
Sistematización de procesos administrativos en clínicas pediátricas mediante el desarrollo de un sistema web

Gabriela Paola Contreras Guerra



UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería




**Sistematización de procesos administrativos en
clínicas pediátricas mediante el desarrollo de un
sistema web**

Trabajo de graduación en modalidad de tesis presentado por
Gabriela Paola Contreras Guerra
para optar al grado académico de Licenciada en Ingeniería en Ciencia de
la Computación y Tecnologías de la Información


Guatemala,
2024

Vo.Bo.:


Ronald Giovanni Estrada Avila
Ingeniero en Sistemas de
Información UMG
Colegiado No. 6136


(f) _____
Ing. Ronald Giovanni Estrada Avila

Tribunal Examinador:


Ronald Giovanni Estrada Avila
Ingeniero en Sistemas de
Información UMG
Colegiado No. 6136

(f) _____
Ing. Ronald Giovanni Estrada Avila


(f) _____
Ing. Eddy Omar Castro Jauregui


(f) _____
Ing. Bacilo Alexander Bolaños Lima

Fecha de aprobación: Guatemala, 04 de diciembre de 2024—.

Índice	III
Lista de figuras	V
Lista de cuadros	VIII
Resumen	IX
Abstract	X
1 Introducción	1
2 Justificación	2
3 Objetivos	3
3.1 Objetivo general	3
3.2 Objetivos específicos	3
4 Marco teórico	4
4.1 Medicina	4
4.1.1 Antecedentes	4
4.1.2 Sector privado	5
4.2 Pediatría	5
4.2.1 Pediatra	5
4.2.2 Contexto en Guatemala	6
4.3 Clínicas médicas	6
4.4 Sistematizar	6
4.5 Aplicación web	7
4.6 Interacción humano computador	7
4.6.1 Color	8
4.6.2 Tipografía	11
4.6.3 Prototipo	11
4.6.4 Experiencia de usuario	12
4.7 Backend	12
4.7.1 Base de datos	13
4.8 Frontend	13
4.8.1 HTML	13
4.8.2 CSS	13

4.8.3	JavaScript	14
4.8.4	Frameworks para frontend	14
4.9	Pruebas unitarias	14
4.9.1	Frameworks para pruebas unitarias	15
4.10	Control de versiones	15
4.10.1	GitHub	16
4.10.2	Commits	16
4.10.3	Branches	16
4.10.4	Pull Request	16
4.11	Minimum viable product	16
4.12	Integración continua	16
5	Metodología	17
5.1	Investigación	17
5.2	Diseño de aplicación web	17
5.2.1	Paleta de colores	17
5.2.2	Selección de fuentes	19
5.3	Prototipado	20
5.3.1	Pantallas del prototipo	20
5.3.2	Retroalimentación del prototipo	32
5.4	Selección de tecnologías	32
5.5	Arquitectura de la aplicación web	37
5.6	Desarrollo	38
5.7	Control de versiones	40
5.8	Integración continua	40
5.9	Pruebas a usuarios y retroalimentación	41
6	Resultados	42
6.1	Aplicación web	42
6.1.1	Inicio de sesión	42
6.1.2	Restablecer contraseña	43
6.1.3	Agenda	44
6.1.4	Expedientes	45
6.1.5	Cita actual	47
6.1.6	Historial médico	48
6.1.7	Finanzas	49
6.1.8	Ajustes	50
6.1.9	Gestión de roles	52
6.2	Evaluación de la aplicación web	52
6.3	Presupuesto	55
7	Discusión de resultados	57
8	Conclusiones	58
9	Recomendaciones	59
10	Referencias	60
11	Anexos	65
11.1	Enlace al prototipo funcional	65
11.2	Enlace al repositorio de GitHub	65
11.3	Tareas asignadas al doctor	66
11.4	Tareas asignadas a la secretaria	67

Lista de figuras

Figura 1	<i>Absorción y reflexión de colores en objetos.</i>	8
Figura 2	<i>Círculo cromático de 12 escalones dividido en colores cálidos y fríos</i>	9
Figura 3	<i>Esquema de colores y emociones vinculadas a cada uno</i>	9
Figura 4	<i>Esquemas de colores</i>	10
Figura 5	<i>Categorías de prototipos ilustradas</i>	12
Figura 6	<i>Color primario en hexadecimal</i>	18
Figura 7	<i>Colores secundarios en hexadecimal</i>	18
Figura 8	<i>Colores neutros en hexadecimal</i>	19
Figura 9	<i>Tipografía para el texto del cuerpo</i>	19
Figura 10	<i>Tipografía para títulos</i>	19
Figura 11	<i>Identidad visual de la clínica</i>	20
Figura 12	<i>Mockup de pantalla de inicio de sesión</i>	21
Figura 13	<i>Mockup de pantalla para agenda</i>	21
Figura 14	<i>Mockup de pantalla para ingresar expediente</i>	22
Figura 15	<i>Mockup de pantalla para buscar expediente</i>	23
Figura 16	<i>Mockup de pantalla para historial médico</i>	24
Figura 17	<i>Mockup de pantalla para consultar expediente</i>	25
Figura 18	<i>Mockup de pantalla para modificar expediente</i>	26
Figura 19	<i>Mockup de pantalla para recetar</i>	27
Figura 20	<i>Mockup de pantalla para ingresar cita actual</i>	28
Figura 21	<i>Mockup de pantalla para consultar y modificar cita actual</i>	29
Figura 22	<i>Mockup de pantalla para ingresar un cobro</i>	29
Figura 23	<i>Mockup de pantalla para la gestión de caja del día</i>	30
Figura 24	<i>Mockup de pantalla para registro de finanzas</i>	30
Figura 25	<i>Mockup de pantalla para gestión de aseguradoras</i>	31
Figura 26	<i>Mockup de pantalla para aseguradoras</i>	31
Figura 27	<i>Representación visual de la arquitectura y componentes de la aplicación web</i>	38
Figura 28	<i>Ejecución de ESLint durante commit</i>	41
Figura 29	<i>Ejecución de pruebas unitarias durante push</i>	41
Figura 30	<i>Pantalla que muestra el formulario de inicio de sesión sin datos ingresados</i>	42
Figura 31	<i>Pantalla que muestra el formulario de inicio de sesión con datos ingresados</i>	42
Figura 32	<i>Pantalla que indica un error en el inicio de sesión, y mensaje de advertencia</i>	43
Figura 33	<i>Notificación que informa sobre el proceso para solicitar el cambio de contraseña</i>	43
Figura 34	<i>Correo electrónico que contiene la información necesaria para restablecer la contraseña</i>	43

Figura 35	Notificación que muestra un mensaje de error sobre códigos incorrectos y solicitud de una nueva entrada para continuar	44
Figura 36	Pantalla que permite al usuario ingresar una nueva contraseña	44
Figura 37	Pantalla que muestra la agenda organizada por semanas	44
Figura 38	Pantalla que presenta la agenda organizada por meses	44
Figura 39	Pantalla destinada a programar una nueva cita para el paciente	45
Figura 40	Pantalla que muestra la cita creada en el calendario	45
Figura 41	Pantalla destinada para modificar los detalles de una cita existente	45
Figura 42	Pantalla que muestra la cita actualizada en el calendario	45
Figura 43	Pantalla que refleja la cancelación de una cita en el calendario	45
Figura 44	Sección superior de la ficha médica que muestra algunos datos esenciales a ingresar en el expediente clínico del paciente	46
Figura 45	Sección inferior de la ficha médica que muestra los botones disponibles para completar el ingreso de datos y gestionar el expediente clínico del paciente	46
Figura 46	Mensaje de advertencia que aparece cuando uno de los campos obligatorios se encuentra sin datos	46
Figura 47	Notificación que detalla las condiciones necesarias para la correcta validación del campo	46
Figura 48	Sección superior de la ficha médica que muestra los campos con los datos registrados en el expediente clínico	46
Figura 49	Sección inferior de la ficha médica que muestra los botones disponibles para navegar dentro de la aplicación	46
Figura 50	Sección superior de cita actual que muestra algunos datos que se deben llenar para el historial del paciente	47
Figura 51	Sección inferior de cita actual que muestra los datos complementarios que se pueden detallar y los botones disponibles	47
Figura 52	Sección superior de cita actual que muestra los datos ingresados, relevantes para el historial del paciente	47
Figura 53	Sección inferior de cita actual que muestra algunos datos llenados y los botones disponibles para continuar navegando	47
Figura 54	Sección que presenta la cita actual con datos actualizados	47
Figura 55	Pantalla que permite buscar y acceder a los expedientes clínicos de los pacientes	48
Figura 56	Pantalla que muestra el historial médico completo del paciente	48
Figura 57	Pantalla diseñada para registrar los medicamentos que se prescribirán al paciente	48
Figura 58	Pantalla que permite eliminar un medicamento ingresado	48
Figura 59	Sección superior de la receta médica que presenta información relevante de la clínica y el paciente	49
Figura 60	Pantalla que permite visualizar y preparar la impresión de la receta médica	49
Figura 61	Pantalla que muestra el historial actualizado en el campo de medicamentos	49
Figura 62	Pantalla diseñada para registrar un cobro en el sistema	50
Figura 63	Pantalla que muestra un resumen de todas las transacciones realizadas durante el día	50
Figura 64	Pantalla que presenta un resumen detallado de las ganancias	50
Figura 65	Pantalla que muestra las diferentes opciones y funcionalidades disponibles en el sistema	50
Figura 66	Pantalla que permite observar y editar la información general de la clínica	50
Figura 67	Pantalla destinada al registro y configuración de un nuevo usuario en la aplicación	51
Figura 68	Pantalla que muestra los usuarios habilitados para utilizar la aplicación	51
Figura 69	Pantalla que muestra los seguros utilizados con opciones adicionales para editar o eliminar cada uno	51
Figura 70	Pantalla para modificar y actualizar los detalles de un seguro existente	51
Figura 71	Pantalla que detalla los campos accesibles para gestionar la información en los expedientes médicos	51

Figura 72	<i>Pantalla que muestra una advertencia al intentar eliminar un campo requerido según los lineamientos de la OMS</i>	51
Figura 73	<i>Pantalla que muestra una alerta al agregar un nuevo campo en una sección específica</i>	52
Figura 74	<i>Pantalla que confirma la correcta integración del campo en el expediente médico</i>	52
Figura 75	<i>Notificación de acceso denegado al ingresar a una vista sin autorización</i>	52
Figura 76	<i>Respuestas de los usuarios sobre la estética visual de la interfaz.</i>	53
Figura 77	<i>Calificaciones de los usuarios sobre la intuición de navegación en la aplicación.</i>	53
Figura 78	<i>Calificaciones de los usuarios sobre la facilidad de navegación a través de las diferentes secciones del sistema.</i>	54
Figura 79	<i>Calificaciones de los usuarios sobre la efectividad de la aplicación de roles en las diferentes vistas para el personal.</i>	54
Figura 80	<i>Calificaciones de los usuarios sobre la facilidad de realizar las distintas funciones asignadas al doctor en el sistema.</i>	55
Figura 81	<i>Calificaciones de los usuarios sobre la facilidad de realizar las distintas funciones asignadas a la secretaria en el sistema.</i>	55
Figura 82	<i>Estimación de costos mensuales para el mantenimiento de la aplicación web</i>	56
Figura 83	<i>Listado de instrucciones realizadas por el doctor</i>	66
Figura 84	<i>Listado de instrucciones realizadas por la secretaria</i>	67

Lista de cuadros

Cuadro 1	Ventajas y desventajas de Firebase	33
Cuadro 2	Ventajas y desventajas de MongoDB	33
Cuadro 3	Ventajas y desventajas de React	34
Cuadro 4	Ventajas y desventajas de Vue	35
Cuadro 5	Ventajas y desventajas de Angular	35
Cuadro 6	Ventajas y desventajas de Jest	36
Cuadro 7	Ventajas y desventajas de Jasmine	36
Cuadro 8	Ventajas y desventajas de Vitest	37

En respuesta a la creciente necesidad de digitalización en el sector de la salud, el presente trabajo desarrolla una aplicación web para clínicas pediátricas. Su objetivo principal es reemplazar los métodos tradicionales de manejo de información, los cuales, al implicar el uso de expedientes físicos, presentan riesgos significativos, tales como pérdida, daño o duplicidad de los datos del paciente. La digitalización ofrece a las clínicas la posibilidad de almacenar y administrar la información de manera ágil, eficiente y segura, liberando espacio físico y optimizando la accesibilidad a los registros.

El sistema desarrollado permite gestionar el historial médico de los pacientes, así como mantener un control organizado del calendario de actividades y las finanzas de la clínica. Además, se ha considerado la seguridad y confidencialidad de los datos, integrando roles de acceso que garantizan que solo el personal autorizado tenga acceso a información sensible. De esta manera, la aplicación no solo facilita la administración de la clínica, sino que también contribuye a una experiencia médica más personalizada y segura, enfocada en la mejora de la calidad del servicio ofrecido.

Los resultados de la implementación mostraron que la interfaz es intuitiva y fácil de usar, permitiendo una rápida adopción por parte del personal médico y administrativo sin requerir conocimientos técnicos avanzados. Asimismo, el sistema de roles garantizó la confidencialidad y el acceso restringido a la información sensible, siendo bien recibido por la mayoría de los usuarios. La centralización de la información en un solo módulo facilitó el acceso y la actualización de los datos de los pacientes, contribuyendo a una gestión más eficiente y, por ende, a una mejora en la calidad del servicio. Estos hallazgos confirman que la herramienta representa un avance significativo en la digitalización de procesos clínicos y con el potencial de evolucionar para satisfacer las necesidades cambiantes del entorno médico.

Palabras clave: digitalización, pediatría, sistema web, gestión clínica, historiales médicos, sistematización de procesos.

In response to the growing need for digitalization in the healthcare sector, this study presents the development of a web application for pediatric clinics. Its main objective is to replace traditional methods of information management, which rely on physical records and pose significant risks such as loss, damage, or duplication of patient data. Digitalization enables clinics to store and manage information in a faster, more efficient, and secure manner, while also saving physical space and improving access to medical records.

The system developed allows the management of patients' medical histories, the organization of clinic schedules, and financial control. Data security and confidentiality were also prioritized through the implementation of access roles that ensure only authorized personnel can view sensitive information. Thus, the application not only facilitates clinic administration but also enhances the delivery of personalized and secure medical care, contributing to improved service quality.

The implementation results demonstrated that the interface is intuitive and easy to use, enabling quick adoption by medical and administrative staff without the need for advanced technical knowledge. Furthermore, the role-based system ensured confidentiality and restricted access to sensitive data, which was positively received by most users. Centralizing patient information within a single module simplified access and updates, leading to more efficient management and an overall improvement in service quality. These findings confirm that the tool represents a significant advancement in the digitalization of clinical processes, with the potential to evolve and meet the changing needs of the medical environment.

Keywords: digitalization, pediatrics, web system, clinical management, medical records, process systematization.

CAPÍTULO 1

Introducción

La medicina es un conjunto de conocimientos y técnicas aplicados a la prevención, el diagnóstico y el tratamiento de enfermedades humanas. Se divide en diversas ramas como anatomía, cardiología, cirugía, dermatología, entre otras. La pediatría, la cual surgió a finales del siglo XIX como especialidad médica, es la rama de la medicina que se enfoca en la atención a niños desde su nacimiento hasta la adolescencia, abarcando áreas como el crecimiento, el desarrollo, la nutrición, la salud mental, los cuidados preventivos, etc. Entre la amplia gama de enfoques de la pediatría, resalta la atención preventiva: chequeos regulares, vacunaciones, asesoramiento de la crianza, asesoramiento de la nutrición del niño. Esto contribuye en gran manera a la identificación temprana de problemas de desarrollo y salud.

En Guatemala, el sistema de salud se encuentra conformado por dos sectores: público y privado. El sector privado, con fines de lucro, incluye hospitales del tercer nivel, pequeños hospitales generales y de especialidades (95), clínicas médicas especializadas (2.927), clínicas de medicina general (1.103) y laboratorios (1.373), lo que representa aproximadamente 60% de todos los centros registrados en Guatemala [60].

El presente proyecto consiste en el desarrollo de un sistema dedicado al área de salud, siendo su principal objetivo las clínicas médicas pequeñas ubicadas en municipios o pueblos dentro de la Ciudad de Guatemala. El objetivo de esta herramienta es apoyar a los profesionales de la medicina, facilitando el registro y consulta de historiales médicos, así como la generación de prescripciones y recetas de manera ágil y clara para la adquisición de medicamentos. Además, incorporará un módulo financiero para el control diario y mensual de los ingresos por servicios prestados. El sistema contará con roles que garantizarán que cada profesional acceda únicamente a la información que le corresponde, asegurando la integridad y seguridad de los datos

La transición hacia la digitalización en el sector de la salud es más que una tendencia, ya que ofrece múltiples beneficios tanto para los profesionales de la salud como para el personal de apoyo y los pacientes. La digitalización, especialmente a través de la tecnología de la información de salud (health IT), permite la administración de la información de manera más ágil, eficiente y segura [19].

Los métodos tradicionales pueden resultar inoperantes, ya que inciden en la acumulación de expedientes físicos, ocupando un espacio valioso dentro de las clínicas lo que conlleva riesgos significativos como la pérdida, daño o duplicidad de la información. En contraste, almacenar historias clínicas de pacientes de forma electrónica libera espacio físico, economía en el uso de papel y archivos, optimiza el uso compartido de la información y mejora la seguridad de los datos del paciente los cuales son de carácter confidencial.

El uso de sistemas de registro electrónico de pacientes (EMR) facilita la recopilación sistematizada de datos e historiales de pacientes, lo que a su vez permite una gestión más efectiva de la información, reduciendo la posibilidad de errores manuales y mejorando la calidad de la atención médica, en cuanto a la agilización en la localización y consulta de expedientes, así como la emisión de recetas.

La automatización es un aspecto clave en la transformación digital de la salud, permitiendo la realización autónoma y eficiente de tareas administrativas y logísticas, como la gestión de citas, el registro de pacientes y la organización de historiales médicos. Esto libera al personal médico de tareas rutinarias y les permite concentrarse en la atención directa, contribuyendo así a una experiencia más personalizada y satisfactoria para el paciente. La digitalización y registro de información ofrece ventajas en términos de accesibilidad, permitiendo compartir documentos desde cualquier ubicación y dispositivo con conexión a Internet, seguridad, eficiencia, así como la reducción de costos asociados con la impresión y el almacenamiento físico.

3.1. Objetivo general

Desarrollar una herramienta enfocada en clínicas pediátricas, la cual se encargue de sistematizar la gestión de procesos médicos y administrativos.

3.2. Objetivos específicos

- Implementar una interfaz que sea intuitiva y fácil de usar tanto para el médico, como para su personal.
- Establecer un sistema de roles con el fin de restringir y regular el acceso y la modificación de los datos médicos de los pacientes por parte del personal, garantizando la confidencialidad y seguridad de la información.
- Elaborar un módulo de registros el cual centralice la información general de cada paciente, conteniendo su ficha e historial médico, asegurando que la información sea fácil de actualizar y accesible en todo momento.

