



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

IMPLEMENTACIÓN DE UN SISTEMA WEB PARA GENERACIÓN Y
RECEPCIÓN DE CITAS MÉDICAS E HISTORIAS CLÍNICAS
UTILIZANDO LA METODOLOGÍA SWIRL.

DIAZ HUAMAN JOHN BRYAN
INGENIERO DE SISTEMAS

MACHALA
2022



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

IMPLEMENTACIÓN DE UN SISTEMA WEB PARA GENERACIÓN
Y RECEPCIÓN DE CITAS MÉDICAS E HISTORIAS CLÍNICAS
UTILIZANDO LA METODOLOGÍA SWIRL.

DIAZ HUAMAN JOHN BRYAN
INGENIERO DE SISTEMAS

MACHALA
2022



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN
PROPUESTAS TECNOLÓGICAS

IMPLEMENTACIÓN DE UN SISTEMA WEB PARA GENERACIÓN Y RECEPCIÓN
DE CITAS MÉDICAS E HISTORIAS CLÍNICAS UTILIZANDO LA METODOLOGÍA
SWIRL.

DIAZ HUAMAN JOHN BRYAN
INGENIERO DE SISTEMAS

HONORES TAPIA JOOFRE ANTONIO

MACHALA, 20 DE SEPTIEMBRE DE 2022

MACHALA
2022

Tesis de John Díaz

INFORME DE ORIGINALIDAD

3%

INDICE DE SIMILITUD

3%

FUENTES DE INTERNET

0%

PUBLICACIONES

1%

TRABAJOS DEL
ESTUDIANTE

FUENTES PRIMARIAS

1

repositorio.utmachala.edu.ec

Fuente de Internet

3%

Excluir citas

Activo

Excluir coincidencias < 1%

Excluir bibliografía

Activo

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, DIAZ HUAMAN JOHN BRYAN, en calidad de autor del siguiente trabajo escrito titulado IMPLEMENTACIÓN DE UN SISTEMA WEB PARA GENERACIÓN Y RECEPCIÓN DE CITAS MÉDICAS E HISTORIAS CLÍNICAS UTILIZANDO LA METODOLOGÍA SWIRL., otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 20 de septiembre de 2022



DIAZ HUAMAN JOHN BRYAN
0706116936

DEDICATORIA

El presente trabajo de titulación se lo dedico en primer lugar a Dios, por ser pilar fundamental en mi vida, por guiarme, por darme fuerzas y sabiduría para siempre seguir adelante, a mis amados padres por siempre inculcarme los buenos valores, por haber forjado como la persona que soy hoy en día, por su apoyo incondicional en todos estos años de mi carrera universitaria. A mi enamorada que siempre supo apoyarme y aconsejarme en los momentos difíciles; por estar a mi lado prácticamente en toda mi carrera universitaria. A mis amigos que de otra forma me apoyaron de forma incondicional.

John Bryan Díaz Huamán

AGRADECIMIENTOS

En primer lugar, quiero agradecer a Dios por haberme dado todo este tiempo salud, vida y fortaleza para seguir adelante y seguir cumpliendo mis metas fijadas.

Agradecimiento infinito a mi familia, de forma muy especial a mis padres que siempre han estado ahí para mí, que con su amor y sacrificios ellos me han sabido apoyar en todo momento de mi carrera para que la pueda culminar de la mejor manera. A mis tíos que de una u otra forma siempre me han apoyado y estuvieron presente en momentos complicados. A mi enamorada que también me acompañó en momentos de difícil decisión durante mi carrera y supo apoyarme incondicionalmente en todo lo que estaba a su alcance.

Quiero dejar en constancia de un sincero agradecimiento a la Universidad Técnica de Machala, a todos mis docentes por importarme sus sabias enseñanzas y de manera muy especial al Ing. Joofre Honores Tapia, por el apoyo brindado en la dirección del presente trabajo de titulación.

John Bryan Díaz Huamán

RESUMEN

La aplicación de la tecnología web en el mundo de los negocios, empresas o en el ámbito educativo y muchos aspectos más ha tenido un crecimiento exponencial en las últimas décadas, es preciso mencionar que cada vez son más empresas de todo tipo de actividad; tales como las grandes industrias optan por crear sitios, aplicaciones o sistemas basados en la web con la finalidad de captar la atención de nuevos clientes o usuarios.

Conforme va avanzado la tecnología las empresas optan por la implementación de páginas web para dar a conocer sus productos o servicios ofertados; complementando en ciertas ocasiones con aplicaciones móviles con la colaboración y utilización correcta de herramientas que permiten la reutilización de código fuente, optimizando tiempo y esfuerzo a la hora del desarrollo de las mismas; esto quiere decir que permite crear un proyecto multiplataforma que será funcional tanto en un ordenador como en sistemas operativos de entorno móvil. Debido a estas características de los sistemas web, la demanda de los mismos han aumentado en gran volumen. Esto compromete a los desarrolladores a tener una actualización constante de conocimientos acerca de nuevas tecnologías como frameworks que permite la gestión de nuevos proyectos con buenas características como: usables, seguros, llamativos y capaces de satisfacer todas las necesidades de los clientes.

La presente propuesta tecnológica, implementa un sistema web para la gestión de generación y recepción de citas médicas e historias clínicas de pacientes del consultorio médico Kennedy, utilizando la metodología de desarrollo SWIRL; esta metodología permite automatizar los procesos de: gestión de turnos de citas médicas, gestión de usuarios, gestión de historias clínicas, gestión de fichas médicas, control de citas atendidas, informes de pacientes, informe de exámenes de laboratorios. Adicionalmente el sistema tiene la capacidad de interactuar con el paciente a través de un chatbot implementado.

Los requerimientos en los que se basa el sistema fueron otorgados por el personal que labora en el consultorio médico con quienes se mantuvo reuniones con el fin de recabar toda la información necesaria y dar ideas para su implementación. Además, este sistema está construido con tecnologías como servidor web apache, lenguaje de programación PHP y JavaScript, Framework Laravel, gestor de base de datos

MySQL sumando los estándares de diseño de HTML y CSS para constitución de las interfaces de usuario. Se seleccionó este tipo de tecnología en base a la necesidad de los usuarios del sistema lleguen a través de internet.

Una vez constituido el sistema, el proceso de evaluación se aplicó en dos partes, esto según lo mencionado en la metodología; la primera parte está asociada a un análisis de calidad tomando como referencia la normativa ISO/IEC 9126, y en su segunda etapa se la realiza mediante la aplicación de herramientas de evaluación en línea para verificar el SEO, performance, accesibilidad y usabilidad del sistema, como parte final de la propuesta, se menciona también el proceso de lanzamiento que es considerado un punto fundamental dentro de la metodología, en la cual el sistema es alojado en un servidor o hosting con su respectivo dominio para que este sea totalmente operable para todo el mundo con acceso a internet.

Palabras Claves: SWIRL, Aplicación Web, Agendamiento de Citas, Desarrollo de Software.

John Bryan Díaz Huamán

ABSTRACT

The application of web technology in the world of business, companies or in the educational field and many other aspects has had an exponential growth in recent decades, it is necessary to mention that there are more and more companies of all kinds of activity; such as large industries choose to create web-based sites, applications or systems in order to capture the attention of new customers or users.

As technology advances, companies opt for the implementation of web pages to publicize their products or services offered; complementing on certain occasions with mobile applications with the collaboration and correct use of tools that allow the reuse of source code, optimizing time and effort when developing them; this means that it allows you to create a multiplatform project that will be functional both on a computer and on mobile environment operating systems. Due to these characteristics of web systems, the demand for them has increased in great volume. This commits developers to have a constant update of knowledge about new technologies such as frameworks that allow the management of new projects with good characteristics such as: usable, safe, attractive and capable of satisfying all customer needs.

This technological proposal implements a web system for managing the generation and reception of medical appointments and medical records of patients at the Kennedy medical office, using the SWIRL development methodology; This methodology allows you to automate the processes of: medical appointment shift management, user management, clinical record management, medical record management, control of appointments attended, patient reports, laboratory test reports. Additionally, the system has the ability to interact with the patient through an implemented chatbot.

The requirements on which the system is based were granted by the staff that works in the medical office with whom meetings were held in order to gather all the necessary information and give ideas for its implementation. In addition, this system is built with technologies such as Apache web server, PHP and JavaScript programming language, Laravel Framework, MySQL database manager, adding HTML and CSS design standards for the creation of user interfaces. This type of

technology was selected based on the need for system users to reach it through the Internet.

Once the system was constituted, the evaluation process was applied in two parts, as mentioned in the methodology; The first part is associated with a quality analysis taking the ISO/IEC 9126 standard as a reference, and in its second stage it is carried out through the application of online evaluation tools to verify the SEO, performance, accessibility and usability of the system. As a final part of the proposal, the launch process is also mentioned, which is considered a fundamental point within the methodology, in which the system is to be hosted on a server or hosting with its respective domain so that it is fully operable for all users. world with internet access.

Keywords: SWIRL, Web Application, Appointment Scheduling, Software Development.

John Bryan Díaz Huamán

INDICE GENERAL

DEDICATORIA	1
AGRADECIMIENTOS	2
RESUMEN	3
ABSTRACT	5
INTRODUCCIÓN	12
1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS	14
1.1. Ámbito de aplicación: descripción de contexto y hechos de interés	14
1.2. Establecimiento de requisitos	16
1.3. Justificación de requerimientos a satisfacer	17
2. CAPÍTULO II. DESARROLLO DE PROTOTIPO	18
2.1. DEFINICIÓN DEL PROTOTIPO TECNOLÓGICO	18
2.1.1. Arquitectura de Software	18
2.1.2. Arquitectura de Angular	19
2.1.3. Arquitectura de Laravel	20
2.1.4. Arquitectura Cliente – Servidor	22
2.1.5. Definición del Sistema	24
2.2. FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO	33
2.2.1. Metodología	33
2.2.2. SWIRL: Metodología para desarrollo de aplicaciones web	35
2.2.3. Sistema Web	42
2.3. OBJETIVOS DEL PROTOTIPO	46
2.3.1. Objetivo General	46
2.3.2. Objetivos Específicos	46
2.4. DISEÑO DEL PROTOTIPO	46
2.4.1. Fase de Planificación	46
2.4.2. Fase de Modelado	53
2.5. EJECUCIÓN Y ENSAMBLAJE DE PROTOTIPO	94
2.5.1. Formularios de autenticación	94
3. CAPÍTULO III: EVALUACIÓN DEL PROTOTIPO	105
3.1. Plan de Evaluación	105
3.1.1. Evaluación de Calidad	106
3.1.2. Evaluación con Herramientas SEO	106
3.2. Resultados de Evaluación	107
3.2.1. Evaluación de Calidad	107
3.2.2. Evaluación mediante Herramientas SEO	110

3.3. CONCLUSIONES	113
3.4. RECOMENDACIONES	113
BIOGRAFÍA	114

INDICE DE TABLAS

Tabla 1: Beneficios Tangibles e Intangibles del Sistema	25
Tabla 2: Usuario de Sistema: Administrador (Médico)	25
Tabla 3: Usuario de Sistema: Enfermera	26
Tabla 4: Usuario de Sistema: Paciente	26
Tabla 5: Funcionalidades del Sistema	27
Tabla 6: Limitaciones del Sistema	28
Tabla 7: Estimación de Costo	29
Tabla 8: factibilidad Operativa de Sistema	29
Tabla 9: Factibilidad Técnica del Sistema	30
Tabla 10: Factibilidad Económica del Sistema	30
Tabla 11: Matriz de Stakeholders	31
Tabla 12: RF - Requerimientos Funcionales del Sistema	32
Tabla 13: RNF - Requerimientos No Funcionales del Sistema	32
Tabla 14: Pasos para correcto uso de Metodología	38
Tabla 15: Actividades de Fase de Análisis	39
Tabla 16: Historia de Usuario #1- RF-01	46
Tabla 17: Historia de Usuario #2- RF-02	47
Tabla 18: Historia de Usuario #3- RF-03	47
Tabla 19: Historia de Usuario #4- RF-04	48
Tabla 20: Historia de Usuario #5- RF-05	48
Tabla 21: Historia de Usuario #6- RF-06	49
Tabla 22: Historia de Usuario #7- RF-07	49
Tabla 23: Historia de Usuario #8- RF-08	50
Tabla 24: Historia de Usuario #9 - RF-09	50
Tabla 25: Historia de Usuario #10 - RF-10	51
Tabla 26: Historia de Usuario #11 - RF-11	51
Tabla 27: Cronograma - Iteración #1	52
Tabla 28: Cronograma - Iteración #2	53
Tabla 29: Escala de Likert en Evaluación de Calidad	106
Tabla 30: Evaluación de Calidad Interna - Externa	107
Tabla 35: Control de Cambios de Proyecto	122

ÍNDICE DE FIGURAS

Figura 1: Arquitectura de Angular	20
Figura 2: Patrón MVC del Prototipo	21
Figura 3: Proceso de Comunicación de Protocolo HTTP	23
Figura 4: Estimación de costos - COCOMO II	28
Figura 5: Fundamentación Teórica del Proyecto	33
Figura 6: Modelo SWIRL	37
Figura 7: Ciclo de Vida de Metodología SWIRL	38
Figura 8: Modelo Entidad- Relación del Sistema	54
Figura 9: Modelo Relacional del Sistema	54
Figura 10: Enlaces Navegacionales Generales del Sistema	55
Figura 11: Enlaces Navegacionales Usuario Paciente	56
Figura 12: Enlaces Navegacionales Usuario Paciente	56
Figura 13: Enlaces Navegacionales Usuario Paciente	57
Figura 14: Prototipo - Página Inicial	58
Figura 15: Prototipo - Página Inicial – 2	59
Figura 16: Prototipo - Página Acerca de Nosotros	60
Figura 17: Prototipo - Página Contacto	61
Figura 18: Prototipo - Página Noticias	61
Figura 19: Prototipo - Página Personal Médico	62
Figura 20: Prototipo - Página Login	63
Figura 21: Prototipo - Página Registro de usuario	63
Figura 22: Prototipo - Página Inicio - Usuario Paciente	64
Figura 23: Prototipo - Página Consultas - Usuario Paciente	64
Figura 24: Prototipo - Página Perfil usuario - Usuario Paciente	65
Figura 25: Prototipo - Exámenes Laboratorio - Usuario Paciente	66
Figura 26: Prototipo - Página Principal - Usuario Enfermera	66
Figura 27: Prototipo - Agendar Cita - Usuario Enfermera	67
Figura 28: Prototipo - Historia Clínica - Usuario Enfermera	67
Figura 29: Prototipo - Página Horas - Usuario Enfermera	68
Figura 30: Prototipo - Informe de Turnos - Usuario Enfermera	68
Figura 31: Prototipo - Página Lista de Citas - Usuario Enfermera	69
Figura 32: Prototipo - Página Pacientes- Usuario Enfermera	69
Figura 33: Prototipo - Página Triage - Usuario Enfermera	70
Figura 34: Prototipo - Página Horarios - Usuario Enfermera	70
Figura 35: Prototipo - Página Ficha Clínica - Usuario Doctor	71
Figura 36: Prototipo - Página Roles - Usuario Doctor	72
Figura 37: Prototipo - Página Tipo Pruebas Laboratorio - Usuario Doctor	72
Figura 38: Prototipo - Página Usuarios - Usuario Doctor	73
Figura 39: Prototipo - Página Agendar Cita - Usuario Doctor	73
Figura 40: Prototipo - Página Dashboard - Usuario Doctor	74
Figura 41: Prototipo - Página Historia Clínica - Usuario Doctor	74
Figura 42: Prototipo - Página Horas - Usuario Doctor	75
Figura 43: Prototipo - Página Informe de Turnos - Usuario Doctor	75
Figura 44: Prototipo - Página Lista de Citas - Usuario Doctor	76
Figura 45: Prototipo - Página Paciente - Usuario Doctor	76
Figura 46: Prototipo - Página Triage - Usuario Doctor	77

Figura 47: Prototipo - Página Horarios - Usuario Doctor	77
Figura 48: Diagrama de casos de uso Generalizado	79
Figura 49: Diagrama de Caso de Uso: Panel de Administración	79
Figura 50: Diagrama de caso de Uso: Creación de Cita	80
Figura 51: Diagrama de Caso de Uso: Gestión de Triage	81
Figura 52: Diagrama de Caso de Uso: Gestión de Paciente	81
Figura 53: Diagrama de Caso de Uso: Gestión de Historia Clínica	82
Figura 54: Diagrama de Caso de Uso: Gestión de Turnos	83
Figura 55: Diagrama de Caso de Uso: Gestión de Horas	83
Figura 56: Diagrama de Caso de Uso: Gestión de Roles	84
Figura 57: Diagrama de Caso de Uso: Gestión de Géneros	85
Figura 58: Diagrama de Caso de Uso: Gestión de Usuarios	85
Figura 59: Diagrama de Caso de Uso: Prueba de Laboratorios	86
Figura 60: Diagrama de Actividades: Inicio de Sesión	87
Figura 61: Diagrama de Actividades: Registro del Sistema	87
Figura 62: Diagrama de Actividades: Gestión de Triage	88
Figura 63: Diagrama de Actividades: Gestión Ficha Médica	88
Figura 64: Diagrama de Actividades: Creación de Turnos Masivos	89
Figura 65: Diagrama de Actividades: Creación de Turnos Unitarios	89
Figura 66: Diagrama de Actividad: Agendamiento de Cita	90
<i>Figura 67: Diagrama de Actividad: Historia Clínica</i>	91
Figura 68: Diagrama de Secuencia: Registro Usuario	91
Figura 69: Diagrama de Secuencia: Recuperación de Password	92
Figura 70: Diagrama de Secuencia: Triage	92
Figura 71: Diagrama de Secuencia: Ficha Médica	93
Figura 72: Diagrama de Secuencia: Crear Turno	93
Figura 73: Diagrama de Secuencia: Agendar Cita	94
Figura 74: Interfaz Inicial del Sistema	94
Figura 75: Interfaz de Inicio de Sesión	95
Figura 76: Interfaz de Registro	95
Figura 77: Correo Enviado por Registro	96
Figura 78: Interfaz de Recuperación de Contraseña	97
Figura 79: Interfaz de Menú Admin	97
Figura 80: Interfaz de Dashboard	98
Figura 81: Interfaz de Triage	98
Figura 82: Interfaz de Ficha de Paciente	99
Figura 83: Interfaz Examen de Laboratorio	99
Figura 84: Interfaz de Informe de Paciente	100
Figura 85: Interfaz de Paciente	100
Figura 86: Interfaz de Historia Clínica	101
Figura 87: Interfaz de Gestión de Turnos	101
Figura 88: Interfaz de Gestión Horas	102
Figura 89: Interfaz de Informe de Pacientes	102
Figura 90: Interfaz de verificación de cita	103
Figura 91: Interfaz de Acerca de Nosotros	104
Figura 92: Interfaz de Personal Médico	105
Figura 93: Interfaz de Noticias	105
Figura 94: Evaluación - Gtmatrix	110

Figura 95: Evaluación - Herramienta FAE	111
Figura 96: Evaluación - Herramienta WebSite Grader	112
Figura 97: Evaluación - Herramienta SEO Site Checkup	112
Figura 98: EDT del Sistema por Módulos	121
Figura 99: Cronograma de Actividades	122

INTRODUCCIÓN

La evolución de la tecnología se ha demostrado que avanza a pasos agigantados, esto da lugar a la actualización y optimización de varios procesos administrativos de cualquier área perteneciente a un negocio, las mismas que se daban de forma empírica con procesos manuales; posteriormente la implementación de sistemas de entorno de escritorio dieron lugar a una nueva era, sin embargo existía inconveniente para los usuarios en el ámbito de la accesibilidad conllevando a problemas en su utilización. Con el surgimiento del internet se dio lugar al uso de tecnologías para el acceso a la web la que permite enlazar y acceder a un sistema alojado en un servidor [1], eso quiere decir que un determinado usuario podrá acceder al sistema desde cualquier lugar del mundo solo con tener acceso a la red y un dispositivo tecnológico como Smartphone, Laptop's, Tablets, etc. Por este motivo la utilización de aplicaciones web es una alternativa empresarial viable para la presentación de ofertas, servicios y realización de procesos mejorando la productividad.

La metodología de desarrollo de software posee procedimientos, los cuales proporcionan estructuras que permiten ser implementados en el diseño y desarrollo de un sistema de información como marcos de trabajo que ofrecen una guía para la elaboración del proyecto en sí; donde existen dos tipos tales como la metodología tradicional en la que establece un proceso en secuencia donde va en una sola dirección sin haber opciones a cambios en anteriores etapas, y en cuanto a la metodología ágil es la que brindan como solución principal la flexibilidad al momento de realizar cambios en el desarrollo de un proyecto, además de ser adaptables a proyectos de grande, mediana y pequeña escala.

En la actualidad existen metodologías que combinan ciertos componentes y estructuras tradicionales y ágiles como por ejemplo una de ellas es la metodología SWIRL.

El presente trabajo tiene el enfoque de resolver, automatizar y optimizar los procesos que intervienen en la gestión de una cita médica en el Consultorio Médico Kennedy, mediante el diseño e implementación de un sistema web aplicando la metodología SWIRL al cual se lo denominará SISMEDIK. Este sistema contará con estándares de seguridad, realizará funciones como la gestión de citas médicas,

control de historias clínicas, reporte de exámenes por paciente, reporte de medicinas otorgada a cada paciente, y con un aporte adicional la implementación de un chatbot que ayuda a responder dudas generadas por los pacientes que accedan al sistema. Brinda un ambiente ordenado, sofisticado para los usuarios debido al diseño de interfaces gráficas amigables, será desarrollado utilizando Laravel con el lenguaje PHP para el entorno de backend y el framework Angular para el entorno de frontend, además de utilizar los estándares de HTML, CSS y JavaScript para las interfaces.

El presente documento cuenta con la siguiente estructura:

Capítulo 1:

Diagnóstico de necesidades y requerimientos; en este capítulo hace referencia al ámbito de aplicación donde se detalla el trabajo a realizar, justificación y establecimiento de los respectivos requerimientos tanto funcionales como no funcionales que intervienen en el proyecto.

Capítulo 2: Está enfocado al desarrollo de la propuesta tecnológica planteada, en el presente capítulo se cumple con la definición de los diversos componentes que posee el sistema sustentado por la fundamentación teórica, en este se constituyen los objetivos, cronograma y alcance, además de ello se realiza el análisis de los requerimientos y se los representa mediante la implementación de los modelos UML y prototipos de interfaz, dando lugar a proceder con la ejecución de la implementación o ensamblaje del prototipo.

Capítulo 3: En el capítulo final, se realiza la respectiva evaluación de calidad, empleando en este caso el estándar ISO/IEC 9126, utilizando herramientas disponibles en la web para verificar el SEO como GTmetrix, la cual nos da los resultados de performance y accesibilidad del sistema.

1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS.

1.1. Ámbito de aplicación: descripción de contexto y hechos de interés.

En la actualidad existen un gran número de consultorios o entidades de salud que brindan sus servicios médicos como la gestión de citas utilizando la tecnología; en algunas ocasiones a través de redes sociales, mensajería instantánea, y en otros casos utilizando aplicaciones web o móviles obteniendo como resultado opiniones muy positivas por parte de los usuarios y un nivel de aceptación alto, por la razón que simplifican el proceso de acudir a los consultorios solo por el motivo de reservar una cita para ser atendidos. En la actualidad en el Consultorio Médico Kennedy el proceso de gestión de citas médicas es realizado por una enfermera quien se encarga de recibir a los pacientes y registrar sus datos manualmente de manera escrita causando muchas veces homonimia dando paso a esto a una gran molestia de los usuarios y convirtiéndolo en un proceso lento y poco confiable.

En términos tecnológicos es importante realizar la implementación de un sistema web para automatizar procesos de administración, control y comunicación de información por medio de cualquier dispositivo electrónico como smartphones facilitando su transmisión, almacenamiento y despliegue de datos.[2]

Dentro de los requisitos establecidos a satisfacer se encuentran: La funcionalidad de crear una cita, controlar los horarios disponibles para los usuarios, llevar el control adecuado con la información de historias clínicas, permitir a los usuarios cargar resultados de exámenes para la respectiva evaluación con el médico, además de cargar comprobantes de depósitos bancarios para dicha reservar, se debe destacar que los procesos para la obtención de los requerimientos funcionales como no funcionales presentes en la propuesta tecnológicas se obtienen aplicando los lineamientos y recomendaciones establecidos en la metodología para el desarrollo de software web SWIRL. La recolección de información es identificar los requisitos que se basó en las perspectivas del personal que labora en el Consultorio Médico Kennedy, sin embargo el sistema puede ser adaptado a cualquier otra entidad u organización que brinde servicios médicos y que desee automatizar y optimizar procesos similares.

1.2. Establecimiento de requisitos

El desarrollo del sistema web SISMEDIK, se constituyó obteniendo información por parte del personal del Consultorio Médico Kennedy aplicando ingeniería de requisitos, dando lugar al proceso de gestión, análisis, especificación y validación de requerimientos, mediante la implementación de métodos que permite determinar e identificar la necesidad de procesos y falencias que dan lugar a la problemática y permitiendo determinar la solución más viable y adaptable para la informatización.

La rama de la ingeniería de requisitos se ocupa de los objetivos que se evidencia en el mundo real con un lenguaje natural, las funciones y limitaciones de los sistemas de software, además de ello se involucra el comportamiento del software y su evolución en el tiempo en base a los requisitos dados en el inicio.[3] La ingeniería de requisitos tiene que cumplir con ciertas fases para determinar en sí los requerimientos funcionales como no funcionales y estas son. Obtención de requisitos, gestión de requisitos, documentación de requisitos, validación y verificación de requisitos, convirtiéndose esta rama de la informática en un gran aporte beneficioso para la obtención de resultados finales de calidad y sobre todo satisfaciendo todas o la gran mayoría de las necesidades de los usuarios.[4]

En base a esto se estableció contacto con el Consultorio Médico Kennedy específicamente con el personal que labora en el mismo, con la finalidad de establecer los requisitos a considerar para el diseño, desarrollo e implementación del Sistema de Gestión de Citas Médicas (SISMEDIK), dado esto por la experiencia con los pacientes que atienden.

Entre los requisitos y funcionalidades que brindaron el personal se determinó los siguientes requisitos:

- Gestión de Usuarios (Pacientes – Médicos - Enfermeras).
- Panel de Administración.
- Gestión de Turnos.
- Gestión de Fichas Médicas.
- Control de Historias Clínicas.
- Control de Exámenes de Laboratorios.
- Recepción de pagos para citas médicas.

- Información de Personal Médico.
- Sección de Noticias.
- Chatbot.

Se debe destacar que la metodología SWIRL establece ciertos procesos y características donde hace mención y énfasis al empleo de la recolección de requisitos utilizando ciertos métodos de comunicación como una entrevista, encuesta u otro tipo de método de recolección de información.

1.3. Justificación de requerimientos a satisfacer

La presente propuesta tecnológica tiene como objetivo principal diseñar e implementar un sistema web para la gestión de procesos de agendamiento de citas médicas que optimiza y automatiza las funciones que en la actualidad se los ejecuta de forma manual dejando como resultados pérdidas de información, datos inválidos, datos inconsistentes lo que deriva en malas tomas de decisiones, dicho esto se da lugar a la implementación de componentes y tecnologías que aportan en la calidad del sistema convirtiéndolo en una herramienta útil, de eficiencia y calidad tanto para el médico como la enfermera, debido a que posee un chatbot que permite al usuario interactuar con el sistema respondiendo a sus interrogantes de manera automatizada. Además, gestionar el manejo de la información de una forma más ordenada y segura mitigando errores humanos significativamente debido a que todo el proceso y operaciones las realiza el sistema. La gestión de información dentro de un sistema es importante y de gran responsabilidad debido a que es un proceso que permite evaluar, organizar, presentar y generar información en base a un determinado argumento de una manera exacta, útil y sobre todo que sea disponible en todo momento para los usuarios. El manejo de información es uno de los desafíos más complicados que se da en muchas empresas quienes precisan de una solución óptima con el fin de mitigar riesgos existentes que se presentan en diferentes procesos.

SISMEDIK cuenta con una arquitectura cliente – servidor, esto facilita y da una gran ventaja a que los usuarios tengan disponibilidad del sistema en todo momento, desde cualquier lugar simplemente por medio de un dispositivo inteligente que permita consumir las funciones y servicios que brinda el sistema. Esto gracias a que los servicios implementados y basados en la web facilitan el análisis, visualización y

sobre todo la disponibilidad de la información para convertirlo en accesible desde cualquier lugar del mundo, la implementación de esta en si da un plus en términos de seguridad debido a que brinda un mecanismo central de autenticación esto minimiza las posibilidades de acceso indebido al sistema.

En el proceso de la obtención de requisitos dan a notar tres tipos de perfiles de usuarios: (I) Doctor (Administrador), (II) Enfermera, (III) Pacientes, siendo estos los que tienen diferentes necesidades por lo cual se les otorga niveles de privilegios en cuanto al acceso de funciones en el sistema, los requisitos establecidos para la resolución de la problemática en general y sobre los cuales se diseñará y desarrollará el software serán obtenidos implementando las herramientas y recomendaciones que dispone la metodología SWIRL en base a las características requeridas en la presente propuesta.

2. CAPÍTULO II. DESARROLLO DE PROTOTIPO.

2.1. DEFINICIÓN DEL PROTOTIPO TECNOLÓGICO.

En esta sección se realiza un enfoque en la definición del sistema como tal, sus interesados, arquitecturas y requerimientos funcionales como no funcionales para la implementación de la fase de modelado.

2.1.1. Arquitectura de Software.

El sistema SISMEDIK está basado en una arquitectura de software sintetizado [5], menciona que la arquitectura de software representa la organización del sistema y ofrece una descripción detallada de su funcionamiento, de la misma forma detalla [6], que una arquitectura de software correctamente diseñada es clave para interpretar y realizar un sistema de calidad y eficiente, esto permite además que se pueda adaptar nuevos requerimientos y mantenimientos de una forma eficiente, Una AS (Arquitectura de Software), es el conjunto de estructuras necesarias para razonar sobre el sistema, esta comprende los elementos de software como relaciones entre ellos y propiedades de ambos.

2.1.2. Arquitectura de Angular.

La arquitectura de Angular según [7] está basado en componentes para crear aplicaciones web escalables, proporciona una gran variedad de bibliotecas integradas que cubren la gran mayoría de funciones, incluido el enrutamiento, la gestión e implementación de formularios, la comunicación cliente-servidor y más, brinda un conjunto de herramientas de desarrollo que ayuda a los desarrolladores a en el proceso de desarrollo, compilación, pruebas y actualización de código.

Gracias a esta arquitectura con Angular se puede trabajar desde proyectos de un solo desarrollador hasta aplicaciones de un nivel empresarial. Se debe destacar que Angular está diseñado para que las actualizaciones de proyectos sea lo más sencilla posible y si existen problemas o inconvenientes, gracias a su estabilidad como framework en el mercado de desarrollo y con un número de más de 1.7 millones de desarrolladores en la comunidad facilitan a tener una documentación buena y eficiente.

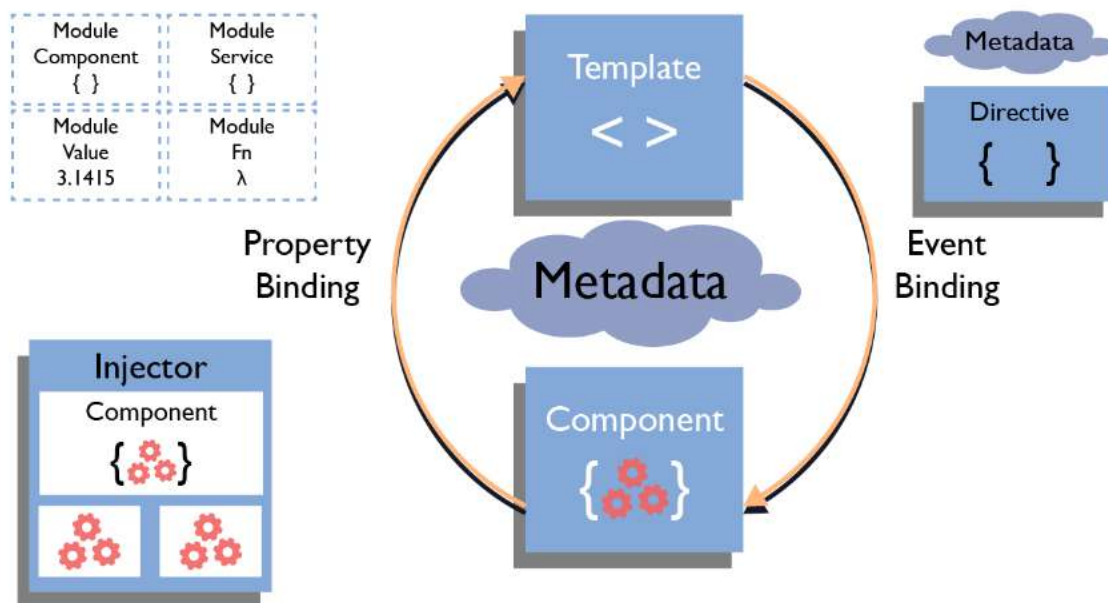


Figura 1: Arquitectura de Angular

Fuente: [8]

2.1.3. Arquitectura de Laravel.

Laravel se establece como un marco de aplicación web que posee una sintaxis expresiva y sobre todo elegante. Este marco web nos brinda una estructura y un punto de inicio para dar origen a una aplicación, lo que da paso en centrarse en el desarrollo de algo increíble mientras nos fijamos en detalles.

