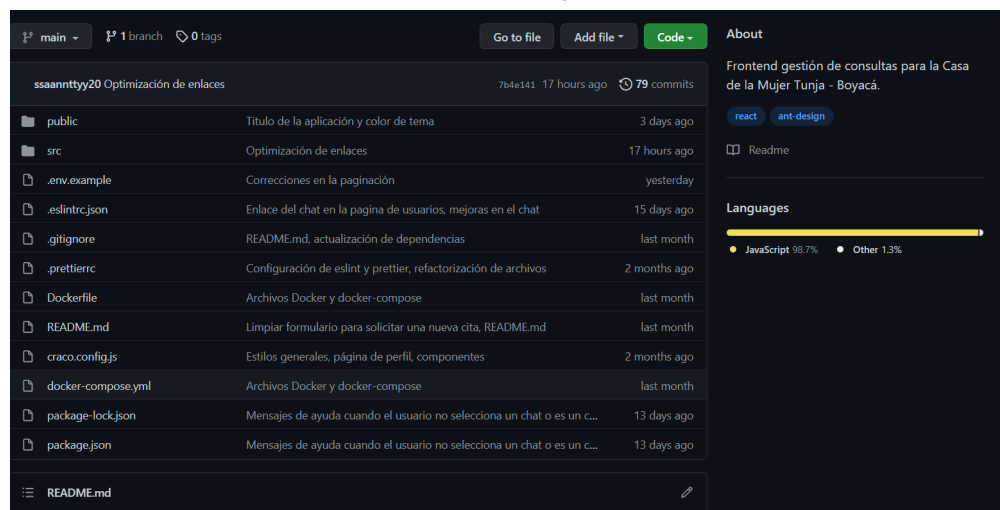


Ilustración 16 Repositorio en GitHub del código fuente del frontend del software.



Adicionalmente se documentó cómo realizar el despliegue del software en desarrollo y producción [Ilustración 17, Ilustración 18].

Ilustración 17 Repositorio en GitHub del README del backend del software.

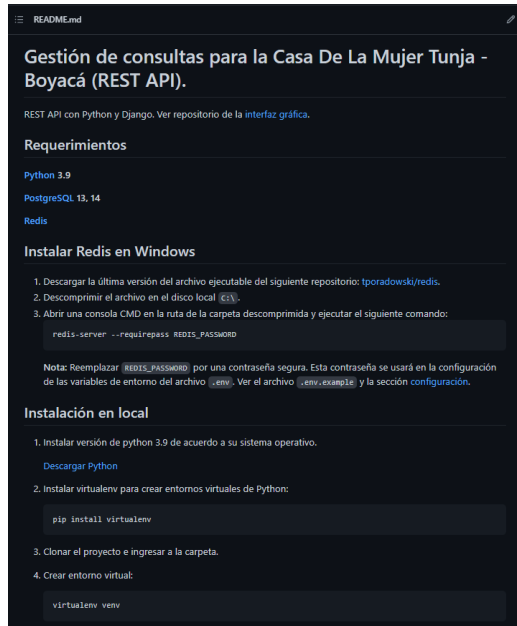
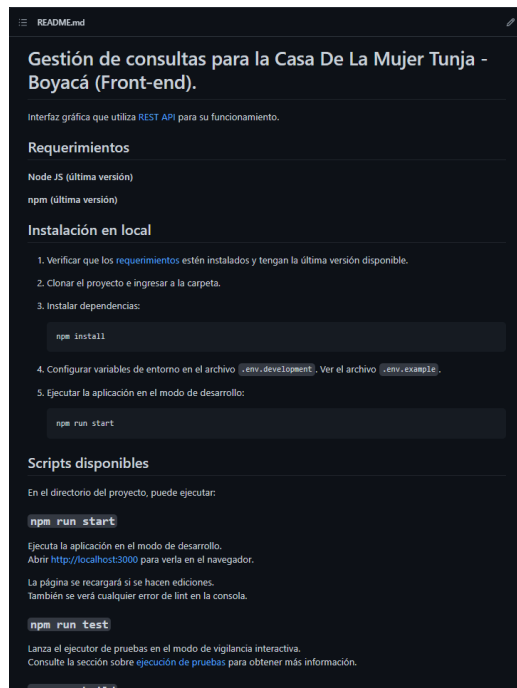


Ilustración 18 Repositorio en GitHub del README del frontend del software.