4.1. Medicina

La medicina es una ciencia enfocada a la prevención y cura de enfermedades del cuerpo humano. Esta se centra en el diagnóstico, prevención, prescripción y tratamiento de enfermedades o cualquier otra afección física o mental del ser humano. El término medicina proviene del latín “medicina” el cual hace referencia a esta disciplina.

El principal objetivo de practicar la medicina es velar por la salud tanto de los individuos como comunidades. Cada acto médico debe realizarse con eficiencia y la menor cantidad de riesgos posibles esto con la finalidad de garantizar la calidad en cuanto al ejercicio de esta profesión. Cabe destacar, que para poder ejercer en este campo es indispensable contar con estudios los cuales proporcionen las habilidades, destrezas y conocimientos necesarios. Estos estudios acreditan a un doctor o médico convertirse en un agente de la salud y poder establecer una relación con una persona enferma.

Es importante señalar que la medicina fue la primera profesión en exigir licencias para su práctica y cualquier individuo que desee diagnosticar o tratar como parte de la profesión, pero no esté debidamente acreditado, puede ser culpable de "practicar la medicina sin licencia".

4.1.1. Antecedentes

La medicina tuvo inicios en Guatemala desde la época precolombina, esta era practicada por sacerdotes quienes utilizaban sus conocimientos heredados para curar enfermedades en un contexto místico y sagrado. Los pueblos indígenas como los K'iche', Q'eqchi' y Mam desarrollaron sus propios sistemas médicos basados en la naturaleza y la transmisión de conocimientos, empleando hierbas medicinales y terapias espirituales [30].

Durante la época de la conquista española la medicina cambió de manera significativa pues se introdujo a la medicina hipocrática y galénica. La llegada de médicos españoles marcó un inicio para estructurar esta disciplina dentro del país, ya que muchos practicaban esta sin respaldo generando prácticas poco éticas. En 1562, se graduó el primer médico en Guatemala, el Dr. Pedro de Pineda, y en 1804, el Dr. José Felipe Flores impulsó la creación de la Escuela de Medicina.

En 1981 se inauguraron los estudios de medicina dentro de la Universidad de San Carlos de Guatemala, siendo este un hito importante para la educación médica dentro del país.

4.1.2. Sector privado

El sector privado de atención de salud en Guatemala representa un segmento grande y en constante crecimiento, este incluye hospitales de tercer nivel, hospitales generales y de especialidades, clínicas particulares, farmacias y proveedores tradicionales. A pesar de los costos de adquirir esta atención, muchos guatemaltecos buscan esta como su primer punto de contacto.

En total, el sector privado representa aproximadamente el 60% de todos los centros de salud registrados en el país, con 95 hospitales generales y de especialidades, 2,927 clínicas médicas especializadas, 1,103 clínicas de medicina general y 1,373 laboratorios [60].

A pesar del alcance y expansión del sector privado, Guatemala enfrenta altos niveles de desigualdad y pobreza, con más de la mitad de sus 15 millones de habitantes viviendo por debajo de la línea de pobreza [60]. Esta desigualdad se ve reflejada en el acceso a los servicios de salud, ya que solo el 49% de los guatemaltecos acuden a proveedores de servicios de salud cuando enfrentan un problema de salud.

4.2. Pediatría

Esta es una rama de la medicina la cual se encarga de la salud y el cuidado médico de los niños y adolescentes desde su nacimiento hasta la edad de los 18 años. El termino pediatría proviene del griego “pais” (niño) y “iatros” (médico) la unión de estos se traduce en sanador de niños.

La pediatría es la especialidad médica que se ocupa de la salud física, emocional y social de los niños desde el nacimiento hasta la edad adulta. Este tipo de atención toma en cuenta una variedad de servicios, dentro de los cuales se encuentran la atención sanitaria, diagnóstico y tratamiento de enfermedades agudas y crónicas.

La atención pediátrica de un niño es esencial pues asegura el crecimiento y desarrollo de estos, los primeros años de vida son cruciales para el desarrollo del cerebro, sistema inmunológico y otros aspectos vitales de la salud física y mental de un niño. Los pediatras cuentan con la capacidad de poder detectar problemas de salud en etapas tempranas, lo que permite un tratamiento oportuno y puede prevenir complicaciones a largo plazo.

4.2.1. Pediatra

Un pediatra es un médico el cual se encuentra especializado para atender a niños, estos reconocen que los niños se encuentran en constante crecimiento y desarrollo físico y mental. Esto abordan cada etapa de desarrollo de una manera adecuada y específica teniendo en cuenta todos aquellos factores que afecten el crecimiento de un niño.

Es importante destacar que como los niños están en constante maduración intelectual y emocional, los pediatras deben ser capaces de manejar eficazmente los desafíos asociados con estos cambios y comunicarse de manera efectiva con los niños y sus familiares por lo que la atención pediátrica no solo requiere de habilidades médicas, sino que también de empatía y paciencia.

4.2.2. Contexto en Guatemala

En 1945 en Guatemala se creó una asociación pediátrica (ASOPEDIA) en respuesta a la necesidad de un grupo pediátrico independiente que abordara los derechos, el respeto y las necesidades especiales de los niños en Guatemala. La creación de esta asociación marco un hito en la atención médica dentro del país ya que cambio la perspectiva de considerar a los niños como “adultos pequeños” y se empezó a reconocer las necesidades médicas que los infantes tienen.

En un inicio esta se enfocó en establecer y promover prácticas de cuidado médico preventivos para los infantes, siendo estas una innovación en su momento. Actualmente, esta tiene como misión lograr el bienestar social y la salud óptima para todos los grupos de edades pediátricas, extendiendo su alcance hasta los adultos jóvenes.

ASOPEDIA se esfuerza por favorecer la inclusión, valoración y protección de los niños en Guatemala [4], desarrollando un papel importante en la mejora continua de la atención pediátrica y en la promoción de políticas que garanticen sus derechos y bienestar.

4.3. Clínicas médicas

Las clínicas médicas son centros de atención ambulatoria las cuales se dedican al tratamiento de pacientes los cuales no requieran de una hospitalización. Estos establecimientos contribuyen al sistema de la salud puesto a que proporcionan atención de problemas de salud rutinarios y episódicos, así como una amplia gama de servicios preventivos; entre ellos chequeos regulares, administración de vacunas y exámenes diagnósticos.

Contar con centros de atención privados ofrece varias ventajas principalmente para aquellos procedimientos que no son de emergencia puesto a que se puede programar una cita en cualquier momento, la atención es más rápida y personalizada por otra parte el paciente puede elegir cuando y con quien desea ser tratado.

Cabe destacar que estos centros proporcionan planes de salud personalizados y hacen uso de tecnología diagnosticar y brindar tratamientos precisos. La disponibilidad continua de servicios médicos, junto con un entorno cómodo y privado, hace que estos centros sean muy solicitados, ya que ofrecen un acceso sin restricciones a los servicios de salud, además ofrecen tranquilidad a quienes buscan una segunda opinión o están preocupados por las tasas de infección en entornos públicos.

4.4. Sistematizar

Esta hace referencia a una forma de organizar y estandarizar diferentes operaciones, tareas y procedimientos comerciales con la finalidad de crear flujos de trabajo los eficientes y repetibles. Para ello este implica implementar sistemas, herramientas y procesos estructurados con el fin de agilizar las operaciones permitiendo mejorar la productividad y el rendimiento de una organización. Cabe resaltar que la sistematización permite eliminar ineficiencias y reducen los errores creando así una base sólida para el crecimiento de esta.

La implementación de la sistematización dentro del ámbito de la salud trae mejoras significativas en cuanto a la eficiencia y la calidad de atención. Dentro de estas mejoras se puede destacar la disponibilidad de los recursos, el ingreso y monitoreo de un paciente y la reducción en los tiempos de espera. Así mismo, la centralización de los datos del paciente disminuye la necesidad de indagar en datos, agiliza el acceso al historial médico y los tiempos de verificación de datos. Esto también contribuye a la reducción de costos y al cuidado del medio ambiente al minimizar el uso de papel y

otros materiales de impresión.

4.5. Aplicación web

Una aplicación web es un programa al que se accede y ejecuta a través de un navegador web, sin necesidad de instalarlo en el dispositivo del usuario. En lugar de almacenar datos y archivos localmente, estos se guardan en una red o en la nube. Para el funcionamiento óptimo de una aplicación web, se requieren tres componentes, siendo estos un servidor web, un servidor de aplicaciones y una base de datos [63].

Los servidores web son los encargados de gestionar las solicitudes que llegan de un cliente, mientras que el servidor de aplicaciones procesa y completa estas solicitudes. La base de datos, por su parte, almacena toda la información necesaria para el funcionamiento de la aplicación, permitiendo la recuperación y manipulación de datos según sea necesario.

La arquitectura de una aplicación web se compone de dos tipos de componentes, los cuales son los estructurales y los de la interfaz de usuario. Los componentes estructurales incluyen elementos como el cliente o navegador web, el servidor de base de datos y el servidor de aplicaciones web, estos forman la infraestructura que permite que la aplicación funcione y gestione las interacciones entre el usuario y la aplicación. Por otro lado, los componentes de la interfaz de usuario se encargan de la presentación visual de la aplicación siendo así fundamental para la experiencia del usuario, su función principal es mostrar la página web y no interactuar directamente con la arquitectura subyacente de la aplicación. [65]

Es importante destacar que las aplicaciones web presentan características específicas que las distinguen de otras aplicaciones de software. Entre estas características se encuentran:

- Es accesible desde cualquier dispositivo que disponga de un navegador web y conexión a internet.
- No requiere ser descargada en el dispositivo desde el cual se accede, por lo que no ocupa espacio en el mismo
- Se accede a través de un navegador web (como Chrome, Safari, Firefox o Internet Explorer).
- Cuenta con un modo de lectura y escritura, a través del cual el usuario puede visualizar y manipular la información.

4.6. Interacción humano computador

La interacción humano computador (HCI) es un campo multidisciplinario el cual se centra en el diseño de la tecnología informática, en particular, en la interacción entre los seres humanos y los ordenadores. Su objetivo principal es desarrollar interfaces de usuario que sean intuitivas, eficientes y agradables de utilizar [31]. Para lograr esto, HCI se basa en la creación de sistemas que sean no solo funcionales y estables, sino también utilizables y accesibles

Este enfoque no solo busca mejorar la productividad y la experiencia del usuario, sino también minimizar los riesgos asociados. Un diseño inadecuado de las interfaces puede dar lugar a una serie de problemas inesperados, lo que destaca la necesidad de llevar a cabo una planificación cuidadosa y detallada durante el proceso de diseño.

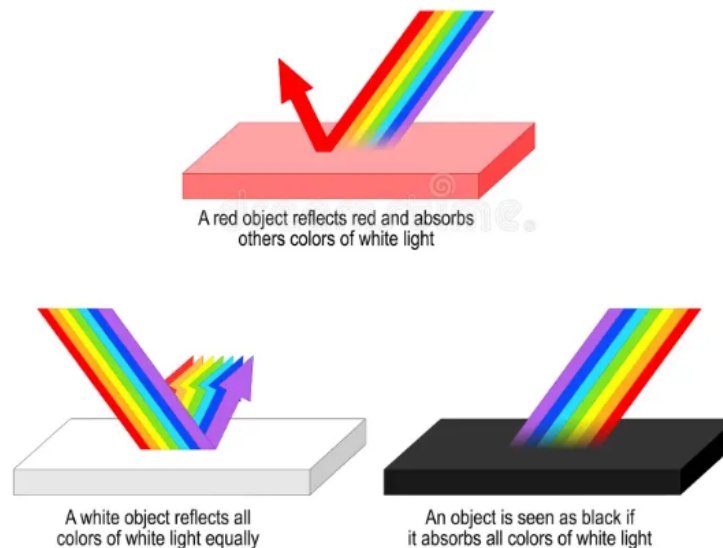
4.6.1. Color

El color es una propiedad visual de un objeto basada en la luz que se refleja sobre él. Específicamente, se refiere a la impresión producida en nuestros ojos e interpretada por nuestro cerebro cuando la luz, en diferentes longitudes de onda, es reflejada o emitida por las cosas que observamos [13].

Cuando un objeto absorbe todos los colores y no refleja ninguno, lo percibimos como negro. En cambio, si un objeto refleja todos los colores de la luz blanca sin absorber ninguno, lo vemos como blanco. Los colores que percibimos son aquellos que el objeto refleja y no los que absorbe. Esto se puede observar de manera gráfica en la figura No.1.

Figura 1 .

Absorción y reflexión de colores en objetos.



Nota: Adaptada de Zhabaska (s.f.)

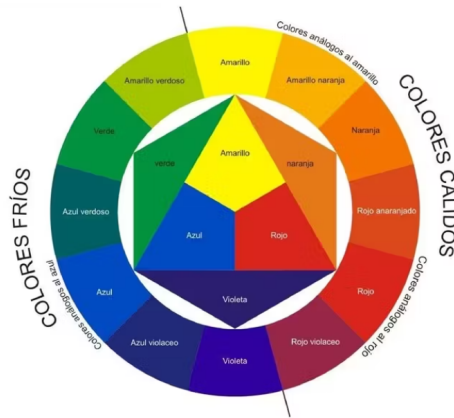
Círculo cromático

El círculo cromático es una representación visual que organiza los colores de acuerdo con sus longitudes de onda. Este muestra la armonía y el contraste entre los colores primarios, secundarios y terciarios.

En total, la rueda de colores básica incluye 12 tonos principales, aunque existen círculos más avanzados que incorporan un espectro más amplio de tintes, sombras, y tonos más matizados [22]. Los colores cálidos, ubicados en una mitad del círculo, están asociados con el calor y la energía. Por otro lado, los colores fríos, situados en la otra mitad, evocan calma y naturaleza [56].

Figura 2 .

Círculo cromático de 12 escalones dividido en colores cálidos y fríos



Nota: Adaptada de Flores (2021)

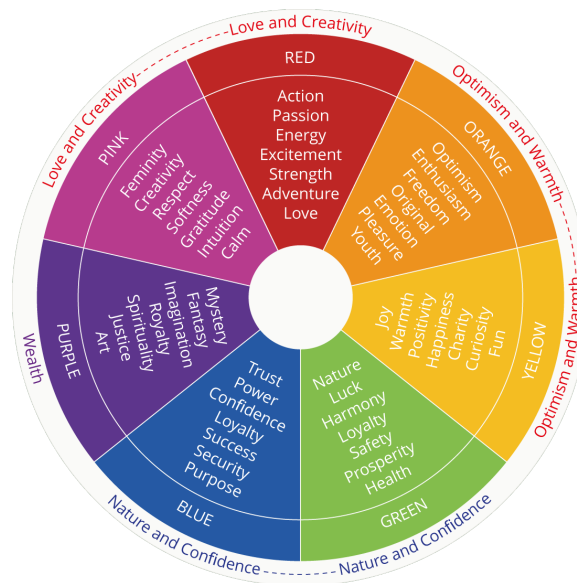
Psicología del color

La psicología del color es el estudio de cómo el color afecta el comportamiento humano. Esta ciencia sostiene que los colores no solo se perciben, sino que también ejercen una influencia significativa sobre el estado de ánimo y mental de un individuo [14].

Los colores forman parte del lenguaje no verbal y poseen el potencial de provocar una variedad de sentimientos, emociones, comportamientos y decisiones. Esta influencia del color se puede ilustrar claramente en la Figura No. 3.

Figura 3 .

Esquema de colores y emociones vinculadas a cada uno



Nota: Adaptada de BeLight Software (2023)

Es importante tener en cuenta que el significado de cada color puede variar según la cultura, las experiencias personales, la religión, el entorno, el género, la raza y la nacionalidad [14].

Teoría del color

La teoría del color examina cómo interactúan los colores y su impacto en nuestras emociones y percepciones. Esta teoría proporciona directrices para seleccionar combinaciones de colores que logren el efecto deseado y transmitan el estado de ánimo o mensaje adecuado en diversas aplicaciones.

Esquemas de color

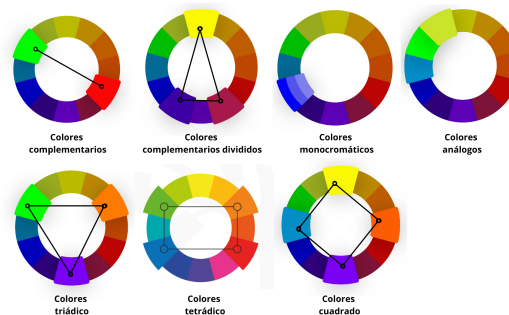
Los esquemas de color son combinaciones estratégicas que ayudan a conseguir un equilibrio visual. Estos esquemas usan los principios de la teoría del color para crear diseños estéticos [22]. Existen diferentes esquemas de color los cuales ofrecen una variedad de cualidades, a continuación se listan algunos de los más comunes:

- Colores complementarios: Consisten en colores ubicados en lados opuestos del círculo cromático.
- Colores complementarios divididos: Se forman combinando un color base con los dos colores terciarios situados a cada lado del color complementario directo
- Colores monocromáticos: Se crean variaciones de un solo color mediante la generación de versiones más claras y más oscuras del tono principal.
- Colores análogos: Comprenden tres colores que se encuentran uno al lado del otro en el círculo cromático
- Colores triádico: Consisten en tres colores equidistantes en el círculo cromático, distribuidos cada 120° , que forman un triángulo equilátero.
- Colores tetrádico: Consisten en cuatro colores organizados en dos pares de colores complementarios que forman un rectángulo en el círculo cromático.
- Colores cuadrado: Utilizan cuatro colores separados a 90° en el círculo cromático para formar un cuadrado.

[22] [10]

Figura 4 .

Esquemas de colores



Nota: Elaboración propia

4.6.2. Tipografía

La tipografía es la representación visual de palabras y frases que combina principios de diseño gráfico [57]. Desde un punto de vista técnico, se refiere a la disposición de los caracteres en una página o diseño, lo que incluye la selección de la fuente, el tamaño, el estilo de las letras, el color y otros aspectos relacionados con la escritura.

La tipografía cuenta con tres elementos fundamentales: el kerning (espaciado entre letras individuales), el interlineado (espaciado entre líneas de texto) y el tracking (ajuste uniforme del espaciado entre caracteres) [9]. Estos elementos no solo mejoran la estética del diseño, sino que también afectan la percepción del espectador, comunicando un tono o estado de ánimo particular.

Elegir la tipografía adecuada es esencial para lograr el propósito del diseño. Resalta los valores de una marca o empresa y contribuye a la armonía del diseño, que implica equilibrio visual y continuidad. Una tipografía incorrecta puede resultar molesta para el espectador, por lo que es crucial considerar el propósito del diseño, el público objetivo y el medio en el que se mostrará el texto [9].