Laravel como tal menciona [9] que su fin es brindar al desarrollador una experiencia increíble al mismo tiempo que da facilidades y características potentes, como una inyección de dependencia completa, colas y trabajos programados, una capa de base de datos expresiva, pruebas unitarias y de integración y más.

En el ámbito para desarrolladores que están iniciando se considera que Laravel el framework con PHP pueda crecer a medida de conocimientos adquiridos por el desarrollador. La arquitectura de Laravel está basada en los patrones de diseños más conocidos para el desarrollo web, es el Modelo – Vista – Controlador, cuya misión de este patrón es separar la lógica de un proyecto en 3 capas.

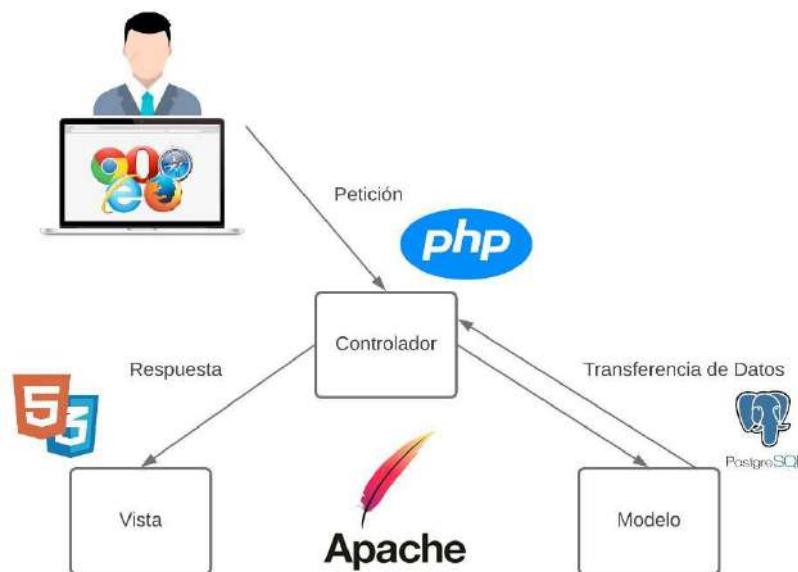


Figura 2: Patrón MVC del Prototipo

Fuente: Elaboración del Autor

En detalle de la figura anterior se evidencia la intervención del servidor web y a su vez el servidor de la base de datos, dando paso a la constitución del patrón MVC (Modelo Vista Controlador), este patrón establece desfragmentar la estructura del

software en 3 capas que son la de Modelo que es el lugar donde se lleva a cabo toda la lógica empresarial de aplicación, la Vista donde se lleva a cabo los escenarios de UI (Interfaz de Usuario) de la aplicación implementada y el último componente que es la de Controlador que se encarga de proporcionar comunicación entre modelos y componentes. [10] [11]

2.1.3.1. Modelo.

La capa Modelo del Patrón de diseño MVC es el apartado que contiene la lógica de negocio, lo ideal de esta capa es que sea independiente del sistema de almacenamiento, además es la encargada de tener mecanismos y funciones para acceder a los datos y actualizar su estado, dado esto aquí estarán todos los mecanismos para acceder a las tablas de la base de datos y ejecutar los respectivos selects, updates o inserts. [12]

2.1.3.2. Vista.

Este elemento del patrón es el encargado de contener el código de la aplicación que va a generar la visualización de las GUI (Interfaces Graficas de Usuarios), es importante mencionar que las vista para mostrar los datos requeridos realiza peticiones a el modelo y estos a su vez genera la salida correspondiente.[13]

2.1.3.3. Controlador.

Según [14] en este componente se da la especificación de métodos, funciones y procedimientos de la aplicación para desarrollar sus respectivas tareas. Se encarga de gestionar las instrucciones que reciben para poder obtener una respuesta y ser finalmente mostradas a través de la vista.

2.1.4. Arquitectura Cliente – Servidor.

La arquitectura cliente - servidor facilita que múltiples usuarios accedan a las funciones o servicios que ofrece un determinado servidor siendo independiente de la ubicación de donde se encuentre, esta arquitectura está fundamentada con la intervención de dos actores que son:

2.1.4.1. Cliente.

De este lado [15] establece que es donde se visualiza las aplicaciones web que juegan un papel importante en el mundo informático contemporáneo, estos ofrecen

una gran cantidad de ventajas sobre los sistemas de escritorios tradicionales, debido a que no son necesarios de algún tipo de instalación, configuración o actualización. La importancia de las aplicaciones web brindadas a los clientes se enfatiza por el continuo aumento de ventas de dispositivos y la capacidad de aplicaciones web.

2.1.4.2. **Servidor.**

En términos informáticos existen tipos de Servidores que desempeñan un papel importante y fundamental en la ejecución de acciones por parte del sistema. En este caso los tipos de servidores implementados en el sistema son los siguientes.

2.1.4.2.1. **Servidor Web.**

La utilización de un servidor web es fundamental en el despliegue de una aplicación de un entorno web, debido a que cumple con las funciones de intercambiar solicitudes entre el navegador y el sistema, las peticiones enviadas por parte del cliente utilizan el protocolo HTTP, y a su vez el servidor responde con encabezados con métodos tipo (GET, POST, PUT, etc.).[16][17].

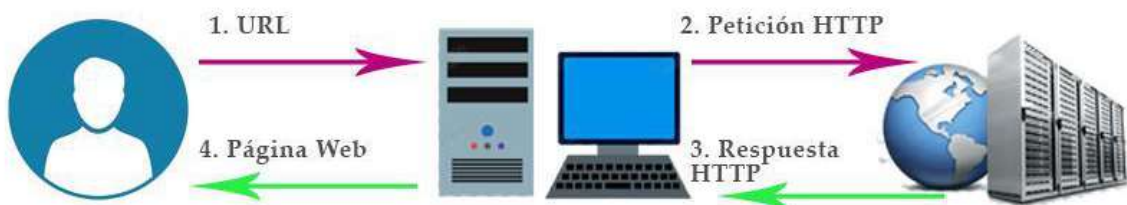


Figura 3: Proceso de Comunicación de Protocolo HTTP

Fuente: Elaboración Propia

El investigador [18] hace mención que la elección de un servidor depende mucho de las necesidades, requerimientos y la experticia del desarrollador, es importante mencionar que algunos ofrecen un entorno de trabajo completo tanto en términos de software como de hardware, sin embargo también hay unos que son privados que tienen un costo por el uso de sus recursos.

2.1.4.2.2. Servidor de Base de Datos

Este servidor tiene como objetivo el almacenamiento de información que recepta el sistema en general, MySQL es la tecnología que es seleccionada para la implementación en el sistema, pero a su vez este es compatible con otros Gestores de Base de Datos como SQL Server, OracleDB y por supuesto PostgreSQL, esto debido a la flexibilidad que permite el framework que se está utilizando como lo es Laravel.

2.1.5. Definición del Sistema.

En base a la arquitectura que tendrá el sistema, es importante reconocer y definir los requerimientos funcionales como no funcionales, con el objetivo de resolver la problemática.

2.1.5.1. Reconocimiento General del Sistema.

Un sistema orientado a la medicina es importante en la sociedad en muchos factores debido a que siempre va a brindar un servicio eficiente, óptimo y sobre todo accesible para la sociedad a la cual está destinada, siendo este el caso SISMEDIK, brinda un entorno accesible a los usuarios del Consultorio Médico Kennedy, aporta muchas ventajas en la administración por el lado del médico, esto se debe a que permite gestionar turnos de citas que se proyectarán a los pacientes, además de eso permite tener un registro ordenado de pacientes, historias clínicas, informes de exámenes, etc. Por el lado del paciente pueden gestionar una cita desde cualquier sitio con acceso a internet utilizando un dispositivo móvil o laptop.

Estas características permiten que este sistema sea una herramienta tecnológica, segura para la gestión de información y para brindar un servicio eficiente a los pacientes.

2.5.1.2. Estudio de Factibilidad.

Con el objetivo de identificar y establecer a los usuarios, definir los beneficios y objetivos que se esperan en la realización del proyecto, definir el alcance y funcionalidades del sistema se detalla el siguiente estudio de factibilidad.

2.5.1.2.1. Beneficios esperados.

El desarrollo e implementación del sistema médico SISMEDIK, se establece como objetivo brindar la solución más óptima a las problemáticas en cuanto al manejo y gestión de información y agendamiento de citas médicas en el Consultorio Médico Kennedy. A continuación, se da a detalle mediante una tabla los beneficios tangibles e intangibles.

Tabla 1: Beneficios Tangibles e Intangibles del Sistema

Beneficios Tangibles	Beneficios Intangibles
<ul style="list-style-type: none">• Automatización de procesos de manejo y gestión de información de pacientes.• Automatización en proceso de gestión de citas médicas.	<ul style="list-style-type: none">• Restablecimiento de cuenta por pérdida de contraseña.• Acceso rápido a información del paciente.

Fuente: Elaboración del Autor

2.1.5.2.2. Usuarios del Sistema.

El sistema SISMEDIK establece inicialmente 3 tipos de usuarios: Administrador (Médico), Enfermera y Paciente, cada uno se detalla a continuación.

Usuario Administrador (Médico): Este rol es el principal dentro del funcionamiento del sistema, posee acceso a todas las funcionalidades del sistema.

Tabla 2: Usuario de Sistema: Administrador (Médico)

Usuario	Administrador (Médico)
Nivel	Primer Nivel
Actividades	<ul style="list-style-type: none">• Acceso al Sistema.• Visualización de Dashboard.• Gestión de Turnos.• Gestión de Pacientes.• Gestión de Roles.• Gestión de Fichas Médicas.• Gestión de Historia Clínica.• Gestión de Informes Médicos• Gestión de Triaje.• Gestión de Exámenes de Laboratorio.• Verificación de Estados de Citas.

	<ul style="list-style-type: none"> ● Configuración de Sistema.
Contenido de Interacción	Acceso a todas las funcionalidades del sistema SISMEDIK.
Habilidades	<ul style="list-style-type: none"> ● Conocimiento de uso de aplicaciones web. ● Conocimiento de Administración de Sistemas Médicos. ● Conocimientos de Procesos de Citas Médicas.

Fuente: Elaboración del Autor.

Usuario Enfermera: Este rol se asigna a la asistente del médico que será la encargada de gestionar y administrar ciertas secciones del panel de administración.

Tabla 3: Usuario de Sistema: Enfermera

Usuario	Enfermera
Nivel	Segundo Nivel
Actividades	<ul style="list-style-type: none"> ● Registro de Sistema. ● Acceso al Sistema. ● Visualización de Dashboard. ● Gestión de Turnos. ● Gestión de Pacientes. ● Gestión de Historia Clínica. ● Gestión de Triaje. ● Verificación de Estados de Citas.
Contenido de Interacción	Acceso a funcionalidades según actividades asignadas.
Habilidades	<ul style="list-style-type: none"> ● Conocimiento de uso de aplicaciones web. ● Conocimientos de Procesos de Citas Médicas.

Fuente: Elaboración del Autor

Usuario Paciente: Este rol se asigna a las personas que van a recibir el servicio y atención médica, poseen acceso a ver información del sistema web como: información de equipo de trabajo, misión, visión del consultorio médico, noticias médicas, y un plus que es un chatbot donde el paciente puede interactuar con el sistema.

Tabla 4: Usuario de Sistema: Paciente

Usuario	Paciente
Nivel	Tercer Nivel
Actividades	<ul style="list-style-type: none"> ● Registro del Sistema. ● Ingreso al Sistema. ● Agendamiento de citas. ● Interacción con ChatBot. ● Carga de Exámenes Médicos. ● Carga de Comprobantes Bancarios para agendamiento de cita.
Contenido de Interacción	Acceso a funcionalidades según actividades asignadas.
Habilidades	<ul style="list-style-type: none"> ● Conocimiento de uso de aplicaciones web.

Fuente: Elaboración Propia

2.1.5.2.3. Alcance del Proyecto.

En base al análisis previamente realizado se estima e identifica las funcionalidades y limitaciones que posee el prototipo a implementar, para detallar de manera simplificada se asume que las gestiones y funciones que realizará cada módulo en su mayoría deben realizar las actividades CRUD (Crear, Editar y Eliminar) registros. A continuación, se detalla las funcionalidades del sistema.

Tabla 5: Funcionalidades del Sistema

Funcionalidades del Sistema
<ul style="list-style-type: none"> ● Gestión de Usuarios. ● Gestión de Roles. ● Gestión de Turnos. ● Gestión de Fichas Médicas. ● Gestión de Informes. ● Gestión Exámenes de Laboratorio. ● Gestión de Triage. ● Gestión de Configuración. ● Chatbot.

Fuente: Elaboración del Autor

Identificadas las funcionalidades del sistema, se identifican también las limitaciones que posee, estas limitaciones se las establece en base a los requerimientos dados por los interesados.

Tabla 6: Limitaciones del Sistema

Limitaciones del Sistema	
•	El sistema no cuenta con verificación RE CAPTCHA.
•	No posee un módulo de facturación.

Fuente: Elaboración del Autor.

2.1.5.2.4. Costo de Inversión.

La Propuesta Tecnológica en un inicio no asume ninguna inversión, ni costos en la implementación ni en el desarrollo, sin embargo, se presenta un valor de forma estimada que se basa en los módulos que presentará el sistema y a su vez en las líneas de código usando la herramienta COCOMO II; mediante la utilización de esta herramienta permite saber un valor estimado del sistema.

A continuación, se muestra el resultado arrojado por la herramienta COCOMO, cada fila representa un módulo y número de líneas que contienen, dado esto se realiza la suma y la herramienta brinda un valor en tres formas que son: optimistas, más probables y pesimista.

Figura 4: Estimación de costos - COCOMO II

Module Name	Module Size	Lines	Effort	Cost	Inst	Staff	Risk			
Accession	8.122	850.00	1.00	Non-Specified	4.3	4.3	291.7	1.0	0.0	
Accession Cases	8.283	870.00	1.00	20%	8.6	8.6	291.7	499.32	2.0	0.0
Accession	8.203	840.00	1.00	20%	1.0	1.0	291.7	143.59	0.5	0.0
Accession Module	8.400	890.00	1.00	20%	1.7	1.7	291.7	342.88	0.7	0.0
Profusers	8.157	80.00	1.00	20%	0.4	0.4	291.7	12.54	0.2	0.0
Sample	8.1	0.00	1.00	Non-Specified	0.0	0.0	0.0	0.00	0.0	0.0

	Estimated	Effort	Staff	PRCD	COST	INST	Staff	RISK
Optimistic	13.0	8.3	8.3	824.4	887.48	1.3	1.4	0.0
Most Likely	16.0	8.3	8.3	291.7	884.38	1.3	1.4	0.0
Pessimistic	20.0	8.3	8.3	229.4	849.44	1.3	1.4	0.0

Fuente: Elaboración del Autor

Mediante el uso de la herramienta COCOMO II, proporcionó como resultado un estimado de 4670 líneas de código, y esto estipula que el sistema puede tener unos costos detallados a continuación.

Tabla 7: Estimación de Costo

Estimación Optimista	\$5507,49
Estimación Probable	\$6884,36
Estimación Pesimista	\$8605,45

Fuente: Elaboración del Autor

El costo del sistema, se estima en \$5000 inicialmente pero este costo es variable y está sujeto a cambios que se pueda realizar en el transcurso de su implementación.

2.1.5.3 Análisis de Factibilidad.

El estudio de factibilidad, establece un punto de origen para establecer si la implementación del proyecto es factible o no, tomando en cuenta la inversión en el inicio del proyecto.

2.1.5.3.1. Factibilidad Operativa.

Proyecta resultados del proyecto en base a un análisis de los recursos que son implementados con el fin de producir en el proceso de implementación del proyecto.

Tabla 8: factibilidad Operativa de Sistema

No.	Actividad	Priorización
1	Capacitación para los usuarios con roles que tendrán acceso al panel de la administración.	Media
2	El sistema requiere de un servidor para operar de manera global a través del uso de Internet.	Alta
3	Capacitación para usuarios con rol de pacientes.	Leve

Fuente: Elaboración del Autor

2.1.5.3.2. Factibilidad Técnica.

De la misma forma que determinamos la factibilidad operativa se debe determinar la factibilidad técnica que hace referencia al uso de herramientas, experiencias, habilidades, etc. Que son necesarias para realizar los procesos, o actividades del proyecto. El sistema SISMEDIK requiere de la implementación de alojamiento en la nube, por lo general el costo de este servicio en el mes es alrededor de \$5, a continuación, en la siguiente tabla se establecen las características que son necesarias en el servidor para la implementación.

Tabla 9: Factibilidad Técnica del Sistema

Recursos de Hardware		Recursos de Software	
Cant.	Descripción	Cant.	Descripción
1	Disco Duro 20GB	1	Servidor web Apache 2
1	Procesador Core i5 7ma Generación en adelante.	1	PHP 7.4+
1	8GB RAM	1	MySQL
1	Tarjeta de Red	1	IP pública e Internet
1	Router	1	Dominio.

Fuente: Elaboración del Autor

2.1.5.3.3. Factibilidad Económica.

El proyecto no va a reflejar gastos en el proceso de desarrollo, debido a que este sistema pertenece a un proceso de titulación.

Tabla 10: Factibilidad Económica del Sistema

RECURSOS TECNOLÓGICOS		
HARDWARE		
CANT.	DESCRIPCIÓN	COSTO TOTAL
1	• Computador Core i5 7ma generación,	\$1019

	<ul style="list-style-type: none"> • Capacidad de almacenamiento interno con Unidad de estado sólido PCIe NVMe M.2 de 256 GB (Unidad óptica no incluida). • Memoria RAM de 8 GB DDR4-3200 MHz (2 x 4 GB) 	
SOFTWARE		
CANT.	DESCRIPCIÓN	COSTO TOTAL
1	Editor de Código VSCode	\$0
1	Servidor web local Laragon	\$0
1	Gestor de BD MySQL	\$0

Fuente: Elaboración del Autor

2.1.5.4. Identificación de Interesados.

La actual propuesta tecnológica hace referencia como interesados a los miembros que son parte del consultorio médico SISMEDIK, que han brindado los requerimientos descritos. A continuación, se conoce a detalle mediante una tabla los stakeholders de más relevancia que son participes de todas las fases que contempla la metodología SWIRL.

Tabla 11: Matriz de Stakeholders

MATRIZ GENERAL DE STAKEHOLDERS			
Nombre	Rol	Profesión	Responsabilidades
Ing. Joofre Honores Tapia	Gerente del Proyecto	Ing. de Sistemas	Revisión de entregables, gestión de requerimientos, pruebas
Dr. Jorge Huamán	Director de Consultorio Médico Kennedy	Médico	Emisión y delimitación de requerimientos.
John Huamán Diaz	Desarrollador	Alumno	Planificación, Implementación, pruebas y lanzamiento.

Fuente: Elaboración del Autor

2.1.5.5. Requerimientos

Los requerimientos detallados en este sistema se los recolectó mediante la aplicación de la ingeniería de requisitos, utilizando el método de entrevistas y diálogos con los interesados.

2.1.5.5.1. Requerimientos Funcionales.

A continuación, en las siguientes tablas se detallan los requerimientos funcionales y no funcionales que posee el sistema, en base a lo descrito por el usuario el sistema debe llevar a cabo todo para dar como resultado final un sistema eficiente y cumplir con el objetivo planteado en el inicio.

Tabla 12: RF - Requerimientos Funcionales del Sistema

REQUERIMIENTOS FUNCIONALES (RF)	
Código	Descripción
RF-01	Sistema de autenticación.
RF-02	Creación de cuenta, recuperación de cuenta mediante correo electrónico.
RF-03	Gestión de Turnos.
RF-04	Gestión de Fichas Médicas.
RF-05	Control de Historias Clínicas.
RF-06	Control de Exámenes de Laboratorio.
RF-07	Recepción de Pago de Citas Médicas.
RF-08	Sección de Noticias.
RF-09	ChatBot.
RF-10	Dashboard de Pacientes y Citas.
RF-11	Reportes de Pacientes, Exámenes y Citas.

Fuente: Elaborado por el autor.

2.1.5.5.2. Requerimientos no Funcionales.

Tabla 13: RNF - Requerimientos No Funcionales del Sistema

REQUERIMIENTOS NO FUNCIONALES (RNF)	
Código	Descripción
RNF-01	En base a la seguridad, los datos sólo pueden ser modificados por los usuarios con acceso al sistema.
RNF -02	El sistema SISMEDIK debe tener una estructura, detallada, fácil de entender.
RNF -03	Los mensajes push deben ser los adecuados según el proceso realizado o los errores para facilitar al usuario la mayor comprensión posible.
RNF -04	La probabilidad de fallo debe ser muy baja.
RNF -05	El sistema debe estar disponible para el acceso al usuario.

RNF -06	Las respuestas del sistema en operaciones realizadas por los usuarios debe ser en menos de 5s promedio.
RNF -07	El sistema debe ser adaptable a todo tipo de dispositivos.
RNF -08	Los datos sensibles como contraseñas deben estar encriptados usando algoritmos como SHA-256.

Fuente: Elaboración del Autor

2.2. Fundamentación Teórica del Prototipo.

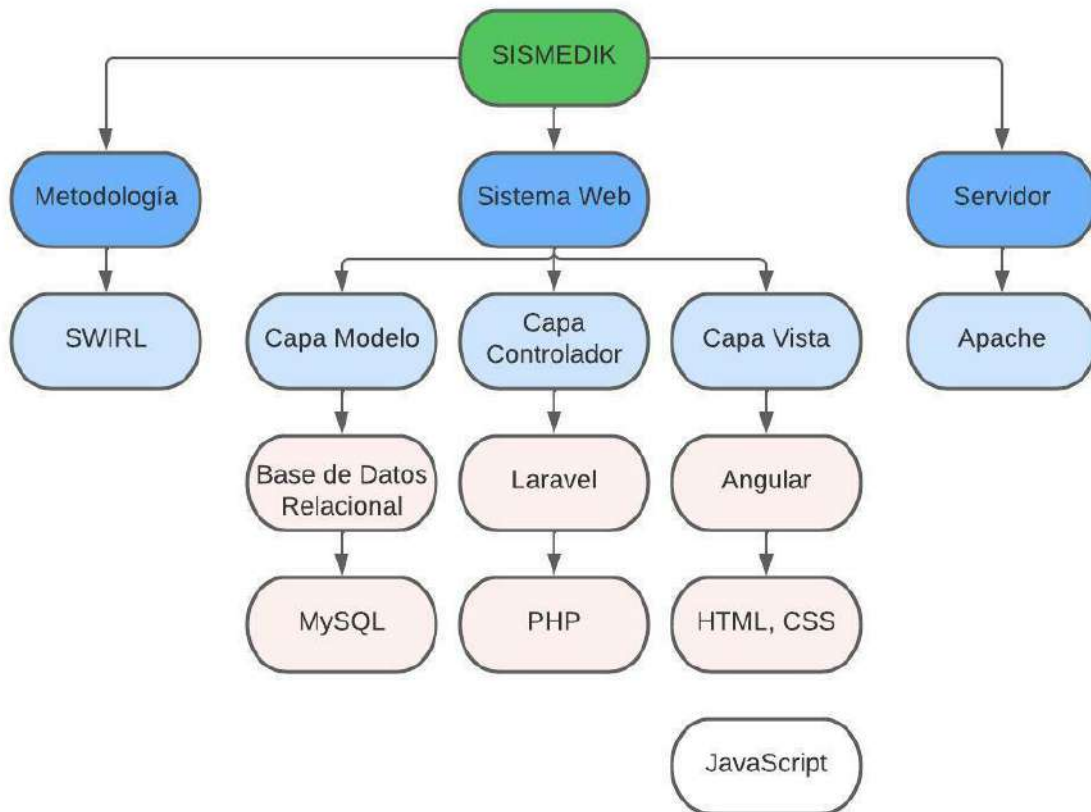


Figura 5: Fundamentación Teórica del Proyecto

Fuente: Elaboración del Autor

2.2.1. Metodología.

Según [19], el establecimiento de una metodología brinda la facilidad de definir un conjunto de actividades, estructuras y procesos que permiten obtener un producto de software final de alta calidad, dado esto facilita a los líderes de proyecto (Project Managment) y desarrolladores en la implementación de sistemas de información, la creación de las metodologías y modelos de desarrollo de sistemas surgieron a la necesidad de resolver la tan llamada “Crisis de Software”, que hacía referencia a la complejidad de desarrollo de un software y también a los cambios que debía

someterse el software para adaptarse a los requerimientos del usuarios y las constantes innovaciones tecnológicas.

A raíz de esto la experiencia y mucha documentación ha dado a notar que los proyectos administrados que siguen una serie de procesos y que luego permiten organizar y contralar el proyecto son los que consiguen a satisfacer por completo o en mayor grado las necesidades del usuario, y que los que no realizan ciertos procesos de la metodología son propensos a correr riesgo de fracasar.

2.2.1.1. Metodologías Tradicionales.

En inicio de los 60, [20], estableció que, el desarrollo de software a través del tiempo se ha venido implementando una serie de procesos que tienen como nombre metodologías de desarrollo los mismos que facilitan el proceso. De la misma forma [21] argumenta que este tipo de metodologías se los llaman también modelos de proceso prescriptivo, y fueron establecidas en un inicio para establecer lineamientos y orden en el caos que se solía darse durante el desarrollo de un software, los inicios de las metodologías tradicionales se dieron en la década de los 60, donde se denomina a un proyecto como uno solo de grandes dimensiones y estructura definida, con un proceso rígido que no va a variar, los requerimientos planteados no están sujetos a variar en ningún momento y una comunicación muy limitada entre el usuario y el jefe de proyecto. El primer modelo publicado para proceso de desarrollo de software se dio origen por procesos grandes de ingeniería de software, debido al paso de una fase en cascada a otra, es por ello que se le dio ese nombre Modelo Cascada.

2.2.1.2. Metodologías Ágiles.

La calidad de un software es medida en términos de métricas, como el número de defectos que se dan en el desarrollo del software. Una metodología de desarrollo ágil se aplica con el objetivo de determinar valores óptimos de parámetros analizados que se da en el desarrollo para determinar un software de calidad, constituyendo un enfoque que captura los defectos en el proceso de desarrollo del software.[22]

[23] Afirma que en la actualidad más del 70% de organizaciones a nivel mundial orientadas al desarrollo de software usan metodologías ágiles en la implementación

de sus proyectos, esto afirma que el desarrollo ágil y entrega rápida son por lo general el requerimiento fundamental de los sistemas en la actualidad, el mundo y los negocios funcionan en un entorno cambiante por lo cual las metodologías ágiles establecen lineamientos y procesos que permiten el desarrollo de una forma rápida y la documentación necesaria para el sistema a desarrollar sea entregado un tiempo menor a lo común.

2.2.1.3. Metodologías Híbridas.

Según [24] y [25], los grandes proyectos son difíciles de manejar y es por ello que no cumplen con la expectativa de satisfacer a las partes interesadas, por ello la aplicación de metodologías desde su creación ha generado un impacto en la industria tecnológica, debido a que permite la adaptación a cualquier entorno de un software, más sin embargo es preciso reconocer que no existe una única metodología aplicable universalmente a proyectos por ello las metodologías ágiles y no ágiles deben adaptarse y por ello se crea lo referente a las metodologías híbridas debido a que es un entorno que permite la combinación de elementos de metodologías tradicionales como ágiles, potencializando la estructura de elaboración del software tanto en el desarrollo como en la documentación.

2.2.2. SWIRL: Metodología para desarrollo de aplicaciones web.

La metodología SWIRL (Software Web Iterativo Relacional Lógico), se enfoca en el desarrollo e implementación de aplicaciones orientadas a la web, esta metodología combina un enfoque híbrido e iterativo.

El usuario tiene una participación total en el proceso de desarrollo debido a que se lo hace partícipe en muchos procesos del desarrollo del software, la metodología SWIRL se basa en la ejecución total de las fases que tiene, todas estas iteraciones dan origen a un dibujo que se representa a la metodología que posteriormente se presentará.[26]

2.2.2.1. Características.

Es aplicable en proyectos de pequeña o gran escala.

Se basa en validaciones que participa el usuario.

Disminuye recursos humanos, costo y tiempo de implementación.

Incluye técnicas SEO, modelo de negocios y Joint Application Desing (JAD).

Da la facilidad de que los requisitos establecidos en el inicio se pueden cambiar en cualquier etapa del proyecto.

2.2.2.2. *Ventajas y Desventajas.*

Ventajas

La metodología SWIRL se caracteriza por ser un modelo flexible y adaptable a las necesidades del usuario.

Presentación continua de resultados.

Disminuye tiempo y costo de recursos dentro del proceso de desarrollo.

Desventajas

Usuarios deben estar definidos desde el inicio.

Poco adaptable para equipos de desarrollo distribuidos.

Se debe finalizar una iteración, y ejecutar una revisión para siguiente.

2.2.2.3. *Modelo SWIRL.*

Esta metodología posee un modelo iterativo la cual considera en cada iteración un grupo de cinco criterios que son partícipes del desarrollo del proyecto: calidad, costo, tiempo, alcance y comunicaciones, todos estos criterios están definidos por todos interesados que son por lo general el product manager y el cliente.

Los criterios de tiempo y costo son definidos por el en la etapa de planificación por el Product Manager, y por otra parte los criterios de calidad y alcance los define el cliente, y el criterio restante de la comunicación lo pueden definir las dos partes por mutuo acuerdo o cualquiera de las partes.

En el siguiente gráfico se evidencia el proceso iterativo que considera los cinco criterios a los que hace referencia la metodología SWIRL.



Figura 6: Modelo SWIRL

Fuente: [26]

2.2.2.4. **Ciclo de Vida.**

Según [26] la metodología SWIRL en su ciclo de vida consta de seis fases que se detalla a continuación:

1. Análisis.
2. Planificación.
3. Modelado.
4. Implementación
5. Revisión y Pruebas.
6. Lanzamiento y marketing.

Durante las iteraciones que da en el proceso de elaboración del proyecto deben ser participe el Product Manager con su equipo de trabajo y los clientes, asegurando de esta forma una continua revisión de todas las funcionalidades que posea el sistema durante todo el desarrollo del proyecto, cabe mencionar que por lo general cada iteración tiene una duración de 15 días a un mes esto depende de la dificultad y duración del proceso.

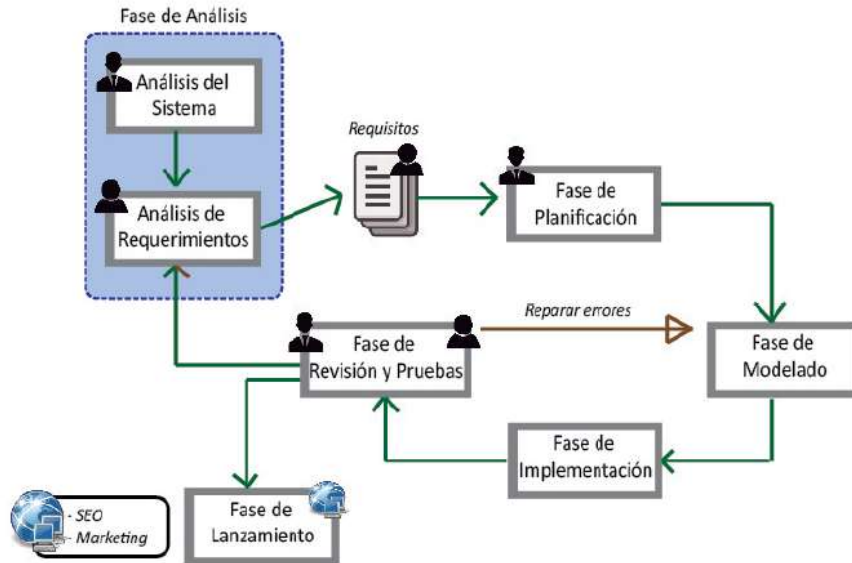


Figura 7: Ciclo de Vida de Metodología SWIRL

Fuente: [26]

2.2.2.5. Pasos para el correcto uso de la metodología.

La metodología Swirl establece una serie de pasos como una guía para su correcta aplicación y uso que son descritos a continuación:

Tabla 14: Pasos para correcto uso de Metodología

El gerente define el sistema y modelo de negocios.
El cliente delimita los requerimientos y calidad del software.
El gerente planifica las actividades, comunicaciones y entregables de cada iteración.
Se diseñan los modelados e implementan las funcionalidades.
Se verifica y se realizan las pruebas de control, en caso de encontrar algún tipo de error en la implementación se procede a verificar los modelados y se vuelve a codificar.
Una vez validado, se retorna al punto 2 para la respectiva revisión de requisitos y seguir con las iteraciones hasta culminar todas las funcionalidades.

Como último punto, cuando la aplicación esté verificada por parte del cliente y sin tener ningún error, se procede a aplicar la última etapa de la SWIRL que es el lanzamiento y marketing.

Fuente: Elaboración del Autor

2.2.2.6 **Fase de Análisis.**

En la fase de análisis se detallan variables relevantes que formarán parte del desarrollo de aplicación web, el modelo de negocio, definición del modelo, análisis sustancial, identificación de interesados y el listado de requerimientos funcionales como no funcionales son partícipes de esta fase. A continuación, se detallan otros elementos que se desarrollan en esta fase.

