

**UNIVERSIDAD NACIONAL DEL SANTA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional de Ingeniería de Sistemas e Informática**



**UNS**  
UNIVERSIDAD  
NACIONAL DEL SANTA

**“Desarrollo de un asistente virtual para la consulta  
automatizada de expedientes judiciales en el módulo penal  
CSJ Santa”**

**Trabajo de Suficiencia Profesional para Obtener el Título Profesional  
de Ingeniero de Sistemas e Informática**

**AUTOR:**

- Bach. Moreno Saavedra, Marco Josué

**ASESOR:**

- Dr. Sánchez Chávez, Juan Pablo

**NUEVO CHIMBOTE – PERÚ**

**2025**

**UNIVERSIDAD NACIONAL DEL SANTA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional de Ingeniería de Sistemas e Informática**

**“Desarrollo de un asistente virtual para la consulta  
automatizada de expedientes judiciales en el módulo penal  
CSJ Santa”**

**Trabajo de Suficiencia Profesional para Obtener el Título Profesional  
de Ingeniero de Sistemas e Informática**

**Revisado y Aprobado por el Asesor:**



---

**Dr. Juan Pablo Sánchez Chávez**  
**DNI: 17808722**  
**Cód. ORCID: 0000-0002-3521-7037**  
**ASESOR**

**NUEVO CHIMBOTE – PERÚ**

**2025**

**UNIVERSIDAD NACIONAL DEL SANTA**

**FACULTAD DE INGENIERÍA**

**Escuela Profesional de Ingeniería de Sistemas e Informática**

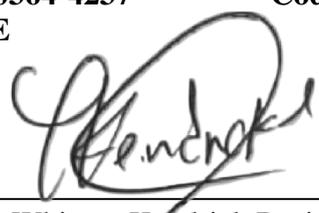
**“Desarrollo de un asistente virtual para la consulta  
automatizada de expedientes judiciales en el módulo penal  
CSJ Santa”**

**Trabajo de Suficiencia Profesional para Obtener el Título Profesional  
de Ingeniero de Sistemas e Informática**

**Revisado y Aprobado por el Jurado Evaluador:**

  
\_\_\_\_\_  
Ms. Mirko Martín Manrique Ronceros  
DNI: 32965599  
Cód. ORCID: 0000-0002-0364-4237  
**PRESIDENTE**

  
\_\_\_\_\_  
Dr. Juan Pablo Sánchez Chávez  
DNI: 17808722  
Cód. ORCID: 0000-0002-3521-7037  
**SECRETARIO**

  
\_\_\_\_\_  
Ms. Whiston Kendrick Borja Reyna  
DNI: 44939310  
Cód. ORCID: 0000-0002-5966-3859  
**INTEGRANTE**

**NUEVO CHIMBOTE – PERÚ**

**2025**

**ACTA DE SUSTENTACIÓN INFORME DE TRABAJO DE SUFICIENCIA  
PROFESIONAL**

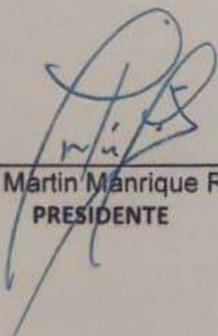
A los veintidós días del mes de agosto del año dos mil veinticinco, siendo las 10:00 am. En el aula S1 del Pabellón de la Escuela Profesional de Ingeniería Sistema e Informática-FI-UNS, se instaló el Jurado Evaluador designado mediante Resolución 448-2025-UNS-CFI, y de expedito según Resolución Decanal N° 551-2025-UNS-FI integrado por los docentes: Ms. Mirko Martin Manrique Ronceros (presidente), Dr. Juan Pablo Sanchez Chavez (secretario) y Ms. Whiston Kendrick Borja Reyna (Integrante), para dar inicio a la sustentación del Informe de Trabajo de Suficiencia Profesional : "DESARROLLO DE UN ASISTENTE VIRTUAL PARA LA CONSULTA AUTOMATIZADA DE EXPEDIENTES JUDICIALES EN EL MÓDULO PENAL CSJ SANTA ", perteneciente al Bachiller: MORENO SAAVEDRA MARCO JOSUÉ, con código de matrícula N°0201414022, quien fue asesorado por el Dr. Juan Pablo Sánchez Chávez, según Resolución N°440-2025-UNS-FI.

El Jurado Evaluador, después de deliberar sobre aspectos relacionados con el trabajo, contenido y sustentación del mismo, y con las sugerencias pertinentes en concordancia con el Reglamento General de Grados y Títulos, vigente, declaran aprobar:

BACHILLER	PROMEDIO VIGESIMAL	PONDERACIÓN
MORENO SAAVEDRA MARCO JOSUÉ	18	BUENO

Siendo las 11.00 am del mismo día, se dio por terminado el acto de sustentación, firmando la presente acta en señal de conformidad.

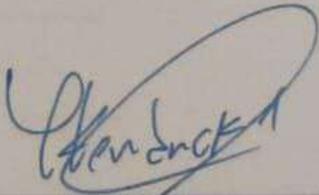
Nuevo Chimbote, 22 de agosto de 2025



Ms. Mirko Martin Manrique Ronceros  
PRESIDENTE



Dr. Juan Pablo Sánchez Chávez  
SECRETARIO



Ms. Whiston Kendrick Borja Reyna  
INTEGRANTE

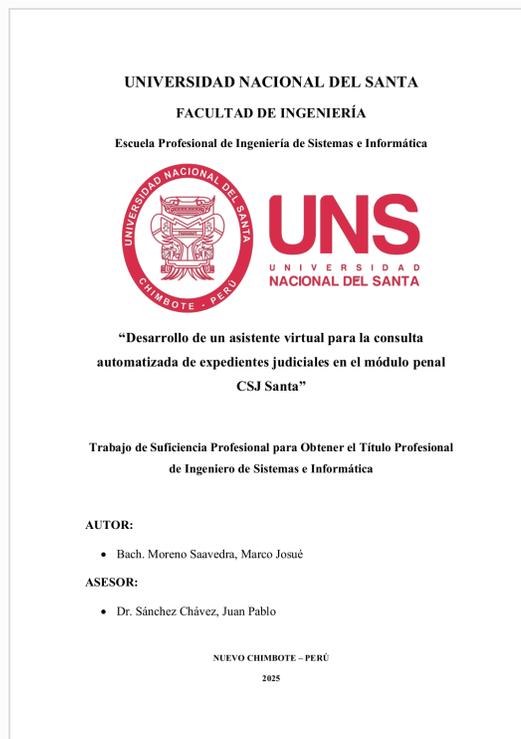


## Recibo digital

Este recibo confirma que su trabajo ha sido recibido por **Turnitin**. A continuación podrá ver la información del recibo con respecto a su entrega.

La primera página de tus entregas se muestra abajo.

Autor de la entrega: Marco Moreno  
Título del ejercicio: Tesis 2025  
Título de la entrega: TESIS-SUFICIENCIA-MARCO.pdf  
Nombre del archivo: TESIS-SUFICIENCIA-MARCO.pdf  
Tamaño del archivo: 3.97M  
Total páginas: 125  
Total de palabras: 18,451  
Total de caracteres: 107,496  
Fecha de entrega: 04-sept-2025 11:04a. m. (UTC-0500)  
Identificador de la entrega: 2741890591



## Fuentes principales

- 16% Fuentes de Internet
- 2% Publicaciones
- 0% Trabajos entregados (trabajos del estudiante)

## Fuentes principales

Las fuentes con el mayor número de coincidencias dentro de la entrega. Las fuentes superpuestas no se mostrarán.

1	Internet	repositorio.uns.edu.pe	7%
2	Internet	www.coursehero.com	2%
3	Internet	alicia.concytec.gob.pe	1%
4	Internet	hdl.handle.net	<1%
5	Internet	medium.com	<1%
6	Internet	en.wikipedia.org	<1%
7	Internet	www.kutombawewe.net	<1%
8	Internet	academica-e.unavarra.es	<1%
9	Internet	www.agilescrum.cl	<1%
10	Internet	repositorio.ucv.edu.pe	<1%
11	Internet	repositorioacademico.upc.edu.pe	<1%

# AGRADECIMIENTO

Quisiera manifestar mi más profundo agradecimiento a:

A mis padres, por su respaldo incondicional,  
su ejemplo de esfuerzo y compromiso, y por  
brindarme siempre el apoyo necesario para  
alcanzar esta meta académica.

A los docentes de la EPISI, cuyo  
compromiso, guía y dedicación fueron clave  
en mi desarrollo académico y profesional  
durante toda mi formación universitaria

# INDICE GENERAL

HOJA DE APROBACIÓN DEL JURADO EVALUADOR	ii
AGRADECIMIENTO	iii
INDICE GENERAL	iv
LISTA DE TABLAS	vi
LISTA DE FIGURAS	vii
PRESENTACIÓN	ix
RESUMEN	xi
ABSTRACT	xii
I. Tema específico abordado	13
II. Contextualización de la experiencia profesional	13
2.1. Cronología de la Trayectoria Profesional	13
2.1.1. Corte Superior de Justicia de Santa	13
2.2. Contexto en que se desarrolló la experiencia	19
2.2.1. La Institución	19
2.2.2. Contexto del proyecto	21
III. Importancia para el ejercicio de la carrera profesional	24
IV. Objetivos logrados	25
4.1. Objetivo General	25
4.2. Objetivos Específicos	25
V. Sustento teórico del tema abordado	26
5.1. Antecedentes	26
5.1.1. Investigaciones Internacionales:	26
5.1.2. Investigaciones Nacionales	29
5.1.3. Investigaciones Locales	32
5.2. Fundamentos teóricos de la investigación	36
5.2.1. Api Rest	36
5.2.2. Chatbot	38
5.2.3. Api Chatbot Telegram	41
5.2.4. Scrum	43
5.2.5. Spring Framework	45
5.2.6. PostgreSQL	48
VI. Organización y sistematización de las experiencias logradas	52

6.1.	Desarrollo de la Metodología SCRUM	52
6.1.1.	Visión general de la situación actual del negocio	52
6.1.2.	Iniciación del proyecto de software	54
6.1.3.	Planificación y estimación de la Iteración (Sprint)	60
6.1.4.	Identificación de tareas por cada iteración (Sprint)	65
6.1.5.	Planificación de las iteraciones (Sprint)	70
6.1.6.	Cronograma de las Iteraciones (Sprints)	71
6.1.7.	Ejecución de la Iteración (Sprint).	73
6.2.	Diagrama de Base de Datos	106
VII.	Ubicación de las experiencias en el marco del sustento teórico	107
VIII.	Aportes logrados para el desarrollo del centro laboral	110
IX.	Aportes para la formación profesional	111
X.	Conclusiones y recomendaciones	113
10.1.	Conclusiones	113
10.2.	Recomendaciones	114
XI.	Referencias bibliográficas	115
ANEXOS		117