4.6.3. Prototipo

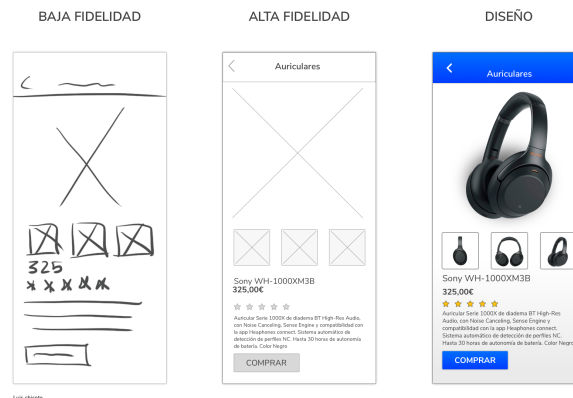
Un prototipo muestra cómo se verá y funcionará un producto de software real [21]. Generalmente, son construidos al inicio del desarrollo lo que permite obtener retroalimentación sobre el diseño y realizar pruebas de usuario. Son ideales para evaluar y validar ideas con usuarios y el equipo interno, facilitando así la prueba y ajuste de conceptos antes de la implementación. Existen tres categorías: baja fidelidad, fidelidad media y alta fidelidad [21].

Los prototipos de baja fidelidad son representaciones que se desarrollan de manera rápida, enfocadas en la estructura y la funcionalidad básica de un diseño sin centrarse en los detalles. Son utilizados al inicio del proceso de diseño para visualizar y probar ideas, facilitando la exploración y validación de conceptos en etapas tempranas [21]. Su simplicidad permite que sean creados y modificados de manera ágil promoviendo la experimentación rápida y la obtención de retroalimentación temprana de los usuarios. Habitualmente, se elaboran en papel mediante bocetos.

Los prototipos de fidelidad media ofrecen más detalle y realismo que los de baja fidelidad, proporcionando una visión más precisa del aspecto y las funcionalidades del producto. Con herramientas y software especializados para crear wireframes o maquetas interactivas, que actúan como planos detallados. Aunque representan un concepto inicial del producto, se centran en la funcionalidad, el flujo del usuario y la relación entre las pantallas con un estilo, color y gráficos mínimos [21].

Los prototipos de alta fidelidad son los más detallados y realistas, ya que incluyen todos los aspectos visuales y funcionales del producto final. Se desarrollan utilizando software y herramientas avanzadas; se emplean principalmente para validar el diseño mediante la simulación precisa de la experiencia del usuario. Pueden adoptar formas como páginas web interactivas o aplicaciones de software, ofreciendo una visión clara de cómo se verá y se sentirá el producto final, lo que permite realizar ajustes según sea necesario.

Figura 5 .
Categorías de prototipos ilustradas



Nota: Adaptada de Chicote (2019)

4.6.4. Experiencia de usuario

Según la norma ISO 9241-210, la experiencia del usuario (UX) se define como “las percepciones y respuestas de una persona resultantes del uso o uso anticipado de un producto, sistema o servicio” [34]. En términos simples, UX se refiere a cómo se siente una persona al usar un producto o servicio digital y a la impresión que le queda tras su uso.

Los diseñadores de UX abarcan todo el proceso relacionado con la adquisición e integración de un producto, incluyendo la investigación de usuarios, la creación de personajes, el diseño de wireframes y la elaboración de prototipos interactivos [34]. Aunque los diseñadores pueden influir significativamente en diversos aspectos del diseño, no tienen control total sobre las percepciones y respuestas de los usuarios.

Una experiencia de usuario de alta calidad se reconoce por la consideración minuciosa de todos los aspectos del producto, desde su diseño y usabilidad hasta el proceso integral de adquisición, utilización y resolución de problemas. Los diseñadores de UX se dedican a crear productos que no solo cumplan con su funcionalidad, sino que también sean agradables, eficientes y satisfactorios para el usuario [34].

4.7. Backend

El *backend* se refiere a la parte de una aplicación web que se ejecuta en el servidor y permite su funcionamiento. Entre sus funciones se incluyen recibir solicitudes, enviar y almacenar datos, así como tomar decisiones basadas en la información procesada.

Este consta de tres componentes principales: el servidor, que recibe las peticiones; la aplicación, que se ejecuta en el servidor, procesa las solicitudes, recupera y envía datos; y la base de datos, que organiza y conserva la información [?].

Existen diversos lenguajes de programación y gestores de bases de datos, tanto relacionales como no relacionales, para el desarrollo del backend. Esta amplia variedad de opciones permite elegir la tecnología más adecuada según las necesidades del proyecto.

4.7.1. Base de datos

Una base de datos es un conjunto de información estructurada la cual se almacena electrónicamente y puede contener diversos tipos de datos, como texto, números, imágenes, videos, entre otros. Su gestión está controlada por un sistema de gestión de bases de datos (DBMS), el cual permite almacenar, recuperar y editar los datos de manera eficiente [1]. El propósito principal de estas es gestionar grandes volúmenes de información, facilitando su almacenamiento, acceso y administración [36].

Base de datos relacionales

Una base de datos relacional es aquella que organiza la información en tablas, esta compuesta por filas y columnas, las cuales permiten establecer relaciones lógicas entre diferentes conjuntos de información. Cada fila representa un registro único identificado por una clave, mientras que las columnas contienen los atributos de dicho registro. Este modelo facilita la vinculación de datos entre tablas mediante claves que relacionan los puntos de datos, proporcionando una estructura clara y eficiente para gestionar y consultar la información [29].

Base de datos no relacionales

Las bases de datos no relacionales (NoSQL), almacenan la información en un formato no tabular lo que permite utilizar estructuras de datos como documentos [45]. Estas utilizan un esquema flexible el cual admite una gran variedad de datos sin estructurar, permitiendo que un documento pueda contener diferentes tipos de información en diversos formatos de manera detallada.

4.8. Frontend

El frontend se enfoca en los aspectos gráficos de un sitio web; es todo lo que el usuario ve o con lo que interactúa. Se compone de dos elementos principales: el diseño gráfico y la funcionalidad [19]. Ambos se desarrollan de manera independiente y, generalmente, se utilizan lenguajes como HTML, CSS y JavaScript para crear tanto elementos estáticos como interactivos.

4.8.1. HTML

El Lenguaje de Marcado de Hipertexto (HTML) es el código utilizado para estructurar, diseñar y desplegar una página web. Este lenguaje permite crear enlaces que conectan páginas web entre sí, ya sea dentro de un mismo sitio o diferentes [47]. Los elementos HTML se definen mediante etiquetas que el navegador interpreta para definir el texto y otros elementos de la página web.

4.8.2. CSS

Cascading Style Sheets (CSS) es el lenguaje utilizado para el formato y la apariencia de las páginas web, como colores, imágenes, fuentes y diseños [46]. Este se aplica a los elementos HTML para definir cómo se mostrará el contenido en diferentes dispositivos.

4.8.3. JavaScript

JavaScript es un lenguaje de programación que permite a los desarrolladores agregar interactividad, animaciones y otros elementos dinámicos a sitios web y aplicaciones web [42]. Esta permite manipular el contenido de la página de manera dinámica, responder a eventos del usuario y comunicarse con servidores a través de peticiones asincrónicas [48]. Además, su versatilidad y compatibilidad con diversas bibliotecas y frameworks lo convierten en una herramienta esencial para el desarrollo web.

4.8.4. Frameworks para frontend

Un framework es una colección de componentes de software reutilizables que facilita el desarrollo de nuevas aplicaciones. En el desarrollo web, los frameworks para frontend son herramientas esenciales que proporcionan los elementos necesarios para diseñar y definir el comportamiento de una interfaz la cual se encuentra orientada al usuario [2].

Hay una variedad de frameworks para frontend. A continuación, se mencionan algunos de los más populares en la actualidad.

Angular

Angular es un marco de trabajo de código abierto para aplicaciones web, mantenido por Google, que está diseñado para crear aplicaciones dinámicas e interactivas de manera eficiente [24]. Esta incluye un extenso conjunto de herramientas, API y bibliotecas que facilitan y optimizan el flujo de trabajo de desarrollo.

Vue

Vue es un marco de JavaScript para crear interfaces de usuario. Este proporciona un modelo de programación declarativo basado en componentes lo que lo vuelve flexible y accesible ayudando a desarrollar de manera eficiente interfaces de usuario de cualquier complejidad [68].

React

React es una biblioteca de JavaScript desarrollada por Facebook, utilizada para crear interfaces de usuario dinámicas e interactivas. Se basa en un enfoque declarativo y en componentes, lo que permite construir componentes de interfaz reutilizables y emplear un DOM virtual para optimizar el rendimiento [23].

4.9. Pruebas unitarias

Las pruebas unitarias son bloques de código diseñados para verificar el correcto funcionamiento de componentes individuales, normalmente estas se aplican a bloques de código pequeños como funciones, métodos, objetos u otras entidades [3]. Su objetivo principal es asegurar que cada unidad del software cumpla con los requisitos y funcione como se espera. Además, son esenciales para garantizar la calidad del software antes de su lanzamiento [8].

Existen varios frameworks para el desarrollo de pruebas unitarias en aplicaciones web, los cuales dependen del lenguaje de programación utilizado. A continuación, se presentan algunas librerías, cada una correspondiente a un framework descrito en el apartado 4.8.4.

4.9.1. Frameworks para pruebas unitarias

Jasmine

Jasmine es un framework de pruebas unitarias de código abierto basado en el comportamiento para JavaScript. Permite escribir pruebas con una sintaxis limpia y sencilla, sin depender de bibliotecas o frameworks externos. Es liviano y no tiene dependencias, lo que lo hace eficiente para probar código JavaScript [35]. Jasmine se utiliza principalmente en proyectos Angular, ya que está incluido de forma nativa.

Vitest

Vitest es un framework de pruebas basado en Vite, enfocado en la construcción y supervisión de aplicaciones web en JavaScript. Se caracteriza por ser rápido y minimalista, ya que requiere una configuración mínima. Además, es compatible con Jest y se integra fácilmente en aplicaciones Vue, lo que lo convierte en una opción ideal para proyectos que buscan simplicidad y eficiencia tanto en el desarrollo como en las pruebas [58].

Jest

Jest es un framework de pruebas de JavaScript enfocado en la simplicidad y eficiencia. Permite desarrollar pruebas con configuraciones mínimas, admite la ejecución de pruebas en paralelo para mejorar el rendimiento y ofrece funciones como pruebas instantáneas, mocks sencillos y cobertura de código integrada. Este framework es compatible con tecnologías como Babel, TypeScript, Node, React, Angular, y Vue, entre otras [37].

4.10. Control de versiones

El control de versiones es una práctica esencial para rastrear y gestionar los cambios en el código de software. Los sistemas de control de versiones son herramientas especializadas que permiten a los equipos de desarrollo administrar eficazmente las modificaciones en el código fuente a lo largo del tiempo [5].

Estos sistemas realizan un seguimiento detallado de cada cambio realizado en el código, ofreciendo la capacidad de regresar a versiones previas, comparar cambios a lo largo del tiempo, identificar al autor de las modificaciones y solucionar problemas de manera eficiente [28]. Además, permiten a los desarrolladores revertir errores y minimizar las interrupciones en el trabajo del equipo. Utilizan una base de datos especializada para proteger el código contra pérdidas causadas por errores humanos o incidentes imprevistos, y facilitan la resolución de conflictos derivados del trabajo concurrente [44].

4.10.1. GitHub

GitHub es el servicio de alojamiento más grande para repositorios de Git. Esta es una plataforma que permite almacenar, compartir y colaborar en la escritura de código [27].

Utilizar repositorios ofrece varias ventajas, entre las que se incluyen la capacidad de mostrar y compartir el trabajo, rastrear y gestionar los cambios a lo largo del tiempo, y facilitar la revisión y sugerencias de mejoras. Además, GitHub permite colaborar en proyectos compartidos sin que los cambios afecten el trabajo de otros hasta que estés listo para integrarlos.

4.10.2. Commits

Un commit en Git registra los cambios en uno o más archivos de la rama que se este trabajando. Cada commit recibe un identificador único que permite rastrear los cambios específicos, la fecha en que se realizaron y el autor de los mismos [26]. Al hacer un commit, se debe proporcionar un mensaje de confirmación que describa brevemente las modificaciones realizadas.

4.10.3. Branches

Una rama en un repositorio permite aislar el trabajo de desarrollo sin afectar otras ramas. Facilita el desarrollo de nuevas características, la corrección de errores o la experimentación en un área específica del repositorio, permitiendo trabajar de manera independiente y sin interferir con los cambios realizados por otros colaboradores. [25]

4.10.4. Pull Request

Un pull request es una solicitud para fusionar el código de una rama a otra. Permite a otros colaboradores revisar las modificaciones propuestas al visualizar las diferencias entre el contenido de la rama de origen y el de la rama de destino [27]. Además, permite agregar comentarios, participar en discusiones y, si es necesario, añadir commits adicionales.

4.11. Minimum viable product

El producto mínimo viable (MVP) es la versión más simple de un producto para lanzarse al mercado, con las características esenciales para satisfacer a los primeros usuarios. Su objetivo principal es validar el concepto del producto, minimizar los costes de desarrollo, recopilar retroalimentación y probar el producto en el mercado [52].

4.12. Integración continua

La integración continua una práctica de desarrollo de software que permite a los desarrolladores incorporar frecuentemente los cambios de código en un repositorio. Este proceso incluye la ejecución automatizada de compilaciones y pruebas para verificar la corrección del nuevo código antes de su integración final [7]

Su objetivo principal es detectar y solucionar errores de manera temprana, elevar la calidad del software y acortar el tiempo necesario para validar y desplegar nuevas actualizaciones. [6]

5.1. Investigación

Se realizó una investigación cuyo objetivo fue conocer las herramientas y sistemas actualmente utilizados en el ámbito de la salud. Durante esta investigación, se identificaron diversos sistemas implementados a nivel mundial; sin embargo, para Guatemala, solo algunos de ellos están disponibles, tales como “XClinics”, “Doctorcy” y “DriCloud”. Este análisis permitió extraer las características más relevantes de estos sistemas y dar una idea inicial sobre las posibles vistas a desarrollar.

Además, se llevaron a cabo entrevistas con médicos de distintas especialidades para validar la funcionalidad del programa que se buscaba desarrollar, así como para entender la organización y los procedimientos habituales dentro de una clínica. Estas entrevistas proporcionaron diversas perspectivas y aspectos que, al ser considerados, aportarían un valor adicional a la aplicación que se desea desarrollar. Al finalizar las entrevistas y la investigación sobre cada especialidad, se decidió enfocar el proyecto exclusivamente a la especialidad de pediatría.

Posteriormente, se buscó el apoyo de un profesional en el área para recibir asesoramiento y asegurar el cumplimiento de los lineamientos básicos requeridos para el desarrollo de la aplicación. Se optó por trabajar con el Doctor Juan Antonio Contreras, quien validó algunos conceptos técnicos, explicó el proceso de atención en su clínica, y facilitó una entrevista con su secretaria para comprender ciertos procesos administrativos. Esta orientación resultó valiosa para consolidar las ideas iniciales y agregar nuevas funcionalidades que aportarían valor a la aplicación.

5.2. Diseño de aplicación web

5.2.1. Paleta de colores

Para el diseño de la aplicación web, se comenzó con la selección de la gama de colores a utilizar. Para ello, se creó una paleta basada en colores análogos y se eligieron tonos neutros para el texto. En total, se seleccionaron seis colores distintos, cada uno destinado a cumplir una función específica dentro de la aplicación.

Se comenzó a formar la paleta mediante la selección del color primario para la aplicación. Considerando el sector al que va dirigida, se buscó un color unisex y atractivo. Por ello, se eligió un tono de naranja (#DC7829 en código hexadecimal), que permite transmitir una sensación de originalidad, optimismo, energía y juventud.

Figura 6 .

Color primario en hexadecimal



Posteriormente, se definieron los colores secundarios a partir de tonos análogos, empleando la herramienta web Adobe Color para explorar diferentes opciones de tonalidades y ajustar la luz e intensidad. Con el objetivo de mantener una apariencia armoniosa, unificada y ordenada, se seleccionaron los colores #CD5B2E y #AE3C2D como complementos al color primario.

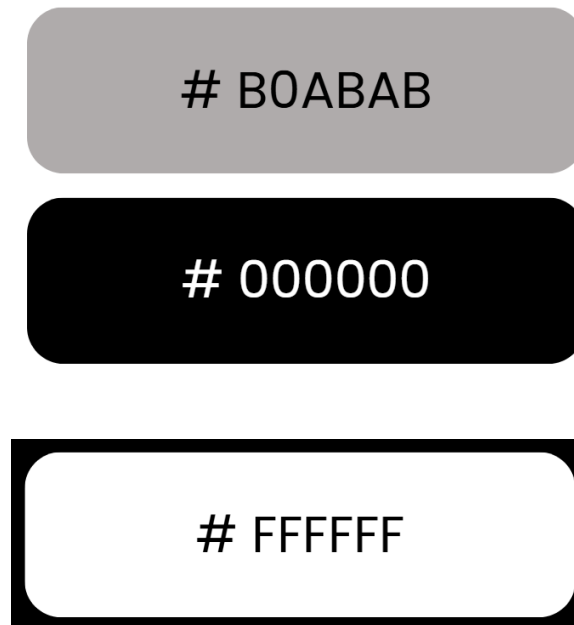
Figura 7 .

Colores secundarios en hexadecimal



Para el texto, se optó por colores neutros que garantizaran la máxima legibilidad y un adecuado contraste con la paleta elegida. Se usaron los colores #FFFFFF (blanco), #000000 (negro) y #B0ABAB (gris oscuro). Cabe destacar que, en ciertos casos, se emplearon los colores secundarios para resaltar textos importantes dentro de las vistas, logrando así un equilibrio visual que contribuye a una experiencia de usuario óptima.

Figura 8 .
Colores neutros en hexadecimal



5.2.2. Selección de fuentes

Para la selección de fuentes, se consideró el grupo objetivo y su rango de edades, eligiendo tipografías de letra de molde para asegurar su legibilidad. Se utilizó la herramienta Google Fonts para explorar la variedad de fuentes disponibles y visualizar cómo se verían en la web.

La fuente seleccionada para los textos fue Poppins Sans-serif, caracterizada por su estilo moderno, minimalista y sus trazos uniformes, lo cual facilitó su integración con el contraste de la aplicación. Para los títulos, se optó por la fuente Della Respira, que permite destacar los encabezados sobre el resto del texto en la aplicación web.

Figura 9 .
Tipografía para el texto del cuerpo

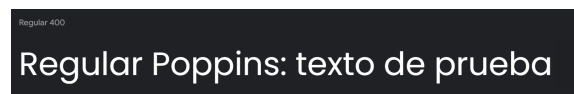
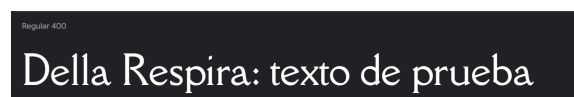


Figura 10 .
Tipografía para títulos



5.3. Prototipado

Con base en el diseño establecido previamente, se creó un prototipo utilizando la aplicación web Figma, especializada en el desarrollo de interfaces gráficas. Este prototipo sirvió para llevar a cabo pruebas de concepto. Es importante destacar que, con el fin de simular un caso real y ofrecer una representación más tangible del producto final, se diseñó un logo para una clínica ficticia. A continuación, se presentan las pantallas desarrolladas.

