

UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

Tema:

APLICACIÓN WEB USANDO PYTHON PARA EL REGISTRO DE LA FICHA MÉDICA Y CONTROL DE LA POSOLOGÍA PARA EL CONSULORIO MÉDICO "DUOMEDIC"

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la obtención del título de Ingeniero en Tecnologías de la Información.

ÁREA: Base de Datos y Sistemas informáticos.

LÍNEA DE INVESTIGACIÓN: Tecnología de la información y Sistemas de control.

AUTOR: José Javier Villacrés López

TUTOR: Ing. Hernán Fabricio Naranjo Ávalos Mg.

Ambato - Ecuador

agosto – 2024

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: APLICACIÓN WEB USANDO PYTHON PARA EL REGISTRO DE LA FICHA MÉDICA Y CONTROL DE LA POSOLOGÍA PARA EL CONSULORIO MÉDICO "DUOMEDIC", desarrollado bajo la modalidad Proyecto de Investigación por el señor José Javier Villacrés López, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, agosto 2024.

Firmado electrónicamente por:
HERNAN FABRICIO NARANJO AVALOS

Ing. Hernán Fabricio Naranjo Ávalos, Mg.

TUTOR

AUTORÍA

El presente trabajo de titulación con el tema: APLICACIÓN WEB USANDO PYTHON PARA EL REGISTRO DE LA FICHA MÉDICA Y CONTROL DE LA POSOLOGÍA PARA EL CONSULORIO MÉDICO “DUOMEDIC” es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, agosto 2024.

Firmado electrónicamente por:
JOSE JAVIER VILLACRES LOPEZ

José Javier Villacrés López

C.C. 1803881133

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, agosto 2024.



José Javier Villacrés López

C.C. 1803881133

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por el señor José Javier Villacrés López, estudiante de la Carrera de Tecnologías de la Información de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado APLICACIÓN WEB USANDO PYTHON PARA EL REGISTRO DE LA FICHA MÉDICA Y CONTROL DE LA POSOLOGÍA PARA EL CONSULORIO MÉDICO "DUOMEDIC", nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, agosto 2024.

Firmado electrónicamente por:
ELSA PILAR URRUTIA URRUTIA

Ing. Elsa Pilar Urrutia Urrutia, Mg.
PRESIDENTE DEL TRIBUNAL

Firmado electrónicamente por:
SANTIAGO DAVID JARA MOYA

Ing. Santiago David Jara Moya, Mg.
PROFESOR CALIFICADOR

Firmado electrónicamente por:
SANDRA LUCRECIA CARRILLO RIOS

Ing. Sandra Lucrecia Carrillo Ríos, Mg.
PROFESOR CALIFICADOR

DEDICATORIA

Dedico este trabajo a mi primer amor, mi amada madre Pilar López. Quien es la persona que siempre ha estado a mi lado, brindándome su amor y apoyo incondicional. En cada paso, en cada desafío, en cada momento de duda y tristeza, tu presencia ha sido mi guía y mi consuelo. Gracias, madre, por ser mi motivación y por enseñarme el verdadero significado del amor y la perseverancia. Este logro es un reflejo de todo lo que me has dado.

A mi querida hermana María José, quien ha sido mi aliada para seguir adelante. Aunque eres la menor, tu fortaleza y valentía me han inspirado a seguir adelante y a no rendirme nunca. Este triunfo es tanto tuyo como mío.

A mis abuelitos, Esther y José, que con su sabiduría y amor han dejado una huella imborrable en mi vida. Gracias por sus abrazos, sus enseñanzas y su cariño han sido fundamentales para alcanzar este objetivo.

A la familia Salazar Silva (tíos Nelson y Fanny, y primos Orlando, Mathi y Daniel), quienes nos abrieron las puertas de su hogar a mi madre, hermana y a mí, a los que considero como mi segunda familia, gracias por apoyarnos y ayudarnos en los momentos más difíciles. Son una parte fundamental en nuestras vidas; las risas, juegos, su compañía, cariño y esos momentos inolvidables han sido una fuente de alegría y motivación. Este logro también es de ustedes, por ser una parte fundamental de mi vida.

José Javier Villacrés López

AGRADECIMIENTO

Al culminar esta etapa de mi vida en la que he pasado alegrías y tristezas. agradezco a mi familia y amigos quienes me apoyaron en cada paso que Dios supo guiarme.

Mil gracias a mi madre y hermana por su apoyo y cariño durante todo este proceso de mi vida universitaria, por tenerme paciencia y entenderme en los momentos difíciles que he pasado, las amo mucho.

Un agradecimiento enorme al Ing. Hernán Naranjo, por su apoyo, por su paciencia, por haberme guiado y compartido sus conocimientos para el desarrollo del este proyecto.

Y como no agradecer a las personas que compartieron aula y laboratorio conmigo durante años, donde me demostraron el significado de la palabra amistad si existe, agradezco a mi grupo de amigos NSDT1P (Ronny, Mario, Jara, Bryan) gracias por esos momentos inolvidables.

Y a la Universidad Técnica de Ambato le agradezco por darme las lecciones de vida más valiosas en el ámbito académico y formarme como profesional.

José Javier Villacrés López

ÍNDICE GENERAL DE CONTENIDOS

PORTADA ..... **i**

APROBACIÓN DEL TUTOR ..... **ii**

AUTORÍA ..... **iii**

DERECHOS DE AUTOR ..... **iv**

APROBACIÓN DEL TRIBUNAL DE GRADO ..... **v**

DEDICATORIA ..... **vi**

AGRADECIMIENTO ..... **vii**

ÍNDICE GENERAL DE CONTENIDOS ..... **viii**

ÍNDICE DE TABLAS ..... **xi**

ÍNDICE DE FIGURAS ..... **xv**

ÍNDICE DE ANEXOS ..... **xx**

RESUMEN EJECUTIVO ..... **xxi**

ABSTRACT ..... **xxii**

CAPÍTULO I. MARCO TEÓRICO **1**

1.1 Tema de investigación 1

1.1.1 Planteamiento del problema 1

1.2 Antecedentes investigativos 2

1.3 Fundamentación teórica 4

1.3.1 Desarrollo de software 4

1.3.2 Metodología Ágil 4

1.3.3 Aplicaciones web 8

1.3.4 Frameworks de desarrollo 9

1.3.5 Frameworks de Python 9

1.3.6 Gestión de salud 15

1.3.7 Control y seguimiento 15

1.3.8 Ficha médica 16

1.3.9 Prescripción médica 16

1.3.10 Control de prescripción médica 18

1.4 Objetivos 20

1.4.1 Objetivo general 20

1.4.2 Objetivos específicos 20

CAPÍTULO II. METODOLOGÍA 21

2.1 Materiales 21

2.2 Métodos 21

2.2.1 Modalidad de la investigación 21

2.2.2 Población y muestra 22

2.2.3 Recolección de información 23

2.2.4 Procesamiento y análisis de datos 40

CAPÍTULO III. RESULTADOS Y DISCUSIÓN 41

3.1 Análisis y discusión de los resultados 41

3.1.1 Proceso y descripción del registro de ficha, consulta y prescripción médica. .. 41

3.1.2 Desarrollo de aplicaciones web 45

3.2 Desarrollo de la propuesta 50

3.2.1 Fase I: Planificación 51

3.2.2 Fase II: Diseño 63

3.2.3 Fase III: Codificación 79

3.2.4 Fase IV: Pruebas 102

3.2.5 Fase V: Implantación 105

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES 108

4.1 Conclusiones 108

4.2 Recomendaciones 109

REFERENCIAS BIBLIOGRÁFICAS 110

ANEXOS 113

ÍNDICE DE TABLAS

Tabla 1. Ventajas y desventajas de Django 11

Tabla 2. Ventajas y desventajas de Flask 13

Tabla 3. Ventajas y desventajas de Tornado 14

Tabla 4. Población de Estudio 22

Tabla 5. Muestra 23

Tabla 6. Confiabilidad alfa de Cronbach en la encuesta de pacientes 24

Tabla 7. Interpretación de la escala de confiabilidad Alfa de Cronbach. 24

Tabla 8. Resultado de la pregunta 1 25

Tabla 9. Resultado de la pregunta 2 26

Tabla 10. Resultado de la pregunta 3 27

Tabla 11. Resultado de la pregunta 4 28

Tabla 12. Resultado de la pregunta 5 29

Tabla 13. Resultado de la pregunta 6 30

Tabla 14. Resultado de la pregunta 7 31

Tabla 15. Resultado de la pregunta 8 32

Tabla 16. Resultado de la pregunta 9 33

Tabla 17. Resultado de la pregunta 10 34

Tabla 18. Resultado de la pregunta 11 35

Tabla 19. Resultado de la pregunta 12 36

Tabla 20. Resultados de la entrevista 37

Tabla 21. Descripción del proceso actual del registro de ficha, consulta y prescripción médica 42

Tabla 22. Descripción del proceso optimizado del registro de ficha, consulta y prescripción médica 44

Tabla 23. Comparativa de IDEs de desarrollo web con compatibilidad con Python. 45

Tabla 24. Comparativa de Frameworks basados en el lenguaje Python 46

Tabla 25. Comparativa de las versiones del framework Django 47

Tabla 26. Comparativa de gestores de Bases de Datos 48

Tabla 27. Comparativa de las metodologías ágiles para desarrollo web 49

Tabla 28. Requerimientos del sistema 51

Tabla 29. Roles del proyecto 52

Tabla 30. Historia de usuario definición de la estructura de la aplicación 53

Tabla 31. Historia de usuario módulo de autenticación 54

Tabla 32. Historia de usuario módulo de usuarios 54

Tabla 33. Historia de usuario módulo de pacientes 54

Tabla 34. Historia de usuario gestión de registro de ficha médica 55

Tabla 35. Historia de usuario registro de signos vitales 55

Tabla 36. Historia de usuario gestión de prescripción médica 55

Tabla 37. Historia de usuario envío de prescripción médica por correo electrónico. 56

Tabla 38. Historia de usuario generador de reportes 56

Tabla 39. Historia de usuario cierre de aplicación 56

Tabla 40. Estimación de historias de usuario 57

Tabla 41. Plan de entrega 57

Tabla 42. Historias de usuario de la iteración 1 58

Tabla 43. Actividades de la definición de la estructura de la aplicación 58

Tabla 44. Actividades para el módulo de autenticación 59

Tabla 45. Historias de usuario de la iteración 2 59

Tabla 46. Actividades para el módulo de usuarios 59

Tabla 47. Historias de usuario de la iteración 3 60

Tabla 48. Actividades para el módulo de registro de ficha médica 60

Tabla 49. Actividades para el registro de signos vitales 60

Tabla 50. Actividades para el módulo de prescripción médica 61

Tabla 51. Historias de usuario de la iteración 4 61

Tabla 52. Actividades para el envío de la prescripción médica 61

Tabla 53. Historias de usuario de la iteración 5 62

Tabla 54. Actividades para el módulo de reportes 62

Tabla 55. Actividades para el cierre de aplicación 62

Tabla 56. Tarjeta CRC módulo de autenticación 63

Tabla 57. Tarjeta CRC módulo de usuarios 63

Tabla 58. Tarjeta CRC módulo de pacientes 63

Tabla 59. Tarjeta CRC módulo de registro de ficha médica 64

Tabla 60. Tarjeta CRC registro de signos vitales 64

Tabla 61. Tarjeta CRC módulo de prescripción médica 64

Tabla 62. Tarjeta CRC envío de la prescripción médica por correo electrónico 64

Tabla 63. Tarjeta CRC módulo de reportes 65

Tabla 64. Prueba de aceptación 1: Módulo de autenticación..... 102

Tabla 65. Prueba de aceptación 2: Módulo de usuarios..... 103

Tabla 66. Prueba de aceptación 3: Módulo de pacientes 103

Tabla 67. Prueba de aceptación 4: Módulo de registro de ficha médica..... 103

Tabla 68. Prueba de aceptación 5: Registro de signos vitales..... 104

Tabla 69. Prueba de aceptación 6: Módulo de prescripción médica..... 104

Tabla 70. Prueba de aceptación 7: Envío de la prescripción médica por correo electrónica 104

Tabla 71. Prueba de aceptación 8: Modulo de reportes 105

ÍNDICE DE FIGURAS

Figura 1. Estructura de un proyecto Django 10

Figura 2. Estructura de un proyecto Flask (MVC) 12

Figura 3. Resultado de la pregunta 1 25

Figura 4. Resultado de la pregunta 2 26

Figura 5. Resultado de la pregunta 3 27

Figura 6. Resultado de la pregunta 4 28

Figura 7. Resultado de la pregunta 5 29

Figura 8. Resultado de la pregunta 6 30

Figura 9. Resultado de la pregunta 7 31

Figura 10. Resultado de la pregunta 8 32

Figura 11. Resultado de la pregunta 9 33

Figura 12. Resultado de la pregunta 10 34

Figura 13. Resultado de la pregunta 11 35

Figura 14. Resultado de la pregunta 12 36

Figura 15. Proceso actual del registro y prescripción médica 41

Figura 16. Proceso optimizado del registro y prescripción médica 43

Figura 17. Arquitectura del sistema 52

Figura 18. Versión de Python 65

Figura 19. Versión de PostgreSQL 66

Figura 20. Entorno virtual “virtualenv” 66

Figura 21. Instalación de dependencias en el entorno virtual 67

Figura 22. Creación del proyecto Django 67

Figura 23. Mockup de la interfaz para la autenticación de usuario 67

Figura 24. Mockup de la interfaz para búsqueda de médicos 68

Figura 25. Mockup de la interfaz para ingresar datos del médico 68

Figura 26. Mockup de la interfaz para editar datos del médico 69

Figura 27. Mockup de la interfaz para eliminar un médico 69

Figura 28. Mockup de la interfaz para búsqueda de pacientes 70

Figura 29. Mockup de la interfaz para el paciente encontrado 70

Figura 30. Mockup de la interfaz para la visualización de la ficha médica Parte 1 (Datos personales) 71

Figura 31. Mockup de la interfaz para la visualización de la ficha médica Parte 2 (Antecedentes) 71

Figura 32. Mockup de la interfaz para la visualización de la ficha médica Parte 3 (Traumas y alergias) 72

Figura 33. Mockup de la interfaz para el ingreso de datos personales en la ficha médica Parte 1 72

Figura 34. Mockup de la interfaz para el ingreso de antecedentes en la ficha médica Parte 2 73

Figura 35. Mockup de la interfaz para el ingreso de traumas y alergia en la ficha médica Parte 3 73

Figura 36. Botones de acción 73

Figura 37. Mockup de la interfaz para editar datos personales en la ficha médica Parte 1 74

Figura 38. Mockup de la interfaz para editar antecedentes en la ficha médica Parte 2 74

Figura 39. Mockup de la interfaz para editar de traumas y alergia en la ficha médica Parte 3..... 75

Figura 40. Mockup de la interfaz para eliminar un paciente..... 75

Figura 41. Mockup de la interfaz para la visualización de la ficha médica Parte 4 (Registros médicos)..... 76

Figura 42. Mockup de la interfaz para la visualización de la ficha médica Parte 5 (Signos vitales)..... 77

Figura 43. Mockup de la interfaz para la visualización de la ficha médica Parte 6 (diagnósticos y sus prescripciones médicas)..... 77

Figura 44. Mockup de la interfaz para la prescripción médica 78

Figura 45. Mockup del envió automático de la prescripción médica 79

Figura 46. Mockup de la interfaz para cerrar la aplicación..... 79

Figura 47. Conexión a la base de datos PostgreSQL 80

Figura 48. Tablas de administración propias de Django..... 80

Figura 49. Archivo model parte 1 81

Figura 50. Archivo model parte 2 82

Figura 51. Archivo model parte 3 83

Figura 52. Tablas de la base de datos generada por models 84

Figura 53. Archivo admin.py para la administración de datos con Django..... 84

Figura 54. Ventana administrativa de Django..... 85

Figura 55. Template de autenticación de usuario 85

Figura 56. Método de autenticación login_view 86

Figura 57. Página de autenticación de usuario 86

Figura 58. Mensaje de credenciales invalidad 87

Figura 59. Listado de médicos registrados 87

Figura 60. Métodos CRUD para el módulo de usuarios 88

Figura 61. Templates del módulo usuarios 88

Figura 62. Rutas del módulo de usuarios 89

Figura 63. Editar los datos del médico 89

Figura 64. Eliminación de médico 90

Figura 65. Crear un nuevo médico 90

Figura 66. Listado de pacientes registrados 91

Figura 67. Métodos CRUD para el módulo de paciente 91

Figura 68. Templates del módulo paciente 92

Figura 69. Rutas del módulo de pacientes 92

Figura 70. Crear un nuevo paciente 93

Figura 71. Editar un paciente 93

Figura 72. Eliminar un paciente 94

Figura 73 Acción mostrar ficha médica 94

Figura 74. Visualización de la ficha médica de un paciente existente parte 1 95

Figura 75. Visualización de la ficha médica de un paciente existente parte 2 95

Figura 76. Visualización de la ficha médica de un paciente existente parte 3 96

Figura 77. Vista para listar fichas médicas por paciente 96

Figura 78. Registro de signos vitales 97

Figura 79. Registro de los diagnósticos médicos 97

Figura 80. Registro las prescripciones médicas 98

Figura 81. Registro de prescripción médica 98

Figura 82. Configuración correo electrónico 99

Figura 83. Vista para el envío de correo electrónico 100

Figura 84. Reporte de consultas por mes 101

Figura 85.Reporte de consultas por día 101

Figura 86. Vista para visualizar los reportes 102

Figura 87. Proyecto backend en Heroku 105

Figura 88. Credenciales de la base de datos de Heroku 106

Figura 89. Conexión a la base de datos de Heroku 106

Figura 90. Base de datos alojada en Heroku 107

Figura 91. Ruta del proyecto Django en el terminal 107

Figura 92. Aplicación en el navegador web de manera local 107

ÍNDICE DE ANEXOS

Anexo A Alfa de Cronbach en la encuesta de pacientes 113

Anexo B Cuestionario de la encuesta dirigida a los pacientes del consultorio Médico "DuoMedic" 115

Anexo C Guía de entrevista dirigida al profesional de la salud del consultorio Médico "DuoMedic" 119

Anexo D Diseño de la base de datos 120

RESUMEN EJECUTIVO

El presente proyecto de investigación se centra en el desarrollo e implantación de una aplicación web para el control de fichas médica y control de posología para el consultorio médico “DuoMedic”, el cual se desarrolló para que el usuario de la aplicación tenga acceso y permita el registro de datos médicos de los pacientes de manera eficiente y optimizada.

La aplicación permite la manipulación de datos como crear, eliminar, editar y mostrar fichas médicas de los datos de los pacientes, así como también generar una receta médica digital detalla para evitar errores de prescripción para proceder enviarla al paciente. Se desarrolló la aplicación aplicando la metodología Extreme Programming (XP) para gestionar el desarrollo del mismo, permitiendo realizar cambios de manera adecuada y organizada. La aplicación tiene como framework de desarrollo Django en su versión 5.0.6 que tiene incorporado la estructura Modelo Vista Template (MVT) el mismo que ayuda al registro de flujo de datos, el gestor de base de datos utilizada para la manipulación de datos fue PostgreSQL en su versión 16.3 que proporciona seguridad, robustez y accesibilidad a los datos. Para la implantación de backend se lo realizado en la plataforma de servicios en la nube Heroku.

Palabras clave: Aplicación web, metodología XP, Django, MVT, PostgreSQL, fichas médicas, prescripción médica

ABSTRACT

The present research project focuses on the development and implementation of a web application for médical record management and dosage control for the médical office "DuoMedic." The application was developed to provide users with efficient and optimized access to médical records of patients.

The application allows for the manipulation of data, including creating, deleting, editing, and displaying patients' médical records. It also generates detailed digital médical prescriptions to avoid prescription errors, which can then be sent to the patient. The application was developed using the Extreme Programming (XP) methodology to manage its development, enabling appropriate and organized changes.

The application utilizes Django version 5.0.6 as its development framework, which incorporates the Model-View-Template (MVT) structure to assist with data flow registration. The database management system used for data manipulation was PostgreSQL version 16.3, providing security, robustness, and data accessibility. The backend was deployed on the cloud service platform Heroku.

Keywords: Web application, XP methodology, Django, MVT, PostgreSQL, médical records, médical prescription.

CAPÍTULO I. MARCO TEÓRICO

1.1 Tema de investigación

APLICACIÓN WEB USANDO PYTHON PARA EL REGISTRO DE LA FICHA MÉDICA Y CONTROL DE LA POSOLOGÍA PARA EL CONSULORIO MÉDICO "DUOMEDIC"

1.1.1 Planteamiento del problema

El uso de las Tecnologías de la Información y Comunicaciones (TIC 's) en el campo de la medicina se ha expandido y evolucionado de manera exponencial en las últimas décadas [1]. La innovación tecnológica actual permite tener un mejora en la eficiencia y eficacia al momento de brindar una atención médica a los pacientes [2]. La combinación de TIC 's con la medicina moderna ha creado un panorama favorable en la prevención, diagnósticos, tratamientos y seguimientos de enfermedades permitiendo establecer una atención médica más accesible y equitativa para todos [1].

En Ecuador hasta el año 2023 existen 3.847 establecimientos de atención a la salud, de los cuales 1.674 son establecimientos de salud ambulatorios incluyendo centros de salud, consultorios médicos, clínicas privadas entre otros y 125 son hospitales [3], [4]. El sistema de salud en el Ecuador está en constante evolución, los hospitales se están equipando con las nuevas tecnologías para mejorar la atención médica ofreciendo un servicio de calidad a los pacientes [5]. Sin embargo, a pesar de los avances tecnológicos, se identificaron deficiencias en el registro de ficha médica y el control de Posología en los Centros de Salud tipo B del Distrito 17D04 [6].

En la ciudad de Ambato, la situación de las prescripciones o recetas médicas en hospitales es similar a la de los consultorios médicos [7]. Algunos de ellos aún utilizan métodos obsoletos, como formularios en papel o recetas escritas a mano, lo cual incrementa el riesgo de errores o confusiones en los medicamentos recetados. En el caso del consultorio médico "DUOMEDIC" busca implementar tecnologías que permitan el registro de fichas médicas y el control de Posología permitiendo brindar un servicio de calidad y procurar la salud integral de cada uno de sus pacientes. La

adopción de tecnologías más modernas y eficientes en el proceso de prescripción médica puede contribuir a mejorar la atención y seguridad de los pacientes, evitando posibles errores en la dosificación y administración de medicamentos.

1.2 Antecedentes investigativos

En el presente trabajo, se llevó a cabo una revisión minuciosa de literatura de artículos publicados en las siguientes bases de datos científicas: SCOPUS, IEEE Explorer, Google Académico y repositorios de universidades. Se utilizó cadenas de búsquedas claves tales como: Desarrollo web, Frameworks basados en Python, Historias Clínicas, Fichas Médicas, Prescripción Médica, Receta Médica bajo el límite de publicación entre los años 2020 y 2024.

El autor [8], propone la implementación de un aplicativo tipo web utilizando la tecnología Django, perteneciente al lenguaje de programación Python, con el objetivo de gestionar y optimizar las actividades de venta de productos y artículos deportivos dentro de un establecimiento deportivo. En el trabajo, se detalla el flujo de desarrollo del aplicativo, el cual permite realizar operaciones CRUD (Crear, Leer, Actualizar y Eliminar) sobre los diferentes tipos de entradas, ya sean actividades o productos ofrecidos en el centro deportivo. Para llevar a cabo este proyecto, se utilizaron herramientas de código abierto y gratuito, como el framework web Django y el sistema de gestión de base de datos SQLite3. Además, la arquitectura de software empleada fue el patrón Modelo-Vista-Controlador (MVC). Sin embargo, el investigador resalta que, si bien el uso inicial de Django puede resultar un poco complejo, una vez que se le dedica el tiempo necesario, se convierte en un entorno de desarrollo web bastante sencillo y completo.

En [9] presenta el desarrollo del proyecto investigativo de un sistema web para la gestión de las historias clínicas, asignación de citas y control de insumos y medicamentos en el dispensario médico de la empresa Leterago del Ecuador S.A. sede Quito (SWL). En el trabajo se detalla la arquitectura del sistema, el cual utiliza tecnologías web como Boopstrap, MySQL, AJAX y Crystal Report 2017 como sistema de emisión de reportes. El autor menciona que el sistema cuenta con más de 25,000 registros de pacientes, 12,000 historias clínicas y 5,000 registros de inventario

de medicamentos. Las principales características contempladas en el desarrollo fueron: gestión de pacientes, agendamiento de citas, control de inventario, generación de reportes y autenticación de usuarios con diferentes perfiles de acceso. El autor concluye que el uso de una aplicación web facilita el diseño, construcción e implementación de soluciones informáticas que permiten automatizar los diferentes procesos y necesidades que surgen a diario en la empresa.

El autor [10], contribuyó a la empresa INGELEC-BM con un proyecto de aplicación web para la regulación de presupuestos en proyectos eléctricos. Después de realizar investigaciones de las diferentes herramientas de desarrollo web opto por uso del Framework Django ya que contaba con conocimientos básicos para el desarrollo y para el almacenamiento de datos el MySQL. El proyecto se basa en la arquitectura del framework Django utilizando el patrón Modelo-Vista-Controlador (MVC) para el trabajo y representación del proyecto, lo que automatiza varios procesos y aligerar la carga del usuario. En el análisis, se consideraron las ventajas que posee, como su versatilidad y flexibilidad al utilizar o importar bases de datos, así como una gran cantidad de librerías que complementan al framework. Sin embargo, se señaló como desventaja la abundante documentación que lo acompaña, la cual puede resultar confusa o de difícil comprensión para el usuario [10].

Según [11], propuso desarrollar un sistema web para la gestión de fichas médicas para la organización “Salud Integral para Todos”, usando la plataforma de programación JEE (Java Enterprise Edition) para aplicaciones web. El autor menciona que, dentro de este proyecto, se optó por usar la distribución de Ubuntu Server como servidor principal para el desarrollo, PostgreSQL como gestor de base de datos y JEE junto con Wildfly, un servidor de aplicaciones de código abierto, como lenguaje y entorno de programación. Además, se adoptó la metodología ágil Scrum para gestionar el desarrollo del sistema. El autor concluye que el sistema presentado tiene un comportamiento adecuado para una concurrencia de 50 a 100 usuarios, sin que ello afecte el tiempo de respuesta a cada petición.

1.3 Fundamentación teórica

La fundamentación teórica de este proyecto abarca aspectos clave del desarrollo de software aplicado en la gestión de la salud. Comienza con una visión general del desarrollo de software y las metodologías ágiles, se consideran pilares fundamentales para la creación de aplicaciones eficientes y adaptables. Las aplicaciones web, soportadas por frameworks de desarrollo y específicos de Python para facilitar su desarrollo. En el ámbito de la salud, se destaca la gestión como la coordinación integral de servicios médicos y recursos, haciendo constancia en la ficha médica y la necesidad de una prescripción segura conforme a la normativa de Ecuador, incluyendo el control de medicamentos para garantizar su uso adecuado y seguro.

1.3.1 Desarrollo de software

El desarrollo del software se puede definir como el proceso de transformar una idea abstracta en un producto funcional que va a ser utilizado por usuarios finales. Este proceso involucra las actividades como el diseño, la programación, las pruebas y el mantenimiento, todas ellas realizadas con diferentes habilidades técnicas y creativas [12].

1.3.2 Metodología Ágil

La metodología ágil es un enfoque de gestión de proyectos de desarrollo de software que promueven la flexibilidad, la colaboración y la mejora continua. Se centran en la entrega rápida de productos funcionales mediante ciclos iterativos y cortos llamados "Sprints" o "Iteraciones". En lugar de seguir un plan fijo, las metodologías ágiles permiten adaptarse a los cambios y responder a las necesidades emergentes del proyecto y del cliente. Equipos pequeños y multifuncionales trabajan juntos, comunicándose de manera constante para resolver problemas y ajustar prioridades [13].

a. Scrum

Scrum es una metodología ágil diseñada para administrar y controlar el desarrollo de software. Utiliza ciclos cortos conocidos como sprints, que normalmente duran entre dos y cuatro semanas. Durante cada sprint, se desarrollan y completan tareas específicas con el objetivo de mejorar continuamente el producto [14].

- ***Características***

La transparencia, la inspección y la adaptación son características fundamentales en Scrum. Promueve la colaboración continua entre los miembros del equipo y la mejora constante tanto del producto como del proceso de trabajo. Estas características permiten una rápida identificación y solución de problemas, asegurando que el equipo se mantenga en el camino correcto hacia la entrega de valor al cliente [14].

- ***Proceso***

El ciclo de Scrum incluye varias fases clave. Comienza con la planificación del sprint, donde el equipo define qué trabajo se realizará. Durante el sprint, se llevan a cabo reuniones diarias, conocidas como daily stand-ups, para revisar el progreso y ajustar el plan según sea necesario. Al final del sprint, se realiza una revisión para evaluar el incremento de producto y una retrospectiva para identificar áreas de mejora en el proceso [14].

- ***Roles***

En Scrum se implementa tres roles principales. El Product Owner que es responsable de gestionar y priorizar el backlog (lista de tareas) del producto, asegurando que el equipo trabaje en las tareas más valiosas. El Scrum Master facilita el proceso, ayudando al equipo a seguir las prácticas de Scrum y eliminando cualquier impedimento que pueda surgir. El equipo de desarrollo se enfoca en crear el producto, trabajando de manera colaborativa para cumplir con los objetivos del sprint [14].

- ***Orientación***

Scrum está dirigido a equipos que buscan mejorar continuamente y entregar valor de manera rápida. Este enfoque permite una rápida adaptación a los cambios y una

constante alineación con las necesidades del cliente, asegurando que el producto evolucione de manera efectiva y eficiente [14].

b. Kanban

La metodología Kanban se centra en visualizar el flujo de trabajo mediante el uso de un tablero. Este enfoque permite a los equipos gestionar sus tareas de manera eficiente y mejorar continuamente sus procesos. La adaptabilidad de Kanban lo hace adecuado para cualquier tipo de proceso existente, facilitando su implementación en diversos contextos [10].

- ***Características***

La visualización del trabajo es un principio clave en Kanban. Además, se enfoca en la limitación del trabajo en curso (WIP) para evitar la sobrecarga y asegurar que las tareas se completen de manera eficiente. La mejora continua es otro pilar fundamental, fomentando ajustes y optimizaciones constantes en el flujo de trabajo. A diferencia de Scrum, Kanban no requiere iteraciones fijas, ofreciendo mayor flexibilidad [15].

- ***Proceso***

El proceso en Kanban se organiza mediante un tablero que se divide en columnas representando diferentes etapas del flujo de trabajo, como "Por hacer", "En progreso" y "Hecho". Las tareas se mueven de una columna a otra a medida que avanzan, proporcionando una vista clara del estado de cada actividad. Este método facilita la identificación de cuellos de botella y ayuda a mantener el trabajo bien gestionado [15].

- ***Roles***

A diferencia de otras metodologías ágiles, Kanban no establece roles específicos. Sin embargo, es habitual contar con un facilitador que apoye en la implementación y optimización del uso del tablero. Este facilitador se asegura de que el equipo siga las prácticas recomendadas y de que el flujo de trabajo se mantenga eficiente y sin interrupciones [15].

- ***Orientación***

Kanban se orienta a mejorar el flujo de trabajo continuo, siendo ideal para equipos que no requieren iteraciones definidas. Es especialmente beneficioso en entornos donde las tareas llegan de manera impredecible y necesitan ser atendidas de inmediato. Este enfoque ayuda a los equipos a adaptarse rápidamente a los cambios y a mantener un rendimiento constante [15].

c. Programación Extrema XP

Extreme Programming (XP) es una metodología ágil enfocada en mejorar tanto la calidad del software como la capacidad de respuesta ante cambios. Fue pensada para aumentar la satisfacción del cliente mediante prácticas de desarrollo estrictas y eficientes. La metodología busca optimizar cada aspecto del proceso de desarrollo para garantizar resultados de alta calidad [16].

- ***Características***

La metodología XP fomenta prácticas como la programación en parejas, donde los desarrolladores trabajan juntos en una sola estación de trabajo, pero también se la puede trabajar con un solo desarrollador cumpliendo con el plan del trabajo [14]. XP promueve la integración continua, asegurando que el código se combine y se pruebe frecuentemente. Las pruebas unitarias frecuentes permiten detectar errores rápidamente, mientras que la retroalimentación constante del cliente asegura que el producto se ajuste a sus necesidades. Estas prácticas están diseñadas para mitigar riesgos y mejorar la calidad del software [16].

- ***Proceso***

El ciclo de XP incluye iteraciones cortas, donde el equipo planifica y ejecuta liberaciones incrementales. Durante estas iteraciones, los desarrolladores colaboran en parejas para escribir y revisar código continuamente. La integración continua asegura que cualquier cambio se pruebe de inmediato, permitiendo la detección temprana de problemas. Las pruebas constantes, tanto unitarias como de integración, son esenciales para mantener la estabilidad del código [16].

- ***Roles***

En XP, los roles son claros y colaborativos. El cliente proporciona requisitos detallados y retroalimentación constante, asegurando que el producto final cumpla con sus expectativas. Los desarrolladores se encargan de crear el software en un entorno altamente colaborativo, mientras que el coach guía al equipo en la adopción y mantenimiento de las prácticas de XP, asegurando que se sigan los principios fundamentales de la metodología [16].

- ***Orientación***

XP es ideal para equipos de desarrollo de software que necesitan mejorar la calidad del código y adaptarse rápidamente a los cambios en los requisitos del cliente. Es especialmente útil en proyectos donde la calidad del software es fundamental y donde se espera una evolución constante de los requisitos. La metodología permite a los equipos responder con agilidad a nuevos desafíos, manteniendo siempre un alto estándar de calidad en el desarrollo [16].

1.3.3 Aplicaciones web

Las aplicaciones web son programas diseñados para operar en navegadores web y se ejecutan en servidores remotos. Estas aplicaciones ofrecen una amplia variedad de servicios y funcionalidades accesibles a través de Internet. Desde herramientas de productividad y colaboración hasta servicios de Streaming y Comercio Electrónico, las aplicaciones web han revolucionado la forma en que las personas interactúan en línea [17].

Su accesibilidad desde cualquier dispositivo con conexión a Internet proporciona una experiencia flexible y conveniente, sin necesidad de instalaciones o actualizaciones locales. Las aplicaciones web se adaptan fácilmente a diferentes plataformas, y su enfoque en la conectividad en línea las convierte en una parte integral de nuestra vida digital. Continúan facilitando nuestra interacción en diversos ámbitos de nuestras actividades cotidianas [18].

1.3.4 Frameworks de desarrollo

Un framework de desarrollo es un conjunto de herramientas, librerías y convenciones que se utilizan para facilitar la creación de aplicaciones de software. Los frameworks proporcionan una estructura y un conjunto de reglas para el desarrollo de software, lo que permite a los desarrolladores enfocarse en la lógica de negocio en lugar de preocuparse por la infraestructura y la arquitectura subyacente [19].

Algunas características comunes de los frameworks son:

a. Arquitectura:

Los frameworks suelen proporcionar una arquitectura base para las aplicaciones, lo que permite a los desarrolladores enfocarse en la lógica de negocio en lugar de preocuparse por la estructura subyacente [19].

b. Librerías:

Los frameworks suelen incluir librerías predefinidas para tareas comunes, como el manejo de bases de datos, la validación de formularios, la autenticación de usuarios [19].

c. Convenciones:

Los frameworks suelen imponer convenciones y buenas prácticas para el desarrollo, lo que ayuda a mantener un código consistente y fácilmente mantenible [19].

d. Reutilización:

Los frameworks permiten la reutilización del código, lo que ayuda a reducir el tiempo y costo de desarrollo [19].

