



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE UNA APLICACIÓN WEB PARA EL
AGENDAMIENTO Y CONTROL DE CITAS MÉDICAS UTILIZANDO
SEGURIDAD A NIVEL BASE DE DATOS

VILLAMAR ESPINOZA JOSUE EFRAIN
INGENIERO DE SISTEMAS

MACHALA
2022



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

DESARROLLO DE UNA APLICACIÓN WEB PARA EL
AGENDAMIENTO Y CONTROL DE CITAS MÉDICAS
UTILIZANDO SEGURIDAD A NIVEL BASE DE DATOS

VILLAMAR ESPINOZA JOSUE EFRAIN
INGENIERO DE SISTEMAS

MACHALA
2022



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN
PROPUESTAS TECNOLÓGICAS

DESARROLLO DE UNA APLICACIÓN WEB PARA EL AGENDAMIENTO Y
CONTROL DE CITAS MÉDICAS UTILIZANDO SEGURIDAD A NIVEL BASE DE
DATOS

VILLAMAR ESPINOZA JOSUE EFRAIN
INGENIERO DE SISTEMAS

VALAREZO PARDO MILTON RAFAEL

MACHALA, 23 DE FEBRERO DE 2022

MACHALA
2022

DESARROLLO DE UNA APLICACIÓN WEB PARA EL AGENDAMIENTO Y CONTROL DE CITAS MÉDICAS UTILIZANDO SEGURIDADES A NIVEL DE BASE DE DATOS

INFORME DE ORIGINALIDAD

10%

INDICE DE SIMILITUD

7%

FUENTES DE INTERNET

2%

PUBLICACIONES

3%

TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	Submitted to Universidad Internacional de la Rioja Trabajo del estudiante	1%
2	www.coursehero.com Fuente de Internet	<1%
3	1library.co Fuente de Internet	<1%
4	Submitted to Universitat Politècnica de València Trabajo del estudiante	<1%
5	cia.uagraria.edu.ec Fuente de Internet	<1%
6	doaj.org Fuente de Internet	<1%
7	scienti.minciencias.gov.co Fuente de Internet	<1%
8	hdl.handle.net	

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

El que suscribe, VILLAMAR ESPINOZA JOSUE EFRAIN, en calidad de autor del siguiente trabajo escrito titulado DESARROLLO DE UNA APLICACIÓN WEB PARA EL AGENDAMIENTO Y CONTROL DE CITAS MÉDICAS UTILIZANDO SEGURIDAD A NIVEL BASE DE DATOS, 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, 23 de febrero de 2022



VILLAMAR ESPINOZA JOSUE EFRAIN
0706794559

DEDICATORIA

Es trabajo de titulación producto de mi dedicación y esfuerzo lo dedico con gran cariño y agradecimiento a toda mi familia, de manera especial a mi padre Alpino Villamar, mi madre Yudi Espinoza y a mi hermana Juleysi Belén, que con su amor y sacrificio me han apoyado en todo momento para que culmine de la mejor manera mis estudios universitarios.

Sr. Villamar Espinoza Josué Efraín

AGRADECIMIENTO

Quiero dejar constancia de un sincero agradecimiento a la Universidad Técnica de Machala, a todos mis docentes por impartirnos sus sabias enseñanzas y, de manera especial al Ing. Milton Valarezo, por su apoyo incondicional en la dirección del presente trabajo.

Gracias a todos, pero sobre todo GRACIAS A DIOS.

Sr. Villamar Espinoza Josué Efraín

RESUMEN

Debido a que cada día nuestra sociedad es más dinámica, globalizada e interconectada, por lo que la utilización de una aplicación web ya no es una alternativa sino un requerimiento esencial en las distintas áreas como: salud, educación, comercio, finanzas, entre otras. La realidad de la crisis sanitaria que se vive a nivel mundial, en el área de la salud se ha incrementado la demanda de pacientes, para responder de una forma idónea todos los requerimientos de la población y brindar una adecuada atención médica es urgente que todos los centros hospitalarios, tanto públicos como privados utilicen la tecnología como un recurso estratégico.

Específicamente en la Clínica Johnson actualmente los pacientes acuden cuando sienten algún quebranto de su salud, son atendidos por el personal disponible en ese momento y solicitan una cita médica con profesionales especialistas que en ciertas ocasiones no tienen disponibilidad, esto genera un malestar tanto para el paciente como para el personal de la clínica, debido a la falta de una persona encargada exclusivamente para el agendamiento; el propietario de la clínica Johnson sintió la necesidad de optimizar recursos, maximizar tiempo y mejorar la eficiencia en cuanto al agendamiento de citas médicas, como respuesta a esta problemática nace la idea de desarrollar una aplicación web que permita la gestión y control de la citas médicas.

Para el desarrollo de esta propuesta tecnológica se utiliza la metodología Extreme Programming, puesto que es una de las metodologías ágiles con más auge en la actualidad debido a que está basada en la comunicación y las necesidades del cliente. Para la implementación de este proyecto se utilizó principalmente el lenguaje JavaScript el cual es muy utilizado en aplicaciones orientadas a la web en vista de la abundancia de librerías y frameworks, para el lado del front-end se empleó ReactJS, un framework diseñado para crear interfaces de usuario interactivas, mientras que para el back-end se utilizó NodeJS y la tecnología Express que proporciona módulos para escritura de manejadores con verbos HTTP, simplificando la configuración de un servidor web. Para almacenar la información se utilizó MongoDB como base de datos no relacional y se implementó una arquitectura de 3 nodos que permite la replicación de la información asegurando la disponibilidad y fiabilidad de los datos.

La aplicación web cuenta con 2 tipos de usuarios: Doctor y Paciente. El paciente puede escoger al doctor con la especialidad necesaria de acuerdo a sus síntomas, al elegir al doctor este visualizará el calendario con la disponibilidad del profesional y finalmente podrá seleccionar la fecha y hora para solicitar una cita; el doctor puede configurar su disponibilidad definiendo un horario de trabajo con hora de entrada, hora de salida y hora de descanso, terminado este proceso, está disponible para agendar o rechazar citas médicas solicitadas.

Con esta aplicación se optimiza notablemente el agendamiento de citas médicas aumentando la productividad de la clínica y generando satisfacción a los pacientes. En la sección final de este trabajo se encuentra la evaluación del sistema basado en el estándar de calidad ISO/IEC-9126 obteniendo calificaciones con resultados aceptables.

Palabras claves: Desarrollo web, Gestión cita médica, ReactJS, NodeJS, Replicación, Programación Extrema

ABSTRACT

Because every day our society is more dynamic, globalized and interconnected, so the use of a web application is no longer an alternative but an essential requirement in different areas such as: health, education, commerce, finance, among others. The reality of the health crisis that is being experienced worldwide, in the area of health, the demand for patients has increased, in order to respond in an appropriate way to all the requirements of the population and provide adequate medical care, it is urgent that all Hospital centers, both public and private, use technology as a strategic resource.

Specifically, at the Johnson Clinic, patients currently come when they feel a breakdown in their health, they are treated by the staff available at that time and request a medical appointment with specialist professionals who on certain occasions are not available, this generates discomfort for both the patient as to the clinic staff, due to the lack of a person in charge exclusively for scheduling; The owner of the Johnson clinic felt the need to optimize resources, maximize time and improve efficiency in terms of scheduling medical appointments. In response to this problem, the idea of developing a web application that allows the management and control of medical appointments was born.

For the development of this technology proposal, the Extreme Programming methodology is used, since it is one of the most popular agile methodologies today because it is based on communication and customer needs. For the implementation of this project, the JavaScript language was used mainly, which is widely used in web-oriented applications in view of the abundance of libraries and frameworks, for the front-end side ReactJS were used, a framework designed to create user interfaces while NodeJS and Express technology were used for the back-end, which provides modules for writing handlers with HTTP verbs, simplifying the configuration of a web server. To store the information, MongoDB was used as a non-relational database and a 3 node architecture was implemented that allows the replication of the information, ensuring the availability and reliability of the data.

The web application has 2 types of users: Doctor and Patient. The patient can choose the doctor with the necessary specialty according to her symptoms, when choosing the doctor, he will see the calendar with the availability of the professional and finally he will be able to select the date and time to request an appointment. The doctor can configure his availability by defining a work schedule with a check-in time, a checkout time and a break time. After this process, he is available to schedule or reject requests medical appointments.

With this application, the scheduling of medical appointments is significantly optimized, increasing the productivity of the clinic and generating patient satisfaction. In the final section of this work is the evaluation of the system based on the ISO/IEC-9126 quality standard, obtaining qualifications with acceptable results.

Keywords: Web Development, Medical Appointment Management, ReactJS, NodeJS, ReplicaSet, Extreme Programming.

ÍNDICE GENERAL

DEDICATORIA	1
AGRADECIMIENTO.....	2
RESUMEN.....	3
ABSTRACT.....	5
INTRODUCCIÓN	13
1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDAD Y REQUERIMIENTOS.....	14
1.1 Ámbito de aplicación: descripción del contexto y hechos de interés.....	14
1.2 Establecimiento de requerimientos.....	15
1.3 Justificación del requerimiento a satisfacer	16
2. CAPÍTULO II. DESARROLLO DEL PROTOTIPO.....	17
2.1 DEFINICIÓN DEL PROTOTIPO TECNOLÓGICO	17
2.1.1 Arquitectura de hardware.....	17
2.1.2 Arquitectura de software	18
2.2 FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO.....	19
2.2.1 Metodología de Desarrollo de Software	20
2.2.2 Metodología ágil Extreme Programming	21
2.2.3 Cliente.....	23
2.2.4 Servidor.....	24
2.2.5 Base de datos	24
2.2.6 Herramientas CASE.....	25

2.3 OBJETIVOS DEL PROTOTIPO	26
2.4 DISEÑO DEL PROTOTIPO	26
2.4.1 Tecnologías de Desarrollo	26
2.4.2. Fase de Análisis.....	27
2.4.3. Fase de Diseño.....	32
2.5 EJECUCIÓN Y/O ENSAMBLAJE DEL PROTOTIPO	42
2.5.1 Estructura de desglose de trabajo (EDT)	42
2.5.2. Portal Web	42
2.5.3. Inicio de sesión.	43
2.5.4. Recuperación de cuenta	44
2.5.5 Registro de usuarios	45
2.5.6 Configuración.....	50
2.5.7 Pacientes	51
2.5.8 Perfil.....	51
2.5.9 Cita médica (Doctor)	52
2.5.10 Cita médica (Paciente).....	53
3. CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO	55
3.1 Plan de evaluación	55
3.2.1 GtMetrix	56
3.2.2 Validator HTML	56
3.2.3 Validator (HTML y CSS).....	57
3.2.4 Seo Site Checkup	58
3.2.5 Metric Spot.....	58
3.2.6 Website grader.....	59

CONCLUSIONES	60
RECOMENDACIONES	61
BIBLIOGRAFÍA	62
ANEXOS.....	66
Anexo 1: Plantilla historias de usuarios	66
Anexo 2: Diseño responsivo	66
Anexo 3: Evaluación de calidad - Herramienta Website grader	67

ÍNDICE DE FIGURAS

Figura 1 Arquitectura del Cliente Servidor	17
Figura 2 Esquema del prototipo.....	20
Figura 3 Modelo Extreme Programming	22
Figura 4: Autenticación	32
Figura 5: BackUp.....	33
Figura 6: Replicación.....	34
Figura 7: Infraestructura de la base de datos	35
Figura 8: Esquema de la base de datos	36
Figura 9: Diseño de Interfaz - Portal Web.....	37
Figura 10: Diseño de Interfaz - Autenticación de usuarios.....	37
Figura 11: Diseño de Interfaz - Registro de usuario I.....	38
Figura 12: Diseño de Interfaz - Registro de usuario II.....	38
Figura 13: Diseño de Interfaz - Registro de usuario III.....	39
Figura 14: Diseño de Interfaz - Registro de pacientes.....	39
Figura 15: Diseño de Interfaz - Registro de doctores.....	40
Figura 16: Diseño de Interfaz - Recuperación de cuenta I.....	40
Figura 17: Diseño de Interfaz - Recuperación de cuenta II.....	41
Figura 18: Diseño de Interfaz - Perfil del paciente	41
Figura 19: Estructura de desglose de trabajo (EDT).....	42
Figura 20: Ejecución del prototipo - Portal Web.....	43
Figura 21: Ejecución del prototipo - Inicio de sesión.....	44
Figura 22: Ejecución del prototipo - Recuperación de cuenta I.....	44
Figura 23: Ejecución del prototipo - Recuperación de cuenta II.....	45
Figura 24: Ejecución del prototipo - Crear una cuenta.....	46
Figura 25: Ejecución del prototipo - Crear cuenta: Información básica.....	47
Figura 26: Ejecución del prototipo - Crear cuenta: Licencia médica	48
Figura 27: Ejecución del prototipo - Crear cuenta: Condiciones médicas.....	49
Figura 28: Ejecución del prototipo - Configuración de la cuenta.....	50
Figura 29: Ejecución del prototipo - Pacientes	51
Figura 30: Ejecución del prototipo - Perfil	51
Figura 31: Ejecución del prototipo - Cita médica (Doctor)	52

Figura 32: Ejecución del prototipo - Reservas / Doctor	52
Figura 33: Ejecución del prototipo - Cita médica (Paciente)	53
Figura 34: Ejecución del prototipo - Reservas / Paciente	54
Figura 35: Ejecución del prototipo - Comprobante.....	54
Figura 36: Evaluación - GtMetrix	56
Figura 37: Evaluación - Validator html	57
Figura 38: Evaluación - Validator (html y css juntos)	57
Figura 39: Evaluación - Seo Site Checkup	58
Figura 40: Evaluación - Metric Spot.....	59
Figura 41: Evaluación - Website grader.....	59

ÍNDICE DE TABLAS

Tabla 1: Tecnologías utilizadas	26
Tabla 2: Ingreso al sistema.....	27
Tabla 3: Gestión de Pacientes.....	28
Tabla 4: Gestión de Doctores	28
Tabla 5: Agendamiento de citas médicas	29
Tabla 6: Mensajería Instantánea	29
Tabla 7: Recordatorios	30
Tabla 8: Seguridad y Validación de la información	30
Tabla 9: Disponibilidad de la información	31
Tabla 10: Fácil mantenimiento y mejora	31
Tabla 11: Resultados de la evaluación	55
Tabla 12: Plantilla historia de usuario	66

INTRODUCCIÓN

Cada día nuestra sociedad es más dinámica, globalizada e interconectada, por lo que la utilización de una aplicación web ya no es una alternativa sino un requerimiento esencial en las distintas áreas como: salud, educación, comercio, finanzas, entre otras.

En el área de la salud, refiriéndonos específicamente a la Clínica Johnson sabemos que tiene la necesidad de optimizar recursos, maximizar tiempo y mejorar la eficiencia en cuanto al agendamiento de citas médicas. Por este motivo se decidió desarrollar una aplicación web de gestión de citas médicas aplicando técnicas de seguridad en base de datos.

La estructura del presente documento es la siguiente:

El **Capítulo 1**, denominado el diagnóstico de necesidad y requerimientos, detallará el ámbito de aplicación donde se establecen los requerimientos, y la justificación del requerimiento a satisfacer, describiendo el contexto y hechos de interés respectivos del problema que se busca resolver

Seguido en el **Capítulo 2**, se puntualiza el desarrollo del prototipo, donde se definirá el prototipo tecnológico, además de la fundamentación teórica necesaria, cómo la metodología de desarrollo de software, la base de datos o las herramientas CASE, conjuntamente se explicará la ejecución y/o ensamblaje del prototipo.

Finalmente, en el **Capítulo 3**, encontraremos los planes de evaluaciones mediante páginas web como GtMetrix, Metric Spot, Website grader, entre otras, así mismo las recomendaciones y conclusiones de la aplicación web propuesta.

1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDAD Y REQUERIMIENTOS

1.1 ÁMBITO DE APLICACIÓN: DESCRIPCIÓN DEL CONTEXTO Y HECHOS DE INTERÉS

Vivimos en la era tecnológica, no es una realidad lejana que las empresas busquen la innovación, con el fin de satisfacer las necesidades de los clientes, dichas empresas buscan realizar cambios ya sean en su estructura, procesos, productos, sistemas, tecnología, u otros aspectos importantes dentro de la organización [1].

Bajo estos lineamientos nos referiremos a La Clínica Johnson como el sujeto de investigación, dicha clínica ubicada en la ciudad de Machala busca brindar a la sociedad la mejor atención médica, al paciente y su familia, el centro de salud ofrece asistencia en Ginecología, Pediatría, Medicina interna y general, Cirugía General y Laparoscópica, Apendicectomía, Colecistectomía, Traumatología, Ecografías y Exámenes de Laboratorio.

Actualmente los pacientes acuden a la clínica cuando estos sienten alguna molestia o malestar, y son atendidos por el personal disponible en ese momento, algunos prefieren agendar una cita médica, generalmente ellos solicitan una hora para ser atendidos y la persona correspondiente indica el personal apto para atenderla, sin embargo, existe un déficit notable al momento de realizar los agendamientos.