5.3.1. Pantallas del prototipo

Logo

Se diseñó un logo llamativo y colorido que hacía referencia a niños y niñas. Se buscó que los colores utilizados en el logo se combinaran armónicamente con la paleta establecida previamente.

La tipografía elegida fue divertida, reflejando el espíritu infantil, y se incluyó un estetoscopio para aludir al ámbito de la salud.

Figura 11 .

Identidad visual de la clínica



Nota: Elaboración propia

Inicio de sesión

Esta pantalla solicita a los doctores y al personal que ingresen sus credenciales para acceder a la aplicación.

Figura 12 .
Mockup de pantalla de inicio de sesión

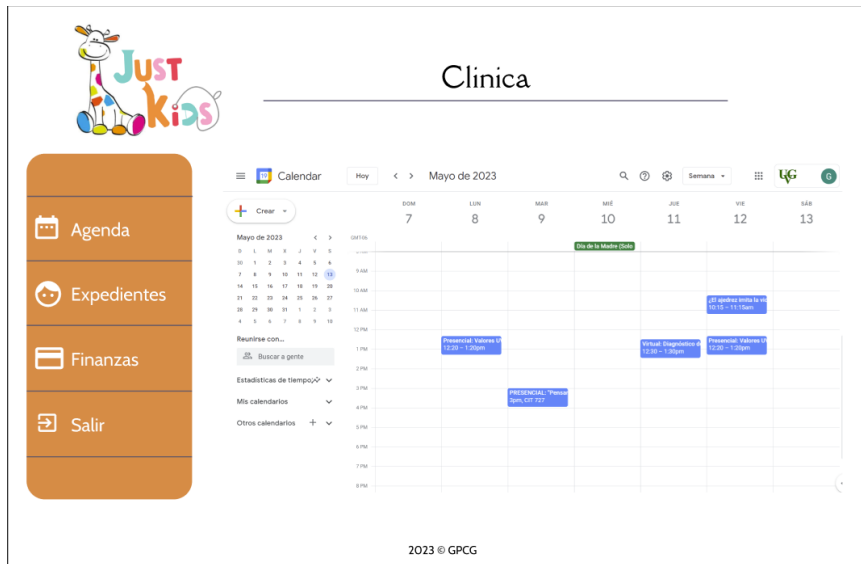


Nota: Elaboración propia

Agenda

Esta es la pantalla principal de la aplicación, donde se pueden ingresar, eliminar y visualizar las citas que el doctor tiene programadas para la semana.

Figura 13 .
Mockup de pantalla para agenda



Nota: Elaboración propia

Ingresar expediente

Esta pantalla presenta la ficha médica que los doctores utilizan para registrar datos importantes

de cada paciente. En ella, se incluyen secciones dedicadas a la información personal del paciente, antecedentes médicos, entre otros.

Figura 14 .
Mockup de pantalla para ingresar expediente

JUST KIDS Clínica

< Nuevo expediente

Datos personales

Ingrese el nombre del paciente Ingrese los apellidos del paciente

Ingrese sexo Ingrese edad Ingrese fecha de nacimiento

Descripción de residencia

Datos de los encargados

Ingrese nombre de la madre Ingrese los apellidos de la madre

Ingrese edad de la madre Ingrese ocupación de la madre

Ingrese nombre del padre Ingrese los apellidos del padre

Ingrese edad del padre Ingrese ocupación del padre

Datos del seguro

¿Cuenta con seguro?

Sí No

Ingrese el nombre del seguro Ingrese en No. de póliza

Ingrese número de CUI del paciente Ingrese DPI del encargado

Antecedentes personales

Tipo de parto

PES CSIP Ingrese lugar donde se le atendió Complicaciones durante el parto

Ingrese peso al nacer Ingrese talla del paciente G P C ab

Ingrese CC Ingrese HV Ingrese HM

Vacunas

BCG:

Hepatitis B:

Pentavalente: 1ra 2da 3ra 1ra Ref 2da Ref

Neumococo: 1ra 2da 3ra

Influenza: 1ra 2da 3ra

Rotavirus: 1ra

Hepatitis A: 1ra 2da

Varicela: 1ra

Covid 19: 1ra 2da 3ra

Antecedentes médicos

Antecedentes médicos Antecedentes quirúrgicos Antecedentes alérgicos

Antecedentes alimenticios

Alimentación actual Edad de Ab lactación Marlas

Ingresar ficha

2023 © GPCG

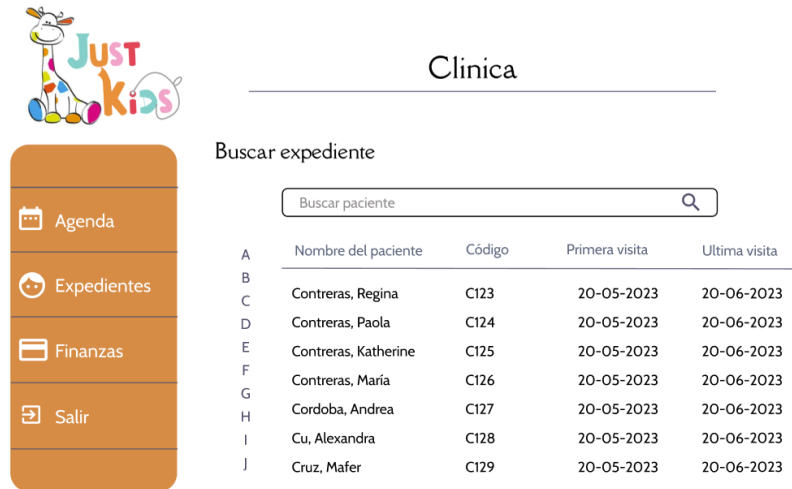
Nota: Elaboración propia

Buscar expediente

Esta pantalla lista en orden alfabético todos los pacientes que han sido ingresados en la aplicación. Además, incluye una barra de búsqueda que permite localizar rápidamente a un paciente en específico.

Figura 15 .

Mockup de pantalla para buscar expediente



2023 © GPCG

Nota: Elaboración propia

Historial médico

Esta pantalla muestra un breve resumen del historial de un paciente en específico. Incluye información clave como las visitas anteriores, diagnósticos y tratamientos recibidos.

Figura 16 .
Mockup de pantalla para historial médico



Nota: Elaboración propia

Consultar expediente

Esta pantalla muestra la ficha médica completa con los datos de un paciente en específico.

Figura 17 .
Mockup de pantalla para consultar expediente

JUST Kids Clínica

Arevalo, Carol
 Ficha médica Historial Recetar

Datos personales

Nombre: Carol Rene Apellido: Arevalo Gómez
 Sexo: Femenino Edad: 10 Fecha de nacimiento: 12-06-2011
 Dirección: Sex 2-09 zona 1 Santa Catalina Píñula

Datos de los encargados

Nombre de la madre: Ximena Apellido de la madre: Gómez
 Edad de la madre: 30 Ocupación de la madre: Secretaria
 Nombre del padre: Fernando Apellido del padre: Arevalo
 Edad del padre: 33 Ocupación del padre: Perito Contador

Datos del seguro

¿Cuenta con seguro?
 Si No

Nombre del seguro: La Ceiba Numero de póliza: 12378901
 Número de CUI: 89021344101 DPI del encargado: 12345678901

Antecedentes personales

Tipo de parto
 PES CSTP Lugar: IGSS Complicaciones: NA

Peso al nacer: 7 lb Talla: 53 cm G C P ab

CC: 45 HV: 35 HM: 54

Vacunas

BCG:
 Hepatitis B:
 Pentavalente: 1ra 2 da 3ra 1ra Ref 2 da Ref
 Neumococo: 1ra 2 da 3ra
 Influenza: 1ra 2 da 3ra
 Rotavirus: 1ra
 Hepatitis A: 1ra 2 da
 Varicela: 1ra
 Covid-19: 1ra 2 da 3ra

Antecedentes médicos

Antecedentes médicos: NA Antecedentes quirúrgicos: NA Antecedentes alérgicos: NA

Antecedentes alimenticios

Alimentación actual: NA Edad de ab lactación: NA Morias: NA

Editar ficha Actualizar ficha Cita actual

Nota: Elaboración propia

Modificar expediente

Esta pantalla permite a los doctores modificar o actualizar los campos de la ficha médica que previamente fueron completados.

Figura 18 .
 Mockup de pantalla para modificar expediente

JUST Kids Clínica

Arevalo, Carol

Ficha médica Historial Recetar

Datos personales

Carol Riera Arevalo Gómez

Femenino 12 20-06-2012

Sav 2-03 zona 1 Santa Catarina Pinula

Datos de los encargados

Juleta Gómez

30 Secretaria

José Arevalo

33 Perito Contador

Datos del seguro

¿Cuenta con seguro?

Sí No

La Caba 1237899H

89021344101 12345698101

Antecedentes personales

Tipo de parto

PES CSTP Ingrese lugar donde se le atendió Complicaciones durante el parto

Ingrese peso al nacer Ingrese talla del paciente G P C ab

Ingrese CC Ingrese HIV Ingrese HM

Vacunas

BCG:

Hepatitis B:

Pentavalente: 1ra 2 da 3ra 1 ra Ref 2 da Ref.

Neumococo: 1ra 2 da 3ra

Influenza: 1ra 2 da 3ra

Rotavirus: 1ra

Hepatitis A: 1ra 2 da

Váricela: 1ra

Covid 19: 1ra 2 da 3ra

Antecedentes médicos

Antecedentes médicos Antecedentes quirúrgicos Antecedentes alérgicos

Antecedentes alimenticios

Alimentación actual Edad de ABlactación Marías

Editar ficha Actualizar ficha Cita actual

Nota: Elaboración propia

Recetar

Esta pantalla permite al doctor recetar medicamentos al paciente que está evaluando. En esta

sección, el médico puede ingresar los medicamentos, indicar las dosis y especificar la duración del tratamiento.

Figura 19 .
Mockup de pantalla para recetar



Nota: Elaboración propia

Ingresar cita actual

En esta pantalla, se pueden ingresar datos relevantes durante la evaluación que se le está realizando al paciente. Este espacio permite a los doctores registrar observaciones, síntomas, y cualquier información importante que pueda ser útil para el diagnóstico.

Figura 20 .
Mockup de pantalla para ingresar cita actual



Logo: JUST Kids

Clinica " Los Angeles "

Cita Actual
 Kathemine Andrea Contreas

Datos Generales

Fecha Edad Alimentación

Examen Fisio E/F

Signos vitales

FC Temperatura FR

P/A SAT Glucometro

Peso Talla POD

Evaluación

Diagnostico:

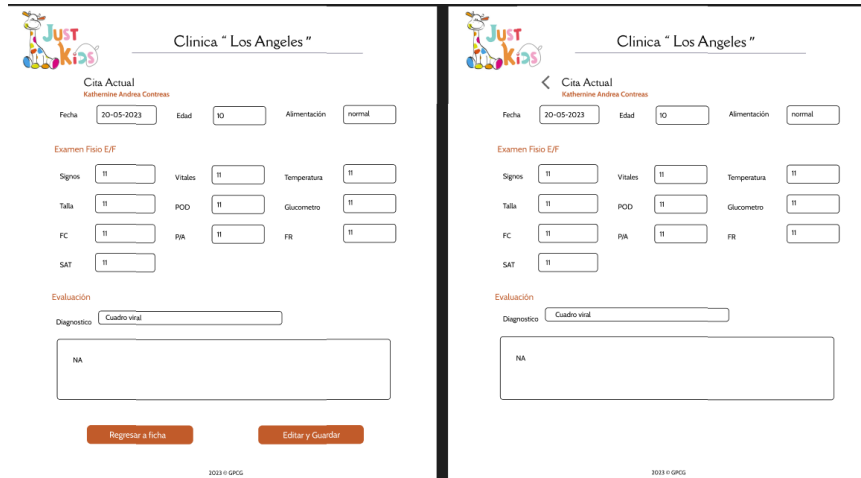
2023 © GPCG

Nota: Elaboración propia

Consultar y Modificar cita actual

En esta pantalla, se pueden observar los datos previamente ingresados de la cita actual. Además, se permite modificar y/o agregar información relevante a la consulta.

Figura 21 .
Mockup de pantalla para consultar y modificar cita actual



Nota: Elaboración propia

Efectuar un cobro

Esta pantalla permite al personal médico registrar un cobro realizado. En esta sección, se ingresan el nombre del paciente, el motivo, el monto a pagar y la información sobre si el paciente cuenta con un seguro.

Figura 22 .
Mockup de pantalla para ingresar un cobro



Nota: Elaboración propia

Caja del día

Esta pantalla permite al doctor observar todos los movimientos realizados durante un día de manera detallada. Aquí se presenta un desglose de la cantidad pagada, especificando si el paciente cuenta o no con un seguro médico, así como el método de pago utilizado.

Figura 23 .
Mockup de pantalla para la gestión de caja del día



Nota: Elaboración propia

Registro de finanzas

Esta pantalla presenta un detalle histórico consolidado que permite al doctor visualizar la cantidad ingresada según cada método de pago y la fecha correspondiente.

Figura 24 .
Mockup de pantalla para registro de finanzas



Nota: Elaboración propia

Configuraciones

Esta pantalla está diseñada para permitir al personal médico gestionar y configurar la información relacionada con las aseguradoras.

Figura 25 .
Mockup de pantalla para gestión de aseguradoras



Nota: Elaboración propia

Figura 26 .
Mockup de pantalla para aseguradoras



Nota: Elaboración propia

5.3.2. Retroalimentación del prototipo

Para asegurar que el prototipo cumpliera con los requisitos esenciales para la gestión de una clínica, se llevó a cabo una prueba de concepto con el Dr. Juan Antonio Contreras y su equipo. Durante esta evaluación, el personal proporcionó observaciones funcionales y sugerencias de mejora sobre algunas de las pantallas, las cuales se detallan a continuación. **Buscar expediente** En esta pantalla, uno de los usuarios comentó que sería conveniente contar con un botón para eliminar expedientes en caso de duplicados o en caso de que el paciente deje de asistir a la clínica. Además, sugirió que los botones del alfabeto a un lado de la pantalla funcionaran únicamente con el código del paciente y no con el nombre, para asegurar que solo se muestren expedientes cuyo apellido comience con la letra seleccionada.

Expedientes médicos En cuanto a esta pantalla, el doctor observó que los campos considerados eran adecuados para sus necesidades. No obstante, recomendó basarse en los lineamientos de la Organización Mundial de la Salud (OMS) para el cumplimiento de sus estándares y, al mismo tiempo, incluir los elementos más relevantes para hacer la aplicación funcional para cualquier pediatra.

Cita actual Para esta pantalla, se sugirió agregar más campos que permitan almacenar un análisis completo de cada consulta. Entre sus recomendaciones, el doctor propuso que el área de examen físico incluya un espacio para notas, así como la incorporación de apartados para reconsultas y tratamientos.

Receta médica En esta pantalla, el médico sugirió que, al acceder, sería funcional que se registre automáticamente la fecha y el nombre del paciente evaluado, de modo que solo sea necesario ingresar el medicamento y la dosis. Además, recomendó diseñar la receta médica de forma más sencilla o estructurarla de manera que permita identificar la información con mayor facilidad y sea, visualmente atractiva.

Ingresar un cobro Para esta pantalla, se sugirió implementar campos desplegables en la sección de seguros y métodos de pago, permitiendo seleccionar entre los disponibles en la clínica. Esta modificación ayudaría a garantizar una estructura homogénea en toda la aplicación.

Configuraciones En esta pantalla, los usuarios observaron que no existía una forma de acceder a ella, ya que no se encontraba detallada en el menú. Asimismo, recomendaron evaluar opciones que podrían beneficiar a la clínica, como incluir la información general de la clínica, visualizar los usuarios disponibles, realizar modificaciones en la ficha médica, entre otros.

Adicionales En términos generales, ambos usuarios indicaron que la aplicación cumplía con las funcionalidades necesarias para la clínica. Sin embargo, recomendaron añadir más color para evitar que algunas pantallas se vean vacías o poco llamativas. También sugirieron evaluar la implementación de notificaciones que confirmen cuando una acción ha sido completada o está en proceso, evitando así la duplicación de procesos o datos mediante múltiples clics.

5.4. Selección de tecnologías

Para seleccionar las tecnologías que cumplieran con los requisitos del proyecto y permitieran desarrollar todas las funcionalidades dentro del tiempo establecido, se realizó una investigación detallada de cada tecnología considerada. A continuación, se presentan los aspectos clave que influyeron en su selección:

Motor de base de datos

Tabla 1: Ventajas y desventajas de Firebase

Ventajas	Desventajas
Al ser un producto de Google, Firebase se integra fácilmente con otras herramientas de Google, facilitando el desarrollo, análisis y monitoreo de la aplicación.	Las bases de datos de Firebase, como Firestore y Realtime Database, no son adecuadas para proyectos que requieran consultas avanzadas o complejas, lo que limita su uso en algunos contextos.
Permite almacenar y sincronizar datos en tiempo real, lo que garantiza que los cambios se reflejen de inmediato en todas las aplicaciones.	Migrar datos desde Firebase a otra plataforma puede ser complicado, lo que representa una dificultad si se planea un cambio de proveedor en el futuro.
Ofrece una solución fácil de implementar para la autenticación de usuarios, compatible con varios métodos (correo, redes sociales, etc.), ahorrando tiempo y esfuerzo en el desarrollo.	Al usar Firebase, se crea una dependencia de la plataforma y las políticas de Google, lo cual puede ser un riesgo si las condiciones de uso o los precios cambian.
Firebase ofrece múltiples servicios que pueden integrarse fácilmente en la aplicación, facilitando el desarrollo de funcionalidades adicionales sin complicaciones.	con

(Elaboración propia)

Tabla 2: Ventajas y desventajas de MongoDB

Ventajas	Desventajas
Idónea para desarrolladores con limitados recursos de computación, ya que se puede descargar en cualquier servidor u ordenador.	Aunque es utilizada por varios desarrolladores actualmente, la tecnología que utiliza es demasiado nueva.
En comparación con otras bases de datos NoSQL, la documentación de soporte que ofrece MongoDB es bastante extensa y detallada, lo que facilita la comprensión y el uso efectivo de la plataforma.	Al tener un formato BSON ocupa más espacio que otros formatos de almacenamiento de datos, consumiendo en mayor escala la memoria.
MongoDB no es completamente gratuito, ya que se requiere un pago de licencia para acceder al soporte, no obstante su uso y acceso inicial no tienen ningún costo, lo que la convierte en una opción económica para muchos desarrolladores.	Esta requiere un enfoque diferente para gestionar consultas que involucren múltiples fuentes de datos, lo que puede dificultar su implementación en ciertas aplicaciones.
Todas las funciones y componentes de JavaScript se ejecutan en su totalidad, lo que permite un rendimiento óptimo y una integración fluida de la lógica en las aplicaciones.	