1.3.5 Frameworks de Python

Python es un lenguaje de programación de alto nivel, interpretado y multiparadigma que se destaca por su sintaxis clara y legible. Ofreciendo varios frameworks que simplifican y aceleran la creación de aplicaciones web. Estos frameworks

proporcionan un conjunto de herramientas y patrones de diseño predefinidos para manejar tareas comunes, como enrutamiento de URL (Uniform Resource Locator), manejo de solicitudes HTTP (Hypertext Markup Language), acceso a bases de datos y seguridad [20].

a. Django

Django es un framework web de alto nivel para Python, diseñado para facilitar el desarrollo rápido y un diseño limpio y pragmático. Está pensado para ayudar a los desarrolladores a crear aplicaciones desde la idea inicial hasta la implementación final de manera eficiente [21].

- ***Características***

Django ofrece una interfaz administrativa lista para usar, un mapeo objeto-relacional (ORM) para interactuar con bases de datos, y múltiples protecciones integradas contra vulnerabilidades comunes ofreciendo seguridad robusta de datos. Además, soporta la internacionalización, permitiendo el uso de múltiples idiomas y localizaciones en una aplicación [19].

- ***Arquitectura de un proyecto Django***

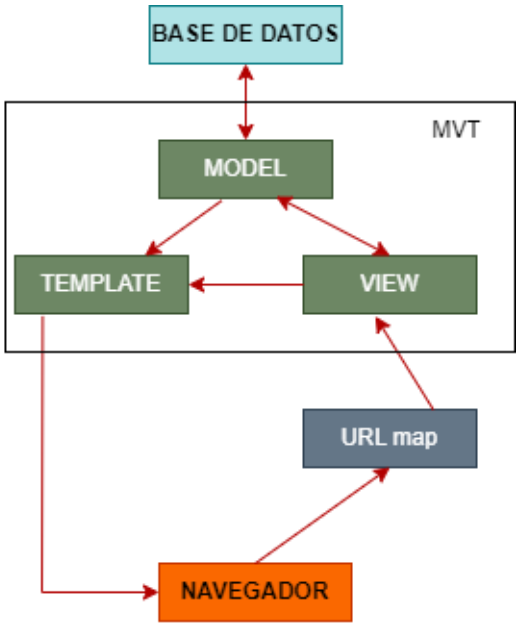


Figura 1. Estructura de un proyecto Django

La arquitectura básica de un proyecto Django, involucra y relaciona los diferentes componentes como lo es el navegador, URL Map, Modelo (MVT) y un motor de base de datos. El flujo comienza cuando el navegador realiza una solicitud HTTP. Django utiliza su URL Map para determinar qué vista de Python debe manejar esa solicitud entrante. La vista recupera los datos necesarios del modelo y los procesa según la lógica de la aplicación. Luego, la vista renderiza una plantilla con esos datos procesados, generando una respuesta HTML final. Esta respuesta HTML es enviada de vuelta al navegador para ser mostrada al usuario [21].

- ***Tecnología***

Django utiliza Python y es compatible con bases de datos como PostgreSQL, MySQL, SQLite y Oracle. También se puede integrar con otras tecnologías a través de sus APIs y middleware, lo que facilita la extensión de sus capacidades [21].

Django se utiliza ampliamente para desarrollar aplicaciones web complejas y robustas de manera rápida y segura. Es muy popular para sitios web de noticias, redes sociales, blogs, aplicaciones empresariales, portales de administración, comercio electrónico, servicios web RESTful, APIs, proyectos de ciencia de datos e inteligencia artificial, entre otros [19].

- ***Ventajas y desventajas***

Tabla 1. Ventajas y desventajas de Django

Ventajas	**Desventajas**
• Promueve el desarrollo rápido gracias a su filosofía DRY (Don't Repeat Yourself). • Fuerte enfoque en la seguridad, lo que lo hace ideal para aplicaciones donde la protección de datos es crucial. • Cuenta con una amplia comunidad y muchos recursos disponibles para los desarrolladores.	• Complejo para desarrolladores principiantes debido a su curva de aprendizaje. • Es más pesado en comparación con microframeworks como Flask, lo que puede ser una desventaja para aplicaciones pequeñas

b. Flask

Flask es un microframework web minimalista de codigo abierto escrito en Python. Este framework se centra en la simplicidad y la flexibilidad en el desarrollo del

proyecto de diferentes tamaños y complejidades. Este framework permite agregar nuevas extensiones adicionales según sea la necesidad para el desarrollo web [22].

- ***Características***

Este microframework es ligero que no utiliza bibliotecas externas en su desarrollo, lo que lo hace sencillo y manejable. No incluye una capa de base de datos ni arreglos para la aprobación de formularios, pero admite expansiones. Una de sus características más destacadas es su sistema integrado de pruebas unitarias, que facilita la depuración y promueve un desarrollo robusto. Además, Flask soporta la introducción de extensiones de terceros, como Flask SQLAlchemy para bases de datos y WTForms para formularios web, lo que le permite añadir funcionalidades adicionales según las necesidades de la aplicación.

- ***Arquitectura de un proyecto Flask (MVC)***

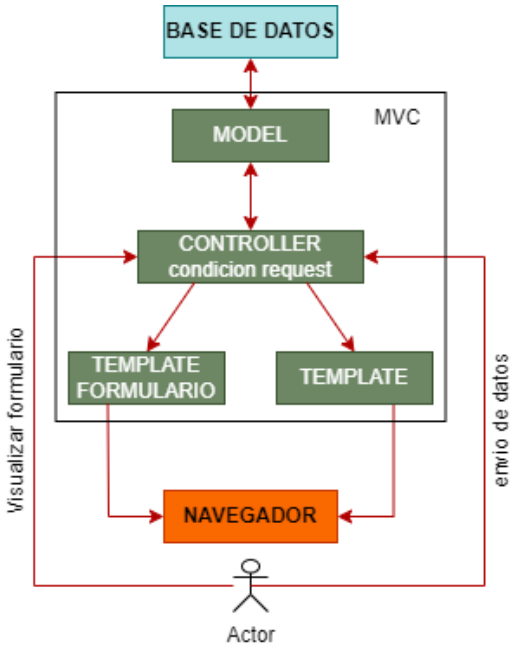


Figura 2. Estructura de un proyecto Flask (MVC)

Flask sigue un patrón de diseño similar al Modelo-Vista-Controlador (MVC), pero con algunas variaciones adaptando un enfoque más flexible y ligero. A diferencia de otros frameworks, Flask no tiene una arquitectura predeterminada, lo que permite a los desarrolladores elegir cómo estructurar su aplicación. Normalmente, sigue una

estructura modular con rutas para manejar las URLs y solicitudes, plantillas para generar HTML, y extensiones para añadir funcionalidades adicionales [23].

- ***Tecnología***

Flask se puede integrar con una variedad de bases de datos y herramientas de terceros gracias a su arquitectura flexible. Utiliza Werkzeug como servidor web y Jinja2 como motor de plantillas, lo que le permite ser flexible y adaptable [23].

- ***Ventajas y desventaja***

Tabla 2. Ventajas y desventajas de Flask

Ventajas	**Desventajas**
• Su ligereza lo hace ideal para proyectos pequeños y medianos. • Es fácil de aprender y usar, lo que lo hace accesible para desarrolladores de todos los niveles. • Su flexibilidad permite gran libertad en la estructura del proyecto, lo que es útil para aplicaciones que requieren personalización.	• Ofrece menos funcionalidades integradas en comparación con frameworks más grandes, lo que puede requerir más configuración para proyectos grandes. • No es tan robusto en términos de seguridad y escalabilidad como Django, lo que puede ser una limitación para aplicaciones más complejas.

c. Tornado

Tornado es un framework web de Python de código abierto y multiplataforma. Utiliza un patrón de programación asíncrona, conocido como "callbacks", que permite que las operaciones de entrada/salida no bloqueen la ejecución del programa. En lugar de esperar a que una operación se complete, el programa registra una devolución de llamada que se ejecutará cuando la operación finalice. Esto lo convierte en ideal para aplicaciones web de alto tráfico, servicios web requieren un rendimiento elevado y alta concurrencia [24].

- ***Características***

Tornado cuenta con un servidor web asíncrono capaz de manejar miles de conexiones abiertas simultáneamente, lo que lo hace ideal para servicios de chat y servidores de juegos. Permitiendo comunicación bidireccional en tiempo real entre el servidor y el cliente. Incluye herramientas integradas para la autenticación y la seguridad, garantizando la protección de las aplicaciones. Además, tiene soporte para la

internacionalización, permitiendo a las aplicaciones servir contenido en múltiples idiomas. Tornado no incluye un sistema de mapeo (ORM) ni plantillas integradas. Sin embargo, es compatible con muchas bibliotecas y herramientas externas populares de Python. Estas características hacen de Tornado una opción potente y flexible para el desarrollo de aplicaciones web modernas y escalables [24].

- ***Arquitectura***

Tornado utiliza un modelo de programación asincrónica basado en eventos, que le permite gestionar múltiples conexiones simultáneas sin bloquear. Su arquitectura se centra en componentes clave como, IOLoop: Es el manejador de eventos principal que realiza un bucle infinito, procesando eventos y gestionando entradas/salidas no bloqueantes. HTTPServer: Módulo que maneja las solicitudes HTTP y distribuye las peticiones a los manejadores correspondientes. Coroutines: Facilitan la escritura de código asincrónico de manera secuencial mediante el uso de la palabra clave async y await.

- ***Tecnología***

Tornado está escrito en Python y se integra bien con otras tecnologías asincrónicas y bases de datos que soportan operaciones no bloqueantes, como MongoDB y Redis. Tornado puede ser utilizado como un servidor web independiente o como parte de una aplicación más grande, y se puede combinar con otros frameworks y bibliotecas para extender su funcionalidad [25].

- ***Ventajas y desventajas***

Tabla 3. Ventajas y desventajas de Tornado

Ventajas	**Desventajas**
• Eficiente en la gestión de conexiones concurrentes, lo que lo hace ideal para aplicaciones que requieren baja latencia y alta capacidad de respuesta. • Su capacidad para manejar una gran cantidad de conexiones simultáneas lo hace extremadamente escalable. • El modelo de programación asincrónica permite que las operaciones se ejecuten de manera no bloqueante, mejorando el rendimiento y la eficiencia.	• La programación asincrónica puede ser difícil de dominar para los desarrolladores que no están familiarizados con este paradigma. • Recursos, tutoriales y una comunidad más pequeña. • Puede requerir más configuración inicial en comparación con otros frameworks que ofrecen más funcionalidades integradas.

1.3.6 Gestión de salud

La gestión de salud se refiere a la planificación, organización, coordinación y supervisión de los servicios y recursos relacionados con la atención médica y el bienestar de las personas. Su objetivo principal es garantizar la prestación efectiva y eficiente de servicios de salud de calidad, así como mejorar la salud y el bienestar de la población en general [26].

La gestión de salud abarca diversos aspectos, incluyendo la administración de hospitales y centros de atención médica, la planificación de recursos humanos y materiales, la gestión financiera y presupuestaria, la regulación y el cumplimiento de normativas sanitarias, la gestión de tecnologías de la información en salud, la gestión de datos y registros médicos, la evaluación y mejora de la calidad de los servicios, y la gestión de la seguridad y riesgos en el ámbito sanitario [27].

Los profesionales de la gestión de salud, como administradores de hospitales, directores de servicios de salud y gerentes de programas de atención médica, desempeñan un papel crucial en la coordinación y optimización de los recursos para garantizar la provisión de servicios de salud adecuados y accesibles. Además, la gestión de salud también implica la colaboración con profesionales clínicos y proveedores de servicios de salud para asegurar una atención integral y de calidad [26].

1.3.7 Control y seguimiento

El control y seguimiento se refiere al proceso de supervisar y evaluar de manera continua y sistemática el progreso, rendimiento o cumplimiento de un proyecto, proceso, actividad o conjunto de objetivos. Implica monitorear de cerca el desarrollo y los resultados, así como compararlos con los planes establecidos o los estándares predefinidos [28].

Esto implica asegurarse de que se cumplan los estándares de calidad, se alcancen los resultados esperados y se optimice el uso de los recursos [28].

1.3.8 Ficha médica

La ficha médica se define como el documento médico legal donde se queda registrada toda la información personal y estados de salud de un paciente [29]. Esta herramienta tiene como objetivo principal registrar y monitorear el proceso individual de atención de un paciente facilitando la toma de decisiones clínicas y el seguimiento de los tratamientos prescritos [30].

Además de contener información detallada sobre los antecedentes, diagnósticos y tratamientos recibidos, también registra la evolución y respuesta del paciente a lo largo del tiempo. Este registro histórico completo permite a los profesionales de la salud tomar decisiones informadas y personalizadas basadas en el historial clínico del paciente [29].

1.3.9 Prescripción médica

La prescripción médica es el acto mediante el cual un profesional de la salud autorizado, como un médico, emite una orden por escrito o electrónica para el suministro de medicamentos, tratamientos o terapias específicas a un paciente. La prescripción médica se realiza con el propósito de diagnosticar, prevenir, tratar o aliviar una condición médica [31].

Generalmente incluye información detallada sobre el medicamento o tratamiento prescrito, como el nombre genérico o comercial, la dosis, la forma de administración, la frecuencia y la duración del tratamiento. También puede incluir instrucciones adicionales, como las condiciones o restricciones de uso, así como advertencias o precauciones pertinentes [31].

a. Normativa que regula la receta médica en Ecuador

En [32],se establecen normas técnicas para el control de la prescripción de medicamentos y la emisión de recetas en establecimientos de salud en Ecuador, con el objetivo de garantizar la calidad y seguridad en la atención médica.

El Acuerdo Ministerial 0031-2020 publicado en el Registro Oficial 251 de 22 de julio de 2020, establece el "Reglamento para establecer el contenido y requisitos de la receta médica y control de la prescripción, dispensación y expendio para medicamentos de uso y consumo humano" [33].

En su artículo 4, este acuerdo detalla los requisitos específicos que debe cumplir una receta médica para ser válida. Entre estos requisitos se encuentran:

- Datos del paciente: Nombres y apellidos completos, cédula de identidad o pasaporte, edad, sexo, peso (opcional), fecha de nacimiento, dirección y número de contacto telefónico.

- Datos del profesional de la salud: Nombres y apellidos completos, cédula de identidad profesional, especialidad, número de licencia, firma y sello.

- Datos del médicamento: Nombre genérico del médicamento (sin siglas ni abreviaturas), forma farmacéutica, concentración del médicamento, cantidad de médicamento a dispensar (en números y letras), vía de administración, posología (dosis y frecuencia de administración), instrucciones especiales (si las hay) y fecha de prescripción.

b. Instructivo para el uso de la receta médica

El Instructivo para el uso de la receta médica, emitido por el Ministerio de Salud Pública, complementa al Acuerdo Ministerial 0031-2020 y proporciona directrices adicionales para la correcta prescripción y dispensación de medicamentos [33].

En este instructivo se destacan los siguientes aspectos:

- ***Profesionales autorizados para prescribir:*** Médicos, odontólogos y obstetrices, estrictamente dentro de su ámbito de competencia y respetando los niveles de prescripción establecidos en el Cuadro Nacional de Medicamentos Básicos.

- ***Prescripción de un solo médicamento por receta:*** Cada receta debe incluir la prescripción de un solo médicamento.

- ***Legibilidad de la receta:*** La receta debe estar escrita con letra legible, utilizando el nombre genérico del médicamento sin siglas ni abreviaturas.

- ***Validación de la receta:*** La receta debe estar firmada y sellada por el profesional de la salud que la prescribe.

- ***Formato de la receta:*** La receta puede ser física (impresa o manuscrita) o electrónica. En el caso de las recetas electrónicas, el sistema utilizado debe garantizar la seguridad y confiabilidad de la información.

1.3.10 Control de prescripción médica

Conjunto de medidas y procedimientos implementados para asegurar el uso adecuado, seguro y responsable de los medicamentos prescritos por profesionales de la salud. El objetivo principal del control de prescripción es garantizar que los pacientes reciban los medicamentos correctos, en las dosis adecuadas, con las instrucciones de uso claras y en el momento adecuado [34].

El control de prescripción médica implica diversos aspectos, entre ellos:

a. Verificación y validación

Se verifica la legitimidad de la prescripción médica, asegurándose de que sea emitida por un profesional de la salud debidamente autorizado. Se valida que la prescripción sea clara, completa y cumpla con los estándares establecidos.

b. Interacción farmacológica

Se analiza la prescripción para identificar posibles interacciones o contraindicaciones entre los medicamentos prescritos y otros que el paciente pueda estar tomando actualmente. Esto ayuda a prevenir riesgos asociados a la combinación de medicamentos.

c. Dosificación adecuada

Se asegura de que la dosis prescrita sea la apropiada para el paciente, teniendo en cuenta su edad, peso, estado de salud y otras condiciones relevantes. Esto ayuda a evitar la administración de dosis excesivas o insuficientes.

d. Cumplimiento y adherencia

Se fomenta el cumplimiento de las instrucciones de uso por parte del paciente, asegurándose de que entienda y siga correctamente las indicaciones del medicamento. Esto puede incluir brindar información sobre la forma de administración, la frecuencia y la duración del tratamiento.

e. Registro y seguimiento

Se realiza un registro adecuado de las prescripciones médicas y se lleva un seguimiento de su cumplimiento. Esto puede incluir la revisión de historiales médicos, la actualización de registros electrónicos y la comunicación con el paciente y otros profesionales de la salud involucrados en el cuidado del paciente.

1.4 Objetivos

1.4.1 Objetivo general

Implantar una aplicación web usando Python para el registro de la ficha médica y control de la Posología para el consultorio médico DuoMedic.

1.4.2 Objetivos específicos

- Investigar los procesos relacionados con los registros de fichas médicas y control de la Posología.
- Investigar sobre los frameworks basados en Python para la creación de la aplicación web.
- Desarrollar la aplicación web usando en Python para el registro de la ficha médica y control de la Posología para el consultorio médico DuoMedic

CAPÍTULO II. METODOLOGÍA

2.1 Materiales

En el presente proyecto de investigación se realizó una evaluación al Consultorio Médico "DuoMedic" con el objetivo de identificar las dificultades y oportunidades de mejora en el manejo de las fichas médicas y el control de la posología de los pacientes. Se aplicó una encuesta a los pacientes para conocer su nivel de satisfacción con los procesos actuales, así como su apertura a la implementación de una aplicación web para el registro de información y el envío digital de recetas. No obstante, también se llevó a cabo una entrevista con el médico personal del consultorio para comprender los procesos, entender el flujo de trabajo como el registro de fichas médicas, presciencia de la receta médica y la gestión de la información clínica de los pacientes.

Los instrumentos utilizados para la recolección de datos **Anexo B**, **Anexo C**.

2.2 Métodos

2.2.1 Modalidad de la investigación

EL presente estudio se llevó a cabo bajo una perspectiva que abarca los enfoques de investigación de la modalidad bibliográfica, descriptiva y aplicada.

Investigación Bibliográfica

Se llevó a cabo una investigación bibliográfica-documental, en la cual se recopiló información importante de libros virtuales, tesis y artículos científicos. El objetivo era obtener información sobre los procesos para el registro de la ficha médica, el control de la posología de las prescripciones y el desarrollo de aplicaciones web. La cual permitió obtener información esencial y útil para proceso de desarrollo del proyecto.

Investigación Descriptiva

Se utilizó la investigación descriptiva para obtener conocimientos necesarios sobre las metodologías de diseño web y su aplicación basada en un framework de Python para

el registro de la ficha médica y el control de la posología de prescripciones. Se recopiló información de expertos en el tema para asegurar el cumplimiento de los objetivos establecidos.

Investigación Aplicada

Esta investigación se llevó a cabo en un enfoque aplicado, que implicó la recopilación y análisis de datos relevantes utilizando técnicas e instrumentos de recolección de datos previamente estudiados. Estos datos se utilizaron para el diseño y desarrollo de una aplicación web que cumplió con los objetivos establecidos en la investigación.

2.2.2 Población y muestra

La población con la que se consideró, a los pacientes y al profesional de la salud del consultorio médico "DuoMedic".

Tabla 4. Población de Estudio

Población	Número	Porcentaje
Pacientes	200	99.51%
Doctor	1	0.49%
Total	201	100%

Muestra

Debido a que la población de paciente es mayor a 100, se procederá a calcular el tamaño de muestra representativa.

Aplicando la formula del cálculo de muestra con población finita, se procede a obtener el tamaño de la muestra.

Formula:

$$\boldsymbol{n = \frac{N\sigma^2 Z^2}{(N-1)e^2 + \sigma^2 Z^2}} \quad (1)$$

En donde:

n = Tamaño de la muestra

N = Población: 200

Z = Nivel de Confianza 95% Equivalente al 1.96

σ = Desviación estándar constante: 0.5

e = Margen de error: 0.1

$$\boldsymbol{n = \frac{(200) * (0.5)^2 * (1.96)^2}{(200-1) * (0.1)^2 + (0.5)^2 * (1.96)^2}} \quad (2)$$

$$n = 65{,}103$$

Por lo que la muestra de pacientes se puede aproximar a 65 Pacientes.

El valor del tamaño de la muestra será 65 pacientes al haber realizado el respectivo calculo. En el caso doctor la muestra no excede los 100 entonces se trabajará con el valor total por tanto el tamaño de muestra corresponde a 66 personas.

Tabla 5. Muestra

Población	**Número**	**Porcentaje**
Pacientes	65	98.48%
Doctor	1	1.52%
Total	66	100%

2.2.3 Recolección de información

Las técnicas que se realizaron para la recolección de la información fueron las encuestas para los pacientes conformada por 12 preguntas de escala de Likert, la entrevista estructura dirigida al profesional de la salud del consultorio médico "DuoMedic", donde se recolecto información para ser analizadas y obtener una respuesta optima hacia la solución.

a. Validación del instrumento

Alfa de Cronbach en la encuesta de pacientes

Para validar el nivel de confiabilidad de la encuesta se utilizó el algoritmo de alfa de Cronbach, aplicando a las 12 preguntas dirigidas a los pacientes del consultorio médico "DuoMedic" que fueron elaboradas en formato de escala de Likert.

Para el cálculo de la confiabilidad del alfa de Cronbach se aplicó la siguiente formula:

$$\boldsymbol{\alpha = \left(\frac{k}{k-1}\right)\left(1 - \frac{\sum v_i}{vt}\right)} \quad (3)$$

Los cálculos fueron elaborados en Excel **Anexo A**, para una mayor facilidad al utilizar la fórmula.

Tabla 6. Confiabilidad alfa de Cronbach en la encuesta de pacientes

Simbología	**Valor**
α (alfa) =	0,749
k (Número de items)	12
$\sum v_i$ (Varianza de cada item)	13,725
vt (varianza total)	43,831

Tabla 7. Interpretación de la escala de confiabilidad Alfa de Cronbach.

Rango	**Confiabilidad**
0.53 a menos	Confiabilidad nula
0.54 a 0.59	Confiabilidad baja
0.60 a 0.65	Confiable
0.66 a 0.71	Muy confiables
0.72 a 0.99	**Excelente confiabilidad**
1	Confiabilidad perfecta

Como se muestra en la **Tabla 6**, el resultado del nivel de confiabilidad del alfa de Cronbach dio un valor de 0,749, indicando que el conjunto de ítems establece un nivel de Excelente confiabilidad como se muestra en la **Tabla 7**.

b. Resultados de la encuesta aplicada a los pacientes

La encuesta fue aplicada a los pacientes del consultorio médico “DuoMedic”, aun total de 66 pacientes mediante la herramienta de Google Forms.

Para generar las gráficas correspondientes a las respuestas de cada pregunta, se empleó un archivo CSV generado por la herramienta de Google Forms. Posteriormente, se utilizó Python en conjunto con las bibliotecas pandas y matplotlib para este propósito.

Pregunta 1: ¿Desde hace cuánto tiempo es paciente del Consultorio Médico "DuoMedic"?

Tabla 8. Resultado de la pregunta 1

Indicador	Frecuencia	Porcentaje
Menos de 3 meses	10	15%
Entre 3 meses y 1 año	12	18%
Entre 1 y 2 años	16	24%
Entre 2 y 3 años	16	24%
Más de 3 años	12	18%
Total	66	100%

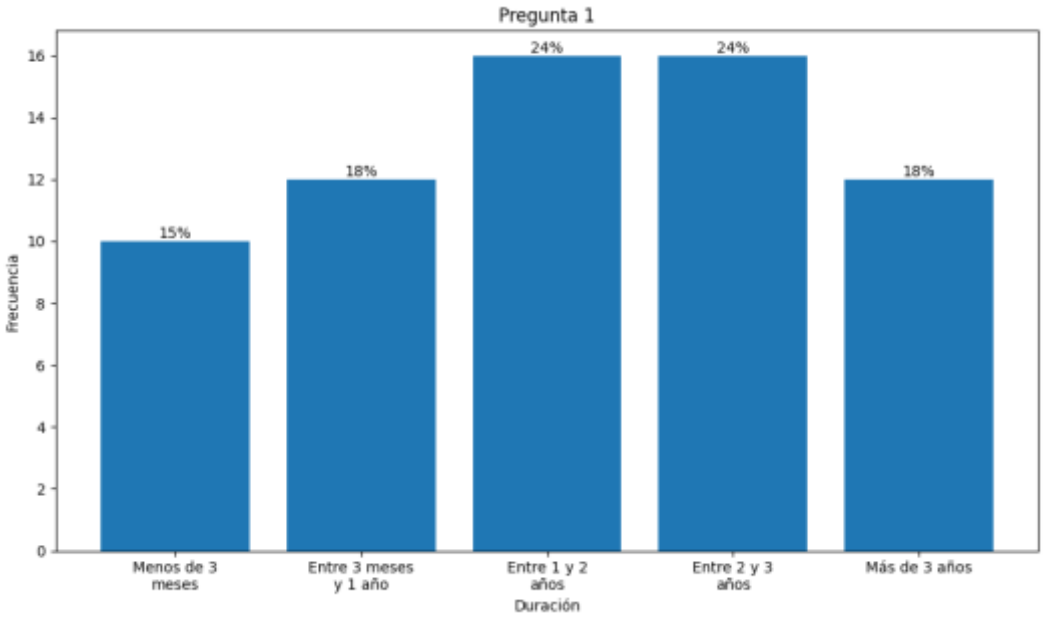


Figura 3. Resultado de la pregunta 1

Análisis e interpretación de resultados

Según los resultados presentados en la **Figura 3**, muestran la distribución de pacientes según su antigüedad en el consultorio. Se puede concluir que un paciente con más de un año de visitas se considera antiguo. La mayoría de los encuestados son pacientes antiguos que continúan acudiendo y recibiendo tratamiento en el consultorio médico, representando el 66% del total. No obstante, se evidencia que el 33% son pacientes nuevos que realizan sus chequeos desde hace menos de un año.

Estos datos indican que el consultorio no solo es capaz de atraer nuevos pacientes, sino también de mantener una relación continua con ellos a lo largo del tiempo, lo cual es un indicador positivo de la calidad y la satisfacción con los servicios ofrecidos.

Pregunta 2: ¿Su médico tratante revisa su historial médico completo durante las consultas?

Tabla 9. Resultado de la pregunta 2

Indicador	Frecuencia	Porcentaje
Siempre	18	27%
Casi Siempre	25	38%
A veces	15	23%
Casi nunca	5	8%
Nunca	3	5%
Total	66	100%

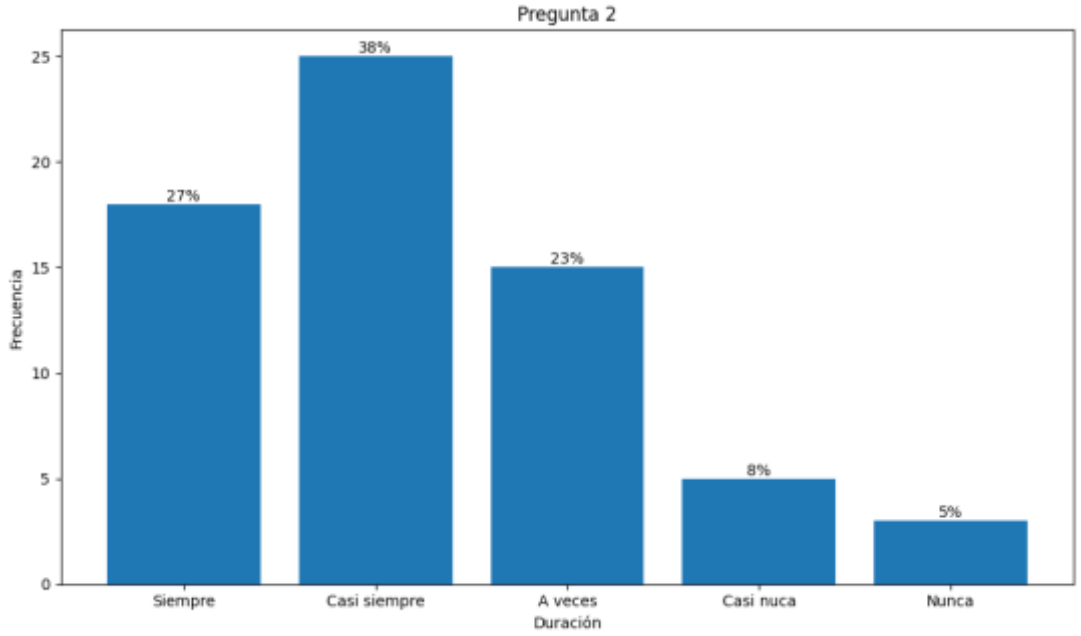


Figura 4. Resultado de la pregunta 2

Análisis e interpretación de resultados

Con los resultados obtenidos en la **Figura 4**, indican que una buena parte de los encuestados, con un total del 65% sumando las respuestas de "siempre" y "casi siempre", observan que el médico revisa el historial médico de manera constante durante las consultas, lo cual es un indicativo positivo de la calidad de atención brindada. Sin embargo, el hecho de la otra parte restante de los encuestados afirme que el médico revisa su historial médico con menos frecuencia o no lo revisa en absoluto, indica un área que se debe mejorar. Es crucial que el médico consulte el historial médico de todos sus pacientes para proporcionar una atención integral, asegurando que se consideren todos los aspectos de la salud del paciente en cada consulta.

Pregunta 3: ¿Cuánto tiempo le toma a su médico encontrar y revisar su historial médico durante las consultas?

Tabla 10. Resultado de la pregunta 3

Indicador	Frecuencia	Porcentaje
Menos de 2 minutos	12	18%
Entre 2 y 5 minutos	15	23%
Entre 5 y 10 minutos	20	30%
Entre 10 y 15 minutos	16	24%
Más de 15 minutos	3	5%
Total	66	100%

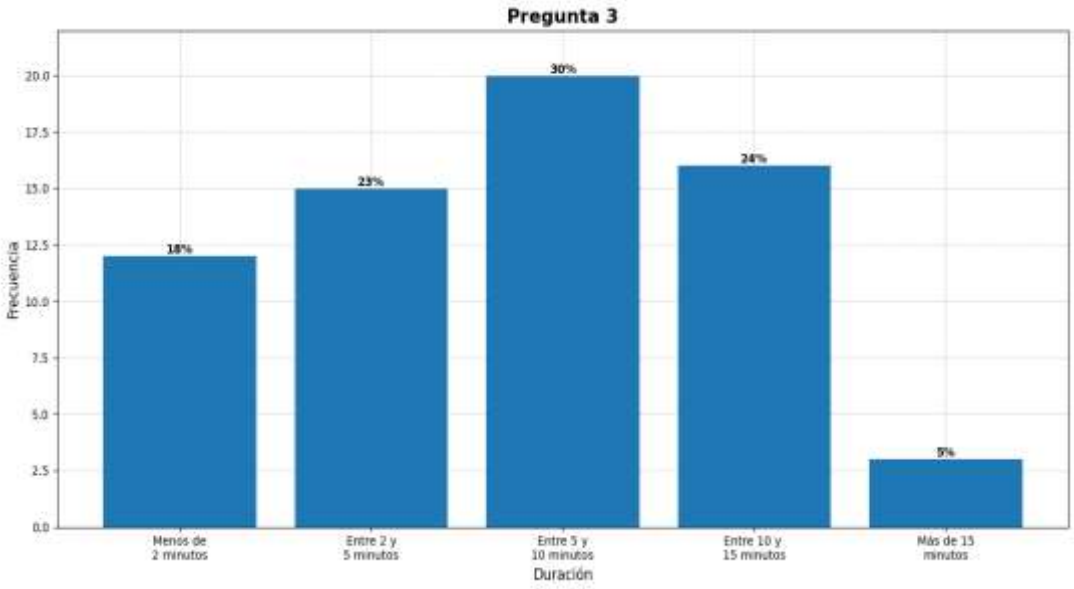


Figura 5. Resultado de la pregunta 3

Análisis e interpretación de resultados

Los resultados presentados en la **Figura 5**, indican que el médico requiere un tiempo considerable para acceder y revisar los historiales médicos durante las consultas. Estableciendo que tarda más de 5 minutos para esta tarea, se evidencia una oportunidad significativa para optimizar este proceso. La implementación de sistemas más eficientes de gestión de registros médicos podría reducir estos tiempos, mejorando la productividad de las consultas y potencialmente aumentando el tiempo de atención directa al paciente.

Pregunta 4: ¿Cómo calificaría la eficiencia del proceso de registro manual de su ficha médica en el consultorio?

Tabla 11. Resultado de la pregunta 4

Indicador	Frecuencia	Porcentaje
Muy eficiente	3	5%
Eficiente	7	11%
Ni eficiente ni deficiente	10	15%
Poco eficiente	30	45%
Nada eficiente	16	24%
Total	66	100%

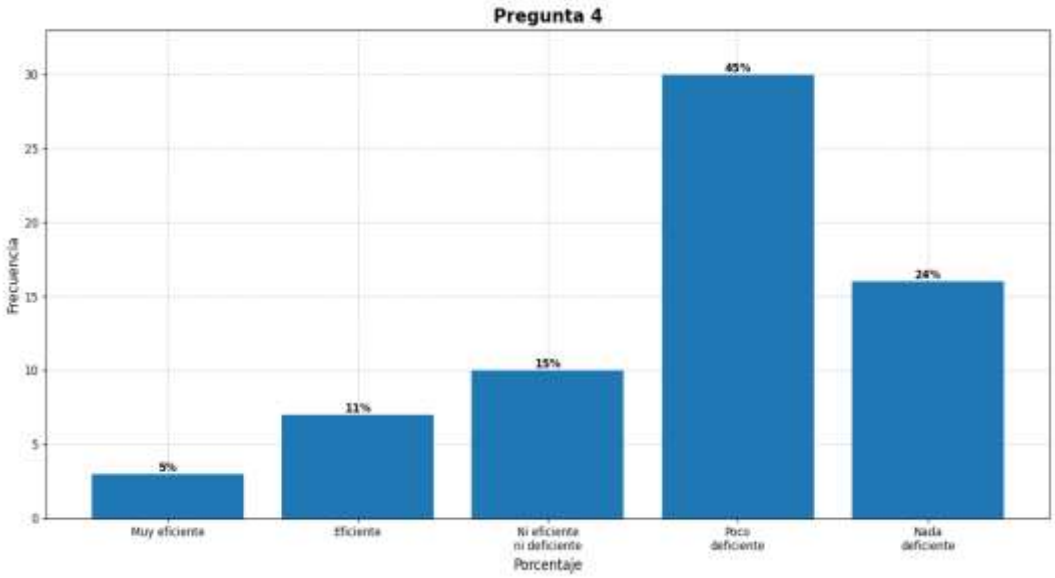


Figura 6. Resultado de la pregunta 4

Análisis e interpretación de resultados

Los resultados de la **Figura 6**, dejan como evidencia que el 69% de opiniones negativas sobre el registro manual de fichas médicas muestra una gran insatisfacción. Esto indica una necesidad urgente de mejorar este proceso. La percepción de ineficiencia sugiere que implementar sistemas más modernos y automatizados para manejar los registros médicos podría mejorar mucho la satisfacción de pacientes y profesionales de la salud. Así, se podría mejorar la calidad y eficiencia de la atención médica en general.