Tabla 15: Actividades de Fase de Análisis

Fase I: Fase de Análisis	
1. Definir el Sistema	Reconocimiento General. Estudio de Factibilidad. Análisis de Factibilidad de Sistema. Perspectiva del Producto. Funcionalidad de Sistema. Análisis de Riesgo.
2. Modelado de Negocio	Descripción del Modelo de Negocio. Evaluación. Mejora / Innovación (Opcional)
3. Identificación de Interesados	Identificación de roles y stakeholders. Definición de las actividades.
4. Requerimientos	Elicitación. Análisis. Validación.

Fuente: [26]

2.2.2.7. **Fase de Planificación.**

En lo correspondiente a la etapa de planificación, el gerente debe tomar en cuenta algunos factores con el objetivo de realizar los planes de tiempo, comunicación,

riesgos, calidad y entre otros; que son necesarios para que el proceso de desarrollo y las iteraciones se puedan realizar de manera sencilla.

Las actividades que se deben realizar en esta fase son:

- Historias de usuario.
- Definición de entregables.
- Gestión de cronograma, riesgos y comunicaciones.
- Estructura de desglose de trabajo.
- Velocidad de proyecto y estimación de esfuerzo.
- Gestión de cambios y de calidad.

2.2.2.8. Fase de Modelado.

En lo correspondiente a fase de modelado se realizan diseños de base de datos, modelos conceptuales y modelos navegacionales del proyecto. Además, se debe aplicar los diagramas UML (Lenguaje de Modelado Unificado) para el modelado de datos. Las actividades a desarrollar son:

- Diseño de modelo Conceptual.
- Diseño de modelo navegacional.
- Diseño de interfaz abstracta de usuario.
- Modelado de diagramas UML.

2.2.2.9. Fase de Implementación.

En la fase de implementación se realiza el proceso de codificación de las funcionalidades que va a tener el sistema, según las iteraciones especificadas en la planificación definida en un inicio. Dentro de este proceso se debe aplicar una nomenclatura general, independientemente al lenguaje de programación escogido. Las actividades a desarrollar en esta fase son:

- Codificación
- Estándares
- Definición de módulos e implementación.
- Diseño y codificación de páginas principales.
- Interconexiones.

2.2.2.10. Fase de Revisión y Pruebas.

Dentro de esta fase se procede a evaluar y testear todas las funcionalidades implementadas en el sistema para la correspondiente verificación de la correcta funcionalidad. Además, en cada iteración realizada se debe realizar un cierre en conjunto con la presentación de entregables.

- Control Integrado de Cambios.
- Pruebas de integración.
- Detección y corrección de errores.
- Pruebas de sistema y aceptación.
- Control de alcance.
- Cierre.

2.2.2.11. Fase de Lanzamiento y Marketing.

En lo correspondiente a esta etapa no se la considera en cada iteración, se la realiza en el final del proyecto, para esto la aplicación debe ser aceptada por el cliente y no debe tener errores al momento de la ejecución. En este punto se trabaja en el SEO del portal con el objetivo de posicionarlo correctamente en los buscadores.

Las actividades a desarrollar son:

- Selección y alojamiento del hosting.
- Preparación del dominio.
- Configuración del certificado SSL (Opcional)
- Campaña de Marketing / SEO.

Esta fase de Lanzamiento y Marketing es un plus diferenciador que posee esta metodología de SWIRL ante otras metodologías, debido a que es necesario también para una mejor calidad del sistema posicionarlo y promocionarlo para que llegue al público con un mayor índice de aceptación y mayor alcance.

2.2.1. Sistema Web

Según [27] , Los sitios web en las últimas décadas se han convertido en un elemento tecnológico usable y atractivo en muchos campos como el turismo, agencias de viajes, gracias a estos han aumentado en importancia a medida que

más personas confían en ellos para la planificación de viajes, además de ellos el acceso a servicios esenciales gubernamentales los cuales se han convertido entre los sitios más buscados en la web, es por ellos que los gobiernos se ven en la obligación de trasladar todos sus servicios a un entorno web mitigando las largas colas físicas de personas en entidades y el colapso de llamadas telefónicas, esto en conjunto es un aporte a personas con discapacidad o mayores a que puedan realizar sus trámites a través de un sitio web sin tener que movilizarse a las entidades.

Argumentando a esto [28], menciona que un sitio web debe contar con el máximo nivel de seguridad en vista que muchos manejan información sensible de los usuarios que acceden y así protegerlos de ataques informáticos que ponga en riesgo la información almacenada.

2.2.1.1. Servidor Web.

En un artículo elaborado por [29], establece que un servidor web es una aplicación que da respuesta a solicitudes que provienen de navegadores web, estos proporcionan los recursos que son solicitados por medio del protocolo HTTP (Hypertext Transfer Protocol) o de manera segura a través del protocolo HTTPS (Hypertext Transfer Protocol Secure), en la actualidad existe una gran variedad de servidores web, como Apache 2, Nginx, LiteSpeed, Google entre otros. Según Netcraft, los servidores de Apache con 33.89% y Nginx con 29.61% son los dos servidores web de código abierto más usados. [30]

2.2.1.1.1. Servidor Web Apache.

La utilización de este tipo de servidor de código abierto, es un método para garantizar flexibilidad, rapidez y eficiencia, añadiendo a esto es capaz de realizar una actualización y capacidad de asumir nuevos protocolos, en vista a su gran y robusta arquitectura permite un alto índice de calidad y fortaleza para implementaciones que utilizan el protocolo HTTP.[31] [32]

2.2.1.1. MySQL.

El empleo del Gestor MySQL aporta en el almacenamiento de datos de la aplicación, es considerado un sistema potente de bases de datos relacionales de

objetos. Posee una licencia BSD por lo que lo convierte en un software gratuito y de código abierto.

MySQL posee un lenguaje que es el SQL estándar, este da facilidad al momento de realizar inserciones, actualizaciones y consulta de datos que se encuentran almacenados en las tablas que contiene una base de datos.[33][34]

2.2.1.2. Laragon.

Laragon es creado por Leo Khoa, es un director de DevOps que tiene una empresa SaaS de rápido crecimiento. Construyó Laragon con un único objetivo de hacer que el desarrollo web sea rápido, divertido y agradable.[35]

Laragon es un entorno de desarrollo local moderno, robusto y rico en características. Muchos desarrolladores web de todo el mundo lo utilizan para crear aplicaciones debido a que es más rápido y sencillo de configurar y usar.

2.2.1.3. PHP.

Según [36], PHP se dio origen como un lenguaje que se usaba para realizar seguimiento de visitas a páginas personales, pero en la actualidad es considerado uno de los lenguajes de scripts más populares a nivel mundial.

Es uno de los más utilizados debido a que es un software libre, en modo de costo e implementación es la mejor opción debido a que tiene mucha documentación y una arquitectura con procesos fáciles de entender.

2.2.1.4. Laravel.

Menciona [37] Laravel está basado en algunos componentes de Symfony, se lo considera en un marco de desarrollo de Back – End como un marco web PHP de código abierto y de acceso gratuito, Fue creado por Taylor Otwell, y está hecho para el desarrollo de aplicaciones web, si arquitectura se basa en el patrón modelo – vista – controlador (MVC), debido a que emplea este patrón lo convierte en un framework fácil de entender y poderoso, de la misma forma [38] hace énfasis en que Laravel es un marco que realiza sus funciones del lado del servidor y que alivia costos en el desarrollo de proyectos de gran escala, posee una sintaxis y herramientas fáciles de entender que pueden ser implementadas en el proceso de desarrollo.

Características:

Implementa componentes de Symfony.

Motor de Vistas Blade.

Emplea Composer para el manejo de librerías.

Tiene un ORM denominado Eloquent para mapeo de relaciones.

2.2.1.5. HTML – CSS.

HTML: [39] menciona que HTML es un lenguaje para desarrollo web, es considerado un método de descripción de documentos que no depende de plataforma, se proyectan a través de un navegador web, además se lo considera como algoritmos que procesan lenguaje natural, son implementados para procesar texto libre. Añadiendo a lo mencionado [40] establece que la fusión de HTML con otras tecnologías web de última generación hacen de él un componente fundamental en la tecnología web.

CSS: Conocido como Hojas de Estilo en Cascada (Cascading Styles Sheets), es implementado en proyectos de pequeña y gran escala, se lo utiliza para establecer la presentación, el lenguaje CSS, surge ante la necesidad de dar formatos y diseños a componentes y establecer un diseño usable como colores, animaciones, efectos, bordes y todo tipo de herramientas para dar un aspecto visual a los documentos web.[41]

2.2.1.1. Otros Conceptos.

2.2.1.1.1. Herramientas CASE.

La Ingeniería de Software plantean áreas de conocimiento específico que requieren el uso de herramienta, para el modelado de software son utilizadas herramientas que nos ayuden a la elaboración de diagramas estas son llamadas Herramientas CASE, son consideradas toda herramienta que brindan soporte a cualquier actividad que se lleve a cabo dentro de la IS, incluso en las actividades de modelado.[42]

Según asegura que las herramientas CASE ayudan en garantizar que la calidad se diseñe antes de iniciar la implementación del proyecto, en un inicio las herramientas CASE se las utilizaba en la automatización de documentación de proyecto pero en la actualidad intervienen en muchos procesos de la

implementación de proyectos con la finalidad de mejorar la proactividad del producto final.[43]

2.2.1.1.2. Responsive Desing.

El diseño responsivo es empleado en una estandarización y adaptación en los sistemas web que son desarrollados para dispositivos móviles y equipos de cómputo, esto significa en optimizar el diseño de una página web para dimensiones de la pantalla del usuario y orientación de la misma, el responsive design es un enfoque con el objetivo de proporcionar a los usuarios una visualización óptima, esto conlleva a una lectura, visualización y navegación más sencilla con un mínimo de cambio de tamaño, panorámica y desplazamiento dependiendo del dispositivo inteligente en uso.[44] [45]

2.3. Objetivos del Prototipo.

2.3.1. Objetivo General.

Desarrollar una aplicación web para la generación y recepción de citas médicas e historia clínicas de pacientes del consultorio médico Kennedy aplicando la metodología SWIRL.

2.4. Objetivos Específicos.

Obtener información de procesos que intervienen en la gestión de citas médicas e historias clínicas, para la determinación de requerimientos funcionales y no funcionales.

Desarrollar e implementar el prototipo.

Evaluar el prototipo aplicando herramientas y estándares de calidad.

2.5. Diseño del Prototipo.

Para el diseño de este prototipo se establece a SWIRL como metodología, es por ello que esta sección hace énfasis a la planificación y modelado, cronología, análisis de procesos y la definición de la estructura del sistema, flujo de datos e interfaces.

2.2.1. Fase de Planificación.
2.2.1.1. Historias de Usuario.

Tabla 16: Historia de Usuario #1- RF-01

Historia de Usuario	
Número: 1	Usuario: Cualquiera
Nombre Historia: Inicio de Sesión en el Sistema	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-01	Iteración asignada: 1
Programador Responsable: John Diaz	
Descripción: En esta sección se puede ingresar como cualquier tipo de usuario ya sea administrador, enfermera, paciente, solo se debe ingresar el correo que utilizó en el registro de cuenta y la respectiva contraseña.	
Observaciones: Si un usuario olvidó la contraseña, el sistema proporciona la facilidad de recuperar la contraseña.	

Fuente: Elaboración del Autor

La interfaz a mostrar al usuario, depende del rol que posee esto debido a que el usuario de tipo Doctor tiene acceso a todas las funciones del sistema.

Tabla 17: Historia de Usuario #2- RF-02

Historia de Usuario	
Número: 2	Usuario: Cualquiera
Nombre Historia: Abrir Cuenta y Recuperar Contraseña vía email.	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-02	Iteración asignada: 2
Programador Responsable: John Diaz	
Descripción: Cualquier tipo de persona tiene acceso a registrarse al sistema, solo debe ingresar los datos solicitados, posterior a eso le llegará un mensaje al correo dándole la bienvenida al sistema, además de esto el sistema brinda la opción de recuperación de contraseña en caso de no recordar.	
Observaciones: El usuario tiene la facilidad de recuperar la cuenta por medio de la opción de recuperar contraseña.	

Fuente: Elaboración del Autor

Para el registro se presenta un formulario donde debe llenar los datos solicitados.

Tabla 18: Historia de Usuario #3- RF-03

Historia de Usuario	
Número: 3	Usuario: Doctor, Enfermera, Paciente
Nombre Historia: Gestión de Turnos	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-03	Iteración asignada: 1
Programador Responsable: John Diaz	
Descripción: Los usuarios de tipo médico y enfermera tienen el acceso a la gestión de turnos, es decir ellos pueden abrir, modificar o eliminar un horario de atención, pero a su vez aquí se considera a los usuarios de tipo paciente ya que son ellos los que intervienen en el proceso de agendamiento. Ellos tienen el derecho a realizar un agendamiento en los horarios pre dispuestos por el médico.	
Observaciones: Para que el usuario de tipo paciente pueda agendar una cita debe estar previamente logueado.	

Fuente: Elaboración del Autor

El usuario de tipo paciente puede agendar una cita siempre y cuando este logueado, caso contrario no podrá acceder al servicio.

Tabla 19: Historia de Usuario #4- RF-04

Historia de Usuario	
Número: 4	Usuario: Doctor
Nombre Historia: Gestión de Fichas Médicas.	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-04	Iteración asignada: 2
Programador Responsable: John Diaz	
Descripción: Solo el tipo de usuario médico tiene la opción de acceder a realizar fichas médicas a los pacientes, en esta sección tiene que haber una cita gestionada por un paciente para proceder a elaborar la ficha médica.	
Observaciones: El médico tiene el acceso a datos de triaje del paciente y datos personales.	

Fuente: Elaboración del Autor

Mediante este proceso se puede ver los últimos 3 datos triajes del paciente como medio de control.

Tabla 20: Historia de Usuario #5- RF-05

Historia de Usuario	
Número: 5	Usuario: Doctor
Nombre Historia: Control de Historias Clínicas.	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-05	Iteración asignada: 2
Programador Responsable: John Diaz	
Descripción: El usuario tipo doctor tiene el acceso a la vista de historias clínicas, esto es un tipo informe de las veces que ha sido atendido el paciente.	
Observaciones: Se tiene acceso a la vista de diagnóstico del paciente en las diferentes fechas que ha sido atendido.	

Fuente: Elaboración del Autor

El doctor tiene el acceso a ver una recopilación de las veces que ha sido atendido un paciente.

Tabla 21: Historia de Usuario #6- RF-06

Historia de Usuario	
Número: 6	Usuario: Doctor, Paciente
Nombre Historia: Exámenes de Laboratorio.	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-06	Iteración asignada: 2
Programador Responsable: John Diaz	
Descripción: El usuario de tipo doctor, realiza la gestión de enviar exámenes médicos según sea el caso del paciente. A su vez el paciente tiene la facilidad de cargar los resultados de exámenes médicos para que sean revisados por el médico y éste realizar un diagnóstico.	
Observaciones: El doctor tiene la posibilidad de enviar los exámenes necesarios al paciente.	

Fuente: Elaboración del Autor

Tabla 22: Historia de Usuario #7- RF-07

Historia de Usuario	
Número: 7	Usuario: Paciente, Doctor
Nombre Historia: Confirmación de cita médica.	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-07	Iteración asignada: 2
Programador Responsable: John Diaz	
Descripción: El paciente tiene la posibilidad para confirmar su cita médica mediante el envío de comprobante de pago a través del sistema, simplemente debe subir un archivo tipo pdf de preferencia, el doctor debe revisar y hacer efectiva la reservación.	
Observaciones: Cabe mencionar que el doctor puede revisar el comprobante de pago, si es un caso conocido el paciente puede asistir en el horario escogido y pagar en el consultorio.	

Fuente: Elaboración del Autor

Tabla 23: Historia de Usuario #8- RF-08

Historia de Usuario	
Número: 8	Usuario: Cualquiera
Nombre Historia: Sección de Noticias	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-08	Iteración asignada: 1
Programador Responsable: John Diaz	
Descripción: Cualquier tipo de persona tiene acceso a esta sección de la aplicación web, debido a que es un medio de conocer noticias de portales médicos y de la misma forma noticias de cuidado de salud.	
Observaciones: El sistema abre en nuevas pestañas los enlaces de interés de noticias médicas.	

Fuente: Elaboración del Autor

Tabla 24: Historia de Usuario #9 - RF-09

Historia de Usuario	
Número: 9	Usuario: Cualquiera

Nombre Historia: ChatBot	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-09	Iteración asignada: 2
Programador Responsable: John Diaz	
Descripción: Cualquier tipo de persona tiene acceso a entablar un diálogo con el asistente virtual Kennedy, este ayudará a guiar al usuario a realizar los diferentes procesos que ofrece la aplicación.	
Observaciones: El chatbot está enfocado en el uso para los pacientes, responder sus interrogantes acerca de la aplicación y servicios que ofrece.	

Fuente: Elaboración del Autor

Tabla 25: Historia de Usuario #10 - RF-10

Historia de Usuario	
Número: 10	Usuario: Cualquiera
Nombre Historia: Dashboard de Pacientes y Citas.	
Prioridad en negocio: Baja	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-10	Iteración asignada: 2
Programador Responsable: John Diaz	
Descripción: Los usuarios de tipo enfermera y doctor tienen acceso a ver un pequeño Dashboard acerca del número de usuarios registrados en el sistema, y el número de citas creadas y atendidas.	
Observaciones: El Dashboard es un medio para llevar a cabo el control de los servicios que tiene la aplicación.	

Fuente: Elaboración del Autor

Tabla 26: Historia de Usuario #11 - RF-11

Historia de Usuario	
Número: 11	Usuario: Cualquiera
Nombre Historia: Reportes de pacientes.	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Baja
Requerimiento Funcional: RF-11	Iteración asignada: 2
Programador Responsable: John Diaz	

Descripción: Los usuarios de tipo doctor y enfermera tienen el acceso a generar reportes de pacientes, exámenes y citas que se han realizado a través del sistema.

Observaciones: Los informes son de importancia en el proceso de control de la aplicación.

Fuente: Elaboración del Autor

2.2.1.1. Gestión de Cronograma.

Esta sección se basa en los requerimientos funcionales como no funcionales, basado en esto se considera para esta propuesta tecnológica realizar dos iteraciones con una duración de 30 días cada una. Para mostrar con más detalle esta estructura se procedió a realizar un cronograma para que aprecie los procesos considerados y las iteraciones del proyecto. Ver el ANEXO 3 CRONOGRAMA, a continuación, se procede a mostrar las actividades de historias de usuario.

Tabla 27: Cronograma - Iteración #1

No. Iteración	1	Duración en días		25
Observaciones	Se aplica un margen de 5 días para realizar pruebas y entregables.			
#	Historia de Usuario	Encargados	Duración	Prioridad
1	HU-1	John Diaz	3	Alta
2	HU-2	John Diaz	2	Alta
3	HU-4	John Diaz	4	Alta
4	HU-7	John Diaz	3	Alta
5	HU-8	John Diaz	5	Alta
6	HU-9	John Diaz	4	Alta
7	HU-10	John Diaz	4	Alta

Fuente: Elaborado del Autor

La iteración dos se realiza con la finalidad de revisar todas las funcionalidades, además se debe implementar nuevas acciones y realizar cambios dados por los usuarios interesados.

Tabla 28: Cronograma - Iteración #2

No. Iteración	2	Duración en días		25
---------------	---	------------------	--	----

Observaciones	Se aplica un margen de 5 días para realizar pruebas y entregables.			
#	Historia de Usuario	Encargados	Duración	Prioridad
1	HU-5	John Diaz	8	Alta
2	HU-6	John Diaz	10	Alta
3	HU-11	John Diaz	7	Alta

Fuente: Elaboración del Autor

2.2.1.2. Estructura de Desglose de Trabajo - EDT.

El objetivo de realizar un EDT (Estructura de Desglose de Trabajo) es identificar y descomponer las tareas efectuadas en el sistema para llevarlas a un nivel más sencillo y de fácil comprensión, en el ANEXO 2: EDT DEL SISTEMA se evidencia la estructura del trabajo.

2.2.1.1. Gestión de Cambios.

En lo relacionado a la Gestión de Cambios durante el proceso de desarrollo para controlar de una mejor forma, esto se detalla en el ANEXO 4: CONTROL DE CAMBIOS se registran los cambios que se realizan en el proceso de implementación del proyecto.

2.2.2. Fase de Modelado.

En esta fase se da inicio a la definir la estructura lógica que va a tener el proyecto y los datos que lo componen, además se implementarán los diagramas UML que aportan en la fase de implementación.

2.2.2.1. Diseño de Base de Datos.

2.2.2.1.1. Modelo Entidad Relación.

A continuación, en la siguiente figura, se representa el diagrama entidad relación que dio como resultado la obtención de requerimientos especificados en puntos anteriores.

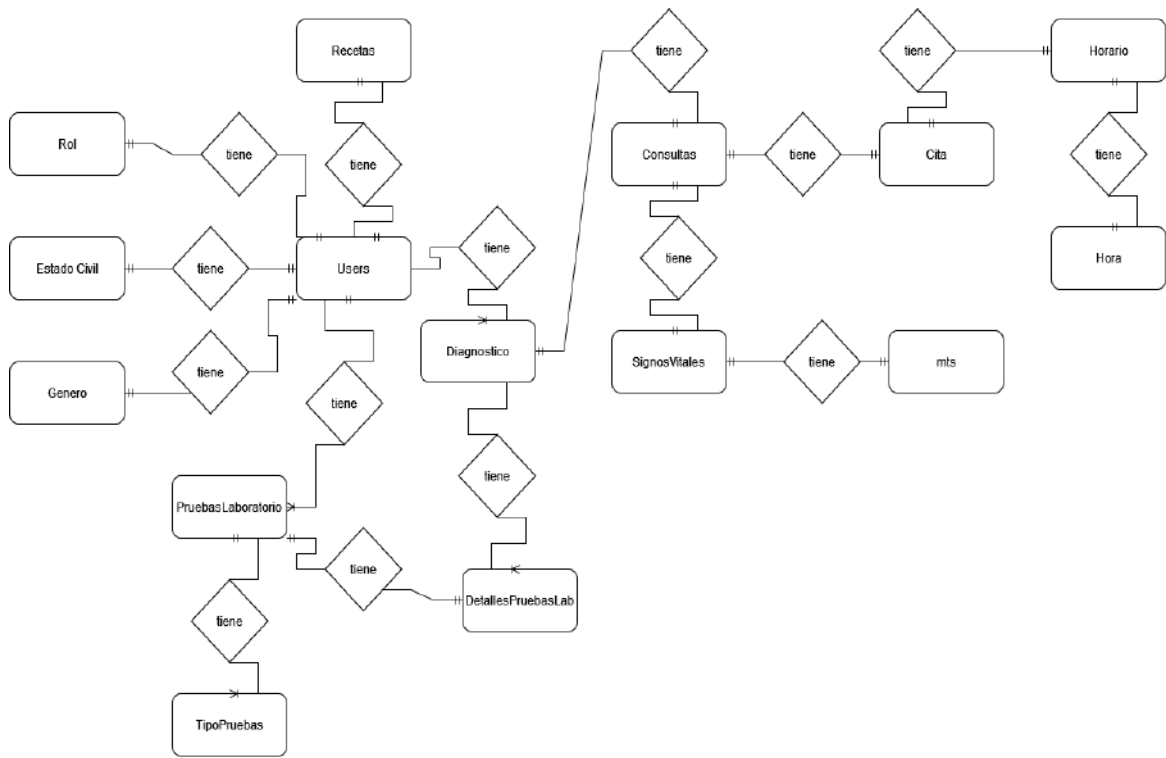


Figura 8: Modelo Entidad- Relación del Sistema

Fuente: Elaboración del Autor

2.2.1.1.1. Modelo Relacional.

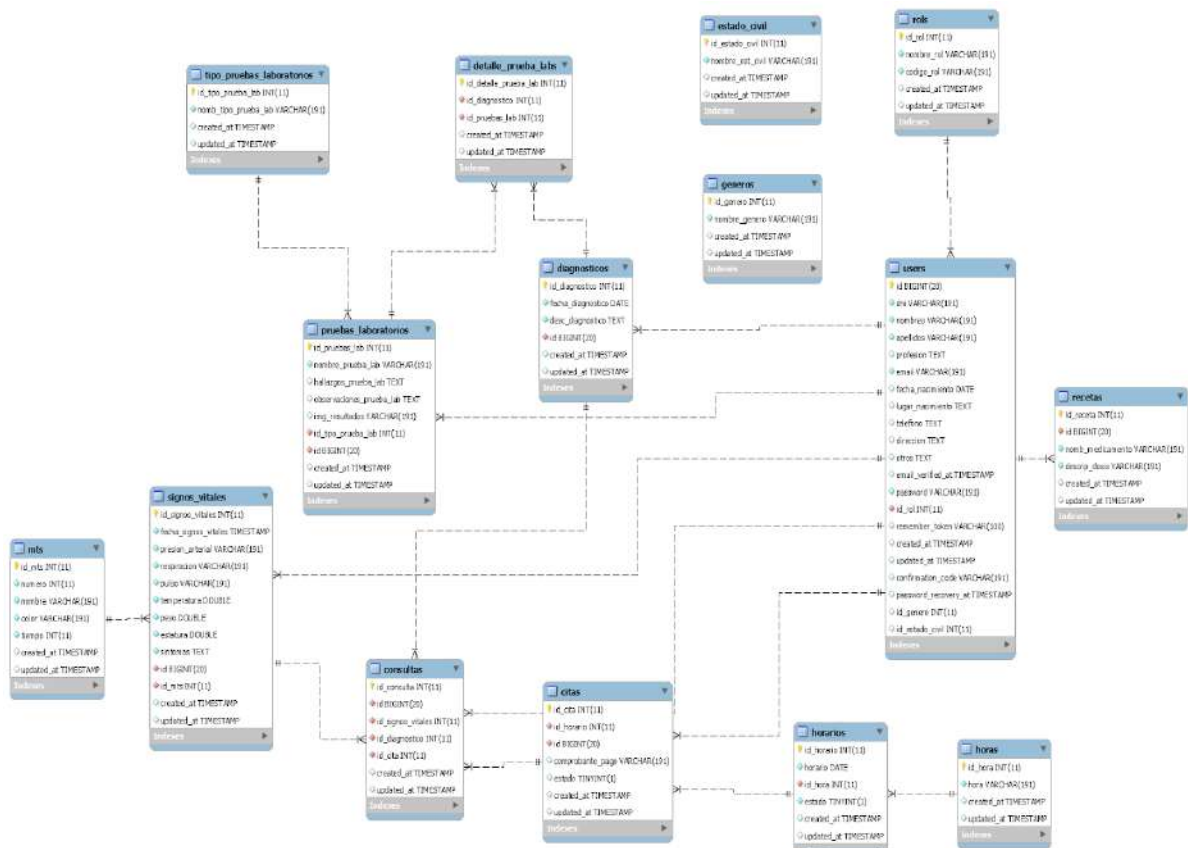


Figura 9: Modelo Relacional del Sistema

Fuente: Elaboración del Autor

2.2.1.1.2. Diseño de Enlaces Navegacionales.

El diseño de enlaces navegacionales se realiza con el objetivo de identificar los componentes gráficos que posee el sistema y la respuesta que efectúa ante la acción de los usuarios.

A continuación, se presenta el diseño de modelo de navegación de forma general que presenta el sistema, en el que se muestran sólo las páginas de acceso general debido a que las demás dependen que el usuario esté logueado.

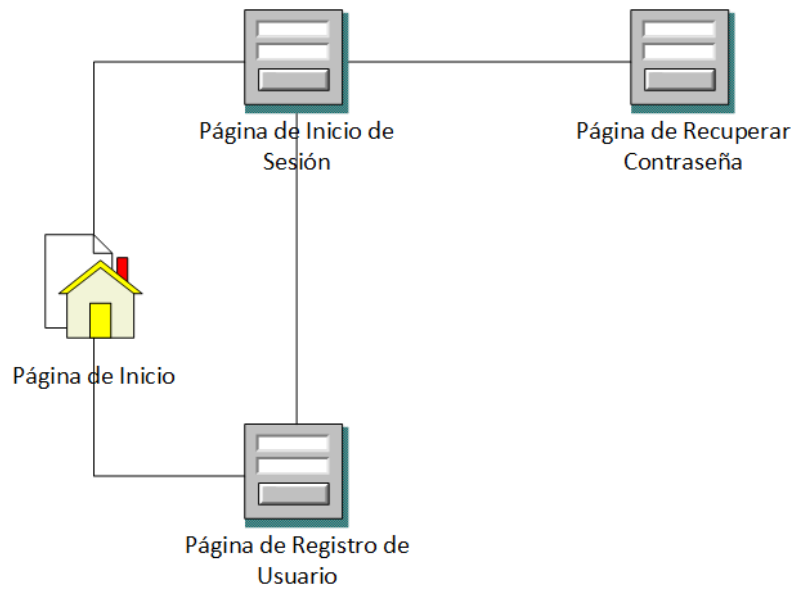


Figura 10: Enlaces Navegacionales Generales del Sistema

Fuente: Elaboración del Autor

En la figura anterior muestra en enlace navegacional con las limitaciones que el sistema brinda para el acceso a usuarios sin tener un previo registro en el sistema, el usuario tipo paciente tiene acceso a lo inicial y si se loguea como paciente puede visualizar un grupo de opciones como, cargar exámenes, cargar comprobante de pago, ver historial clínico y por supuesto agendar una cita.

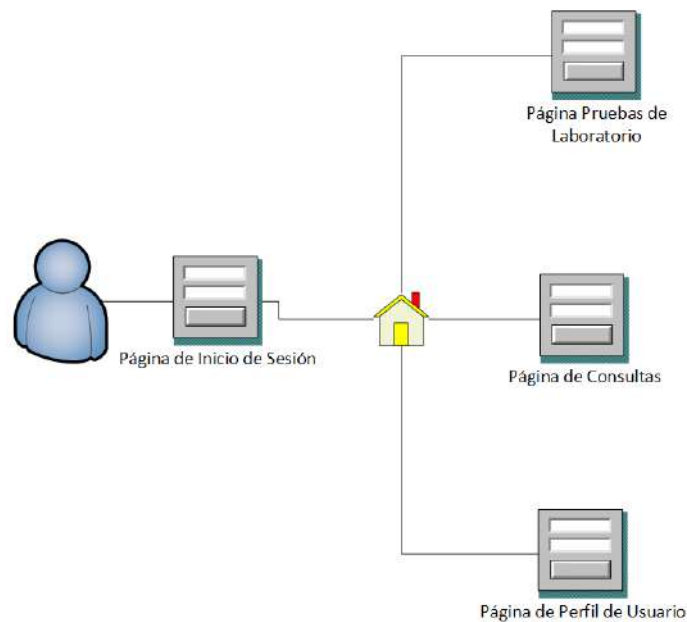


Figura 11: Enlaces Navegacionales Usuario Paciente

Fuente: Elaboración del Autor

El usuario tipo enfermera tiene acceso en el menú con la opción de panel de administración, esta opción permite tener acceso a algunas funciones del panel de administración.

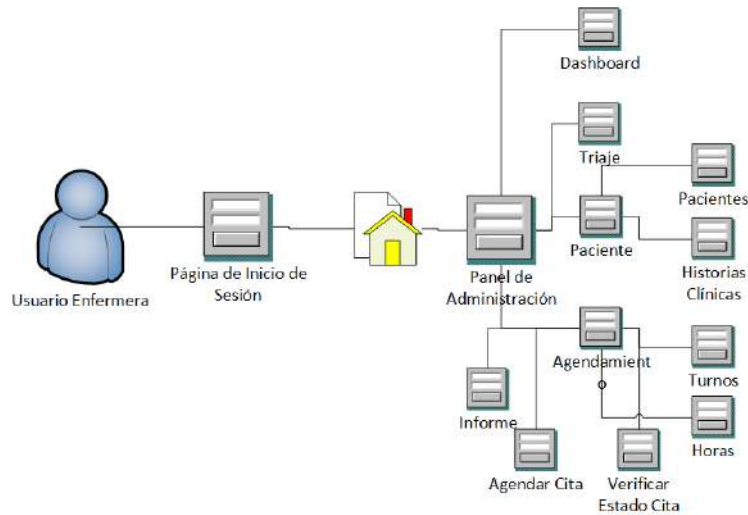


Figura 12: Enlaces Navegacionales Usuario Paciente

Fuente: Elaboración del Autor

Y por último el usuario de tipo doctor tiene acceso a todas las funcionalidades del sistema, a continuación, se presenta el enlace de navegación.

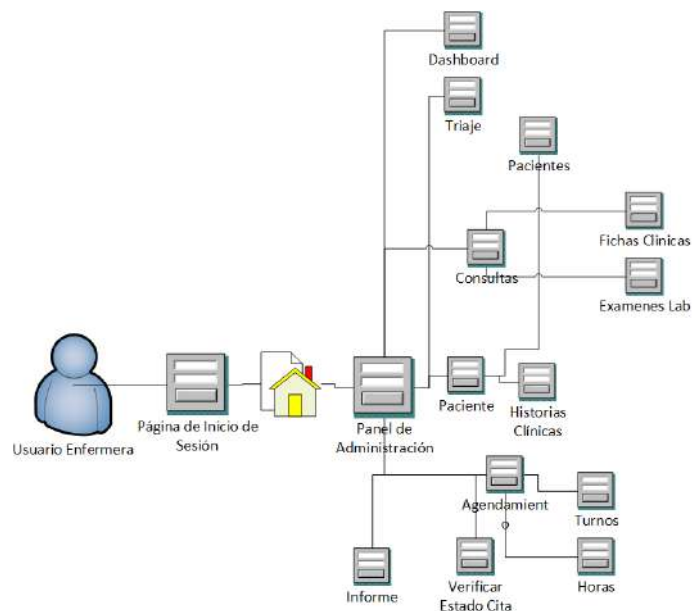


Figura 13: Enlaces Navegacionales Usuario Paciente

Fuente: Elaboración del Autor

2.2.1.3. Tendencias de Diseño.

La metodología SWIRL, brinda un grupo de recomendaciones que son importantes en el proceso de implementación de interfaces, por ello se debe tomar en cuenta algunas tendencias de diseño que aporten al desarrollo del sistema y a la calidad del mismo como:

Flat Design (Diseño Plano).

