**PROYECTO AURORA**

**ARANGO DÍAZ SAMUEL**

**FLOREZ JULIO JEISON STEVEN**

**UNIVERSIDAD PONTIFICIA BOLIVARIANA**

**ESCUELA DE INGENIERÍAS**

**FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**JUAN CARLOS MORANTES (Docente Universitario)**

**BUCARAMANGA, COLOMBIA**

**2024**

**ÍNDICE**

1. **Introducción**
2. **Situación Problema**
   1. Pregunta Problema
3. **Justificación**
4. **Objetivo General**
   1. Objetivos Específicos
5. **Metodología**
   1. Fases del proyecto
6. **Marco tecnológico**
7. **Marco Conceptual**
8. **Marco Legal**
9. **Diseños**
   1. **Modelo de datos**
   2. **Diagramas**
      1. Diagrama de clase
      2. Diagrama de casos de uso
      3. Diagrama de secuencia
   3. **Actividades**
   4. **Interfaces**
      1. Vistas de diseño
      2. Visualización de código
10. **Especificación de Requerimientos**
11. **Implementaciones**
    1. Tecnologías
    2. Software
    3. Pruebas
12. **Cronograma**
13. **Conclusiones**
14. **Resultados**
15. **Referencias Bibliográficas**

**Introducción**

El presente documento aborda el desarrollo de un sistema integral de gestión de citas médicas diseñado para optimizar los procesos operativos de un centro médico privado de alta calidad denominado Proyecto Aurora, situado en la localidad de Bucaramanga, Colombia. Este proyecto surge como respuesta a la imperiosa necesidad de implementar una plataforma de asistencia y gestión médica eficiente y especializada, dirigida específicamente al ámbito de la salud, con el objetivo de mejorar la experiencia de los usuarios y garantizar la excelencia en la atención médica brindada. Donde el sistema incluye autorización de órdenes, autorización de exámenes, atención de citas médicas con el médico de su preferencia y especialidad deseada.

Ofrecemos una visión integral que busca armonizar las expectativas de eficiencia, optimización y excelencia en la gestión de citas médicas. Este enfoque integrador tiene como objetivo principal satisfacer las necesidades tanto de los usuarios como del personal médico, garantizando una atención fluida y de alta calidad en todo el proceso de gestión de citas.

El sistema busca atender los desafíos de largas esperas y gestiones tardías de citas y exámenes en el área médica, con el propósito de crear procesos internos en el centro de salud de forma segura, rápida, mejorando la experiencia del paciente. Además, contará con el cumplimiento de los reglamentos legales de la manipulación de datos personales, como son la confidencialidad de la información de cada uno de los usuarios, satisfaciendo las necesidades del cuidado de datos sensibles que se desempeña principalmente en el ámbito a tratar.

A continuación, se detalla la situación con base a la problemática, la justificación y metodología a seguir en el desarrollo del sistema, junto con la presentación de los respectivos marcos tecnológicos, conceptuales y de requerimientos. Asimismo, se incluye un cronograma de actividades diseñado para orientar el proceso de implementación del sistema de gestión de citas médicas bajo el nombre de proyecto Aurora.

**Situación Problema**

En la localidad de Bucaramanga, se identifica la necesidad de implementar un sistema de gestión de pacientes en un centro de salud privado. Enfocándonos en dicha necesidad, es importante establecer este nuevo centro que permita a los residentes de Bucaramanga y gestionar sus citas y exámenes de manera eficiente.

Uno de los principales problemas que pueden surgir en una EPS (Entidades Promotoras de Salud) es la dificultad para manejar adecuadamente el sistema de gestión de citas y exámenes médicos, tanto para el personal como para los pacientes al agendar, programar o cancelar citas y exámenes de distintas especialidades. Estos problemas suelen derivarse de diversos factores, como un mal funcionamiento del sistema, un diseño poco intuitivo para navegar entre las funciones y la falta de capacidad del personal del centro de salud para utilizar el sistema de manera efectiva, entre otros.

Entre los numerosos desafíos que pueden surgir, la gestión eficiente de las colas de espera para citas y exámenes, así como la autorización de órdenes dentro de las especialidades disponibles, destaca como uno de los más importantes. En algunas sedes de salud, la situación actual se caracteriza por largas horas de espera para los pacientes, tanto en la solicitud de citas como en la realización de exámenes necesarios. Esta congestión no solo genera incomodidad e insatisfacción en los usuarios, sino que también cuestiona la eficacia de un sistema de salud en el cual el servicio de atención médica no satisface las necesidades de los usuarios.

Además, la falta de un sistema de autorización de órdenes para exámenes dentro de las especialidades disponibles puede resultar en retrasos en el tratamiento y diagnóstico de los pacientes, lo cual puede tener un impacto negativo en su salud y bienestar.

Este proyecto no solo busca abordar los desafíos actuales en la atención médica, sino también establecer las bases para un modelo de atención centrado en el paciente, donde el sistema y la tecnología desempeñen un papel fundamental. La implementación exitosa de este sistema beneficiará no solo a la comunidad de Bucaramanga en general, sino que también proporcionará una atención médica satisfactoria y accesible en un entorno local.

**Pregunta Problema**

¿Cómo implementar un sistema de gestión de pacientes en Java para el agendamiento de citas, la administración de colas de espera y la autorización de órdenes para exámenes médicos en un centro de salud privado en Bucaramanga, asegurando un despliegue adecuado en el entorno clínico?

**Justificación**

El desarrollo del proyecto busca implementar un sistema de gestión para el apartado de citas médicas y su correspondiente despliegue en un entorno clínico en el área metropolitana de Bucaramanga, con el propósito de mejorar la eficiencia y competitividad en el sector de la salud. La automatización del proceso de toma de citas, autorización y registro de exámenes contribuirá a una expansión en el sistema, mejorando los procesos internos del centro de salud y la experiencia de los usuarios, lo que se traducirá en una mayor satisfacción del cliente. Además, el sistema será escalable para hacer frente al crecimiento de la demanda y futuras expansiones.

La automatización del proceso de agendamiento de citas y exámenes médicos permitirá una atención médica más eficiente, mejorando significativamente la experiencia del usuario/paciente. Al eliminar las largas filas y el tedioso proceso de agendamiento manual, el sistema proporcionará una forma más conveniente para que los pacientes accedan a la atención médica que necesitan.

Además, la implementación de este sistema de gestión de citas médicas aportará conocimientos y beneficios tanto al campo tecnológico como al sector de la salud en general en el área metropolitana de Bucaramanga. Este proyecto servirá como un ejemplo de cómo la tecnología puede transformar y mejorar los procesos en el ámbito de la salud, fomentando la adopción de soluciones innovadoras en otros centros médicos y clínicas de la región.

En este proyecto se contribuirá al avance del campo tecnológico y el mejoramiento del sistema de atención de citas médicas y el sector en general. También buscará elevar los estándares de calidad, eficiencia y accesibilidad en los servicios de salud, beneficiando directamente a los pacientes, al personal médico y administrativo, así como a la comunidad en su conjunto. En resumen, la implementación del sistema de gestión de citas será fundamental para el crecimiento sostenible del proyecto Aurora y la innovación en la atención médica en el área metropolitana de Bucaramanga, permitiendo un mejor uso del tiempo del personal médico y reduciendo los tiempos inactivos, lo que mejorará la productividad del centro de salud.

**Objetivo General**

El objetivo principal de este proyecto es diseñar, desarrollar e implementar un sistema de gestión de pacientes para un sistema integral de atención médica en la ciudad de Bucaramanga, con el fin de mejorar la eficiencia operativa y la calidad de la atención proporcionada. Este sistema estará basado en la metodología de espiral, que permite una adaptación continua a medida que se van identificando y resolviendo los requisitos y desafíos del proyecto.

Utilizando el lenguaje de programación Java y adoptando las mejores prácticas de desarrollo de software, el sistema se diseñará para ofrecer una interfaz intuitiva y amigable para el usuario, garantizando una experiencia fluida tanto para el personal médico como para los pacientes. Además, se emplearán archivos de texto (.txt) para la persistencia y el almacenamiento de datos, asegurando la integridad y confidencialidad de la información del paciente.

El propósito principal de este sistema es automatizar y optimizar el proceso completo de gestión de pacientes en el entorno de salud. Esto incluye, pero no se limita a, el agendamiento eficiente de citas médicas, la autorización oportuna de exámenes clínicos, el registro preciso de información del paciente y la prestación de una atención médica oportuna y de calidad por parte del personal debidamente calificado.

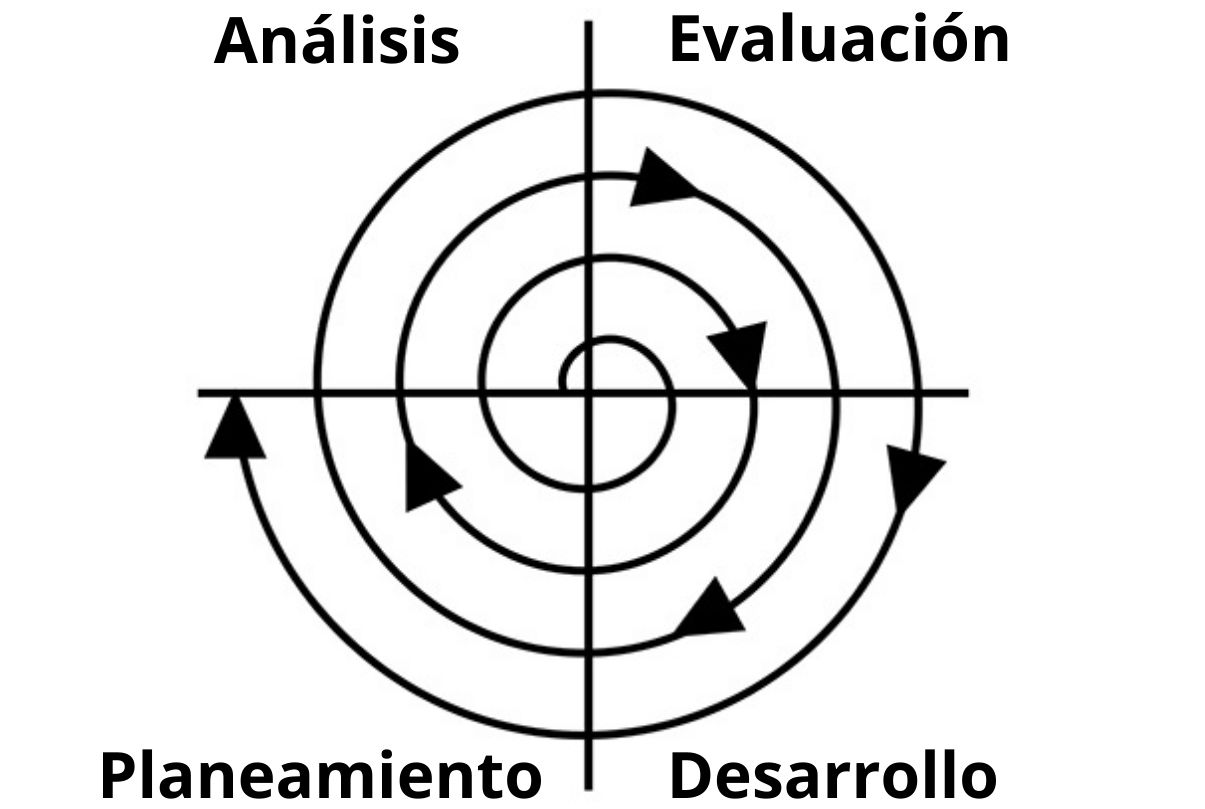
Se espera que este sistema abarque diversas especialidades médicas, permitiendo la gestión integral de pacientes con diferentes necesidades de atención médica. Se priorizará la interoperabilidad con otros sistemas de salud y la integración de estándares de seguridad y privacidad de datos para garantizar la confidencialidad y protección de la información del paciente.

**Objetivos Específicos**

* Definir indicadores de calidad para el proceso de agendamiento de citas, estableciendo métricas para evaluar la calidad del proceso de agendamiento que permitirá identificar áreas que se deben tratar con prioridad y a futuro mejorar con base a cómo evoluciona el sistema y garantizar una atención óptima para los pacientes.
* Diseñar una interfaz visualmente cómoda para los usuarios del sistema donde se busca ir de la mano con la funcionalidad y adaptabilidad de una forma práctica y precisa, el diseño de una interfaz que cubra todas las implicaciones que demandan contribuirá de forma positiva a la eficacia general del sistema y a su buena reputación.
* Desarrollar un prototipo funcional del sistema, centrándonos en los requerimientos que se definen por el cliente, donde la creación de un prototipo funcional permitirá validar y refinar los requisitos del sistema antes de avanzar en etapas que requieren de una atención mayor del desarrollo.
* Optimizar la gestión de citas y registros médicos mediante el desarrollo del sistema, donde la implementación de sistemas de gestión robustos como la aplicación de las estructuras de datos y la programación orientada a objetos/eventos es vital para asegurar la eficiencia y optimización en la gestión de citas y registros sin comprometer la privacidad de los pacientes.
* Evaluar el funcionamiento integral del sistema asegurando el cumplimiento de los requisitos acordados con el cliente y satisfacer las necesidades del centro de salud, esta etapa de evaluación exhaustiva busca garantizar que el sistema cumpla con los estándares acordados y sea capaz de satisfacer las necesidades específicas del centro de salud.
* Mitigar los riesgos potenciales asociados al desarrollo e implementación del sistema y eliminar algún tipo de error, este objetivo es fundamental para identificar posibles obstáculos y establecer estrategias para abordarlos antes de la entrega del producto garantizando una base sólida para el desarrollo del proyecto.
* Garantizar la claridad y precisión en la accesibilidad a los servicios, la interfaz de usuario y el funcionamiento general del sistema deben ser intuitivos y fácil de usar donde es esencial asegurar que los usuarios puedan interactuar con el sistema de manera efectiva y obtener los servicios que necesitan de manera clara, oportuna y precisa.