Entre las personas que allí laboran, se encuentra a doctores, enfermeras o enfermeros, más no a una persona encargada solo del agendamiento, recordemos que este trabajo lleva algo de tiempo, puesto que necesita haber un intercambio considerable de información, entre ellos los datos del paciente y sus síntomas, la idea de desarrollar una aplicación web nace a raíz de esa falencia.

El diseño del sistema web asegurará la confidencialidad de los pacientes, es decir ya no se necesitará a una persona intermediaria, proporcionando una seguridad extra en los

datos, mejorará la gestión de las citas médicas, optimizando tiempo, y recursos. Se resolvió implementar la aplicación web basándose en el modelo cliente/servidor gracias a las ventajas que esta arquitectura ofrece. Dado que brinda la posibilidad de que los cambios únicamente se realicen en el servidor, que se logre entrar a ella a partir de cualquier máquina conectada a la red por medio de un navegador web sin necesidad de instalarse, y que la logren utilizar diversos usuarios a la vez [2].

1.2 ESTABLECIMIENTO DE REQUERIMIENTOS

El proceso administrativo de cualquier empresa consta de 4 pilares fundamentales: la planificación, organización, dirección, y control, cada uno de ellos proyectan una escalera, donde se necesita subir uno por uno, al momento de planificar que es el primer escalón que se debe subir, se necesita efectuar un inventario acerca del qué y cómo se están realizando las operaciones, además de la persona encargada de realizar esa acción [3].

En las redes sociales se encuentran los números de contacto de la clínica, sin embargo, el número principal de conexión responde a un teléfono que es manejado por una de las propietarias de la clínica, entre sus responsabilidades no están como prioridad atender los mensajes y demás consultas que los pacientes realizan vía WhatsApp, se realizó una entrevista a la administradora de la clínica para conocer de cerca el proceso administrativo, y la experiencia que han tenido a lo largo de los años con el agendamiento de las citas médicas.

Ella nos explicó que el problema surgió a raíz de la pandemia cuando el número de pacientes incrementó, implicando mayor trabajo para el personal de la clínica, además de que un porcentaje considerable de personas empezaron a utilizar los medios digitales para evitar salir y contagiarse, fue entonces cuando las redes sociales empezaron a tomar mayor fuerza, y el responder a todas las personas de manera personalizada se volvió complejo.

Los medios digitales tienen varias ventajas, considero principal a la interacción, las aplicaciones web se caracterizan no solo por ser medios donde el usuario obtenga información, sino un espacio donde interactuar y navegar entramado en información [4]. En este proyecto la interacción partirá del médico al paciente, donde el servicio de mensajería cumplirá con la necesidad de comunicación confidencial entre los servidores.

1.3 JUSTIFICACIÓN DEL REQUERIMIENTO A SATISFACER

El propósito del desarrollo de la aplicación web es asegurar la confidencialidad de los pacientes, entre las ventajas que se pueden considerar están: reducir el volumen del archivo, optimización del tiempo y ahorro de los recursos, la Clínica Johnson podrá modernizarse en el ámbito digital contando con presencia de la misma.

La diferencia de ese sistema de reserva con respecto a otros es que será el usuario quién elija según sus necesidades y preferencias al médico que pueda tratarlo, de esta manera la clínica podrá asegurar una verdadera atención personalizada, además que el usuario no necesita instalar una aplicación para poder realizar dicha reserva, lo podrá hacer desde su ordenador, smartphone o cualquier dispositivo con acceso a internet.

Se considera, además que este proyecto prioriza la seguridad de los datos, según la Ley Orgánica de Protección de Datos Personales, el tratamiento de los datos deberá ser sobre la base del sigilo y secreto, esto quiere decir, la información obtenida no debe ser usada para un fin diferente para el que fue obtenido [5], [6]. En tal caso la historia clínica sólo podrá ser manejada por personal de la cadena sanitaria [7].

El Convenio 108 ya se encargó de regular este tema en su artículo 7, el cual establece que “se tomarán medidas de seguridad apropiadas para la protección de datos de carácter personal registrados en ficheros automatizados contra la destrucción accidental o no autorizada, o la pérdida accidental, así como contra el acceso, la modificación o la difusión no autorizados”.

2. CAPÍTULO II. DESARROLLO DEL PROTOTIPO

2.1 DEFINICIÓN DEL PROTOTIPO TECNOLÓGICO

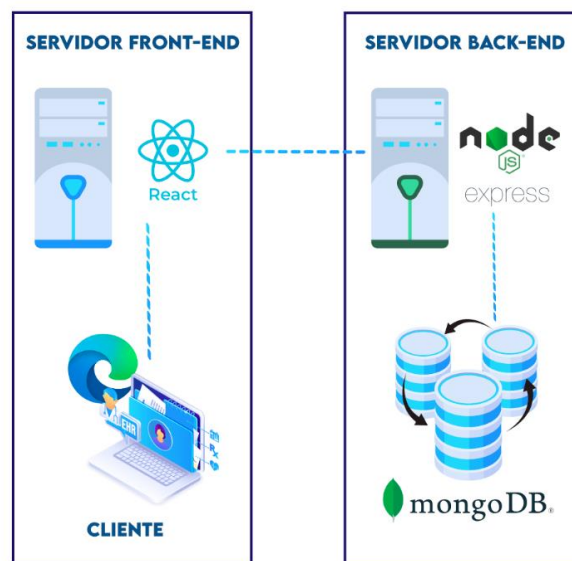
En el siguiente apartado del actual trabajo de titulación, se presentará la conceptualización de la aplicación web, la arquitectura, los interesados, el lenguaje de programación y las tecnologías utilizadas en cada una de las etapas de desarrollo.

2.1.1 Arquitectura de hardware

En el prototipo piloto que se aprecia en la **Figura 1** se puede notar la arquitectura cliente-servidor que se aplica, este prototipo tecnológico divide sus responsabilidades en el front-end y back-end.

El back-end es “el lado del servidor”, en este lado se toman los datos, se procesa la información y se envía al usuario, todo a través de una computadora diseñada para operar las 24 horas del día, los 7 días de la semana. Por su parte el front-end es “el lado del cliente”, para la implementación de la aplicación web se involucran diversas tecnologías desarrolladas en los lenguajes de HTML, CSS y JavaScript; con las que se pueden interactuar directamente con el usuario [8].

Figura 1 Arquitectura del Cliente Servidor



Fuente: Elaboración Propia

2.1.2 Arquitectura de software

La arquitectura de software hace referencia a la composición de los elementos de una técnica, sus interrelaciones, principios y directrices que rigen su diseño y evolución en todo el tiempo [9]. Para el desarrollo de la aplicación web en cuestión se ha propuesto un modelo de 3 capas, puesto que esta arquitectura es la más utilizada en el mundo de los sistemas. Se decidió utilizar este modelo basado en capas, para no modificar los detalles que se trabajen en el programa, viéndose ventajoso la manipulación del sistema, sin alterar ni perjudicar las demás capas. Las capas se detallan a continuación:

2.1.2.1 Capa de presentación.

Este primer nivel del sistema es la parte que se logra observar en el navegador, es la responsable de llevar a cabo la funciones solicitadas para que los usuarios puedan interactuar con la aplicación [10]. Esta capa es popularmente conocida como la interfaz gráfica, en este espacio el usuario podrá interactuar, enviar y recibir información.

Dado que el navegador web puede entender únicamente el lenguaje JavaScript, se suele programar con lenguajes de programación Frameworks como: Angular, Vue, o ReactJS, siendo este último el lenguaje utilizado para el desarrollo de la aplicación, a diferencia de los otros lenguajes ReactJS ayuda a producir interfaces interactivas de una manera sencilla, además este lenguaje se actualiza y renderiza de forma eficiente los elementos correctos cuando la página lo necesite [11].

2.1.2.2 Capa de negocios.

Es donde se programa la lógica del negocio, a diferencia de la capa de presentación que se relaciona con el front-end, la capa de negocios se relaciona con el back-end, este nivel intermedio se encarga de vigilar la potencia y la secuencia de tareas, además esta capa asegura la integridad de todos aquellos lineamientos necesarios para cumplir con todos los requerimientos [12].

Para el servidor back-end existen varios tipos de lenguajes de programación, mismos que permiten crear servidores web por mencionar algunos ejemplos: Php, Python o Java, en el caso de la propuesta presentada se utilizó JavaScript, entre sus ventajas se puede

mencionar su versatilidad, lo que facilita el desarrollo de una aplicación completa utilizando únicamente este lenguaje [13], hoy en día es uno de los lenguajes de programación más utilizados gracias a su flexibilidad y potencia [14], cabe recalcar que en esta capa se utilizará la tecnología Express conocida como un Framework de NodeJS para el manejo de peticiones HTTP.

2.1.2.3 Capa de datos.

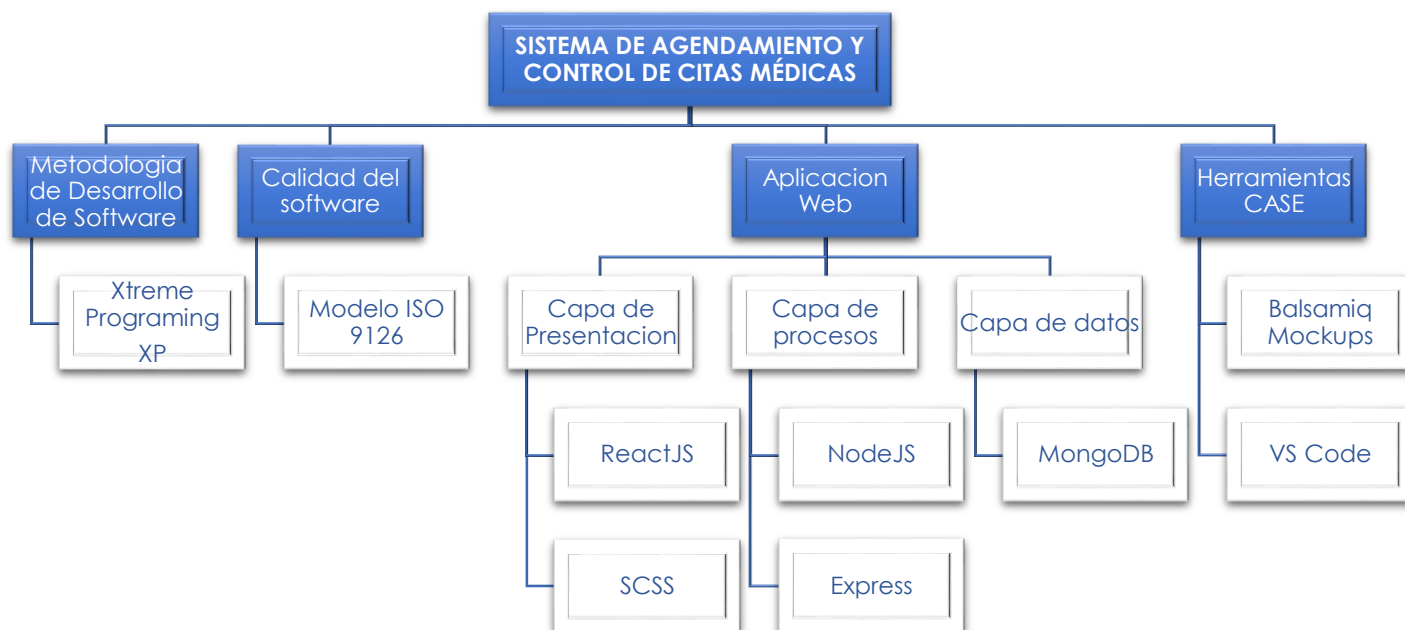
En esta tercer nivel se utiliza un motor de base de datos el cual está encargado de almacenar la información [15] y modificar los datos ya existentes. Actualmente se pueden trabajar con 2 tipos de base de datos: Structured Query Language en sus siglas “SQL” y Not only SQL en sus siglas “NoSQL”, el cual puede contribuir al manejo de la información [16], la principal diferencia entre las bases de datos radica en la forma de guardar los datos, donde la segunda almacena la información como documentos lo que permite ser más escalables y ofrecer un mayor rendimiento.

El motor de almacenamiento o motor de base de datos utilizado es MongoDB, la empresa que administra esta base de base datos en la nube es Mongo Atlas, se hace mención de lo anterior para indicar que la Database necesita estar en la nube para que el servidor tenga acceso a los datos y estos puedan ser brindados a los clientes en formato Json.

2.2 FUNDAMENTACIÓN TEÓRICA DEL PROTOTIPO

En la **Figura 2** se puede observar en forma de organizador gráfico la composición en elementos de los instrumentos y de las tecnologías aplicadas jerárquicamente para el correcto desarrollo del proyecto tecnológico propuesto.

Figura 2 Esquema del prototipo



Fuente: Elaboración Propia

2.2.1 Metodología de Desarrollo de Software

El método de desarrollo muestra el proceso para desarrollar un sistema de información competente, por su parte la metodología de desarrollo de software es definida como el conjunto de técnicas para analizar y desarrollar un software, esta agrupación controla e introduce ejercicios concretos para la estabilidad a lo largo de las etapas [17]. La metodología concibe 2 arquetipos, conocidas como tradicionales y ágiles [18].

2.2.1.1 Metodología de Desarrollo de Software Tradicional

Las metodologías tradicionales suelen centrar su vasta información en el desarrollo del sistema [19], sin embargo, estas metodologías no son las ideales si se busca un espacio que esté en constante cambio dado que no soporta las modificaciones de la planificación por completo.

Tienen como propósito generar software eficaces y eficientes en un tiempo específico, para ello hacen hincapié en la fase de planeación, donde puntualiza todas las técnicas, ocupaciones y equipos que se involucran en el ciclo del programa. En las metodologías tradicionales la fase de diseño no empieza a ejecutarse si no hasta que culmine la fase

de análisis, a su vez la fase de codificación no empieza hasta que la fase anterior haya finalizado, generando un proceso lineal conocido también como modelo en cascada.

2.2.1.2 Metodología de Desarrollo de Software Ágil

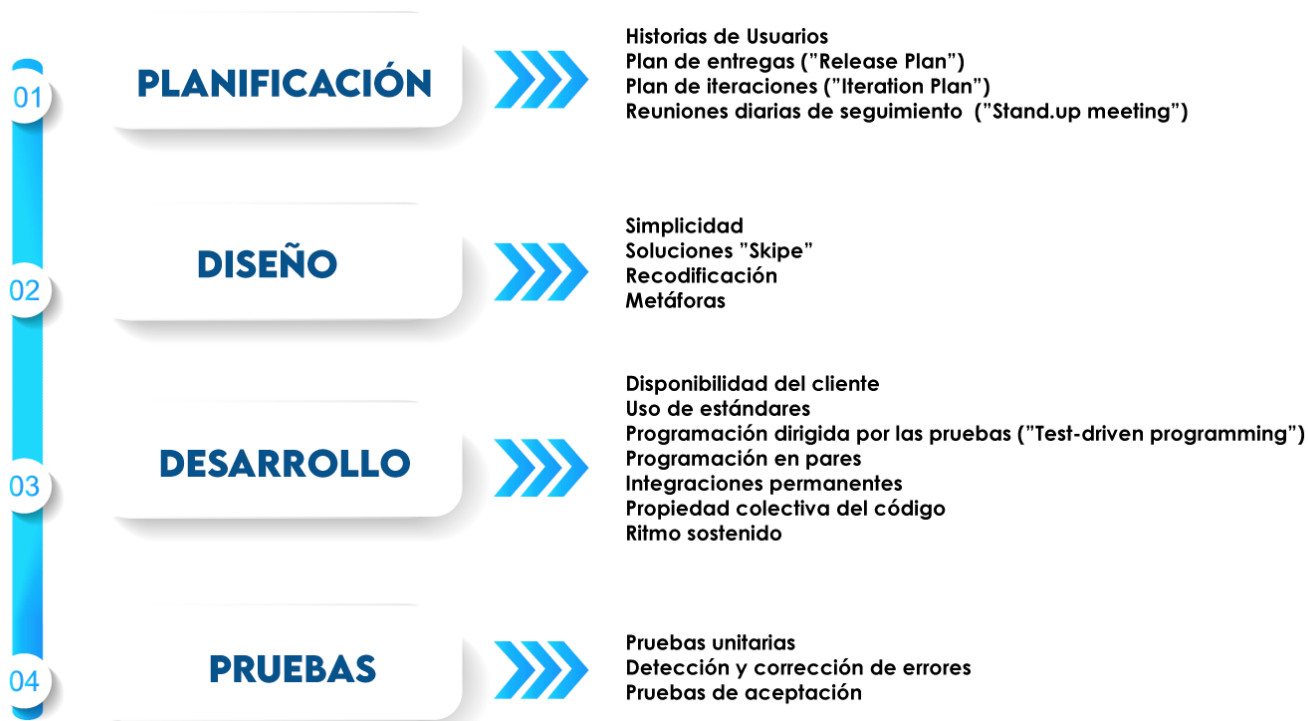
El actual trabajo de titulación emplea una metodología ágil, el desarrollo del software revela su complejidad la cual fue analizada desde el uso de las metodologías ágiles y las prácticas de su misma apariencia, ambas surgieron a partir del manifiesto ágil y sus principios a diferencias de la metodología tradicional [20].