## LISTA DE TABLAS

Tabla 1: Roles Scrum del proyecto .....	55
Tabla 2: Requisitos funcionales del sistema .....	55
Tabla 3: Requisitos no funcionales del sistema .....	57
Tabla 4: Backlog del producto e historias de usuario .....	59
Tabla 5: Historias de usuario priorizadas.....	62
Tabla 6: Tareas y descripción de cada historia de Usuario.....	65
Tabla 7: Planificación de las iteraciones (sprints).....	70
Tabla 8: Cronograma para los Sprints(Año 2019).....	72
Tabla 9: Especificación de la HU01: Gestionar usuarios y roles del sistema.....	74
Tabla 10: Especificación de la HU02: Autenticar usuario en el Sistema .....	77
Tabla 11: Especificación de la HU03: Recibir instrucciones de registro .....	79
Tabla 12: Especificación de la HU04: Consultar expedientes por número .....	82
Tabla 13: Especificación de la HU05: Consultar expedientes por DNI del imputado ..	84
Tabla 14: Especificación de la HU06: Actas de audiencia de un expediente .....	86
Tabla 15: Especificación de la HU07: Acceder a resoluciones de un expediente .....	88
Tabla 16: Especificación de la HU08: Acceder a digitalizados de un expediente.....	90
Tabla 17: Especificación de la HU09: Consultar depósitos judiciales .....	91
Tabla 18: Especificación de la HU10: Consultar registros de firmas. ....	93
Tabla 19: Especificación de la HU11: Seleccionar recepción de documentos .....	95
Tabla 20: Especificación de la HU12: Consultar principio de oportunidad .....	96
Tabla 21: Especificación de la HU13: Navegar entre diferentes expedientes .....	99
Tabla 22: Especificación de la HU14: Acceder a estadísticas de uso del sistema.....	100
Tabla 23: Especificación de la HU15: Enviar mensajes masivos a los usuarios .....	102

## LISTA DE FIGURAS

Figura 1 Organigrama CSJ Santa.....	21
Figura 2 Etapas Scrum .....	45
Figura 3 Módulos de Spring Boot.....	47
Figura 4 Diagrama del proceso de negocio de la gestión de solicitudes.....	54
Figura 5 Secuencia de ejecución de los Sprints .....	70
Figura 6 Diagrama de Capas-MVC para la HU01 .....	75
Figura 7 UI para la HU01: Gestionar usuarios y roles del sistema.....	76
Figura 8 Diagrama de Capas-MVC para la HU02.....	78
Figura 9 UI para la HU02: Autenticar Usuarios .....	79
Figura 10 Diagrama de Capas-MVC para la HU3.....	80
Figura 11 UI para la HU03: Recibir instrucciones de registro .....	81
Figura 12 Diagrama de Capas-MVC para la HU04.....	83
Figura 13 UI para la HU4: Consultar expedientes por número .....	84
Figura 14 Diagrama Capas-MVC para la HU05.....	85
Figura 15 UI para la HU05: Consultar expedientes por DNI del imputado .....	86
Figura 16 Diagrama de Capas-MVC para la HU06.....	87
Figura 17 UI para la HU06: Acceder a actas de audiencia de un expediente .....	88
Figura 18 Diagrama de Capas-MVC para la HU07.....	89
Figura 19 UI para la HU07: Acceder a resoluciones de un expediente .....	89
Figura 20 Diagrama Capas-MVC para la HU08.....	90
Figura 21 UI para la HU08: Acceder a documentos digitalizados de un expediente .....	91
Figura 22 Diagrama de Capas-MVC para la HU09.....	92
Figura 23 UI para la HU09: Consultar información sobre depósitos judiciales .....	93
Figura 24 Diagrama de Capas-MVC para la HU10.....	94
Figura 25 UI para la HU10: Consultar el registro de firmas en un expediente.....	94
Figura 26 UI para la HU11: Seleccionar el modo de recepción de documentos .....	95
Figura 27 Diagrama de Capas-MVC para la HU11.....	96
Figura 28 UI para la HU12: Consultar información sobre principio de oportunidad .....	97
Figura 29 Diagrama de Capas-MVC para la HU12.....	98
Figura 30 UI para la HU13: Navegar entre diferentes expedientes .....	99
Figura 31 Diagrama de Capas-MVC para la HU13.....	100
Figura 32 Diagrama de Capas-MVC para la HU14.....	101

<b>Figura 33</b> Diagrama Capas-MVC para la HU15 .....	103
<b>Figura 34</b> <i>Arquitectura de la aplicación</i> .....	104
<b>Figura 35</b> <i>Arquitectura del dominio de la aplicación</i> .....	105
<b>Figura 36</b> <i>Diagrama de Base de Datos de la aplicación</i> .....	106

# PRESENTACIÓN

En el contexto actual donde la transformación digital es un elemento clave para mejorar la eficiencia en los servicios públicos, existe la necesidad de implementar de soluciones tecnológicas innovadoras que hagan frente a estas necesidades. Esta problemática se manifiesta claramente en el Módulo Penal CSJ Santa, donde es muy importante que la atención a los usuarios sea rápida y precisa, y que la documentación judicial esté disponible en todo momento mejorando de esta manera la transparencia y eficiencia institucional.

Con el objetivo claro y definido: “optimizar la atención y consulta de expedientes judiciales” se desarrolló un asistente virtual que, en base a la experiencia, representa un avance significativo en la automatización de estos procesos. Los usuarios autorizados pueden ahora acceder de manera segura a la información procesal a través de una interfaz conversacional intuitiva y eficiente.

Esta solución se integra a los sistemas propios del Módulo Penal CSJ Santa, y se ha observado que mejora considerablemente la comunicación, reduce los tiempos de respuesta y, probablemente lo más importante, descongestiona los canales tradicionales de atención que suelen estar saturados.

La estructura del presente trabajo se organiza de la siguiente manera:

1. **Capítulo I:** Presentación del tema de estudio.
2. **Capítulo II:** Descripción del contexto de la experiencia profesional.
3. **Capítulo III:** Importancia de la experiencia en el ejercicio profesional.
4. **Capítulo IV:** Exposición de los objetivos generales y específicos alcanzados.
5. **Capítulo V:** Fundamento teórico relacionado al tema abordado.
6. **Capítulo VI:** Organización y sistematización de las actividades realizadas.
7. **Capítulo VII:** Relación entre la experiencia desarrollada y el marco teórico.

8. **Capítulo VIII:** Aportes obtenidos para el desarrollo del centro laboral.

9. **Capítulo IX:** Aportes a la formación profesional del autor.

10. **Capítulo X:** Conclusiones y recomendaciones finales del trabajo.

Se tiene la expectativa que el presente proyecto sirva como referencia para futuras implementaciones de tecnologías conversacionales (chat bots), con la motivación de que el desarrollo de esta solución se pueda replicar y mejorar, siempre con el objetivo de contribuir a mejorar el acceso a la justicia y la calidad del servicio al ciudadano.

## RESUMEN

Este trabajo de suficiencia profesional se llevó a cabo en el Módulo Penal de la Corte Superior de Justicia del Santa, con el objetivo de implementar un asistente virtual que permita consultar expedientes judiciales a través de un chat directo con los usuarios, utilizando la plataforma de mensajería Telegram. La idea surgió como respuesta a la necesidad de agilizar el acceso a la información judicial para usuarios autorizados, al mismo tiempo que se optimiza el servicio, se reducen los tiempos de atención y se alivia la carga de los canales presenciales.

Teniendo en cuenta las anteriores consideraciones, el enfoque seleccionado en primera instancia fue el de una solución sólida y con una arquitectura orientada a la seguridad, disponibilidad y escalabilidad de todo el proyecto. Esto permitió integrar el asistente virtual a los sistemas de administración interna del Módulo Penal, asegurándonos su continuidad y disponibilidad sin interrupciones. Para implementar el proyecto se seleccionó trabajar con la metodología Scrum, un marco ágil que nos permitió organizar el proyecto de forma estructurada y flexible, siempre desde la base de la iteración desde el análisis hasta la implementación.

En cuanto al desarrollo técnico, se usó Spring Boot para crear la parte interna y estructural del sistema, y la API oficial de Telegram permitió manejar la comunicación en tiempo real con los usuarios de forma eficiente.

Esta integración implementada facilita que el asistente proporcione información procesal relevante: estado de expedientes, fechas de audiencias y datos complementarios. Esto se logra mediante medidas de autenticación que, mediante un análisis previo, garantizan el acceso exclusivo a personas autorizadas.

Al terminar el desarrollo, se realizaron pruebas tanto técnicas como con usuarios, y los resultados fueron muy positivos con una mejora evidente en la velocidad y la calidad del servicio de información. En resumen, este asistente virtual representa un paso concreto hacia la modernización del sistema judicial, alineado con los esfuerzos del Estado por lograr una gestión más eficiente, transparente y digital.

AUTOR:

- Bach. Moreno Saavedra, Marco Josué

ASESOR:

- Dr. Sánchez Chávez, Juan Pablo

## ABSTRACT

This professional competency project was carried out at the Criminal Module CSJ-Santa, with the aim of implementing a virtual assistant capable of providing access to judicial case files through a direct chat with users, using the Telegram messaging platform. The idea emerged in response to the need to streamline access to judicial information for authorized users, while also optimizing the service, reducing response times, and relieving pressure on in-person service channels. Given these considerations, the initial approach focused on building a robust solution with an architecture designed around **security, availability, and scalability**. This made it possible to integrate the virtual assistant with the internal management systems of the Criminal Module, ensuring its continuity and uninterrupted availability. To carry out the project, the Scrum methodology was chosen—an agile framework that allowed the work to be organized in a structured yet flexible way, with a strong emphasis on iteration from analysis through to implementation. In terms of technical development, Spring Boot was used to build the system’s core logic and structure, while the official Telegram API enabled efficient, real-time communication with users. The virtual assistant securely delivers key judicial information—such as case status and hearing dates—to authorized users. After successful technical and user testing, the solution proved to enhance both speed and quality of service, marking a significant step toward modernizing the judicial system in line with the State’s push for digital efficiency and transparency.