**Metodología**

La metodología empleada para el desarrollo del sistema es la metodología de espiral. Esta metodología se fundamenta en una planificación meticulosa del proyecto, la identificación y evaluación exhaustiva de riesgos, un enfoque iterativo de desarrollo y una evaluación continua, lo que permite minimizar eficientemente los errores. Avanzar en las tareas solo se llevará a cabo si se completa un sprint con eficacia y se cumplen las expectativas establecidas. El sistema basado en esta metodología permitirá crear un programa de Asistente de Citas Médicas eficiente, completo y con una reducción de los riesgos potenciales durante todo el proceso de creación. Se garantiza, de esta manera, que solo se avanzará a la siguiente fase si la fase actual está completamente finalizada, asegurando así el correcto funcionamiento de la tarea en curso.



**Fases del proyecto**

* **Análisis de Riesgos**

Se identificarán los posibles riesgos asociados al desarrollo e implementación del sistema. Entre los riesgos potenciales se pueden incluir problemas de carga de datos, retrasos en la asignación de exámenes u citas, fallos en la integración de la sala de espera, suplantación de identidad, entre otros.

Se evaluará el impacto y la probabilidad de ocurrencia de cada riesgo de manera rigurosa, utilizando matrices de riesgo u otras herramientas apropiadas. Se diseñarán planes de contingencia y mitigación detallados para reducir su impacto en el proyecto.

* **Evaluación y planificación**

Se llevarán a cabo planificaciones y evaluaciones detalladas. En estas planificaciones se definirán funcionalidades que debe incluir y los requisitos necesarios para su correcto funcionamiento y aplicación de estructuras sólidas para garantizar una correcta modularidad y mantenimiento.

Se establecerá un cronograma para el desarrollo del sistema y se asignarán los recursos necesarios para llevar a cabo el proyecto. Además, los roles y responsabilidades de los miembros del equipo encargados del desarrollo serán de forma donde contribuirán de forma colectiva para una mejora continua desde distintos puntos.

Asimismo, se analizarán minuciosamente los requisitos del sistema, centrándose en las necesidades específicas de un Asistente de Citas Médicas, y se recopilará información detallada sobre las funcionalidades requeridas.

* **Desarrollo y prueba**

Se seguirá un enfoque iterativo comenzando con la definición de los requisitos del usuario y la especificación de las funcionalidades clave del sistema.

Se procederá a la creación de un prototipo inicial del sistema, y se realizarán ciclos de refinamiento y mejora para garantizar la calidad del producto final.

Durante esta etapa, se implementarán funcionalidades como el Agendamiento de Citas, la autorización de exámenes específicos, el registro de usuarios, el seguimiento médico, entre otros.

* **Planeamiento**

Se llevarán a cabo pruebas exhaustivas y planificaciones detalladas. Todas las funcionalidades del sistema serán probadas para identificar problemas y oportunidades de mejora. Se implementarán las correcciones y mejoras necesarias para asegurar que el sistema funcione de manera óptima y cumpla con los requisitos establecidos en la fase de planificación. Además, se realizará una evaluación continua del rendimiento del sistema utilizando métricas relevantes, como el tiempo de procesamiento para agendar citas, la asignación correspondiente de profesionales al programar un examen y la eficacia de los resultados obtenidos.

**Marco tecnológico**

El presente marco tecnológico se aborda el panorama actual de las tecnologías que están en juego dentro del proyecto. En un mundo donde la innovación tecnológica avanza a un ritmo vertiginoso, es fundamental comprender las herramientas, plataformas y sistemas que impactan directamente para la correcta realización de un asistente de citas médicas que cubran todos los requerimientos demandados por el cliente e ir más allá de la satisfacción de este mismo.

1. **Plataforma de Desarrollo**

* **IDE,** Un entorno de desarrollo integrado (IDE) es un sistema de software para el diseño de aplicaciones que combina herramientas comunes para desarrolladores en una sola interfaz gráfica de usuario (GUI). Se utiliza el IDE NetBeans para el despliegue de la aplicación. Sin embargo, cada participante en el desarrollo del código del programa.

1. **Entorno de Desarrollo integrado (IDE)**

* **NetBeans,** Es un entorno de desarrollo integrado, gratuito y de código abierto para el desarrollo de aplicaciones en los sistemas operativos Windows, Mac, Linux y Solaris. Desarrollaremos el proyecto y realizaremos el completo despliegue en el IDE NetBeans. Siendo así, un IDE popular y de positiva confianza para el desarrollo de aplicaciones Java que ofrece múltiples ventajas y herramientas útiles para la edición de código, depuración y pruebas.

1. **Lenguaje de programación**

* **Java**, Java es un tipo de lenguaje de programación y una plataforma informática, creada y comercializada por Sum Microsystems en el año 1995. Es funcional para este proyecto, ya que se constituye como un lenguaje orientado a objetos el cual está fuertemente equipado, permitiendo que los desarrolladores implicados en la creación de este sistema escriban el programa una sola vez y lo ejecuten en cualquier sistema operativo y es altamente adoptado en el desarrollo empresarial ofreciendo una plataforma confiable y escalable.

1. **Control de versiones**

* **Git,** Git es una herramienta que realiza el sistema de control de versiones de código de forma distribuida. Es de código abierto, con mantenimiento activo y es la herramienta de este tipo más empleada en el mundo. Utilizaremos Git como el sistema de control de versiones para rastrear y administrar los cambios en el código, creando así diferentes ramificaciones para mantener un historial de desarrollo confiable y sólido, y asegurar la fiabilidad del proyecto.

1. **Diseños de interfaces**

* **Figma,** Esta es una herramienta para diseñar prototipos, wireframes e interfaces. Todo aquello que posea una interfaz gráfica se puede diseñar desde Figma, ya sean páginas web, pantallas de móvil e incluso para smartwatches. Utilizaremos Figma para diseñar interfaces gráficas de usuario (GUI) del sistema. Figma permite crear prototipos interactivos y diseños de alta calidad que facilitan la comunicación visual entre el equipo de diseño y el desarrollo. Y aunque es una plataforma para el diseño más que todo implementado en aplicativos webs, se usó para el trabajo en tiempo real con las personas pertenecientes al equipo de trabajo.
* **IntelliJ IDEA,** Este es un entorno de desarrollo integrado(IDE) que ofrece características avanzadas para el desarrollo de aplicaciones Java, incluidas las aplicaciones de escritorio con JavaFX. Tiene un editor de GUI visual que te permite diseñar interfaces de usuario de forma visual y luego general el código correspondiente. Se aplicó también la creación de distintas vistas desde otras plataformas de desarrollo para ver su capacidad de creación de diseño.
* **JavaFX,** Esta es una plataforma de software donde las funcionalidades para la creación de interfaces de usuario modernas y dinámicas en aplicaciones de escritorio. Proporciona una variedad de componentes gráficos, animaciones, efectos visuales y herramientas para el desarrollo de aplicaciones interactivas. Permitiendo diseñar y obtener el código del diseño implementado.

1. **Persistencia de la información**

* **Archivos de texto,** Los archivos de texto son una forma común y sencilla de almacenar la información de manera legible para los humanos y fácilmente manipulable por los dispositivos, en este caso las computadoras. Son ampliamente utilizados en una variedad de aplicaciones debido a su simplicidad y portabilidad. Los datos almacenados en archivos de texto pueden ser fácilmente editados, compartidos, almacenados de forma constante entre diferentes sistemas y aplicaciones. Además, los archivos de texto son adecuados para almacenar información estructurada o no estructurada, como registros de eventos, configuraciones de aplicaciones, datos de configuración, entre otros. Su simplicidad y práctica y versatilidad los convierten en una opción popular para la persistencia de datos informáticos.

**Marco Conceptual**

Este marco conceptual proporciona una base fundamental para comprender los elementos clave relacionados con el funcionamiento y los desafíos inherentes al proyecto para el correcto funcionamiento de este donde se pueden identificar áreas de mejora, diseñar estrategias para aumentar la eficiencia y mejorar el proceso de agendamiento de citas y asignación de exámenes médicos yendo de la mano con la satisfacción del cliente.

* El ciclo de vida del desarrollo de sistemas (SDLC) es aquel proceso que abarca el desarrollo de nuevo software desde la etapa de planificación inicial hasta la implementación y el mantenimiento a largo plazo. Es una herramienta de mapeo que ayuda a los desarrolladores de software a medida que crean un nuevo software.
* Esto yendo de la mano con la metodología de espiral que fue aplicada, permite hacer una combinación entre el modelo WaterFall y un modelo por iteraciones. Donde el proceso pasa por distintas etapas, desde la de conceptualización siguiendo el desarrollo, luego una fase de mejoras, para finalizar con el mantenimiento y así haciendo el proyecto de una forma más escalable y práctica.
* Aplicando a su vez las estructuras de datos para la creación de un proyecto mucho más modular y sólido. Implementando las ventajas como son la aplicación de las Listas enlazadas, que son aquellas que generan un vínculo entre una lista de elementos importantes de un programa. Organizando el orden de la información dentro del sistema cuando se ejecuta un programa de gestión de información, a su vez la aplicación de Listas doblemente enlazadas, pilas, colas las cuales nos permiten avanzar como retroceder. Cada nodo de la lista replicada tiene dos referencias, una al siguiente nodo y otra al anterior, además del campo de datos. Se utiliza un enlace a la derecha para avanzar. Otros enlaces a la izquierda que se utiliza para regresar, a excepción de una pila que usa un sistema LIFO (Last In First Out). Que como lo indican sus siglas en inglés es el último en entrar, es el primero en salir, entonces se podría decir que el último elemento en entrar desplaza a todos los demás posicionándose en la cabeza y así sucesivamente. Cola es un sistema FIFO(First In First Out) donde el primero en entrar es el último en salir, permitiendo un funcionamiento conforme al orden de llegada.
* Permitiendo una estructura sólida para la accesibilidad, monitorización, escalabilidad, manejo de datos de forma adecuada y una seguridad de la información completa. Ya que el sistema permite un acceso a sus funcionamientos de una forma práctica para adaptarse a cualquier situación con una monitorización eficiente para hacer el proceso de seguimiento del rendimiento para identificar y solucionar problemas yendo de la mano con la escalabilidad para adaptarse y evolucionar como proyecto para un mayor número de usuarios y a una mayor demanda de servicios con un manejo de datos eficiente y práctico permitiendo los procesos que se utilizan para el manejo de datos de forma eficiente haciendo la debida protección de los datos sensibles de las personas, protegiéndolos contra errores, daños o accesos no autorizados.
* Y todo esto, teniendo siempre presente el Diseño centrado en el usuario (UCD) para garantizar así la completa comprensión y funcionalidad del programa, donde se basa en comprender las necesidades, expectativas y comportamientos de los usuarios para crear productos y servicios que se ajusten a ellos.

**Marco Legal**

El marco legal que rige el desarrollo y la implementación del sistema para la atención médica desempeña un papel crucial en la garantía de la calidad, la seguridad y la protección de los derechos de los pacientes y los profesionales de la salud. En un entorno delicado y sensible como el sector de la salud, el cumplimiento de las leyes y regulaciones es esencial para salvaguardar la integridad y la confianza en los servicios prestados.

El marco legal presente en este documento abarca una amplia gama de áreas, desde la protección de datos personales y la confidencialidad médica hasta las normativas específicas para el desarrollo del software en el ámbito de la salud. Cada una de estas áreas tiene como objetivo asegurar que el sistema cumpla con los estándares éticos y legales necesarios para el correcto funcionamiento.

Al comprender y adherirse a nuestro marco legal aplicable, los desarrolladores y proveedores de servicios de salud se pueden mitigar riesgos legales, evitar sanciones y, lo que es más importante, proteger la privacidad y derechos de los pacientes. En esta introducción al marco legal, se explorarán las principales leyes y regulaciones que deben tenerse en cuenta en el desarrollo e implementación de un sistema relacionado con la salud, así como las implicaciones prácticas de cumplir con estas normativas en la práctica clínica y administrativa.

1. **Confidencialidad médica**

La confidencialidad médica es importante para los pacientes y sus profesionales de la salud. Con solo unas pocas excepciones, todo lo que hable con su médico debe, por ley, mantenerse en privado entre ambas partes como usuario y médico y la organización para la que trabajan o donde se emplea el servicio. Esto también se conoce como confidencialidad médico-paciente.

1. **Privacidad en la atención sanitaria**

Bajo las leyes de privacidad en la atención sanitaria, los pacientes tienen el derecho fundamental a controlar quién puede acceder a su información médica, cómo se utiliza y divulga, y con qué fines. Esta protección se extiende a todos los aspectos relacionados con la atención médica, incluyendo el almacenamiento de registros médicos, la comunicación con proveedores de servicios de salud, y la participación en investigaciones médicas.

1. **Cumplimiento Normativo**

Hay que asegurar que el sistema cumpla con todas las regulaciones y normativas aplicables en el ámbito de la salud, incluyendo leyes locales, regionales y nacionales, así como estándares internacionales si son relevantes. Como lo dice La Ley Estatutaria De Salud (Ley 1751 de 2015). Esta ley establece los principios y los derechos relacionados con la salud en Colombia, incluyendo, la participación ciudadana en el sistema de salud, entre otros aspectos fundamentales. Esto implica asegurar que el sistema cumpla con los estándares de calidad y seguridad requeridos, protegiendo los derechos de los pacientes y promoviendo una atención médica equitativa y accesible para todos los ciudadanos colombianos.

1. **Ley General de Salud (Ley 1438 de 2011)**

La Ley 1438 de 2011 introduce importantes cambios en el sistema de salud colombiano, con el objetivo de mejorar la calidad, la eficiencia y la equidad en la prestación de servicios de salud. Entre sus disposiciones, la ley establece la creación del Sistema General de Seguridad Social en Salud (SGSSS), que tiene como objetivo garantizar el acceso universal a los servicios de salud y proteger el derecho a la salud de todos los ciudadanos colombianos.

Además, la Ley 1438 de 2011 aborda temas como la regulación de la prestación de servicios de salud, la gestión de los recursos en el sistema de salud, la promoción de estilos de vida saludables, la participación ciudadana en la toma de decisiones en salud, entre otros aspectos relevantes para el funcionamiento del sistema de salud en Colombia.