(Elaboración propia)

Se optó por utilizar una base de datos no relacional debido a que el proyecto buscaba ser personalizable, lo cual implicaba la necesidad de tablas con campos variables. Por ello, el dinamismo que ofrecen las bases de datos no relacionales resultaba adecuado para los requisitos planteados.

En cuanto al motor de base de datos, se decidió usar Firebase, ya que esta herramienta cuenta

con múltiples servicios integrados, lo cual facilitó el desarrollo de la aplicación. Además, Firebase ofrece una infraestructura escalable que permite manejar el crecimiento de usuarios y datos sin la necesidad de realizar ajustes manuales, y proporciona seguridad integrada a través de reglas de acceso personalizables, lo que garantiza una gestión segura de los datos. Finalmente, Firebase ofrecía una versión gratuita de la mayoría de sus servicios, lo cual fue ventajoso en la etapa de desarrollo y pruebas.

Frontend

Tabla 3: Ventajas y desventajas de React

Ventajas	Desventajas
React utiliza una arquitectura de componentes que permite dividir la aplicación en partes más pequeñas, facilitando su comprensión y modificación.	La curva de aprendizaje de React es más pronunciada en comparación con otros frameworks, ya que se requieren conocimientos previos para integrarlo con el framework Modelo-Vista-Controlador (MVC).
React cuenta con una variedad de herramientas y extensiones que ayudan a seleccionar las más adecuadas para las necesidades específicas de cada proyecto, optimizando la creación y el mantenimiento del código.	En proyectos de gran tamaño, el peso de la aplicación puede aumentar considerablemente, afectando la velocidad de carga y el rendimiento general.
Posee una comunidad activa y extensa de desarrolladores que contribuyen a su soporte, facilitando el aprendizaje y la resolución de problemas.	React requiere un código base que, en muchas ocasiones, es repetitivo, lo que puede incrementar el tiempo de desarrollo.
Su arquitectura flexible y escalable permite adoptar y ampliar la aplicación en función de los requisitos del proyecto.	

(Elaboración propia)

Tabla 4: Ventajas y desventajas de Vue

Ventajas	Desventajas
Vue cuenta con un buen rendimiento ya que tiene un peso mínimo, lo que hace que las aplicaciones sean rápidas y responsivas.	Vue cuenta con una comunidad más pequeña en comparación con otros frameworks, lo que reduce la disponibilidad de recursos y tutoriales, haciendo más difícil encontrar soporte extenso.
Vue ofrece una sintaxis muy similar a HTML, lo cual mejora la legibilidad y el mantenimiento del código.	Al ser un framework muy nuevo no cuenta con el respaldo financiero de empresas como Facebook o Google, por lo que no ofrece el mismo nivel de confiabilidad en sus actualizaciones en comparación a frameworks más consolidados. Esta limitación podría afectar en su capacidad para resolver problemas de manera ágil.
La curva de aprendizaje es suave puesto a que cuenta con una sintaxis sencilla e intuitiva haciendo que esta sea accesible para desarrolladores con distintos niveles de experiencia.	Vue ofrece herramientas más limitadas en comparación con otros frameworks, lo que hace que la curva de aprendizaje para sus características avanzadas sea más pronunciada. Aunque es fácil comenzar a usarlo, dominar sus funciones avanzadas puede requerir más tiempo.
Se enfoca en la creación eficiente de interfaces de usuario mediante el uso de herramientas que permiten que estas sean interactivas y dinámicas sin tener que manipular el DOM constantemente.	

(Elaboración propia)

Tabla 5: Ventajas y desventajas de Angular

Ventajas	Desventajas
Este permite la reutilización de código, ya que tiene componentes que encapsulan la lógica y presentación de códigos existentes, admitiendo la reutilización y personalización de los mismos en las aplicaciones.	Debido al framework y sus requisitos, el tamaño de los archivos generados tienden a ser grandes, lo que prolonga el tiempo de descarga e inicio de la aplicación.
Angular cuenta con una gran comunidad, lo que permite que la documentación en línea sea lo suficientemente extensa para brindar orientación y soporte a los nuevos usuarios en la resolución de problemas.	Al ser un framework en constante actualización, tiende a afectar la compatibilidad de la aplicación con versiones anteriores, especialmente en proyectos de gran escala.
Angular sigue un patrón de arquitectura MVC, lo cual garantiza que las aplicaciones tengan una estructura clara y modular, permitiendo a los desarrolladores trabajar de manera más rápida y escalable.	La curva de aprendizaje es alta ya que al tener una estructura y serie de propiedades complejas, habituarse a su framework y ecosistema puede resultar dificultoso.

(Elaboración propia)

Se optó por utilizar Vue para el frontend debido a su estructura clara, su baja curva de aprendizaje y su capacidad para simplificar el manejo de estados en comparación con otros frameworks, lo cual facilitó la sincronización de datos entre componentes. Además, Vue optimiza automáticamente

las dependencias reactivas, lo que reduce las actualizaciones innecesarias en el *DOM* y mejora el rendimiento de la aplicación. Además, su amplia variedad de librerías, desarrolladas por terceros y la comunidad de Vue, permitió aprovechar al máximo sus capacidades y facilitó el diseño de componentes sin configuraciones complejas, acelerando el desarrollo.

Pruebas unitarias

Tabla 6: Ventajas y desventajas de Jest

Ventajas	Desventajas
Jest puede utilizarse en diferentes entornos, como React, Vue.js, Angular y Node.js, lo que lo convierte en una herramienta versátil para pruebas en diversas plataformas.	Aunque Jest ofrece un rendimiento rápido gracias a su ejecución paralela, puede requerir una cantidad considerable de recursos del sistema durante las pruebas
Permite capturar y almacenar resultados de pruebas como "snapshots", facilitando la comparación con ejecuciones posteriores para detectar cambios inesperados y asegurar la calidad del código.	Debido a su evolución constante, Jest puede experimentar cambios en su API que afecten el código existente, lo que puede implicar un esfuerzo adicional para adaptar las pruebas previamente programadas.
proporciona una sintaxis clara y expresiva para escribir pruebas, lo que facilita su lectura y simplifica el proceso de escritura, depuración y mantenimiento.	Las características avanzadas que ofrece este framework hacen que la curva de aprendizaje para los nuevos usuarios sea más pronunciada, lo que requiere tiempo para dominarlo.
Se integra fácilmente con herramientas comunes como Babel, Webpack y ESLint, lo que facilita la incorporación de este en el flujo de trabajo existente.	

(Elaboración propia)

Tabla 7: Ventajas y desventajas de Jasmine

Ventajas	Desventajas
Cuenta con una sintaxis fácil de entender, lo que la hace accesible y adecuada para desarrolladores con distintos niveles de experiencia.	Esta no es tan flexible como otros marcos de prueba, lo que dificulta su adaptación en proyectos con requisitos específicos.
Admite pruebas asíncronas, espías, simulacros y emparejadores personalizados, lo que permite realizar pruebas más complejas.	Presenta restricciones en algunas de sus funciones integradas, lo cual la vuelve insuficiente para escenarios de prueba más complejos.
Proporciona una forma clara y organizada de estructurar las pruebas, lo que facilita el mantenimiento y la actualización de los conjunto	Jasmine no incluye soporte para pruebas instantáneas, lo que limita la capacidad de verificar rápidamente la salida de los componentes y detectar cambios.

(Elaboración propia)

Tabla 8: Ventajas y desventajas de Vitest

Ventajas	Desventajas
Permite personalizar las configuraciones de las pruebas según los requisitos de cada proyecto, facilitando así su integración a diversas necesidades.	Su comunidad aun es pequeña, lo que significa que hay menos recursos, tutoriales y complementos disponibles en comparación con otros frameworks.
Se integra con diversas herramientas y utilidades de prueba, lo que proporciona una experiencia de desarrollo y prueba fluida y cohesiva	Vitest requiere Vite como dependencia para funcionar en su totalidad; sin embargo, en desarrollos que no lo utilizan, puede carecer de ciertas características o capacidades.
Proporciona un proceso de configuración simple y requisitos mínimos, lo que permite iniciar el desarrollo de pruebas de manera rápida.	Esta carece de algunas funcionalidades avanzadas que están disponibles en otros frameworks, lo que podría limitar su uso en escenarios de prueba más complejos.

(Elaboración propia)

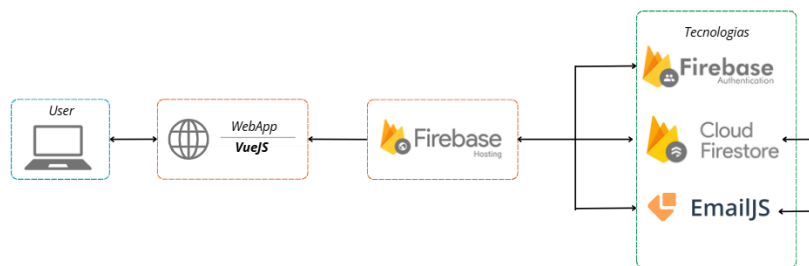
Para las pruebas unitarias, se optó por utilizar Jest debido a su robustez como framework, lo cual permitía realizar pruebas más completas y exhaustivas. Una de sus principales ventajas fue la capacidad de simular y mantener ciertos datos en memoria, lo que permitió ejecutar pruebas complejas de forma controlada y replicable. Además, Jest ofrecía flexibilidad para integrarse con distintos frameworks de frontend y contaba con una documentación detallada, lo cual facilitó su configuración y uso durante el desarrollo.

5.5. Arquitectura de la aplicación web

La aplicación web emplea una arquitectura híbrida en la nube, ya que combina el procesamiento en el cliente con servicios en la nube, lo que permite una distribución eficiente de las cargas de trabajo. En el lado del cliente, la aplicación está desarrollada en Vue.js, lo que ofrece la posibilidad que gran parte de la lógica, como la navegación y la validación de datos, se ejecute directamente en el navegador del usuario, optimizando así el rendimiento y la experiencia de uso. Por otro lado, el backend se basa en un enfoque serverless, utilizando Firebase Authentication para la gestión de usuarios, Cloud Firestore para el almacenamiento y sincronización de datos, y Firebase Hosting para el despliegue de la aplicación. Además, se integra EmailJS para el envío de correos electrónicos sin requerir infraestructura adicional. Esta arquitectura permite una mayor escalabilidad, disponibilidad y reducción de costos operativos, eliminando la necesidad de servidores dedicados y facilitando el mantenimiento del sistema.

Figura 27 .

Representación visual de la arquitectura y componentes de la aplicación web



Nota: Elaboración propia

5.6. Desarrollo

Para el desarrollo de la aplicación, se comenzó elaborando una lista detallada de las vistas requeridas y el orden en el que debían desarrollarse. A partir de esta lista, se identificaron los elementos recurrentes, lo que permitió definir un listado de componentes a implementar.

Una vez definidos los componentes y establecido el orden de desarrollo, se creó un repositorio en *GitHub* y un proyecto en *Firebase*. Posteriormente, se configuró *Firestore* en el proyecto de *Firebase* para el almacenamiento de datos en tiempo real. Luego, se clonó el repositorio y se instaló *Webpack* para gestionar las dependencias del proyecto, lo que facilitó una estructura organizada y coherente de los archivos.

Finalizada la configuración de *Webpack*, se procedió a instalar las principales librerías necesarias: *Firebase*, *Vue/CLI*, *ESLint* y *Vue Template Compiler*. Con el entorno de desarrollo listo, se inició la programación de los componentes, creando documentos independientes y nombrándolos de manera clara y sencilla, siguiendo las reglas de diseño establecidas. La estructura de cada archivo incluía *HTML*, seguido de *CSS* y, finalmente, el *Script* correspondiente. Según los requisitos de cada componente, se importaron librerías como *Fontawesome*, para iconos, y *Google Fonts*, para las fuentes definidas en el diseño.

Al finalizar los componentes, se comenzó a desarrollar cada una de las vistas del prototipo. Para ello, se crearon ramas independientes en el repositorio principal, permitiendo manejar cada sección de manera organizada y aislada. En el desarrollo de las vistas, se importaron los componentes necesarios, y con estos elementos ya disponibles, se procedió a estructurar cada vista utilizando *HTML* y los componentes previamente creados. Una vez estructurada la vista, se aplicaron estilos *CSS* para recrear el diseño definido en el prototipo.

Con la interfaz visual completada, se procedió a implementar la lógica de cada vista, asegurando que cada una cumpliera con las funcionalidades requeridas. Para gestionar la manipulación de datos, se utilizaron las funciones proporcionadas por *Firebase*, especialmente las de *Firestore*, lo que facilitó las operaciones de escritura, lectura, actualización y eliminación de datos en tiempo real. Esto permitió que las vistas interactuaran de manera dinámica con la base de datos, reflejando cualquier cambio al instante.

Es importante señalar que, en algunas vistas, se realizaron modificaciones significativas respecto al diseño original del prototipo o a la lógica prevista. A continuación, se detallan los casos relevantes.

Inicio de sesión

Para esta vista se utilizó el servicio de autenticación de Firebase, el cual resultó fundamental para asegurar que solo usuarios registrados y autorizados pudieran acceder a la aplicación. Este servicio permitió implementar un sistema de inicio de sesión que exigía a los usuarios autenticarse antes de poder visualizar o interactuar con las funcionalidades principales de la aplicación.

Restablecer contraseña

Esta vista no había sido diseñada en la fase inicial, pero surgió como respuesta a la necesidad de ofrecer una opción para cambiar la contraseña, según el feedback recibido. La funcionalidad incluye un sistema de verificación de dos factores, que consiste en dos códigos: uno definido por el usuario y otro generado aleatoriamente y enviado al correo electrónico del solicitante. Este enfoque permitió aumentar la seguridad, evitando que cualquier persona que conociera el correo electrónico o el nombre de usuario pudiera cambiar la contraseña sin autorización.

Para enviar el código aleatorio, se utilizó la librería *EmailJS*, la cual permitía enviar un correo electrónico con un diseño personalizado. Se optó por *EmailJS* en lugar del servicio de correo de *Firebase*, ya que este último no permitía modificar el diseño de sus plantillas predefinidas, lo cual interfería con la estética previamente establecida para la aplicación. Además, el uso de *EmailJS* permitió implementar el sistema de autenticación adicional, reforzando la seguridad en el proceso de cambio de contraseña.

Agenda

Inicialmente, se consideró utilizar Google Calendar mediante su API para esta vista. Sin embargo, se optó por emplear *Vue Cal*, ya que el uso de Google Calendar requería un inicio de sesión adicional. Aunque la autenticación de Firebase ya permitía el inicio de sesión con Google, el acceso al calendario exigía una nueva autenticación, lo cual afectaba la practicidad y fluidez de la aplicación. Esta doble autenticación pudo deberse al uso de la versión gratuita de Google Calendar.

La implementación de *Vue Cal* resultó ventajosa, ya que permitió adaptar el calendario a la estética de la aplicación. Se logró personalizar los colores y el estilo visual, así como bloquear días específicos y ofrecer diferentes vistas del calendario (como vista mensual, semanal o diaria).

Para gestionar los eventos, se creó una colección específica en *Firestore*. Esta colección almacenaba la información de los eventos en tiempo real, permitiendo que los datos del calendario se actualizaran dinámicamente. Al ser una biblioteca independiente, *Vue Cal* no incluía soporte para la persistencia de datos, por lo que *Firestore* proporcionó una solución eficaz para el almacenamiento y sincronización de eventos.

Expedientes

Para realizar esta vista, se llevó a cabo una investigación previa para identificar los campos necesarios para desarrollar una ficha médica. Se consultó información de la Organización Mundial de la Salud (OMS), ya que esta institución proporciona una lista de requisitos mínimos que debe contener una ficha pediátrica. Además, se añadió un apartado donde el doctor pudo ingresar campos adicionales no preestablecidos, permitiendo así que este continuara trabajando de la manera habitual.

Ajustes

Inicialmente, esta vista fue diseñada exclusivamente para la gestión de seguros. Sin embargo, durante el desarrollo, se identificaron elementos adicionales que resultarían útiles para la administración general de la clínica. Entre estos elementos se implementaron la administración de usuarios, la creación de nuevos usuarios, el manejo de información general de la clínica, la gestión de aseguradoras y la configuración de los campos de los expedientes. Estas mejoras permitieron que la aplicación alcanzara un nivel de personalización y funcionalidad acorde a las necesidades de diferentes clínica

Vuex y Localstorage

El uso de Vuex para el manejo de estados y de localStorage para el almacenamiento local de información resultó de gran utilidad en la gestión de roles entre vistas. Esto permitió recargar la página sin perder el rol activo en cada sesión. Además, Vuex facilitó el uso de getters y setters, lo que simplificó el traspaso de información entre vistas sin necesidad de almacenar dichos datos en la base de datos.

5.7. Control de versiones

Para el control de versiones del proyecto, se utilizó la herramienta GitHub. Se estableció una rama principal, denominada "main", que contenía la versión más actualizada y unificada del proyecto. Para cada característica o vista de la aplicación, se creó una nueva rama a partir de la rama principal. A cada una de estas ramas se le realizaron commits, los cuales representaban avances significativos, ya sean pequeños o grandes, en el desarrollo del código.

Al finalizar el desarrollo de cada vista, se llevaron a cabo pull requests con el objetivo de incorporar todos los cambios realizados en la rama alterna a la rama principal. Este enfoque facilitó un manejo efectivo de las versiones y garantizó la integridad del código a lo largo del desarrollo.

5.8. Integración continua

Para las pruebas de integración continua, se llevó a cabo una investigación sobre las herramientas disponibles que podrían integrarse fácilmente en el proyecto. Tras evaluar las distintas opciones, se optó por utilizar "Husky", dada su capacidad para configurar hooks en Git. Esta herramienta se implementó con el propósito de automatizar diversas tareas dentro del flujo de trabajo del desarrollo.

Se configuraron dos hooks para asegurar la calidad del código y mejorar el proceso de desarrollo; estos fueron:

- **Pre-commit hook:** Este hook se ejecuta antes de realizar un commit y se utilizó para ejecutar automáticamente las pruebas de Eslint y asegurarse de que el código cumpliera con las reglas de estilo definidas.

Figura 28 .*Ejecución de ESLint durante commit*

```

Terminate batch job (Y/N)? y
PS C:\Users\Gaby\Documents\GitHub\Tesis2024> git add .
PS C:\Users\Gaby\Documents\GitHub\Tesis2024> git commit -m 'change based on feedback'

> project@0.1.0 lint:fix
> vue-cli-service lint --fix

Browserslist: caniuse-lite is outdated. Please run:
  npx update-browserslist-db@latest
  Why you should do it regularly: https://github.com/browserslist/update-db#readme
  DONE No lint errors found!

> project@0.1.0 lint
> vue-cli-service lint

Browserslist: caniuse-lite is outdated. Please run:
  npx update-browserslist-db@latest

```

Nota: Elaboración propia

- **Pre-push hook:** Este hook se ejecutó antes de enviar cambios a una branch. Se configuró para ejecutar pruebas unitarias, garantizando que el código pasara todas las pruebas antes de ser integrado.