### AUTHOR:

- Bachelor Moreno Saavedra, Marco Josué

### ADVISOR:

- Dr. Sánchez Chávez, Juan Pablo

## **I. Tema específico abordado**

“Desarrollo de un Asistente Virtual para la consulta automatizada de Expedientes Judiciales en el Módulo Penal CSJ Santa”

## **II. Contextualización de la experiencia profesional**

### **2.1. Cronología de la Trayectoria Profesional**

#### **2.1.1. Corte Superior de Justicia de Santa**

La Corte Superior de Justicia del Santa es una entidad que administra justicia con equidad y transparencia, a través de sus órganos jurisdiccionales, siguiendo los principios de la Constitución y a las leyes, garantizando la seguridad y tutela jurisdiccional, para contribuir al estado de derecho, a la paz social y al desarrollo del país

**Tiempo en la institución:** Del 01/01/2019 al 31/05/2023

**Puesto:** Asistente Judicial NCPP

##### **2.1.1.1. Principales actividades**

El desarrollo del asistente virtual para la consulta de expedientes judiciales conllevó una serie de actividades técnicas que se ejecutaron de forma coordinada con el fin de garantizar su funcionamiento óptimo.

El desarrollo del sistema comenzó con la fase de análisis y diseño, en el que se ha realizado un diagnóstico de los procesos que son materia de estudio en el Módulo Penal CSJ Santa y se ha identificado que los usuarios del sistema serán fiscales, abogados y personal administrativo. En esta etapa se han definido los requerimientos funcionales como las modalidades de búsqueda de expedientes y tipos de documentos que se van a poder consultar, para luego crear las historias de usuarios que plasmen estos requerimientos.

Durante el desarrollo, se diseñó la interfaz conversacional del asistente virtual, estableciendo flujos de diálogo que, según el análisis realizado, logran una interacción natural entre usuarios y sistema. Esta fase requirió múltiples iteraciones para optimizar la experiencia conversacional.

La arquitectura técnica fue planificada minuciosamente y que determinó la metodología de conexión con las bases de datos institucionales existentes. Según nuestro criterio, esta integración representó uno de los mayores desafíos técnicos del proyecto. Paralelamente, se implementó medidas de seguridad específicas para proteger la información de los expedientes judiciales, considerando que el manejo de datos sensibles exige estándares de protección elevados.

La fase de desarrollo se constituyó en la etapa principal donde se ha construido el sistema propiamente dicho, teniendo como punto de partida la programación del backend, que dicho de paso es el componente desde donde se gestiona todas las consultas a la base de datos. Se han creado módulos para buscar expedientes por número o por DNI del imputado. También se han implementado funcionalidades para recuperar documentos digitalizados, actas de audiencia y resoluciones. Se ha creado la lógica que controla los permisos de acceso según el tipo de usuario, de modo que cada usuario sólo ha podido acceder y ver la información que es suya.

También se ha implementado el sistema de autenticación que verifica la identidad de los usuarios, un aspecto que se consideró crítico desde el inicio del proyecto. Adicionalmente se crearon los servicios que han permitido enviar documentos tanto por correo electrónico como a través del chat.

La fase de integración, represento uno de los momentos más desafiantes del proyecto. Conectar todos los componentes desarrollados para que operen como un sistema unificado, requirió múltiples iteraciones y ajustes que inicialmente no se contemplaron.

La comunicación entre el asistente virtual y la plataforma de mensajería, fue un proceso que implicó configurar servicios y APIs específicos para garantizar el flujo correcto de mensajes. La integración con las bases de datos del Poder Judicial fue particularmente meticulosa: se realizaron pruebas exhaustivas de conectividad y configuraciones de accesos seguros siguiendo los protocolos institucionales. Finalmente, se implementó los servidores de producción, configurando el entorno operativo y estableciendo conexiones estables con los servicios de correo electrónico.

En la fase de pruebas, se desarrolló un protocolo para verificar el funcionamiento del sistema en diversos escenarios operativos. Las pruebas de funcionalidad que se ejecutaron confirmaron la precisión de las búsquedas de expedientes y la correcta entrega de documentos. Cada componente fue evaluado sistemáticamente.

Las pruebas de rendimiento constituyeron un aspecto crítico del proceso de validación. Se debía asegurar respuestas eficientes del

sistema ante concurrencia múltiple de usuarios. Las pruebas de seguridad implementadas garantizaron acceso restringido a usuarios autorizados y transición segura de información judicial.

Durante las pruebas de usabilidad con usuarios finales, se pudo constatar que la interfaz conversacional desarrollada resultaba intuitiva y funcional. Esta validación con usuarios reales confirmó la efectividad del diseño conversacional implementado

La fase de implementación y mantenimiento marcó el inicio de la operación en un entorno real. Se configuró un sistema de monitoreo continuo de los servidores para detectar posibles problemas de rendimiento o disponibilidad. En la experiencia aprendida, este monitoreo preventivo es fundamental para garantizar la estabilidad del sistema.

Se definió un plan de mantenimiento que contempla actualizaciones periódicas, respaldos de información y optimización de consultas a la base de datos. El objetivo era conservar tiempos de respuesta eficientes a largo plazo. También se brindó soporte técnico a los usuarios, atendiendo consultas sobre el uso del sistema y resolviendo inconvenientes.

La generación de reportes periódicos sobre el uso del sistema ha permitido identificar oportunidades de mejora y planificar futuras actualizaciones.

#### **2.1.1.2. Logros obtenidos**

Durante el desarrollo del sistema, se implementó una arquitectura de autenticación y autorización que diferencia tipos de usuarios del

Módulo Penal CSJ Santa. Los perfiles específicos que se crearon para fiscales, abogados y personal administrativo garantizan el acceso restringido a información correspondiente a cada rol y nivel de permisos institucional.

La selección de la API para el asistente virtual constituyó una decisión técnica imprescindible, donde se evaluó la flexibilidad, apertura y el costo. Se optó por la API de Telegram, asegurando la compatibilidad universal: dispositivos móviles, computadoras de escritorio y navegadores web. En nuestro análisis, facilitar acceso desde cualquier dispositivo con Telegram preinstalado optimizaba la adopción del sistema, eliminando barreras de instalación adicional.

La integración con bases de datos existentes del Poder Judicial del Santa permitió automatizar integralmente el proceso de consulta de expedientes judiciales. Esta integración eliminó dependencia de búsquedas manuales en archivos físicos y redujo sustancialmente los tiempos de respuesta para acceso a información judicial.

Durante la automatización del proceso de entrega de documentos, se implementó canales múltiples: distribución por correo electrónico y entrega directa mediante chat. Esta funcionalidad garantiza que usuarios reciban documentos solicitados inmediatamente, optimizando la eficiencia del servicio y minimizando tiempos de espera.

Complementariamente, se desarrolló un sistema de notificaciones automáticas que informa sobre estado de consultas y disponibilidad de documentos solicitados. En la evaluación técnica, este componente

fortaleció significativamente la comunicación con usuarios y redujo la frecuencia de consultas reiterativas respecto al estado de solicitudes procesadas.

### **2.1.1.3. Aprendizaje**

A lo largo del proyecto, se consolidó mi experiencia en el desarrollo y configuración de chatbots mediante la integración con la API de Telegram. Este proceso me permitió dominar el manejo de webhooks, la configuración de respuestas automáticas y la gestión de flujos conversacionales complejos. Logré crear una interacción que considero fluida y natural entre usuarios y sistema. Se profundizó significativamente en la implementación de sistemas de autenticación y autorización robustos empleando Spring Security. Experiencia valiosa a la hora de trabajar con información sensible del ámbito judicial. Se aprendió a configurar tokens JWT, validar permisos de usuario y aplicar controles de acceso basados en roles específicos del entorno judicial.

También adquirí conocimiento práctico en el desarrollo de aplicaciones backend utilizando Spring Boot, junto con la creación de APIs REST. En mi experiencia, esto implicó gestionar servicios web complejos e implementar arquitecturas escalables capaces de procesar múltiples consultas simultáneas eficientemente. En el ámbito de bases de datos, fortalecí mi competencia en integración y optimización de sistemas relacionales para consultas intensivas mediante Spring Data JPA. Este trabajo abarcó el diseño de consultas eficientes, la optimización de índices y la implementación de estrategias de

respaldo. Garantizar la integridad de la información judicial fue una prioridad constante.

Desarrollé capacidades importantes en la automatización de procesos de entrega documental a través de múltiples canales de comunicación. La integración de servicios REST con la API de Telegram incluyó configurar servicios de correo electrónico, gestionar archivos digitales e implementar sistemas de notificación que mejoran la experiencia del usuario final.

Finalmente, se obtuvo experiencia valiosa en el monitoreo y mantenimiento de sistemas en producción basados en Spring Boot. Esto incluyó configurar alertas automáticas, analizar registros del sistema y aplicar estrategias de recuperación ante fallos para asegurar la continuidad operativa del chatbot y las APIs REST asociadas.

## **2.2. Contexto en que se desarrolló la experiencia**

### **2.2.1. La Institución**

#### **2.2.1.1. PJ Corte Superior de Justicia de Santa**

La Corte Superior de Justicia del Santa es una institución encargada de administrar justicia con imparcialidad y transparencia, mediante sus órganos jurisdiccionales y en conformidad con la Constitución y el marco legal vigente. Su actuación asegura la protección de los derechos ciudadanos, consolidando el estado de derecho, fomentando la convivencia pacífica y aportando al progreso del país.

#### **2.2.1.2. Misión**

Administrar justicia con equidad y transparencia, a través de sus órganos jurisdiccionales, con arreglo a la constitución y a las leyes,

garantizando la seguridad y tutela jurisdiccional, para contribuir al estado de derecho, a la paz social y al desarrollo del país.

Para el cumplimiento cabal de esta misión, se cuenta con magistrados respetados, probos, capaces, éticos y justos, con competencias que respondan a las exigencias de una sociedad moderna.