1. **Ley de protección de datos personales (Ley 1581 de 2012)**

Esta ley establece los principios y disposiciones para la protección de los datos personales en Colombia, incluyendo la recolección, almacenamiento, uso, circulación y supresión de dichos datos, lo cual es especialmente relevante en el contexto de la gestión de información de pacientes en el Software.

1. **Resolución 1995 de 1999 de Ministerio de Salud y protección Social**

Esta resolución establece los requisitos técnicos y operativos que deben cumplir los establecimientos de salud en Colombia, incluyendo aspectos relacionados con la gestión de la información de pacientes y la interoperabilidad de sistemas de información en salud.

1. **Normas internacionales de información de Salud (CIE-10, CIE-11)**

Estas normas internacionales establecen un marco para la codificación de enfermedades, procedimientos y otros conceptos relacionados con la salud, lo cual puede ser relevante para interoperabilidad y la estandarización de datos en el software.

1. **Protección de datos personales**

Garantizar el cumplimiento de las leyes de protección de datos personales, como el reglamento General de protección de datos (GDPR) en la Unión Europea o leyes similares en otras jurisdicciones, para proteger la información personal de los pacientes.

1. **Consentimiento informado**

Establecer procedimientos para obtener el consentimiento informado de los pacientes para el uso y la divulgación de su información médica, de acuerdo con las leyes y regulaciones aplicables.

1. **Seguridad de la información**

Implementar medidas de seguridad robustas para proteger la información de salud de los pacientes contra accesos no autorizados, divulgación indebida, alteración o destrucción, de acuerdo con las prácticas y estándares de seguridad de la industria.

1. **Responsabilidad legal**

Definir claramente las responsabilidades legales de todas las partes involucradas en el desarrollo, implementación y so del sistema, incluyendo el proveedor del sistema, los profesionales de la salud, y los usuarios finales.

1. **Registro y auditoria**

Establecer procedimientos para el registro y la auditoria de todas las actividades relacionados con el sistema, incluyendo el acceso a la información de salud de los pacientes, para garantizar la transparencia y la rendición de cuentas.

1. **Notificación de brechas de seguridad**

Establecer protocolos para la notificación oportuna de cualquier brecha de seguridad que pueda afectar la privacidad o seguridad de la información de salud de los pacientes, de acuerdo con las leyes y regulaciones aplicables.

1. **Ley de 1551 de 2015**

Una de las leyes fundamentales que respaldan estos principios es la Ley Estatutaria de Salud en Colombia, Esta ley establece los principios y derechos relacionados con la salud en el país, incluyendo la garantía de acceso a servicios de salud de calidad, la protección de los derechos de los pacientes, la participación ciudadana en el sistema de salud, entre otros aspectos relevantes.

Además, el Código de Ética Médica Colombiano, establecido por la Asociación Médica Colombiana, también aborda principios éticos y estándares de práctica clínica que deben seguir los profesionales de la salud en el país. Este código incluye disposiciones sobre el consentimiento informado, la confidencialidad médica, la relación médico-paciente, entre otros aspectos importantes para garantizar una atención médica ética y segura.

1. **Ley 39 de 1990**

La ley colombiana que establece que no pueden existir dos personas con el mismo documento de identidad es la Ley 39 de 1990, específicamente en su artículo 11. Esta ley regula el Registro Civil y establece los principios y procedimientos para la identificación de las personas en Colombia. El artículo 11 de esta ley establece que "en ningún caso se asignará a una persona un número de identificación que haya sido asignado previamente a otra". Esto significa que el número de identificación, como el documento de identidad (cédula de ciudadanía o tarjeta de identidad), debe ser único para cada individuo en Colombia.

**Diseños**

Los diseños en este proyecto desempeñan un papel crucial en el éxito del proyecto Aurora, ya que proporciona la estructura y el plan detallado para la implementación del sistema de gestión de citas médicas. Estas etapas de diseño son fundamentales para traducir los objetivos del proyecto en soluciones tangibles y funcionales. A través de los diseños, se establecerán los fundamentos para una implementación eficaz y eficiente del sistema, garantizando que cumpla con las necesidades y expectativas de los usuarios finales.

* **Aspectos clave de los diseños en el proyecto**

Las etapas específicas de diseño, aborda diversas etapas de diseño, incluyendo el diseño de la arquitectura del sistema, el diseño de la interfaz de usuario y el diseño de la persistencia de la información y como se transmitirá la estructura del programa con los miembros del equipo haciendo el uso de diagramas de clase, entre otros. Cada etapa será cuidadosamente planificada y ejecutada para garantizar la coherencia y funcionalidad del sistema en su conjunto.

Donde el enfoque en la usabilidad y experiencia del usuario se pondrá un énfasis especial en la usabilidad y la experiencia del usuario durante el diseño del sistema. Se implementarán principios de diseño centrado en el usuario para crear una interfaz intuitiva y fácil de usar que mejore la experiencia del paciente en la gestión de citas médicas.

Por otro lado, el enfoque en la escalabilidad y flexibilidad donde se diseñarán un sistema que sea escalable y flexible para adaptarse a futuras necesidades y cambios en el entorno de atención médica. Se considerarán cuidadosamente aspectos de escalabilidad y flexibilidad para asegurar la viabilidad a largo plazo del sistema.

**Modelo de datos**

El modelo de datos que se presenta a continuación es el resultado del análisis que se hizo a las necesidades y desafíos identificados en el proceso de implementación de un sistema de gestión de pacientes. Este modelo se ha desarrollado con el objetivo de proporcionar una estructura sólida y eficiente para la gestión de citas médicas y exámenes y los demás requisitos solicitados por el cliente, con el fin de mejorar la calidad y la accesibilidad de la atención médica para los residentes de Bucaramanga.

Para satisfacer de forma efectiva en el entorno de salud que es dinámico y exigente, la gestión eficiente, pago de citas, cancelación, registro de autorizaciones y exámenes es fundamental para garantizar la satisfacción de los usuarios y la eficacia del sistema de salud en su conjunto.

**Como objetivo,** de este modelo de datos presentado tiene como objetivo principal proporcionar una estructura coherente y bien definida para la gestión de datos relacionados con citas médicas y exámenes en el centro de salud. Permitiendo así una eficiente gestión, facilitando la programación, la coordinación y el seguimiento de las actividades relacionadas. Además, el modelo se diseñará teniendo en cuenta la escalabilidad y flexibilidad, para adaptarse a las necesidades cambiantes del sistema, garantizando así su viabilidad a largo plazo.

**Entidades Principales**

1. **Paciente/Usuario:** Representa a los individuos que solicitan citas médicas y exámenes.

* **Atributos:** Nombre, apellido, identificación, edad, teléfono, contraseña.

1. **Cita Médica:** Representa una cita programada entre un paciente y un médico.

* **Atributos:** Nombre, apellido, identificación, identificador, ticket, costo, doctor, estado, especialidad, asistencia, motivo.

1. **Autorización:** Representa la autorización para los exámenes solicitados para un paciente.

* **Atributos:** Nombre, apellido, identificación, identificadordelacita, ticketdelacita, precio, tipoExamen, identificadordeautorizacion, descripción, Estado, ticketDeAutorizacion.

1. **Examen Médico:** Representa un examen médico solicitado por un paciente para ser atendido en la categoría de exámenes. Para esta solicitud necesita previamente una cita médica de tipo control para gestionar que su estado de salud no se verá afectado después de los exámenes y una autorización exitosa para ser atendido con éxito.

* **Atributos:** nombreUsuario, apellidoUsuario, identificacionDeExamen, ticketDeExamen, costo, tipoExamen, descripción, estadoDePago, asistencia

**Relaciones**

1. **Relación Paciente - Cita Médica:** Un paciente puede tener múltiples citas médicas independientemente de su especialidad, motivo u doctor. Y cada cita médica está asociada a un único paciente.
2. **Relación Pago de Cita – Cita Médica:** Usted para ser atendido necesita previamente pagar su cita médica a no ser que se trate de un control que por defecto viene pago por ser gratuito.
3. **Relación Cita Médica - autorización:** Un paciente puede tener múltiples autorizaciones médicos, y cada autorización está asociada a un único paciente con su debida cita de control para apoyarnos en el marco legal evitando posibles situaciones donde se afecte la vida del usuario por practicar exámenes donde no estaba completamente bien su estado para ser tratado con base a un examen.
4. **Autorización - Examen Médico:** Un paciente puede tener múltiples exámenes médicos, y cada examen médico está asociado a un único paciente. Para ser atendido con un examen usted necesita una autorización exitosa para poder ser atendido con un examen.

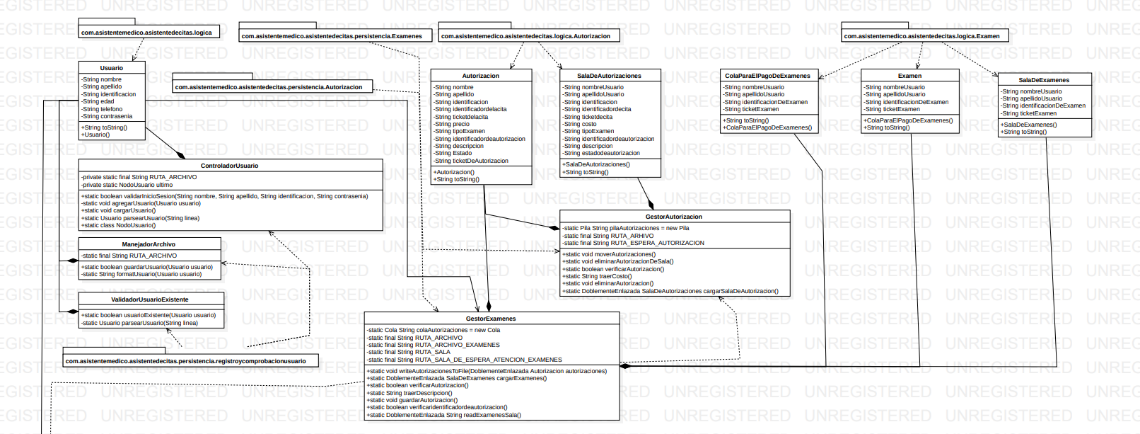
* **Consideraciones Adicionales**
* Se podría agregar una tabla para gestionar los médicos, incluyendo su nombre, especialidad.
* Se puede incluir tablas adicionales para la visualización de las citas , autorizaciones o exámenes relacionadas con el usuario.
* Se puede establecer restricciones y reglas de negocio para garantizar la integridad y la consistencia de los datos en el modelo, como términos y condiciones.

**Diagramas**

Los diagramas en este proyecto desempeñan un papel fundamental en el desarrollo y la implementación del proyecto Aurora, proporcionando herramientas visuales que permiten una comprensión clara y estructurada de los diferentes aspectos del sistema de gestión de citas médicas. En particular, los diagramas de clase, caso de uso y de secuencia son elementos esenciales que ayudan a definir la arquitectura, la funcionalidad y la interacción del sistema de manera precisa y detallada.

**Diagrama de clase**

El diagrama de clase presente permite representar nuestra estructura estática del sistema, mostrando las clases, atributos, métodos y relaciones entre ellas. En el contexto del proyecto Aurora, el diagrama de clase es crucial para definir la organización y la interacción de los diferentes componentes del sistema, incluyendo entidades principales como usuario, citas médicas, autorizaciones y exámenes médicos. Además, los diagramas de clase proporcionan una base sólida para el diseño de la persistencia de datos y la implementación del modelo de datos.

****

**Imagen de la pantalla de un computador

Descripción generada automáticamente con confianza baja**

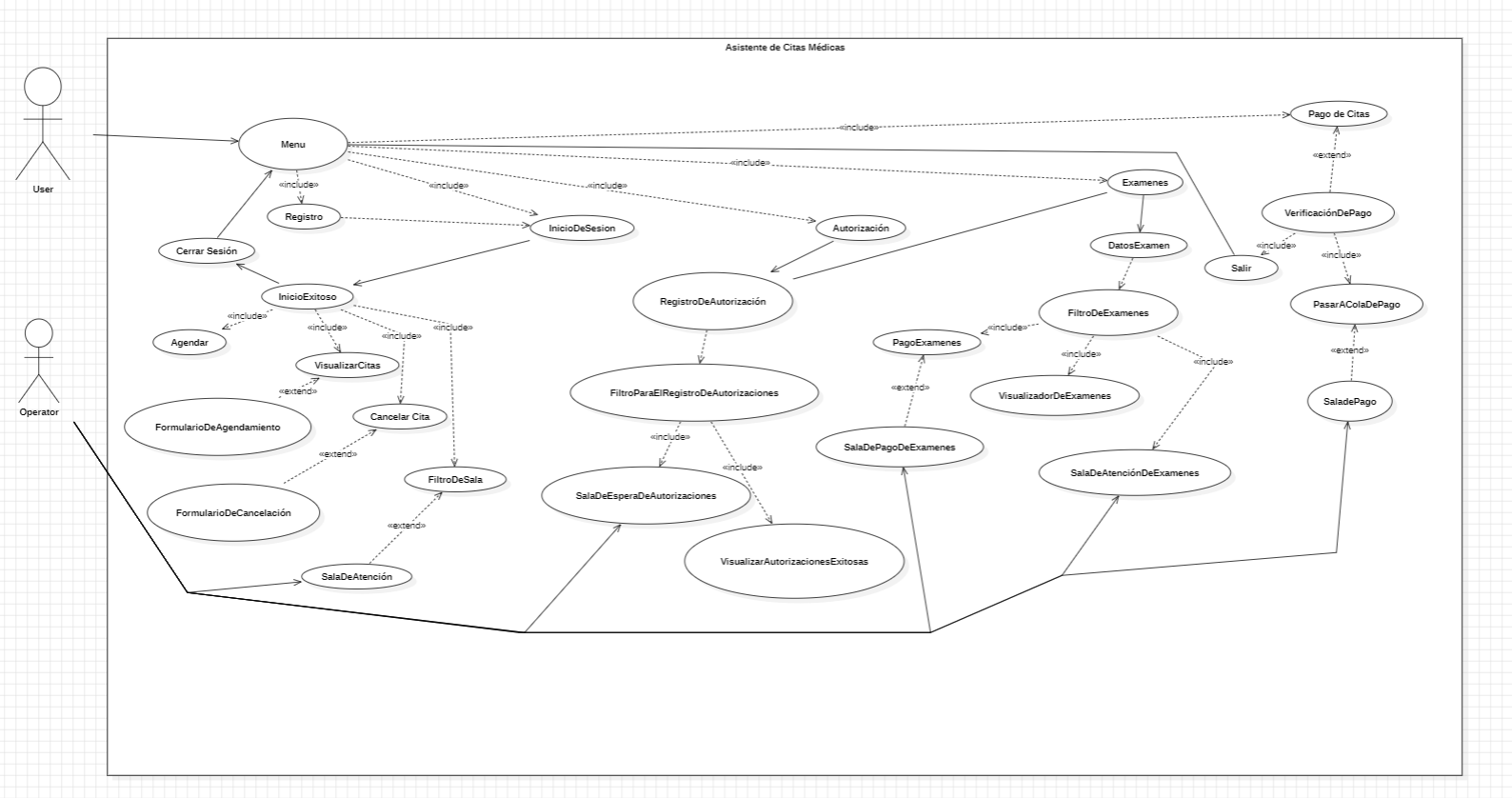
**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Diagrama de casos De Uso**