Figura 29 .*Ejecución de pruebas unitarias durante push*

```

  PASS  tests/unit/BoxComplete.spec.ts

Test Suites: 16 passed, 16 total
Tests:      66 passed, 66 total
Snapshots:  0 total
Time:       7.443 s, estimated 15 s
Ran all test suites.
Enumerating objects: 19, done.
Counting objects: 100% (19/19), done.
Delta compression using up to 8 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (10/10), 2.92 KiB | 2.92 MiB/s, done.
Total 10 (delta 9), reused 2 (delta 1), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (9/9), completed with 8 local objects.
To https://github.com/Paola-Contreras/Tesis2024.git
 54094b7..eb8d722  main -> main

```

Nota: Elaboración propia

5.9. Pruebas a usuarios y retroalimentación

Para las pruebas, se seleccionó una muestra de 14 personas, compuesta por 7 doctores y 7 secretarías. Cabe destacar que los usuarios tenían edades variadas y, en el caso de las secretarías, contaban con distintos niveles de escolaridad.

Para la evaluación de la aplicación, se diseñaron dos encuestas específicas para cada grupo de usuarios, en las cuales se detallaban las actividades que debían realizar (Ver anexos 11.3 y 11.4). Estas encuestas se enfocaron en evaluar aspectos clave, como la facilidad de uso, la comprensión de las funcionalidades y la eficiencia en la realización de tareas dentro de la aplicación. Cada usuario completó las actividades asignadas, proporcionando retroalimentación sobre su experiencia y señalando posibles áreas de mejora. Esta información fue fundamental para ajustar la aplicación y adaptarla a las necesidades reales de cada perfil.

6.1. Aplicación web

Se desarrolló una aplicación web siguiendo las técnicas de interacción humano computador. Su uso permitió crear un sistema con lo necesario para poder ser utilizado dentro de una clínica pediátrica.

A continuación, se presentan imágenes que ilustran las principales funcionalidades del sistema clínico desarrollado. Estas capturas de pantalla reflejan la interfaz de usuario, los módulos implementados y el flujo de trabajo.

6.1.1. Inicio de sesión

Estas pantallas solicitan al usuario que inicie sesión. Como se muestra en la Figura 31, se incluyen mensajes informativos que le indican al usuario el estado de su ingreso, así como la existencia de errores en algún campo. De esta manera, se mejora la experiencia del usuario, propiciando un acceso más fluido y eficiente.

Figura 30 .

Pantalla que muestra el formulario de inicio de sesión sin datos ingresados



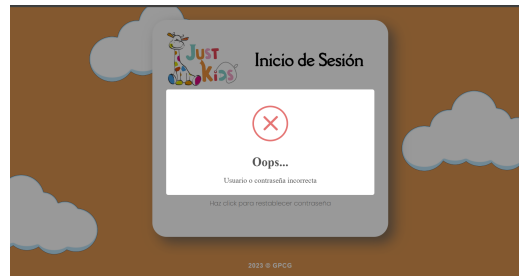
Figura 31 .

Pantalla que muestra el formulario de inicio de sesión con datos ingresados



Figura 32 .

Pantalla que indica un error en el inicio de sesión, y mensaje de advertencia



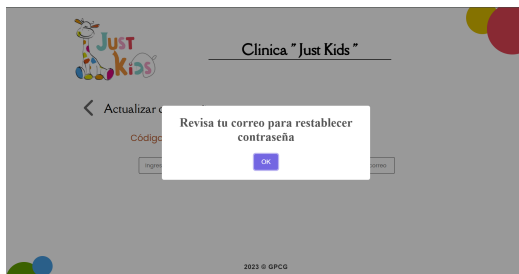
Nota: Elaboración propia

6.1.2. Restablecer contraseña

En las pantallas del proceso de actualización de contraseña, como se observa en las Figuras 32 y 33, se destaca el envío automático de un correo electrónico. Este mensaje incluye un cuerpo de texto genérico, claro y redactado en español, lo que garantiza que el usuario comprenda con sencillez la información proporcionada. Este diseño intuitivo busca facilitar la experiencia del usuario, asegurando que reciba la notificación de manera eficiente y pueda seguir los pasos necesarios para completar el proceso de actualización de su contraseña sin inconvenientes.

Figura 33 .

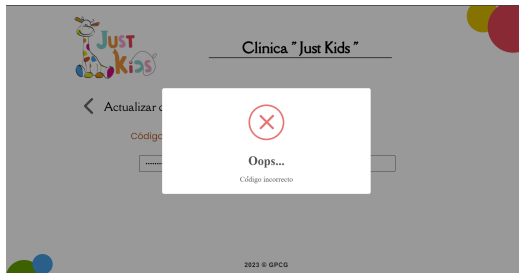
Notificación que informa sobre el proceso para solicitar el cambio de contraseña

**Figura 34 .**

Correo electrónico que contiene la información necesaria para restablecer la contraseña



Figura 35 .
Notificación que muestra un mensaje de error sobre códigos incorrectos y solicitud de una nueva entrada para continuar



Nota: Elaboración propia

Figura 36 .
Pantalla que permite al usuario ingresar una nueva contraseña



6.1.3. Agenda

Las pantallas de la agenda presentan de forma detallada las distintas acciones disponibles y los resultados de cada una, brindando al usuario una interfaz clara y funcional para gestionar sus tareas. Como se muestra en la Figura 37, la agenda ofrece varias opciones de visualización, lo cual agiliza el control y organización de las actividades programadas.

Además, el sistema integra notificaciones en tiempo real que confirman cada acción antes de su ejecución, asegurando que el usuario esté al tanto de los procesos y pueda evitar errores. Estas notificaciones también informan cuando una acción se ha completado con éxito, mejorando así la transparencia y el seguimiento de las actividades realizadas.

Figura 37 .
Pantalla que muestra la agenda organizada por semanas

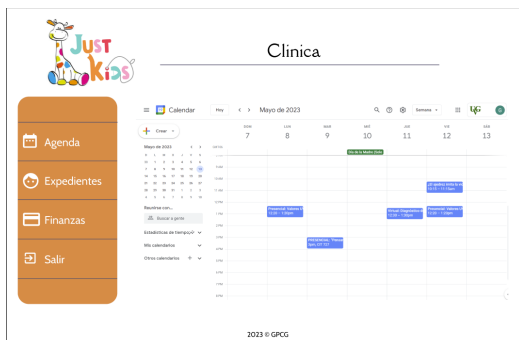


Figura 38 .
Pantalla que presenta la agenda organizada por meses

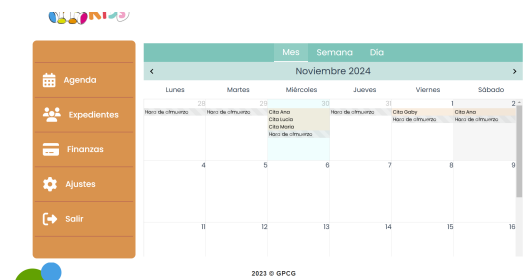


Figura 39 .
 Pantalla destinada a programar una nueva cita para el paciente

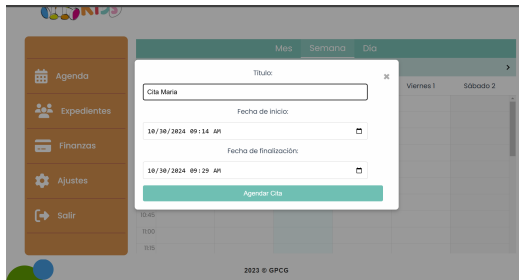


Figura 40 .
 Pantalla que muestra la cita creada en el calendario

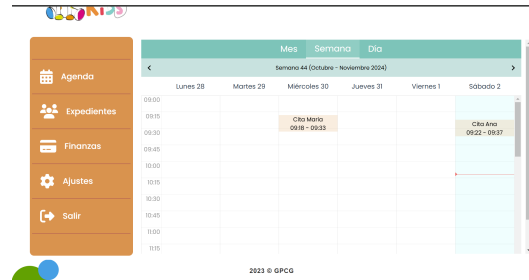


Figura 41 .
 Pantalla destinada para modificar los detalles de una cita existente

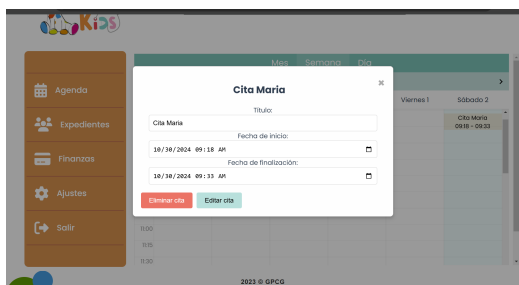


Figura 42 .
 Pantalla que muestra la cita actualizada en el calendario

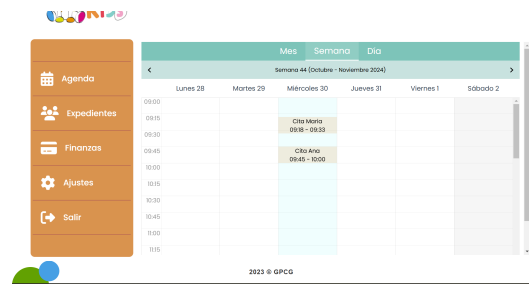
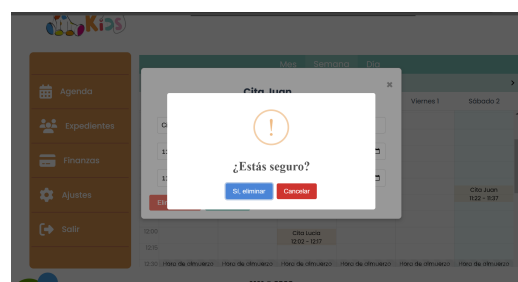


Figura 43 .
 Pantalla que refleja la cancelación de una cita en el calendario



Nota: Elaboración propia

6.1.4. Expedientes

En las pantallas, se solicita al usuario completar campos obligatorios en la ficha médica, como se muestra en la Figura 43. Si algún campo está vacío o contiene un error, la aplicación le notifica al usuario de manera inmediata, como se observa en las Figuras 45 y 46. Esto asegura que la información ingresada sea precisa y completa, permitiendo al usuario corregir cualquier dato antes de avanzar.

Figura 44 .
Sección superior de la ficha médica que muestra algunos datos esenciales a ingresar en el expediente clínico del paciente

Clínica "Just Kids"

< Nuevo Expediente

Datos personales

Ingresar los nombres del paciente * Ingresar los apellidos del paciente *

Ingresar sexo * Ingresar edad * Fecha de nacimiento *

Dirección de residencia *

Datos del encargado

Ingresar los nombres del encargado * Ingresar los apellidos del encargado *

Ingresar edad del encargado Ingresar parentesco del encargado

Figura 45 .
Sección inferior de la ficha médica que muestra los botones disponibles para completar el ingreso de datos y gestionar el expediente clínico del paciente

Antecedentes médicos Antecedentes quirúrgicos Antecedentes traumáticos

Otros antecedentes

Ingresar Hábitos Ingresar Alimentación Antecedentes familiares

Ingresar ficha

2023 © GPCG

Figura 46 .
Mensaje de advertencia que aparece cuando uno de los campos obligatorios se encuentra sin datos

< Nuevo Expediente

Datos personales

Regina Lucía Contreras Guerra

femenino 8 meses 07/16/2024

Ciudad

Datos del encargado

Lilán Guerra

Ingresar edad del encargado Ingresar parentesco del encargado

Ingresar número telefónico del encargado Ingresar DNI del encargado *

Antecedentes Prenatales

Uno o más campos obligatorios están vacíos.

Figura 47 .
Notificación que detalla las condiciones necesarias para la correcta validación del campo

Datos del encargado

Lilán Guerra

45 Madre

Ingresar número telefónico del encargado 12345

Error: el DNI debe tener 12 caracteres.

Figura 48 .
Sección superior de la ficha médica que muestra los campos con los datos registrados en el expediente clínico

Clínica "Just Kids"

< CONTRERAS GUERRA, REGINA LUCIA

Ficha Médica Historial Recetar

Datos Personales

Nombres: regina lucia Apellidos: contreras guerra

Sexo: femenino Edad: 8 meses Fecha de nacimiento: 2024-07-15

Figura 49 .
Sección inferior de la ficha médica que muestra los botones disponibles para navegar dentro de la aplicación

Otros antecedentes

Hábitos Alimentación Antecedentes familiares

NA NA NA

Editar ficha Cita Actual

2023 © GPCG

Nota: Elaboración propia

6.1.5. Cita actual

Estas pantallas permiten solicitar y visualizar la información de la evaluación en curso del paciente, proporcionando un acceso claro y detallado a los datos clínicos obtenidos durante la consulta. Esto facilita un seguimiento continuo y actualizado de la condición del paciente, permitiendo al médico tomar decisiones informadas en función de la evolución y los resultados observados en tiempo real.

Figura 50 .

Sección superior de cita actual que muestra algunos datos que se deben llenar para el historial del paciente

Figura 51 .

Sección inferior de cita actual que muestra los datos complementarios que se pueden detallar y los botones disponibles

Figura 52 .

Sección superior de cita actual que muestra los datos ingresados, relevantes para el historial del paciente

Figura 53 .

Sección inferior de cita actual que muestra algunos datos llenados y los botones disponibles para continuar navegando

Figura 54 .

Sección que presenta la cita actual con datos actualizados

Nota: Elaboración propia

6.1.6. Historial médico

Estas pantallas, muestra el registro detallado de citas que el paciente ha tenido durante su tiempo en la clínica (Figura 55). En esta misma sección, el doctor puede emitir recetas e imprimir la ficha médica, como se aprecia en la Figura 59. Es importante destacar que la receta médica está integrada con el historial, de modo que, al actualizarse, los medicamentos recetados queden reflejados en el historial y que corresponden a la cita actual del día.

Figura 55 .
Pantalla que permite buscar y acceder a los expedientes clínicos de los pacientes



Figura 56 .
Pantalla que muestra el historial médico completo del paciente

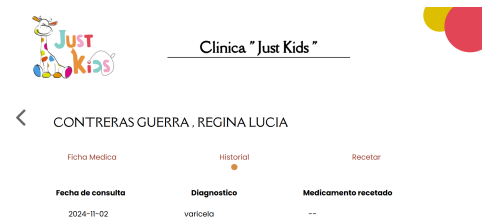


Figura 57 .
Pantalla diseñada para registrar los medicamentos que se prescribirán al paciente



Figura 58 .
Pantalla que permite eliminar un medicamento ingresado



Figura 59 .
 Sección superior de la receta médica que presenta información relevante de la clínica y el paciente



Figura 60 .
 Pantalla que permite visualizar y preparar la impresión de la receta médica

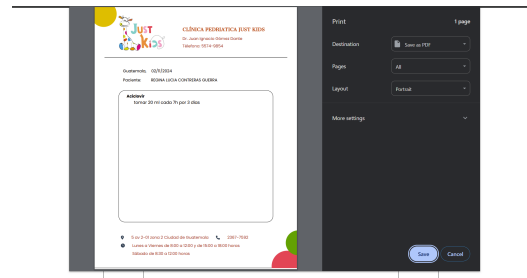


Figura 61 .
 Pantalla que muestra el historial actualizado en el campo de medicamentos



Nota: Elaboración propia

6.1.7. Finanzas

Estas pantallas muestran la gestión económica de la clínica, organizada en varias secciones clave para facilitar el control financiero. La opción de ingresar cobro permite registrar cada transacción de forma rápida y precisa. La sección caja del día ofrece un desglose detallado de todas las transacciones realizadas en la jornada. Finalmente, el historial de ganancias presenta un resumen financiero de los pagos recibidos por fecha, lo que simplifica el análisis de ingresos acumulados a lo largo del tiempo. Esta estructura integral permite mantener un control eficiente y detallado de las finanzas de la clínica.

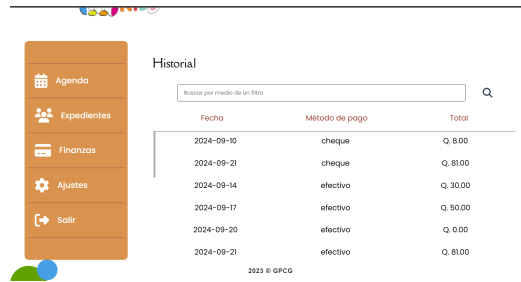
Figura 62 .
Pantalla diseñada para registrar un cobro en el sistema



Figura 63 .
Pantalla que muestra un resumen de todas las transacciones realizadas durante el día



Figura 64 .
Pantalla que presenta un resumen detallado de las ganancias



Nota: Elaboración propia

6.1.8. Ajustes

En estas pantallas se presentan todas las configuraciones desarrolladas, cada una de las cuales desempeña un papel esencial en la administración de la clínica. Estas configuraciones permiten al doctor personalizar y realizar cambios en la aplicación de manera autónoma, sin necesidad de recurrir a un tercero. Esta capacidad de personalización garantiza que el sistema se adapte a los procesos específicos de la clínica, optimizando así la atención al paciente y el flujo de trabajo.

Figura 65 .
Pantalla que muestra las diferentes opciones y funcionalidades disponibles en el sistema



Figura 66 .
Pantalla que permite observar y editar la información general de la clínica



Figura 67 .
 Pantalla destinada al registro y configuración de un nuevo usuario en la aplicación



Figura 68 .
 Pantalla que muestra los usuarios habilitados para utilizar la aplicación

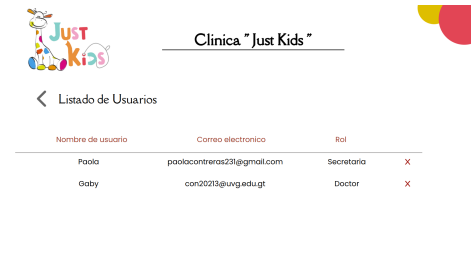


Figura 69 .
 Pantalla que muestra los seguros utilizados con opciones adicionales para editar o eliminar cada uno

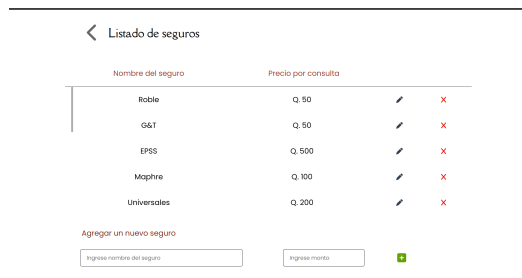


Figura 70 .
 Pantalla para modificar y actualizar los detalles de un seguro existente

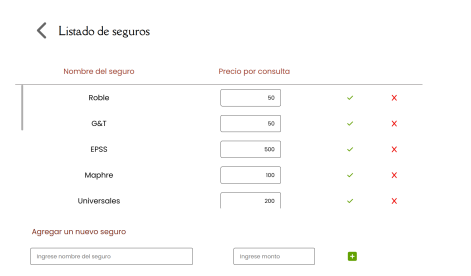


Figura 71 .
 Pantalla que detalla los campos accesibles para gestionar la información en los expedientes médicos



Figura 72 .
 Pantalla que muestra una advertencia al intentar eliminar un campo requerido según los lineamientos de la OMS



Figura 73 .