Pregunta 5: ¿Cuánto tiempo suele tardar el proceso de registro de su información médica en el consultorio?

Tabla 12. Resultado de la pregunta 5

Indicador	Frecuencia	Porcentaje
Menos de 5 minutos	15	23%
Entre 5 y 10 minutos	18	27%
Entre 10 y 15 minutos	20	30%
Entre 15 y 20 minutos	10	15%
Más de 20 minutos	3	5%
Total	66	100%

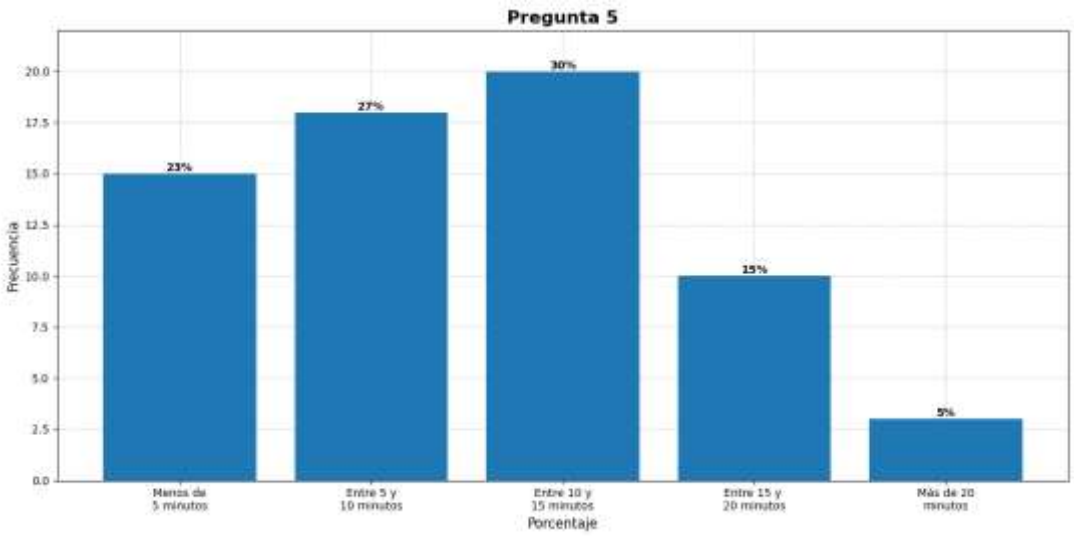


Figura 7. Resultado de la pregunta 5

Análisis e interpretación de resultados

Según los resultados evidenciados en la **Figura 7**, se observa que la diversidad en los tiempos de registro sugiere una falta de estandarización en el proceso. Aunque la mitad de los casos se resuelven en 10 minutos o menos, existe un margen significativo para mejorar la eficiencia, especialmente considerando que el 50% de los registros toman más de 10 minutos. Esta variabilidad indica que la implementación de sistemas más ágiles y uniformes podría reducir los tiempos de espera y mejorar la experiencia tanto de pacientes como de personal médico. La optimización de este proceso podría llevar a una mayor satisfacción y eficiencia en la atención médica.

Pregunta 6: ¿Qué tan satisfecho está con el proceso actual de registro y manejo de su ficha médica y posología (dosis y frecuencia de los medicamentos) en el consultorio?

Tabla 13. Resultado de la pregunta 6

Indicador	Frecuencia	Porcentaje
Muy satisfecho	15	23%
Satisfecho	30	45%
Ni satisfecho ni insatisfecho	15	23%
Insatisfecho	5	8%
Muy insatisfecho	1	2%
Total	66	100%

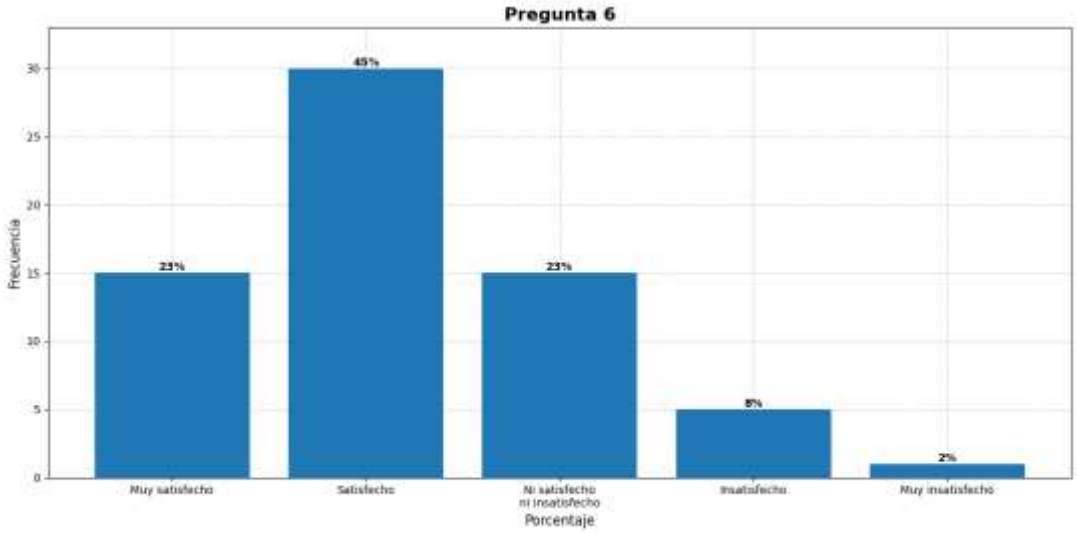


Figura 8. Resultado de la pregunta 6

Análisis e interpretación de resultados

Según los resultados presentados en la **Figura 8**, se observa que, a pesar de las ineficiencias percibidas en el proceso de registro manual y los tiempos variables de registro, la satisfacción general con el manejo de fichas médicas y posología es sorprendentemente alta. Con un 68% de opiniones positivas, parece que los pacientes valoran otros aspectos del servicio que compensan las posibles deficiencias en eficiencia. Sin embargo, el margen de mejora sigue siendo significativo, ya que un tercio de los encuestados no está plenamente satisfecho. Esta discrepancia entre la eficiencia percibida y la satisfacción general sugiere que, aunque los pacientes están mayormente contentos, podrían beneficiarse aún más de mejoras en el sistema de registro y manejo de información médica, potencialmente aumentando los niveles de satisfacción a cifras aún más altas.

Pregunta 7: ¿Ha tenido alguna vez problemas o confusiones con las indicaciones de la posología (dosis y frecuencia de los medicamentos) dadas por su médico?

Tabla 14. Resultado de la pregunta 7

Indicador	Frecuencia	Porcentaje
Siempre	12	18%
Frecuentemente	18	27%
A veces	25	38%
Casi nunca	8	12%
Nunca	3	5%
Total	66	100%

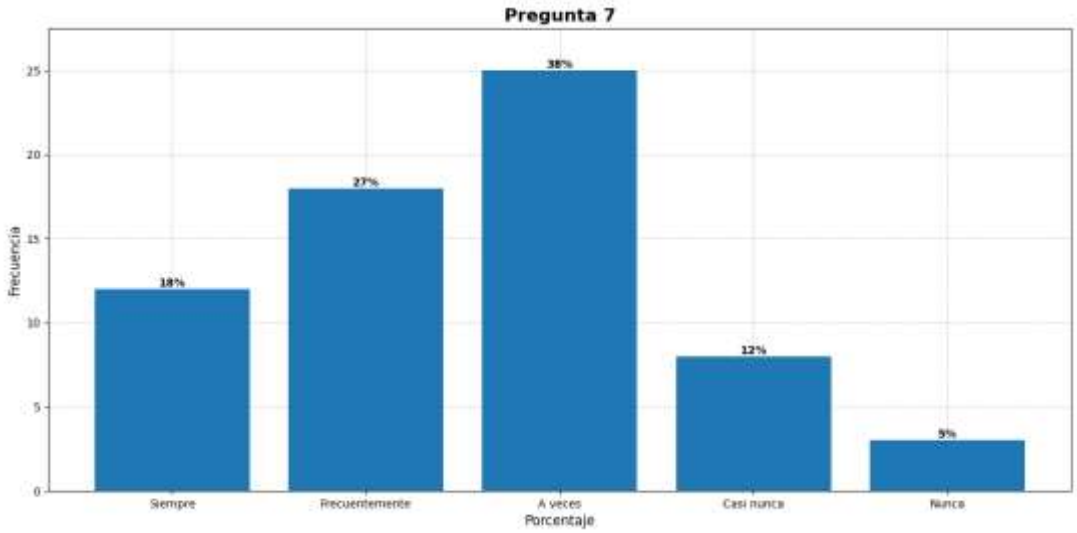


Figura 9. Resultado de la pregunta 7

Análisis e interpretación de resultados

Según los resultados presentados en la **Figura 9**, se observa que la alta incidencia de problemas y confusiones con las indicaciones de posología es un hallazgo crítico que requiere atención inmediata. Con un 83% de pacientes experimentando dificultades en diversos grados, se evidencia un riesgo significativo para la seguridad del paciente y la efectividad del tratamiento. Esta situación subraya la necesidad urgente de mejorar la comunicación entre médicos y pacientes, así como de implementar sistemas más claros y efectivos para transmitir las instrucciones de medicación. Lo que permite concluir que la gran mayoría de los pacientes tiene dificultades para entender las instrucciones de dosificación.

Pregunta 8: En relación con la prescripción de medicamentos, ¿cuánto tiempo le toma a su médico buscar y registrar la posología (dosis y frecuencia de los medicamentos) adecuada para su tratamiento?

Tabla 15. Resultado de la pregunta 8

Indicador	Frecuencia	Porcentaje
Menos de 2 minutos	10	15%
Entre 2 y 5 minutos	45	68%
Entre 5 y 10 minutos	7	11%
Entre 10 y 15 minutos	3	5%
Más de 15 minutos	1	2%
Total	66	100%

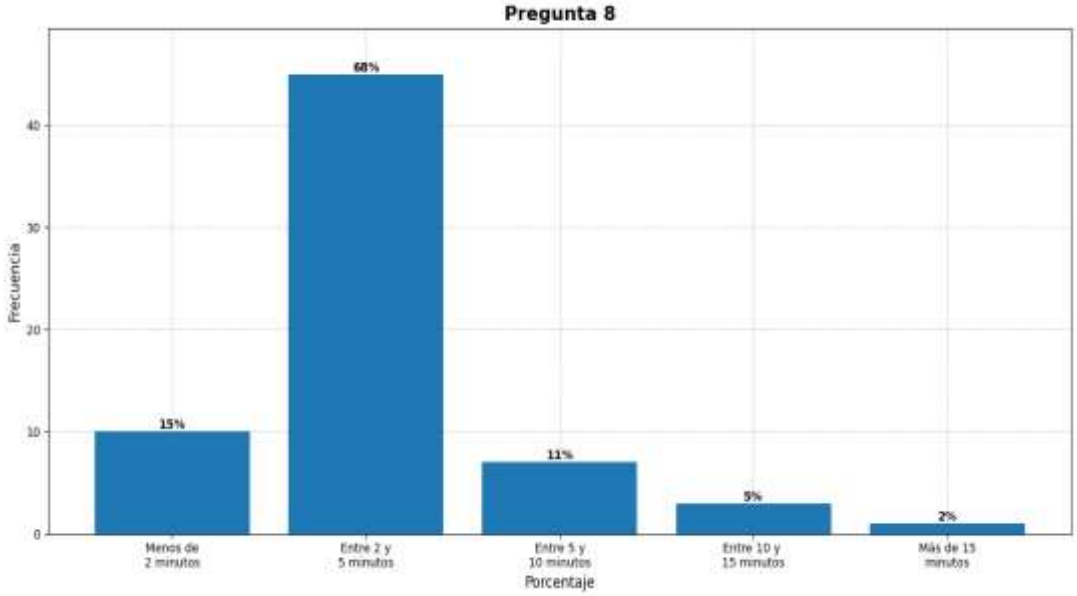


Figura 10. Resultado de la pregunta 8

Análisis e interpretación de resultados

Los resultados de la **Figura 10**, dejan en evidencia que la distribución de tiempos para buscar y registrar la posología sugiere un proceso relativamente eficiente para la mayoría de los médicos. Con el 83% de los casos resueltos en 5 minutos o menos, parece que los profesionales de la salud generalmente tienen un buen manejo de esta tarea. Sin embargo, existe un margen de mejora para el 18% que tarda más de 5 minutos. La implementación de sistemas de apoyo a la decisión clínica o bases de datos de fácil acceso podría potencialmente reducir aún más estos tiempos, especialmente para los casos que superen los 5 minutos. Optimizar este proceso no solo podría

mejorar la eficiencia de las consultas, sino también reducir la probabilidad de errores en la prescripción, contribuyendo así a una atención médica más segura y efectiva.

Pregunta 9: ¿Considera que el uso de la aplicación web para el registro de la ficha médica y control de la posología (dosis y frecuencia de los medicamentos) mejoraría la organización y eficiencia en el consultorio médico?

Tabla 16. Resultado de la pregunta 9

Indicador	Frecuencia	Porcentaje
Totalmente de acuerdo	20	30%
De acuerdo	30	45%
Ni de acuerdo ni en desacuerdo	10	15%
En desacuerdo	5	8%
Totalmente en desacuerdo	1	2%
Total	66	100%

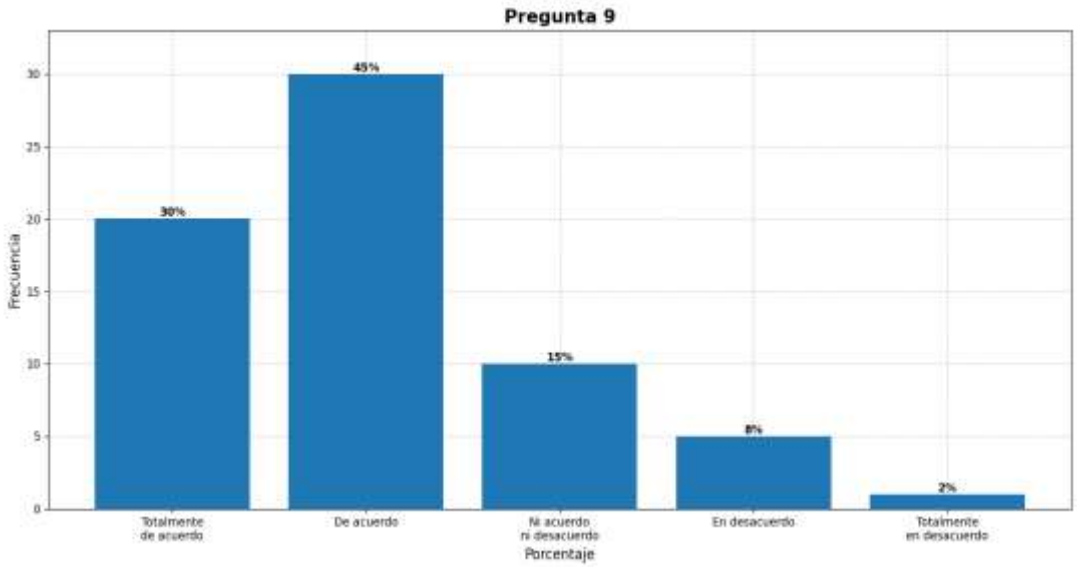


Figura 11. Resultado de la pregunta 9

Análisis e interpretación de resultados

Los datos reflejados en la **Figura 11** son contundentes, la alta aceptación de una aplicación web para el manejo de fichas médicas y control de posología, con un 75% de opiniones favorables, indica un contexto favorable para la digitalización en la atención médica. Solo un 10% de opiniones desfavorables sugiere pocos obstáculos para su adopción. La implementación de esta tecnología puede mejorar la organización y eficiencia en los consultorios, reducir errores y ahorrar tiempo.

Pregunta 10: ¿Cree que una aplicación web facilitaría la comunicación y seguimiento con su médico tratante con relación a su ficha médica y posología (dosis y frecuencia de los medicamentos)?

Tabla 17. Resultado de la pregunta 10

Indicador	Frecuencia	Porcentaje
Totalmente de acuerdo	25	38%
De acuerdo	30	45%
Ni de acuerdo ni en desacuerdo	8	12%
En desacuerdo	2	3%
Totalmente en desacuerdo	1	2%
Total	66	100%

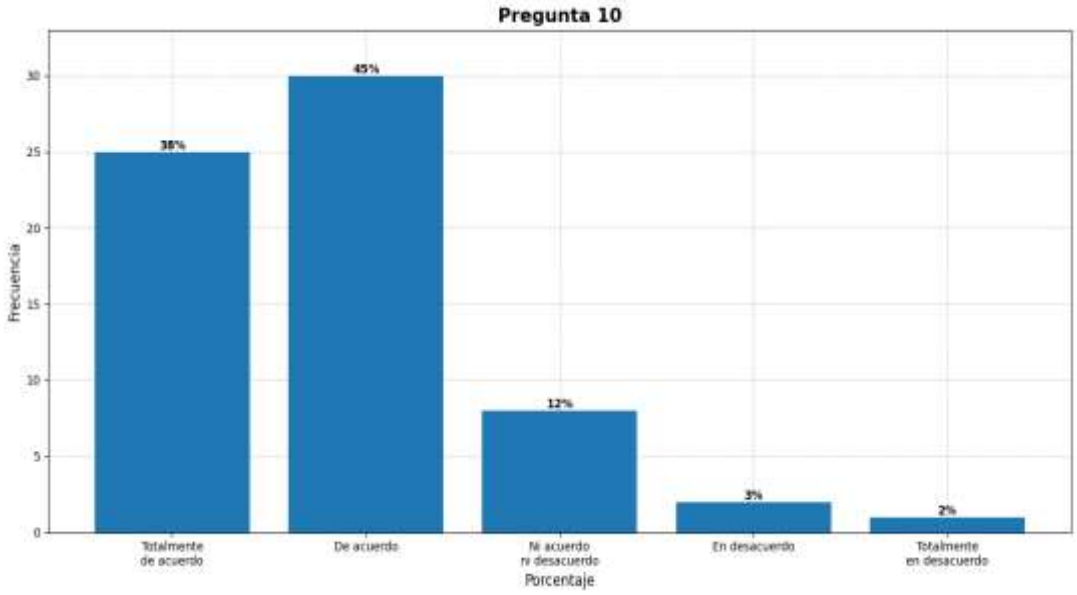


Figura 12. Resultado de la pregunta 10

Análisis e interpretación de resultados

Los resultados de la **Figura 12** son claros, la aceptación del uso de una aplicación web para la comunicación y seguimiento médico, con un 83% de opiniones favorables, indica una fuerte disposición hacia la tecnología en la atención médica. Pero solo el 5% de los encuestados no ven beneficio alguno sobre esta propuesta. No obstante, es crucial abordar las necesidades del 17% que no está completamente convencido, garantizando que la aplicación sea accesible, segura y fácil de usar.

Pregunta 11: ¿Considera que el envío de la receta médica por correo electrónico facilitaría el seguimiento y cumplimiento de su tratamiento, cumpliendo con todas las normas y leyes legales para que sea una receta válida?

Tabla 18. Resultado de la pregunta 11

Indicador	Frecuencia	Porcentaje
Totalmente de acuerdo	18	27%
De acuerdo	25	38%
Ni de acuerdo ni en desacuerdo	15	23%
En desacuerdo	5	8%
Totalmente en desacuerdo	3	5%
Total	66	100%

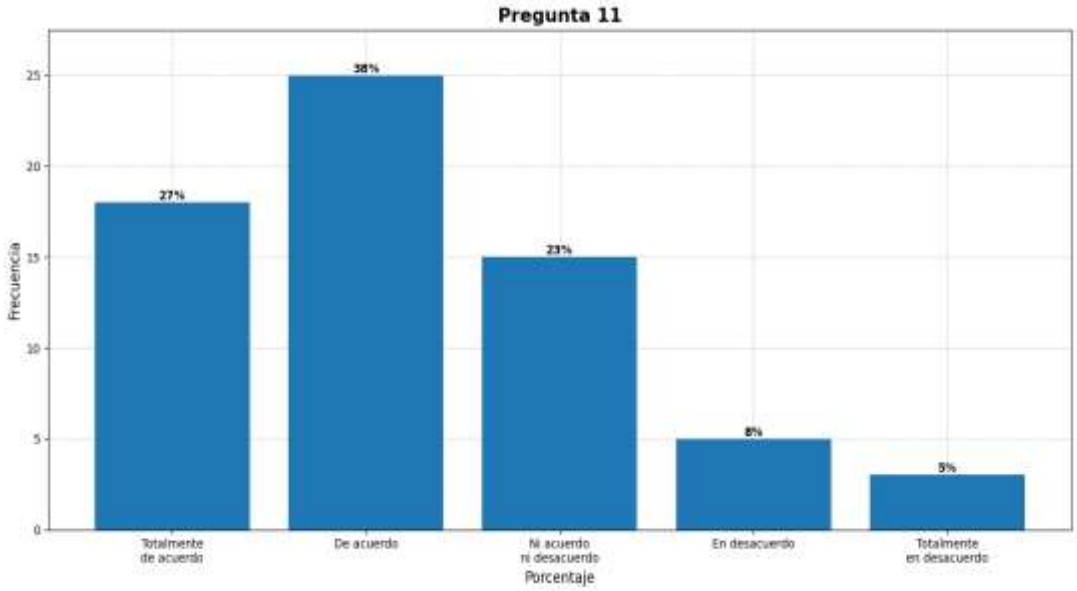


Figura 13. Resultado de la pregunta 11

Análisis e interpretación de resultados

Los resultados presentados en la **Figura 13**, muestran que un 65% está de acuerdo o totalmente de acuerdo con que el envío de la receta médica sea por medio de correo electrónico facilitando el seguimiento y cumplimiento del tratamiento, siempre que se cumplan todas las normas y leyes legales para que sea una receta válida. Por lo que se concluye que existe un amplio respaldo a la idea de enviar recetas médicas por correo electrónico, siempre que se garantice su validez legal.

Pregunta 12: En base a su experiencia general con el Consultorio Médico "DuoMedic", ¿qué tan probable es que lo recomiende a un familiar o amigo?

Tabla 19. Resultado de la pregunta 12

Indicador	Frecuencia	Porcentaje
Definitivamente lo recomendaría	17	26%
Probablemente lo recomendaría	25	38%
Neutral	15	23%
Probablemente no lo recomendaría	6	9%
Definitivamente no lo recomendaría	3	5%
Total	66	100%

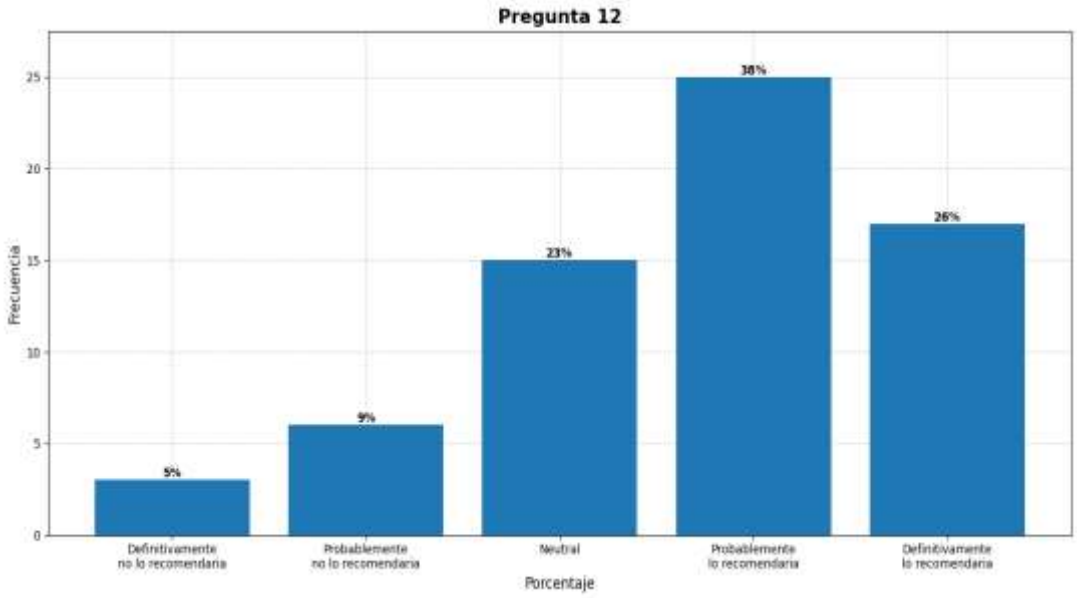


Figura 14. Resultado de la pregunta 12

Análisis e interpretación de resultados

Según los datos presentados en la **Figura 14**, se puede observar que la mayoría de los encuetados recomendarían el Consultorio Médico "DuoMedic" a un familiar o amigo. Proporcionando una base sólida para el crecimiento y la reputación del consultorio. Esta disposición positiva refleja una alta satisfacción con los servicios prestados y sugiere una experiencia de atención al paciente de calidad.

c. Resultados de la entrevista aplicada al médico del consultorio

Los resultados de le entrevista realizada al profesional de la salud del consultorio médico "DuoMedic" son los siguientes:

Tabla 20. Resultados de la entrevista

Nombre del entrevistado: Dr. Cesar Guerrero **Empresa:** Consultorio Médico "DuoMedic" **Cargo:** Médico General **Entrevistador:** José Villacrés			
Objetivo: Identificar las dificultades que enfrenta el médico personal en el proceso de registro manual de fichas médicas y control de la posología, además de conocer su disposición a adoptar una aplicación web que facilite el acceso a la información clínica completa de los pacientes y el envío digital de recetas médicas.			
N°	**Pregunta**	**Respuesta**	**Observación**
1	¿Podría describir el proceso actual que sigue para el registro de la ficha médica de un paciente?	El proceso inicia con la recepción del paciente y la recopilación de su información básica. Luego, verificamos si es un paciente nuevo o existente. Si es nuevo, creamos una ficha médica física con sus datos personales y antecedentes médicos relevantes. Si es un paciente existente, consultamos su ficha médica previa. Después de la consulta médica, registramos los hallazgos, diagnósticos y tratamientos en la ficha.	El proceso actual es manual y con fichas físicas, lo que puede generar demoras, desorden y riesgo de pérdida de información.
2	¿Podría describir el proceso actual que sigue para la prescripción de medicamentos a los pacientes?	Después de evaluar al paciente y determinar el tratamiento necesario, prescribimos los medicamentos de forma manual en una receta física, detallando el nombre del medicamento, dosis, frecuencia y duración del tratamiento. Explicamos las indicaciones al paciente y le entregamos la receta impresa.	El proceso manual de prescripción puede ser propenso a errores, y la receta física puede extraviarse o ser difícil de leer para el paciente.
3	¿Ha tenido alguna vez problemas o pérdidas de información debido al manejo manual de las fichas médicas y recetas de pacientes?	Sí, lamentablemente hemos tenido casos en los que se han extraviado o dañado fichas médicas físicas, o se han perdido recetas de pacientes. Esto genera inconvenientes y retrasos en la atención médica.	La pérdida de información médica es un riesgo importante que puede afectar la calidad de la atención y generar problemas legales.
4	¿Qué dificultades o problemas ha encontrado con el método actual de registro de la ficha médica?	Una de las principales dificultades es el manejo físico de las fichas médicas. Es complicado almacenar y organizar tantas fichas, y acceder rápidamente a la información cuando se necesita. Además, la información está dispersa y no siempre está actualizada. Otro	El método actual de registro en fichas físicas es poco eficiente, propenso a errores y dificulta el acceso rápido a la información médica completa del paciente.

		problema es la legibilidad de las anotaciones manuscritas, que a veces son difíciles de leer o interpretar.	
5	¿Utiliza alguna aplicación informática para gestionar la información clínica de los pacientes?	Actualmente no utilizamos ninguna aplicación informática para gestionar las fichas médicas ni la información clínica de los pacientes. Todo nuestro registro y manejo de datos es de forma manual y en papel.	La falta de un sistema digital representa una oportunidad de mejora significativa para optimizar los procesos y brindar una atención más eficiente y segura.
6	¿Cómo maneja actualmente el control de la posología (dosis y frecuencia de medicamentos) de sus pacientes?	El control de la posología se realiza de forma manual. Cuando prescribimos medicamentos, anotamos las indicaciones de dosis y frecuencia en la receta física que entregamos al paciente. También registramos estas indicaciones en la ficha médica del paciente. Sin embargo, depende del paciente seguir correctamente las instrucciones y no contamos con un sistema de seguimiento o recordatorios.	El manejo manual de la posología implica un riesgo de errores y falta de seguimiento, lo que puede afectar la adherencia y efectividad del tratamiento.
7	¿Ha tenido casos en los que los pacientes han reportado confusiones o problemas al seguir las indicaciones de la posología?	Sí, lamentablemente hemos tenido varios casos en los que los pacientes han reportado confusiones al seguir las indicaciones de la posología. Muchas veces se deben a recetas poco claras, instrucciones verbales mal comprendidas o olvidos por parte del paciente. Esto puede llevar a errores en la dosis o frecuencia de los medicamentos, lo cual puede comprometer la efectividad del tratamiento.	Las confusiones en la posología son un problema recurrente que evidencia la necesidad de implementar un sistema más claro y accesible para los pacientes, que les permita comprender y seguir adecuadamente las indicaciones médicas.
8	¿Cree que el acceso a la ficha médica completa de los pacientes a través de una aplicación web facilitaría su trabajo y la atención brindada?	Definitivamente, contar con el acceso a la ficha médica completa de los pacientes a través de una aplicación web facilitaría enormemente mi trabajo y la atención brindada. Tener toda la información médica del paciente disponible de forma organizada y actualizada me permitiría brindar una atención más eficiente, precisa y segura. Además, agilizaría los procesos de registro y consulta de datos.	Una aplicación web que centralice y organice las fichas médicas de los pacientes representaría un avance significativo en la calidad y eficiencia de la atención médica brindada.
9	Desde su punto de vista como médico, ¿qué tipo de información le gustaría poder acceder fácilmente a través de la aplicación mientras está interactuando con un paciente?	Me sería muy útil poder acceder fácilmente a través de la aplicación a información como el historial médico completo del paciente, incluyendo antecedentes, diagnósticos previos, tratamientos, alergias, exámenes de laboratorio e imágenes médicas. También sería valioso contar con un registro detallado de las consultas previas,	Poder acceder de manera rápida y completa al historial médico del paciente a través de la aplicación mejoraría significativamente la calidad de la atención brindada y la toma de decisiones médicas.

		evolución de la condición del paciente y medicamentos prescritos anteriormente. Toda esta información es crucial para brindar una atención integral y tomar decisiones médicas adecuadas.	
10	¿Estaría dispuesto a utilizar una aplicación web para el registro y actualización de las fichas médicas de sus pacientes?	Sí, definitivamente estaría dispuesto a utilizar una aplicación web para el registro y actualización de las fichas médicas de mis pacientes. Considero que sería una herramienta muy valiosa y necesaria para optimizar nuestros procesos y brindar una atención médica más eficiente y segura. Contar con un sistema digital y centralizado para manejar la información clínica de los pacientes representaría un gran avance en comparación con el método actual de registros físicos.	La disposición del personal médico a adoptar una aplicación web para el manejo de las fichas médicas es fundamental para el éxito de su implementación. Esta respuesta positiva demuestra la apertura al cambio y la mejora en los procesos de atención médica.
11	¿Le resultaría útil poder enviar las recetas médicas de sus pacientes por correo electrónico a través de la aplicación web?	Sí, me resultaría extremadamente útil poder enviar las recetas médicas de mis pacientes por medios digitales a través de la aplicación web. Esto agilizaría y simplificaría el proceso de entrega de recetas, evitando problemas como extravíos, recetas ilegibles o confusiones por parte de los pacientes. Además, permitiría un mejor seguimiento y control de las prescripciones médicas realizadas.	La capacidad de enviar recetas médicas por medios digitales seguros y confiables representaría una mejora sustancial en la eficiencia y calidad del proceso de prescripción médica, beneficiando tanto al personal médico como a los pacientes.
12	¿Cree que el envío digital de recetas médicas válidas y legales podría mejorar el seguimiento y cumplimiento del tratamiento por parte de los pacientes?	Sí, estoy convencido de que el envío digital de recetas médicas válidas y legales podría mejorar significativamente el seguimiento y cumplimiento del tratamiento por parte de los pacientes. Al recibir las recetas de forma digital y clara, los pacientes tendrían menos dudas sobre las indicaciones y podrían acceder fácilmente a la información en cualquier momento. Además, un sistema digital podría incluir recordatorios y seguimiento de la adherencia al tratamiento, lo cual es crucial para su efectividad.	La mejora en el seguimiento y cumplimiento del tratamiento gracias al envío digital de recetas médicas tendría un impacto directo y positivo en la calidad de la atención médica y los resultados de salud de los pacientes.

Conclusión:

Según los resultados de la entrevista el personal médico reconoce las limitaciones y problemas del método actual de registro manual de fichas médicas y prescripción de medicamentos, como riesgo de pérdida de información, falta de organización, confusiones en la posología de los pacientes y acceso limitado a los datos clínicos. Sin embargo, existe una disposición positiva hacia la implementación de una aplicación web que permita el registro digital y acceso completo a las fichas médicas, incluyendo el historial clínico detallado, así como el envío de recetas por medios digitales seguros, lo cual facilitaría su trabajo, agilizaría procesos, mejoraría el seguimiento de los tratamientos y brindaría una atención más eficiente, precisa y segura para los pacientes.

2.2.4 Procesamiento y análisis de datos

Conforme a los resultados obtenidos a través de los instrumentos de recolección de información en el consultorio médico “DuoMedic”, se concluye lo siguiente:

- El proceso de registro y manejo del historial médico se realiza de manera manual con fichas físicas, lo cual provoca demoras, desorganización y un alto riesgo de pérdida de información. Esta metodología es ineficiente y propensa a errores, afectando la calidad de la atención médica.

- La prescripción de medicamentos también se lleva a cabo manualmente, utilizando recetas físicas. Este método es propenso a errores de legibilidad y a la pérdida de recetas, comprometiendo el seguimiento del tratamiento y la seguridad del paciente

- Existen problemas con el manejo manual de las fichas y recetas, incluyendo la pérdida y el daño de estos documentos. Esto no solo causa inconvenientes y retrasos en la atención, sino que también pone en riesgo la integridad de la información médica de los pacientes.

- El almacenamiento y la organización de las fichas médicas físicas presentan dificultades, especialmente en el acceso rápido y eficiente a la información necesaria. Esta situación genera retrasos y afecta negativamente la eficiencia del consultorio.

- El personal médico está de acuerdo en que el proceso actual es inadecuado y no satisface las necesidades del consultorio. Coinciden en la necesidad de adoptar soluciones más eficientes y tecnológicas.

- El personal médico está de acuerdo en que la implementación de una aplicación web para la gestión de fichas médicas y recetas mejoraría significativamente la eficiencia, la precisión y la seguridad de los procesos del consultorio. Esta solución tecnológica podría resolver muchos de los problemas actuales asociados con el manejo manual de la información.