#### **2.2.1.3. Visión**

Ser una institución autónoma y eficiente, integrada por magistrados y servidores con vocación de servicio, comprometidos en lograr la excelencia, modernidad y superación constante, que privilegie la solución de conflictos, el acceso oportuno y en condiciones de igualdad a la justicia, respetando los derechos de las personas, contando para ello con un adecuado soporte administrativo y tecnológico

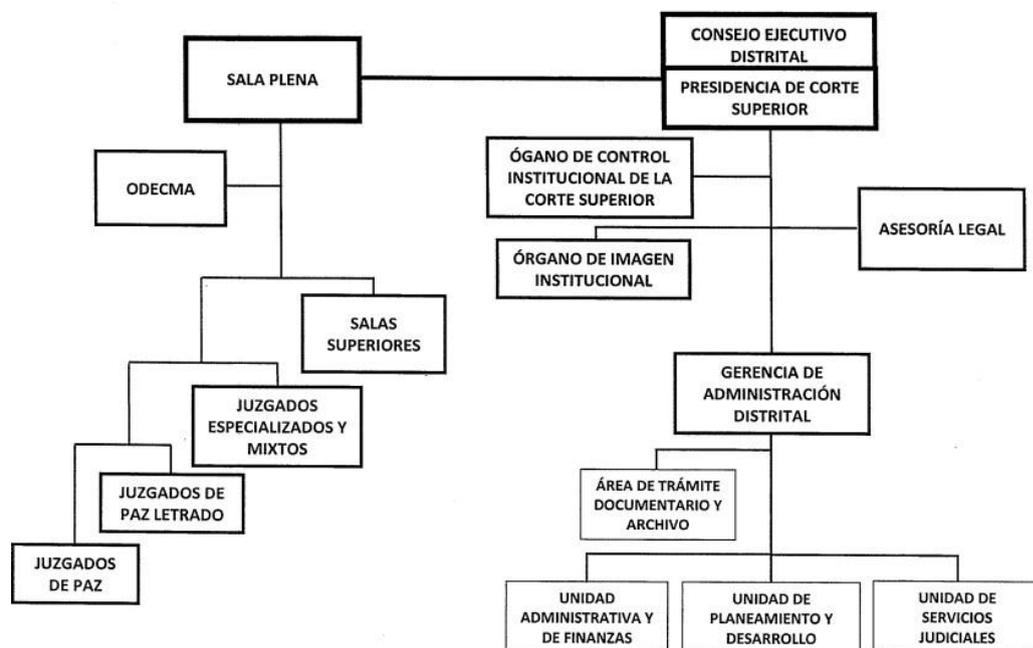
#### **2.2.1.4. Ubicación**

La sede central de la Corte Superior de Justicia del Santa se encuentra en: Av. Pardo N°- Chimbote - Santa - Áncash - Perú.

### 2.2.1.5. Organigrama de la CSJ de Santa

Figura 1

Organigrama CSJ Santa



Nota. Elaboración propia

### 2.2.2. Contexto del proyecto

La implementación de este asistente virtual para consultas automatizadas de expedientes judiciales en el módulo penal CSJ Santa, representa un hito en la modernización de servicios judiciales y el acceso a la información pública.

Durante el análisis de los procesos bajo estudio en el Módulo Penal se observó que se maneja diariamente una gran cantidad de expedientes judiciales que requieren consulta por parte de fiscales, abogados y personal administrativo. Tradicionalmente, estas consultas se realizaban de manera presencial en las oficinas del módulo penal o simplemente mediante llamadas telefónicas, con un resultado bastante predecible: demoras, colas

de espera y una carga considerable de trabajo para el personal administrativo.

Para enfrentar esta problemática, el Poder Judicial del Santa a través de la la Unidad de Coordinación de Informática, se decidió implementar un asistente virtual que automatice las consultas de expedientes judiciales, sistema que permite a los usuarios autorizados acceder a la información que necesitan de manera más rápida y eficiente, sin necesidad de desplazamientos físicos a las oficinas ni esperas telefónicas.

El asistente virtual opera como un intermediario entre los usuarios y las bases de datos del sistema judicial, donde se desarrolló diferentes formas de búsqueda según la información disponible:

- **Consulta por número de expediente:** En el caso de tener el número de expediente y se desea tener acceso a toda la información sobre el caso en cuestión, es factible recurrir a esta opción.
- **Consulta por DNI del imputado:** En aquellos casos donde no se conoce el número de expediente, pero sí se dispone del DNI de la persona imputada, puede elegirse esta alternativa para que el sistema pueda encontrar el expediente involucrado.
- **Acceso a documentos digitalizados:** Una vez que el sistema ha encontrado el expediente que se busca, se puede acceder a la obtención de los documentos como, por ejemplo, el acta en la que se registran las actuaciones judiciales y las resoluciones de la audiencia.
- **Funciones especializadas:** en donde la aplicación puede llevar a cabo búsquedas de depósitos judiciales, firmas, entre otros, según las necesidades de cada usuario.

Mediante la implementación de el asistente virtual se logró una mejora significativa en la atención a los usuarios del sistema judicial. Es posible acceder hoy en día a los expedientes judiciales, de manera automatizada, con la reducción de tiempos de espera y la liberación del tiempo para personal administrativo. Esto ayuda a los usuarios porque obtienen respuestas más rápidas y a la institución ya que puede brindar un mejor servicio con los mismos recursos.

**Labor realizada:** El desarrollo y mantenimiento de este sistema está a cargo del equipo técnico del área de informática del Poder Judicial del Santa, quienes utilizan su conocimiento de los procesos judiciales y las bases de datos institucionales para garantizar que el sistema funcione correctamente y mantenga la seguridad de la información.

### **III. Importancia para el ejercicio de la carrera profesional**

El desarrollo de este tipo de proyectos, específicamente el que es materia de este estudio: El **Asistente** Virtual para la Consulta Automatizada de Expedientes Judiciales, representa una oportunidad valiosa para el crecimiento profesional en Ingeniería de Sistemas e Informática. Este tipo de proyectos ofrece la oportunidad de aplicar conocimientos teóricos en contextos reales, utilizando tecnologías modernas de desarrollo web y aplicaciones móviles que he visto ampliamente adoptadas en la industria actual. La experiencia de construir un sistema que facilite el acceso automatizado a información judicial me ha permitido formar una base técnica sólida para abordar desafíos similares en otros sectores. Durante el desarrollo, he gestionado múltiples componentes técnicos que se consideró esenciales en el ejercicio profesional: diseño de interfaces de usuario, creación de servicios web, administración de bases de datos e implementación de mecanismos de seguridad.

El trabajo articulado de estos elementos permitió desarrollar una comprensión integral del proceso de desarrollo de software, desde la planificación hasta la implementación operativa. La gestión de información sensible, como los expedientes judiciales, hizo valorar profundamente la importancia de aplicar buenas prácticas en seguridad y privacidad de datos.

Una de las ventajas más destacadas del proyecto, radica en la necesidad de considerar la experiencia del usuario final. Diseñar un sistema accesible para fiscales, abogados y personal administrativo exigió comprender sistemáticamente todas sus necesidades específicas y traducirlas en soluciones tecnológicas intuitivas. Esta habilidad de transformar requerimientos complejos en interfaces simples es altamente valorada profesionalmente.

El proyecto también posee, desde nuestro punto de vista, un impacto social relevante. Su implementación, orientada a mejorar la eficiencia de los servicios públicos, ha mostrado el papel transformador que puede desempeñar la tecnología en el fortalecimiento institucional. Esta experiencia me ha promovido una mentalidad enfocada en crear soluciones con valor social, contribuyendo a formarme como profesional consciente del alcance de mi labor y comprometido con un uso ético de la tecnología.

#### **IV. Objetivos logrados**

##### **4.1. Objetivo General**

Crear un asistente virtual que permita la consulta automatizada de expedientes judiciales en el Módulo Penal CSJ Santa, con acceso autorizado solo a los usuarios vinculados al servicio y/o a los procesos judiciales, a través de un Chat.

##### **4.2. Objetivos Específicos**

- Diseñar la arquitectura del asistente virtual, considerando seguridad, accesibilidad y escalabilidad para la consulta de expedientes.
- Desarrollar una interfaz conversacional accesible, que facilite la interacción del usuario en la consulta judicial.
- Integrar el asistente con el sistema de gestión de expedientes para garantizar acceso en tiempo real a la información, manteniendo siempre la confidencialidad de los datos.
- Implementar mecanismos de autenticación que permitan controlar quién puede acceder a la información, asegurando que solo los usuarios autorizados la consulten.
- Evaluar qué tan preciso y fácil de usar es el asistente, realizando pruebas con usuarios reales y analizando los resultados obtenidos.

## **V. Sustento teórico del tema abordado**

### **5.1. Antecedentes**

#### **5.1.1. Investigaciones Internacionales:**

##### **Antecedente 01**

Vassilopoulos, A. (2022)., en su tesis de maestría “Can AI chatbots improve access to legal information?”, explora el papel que pueden desempeñar los chatbots basados en inteligencia artificial en la mejora del acceso a la información legal. Su investigación parte de una premisa clara: muchas personas no tienen acceso a servicios legales tradicionales, ya sea por limitaciones económicas o por barreras de conocimiento. Ante esta realidad, el uso de asistentes virtuales se presenta como una alternativa innovadora y potencialmente transformadora.

El estudio se enfoca en cómo estas herramientas pueden democratizar el conocimiento jurídico, haciéndolo más accesible para el público en general. Se examina en detalle la capacidad de los chatbots para ofrecer respuestas precisas, comprensibles y útiles a preguntas legales frecuentes, lo que podría traducirse en una mayor autonomía y empoderamiento ciudadano frente a trámites o situaciones legales básicas.

Sin embargo, la investigación también aborda los desafíos que acompañan a esta tecnología. Entre ellos, destacan la necesidad de asegurar la precisión de la información proporcionada, la protección de la privacidad de los datos de los usuarios y la importancia de una supervisión humana constante para evitar errores o interpretaciones erróneas. Estos aspectos son cruciales para evitar consecuencias negativas que podrían surgir del uso no controlado de estas herramientas en un ámbito tan sensible como el legal.

Como resultado de su análisis, Vassilopoulos concluye que los chatbots jurídicos pueden ser una herramienta poderosa para mejorar el acceso a la justicia, siempre que su diseño y funcionamiento estén guiados por principios de claridad, exactitud y ética. En lugar de sustituir completamente a los servicios legales profesionales, sugiere que estas tecnologías pueden cumplir un rol complementario, aliviando la carga sobre el sistema legal y brindando asistencia básica a quienes más lo necesitan.

En este sentido, propone un marco orientado al desarrollo responsable de chatbots legales, que contemple tanto las necesidades del usuario como las exigencias del contexto legal en el que operan. Un enfoque centrado en el usuario, pero con la rigurosidad técnica y jurídica necesaria, podría marcar la diferencia entre un simple asistente automatizado y una verdadera herramienta de acceso a derechos.