9. PRESUPUESTO

9.1. Resumen del presupuesto

Presupuestos de los recursos necesarios para desarrollar el proyecto.

Tabla 7 Resumen del presupuesto

Rubros	Fuentes de Financiación			Total
	Propias	Contrapartida	Otras ¹	
Recurso Humano	x			\$21.240.000
Recurso Material y/o Equipos	x			\$6.375.000
Total				\$27.615.000

Fuente: Autor

En las siguientes tablas se detallan los recursos necesarios para el proyecto en términos del recurso humano y recurso material.

9.2. Recurso Humano

A continuación, se indican los datos de los estudiantes y director de proyecto que trabajaran en el proyecto.

Tabla 8 Recurso humano

Nombre	Rol	Dedicación Horas	Valor Hora	Fuentes de Financiación			Total
				Propias	Contrapartida	Otras	
Santiago Andrés Rodríguez Garcés	Estudiante	576	\$20.000	x			\$11.520.000
Mery Yolima Uribe Ríos	Asesora	96	\$45.000	x			\$4.320.000
Iván Fernando Fonseca Barinas	Director	120	\$45.000	x			\$5.400.000
Total							\$21.240.000

Fuente: Autor

9.3. Recurso Material y/o Equipos

Tabla 9 Recurso material y/o equipos

Recurso	Justificación	Fuentes Financiación			Total
		Propios	Contrapartida	Otras	
3 computadores	Desarrollo de la plataforma.	X			\$5.400.000
Conexión a internet	Para investigar, y desarrollar la plataforma.	X			\$640.000
Hosting	Servidor donde se aloja la plataforma para publicarla en internet.	X			\$300.000
Dominio	Dirección de internet donde los usuarios podrán acceder a la plataforma.	X			\$35.000
Total					\$6.375.000

Fuente: Autor

10. Conclusiones

- Mediante la selección y uso de herramientas técnicas y de software específicas, se logró crear una plataforma funcional con tres módulos principales que cubren las necesidades de La Casa de la Mujer en cuanto a brindar sus servicios a todas las personas que necesitan asesoría psicosocial o judicial por una situación de violencia de género en la ciudad de Tunja, Colombia. Debido a la movilidad limitada, el acceso limitado a los servicios en persona y el contacto cara a cara limitado debido al Covid-19, La Casa de la Mujer ha enfrentado muchos problemas al brindar sus servicios. A través de la plataforma desarrollada, la fundación mejorará el acceso a sus servicios, la gestión de sus datos, la aceleración de sus servicios gracias a los datos gestionados y los módulos desarrollados, las estadísticas sobre el comportamiento del centro y las funciones de su personal y administradores.
- Conforme a el planeamiento estipulado, se logró de manera correcta el diseño de la base de datos, mediante la aplicación de distintos criterios como lo fueron: controlar la redundancia de la información, evitar pérdidas de información, corroborar la capacidad para representar toda la información, mantener la consistencia de los datos, seleccionar el tipo y tamaño de datos adecuado.
- Con base en los principios de buenas prácticas de programación empleados (SOLID), se dio cumplimiento a cabalidad y de manera efectiva a los requerimientos dados por la Fundación Casa de la Mujer. La plataforma abre a los usuarios el acceso a: los servicios, el personal profesional de La Casa de la Mujer, los datos de un caso y a un acompañamiento real, característica particularmente necesaria en estos casos de violencia.
- La realización y ejecución satisfactoria de las pruebas unitarias permitió comprobar de manera efectiva la correcta ejecución de cada funcionalidad, descartando así posibles fallos en la calidad de desempeño.
- Gracias a la arquitectura propuesta, la plataforma se desarrolló en poco tiempo y se mejoró su funcionalidad con una interfaz de usuario limpia y sencilla.