Esta metodología ejecuta cada fase del desarrollo de un software: análisis, diseño, desarrollo y pruebas, con el fin para tener resultados en un breve periodo de tiempo, reduciendo la proporción de documentación y centrarse en el desarrollo del sistemas, por medio de la ejecución de prototipos de manera habitual [21]. Actualmente es la metodología ágil la más utilizada por los programadores, tanto así que se decidió usar Extreme Programming (XP) como metodología, además que ayuda a incrementar la productividad del equipo desarrollador [22].

2.2.2 Metodología ágil Extreme Programming

Extreme Programming también entendida como XP, como se observa en la Figura 3 es una metodología liviana, esto resulta beneficioso para el transcurso normal del desarrollo de la aplicación, como la definición de requisitos extensos y la documentación vasta [23], además hace énfasis en que los conjuntos de desarrollo sean pequeños y el código sencillo [24].

Figura 3 Modelo Extreme Programming



Fuente: Elaboración Propia

2.2.2.1 Fase de Planificación

En esta primera etapa, se antepone y se desmonta en mini-versiones, dentro de este proceso es importante contar con pasos como "release planning" donde se planea las historias que el usuario necesita, otro aspecto importante es las interacciones, la velocidad del proyecto, además de la programación en pareja, son aspectos que ayudarán a obtener un programa eficaz, servible, listo para probar y lanzar [25].

2.2.2.2 Fase de Diseño

Dado la metodología aplicada esta etapa sugiere elaborar diseños nada complejos, para que estos sean expeditamente entendible y funcional, se recomienda usar correctamente los términos, para evitar problemas en cuanto a la reutilización del código, cabe recalcar que los diseños elaborados en esta etapa están enfocados en los objetivos ayudándose de Tarjetas CRC (Clase, Responsabilidad y Colaboración) las cuales ayudan al o los programadores concentrarse y valorar los avances del proyecto.

2.2.2.3 Fase de Desarrollo

Al momento de codificar es de vital importancia especificar a detalle los requerimientos del usuario [26], para esta etapa debemos considerar al cliente como un miembro más del equipo, la codificación debe atenerse a los estándares de codificación manejados hasta el momento, es importante reconocer que debemos dejar la optimización del código hasta el final, antes de necesita asegurar que el código esté funcionando, para optimizarlo posteriormente, de esta forma se conseguirá una programación estructurada y planificada.

2.2.2.4 Fase de Pruebas

Entre los pilares de la metodología XP es la implementación de test que ayuden a comprobar el correcto manejo del código que se esté ejecutando, estas pruebas automáticas serán constantemente, el mismo cliente puede realizar las pruebas si así lo desea, validando las mini-versiones, este testeo frecuente nace ante la necesidad de comprobar la eficacia del código en proyectos a corto plazo.

2.2.3 Cliente

El modelo cliente-servidor es un tipo de comunicación que sujeta a diversos dispositivos informáticos por medio de la red, este centra los requerimientos del destinatario a través de la interacción y de las contestaciones [27]. Como ya se había mencionado anteriormente se involucran diversos lenguajes de front-end como HTML, CSS y JavaScript.

2.2.3.1 Framework ReactJS

El framework es definida como una estructura o marco de trabajo que busca reducir la elaboración de las tareas [28], siendo utilizada como un organizador para el desarrollo de software, ReactJS es un framework desarrollado por Facebook, lo que le permite ser una biblioteca JavaScript de código abierto bastante amplia, por este y otros motivos los programadores lo prefieren para crear interfaces de usuario de una forma más ágil [29].

2.2.3.2 Preprocesador SCSS

Hoy en día encontrar un programa desarrollado con el lenguaje CSS es habitual, dado que es una de las tres principales tecnologías web, la mayor parte de preprocesadores CSS añadirán varias propiedades que no hay en CSS puro, como variable, mixins,

selectores anidados, entre otros [30], la diferencia de este preprocesador y SCSS radica en la posibilidad que tiene de añadir muchas más variables en el programa, lo que permite hacerlo más dinámico tanto para el usuario como el programador.

2.2.4 Servidor

Dentro de informática el servidor es el mecanismo informático que pertenece a una red y suministra servicios a los demás conjuntos cliente [31], el servidor puede cumplir con varios roles, refiriéndonos a los servidores de archivo, directorio, impresión, correo, proxy, Base de Datos, Web, DNS, DHCP o FTP respectivamente, todos los servidores anteriormente mencionados buscan cumplir a las demandas de los consumidores.

2.2.4.1 Framework NodeJS

NodeJS está elaborado para producir aplicaciones y servicios escalables [32], gracias a la vasta pluralidad de librerías accesibles para programar, la ejecución se da en tiempo real lo que permite usarse fuera de la trama del navegador [33] e incrementar la interoperabilidad de la información [27].

2.2.4.2 Librería Express

Por su parte, Express es la librería tácita para la mayoría de frameworks web de NodeJS, proporciona módulos para escritura de manejadores con verbos HTTP, modifica ajustes de los programas, agrega requerimientos "middleware" en algún punto necesario dentro del manejo o entrega de archivos estáticos, esta librería cuenta con múltiples ventajas por mencionar a las importantes, cuenta con paquetes de middleware compatibles para solucionar imprevistos durante el desarrollo web [33].

2.2.5 Base de datos

La base de datos es la colección estructurada de información o datos ordenados, en otras palabras es un conjunto de archivos, estratégicamente organizados [34],[35] de esta forma la base de datos está integrada, afianza la obtención de los datos y la seguridad de los mismos.

2.2.5.1 MongoDB

Es un sistema de base de datos no relacional enfocada en documentaciones y considerablemente utilizada en todo el mundo [36], en menos palabras está encaminado a documentos de código abierto [37], es decir entre las características que destacan de esta base están la velocidad, el almacenamiento de archivos, y la seguridad, a pesar de ser una tecnología emergente MongoDB apertura el control de la seguridad integrada en los datos [38], diversas técnicas de seguridad, tales como la encriptación, el respaldo y la replicación.

2.2.6 Herramientas CASE

CASE es la abreviatura de “Computer Aided Software Engineering” o en su significado en español “Ingeniería de Software Asistida por Computación”, estas herramientas figuran maneras posibles de crear procesos de negocios y ejecutar sistemas, ya que son programas que facilitan la creación de una aplicación web, permitiendo el uso de la informática para planificar la programación.

2.2.6.1 VSCode

Visual Studio Code, es el editor de código fuente creado por Microsoft para Windows, Linux y MacOS, esta herramienta adopta la facilidad de un editor de código siendo este accesible a los requerimientos que los desarrolladores buscan cumplir [39]. Lo llamativo de este código es la variación de color que diferencia a cada lenguaje lo que facilita su lectura.

2.2.6.2 Balsamiq Mockups

Balsamiq Mockups es una herramienta que ayuda en la creación de modelos del sitio web, este instrumento juega un papel importante dentro de la planificación de una aplicación software [40] puesto que permite hacer bocetos de la mencionada aplicación, impulsa al desarrollador a organizar el contenido y la distribución de los elementos y mejorar la experiencia del usuario.

2.3 OBJETIVOS DEL PROTOTIPO

Objetivo General

Desarrollar una aplicación web que asegure la confidencialidad de los pacientes utilizando la seguridad a nivel base de datos, permitiendo mejorar la gestión de las citas médicas en la Clínica Johnson de la ciudad de Machala.

Objetivos Específicos

- Implementación de la metodología XP para el desarrollo de la aplicación web.
- Aplicar los conocimientos de programación básica y avanzada, base de datos y análisis de sistemas informáticos para el desarrollo, codificación, y pruebas del programa.
- Evaluar la calidad de la aplicación utilizando distintas herramientas web para asegurar el correcto funcionamiento del mismo.
- Utilizar fuentes bibliográficas confiables para asegurar la fiabilidad de la investigación.

2.4 DISEÑO DEL PROTOTIPO

2.4.1 Tecnologías de Desarrollo

Las tecnologías utilizadas para el desarrollo de la aplicación web se detallan a continuación.

Tabla 1: Tecnologías utilizadas

TECNOLOGÍAS	
JavaScript	Lenguaje de programación
NodeJS	Entorno de ejecución de JavaScript
Express	Framework de NodeJS para manejo de peticiones HTTP
ReactJS	Framework de JavaScript para el cliente
SCSS	Preprocesador de CSS
MongoDB	Base de datos NoSQL
MongoAtlas	Gestor para base de datos MongoDB
VSCode	Editor de código

Fuente: Elaboración Propia

2.4.2. Fase de Análisis

2.4.2.1. Historias de usuario

Luego de una interesante reunión entre el Lcda. Lenis Espinoza, administradora de la clínica Johnson, el Ing. Milton Valarezo y el estudiante Josué Villamar, se llega al planteamiento de los requerimientos y delimitación del sistema, los requisitos fueron planteadas utilizando las historias de usuarios (**Anexo 1**) propuesta por la metodología Extreme Programming.

Tabla 2: Ingreso al sistema

Historia de Usuario	
Número: 1	Usuario: Doctor, Paciente
Nombre Historia: Ingreso al sistema	
Prioridad en negocio: Media	Riesgo en negocio: Alta
Programador Responsable: Josué Efraín Villamar Espinoza	
Descripción: Los usuarios podrán iniciar sesión en el sistema empleando su correo y una contraseña. Cada usuario tendrá las posibilidades de: resetear, cambiar su contraseña y recuperar la contraseña olvidada. La contraseña debe cumplir estándares de seguridad. Entrada de datos validada.	
Observaciones: Solo podrán ingresar al sistema los usuarios con un rol asignado: Doctor, Paciente	

Fuente: Elaboración Propia

Al ingresar el usuario, el sistema será capaz de detectar si es Doctor o Paciente e inmediatamente mostrará su perfil; la información del perfil dependerá del rol del usuario y, éste podrá resetear, cambiar y recuperar la contraseña.

Tabla 3: Gestión de Pacientes

Historia de Usuario	
Número: 2	Usuario: Paciente
Nombre Historia: Gestión de Pacientes	
Prioridad en negocio: Alta	Riesgo en negocio: Alta
Programador Responsable: Josué Efraín Villamar Espinoza	
<p>Descripción: Los pacientes tendrán acceso al sistema a través del Portal Web para solicitar citas médicas. El usuario para registrarse como paciente, debe ingresar los siguientes datos: condición médica, alergias, medicamentos, tipo de sangre y peso.</p>	
<p>Observaciones: Para la creación de pacientes el correo electrónico de ser único.</p>	

Fuente: Elaboración Propia

Se identifica como Paciente a las personas con una cuenta creada en el sistema; el Paciente podrá modificar la información que se muestra en su perfil o eliminar su cuenta si el caso lo amerita.

Tabla 4: Gestión de Doctores

Historia de Usuario	
Número: 3	Usuario: Doctor
Nombre Historia: Gestión de Doctores	
Prioridad en negocio: Alta	Riesgo en negocio: Alta
Programador Responsable: Josué Efraín Villamar Espinoza	
<p>Descripción: Los doctores tendrán acceso al sistema a través del Portal Web para aceptar o rechazar citas médicas solicitadas. El usuario para registrarse como doctor, debe ingresar los siguientes datos: licencia médica, especialidad, subespecialidad, nivel de estudio.</p>	
<p>Observaciones: Para la creación de doctores el correo electrónico de ser único.</p>	

Fuente: Elaboración Propia

Se identifica como Doctor a los médicos con una cuenta creada en el sistema, donde receptorá las citas médicas; el Doctor podrá modificar la información que se muestra en su perfil o eliminar su cuenta si es necesario.

Tabla 5: Agendamiento de citas médicas

Historia de Usuario	
Número: 4	Usuario: Paciente
Nombre Historia: Agendamiento de citas médicas	
Prioridad en negocio: Alta	Riesgo en negocio: Alta
Programador Responsable: Josué Efraín Villamar Espinoza	
<p>Descripción: El Paciente tendrá la posibilidad de agendar y editar citas médicas. El paciente debe ajustarse a la disponibilidad en el horario de atención del doctor.</p>	
<p>Observaciones: El Doctor no tendrá acceso al historial de citas médicas de otro Doctor.</p>	

Fuente: Elaboración Propia

El paciente puede solicitar su cita médica solo a los Doctores que tengan un horario establecido, al ser así el paciente debe elegir la fecha y hora a su conveniencia.

Tabla 6: Mensajería Instantánea

Historia de Usuario	
Número: 5	Usuario: Doctor, Paciente
Nombre Historia: Mensajería Instantánea	
Prioridad en negocio: Alta	Riesgo en negocio: Media
Programador Responsable: Josué Efraín Villamar Espinoza	
<p>Descripción: El Doctor y el Paciente podrán comunicarse solo si el paciente tiene agendada una cita médica. El sistema permitirá al doctor estar en contacto con el paciente para facilitar el proceso de registro de una nueva cita médica.</p>	
<p>Observaciones:</p>	

Fuente: Elaboración Propia

El paciente que ya tiene registrada una cita médica tiene la posibilidad de hacer uso de esta, consultar alguna inquietud médica al Doctor y agendar una nueva cita.

Tabla 7: Recordatorios

Historia de Usuario	
Número: 6	Usuario: Paciente
Nombre Historia: Recordatorios	
Prioridad en negocio: Medio	Riesgo en negocio: Media
Programador Responsable: Josué Efraín Villamar Espinoza	
<p>Descripción: El sistema enviará mensajes SMS automáticos como recordatorios a los pacientes de sus citas médicas confirmadas.</p>	
<p>Observaciones: Los Doctores son los únicos que pueden confirmar una cita médica.</p>	

Fuente: Elaboración Propia

El paciente se beneficiará de un recordatorio de su cita médica un día antes de la fecha seleccionada, aún si esta fecha ha sido modificada y confirmada por el Doctor.

Tabla 8: Seguridad y Validación de la información

Historia de Usuario	
Número: 7	Usuario: Doctor, Paciente
Nombre Historia: Seguridad y Validación de la información	
Prioridad en negocio: Alta	Riesgo en negocio: Alta
Programador Responsable: Josué Efraín Villamar Espinoza	
<p>Descripción: El sistema almacena los datos de forma segura. Los datos deberán ser validados antes de ser almacenados. El sistema asegura la confidencialidad de los datos.</p>	
<p>Observaciones:</p>	

Fuente: Elaboración Propia

Los datos que tanto el doctor como el paciente ingrese a este aplicativo web deben ser verdaderos, y serán de uso exclusivo por las personas autorizadas.

Tabla 9: Disponibilidad de la información

Historia de Usuario	
Número: 8	Usuario: Doctor, Paciente
Nombre Historia: Disponibilidad de la información	
Prioridad en negocio: Alta	Riesgo en negocio: Alta
Programador Responsable: Josué Efraín Villamar Espinoza	
Descripción: Los datos estarán disponibles 24/7. El sistema responde de forma rápida a las peticiones que se le hacen. El acceso a los datos será de forma fluida.	
Observaciones:	

Fuente: Elaboración Propia

El paciente puede revisar el historial de sus citas médicas y agendar una nueva en cualquier momento.

Tabla 10: Fácil mantenimiento y mejora

Historia de Usuario	
Número: 9	Usuario: Doctor, Paciente
Nombre Historia: Fácil mantenimiento y mejora	
Prioridad en negocio: Media	Riesgo en negocio: Media
Programador Responsable: Josué Efraín Villamar Espinoza	
Descripción: El sistema permite la integración de nuevos módulos que solventen las necesidades futuras de la clínica.	
Observaciones:	

Fuente: Elaboración Propia

El sistema permitirá la posibilidad de adicionar nuevas funcionalidades sin alterar los módulos ya existentes, beneficiando a todos los destinatarios de esta aplicación.

2.4.3. Fase de Diseño

2.4.3.1. Diseño de la base de datos

Para el manejo de un alto volumen de datos, se utilizó el motor de base datos MongoDB, el cual es una infraestructura lógica, robusta y flexible [41] para la administración de datos sanitarios, dispone también de un conjunto de recursos que permiten asegurar la confidencialidad y disponibilidad de estos.

Una de las formas más efectivas de garantizar confidencialidad y disponibilidad en una base de datos es implementando algunas de las mejores prácticas de seguridad [42]. Entre ellas se encuentran: la autenticación y la replicación.