### **Antecedente 02**

En el artículo “*Artificial Intelligence for Legal Chatbot*”, Suryawanshi y Suryawanshi (2024) analizan el papel de la inteligencia artificial en el desarrollo de chatbots orientados al ámbito legal. Su enfoque parte de una pregunta clave: ¿cómo puede la tecnología transformar la manera en que se ofrecen los servicios jurídicos? A partir de esta interrogante, los autores examinan cómo los chatbots pueden ofrecer asistencia automatizada y acceso inmediato a información legal, reduciendo barreras tradicionales como los costos elevados o la complejidad del lenguaje jurídico.

El estudio profundiza en los aspectos técnicos fundamentales para construir un chatbot legal efectivo. Se abordan elementos como el procesamiento del lenguaje natural (PLN) y el aprendizaje automático, tecnologías clave que

permiten a estos sistemas interpretar consultas complejas y responder de manera coherente. La arquitectura de estos chatbots, según los autores, debe estar cuidadosamente diseñada para garantizar un funcionamiento ágil, preciso y adaptable a distintos contextos legales.

Entre los beneficios que destacan, se encuentran la mejora en la eficiencia operativa de los servicios legales, la reducción de costos tanto para proveedores como para usuarios, y la posibilidad de ampliar el acceso a la justicia, especialmente para personas o pequeñas empresas que normalmente no pueden costear asesoría legal profesional.

No obstante, el artículo no omite los desafíos. Se señala, por ejemplo, la dificultad de procesar correctamente el lenguaje legal, que suele ser técnico, ambiguo o con múltiples interpretaciones. Además, se subraya la necesidad de contar con una base de conocimientos jurídica robusta y constantemente actualizada, así como mecanismos que garanticen la precisión y fiabilidad de las respuestas ofrecidas por el chatbot.

A modo de conclusión, los autores sostienen que, pese a los obstáculos técnicos y éticos que aún deben resolverse, los chatbots legales representan una frontera innovadora en la convergencia entre inteligencia artificial y derecho. Su potencial para revolucionar la prestación de servicios legales es significativo, siempre que se les desarrolle con responsabilidad, rigurosidad y una clara orientación al usuario.

### **Antecedente 03**

Al-Ayyoub, M., & Al-Zoubi, A. (2025), en su artículo “Generative vs Intent-based Chatbot for Judicial Advice.” compara dos enfoques principales para el desarrollo de chatbots de asesoramiento judicial: los chatbots generativos

y los basados en intenciones. Los autores investigan las ventajas y desventajas de cada tipo en el contexto de proporcionar orientación legal automatizada. Los chatbots basados en intenciones se centran en reconocer patrones predefinidos en las consultas de los usuarios y proporcionar respuestas específicas de una base de conocimientos. Por otro lado, los chatbots generativos utilizan modelos de lenguaje avanzados para crear respuestas nuevas y coherentes, lo que les permite manejar una gama más amplia de consultas y ofrecer interacciones más fluidas. El estudio evalúa el rendimiento de ambos tipos en términos de precisión, relevancia y satisfacción del usuario en el dominio judicial. Se discuten las implicaciones de cada enfoque para la implementación práctica de sistemas de asesoramiento legal, destacando la importancia de equilibrar la flexibilidad con la fiabilidad en un campo tan crítico como el derecho. Los hallazgos sugieren que, si bien los chatbots generativos ofrecen mayor versatilidad, los basados en intenciones pueden ser más adecuados para tareas que requieren alta precisión y control sobre la información proporcionada, especialmente en el contexto de expedientes judiciales.

### **5.1.2. Investigaciones Nacionales**

#### **Antecedente 04**

En su tesis titulada *“El Expediente Digital y su incidencia en la Administración de Justicia en el Perú”*, Quispe (2018) analiza el impacto que tiene la digitalización de los expedientes judiciales en el funcionamiento del sistema de justicia peruano. El estudio se construye sobre una base empírica compuesta por opiniones y experiencias de magistrados, personal

administrativo y abogados litigantes, lo que le otorga una perspectiva amplia sobre el tema.

La investigación adopta un enfoque cualitativo, utilizando muestras no probabilísticas, y se orienta hacia el análisis de propuestas internacionales sobre expedientes digitales. Además, se incluyen referencias al derecho comparado, una evaluación crítica de los avances tecnológicos implementados en el sistema judicial peruano y una lectura atenta de las percepciones de quienes participan activamente en la administración de justicia.

Los hallazgos del estudio muestran con claridad que la adopción del expediente judicial electrónico conlleva importantes beneficios. Entre los más relevantes, se encuentran la mejora en la interoperabilidad entre distintas instancias del sistema, el fortalecimiento del control y acceso a la información, la aceleración de los procesos judiciales y un incremento en la transparencia institucional.

Más allá de las ventajas técnicas, el expediente digital se presenta como una herramienta que contribuye directamente a acercar la justicia a la ciudadanía. Al reducir los tiempos de espera y simplificar procedimientos, se transforma en un elemento clave para una justicia más eficiente, accesible y confiable.

#### **Antecedente 05**

Morocco Sucapuca, A. L. (2023), en su tesis “El acceso a la justicia a través del uso del aplicativo Chatbot-Pj de los litigantes de la Corte Superior de Justicia de Puno -2023”, estudió el uso del aplicativo Chatbot-PJ como un mecanismo de acceso a la justicia, el referido fue implementado mediante Resolución Administrativa Nro. 000032-2022-CE-PJ, mediante la

herramienta virtual “El servicio de justicia en tus manos”. Este dispositivo permite simular conversación mediante el aplicativo WhatsApp con un asistente virtual de la Corte Superior de Justicia de Puno, debiendo agendar el número de WhatsApp, luego accederán a diferentes servicios como: Demandas, alimentos vía web, notificaciones, estado de expediente, depósito judicial, quejas, orientación judicial, entrevista con juez, certificado judicial, directorios. Los objetivos planteados en la investigación fueron las siguientes: Fundamentar si el aplicativo Chatbot-PJ constituye como un mecanismo de acceso a la justicia de los litigantes del Distrito Judicial de Puno. La investigación efectuada presenta un método mixto (cuantitativo y cualitativo), el propósito del estudio fue medir los resultados obtenidos mediante el diseño descriptivo. De los resultados el aplicativo Chatbot-PJ si constituye un mecanismo de acceso de la justicia de los litigantes del Distrito Judicial de Puno debido a que brinda información sobre las prestaciones judiciales que ofrece la Corte Superior de Justicia de Puno. Como conclusión se afirma que el aplicativo Chatbot-PJ es de utilidad para los litigantes del Distrito Judicial de Puno. La aportación de este estudio fue enriquecer y difundir la calidad y servicio de los medios de atención al usuario judicial con las que cuenta el Poder Judicial.

### **Antecedente 06**

Minaya, M., Saavedra, A. (2024), en su tesis “Implementación de chatbot logístico para mejorar la gestión en servicios farmacéuticos en la droguería Lidufarma, Santa Anita, Lima, Perú – 2023”, estudio la optimización de la gestión del servicio en el área farmacéutica mediante la incorporación de un chatbot logístico. El enfoque metodológico fue cuantitativo, de tipo aplicado,

con diseño pre-experimental, de corte longitudinal y basado en el método hipotético-deductivo. Se aplicó la técnica de encuesta a una muestra de 46 clientes, utilizando un cuestionario como instrumento. Los datos recolectados fueron analizados mediante estadística descriptiva e inferencial.

Antes de la implementación del chatbot, se identificaron deficiencias significativas: el 50% de los usuarios percibía el tiempo de atención como malo, el 78,3% calificaba la calidad del servicio como regular y el 63% consideraba su nivel de satisfacción también como regular. Posteriormente a la implementación del sistema conversacional, se observó una mejora sustancial: el 93,5% calificó el tiempo de atención como excelente, el 84,8% consideró excelente la calidad del servicio y el 91,3% manifestó un alto nivel de satisfacción. En general, se concluyó que la gestión de los servicios farmacéuticos alcanzó un nivel excelente en un 80,4% de los casos.

Este antecedente evidencia la efectividad del uso de chatbots en sectores altamente demandantes, como el farmacéutico, y respalda la aplicabilidad de estas tecnologías para mejorar la atención al cliente, reducir tiempos de respuesta y aumentar la calidad del servicio, lo que lo convierte en un aporte relevante para investigaciones similares en otros contextos organizacionales.

### **5.1.3. Investigaciones Locales**

#### **Antecedente 07**

De, T., (2022), en su tesis “Implementar un chatbot basado en ibm watson para mejorar el servicio de información recurrente en la empresa inversiones Santa SRL, 2022”, abordó la necesidad de modernizar los canales de atención al cliente frente a una creciente demanda de información inmediata y eficaz,

en un entorno marcado por el avance de las tecnologías de la información y la transformación digital de los servicios

La investigación tomó como caso de estudio al restaurante “El Fogón”, un establecimiento reconocido en la ciudad de Huaraz. En este contexto, se diseñó e implementó un chatbot utilizando la tecnología IBM Watson, con el objetivo de automatizar la atención de consultas frecuentes realizadas por los clientes. El enfoque metodológico fue de tipo aplicado, con un diseño preexperimental. La muestra estuvo compuesta por 36 clientes seleccionados mediante un muestreo aleatorio simple, lo que permitió obtener resultados representativos sobre la experiencia del usuario.

Para evaluar el impacto del chatbot, se aplicaron encuestas antes y después de su implementación. Los resultados evidenciaron una mejora notable en los tiempos de atención, disminuyendo las demoras habituales y aumentando la eficiencia en la entrega de información. Los datos recogidos fueron analizados mediante la prueba estadística de Wilcoxon, la cual confirmó la significancia de los cambios observados.

Más allá de los indicadores cuantitativos, el chatbot fue bien recibido por los usuarios gracias a su capacidad para simular conversaciones naturales y resolver dudas en tiempo real. Este tipo de interacción generó una experiencia positiva y demostró el potencial de la inteligencia artificial como herramienta de apoyo en la atención al cliente.

En conjunto, este estudio ofrece una base sólida para comprender los beneficios del uso de chatbots en entornos empresariales. Los hallazgos refuerzan la idea de que la automatización, cuando se implementa adecuadamente, no solo optimiza procesos, sino que también mejora la

percepción del servicio. Por ello, este caso se convierte en un referente relevante para futuras investigaciones que aborden la digitalización de servicios informativos en distintos sectores.