Responsive Design (Diseño Adaptable).

La tendencia de responsive design es de un alto grado de importancia en la aplicación al sistema web que se está desarrollando, esto a qué surge la necesidad de que el sistema sea adaptable a cualquier clase de ordenador o dispositivo móvil. En cuanto a la tendencia de flat design es considerado un estándar que se implementa en el diseño de interfaces UI, esté en la actualidad es tendencia debido a que proporciona muchos recursos que mejora significativamente la calidad visual de un sistema, presentando al usuario interfaces mucho más usables y amigables.

2.2.1.4. Prototipado de Interfaces de Usuario.

Con la ayuda de la herramienta Balsamiq Mockup en la versión 3.5.17, permitió la elaboración de prototipos de interfaces gráficas. A continuación, se muestran los diseños de prototipos elaborados para la implementación en el sistema según los módulos expuestos en secciones anteriores.

Interfaces a Usuarios General

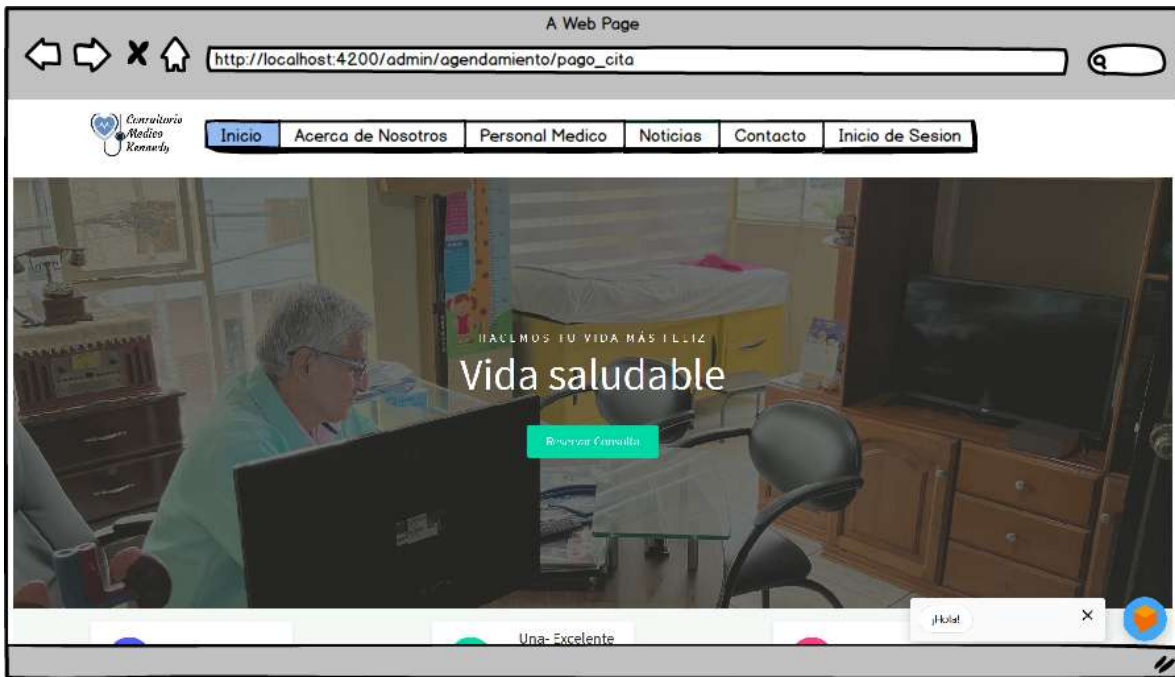


Figura 14: Prototipo - Página Inicial

Fuente: Elaboración del Autor

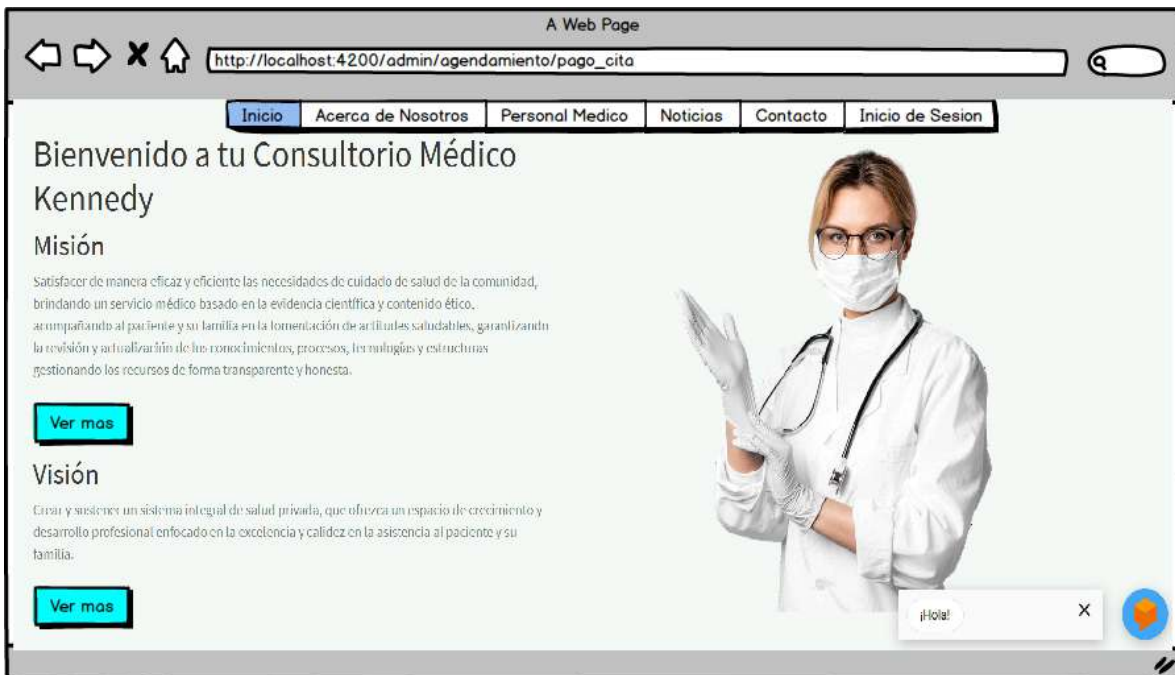


Figura 15: Prototipo - Página Inicial – 2

Fuente: Elaboración del Autor

En las figuras anteriores se puede evidenciar la página de inicio del sistema, dentro de ella pueden acceder cualquier usuario tenga o no cuenta en la aplicación web. El diseño que posee es tipo one page, es decir un grupo de componentes dentro de la

misma página. En ella se puede encontrar con la imagen del médico que atiende en el consultorio y sobre la imagen un botón que envía a la sección para agendar una cita médica, de la misma forma deslizando hacia abajo se puede encontrar con la misión y visión del consultorio, y además de ello contiene un footer con enlaces a redes sociales y la ubicación del consultorio. Como un plus dentro de esta página aparece el chatbot que es un elemento innovador dentro de este tipo de servicios.



Figura 16: Prototipo - Página Acerca de Nosotros

Fuente: Elaboración del Autor

Dentro del prototipo de interfaz se puede observar un diseño tipo one page, dentro de ella se aprecia información como misión, visión, valores que se pone en práctica en el consultorio y la información del médico principal, es un diseño donde se puede apreciar toda la información de una forma detallada, de la misma forma contiene el chatbot para que el usuario pueda interactuar con el sistema.

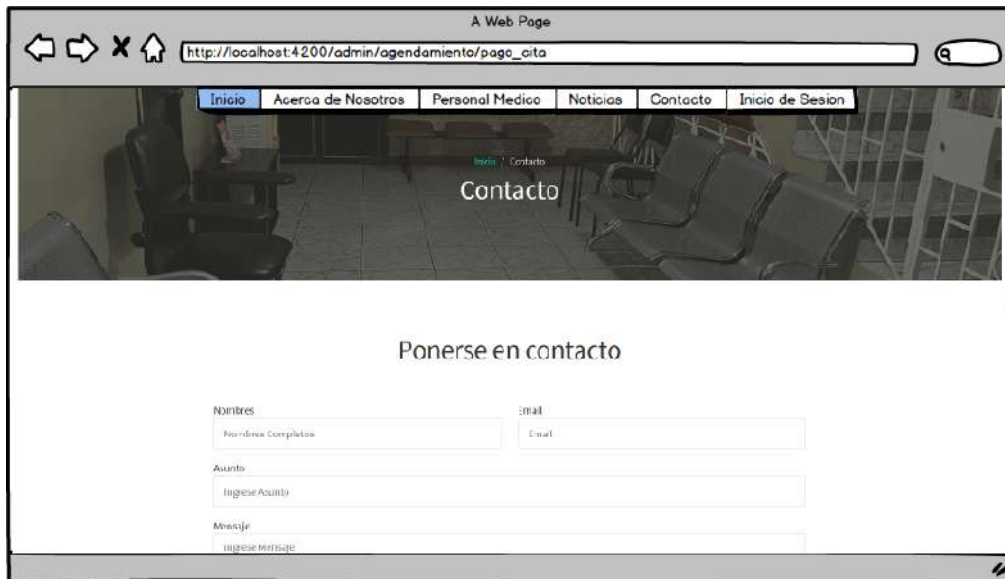


Figura 17: Prototipo - Página Contacto

Fuente: Elaboración del Autor

Dentro del prototipo de interfaz de Contacto se evidencia los campos necesarios a llenar para solicitar algún tipo de información en concreto al personal que labora en el consultorio, es otro medio de comunicación entre el usuario y el personal que labora.



Figura 18: Prototipo - Página Noticias

Fuente: Elaboración del Autor

En la sección de prototipo de interfaz de noticias, es un diseño que proporciona algunas imágenes con títulos que especifican de qué se trata, si se da clic nos enlazará con sitios web de interés acerca del tema seleccionado. Todas las fuentes a las que se enlaza contienen información valiosa para el cuidado de la salud.

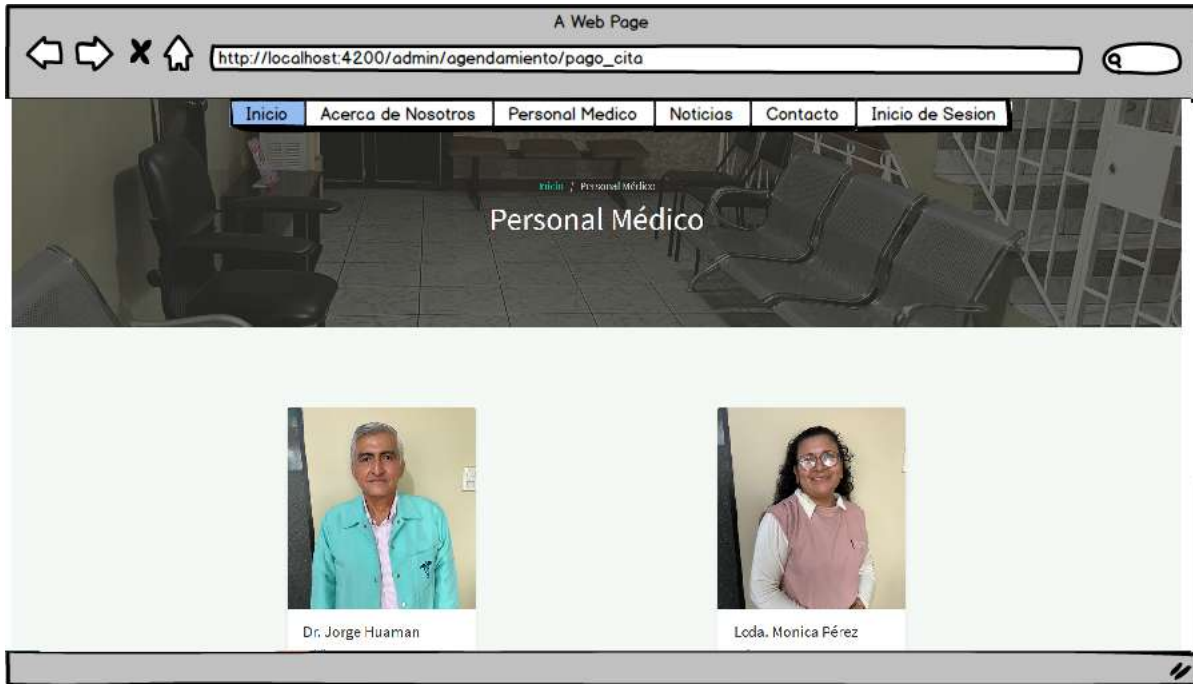


Figura 19: Prototipo - Página Personal Médico

Fuente: Elaboración del Autor

En lo correspondiente al diseño de prototipo de la sección de Personal Médico, es sumamente amigable, contiene información relacionada al personal médico que labora dentro del consultorio, si se pasa el punto sobre las imágenes del médico o enfermera tiene la posibilidad de establecer contacto con ellos, vía llamada telefónica o por medio de WhatsApp.

Login y Registro de Usuario

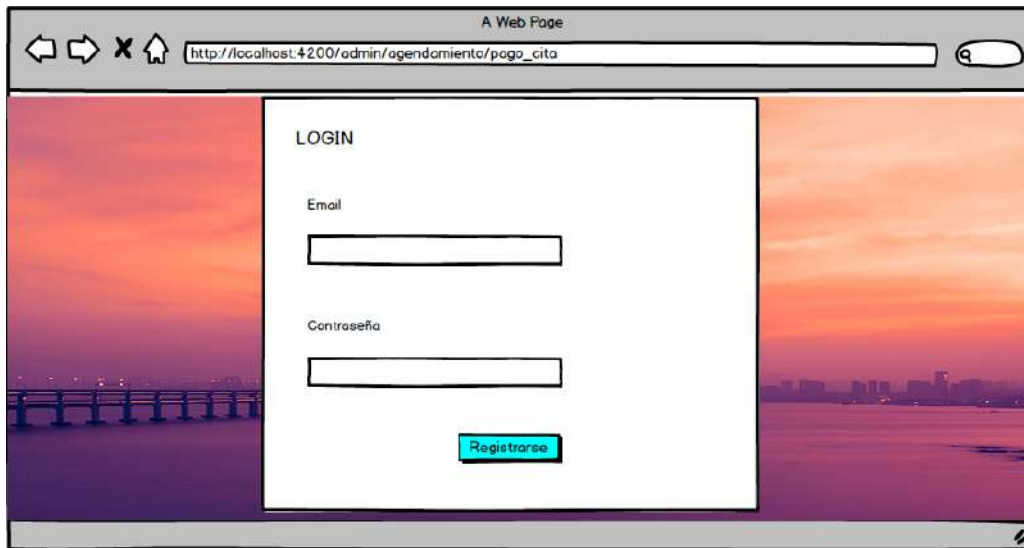


Figura 20: Prototipo - Página Login

Fuente: Elaboración del Autor

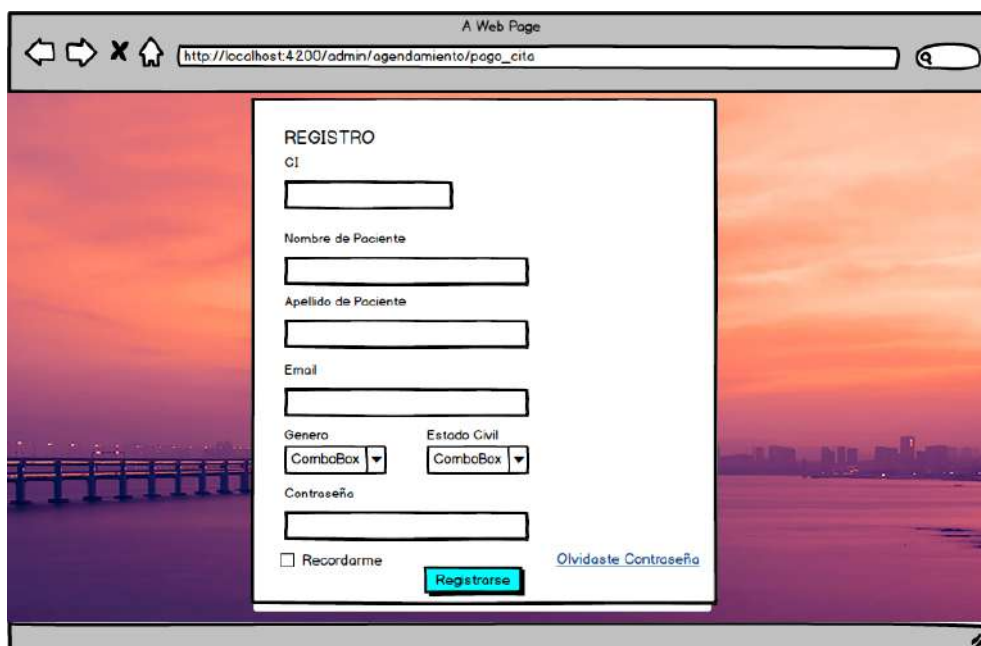


Figura 21: Prototipo - Página Registro de usuario

Fuente: Elaboración del Autor

El diseño del prototipo de la interfaz de registro y login es amigable y sumamente entendible ya que es un diseño estándar de sistemas, en registros cuenta con los campos que solicita el sistema al usuario para crear una respectiva cuenta, los

componentes utilizados son idóneos para cada campo. De la misma forma en login, solo el usuario debe llenar los campos de email y password.

Páginas de usuario Paciente



Figura 22: Prototipo - Página Inicio - Usuario Paciente

Fuente: Elaboración del Autor

Una vez logueado el usuario con el rol de paciente, tiene acceso a más de las opciones que vienen por defecto, también a Consultas donde visualizará las consultas que ha realizado, Pruebas de Laboratorio donde se permite que cargue resultados de exámenes enviados por el médico y por supuesto el perfil de usuario donde puede corregir alguna información personal errónea.

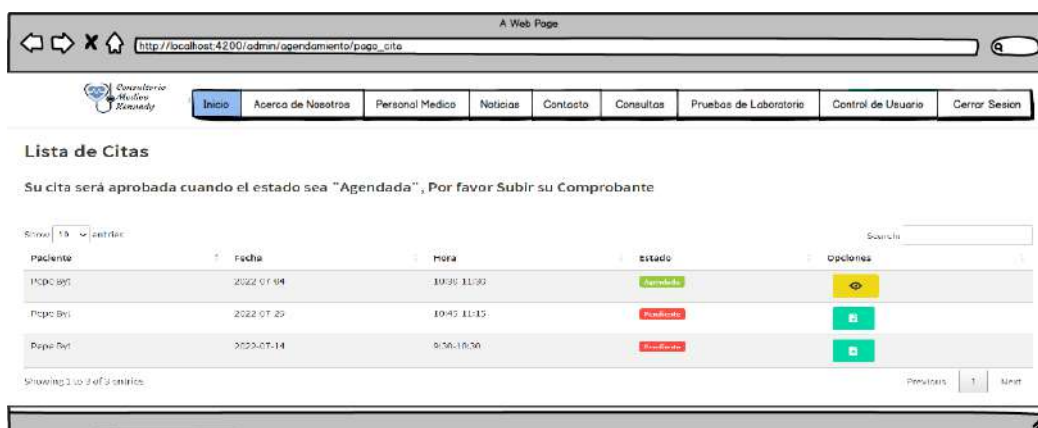


Figura 23: Prototipo - Página Consultas - Usuario Paciente

Fuente: Elaboración del Autor

En la sección de Consultas se evidencia, un listado de las citas médicas realizadas por el usuario, como se puede evidenciar es una tabla con sus respectivos encabezados y sus datos, en ella se muestra el estado de la cita si es color verde es

porque está agendada y validada por el consultorio médico lista para que el paciente asista a atenderse y posee un botón de opciones en el que por defecto viene la simbología de cargar un archivo ya que es para que el paciente cargue en un documento con el comprobante de pago para validar la cita, si el usuario ya cargo el comprobante puede visualizar igual su archivo.

A Web Page
 http://localhost:4200/admin/agendamiento/pago_cita

PERFIL DE USUARIO

CI

Nombre de Paciente

Apellido de Paciente

Email

Genero Estado Civil

Contraseña

Guardar

Figura 24: Prototipo - Página Perfil usuario - Usuario Paciente

Fuente: Elaboración del Autor

El usuario puede visualizar su perfil también accediendo a esa opción, y puede editar su información personal si es que ingresó algún dato erróneo.

A Web Page
 http://localhost:4200/admin/agendamiento/pago_cita

Consultorio Médico Kennedy

Inicio Acerca de Nosotros Personal Médico Noticias Contacto Consultas Pruebas de Laboratorio Control de Usuario Cerrar Sesión

Exámenes de Laboratorio

Cargar los resultados de los Exámenes de Laboratorio en PDF

Show 10 entries

Search:

Prueba	Tipo de Prueba	Opciones
1 to	Indicador biología	<input type="button" value="D"/> <input type="button" value="E"/>

Showing 1 to 1 of 1 entries

Previous 1 Next

Figura 25: Prototipo - Exámenes Laboratorio - Usuario Paciente

Fuente: Elaboración del Autor

En la sección de exámenes de laboratorio se presenta una tabla que consta de títulos en base al módulo y sus respectivos datos, con una sección de opciones que permite cargar un documento que en este caso sería los resultados de exámenes solicitados previamente por el médico.

Páginas de Usuario Enfermera

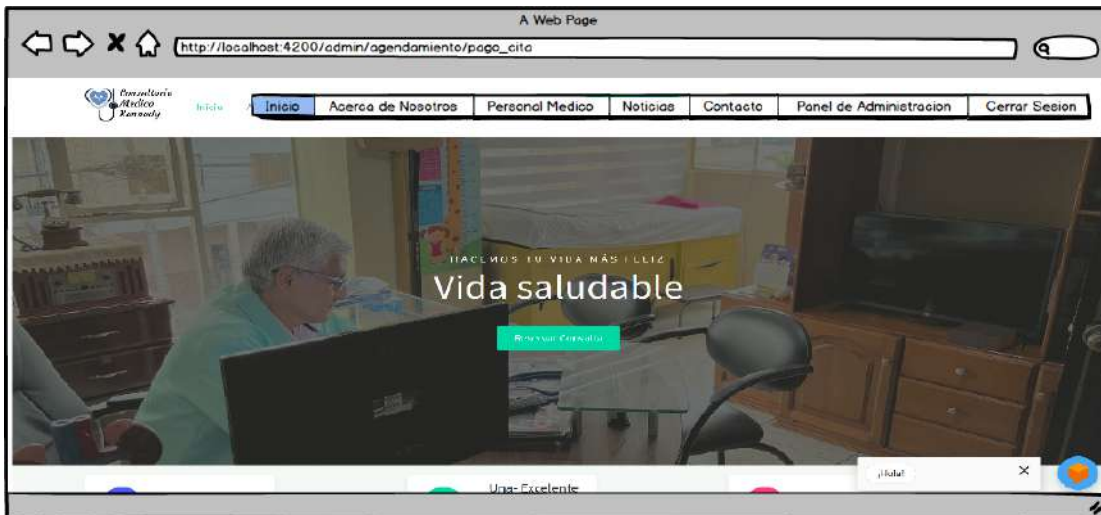


Figura 26: Prototipo - Página Principal - Usuario Enfermera

Fuente: Elaboración del Autor

En el diseño del prototipo de la interfaz correspondiente a la principal para el usuario enfermera, se puede apreciar que dentro del menú bar contiene la opción de Panel de Administración en el cual solo tiene acceso personal que tiene el rol de enfermera o doctor.

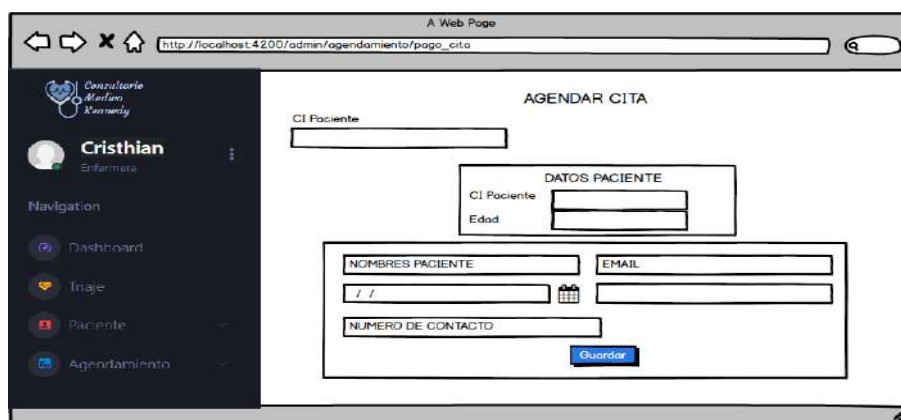


Figura 27: Prototipo - Agendar Cita - Usuario Enfermera

Fuente: Elaboración del Autor

Dentro de la opción del menú del componente sidebar de Agendar Cita se presenta el diseño donde consten los campos necesarios para que la enfermera pueda agendar una cita a un paciente.

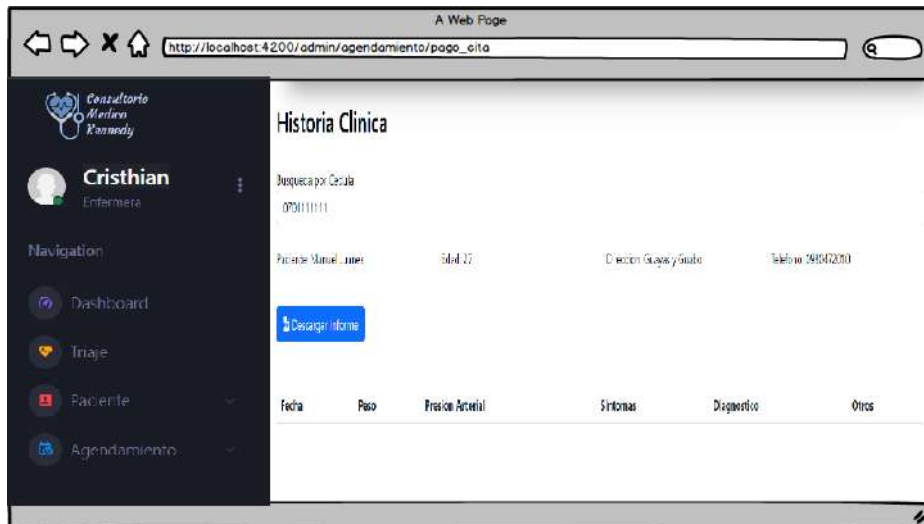


Figura 28: Prototipo - Historia Clínica - Usuario Enfermera

Fuente: Elaboración del Autor

En lo correspondiente al diseño del prototipo de historia clínica, los usuarios de tipo enfermera pueden acceder a esta opción con la finalidad de proporcionar a los pacientes un informe de historias clínicas, deben llenar el campo de cédula para que aparezca la información del paciente seleccionado.

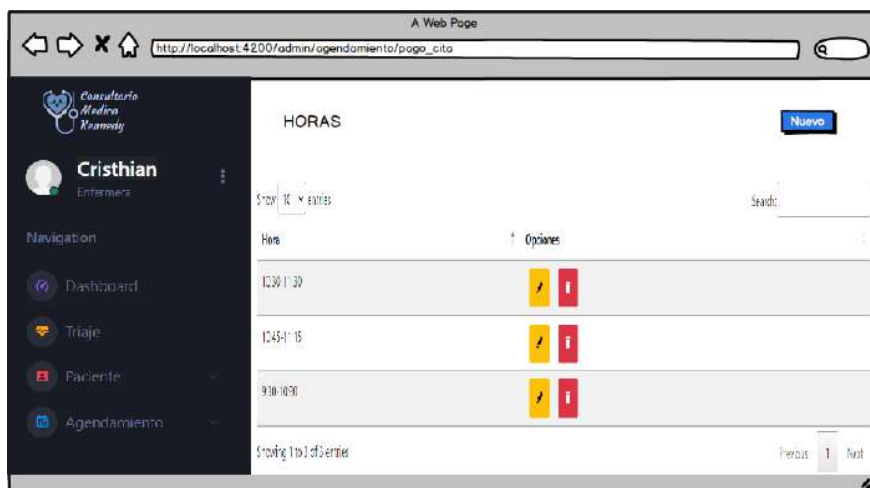


Figura 29: Prototipo - Página Horas - Usuario Enfermera

Fuente: Elaboración del Autor

En el diseño correspondiente a la opción del menú de horas, se presenta una tabla con las horas creadas para los turnos que se proyectarán como opción a los pacientes para que seleccionen y sean atendidos, dentro de la tabla también el usuario enfermera puede editar o borrar dicha hora creada.

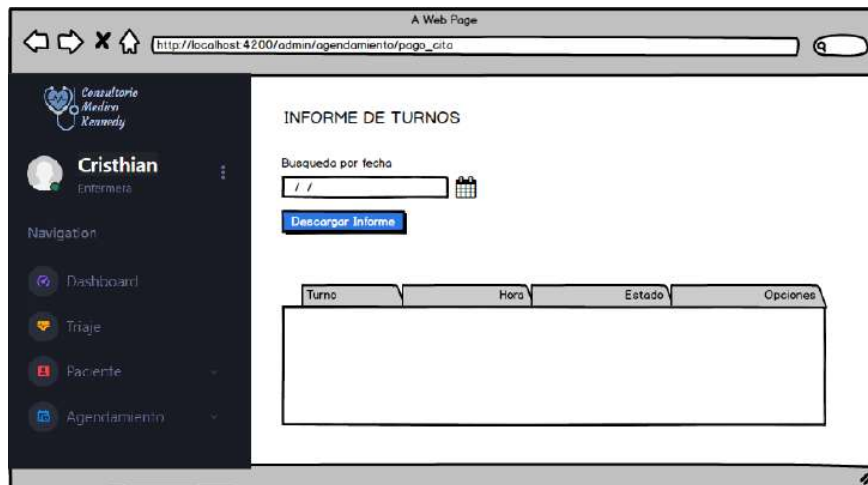


Figura 30: Prototipo - Informe de Turnos - Usuario Enfermera

Fuente: Elaboración del Autor

El usuario tipo enfermera, tiene acceso a descargar informe por fechas para el control de los pacientes atendidos.

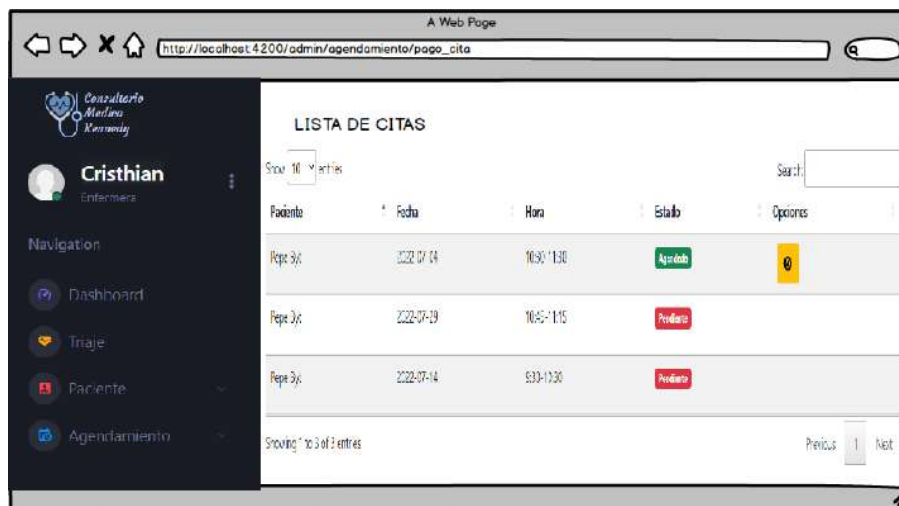


Figura 31: Prototipo - Página Lista de Citas - Usuario Enfermera

Fuente: Elaboración del Autor

En el prototipo de interfaz de lista de citas, la enfermera tiene opción a revisar las citas realizadas por los usuarios pacientes, mediante una tabla se puede apreciar los campos necesarios para que se identifique a un determinado paciente, de la

misma forma se presenta las opciones visualizar si el paciente ha cargado el comprobante de pago para validar la cita.