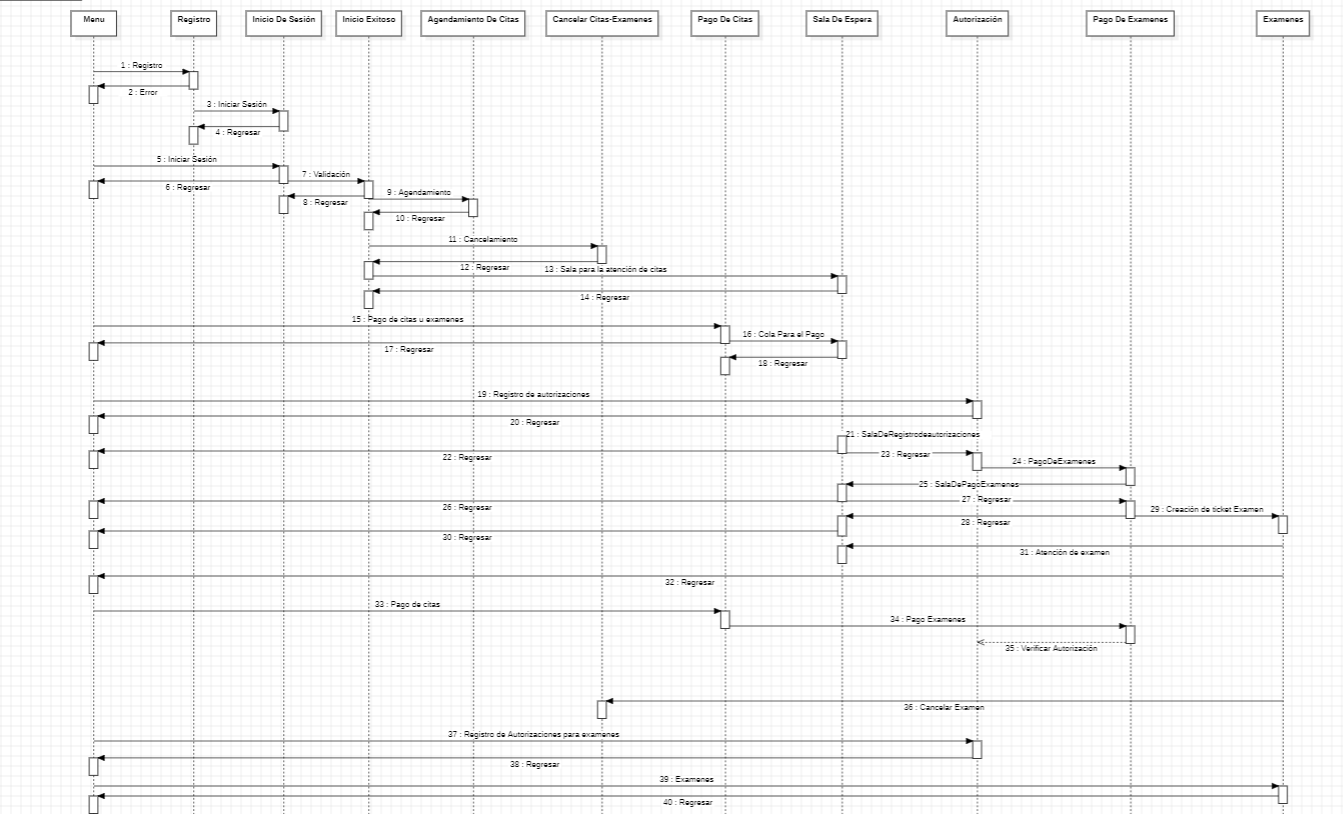
El presente diagrama de casos de uso muestra una seria de casos de uso que describen las funcionalidades y los procesos clave del sistema. Los actores externos que interactúan con el sistema se representan en el borde del diagrama con los nombres de User y Operator. Mientras que los casos de uso se muestran en el interior de este.

Cada caso de uso describe una interacción específica entre un actor y el sistema, identificando las acciones que se realizan y los resultados esperados. Los casos de uso se organizan de manera lógica y se agrupan en función de las funcionalidades relacionadas, lo que permite una comprensión clara y estructurada de las capacidades del sistema.

El objetivo del presente diagrama de casos de uso es proporcionar a una descripción completa y detallada de las funcionalidades del sistema desde la perspectiva del usuario y el operador. Esto incluye identificar las diversas tareas que los usuarios puedan realizar y las tareas que el operador puede intervenir, los requisitos funcionales del sistema y las interacciones entre los usuarios y el sistema. Dando así un funcionamiento como herramienta de comunicación efectiva entre los diferentes interesados en el proyecto, permitiendo una alineación clara de las expectativas y requisitos del sistema.

**Diagrama de secuencia**

El diagrama de secuencia de esta sección muestra una serie de eventos y las interacciones que ocurren entre los objetos del sistema en respuesta a los eventos. Los objetos se representan en el diagrama, junto con las líneas de mensaje que indican la comunicación entre ellos. Estas líneas de mensaje muestran el orden y la dirección en la se envían los mensajes entre los objetos, así como cualquier dato adicional que se transmita.

Cada interacción se representa secuencialmente en el diagrama, lo que permite visualizar el flujo de control y los intercambios de información entre los objetos del sistema. El objetivo principal y especifico de este diagrama de secuencia es proporcionar la representación detallada y práctica y clara de la dinámica del sistema, mostrando como los diferentes objetos colaboran entre sí para lograr ciertos objetivos. Esto incluye identificar los procesos y las interacciones clave del sistema, así como comprender la lógica y el flujo de control detrás de estas interacciones. El presente diagrama cumple la funcionalidad para la identificación de posibles problemas u errores en el diseño del sistema, así como para comunicar de manera efectiva requisitos y especificaciones.

**Actividades**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | ACTIVIDAD | DESCRIPCIÓN | PRIORIDAD |
| 1 | Análisis | Se realiza la respectiva investigación sobre el sistema que se va a desarrollar, sus funcionalidades, requerimientos, para una mejor organización en el momento de realizar el proyecto | ALTA |
| 2 | Primera edición del documento | Se comenzó a editar el documento las primeras características como introducción, situación problema, pregunta problema, justificación y objetivos. | MEDIA |
| 3 | Metodología y documentación | Se conversa acerca de la metodología a emplear para avanzar en el proyecto, la cual fue la metodología espiral, una vez decidida se comienza a aplicar en la realización del proyecto y se agrega a la documentación, | ALTA |
| 4 | Segunda edición del documento | Se agregan los marcos, requerimientos, cronograma y bibliografía, de esta forma se finaliza los aspectos necesarios para la primera entrega del documento | ALTA |
| 5 | Primer avance del sistema de citas medicas | El primer avance consistió en la creación de las interfaces para cada una de las funcionalidades a implementar. | MEDIA |
| 6 | Segundo avance del sistema de citas medicas | Se implementaron funcionalidades principales y adicionales, como sala de espera, autorización de exámenes, pago de citas, registro de autorizaciones, atención de exámenes, cancelación de citas y exámenes. | ALTA |
| 7 | Prueba de las funcionalidades implementadas | Se realizan pruebas para comprobar la funcionalidad del código y mitigar el riesgo de continuar con la creación del sistema, para esto se hace la prueba de cada funcionalidad por separado. | MUY ALTA |
| 8 | Tercer avance del sistema de citas medicas | Se empezaron a implementar otros requerimientos documentados como la interfaz para el inicio de sesión, configuración y selección del tipo de cita, confirmación de la cita, modificación de la cita designada. Buscando una sincronización final entre funcionalidad y algo practico. | ALTA |

**Interfaces**

Las interfaces juegan un papel fundamental tanto visual como en estructura organizada de código en el proyecto Aurora al proporcionar el punto de interacción entre los usuarios y el sistema de gestión de citas médicas. Estas interfaces son la cara visible del sistema, permitiendo a los usuarios acceder y utilizar las diferentes funcionalidades de manera intuitiva y eficiente. En el contexto de este proyecto, las interfaces son elementos clave que facilitan la comunicación y la interacción entre los usuarios y el sistema, asegurando una experiencia de usuario óptima y satisfactoria.

La importancia de una correcta aplicación de interfaces y patrones de diseño son una parte integral de la implementación del sistema de gestión de citas médicas en el proyecto Aurora. Su importancia radica en varios aspectos clave como son:

* La facilidad de la usabilidad, donde esto se puede obtener por medio de interfaces bien diseñadas que permiten a los usuarios interactuar de manera fácil y eficiente con el sistema, facilitando la navegación y el acceso a las funcionalidades necesarias para gestionar citas médicas y exámenes.
* Optimizan la productividad, ya que una interfaz eficiente y bien diseñada ayudan a agilizar los procesos de gestión, lo que resulta en una mayor productividad y eficiencia tanto para los usuarios como para el personal médico.
* Garantizan la accesibilidad, ya que interfaces accesibles y adaptativas aseguran que el sistema pueda ser utilizado por una amplia gama de usuarios, incluyendo aquellos con diferentes niveles de habilidad y capacidades.
* Y uno de los factores más importantes que es el mejoramiento de la experiencia del usuario, donde si es una interfaz intuitiva y atractiva contribuye a una experiencia positiva, aumentando la satisfacción y la aceptación del sistema por parte de los usuarios finales, como pacientes y personal médico.

**Vistas de Diseño**

Las vistas de diseño desempeñan uno de los papeles más importantes en el proyecto ya que son esenciales ya que actúan como el punto de interacción principal entre los usuarios y el sistema de gestión de citas médicas. Estas interfaces representan la interfaz visual a través de la cual los usuarios pueden acceder y utilizar las diversas funcionalidades proporcionadas por el sistema. Son la ventana a través de la cual los usuarios pueden interactuar de manera efectiva y eficiente con las capacidades que ofrecen el sistema.

En el contexto aurora, las interfaces son más que simples representaciones visuales; también están diseñadas para proporcionar diferentes puntos de vista y una estructura organizada en el código subyacente. Esto garantiza una coherencia tanto en la apariencia visual como en la funcionalidad del sistema, lo que contribuye a una experiencia de usuario cohesiva y satisfactoria.

Estas interfaces están cuidadosamente diseñadas para ser intuitivas y fáciles de usar, lo que permite a los usuarios navegar sin problema a través de las diferentes características del sistema. Se han considerado principios de diseño de experiencia de usuario (UX) para garantizar que la interacción con el sistema sea lo más fluida y natural posible.

1. **Ventana Principal/LOBBY**

Frame de ventana Principal, este Frame es la pieza central de la interfaz y por donde la entrada del programa al ejecutarse es el primero en iniciar. Es la pieza central de la interfaz de usuario, que proporciona acceso a las principales funciones y características del sistema. Su diseño minimalista y elegante incorpora una paleta de colores que incluye el tono de gris, blanco, azul marino y verde suave. También un clip de bienvenidos para hacerlo de una forma dinámica y una imagen propia de logo y otra imagen que llena un vacío, creando una atmosfera serena y moderna. Además de ser estéticamente atractivo, este Frame está diseñado

****

1. **Ventana Registro/Inicio De Sesión**

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamenteEn sección se proporciona a los usuarios la capacidad de crear una cuenta en el sistema, permitiéndoles acceder a todas las funciones y características disponibles. Con una paleta igual a la del Principal ya que esa paleta se manejó en todo el sistema. En este proceso de registro se solicita parámetros al usuario como se notan en la imagen y un acepto de términos y condiciones para ir de la mano con el marco legal y su correcta aplicación con base a la ley del trato de datos sensibles del usuario. A parte en la sección de inicio de sesión siguiendo la misma estética solicitando 4 parámetros para acceder a funciones que se especificaran a continuación.