CAPÍTULO III. RESULTADOS Y DISCUSIÓN

3.1 Análisis y discusión de los resultados

En el capítulo II del presente trabajo de investigación, se realizó la recopilación de la información con ayuda de las diversas técnicas y herramientas de recolección de datos. Con la información obtenida, se detectaron varias necesidades que se pueden solucionar. Fue necesario analizar el proceso, las tecnologías y la metodología que se utilizarían para el desarrollo del proyecto.

3.1.1 Proceso y descripción del registro de ficha, consulta y prescripción médica.

Una vez recogida la información de cómo se lleva a cabo los procesos del registro y prescripción médica, se presentan diagramas de cada proceso involucrado y la interacción del médico con el paciente.

a. Proceso actual

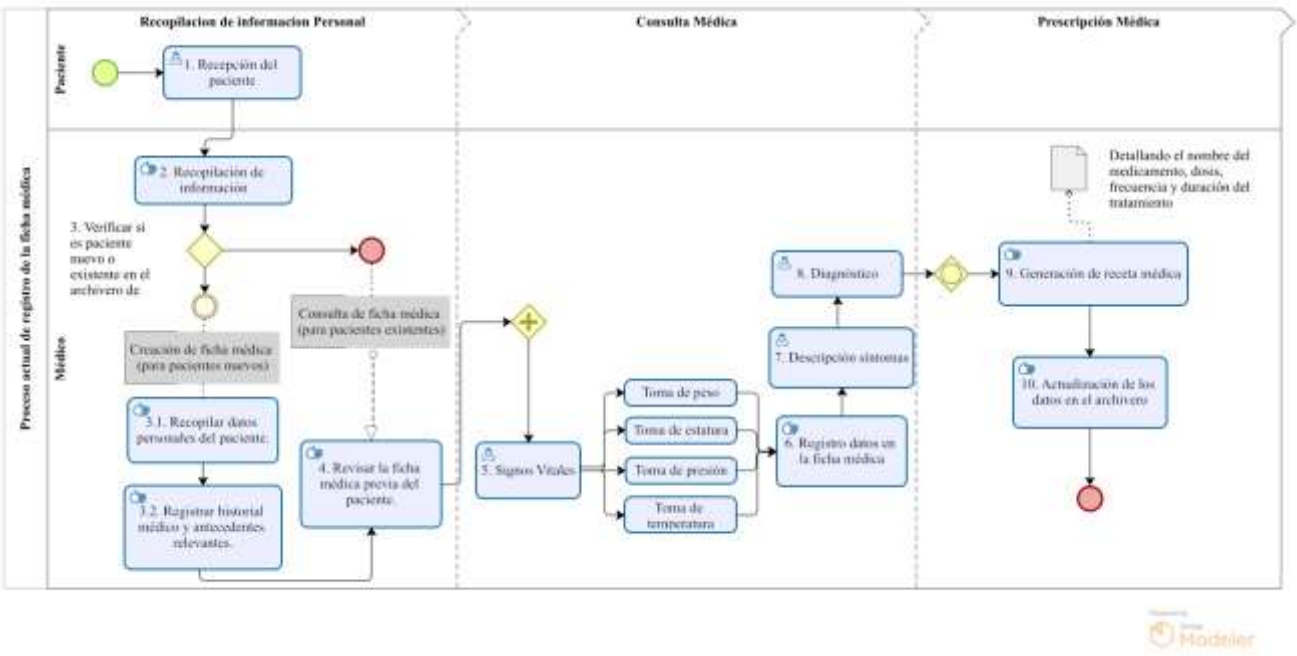


Figura 15. Proceso actual del registro y prescripción médica

b. Descripción del proceso actual

Como se observa en la **Figura 15**, el proceso actual que realiza el consultorio médico para el registro de la ficha y prescripción médica se detallan a continuación:

Tabla 21. Descripción del proceso actual del registro de ficha, consulta y prescripción médica

N°	Tarea	Responsable	Descripción
1	Recepción del paciente	Paciente	El paciente llega al consultorio médico y se presenta en la recepción para su consulta.
2	Recopilación de información	Médico	El personal solicita al paciente información básica como cédula de identidad y nombre complete para la búsqueda del paciente.
3.	Verificar si es paciente nuevo o existente	Médico	Utilizando la información proporcionada en la tarea 2, especialmente la cédula, se verifica en el archivero de pacientes si el paciente ya tiene un historial en el centro médico o si es su primera visita.
3.1	Recopilar datos personales del paciente (para pacientes nuevos)	Médico	Para pacientes nuevos, se llena un formulario de manera manual detallando la información personal del paciente.
3.2	Registrador historial médico y antecedentes relevantes (para pacientes nuevos)	Médico	Se registran detalles de manera manual sobre cirugías previas, medicamentos actuales, historial familiar de enfermedades y cualquier otra información médica relevante proporcionada por el paciente nuevo.
4	Revisar la ficha médica previa del paciente	Médico	Para pacientes existentes, el médico revisa la ficha médica anterior, incluyendo consultas previas, tratamientos anteriores.
5	Signos Vitales	Médico	Se toman y registran los signos vitales del paciente: peso en una báscula, estatura con un estadiómetro, presión arterial con un tensiómetro y temperatura corporal con un termómetro.
6	Registro de datos en la ficha médica	Médico	Los signos vitales medidos se anotan en la ficha médica del paciente, junto con la fecha y hora de la toma de datos.
7	Descripción de los síntomas	Médico	El médico escucha atentamente al paciente describir sus síntomas, hace preguntas adicionales para aclarar y registrar esta información en la ficha médica.
8	Diagnostico	Médico	Basándose en los síntomas, signos vitales, historial médico y posiblemente un examen físico, el médico realiza un diagnóstico preliminar o solicita exámenes adicionales si es necesario.
9	Generación de receta médica	Médico	Si se requiere tratamiento, el médico escribe una receta médica detallando el nombre del medicamento, dosis, frecuencia y duración del tratamiento. También puede incluir recomendaciones adicionales.
10	Actualización de los datos en el archivo	Médico	Toda la información nueva generada durante la consulta (diagnóstico, tratamiento, recetas)

			se agrega al expediente del paciente para futuras referencias.

c. Proceso optimizado

Una vez analizado el proceso de cómo se realiza el registro de ficha y prescripción médica, se constató que, de la forma tradicional utilizando archivos físicos, formularios y registros manuales es un poco lento y propenso a errores. Por ejemplo, la búsqueda de pacientes entre numerosos archivos puede llevar mucho tiempo, especialmente si no están organizados adecuadamente. Además, el registro manual de la información es más lento y susceptible a errores, mientras que las prescripciones médicas escritas a mano pueden generar confusiones en la interpretación y dificultades en la dispensación de medicamentos. Por ende, para solucionar estos problemas se crucial migrar el proceso manual a un sistema digital, permitiendo optimizar y reducir el tiempo de cada proceso.

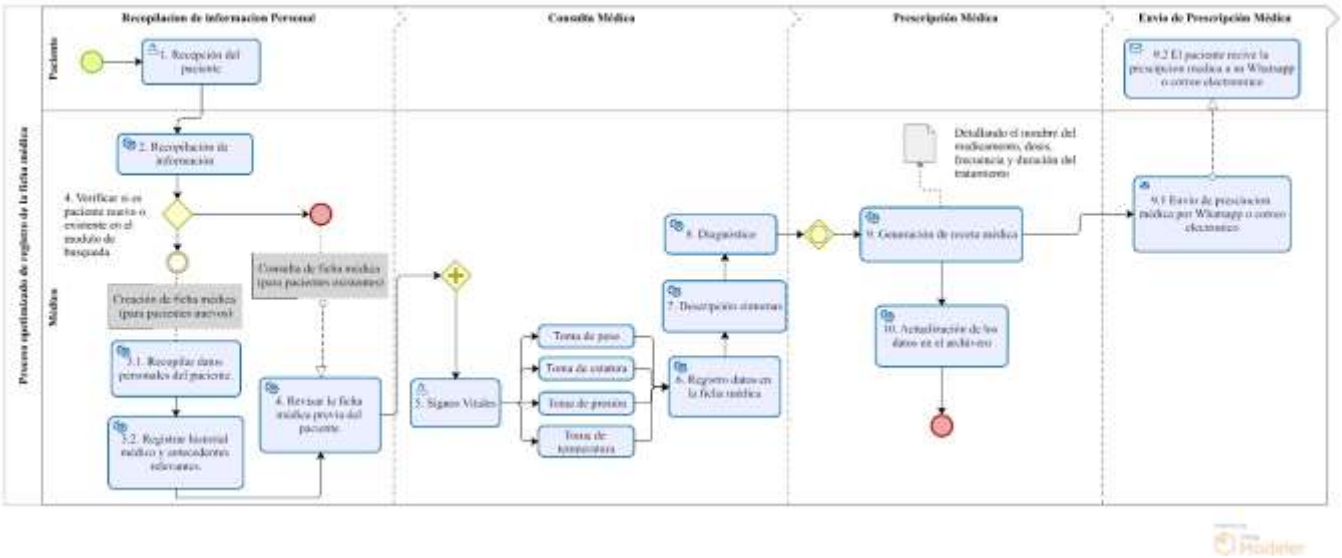


Figura 16. Proceso optimizado del registro y prescripción médica

d. Descripción del proceso optimizado

Como se observa en la **Figura 16**, se evidencian claramente los nuevos cambios implementados en las actividades del consultorio médico para mejorar el proceso de registro de ficha y prescripción médica. A continuación, se detallan estos cambios:

Tabla 22. Descripción del proceso optimizado del registro de ficha, consulta y prescripción médica

N°	Tarea	Responsable	Descripción
1	Recepción del paciente	Paciente	El paciente llega al consultorio médico y se presenta en la recepción para su consulta.
2	Recopilación de información	Médico	El personal solicita al paciente información básica como cédula de identidad y nombre complete para la búsqueda del paciente. La búsqueda del paciente se lo realiza en el módulo de búsqueda del paciente.
3.	Verificar si es paciente nuevo o existente	Sistema	El módulo de búsqueda verifica automáticamente si el paciente ya existe en la base de datos o si es nuevo.
3.1	Recopilar datos personales del paciente (para pacientes nuevos)	Médico	Para pacientes nuevos, el médico ingresar datos personales del paciente detallados en un formulario digital.
3.2	Registrador historial médico y antecedentes relevantes (para pacientes nuevos)	Médico	El sistema facilita un formulario para el registro sobre cirugías previas, medicamentos actuales, historial familiar de enfermedades y cualquier otra información médica relevante proporcionada por el paciente nuevo.
4	Revisar la ficha médica previa del paciente	Médico	El médico accede y revisa el historial médico en el módulo de fichas médicas del paciente que está haciendo tartar.
5	Signos Vitales	Médico	Se toman los signos vitales del paciente: peso en una báscula, estatura con un estadiómetro, presión arterial con un tensiómetro y temperatura corporal con un termómetro.
6	Registro de datos en la ficha médica	Médico	Los signos vitales medidos se ingresan en la ficha médica del paciente actual en el sistema.
7	Descripción de los síntomas	Médico	El médico escucha atentamente al paciente describir sus síntomas, hace preguntas adicionales para aclarar y registrar esta información en el módulo de la ficha médica.
8	Diagnostico	Médico	Basándose en los síntomas, signos vitales, historial médico y posiblemente un examen físico, el médico realiza un diagnóstico que se registra en el sistema.
9	Generación de receta médica	Médico	El médico genera una receta médica digital, detallando el nombre del médicamento, dosis, frecuencia y duración del tratamiento.
9.1	Envío de prescripción médica	Sistema	El sistema envía automáticamente la receta médica al paciente a través de correo electrónico.
9.2	El paciente recibe la receta médica.	Sistema	El paciente recibe la receta digital en su dispositivo móvil.
10	Actualización de los datos en el archivo	Médico	Toda la información nueva generada durante la consulta (diagnóstico, tratamiento, recetas) se agrega en el sistema para futuras referencias acerca del paciente que hizo la consulta médica.

3.1.2 Desarrollo de aplicaciones web

Las aplicaciones web son muy importantes hoy en día, ya que permiten a las personas acceder a productos y servicios desde cualquier lugar y en cualquier momento. Han cambiado la manera en que interactuamos con el mundo, haciéndolo más fácil y rápido.

Para crear una aplicación web, se requiere una combinación de habilidades técnicas, además de la selección de una plataforma y herramientas necesarias para su construcción. Es importante planificar bien y realizar una buena investigación del mercado. Todo esto ayuda a cumplir con los objetivos del planteados, utilizando una metodología de desarrollo adecuada.

a. IDEs (Entorno de Desarrollo Integrado) para el desarrollo del proyecto

Las plataformas de desarrollo son entornos en los cuales los desarrolladores crean, realizan pruebas y despliegan programas desarrolladas para un propósito en específico. Cada plataforma tiene sus propias herramientas, extensiones, gadgets entre otras ayudas para el desarrollo del proyecto. Los IDEs presentados en la siguiente tabla comparativa son los más utilizados para el desarrollo web, en la cual se realiza una comparativa de sus características. Tiendo en cuenta que las IDEs que se comparar son aquellas que tiene compatibilidad con Python.

Tabla 23. Comparativa de IDEs de desarrollo web con compatibilidad con Python

IDEs	Plataforma	Lenguajes de Programación Compatibles	Características destacadas	Ventajas	Desventajas
Visual Studio	Windows, macOS	C#, VB.NET, F#, C++, Python, JavaScrip	Integración con Azure, herramientas avanzadas de depuración, integración con control de versiones, herramientas de prueba, soporte para desarrollo web.	Gran soporte para desarrollo en .NET, muchas herramientas útiles, excelente integración con bases de datos y sistemas de control de versiones	Requiere muchos recursos del sistema, instalación pesada

Visual Studio Code	Windows, Linux, macOS	JavaScript, Python, TypeScript, Java, C#	Ligero, extensiones para casi cualquier lenguaje, soporte para control de versiones, integración con bases de datos, soporte para desarrollo web.	Muy personalizable, gratuito, ligero, gran comunidad de usuarios	Puede necesitar configuración adicional, menos herramientas nativas que IDEs más completos
IntelliJ IDEA	Windows, Linux, macOS	Java, Kotlin, Groovy, Scala, Python, Ruby	Soporte avanzado para desarrollo en Java, herramientas de refactorización automáticas, integración con control de versiones, herramientas de prueba, soporte para desarrollo web.	Excelente para desarrollo en Java, potente motor de análisis de código, buena integración con bases de datos	Versión completa es de pago, puede ser pesado en recursos

Acorde a la **Tabla 23**, se optó por Visual Studio Code como plataforma de desarrollo principal para el desarrollo del aplicativo web basado en Python, por su comodidad y experiencia previa sobre el uso de la IDE, lo que facilita tener un manejo del software.

b. Frameworks basados en el lenguaje Python

Para definir qué framework es la adecuada para el desarrollo de la aplicación web basada en Python, es importante considerar algunas características que nos permitirán definir factores que ayuden a identificar la adecuada para el desarrollo del proyecto. A continuación, se presenta una tabla comparativa entre los frameworks de Python.

Tabla 24. Comparativa de Frameworks basados en el lenguaje Python

Características	Django	Flask	Tornado
Escalabilidad	Alta, adecuado para aplicaciones grandes y complejas	Media, adecuado para proyectos pequeños a medianos	Alta, ideal para aplicaciones en tiempo real y de gran concurrencia
Curva de aprendizaje	Moderada, debido a su gran cantidad de funcionalidades	Baja, fácil de aprender y usar para principiantes	Moderada, requiere comprensión de programación asincrónica
Modelo	Modelo-View-Template (MVT)	No sigue un modelo específico, microframework	Modelo de programación asincrónica basado en eventos
Compatibilidad con bases de datos	Compatible con PostgreSQL, MySQL, SQLite y Oracle	Extensible con SQLAlchemy y otros ORM	Compatible con bases de datos que soportan operaciones no

			bloqueantes como MongoDB y Redis
Características Destacadas	Interfaz administrativa, ORM, seguridad robusta, soporte para internacionalización	Ligero, extensible con plugins, sistema integrado de pruebas unitarias	Servidor web asíncrono, manejo de miles de conexiones simultáneas, comunicación bidireccional en tiempo real
Ventajas	Promueve desarrollo rápido, enfoque en seguridad, amplia comunidad	Ligero y flexible, fácil de aprender, ideal para personalización	Eficiente en la gestión de conexiones concurrentes, escalabilidad, mejora el rendimiento y la eficiencia
Desventajas	Curva de aprendizaje compleja, más pesado que microframeworks	Menos funcionalidades integradas, menos robusto en seguridad y escalabilidad	La programación asincrónica puede ser difícil de dominar, menos recursos y tutoriales, requiere más configuración inicial

Conforme a la información resumida en la **Tabla 24**, se eligió a Django como framework para el desarrollo, por su robustez y amplia gama de funcionalidades integradas. En definitiva, Django es ideal para aplicaciones web escalables y seguras, lo cual brinda más opciones de crecimiento y mantenibilidad a futuro.

c. Versiones del framework Django

Una vez que se determinó el framework con la que se va a desarrollo del proyecto, se procede a analizar cada una de las versiones del framework considerando sus aspectos técnicos y específicos como lo presenta la siguiente tabla comparativa.

Tabla 25. Comparativa de las versiones del framework Django

Versión	Django 5.0.6	Django 5.0.5	Django 4.2.5	Django 4.2.4
Fecha de lanzamiento	2024-05-07	2024-04-04	2023-12-02	2023-10-05
Detalles	Corrección de errores de empaquetado.	Introducción de nuevas funciones de ORM.	Mejoras en la seguridad y soporte extendido.	Correcciones menores y optimizaciones de rendimiento.
Ventajas	Mayor estabilidad y corrección de errores.	Nuevas funciones en el ORM, mayor flexibilidad.	Mejoras de seguridad, soporte extendido.	Mejor rendimiento y eficiencia.
Desventajas	Actualización menor, sin nuevas características.	Posibles errores iniciales con nuevas funciones.	No incluye nuevas características, solo mejoras.	Menos cambios significativos.

Soporte para plataformas	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows, MacOS, Linux	Windows, MacOS, Linux
Rendimiento	Alto	Alto	Alto	Alto
Curva de aprendizaje	Bueno	Bueno	Moderado	Moderado
Comunidad	Grande	Grande	Grande	Grande
Documentación	Completa y actualizada	Completa y actualizada	Completa y actualizada	Completa y actualizada
Compatibilidad	Alta	Alta	Alta	Alta
Licencia	BSD	BSD	BSD	BSD

Conforme a la información resumida en la **Tabla 25**, se escogió a Django 5.0.6 como la versión del framework para el desarrollo del proyecto, por su mayor estabilidad y las correcciones de errores implementadas. En definitiva, Django 5.0.6 es ideal para aplicaciones que requieren una base sólida y confiable, lo cual brinda más seguridad y robustez en el desarrollo a futuro.

d. Gestores de base de datos

Una base de datos es un sistema de almacenamiento de información, permitiendo realizar acciones como gestionar, ordenar, recuperar y manipular data de manera eficiente. Las bases de datos son fundamentales para todo tipo de los sistemas informáticos como sitios web y aplicaciones.

Para determinar que gestor de base de datos es el adecuado para el desarrollo del proyecto, se procede analizar el gestor adecuado por medio de la siguiente tabla comparativa.

Tabla 26. Comparativa de gestores de Bases de Datos

Gestores de Base de Datos	**MySQL**	**SQLite**	**PostgreSQL**
Modelo de datos	Relacional	Relacional	Relacional
Compatibilidad con SQL	Alta	Buena	Alta
Compatibilidad con Python	Alta (MySQL Connector, PyMySQL)	Alta (sqlite3)	Alta (Psycopg2, SQLAlchemy)
Comunidad	Grande	Grande	Grande
Documentación	Amplia y detallada	Amplia	Amplia
Resistencia a fallos	Alta	Buena	Alta
Seguridad	Alta	Buena	Buena
Usos recomendados	Aplicaciones web, comercio electrónico,	Aplicaciones pequeñas, prototipos, pruebas	Aplicaciones empresariales, análisis de datos, sistemas financieros,

	aplicaciones móviles		aplicaciones web, aplicaciones comerciales
Ventajas	Rendimiento alto, ampliamente soportado, replicación y clustering	Ligero, sin necesidad de servidor, fácil de usar	Altamente extensible, soporte para consultas complejas
Desventajas	Configuración y mantenimiento más complejos, licencias duales	Limitaciones en concurrencia, no adecuado para aplicaciones grandes	Curva de aprendizaje más pronunciada, consumo de recursos

Conforme a la información resumida en la **Tabla 26**, se escogió a PostgreSQL como el sistema de gestión de base de datos para el proyecto en Django, por su alta compatibilidad con Python y su capacidad para manejar consultas complejas. PostgreSQL se destaca por ser altamente extensible y ofrecer un sólido soporte para aplicaciones empresariales y análisis de datos. En definitiva, PostgreSQL es ideal para proyectos Django que requieren escalabilidad, integridad de datos y funcionalidades avanzadas, lo cual brinda una base robusta para el crecimiento y la evolución del proyecto a largo plazo, especialmente en entornos web y aplicaciones comerciales complejas.

e. Metodologías para el desarrollo web

La tabla comparativa a continuación destaca las principales características de cada una de las metodologías ágiles, resaltando su enfoque en la entrega constante y rápida de resultados.

Tabla 27. Comparativa de las metodologías ágiles para desarrollo web

Metodología	**Scrum**	**Kanban**	**XP (Programación Extrema)**
Tamaño de los proyectos	Pequeños a medianos	Cualquier tamaño	Pequeños a medianos
Tamaño de equipo	3-9 personas	Flexible, cualquier tamaño	1-10 personas
Estilo de desarrollo	Iterativo e incremental	Flujo continuo	Iterativo e incremental
Estilo de Código	Se enfoca en entregas incrementales	Sin estilo de código específico	Código simple, pruebas constantes, programación en pareja
Fases o etapas	Sprints	Flujo de trabajo	Iteraciones
Duración	2-4 semanas por sprint	Continua	1-2 semanas por iteración

Complejidad	Moderada	Baja a moderada	Alta
Modificable a cambios	Alta	Muy alta	Muy alta
Roles	• Scrum Master • Product Owner • Equipo de Desarrollo	No hay roles específicos.	• Cliente • Programador • Coach
Proceso	• Planificación • Ejecución • Revisión • Retrospectiva	• Visualización • Límites de WIP • Mejora continua	• Planificación • Diseño • Codificación • Pruebas • Lanzamiento

Según las características identificadas anteriormente en la **Tabla 27**, Scrum no es factible porque el tamaño mínimo del equipo requerido es de 3 a 9 personas, y nuestro equipo tiene menos de 3 personas. Por otro lado, Kanban no se considera adecuado porque no se ajusta al estilo de desarrollo iterativo e incremental que necesitamos para este proyecto. Por lo tanto, la metodología más apropiada sería la Programación Extrema (XP), debido a su enfoque en la simplicidad del código, pruebas constantes y programación en pareja basadas en historias de usuario, lo cual es práctico para proyectos pequeños a medianos con equipos reducidos y cambiantes.

3.2 Desarrollo de la propuesta

Para el desarrollo de este proyecto, se aplicaron las cinco fases del proceso que conforman la metodología XP: Planificación, Diseño, Codificación, Pruebas y Lanzamiento. Estas etapas facilitan un desarrollo ágil y adaptable de las aplicaciones web.

Planificación**:** En esta fase, se identifican los requisitos del sistema y se asignan roles dentro del equipo. Se crean historias de usuario y se estiman, estableciendo un plan detallado de entregas.

Diseño**:** Durante esta etapa, se define las tarjeras CRC (Clasificación, Responsabilidad y Colaboración), la arquitectura del proyecto, diseño de la base de datos y los mockups de la interfaz de usuario, guiando así el desarrollo.

Codificación**:** Aquí se implementa la lógica de negocio y se desarrolla tanto el backend como el frontend del sistema en iteraciones cortas y frecuentes.

Pruebas**:** Se llevan a cabo pruebas exhaustivas de aceptación y usabilidad para validar el software desarrollado y ajustarlo según sea necesario.

Lanzamiento: Finalmente, el software se prepara y se despliega en el entorno de producción, proporcionando soporte inicial y asegurando que el sistema esté listo para su uso continuo.

3.2.1 Fase I: Planificación

La fase inicial en la metodología XP es la planificación, donde se establece una comunicación efectiva entre el equipo de desarrollo y el cliente para definir los requisitos del sistema y su operación.

a. Requerimientos del sistema

De acuerdo con los datos obtenidos con las diversas técnicas y herramientas de recolección de datos, se concluye que en consultorio médico "DuoMedic" existen falencias en los procesos de manejo manual utilizando documentos físicos, por ende, se determinó que se debe usar las nuevas tecnologías para realizar un proceso de registro de ficha y prescripción médica. Para la mejora de estos procesos de tiene en cuenta los siguientes requerimientos como se detalla en la **Tabla 28**.

Tabla 28. Requerimientos del sistema

Requerimiento	Descripción
Módulo de autenticación	Autenticación segura con credenciales para el administrador y médico del consultorio.
Módulo de usuario	Creación de usuarios (médico) con sus datos personales.
Módulo de paciente	Creación del paciente por parte del médico logueado en el sistema
Módulo de registro de ficha médica	Crear y almacenar fichas médicas digitales para nuevos pacientes, incluyendo información personal y médica relevante. Permitir la actualización y edición de fichas médicas existentes con nueva información clínica. Facilitar el acceso rápido y seguro a la ficha médica de los pacientes
Registro de signos vitales	Registro de los signos vitales en la ficha médica del paciente
Módulo de prescripción médica	Crear recetas médicas digitales detallando el nombre del medicamento, dosis, frecuencia y duración del tratamiento. Enviar recetas médicas a los pacientes mediante correo electrónico. Mantener un historial de todas las prescripciones médicas realizadas a cada paciente.
Envío de la prescripción médica por correo electrónico	Envió de la prescripción médica en formato pdf al correo electrónico del paciente
Módulo de reportes	Visualización en pantalla o puede mostrar en formato pdf de la ficha médica de un paciente.

b. Roles asignados en el proyecto

En la metodología XP, se establece que cada miembro del equipo de desarrollo debe tener un rol asignado con sus respectivas funciones. La Tabla muestra los roles y sus descripciones:

Tabla 29. Roles del proyecto

Nombre	Rol	Descripción
José Villacrés	Programador	Responsable de llevar a cabo el desarrollo activo del sistema, cumpliendo con los plazos establecidos, asegurando la calidad del código a través de pruebas y retroalimentación, y colaborando estrechamente con el Cliente.
Dr. César Guerrero	Cliente – Tester	Representa a los usuarios finales, define las características y prioridades del Sistema y su correcto funcionamiento.
Ing. Hernán Naranjo	Coach	Vigila constantemente los avances del proyecto e implementa sus conocimientos con mejores alternativas o practicas más efectivas.

c. Arquitectura del sistema

El proyecto de la aplicación web se desarrolló con Django como framework principal de desarrollo. Se implementó el patrón de diseño Modelo-Vista-Plantilla (MTV), que es una variante del patrón Modelo-Vista-Controlador (MVC). Para el almacenamiento y manejo de datos, se empleó PostgreSQL como sistema de gestión de bases de datos relacional.

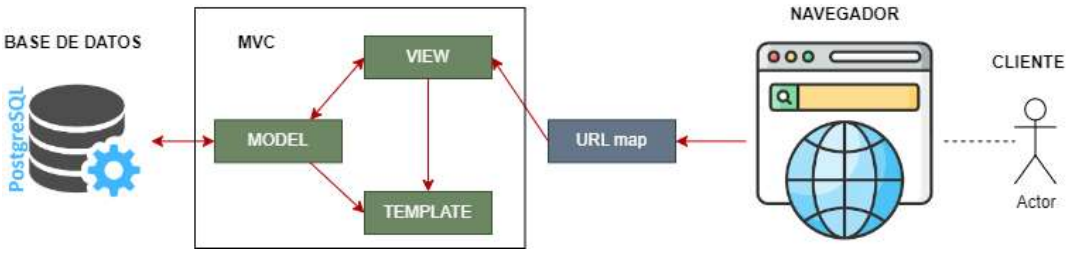


Figura 17. Arquitectura del sistema

d. Requerimientos de Software

Se realizó un análisis de las herramientas para el desarrollo del sistema, que se describen a continuación:

- ***Visual Studio Code.*** Este editor de código se utilizó debido a su robustez, flexibilidad y amplio ecosistema de herramientas, lo que lo convierte en una opción ideal para el desarrollo de software.

- ***Django.* –** Se utilizó la versión 5.0.6 de este framework web de alto nivel en Python debido a su capacidad para fomentar un desarrollo rápido y un diseño limpio y pragmático.

- ***Bootstrap.* –** Se empleó la versión 5 de esta biblioteca multiplataforma para proporcionar un estilo y diseño agradables al entorno de desarrollo de software.

- ***PostgreSQL.* –** Este gestor de base de datos de código abierto fue utilizado por su popularidad y eficacia en el mundo de la programación y desarrollo.

- ***Heroku.* -** Esta plataforma en la nube se utilizó para desplegar, ejecutar y administrar aplicaciones de manera sencilla y eficiente.

e. Historias de usuario

Después de definir los roles de los usuarios, establecer la arquitectura sistemas y requisitos del software a seguir, se elaboraron historias de usuario con el objetivo de especificar los requisitos fundamentales del proyecto.

Tabla 30. Historia de usuario definición de la estructura de la aplicación

Historias de usuario	
Número: 1	**usuario:** Cliente
Nombre de la historia: Definición de la estructura de la aplicación	
Prioridad: Alta	**Nivel de riesgo:** Alta
Puntos estimados: 2	**Iteración asignada:** 1
Programador responsable: José Villacrés	
Descripción: El programador puede definir la construcción de la estructura y entorno de la aplicación, incluyendo módulos que se van a desarrollar.	
Observación: Definir el ambiente de trabajo con las instalación y diseños establecidos	

Tabla 31. Historia de usuario módulo de autenticación

Historias de usuario	
Número: 2	**usuario:** Cliente
Nombre de la historia: Módulo de autenticación	
Prioridad: Alta	**Nivel de riesgo:** Alta
Puntos estimados: 3	**Iteración asignada:** 1
Programador responsable: José Villacrés	
Descripción: Creación del módulo de acceso al Sistema, donde cada usuario que vaya a tener acceso deberá ingresar con sus credenciales correspondientes (correo electrónico y contraseña) registradas previamente. Si el usuario no existe, el administrador debe crear un usuario para el cliente que lo usará.	
Observación: El correo electrónico de las credenciales ingresadas debe ser válido. Para mayor seguridad, la contraseña debe cumplir con un formato estándar que incluya una longitud mínima de 8 caracteres, al menos un número, una letra mayúscula y un carácter especial.	

Tabla 32. Historia de usuario módulo de usuarios

Historias de usuario	
Número: 3	**usuario:** Cliente
Nombre de la historia: Módulo de usuarios	
Prioridad: Alta	**Nivel de riesgo:** Alta
Puntos estimados: 3	**Iteración asignada:** 2
Programador responsable: José Villacrés	
Descripción: Creación del módulo para visualizar los médicos registrados, permitiendo realizar operaciones como crear, leer, actualizar y eliminar en caso de ser necesario.	
Observación: Desarrollar una interfaz para las operaciones CRUD (Crear, Leer, Actualizar y Eliminar) que permita a los usuarios autorizados agregar nuevos médicos.	

Tabla 33. Historia de usuario módulo de pacientes

Historias de usuario	
Número: 4	**usuario:** Cliente
Nombre de la historia: Módulo de pacientes	
Prioridad: Alta	**Nivel de riesgo:** Alta
Puntos estimados: 3	**Iteración asignada:** 2
Programador responsable: José Villacrés	
Descripción: Creación del módulo para visualizar los pacientes registrados, permitiendo realizar operaciones como crear, leer, actualizar y eliminar en caso de ser necesario. Esto implica la capacidad de agregar un nuevo paciente y guardarlo, así como un buscador de pacientes por filtros específicos, como cédula, nombre o apellidos.	
Observación: Para facilitar la gestión eficiente de los pacientes, se debe optar por un buscador robusto que permita localizar a los pacientes de manera rápida y precisa. Además, se desarrollar una vista completa de operaciones CRUD (Crear, Leer, Actualizar y Eliminar) que permita a los usuarios autorizados agregar nuevos pacientes.	

Tabla 34. Historia de usuario gestión de registro de ficha médica

Historias de usuario	
Número: 5	**usuario:** Cliente
Nombre de la historia: Módulo de registro de ficha médica	
Prioridad: Alta	**Nivel de riesgo:** Alta
Puntos estimados: 4	**Iteración asignada:** 3
Programador responsable: José Villacrés	
Descripción: Creación del módulo para gestionar fichas médicas digitales de pacientes, permitiendo crear y almacenar fichas completas con información personal y médica relevante para nuevos pacientes, así como facilitar el acceso rápido y seguro para actualizar y editar los datos clínicos en fichas existentes. De este modo, se mantendrá un registro actualizado de la historia clínica de cada paciente, tanto para quienes recién ingresan al sistema como para aquellos con un historial médico previo.	
Observación: Para este módulo, el paciente debe estar dispuesto al registro de sus datos personales y de salud, detallando los antecedentes clínicos anteriores y nuevos, pero para que tenga confidencialidad, el aplicativo web con los datos del paciente se debe cumplir con la LOPD (Ley Orgánica de Protección de Datos Personales), garantizando la protección adecuada de los datos personales.	

Tabla 35. Historia de usuario registro de signos vitales

Historias de usuario	
Número: 6	**usuario:** Cliente
Nombre de la historia: Registro de signos vitales	
Prioridad: Media	**Nivel de riesgo:** Media
Puntos estimados: 2	**Iteración asignada:** 3
Programador responsable: José Villacrés	
Descripción: En este módulo el médico registra los signos vitales ya previamente obtenidos, en el módulo permitirá ingresar la descripción de los síntomas para proceder a diagnosticar y registrar en la ficha médica.	
Observación: El sistema deberá incorporar la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud (CIE-11) de la Organización Mundial de la Salud al momento de diagnosticar una enfermedad.	

Tabla 36. Historia de usuario gestión de prescripción médica

Historias de usuario	
Número: 7	**usuario:** Cliente
Nombre de la historia: Módulo de prescripción médica	
Prioridad: Alta	**Nivel de riesgo:** Alta
Puntos estimados: 3	**Iteración asignada:** 3
Programador responsable: José Villacrés	
Descripción: El módulo debe permitir al médico crear recetas médicas digitales de manera fácil e intuitiva. Durante este proceso, se deberá ingresar información detallada sobre los medicamentos prescritos, incluyendo el nombre del medicamento, la dosis recomendada, la frecuencia de administración y la duración del tratamiento.	
Observación: Al momento de generar la prescripción médica se debe tener en cuenta la implementación del Índice ATC/DDD 2023, que es un sistema de clasificación de medicamentos ampliamente utilizado en el ámbito médico. También se debe realizar una validación y autocompletado de medicamentos.	

Tabla 37. Historia de usuario envío de prescripción médica por correo electrónico

Historias de usuario	
Número: 8	**usuario:** Cliente
Nombre de la historia: Envío de la prescripción médica por correo electrónico	
Prioridad: Alta	**Nivel de riesgo:** Media
Puntos estimados: 2	**Iteración asignada:** 4
Programador responsable: José Villacrés	
Descripción: Creación del módulo para el envío la prescripción médica en un formato digital estándar, como PDF, garantizando su legibilidad e integridad.	
Observación: Para que la prescripción médica sea válida se debe cumplir con las normativas establecidas por el Ministerio de Salud Pública y la Ley Orgánica de Protección de Datos Personales (LOPDP).	