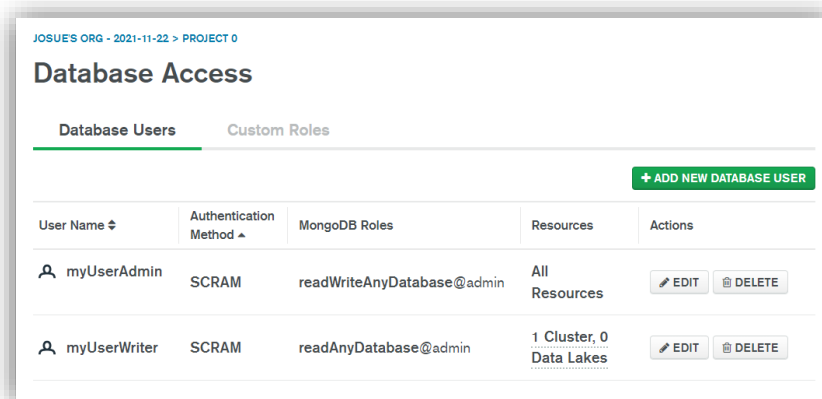
- Autenticación

La autenticación es el proceso de verificar la identidad de un cliente. En este caso se utilizaron 2 usuarios: myUserAdmin y myUserWriter.

El usuario **myUserAdmin** ejecuta el papel de administrador, es decir, tiene acceso de lectura y escritura en todo la base de datos, con la capacidad de modificar los privilegios de todos los usuarios existentes en esta base de datos

Es recomendable la utilización de un segundo usuario, debido a los altos privilegios y permisos de un usuario administrador. El usuario **myUserWriter** solamente posee permisos para lectura y escritura, careciendo de permisos para crear o eliminar usuarios, protegiendo de esta forma la información [43].

Figura 4: Autenticación



The screenshot shows the 'Database Access' page in a MongoDB management interface. It features a table with columns for 'User Name', 'Authentication Method', 'MongoDB Roles', 'Resources', and 'Actions'. Two users are listed: 'myUserAdmin' and 'myUserWriter'. The 'myUserAdmin' user has the role 'readWriteAnyDatabase@admin' and access to 'All Resources'. The 'myUserWriter' user has the role 'readAnyDatabase@admin' and access to '1 Cluster, 0 Data Lakes'. Both users have 'EDIT' and 'DELETE' actions available. A green button '+ ADD NEW DATABASE USER' is located at the top right of the table.

User Name	Authentication Method	MongoDB Roles	Resources	Actions
myUserAdmin	SCRAM	readWriteAnyDatabase@admin	All Resources	EDIT DELETE
myUserWriter	SCRAM	readAnyDatabase@admin	1 Cluster, 0 Data Lakes	EDIT DELETE

Fuente: Elaboración Propia

- BackUp

Es una copia de seguridad de los datos a partir de la cual se puede restaurar posteriormente en caso de pérdida [44]. Para poder elegir la estrategia más conveniente, es importante conocer las diferencias entre los backups completos, los diferenciales y los incrementales.

- **Completa:** Este tipo de copia se suele hacer una vez a la semana. Es la más importante porque almacena todos los datos de la empresa.
- **Diferencial:** Este tipo de copia almacena sólo aquellos datos que se hayan cambiado desde el momento en el que se realizó la copia completa.
- **Incremental:** Este tipo de copia guarda sólo aquellos datos que se hayan modificado desde el momento en el que se realizó la copia (ya sea completa o diferencial).

La función de copia de seguridad (backup) de MongoDB Atlas realiza una copia de seguridad incremental de los datos en un clúster específico y puede restaurar desde estas instantáneas o desde cualquier momento dentro de las últimas 24 horas

Figura 5: BackUp

JOSUES ORG - 2021-11-22 > PROJECT 0 > DATABASES

Cluster0 BACKUP TYPE M2/M5 Snapshots

Overview Real Time Metrics Collections Search Profiler Performance Advisor **Backup** Online Archive

Snapshots Restores & Downloads Backup Policy

LAST SNAPSHOT 02/24/22 - 04:49 AM NEXT ESTIMATED SNAPSHOT 02/25/22 - 04:48 AM TOTAL SNAPSHOT COUNT 4 TAKE SNAPSHOT NOW POINT IN TIME RESTORE

All Daily Snapshots

Completed Time	Expiration Time	Used Size	MongoDB Version	Actions
02/21/22 - 04:49 AM	03/01/22 - 04:48 AM	301.68 MB	5.0.6	RESTORE DOWNLOAD
02/22/22 - 04:49 AM	03/02/22 - 04:48 AM	301.69 MB	5.0.6	RESTORE DOWNLOAD
02/23/22 - 04:49 AM	03/03/22 - 04:48 AM	301.69 MB	5.0.6	RESTORE DOWNLOAD
02/24/22 - 04:49 AM	03/04/22 - 04:48 AM	301.68 MB	5.0.6	RESTORE DOWNLOAD

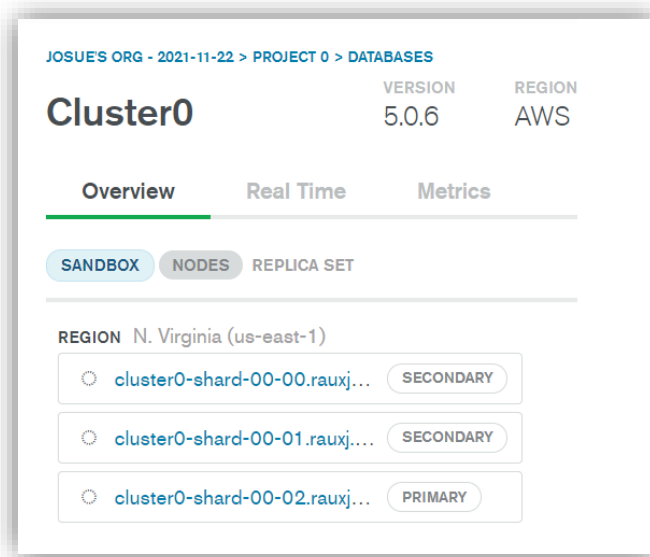
Fuente: Elaboración Propia

- Replicación

Es el proceso de crear un duplicado de un conjunto de datos en varios servidores.

El diseño de la base de datos permite crear un conjunto de 3 nodos de MongoDB los cuales proporcionan suficiente redundancia para resistir a la mayoría de errores y fallos en el sistema. Esta práctica es realmente importante, se presenta como una solución a los problemas de protección y fiabilidad del sistema

Figura 6: Replicación

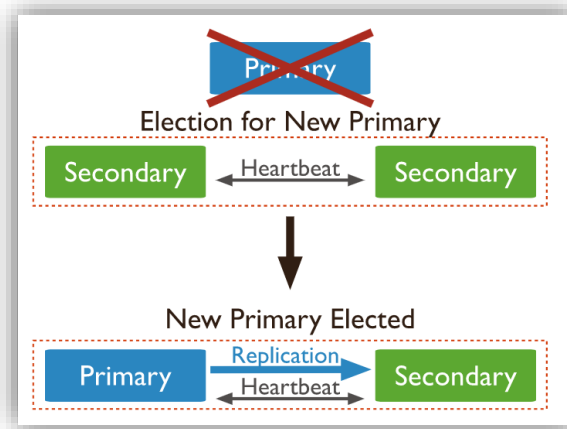


Fuente: Elaboración Propia

Es importante recalcar que la infraestructura de la base de datos utilizada en este trabajo posee 3 nodos: 1 nodo primario y 2 nodos secundarios.

- El nodo primario: Es el encargado de almacenar la información generada en la aplicación web.
- Los nodos secundarios: Son los encargados de salvaguardar una copia exacta de los datos para cumplir la función de nodo primario en caso de fallo.

Figura 7: Infraestructura de la base de datos



Fuente: [45]

A continuación, se presenta el esquema de la base de datos implementada [46] en formato tipo JSON en Mongo Atlas, este es un servicio global de base de datos de documentos en la nube para las aplicaciones que exigen flexibilidad y rendimiento.

Figura 8: Esquema de la base de datos

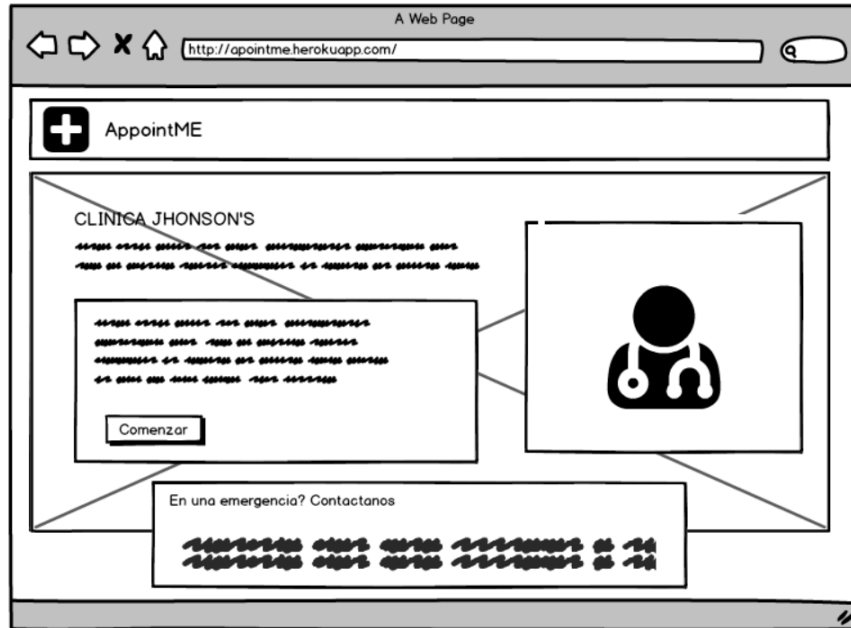
```
{
  "_id": {
    "$oid": "62115a4c2ba1d332d08fae29"
  },
  "doctorInfo": {
    "accreditations": [],
    "education": [
      "UTMACH"
    ],
    "languagesSpoken": [
      "Español"
    ],
    "licence": "MIT",
    "specialtyField": "Cardiología",
    "subSpecialtyField": "Cardiología Intervencionista",
    "yearsExperience": 10,
    "rating": 1
  },
  "clientInfo": {
    "medicalHistory": [],
    "allergies": [],
    "medication": []
  },
  "firstName": "usuario Doctor",
  "lastName": "de prueba",
  "title": "Dr.",
  "sex": "masculino",
  "dateOfBirth": "1999-01-01",
  "phoneNumber": "+593987466964",
  "email": "usuario.doctor.001@gmail.com",
  "password": "$2a$08$tJxwJ/vGSqAeM5ME1HNS2uI/JHEQP3GGt0qVqLGSvJQN64QIB.fxu",
  "isDoctor": true,
  "address": {
    "principalStreet": "Guayacanes",
    "street": "Rocafuerte",
    "city": "Machala",
    "state": "El Oro",
    "country": "Ecuador",
    "postcode": 70206
  },
  "tokens": [
    {
      "_id": {
        "$oid": "62115a4c2ba1d332d08fae2a"
      },
      "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiIiImJjExNWE0YzJiYTFkMzMyZDA4ZmF1MjkiLCJpYXQiOiJlE2NDUzMDQzOTYsImV4cCI6IjE2MTY0NTM0MDM5Nn0.8PIT-Z9MJBw18hk7J73QMRawkQ1YTA81tbhuU92dUbo"
    }
  ]
}
```

Fuente: Elaboración Propia

2.4.3.2. Diseño de interfaces

- Portal Web

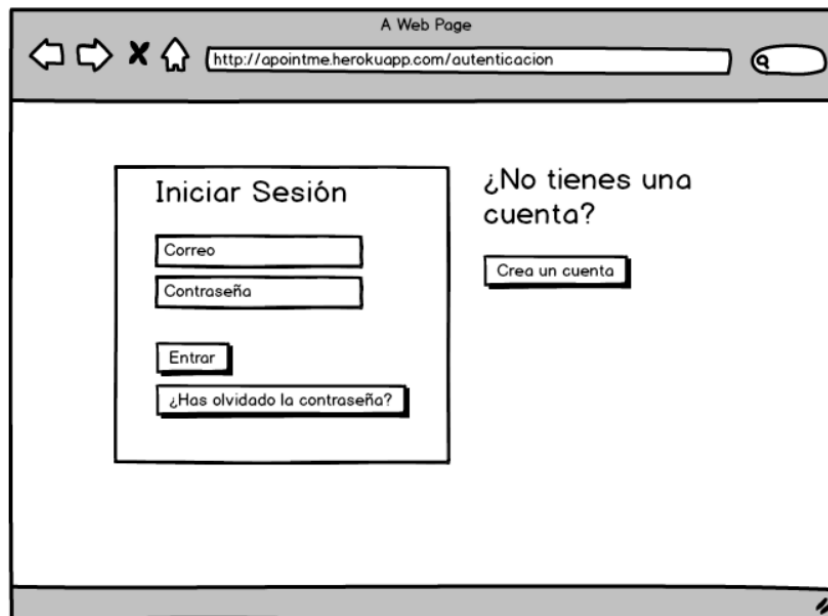
Figura 9: Diseño de Interfaz - Portal Web



Fuente: Elaboración Propia

- Autenticación de usuarios

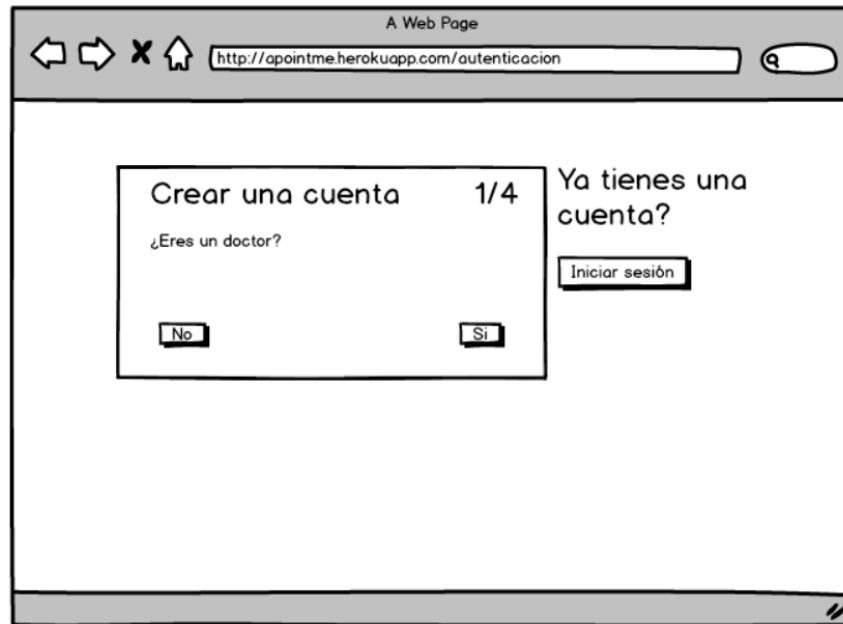
Figura 10: Diseño de Interfaz - Autenticación de usuarios



Fuente: Elaboración Propia

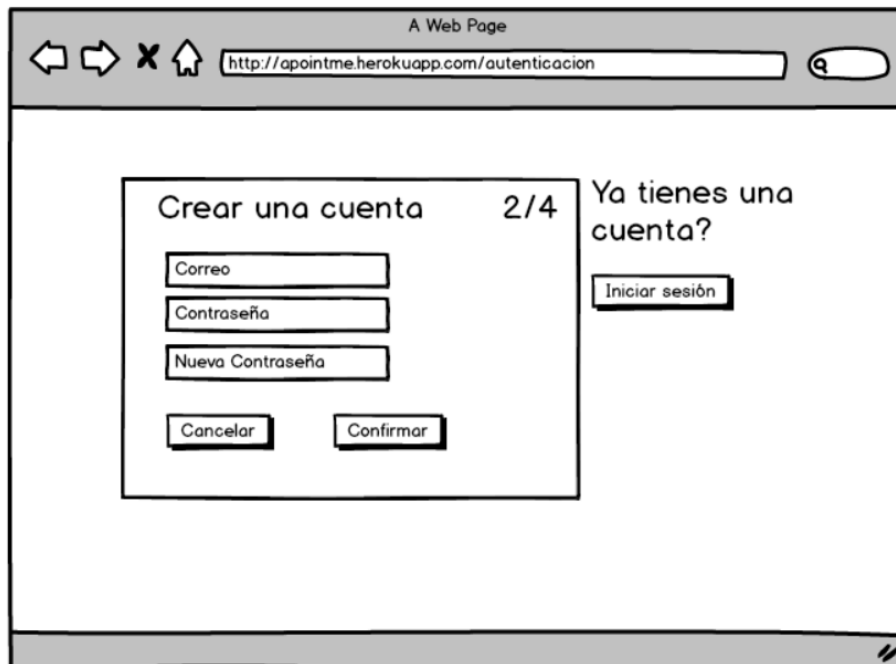
- Registro de usuario

Figura 11: Diseño de Interfaz - Registro de usuario I



Fuente: Elaboración Propia

Figura 12: Diseño de Interfaz - Registro de usuario II



Fuente: Elaboración Propia

Figura 13: Diseño de Interfaz - Registro de usuario III

A Web Page
http://apointme.herokuapp.com/autenticacion

Crear una cuenta 3/4

Ya tienes una cuenta?
[Iniciar sesión](#)

IDENTIFICACIÓN	DIRECCIÓN
NOMBRES	CALLE PRINCIPAL
APELLIDOS	CALLE SECUNDARIA
SEXO	CIUDAD
FECHA DE NACIMIENTO 10 / 12 / 1988	PROVINCIA
CELULAR 0999999999	PAIS
	POSTAL 70203

[ANTERIOR](#) [SIGUIENTE](#)

Fuente: Elaboración Propia

- Registro de pacientes

Figura 14: Diseño de Interfaz - Registro de pacientes

A Web Page
http://apointme.herokuapp.com/autenticacion

Crear una cuenta 4/4

Ya tienes una cuenta?
[Iniciar sesión](#)

CONDICIONES MÉDICAS ACTUALES Enumere cualquier condición médica
CONDICION
10 / 12 / 2021
COMENTARIO
+
ALERGIAS
TIPO
GRAVEDAD 3
+
MEDICACIONES
MEDICAMENTO
DOSIS
FABRICANTE
+
TIPO DE SANGRE
A+ A- B+
PESO 80 (kg)

[ANTERIOR](#) [ENVIAR](#)

Fuente: Elaboración Propia

- Registro de doctores

Figura 15: Diseño de Interfaz - Registro de doctores

A Web Page
http://apointme.herokuapp.com/autenticacion

Crear una cuenta 4/4

Ya tienes una cuenta?