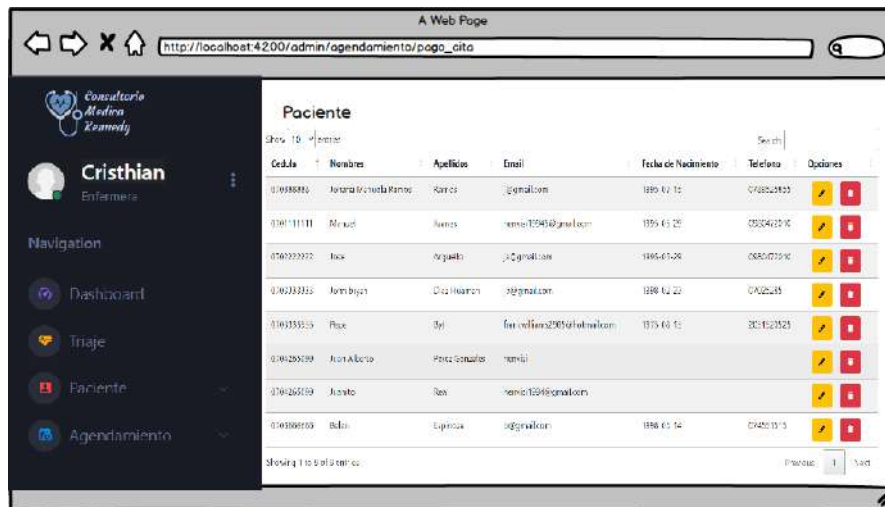


Figura 32: Prototipo - Página Pacientes- Usuario Enfermera

Fuente: Elaboración del Autor

En lo correspondiente a la sección de pacientes se presenta un diseño básico y entendible que contiene una tabla en la cual consta la lista de pacientes y una columna de opciones donde la enfermera puede editar los campos o borrar el usuario.

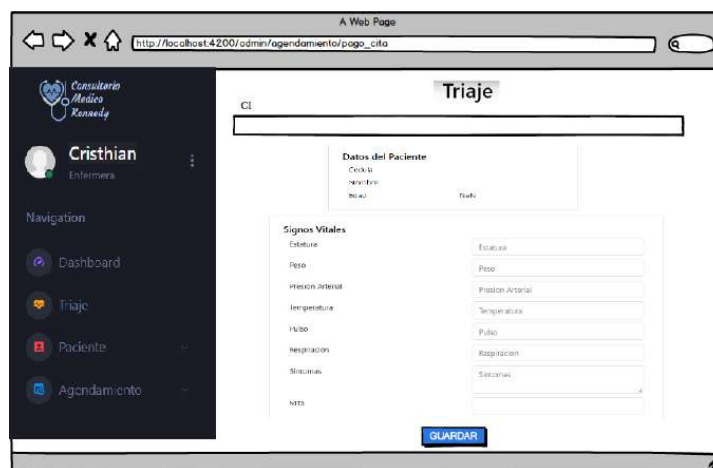


Figura 33: Prototipo - Página Triage - Usuario Enfermera

Fuente: Elaboración del Autor

En el diseño del prototipo de interfaz de triaje se presenta, un campo de texto que es principal que es el de cédula de Identidad, la enfermera debe llenar ese campo para que el sistema cargue el resto de datos de paciente, una vez realizado ese

proceso, el siguiente paso corresponde a llenar los campos de signos vitales previa a la atención con el médico.

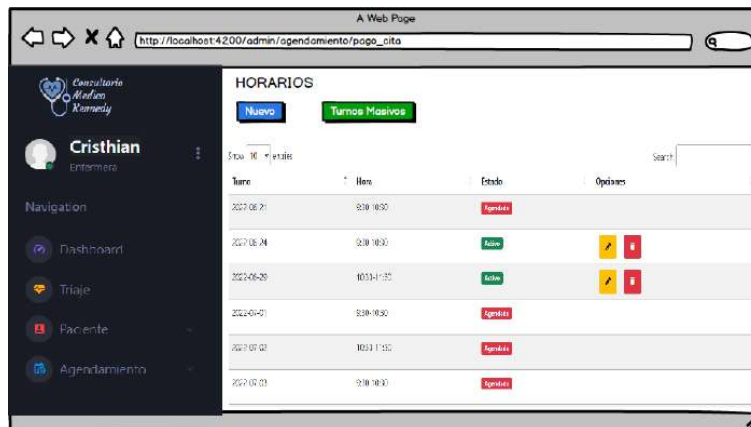


Figura 34: Prototipo - Página Horarios - Usuario Enfermera

Fuente: Elaboración del Autor

En la sección de horarios, se va a mostrar un diseño que contiene en la parte superior dos botones, el de nuevo permitirá que el usuario pueda registrar un nuevo horario, y si se da clic en el botón de turnos masivos permite cargar un archivo .xls con los turnos de un mes citando un ejemplo, además puede mediante las opciones eliminar o editar algún turno siempre y cuando no esté seleccionado por algún paciente.

Páginas de Usuario Doctor

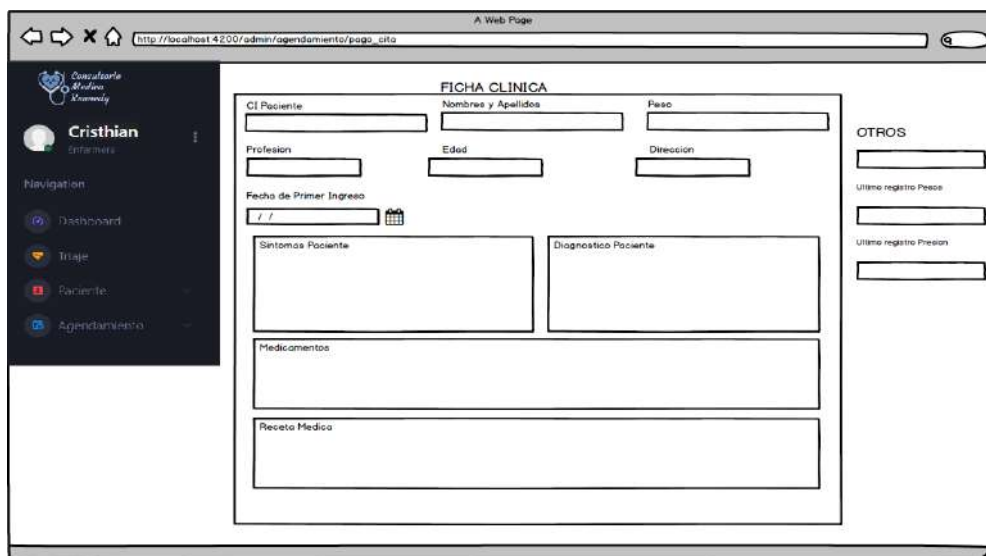


Figura 35: Prototipo - Página Ficha Clínica - Usuario Doctor

Fuente: Elaboración del Autor

EL usuario de tipo doctor tiene acceso a esta opción que corresponde a la ficha clínica, en ella se puede apreciar que contiene información referente al paciente que está siendo atendido, además de ellos se debe poner la descripción de síntomas, además de ellos el respectivo diagnóstico, también tiene un cuadro de texto donde el doctor debe escribir los medicamentos que le envía y de la misma forma la receta médica que en este caso sería la forma de tomar los medicamentos.

En la parte de la derecha se encuentra unos registros de otros, aquí se detalla alguna anomalía o tipo de alergia que tenga el paciente, además de ellos los 3 últimos registros de peso y presión arterial como método para visualizar el estado del paciente.

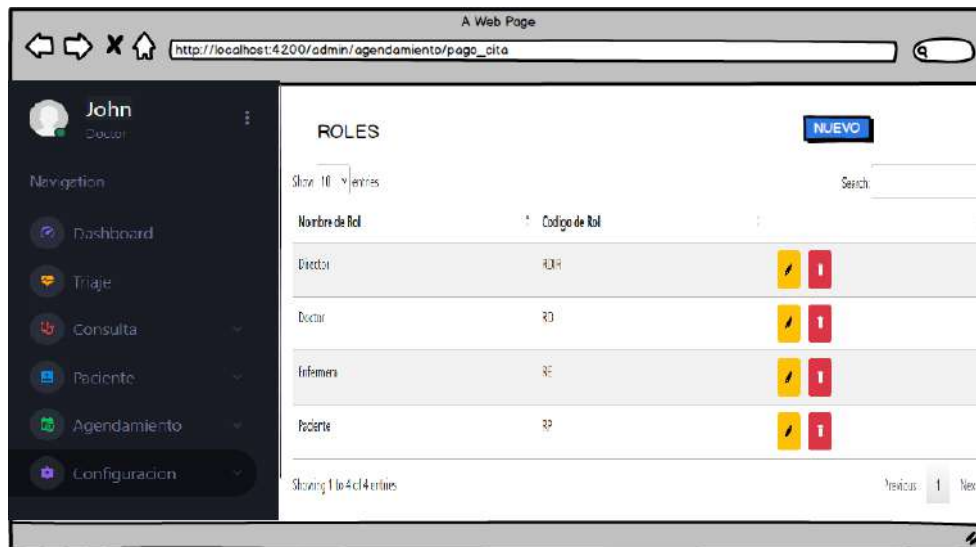


Figura 36: Prototipo - Página Roles - Usuario Doctor

Fuente: Elaboración del Autor

El doctor tiene acceso a la sección de configuración en el componente sidebar, dentro de la misma el doctor puede gestionar nuevos roles para el sistema.



Figura 37: Prototipo - Página Tipo Pruebas Laboratorio - Usuario Doctor

Fuente: Elaboración del Autor

En esta sección de tipos de prueba de laboratorio se la diseñó con el afán de que el doctor registre algún tipo de prueba de laboratorio a enviar al paciente, esto como plan para que el sistema tenga un gran alcance en este ámbito debido a que existen muchos tipos de pruebas de laboratorio.

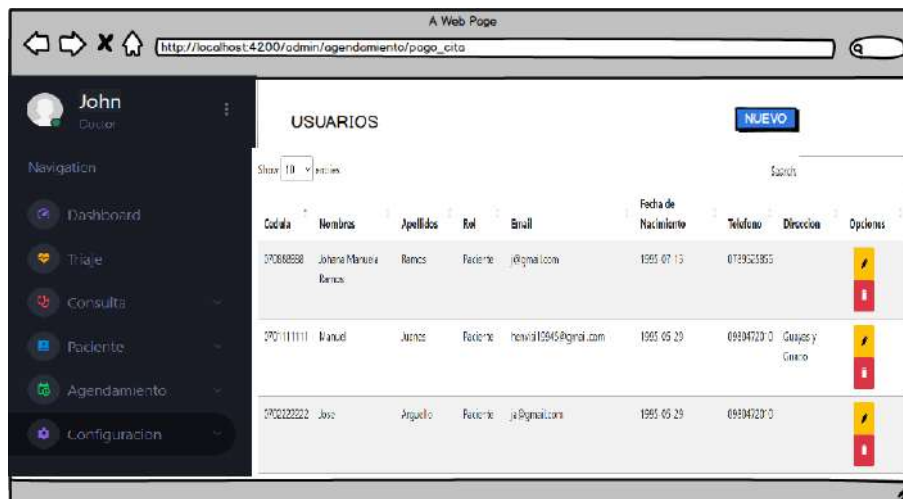


Figura 38: Prototipo - Página Usuarios - Usuario Doctor

Fuente: Elaboración del Autor

El doctor al ser el usuario con acceso a todas las funcionalidades, puede dar de baja a algún usuario, en el diseño se puede ver que contiene una tabla con los datos de los usuarios y la columna de opciones donde tiene los botones de eliminar y editar.

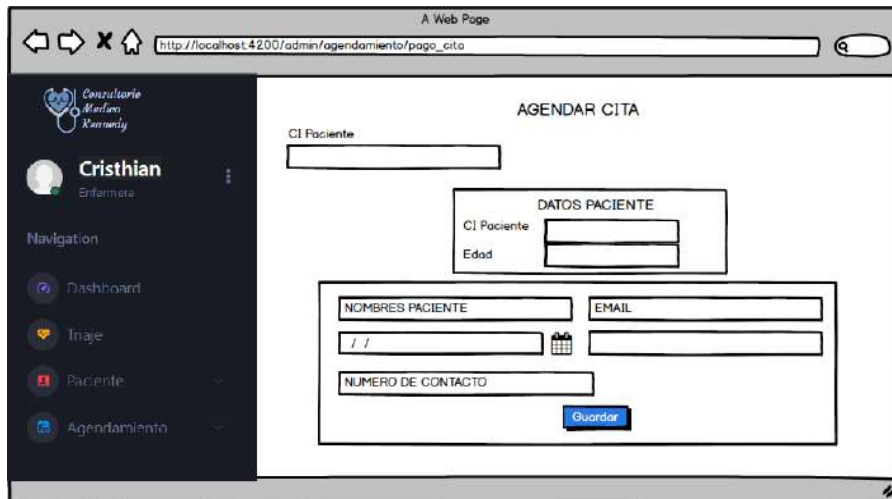


Figura 39: Prototipo - Página Agendar Cita - Usuario Doctor

Fuente: Elaboración del Autor

De la misma forma que el usuario tipo enfermera puede agendar una cita, el usuario tipo doctor también puede realizar esta función y de la misma forma solo se debe llenar los datos con la información del paciente solicitante de la cita.

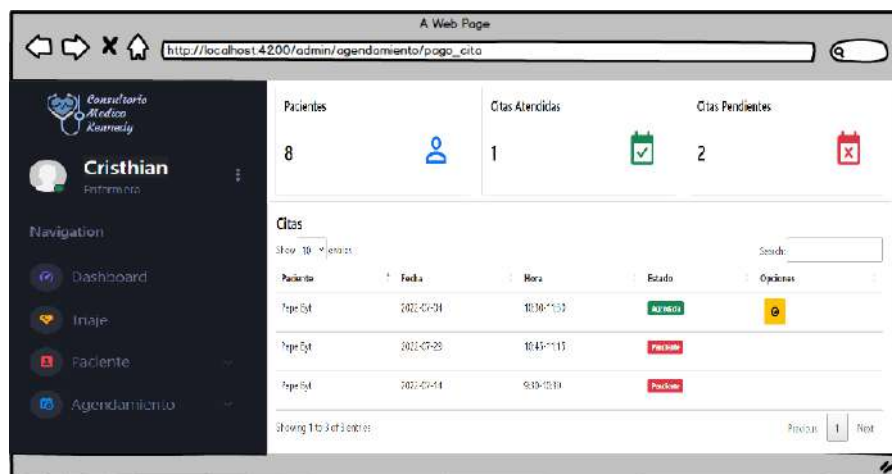


Figura 40: Prototipo - Página Dashboard - Usuario Doctor

Fuente: Elaboración del Autor

Lógicamente el usuario tipo doctor tiene acceso a el dashboard del sistema, en el que consta las estadísticas de los pacientes registrados, las citas médicas atendidas y por supuesto las citas de los pacientes en total que están pendientes. Además contiene la tabla de las citas de los pacientes con el estado de la cita para mayor referencia.

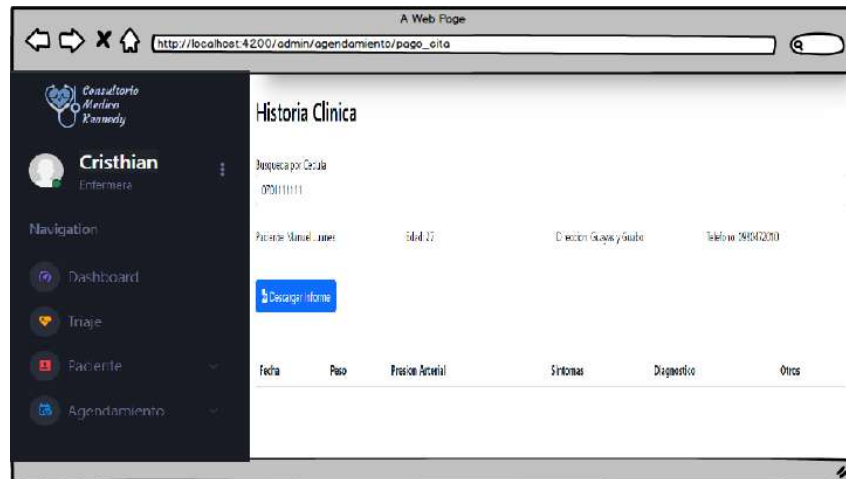


Figura 41: Prototipo - Página Historia Clínica - Usuario Doctor

Fuente: Elaboración del Autor

En lo correspondiente a la sección de historia clínica es el mismo diseño para el usuario con tipo doctor, donde puede proporcionar al paciente la historia clínica dentro del consultorio con sus registros de atención médica en un informe.

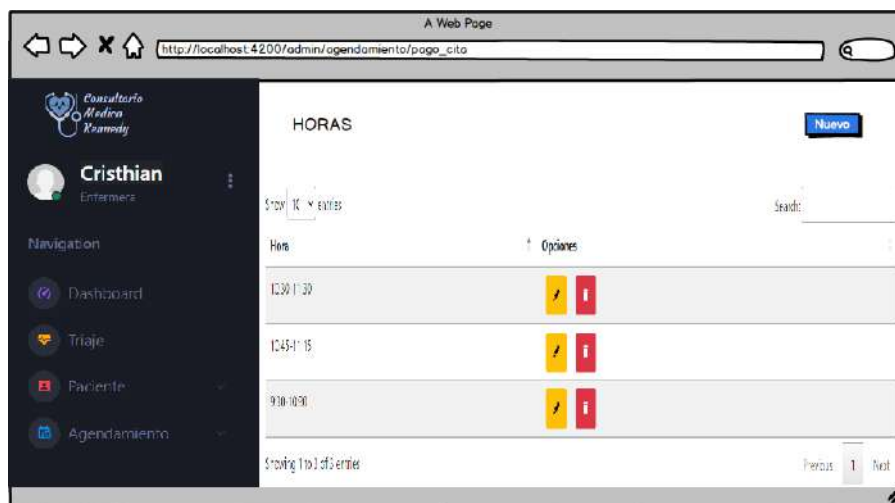


Figura 42: Prototipo - Página Horas - Usuario Doctor

Fuente: Elaboración del Autor

El usuario de tipo doctor también presenta una interfaz para crear horas de atención lógicamente que proporcionará a sus pacientes para que ellos elijan el horario a su gusto y disposición, es un diseño con una tabla que contiene los datos y la columna con los botones de editar y eliminar.

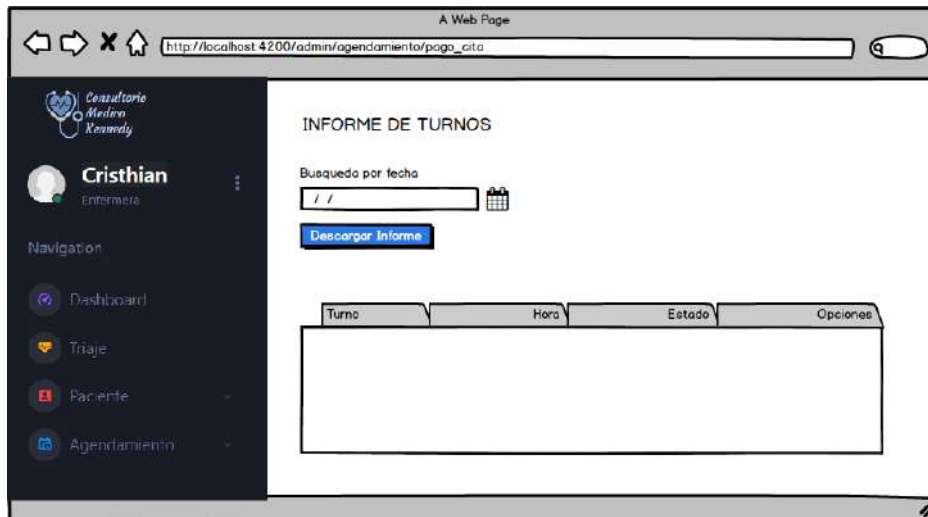


Figura 43: Prototipo - Página Informe de Turnos - Usuario Doctor

Fuente: Elaboración del Autor

Como medio de control de turnos el doctor tiene acceso a generar un informe respecto a este tipo de información, posee un diseño con una tabla y el botón de descargar el informe solicitado filtrado por fecha.

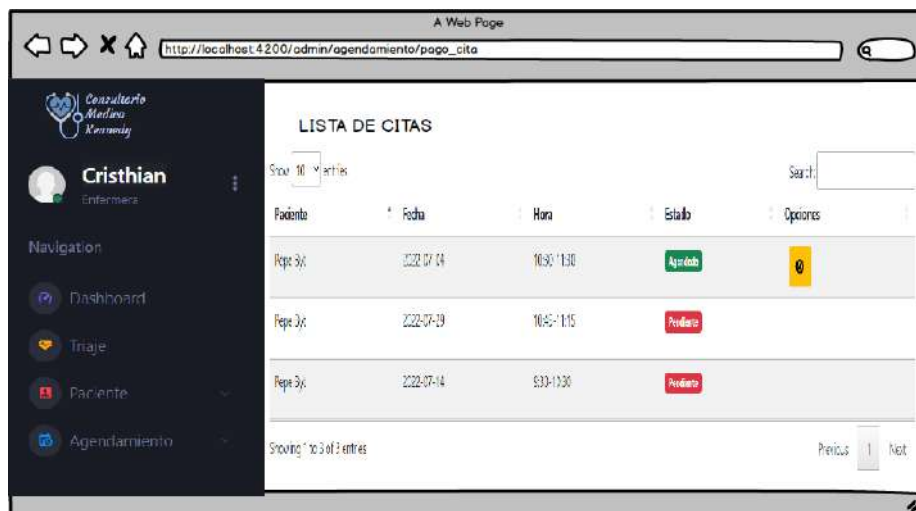


Figura 44: Prototipo - Página Lista de Citas - Usuario Doctor

Fuente: Elaboración del Autor

El diseño de la interfaz de lista de citas de la misma forma contiene una tabla con los datos de las citas y la opción de visualizar el archivo si el paciente ya cargo el comprobante, caso contrario no se activa este botón y el estado de la cita permanecerá en Pendiente.

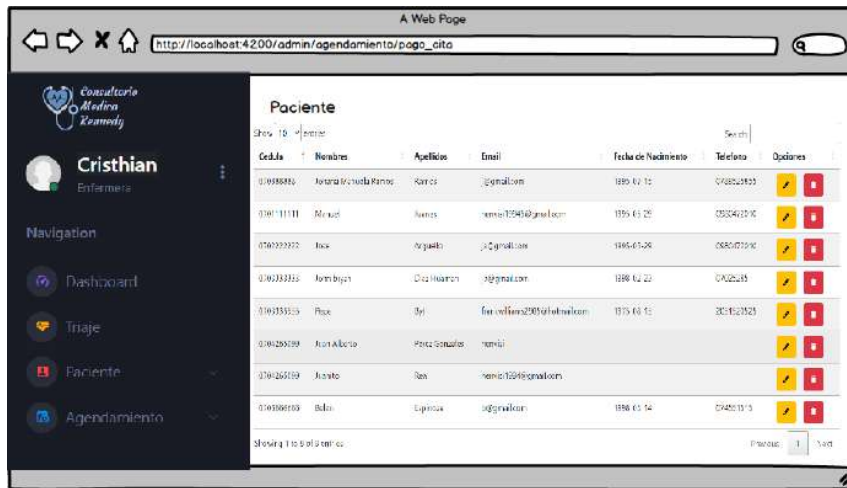


Figura 45: Prototipo - Página Paciente - Usuario Doctor

Fuente: Elaboración del Autor

El doctor tiene acceso a ver la lista de pacientes que posee el sistema médico, puede ver sus datos mediante una tabla que los contiene, y de la misma manera si lo amerita puede editar la información o a su vez eliminar a algún usuario.

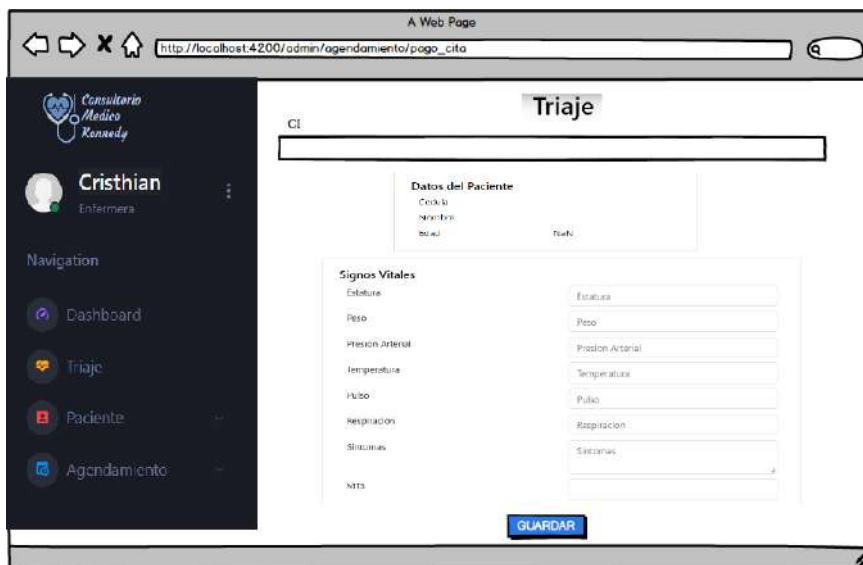


Figura 46: Prototipo - Página Triaje - Usuario Doctor

Fuente: Elaboración del Autor

El diseño de la interfaz de triaje contiene los campos necesarios para hacer el registro previo del paciente de sus signos vitales y posteriormente pueda acudir a su cita.

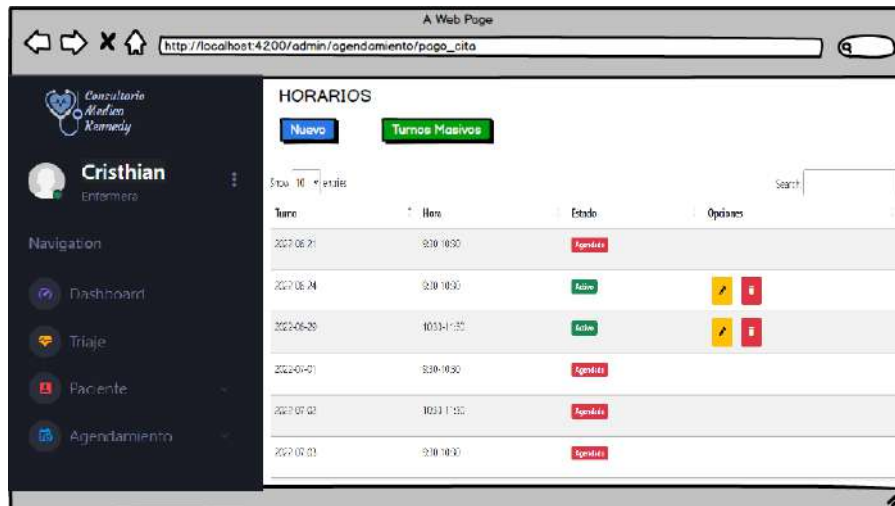


Figura 47: Prototipo - Página Horarios - Usuario Doctor

Fuente: Elaboración del Autor

El diseño del prototipo de interfaz de horarios, tiene la misma línea que los demás, contiene una tabla con los datos, dos botones en la parte superior donde el doctor puede crear un solo turno dando clic en nuevo y si se da clic en turnos masivos puede cargar un archivo .xls con los horarios y fechas, cabe mencionar que en ese documento sólo pueden tener las horas creadas en el sistema si se pone otras horas no se guardarán.

2.2.1.1. Diseño de Diagramas UML.

En la elaboración de esta propuesta tecnológica es de suma importancia considerar los diagramas UML debido a que ayudan a la identificación de los componentes, estructura y flujo por medio del tiempo, lógica y condiciones.

Como diagramas de importancia en la presente propuesta tecnológica del sistema de agendamiento e historias clínicas de consultorio Kennedy se considera a: diagramas de secuencia, diagramas de caso de uso, diagramas de actividades.

2.2.1.1.1. Diagrama de Casos de Uso.

La elaboración de diagramas de caso de uso, aporta en el proceso de interpretación de actividades que realiza cada usuario participante del sistema. De esta forma los diagramas de caso de uso es el medio de representación gráfica de los requisitos funcionales solicitados por los interesados,

constituyendo de esta forma en los diagramas de caso de uso como medio de representar los CRUD y funciones de cada usuario participante en el sistema.

A continuación, el diagrama de caso de uso general del sistema.

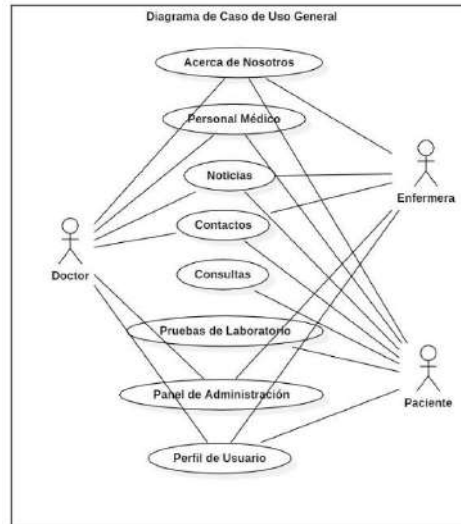


Figura 48: Diagrama de casos de uso Generalizado

Fuente: Elaboración del Autor

En la figura anterior se detalla el caso de uso de las opciones que el usuario se puede encontrar en el menú bar del sitio web, se aprecia que algunas opciones se restringen a ciertos roles de usuario.

Diagramas de Caso de Uso Panel de Administración

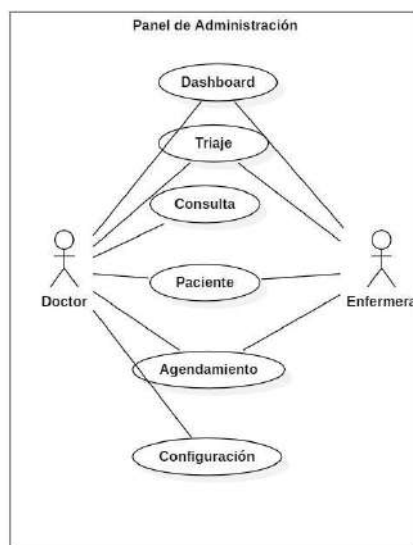


Figura 49: Diagrama de Caso de Uso: Panel de Administración

Fuente: Elaboración del Autor

En el diagrama de caso de uso del panel de administración se puede apreciar que algunas opciones se restringen para el usuario de tipo enfermera, sin embargo, el usuario doctor si tiene permiso y acceso a todas las funcionalidades.

Diagramas de Caso de Uso Paciente

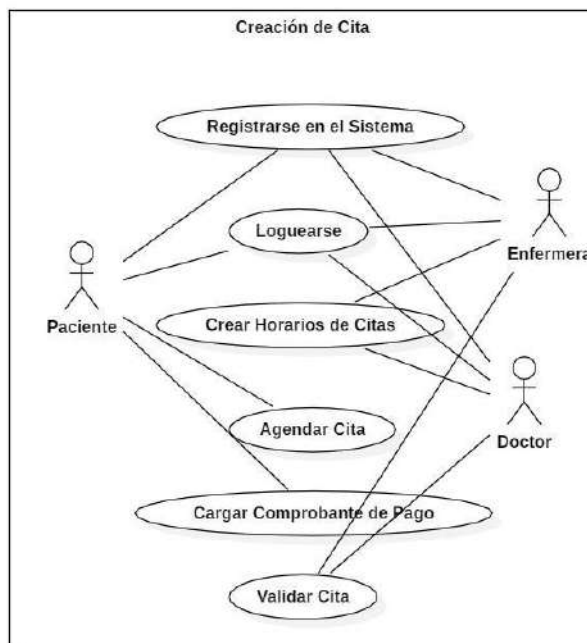


Figura 50: Diagrama de caso de Uso: Creación de Cita

Fuente: Elaboración del Autor

Para la creación de cita interactúan en las actividades a realizar el usuario tipo paciente que tiene acceso a agendar una cita y validar su cita, y de la misma forma el usuario tipo enfermera y doctor que son los encargados de crear horarios de citas.

Diagrama de Caso de Uso Gestión Triaje

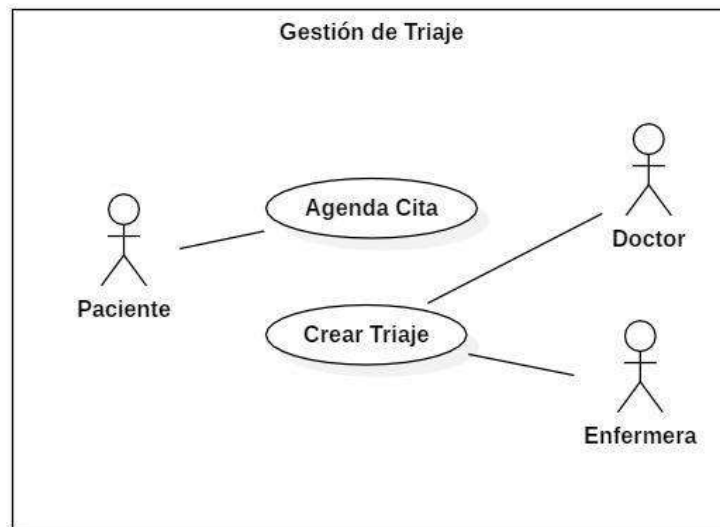


Figura 51: Diagrama de Caso de Uso: Gestión de Triaje

Fuente: Elaboración del Autor

En el diagrama de caso de uso de Gestión de Triaje, participan los 3 actores, el actor paciente es el encargado de agendar una cita, y el doctor o enfermera puede crear el triaje.