### **Antecedente 08**

Castillo, R., (2022), en su tesis “Chatbot para mejorar la atención de consultas de los estudiantes en la institución educativa niño Jesús de Praga” plantea como objetivo principal optimizar la eficiencia y la calidad del servicio brindado mediante un chatbot diseñado para atender las consultas estudiantiles dentro de dicha institución. La investigación se enmarca en un enfoque cuantitativo y adopta un diseño experimental, prospectivo y longitudinal, siendo clasificada como una investigación de nivel aplicativo. Para llevar a cabo el estudio, se utilizaron diversas técnicas de recolección de datos que permitieron obtener una visión amplia del impacto del sistema. Entre estas, se incluyen encuestas, entrevistas y observación directa. Como instrumento central se empleó una guía estructurada de observación, la cual facilitó el análisis sistemático de las interacciones y permitió registrar con precisión los efectos de la implementación del chatbot en el entorno educativo.

El trabajo de Castillo aporta evidencia sobre la utilidad de este tipo de herramientas en el ámbito escolar, mostrando cómo la tecnología puede contribuir a mejorar la comunicación y atención hacia los estudiantes. Su enfoque metodológico y la variedad de técnicas aplicadas fortalecen la validez de los resultados, convirtiendo esta investigación en un aporte relevante para futuras iniciativas que busquen digitalizar y optimizar servicios de atención educativa. La población estuvo compuesta por un total de 288

estudiantes de los niveles inicial, primario y secundario. De esta población, se seleccionó una muestra de 64 alumnos pertenecientes al tercer y cuarto grado de secundaria.

Los resultados evidenciaron una mejora significativa en la atención ofrecida a los estudiantes tras la implementación del chatbot. Se observó una atención más rápida, eficiente y de mayor calidad, una notable reducción en el tiempo de respuesta a las consultas, así como una disminución en los costos que anteriormente debían asumirse para obtener dicha información.

## 5.2. Fundamentos teóricos de la investigación

### 5.2.1. Api Rest

Según Fielding, R. T. (2000)., REST (Representational State Transfer) es un estilo arquitectónico fundamental para el diseño de sistemas distribuidos de hipertexto, especialmente en el contexto de la World Wide Web. Fue definido por Roy Fielding en su disertación doctoral en el año 2000, y aunque no es un protocolo ni un estándar en sí mismo, proporciona un conjunto de principios y restricciones que guían la creación de servicios web escalables, eficientes y fáciles de mantener.

La idea central de REST es que la información se organiza en recursos. Un recurso es cualquier entidad que puede ser identificada, nombrada y manipulada. Esto puede incluir documentos, imágenes, servicios temporales, colecciones de otros recursos, o incluso objetos no virtuales como personas. La clave es que cualquier concepto que pueda ser el objetivo de una referencia de hipertexto debe encajar en la definición de un recurso. Un recurso es un mapeo conceptual a un conjunto de entidades, no la entidad que corresponde al mapeo en un momento dado

Los principios clave que definen el estilo arquitectónico REST son:

- **Cliente-Servidor:** La arquitectura REST separa las preocupaciones de la interfaz de usuario (cliente) de las preocupaciones de almacenamiento de datos (servidor). Esto mejora la portabilidad de la interfaz de usuario a través de múltiples plataformas y la escalabilidad del servidor.
- **Sin Estado (Stateless):** Cada solicitud del cliente al servidor debe contener toda la información necesaria para que el servidor la entienda

y la procese. El servidor no debe almacenar ningún contexto de la sesión del cliente entre solicitudes. Esto mejora la escalabilidad y la fiabilidad.

- **Cacheable:** Las respuestas del servidor deben indicar si los datos pueden ser almacenados en caché por el cliente. Esto mejora el rendimiento y la escalabilidad al reducir la necesidad de que el cliente realice solicitudes repetidas al servidor.
- **Interfaz Uniforme:** Este es el principio más importante y el que distingue a REST de otros estilos arquitectónicos. Requiere que los recursos se identifiquen de forma única (mediante URIs), que los recursos se manipulen a través de representaciones (ej. JSON, XML), que los mensajes sean auto-descriptivos (incluyendo metadatos sobre cómo procesar el mensaje), y que se utilice el hipermedia como motor del estado de la aplicación (HATEOAS), lo que significa que las interacciones con los recursos se guían por enlaces proporcionados en las representaciones de los recursos.
- **Sistema en Capas:** Un cliente no puede saber si está conectado directamente al servidor final o a un intermediario. Los servidores proxy, pasarelas y firewalls pueden ser introducidos entre el cliente y el servidor sin afectar las interacciones.
- **Código bajo Demanda (Opcional):** Los servidores pueden extender la funcionalidad del cliente transfiriendo código ejecutable (como JavaScript). Este principio es opcional.

Los servicios web que adhieren a estos principios se conocen como servicios web RESTful o APIs RESTful. Utilizan los métodos estándar del protocolo

HTTP (GET para recuperar datos, POST para crear, PUT para actualizar y DELETE para eliminar) para realizar operaciones sobre los recursos.

La simplicidad de REST, junto con su enfoque en el uso de estándares web ampliamente adoptados, ha sido una cuestión clave para su integración masiva en el desarrollo de aplicaciones tanto web como móviles. Gracias a estas características, las APIs REST han facilitado significativamente la interoperabilidad entre distintos sistemas y plataformas, lo que las convierte en una opción preferida para quienes buscan soluciones prácticas y escalables.

En la práctica, las APIs RESTful se han consolidado como el pilar estructural de muchas aplicaciones modernas. Desde redes sociales hasta plataformas de comercio electrónico y servicios en la nube, este tipo de arquitectura permite construir sistemas ágiles, distribuidos y capaces de operar de forma eficiente bajo altos niveles de demanda.

Su diseño desacoplado y sin estado las hace especialmente adecuadas para arquitecturas basadas en microservicios, donde la alta disponibilidad y la escalabilidad son fundamentales. Por ello, comprender los principios que sustentan REST no solo es recomendable, sino esencial para cualquier desarrollador que trabaje en el ecosistema web actual. Dominar estos conceptos permite construir aplicaciones más robustas, eficientes y sostenibles a lo largo del tiempo.

### **5.2.2. Chatbot**

Según Adamopoulou y Moussiades (2020), los chatbots, también conocidos como bots conversacionales, son programas informáticos diseñados para simular conversaciones humanas, permitiendo la interacción entre personas y sistemas digitales. Representan una de las aplicaciones más comunes de

la inteligencia artificial en el campo de la Interacción Persona-Computadora. Su funcionamiento se basa en técnicas de Procesamiento del Lenguaje Natural (PLN), que les permiten interpretar las intenciones del usuario y generar respuestas coherentes. Gracias a su versatilidad, pueden integrarse en diversas plataformas como sitios web, aplicaciones de mensajería (ej. Telegram, WhatsApp), o asistentes de voz. Su estudio continúa evolucionando, abordando no solo aspectos técnicos, sino también desafíos en diseño conversacional, usabilidad y ética en la interacción.

Actualmente, los chatbots se pueden clasificar en dos grandes categorías según su estructura y nivel de complejidad. Por un lado, se encuentran los *chatbots basados en reglas*, que funcionan siguiendo un conjunto fijo de instrucciones predefinidas. Estos sistemas son particularmente eficaces en tareas específicas y repetitivas, como responder preguntas frecuentes o guiar al usuario a través de procesos estructurados. Sin embargo, su capacidad de respuesta está limitada por las reglas establecidas, lo que reduce su flexibilidad ante consultas inesperadas o poco convencionales.

Por otro lado, los *chatbots basados en inteligencia artificial*, también conocidos como chatbots inteligentes, utilizan técnicas avanzadas de aprendizaje automático y procesamiento del lenguaje natural (PLN). A diferencia de los anteriores, estos modelos pueden aprender de interacciones previas, adaptarse a nuevos escenarios y manejar conversaciones más abiertas y complejas. Gracias a los avances en aprendizaje profundo, han mejorado significativamente en su capacidad para mantener diálogos naturales, interpretar el contexto y ofrecer respuestas más precisas y personalizadas.

Desde un enfoque evolutivo, el desarrollo de los chatbots ha sido notable. Desde los primeros intentos, como el emblemático *ELIZA* en los años 60, hasta los sistemas actuales basados en redes neuronales, se ha transitado de simples programas de respuesta a herramientas sofisticadas capaces de ejecutar tareas complejas, automatizar servicios y enriquecer la experiencia del usuario. Su popularidad se ha incrementado no solo por su capacidad para operar de forma continua (24/7), sino también por su impacto en la reducción de costos operativos y mejora en la eficiencia comunicacional en distintos contextos organizacionales.

En términos de aplicación, los chatbots han encontrado un lugar en múltiples sectores. En atención al cliente, ayudan a resolver dudas comunes y canalizar casos hacia personal especializado cuando es necesario. En el comercio electrónico, asisten a los usuarios en la búsqueda de productos, realizan recomendaciones y acompañan el proceso de compra. En el sector salud, ofrecen información preliminar sobre síntomas, facilitan la programación de citas y generan recordatorios. En el ámbito educativo, actúan como asistentes virtuales que responden consultas académicas y orientan en el uso de recursos. Finalmente, en el sector financiero, colaboran en gestiones bancarias, consultas de cuentas y realización de transacciones básicas.

No obstante, a pesar de los avances, aún persisten desafíos importantes. La interpretación de ambigüedades lingüísticas, la gestión de emociones humanas y la capacidad para mantener conversaciones prolongadas y coherentes son algunas de las limitaciones que se siguen investigando. La

mejora de estas capacidades sigue siendo un objetivo prioritario en el campo del PLN y del desarrollo de modelos más humanizados.

En conclusión, los chatbots representan una tecnología en constante evolución que está transformando la manera en que los usuarios interactúan con los sistemas. Su habilidad para automatizar la comunicación y ofrecer asistencia inmediata los convierte en una herramienta clave en la era digital. A medida que se perfeccionan sus capacidades, su impacto en la eficiencia operativa y en la experiencia del usuario promete seguir creciendo en los próximos años.

### **5.2.3. Api Chatbot Telegram**

Según Sebastian, D., & Nugraha, K. A. (2021), la API de Telegram Bot es una interfaz de programación de aplicaciones (API) que permite a los desarrolladores crear programas automatizados, conocidos como bots, que interactúan directamente con la plataforma de mensajería Telegram. Esta API facilita la comunicación bidireccional entre los usuarios de Telegram y los servicios externos, permitiendo a los bots enviar y recibir mensajes, gestionar grupos, y realizar una amplia gama de funciones a través de solicitudes HTTPS.