LICENCIA MÉDICA
10 / 12 / 2021
ACREDITACIONES +
ESPECIALIDAD
SUBESPECIALIDAD
EDUCACIÓN +
AÑOS DE EXPERIENCIA 3
IDIOMA +

Fuente: Elaboración Propia

- Recuperación de cuenta

Figura 16: Diseño de Interfaz - Recuperación de cuenta I

A Web Page
http://apointme.herokuapp.com/contrasena

Recupera tu cuenta

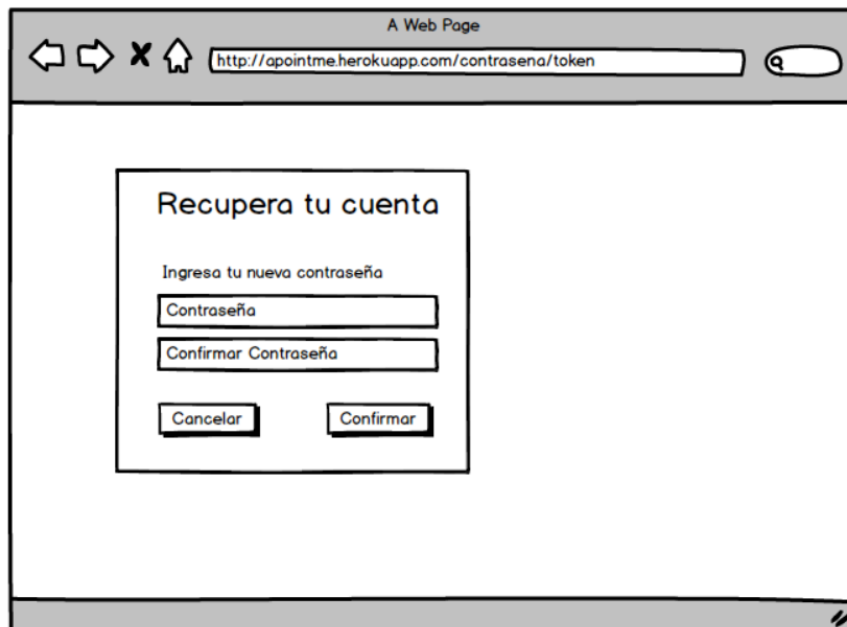
Introduce tu correo electrónico para buscar tu cuenta

Correo

¿No tienes una cuenta?

Fuente: Elaboración Propia

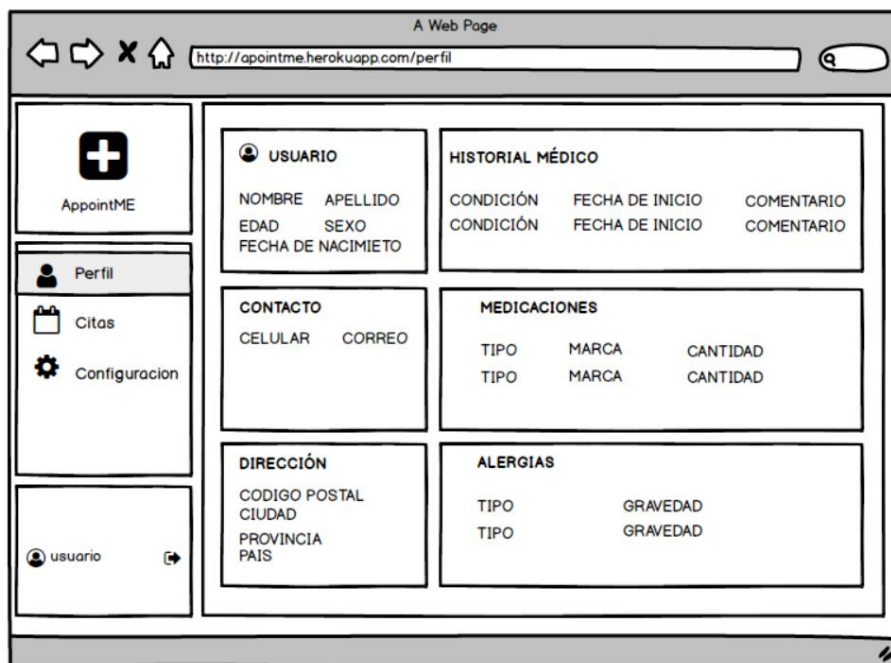
Figura 17: Diseño de Interfaz - Recuperación de cuenta II



Fuente: Elaboración Propia

- Perfil del Paciente

Figura 18: Diseño de Interfaz - Perfil del paciente



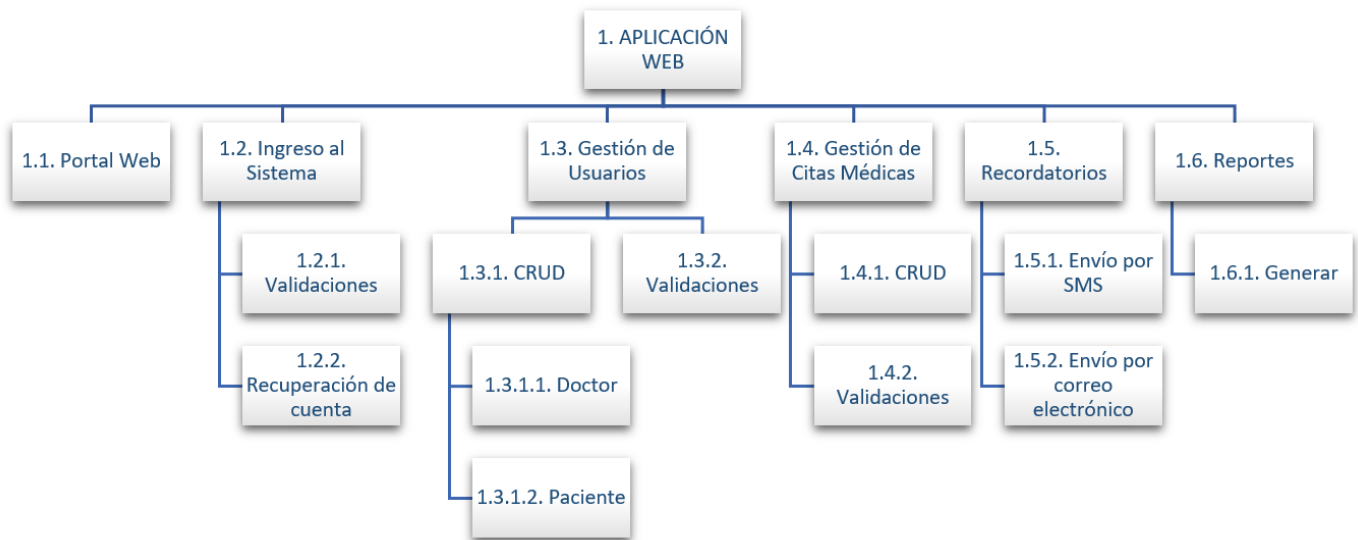
Fuente: Elaboración Propia

2.5 EJECUCIÓN Y/O ENSAMBLAJE DEL PROTOTIPO

2.5.1 Estructura de desglose de trabajo (EDT)

Como parte del presente proyecto se establece una Estructura de desglose de trabajo (EDT), la cual es una herramienta y fundamental eficaz en el control y administración de planes [47], además se utiliza para identificar los módulos de la aplicación web y las actividades necesarias para la implementación de los mismos.

Figura 19: Estructura de desglose de trabajo (EDT)

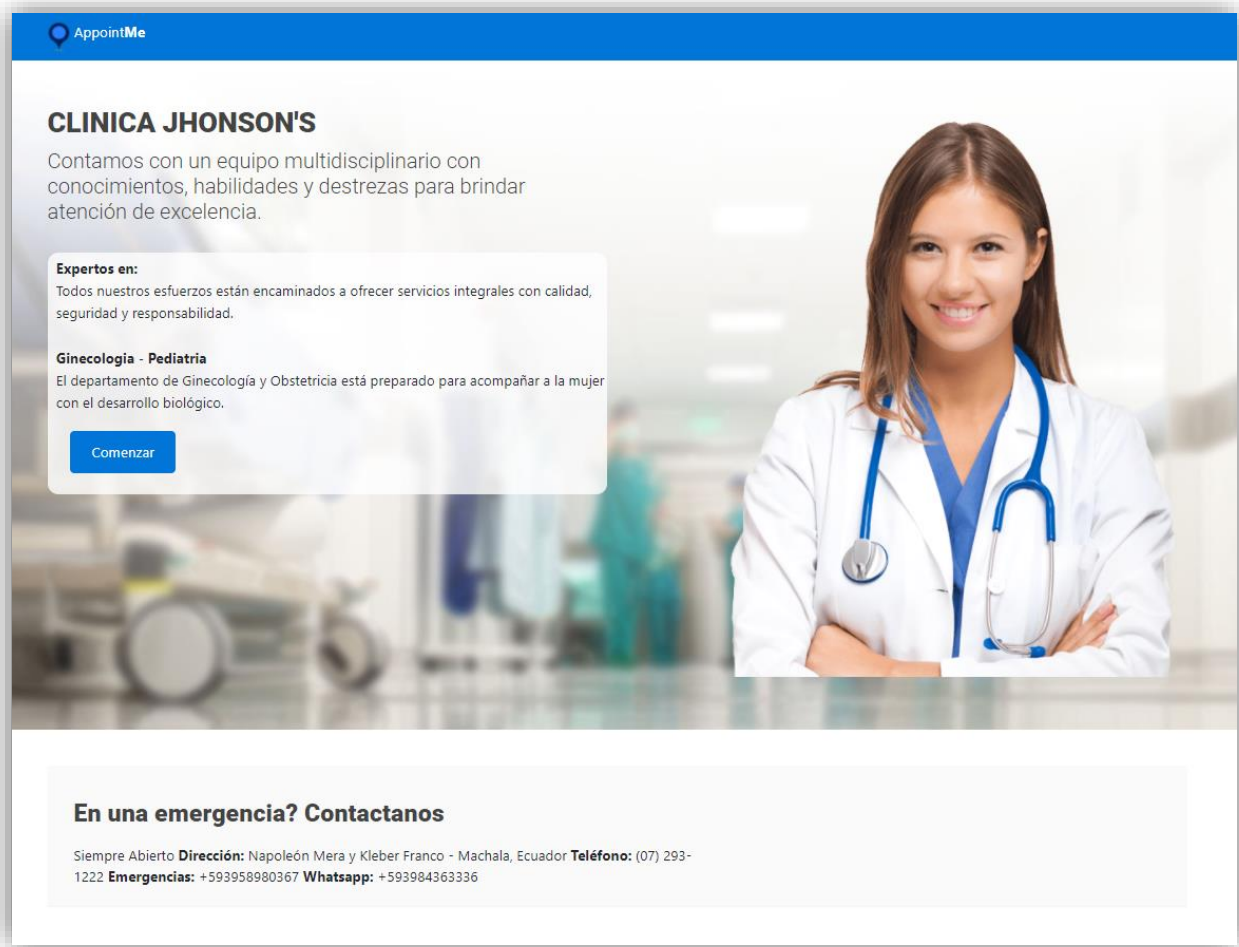


Fuente: Elaboración Propia

2.5.2. Portal Web

El Portal Web es una página de bienvenida la cual contiene información de la Clínica Johnson sobre las especialidades médicas que ellos ofrecen, dirección y números telefónicos de contacto.

Figura 20: Ejecución del prototipo - Portal Web



Fuente: Elaboración Propia

2.5.3. Inicio de sesión.

La página de inicio de sesión sirve para la autenticación del usuario, para esto se solicita un correo electrónico único y una contraseña segura; si se ingresa un correo electrónico que no existe el sistema genera una advertencia; la contraseña debe contener mínimo 6 dígitos.

Figura 21: Ejecución del prototipo - Inicio de sesión

The screenshot shows a login interface with the following elements:

- Header: "Autenticación" with a horizontal line.
- Section: "Iniciar sesión" (Login).
- Form fields: "Correo" (Email) and "Contraseña" (Password).
- Buttons: "Entrar →]" (Login) and "¿Has olvidado la contraseña?" (Forgot password?).
- Link: "¿No tienes una cuenta?" (Don't you have an account?) with a "Crea una cuenta" (Create an account) button.

Fuente: Elaboración Propia

2.5.4. Recuperación de cuenta

En caso de olvidar la contraseña la página Recuperación de cuenta te permite solicitar un cambio de contraseña, para esto solicita un correo electrónico (correo que debe pertenecer a un usuario del sistema) mismo que recibirá un enlace para reasignar una nueva contraseña

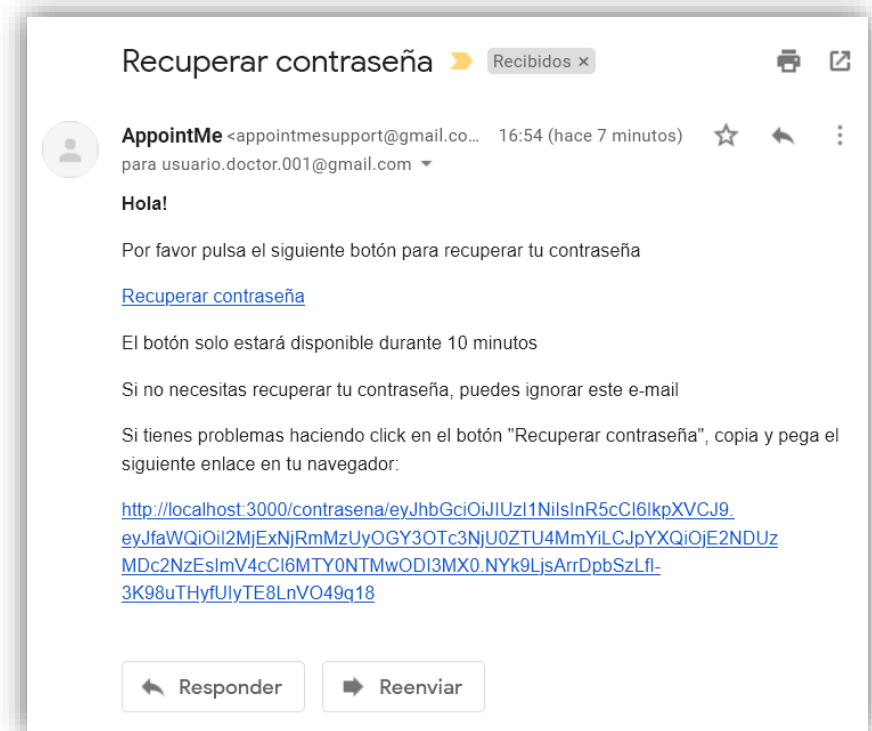
Figura 22: Ejecución del prototipo - Recuperación de cuenta I

The screenshot shows a password recovery interface with the following elements:

- Header: "Contraseña olvidada" with a horizontal line.
- Section: "Recupera tu cuenta" (Recover your account).
- Text: "Introduce tu correo electrónico para buscar tu cuenta" (Enter your email to find your account).
- Form field: "Correo" (Email) with the value "usuario.doctor.001@gmail.com".
- Buttons: "← Cancelar" (Cancel) and "Buscar →]" (Search).