11. REFERENCIAS

- [1]Guerrero, R., Gallego, A., Becerril-Montekio, V. and Vásquez, J., 2011. Sistema De Salud De Colombia. [online]. Scielosp.org. Available at: <https://www.scielosp.org/article/spm/2011.v53suppl2/s144-s155/es/>
- [2]fundación casa de mujer, 2017. [Online]. Available: <https://www.casmujer.com/casamujer2017/%23quienes-somos>.
- [3]"Telehealth | CMS", Cms.gov, 2020. [Online]. Available: <https://www.cms.gov/Medicare/Medicare-General-Information/Telehealth/index>
- [4] Smith, A., 2010. Mobile Access 2010. [online] Pew Research Center: Internet, Science & Tech. Available: <http://www.pewinternet.org/Reports/2010/Mobile-Access>
- [5] Netux.com. 2020. Mi Paciente | Monitoreo Remoto De Paciente - Telemonitoreo. [online]. Available: https://www.netux.com/es/mipaciente?gclid=EAlaIQobChMI-5Can8T96gIVkP7jBx10UgN9EAAYAiAAEgJ-NPD_BwE
- [6] Letelier, P. and Penadés, C., 2006. Cyta. [online] Cyta.com.ar. Available: <http://www.cyta.com.ar/ta0502/v5n2a1.htm>
- [7] Viewnext. 2020. ¿Qué Ventajas Aporta Extreme Programming?. [online]. Available: <https://www.viewnext.com/ventajas-extreme-programming/>
- [8] Pycryptodome, "PyCryptodome." 2021, Accessed: Sep. 24, 2021. [Online]. Available: <https://pycryptodome.readthedocs.io/en/latest/>.
- [9] World Wide Web Foundation, "Tech Policy Design Lab: Online Gender-Based Violence and Abuse," 2021. <https://ogbv.webfoundation.org/> (accessed Oct. 11, 2021).

[10] Ant Desing, "Ant Design of React." 2021, Accessed: Sep. 24, 2021. [Online]. Available: <https://ant.design/docs/react/introduce>.

[11] Information is Beautiful, "Colours in Cultures," 2021. <https://informationisbeautiful.net/visualizations/colours-in-cultures/> (accessed Sep. 24, 2021).

[12] Neil Patel, "How to Use the Psychology of Color to Increase Website Conversions," 2021. <https://neilpatel.com/blog/psychology-of-color-and-conversions/> (accessed Sep. 24, 2021).

[13] Toptal, "The Role of Color in UX," 2021. <https://www.toptal.com/designers/ux/color-in-ux> (accessed Sep. 24, 2021).

[14] Django Framework. (10 de Mayo de 2021). Applications. Obtenido de <https://docs.djangoproject.com/es/3.2/ref/applications/#projects-and-applications>

[15] Django Framework. (10 de Mayo de 2021). Models. Obtenido de Documentation: <https://docs.djangoproject.com/en/dev/topics/db/models/>

[16] GeeksforGeeks. (17 de Junio de 2021). Best Practices of Object Oriented Programming (OOP). Obtenido de GeeksforGeeks: <https://www.geeksforgeeks.org/best-practices-of-objectoriented-programming-oop/>

[17] Suárez-Rozo LF, Puerto-García S, Rodríguez-Moreno LM, Ramírez-Moreno. La crisis del sistema de salud colombiano: una aproximación desde la legitimidad y la regulación. Rev Gerenc Polít Salud. 2017; 16 (32): <https://doi.org/10.11144/Javeriana.rgps16-32.cssc>