Pantalla que muestra una alerta al agregar un nuevo campo en una sección específica

The screenshot shows a web interface titled 'Manejo de Campos de Expedientes'. At the top, there is a green notification box with a checkmark and the text 'El campo ha sido añadido correctamente'. Below this, there are two sections: 'Datos Personales' and 'Datos del encargado'. The 'Datos Personales' section includes fields for 'Edad', 'FechaNacimiento', 'Residencia', and 'apodo'. Below these fields are two buttons: 'Agregar un nuevo campo' and 'Eliminar un campo'. Under 'Agregar un nuevo campo', there is a text input field with a green plus icon and a green checkmark. Under 'Eliminar un campo', there is a text input field with a red minus icon and a red X. The 'Datos del encargado' section includes fields for 'Nombre', 'Edad', 'Parentesco', 'DPI', and 'Telefono'.

Figura 74 .

Pantalla que confirma la correcta integración del campo en el expediente médico

The screenshot shows a web interface titled 'Datos Personales'. It displays the following information: 'Nombres: regina lucia', 'Apellidos: contreras guerra', 'Sexo: femenino', 'Edad: 8 meses', 'Fecha de nacimiento: 2024-07-15', 'Dirección: Ciudad', and 'apodo:'. The fields are presented as a form with horizontal lines for input.

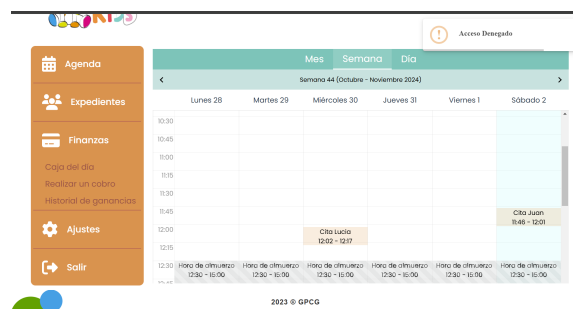
Nota: Elaboración propia

6.1.9. Gestión de roles

Como se puede apreciar en esta pantalla, se muestra una notificación que dice *acceso denegado*. Esta alerta informa a las secretarías que existen secciones del sistema a las cuales no tienen permitido acceder. Esta medida de seguridad es crucial para proteger la información sensible y garantizar que solo el personal autorizado pueda visualizar y gestionar ciertos datos.

Figura 75 .

Notificación de acceso denegado al ingresar a una vista sin autorización



Nota: Elaboración propia

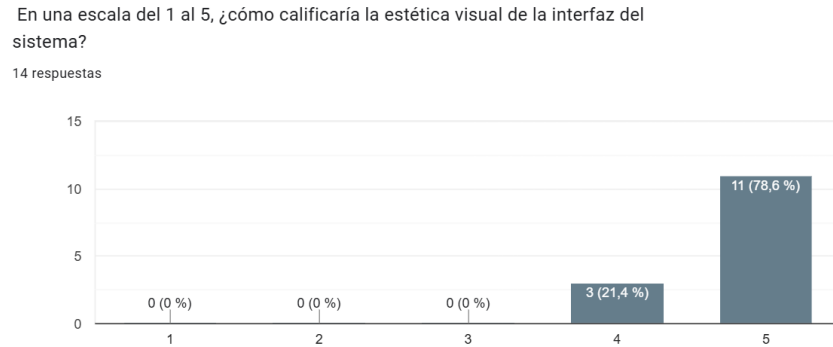
6.2. Evaluación de la aplicación web

Para evaluar el diseño y la funcionalidad de la aplicación, se realizaron encuestas dirigidas al grupo objetivo. Los resultados proporcionan una visión detallada sobre la satisfacción general, la facilidad de uso y la efectividad de las diferentes características implementadas. Partiendo de una escala de 1 a 5, donde 1 revela una opinión negativa y 5 una positiva, se presentan los estadísticos obtenidos:

La Figura 75 muestra los resultados de la evaluación estética de la interfaz. De las 14 respuestas, el 78.6% calificó con 5 y el 21.4% con 4, lo que indica que la mayoría de los usuarios encontró atractiva la estética visual. La ausencia de calificaciones por debajo de 4, sugiere una valoración positiva de la apariencia de la aplicación.

Figura 76 .

Respuestas de los usuarios sobre la estética visual de la interfaz.

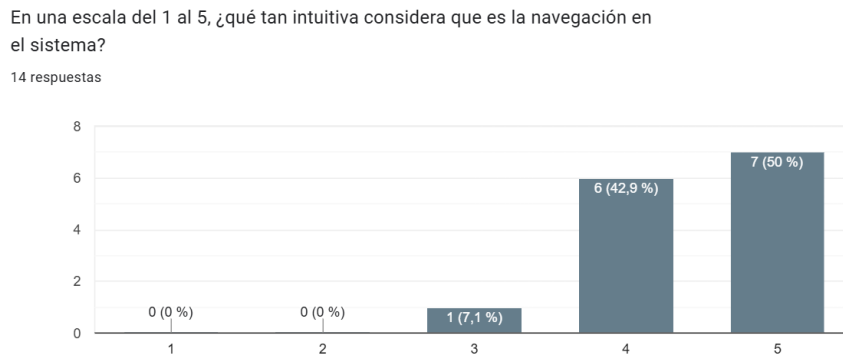


Nota: Elaboración propia

La Figura 76 muestra que la mayoría de los encuestados consideró la navegación de la aplicación como intuitiva: un 50 % otorgó la máxima calificación de 5, mientras que el 42.9 % calificó con 4, reflejando una experiencia de usuario positiva en términos de accesibilidad y facilidad de uso. Solo un 7.1 % dio 3 puntos, lo que sugiere que algunos usuarios percibieron aspectos menores que podrían mejorarse, sin considerarlos insatisfactorios. La ausencia de calificaciones de 1 y 2 indica que ninguno encontró la navegación confusa o difícil de entender.

Figura 77 .

Calificaciones de los usuarios sobre la intuición de navegación en la aplicación.

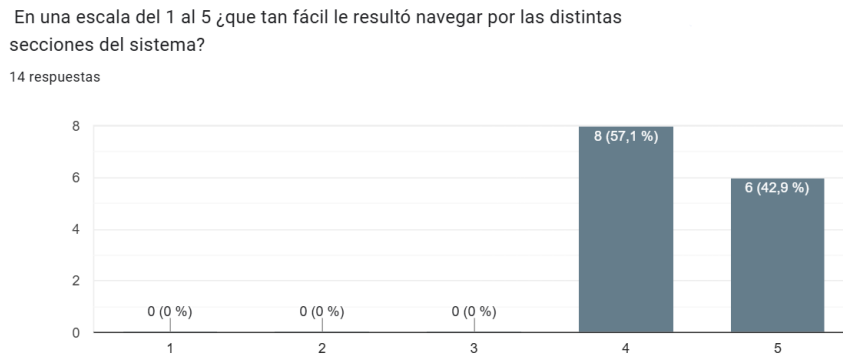


Nota: Elaboración propia

En la Figura 77, se observa que el 57.1 % de los usuarios calificó la navegación con 4, y el 42.9 % con 5, reflejando una experiencia de uso positiva. Esto sugiere que la navegación fue fácil y no causó problemas, aunque se identifican posibles áreas de mejora.

Figura 78 .

Calificaciones de los usuarios sobre la facilidad de navegación a través de las diferentes secciones del sistema.

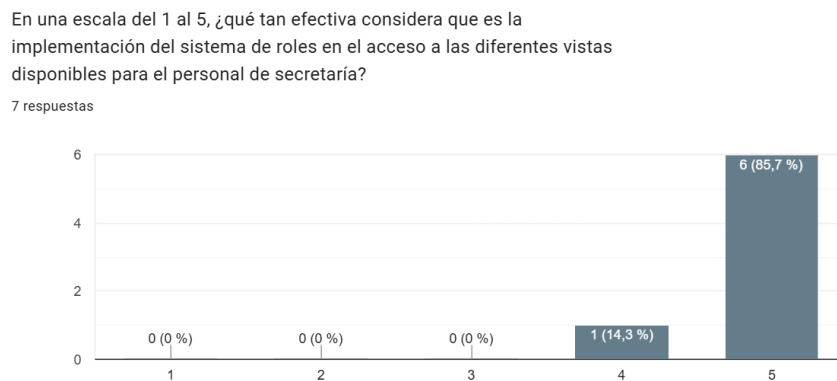


Nota: Elaboración propia

La Figura 78 muestra la percepción sobre la efectividad del sistema de roles en el acceso a las distintas vistas para el personal de secretaría. De las 7 respuestas obtenidas, el 85.7% calificó la implementación con un 5, lo que indica un alto nivel de satisfacción con la organización y acceso de roles. El 14.3% restante otorgó una calificación de 4, reflejando una valoración positiva y sugerencias menores de mejora.

Figura 79 .

Calificaciones de los usuarios sobre la efectividad de la aplicación de roles en las diferentes vistas para el personal.

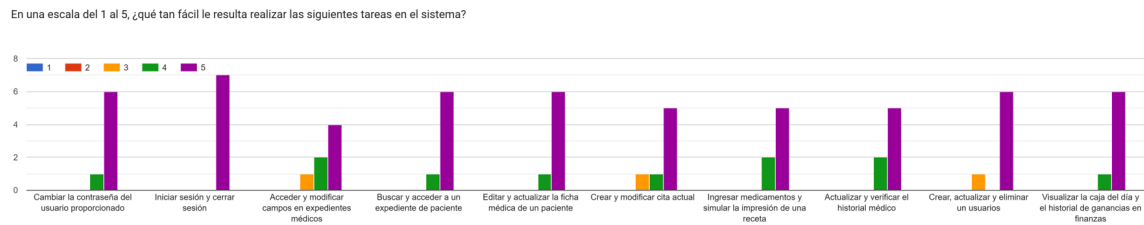


Nota: Elaboración propia

La evaluación de usabilidad realizada por los doctores (Figura 79) muestra puntajes altos, al recibir una valoración general de 4 y 5 puntos, indicando una experiencia satisfactoria en la mayoría de las áreas. Sin embargo, tres actividades específicas recibieron una calificación de 3, lo que sugiere un nivel de complejidad mayor en estas tareas en comparación con las demás. Los resultados destacan una usabilidad general positiva, aunque estas áreas específicas podrían beneficiarse de algunos cambios para optimizar la experiencia.

Figura 80 .

Calificaciones de los usuarios sobre la facilidad de realizar las distintas funciones asignadas al doctor en el sistema.

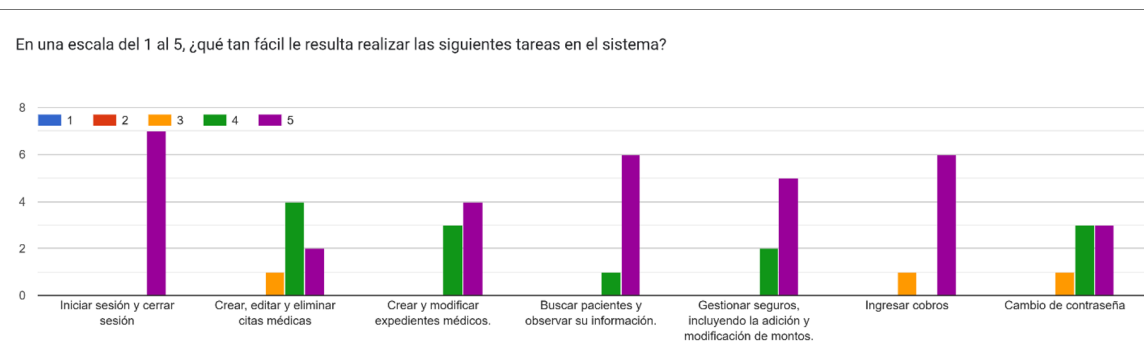


Nota: Elaboración propia

La evaluación de usabilidad realizada por las secretarias (Figura 80) muestra que la mayoría de las actividades fueron calificadas con un 5, indicando que las usuarias completaron las tareas sin dificultad y tuvieron una experiencia de uso positiva y autónoma. Algunas actividades obtuvieron una calificación de 4, reflejando un manejo satisfactorio del sistema. Sin embargo, la tarea de cambio de contraseña presentó una dificultad mayor en comparación con las demás, probablemente debido al proceso de doble autenticación. En general, los resultados sugieren que el sistema fue accesible y funcional.

Figura 81 .

Calificaciones de los usuarios sobre la facilidad de realizar las distintas funciones asignadas a la secretaria en el sistema.



Nota: Elaboración propia

6.3. Presupuesto

El presupuesto contempla los recursos esenciales para el correcto funcionamiento del sistema, incluyendo los servicios de Firestore Database y Hosting de Firebase, con una estimación de costos mensuales basada en el uso proyectado.

El servicio de Firestore Database incluye operaciones de lectura, escritura y eliminación de datos, así como almacenamiento y salida de red. Firebase ofrece un límite gratuito diario para las operaciones de lectura (50,000), escritura (20,000) y eliminación (20,000). No obstante, debido a la demanda del sistema, estos límites pueden llegar a ser superados, generando costos adicionales. Se ha estimado un consumo de 350,000 lecturas, 150,000 escrituras y 70,000 eliminaciones, lo que

representa un costo mensual de \$15.31. Adicionalmente, el almacenamiento de datos y la salida de red tienen un consumo proyectado de 120 GB y 70 GB, respectivamente, lo que genera un costo de \$28.62.

Por otro lado, el servicio de Hosting de Firebase, utilizado para la transferencia y almacenamiento de datos, también cuenta con una versión gratuita. Sin embargo, al superar los límites establecidos, se generan costos adicionales. Se ha estimado un consumo de 60 GB en transferencia de datos y 70 GB en almacenamiento, lo que implica un costo mensual de \$55.86.

En total, la suma de los costos de Firestore Database y Hosting asciende aproximadamente \$100 mensuales. Este monto es una estimación y puede variar en función del uso real del sistema, pudiendo aumentar o disminuir dependiendo de la cantidad de datos transferidos, almacenados y procesados.

Cabe destacar que este presupuesto se ha calculado considerando que la clínica médica opera con un doctor y una secretaria, y que el doctor atiende en promedio 30 pacientes por día. Estos parámetros han sido tomados en cuenta para realizar una proyección de consumo acorde a la actividad diaria de la clínica.

Figura 82 .

Estimación de costos mensuales para el mantenimiento de la aplicación web

<i>Firestore Database</i>				
Lectura y escritura				
Métrica	Limite Gratuito	Uso Estimado	Costo por cada 100,000	Subtotal Mensual (USD)
Lecturas	50,000	350,000	\$ 0.06	\$ 6.51
Escrituras	20,000	150,000	\$ 0.18	\$ 8.37
Eliminaciones	20,000	70,000	\$ 0.02	\$ 0.43
Subtotal				\$ 15.31
Almacenamiento y redes				
Salida de red	10 GB	70 GB	\$ 0.12	\$ 7.20
Almacenamiento de datos	1 GB	120 GB	\$ 0.18	\$ 21.42
Subtotal				\$ 28.62
<i>Hosting</i>				
Transferencia de datos	10 GB	60 GB	\$ 0.15	\$ 7.50
Almacenamiento	10 GB	70 GB	\$ 0.03	\$ 48.36
Subtotal				\$ 55.86
Total				\$ 99.79

Nota: Elaboración propia

Discusión de resultados

La aplicación desarrollada para clínicas pediátricas tuvo como objetivo principal la sistematización de procesos médicos y administrativos. Mediante esta herramienta, se buscó reducir el tiempo empleado en buscar, archivar y transferir expedientes, simplificando así el trabajo del personal y mejorando la eficiencia de la clínica. Además, la aplicación brindó al doctor la posibilidad de acceder de forma remota a la información de sus pacientes, sin necesidad de estar físicamente en la clínica.

Para evaluar el primer objetivo específico, enfocado en la intuitividad de la aplicación, se realizaron entrevistas con un grupo de usuarios conformado por siete doctores pediátricos y siete secretarías. Los resultados indicaron que, en general, la aplicación es fácil de usar y no requiere conocimientos previos para su manejo. Como se observa en la Figura 76, solo uno de los usuarios presentó algunas dificultades, probablemente debido a la necesidad de familiarizarse con la nueva herramienta. Las secciones de la aplicación están organizadas de manera lógica, con títulos claros y una distribución accesible de botones y menús, lo cual facilita la navegación y el uso de las distintas funcionalidades, como se evidencia en las Figuras 77, 79 y 80, desglosado por rol.

El segundo objetivo específico abordó la gestión de roles dentro de la aplicación, con el fin de asegurar la privacidad y seguridad de los datos. En este sentido, se evaluaron las funciones disponibles para el rol de secretaria, limitando su acceso a información confidencial de los pacientes y de la clínica. En la Figura 74 se muestra cómo el sistema emite una alerta a los usuarios con este rol cuando intentan acceder a vistas restringidas, reforzando así las limitaciones de acceso. Además, en la Figura 78 se observa que la mayoría de los médicos entrevistados consideró adecuada esta restricción. No obstante, uno de los entrevistados sugirió mejorar la interfaz ocultando por completo las vistas no accesibles, para evitar que el personal conozca la existencia estas.

El tercer objetivo específico se enfocó en la centralización de la información médica de los pacientes, desarrollando un módulo que permitiera almacenar y acceder de manera sencilla y rápida a todos los datos relevantes en un único lugar. Durante el desarrollo de este módulo, se priorizó la organización y accesibilidad de la información, buscando reducir la dispersión de datos y facilitar el acceso a los expedientes médicos. Como se observa en la Figura 79, la centralización de la información en un solo módulo ha facilitado significativamente tanto el acceso como la actualización de los datos de los pacientes.

- La herramienta desarrollada para clínicas pediátricas cumplió con el objetivo de sistematizar la gestión de procesos médicos y administrativos, mejorando la eficiencia en la organización y el acceso a la información.
- La aplicación resultó intuitiva y fácil de usar para el personal médico y administrativo, facilitando su adopción sin necesidad de conocimientos técnicos avanzados.
- La implementación de un sistema de roles garantizó la confidencialidad y seguridad de la información de los pacientes, al restringir adecuadamente el acceso y la modificación de los datos sensibles.
- En general, la herramienta representa un avance significativo en la administración clínica, aportando una gestión más organizada y segura, lo cual establece las bases para futuras mejoras según las necesidades del entorno médico.