Fuente: Elaboración Propia

Figura 23: Ejecución del prototipo - Recuperación de cuenta II



Fuente: Elaboración Propia

2.5.5 Registro de usuarios

Para crear una cuenta existen 4 formularios para completar:

- Crear una cuenta (Doctores y Pacientes)
- Crear cuenta: Información básica (Doctores y Pacientes)
- Crear cuenta: Licencia médica (Doctores)
- Crear cuenta: Condiciones médicas (Pacientes)

En la **Figura 24** se puede observar el formulario **Crear una cuenta** para crear una cuenta, dentro se encuentran los campos de correo, contraseña y confirmar contraseña. La misma contraseña debe ser escrita en 2 campos diferentes para que esta sea confirmada, caso contrario el sistema genera una advertencia

Figura 24: Ejecución del prototipo - Crear una cuenta

The image shows a mobile application interface for user authentication. At the top, the word "Autenticación" is displayed with a blue underline. Below this, there is a card titled "Crear una cuenta" with a progress indicator "2/4". The card contains three input fields: "Correo" with the value "usuario.doctor.001@gmail.", "Contraseña" with masked characters and a green checkmark, and "Confirmar contraseña" also with masked characters and a green checkmark. At the bottom of the card are two buttons: "Anterior" (dark grey) and "Siguiente" (blue). To the right of the card, the text "¿Ya tienes una cuenta?" is followed by a blue button labeled "Iniciar sesión".

Fuente: Elaboración Propia

El formulario **Crear cuenta: Información básica** solicita información básica del usuario, todos los campos son necesarios, sin embargo, si es Paciente es muy importante el número celular para recibir los recordatorios.

Figura 25: Ejecución del prototipo - Crear cuenta: Información básica

Autenticación

3/4

Crear una cuenta

¿Ya tienes una cuenta?

Iniciar sesión

Información Básica

Título
Dr.

Primer nombre
usuario Doctor

Apellido
de prueba

Sexo
masculino

Fecha de nacimiento
01/01/1999

Celular
+593987466964

Dirección

Calle Principal
Guayacanes

Calle
Rocafuerte

Ciudad
Machala

Provincia
El Oro

País
Ecuador

Código postal
70206

← Anterior Siguiente →

Fuente: Elaboración Propia

Todos los campos solicitados en el formulario **Crear cuenta: Licencia médica** deben ser llenados solamente si el usuario es Doctor. Los campos Educación e Idioma son campos dinámicos, ya que permiten el ingreso de varios valores al dar clic en el botón adjunto.

Figura 26: Ejecución del prototipo - Crear cuenta: Licencia médica

The image shows a mobile application interface for creating an account. The main heading is 'Autenticación'. On the left, there is a card titled 'Crear una cuenta' with a '4/4' indicator. The card contains several input fields: 'Licencia médica' with the value 'MIT', 'Especialidad' with 'Cardiología', 'Subespecialidad' with 'Cardiología Intervencionist', 'Educación' with 'UTMACH' and a plus sign, 'Años de experiencia' with '# 10', and 'Idioma' with 'Español' and a plus sign. At the bottom of the card are two buttons: 'Anterior' (left arrow) and 'Enviar' (right arrow with checkmark). To the right of the card, the text '¿Ya tienes una cuenta?' is displayed above a blue button labeled 'Iniciar sesión'.

Fuente: Elaboración Propia

El formulario **Crear cuenta: Condiciones médicas** será utilizado solamente si el usuario es Paciente. Al final se encuentra el botón Enviar que culmina con el proceso de creación de cuenta.

Figura 27: Ejecución del prototipo - Crear cuenta: Condiciones médicas

Autenticación

Crear una cuenta 4/4

Pacientes:
Condiciones medicas

Enumere cualquier condición médica actual y su fecha de inicio aproximada.

Condición
Presión Alta

Fecha de inicio
03/05/2021

Comentario
Por vejez

+

Alergias

Alergia
Alergia al polvo

Gravedad
1

+

Medicamentos

Medicamento
Paracetamol

Dosis (mg)
200

Fabricante
Ecuador

+

Tipo de Sangre & Peso

Tipo de sangre
A+

Peso (kg)
82

← Anterior Enviar ✓

¿Ya tienes una cuenta?

Iniciar sesión

Fuente: Elaboración Propia

2.5.6 Configuración

En la **Figura 28** se puede observar el formulario Configuración, este permite modificar la información de los usuarios, actualizar el avatar, cambiar la contraseña actual y eliminar la cuenta de forma permanente.

Figura 28: Ejecución del prototipo - Configuración de la cuenta

AppointMe

Perfil

Mensajes

Configuración

Pacientes

Citas

Reportes

usuario Doctor

Configuración

Configuración de tu cuenta

Eliminar tu cuenta ✕ Cerrar sesión ↗

Actualiza tu avatar

Elegir archivo No se eligió ningún archivo

Subir avatar ↗

Detalles del inicio de sesión

Contraseña

Confirmar contraseña

Información Básica

Primer nombre usuario Doctor

Apellido de prueba

Sexo masculino

Fecha de nacimiento 01/01/1999

Celular +593987466964

Dirección

Calle Principal # Guayacanes

Calle Rocafuerte

Ciudad Machala

Provincia El Oro

País Ecuador

Código postal 70206

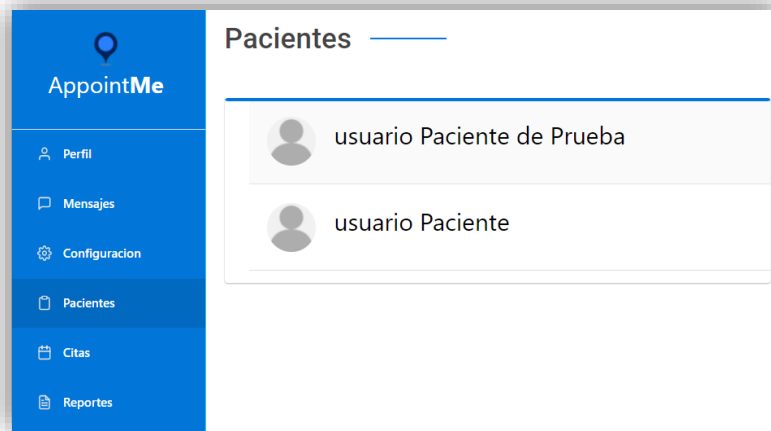
Actualizar ✓

Fuente: Elaboración Propia

2.5.7 Pacientes

En el sistema el Doctor puede visualizar una lista de sus pacientes y al dar clic en un nombre determinado se muestra el perfil de paciente seleccionado

Figura 29: Ejecución del prototipo - Pacientes

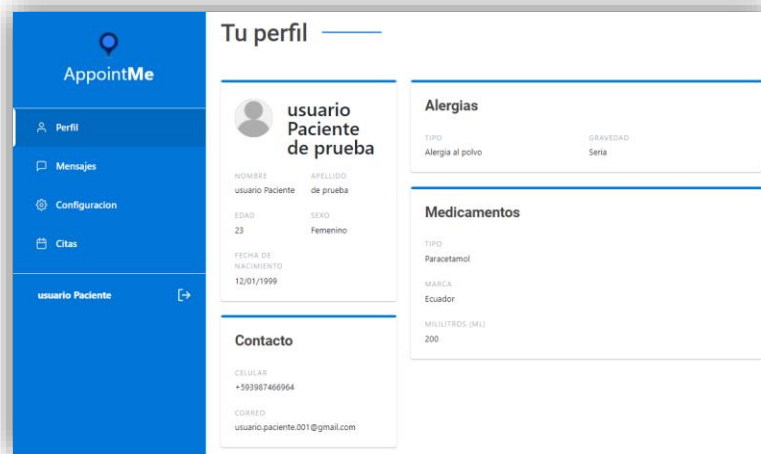


Fuente: Elaboración Propia

2.5.8 Perfil

En la Figura 30 se observa la información referente a la cuenta del paciente: alergias, medicamentos, información de contacto y su dirección.

Figura 30: Ejecución del prototipo - Perfil

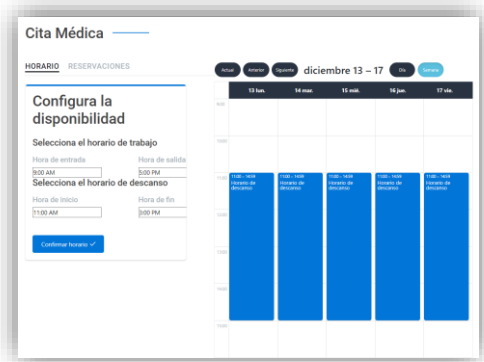


Fuente: Elaboración Propia

2.5.9 Cita médica (Doctor)

En este formulario el Doctor configura su disponibilidad definiendo un horario de trabajo con hora de entrada, hora de salida y hora de descanso, este proceso es necesario para empezar a recibir solicitudes de citas médicas; una vez elegido el horario de trabajo se despliega un calendario adaptado con las horas disponibles de cada doctor.

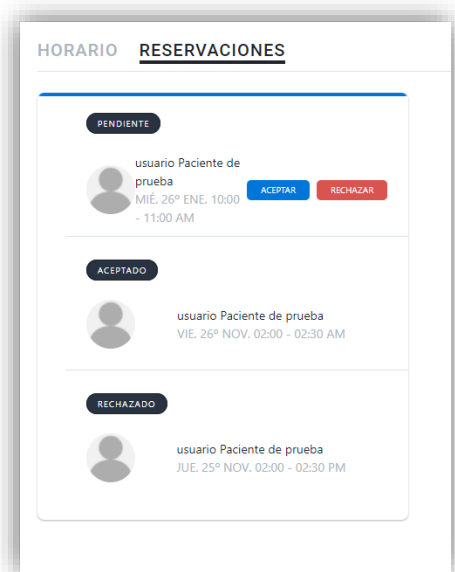
Figura 31: Ejecución del prototipo - Cita médica (Doctor)



Fuente: Elaboración Propia

En la sección de reservaciones el Doctor encontrará todas las solicitudes de citas las cuales, el podrá aceptar o rechazar.

Figura 32: Ejecución del prototipo – Reservaciones / Doctor

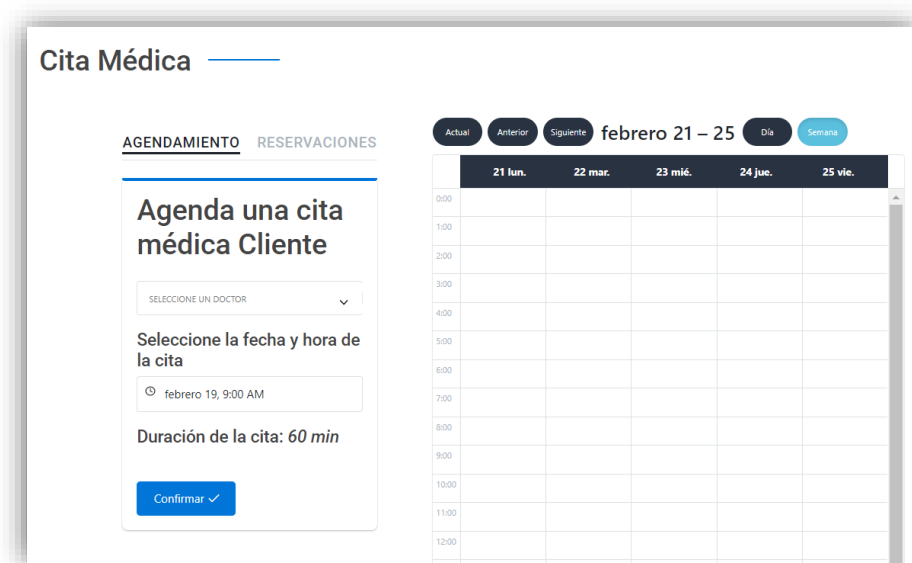


Fuente: Elaboración Propia

2.5.10 Cita médica (Paciente)

En la sección de agendamiento el usuario al elegir el Doctor con la especialización que necesita encontrará el calendario con la disponibilidad de éste. A continuación, podrá seleccionar la fecha y hora de la cita.

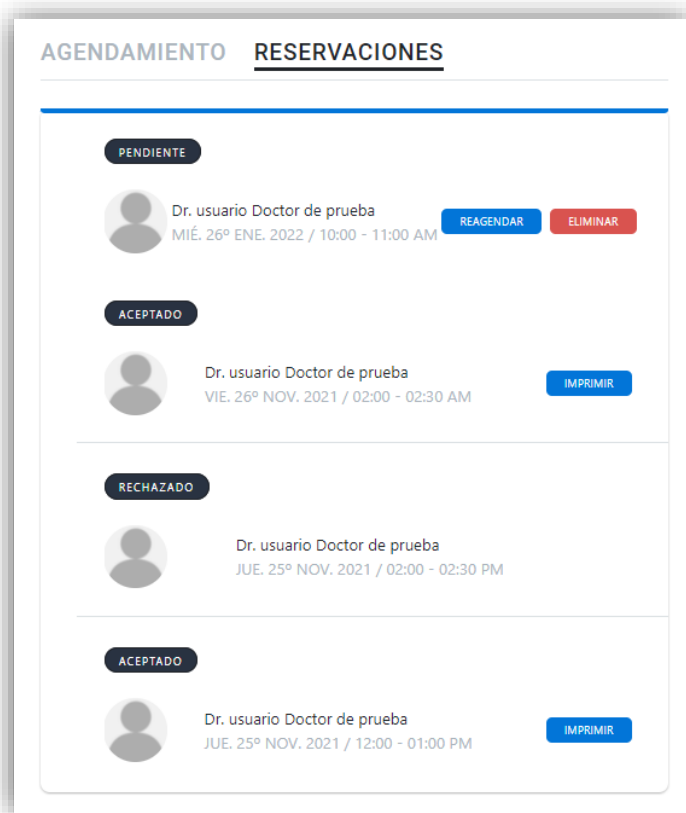
Figura 33: Ejecución del prototipo - Cita médica (Paciente)



Fuente: Elaboración Propia

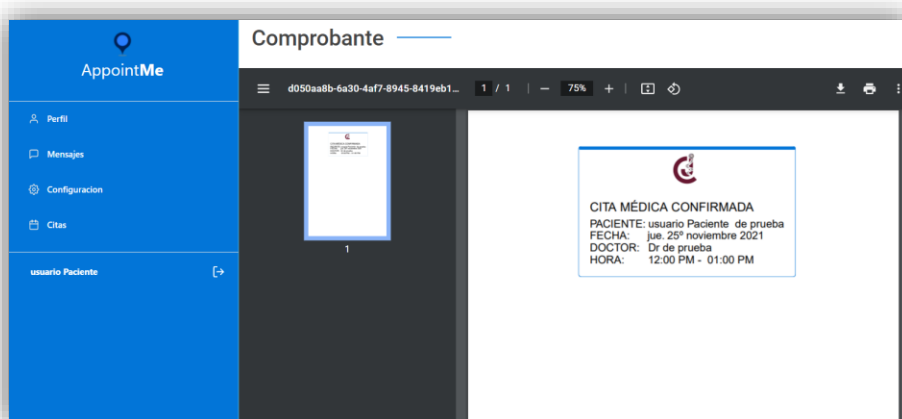
En la sección de reservaciones el Paciente encontrará tres tipos de citas: pendientes, aceptadas y rechazadas. Las citas pendientes mostrarán las opciones de Reagendar o Eliminar, las citas rechazadas y, finalmente se encontrará las citas confirmadas con la opción de Imprimir

Figura 34: Ejecución del prototipo – Reservaciones / Paciente



Fuente: Elaboración Propia

Figura 35: Ejecución del prototipo – Comprobante



Fuente: Elaboración Propia

3. CAPÍTULO III. EVALUACIÓN DEL PROTOTIPO

3.1 Plan de evaluación

La fase de evaluación en el desarrollo de software se realiza para asegurar un nivel de calidad óptimo y reducir las posibilidades de fracaso del proyecto [48], esta fase es realmente importante ya que ayuda a descubrir posibles errores que afecten la experiencia del usuario [49].

3.2 Resultados de la evaluación

Para obtener la evaluación de calidad fue empleada la norma ISO/IEC 9126 [49], para realizar esta medición las características son divididas en diferentes categorías: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad [50], [51], tal como se presenta en la tabla.