Tabla 38. Historia de usuario generador de reportes

Historias de usuario	
Número: 9	**usuario:** Cliente
Nombre de la historia: Módulo de reportes	
Prioridad: Alta	**Nivel de riesgo:** Media
Puntos estimados: 2	**Iteración asignada:** 5
Programador responsable: José Villacrés	
Descripción: Creación de un apartado para la visualización de datos del paciente de manera detalla.	
Observación: El personal médico, selecciona el reporte solicitado permitiendo visualizar en pantalla o puede mostrar en formato pdf.	

Tabla 39. Historia de usuario cierre de aplicación

Historias de usuario	
Número: 10	**usuario:** Cliente
Nombre de la historia: Cierre de aplicación	
Prioridad: Alta	**Nivel de riesgo:** Media
Puntos estimados: 2	**Iteración asignada:** 5
Programador responsable: José Villacrés	
Descripción: Creación de un apartado para que se cierre la aplicación de forma segura.	
Observación: El personal médico, deberá tener un apartado de cierre de aplicación superior derecha en donde aparecerá el nombre de su usuario con el apartado cerrar sesión	

f. Técnicas para estimar el tiempo

Para estimar el tiempo de cada historia de usuario, se empleará la técnica de puntos de historia. Esta técnica ayuda para una comprensión más clara de la carga del trabajo en términos relativos facilitando la planificación y asignación de tareas.

Se evalúa la técnica a través de los siguientes enfoques:

- En cada historia de usuario cuenta con un punto estimado que se traduce a una jordana laboral de horas de trabajo.
- La planificación de tareas se basa en 6 días de trabajo por semana, que da como resultado un total de 48 horas de trabajo semanales.

g. Valoración de historias de usuario

Una vez creadas las historias de usuario, se estima el tiempo necesario para desarrollar cada una de ellas utilizando la técnica de estimación por tiempo.

Tabla 40. Estimación de historias de usuario

N°	Historia de usuario	Puntos estimados	Tiempo estimado		
			Días	Horas	Semanas
1	Definición de la estructura de la aplicación	2	2	10	0.3
2	Módulo de autenticación	3	3	15	0.5
3	Módulo de usuarios	3	3	15	0.5
4	Módulo de paciente	3	3	15	0.5
5	Módulo de registro de ficha médica	4	4	20	0.7
6	Registro de signos vitales	2	2	10	0.3
7	Módulo de prescripción médica	3	3	15	0.5
8	Envío de la prescripción médica por correo electrónico	2	2	10	0.3
9	Módulo de reportes	2	2	10	0.3
10	Cierre de aplicación	2	2	10	0.3
Total		**26**	**26**	**130**	**4.2**

h. Planificación de entregas

A continuación, la **Tabla 41** presenta la planificación de entregas basada en las iteraciones, teniendo en cuenta el tiempo de desarrollo requerido para cada historia de usuario.

Tabla 41. Plan de entrega

N°	Historia de usuario	Puntos estimados	Tiempo estimando		Iteraciones				
			Días	Horas	1	2	3	4	6
1	Definición de la estructura de la aplicación	2	2	10	X				
2	Módulo de autenticación	3	3	15	X				
3	Módulo de usuarios	3	3	15		X			
4	Módulo de usuarios	3	3	15		X			
5	Módulo de registro de ficha médica	4	4	20			X		

6	Registro de signos vitales	2	2	10			X		
7	Módulo de prescripción médica	3	3	15			X		
8	Envío de la prescripción médica por correo electrónico	2	2	10				X	
9	Módulo de reportes	2	2	10					X
10	Cierre de aplicación	2	2	10					X

i. Plan de iteraciones

Al tener ya definido y estimado el tiempo para cada historia de usuario relacionadas con su respectiva iteración, se plantea un el plan de iteraciones que detalla las actividades y el desarrollo para la aplicación.

- ***Iteración 1***

Tabla 42. Historias de usuario de la iteración 1

N°	**Historia de usuario**	**Prioridad**	**Riesgo**
1	Definición de la estructura de la aplicación	Alta	Alta
2	Módulo de autenticación	Alta	Alta

- **Definición de la estructura de la aplicación**

Se define la estructura de la aplicación web, eligiendo la versión del framework de Django, preparación del entorno con todas la instalación y extensiones. Además, se selecciona la base de datos para el desarrollo de la aplicación.

Tabla 43. Actividades de la definición de la estructura de la aplicación

Número de historia: 1	**Responsable:** José Villacrés
Lista de Actividades	
• Establecer e identificar la estructura principal para la aplicación web. • Instalación de las dependencias necesarias para el ambiente de trabajo. • Diseño entidad relación de la base de datos. • Configuración para la conectividad entre el framework Django y la base de datos PostgreSQL.	

- **Módulo de autenticación**

Implementación de un sistema de control de acceso que permitirá a los usuarios autenticar sus credenciales para ingresar a la aplicación web. Esto incluirá la creación de una interfaz de inicio de sesión, la verificación de usuarios y contraseñas, y la gestión de sesiones para asegurar que solo los usuarios autorizados puedan acceder a las funcionalidades del sistema,

Tabla 44. Actividades para el módulo de autenticación

Número de historia: 2	**Responsable:** José Villacrés
Lista de Actividades	
• Definir los requisitos del sistema de autenticación. • Diseñar la interfaz de usuario para el formulario de inicio de sesión. • Especificar los criterios de validación para usuarios y contraseñas. • Configurar el entorno de desarrollo.	

- ***Iteración 2***

Tabla 45. Historias de usuario de la iteración 2

N°	**Historia de usuario**	**Prioridad**	**Riesgo**
3	Módulo de usuarios	Alta	Alta
4	Módulo de pacientes	Alta	Alta

- **Módulo de usuarios**

Este módulo permite gestionar usuarios, proporcionando funcionalidades para crear, modificar y eliminar cuentas de usuario para los médicos. Esto incluye la actualización de la información de usuario. Además, asegura que solo los administradores o usuarios con privilegios adecuados puedan realizar estas acciones.

Tabla 46. Actividades para el módulo de usuarios

Número de historia: 3	**Responsable:** José Villacrés
Lista de Actividades	
• Implementar funciones que permitan al administrador agregar, editar y eliminar información • Crear la interfaz de usuario para la gestión de médicos • Asegurar que la interfaz sea intuitiva y fácil de usar.	

- **Módulo de pacientes**

Este módulo permite gestionar paciente, proporcionando funcionalidades para crear, modificar y eliminar. En este módulo se establece un buscador de pacientes por medio de cedula o nombre completo.

Número de historia: 4	**Responsable:** José Villacrés
Lista de Actividades	
• Implementar funciones que permitan al administrador agregar, editar y eliminar información • Crear la interfaz de usuario para la gestión de paciente • Búsqueda de paciente por cédula o nombres completes. • Asegurar que la interfaz sea intuitiva y fácil de usar.	

- ***Iteración 3***

Tabla 47. Historias de usuario de la iteración 3

N°	Historia de usuario	Prioridad	Riesgo
5	Módulo de registro de ficha médica	Alta	Alta
6	Registro de signos vitales	Media	Media
7	Módulo de prescripción médica	Alta	Alta

- **Módulo de registro de ficha médica**

Este módulo permite gestionar fichas médicas digitales de pacientes, permitiendo crear, almacenar, actualizar y editar fichas completas con información personal y médica relevante. Está diseñado para mantener un registro actualizado de la historia clínica de cada paciente, cumpliendo con la Ley Orgánica de Protección de Datos Personales (LOPD) para garantizar la confidencialidad y seguridad de los datos.

Tabla 48. Actividades para el módulo de registro de ficha médica

Número de historia: 5	**Responsable:** José Villacrés
Lista de Actividades	
• Definir los requisitos del módulo de registro de ficha médica. • Diseñar la estructura de las fichas médicas. • Desarrollar la interfaz de usuario para la creación y edición de fichas médicas. • Implementar formularios para ingresar y actualizar la información personal y médica del paciente. • Implementar la lógica para el manejo de datos clínicos y personales de los pacientes. • Implementar medidas de seguridad para proteger la información médica y personal.	

- **Registro de signos vitales**

Permite al personal médico registrar los signos vitales previamente obtenidos de los pacientes. Además, facilitará la entrada de descripciones de síntomas para el diagnóstico y registro en la ficha médica correspondiente. Además, el sistema incorporará la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud (CIE-11) de la OMS para la codificación de diagnósticos.

Tabla 49. Actividades para el registro de signos vitales

Número de historia: 6	**Responsable:** José Villacrés
Lista de Actividades	
• Diseñar la interfaz de usuario para la entrada de datos de signos vitales y síntomas. • Integrar controles para ingresar datos como presión arterial, frecuencia cardíaca, temperatura, entre otros. • Incorporar la codificación CIE-11 para diagnósticos.	

- **Módulo de prescripción médica**

Este módulo permite al médico crear recetas médicas digitales de manera fácil e intuitiva. Podrá ingresar información detallada sobre los medicamentos prescritos, incluyendo el nombre del medicamento, la dosis recomendada, la frecuencia de administración y la duración del tratamiento. Además, el sistema debe implementar el Índice ATC/DDD 2023 para la clasificación de medicamentos y proporcionar funciones de validación y autocompletado de medicamentos.

Tabla 50. Actividades para el módulo de prescripción médica

Número de historia: 7	**Responsable:** José Villacrés
Lista de Actividades	
• Definir los requisitos del módulo de prescripción médica. • Desarrollar la interfaz de usuario para la creación de recetas médicas. • Implementar formularios para ingresar y actualizar la información de los medicamentos prescritos. • Añadir funciones de autocompletado y validación de medicamentos. • Implementar la lógica para el manejo de la información de las prescripciones médicas. • Integrar el Índice ATC/DDD 2023 para la clasificación de medicamentos.	

- ***Iteración 4***

Tabla 51. Historias de usuario de la iteración 4

N°	**Historia de usuario**	**Prioridad**	**Riesgo**
8	Envío de la prescripción médica por Correo electrónico	Alta	Media

- **Envío de la prescripción médica por correo electrónico**

Este módulo permite enviar prescripciones médicas en un formato digital estándar, como PDF, a través de correo electrónico. La herramienta garantiza la legibilidad e integridad de las prescripciones y asegura que cumplan con las normativas establecidas por el Ministerio de Salud Pública y la Ley Orgánica de Protección de Datos Personales (LOPDP).

Tabla 52. Actividades para el envío de la prescripción médica

Número de historia: 8	**Responsable:** José Villacrés
Lista de Actividades	
• Definir los requisitos del módulo de envío de prescripciones. • Diseñar el formato digital estándar (PDF) para las prescripciones médicas. • Implementar la funcionalidad para generar prescripciones médicas en formato PDF. • Integrar servicios SMTP de correo electrónico para el envío de prescripciones	

• Implementar validaciones para asegurar que las prescripciones cumplan con las normativas del Ministerio de Salud Pública.

- ***Iteración 5***

Tabla 53. Historias de usuario de la iteración 5

N°	Historia de usuario	Prioridad	Riesgo
9	Módulo de reportes	Alta	Media
10	Cierre de aplicación	Alta	Media

- **Módulo de reportes**

El módulo de reportes permite al personal médico visualizar de manera detallada los datos de los pacientes y seleccionar reportes para ser mostrados en pantalla o generados en formato PDF. Esta funcionalidad es crucial para facilitar el acceso rápido y preciso a la información médica necesaria para la atención y tratamiento de los pacientes, cumpliendo con estándares de seguridad y privacidad de datos establecidos por las normativas pertinentes.

Tabla 54. Actividades para el módulo de reportes

Número de historia: 9	**Responsable:** José Villacrés
Lista de Actividades	
• Definir los requisitos específicos del módulo de reportes. • Diseñar la interfaz de usuario para la visualización detallada de datos. • Especificar los formatos de visualización. • Implementar la opción de seleccionar y mostrar reportes específicos.	

- **Cierre de aplicación**

Facilitar al personal médico en cerrar la aplicación de manera segura. Esto se logrará mediante la implementación de un apartado específico en la esquina superior derecha de la interfaz de usuario, donde aparecerá el nombre de usuario del médico junto con la opción “Cerrar sesión”.

Tabla 55. Actividades para el cierre de aplicación

Número de historia: 10	**Responsable:** José Villacrés
Lista de Actividades	
• Diseñar la ubicación y apariencia del botón de cierre de aplicación dentro de la interfaz. • Implementar la lógica necesaria para el cierre seguro de la aplicación. • Asegurar que todos los procesos y conexiones se manejen adecuadamente antes de cerrar la aplicación.	

3.2.2 Fase II: Diseño

La segunda fase del diseño de la metodología XP es el diseño, y se centra en estructura y el diseño que se llevara a cabo para el desarrollo del proyecto.

a. Tarjetas CRC

Las tarjetas CRC (Clasificación, Responsabilidad y Colaboración) son herramientas de análisis cruciales en el desarrollo de software. Se utilizan para mapear las responsabilidades y relaciones entre las diferentes clases de un sistema. Al asignar roles específicos a cada clase y definir cómo interactúan entre sí, estas tarjetas proporcionan una representación visual que ayuda a los equipos a entender y planificar de manera efectiva la estructura y funcionalidad del software en desarrollo.

Tabla 56. Tarjeta CRC módulo de autenticación

Acceso al aplicativo web -login	
Responsabilidades	**Colaboradores**
Mostrar la interfaz para el ingreso de las credenciales Validación del email del usuario	Modelo Usuario Controlador Usuario Clase Auth Vista Index
Observaciones: Para el acceso al aplicativo se debe validar que las credenciales sean correctas	

Tabla 57. Tarjeta CRC módulo de usuarios

Módulo de usuarios	
Responsabilidades	**Colaboradores**
Crear, editar y eliminar perfiles de usuarios. Presentar formulario intuitivo para el ingreso de usuarios	Modelo médico Controlador médico Vista médico
Observaciones: Validación de datos, cédula, formato del correo electrónico.	

Tabla 58. Tarjeta CRC módulo de pacientes

Módulo de pacientes	
Responsabilidades	**Colaboradores**
Crear, editar y eliminar perfiles de pacientes. Presentar formulario intuitivo para el ingreso de paciente, Busqueda de paciente.	Modelo paciente Controlador paciente Vista paciente Pipe búsqueda paciente
Observaciones: Ninguna	

Tabla 59. Tarjeta CRC módulo de registro de ficha médica

Módulo de registro de ficha médica	
Responsabilidades	**Colaboradores**
Gestiona las operaciones CRUD (crear, leer, actualizar, eliminar) de las fichas médicas. Presentar formulario intuitivo para el ingreso de los datos médicos del paciente.	Modelo paciente, Modelos antecedentes (personales, familiares, clínicos), Modelo alergias, Modelo hábitos, Modelo traumas Controlador ficha médica Vista ficha médica
Observaciones: Ninguna	

Tabla 60. Tarjeta CRC registro de signos vitales

Registro de signos vitales	
Responsabilidades	**Colaboradores**
Presentar formulario para el ingreso de datos signos vitales Registro del diagnostico	Modelo signos vitales Modelo diagnostico Controlador ficha médica Vista ficha médica
Observaciones: Tomar en cuenta la codificación CIE-11 para el diagnóstico de la enfermedad	

Tabla 61. Tarjeta CRC módulo de prescripción médica

Módulo de prescripción médica	
Responsabilidades	**Colaboradores**
Presentar formulario para registrar la información detallada sobre los medicamentos, el nombre del médicamento, la dosis recomendada, la frecuencia de administración y la duración del tratamiento.	Modelo prescripción médica Modelo prescripción médica Controlador prescripción médica Vista prescripción médica
Observaciones: Tomar en cuenta la codificación del Índice ATC/DDD 2023, que es un sistema de clasificación de medicamentos ampliamente utilizado en el ámbito médico.	

Tabla 62. Tarjeta CRC envío de la prescripción médica por correo electrónico

Envío de la prescripción médica por correo electrónico	
Responsabilidades	**Colaboradores**
Apartado para el envío de correo electrónico automáticamente de la prescripción médica.	Extensiones de las librerías para el envío de correo SMTP.
Observaciones: se debe asignar el correo electrónico del médico que envía, para realizar al paciente.	

Tabla 63. Tarjeta CRC módulo de reportes

Módulo de reportes	
Responsabilidades	**Colaboradores**
Apartado para visualizar de manera detallada los datos de los pacientes y seleccionar reportes de los mismos.	Modelo paciente Modelo ficha médica Vista reportes
Observaciones: Mostrar detalladamente los datos más relevantes del paciente	

b. Diseño de base de datos

El diseño de la base de datos relacional se realizó en el gestor de base de datos PostgreSQL que funciona como principal medio de almacenamiento de las operaciones y transacciones que se va llevan a cabo al momento de realizar el registro de ficha y prescripción médica como se observa en el **Anexo D**.

c. Iteración 1

Conforme al plan de entrega descrito en la **Tabla 41**, se continúa avanzando en la realización de las historias de usuario. A continuación, se muestran las interfaces de cada historia de usuario correspondientes a la iteración 1 que conforman la aplicación web. Estas fueron diseñadas con el software Miro debido a su interfaz intuitiva y su amplia biblioteca de elementos y componentes.

- ***Definición de la estructura de la aplicación***

Durante el proceso de desarrollo de la aplicación web, se seleccionó Django en su versión 5.0.6 como framework principal y PostgreSQL como sistema de gestión de bases de datos. Además, se adoptó Bootstrap 5 para el diseño y estilos del front-end. A continuación, se detalla el procedimiento de configuración y desarrollo:

- **Configuración del Entorno de Desarrollo con sus dependencias**

Para el desarrollo con Django 5.0.6 se debe tener instalado Python con una versión mínimo 3.10 o superior de manera global. Para el proyecto se utilizó la versión 3.12.1 siendo una de las versiones más actuales hasta el momento que se realizó el proyecto.

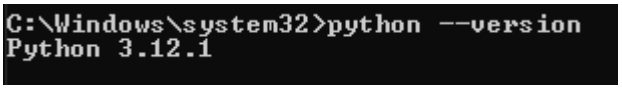


Figura 18. Versión de Python

La versión del gestor de basa de datos utilizada para el desarrollo del proyecto fue la 16.3 siendo una de las versiones más actuales hasta el momento que se realizó el proyecto.

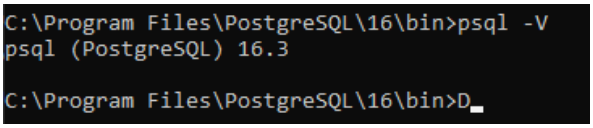


Figura 19. Versión de PostgreSQL

Una vez definidas las versiones de compatibilidad para el desarrollo de la aplicación, se procedió a crear un entorno virtual utilizando "virtualenv". Este paso permitió aislar las dependencias del proyecto dentro de una carpeta específica donde se alojará el proyecto.



Figura 20. Entorno virtual “virtualenv”

Después de crear el entorno virtual se procede con la activación de la virtualización, para instalar las dependencias que son Django 5.0.6 y el conector de PostgreSQL.

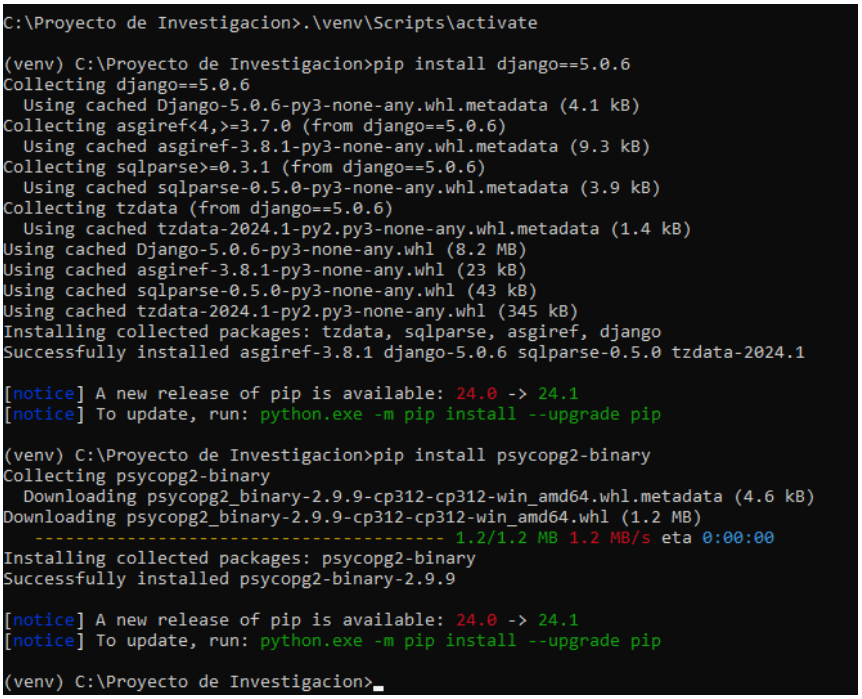


Figura 21. Instalación de dependencias en el entorno virtual

Para finalizar con la configuración del entorno se procedió a crear el proyecto Django.

```
(venv) C:\Proyecto de Investigacion>django-admin startproject duomedic .

(venv) C:\Proyecto de Investigacion>
```

Figura 22. Creación del proyecto Django

- ***Módulo de autenticación***

La interfaz de autenticación conta de dos credenciales las cuales son el correo y la contraseña. Permitiendo el acceso al sistema

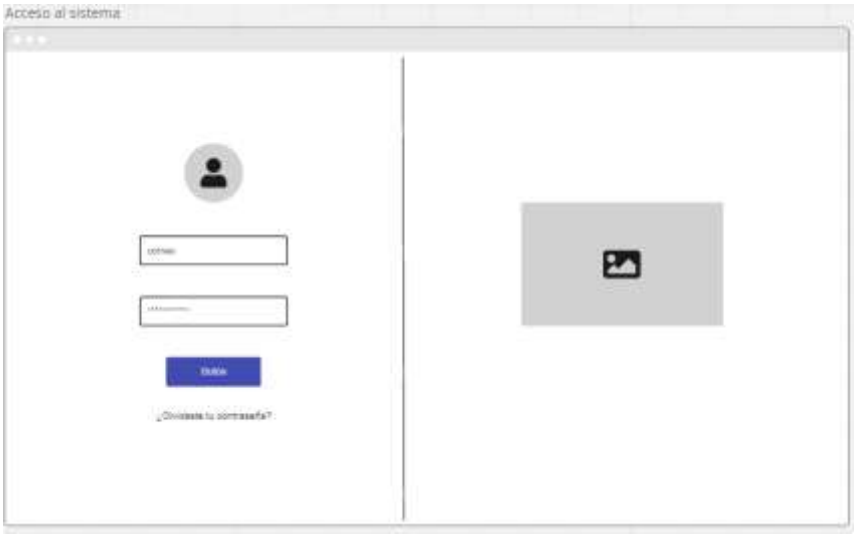


Figura 23. Mockup de la interfaz para la autenticación de usuario

d. Iteración 2

A continuación, se muestran las interfaces de cada historia de usuario correspondientes a la iteración 2 que conforman la aplicación web.

- ***Módulo de usuarios***

El módulo de médico permite la búsqueda, creación o manipulación de los datos de los médicos por parte del usuario o médico administrador.

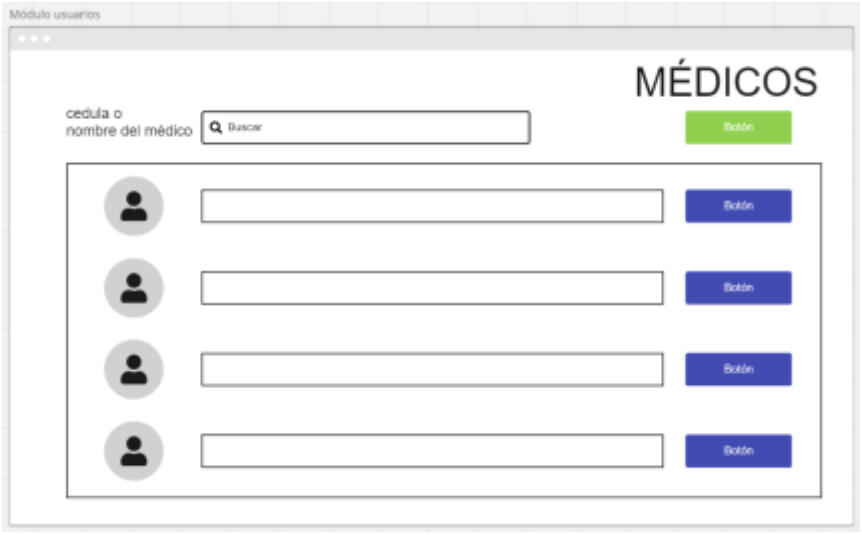


Figura 24. Mockup de la interfaz para búsqueda de médicos

Al no existe un médico en la base de datos se debe ingresar uno nuevo al dar click en el botón de color verde, donde enviará le enviará a una nueva interfaz para el ingreso de información del médico.



Figura 25. Mockup de la interfaz para ingresar datos del médico

Al dar clic en el grupo de botones de acciones se despliegan el botón editar y eliminar médico.



Figura 26. Mockup de la interfaz para editar datos del médico

Dentro de los botones de acciones, se encuentra el botón de eliminar, al dar clic en eliminar mostrara un modal de advertencia sobre eliminación del médico, **Figura 27**.



Figura 27. Mockup de la interfaz para eliminar un médico

- ***Módulo de pacientes***

El módulo de pacientes permite la búsqueda, creación o manipulación de los datos de los pacientes por marte del médico administrativo. Este módulo permite el ingreso de los datos médicos para los pacientes que son nuevos en el consultorio.



Figura 28. Mockup de la interfaz para búsqueda de pacientes

Si en la búsqueda de pacientes, el paciente existe me permitirá visualizar en la tabla y pacientes para su vez poder ver la ficha médica de ese paciente.



Figura 29. Mockup de la interfaz para el paciente encontrado

Al seleccionar en mostrar ficha médica se podrá visualizar los datos personales, datos clínicos del paciente como antecedentes (personales, familiares, clínicos), traumas y alergias.

En la **Figura 30** se despliega los datos del paciente personales e informativos.



Figura 30. Mockup de la interfaz para la visualización de la ficha médica Parte 1 (Datos personales)

En la **Figura 31** se despliega los datos correspondientes del paciente seleccionado visualizando antecedentes personales, familiares y clínicos.



Figura 31. Mockup de la interfaz para la visualización de la ficha médica Parte 2 (Antecedentes)

Dentro de esa misma ventana se visualizará los traumas clínicos y alergias que el paciente presenta como se visualiza en la **Figura 32**.



Figura 32. Mockup de la interfaz para la visualización de la ficha médica Parte 3 (Traumas y alergias)

Si en la búsqueda de pacientes, el paciente no existe se procederá al ingreso de los datos del paciente.



Figura 33. Mockup de la interfaz para el ingreso de datos personales en la ficha médica Parte 1

En la **Figura 34**, se visualiza los campos que son necesarios para el ingreso de datos médicos del paciente relacionados con los antecedentes personales, familiares y clínicos.



Figura 34. Mockup de la interfaz para el ingreso de antecedentes en la ficha médica Parte 2

En la **Figura 35**, se visualiza los campos que son necesarios para el ingreso de datos médicos del paciente relacionados con los traumas clínicos y alergias.



Figura 35. Mockup de la interfaz para el ingreso de traumas y alergia en la ficha médica Parte 3

Al dar clic en el grupo de botones de acciones se despliegan el botón editar y eliminar paciente, **Figura 36.**



Figura 36. Botones de acción

En la **Figura 37** se visualiza la venta que permite editar los datos personales e informativos del paciente seleccionado.



Figura 37. Mockup de la interfaz para editar datos personales en la ficha médica Parte 1

En la **Figura 38** se visualiza los campos que se pueden editar, como los son los atributos de los antecedentes personales, familiares y clínicos.



Figura 38. Mockup de la interfaz para editar antecedentes en la ficha médica Parte 2

En la **Figura 39.** se visualiza los campos que se pueden editar, como los son los atributos de los antecedentes personales, familiares y clínicos.



Figura 39. Mockup de la interfaz para editar de traumas y alergia en la ficha médica Parte 3

Dentro de los botones de acciones, se encuentra el botón de eliminar, al dar clic en eliminar mostrara un modal de advertencia sobre eliminación del paciente, **Figura 40**.



Figura 40. Mockup de la interfaz para eliminar un paciente

e. Iteración 3

A continuación, se muestran las interfaces de cada historia de usuario correspondientes a la iteración 3 que conforman la aplicación web.

- ***Módulo de registro de ficha médica***

En este módulo se visualiza los daos de la ficha médica, permitiendo ver detalladamente cada uno de los aspectos clínicos presentados por el paciente en las revisiones anteriores.

Los mockups de las interfaces son las mismas que se presentaron en el módulo de pacientes **Figura 30, Figura 31, Figura 32.**

Adicional a esos mockups se podrá visualizar los apartados de registros médicos con sus respectivos signos vitales, y prescripciones médicas anterior con su respectiva dosificación.



Figura 41. Mockup de la interfaz para la visualización de la ficha médica Parte 4 (Registros médicos)

En la **Figura 42** Figura 39. se visualiza los campos relacionados con el registro de los signos vitales, en donde el médico lo realiza tomando con sus implementos médicos.



Figura 42. Mockup de la interfaz para la visualización de la ficha médica Parte 5 (Signos vitales)

Para el diagnóstico se tiene presente la asignación de la posible enfermedad presentado por el paciente, para posterior a eso determinar que medicamentos son lo que requiere para tratarlo, como se visualiza en la **Figura 43.**



Figura 43. Mockup de la interfaz para la visualización de la ficha médica Parte 6 (diagnósticos y sus prescripciones médicas)

- ***Registro de signos vitales***

Después que del médico realiza y toma los signos vitales al paciente se procese al ingreso de datos obtenidos en la interfaz visualizada en la **Figura 42**, la cual será reutilizada para el ingreso de un nuevo registro de signos vitales. Para acceder a este formulario se puede acceder de dos maneras, la primara mediante el grupo de botones

de acción la cual dice generar consulta nueva, la otra forma de acceder es en la interfaz de ficha médica generando un botón de generar consulta nueva.

- ***Módulo de prescripción médica***

Después de que el médico toma los signos vitales del paciente y registrarlos en la ficha médica. El médico realiza la descripción de sistemas haciendo preguntas al paciente para tomar una decisión basado en esa información. Esa decisión del médico formula un diagnóstico de la enfermedad posible para después genera la prescripción correspondiente en el sistema. La prescripción incluye detalles sobre los medicamentos, dosis, frecuencia, y cualquier otra instrucción relevante.



Figura 44. Mockup de la interfaz para la prescripción médica

f. Iteración 4

A continuación, se muestra la interfaz de la historia de usuario correspondiente a la iteración 4 que conforma para la aplicación web.

- ***Envío de la prescripción médica por correo electrónico***

Al finalizar la prescripción médica el sistema envía automáticamente la receta médica al paciente a través de correo electrónico.

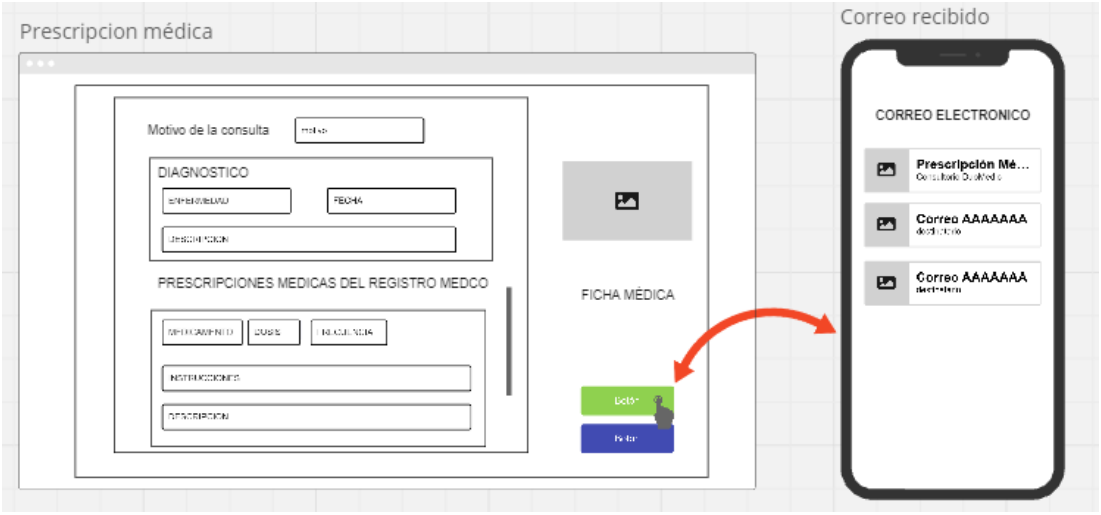


Figura 45. Mockup del envió automático de la prescripción médica

g. Iteración 5

A continuación, se muestra la interfaz de la historia de usuario correspondiente a la iteración 5 que conforma para la aplicación web.

- ***Cierre de aplicación***

Para el cierre de la aplicación se localiza la parte superior derecha, donde el usuario podrá cerrar sesión y salir del sistema.

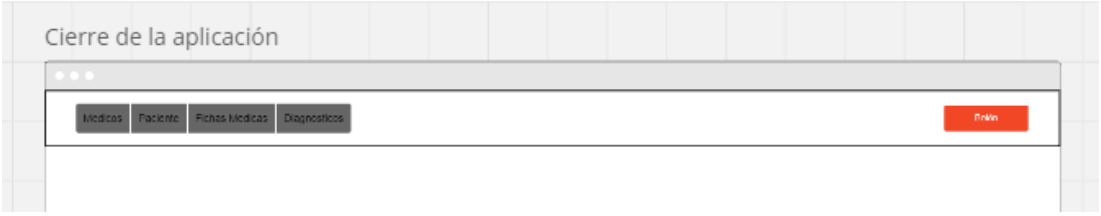


Figura 46. Mockup de la interfaz para cerrar la aplicación

3.2.3 Fase III: Codificación

En esta fase, se llevó a cabo el desarrollo del código de las iteraciones de la **Tabla 41**Figura 31**,** creando las funcionalidades con el framework Django 5.0.6 y el gestor de base de datos PostgreSQL 16.03.

a. Conexión a la base de datos

Para establecer la conexión de la base de datos PostgreSQL con Django se debe configurar el archivo “settings.py” donde se ingresa los respectivos parámetros como el usuario de administrador de la base de datos con su respectiva contraseña, puerto de conexión de la base que por defecto es el 5432, host de la base de datos que en donde esta alojada la base de datos por el momento se estable el localhost antes de subirlo a producción, el nombre de dela base de datos y el drive de conexión para establecer la conexión.



Figura 47. Conexión a la base de datos PostgreSQL

Django ofrece un sistema integrado robusto de administración y autenticación de usuario. Permitiendo gestionar los datos.