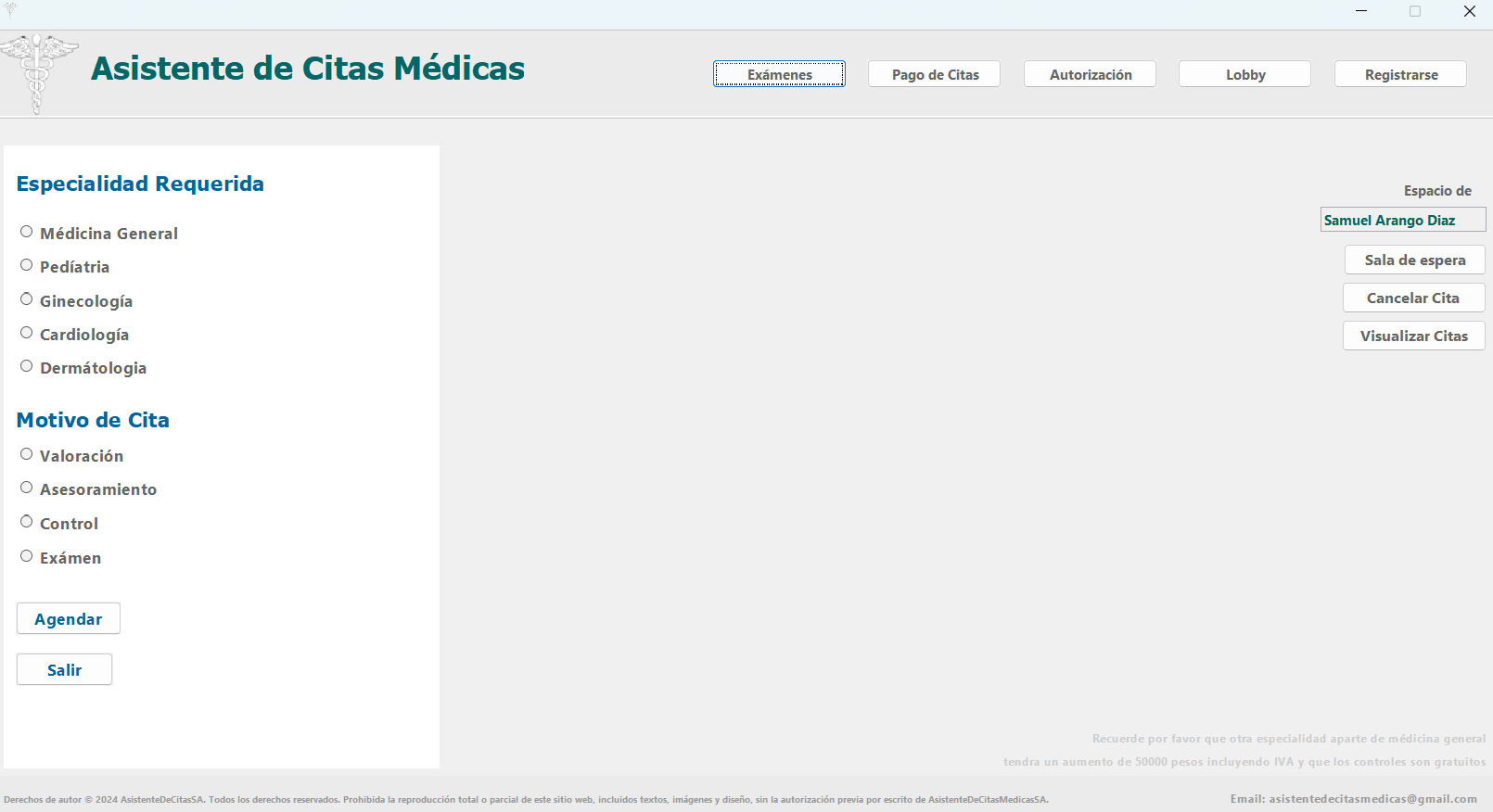
1. **Entrada Home de inicioExitoso**

Esta página de inicio da la bienvenida a los usuarios de forma exitosa después de iniciar sesión en el sistema. Al acceder con éxito, los usuarios son dirigidos a la página principal, donde se les presentan una serie de nuevas opciones y funcionalidades para explorar.

Las opciones disponibles después de iniciar Sesión exitosamente incluyen:

1. Agendamiento de citas
2. Cancelación de citas
3. Visualizar citas a nombre del usuario (Historia Clínica)
4. Sala de espera

Como diseño adicional se aplicó que fuera un espacio personalizado como se puede apreciar teniendo una persistencia de quien es el usuario que está haciendo las acciones en este caso Samuel Arango Diaz.



1. **Agendamiento de citas**

El agendamiento de citas como se aprecia proporciona a los usuarios la capacidad de programas nuevas citas de manera rápida y eficiente. Esta es una parte que se encuentra 1 anterior a la confirmación de la cita, pasando por diferentes filtros inicialmente donde el usuario debe seleccionar la especialidad y el motivo luego el doctor generando así el ticket con base a lo seleccionado el identificador de la cita y el costo respectivo. Inicialmente para todas las citas excepto para los controles el botón de pagar estará inicialmente desactivado ya que obligatoriamente se debe pasar a pago de citas para hacer el respectivo pago excepto para los controles donde este flag estará activado de pago. A continuación, se visualizará también el agendamiento exitoso.

****

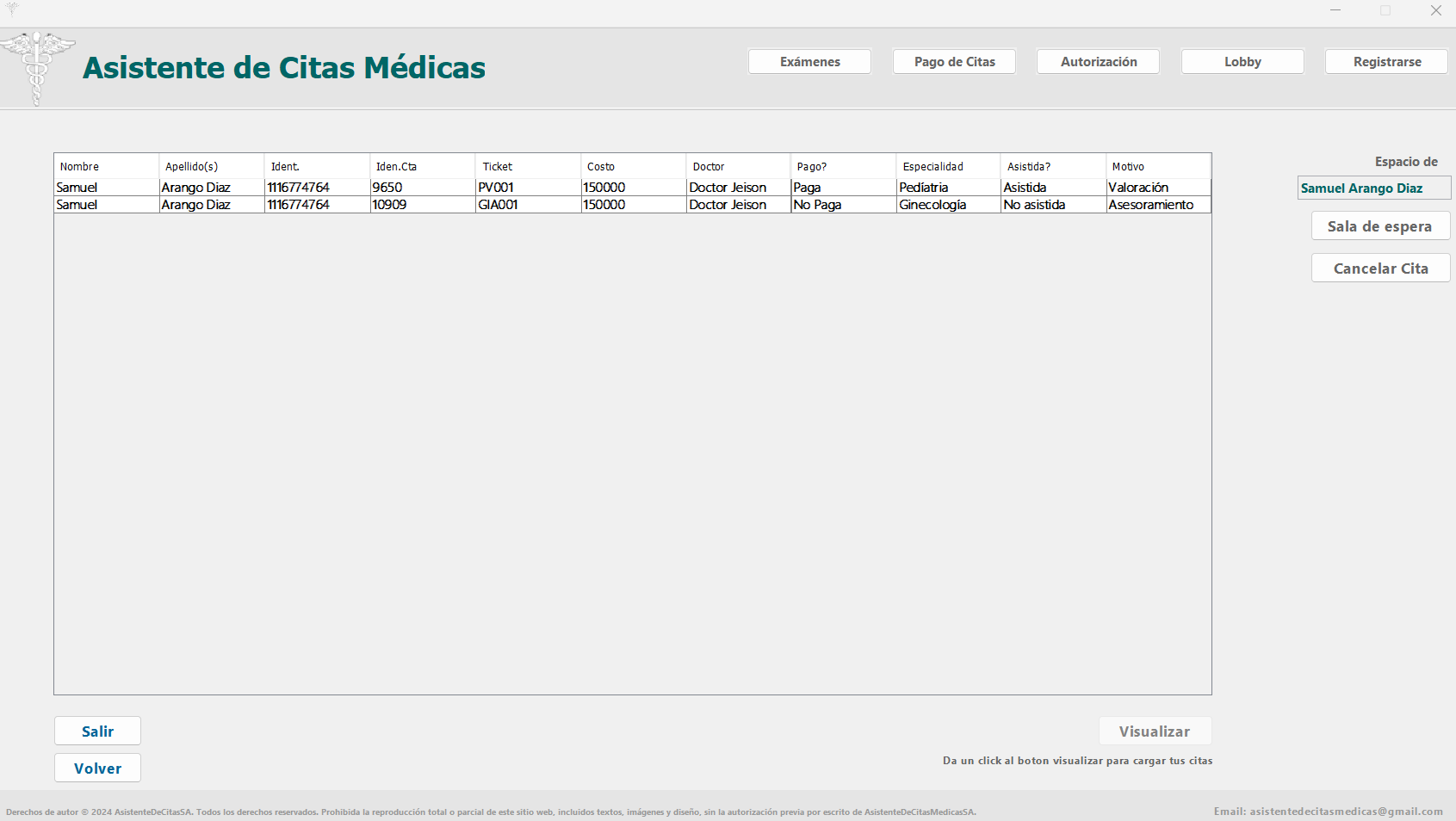
****

1. **Visualización de citas**

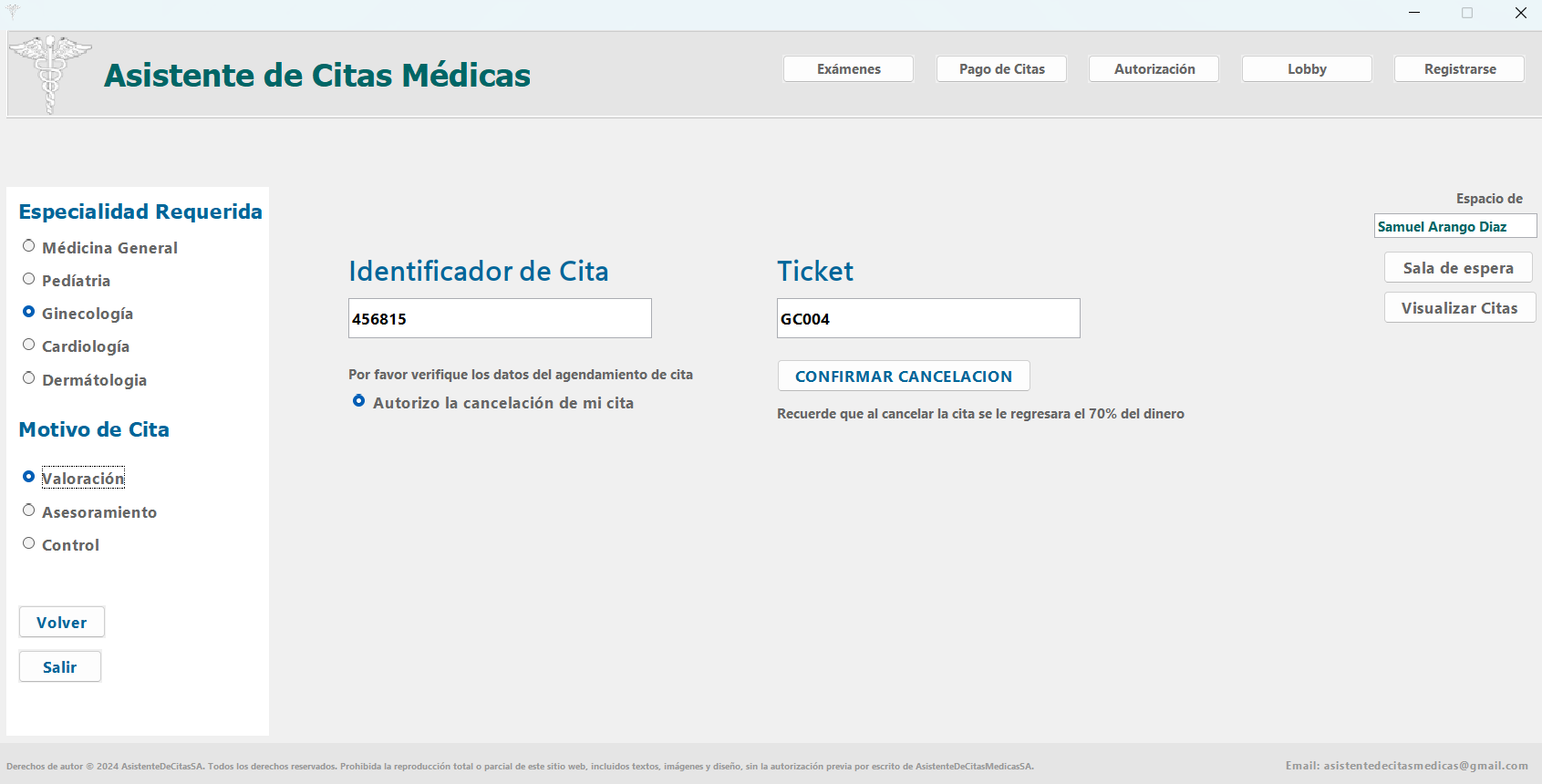
Esta página muestra una lista completa de todas las citas programadas a nombre del usuario actual. Proporciona una visión general de las citas próximas y pasadas, permitiendo al usuario acceder fácilmente a la información detallada de cada Cita.

Características clave:

1. Lista de citas, muestra todas las citas programadas en formato sencillo de entender con detalles exactos.
2. Filtros de búsqueda ofrece solamente las citas con base al usuario que solicito visualizar.
3. Detalles y continúan las opciones de gestión.

****

1. **Cancelación de citas**

****Aquí se centra en proporcionar al usuario la capacidad de cancelar citas médicas de manera eficiente y sin complicaciones. Aquí hay puntos a resaltar como que se le piden 4 datos respecto a la cita a cancelar que son el identificador, especialidad, motivo y ticket el proceso de cancelación simplificado asegura de que el proceso de cancelación sea lo más sencillo posible para el usuario. Implicando una interfaz intuitiva fácil de navegar que permita al usuario cancelar una cita con unos pocos clics o toques. Cabe resaltar que usted no puede cancelar citas que donde ya ha asistido ya que no tendría lógica eliminar ese historial. Pero si pudiera cancelar citas pagas, no pagas, no asistidas. El manejo de excepciones del ticket ya que por ejemplo uno de los requerimientos era tener en cuenta cuantas citas ya se han confirmado para crear el ticket. Pues continuaría su flujo normal del último ticket.