El funcionamiento de la API de Telegram Bot se basa en un conjunto de métodos y objetos que los desarrolladores pueden utilizar para interactuar con la plataforma. Los bots pueden ser controlados mediante el envío de comandos y la recepción de actualizaciones a través de webhooks o de la obtención de actualizaciones a largo plazo. Esto significa que un bot puede reaccionar en tiempo real a las interacciones de los usuarios, lo que lo convierte en una herramienta poderosa para la automatización y la prestación de servicios.

Las características clave de la API de Telegram Bot incluyen:

- Envío y recepción de mensajes: Los bots pueden enviar mensajes de texto, fotos, videos, documentos, ubicaciones, contactos y más, así como recibir estos tipos de mensajes de los usuarios.
- Teclados personalizados: Permite a los desarrolladores crear teclados personalizados con botones para guiar la interacción del usuario y simplificar la entrada de comandos.
- Modo inline: Los bots pueden funcionar en modo inline, lo que permite a los usuarios interactuar con el bot directamente desde cualquier chat de Telegram sin necesidad de añadir el bot a un grupo o iniciar una conversación privada.
- Webhooks: La API soporta webhooks, lo que permite a los desarrolladores configurar una URL a la que Telegram enviará actualizaciones en tiempo real, eliminando la necesidad de que el bot sondee constantemente los nuevos mensajes.
- Gestión de grupos: Los bots pueden ser miembros de grupos y canales, lo que les permite moderar contenido, enviar notificaciones o interactuar con los miembros del grupo.

La versatilidad de la API de Telegram Bot ha llevado a su adopción en una multitud de aplicaciones. Por ejemplo, en el ámbito académico, se ha utilizado para desarrollar chatbots de servicio al cliente que proporcionan información a estudiantes y administradores, automatizando consultas frecuentes y mejorando la eficiencia de la comunicación en instituciones educativas. Esto es particularmente útil en entornos donde la gestión de

múltiples canales de comunicación puede ser un desafío, ya que un bot puede centralizar y optimizar la interacción con los usuarios.

La integración de funcionalidades de chatbot directamente en la plataforma de mensajería Telegram aprovecha su amplia base de usuarios y su infraestructura robusta, lo que permite a los desarrolladores llegar a una audiencia masiva con relativa facilidad. La simplicidad de la API y la documentación clara contribuyen a una curva de aprendizaje baja, lo que la hace accesible para una amplia gama de desarrolladores.

En resumen, la API de Telegram Bot es una herramienta fundamental para la creación de soluciones automatizadas y conversacionales dentro del ecosistema de Telegram. Su diseño flexible y sus amplias capacidades la convierten en una opción atractiva para desarrolladores que buscan construir aplicaciones interactivas y eficientes que mejoren la experiencia del usuario y optimicen los procesos de comunicación.

#### **5.2.4. Scrum**

Según Schwaber, K., & Sutherland, J. (2020), Scrum es un marco de trabajo ágil ampliamente adoptado en el desarrollo de software y en otros dominios para gestionar proyectos complejos. Su esencia radica en la entrega de valor de forma incremental y colaborativa, fomentando la experimentación continua y los ciclos de retroalimentación para el aprendizaje y la mejora constante. Este enfoque empírico, donde las decisiones se basan en la observación, la experiencia y la experimentación, se sustenta en tres pilares fundamentales: transparencia, inspección y adaptación.

La transparencia asegura que los aspectos significativos del proceso sean visibles para aquellos que son responsables del resultado. Esto implica que el trabajo y el progreso deben ser claros y entendibles para todos los

miembros del equipo y las partes interesadas. La inspección se refiere a la necesidad de revisar frecuentemente los artefactos de Scrum y el progreso hacia un objetivo acordado para detectar variaciones indeseables. Finalmente, la adaptación es el ajuste del proceso o del producto tan pronto como sea posible para minimizar cualquier desviación de los límites aceptables o para optimizar los objetivos.

El marco de trabajo Scrum se define por un conjunto de roles, eventos y artefactos. El Equipo Scrum es la unidad fundamental de Scrum y está compuesto por un Product Owner, un Scrum Master y Desarrolladores. Cada uno de estos roles tiene responsabilidades específicas. El Product Owner maximiza el valor del producto resultante del trabajo de la metodología y es responsable de asegurar que Scrum sea entendido y ejecutado, y actúa como un líder de servicio para el Equipo Scrum y la organización. Los desarrolladores son las personas en el Equipo Scrum que se comprometen a crear cualquier aspecto de un incremento utilizable en cada Sprint.

Los eventos de Scrum son oportunidades formales para inspeccionar y adaptar. Estos incluyen el Sprint (un contenedor para todos los demás eventos, de duración fija de un mes o menos), la Planificación del Sprint, el Daily Scrum, la Revisión del Sprint y la Retrospectiva del Sprint. Cada evento tiene un propósito específico y un límite de tiempo. Los artefactos de Scrum representan el trabajo o el valor. Son el Product Backlog, el Sprint Backlog y el Incremento. Estos artefactos están diseñados para maximizar la transparencia de la información clave, asegurando que todos tengan la misma comprensión del trabajo que se está realizando.

Scrum fue introducido al mundo en 1995 por Ken Schwaber y Jeff Sutherland como una forma mejor de colaboración en equipo para resolver problemas complejos. Ellos son los co-creadores de Scrum y mantienen la Guía Scrum, que es la definición oficial y concisa de Scrum. La Guía Scrum es considerada la fuente autorizada de Scrum y describe las responsabilidades, eventos y artefactos de Scrum, así como la guía que los une.

**Figura 2**  
*Etapas Scrum*



*Nota.* Adaptado de Schwaber, K., & Sutherland, J. (2020),

### 5.2.5. Spring Framework

Khandal, R., Meshram, M., & Mahatme, S. (2017), Spring Framework es un marco de trabajo de código abierto y una plataforma integral para el desarrollo de aplicaciones empresariales en Java. Lanzado inicialmente en 2004, Spring ha evolucionado hasta convertirse en el ecosistema dominante para el desarrollo de aplicaciones basadas en Java, gracias a su diseño modular, su flexibilidad y su amplio conjunto de características. Su objetivo

principal es simplificar el desarrollo de aplicaciones empresariales, permitiendo a los desarrolladores centrarse en la lógica de negocio en lugar de en la infraestructura subyacente.

El corazón de Spring Framework reside en dos conceptos fundamentales: la Inversión de Control (IoC) y la Inyección de Dependencias (DI). La IoC es un principio de diseño de software en el que el control de los objetos o partes de un programa se transfiere a un contenedor o marco de trabajo. En Spring, esto se logra a través del contenedor IoC, que es responsable de instanciar, configurar y ensamblar los objetos. La DI es una forma específica de IoC donde las dependencias de un objeto se proporcionan externamente en lugar de ser creadas por el propio objeto. Esto promueve un acoplamiento flexible entre los componentes, lo que facilita la prueba, el mantenimiento y la escalabilidad de las aplicaciones.

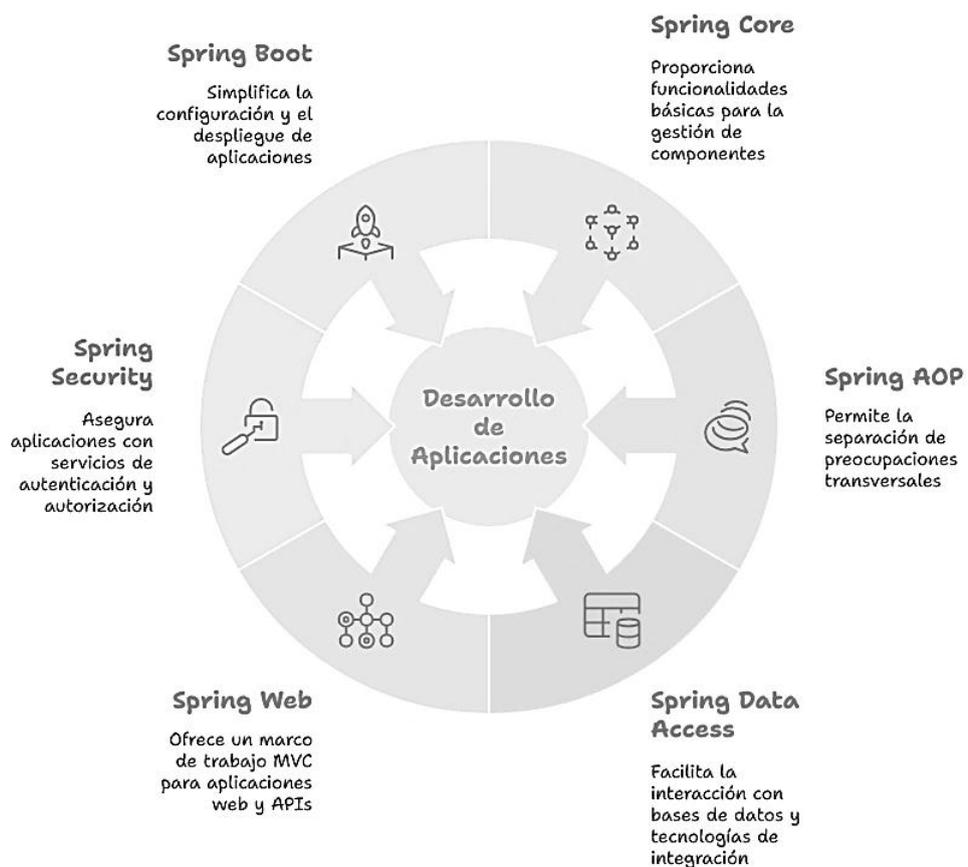
Spring Framework ofrece una amplia gama de módulos que cubren diversas áreas del desarrollo de aplicaciones:

- Spring Core: Proporciona las funcionalidades básicas del contenedor IoC y DI.
- Spring AOP (Aspect-Oriented Programming): Permite la separación de preocupaciones transversales, como la seguridad o el registro, de la lógica de negocio principal.
- Spring Data Access/Integration: Facilita la interacción con bases de datos (JDBC, ORM como Hibernate y JPA), JMS, JMX, y otras tecnologías de integración.

- Spring Web (Spring MVC): Un robusto marco de trabajo MVC (Modelo-Vista-Controlador) para el desarrollo de aplicaciones web y APIs RESTful.
- Spring Security: Proporciona servicios de autenticación y autorización para proteger las aplicaciones.
- Spring Boot: Una extensión de Spring que simplifica enormemente la configuración y el despliegue de aplicaciones Spring, permitiendo la creación de aplicaciones autónomas y listas para producción con un mínimo esfuerzo de configuración.