Diagrama de Caso de Uso Gestión Paciente

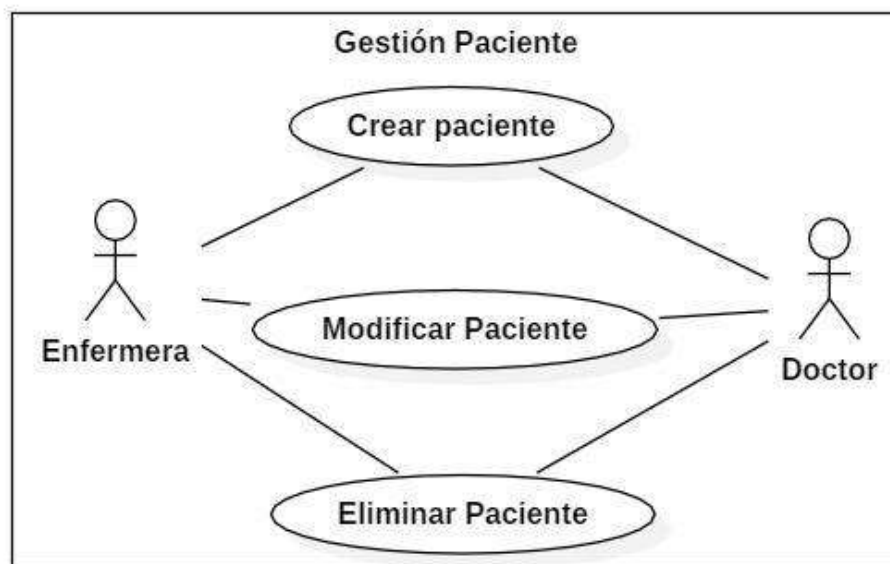


Figura 52: Diagrama de Caso de Uso: Gestión de Paciente

Fuente: Elaboración del Autor

Las acciones a realizarse en el proceso de gestión de paciente es la de un CRUD Crear, eliminar, modificar y listar. Estas funciones las puede realizar el usuario de tipo enfermera y doctor.

Gestión Historia Clínica

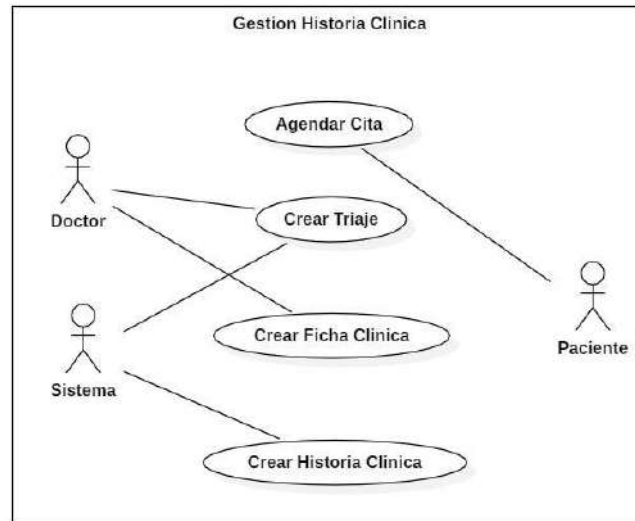


Figura 53: Diagrama de Caso de Uso: Gestión de Historia Clínica

Fuente: Elaboración del Autor

Para realizarse la gestión de historia clínica previamente, debe el usuario paciente previamente agendar una cita, luego de ellos se realiza la actividad de triaje, posterior a eso el usuario de tipo doctor realiza la ficha clínica del paciente y el sistema se encargará de almacenar esa información y guardarla como historia clínica.

Gestión Turnos

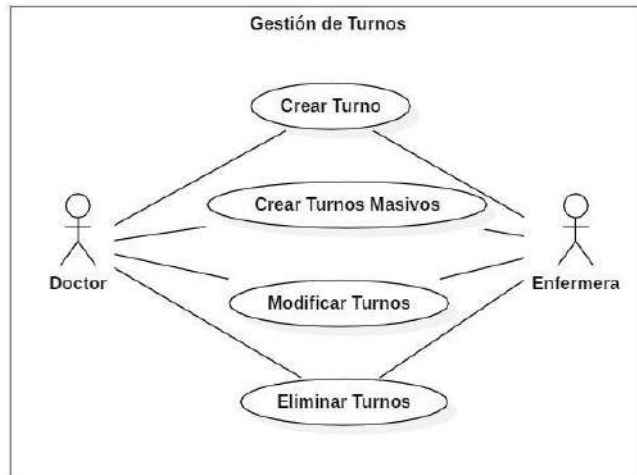


Figura 54: Diagrama de Caso de Uso: Gestión de Turnos

Fuente: Elaboración del Autor

En el proceso de Gestión de Turnos los actores son los usuarios con rol de doctor y el usuario con rol de enfermera, estos pueden realizar las actividades de crear turno, es decir un solo turno, crear turnos masivos que mediante un archivo .xls puede cargar una lista de turnos y además de ellos pueden ejecutar las opciones de modificar y eliminar.

Gestión Horas

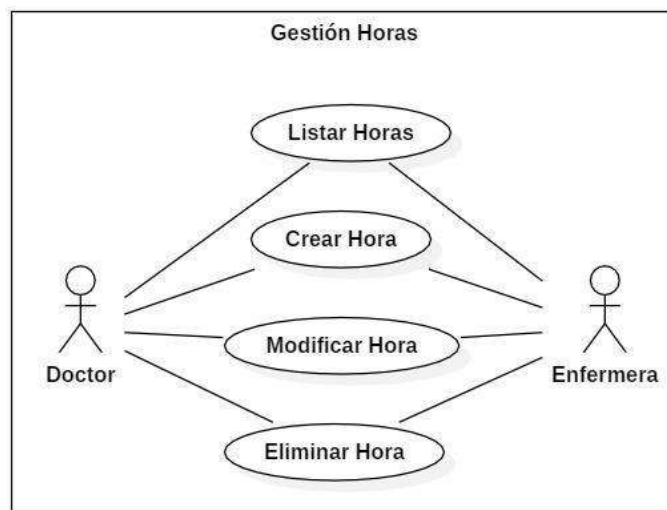


Figura 55: Diagrama de Caso de Uso: Gestión de Horas

Fuente: Elaboración del Autor

En la gestión de horas, se realizan las funciones de CRUD (Listar, Crear, Modificar, Eliminar) registros y los actores en esta actividad son el doctor y enfermera.

Gestión Roles

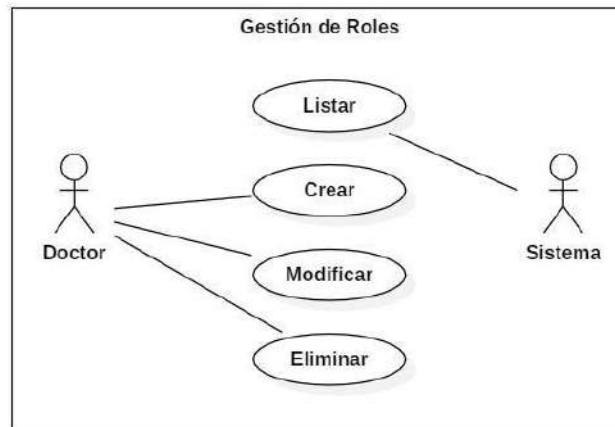


Figura 56: Diagrama de Caso de Uso: Gestión de Roles

Fuente: Elaboración del Autor

En lo correspondiente a la gestión de roles, se realiza las actividades de listar que lo realiza el sistema, crear, modificar y eliminar lo ejecuta el actor que sería el doctor.

Gestión Géneros

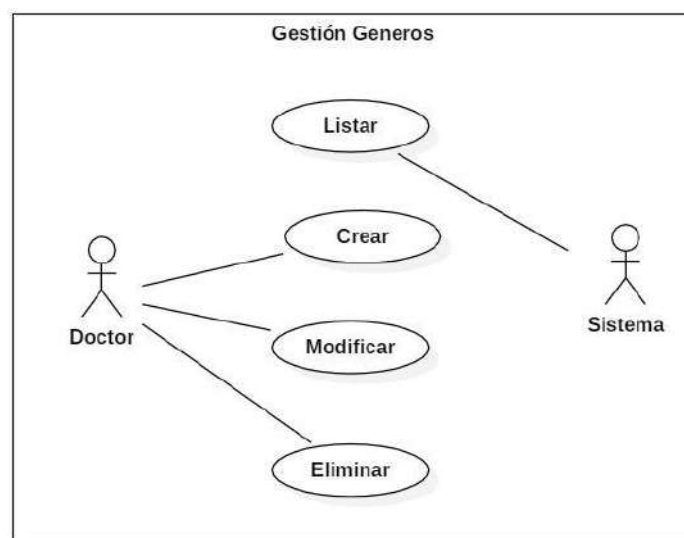


Figura 57: Diagrama de Caso de Uso: Gestión de Géneros

Fuente: Elaboración del Autor

Las actividades que se realizan en la gestión de géneros es listar que lo realiza el sistema, y las actividades de crear, modificar y eliminar registros lo realiza el usuario con rol de doctor.

Gestión Usuarios

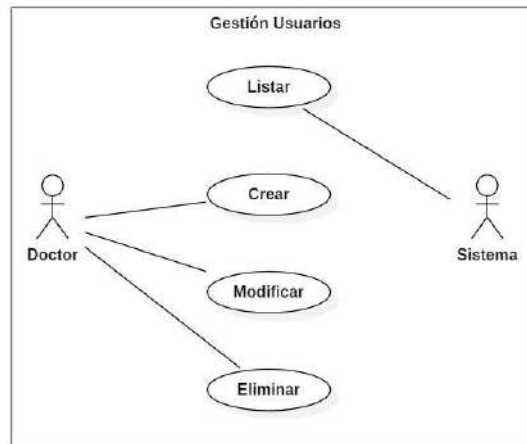


Figura 58: Diagrama de Caso de Uso: Gestión de Usuarios

Fuente: Elaboración del Autor

En la gestión de Gestión de usuarios participan como actores el usuario con rol de Doctor y el sistema, este es el que lista los registros, mientras que el doctor puede realizar las actividades de crear, modificar, y eliminar.

Gestión de Prueba de Laboratorios

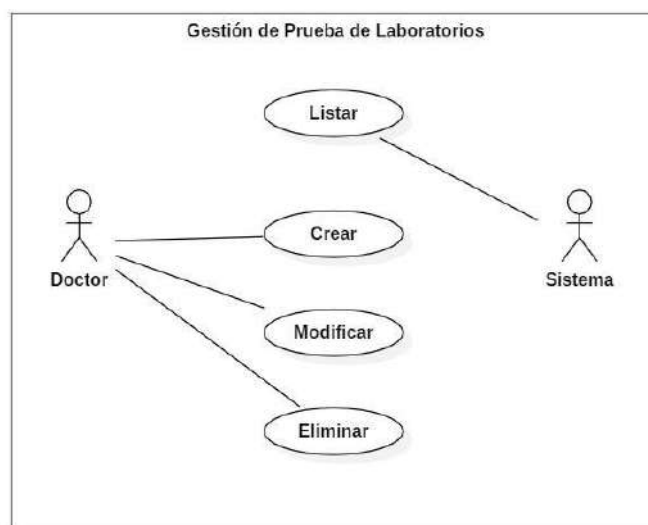


Figura 59: Diagrama de Caso de Uso: Prueba de Laboratorios

Fuente: Elaboración del Autor

En lo correspondiente a la gestión de pruebas de laboratorios, las actividades son de listar que le corresponde realizar al sistema, y crear, modificar y eliminar que lo realiza el doctor.

2.2.1.1.1. Diagrama de Actividades.

En esta sección se presentan los diagramas de actividades de registro de usuario, donde interviene el usuario y sistema realizando cada uno las actividades necesarias para el correcto registro de usuario.

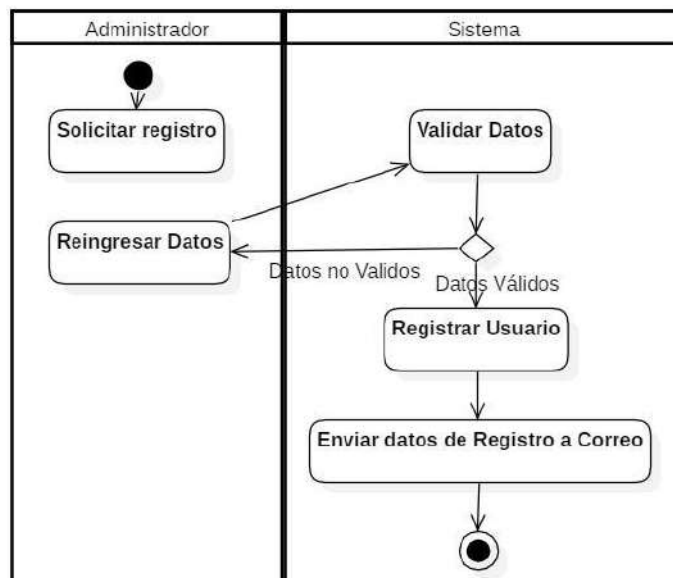


Figura 60: Diagrama de Actividades: Inicio de Sesión

Fuente: Elaboración del Autor

En el siguiente diagrama se presenta el flujo del sistema que realiza el sistema y el usuario para que inicie sesión.

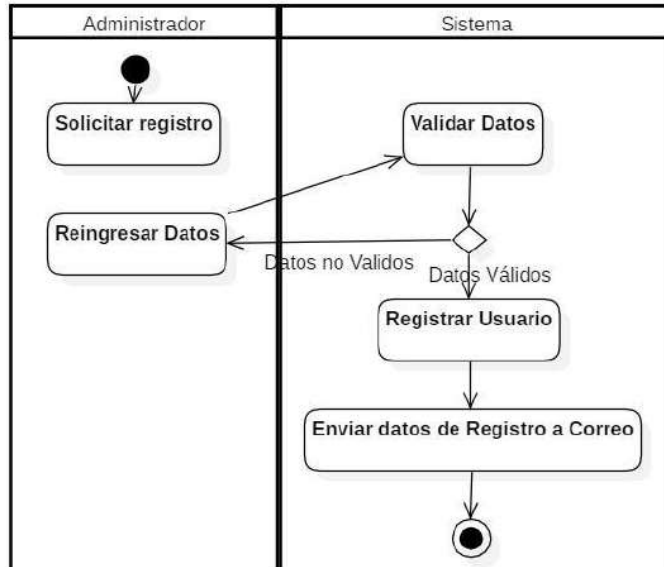


Figura 61: Diagrama de Actividades: Registro del Sistema

Fuente: Elaboración del Autor

Una vez realizado el proceso de registro, se envía un mensaje al correo dándole la bienvenida al nuevo usuario.

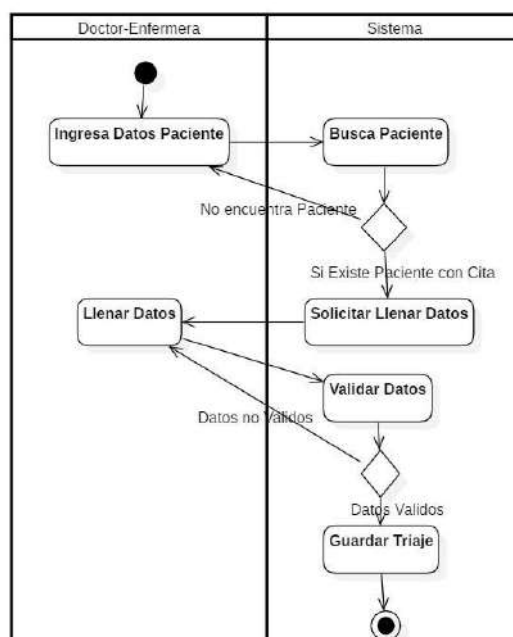


Figura 62: Diagrama de Actividades: Gestión de Triaje

Fuente: Elaboración del Autor

En la imagen anterior, se detalla las actividades que se realizan en el proceso de triaje, en ellas interactúan el doctor o enfermera y el sistema que es el encargado de validar datos.

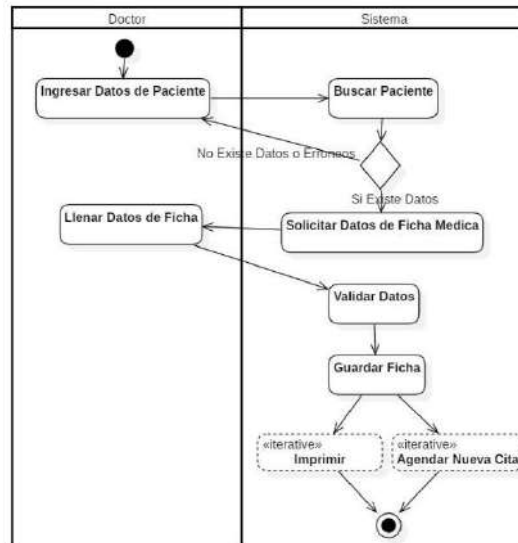


Figura 63: Diagrama de Actividades: Gestión Ficha Médica

Fuente: Elaboración del Autor

En el proceso de la Gestión de Ficha médica, intervienen en las actividades a desarrollar el doctor que es el que emite la ficha y llena según diagnóstico del paciente y por supuesto el sistema que es el encargado de realizar los procesos.

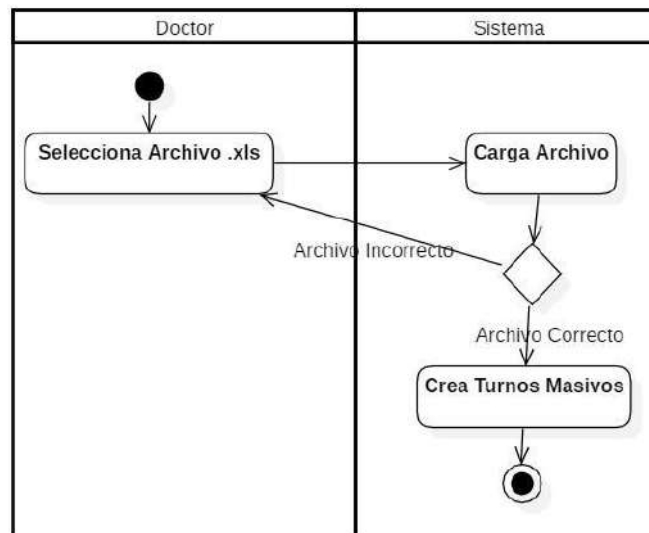


Figura 64: Diagrama de Actividades: Creación de Turnos Masivos

Fuente: Elaboración del Autor

En cuanto al proceso de creación de turnos se puede realizar de dos formas en este caso, los turnos masivos se realizan con la participación de él doctor y el sistema.

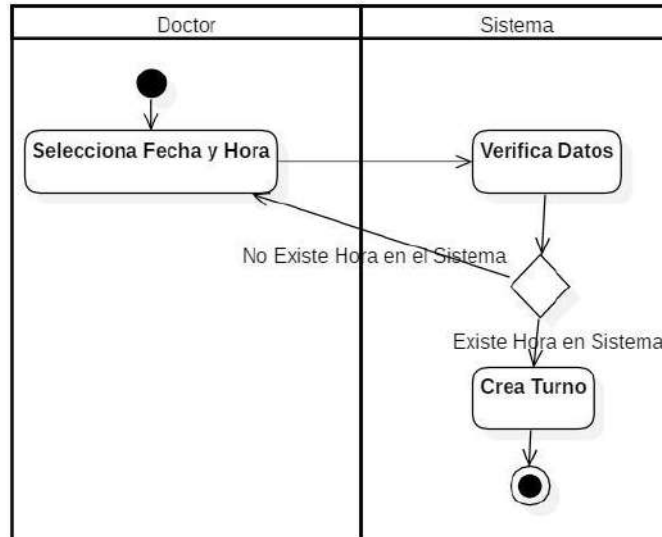


Figura 65: Diagrama de Actividades: Creación de Turnos Unitarios

Fuente: Elaboración del Autor

En el proceso de carga de turno unitario, intervienen en sus actividades de la misma forma el doctor encargado de gestionar la información, y el sistema que se encarga de validar y crear turnos.

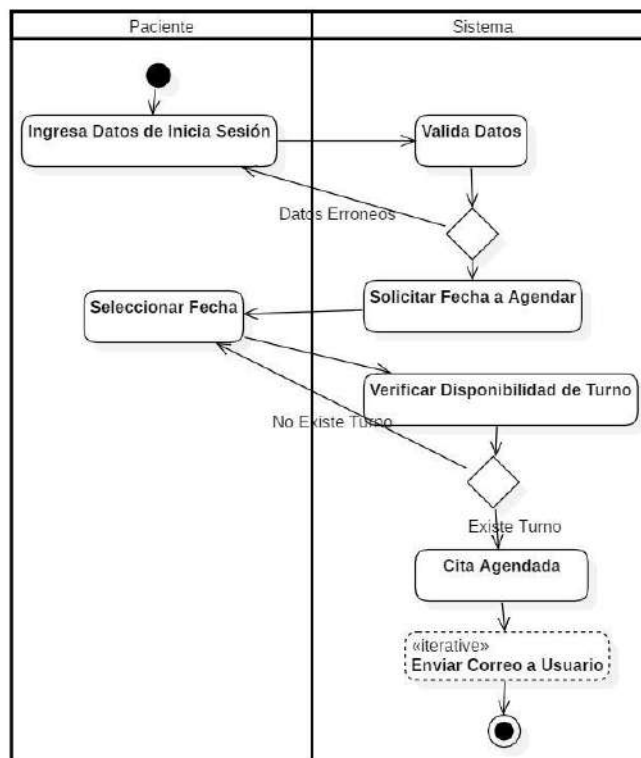


Figura 66: Diagrama de Actividad: Agendamiento de Cita

Fuente: Elaboración del Autor

El agendamiento de cita, se da con la intervención del paciente que es el ente principal en esta acción, debido a que debe seleccionar fecha y hora, una vez realizado este proceso participa el sistema validando la información y verificando disponibilidad del turno solicitado.

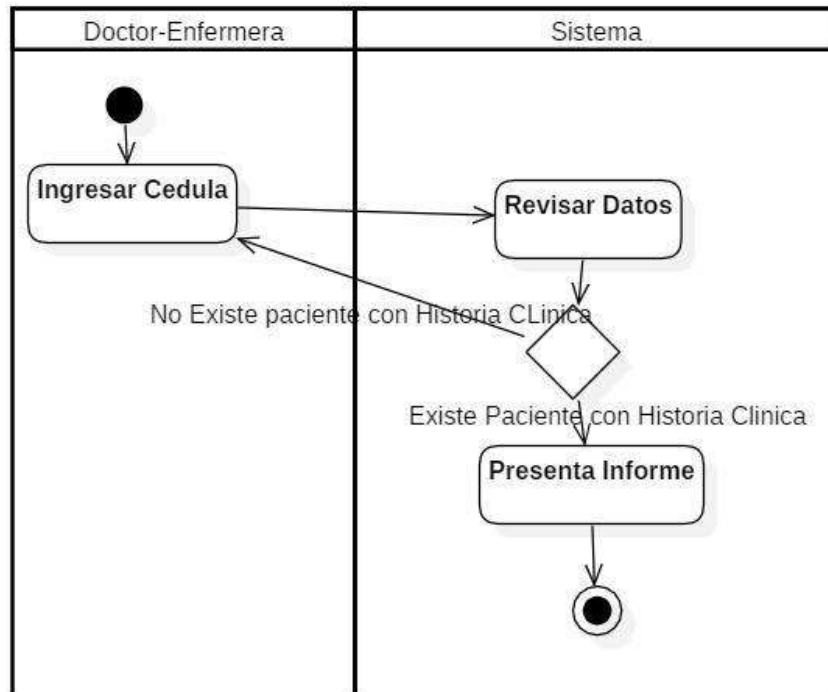


Figura 67: Diagrama de Actividad: Historia Clínica

Fuente: Elaboración del Autor

2.2.1.1.1. Diagrama de Secuencias.

Los diagramas de secuencia, describen el comportamiento dinámico que tiene el sistema en función de los mensajes que proyecta hacia los usuarios, en la siguiente figura expresa la interacción entre el sistema y el usuario final.

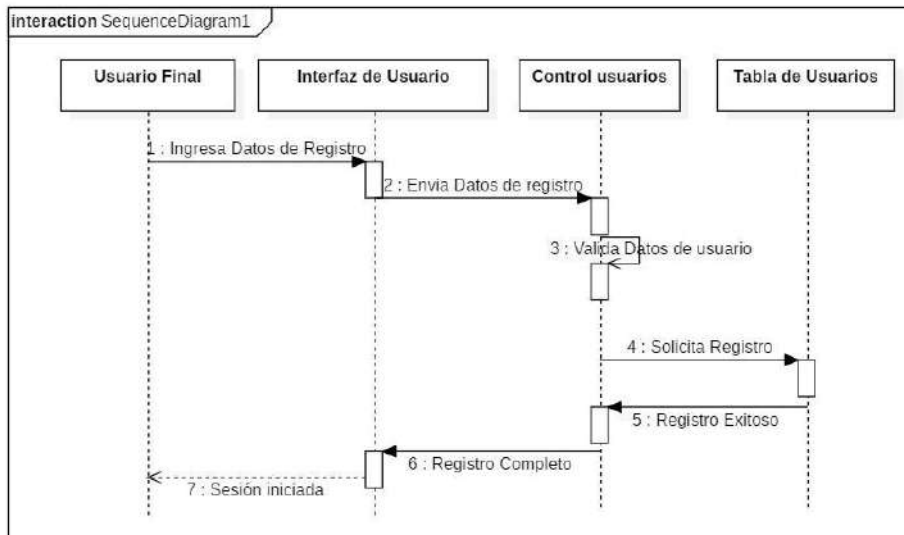


Figura 68: Diagrama de Secuencia: Registro Usuario

Fuente: Elaboración del Autor

En la imagen anterior se realiza el recuento de la secuencia del proceso que se da para la gestión de usuario dentro del sistema.

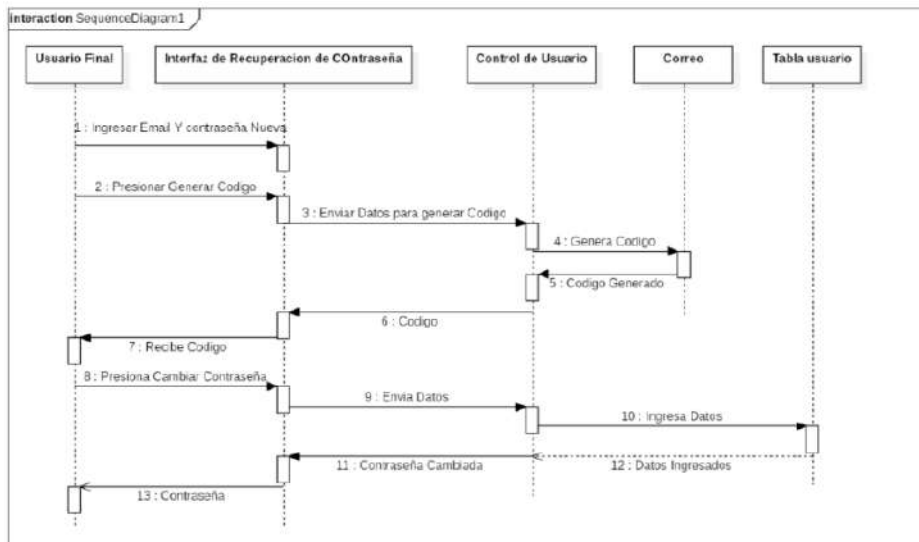


Figura 69: Diagrama de Secuencia: Recuperación de Password

Fuente: Elaboración del Autor

El sistema también permite la recuperación de password, mediante el proceso detallado en la figura anterior se evidencia cómo el sistema y el usuario interactúan en función de realizar esa opción de recuperación de password.

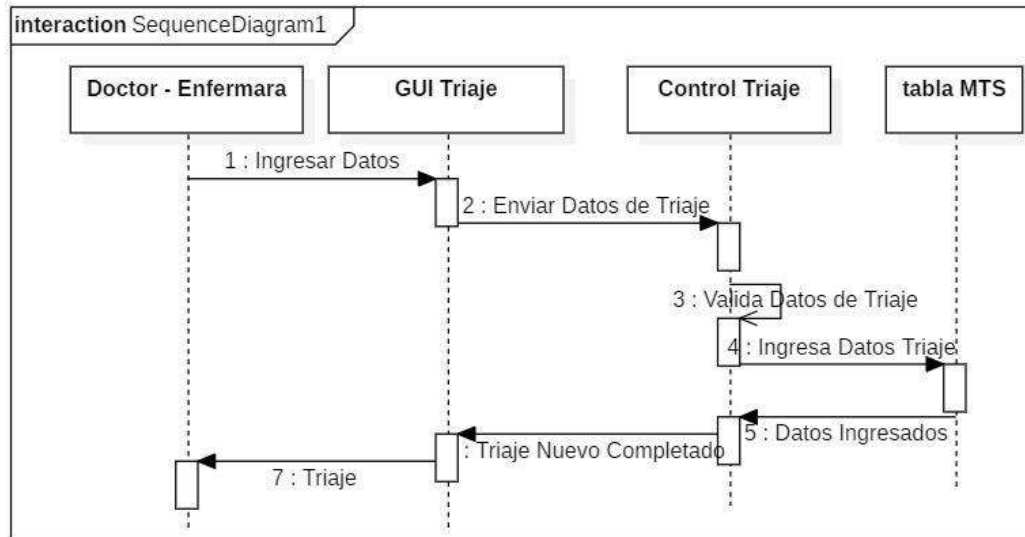


Figura 70: Diagrama de Secuencia: Triage

Fuente: Elaboración del Autor

El sistema facilita la opción de gestionar un triaje previa a la consulta, en este proceso se realiza un grupo de actividades que deben cumplir con una secuencia como se detalla en la figura anterior.

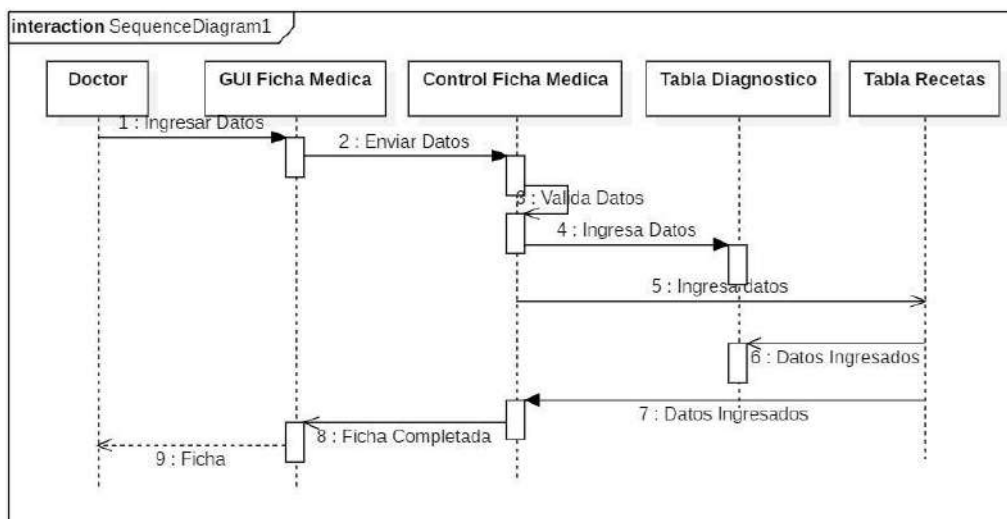


Figura 71: Diagrama de Secuencia: Ficha Médica

Fuente: Elaboración del Autor

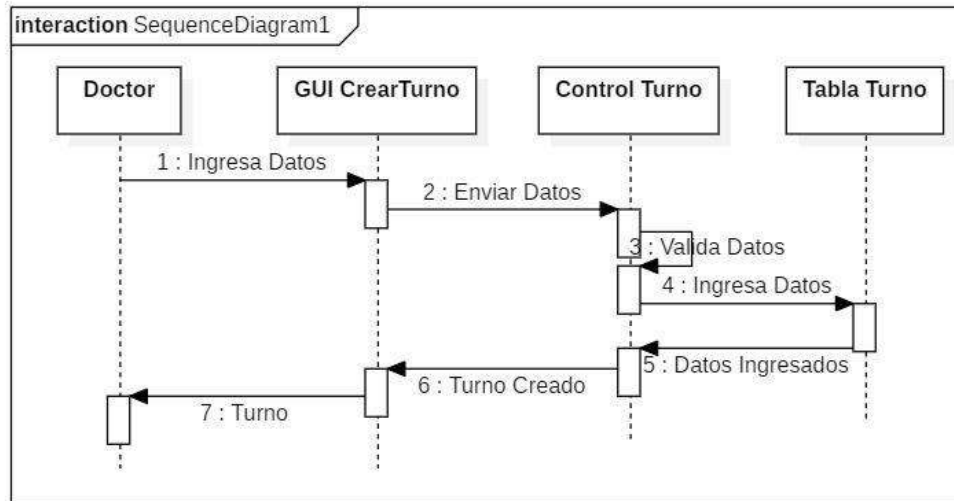


Figura 72: Diagrama de Secuencia: Crear Turno

Fuente: Elaboración del Autor

El sistema permite al doctor o enfermera, poder crear un turno, para ello es indispensable que se cumpla con un grupo de pasos secuenciales que se evidencian en la figura anterior.