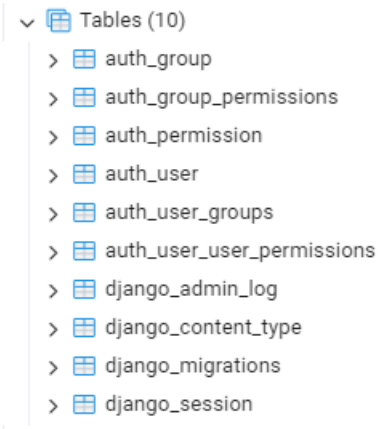


Figura 48. Tablas de administración propias de Django

b. Creación de las tablas en la base de datos

Para la creación de las tablas se optó en crear de manera de tipo models con la ayuda de Django, ya que este framework nos permite crear tablas con ayuda de models.



```python
from django.db import models

# modelo usuario
class Usuario(models.Model):
    email = models.EmailField(max_length=50, unique=True)
    password = models.CharField(max_length=25)

    def __str__(self):
        return self.email

# modelo especialidad
class Especialidad(models.Model):
    especialidad = models.CharField(max_length=50)
    descripcion = models.CharField(max_length=255, blank=True, null=True)

    def __str__(self):
        return self.especialidad

# modelo medico
class Medico(models.Model):
    cedula = models.CharField(max_length=10, unique=True)
    nombre1 = models.CharField(max_length=50)
    nombre2 = models.CharField(max_length=50, blank=True, null=True)
    apellido1 = models.CharField(max_length=50)
    apellido2 = models.CharField(max_length=50, blank=True, null=True)
    celular = models.CharField(max_length=10)
    email = models.EmailField(max_length=50)
    direccion = models.CharField(max_length=50)
    especialidad = models.ForeignKey(Especialidad, on_delete=models.CASCADE)
    usuario = models.OneToOneField(Usuario, on_delete=models.CASCADE)

    def __str__(self):
        return f"{self.nombre1} {self.apellido1}"

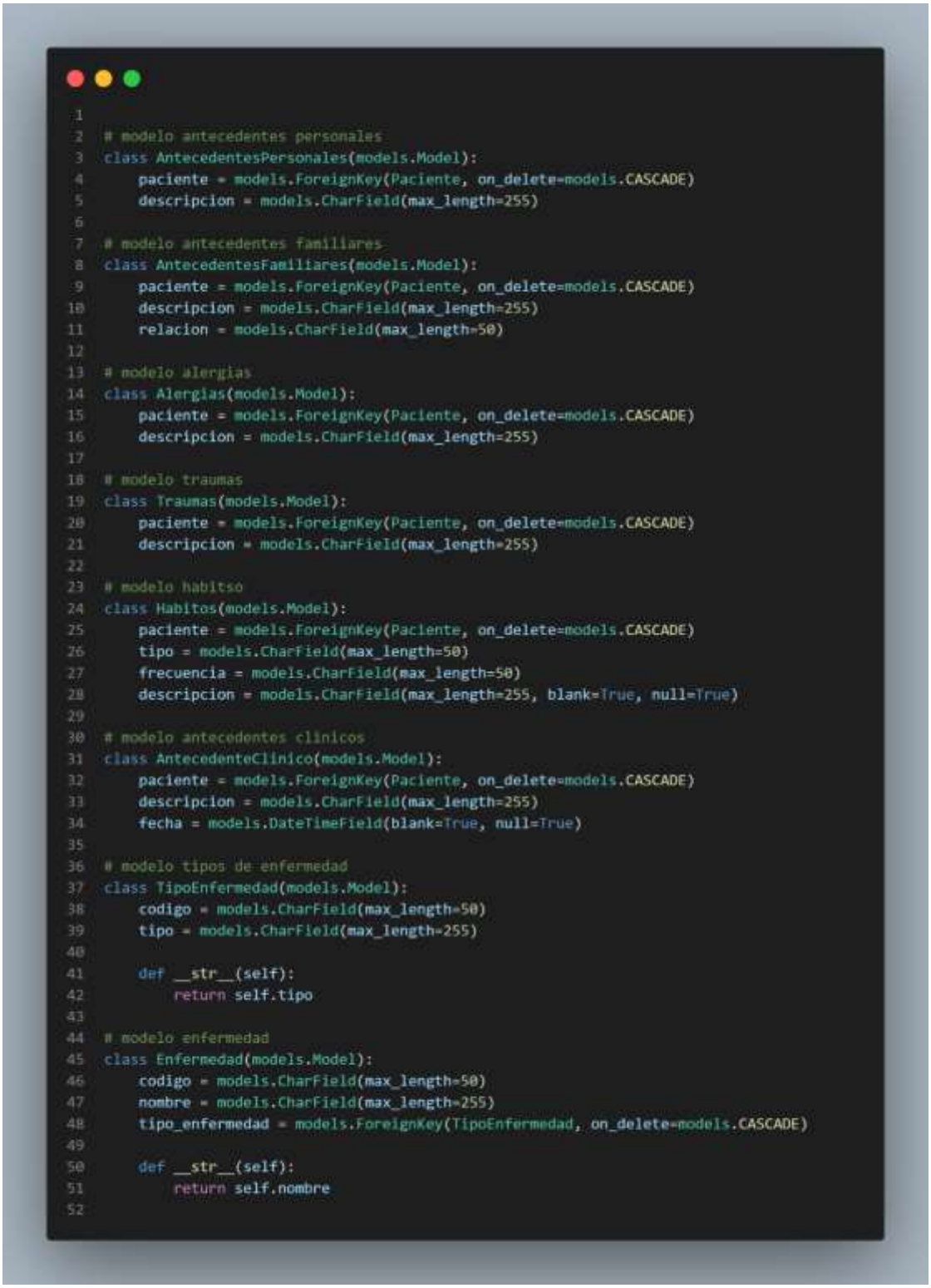
# modelo tipo de sangre
class TipoSangre(models.Model):
    sangre = models.CharField(max_length=50)
    descripcion = models.CharField(max_length=255, blank=True, null=True)

    def __str__(self):
        return self.sangre

# modelo paciente
class Paciente(models.Model):
    cedula = models.CharField(max_length=10, unique=True)
    nombre1 = models.CharField(max_length=50)
    nombre2 = models.CharField(max_length=50, blank=True, null=True)
    apellido1 = models.CharField(max_length=50)
    apellido2 = models.CharField(max_length=50, blank=True, null=True)
    fecha_nacimiento = models.DateField()
    genero = models.CharField(max_length=10)
    celular = models.CharField(max_length=10)
    email = models.EmailField(max_length=50)
    direccion = models.CharField(max_length=50)
    ocupacion = models.CharField(max_length=50)
    tipo_sangre = models.ForeignKey(TipoSangre, on_delete=models.CASCADE)

    def __str__(self):
        return f"{self.nombre1} {self.apellido1}"
```

Figura 49. Archivo model parte 1



```python

# modelo antecedentes personales
class AntecedentesPersonales(models.Model):
    paciente = models.ForeignKey(Paciente, on_delete=models.CASCADE)
    descripcion = models.CharField(max_length=255)

# modelo antecedentes familiares
class AntecedentesFamiliares(models.Model):
    paciente = models.ForeignKey(Paciente, on_delete=models.CASCADE)
    descripcion = models.CharField(max_length=255)
    relacion = models.CharField(max_length=50)

# modelo alergias
class Alergias(models.Model):
    paciente = models.ForeignKey(Paciente, on_delete=models.CASCADE)
    descripcion = models.CharField(max_length=255)

# modelo traumas
class Traumas(models.Model):
    paciente = models.ForeignKey(Paciente, on_delete=models.CASCADE)
    descripcion = models.CharField(max_length=255)

# modelo habitso
class Habitos(models.Model):
    paciente = models.ForeignKey(Paciente, on_delete=models.CASCADE)
    tipo = models.CharField(max_length=50)
    frecuencia = models.CharField(max_length=50)
    descripcion = models.CharField(max_length=255, blank=True, null=True)

# modelo antecedentes clinicos
class AntecedenteClinico(models.Model):
    paciente = models.ForeignKey(Paciente, on_delete=models.CASCADE)
    descripcion = models.CharField(max_length=255)
    fecha = models.DateTimeField(blank=True, null=True)

# modelo tipos de enfermedad
class TipoEnfermedad(models.Model):
    codigo = models.CharField(max_length=50)
    tipo = models.CharField(max_length=255)

    def __str__(self):
        return self.tipo

# modelo enfermedad
class Enfermedad(models.Model):
    codigo = models.CharField(max_length=50)
    nombre = models.CharField(max_length=255)
    tipo_enfermedad = models.ForeignKey(TipoEnfermedad, on_delete=models.CASCADE)

    def __str__(self):
        return self.nombre

```

Figura 50. Archivo model parte 2



```python

# modelo Registro medico
class RegistroMedico(models.Model):
    paciente = models.ForeignKey(Paciente, on_delete=models.CASCADE)
    medico = models.ForeignKey(Medico, on_delete=models.CASCADE)
    fecha_registro_medico = models.DateTimeField()
    razon = models.CharField(max_length=255)

# modelo signos vitales
class SignosVitales(models.Model):
    registro_medico = models.OneToOneField(RegistroMedico, on_delete=models.CASCADE)
    presion = models.CharField(max_length=7, blank=True, null=True)
    temperatura = models.DecimalField(max_digits=4, decimal_places=2, blank=True, null=
True)
    talla = models.DecimalField(max_digits=4, decimal_places=1, blank=True, null=True)
    peso = models.DecimalField(max_digits=4, decimal_places=1, blank=True, null=True)
    imc = models.DecimalField(max_digits=4, decimal_places=1, blank=True, null=True)

# modelo diagnostico
class Diagnostico(models.Model):
    registro_medico = models.ForeignKey(RegistroMedico, on_delete=models.CASCADE)
    enfermedad = models.ForeignKey(Enfermedad, on_delete=models.CASCADE)
    descripcion = models.CharField(max_length=255, blank=True, null=True)
    fecha_diagnostico = models.DateTimeField()

# modelo medicamento
class Medicamento(models.Model):
    codigo = models.CharField(max_length=10)
    nombre = models.CharField(max_length=255)
    forma_farmaceutica = models.CharField(max_length=255)
    concentracion = models.TextField(blank=True, null=True)

    def __str__(self):
        return self.nombre

# modelo prescripcion medica
class PrescripcionMedica(models.Model):
    registro_medico = models.ForeignKey(RegistroMedico, on_delete=models.CASCADE)
    medicamento = models.ForeignKey(Medicamento, on_delete=models.CASCADE)
    dosis = models.CharField(max_length=50)
    frecuencia = models.CharField(max_length=50, blank=True, null=True)
    duracion = models.CharField(max_length=50, blank=True, null=True)
    instrucciones = models.CharField(max_length=255, blank=True, null=True)

# modelo envio prescripcion medica
class EnvioPrescripcion(models.Model):
    prescripcion = models.OneToOneField(PrescripcionMedica, on_delete=models.CASCADE)
    metodo_envio = models.CharField(max_length=50, blank=True, null=True)
    estado_envio = models.CharField(max_length=1, blank=True, null=True)
    fecha_envio = models.DateTimeField(blank=True, null=True)
```

Figura 51. Archivo model parte 3

Al tener listo el model para la creación de la base de datos se procede a migrar los cambios realizados para la creación de las tablas.



Figura 52. Tablas de la base de datos generada por models

Django como es un framework robusto, tiene la opción de administrar datos de manera de formulario y para acceder a esos privilegios se codificar el archivo admi.py



Figura 53. Archivo admin.py para la administración de datos con Django

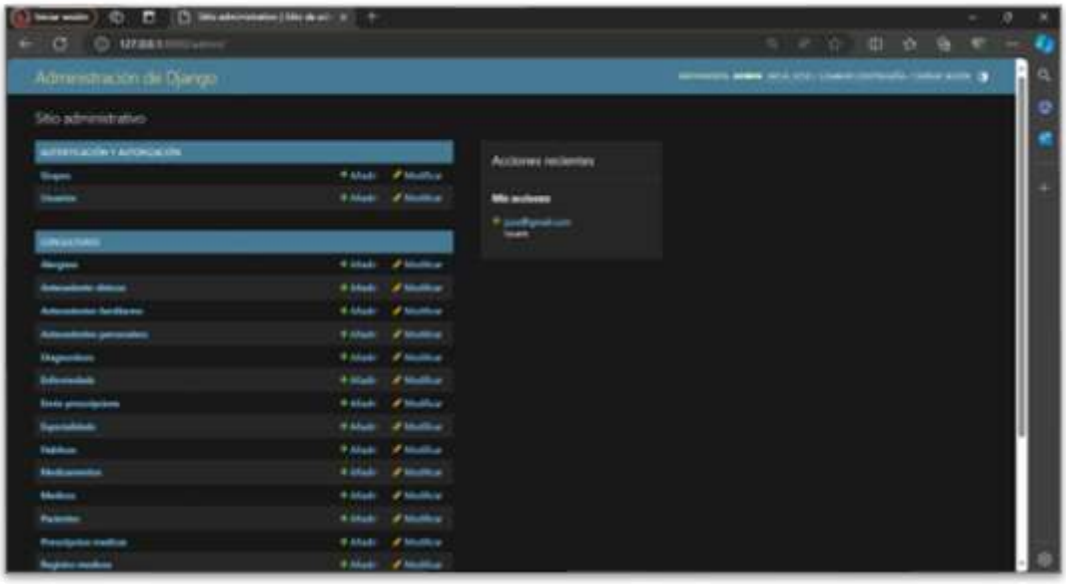


Figura 54. Ventana administrativa de Django

c. Desarrollo del login de autenticación

Al momento de ejecutar el aplicativo la primera página que se visualiza en la de la autenticación, para acceder al aplicativo se debe ingresar las credenciales que son el correo y la contraseña. Para el módulo de autenticación se creó un templete en formato HTML que permita el ingreso de las credenciales.

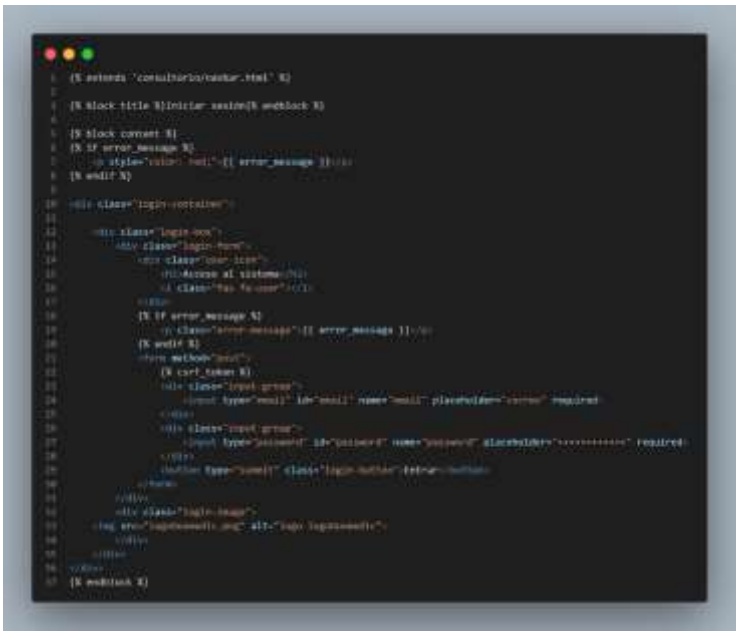


Figura 55. Template de autenticación de usuario

Para verificar que los datos del usuario sean correctos se realizó una validación, en la cual se generó un método login de solicitud de tipo POST, se obtiene el correo electrónico y la contraseña del formulario y si las credenciales son correctas, inicia la sesión y redirige a la página 'pacientes'. Si las credenciales son incorrectas, muestra un mensaje de error en la misma página de inicio de sesión.



```python
def login_view(request):
    if request.method == 'POST':
        email = request.POST.get('email')
        password = request.POST.get('password')
        user = authenticate(request, username=email, password=password)
        if user is not None:
            login(request, user)
            return redirect('pacientes')
        else:
            error_message = "Credenciales inválidas"
            return render(request, 'consultorio/login.html', {'error_message': error_message})
    return render(request, 'consultorio/login.html')
```

Figura 56. Método de autenticación login_view

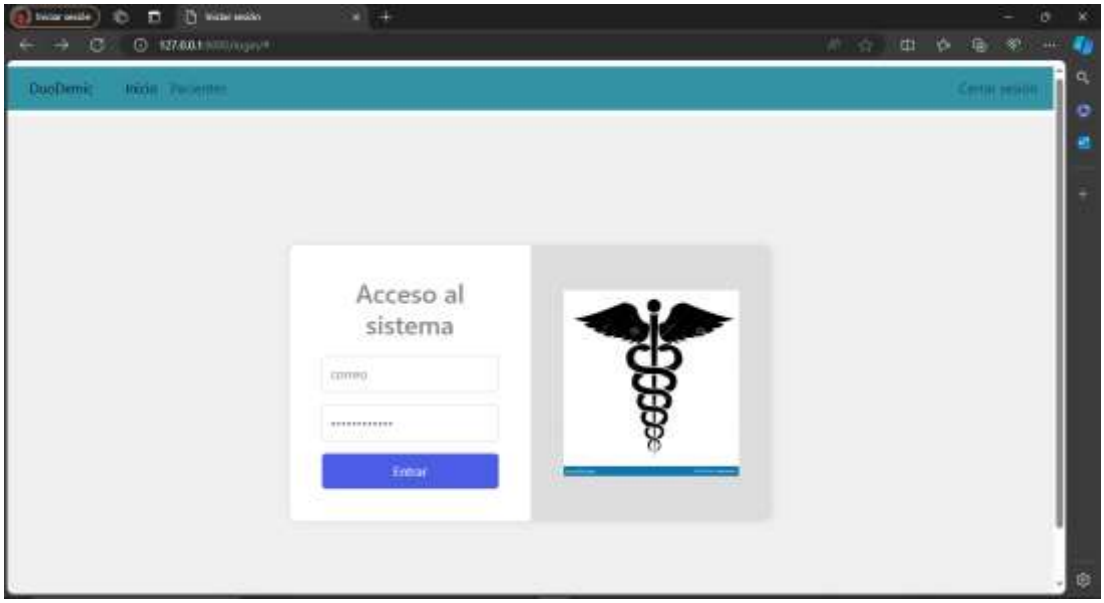


Figura 57. Página de autenticación de usuario

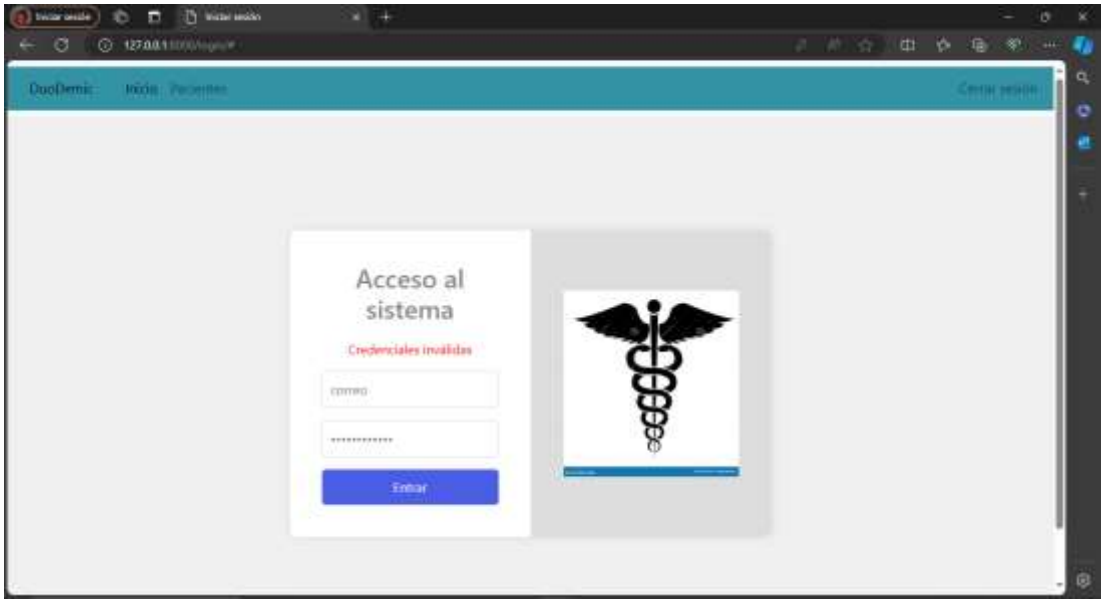


Figura 58. Mensaje de credenciales invalidad

- ***Creación del módulo de usuarios***

En este módulo se permitirá la gestión de la información de los médicos. En este apartado se puede realizar una búsqueda, inserción, manipulación y eliminación de datos.

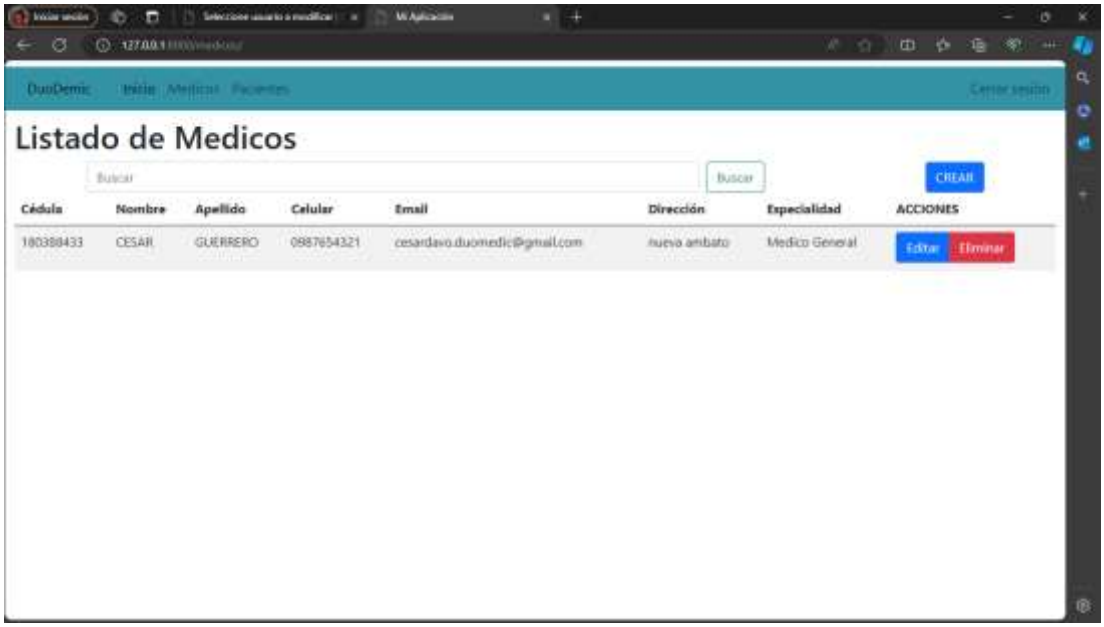


Figura 59. Listado de médicos registrados

Para realizar las operaciones de CRUD, se realizó la siguiente lógica utilizando la ORM para la manipulación de datos de los médicos. Cada view tiene su funcionalidad correspondiente.



```python

# listar medicos
def listar_medicos(request):
    if not request.user.is_authenticated:
        return redirect('login')
    medicos = Medico.objects.all()
    return render(request, 'consultorio/medicos.html',{'medicos': medicos})

# agregar medico
def agregar_medico(request):
    if request.method == 'POST':
        form = MedicoForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('listar_medicos')
    else:
        form = MedicoForm()
    return render(request, 'consultorio/agregar_medico.html', {'form': form})

# editar medico
def editar_medico(request, pk):
    medico = get_object_or_404(Medico, pk=pk)
    if request.method == 'POST':
        form = MedicoForm(request.POST, instance=medico)
        if form.is_valid():
            form.save()
            return redirect('medicos')
    else:
        form = MedicoForm(instance=medico)
    return render(request, 'consultorio/editar_medico.html', {'form': form})

# eliminar medico
def eliminar_medico(request, pk):
    medico = get_object_or_404(Medico, pk=pk)
    if request.method == 'POST':
        medico.delete()
        return redirect('medicos')
    return render(request, 'consultorio/confirmar_eliminar_medico.html', {'medico': medico})
```

Figura 60. Métodos CRUD para el módulo de usuarios

Para la gestión de cada CRUD se creó templates específicos para acción que el usuario administrador va a realizar.

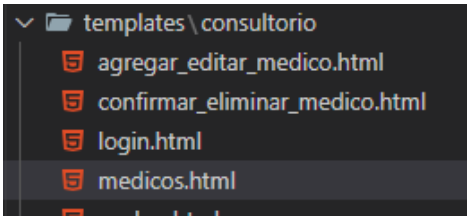


Figura 61. Templates del módulo usuarios

Después de definen las rutas o paths de las vistas puedan establecer la conectividad del modelo MVT.



```
urlpatterns = [
    path('medicos/', views.listar_medicos, name='medicos'),
    path('medicos/agregar/', views.agregar_medico, name='agregar_medico'),
    path('medicos/editar/<int:pk>/', views.editar_medico, name='editar_medico'),
    path('medicos/eliminar/<int:pk>/', views.eliminar_medico, name='eliminar_medico'),
]
```

Figura 62. Rutas del módulo de usuarios

El formulario de editar médico permite modificar los datos de los médicos existentes en el sistema. Esta es una función útil para mantener actualizada y precisa la información de los médicos.

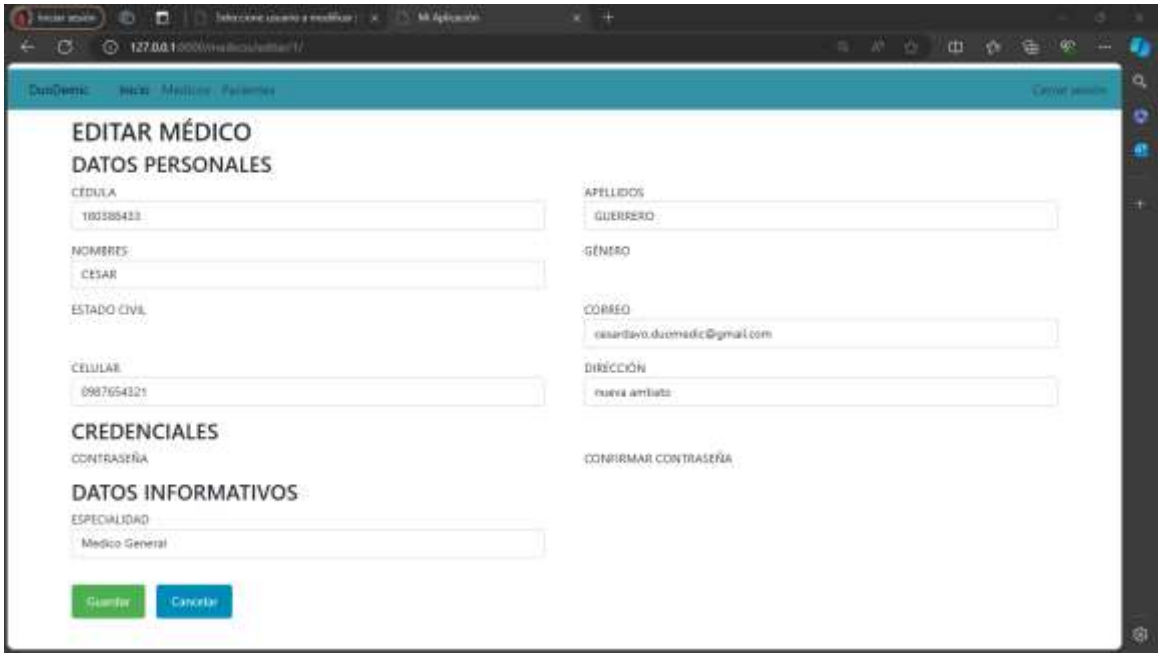


Figura 63. Editar los datos del médico

La funcionalidad de Eliminar Médico permite a los usuarios con los permisos adecuados eliminar el registro de un médico del sistema. Esta opción se puede utilizar cuando un médico ya no forma parte del personal o si se ha ingresado información errónea.

Una vez que se confirma la eliminación, el registro del médico se eliminará de manera permanente del sistema. Es importante tener en cuenta que esta acción no se puede deshacer, por lo que se debe proceder con precaución. La funcionalidad de eliminar médico ayuda a mantener la base de datos del personal médico limpia y actualizada.

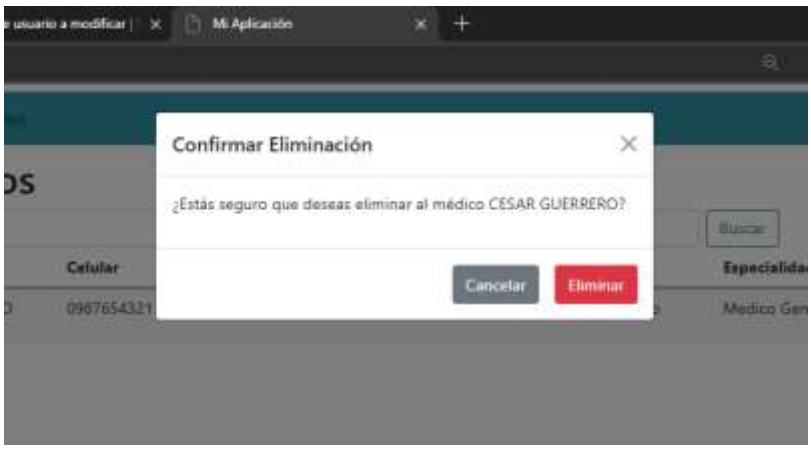


Figura 64. Eliminación de médico

La funcionalidad de Agregar Médico permite registrar a nuevos médicos en el sistema. Esta opción se utiliza cuando se contrata a un nuevo miembro del personal médico o cuando se necesita ingresar la información de un médico que no está en la base de datos.

Los usuarios deben completar todos los campos obligatorios para poder guardar la información del nuevo médico en el sistema.

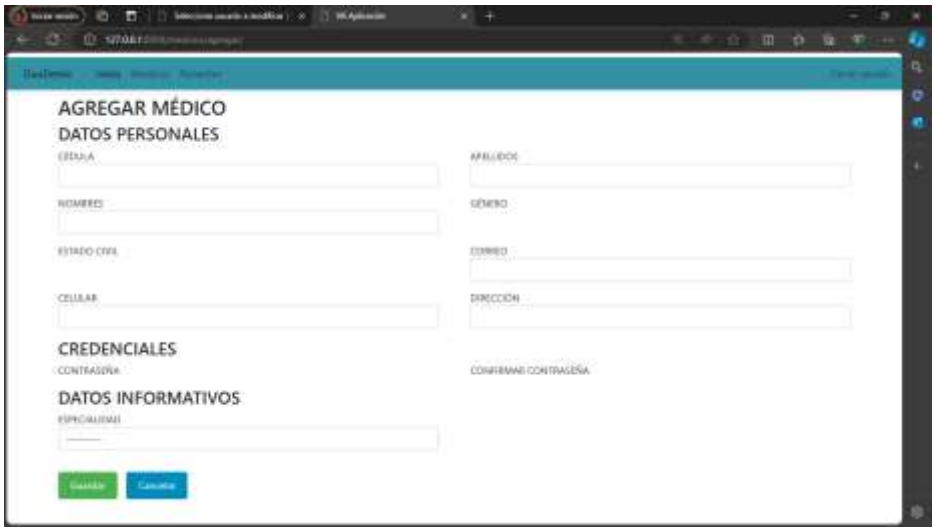


Figura 65. Crear un nuevo médico

El template utilizado para crear un nuevo médico fue el mimo que el editar, **Figura 63** ya que se establece los mismos atributos y de esta manera reutilizamos código.

- ***Creación del módulo de pacientes***

En este módulo se permitirá la gestión de la información de los pacientes. En este apartado se puede realizar una búsqueda, inserción, manipulación y eliminación de datos. Si un paciente no se encuentra en la base de datos se procesa a la creación de la ficha médica del paciente.

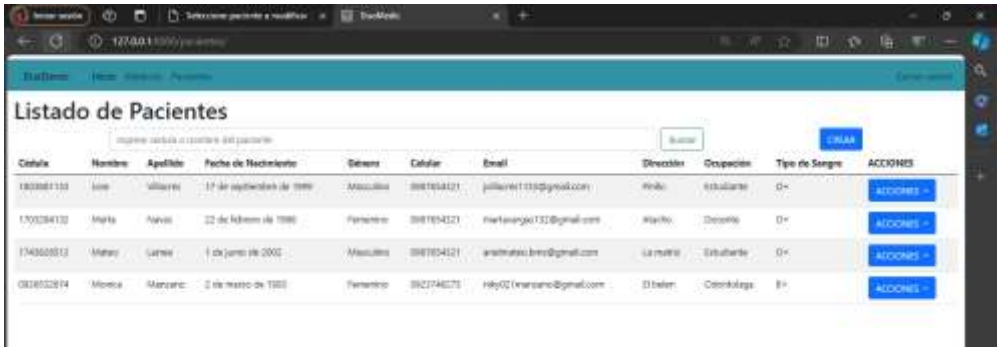


Figura 66. Listado de pacientes registrados



```python
# listar pacientes
def listar_pacientes(request):
    if not request.user.is_authenticated:
        return redirect('login')
    pacientes = Paciente.objects.all()
    return render(request, 'consultorio/pacientes.html',{'pacientes': pacientes})

# crear paciente
def crear_paciente(request):
    if request.method == 'POST':
        form = PacienteForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('lista_pacientes')
    else:
        form = PacienteForm()
    return render(request, 'consultorio/paciente_form.html', {'form': form})

# crear modificar
def modificar_paciente(request, pk):
    paciente = get_object_or_404(Paciente, pk=pk)
    if request.method == 'POST':
        form = PacienteForm(request.POST, instance=paciente)
        if form.is_valid():
            form.save()
            return redirect('lista_pacientes')
    else:
        form = PacienteForm(instance=paciente)
    return render(request, 'consultorio/paciente_form.html', {'form': form})

# crear eliminar
def eliminar_paciente(request, pk):
    paciente = get_object_or_404(Paciente, pk=pk)
    if request.method == 'POST':
        paciente.delete()
        return redirect('lista_pacientes')
    return render(request, 'consultorio/paciente_eliminar.html', {'paciente': paciente})
```

Figura 67. Métodos CRUD para el módulo de paciente

Para la gestión de cada CRUD de paciente se creó templates específicos para acción que el usuario administrador va a realizar.

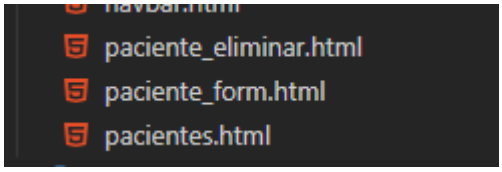


Figura 68. Templates del módulo paciente

Después se establecen los paths para establecer la conexión entre las vistas, templates y modelos.



Figura 69. Rutas del módulo de pacientes

La funcionalidad de Agregar Paciente permite a los médicos registrar a nuevos pacientes en el sistema. Esto se utiliza cuando se recibe un nuevo paciente en el consultorio.

Una vez completados todos los campos obligatorios, el usuario puede guardar la información del nuevo paciente en el sistema. Estas funcionalidades de gestión de pacientes son esenciales para mantener actualizada y organizada la información de los pacientes en la aplicación

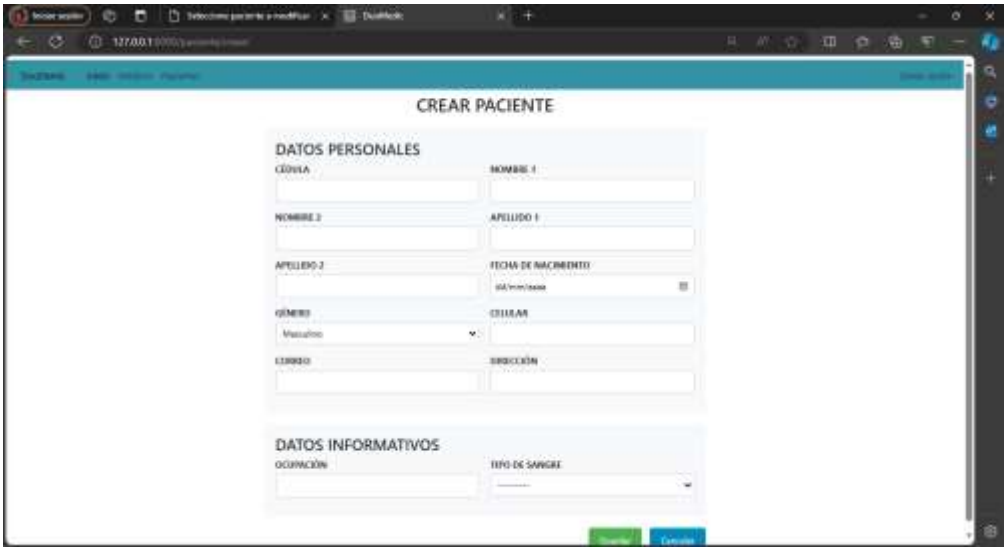


Figura 70. Crear un nuevo paciente

El formulario de Editar Paciente permite a los médicos modificar los detalles de los pacientes existentes en el sistema. Esto es útil para mantener actualizada y precisa la información de los pacientes, como cambios en sus datos de contacto o un dato registrado erróneo anteriormente.