Tabla 11: Resultados de la evaluación

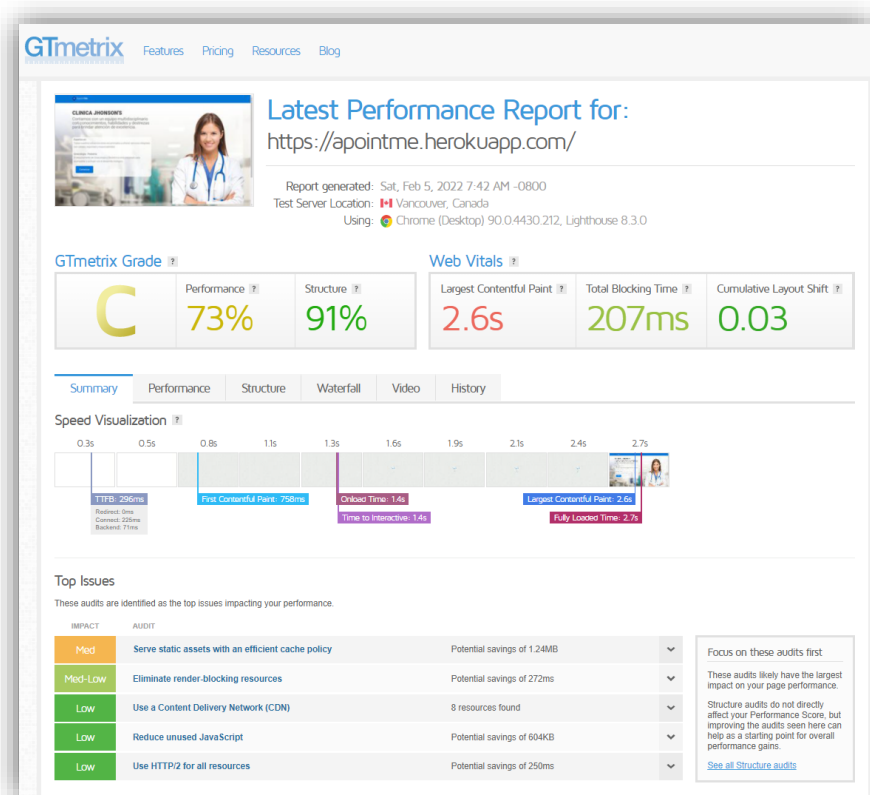
#	Herramienta	Parámetros					
	Nombre	Accesibilidad	Funcionalidad	Eficiencia	Usabilidad	Portabilidad	Posicionamiento
1	Validator html		X				
2	Validator (HTML y CSS juntos)		X				
3	GtMetrix	X		X			
4	Website grader					X	
5	Metric Spot				X		X
6	Seo Site Checkup						X

Fuente: Elaboración Propia

3.2.1 GtMetrix

Este instrumento de evaluación determina la eficiencia y accesibilidad del sistema [52], en donde se observan resultados aceptables con una calificación C (73%) en el rendimiento y una calificación A (91%) en la estructura.

Figura 36: Evaluación - GtMetrix

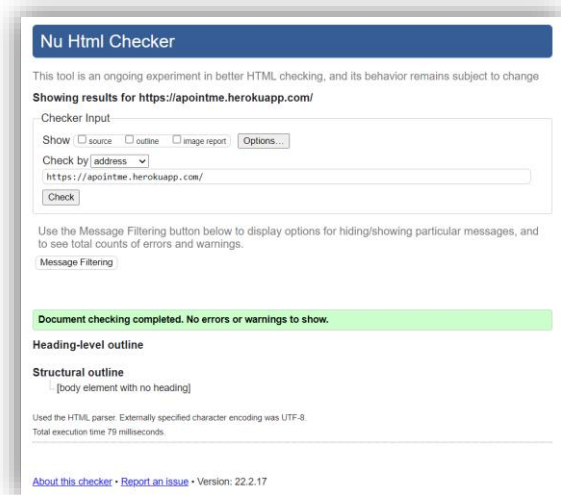


Fuente: Elaboración Propia

3.2.2 Validator HTML

Los resultados obtenidos al utilizar esta herramienta fueron favorables puesto que no presentó ningún tipo de error o advertencia como se observa en la Figura 37.

Figura 37: Evaluación - Validator html

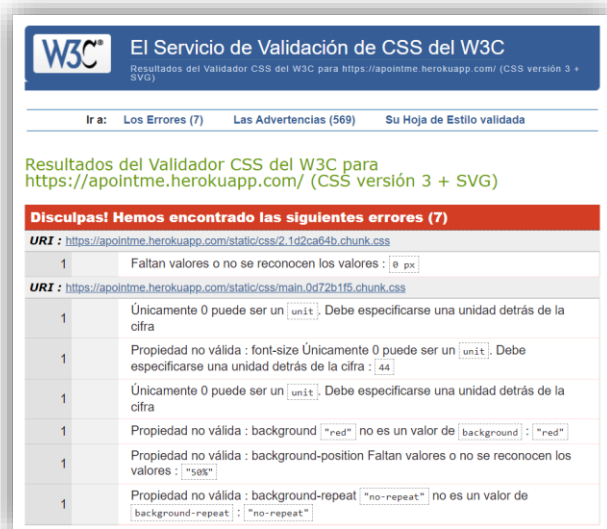


Fuente: Elaboración Propia

3.2.3 Validator (HTML y CSS)

Los archivos CSS que se encuentran dentro de la aplicación al ser evaluados con esta herramienta presentaron resultados aceptables, puesto que encontraron 7 errores y 569 advertencias, estas se deben a que poseen propiedades no válidas y estilos sin definir originarias de librerías externas [53], estas librerías ofrecen la posibilidad de mantener un diseño responsivo , [54] (Anexo 2).

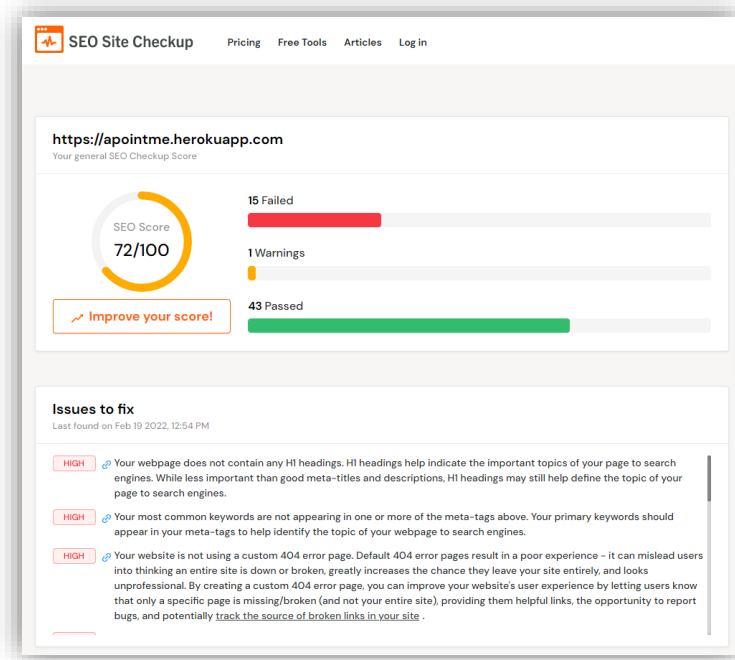
Figura 38: Evaluación - Validator (html y css juntos)



3.2.4 Seo Site Checkup

Esta herramienta evalúa el posicionamiento y visibilidad de la aplicación web, donde se cumplen 43 recomendaciones de 59 aspectos evaluados con una calificación aceptable de 72/100.

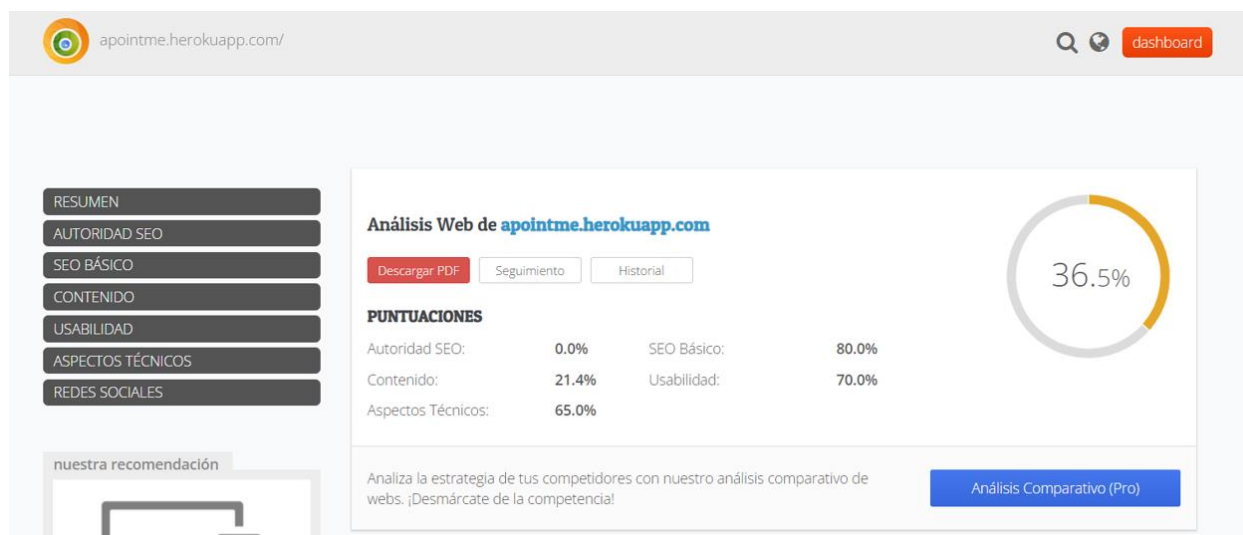
Figura 39: Evaluación - Seo Site Checkup



3.2.5 Metric Spot

Esta herramienta evalúa la usabilidad y al igual que Seo Site Checkup evalúa también el posicionamiento. Entre otras medidas señala un SEO con una calificación del 80% y la usabilidad con una calificación del 70%.

Figura 40: Evaluación - Metric Spot

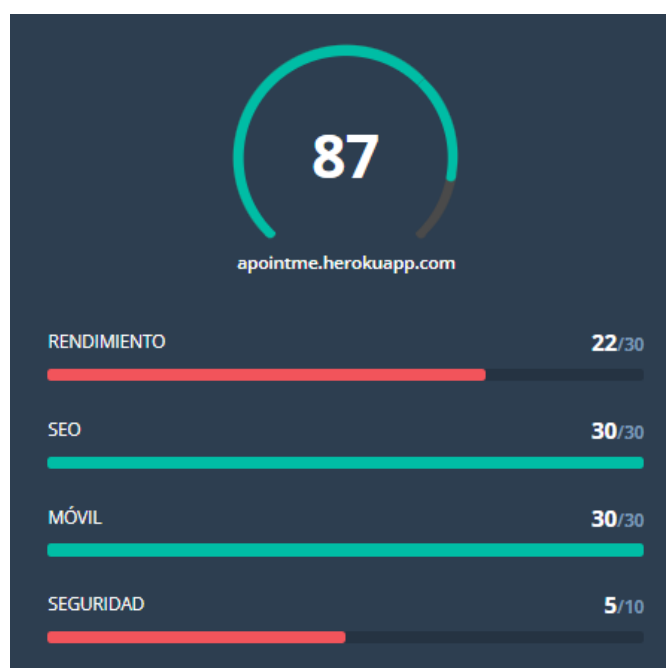


Fuente: Elaboración Propia

3.2.6 Website grader

Esta herramienta permite evaluar el rendimiento, SEO, adaptabilidad y seguridad otorgando una calificación favorable del 87%, cabe recalcar una calificación perfecta de 30/30 en el parámetro de portabilidad (Anexo 3).

Figura 41: Evaluación - Website grader



Fuente: Elaboración Propia

CONCLUSIONES

- La utilización de una metodología ágil como Extreme Programming para el desarrollo de la aplicación permitió asegurar la calidad, la optimización del tiempo y reducir probabilidades de fracaso al mantener un contacto permanente con el beneficiario del programa.
- Se desarrolló una aplicación web que permite la correcta gestión de las citas médicas en la Clínica Johnson utilizando una arquitectura de hardware que permite asegurar la información sensible de los doctores que elaboran en la clínica y los pacientes que solicitan citas médicas en este establecimiento.
- Una vez finalizado la implementación de la aplicación y realizado el proceso de evaluación mediante distintas herramientas web se obtuvieron calificaciones con resultados favorable, indicando que el sistema tiene un nivel de calidad aceptable y está optimizado para cualquier dispositivo.
- La información obtenida de fuentes bibliográficas confiables permitió que este trabajo cuente con la fiabilidad y seguridad de un documento fidedigno.

RECOMENDACIONES

- Mantener una comunicación constante con los stakeholders de la aplicación, esto evitaría cambios excesivos durante el desarrollo del sistema, optimizando valiosos recursos como tiempo y dinero.
- Realizar pruebas unitarias y de integración que verifiquen el funcionamiento de la aplicación, es recomendable también ejecutar pruebas que generen datos aleatorios en la base de datos comprobando de esta manera el rendimiento de esta al existir un número bastante considerable de registros.
- Dependiendo del tamaño del sistema se recomienda el uso de frameworks en vista de que se basan en arquitecturas y estándares que facilitan la escalabilidad y mantenimiento del código.
- Utilizar constantemente las herramientas de evaluación de calidad que permiten corregir los errores para mejorar los diferentes aspectos que aseguran una buena experiencia de usuario.

BIBLIOGRAFÍA

- [1] G. C. Tejada Estrada, J. M. Cruz Montero, Y. C. Uribe Hernandez, y J. J. Rios Herrera, «Innovación tecnológica: Reflexiones teóricas», *Rev. Venez. Gerenc.*, vol. 24, n.º 85, 2019, Accedido: 22 de enero de 2022. [En línea]. Disponible en: <https://www.redalyc.org/journal/290/29058864011/>
- [2] Y. C. Blanco y R. G. Hernández, «Aplicación web para gestionar información sobre personas discapacitadas», *Rev. Inf. Científica*, vol. 97, n.º 4, pp. 710-721, 2018.
- [3] L. O. Señalín Morales, R. L. Olaya Cum, y J. N. Herrera Peña, «Gestión presupuestaria y planificación empresarial: algunas reflexiones», *Rev. Venez. Gerenc.*, vol. 25, n.º 92, pp. 1704-1715, 2020.
- [4] C. Pla y C. Niqui, «Modelo de negocio en medios de comunicación de proximidad y creación de un prototipo comunicativo digital basado en Wordpress», p. 69.
- [5] M. V. Raso, «Regulación y realidad de las bases de datos médicos desde las perspectivas de la protección de datos y la salud», <http://purl.org/dc/dcmitype/Text>, Universidad Complutense de Madrid, 2016. Accedido: 26 de enero de 2022. [En línea]. Disponible en: <https://dialnet.unirioja.es/servlet/tesis?codigo=149288>
- [6] Asamblea Nacional de la República del Ecuador, *Ley Organica de Protección de Datos Personales*. 2021, p. 70. Accedido: 26 de enero de 2022. [En línea]. Disponible en: <https://www.telecomunicaciones.gob.ec/wp-content/uploads/2021/06/Ley-Organica-de-Datos-Personales.pdf>
- [7] Ministerio de Salud Pública, *Reglamento de Información confidencial en Sistema Nacional de Salud*. 2015. Accedido: 26 de enero de 2022. [En línea]. Disponible en: <http://instituciones.msp.gob.ec/cz6/images/lotaip/Enero2015/Acuerdo%20Ministerial%205216.pdf>
- [8] J. J. Valdivia Caballero, «Modelo de procesos para el desarrollo del front-end de aplicaciones web», *Interfases*, n.º 9, pp. 187-208, 2016.
- [9] O. Agudelo, F. R. Sanabria, y S. V. Rodríguez, «Evaluación de una Arquitectura de Software», *Prospectiva*, vol. 19, n.º 2, Art. n.º 2, jul. 2021, doi: 10.15665/rp.v19i2.2636.
- [10] H. J. C. Benavides, «Una aproximación dirigida por modelos para la caracterización de la capa de presentación web de aplicaciones empresariales», <http://purl.org/dc/dcmitype/Text>, Universidad Complutense de Madrid, 2017. Accedido: 27 de enero de 2022. [En línea]. Disponible en: <https://dialnet.unirioja.es/servlet/tesis?codigo=151551>
- [11] «React – Una biblioteca de JavaScript para construir interfaces de usuario». <https://es.reactjs.org/> (accedido 27 de enero de 2022).
- [12] B. M. Coti Colop, «Reglas del Negocio en Arquitectura Tres Capas», Universidad de San Carlos de Guatemala, Guatemala, 2003. Accedido: 20 de febrero de 2022. [En línea]. Disponible en: http://biblioteca.usac.edu.gt/tesis/08/08_0170_CS.pdf
- [13] H. S. Krohn, «Programación de buscadores en JavaScript para diccionarios digitales», *Cuad. Lingüíst. Hispánica*, n.º 34, pp. 109-130, 2019.
- [14] V. Subramaniam, *Rediscovering JavaScript*. 2018. Accedido: 20 de febrero de 2022. [En línea]. Disponible en: <https://pragprog.com/titles/ves6/rediscovering-javascript>
- [15] M. P. Ascue y J. L. M. Aroni, «Integración de procesos de negocio aplicando la arquitectura orientada a servicios (SOA)», *Interfases*, n.º 10, pp. 93-121, 2017.