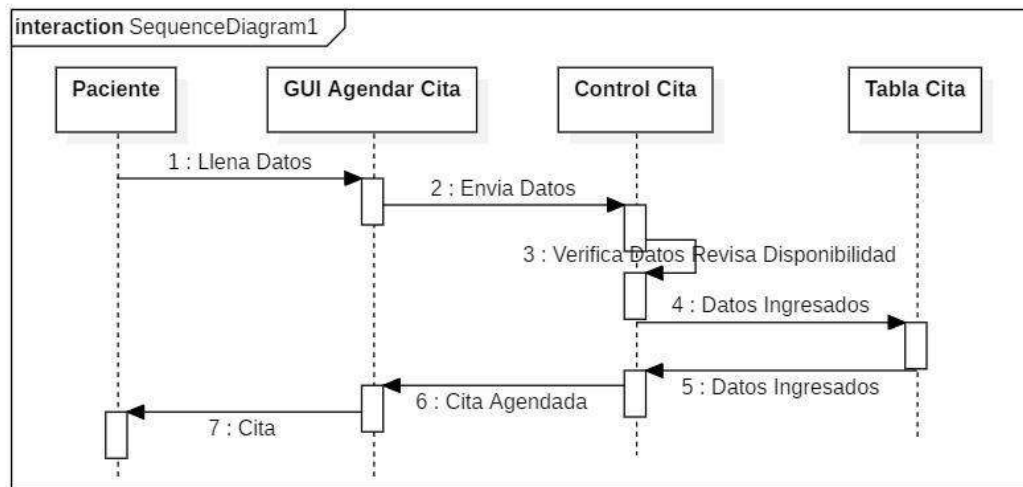


Figura 73: Diagrama de Secuencia: Agendar Cita

Fuente: Elaboración del Autor

Para la gestión de agendar una cita el sistema debe interactuar con el paciente, pero en este proceso se da una secuencia de pasos para realizar el objetivo de agendar una cita.

2.3. Ejecución y Ensamblaje de Prototipo.

2.3.1. Formularios de autenticación.

A continuación, se detalla el desarrollo del prototipo web por medio de interfaces.

Como interfaz inicial disponemos de un diseño tipo one page que cuenta con información del consultorio médico.

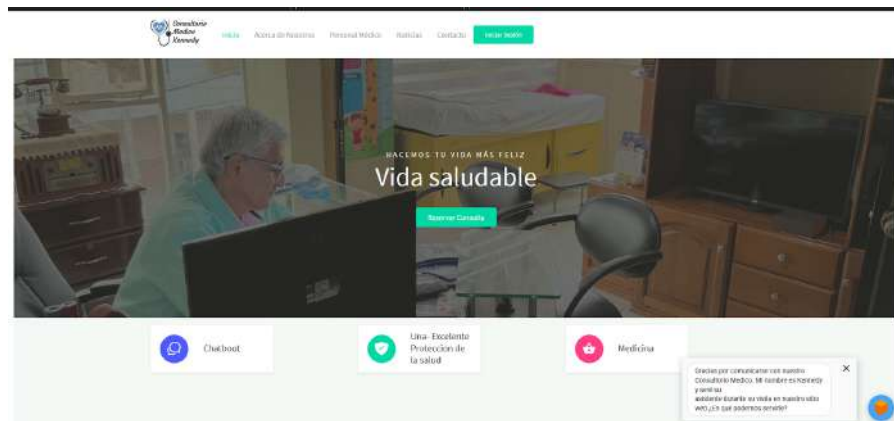


Figura 74: Interfaz Inicial del Sistema
Fuente: Elaboración del Autor

Inicio de Sesión.

Mediante la interfaz del Loguin el usuario tiene la posibilidad de iniciar sesión, utilizando el correo y contraseña que utilizó al momento de registrarse.

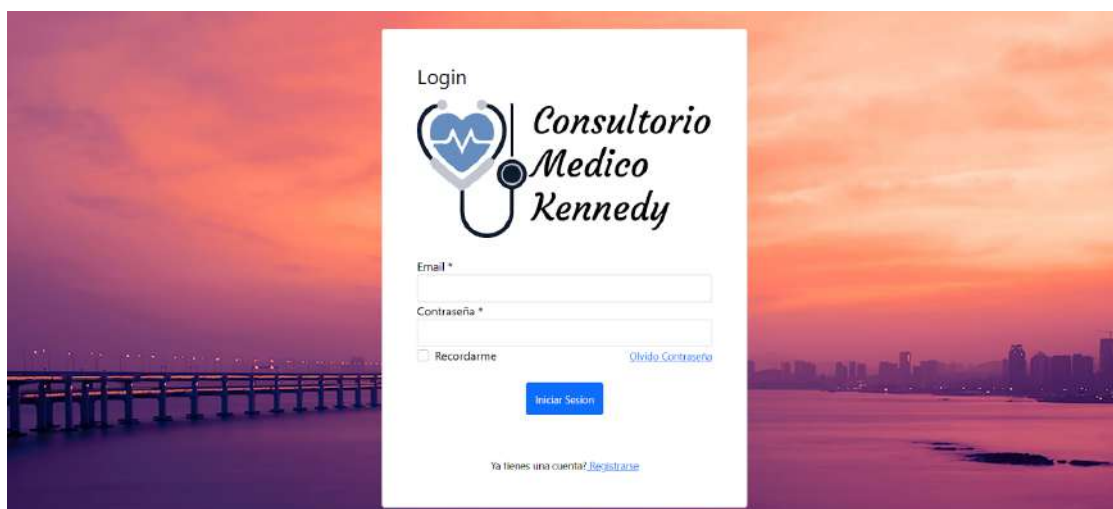


Figura 75: Interfaz de Inicio de Sesión
Fuente: Elaboración del Autor

Registro de Usuarios.

En el formulario de registros, el usuario debe ingresar todos los datos solicitados por el sistema, cabe mencionar que una vez que el usuario se registre, llegará un mensaje al correo que ingresó.

Registro

C.I.

Nombre de Paciente

Apellidos de Paciente

Fecha de Nacimiento

Lugar de Nacimiento

Email

Telefono

Contraseña

Recordarme

Figura 76: Interfaz de Registro
Fuente: Elaboración del Autor

A continuación, se presenta imagen de correo enviado.

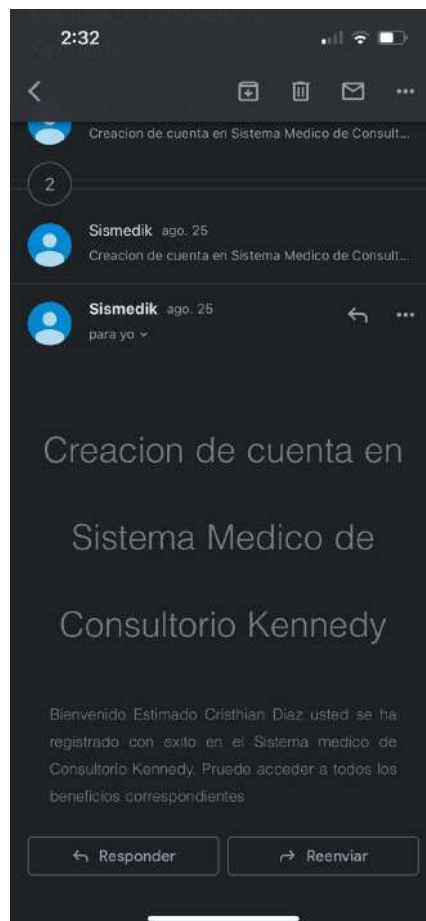


Figura 77: Correo Enviado por Registro
Fuente: Elaboración del Autor

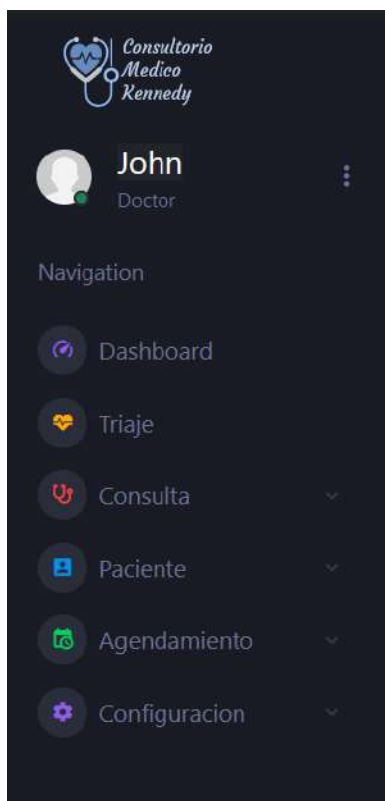
Recuperación de contraseña.

El sistema gestiona también la facilidad de que un usuario en caso de pérdida de contraseña pueda recuperar su cuenta, esto lo hace mediante un correo electrónico donde le brinda un enlace y un código.



Figura 78: Interfaz de Recuperación de Contraseña

Fuente: Elaboración del Autor



1. Dashboard, los usuarios que tienen acceso al panel de administración pueden visualizar un dashboard de los datos principales del sistema.

2. Gestión de Triaje, en base al proceso de cita médica el usuario tipo doctor o enfermera debe registrar un triaje previo a la cita médica.

3. En consultas, contiene un submenú con las opciones de fichas clínicas, exámenes de laboratorio e informe.

4. En la opción de Paciente, cuenta con las funciones de visualizar los pacientes e historia clínica.

5. En agendamiento, se despliega otras opciones como turnos, horas, informe y verificación de estado de citas.

6. En configuración, cuenta con algunas opciones como roles, géneros, usuarios, prueba de laboratorio y estado civil.

Interfaces de Usuario Doctor

Algunas interfaces que tiene el doctor cuentan también el usuario enfermera, en este caso se describirá todas las acciones que puede gestionar un doctor.

Se presenta una interfaz con el dashboard que gestiona el sistema, es un conteo de los pacientes que tiene, de las citas atendidas y citas pendientes.

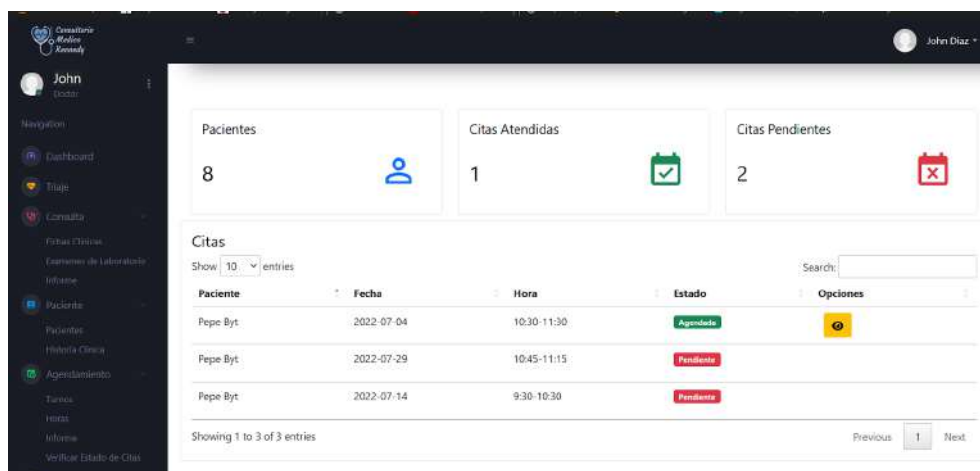


Figura 80: Interfaz de Dashboard
Fuente: Elaboración del Autor

El usuario tipo doctor y enfermera tienen la opción de crear un triaje, este se habilita ingresando la cedula del paciente que tiene cita en el día en curso, sino existe cita no saldrá para llenar los datos.

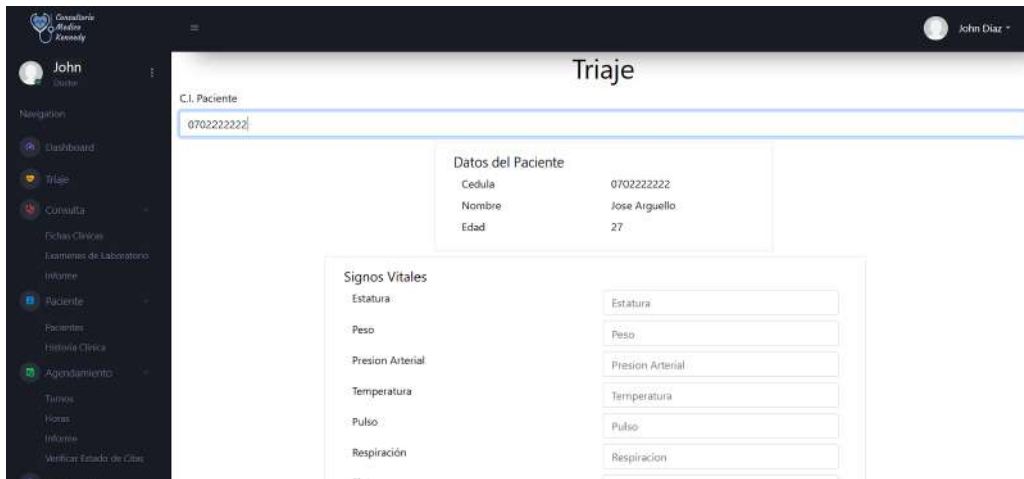


Figura 81: Interfaz de Triage
Fuente: Elaboración del Autor

De la misma forma una vez realizado el triaje el sistema permite que el doctor proceda a llenar la ficha del paciente, donde debe llenar los síntomas, dar el diagnóstico, enviar los medicamentos y la prescripción médica, en la parte derecha se puede evidenciar en otros, si el paciente tiene algún caso especial o alguna alergia, se puede ver los últimos tres registros de presión arterial, peso y las últimas citas médicas.

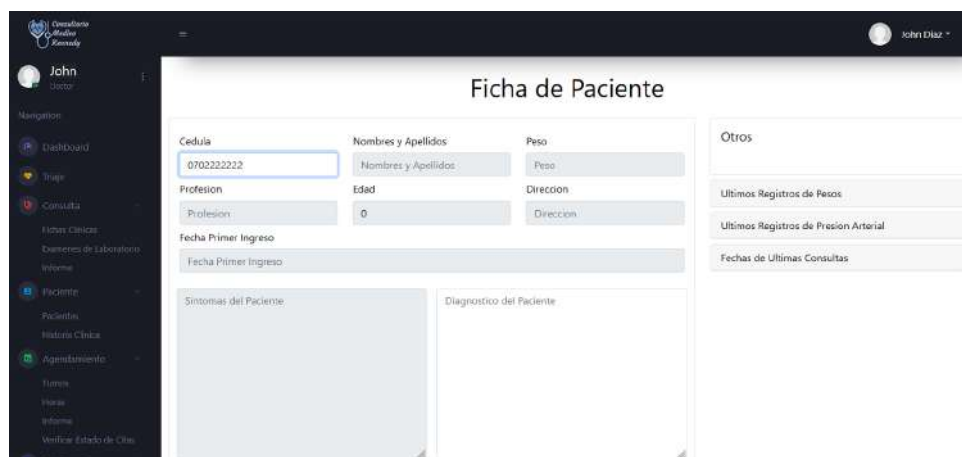


Figura 82: Interfaz de Ficha de Paciente
Fuente: Elaboración del Autor

Si el usuario doctor realizó la ficha médica puede revisar los datos de exámenes del paciente y escribir los hallazgos.

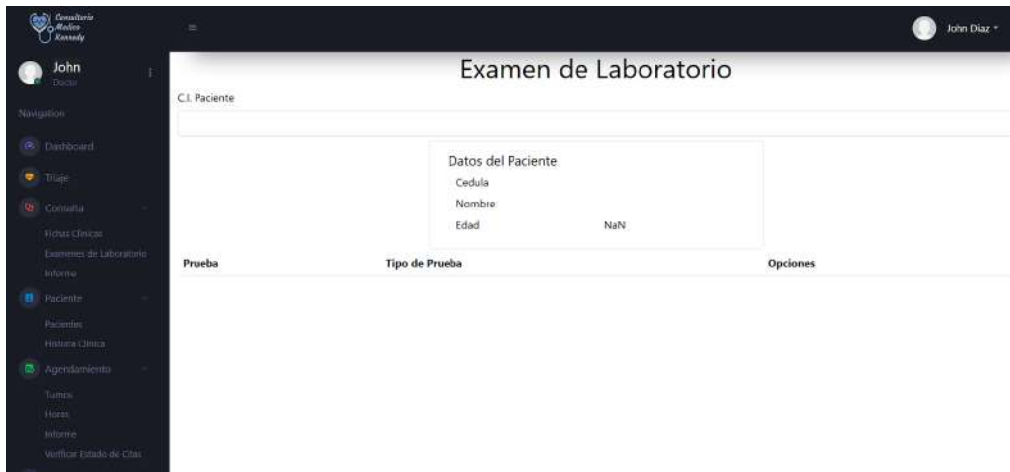


Figura 83: Interfaz Examen de Laboratorio
Fuente: Elaboración del Autor

El doctor también puede visualizar un informe de los pacientes, en el cual consta de tres opciones que son receta, diagnósticos y los exámenes valorados de los pacientes.

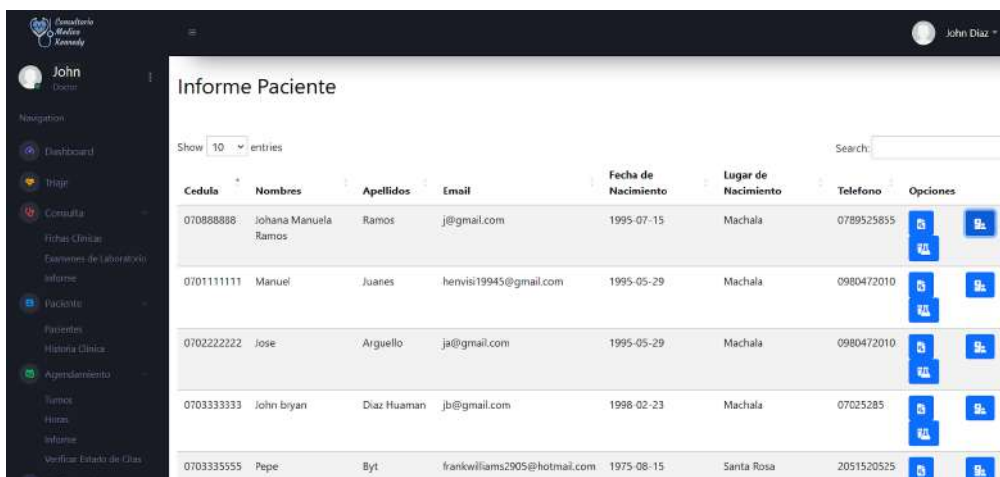


Figura 84: Interfaz de Informe de Paciente
Fuente: Elaboración del Autor

El sistema también permite al doctor gestionar pacientes, es decir modificar información o crear un nuevo registro y también eliminar.

Cedula	Nombres	Apellidos	Email	Fecha de Nacimiento	Telefono	Opciones
070888888	Johana Manuela Ramos	Ramos	j@gmail.com	1995-07-15	0789525855	[Edit] [Delete]
070111111	Manuel	Juanes	henvisi19945@gmail.com	1995-05-29	0980472010	[Edit] [Delete]
070222222	Jose	Arguello	ja@gmail.com	1995-05-29	0980472010	[Edit] [Delete]
070333333	John bryan	Diaz Huaman	jb@gmail.com	1998-02-23	07025285	[Edit] [Delete]
070333555	Pape	Byt	frankwilliams2905@hotmail.com	1975-08-15	2051520525	[Edit] [Delete]
0704265099	Juan Alberto	Perez Gonzales	henvisi			[Edit] [Delete]
0704265099	Juanito	Roa	henvisi1994@gmail.com		2051520525	[Edit] [Delete]
0706666666	Belen	Espinoza	b@gmail.com	1998-05-14	074551515	[Edit] [Delete]

Figura 85: Interfaz de Paciente
Fuente: Elaboración del Autor

El sistema también permite al médico gestionar la historia clínica del paciente, donde se aprecia mediante un documento pdf que lo puede imprimir y dar al paciente para que vea su historia clínica, en el se puede apreciar datos de cada cita que se haya realizado en consultorio.

Fecha	Peso	Presion Arterial	Sintomas	Diagnostico	Otros
[Empty table rows]					

Figura 86: Interfaz de Historia Clínica
Fuente: Elaboración del Autor

En cuanto a la gestión de turnos el sistema presenta la acción de crear un solo nuevo turno o cargar un archivo xls con la lista de turnos creados, esta es la opción de turnos masivos, además puede eliminar y registrar.

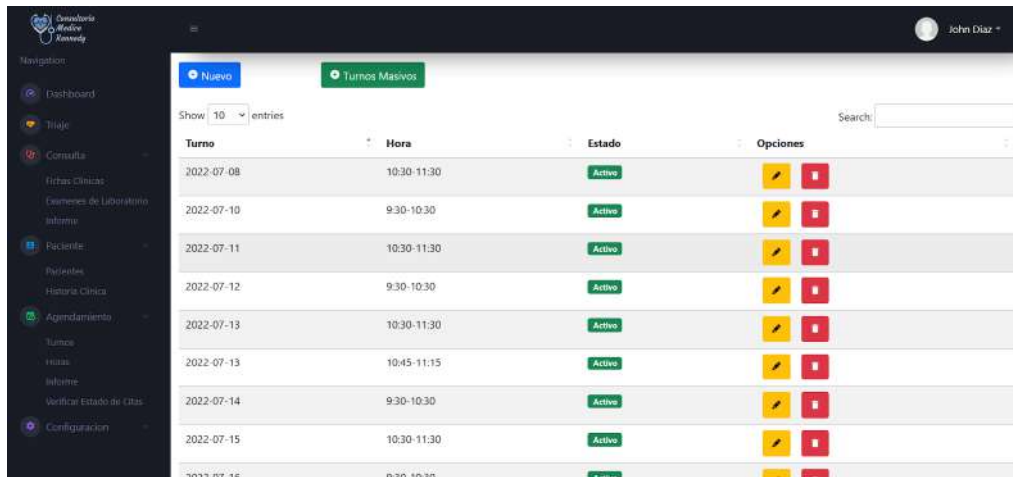


Figura 87: Interfaz de Gestión de Turnos
Fuente: Elaboración del Autor

En cuanto a horas el sistema permite crear nuevos rangos de horas para la utilización de creación de turnos.

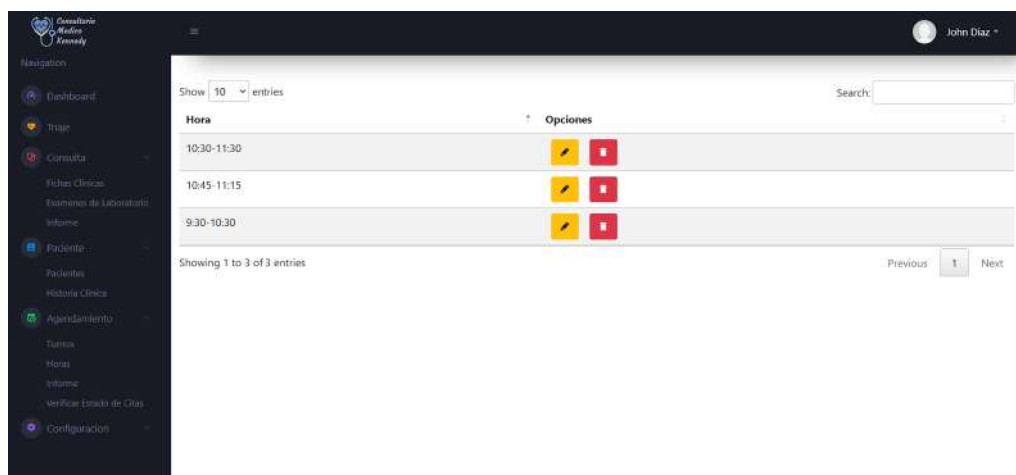


Figura 88: Interfaz de Gestión Horas
Fuente: Elaboración del Autor

El sistema permite descargar informes de pacientes atendidos por fecha.

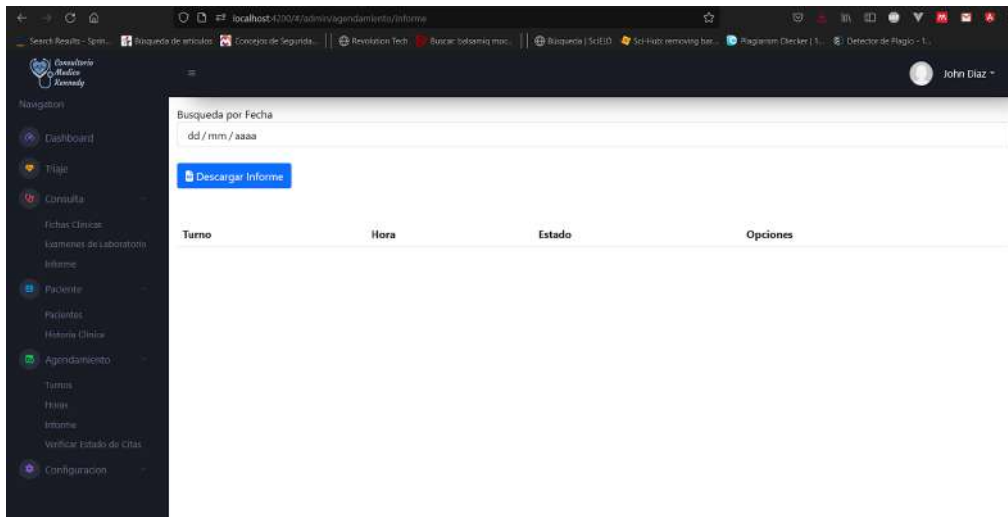


Figura 89: Interfaz de Informe de Pacientes
Fuente: Elaboración del Autor

En cuanto a la verificación de citas el usuario de tipo doctor debe verificar la cita, es decir validarla siempre y cuando el usuario haya cargado el comprobante de pago.

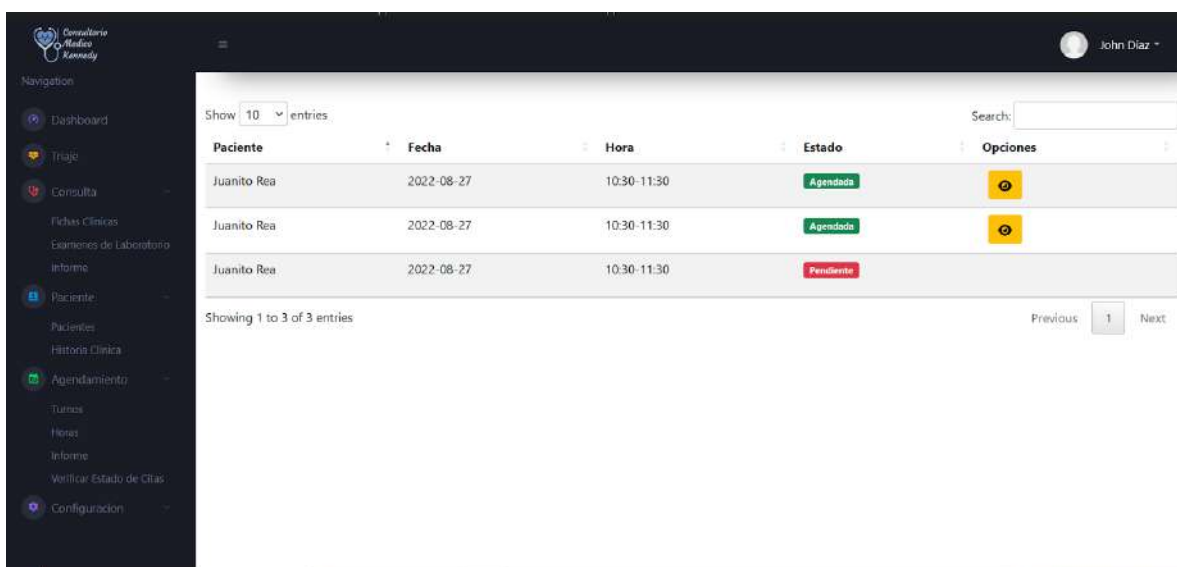
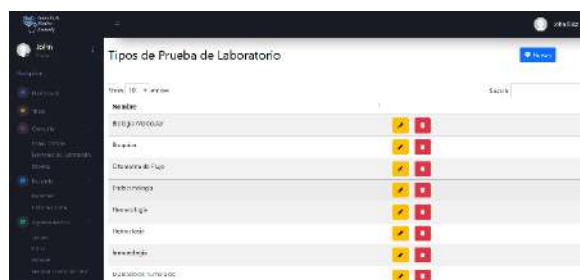
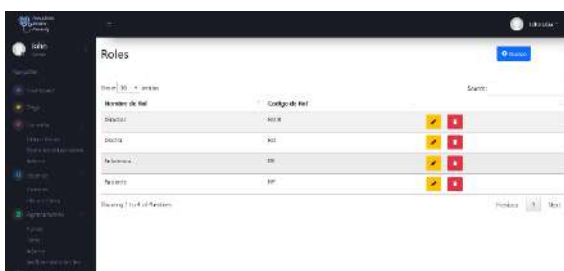


Figura 90: Interfaz de verificación de cita
Fuente: Elaboración del Autor



El sistema permite la gestión de roles y tipos de prueba de laboratorios, este último con la finalidad de registrar un nuevo tipo de laboratorio debido a que cada vez se realizan más hallazgos en la medicina.

Páginas Públicas

En cuanto a la página acerca de nosotros, cuenta con información del consultorio médico Kennedy como los valores que se practica además de información del doctor y si pasa el cursor por la imagen del mismo permite que se pueda poner en contacto por medio de llamada o WhatsApp.



Figura 91: Interfaz de Acerca de Nosotros
Fuente: Elaboración del Autor

En cuanto a personal médico permite al usuario conocer el personal que labora dentro del consultorio médico.

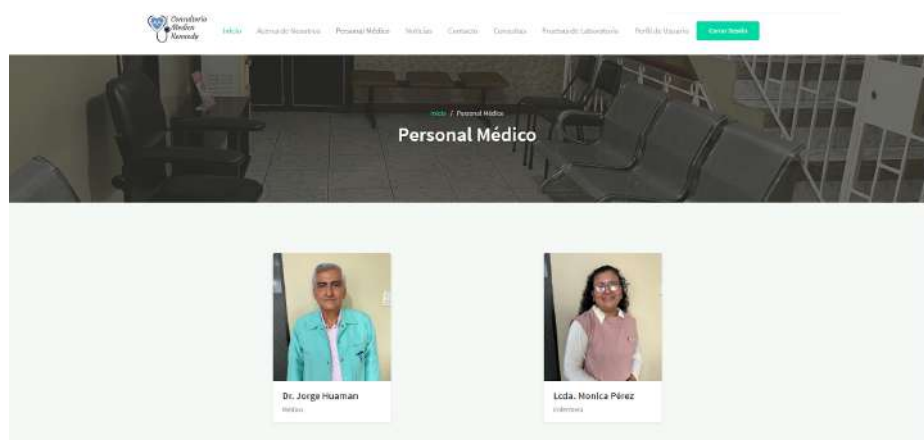


Figura 92: Interfaz de Personal Médico
Fuente: Elaboración del Autor

El sistema brinda información médica en la página de noticias que contiene enlaces de interés para un mejor cuidado de la salud.

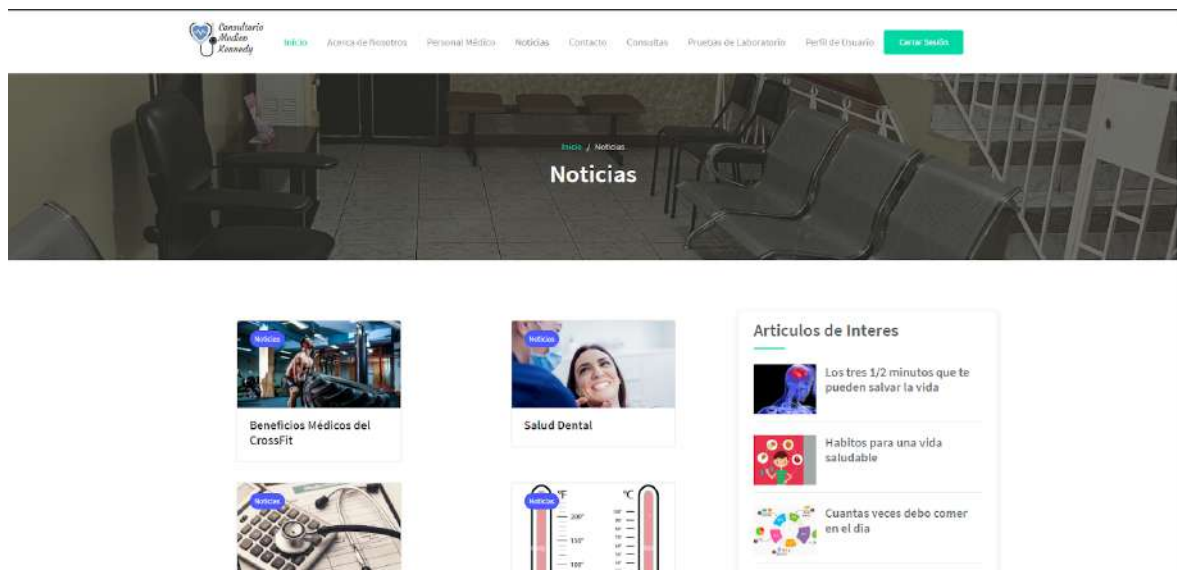


Figura 93: Interfaz de Noticias
Fuente: Elaboración del Autor

3. CAPÍTULO III: EVALUACIÓN DEL PROTOTIPO.

3.1. Plan de Evaluación.

La aplicación de la evaluación del sistema es de gran aporte en el ámbito de desarrollo y de pruebas del proyecto finalizado y el transcurso del mismo; de esta forma se mitiga la cantidad de errores o fallos de las acciones que realiza el sistema, permitiendo obtener un sistema de calidad y eficiente. La metodología SWIRL permite realizar un plan de evaluación en 3 pruebas a considerar, evaluación de calidad, evaluación de tendencia y evaluación con la participación de herramientas SEO.