1. **Sala de espera para atención de citas**

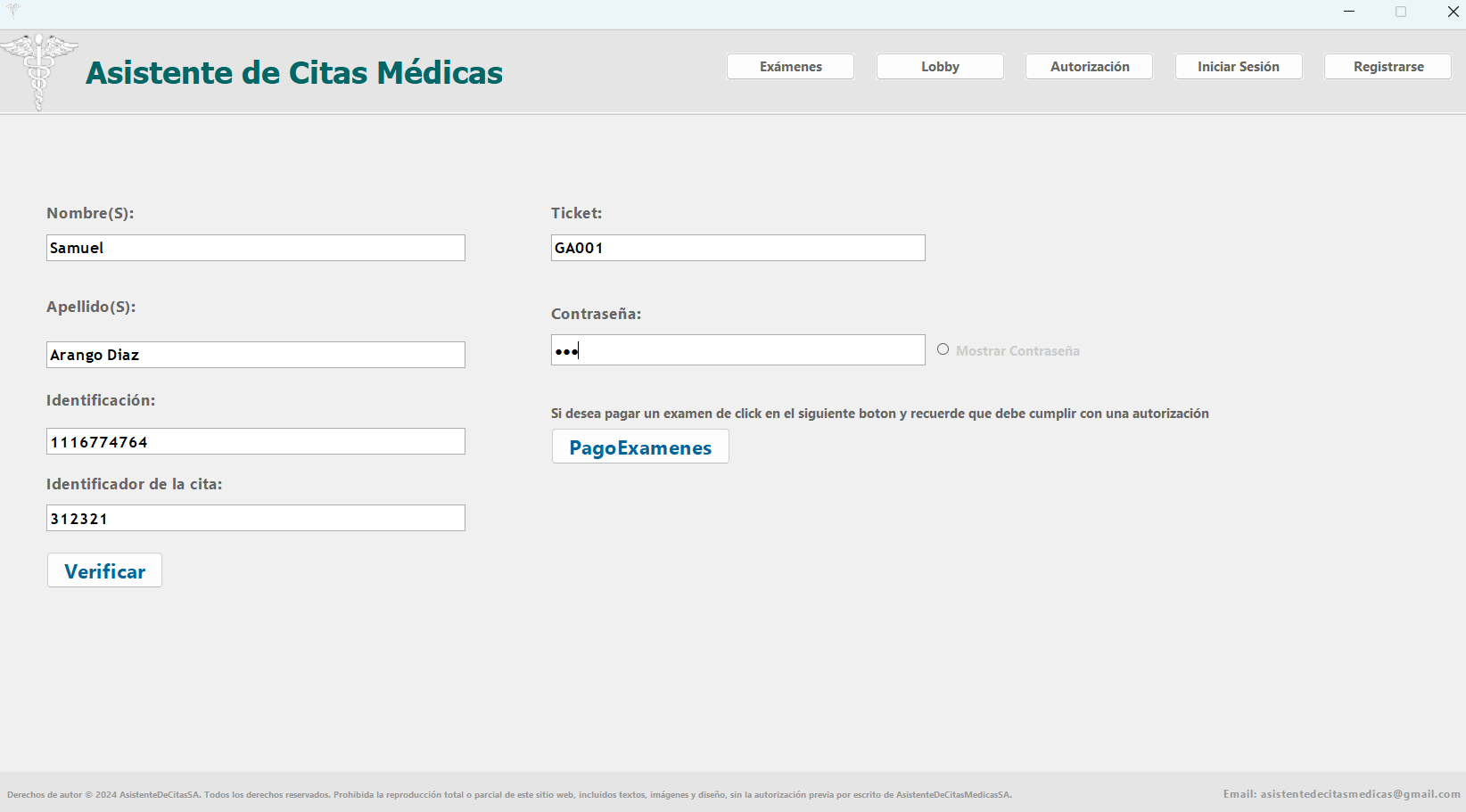
**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**La sala de espera es un componente crucial en el proceso de atención de citas médicas, ya que proporciona un espacio virtual donde los pacientes pueden esperar su turno para ser atendidos. La interfaz es intuitiva para hacer fácil su hizo y accesible para los pacientes. Esto implica integrar un enlace directo en la plataforma de citas médicas. Cabe resaltar que también tiene la debida intervención de un operador para detener o dar paso a las personas con el flujo debido de las personas.

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

1. **Pago de citas**

****Para el proceso de pago de citas, fue esencial diseñar una interfaz intuitiva y con diferentes filtros, así como en la demás sección donde si la contraseña es incorrecta o si no se encuentra la cita buscada para pagar o si el usuario no existe. Asegurando a los usuarios una interacción de forma rápida , intuitiva y practica y sin complicaciones. Los parámetros solicitados para revisar si esa cita ya se encuentra paga se digitan los datos solicitados por la interfaz y sus debidos parámetros al verificar se despliegan dos posibles rutas 1 de que la cita ya es paga y usted puede ser atendido y la 2 es que esa cita no está paga y usted debe pagarla en una cola de caja que también es monitoreada por el operador.

****

1. **Autorización/Registro de exámenes**

Para el proceso de autorización o registro de exámenes médicos, es importante diseñar la interfaz que permite a los usuarios la solicitud y autorización de la realización de pruebas médicas de forma eficiente cabe resaltar que si es un examen dejado luego de una cita médica el examen se selecciona de forma automática y si es un examen solicitado por el usuario de forma independiente deberá tener mínimamente un control con nosotros para crear una historia clínica e ir de la mano con el marco legal. Bajo la ley de **Ley de 1551 de 2015** que resalta que ningún establecimiento que presta sus servicios a la salud realizara exámenes médicos a personas sin antes saber su correcto estado de salud para su debido examen. Cabe decir que la autorización tiene su sala de espera que va ligado al sistema LIFO monitoreado por el operador para tener y continuar con el flujo de la sala para los usuarios y su correcta autorización de examen. En la parte de autorizaciones también se puede visualizar todas las autorizaciones a nombre de cierto usuario y cancelarlas si es el caso deseado por el usuario.



****

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

1. **Pago de exámenes/A partir de una autorización exitosa**

Sección de exámenes aquí se encuentran 3 diferentes opciones para proceder la primera es proceder con el pago que se pagara el examen a partir de autorización de exámenes es exitosa es decir el registro de los exámenes es exitosa se procede a pagar para en consecuencia pasar a ser atendidos y asegura para los usuarios una experiencia fluida y coherente. También cuenta con una visualización de exámenes a nombre del usuario también hay distintos casos si el usuario digita los datos de un examen ya pago saldrá la notificación de que ese examen ya este pago y que puede proceder a la sala de atención gestionada por el operador. También se cuenta con una cancelación de exámenes ya que examen es como una cita la compañía que preste el servicio entonces se puede cancelar un examen ya pago y no asistido. Pero si ya está asistido no podrá cancelarse y para cancelar un examen no pago que es su debida autorización solamente se deberá cancelar dicha autorización.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

****

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

1. **Atención de exámenes/A partir de un Pago exitoso**

Una vez que el pago por los exámenes médicos ha sido exitoso, es crucial asegurarse de que el proceso de atención de los exámenes sea eficiente y satisfactorio para los usuarios. Aquí hay algunas consideraciones importantes para esta parte:

Confirmación de pago: Después de que el pago sea exitoso, proporciona al usuario una confirmación clara de que la transacción se ha completado con éxito. Esto incluye un mensaje en pantalla.

Programación de exámenes: Una vez que se haya confirmado el pago, permite al usuario programar los exámenes médicos conveniente para ellos. Esto puede hacerse a través de la sección de autorizaciones.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

El prototipo del diseño inicial lo podrá encontrar adjuntado en el siguiente link <https://www.figma.com/file/z6BQy7Ulay65LyDyWPGAgX/Untitled?type=design&node-id=0%3A1&mode=design&t=V8EQUcQML9dum6BY-1>

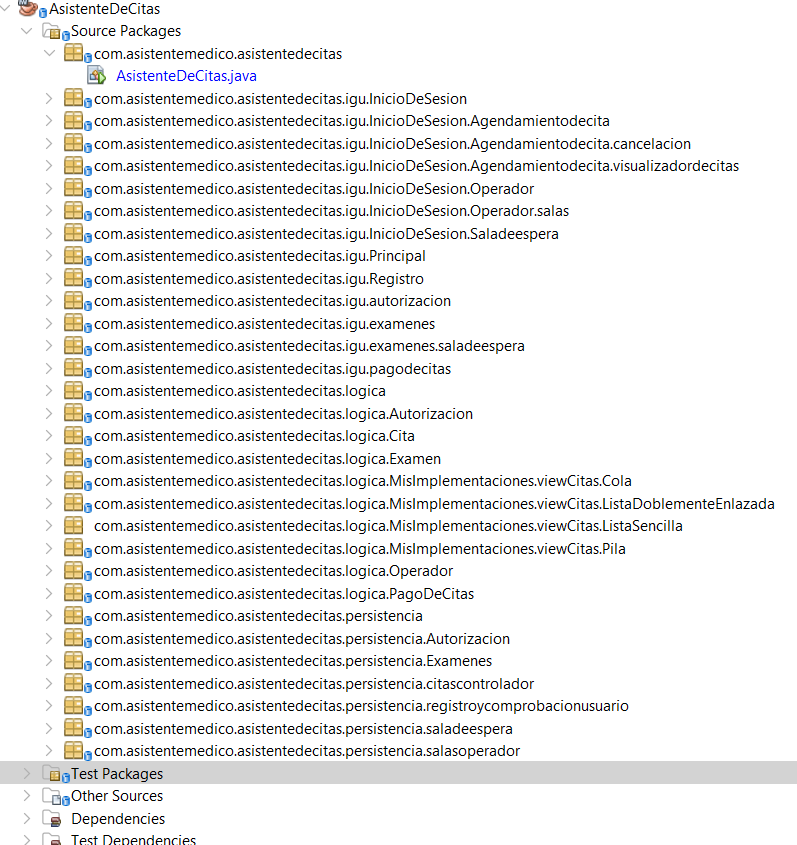
que contiene todas las vistas del aplicativo.

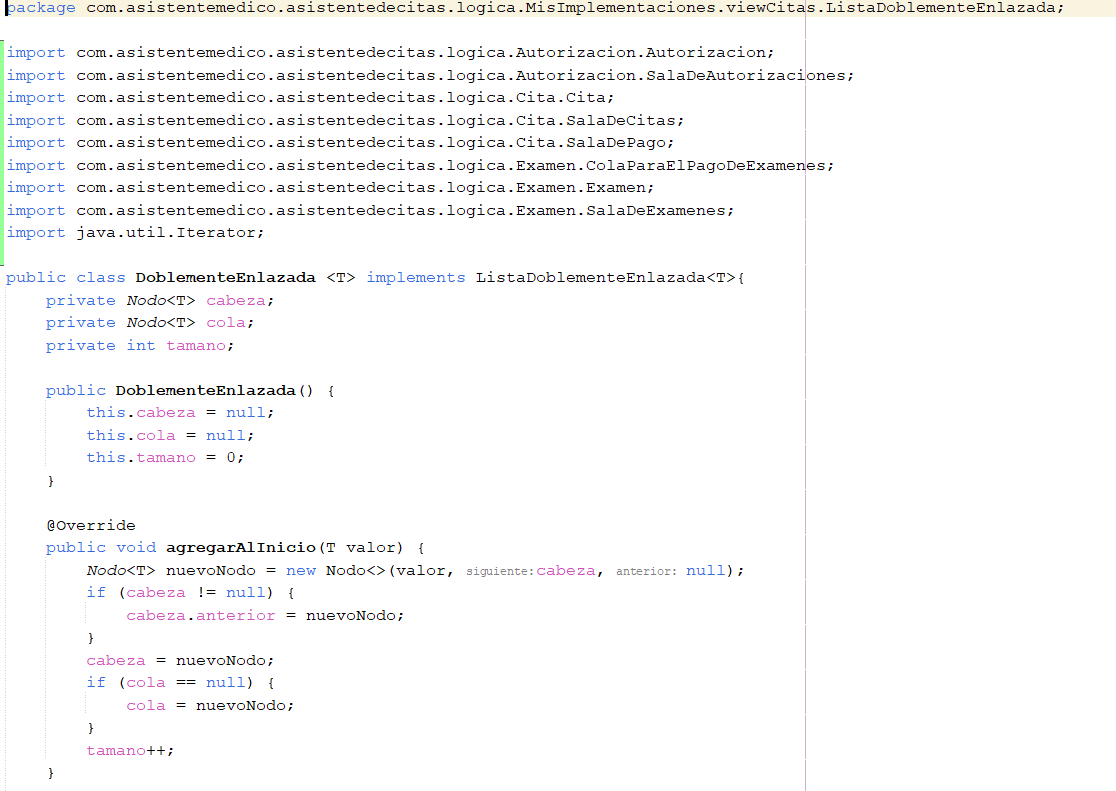
**Visualización de código**

En el desarrollo de las interfaces de usuario para el sistema de gestión de citas médicas, se ha prestado especial atención a la estructura y organización del código subyacente que define estas vistas. Aquí hay una de las consideraciones más importantes que tuvimos en cuenta al momento de desempeñar la elaboración.

1. **Modularidad y reutilización,** el código que define las vistas se ha estructurado de distintas formas de forma modular, lo que permite la reutilización de componentes en diferentes partes del sistema. Esto facilita la consistencia visual y el funcional en todas las vistas y promueve la eficiencia en el desarrollo.
2. **Separación de responsabilidades,** Se ha aplicado el principio de separación de responsabilidades para garantizar que el código que define las vistas este claramente separado de la lógica de la programación orientada a eventos que son las acciones con los frames y el manejo de datos. Mejorando la legibilidad del código y facilitando la colaboración entre desarrolladores.
3. **Accesibilidad,** Se han seguido las prácticas de accesibilidad garantizando que las vistas y el funcionamiento vayan de la mano. De forma práctica y funcional para todos los usuarios, incluyendo aquellos con discapacidades visuales o motoras.