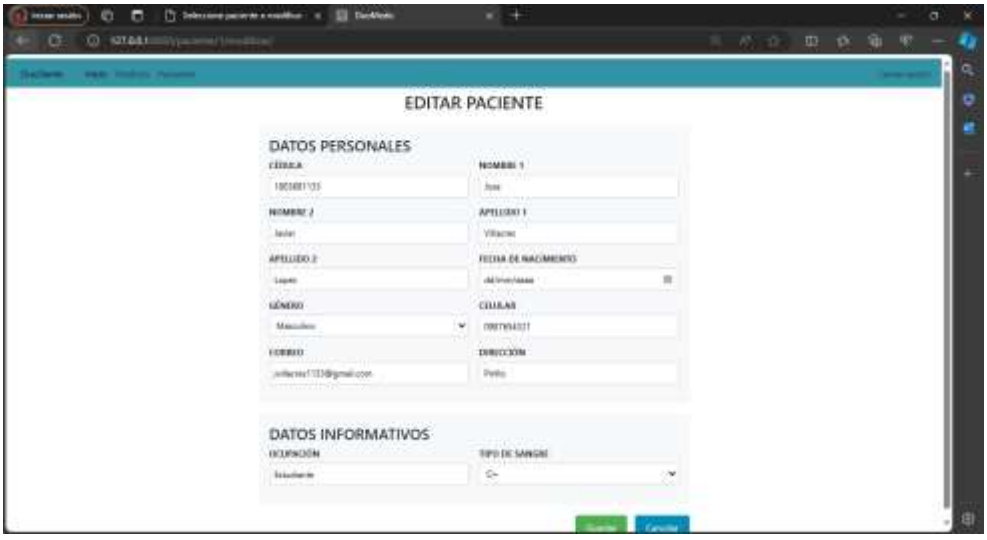


Figura 71. Editar un paciente

La funcionalidad de eliminar paciente, permite al médico eliminar el registro de un paciente del sistema. Esto puede ser necesario cuando un paciente ya no forma parte de la clínica o si se ha ingresado información errónea.

Una vez confirmada la eliminación, el registro del paciente se eliminará de manera permanente del sistema. Es importante tener en cuenta que esta acción no se puede deshacer.

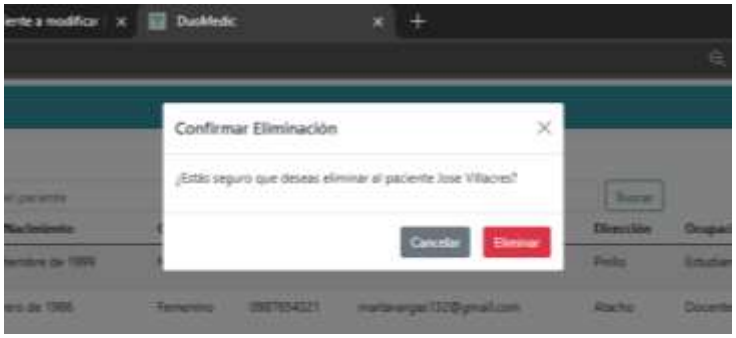


Figura 72. Eliminar un paciente

d. Registro de ficha médica

La ficha médica del paciente recopila información detallada de los datos personales y clínicos.

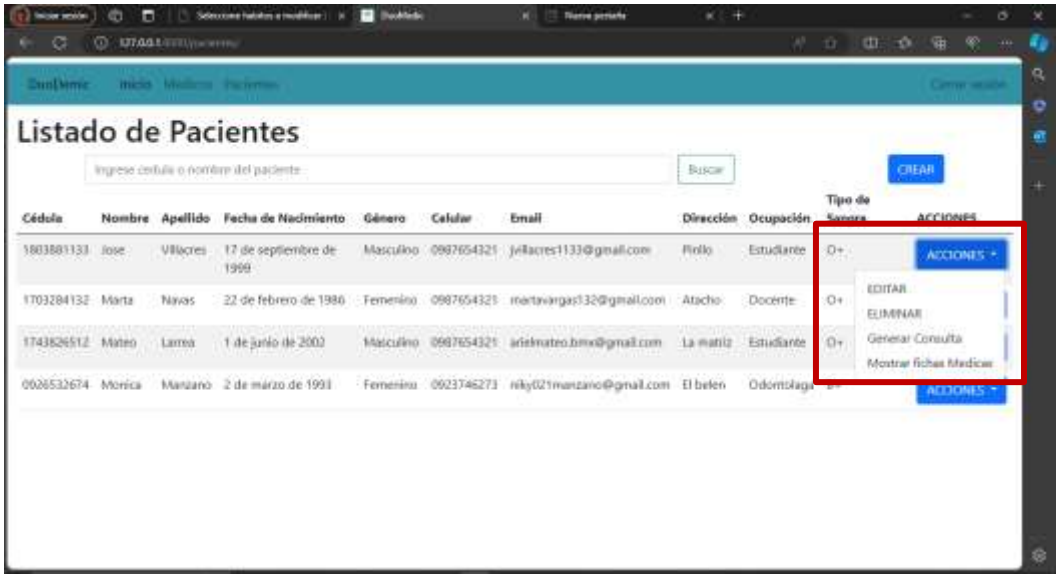


Figura 73 Acción mostrar ficha médica

La ficha médica del paciente recopila información detallada de los datos personales, incluyendo sus datos básicos como nombre completo, fecha de nacimiento, sexo, información de contacto y cédula de ciudadanía ecuatoriana. Además, contiene un registro de sus antecedentes médicos, como enfermedades previas, cirugías, hospitalizaciones, medicamentos recetados y esquema de vacunación.



Figura 74. Visualización de la ficha médica de un paciente existente parte 1

Los antecedentes familiares también se documentan, abarcando enfermedades hereditarias, historial de cáncer, diabetes, problemas cardiovasculares y trastornos mentales.

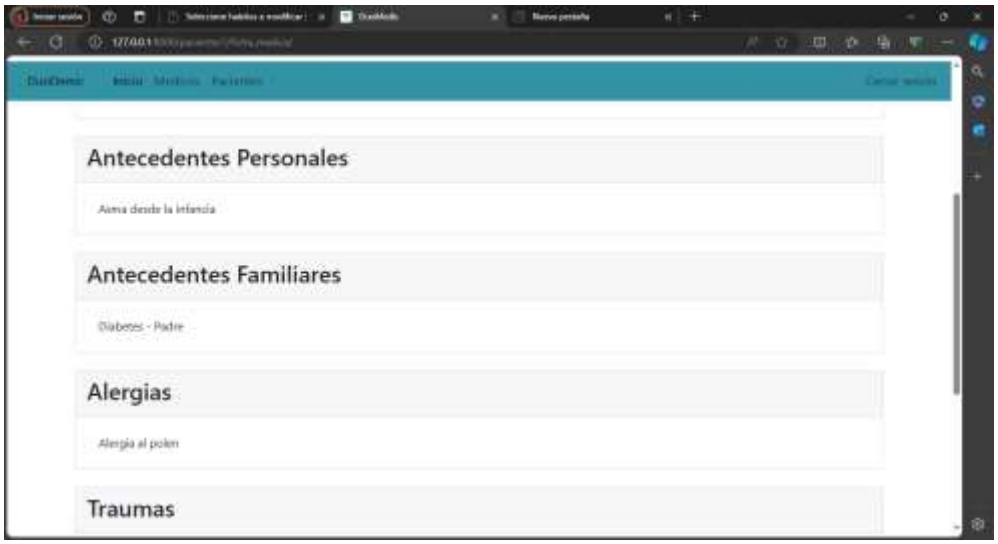


Figura 75. Visualización de la ficha médica de un paciente existente parte 2

La ficha también incluye un historial de traumas sufridos, como accidentes, lesiones, fracturas y traumatismos craneoencefálicos. Se detalla información sobre los hábitos y el estilo de vida del paciente, como consumo de tabaco y alcohol, nivel de actividad física, dieta y nutrición, y estado de salud mental y emocional. Esta información permite al personal médico contar con un panorama integral del estado de salud y las necesidades del paciente, facilitando una atención más personalizada y efectiva.

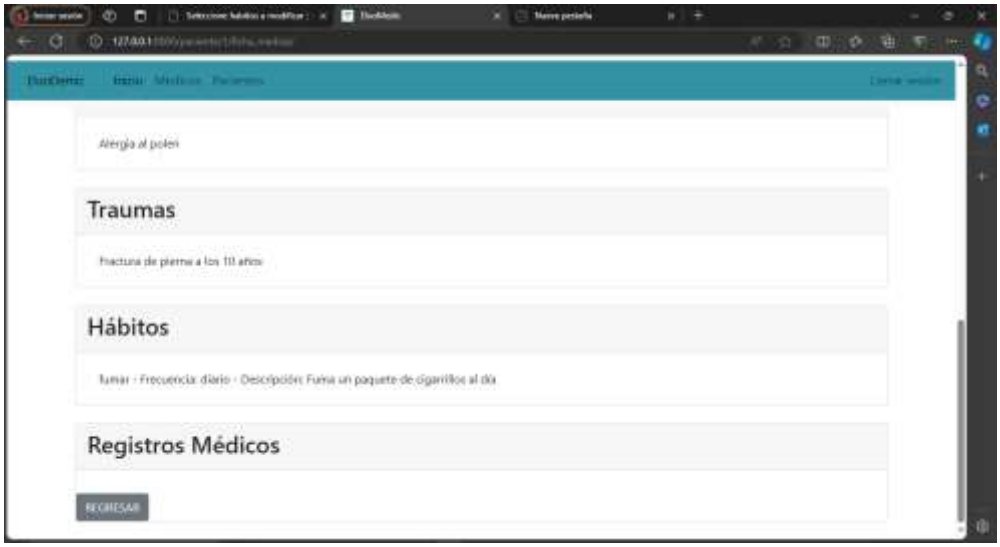


Figura 76. Visualización de la ficha médica de un paciente existente parte 3

```python
# Ficha medica
def ficha_medica(request, paciente_id):
    paciente = Paciente.objects.get(id=paciente_id)
    antecedentes_personales = AntecedentesPersonales.objects.filter(paciente=paciente)
    antecedentes_familiares = AntecedentesFamiliares.objects.filter(paciente=paciente)
    alergias = Alergias.objects.filter(paciente=paciente)
    traumas = Traumas.objects.filter(paciente=paciente)
    habitos = Habitos.objects.filter(paciente=paciente)
    registros_medicos = RegistroMedico.objects.filter(paciente=paciente)
    diagnosticos = Diagnostico.objects.filter(registro_medico__paciente=paciente)
    prescripciones = PrescripcionMedica.objects.filter(registro_medico__paciente=paciente)

    context = {
        'paciente': paciente,
        'antecedentes_personales': antecedentes_personales,
        'antecedentes_familiares': antecedentes_familiares,
        'alergias': alergias,
        'traumas': traumas,
        'habitos': habitos,
        'registros_medicos': registros_medicos,
        'diagnosticos': diagnosticos,
        'prescripciones': prescripciones,
    }
    return render(request, 'consultorio/ficha_medica.html', context)
```

Figura 77. Vista para listar fichas médicas por paciente

e. Registro de signos vitales y prescripción médica

La toma de los signos vitales es una parte integral de la evaluación médica. Esto implica registrar de manera sistemática parámetros como la temperatura, talla, peso, presión arterial.



Figura 78. Registro de signos vitales

El diagnóstico médico es la determinación de la enfermedad, lesión o condición que afecta al paciente. Este apartado cuenta con un listado de todas las enfermedades hasta la actualidad acorde a la norma de código y nomenclatura (CIE-11).

Las nomenclaturas (CIE-11) se la obtuvo de la página oficial Organización Mundial de la Salud [35].

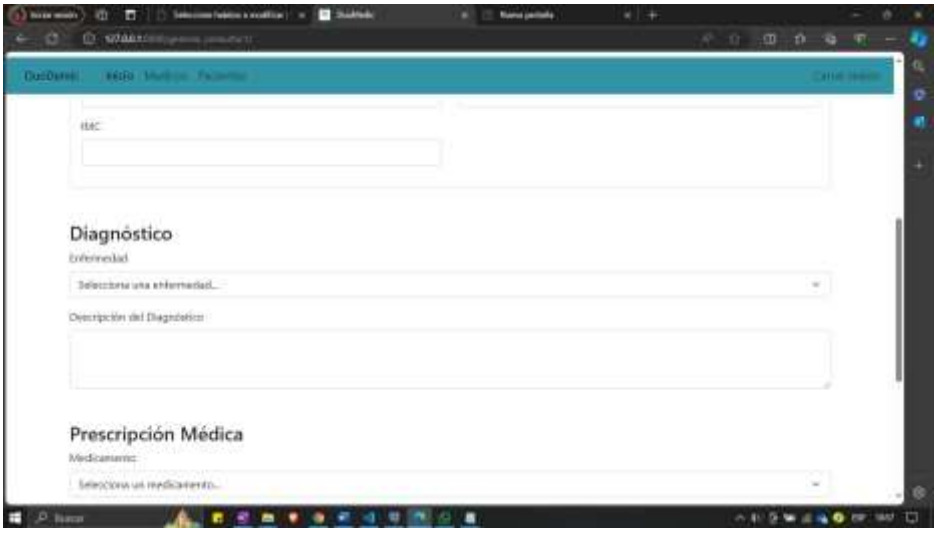


Figura 79. Registro de los diagnósticos médicos

Gestión médica donde los profesionales de la salud pueden ingresar detalles de prescripciones para los pacientes. Este apartado selecciona el medicamento acorte al código y nomenclatura (ATC/DDD) 2023.

El listado se lo obtuvieron de página de la Consejo Nacional de Salud CONASA 2024 (CONASA) del libro: Libro-Cuadro-Medicamentos-Basicos-11a-revision [36].

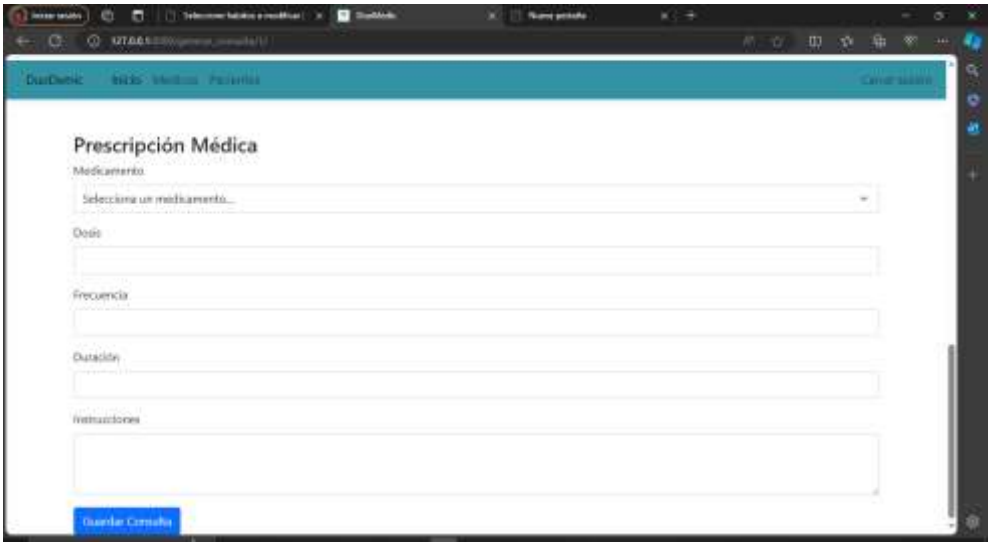
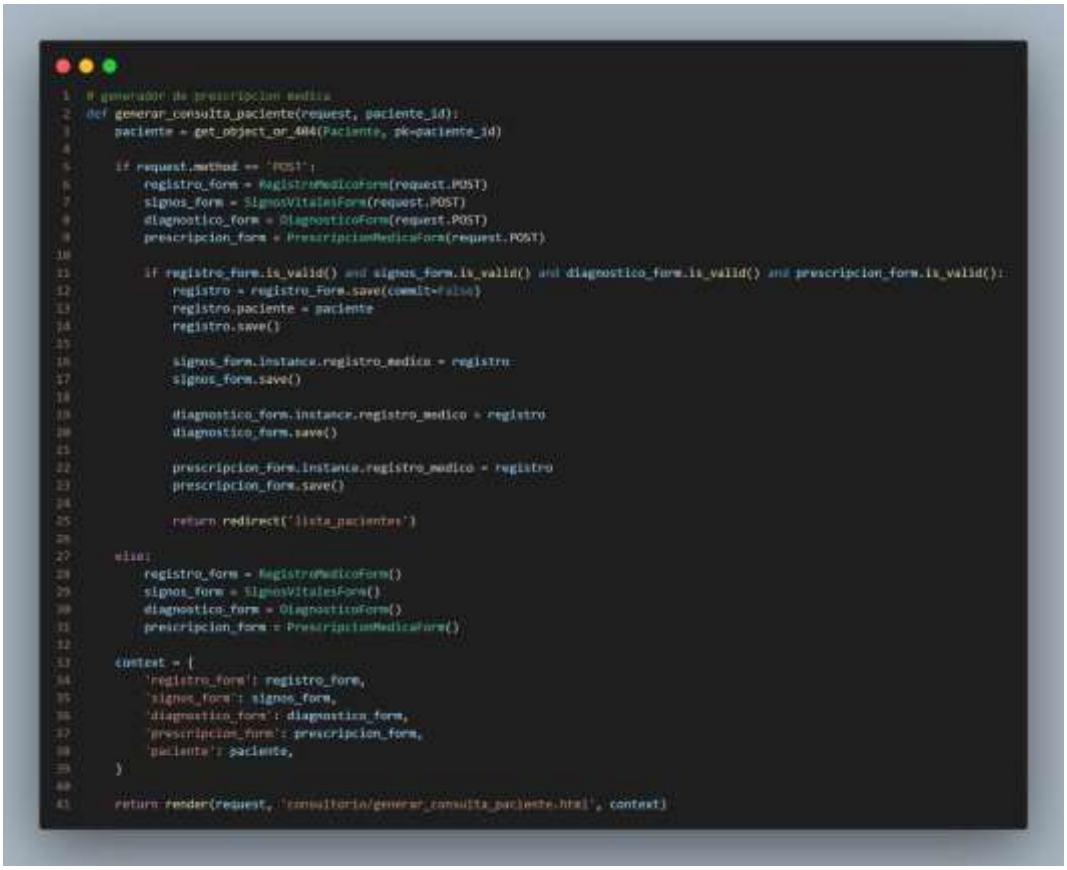


Figura 80. Registro las prescripciones médicas

Lógica para el registro y validación de la presión médica acorde a los entandares establecidos.



```python
# generador de prescripcion medica
def generar_consulta_paciente(request, paciente_id):
    paciente = get_object_or_404(Paciente, pk=paciente_id)

    if request.method == 'POST':
        registro_form = RegistroMedicoForm(request.POST)
        signos_form = SignosVitalesForm(request.POST)
        diagnostico_form = DiagnosticoForm(request.POST)
        prescripcion_form = PrescripcionMedicaForm(request.POST)

        if registro_form.is_valid() and signos_form.is_valid() and diagnostico_form.is_valid() and prescripcion_form.is_valid():
            registro = registro_form.save(commit=False)
            registro.paciente = paciente
            registro.save()

            signos_form.instance.registro_medico = registro
            signos_form.save()

            diagnostico_form.instance.registro_medico = registro
            diagnostico_form.save()

            prescripcion_form.instance.registro_medico = registro
            prescripcion_form.save()

            return redirect('lista_pacientes')

    else:
        registro_form = RegistroMedicoForm()
        signos_form = SignosVitalesForm()
        diagnostico_form = DiagnosticoForm()
        prescripcion_form = PrescripcionMedicaForm()

    context = {
        'registro_form': registro_form,
        'signos_form': signos_form,
        'diagnostico_form': diagnostico_form,
        'prescripcion_form': prescripcion_form,
        'paciente': paciente,
    }

    return render(request, 'consultorio/generar_consulta_paciente.html', context)
```

Figura 81. Registro de prescripción médica

f. Envío de prescripción médica

Para en el envío de la prescripción médica a través del correo electrónico se debe configurar el archivo setting.py colocando las credencias del correo host, el puerto SMTP y la librería propia de Django para el envío de correo electrónico.

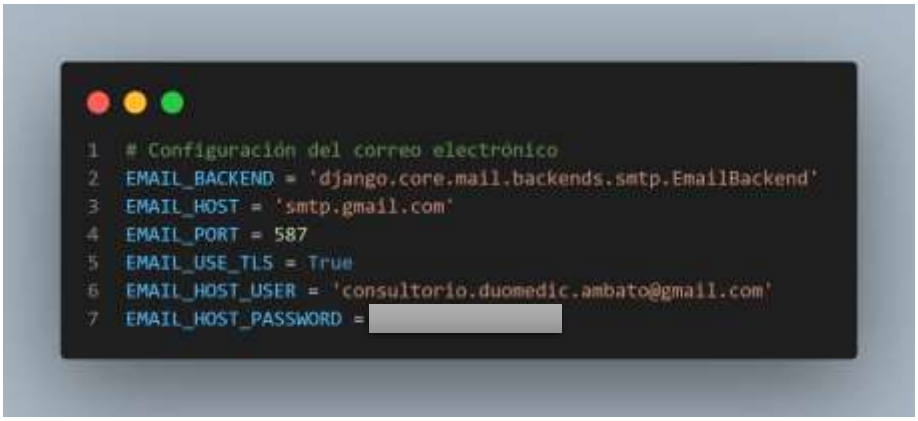
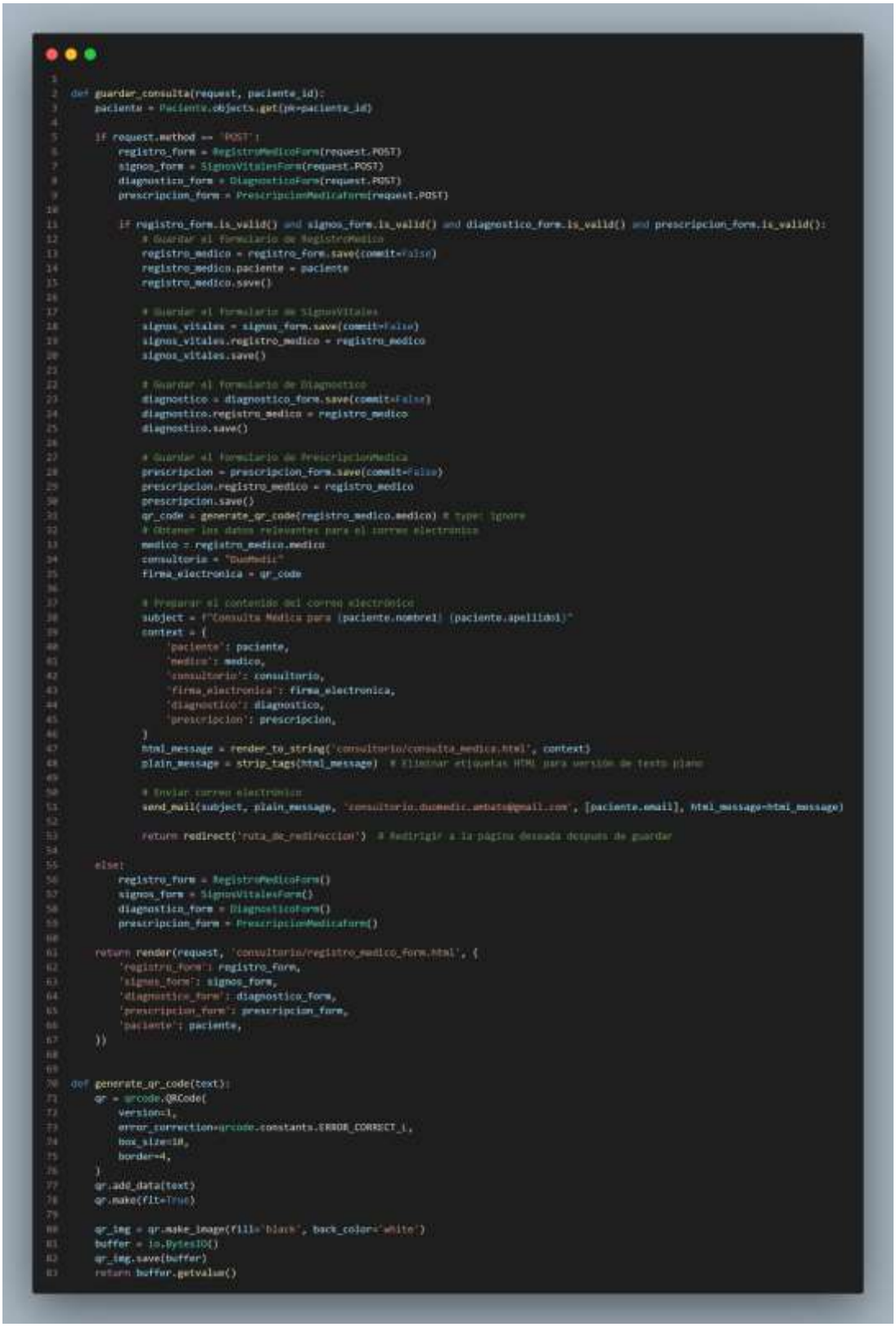


Figura 82. Configuración correo electrónico



```python

def guardar_consulta(request, paciente_id):
    paciente = Paciente.objects.get(pk=paciente_id)

    if request.method == 'POST':
        registro_form = RegistroMedicoForm(request.POST)
        signos_form = SignosVitalesForm(request.POST)
        diagnostico_form = DiagnosticoForm(request.POST)
        prescripcion_form = PrescripcionMedicaForm(request.POST)

        if registro_form.is_valid() and signos_form.is_valid() and diagnostico_form.is_valid() and prescripcion_form.is_valid():
            # Guardar el formulario de RegistroMedico
            registro_medico = registro_form.save(commit=False)
            registro_medico.paciente = paciente
            registro_medico.save()

            # Guardar el formulario de SignosVitales
            signos_vitales = signos_form.save(commit=False)
            signos_vitales.registro_medico = registro_medico
            signos_vitales.save()

            # Guardar el formulario de Diagnostico
            diagnostico = diagnostico_form.save(commit=False)
            diagnostico.registro_medico = registro_medico
            diagnostico.save()

            # Guardar el formulario de PrescripcionMedica
            prescripcion = prescripcion_form.save(commit=False)
            prescripcion.registro_medico = registro_medico
            prescripcion.save()
            qr_code = generate_qr_code(registro_medico.medico) # type: ignore
            # Obtener los datos relevantes para el correo electrónico
            medico = registro_medico.medico
            consultorio = "DuoMedic"
            firma_electronica = qr_code

            # Preparar el contenido del correo electrónico
            subject = f"Consulta Médica para {paciente.nombre1} {paciente.apellido1}"
            context = {
                'paciente': paciente,
                'medico': medico,
                'consultorio': consultorio,
                'firma_electronica': firma_electronica,
                'diagnostico': diagnostico,
                'prescripcion': prescripcion,
            }
            html_message = render_to_string('consultorio/consulta_medica.html', context)
            plain_message = strip_tags(html_message)  # Eliminar etiquetas HTML para versión de texto plano

            # Enviar correo electrónico
            send_mail(subject, plain_message, 'consultorio.duomedic.ambato@gmail.com', [paciente.email], html_message=html_message)

            return redirect('ruta_de_redireccion')  # Redirigir a la página deseada después de guardar

    else:
        registro_form = RegistroMedicoForm()
        signos_form = SignosVitalesForm()
        diagnostico_form = DiagnosticoForm()
        prescripcion_form = PrescripcionMedicaForm()

    return render(request, 'consultorio/registro_medico_form.html', {
        'registro_form': registro_form,
        'signos_form': signos_form,
        'diagnostico_form': diagnostico_form,
        'prescripcion_form': prescripcion_form,
        'paciente': paciente,
    })


def generate_qr_code(text):
    qr = qrcode.QRCode(
        version=1,
        error_correction=qrcode.constants.ERROR_CORRECT_L,
        box_size=10,
        border=4,
    )
    qr.add_data(text)
    qr.make(fit=True)

    qr_img = qr.make_image(fill='black', back_color='white')
    buffer = io.BytesIO()
    qr_img.save(buffer)
    return buffer.getvalue()
```

Figura 83. Vista para el envío de correo electrónico

g. Módulo de reportes

Este apartado permite recopilar, analizar y presentar información relevante sobre la gestión. Este apartado permite mostrar los datos de los pacientes de manera grafica.

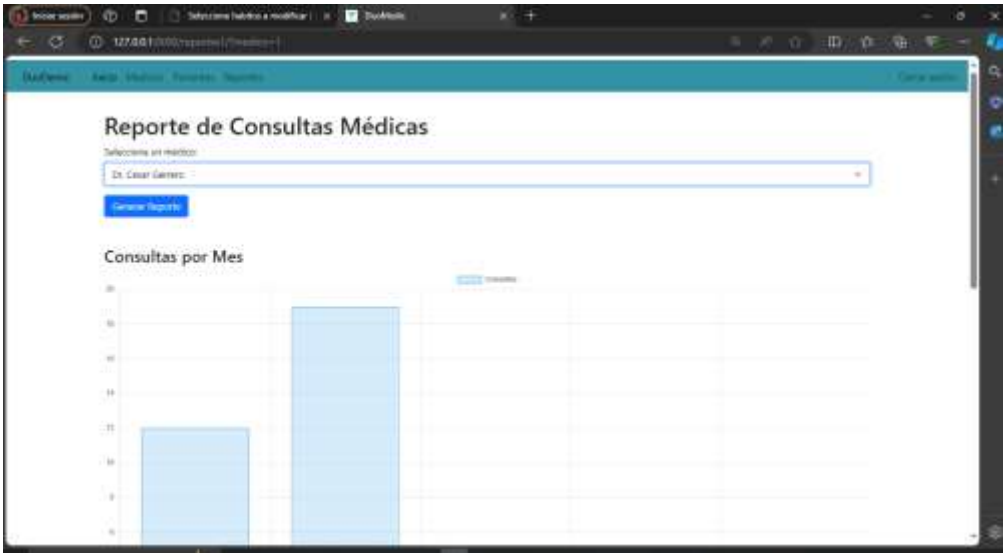


Figura 84. Reporte de consultas por mes

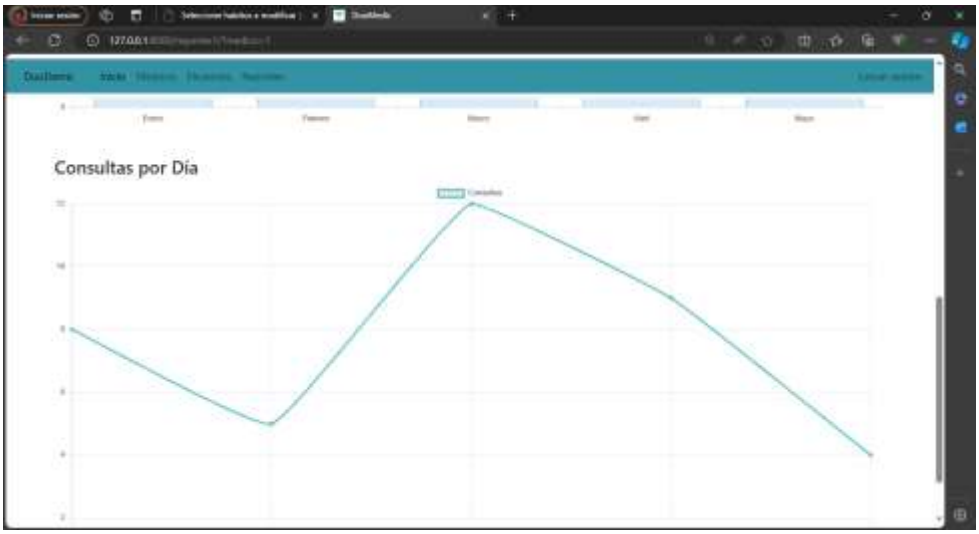


Figura 85.Reporte de consultas por día



```python
def reporte_consultas(request):
    # Obtener todos los médicos para el combo box
    medicos = Medico.objects.all()

    # Inicializar variables para gráficos
    datos_consultas_por_mes = []
    datos_consultas_por_dia = []

    # Si se ha enviado el formulario (seleccionado un médico)
    if 'medico' in request.GET:
        medico_id = request.GET['medico']

        # Obtener consultas por mes para el médico seleccionado
        consultas_por_mes = (
            RegistroMedico.objects
            .filter(medico_id=medico_id)
            .annotate(mes=Count('fecha_registro_medico__month'))
            .values('mes')
        )

        # Generar datos para el gráfico de consultas por mes
        datos_consultas_por_mes = [consulta['mes'] for consulta in consultas_por_mes]

        # Obtener consultas por día para el médico seleccionado (ejemplo ficticio)
        # En una aplicación real, esto debería ser más complejo y basado en datos reales
        datos_consultas_por_dia = [8, 5, 12, 9, 4]  # Datos ficticios para demostración

    context = {
        'medicos': medicos,
        'datos_consultas_por_mes': datos_consultas_por_mes,
        'datos_consultas_por_dia': datos_consultas_por_dia,
    }

    return render(request, 'consultorio/reporte_consultas.html', context)
```

Figura 86. Vista para visualizar los reportes

3.2.4 Fase IV: Pruebas

En esta fase se realizaron pruebas de aceptación para verificar el correcto funcionamiento de todas las funciones de la aplicación, asegurando un rendimiento óptimo en todos los aspectos del proyecto.

Tabla 64. Prueba de aceptación 1: Módulo de autenticación

Pruebas de aceptación	
Número: 1	**Historia de Usuario**:1
Nombre: Módulo de autenticación	
Descripción: Permite a los usuarios registrarse e iniciar sesión en la aplicación.	
Condiciones de ejecución: 1. Un usuario existente intenta iniciar sesión. 2. Se prueba el acceso con credenciales incorrectas.	