- [18] Viloria Núñez, C. (2009). Tecnologías de la información para la educación, investigación y aplicación en el área de la salud. Bondades y retos. Revista Salud Uninorte. 25(2), 331-349.
- [19] González Ambriz, S. (2013). Evaluación de capacidad de redes wimax para aplicaciones de telemedicina. México: Escuela Superior de Ingeniería en Mecánica y Eléctrica. Recuperado de tesis.ipn.mx/.../Evaluacion%20de%20capacidad%20de%20redes%20wimax.pdf
- [20] Segrelles, G. W. (2013). Telemonitorización domiciliaria en pacientes con enfermedad pulmonar obstructiva crónica grave. Madrid: Universidad Autónoma de Madrid. Departamento de Medicina. Recuperado el 10 de abril de 2017, de https://repositorio.uam.es/bitstream/handle/10486/13641/64115_segrelles%20calvo%20gonzalo%20walderico.pdf?sequence=1
- [21] Soriano, O. & Martín, L. (2010). Telemedicina: ¿futuro o presente?. Revista Cubana Habanera de Ciencias Médicas. 9(1), 127-139.
- [22] Organización de Estados Americanos (OEA). (2010). ¿Que es Telemedicina? Obtenido de Comisión Interamericana de Telecomunicaciones: 75 http://www.oas.org/es/citel/infocitel/2010/junio/salud-1_e.asp Redalyc. TELEMEDICINA. (Enero de 2007). Recuperado el 20 de Febrero de 2012, de <http://redalyc.uaemex.mx/redalyc/pdf/2611/261120984009.pdf>
- [23] González Aguilar, J. M. (2015). Estudio para la aplicación de la telemedicina en consultas y diagnósticos a distancia en el hospital Teodoro Maldonado Carbo de Guayaquil. Guayaquil: Universidad Católica Santiago de Guayaquil. Recuperado de <http://repositorio.ucsg.edu.ec/bitstream/3317/4115/1/T-UCSG-POS-MTEL-46.pdf>

[24] López, M. (20 de Enero de 2015). mialtoweb. Obtenido de <http://mialtoweb.es/definicion-de-aplicacion-web>

[25] Silva, H. (25 de 07 de 2018). ILMAISTRO.COM. Obtenido de ILMAISTRO.COM: <https://ilmaistro.com/tipos-aplicaciones-web/>

[26] V. (2021, 7 enero). 7 Reasons to choose React for your project in 2021. Nearshore Software Development Company in Ukraine - VILMATE. <https://vilmate.com/blog/why-choose-react-in-2021/>

[27] GeeksforGeeks. (2020, 30 agosto). Top 10 Reasons to Choose Django Framework For Your Project. <https://www.geeksforgeeks.org/top-10-reasons-to-choose-django-framework-for-your-project/>

[28] Surah. (2020) <https://www.velotio.com/engineering-blog/use-pytest-fixtures-with-django-models>

12. ANEXOS

■ Anexo 1. Actas de Reuniones

Enlace: <https://drive.google.com/drive/folders/1-zBW6vtmRZtU47jYJeP5sc6-obmKXbx?usp=sharing>

■ Anexo 2. Historias de Usuario

Enlace: <https://docs.google.com/document/d/13JYR2dHhJjVm2ulSu372W4oP3MKz789t/edit?usp=sharing&oid=112761501588383553733&rtpof=true&sd=true>

■ Anexo 3. Artículo investigativo

Enlace: https://docs.google.com/document/d/1RFJ35e_QDq1t4iqQD4bILPG3-fCnEoTE/edit?usp=sharing&oid=112761501588383553733&rtpof=true&sd=true

■ Anexo 4. Diagrama Entidad-Relación Base de Datos

Enlace: <https://drive.google.com/file/d/1i-7kt7xt4eW2Y6FUXAFIO86--1114M6X/view?usp=sharing>

■ Anexo 5. Mockups diseño Interfaz

Enlace: <https://drive.google.com/file/d/1XOVts0oZ4qi6R8eo04ipA76jupQZvsWA/view?usp=sharing>

■ Anexo 6. Manual de Usuario

Enlace: https://docs.google.com/document/d/11rouZsEFQT_PLDN8NIS-mNQSsxT7e6wr/edit?usp=sharing&oid=112761501588383553733&rtpof=true&sd=true

■ Anexo 7. Descripción de Software

Enlace: <https://docs.google.com/document/d/1wGSCSENqAKrzpToHTC2ur6V6CqFn-cyMb4m71ptRJQE/edit?usp=sharing>

■ Anexo 8. Diccionario de Datos Base de Datos

Enlace: <https://drive.google.com/file/d/1SCv9HLtDK7REvny99xOSyRh3l0IfUxh/view?usp=sharing>