Recomendaciones

- Con el fin de mejorar la validez de los resultados, se recomienda ampliar el tamaño de la muestra. Contar con un número de usuarios superior a 14 ofrecería evidencia más sólida sobre la eficacia del sistema.
- Se sugiere añadir un módulo de gráficos que permita a los pediatras visualizar las curvas de crecimiento de cada paciente, facilitando el seguimiento y análisis del desarrollo infantil para un monitoreo clínico más completo.
- Se recomienda desarrollar una versión móvil de la aplicación, permitiendo a los usuarios acceder de forma práctica y portátil desde cualquier lugar. Esto mejorará la disponibilidad de la herramienta y facilitará el acceso a la información de manera inmediata.
- Se recomienda ampliar la personalización para que cada pediatra adapte la aplicación a su estilo de trabajo, además de evaluar la integración de servicios externos para un mayor alcance.

-
- [1] Amazon Web Services: *What is a Database? - Databases Explained*. Disponible en: <https://aws.amazon.com/what-is/database/>.
 - [2] Amazon Web Services: *What is a Framework? - Framework in Programming and Engineering Explained*. Disponible en: <https://aws.amazon.com/what-is/framework/>.
 - [3] Amazon Web Services: *¿Qué son las pruebas unitarias?: explicación de las pruebas unitarias en AWS*. Disponible en: <https://aws.amazon.com/es/what-is/unit-testing/>.
 - [4] Asociación Pediátrica de Guatemala: *Historia - ASOPEDIA*, 2024. Disponible en: <https://asopedia.org/historia/>.
 - [5] Atlassian: *Qué es el control de versiones | Atlassian Git Tutorial*. Disponible en: <https://www.atlassian.com/es/git/tutorials/what-is-version-control>.
 - [6] Atlassian: *¿En qué consiste la integración continua?* Disponible en: <https://www.atlassian.com/es/continuous-delivery/continuous-integration>.
 - [7] AWS: *Integración continua del software | Pruebas automatizadas | AWS*. Disponible en: <https://aws.amazon.com/es/devops/continuous-integration/#:~:text=La%20integraci%C3%B3n%20continua%20se%20refiere%20a%20la%20fase%20de%20creaci%C3%B3n,la%20creaci%C3%B3n%20y%20las%20pruebas>.
 - [8] Bakharev, N.: *Unit Testing: Definition, Examples, and Critical Best Practices*, septiembre 2024. Disponible en: <https://brightsec.com/blog/unit-testing/>.
 - [9] British Academy Of Graphic Design: *British Academy Of Graphic Design: Why is typography important in graphic design?* Disponible en: <https://www.graphicdesignacademy.uk/blog/why-is-typography-important-in-graphic-design->.
 - [10] Brittany Anas: *InVision: Understanding color theory: the color wheel and finding complementary colors*. Disponible en: <https://www.invisionapp.com/inside-design/understanding-color-theory-the-color-wheel-and-finding-complementary-colors/>.
 - [11] Bruke, H.: *Cómo eliminar el uso de papel en un hospital*, 2020. Disponible en: <https://www.linkedin.com/pulse/c3%B3mo-eliminar-el-uso-del-papel-en-un-hospital-hern%C3%A1n-bruke/>.

- [12] buzonuv: *Medicina – Centro de Estudios y Servicios en Salud*. Disponible en: <https://www.uv.mx/veracruz/cess/vinculacion-y-extension/medicina/>.
- [13] Canva: *Canva: Color*. Disponible en: <https://www.canva.com/colors/color-wheel/>.
- [14] Canva: *Canva: La psicología del color: el significado de los colores y cómo aplicarlos a tu marca*. Disponible en: https://www.canva.com/es_mx/abrilende/psicologia-del-color/.
- [15] Carecloud: *Chronic Care Management (CCM) service - CareCloud Wellness*, 2023. Disponible en: <https://www.carecloud.com/chronic-care-management/>.
- [16] Chawla, B.: *Vitest vs Jest: A Personal Take on Testing Frameworks*, mayo 2024. Disponible en: <https://www.linkedin.com/pulse/vitest-vs-jest-personal-take-testing-frameworks-bhuvan-chawla-uvjfc>.
- [17] De Imagina, E.: *¿Qué es Angular y cuáles son sus ventajas?*, noviembre 2024. Disponible en: <https://imaginaformacion.com/tutoriales/conoce-las-ventajas-de-utilizar-angular>.
- [18] De Impresión, S. A.: *Automatización de procesos en el sector salud*, 2023. Disponible en: <https://www.linkedin.com/pulse/automatizaci%C3%B3n-de-procesos-en-el-sector/>.
- [19] DocuWare Europe GmbH: *Transformación digital para una oficina sin papeles | DocuWare*. Disponible en: <https://start.docuware.com/es/una-oficina-sin-papeles>.
- [20] easiio: *Jazmín Javascript*. Disponible en: <https://www.easiio.com/es/easiio-jasmine-javascript/>.
- [21] EsmeeWevers: *Various types of prototypes, 3 categories - Beeliners English*, marzo 2023. Disponible en: <https://beeliners.com/blog/prototype-types/#:~:text=Prototypes%20come%20in%20many%20different,medium%20fidelity%20and%20high%20fidelity>.
- [22] Figma: *Figma: Círculo Cromático*. Disponible en: <https://www.figma.com/es-es/circulo-cromatico/>.
- [23] GeeksforGeeks: *React Introduction*, oct 2024. Disponible en: <https://www.geeksforgeeks.org/reactjs-introduction/>.
- [24] GeeksforGeeks: *What is Angular?*, abril 2024. Disponible en: <https://www.geeksforgeeks.org/what-is-angular/>.
- [25] GitHub: *About branches - GitHub Docs*. Disponible en: <https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/about-branches>.
- [26] GitHub: *About commits - GitHub Docs*. Disponible en: <https://docs.github.com/en/pull-requests/committing-changes-to-your-project/creating-and-editing-commits/about-commits>.
- [27] GitHub: *About GitHub and Git - GitHub Docs*. Disponible en: <https://docs.github.com/en/get-started/start-your-journey/about-github-and-git>.
- [28] GitHub: *Git - Acerca del control de versiones*. Disponible en: <https://git-scm.com/book/es/v2/Inicio---Sobre-el-Control-de-Versiones-Acerca-del-Control-de-Versiones>.
- [29] Google Cloud: *¿Qué es una base de datos relacional (RDBMS)?* Disponible en: <https://cloud.google.com/learn/what-is-a-relational-database?hl=es#what-is-a-relational-database>.

- [30] Gragnolati, M. y A. Marini: *Health and Poverty in Guatemala*, 2003. Disponible en: https://documents1.worldbank.org/curated/en/854931468752070020/114514322_20041117164032/additional/multi0page.pdf.
- [31] Henry Vasconcellos: *Softplan: HCI: Descubra qué es la Interacción Humano-Computadora y en qué se centra el estudio*. Disponible en: <https://www.softplan.com.br/es/tech-writers/ihc/>.
- [32] How Pediatric Medicine differs from Adult Medicine. Disponible en: <https://www.grandlakehealth.org/patients-and-visitors/information-center/resources/health-library/pediatric-health.html>.
- [33] Ilyusenko, P.: *MVP Development*. Disponible en: <https://www.scnsoft.com/software-development/mvp>.
- [34] Interaction Design Foundation: *What is User Experience (UX) Design?*, septiembre 2024. Disponible en: [https://www.interaction-design.org/literature/topics/ux-design#what_is_user_experience_\(ux\)_design?-0](https://www.interaction-design.org/literature/topics/ux-design#what_is_user_experience_(ux)_design?-0).
- [35] Jasmine: *Jasmine Documentation*. Disponible en: <https://jasmine.github.io/>.
- [36] Javatpoint: *What is a Database? Definition, Types, Examples*. Disponible en: <https://www.javatpoint.com/what-is-database>.
- [37] Jest: *Jest*. Disponible en: <https://jestjs.io/>.
- [38] Jesús: *Ventajas y desventajas de MongoDB*, septiembre 2024. Disponible en: <https://www.dongee.com/tutoriales/ventajas-y-desventajas-de-mongodb/>.
- [39] Key Benefits of Private Healthcare | Vista Health. Disponible en: <https://www.vista-health.co.uk/blog/key-benefits-private-healthcare>.
- [40] La automatización de procesos en el sector salud (De AdeA), 2022. Disponible en: <https://www.adea.es/blog/automatizacion-salud/>.
- [41] M, E. A. G.: *Almacenamiento en la Nube VS almacenamiento tradicional. ¿Cuál elegir?*, 2023. Disponible en: <https://www.linkedin.com/pulse/almacenamiento-en-la-nube-vs-tradicional-cu%C3%A1l-elegir-garc%C3%ADa-m-/>.
- [42] Masters, F.: *What Does a Front-End Developer Do? Complete Guide to the Front-End Developer Profession*, marzo 2023. Disponible en: <https://frontendmasters.com/guides/front-end-handbook/2018/what-is-a-FD.html>.
- [43] Medicaladmin: *Beneficios de usar un software médico en tu consultorio o clínica*, 2023. Disponible en: <https://inmedical.net/beneficios-software-medico/>.
- [44] Mijacobs: *¿Qué es el control de versiones? - Azure DevOps*, oct 2023. Disponible en: <https://learn.microsoft.com/es-es/devops/develop/git/what-is-version-control>.
- [45] MongoDB: *What is a Non-Relational Database?* Disponible en: <https://www.mongodb.com/resources/basics/databases/non-relational>.
- [46] Mozilla Corporation's : *CSS basics - Learn web development | MDN*, oct 2024. Disponible en: https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/CSS_basics.
- [47] Mozilla Corporation's : *HTML: HyperText Markup Language | MDN*, septiembre 2024. Disponible en: <https://developer.mozilla.org/en-US/docs/Web/HTML>.
- [48] Mozilla Corporation's: *JavaScript | MDN*, marzo 2024. Disponible en: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>.

- [49] News Atlantis: *Cinco beneficios de sistematizar la información de los pacientes – Landing / Atlantis University*, 2020. Disponible en: <https://landing.atlantisuniversity.edu/cinco-beneficios-de-sistematizar-la-informacion-de-los-pacientes/>.
- [50] OVHcloud: *OVHcloud: ¿Qué es una aplicación web?* Disponible en: <https://www.ovhcloud.com/es/learn/what-is-web-application/>.
- [51] Pediatrics | Careers in Medicine. Disponible en: <https://careersinmedicine.aamc.org/explore-options/specialty-profiles/pediatrics#:~:text=Pediatricians%20practice%20the%20specialty%20of,of%20acute%20and%20chronic%20diseases.>
- [52] Powercode - Ultimate Software Development: *What is MVP in software development?*, abril 2023. Disponible en: <https://www.linkedin.com/pulse/what-mvp-software-development-powercodiciembreuk>.
- [53] Private healthcare - Freedom Health insurance. Disponible en: <https://www.freedomhealthinsurance.co.uk/about-private-healthcare#:~:text=Private%20healthcare%20treatment%20centres%20can,choose%20your%20consultant%20or%20specialist>.
- [54] Profesión médica: MedlinePlus enciclopedia médica, 2023. Disponible en: <https://medlineplus.gov/spanish/ency/article/001936.htm>.
- [55] Rangnekar, P.: *The Benefits and Downsides of Angular Development!*, diciembre 2023. Disponible en: <https://www.linkedin.com/pulse/benefits-downsides-angular-development-parija-rangnekar-ska5f>.
- [56] Rock Content: *RockContent: ¿Qué es el círculo cromático y cuál es su importancia en el diseño?* Disponible en: <https://rockcontent.com/es/blog/circulo-cromatico/>.
- [57] Sandra Coffey: *Indeed: What Is Typography? (Plus Its Purpose, Rules and Elements)*. Disponible en: <https://www.indeed.com/career-advice/career-development/typography>.
- [58] Sivan, V.: *Introducing Vitest: the super fast testing framework*, oct 2023. Disponible en: <https://codemaker2016.medium.com/introducing-vitest-the-super-fast-testing-framework-c4a86b431f8d#d36d>.
- [59] Stouse, P.: *What is ‘pediatric’?*, 2022. Disponible en: <https://pubmed.ncbi.nlm.nih.gov/36707196/>.
- [60] Strengthening Health Outcomes through the Private Sector (SHOPS): *Guatemala Diagnóstico del sector privado de salud*, 2016. Disponible en: <https://2012-2017.usaid.gov/sites/default/files/documents/1862/GUATEMALA-Analysis-Sector-Privado-INFORME-EJECUTIVO-Final.pdf>.
- [61] Systemize - Definition & Benefits. Disponible en: <https://www.activecampaign.com/glossary/systemize>.
- [62] TAG MedStaffing: *What is a Medical Clinic? Definition, Types, When to Visit & More*, 2024. Disponible en: <https://www.tagmedstaffing.com/what-is-a-clinic/>.
- [63] TechTarget: *TechTarget: What is the difference between website and web application.* Disponible en: <https://www.techtarget.com/searchsoftwarequality/definition/Web-application-Web-app>.
- [64] Teresa, M. S. M.: *Médicos y antropólogos que descifran y tratan males: El desarrollo de la antropología de la medicina en Guatemala*. Disponible en: https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1607-050X2007000100011.

- [65] Thinksys: *Thinksys: Web Application Architecture: Working, Components, Types, Trends.* Disponible en: <https://thinksys.com/development/web-application-architecture-complete-guide/>.
- [66] Villalta Ortiz, D: *Desventajas de JEST.* Disponible en: https://issuu.com/denisvillalta/docs/presentaci_n_jest_grupo_az_1.pptx/s/29246698.
- [67] Villalta Ortiz, D: *Ventajas de JEST.* Disponible en: https://issuu.com/denisvillalta/docs/presentaci_n_jest_grupo_az_1.pptx/s/29246697.
- [68] Vue: *Vue.js.* Disponible en: <https://vuejs.org/guide/introduction.html#what-is-vue>.
- [69] White Jacome, W.: *Informe final de tesis Opinión de los egresados de la Facultad de Ciencias Medicas de la Universidad de San Carlos de Guatemala De la promoción del año de 1979, en relación a su formación y desempeño profesional.* Disponible en: http://biblioteca.usac.edu.gt/tesis/05/05_0206.pdf.
- [70] ¿Qué es el SaaS? - Explicación del software como servicio - AWS. Disponible en: [https://aws.amazon.com/es/what-is/saas/#:~:text=El%20software%20como%20servicio%20\(SaaS\)%20es%20un%20modelo%20de%20software,acceder%20a%20ellos%20bajo%20demanda.](https://aws.amazon.com/es/what-is/saas/#:~:text=El%20software%20como%20servicio%20(SaaS)%20es%20un%20modelo%20de%20software,acceder%20a%20ellos%20bajo%20demanda.)

11.1. Enlace al prototipo funcional

www.figma.com/proto/o7vpQdvMaj68181NEBu00B/Tesis?node-id=57-640&t=qAj3ENxtndhcbQej-1

11.2. Enlace al repositorio de GitHub

<https://github.com/Paola-Contreras/Tesis2024>

11.3. Tareas asignadas al doctor

Figura 83 .

Listado de instrucciones realizadas por el doctor

Acciones a realizar

Para proceder con las pruebas, es necesario que utilices las siguientes credenciales de acceso:

- **Usuario:**gaby
- **Código personal:**1234567890ga

A continuación, se presenta una lista de actividades que deberá llevar a cabo. Una vez que haya completado todas las tareas, le agradeceríamos que respondiera a una breve encuesta que nos ayudará a evaluar la calidad del sistema desarrollado.

- Cambiar la contraseña del usuario proporcionado.
- Iniciar sesión utilizando el usuario proporcionado y la contraseña establecida anteriormente.
- Diríjase a la sección de configuraciones y seleccione "campos de expedientes".
- Verifique los campos existentes en cada apartado de la pantalla y añada un campo en alguna de las áreas disponibles.
- Busque un expediente.
- Acceda al expediente de un paciente y observe los datos registrados.
- Edite la ficha médica, añadiendo un valor al campo creado previamente.
- Confirme que el campo modificado haya sido actualizado correctamente.
- Cree una cita actual para un paciente con la fecha de hoy.
- Acceda a la cita creada y revise la información registrada.
- Actualice un campo de la cita actual para verificar su correcta modificación.
- Diríjase al apartado de recetar en la parte superior del expediente.
- Ingrese medicamentos y dosis en el sistema para realizar una receta.
- Presione el cheque verde para ir creando el listado de medicamentos.
- Simule la impresión de la receta para confirmar su funcionalidad.
- Diríjase al apartado de historial en la parte superior del expediente.
- Actualice el historial médico del paciente.
- Verifique que los cambios en los campos del historial se hayan realizado correctamente.
- Acceda a una visita que se encuentre previamente registrada en el historial.
- Regrese a la página principal y busque la sección de configuraciones.
- Cree un nuevo usuario con el rol de secretaria.
- Revise la lista de usuarios disponibles y confirme que el nuevo usuario esté registrado.
- Edite un campo del usuario creado previamente.
- Elimine el usuario creado en el paso anterior.
- Regrese a la página principal y busque la sección de finanzas
- Diríjase a la sección de realizar un cobro.
- Ingrese un cobro nuevo, colocando que el mismo se realiza SIN seguro.
- Diríjase a la sección de finanzas y visualice la caja del día.
- Diríjase a la sección de finanzas y visualice el historial de ganancias.
- Cerrar sesión una vez finalizadas las pruebas detalladas anteriormente.

Nota: Elaboración propia

11.4. Tareas asignadas a la secretaria

Figura 84 .

Listado de instrucciones realizadas por la secretaria

Acciones a realizar:

Para proceder con las pruebas, es necesario que utilices las siguientes credenciales de acceso:

- **Usuario:** paola
- **Código personal:**1234567890pa

A continuación, se presenta una lista de actividades que deberá llevar a cabo. Una vez que haya completado todas las tareas, le agradeceríamos que respondiera a una breve encuesta que nos ayudará a evaluar la calidad del sistema desarrollado.

- Cambiar la contraseña del usuario proporcionado.
- Iniciar sesión utilizando el usuario proporcionado y la contraseña ingresada previamente.
- Crear una cita médica para un paciente existente o nuevo.
- Editar una cita médica existente.
- Eliminar la cita creada previamente y verificar la correcta actualización en el sistema.
- Observar el calendario en una presentación diferente a la vista semanal.
- Crear un expediente para un nuevo paciente con los datos requeridos.
- Buscar al paciente ingresado previamente.
- Observar el expediente del paciente ingresado previamente para verificar los datos.
- Modificar un campo específico del expediente médico del paciente y confirme que el cambio se refleja adecuadamente.
- Regresar a la vista de búsqueda de pacientes y eliminar el paciente registrado como "prueba".
- Dirigirse a la vista de administración de clínica.
- Modificar algún dato de la clínica.
- Dirigirse a la vista de gestión de aseguradoras.
- Agregar un nuevo seguro.
- Modificar el monto de un seguro existente y confirmar que el cambio se refleja adecuadamente.
- Eliminar el seguro registrado como "prueba".
- Dirigirse a la vista de ingresar cobro.
- Ingresar un cobro utilizando el seguro previamente añadido.
- Acceda a una sección que no se haya mencionado anteriormente.
- Cerrar sesión una vez finalizadas las pruebas detalladas anteriormente.

Nota: Elaboración propia