- [16] A. C. Romero, J. González, y M. C. Cuervo, «Utilidad y funcionamiento de las bases de datos NoSQL», *Fac. Ing.*, vol. 21, n.º 33, pp. 21-32, 2012.
- [17] J. C. S. Núñez, «Implementación de un modelo de desarrollo de software seguro», <http://purl.org/dc/dcmitype/Text>, Universidad de Extremadura, 2021. Accedido: 27 de enero de 2022. [En línea]. Disponible en: <https://dialnet.unirioja.es/servlet/tesis?codigo=289165>
- [18] J. Z. Gamboa, «Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software», *INNOVA Res. J.*, vol. 3, n.º 10, pp. 20-33, 2018.
- [19] J. R. M. Ríos, M. P. Z. Ordóñez, M. J. C. Segarra, y F. G. G. Zerda, «Estado del arte: Metodologías de desarrollo en aplicaciones web», *3c Tecnol. Glosas Innov. Apl. Pyme*, vol. 6, n.º 3, pp. 54-71, 2017.
- [20] F. Vera-Rivera, «A development process of enterprise applications with microservices», *J. Phys. Conf. Ser.*, vol. 1126, p. 012017, nov. 2018, doi: 10.1088/1742-6596/1126/1/012017.
- [21] J. C. Pereira y R. de F. S. M. Russo, «Design Thinking Integrated in Agile Software Development: A Systematic Literature Review», *Procedia Comput. Sci.*, vol. 138, pp. 775-782, ene. 2018, doi: 10.1016/j.procs.2018.10.101.
- [22] J. C. Vargas-Fandiño, J. J. Sandoval-Ramírez, y F. Vera-Rivera, «Implementación de un repositorio para el catálogo, búsqueda y uso de componentes software reutilizables en el desarrollo de aplicaciones web», *Rev. UIS Ing.*, vol. 19, n.º 2, Art. n.º 2, mar. 2020, doi: 10.18273/revuin.v19n2-2020002.
- [23] Y. Molina Hernández, A. Granda Dihigo, y A. Velázquez Cintra, «Los requisitos no funcionales de software. Una estrategia para su desarrollo en el Centro de Informática Médica», *Rev. Cuba. Cienc. Informáticas*, vol. 13, n.º 2, pp. 77-90, jun. 2019.
- [24] L. Copeland, «Extreme Programming», *Computerworld*, 3 de diciembre de 2001. <https://www.computerworld.com/article/2585634/extreme-programming.html> (accedido 28 de enero de 2022).
- [25] «Metodología XP o Programación Extrema: ¿Qué es y cómo aplicarla?», *Gestor de proyectos online*. <https://www.sinnaps.com/blog-gestion-proyectos/metodologia-xp> (accedido 28 de enero de 2022).
- [26] J. Medina Cruz, E. Pineda Ballesteros, F. R. Téllez Acuña, J. Medina Cruz, E. Pineda Ballesteros, y F. R. Téllez Acuña, «Requerimientos de software: prototipado, software heredado y análisis de documentos», *Ing. Desarro.*, vol. 37, n.º 2, pp. 327-345, dic. 2019, doi: 10.14482/inde.37.2.1053.
- [27] A. O. Duarte, L. D. C. Aljure, y T. Delgado, «Alternativa Open Source en la implementación de un sistema IoT para la medición de la calidad del aire.», *Rev. Cuba. Cienc. Informáticas*, vol. 12, n.º 1, pp. 189-204, 2018.
- [28] G. Muelle, «Framework: ¿qué es y cuál es su función en Internet?», *Rock Content - ES*, 8 de enero de 2020. <https://rockcontent.com/es/blog/framework/> (accedido 29 de enero de 2022).
- [29] «¿Qué es ReactJS? Conceptos básicos», *IfgeekthenNTTdata*, 26 de marzo de 2020. <https://ifgeekthen.nttdata.com/es/que-es-reactjs-conceptos-basicos> (accedido 29 de enero de 2022).
- [30] «Preprocesador CSS - Glosario | MDN». https://developer.mozilla.org/es/docs/Glossary/CSS_preprocessor (accedido 29 de enero de 2022).

- [31] «¿Qué es un servidor?» <https://www.onyxsystems.es/que-es-un-servidor.html> (accedido 29 de enero de 2022).
- [32] Node.js, «Acerca», *Node.js*. <https://nodejs.org/es/about/> (accedido 29 de enero de 2022).
- [33] «Introducción a Express/Node - Aprende sobre desarrollo web | MDN». https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs/Introduction (accedido 29 de enero de 2022).
- [34] G. Trentin, «Estructura y organización de una base de datos», *CL E Comun. Leng. Educ.*, n.º 13, pp. 81-88, 1992.
- [35] L. Fattori, «Narrativa y estética de la database», *Rev. Razón Crítica*, n.º 6, pp. 33-59, 2019.
- [36] L. P. Moreno y C. A. Calderón, «Arquitectura referencial de Big Data para la gestión de las telecomunicaciones», *Ingeniare Rev. Chil. Ing.*, vol. 25, n.º 4, pp. 566-577, 2017.
- [37] «Qué es MongoDB y características», *OpenWebinars.net*, 28 de octubre de 2019. <https://openwebinars.net/blog/que-es-mongodb/> (accedido 29 de enero de 2022).
- [38] «Seguridad», *MongoDB*. <https://www.mongodb.com/es/cloud/atlas/security> (accedido 29 de enero de 2022).
- [39] *Visual Studio Code - Open Source* («Code - OSS»). Microsoft, 2022. Accedido: 30 de enero de 2022. [En línea]. Disponible en: <https://github.com/microsoft/vscode>
- [40] A. C. Mondeja y O. G. Baryolo, «Estado del arte en el desarrollo de aplicaciones web basado en ingeniería dirigida por modelos.», *Rev. CIENTÍFICA ECOCIENCIA*, vol. 4, n.º 1, Art. n.º 1, 2017, doi: 10.21855/ecociencia.41.23.
- [41] G. E. Vega, M. T. Villalobos, L. V. Acuña, y R. Q. Oviedo, «Una comparación de rendimiento entre bases de datos NoSQL: MongoDB y ArangoDB», *Tecnol. En Marcha*, vol. 32, n.º Extra 9, pp. 5-15, 2019.
- [42] M. R. M. Morales, J. W. D. Cazar, E. J. T. Gaona, y S. M. Cardoso, «Rendimiento de bases de datos columnares», *Ingenius Rev. Cienc. Tecnol.*, n.º 22, pp. 47-58, 2019.
- [43] «Authentication — MongoDB Manual», <https://github.com/mongodb/docs-bi-connector/blob/DOCSP-3279/source/index.txt>. <https://docs.mongodb.com/manual/core/authentication/> (accedido 20 de febrero de 2022).
- [44] «How To Back Up And Restore MongoDB», *MongoDB*. <https://www.mongodb.com/basics/backup-and-restore> (accedido 14 de febrero de 2022).
- [45] «Replica Set Elections — MongoDB Manual», <https://github.com/mongodb/docs-bi-connector/blob/DOCSP-3279/source/index.txt>. <https://docs.mongodb.com/manual/core/replica-set-elections/> (accedido 20 de febrero de 2022).
- [46] M. Rivera López, R. Santander Acosta, S. Sixto Fuentes, M. Rivera López, R. Santander Acosta, y S. Sixto Fuentes, «Arquitectura de información para la gestión de la historia clínica digital en oftalmopediatría», *Rev. Cienc. Médicas Pinar Río*, vol. 25, n.º 2, abr. 2021, Accedido: 14 de febrero de 2022. [En línea]. Disponible en: http://scielo.sld.cu/scielo.php?script=sci_abstract&pid=S1561-31942021000200009&lng=es&nrm=iso&tlng=es

- [47] R. P. Lima, M. A. T. Elmiro, M. A. Nero, P. da C. Temba, B. M. Fonseca, y L. H. G. Castiglione, «ASSESSMENT OF DIGITAL TERRAIN MODELS IN DAM BREAK SIMULATION STUDIES», *Bol. Ciênc. Geodésicas*, vol. 27, n.º 1, Accedido: 20 de febrero de 2022. [En línea]. Disponible en: <https://www.redalyc.org/journal/3939/393966213005/>
- [48] J. R. M. Ríos, «Modelo de evaluación de metodologías de desarrollo de software web», <http://purl.org/dc/dcmitype/Text>, Universidade da Coruña, 2021. Accedido: 20 de febrero de 2022. [En línea]. Disponible en: <https://dialnet.unirioja.es/servlet/tesis?codigo=295964>
- [49] M. Callejas-Cuervo, A. C. Alarcón-Aldana, y A. M. Álvarez-Carreño, «Modelos de calidad del software, un estado del arte», *Entramado*, vol. 13, n.º 1, pp. 236-250, 2017.
- [50] F. F. R. Castillo, N. M. L. Mora, K. D. C. Elizaldes, y J. I. P. Orozco, «Comparación de métricas de calidad para el desarrollo de aplicaciones web», *3c Tecnol. Glosas Innov. Apl. Pyme*, vol. 7, n.º 3, pp. 94-113, 2018.
- [51] D. Carrizo y A. Alfaro, «Método de aseguramiento de la calidad en una metodología de desarrollo de software: un enfoque práctico», *Ingeniare Rev. Chil. Ing.*, vol. 26, n.º 1, pp. 114-129, mar. 2018, doi: 10.4067/S0718-33052018000100114.
- [52] V. Pagnoni y S. I. M. Fernández, «Validación de contenido de un portal educativo centrado en la Accesibilidad Web», *IE Comun. Rev. Iberoam. Informática Educ.*, n.º 29 (Enero-Junio), pp. 14-22, 2019.
- [53] D. J. B. Logroño, O. O. E. Lara, y A. D. P. Rivera, «Implementación del bootstrap como una metodología ágil en la web», *Rev. Arbitr. Interdiscip. Koinonía*, vol. 5, n.º 9 (Enero-Julio), pp. 268-286, 2020.
- [54] Y. Riccardi Sabatier, R. L. Vega Almeida, y E. Miyares Díaz, «Aplicación del Responsive Web Design en la creación e implementación del sitio Web del Centro de Histoterapia Placentaria», *Rev. Cuba. Informática Médica*, vol. 10, n.º 1, pp. 16-27, jun. 2018.

ANEXOS

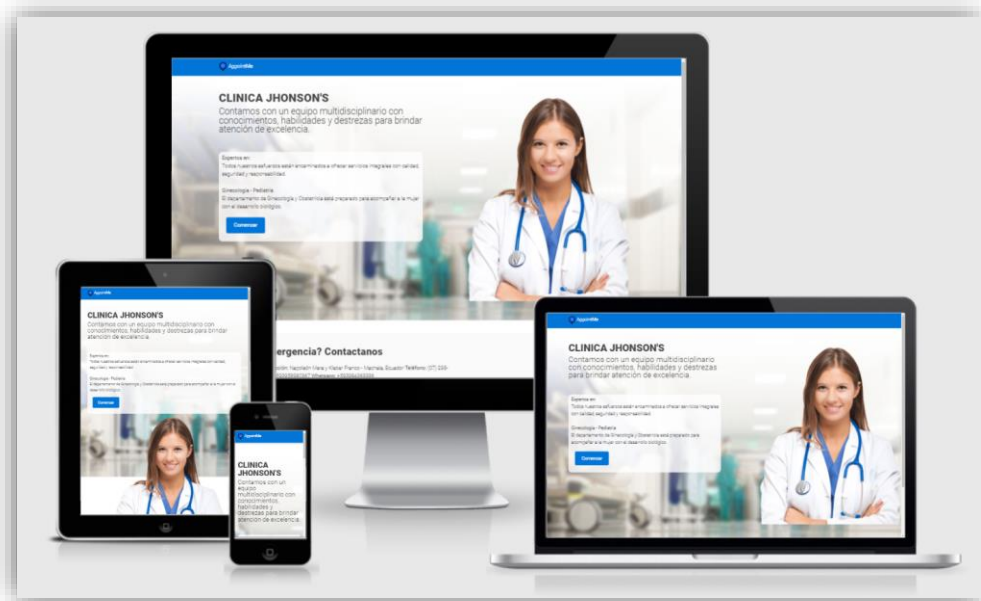
Anexo 1: Plantilla historias de usuarios

Tabla 12: Plantilla historia de usuario

Historia de Usuario	
Número:	Usuario:
Nombre Historia:	
Prioridad en negocio:	Riesgo en negocio:
Programador Responsable:	
Descripción:	
Observaciones:	

Fuente: Elaboración Propia

Anexo 2: Diseño responsivo



Anexo 3: Evaluación de calidad – Herramienta Website grader



30/30

Móvil

El tráfico de los dispositivos móviles crece con rapidez. Optimiza tu sitio web para dispositivos móviles o perderás tráfico, leads e ingresos valiosos.



Mejora el rendimiento en dispositivos móviles con una [guía gratuita](#)

APROBADO

TAMAÑO DE FUENTE LEGIBLE



Fácil de leer.

Los visitantes pueden tener dificultades para leer texto pequeño, especialmente en dispositivos móviles. Recomendamos al menos 12px.

APROBADO

ELEMENTOS INTERACTIVOS



Fácil de usar.

Las páginas compatibles con dispositivos móviles tienen mejor rendimiento en los resultados de búsqueda. Asegúrate de que los elementos interactivos como botones y enlaces no sean demasiado pequeños o estén demasiado cerca.

APROBADO

ADAPTABILIDAD



Un sitio bien diseñado.

Un diseño adaptable mejora tu clasificación en las búsquedas en dispositivos móviles.

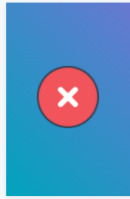


5/10

Seguridad

Un sitio web seguro, equipado con un certificado SSL y libre de vulnerabilidades, es ahora el estándar en línea. A los internautas y a los motores de búsqueda les encantan los sitios web seguros.

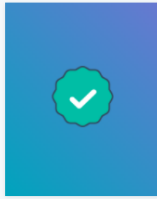
Mejora la seguridad de tu sitio web con una [guía gratuita](#)



HTTPS

Cuidado. No todos los recursos se cargan en HTTPS.

HTTPS protege a los sitios web de ataques y les da a los visitantes la confianza de que tu sitio es auténtico y fiable.



BIBLIOTECAS SEGURAS DE JAVASCRIPT

Protegido.

Los intrusos pueden aprovechar las bibliotecas de JavaScript desactualizadas. Usar la última versión de cada biblioteca y actualizarla con regularidad te mantendrá protegido.



30/30

SEO

Optimizar el contenido de tu sitio web para las búsquedas te ayuda a atraer tráfico orgánico hacia tu sitio web. Puedes hacerlo ofreciendo una excelente experiencia para los visitantes y rastreadores web por igual.

Mejora tu clasificación de SEO con una [guía gratuita](#)

APROBADO

PERMISO PARA INDEXAR



Concedido.

Para que una página aparezca en los resultados de búsqueda, los motores de búsqueda deben tener permiso para almacenarla en su indexación. Si no pueden almacenarla, ningún otro cambio importa.

APROBADO

METADESCRIPCIÓN



¡Muy bien!

Las metadescripciones permiten que los usuarios sepan de qué trata tu página en los resultados de búsqueda.

APROBADO

PLUGINS DE CONTENIDO



Estás haciendo lo correcto.

Los motores de búsqueda no siempre pueden comprender el contenido que se basa en plugins, como Flash.

APROBADO

TEXTO DESCRIPTIVO DEL ENLACE



Tus descripciones son buenas.

El texto descriptivo de un enlace ayuda a los visitantes a saber qué verán si hacen clic en el enlace. Una CTA para hacer clic no es suficiente.



22/30

Rendimiento

Optimizar el rendimiento de tu sitio web es crucial para incrementar el tráfico, mejorar las tasas de conversión, generar más leads y aumentar los ingresos.



Mejora el rendimiento de tu sitio web con una [guía gratuita](#)

TAMAÑO DE LA PÁGINA



1.3 MB

¡Qué rapidez!

Mientras más pesado es el sitio web, más lenta es la carga. Para un rendimiento óptimo, intenta mantener el tamaño de la página por debajo de los 3MB.

SOLICITUDES DE PÁGINAS



11

Lograste el equilibrio perfecto.

Mientras más solicitudes de HTTP realiza tu sitio web, más lento se vuelve. Prueba reduciendo la cantidad de archivos que carga tu sitio.

VELOCIDAD DE LA PÁGINA



6.4 SEG

Podría ser más rápida.

Las mejores páginas web deberían volverse interactivas en máximo 5.3 segundos. Si son más lentas, los visitantes se irán y esto afectará las conversiones y las ventas.

APROBADO

CACHÉ WEB



¡Muy bien! Tu caché web es de primera.

La caché web incrementa la velocidad de tu sitio web almacenando contenido utilizado con frecuencia en una memoria local.

APROBADO

REDIRECCIONAMIENTOS DE PÁGINA MÍNIMOS



Directo al grano.

Múltiples redireccionamientos pueden hacer que tu sitio se cargue más lentamente. Intenta no tener más de uno.

APROBADO

TAMAÑO DE LA IMAGEN



Se adaptan a la perfección.

Las imágenes pueden tardar mucho tiempo en cargarse. Usa imágenes adaptables o SVG para optimizarlas según distintos tamaños de pantalla.

APROBADO

JAVASCRIPT MINIMIZADO



Todo en orden.

Cuando tu JavaScript se comprime correctamente, tu sitio web funciona con mucha más rapidez.

APROBADO

CSS MINIMIZADO



Breve, como debe ser.

Cuando tu CSS se comprime correctamente, tu sitio web funciona con mucha más rapidez.