1. **Patrón MVC,** el sistema de asistente de citas médicas se divide en tres componentes principales uno de ellos es el modelo que representa la estructura de datos y la lógica de negocio del sistema. En la división por paquetes y clases, las clases que representan el modelo suelen encontrarse en paquetes que describen su funcionamiento con la etiqueta de (Gestor) otra de las responsabilidades son las vistas de mostrar la interfaz al usuario. En la división por paquetes y clases, que representan las vistas suelen encontrarse en el paquete vistas. Estas clases definen la presentación de los datos y manejan la interacción con el usuario estos paquetes llevan la etiqueta de (IGU) y finalmente y controlador que actúa como intermediario entre el modelo y la vista con la parametrización de datos respondiendo a las solicitudes del usuario y actualizando la vista o el modelo estos paquetes llevan como etiqueta (Lógica)

****

****

**Texto

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

**Texto, Aplicación

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

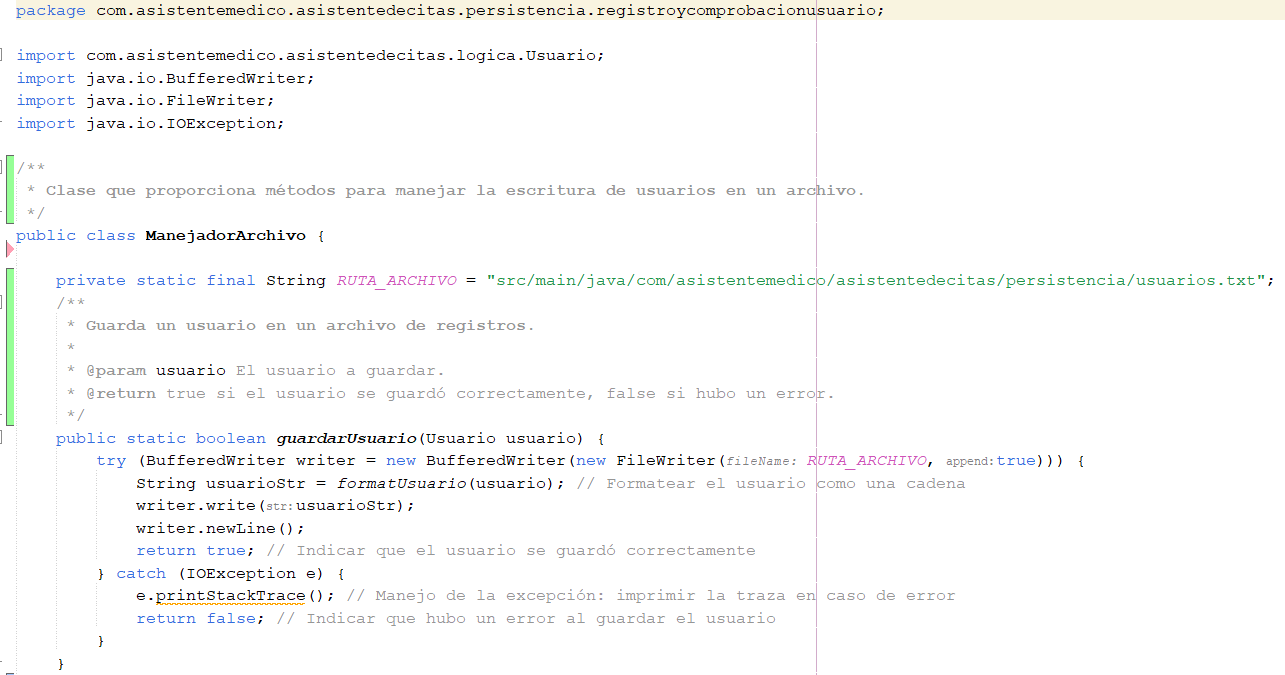
**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

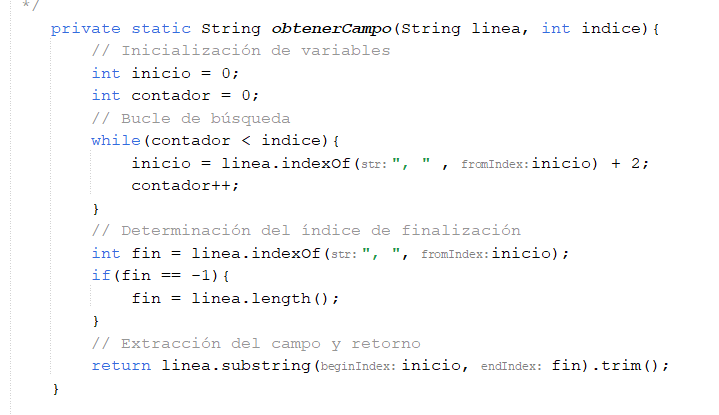
**Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente**

**Texto

Descripción generada automáticamente**

**Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Texto

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente**

**Especificación de Requerimientos**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | REQUERIMIENTO | DESCRIPCIÓN | PRIORIDAD |
| 110 | Registro de Usuarios por Número Telefónico | Como usuario, quiero poder registrar mi información utilizando mi número telefónico como identificador. En caso de no contar con un número telefónico, deseo tener la opción de utilizar un identificador de llamadas para acceder al sistema. | MEDIA |
| 220 | Interfaz Dinámica para el Inicio de Sesión | Como usuario, quiero una interfaz de inicio de sesión dinámica y sencilla, para que las distintas características del programa estén organizadas en páginas y se facilite la navegación. | MEDIA |
| 330 | Configuración de Tipo de Cita | Como usuario, quiero tener una casilla en la interfaz donde pueda configurar mi cita, para poder seleccionar el tipo de cita que deseo, ya sea general o de un área específica. | MEDIA |
| 440 | Selección de Tipo de Cita y Categoría | Como usuario, quiero poder seleccionar el tipo de cita que deseo y categoría, y tener la opción de decidir si pagaré la cita de una vez o después, para tener mayor flexibilidad y eficiencia en los tiempos de pago. | ALTA |
| 550 | Confirmación de Cita | Como usuario, deseo poder confirmar mi cita a través del sistema y que el sistema genere automáticamente el total a pagar, para agilizar el proceso de pago de las citas. | MEDIA |
| 660 | Modificación de Cita Designada | Como usuario, quiero poder modificar mi cita designada según el motivo y el tipo de cita que seleccione, para tener la comodidad de poder editar las especificaciones de la cita acorde al tipo de atención médica que deseo. | ALTA |
| 7 | Agendar Citas Médicas | Como usuario, deseo poder agendar citas médicas una vez iniciada la sesión en el sistema y tener la opción de seleccionar lo necesario para mi cita, así como la especialidad médica requerida. Además, deseo recibir una confirmación de la cita una vez que haya sido programada con éxito. | ALTA |
| 8 | Acceder a la Sala de Espera | Como usuario, deseo tener la capacidad de acceder a la sala de espera del centro de salud. Quiero poder verificar mi cita programada y, si es necesario, unirme a la sala de espera virtual para esperar mi turno de atención médica. Además, deseo recibir notificaciones o recordatorios cuando sea mi turno de ser atendido, para que haya una mejor organización en la sección de sala de espera. | ALTA |
| 9 | Autorización de Exámenes desde el Lobby | Como usuario, deseo tener la opción de acceder a la autorización de exámenes desde el lobby del centro de salud. Quiero poder revisar los exámenes recomendados por mi médico y autorizar su realización a través del sistema. Además, deseo recibir información sobre la preparación necesaria para cada examen y los detalles sobre cómo proceder después de su realización para estar mejor orientado y notificado acerca de las recomendaciones y cómo proceder con esta. | ALTA |
| 10 | Pago de citas | Como usuario, deseo tener la opción de realizar el pago de mi cita médica una vez establecida las especificaciones sobre el tipo de cita que quiero agendar, para cumplir de una forma más optima el respectivo pago antes de proceder con la cita médica o examen programado | ALTA |
| 11 | Atención de cita médica o examen | Como usuario, quiero tener la opción de elegir que doctor deseo que atienda mi cita médica o examen siempre y cuando esté libre y haga parte de los médicos autorizados a atender ese tipo de cita o examen, pero en caso de agendar un examen hacer tener antes la respectiva cita de control necesaria para la autorización del agendamiento de este, para tener una mayor flexibilidad y comodidad a la hora de ser atendido. | MEDIA |
| 12 | Cancelación de citas y exámenes | Como usuario, quiero tener la capacidad de cancelar mi cita médica en caso de que ocurra alguna situación por la que no pueda ser atendido, y que me devuelvan el 70% de lo pagado, para poder recibir una gran parte de lo pagado y tener la ventaja de usarlo para reagendar otra cita o examen. | ALTA |

**Implementaciones**

En el proyecto Aurora, las implementaciones representan el puente crucial entre la concepción de ideas y la materialización de soluciones tangibles. Cada implementación no solo implica la aplicación de tecnologías y herramientas específicas, sino también una cuidadosa planificación y ejecución para garantizar que las funcionalidades diseñadas cumplan con los requisitos del usuario y las expectativas del proyecto.

Las implementaciones en este proyecto se abordaron con un enfoque centrado en la eficiencia y la calidad, priorizando la entrega oportuna de funcionalidades clave sin comprometer la estabilidad del proyecto ni la escalabilidad del sistema.

Se establecieron procesos de desarrollo ágiles que fomentaron la colaboración entre los equipos, la iteración continua y la adaptación rápida a los cambios en los requisitos del proyecto.

Continuando con la importancia de las correctas implementaciones y el correcto abordamiento en este proyecto, las implementaciones son el motor que impulsa el progreso y la evolución del sistema de gestión de citas médicas. Cada una de las importantes características, mejora y corrige los errores llevados a cabo mediante las implementaciones efectivas contribuye a fortalecer la robustez y la utilidad del sistema.

**Tecnologías**

En el proyecto Aurora, se emplearon diversas tecnologías a destacar para dar vida al sistema de gestión de citas médicas, cada una desempeñando un papel crucial en la construcción del proyecto y operación del sistema.

Estas tecnologías abarcaron de forma eficiente desde el lenguaje de programación Java hasta plataformas de desarrollo y herramientas de persistencia de la información.

1. **Entorno de Desarrollo y Lenguaje de Programación:** Se utilizó Java como el lenguaje de programación principal debido a su robustez, portabilidad y amplia adopción en el desarrollo de aplicaciones empresariales. Además, se empleó NetBeans como entorno de desarrollo integrado (IDE), aprovechando su conjunto de herramientas avanzadas para agilizar el proceso de desarrollo y mejorar la productividad del equipo.
2. **Interfaz de Usuario (UI),** Se creó un frontend para la aplicación de escritorio utilizando Java Swing, una biblioteca gráfica que ofrece una amplia gama de componentes visuales y una fácil integración con el entorno de desarrollo NetBeans. Esto permitió diseñar interfaces de usuario intuitivas y atractivas para mejorar la experiencia del usuario.
3. **División de clases por Responsabilidad Única (SRP),** Se aplicó el principio de responsabilidad única para dividir las clases en unidades cohesivas y de responsabilidad única. Esto facilitó el mantenimiento del código, mejoró su legibilidad y promovió la reutilización de componentes, lo que contribuyó a la escalabilidad y mantenibilidad del sistema.
4. **Estructuras de Datos,** Se implementaron diversas estructuras de datos para el correcto funcionamiento y la dependencia de otras estructuras como ArrayList, HashMap, List, Arreglos, entre otros. Se implemento como doblemente Enlazada, pilas, colas y listas sencillas. Para organizar y manipular eficientemente la información relacionada con citas médicas, pacientes y médicos, exámenes, autorizaciones. Entre otros.
5. **Clases para manejo de Parámetros,** Se desarrollaron clases específicas para gestionar los parámetros de la información, lo que facilitó la manipulación y el acceso a los datos de manera estructurada y coherente. Esto contribuyó a la claridad y coherencia del código, así como a la escalabilidad del sistema.
6. **Persistencia de la información con archivos de Texto,** La persistencia de la información se llevó a cabo utilizando archivos de texto, que ofrecen una solución simple y efectiva para almacenar y recuperar datos sin la necesidad de un sistema de gestión de base de datos complejo. Escogimos algo practico y un sistema de citas portable en cualquier sistema sin descargar otras características externas.
7. **Junit Test,** Se utilizaron pruebas unitarias con JUnit para garantizar la calidad y el correcto funcionamiento del código. Estas pruebas permitieron la detección temprana de errores y facilitaron la refactorización del código, asegurando la estabilidad y fiabilidad del sistema en todas las etapas del desarrollo.
8. **Control de versiones,** Se utilizó Git como sistema de control de versiones para gestionar el código fuente del proyecto. Git proporcionó un historial completo de cambios, permitiendo al equipo de desarrollo colaborar de manera efectiva y mantener un registro detallado de todas las modificaciones realizadas en el código.

GitHub se empleó como plataforma de alojamiento remoto para el repositorio Git del proyecto. Esta elección permitió al equipo de desarrollo acceder al código desde cualquier lugar, facilitando la colaboración entre miembros del equipo ubicados en diferentes ubicaciones geográficas.

**Software**

El software implemento diferentes herramientas de software y estructuras utilizadas durante el desarrollo del proyecto.

El propósito del software. Principalmente es automatizar y optimizar el proceso de gestión de citas médicas en un entorno hospitalario o clínico. Facilitando la coordinación entre pacientes y médicos, garantizando una distribución eficiente de los recursos y una atención medica oportuna.

**Funcionalidades Principales**

* Gestión de citas médicas, Registro de pacientes, Asignación de médicos, gestión de exámenes, pago de citas, autorizaciones médicas. Entre otros.

**Arquitectura**

* El software sigue una arquitectura basada en capas, con una capa de presentación (UI) desarrollada en Java Swing para la interfaz de usuario de la aplicación de escritorio.
* La lógica está encapsulada en clases Java que implementan principios de diseño como la Responsabilidad Única (SRP) y la cohesión.
* Se utilizan estructuras de datos como listas doblemente enlazadas y clases específicas para el manejo de parámetros y estados de flujo.

**Capa de presentación (UI)**

* La interfaz de usuario se desarrollo con distintos componentes como Lbales, Tables, Botton, etc. El diseño de la interfaz se hizo de forma practica y funcional para que cualquier tipo de usuario pudiera operarla. También el manejo de eventos donde se desempeñaba un papel crucial si el usuario digita una contraseña errónea no simplemente quedarse en esa pestaña por el contrario tener un evento pasando a una sección de contraseñas incorrectas y su debida vinculación a acciones correspondientes. Finalmente, la accesibilidad y usabilidad según su acción o botón describe lo que se realiza en esa parte del programa.
* Se han implementado controladores de eventos que responden a las acciones del usuario, como clics de botón o selección de elementos, para ejecutar las acciones correspondientes, como programar citas o realizar pagos. Se ha priorizado la respuesta rápida y la retroalimentación visual para mejorar la interactividad.

**Capa de lógica**

* La capa de lógica del software incluye clases como FlujoDeSalas, que se encargan de gestionar el flujo de operaciones en las diferentes salas del sistema, como la sala de atención de citas, la sala de pagos y la sala de autorizaciones. Estas clases implementan métodos para cargar, modificar y guardar el estado de las salas, así como para realizar acciones específicas en función de los eventos del sistema.
* Las clases de la capa de lógica siguen los principios de diseño como la Responsabilidad Única (SRP) y la cohesión, donde cada clase tiene una única responsabilidad y está altamente relacionada con su función específica en el sistema. Se han evitado acoplamientos innecesarios entre clases para mejorar la mantenibilidad y la reutilización del código.
* La lógica del software se ha diseñado con una clara separación de preocupaciones y una estructura modular, donde cada componente se encarga de una tarea específica y puede ser fácilmente reemplazable o extendido sin afectar otras partes del sistema. Esto permite una mayor flexibilidad y escalabilidad en el desarrollo y mantenimiento del software.