Entrada: Los usuarios pueden, iniciar sesión con credenciales correctas y se les niega el acceso con credenciales incorrectas.
Resultado esperado: Los usuarios pueden acceder a la aplicación con credenciales válidas.
Evaluación de prueba: Satisfactoria

Tabla 65. Prueba de aceptación 2: Módulo de usuarios

Pruebas de aceptación	
Número: 2	**Historia de Usuario**: 2
Nombre: Módulo de usuarios	
Descripción: Gestiona la información y permisos de los médicos del sistema.	
Condiciones de ejecución: El usuario médico es el administrador del sistema y l módulo de autenticación debe estar funcionando.	
Entrada: Un administrador o médico accede al módulo de usuarios y realiza operaciones CRUD.	
Resultado esperado: Se pueden crear, leer, actualizar y eliminar médicos	
Evaluación de prueba: Satisfactoria	

Tabla 66. Prueba de aceptación 3: Módulo de pacientes

Pruebas de aceptación	
Número: 3	**Historia de Usuario**: 3
Nombre: Módulo de pacientes	
Descripción: Permite realizar una búsqueda de pacientes, para después realizar una manipulación de los datos permitiendo realizar un CRUD.	
Condiciones de ejecución: El usuario médico debe estar logueo para el respectivo funcionamiento.	
Entrada: Un usuario autorizado accede al módulo de pacientes y para realizar la manipulación de datos.	
Resultado esperado: Se pueden realizar todas las operaciones de gestión de pacientes de manera efectiva y los datos se almacenan correctamente.	
Evaluación de prueba: Satisfactoria	

Tabla 67. Prueba de aceptación 4: Módulo de registro de ficha médica

Pruebas de aceptación	
Número: 4	**Historia de Usuario**: 4
Nombre: Módulo de registro de ficha médica	
Descripción: El médico puede visualizar todos los datos clínicos de un paciente, y si el paciente es nuevo en el consultorio procede a crear la ficha médica para el nuevo paciente	
Condiciones de ejecución: Autenticación de usuario funcional	
Entrada: 1. Un profesional médico selecciona un paciente. 2. Crea una nueva ficha médica o accede a una existente. 3. Ingresa o actualiza la información médica relevante. 4. Guarda los cambios en la ficha.	
Resultado esperado: Las fichas médicas se crean, actualizan y almacenan correctamente, asociadas al paciente correspondiente.	
Evaluación de prueba: Satisfactoria	

Tabla 68. Prueba de aceptación 5: Registro de signos vitales

Pruebas de aceptación	
Número: 5	**Historia de Usuario**: 5
Nombre: Registro de signos vitales	
Descripción: Registrar los signos vitales de los pacientes.	
Condiciones de ejecución: El módulo de ficha médica debe estar implementado.	
Entrada: 1. El medio tomas los signos vitales 1. Un profesional de salud selecciona un paciente. del sistema 2. Accede a la sección de signos vitales. 3. Ingresa nuevos datos de signos vitales.	
Resultado esperado: Los signos vitales se registran correctamente, se asocian a la ficha del paciente.	
Evaluación de prueba: Satisfactoria	

Tabla 69. Prueba de aceptación 6: Módulo de prescripción médica

Pruebas de aceptación	
Número: 6	**Historia de Usuario**: 6
Nombre: Módulo de prescripción médica	
Descripción: Permitir al médico crear y gestionar prescripciones para los pacientes.	
Condiciones de ejecución: Los módulos de ficha médica y pacientes deben estar operativos. Ficha médica debe estar abierta con los signos vitales ingresados.	
Entrada: 1. El médico basándose en los síntomas que presenta paciente, realiza el diagnostico. 2. Ingresa la posible enfermedad 3. Descripción detallada de la receta médica, detallando médicamento, dosis, duración y frecuencia. 4. Genera la prescripción	
Resultado esperado: Ingreso exitoso de la prescripción médica	
Evaluación de prueba: Satisfactoria	

Tabla 70. Prueba de aceptación 7: Envío de la prescripción médica por correo electrónica

Pruebas de aceptación	
Número: 7	**Historia de Usuario**: 7
Nombre: Envío de la prescripción médica por correo electrónico	
Descripción: Envió de prescripciones médicas al correo electrónico del paciente.	
Condiciones de ejecución: El módulo de prescripción médica debe estar implementado y se debe contar con un servicio de correo electrónico configurado.	
Entrada: Al finalizar la prescripción médica el sistema envía automáticamente al paciente al correo electrónico	
Resultado esperado: La prescripción se envía correctamente al correo electrónico del paciente en un formato legible y seguro.	
Evaluación de prueba: Satisfactoria	

Tabla 71. Prueba de aceptación 8: Modulo de reportes

Pruebas de aceptación	
Número: 8	**Historia de Usuario**: 8
Nombre: Módulo de reportes	
Descripción: Informes y estadísticas basados en los datos del sistema.	
Condiciones de ejecución: Todos los módulos anteriores deben estar funcionando y con datos suficientes.	
Entrada: en la sección de reportes se visualiza los Visualiza y/o exporta el reporte generado.	
Resultado esperado: El sistema genera reportes basados en los datos del sistema	
Evaluación de prueba: Satisfactoria	

3.2.5 Fase V: Implantación

Previo a los requisitos del sistema se estableció que el servidor de alojamiento será Heroku, en la cual es un servicio (PaaS) en la nube. Para poner a nivel de servidor de producción el backend se crea un proyecto en Heroku con la instalación de Heroku Postgres.

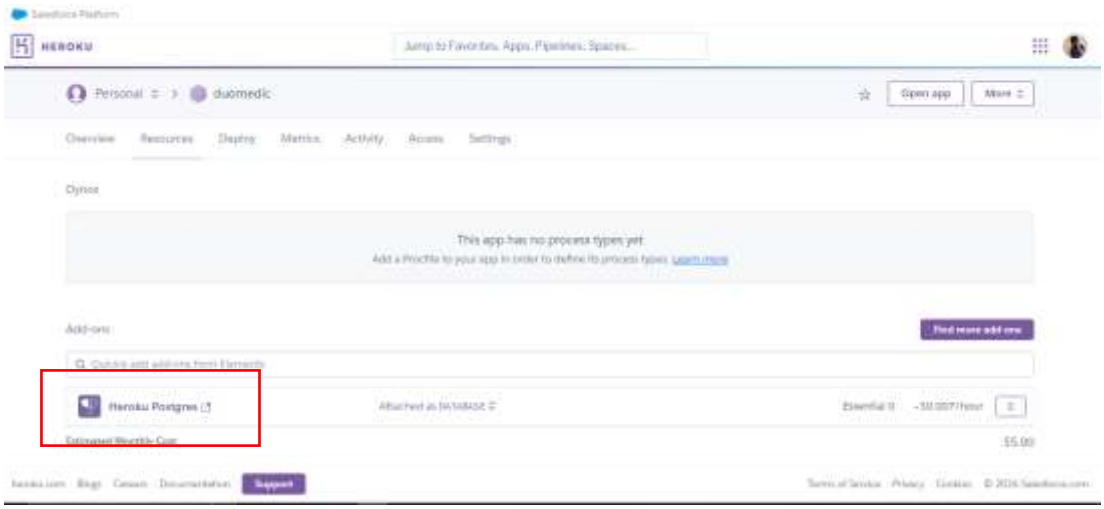


Figura 87. Proyecto backend en Heroku

Antes de desconectar la base de datos de PostgreSQL en Django de manera local, se realiza un Backup de la base por seguridad.

Posterior a ese se configura el archivo setting.py para establecer la conexión de del servidor al proyecto Django.



Figura 88. Credenciales de la base de datos de Heroku

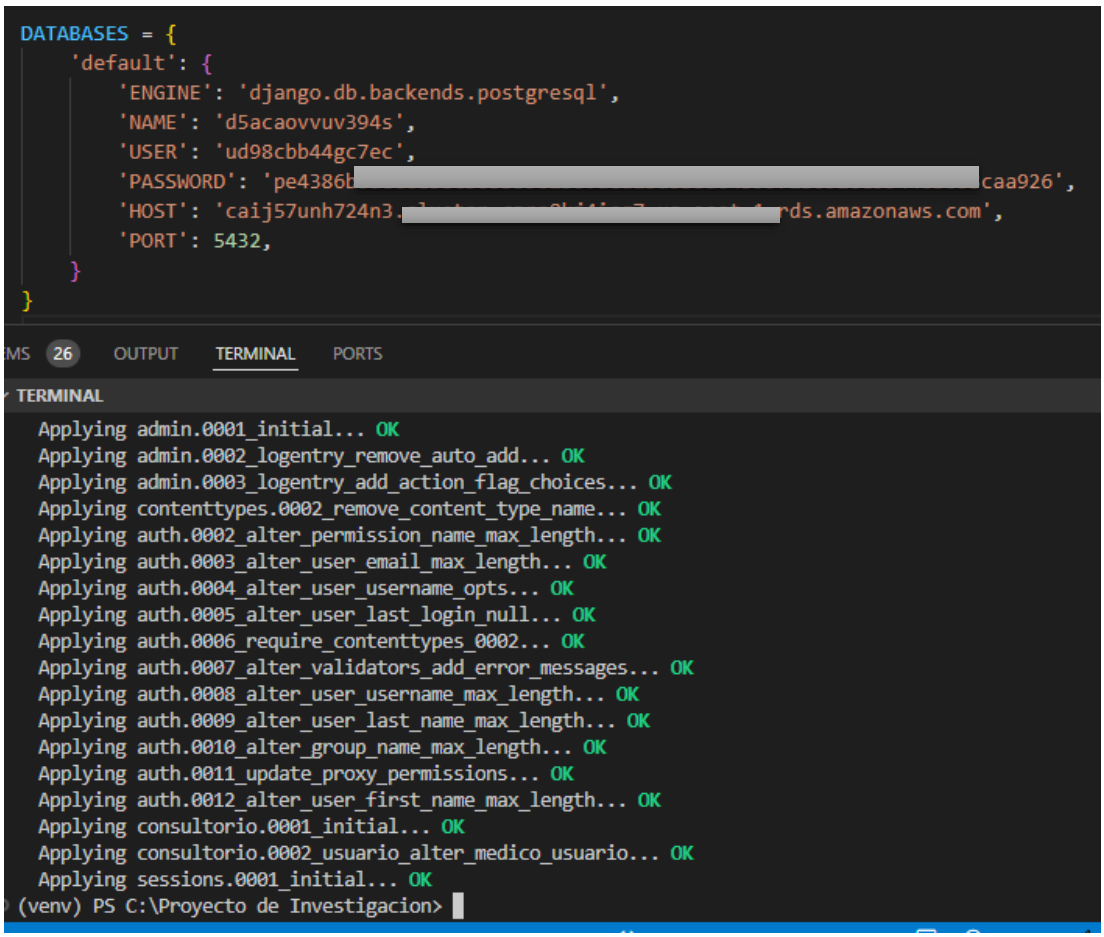


Figura 89. Conexión a la base de datos de Heroku

A través del gestor de base de datos pgAdmin4 se puede ingresar a la base alojada en Heroku para visualizar la nueva conectividad.



Figura 90. Base de datos alojada en Heroku

Para la implementación del Django, el cliente debe abrir el proyecto Django en Visual Studio Code, ingresa al terminar del IDE activando el entorno virtual del proyecto para ejecutar el proyecto de Django con el comando “python manage.py runserver”. Este proceso inicia la ejecución del framework. El servidor local de Django es http://127.0.0.1:8000/

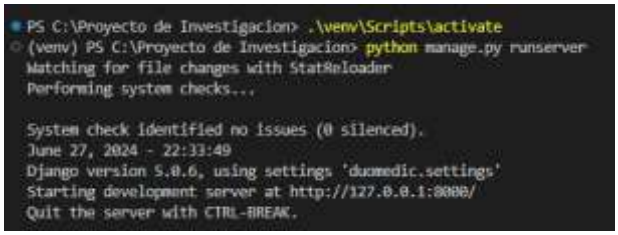


Figura 91. Ruta del proyecto Django en el terminal



Figura 92. Aplicación en el navegador web de manera local

CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- Al momento de aplicar las técnicas de recolección de datos en el consultorio médico “DuoMedic”, se evidenciaron problemas en los procesos que realiza el médico en cuanto a la búsqueda de pacientes. Durante la búsqueda, el médico tenía que buscar en grandes archiveros físicos, lo cual presentaba problemas de tiempo y eficiencia. A demás otra observación clara es que el manejo de documentación física como la ficha y recetas médicas se las sigue realizando de manera manual causando problemas en la integridad de los datos como el ingreso de datos erróneos o falta de legibilidad de la información.

- Entre los diferentes frameworks encontrados durante la investigación, Django fue seleccionado como el framework óptimo para el desarrollo de la aplicación web. Se consideraron factores como la documentación disponible, la curva de aprendizaje, la comunidad de soporte, arquitectura de trabajo MVT y la gestión de datos ORM con ayuda de PostgreSQL para el manejo de datos.

- El desarrollo de una aplicación web enfocada en el registro y control de posología para el consultorio médico “DuoMedic” representa un paso crucial hacia la digitalización y eficiencia en la gestión médica. Esta iniciativa no solo busca facilitar el registro, seguimiento de historias clínicas y prescripciones médicas, sino también optimizar la calidad y tiempo de atención ofrecida a los pacientes, fortaleciendo así el servicio médico integral y personalizado que ofrecemos.

4.2 Recomendaciones

- Se recomienda realizar un respaldo de los datos de los pacientes que actualmente se almacenan de manera física, e integrarlos en el sistema. Este proceso garantizará la integridad y seguridad de los datos, además de mejorar la eficiencia en la búsqueda y manejo de la información.

- Se recomienda revisar las documentaciones oficiales de Django, para asegurar la compatibilidad entra la versión del framework y PostgreSQL, para garantizar un funcionamiento óptimo y aprovechar las últimas mejoras en seguridad y rendimiento.

- Se recomienda planificar la expansión del sistema con nuevas funcionalidades clave para el crecimiento del consultorio, como módulos de telemedicina, gestión de inventario de suministros médicos y un sistema de citas en línea. Paralelamente, es crucial implementar estrategias de optimización de datos en la nube, como el uso de índices eficientes, particionamiento de tablas y almacenamiento en caché, para manejar el incremento en el volumen de datos.

REFERENCIAS BIBLIOGRÁFICAS

[1] M. J. Cervantes-López, A. A. Peña-Maldonado, and A. Ramos-Sánchez, "Uso de las tecnologías de la información y comunicación como herramienta de apoyo en el aprendizaje de los estudiantes de medicina," *CienciaUAT*, vol. 15, no. 1, pp. 162–171, 2020, doi: https://doi.org/10.29059/cienciauat.v15i1.1380.

[2] Y. Damayanthy, S. Anaya, M. Tamaulipas, M. Tamaulipas, and M. Tamaulipas, "El impacto del internet de todas las cosas (IoT) en la vida cotidiana," *Cienc. Lat. Rev. Científica Multidiscip.*, vol. 6, no. 2, pp. 1369–1378, 2022, doi: 10.37811/cl_rcm.v6i2.1959.

[3] INEC, "Establecimientos de Salud," 2022. https://www.ecuadorencifras.gob.ec/establecimientos-de-salud/.

[4] M. de S. Pública, "ESTABLECIMIENTOS DE SALUD," 2023. https://www.salud.gob.ec/establecimientos-de-salud-/#:~:text=ESTABLECIMIENTOS DE SALUD – Ministerio de Salud P%C3.

[5] R. M. Añel Rodríguez, I. García Alfaro, R. Bravo Toledo, and J. D. Carballeira Rodríguez, "Historia clínica y receta electrónica: riesgos y beneficios detectados desde su implantación. Diseno, ˜ despliegue y usos seguros," *Aten. Primaria*, vol. 53, 2021, doi: 10.1016/j.aprim.2021.102220.

[6] L. Cotacachi, "Errores de prescripción en la consulta externa en los Centros de Salud tipo B del Distrito 17D04 (La Tola, La Vicentina y La Libertad B) durante el período 2017 y propuesta de validación de recetas médicas.," Universidad Central del Ecuador, 2019.

[7] M. Changoluisa, "Escuela Superior Politécnica De Chimborazo Facultad De Ciencias Bioquímica Farmacéutica Autora: María Valeria Changoluisa Supe," ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO, 2022.

[8] Álvaro Guerrero Sánchez, "Desarrollo de una aplicación web para un gimnasio en Django," Escuela Técnica Superior de Ingeniería Informática Universidad Politécnica de Valencia, 2022.

[9] M. Yépez, "SISTEMA WEB PARA LA GESTIÓN DE HISTORIAS CLINICAS Y CONTROL DE INSUMOS EN EL DISPENSARIO MÉDICO DE LETERAGO DEL ECUADOR S.A. SEDE QUITO (SWL)," 2020.

[10] O. P. Escobar Chávez, "Desarrollo de una aplicación web para regulación de presupuestos en proyectos eléctricos aplicando la herramienta Django en la empresa INGELEC-BM," Dec. 2022, Accessed: May 13, 2023. [Online]. Available: http://dspace.unach.edu.ec/handle/51000/10170.

[11] L. F. Ordóñez Armijos, "DESARROLLO DE UN SISTEMA WEB DE GESTIÓN DE FICHAS MÉDICAS PARA LA ORGANIZACIÓN 'SALUD INTEGRAL PARA TODOS', BAJO LA METODOLOGÍA SCRUM.," UNIVERSIDAD TECNOLÓGICA ISRAEL, Quito, 2019.

[12] Amrita Pathak, "Ciclo de vida de desarrollo de software (SDLC): una guía completa - Geekflare," Jul. 01, 2022. https://geekflare.com/es/software-development-life-cycle-sdlc-guide/ (accessed May 22, 2023).

[13] S. M. Velásquez, J. D. Vahos Montoya, M. E. Gómez Adasme, E. J. Restrepo Zapata, A. A. Pino, and S. Londoño Marín, "Una revisión comparativa de la literatura acerca

de metodologías tradicionales y modernas de desarrollo de software," *Rev. CINTEX*, vol. 24, no. 2, pp. 13–23, 2019, doi: 10.33131/24222208.334.

[14] E. Bautista-villegas, "Metodologías agiles XP y Scrum, empleadas para el desarrollo de páginas web, bajo MVC, con lenguaje PHP y framework Laravel," vol. 1, pp. 1–7, 2022, doi: https://doi.org/10.55873/rad.v1i1.168.

[15] A. R. León Yacelga and M. A. Checa Cabrera, "USO DE TABLEROS KANBAN COMO APOYO PARA EL DESARROLLO DE LAS METODOLOGÍAS ÁGILES," *Univ. Reg. Autónoma Los Andes. Ecuador*, vol. 7, no. 1, pp. 1–8, 2022.

[16] S. Abel, "Metodologías ágiles Scrum, XP, SLeSS, Scrumban, HME, Mobile-D y MASAN empleadas en la industria de dispositivos móviles: Un contraste en favor de la industria del desarrollo móvil.," *Dir. Gen. Investig.*, pp. 1–93, 2020, doi: http://hdl.handle.net/20.500.12840/3906.

[17] J. CHÓEZ, "APLICACIÓN INFORMÁTICA PARA LA GESTIÓN DE PROCESOS DE RECAUDACIÓN E INVENTARIO DEL CENTRO DE REHABILITACIÓN FÍSICA SAN LORENZO DE JIPIJAPA," 2022, [Online]. Available: http://repositorio.unesum.edu.ec/handle/53000/4315.

[18] D. Telenchana, "APLICACIÓN WEB USANDO EL FRAMEWORK ANGULAR PARA EL CONTROL DE HISTORIAS CLÍNICAS DE LOS PACIENTES DEL CONSULTORIO MÉDICO FISIO&TRAUMA DE LA CIUDAD DE AMBATO," p. 80, 2022.

[19] R. Espinosa-Hurtado, "Análisis comparativo para la evaluación de frameworks usados en el desarrollo de aplicaciones web," *Cedamaz*, vol. 11, no. 2, pp. 133–141, 2021, doi: 10.54753/cedamaz.v11i2.1182.

[20] A. Borrego, D. Ayala, F. Sola, I. Hernández, and D. Ruiz, "Silence : un framework de apoyo a la docencia de desarrollo web Abstract Este framework se ha desarrollado en el contexto de," vol. 6, 2021.

[21] J. E. Chacha Pizarro and D. M. Vargas Cuacés, "Sistema integrado para el Centro de Desarrollo Infantil 'Pequeños Sabios' de la Empresa Pública UPEC-Creativa EP," UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI, 2022.

[22] G. I. NÚÑEZ RIBADENEYRA, "DESARROLLO DE LENGUAJE DE DOMINIO ESPECÍFICO PARA LA GENERACIÓN DE UNA INFRAESTRUCTURA CRUDREST ESCRITA EN PYTHON Y FLASK CON CONEXIÓN A UNA BASE DE DATOS," ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO, 2021.

[23] D. G. NAGUA PULLAGUARI, "DESARROLLO DE SOFTWARE PARA LA GESTIÓN Y MANTENIMIENTO DE VEHÍCULOS EN UNA EMPRESA PÚBLICA BASADOS EN LA METODOLOGÍA SNAIL," 2024, [Online]. Available: http://repositorio.utmachala.edu.ec/bitstream/48000/15882/1/ECFIC-2020-IC-DE-00051.pdf.

[24] J. M. Ortega Candel, *Desarrollo de microservicios con Python Tecnologías y frameworks para aplicaciones basadas en microservicios*. España, 2023.

[25] M. A. Cofre Chango, "APLICACIÓN MÓVIL PARA LA IDENTIFICACIÓN DE LAS PROPIEDADES MEDICINALES DE PLANTAS NATIVAS LOCALIZADAS EN LA MICROEMPRESA 'AGROCOTOPAXI,'" UNIVERSIDAD TÉCNICA DE COTOPAXI, 2020.

[26] J. Pablo and V. Saavedra, "Control interno de la ejecución presupuestal en la gestión de salud 2020," *Cienc. Lat. Rev. Científica Multidiscip.*, vol. 2215, no. 2, pp. 1092–

1110, 2020, doi: 10.37811/cl_rcm.v4i2.140.

[27] S. Esteban, M. B. Islas, and M. Ponce, “10 consejos para la gestión de la información en salud,” 2020. https://blogs.iadb.org/salud/es/10-consejos-para-la-gestion-de-la-informacion-en-salud/.

[28] R. Alejandro, “MANUAL DE PROCESOS, CONTROL Y SEGUIMIENTO DE SUBVENCIONES DEL GOBIERNO ESPAÑOLA LA FUNDACIÓN ESPAÑA SALUD. (FES),” 2022.

[29] Dr. Francisco José Eguez Mora, “Propuesta de una historia clínica electrónica en el pensionado del Hospital Enrique C. Sotomayor.,” p. 123, 2019, [Online]. Available: http://repositorio.ucsg.edu.ec/bitstream/3317/3367/1/T-UCSG-POS-MGSS-37.pdf.

[30] A. D. Bonilla Adriano, “Aplicación web SPA para la gestión de fichas médicas en el Hospital Universitario Andino utilizando servicios REST,” 2021, [Online]. Available: http://dspace.unach.edu.ec/handle/51000/7636.

[31] J. MORENO, “LA PRESCRIPCIÓN MÉDICA,” 2022. https://flasog.org/2022/02/28/la-prescripcion-medica/ (accessed May 13, 2023).

[32] M. de S. Pública, “Resolucion No. ACESS-2021-0006,” *Agencia Asegur. la Calid. los Serv. salud y Med. prepagada*, pp. 1–18, 2021, [Online]. Available: http://www.acess.gob.ec/wp-content/uploads/2021/08/Resolución-Nro.-ACESS-2021-0006.pdf.

[33] Ministerio de Salud, “Reglamento para establecer el contenido y requisitos de la receta médica y control de la prescripción, dispensación y expendio para medicamentos de uso y consumo humano,” no. 00031, pp. 1–14, 2020.

[34] Zambrano, “Reglamento de receta medica y control de prescripcion de medicamentos,” 2020. https://www.meythalerzambranoabogados.com/post/reglamento-de-receta-medica-y-control-de-prescripcion-de-medicamentos (accessed May 13, 2023).

[35] O. M. de la Salud, “Clasificación Internacional de Enfermedades, 11.a revisión,” *2023*. https://icd.who.int/es.

[36] C. del Ecuador, “CUADRO BÁSICO DE MEDICAMENTOS,” *2023*. http://www.saludzona1.gob.ec/cz1/index.php/programas-salud/cuadro-medicamentos.

ANEXOS

Anexo A Alfa de Cronbach en la encuesta de pacientes

Encuestados	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5	Pregunta 6	Pregunta 7	Pregunta 8	Pregunta 9	Pregunta 10	Pregunta 11	Pregunta 12	SUMA
1	3	2	2	1	2	3	2	2	1	1	1	3	23
2	3	2	3	1	3	4	3	2	2	2	2	4	31
3	3	2	3	1	2	4	2	2	2	2	3	4	30
4	3	2	2	2	3	4	3	2	3	2	3	5	34
5	2	1	2	2	3	4	3	2	1	1	1	1	23
6	4	2	3	2	1	2	1	2	2	1	2	3	25
7	3	2	3	2	3	5	3	2	2	2	3	4	34
8	3	2	3	2	5	5	5	2	3	2	3	5	40
9	4	3	3	2	5	5	5	2	2	2	2	4	39
10	4	3	4	2	3	4	3	2	1	1	1	2	30
11	3	2	3	3	3	4	3	2	1	1	1	3	29
12	3	2	2	3	3	4	3	2	1	1	1	1	26
13	1	1	1	3	4	5	4	2	1	1	1	1	25
14	3	2	3	3	2	4	2	2	3	2	3	5	34
15	5	4	4	3	3	4	3	2	2	2	2	4	38
16	3	2	3	3	3	4	3	2	2	2	2	4	33
17	1	1	1	3	2	4	3	2	2	2	2	4	27
18	5	3	4	3	2	3	2	2	3	3	3	5	38
19	4	2	3	3	1	1	1	2	2	2	2	4	27
20	4	3	4	3	4	5	4	2	2	2	3	4	40
21	2	1	1	4	2	4	2	2	5	5	5	5	38
22	5	5	5	4	4	5	4	2	2	2	2	4	44
23	5	3	4	4	1	2	1	2	3	2	3	5	35
24	2	1	2	4	1	2	1	2	3	3	4	5	30
25	3	2	3	4	4	5	3	2	1	1	1	3	32
26	2	1	2	4	2	4	2	2	1	1	1	3	25
27	1	1	1	4	3	4	3	2	2	2	2	4	29
28	5	4	4	4	3	4	3	2	4	4	5	5	47
29	3	2	2	4	1	3	1	2	2	2	2	4	28
30	4	3	4	4	3	4	3	2	2	2	2	4	37
31	2	2	2	4	3	4	3	2	1	1	1	2	27
32	3	2	3	4	2	4	2	2	3	3	3	5	36
33	3	2	3	4	1	3	2	2	2	2	3	4	31
34	2	1	2	4	3	4	3	2	2	2	2	4	31
35	5	5	5	4	3	4	3	2	2	2	3	4	42
36	4	2	3	4	5	5	4	2	2	2	2	3	38
37	5	5	5	4	2	3	2	2	1	1	2	3	35
38	2	1	1	4	1	2	1	2	2	2	2	3	23
39	4	3	4	4	2	3	2	2	4	3	4	5	40
40	2	2	2	4	3	4	3	2	1	1	1	3	28
41	4	3	4	4	4	5	3	2	1	1	1	3	35
42	1	1	1	4	2	3	2	2	4	3	4	5	32
43	1	1	1	4	4	5	3	2	2	2	3	4	32
44	5	3	4	4	3	4	3	2	2	2	2	4	38
45	5	3	4	4	3	4	3	2	1	1	1	3	34
46	3	2	2	4	1	3	1	2	1	1	1	2	23
47	2	2	2	4	2	3	2	2	1	1	1	3	25
48	4	3	4	4	3	4	3	1	2	1	2	3	34
49	4	2	3	4	1	2	1	3	1	1	1	3	26
50	4	3	3	4	1	1	1	4	2	2	3	5	33
51	1	1	1	5	1	3	2	5	2	2	2	4	29
52	4	2	3	5	2	4	2	1	2	1	2	3	31
53	1	1	1	5	2	4	2	1	1	1	1	2	22
54	4	3	3	5	2	4	3	1	2	2	2	4	35
55	5	4	4	5	4	5	4	1	2	2	2	4	42
56	5	4	4	5	4	5	4	3	3	3	4	5	49
57	2	2	2	5	4	5	4	3	2	1	2	3	35
58	3	2	3	5	2	4	2	3	3	3	3	5	38
59	5	3	4	5	2	4	2	1	4	4	5	5	44
60	4	3	3	5	3	5	3	4	2	1	2	3	38
61	1	1	1	5	1	3	2	3	2	2	2	4	27

60	4	3	3	5	3	5	3	4	2	1	2	3	38
61	1	1	1	5	1	3	2	3	2	2	2	4	27
62	2	1	2	5	5	5	5	4	4	3	4	5	45
63	1	1	1	5	4	5	4	4	2	1	2	3	33
64	4	3	4	5	1	1	1	3	1	1	1	2	27
65	1	1	1	5	1	2	1	1	1	1	1	2	18
66	2	1	2	5	1	2	1	1	3	2	3	5	28
VARIANZA	1,770	1,132	1,333	1,179	1,389	1,186	1,140	0,581	0,889	0,746	1,171	1,210	
SUMATORIA DE VARIANZA	13,725												
VARIANZA DE LA SUMA DE LOS ITEMS	43,831												

α (ALFA) =	0,749
K (número de items) =	12
varianza de cada item) =	13,725
Vt (varianza total) =	43,831

$$\alpha = \left(\frac{k}{k-1}\right)\left(1 - \frac{\sum v_i}{vt}\right)$$

RANGO	CONFIABILIDAD
0.53 a menos	Confiabilidad nula
0.54 a 0.59	Confiabilidad baja
0.60 a 0.65	Confiable
0.66 a 0.71	muy confiables
0.72 a 0.99	Exelente Confiabilidad
1	Confiabilidad Perfecta

Anexo B Cuestionario de la encuesta dirigida a los pacientes del consultorio Médico “DuoMedic”

Encuesta dirigida a los pacientes del consultorio Médico “DuoMedic”

Indicaciones para los Encuestados:

Agradecemos su colaboración en esta encuesta. Por favor, responda cada pregunta honesta y detalladamente. Sus respuestas son esenciales para diseñar una aplicación web que se adapte a sus necesidades. Marque la opción que mejor refleje su opinión. Todas las preguntas son obligatorias. Gracias por su participación.

Objetivo: Evaluar la satisfacción de los pacientes con el manejo actual de las fichas médicas y el control de la posología en el Consultorio Médico "DuoMedic", así como conocer su percepción sobre la implementación de una aplicación web para el registro de información médica y el envío digital. de recetas.

1. ¿Desde hace cuánto tiempo es paciente del Consultorio Médico "DuoMedic"?

    - Menos de 3 meses
    - Entre 3 meses y 1 año
    - Entre 1 y 2 años
    - Entre 2 y 3 años
    - Más de 3 años

2. ¿Su médico tratante revisa su historial médico completo durante las consultas?

    - Siempre
    - Casi siempre
    - A veces
    - Casi nunca
    - Nunca

3. ¿Cuánto tiempo le toma a su médico encontrar y revisar su historial médico durante las consultas?

- Menos de 2 minutos
- Entre 2 y 5 minutos
- Entre 5 y 10 minutos
- Entre 10 y 15 minutos
- Más de 15 minutos

4. ¿Cómo calificaría la eficiencia del proceso de registro manual de su ficha médica en el consultorio?

    - Muy eficiente
    - Eficiente
    - Ni eficiente ni deficiente
    - Poco eficiente
    - Nada eficiente

5. ¿Cuánto tiempo suele tardar el proceso de registro de su información médica en el consultorio?

    - Menos de 5 minutos
    - Entre 5 y 10 minutos
    - Entre 10 y 15 minutos
    - Entre 15 y 20 minutos
    - Más de 20 minutos

6. ¿Qué tan satisfecho está con el proceso actual de registro y manejo de su ficha médica y posología (dosis y frecuencia de los medicamentos) en el consultorio?

    - Muy satisfecho
    - Satisfecho
    - Ni satisfecho ni insatisfecho
    - Insatisfecho
    - Muy insatisfecho

7. ¿Ha tenido alguna vez problemas o confusiones con las indicaciones de la posología (dosis y frecuencia de los medicamentos) dadas por su médico?

- Siempre
- Frecuentemente
- A veces
- Casi nunca
- Nunca

8. En relación con la prescripción de medicamentos, ¿cuánto tiempo le toma a su médico buscar y registrar la posología (dosis y frecuencia de los medicamentos) adecuada para su tratamiento?

    - Menos de 2 minutos
    - Entre 2 y 5 minutos
    - Entre 5 y 10 minutos
    - Entre 10 y 15 minutos
    - Más de 15 minutos

9. ¿Considera que el uso de la aplicación web para el registro de la ficha médica y control de la posología (dosis y frecuencia de los medicamentos) mejoraría la organización y eficiencia en el consultorio médico?

    - Totalmente de acuerdo
    - De acuerdo
    - Ni de acuerdo ni en desacuerdo
    - En desacuerdo
    - Totalmente en desacuerdo

10. ¿Cree que una aplicación web facilitaría la comunicación y seguimiento con su médico tratante en relación a su ficha médica y posología (dosis y frecuencia de los medicamentos)?

    - Totalmente de acuerdo
    - De acuerdo
    - Ni de acuerdo ni en desacuerdo
    - En desacuerdo
    - Totalmente en desacuerdo

11. ¿Considera que el envío de la receta médica por correo electrónico facilitaría el seguimiento y cumplimiento de su tratamiento, cumpliendo con todas las normas y leyes legales para que sea una receta válida?

    - Totalmente de acuerdo
    - De acuerdo
    - Ni de acuerdo ni en desacuerdo
    - En desacuerdo
    - Totalmente en desacuerdo

12. En base a su experiencia general con el Consultorio Médico "DuoMedic", ¿qué tan probable es que lo recomiende a un familiar o amigo?

    - Definitivamente no lo recomendaría
    - Probablemente no lo recomendaría
    - Indeciso/Neutral
    - Probablemente lo recomendaría
    - Definitivamente lo recomendaría

Anexo C Guía de entrevista dirigida al profesional de la salud del consultorio Médico "DuoMedic"

Entrevista dirigida al profesional de la salud del consultorio Médico "DuoMedic"

Nombre del entrevistado: Dr. Cesar Guerrero
Empresa: Consultorio Médico "DuoMedic"
Cargo: Médico General
Entrevistador: José Villacrés

Objetivo:
Identificar las dificultades que enfrenta el médico personal en el proceso de registro manual de fichas médicas y control de la posología, además de conocer su disposición a adoptar una aplicación web que facilite el acceso a la información clínica completa de los pacientes y el envío digital de recetas médicas.

N°	Pregunta	Respuesta	Observación
1	¿Podría describir el proceso actual que sigue para el registro de la ficha médica de un paciente?		
2	¿Podría describir el proceso actual que sigue para la prescripción de medicamentos a los pacientes?		
3	¿Ha tenido alguna vez problemas o pérdidas de información debido al manejo manual de las fichas médicas y recetas de pacientes?		
4	¿Qué dificultades o problemas ha encontrado con el método actual de registro de la ficha médica?		
5	¿Utiliza alguna aplicación informática para gestionar la información clínica de los pacientes?		
6	¿Cómo maneja actualmente el control de la posología (dosis y frecuencia de medicamentos) de sus pacientes?		
7	¿Ha tenido casos en los que los pacientes han reportado confusiones o problemas al seguir las indicaciones de la posología?		
8	¿Cree que el acceso a la ficha médica completa de los pacientes a través de una aplicación web facilitaría su trabajo y la atención brindada?		
9	Desde su punto de vista como médico, ¿qué tipo de información le gustaría poder acceder fácilmente a través de la aplicación mientras está interactuando con un paciente?		
10	¿Estaría dispuesto a utilizar una aplicación web para el registro y actualización de las fichas médicas de sus pacientes?		
11	¿Le resultaría útil poder enviar las recetas médicas de sus pacientes por correo electrónico a través de la aplicación web?		
12	¿Cree que el envío digital de recetas médicas válidas y legales podría mejorar el seguimiento y cumplimiento del tratamiento por parte de los pacientes?		

Conclusion:

Anexo D Diseño de la base de datos