3.1.1. Evaluación de Calidad.

Para la correcta evaluación de la calidad se ha optado por aplicar la normativa ISO/IEC 9126, en la metodología SWIRL se hace mención que empleando como una herramienta un cuestionario detallando indicadores establecidos en la métrica. Con el objetivo de una mejor comprensión se elabora una tabla con escala de Likert que tiene valores de 1 (Excelente), hasta 5 (Pésimo) como se detallan en la siguiente tabla.

Tabla 29: Escala de Likert en Evaluación de Calidad

Escala de Likert	
Valoración	Interpretación de Resultados
1	Pésimo
2	Malo
3	Regular
4	Bueno
5	Excelente

Fuente: Elaboración del Autor

El objetivo de realizar la evaluación de calidad es conocer si el sistema cumple con las necesidades del cliente, y si brinda el cumplimiento con los criterios de funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y operatividad.

3.1.2. Evaluación con Herramientas SEO.

Para realizar la evaluación de características SEO, la metodología SWIRL, brinda como propuesta la implementación de herramientas tecnológicas que permita gestionar el contenido semántico del sitio con la finalidad de proporcionar un análisis que proyecte resultados del posicionamiento del sitio frente a búsquedas del usuario en navegadores. Además, las herramientas a utilizar también brindan resultados en cuanto a funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y operatividad.

3.2. Resultados de Evaluación.

3.2.1. Evaluación de Calidad.

Aplicando la norma ISO/IEC 9126, permite obtener resultados de calidad del sistema en base a criterios de funcionalidad, fiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad. A continuación, se detalla la siguiente tabla que posee el análisis de calidad usando escala de Likert.

Tabla 30: Evaluación de Calidad Interna - Externa

Características	Subcaracterísticas	Criterio	Valoración
Funcionalidad	Exactitud	¿El sistema brinda los resultados esperados?	5
	Interoperabilidad	¿El sistema tiene la capacidad de interactuar con otro sistema?	5
	Adecuación	¿El sistema cumple con la capacidad de realizar las tareas establecidas en los requerimientos?	4
	Conformidad	¿El sistema cumple con estándares de funcionalidad?	5
	Seguridad	¿El sistema impide el acceso no autorizado?	5
Total			24/25
Eficiencia	Consumo de Recursos	¿Los correos que emite el sistema se dan de manera óptima y en buen tiempo?	4
	Comportamiento en el Tiempo	¿El tiempo de respuesta del sistema es adecuado?	4
Total			8/10
	Madurez	¿El sistema tiene la capacidad de mitigar	4

Fiabilidad		los fallos como resultado de errores?	
	Recuperabilidad	¿El sistema cuenta con la función de recuperar la información en caso de fallos?	3
Total			7/10
Usabilidad	Comprensión	¿Su entendimiento y comprensión es fácil en cuanto a tareas y funciones que realiza el sistema?	5
	Operatividad	¿La intervención del usuario se da pocas veces para realizar las funciones del sistema?	4
	Atractividad	¿Las interfaces graficas del sistema se adaptan y son amigables para el usuario?	3
	Aprendizaje	¿El usuario aprende de una forma rápida y fácil el uso del sistema?	5
Total			17/20
Portabilidad	Conformidad	¿El sistema se encuentra desarrollado con algún lenguaje multiplataforma?	4
	Facilidad de Instalación	¿La instalación del sistema es fácil?	4
	Remplazabilidad	¿Es fácil reemplazar el sistema por otras funcionalidades similares?	4
	Adaptabilidad	¿El software es fácil de adaptar a otros entornos de trabajo?	5
	Coexistencia	¿El software cuenta con la capacidad de interactuar con otros sistemas?	5
Total			22/25

Mantenibilidad	Capacidad de Modificación	¿El software se puede modificar ante cualquier necesidad?	3
	Estabilidad	¿El software mantiene su funcionalidad pese a cambios?	4
	Capacidad de Análisis	¿Es fácil identificar fallas para ser corregidas o modificadas?	5
	Facilidad de Prueba	¿Es fácil realizar pruebas en el sistema?	5
Total			17/20

Fuente: Elaboración del Autor

Una vez realizada la evaluación mediante la aplicación de características proporcionadas por la normativa ISO/IEC 9126, de esta manera se evidencia que los totales obtenidos hacen del sistema un producto que cumple con la aceptación en todos los criterios.

La aplicación y combinación de tendencias y herramientas tecnológicas elegidas para la implementación del sistema, aportaron significativamente en los criterios de calidad evaluados, destacando las características de mantenibilidad, portabilidad y usabilidad que posee el sistema.

3.2.2. Evaluación mediante Herramientas SEO

3.2.2.1. GTMETRIX.

Esta herramienta facilita la evaluación de la aplicación en base al rendimiento y tiempo de respuesta. La aplicación arrojó como resultados un 96% en cuanto a performance y un 80% en estructura, estos resultados representan que es bueno el performance del sistema más aún con el tiempo de respuesta que es de 1.3 s.

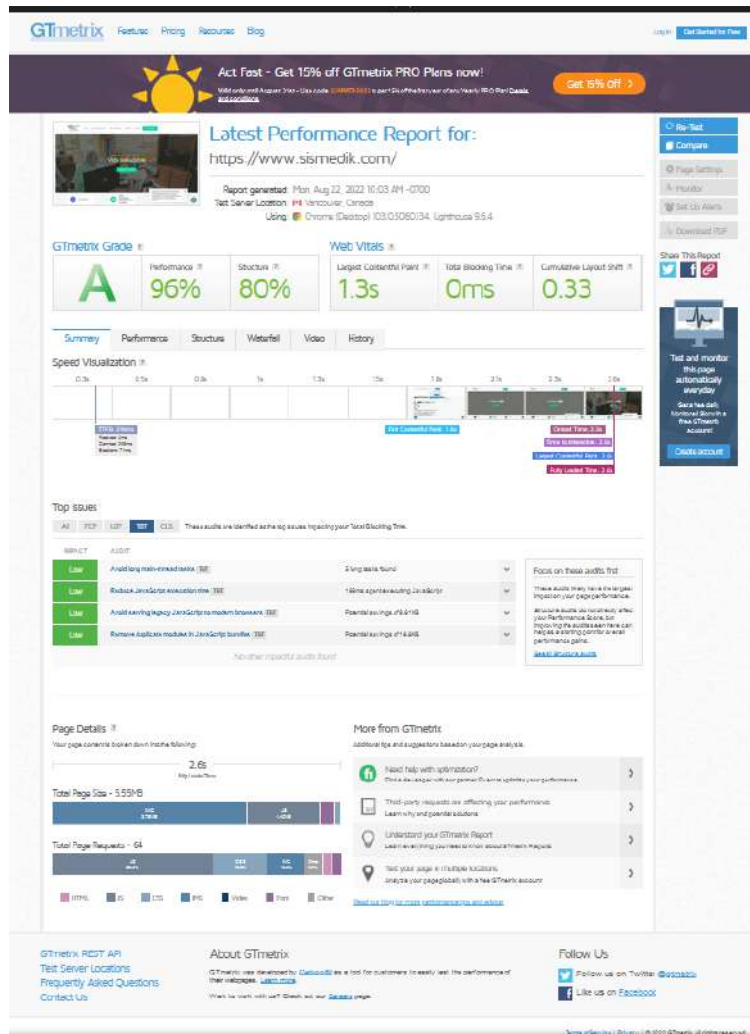


Figura 94: Evaluación - Gtmetrix

Fuente: Elaboración del Autor

3.2.2.2. **Functional Accessibility Evaluator.**

La siguiente herramienta evalúa el sitio en base a requerimientos de pautas de accesibilidad del contenido web 2.0 y se han obtenido los siguientes resultados.

Page 1 Summary

[E-mail Report](#) | [CSV](#)Page 1: [Sismedik](#)

	Violations	Warnings	Manual Checks	Passed
Number of Rules	3	1	14	1

Rule Category: [WCAG Guidelines](#) [Rule Scope](#) Default Sort

Rule Group	Number of Rules				Implementation Level	
	V	W	MC	P	Score	Status
Landmarks	1	-	2	-	0	R
Headings	-	1	-	-	0	R
Styles/Content	-	-	4	1	20	NI-R
Images	-	-	-	-	-	na
Links	-	-	-	-	-	na
Tables	-	-	-	-	-	na
Forms	-	-	-	-	-	na
Widgets/Scripts	-	-	1	-	0	R
Audio/Video	-	-	1	-	0	R
Keyboard	-	-	1	-	0	R
Timing	-	-	3	-	0	R
Site Navigation	2	-	2	-	0	R
All Rule Groups	3	1	14	1	5	NI-R

Figura 95: Evaluación - Herramienta FAE

Fuente: Elaboración del Autor

3.2.2.3. Website Grader.

Esta herramienta da acceso a evaluar el rendimiento SEO, usabilidad y seguridad del sistema, en cuanto a las a la adaptabilidad existen mejoras que hacer debido a que el complemento del chatbot no se ajustó al sistema.

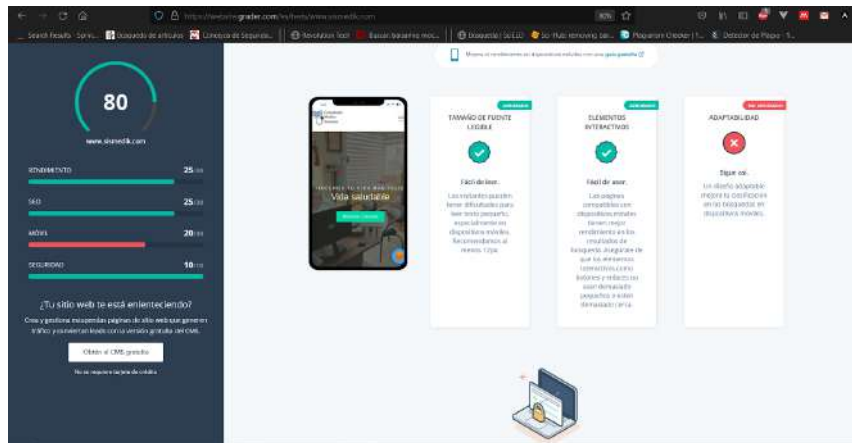


Figura 96: Evaluación - Herramienta Website Grader

Fuente: Elaboración del Autor

3.2.2.4. SEO Site Checkup.

La herramienta SEO Site Checkup permite evaluar al sistema en cuanto a estructura semántica, buscando las mejores recomendaciones a nivel de mejoramiento de SEO, calificación es 84/100 que quiere decir que es aceptable.

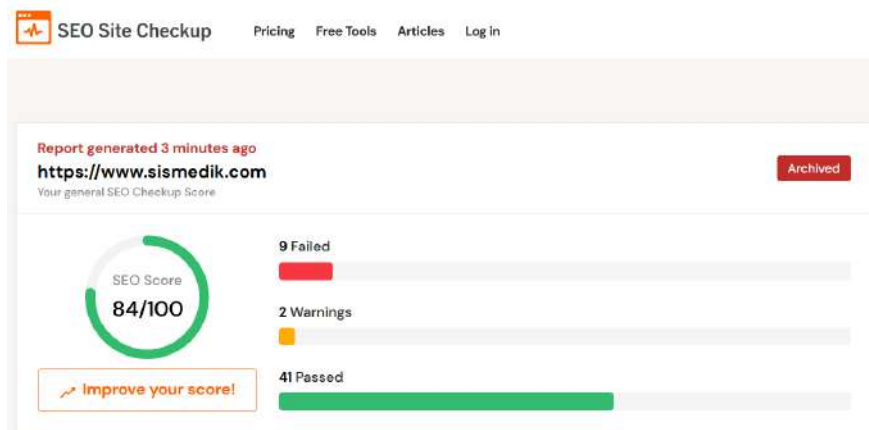


Figura 97: Evaluación - Herramienta SEO Site Checkup

Fuente: Elaboración del Autor

3.3. CONCLUSIONES

Una vez concluido el desarrollo e implementación del sistema web para gestión de agendamiento de citas y control de historias clínicas para el consultorio médico Kennedy utilizando la metodología SWIRL se determinaron las siguientes conclusiones.

El uso y aplicación de una metodología de desarrollo de Software en este caso la SWIRL de una forma correcta y respetando su estructura permitió una fluida comunicación entre cliente y equipo de trabajo.

El uso y aplicación de herramientas CASE, hacen que la elaboración de prototipos de interfaces, diagramas UML y diseño de Base de Datos desempeñen un papel importante debido a que se los utilizó como guía para la construcción de los componentes del proyecto.

La selección de tecnología y herramientas adaptables a las necesidades del proyecto, además de la recolección de requisitos y levantamiento de información dan logro a un desarrollo e implementación exitoso del sistema cumpliendo y satisfaciendo todas las necesidades del cliente.

La aplicación de la normativa 9126, en conjunto con las herramientas SEO, permitieron tener un seguimiento en cuanto a resultados de recomendaciones para mejorar la calidad del sistema.

3.4. RECOMENDACIONES

La implementación de Frameworks en el desarrollo del sistema brinda la opción de tener un producto con alta mantenibilidad, diseño adaptable, eficiencia y estructuralmente ordenado y fácil de editar.

Realizar el número de pruebas que sean necesarias con la finalidad de detectar problemas o fallos y aplicar su respectiva corrección con el objetivo de obtener la mejor versión del sistema.

El uso de herramientas CASE, aportan al diseño y comprensión a la hora del desarrollo e implementación de los componentes del sistema.

El uso constante de herramientas de evaluación para comprobar la funcionalidad y eficiencia de los diferentes módulos del sistema y aplicar las recomendaciones dadas por las herramientas.

BIOGRAFÍA

- [1] C. E. Calle-Medina, D. G. García-Herrera, and J. C. Erazo-Álvarez, "Educación rural conectividad y accesibilidad: Retos y propuestas para el acceso a Internet en Joyagshí," *Cienciamatria*, vol. 7, no. 3, pp. 294–312, 2021, doi: 10.35381/cm.v7i3.581.

- [2] G. Martínez Villalobos, D. Flórez Méndez, and N. Bravo Osorio, “Desarrollo de un sistema web y móvil para la gestión de cultivos agrícolas,” *trilogía Cienc. Tecnol. Soc.*, vol. 10, no. 18, pp. 151–166, 2018, doi: 10.22430/21457778.669.
- [3] M. S. Ivette and B. Núñez, “El uso de ontologías como apoyo a la Ingeniería de Requisitos The use of ontologies to support Requirements Engineering,” vol. 15, no. 1, pp. 20–36, 2021.
- [4] E. Planas and J. Cabot, “Introducción de una herramienta OCL en la asignatura de Ingeniería de Requisitos Abstract,” vol. 6, pp. 35–42, 2021.
- [5] A. Razzaq, “A Systematic Review on Software Architectures for IoT Systems and Future Direction to the Adoption of Microservices Architecture,” *SN Comput. Sci.*, vol. 1, no. 6, 2020, doi: 10.1007/s42979-020-00359-w.
- [6] F. R. Sanabria and S. V. Rodríguez, “Evaluación de una Arquitectura de Software,” *Prospectiva*, vol. 19, no. 2, 2021.
- [7] Angular, “<https://angular.io/guide/what-is-angular>.” .
- [8] Gitbook.io, “<https://ngchallenges.gitbook.io/example-angular/kanban/kanban/arquitectura>.” .
- [9] Laravel, “Laravel.” <https://laravel.com/docs/9.x#meet-laravel>.
- [10] S. I. Ahmad, T. Rana, and A. Maqbool, “A Model-Driven Framework for the Development of MVC-Based (Web) Application,” *Arab. J. Sci. Eng.*, vol. 47, no. 2, pp. 1733–1747, 2022, doi: 10.1007/s13369-021-06087-4.
- [11] N. A. Bahrán, W. El-Shafai, A. Zekry, S. El-Rabaie, M. M. El-Halawany, and F. E. A. El-Samie, “An FPGA design and implementation of EPZS motion estimation algorithm for 3D H.264/MVC standard,” *Multimed. Tools Appl.*, vol. 78, no. 16, pp. 22351–22396, 2019, doi: 10.1007/s11042-019-7562-z.
- [12] R. Laura, “Arquitectura Pervasiva con tecnologías WEBRTC híbridas para el desarrollo de un Framework Modelo Vista Controlador de tiempo real,” *Rev. Investig. la Esc. Posgrado Univ. Nac. del Altiplano*, vol. 9, no. 4, pp. 12–45,

- 2020, [Online]. Available: <http://dx.doi.org/10.26788/riepg.2020.4.202>.
- [13] L.F. Santiago González *et al.*, “Aplicación Web Basada en el Patrón de Arquitectura de Software Modelo-Vista- Controlador (MVC) para Incrementar el Desempeño Académico en la Asignatura de Matemáticas Básicas,” pp. 9–25, 2019.
- [14] R. Luna Santos, R. A. Santos Quiroz, E. Carrera Carrasco, I. González Domínguez, and M. G. Santaella Benavente, “Creación e implementación de una aplicación web para la administración de citas para un autolavado,” *Rev. Investig. en Tecnol. la Inf.*, vol. 9, no. 19, pp. 97–108, 2021, doi: 10.36825/riti.09.19.008.
- [15] M. Ivanović, M. Vidaković, Z. Budimac, and D. Mitrović, “A scalable distributed architecture for client and server-side software agents,” *Vietnam J. Comput. Sci.*, vol. 4, no. 2, pp. 127–137, 2017, doi: 10.1007/s40595-016-0083-z.
- [16] X. Vilajosana and L. Navarro, “Arquitectura de Aplicaciones Web - Capa de Negocio,” *Arquit. Apl. Web - Capa Neg.*, p. 46, 2017.
- [17] I. Contreras and G. Hernández, “Sistema de localización en tiempo real mediante un servidor web y aplicaciones móviles,” *Pist. Educ.*, vol. 39, no. 127, pp. 171–186, 2017, [Online]. Available: <http://itcelaya.edu.mx/ojs/index.php/pistas%0Ahttp://www.itcelaya.edu.mx/ojs/index.php/pistas/article/view/1073>.
- [18] A. Quiroz Estrada, G. Alonso Acosta Amaya, R. Alberto Torres Villa, Q. Estrada, and P. Jaime Isaza Cadavid, “Diseño de un sistema internet de las cosas (IoT) para el monitoreo de la presión arterial Design of an internet of things (IoT) system for blood pressure monitoring,” *Rev. EIA*, vol. 18, no. 35, pp. 1–15, 2021, [Online]. Available: <https://doi.org/10.24050/reia>.
- [19] J. Sucari, “Modelos de desarrollo de software - Modelo V,” vol. 15, no. 1, pp. 37–51, 2021, [Online]. Available: <https://www.redalyc.org/articulo.oa?id=378366538003%0ACómo>.
- [20] B. Montero, H. Cevallos, and J. Dávila, “Metodologías ágiles frente a las

- tradicionales en el proceso de desarrollo de software,” *Espirales Rev. Multidiscip. Investig. ISSN 2550-6862*, vol. 2, no. 17, pp. 114–121, 2018, [Online]. Available:
https://www.researchgate.net/publication/327537074_Metodologias_agiles_frente_a_las_tradicionales_en_el_proceso_de_desarrollo_de_software.
- [21] G. Cicotti, “An evidence-based risk-oriented V-model methodology to develop ambient intelligent medical software,” *J. Reliab. Intell. Environ.*, vol. 3, no. 1, pp. 41–53, 2017, doi: 10.1007/s40860-017-0039-9.
- [22] A. Pai, G. Joshi, and S. Rane, “Integration of agile software development and robust design methodology in optimization of software defect parameters,” *Int. J. Syst. Assur. Eng. Manag.*, vol. 10, no. 5, pp. 1043–1051, 2019, doi: 10.1007/s13198-019-00833-6.
- [23] M. Romero and N. Escudero, “Propuesta de Metodología Híbrida y Base de Documentación para el Desarrollo de Software Actual,” *Concienc. Tecnológica*, vol. 60, no. 2, 2020, [Online]. Available:
<https://www.redalyc.org/articulo.oa?%0Ahttp://www.redalyc.org/articulo.oa?>
- [24] L. Neelu and D. Kavitha, “Estimation of software quality parameters for hybrid agile process model,” *SN Appl. Sci.*, vol. 3, no. 3, pp. 1–11, 2021, doi: 10.1007/s42452-021-04305-0.
- [25] A. Q. Gill, B. Henderson-Sellers, and M. Niazi, “Scaling for agility: A reference model for hybrid traditional-agile software development methodologies,” *Inf. Syst. Front.*, vol. 20, no. 2, pp. 315–341, 2018, doi: 10.1007/s10796-016-9672-8.
- [26] J. R. Molina Ríos and M. de las N. Pedreira-Souto, “*SWIRL*”, *metodología para el diseño y desarrollo de aplicaciones web*. 2019.
- [27] M. Stojmenovic, R. Biddle, J. Grundy, and V. Farrell, “The influence of textual and verbal word-of-mouth on website usability and visual appeal,” *J. Supercomput.*, vol. 75, no. 4, pp. 1783–1830, 2019, doi: 10.1007/s11227-018-2313-x.

- [28] A. Parikh, J. D. Patel, and A. K. Jaiswal, "Managing job applications online: integrating website informativeness and compatibility in theory of planned behaviour and technology acceptance model," *Decision*, vol. 48, no. 1, pp. 97–113, 2021, doi: 10.1007/s40622-020-00266-2.
- [29] N. P. Pérez, "Solución informática para la selección del servidor web durante la migración a código abierto," *Rev. Cuba. Ciencias Informáticas*, vol. 14, no. 2, pp. 49–69, 2020.
- [30] P. S. Atencio-ortiz, "Creación de un ataque DDoS utilizando HTTP-GET Flood a partir de la metodología Cyber Kill Chain Creation of a DDOS attack using HTTP-GET Flood with the Cyber Kill Chain methodology," vol. 16, pp. 41–47, 2020.
- [31] M. E. Chiroides Rojas, N. M. Arencibia Parada, A. A. Vitón Castillo, M. Linares Ríos, and M. Rodríguez Concepción, "Diseño del sitio web de la Universidad Virtual de Salud de Pinar del Río," *Rev. Ciencias Médicas Pinar del Río*, vol. 24, no. 3, pp. 1–8, 2020, [Online]. Available: <http://revcmpinar.sld.cu/index.php/publicaciones/article/view/4388>.
- [32] Y. Castro Blanco and R. González Hernández, "Aplicación web para gestionar información sobre personas discapacitadas," *Rev Inf Cient*, vol. 97, no. 4, pp. 710–721, 2018.
- [33] T. K. Dang, T. M. Huy, L. H. Dang, and N. Le Hoang, "An Elastic Data Conversion Framework: A Case Study for MySQL and MongoDB," *SN Comput. Sci.*, vol. 2, no. 4, 2021, doi: 10.1007/s42979-021-00716-3.
- [34] A. J. Ahmad and S. Priyadarshini, "Design and development of a friendly user interface for building social network traceability system," *Soc. Netw. Anal. Min.*, vol. 10, no. 1, pp. 1–8, 2020, doi: 10.1007/s13278-020-00675-2.
- [35] L. Khoa, "Laragon." <https://laragon.org/>.
- [36] M. A. Espinoza Mina and A. Y. Sierra Cedeño, "Análisis comparativo entre ASP.NETy PHP," *INNOVA Res. J.*, vol. 3, no. 4, pp. 25–43, 2018, doi: 10.33890/innova.v3.n4.2018.474.

- [37] D. Yan, B. Qi, Y. Zhang, and Z. Shao, "M-BiRank: co-ranking developers and projects using multiple developer-project interactions in open source software community," *Eurasip J. Wirel. Commun. Netw.*, vol. 2020, no. 1, 2020, doi: 10.1186/s13638-020-01820-3.
- [38] N. Bessghaier, A. Ouni, and M. W. Mkaouer, *A longitudinal exploratory study on code smells in server side web applications*, vol. 29, no. 4. Springer US, 2021.
- [39] M. M. Mirończuk, "The BigGrams: the semi-supervised information extraction system from HTML: an improvement in the wrapper induction," *Knowl. Inf. Syst.*, vol. 54, no. 3, pp. 711–776, 2018, doi: 10.1007/s10115-017-1097-2.
- [40] R. Xiao, Z. Wu, and K. Sugiura, "A semantic HTML based approach for geosensor media," *Geoinformatica*, vol. 22, no. 1, pp. 105–126, 2018, doi: 10.1007/s10707-016-0273-y.
- [41] C. R. Jaimez-González, "Portal web con recursos didácticos digitales para el aprendizaje de HTML y CSS / Web portal with HTML and CSS digital learning resources," *RIDE Rev. Iberoam. para la Investig. y el Desarro. Educ.*, vol. 8, no. 15, pp. 833–860, 2018, doi: 10.23913/ride.v8i15.323.
- [42] N. Battaglia, "Integración de una Herramienta CASE en un Entorno Académico Colaborativo para la Enseñanza de Ingeniería de Software," vol. 3, p. 211, 2017.
- [43] N. Battaglia, R. Martínez, C. Neil, and M. De Vincenzi, "XXIII Congreso Argentino de Ciencias de la Computación - Una Propuesta de Evaluación de Herramientas CASE para la Enseñanza," pp. 435–444, 2017, [Online]. Available: http://sedici.unlp.edu.ar/bitstream/handle/10915/63631/Documento_completo.pdf-PDFA.pdf?sequence=1.
- [44] T. do Carmo Nogueira, D. J. Ferreira, S. T. de Carvalho, L. de Oliveira Berretta, and M. R. Guntijo, "Comparing sighted and blind users task performance in responsive and non-responsive web design," *Knowl. Inf. Syst.*,

vol. 58, no. 2, pp. 319–339, 2019, doi: 10.1007/s10115-018-1188-8.

- [45] A. Parlakkiliç, “Evaluating the effects of responsive design on the usability of academic websites in the pandemic,” *Educ. Inf. Technol.*, vol. 27, no. 1, pp. 1307–1322, 2022, doi: 10.1007/s10639-021-10650-9.

ANEXOS.

ANEXO 1: ANÁLISIS DE RIESGO

Riesgo No:	RG-1			
Descripción	Poca disponibilidad de Clientes Interesados			
Etapa	Categoría	Stakeholders	Actividad	Probabilidad
Análisis	Alcance	Cliente Interesado	Poca Tiempo disponible de cliente interesado	30%

Riesgo No:	RG-2			
Descripción	Cambios o Añadir nuevos componentes al proyecto			
Etapa	Categoría	Stakeholders	Actividad	Probabilidad
Revisión y Pruebas	Alcance	Cliente Interesado	El cliente desea añadir componentes realizar cambios nuevos al sistema	50%
Implementación	Tiempo	Desarrolladores	Depende de los cambios a realizar o de los componentes a añadir	70%

Riesgo No:	RG-3			
Descripción	Tiempos no cumplidos según el cronograma			
Etapa	Categoría	Stakeholders	Actividad	Probabilidad
Planificación	Modelado	Equipo de Desarrollo	Los procesos de	30%

			diseño de modelos y prototipos están fuera de tiempo	
Implementación	Tiempo	Desarrolladores	Fuera de tiempo el proceso de codificación	20%

ANEXO 2: EDT DEL SISTEMA

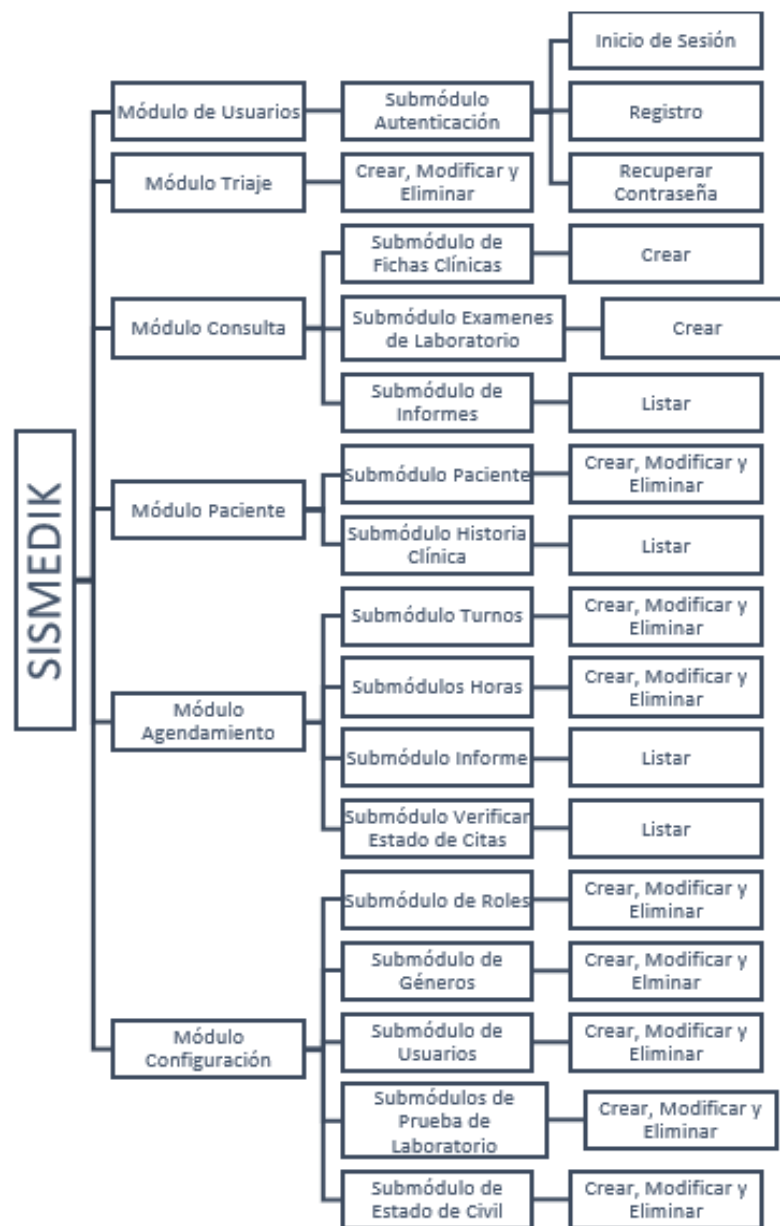


Figura 98: EDT del Sistema por Módulos

Fuente: Elaboración del Autor

ANEXO 3: CRONOGRAMA

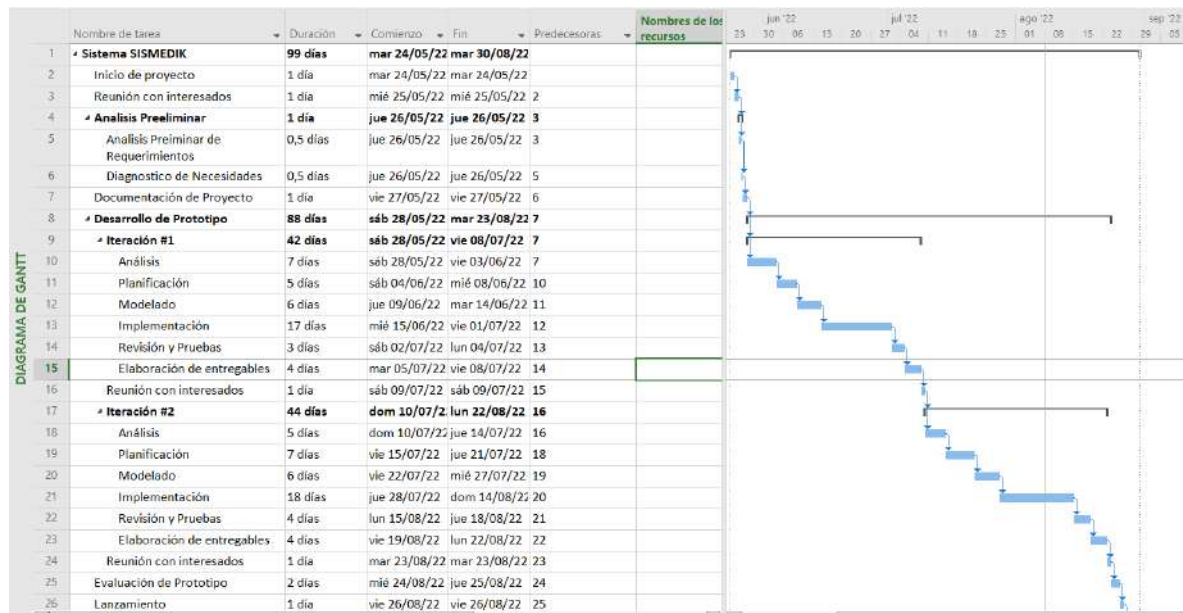


Figura 99: Cronograma de Actividades

Fuente: Elaboración del Autor

ANEXO 4: CONTROL DE CAMBIOS

Tabla 35: Control de Cambios de Proyecto

No	Descripción del Cambio	Fase - Iteración	Encargado	Fecha
1	Cambio de Modelo inicial de la Base de Datos Tabla Usuarios	Modelado: Iteración 1	John Díaz	25/05/2022
2	Cambio en Base de Datos agregar tabla roles	Modelado: Iteración 1	John Díaz	29/05/2022
3	Cambio en CRUD de Diagnóstico	Implementación: Iteración 1	John Díaz	05/06/2022
4	Modificar Interfaz de Registro de Usuario	Implementación: Iteración 1	John Díaz	15/06/2022
5	Modificar Interfaz de Noticias Médicas	Implementación: Iteración 2	John Díaz	02/07/2022
6	Mejoras en Página Home	Implementación: Iteración 2	John Díaz	10/07/2022
7	Control de Acceso a Botones en Página Home	Implementación: Iteración 2	John Díaz	15/07/2022
8	Implementar ChatBot	Implementación: Iteración 2	John Díaz	10/08/2022

Fuente: Elaboración del Autor