**Estructuras de datos y clases especificas**

* Listas doblemente enlazadas jugo un papel crucial en este proyecto al igual que las pilas, colas y listas sencillas. En la implementación del software, se utiliza una lista doblemente enlazada para mantener el flujo de estados en las diferentes salas del sistema. Esta estructura de datos proporciona un acceso eficiente tanto hacia adelante como hacia atrás en la lista, lo que es crucial para manejar el flujo de operaciones en las salas, como cambiar el estado de una cita médica o un proceso de autorización.
* El software incluye clases específicas como EstadoParaSalaDeCitasAtencionMedica, Entre otras. Para representar y manejar los diferentes estados de flujo en las salas del sistema. Todo esto siendo posible gracias a las listas dobles donde estas clases encapsulan la información relevante, como nombres, identificación, estado y proporciona métodos para la debida manipulación de la información coherente y segura.

**Patrones de diseño**

* Se tuvieron en cuenta los patrones de diseño como el patrón de diseño Modelo-Vista-Controlador (MVC). Ya que este patrón separa la lógica (modelo) de la presentación(Vista) y la lógica de control (Controlador), lo que facilita el modularidad y la reutilización del código.
* La aplicación de patrones de diseño para este proyecto funciono para estructurar el código ayudando a organizar y estructuras el código de manera efectiva, separando las preocupaciones y facilitando la comprensión y mantenimiento del software. Por ejemplo, utilizar el patrón MVC, se puede dividir el código en componentes independientes como se desarrollo que son más fáciles de mantener y extender.

**Escalabilidad y flexibilidad / Desempeño y optimización**

* La arquitectura del software está diseñada para facilitar la escalabilidad, permitiendo la incorporación de nuevas funcionalidades y la expansión del sistema para adaptarse a cambios en los requisitos del negocio o del usuario. Esto se logra mediante una estructura modular y una clara separación de preocupaciones que permite agregar nuevas capas o componentes sin afectar el funcionamiento existente del sistema.
* Se han tenido en cuenta las consideraciones de flexibilidad en el diseño del software, asegurando que el sistema pueda adaptarse fácilmente a cambios en los requisitos o tecnologías.
* En la arquitectura del software, se han implementado estrategias de optimización para mejorar el rendimiento del sistema. Esto puede incluir técnicas como la optimización de consultas, la minimización de la carga en la interfaz de usuario y la optimización de algoritmos para reducir el tiempo de procesamiento.

**Pruebas**

Se tuvieron en cuenta distintos estados de prueba como pruebas de integración, escenarios de uso, Junit test para los métodos, Resultados de prueba y validaciones.

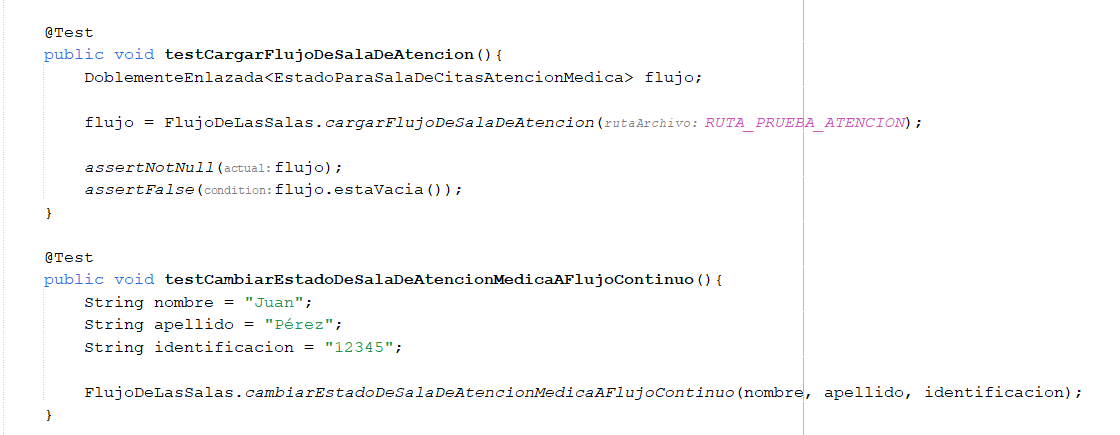
* **Pruebas de unidad**

Se utilizaron validaciones de comportamiento individual de componentes o unidades de código en el módulo de gestión de usuarios. Una prueba de estas fue la verificación que no permitiera grabar dos usuarios con la misma identificación que se centraba en probar cada clase que diera al cumplimiento de este requerimiento y método de forma aislada para asegurar su correcto funcionamiento.

* **FrameWork de pruebas utilizado**

Se empleo Junit para escribir y ejecutar las pruebas de unidad en el modelo de gestión de usuarios.

Algunos ejemplos de estas pruebas fueron, cargar el flujo de sala de atención, cambiar el estado de sala de atención, detener el flujo de atención de citas.





Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* **Resultados esperados y obtenidos**

Se esperaba que todas las pruebas de unidad pasaran correctamente, validando así el comportamiento de las clases y métodos de distintos eventos de gestión de usuarios. Los resultados obtenidos mostraron que todas las pruebas pasaron sin errores, confirmando que el módulo funcionaba según lo esperado.

* **Pruebas de integración**

En este punto una vez se probaron los métodos por medio de Junit test para validar su funcionamiento interno se probó la interacción entre el módulo de gestión de usuarios y otros módulos del sistema. Testeando el programa y usándolo como sería un día a día normal. Donde se centro en probar la comunicación entre el modulo de usuarios y los módulos relacionados, como el de autenticación y el de permisos.

* **Escenario de prueba de integración**

Esta parte de prueba de integración fue la creación de usuarios y su autenticación en el sistema por medio de contraseña, verificando que los datos se sincronicen correctamente entre los diferentes módulos.

* **Resultados de pruebas de integración**

Los resultados finales y obtenidos de las pruebas de integración confirmaron que el módulo de gestión de usuarios funcionaba correctamente en conjunto con otros módulos del sistema. Se verifico correcta comunicación y sincronización de datos entre los diferentes componentes.

* **Pruebas de aceptación**

1. Las pruebas de aceptación se utilizaron para validar que el módulo de gestión de usuarios cumpliera con los requisitos y expectativas del cliente o usuario final. Se centraron en probar los casos de uso principales del sistema relacionados con la gestión de usuarios.
2. Los escenarios de pruebas de aceptación se definieron varios escenarios de prueba de aceptación que representaban las acciones principales que realizarían los usuarios en el sistema, como registrar un nuevo usuario, iniciar sesión y gestionar las citas y su correspondiente visualización de vida clínica.
3. Los resultados de las pruebas de aceptación confirmaron que el módulo de gestión de usuarios cumplía con los criterios de aceptación definidos por el cliente. Se verifico el funcionamiento del módulo y opero correctamente según lo esperado en distintos escenarios de uso y que satisficiera las necesidades y expectativas de los usuarios finales.

**Cronograma**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **BLOQUE 1: PRIMERA ENTREGA** | | **MES** | | | | | | | | | | | | |
| **SEMANA** | **TAREA** | **FEBRERO** | | | | **MARZO** | | | | | **ABRIL** | | | |
| SEM 1 | SEM 2 | SEM 3 | SEM 4 | SEM 1 | SEM 2 | SEM 3 | SEM 4 | SEM 1 | | SEM 2 | SEM 3 | SEM 4 |
| ***Febrero 5-10*** | Definición de equipos y roles |  |  |  |  |  |  |  |  |  | |  |  |  |
| Investigación preliminar |  |  |  |  |  |  |  |  |  | |  |  |  |
| Revisión de formatos de entrega (IEEE o APA) |  |  |  |  |  |  |  |  |  | |  |  |  |
| Elaboración de portadas y formato del documento |  |  |  |  |  |  |  |  |  | |  |  |  |
| ***Febrero 8-15*** | Introducción, Planteamiento del Problema y Justificación |  |  |  |  |  |  |  |  |  | |  |  |  |
| ***Febrero 12-22*** | Definición de Objetivos del Proyecto |  |  |  |  |  |  |  |  |  | |  |  |  |
| Marco Referencial (Conceptual, Tecnológico y/o Legal) |  |  |  |  |  |  |  |  |  | |  |  |  |
| Metodología de Trabajo |  |  |  |  |  |  |  |  |  | |  |  |  |
| ***Febrero 20-28*** | Resultados Preliminares (Especificación de Requerimientos y Diseños) |  |  |  |  |  |  |  |  |  | |  |  |  |
| Elaboración del Cronograma de Actividades |  |  |  |  |  |  |  |  |  | |  |  |  |
| ***Marzo 4-8*** | Revisión, ajustes y entrega |  |  |  |  |  |  |  |  |  | |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **BLOQUE 2: DESARROLLO Y PREPARACIÓN DE LA SEGUNDA ENTEGA** | | **MES** | | | | | | | | | | | |
| **SEMANA** | **TAREA** | **FEBRERO** | | | | **MARZO** | | | | **ABRIL** | | | |
| SEM 1 | SEM 2 | SEM 3 | SEM 4 | SEM 1 | SEM 2 | SEM 3 | SEM 4 | SEM 1 | SEM 2 | SEM 3 | SEM 4 |
| ***Marzo 9-15*** | Planificación detallada del Desarrollo (Fase de Análisis) |  |  |  |  |  |  |  |  |  |  |  |  |
| ***Marzo 16-22*** | Diseño detallado del software |  |  |  |  |  |  |  |  |  |  |  |  |
| ***Marzo 23- 5 abril*** | Desarrollo del código fuente |  |  |  |  |  |  |  |  |  |  |  |  |
| ***Abril 6-12*** | Pruebas Unitarias |  |  |  |  |  |  |  |  |  |  |  |  |
| ***Abril 13-16*** | Preparación de documentación técnica |  |  |  |  |  |  |  |  |  |  |  |  |
| ***Abril 17-18*** | Validación del software |  |  |  |  |  |  |  |  |  |  |  |  |
| ***5 Mayo*** | Segunda Entrega |  |  |  |  |  |  |  |  |  |  |  |  |

**Referencias Bibliográficas**

**(**Home**)** About Version Control-About Version Control. Software Freedom conservancy. <https://git-scm.com/book/es/v2/Inicio---Sobre-el-Control-de-Versiones-Acerca-del-Control-de-Versiones>

Wikipedia contributors. (s/f). Cola de prioridades. Wikipedia, The Free Encyclopedia. <https://es.wikipedia.org/w/index.php?title=Cola_de_prioridades&oldid=145477000>

Wikipedia contributors. (s/f). Modelo–vista–controlador. Wikipedia, The Free Encyclopedia. <https://es.wikipedia.org/w/index.php?title=Modelo%E2%80%93vista%E2%80%93controlador&oldid=138615253>

Wikipedia contributors. (s/f). Control de Versiones. Wikipedia, The Free Encyclopedia. <https://es.wikipedia.org/w/index.php?title=Control_de_versiones&oldid=152003367>

¿Qué es Git? (s/f). Microsoft.com. Recuperado el 3 de septiembre de 2023, de <https://learn.microsoft.com/es-es/devops/develop/git/what-is-git>

Tablado, F. (2020, septiembre 7). Bases de datos XML. Características y tipos. Ayuda Ley Protección Datos; AyudaLeyProteccionDatos. <https://ayudaleyprotecciondatos.es/bases-de-datos/xml/>

Wikipedia contributors. (s/f). Java Remote Method Invocation. Wikipedia, The Free Encyclopedia. <https://es.wikipedia.org/w/index.php?title=Java_Remote_Method_Invocation&oldid=148805182>

Base de Datos TXT : Parte I - Perl En Español. (s/f). Perlenespanol.com. Recuperado el 3 de septiembre de 2023, de <https://perlenespanol.com/tutoriales/bases_de_datos/base_de_datos_txt_parte_i.html>

Software development kits and command line interface. (s/f). Oracle.com. Recuperado el 9 de mayo de 2023, de <https://docs.oracle.com/en-us/iaas/Content/API/Concepts/sdks.htm>

UML Class Diagram tutorial. (s/f). Visual-paradigm.com. Recuperado el 3 de septiembre de 2023, de <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial/>

NetBeans, A. (s/f). Designing a Swing GUI in NetBeans IDE. Apache.org. Recuperado el 3 de septiembre de 2023, de <https://netbeans.apache.org/kb/docs/java/quickstart-gui.html>

What is GitHub? A beginner’s introduction to GitHub. (2018, abril 20). Kinsta®; Kinsta. <https://kinsta.com/knowledgebase/what-is-github/>

Deloitte (Sin fecha). *Qué es el desarrollo en espiral*. Deloitte.com. Recuperado el 17 de febrero de 2024, de<https://www2.deloitte.com/es/es/pages/technology/articles/que-es-el-desarrollo-en-espiral.html>

Ar-racking.com (2019, julio 26). *Método LIFO de Gestión Stock: Qué es y cuando se utiliza*.<https://www.ar-racking.com/co/blog/metodo-lifo-de-gestion-stock-que-es-y-cuando-se-utiliza/>

Arnold, K., Gosling, J., & Holmes, D. (Sin fecha). *THE javaTM programming language, fourth edition*. Ase.ro. Recuperado el 29 de febrero de 2024, de<https://www.acs.ase.ro/Media/Default/documents/java/ClaudiuVinte/books/ArnoldGoslingHolmes06.pdf>

SYDLE. (s. f.). Post title | SYDLE. <https://www.sydle.com/blog/billing-rules-6400e02757aff34f9e0a3155>

Juviler, J. (2024, enero 5). *What is GitHub? (and what is it used for?)*. HubSpot.<https://blog.hubspot.com/website/what-is-github-used-for>

Programiz.com (Sin fecha). *Circular linked list*. Recuperado el 7 de marzo de 2024, de<https://www.programiz.com/dsa/circular-linked-list>

SYDLE. (s. f.). Post title | SYDLE. <https://www.sydle.com/blog/billing-rules-6400e02757aff34f9e0a3155>