**Figura 3**

*Módulos de Spring Boot*



*Nota.* Adaptado de Khandal, R., Meshram, M., & Mahatme, S. (2017)

La modularidad de Spring es una de sus mayores fortalezas. Los desarrolladores pueden elegir qué módulos utilizar en función de las necesidades específicas de su proyecto, evitando la sobrecarga de dependencias innecesarias. Esto permite construir aplicaciones ligeras y eficientes. Además, Spring se integra sin problemas con otras tecnologías y frameworks populares en el ecosistema Java, lo que lo convierte en una opción versátil para una amplia variedad de escenarios de desarrollo.

El éxito de Spring Framework se debe a varios factores:

- **Productividad del desarrollador:** Simplifica tareas complejas y reduce la cantidad de código repetitivo, lo que acelera el desarrollo.
- **Flexibilidad y extensibilidad:** Permite a los desarrolladores adaptar el framework a sus necesidades específicas y extender sus funcionalidades.
- **Comunidad activa y soporte:** Cuenta con una gran comunidad de desarrolladores y una amplia documentación, lo que facilita la resolución de problemas y el aprendizaje.
- **Pruebas unitarias y de integración:** Su diseño basado en IoC/DI facilita la creación de pruebas unitarias y de integración, lo que contribuye a la calidad del software.
- **Adopción industrial:** Es ampliamente utilizado en la industria, lo que garantiza su relevancia y evolución continua.

#### **5.2.6. PostgreSQL**

Según Stonebraker y Rowe (1986), PostgreSQL es un sistema de gestión de bases de datos relacionales de objetos (ORDBMS) de código abierto, reconocido mundialmente por su avanzada funcionalidad, robustez,

escalabilidad y cumplimiento estricto de los estándares SQL. Su origen se remonta al proyecto POSTGRES en el Departamento de Ciencias de la Computación de la Universidad de California, Berkeley, liderado por Michael Stonebraker, con el objetivo de suceder al sistema de bases de datos relacionales.

El diseño de PostgreSQL se centró en superar las limitaciones de los sistemas de bases de datos relacionales tradicionales, incorporando características orientadas a objetos. Esto incluye la capacidad de definir tipos de datos complejos, herencia de tablas, funciones y operadores definidos por el usuario, y reglas de integridad de datos avanzadas. Estas características lo distinguen de los sistemas puramente relacionales y le otorgan la flexibilidad necesaria para manejar una amplia gama de aplicaciones, desde sistemas transaccionales hasta almacenes de datos y aplicaciones geoespaciales.

Entre los objetivos de diseño clave de POSTGRE se encontraban:

- Mejor soporte para objetos complejos: Permitir el almacenamiento y la manipulación de datos más allá de las estructuras relacionales simples.
- Extensibilidad para tipos de datos, operadores y métodos de acceso: Facilitar a los usuarios la definición de sus propios tipos de datos y la extensión de la funcionalidad del sistema.
- Facilidades para bases de datos activas (alertas y disparadores): Implementar mecanismos que permitan al sistema reaccionar automáticamente a ciertos eventos o condiciones en la base de datos.
- Simplificar el código del DBMS para la recuperación de fallos: Mejorar la fiabilidad y la capacidad de recuperación del sistema.

- Producir un diseño que pueda aprovechar los discos ópticos, las estaciones de trabajo con múltiples procesadores estrechamente acoplados y los chips VLSI personalizados: Adaptarse a las tecnologías de hardware emergentes de la época.
- Realizar la menor cantidad de cambios posible (preferiblemente ninguno) al modelo relacional: Mantener la compatibilidad con el modelo relacional mientras se introducen nuevas capacidades.

La arquitectura de PostgreSQL se caracteriza por su proceso basado en procesos, donde cada conexión de cliente se maneja mediante un proceso de servidor dedicado. Esto proporciona una gran estabilidad y aislamiento entre las sesiones de usuario. Además, PostgreSQL implementa un sofisticado sistema de control de concurrencia multiversión (MVCC), que permite que múltiples transacciones accedan y modifiquen los datos simultáneamente sin bloquearse entre sí, lo que mejora significativamente el rendimiento en entornos de alta concurrencia.

La comunidad de PostgreSQL es muy activa y contribuye continuamente al desarrollo y mejora del sistema, lo que ha resultado en un ciclo de lanzamiento constante de nuevas versiones con características innovadoras y mejoras de rendimiento. Su licencia de código abierto permite su uso, modificación y distribución libremente, lo que ha fomentado su adopción en una amplia variedad de industrias y aplicaciones, desde startups hasta grandes empresas.

En conclusión, PostgreSQL se ha consolidado como una de las bases de datos más avanzadas y confiables disponibles, ofreciendo una combinación única de características relacionales y orientadas a objetos, junto con una

sólida arquitectura y una vibrante comunidad de soporte. Su diseño extensible y su compromiso con los estándares lo convierten en una elección excelente para aplicaciones que demandan alta integridad de datos y flexibilidad.

## **VI. Organización y sistematización de las experiencias logradas**

### **6.1. Desarrollo de la Metodología SCRUM**

#### **6.1.1. Visión general de la situación actual del negocio**

Para modernizar el acceso a la información judicial y desarrollar un sistema de consulta de expedientes eficiente, se realizará un diagnóstico de los procesos vigentes en el módulo penal CSJ Santa. Este análisis constituye la base para la identificación de los requerimientos funcionales y no funcionales de una solución de software.

El módulo penal CSJ Santa opera actualmente con procesos de consulta de expedientes predominantemente manuales, físicos y burocráticos, basados en la presentación de formularios y la manipulación de archivos en papel.

Esta metodología genera:

- Tiempos de procesamiento prolongados y demoras significativas, tanto para la obtención de copias como para el avance general de los casos debido a la alta carga procesal.
- Escasa transparencia en el acceso a la información y desventaja informativa para los litigantes, quienes carecen de visibilidad en tiempo real sobre el estado de sus expedientes.
- Elevados costos directos e indirectos para los usuarios (tasas, transporte, tiempo), creando barreras de acceso económicas y geográficas.
- Limitada capacidad para el seguimiento, dependiendo de la presencia física y el seguimiento proactivo del interesado en horarios de oficina.

Para el levantamiento de información se documentó el flujo de actividades "as-is" (tal como es), desde la solicitud de información hasta la entrega de documentos. La información se obtuvo mediante:

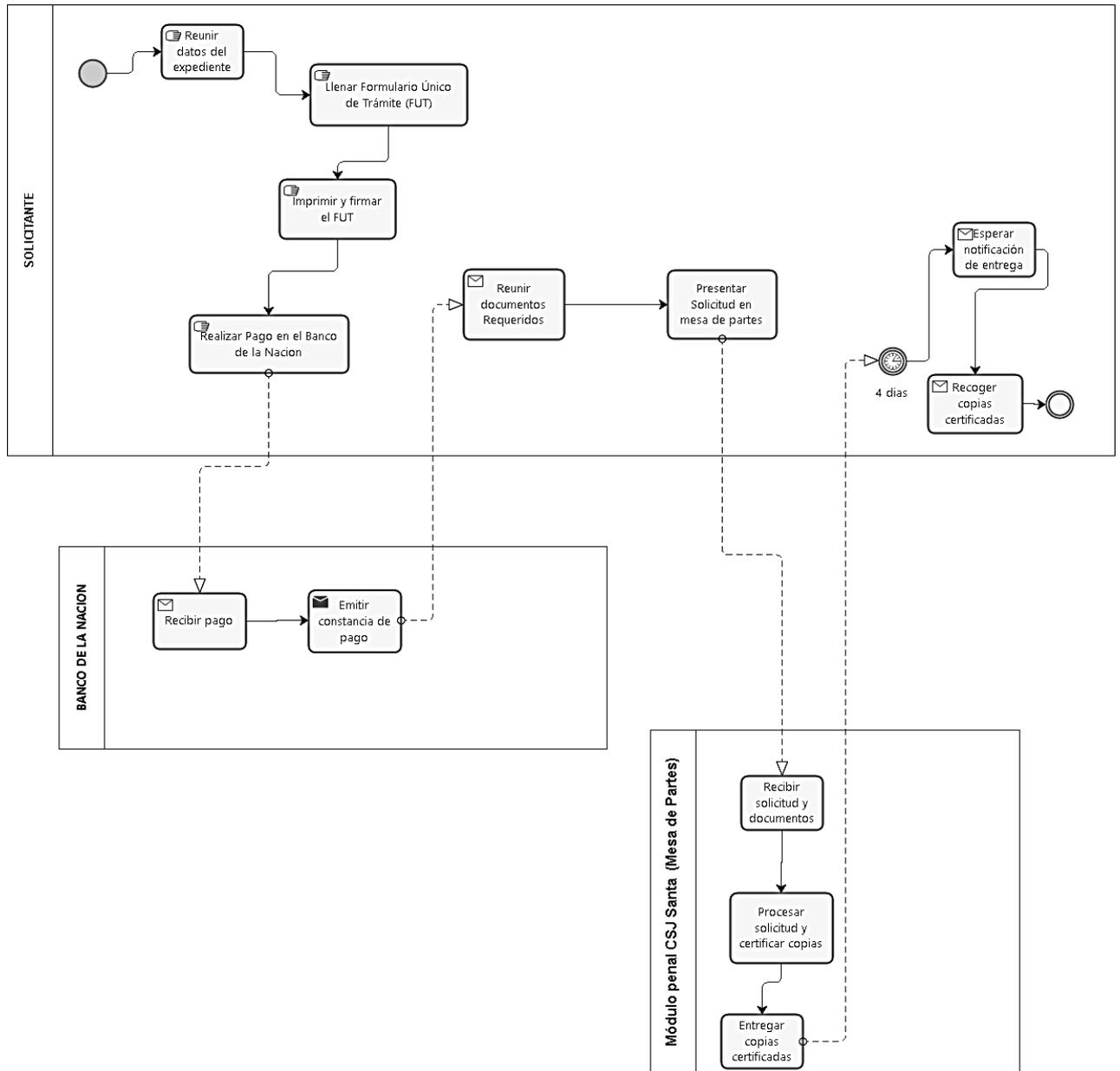
- Análisis de la documentación normativa existente, como el Texto Único de Procedimientos Administrativos (TUPA) y guías de trámite.
- Revisión de informes institucionales y académicos sobre la carga procesal y la eficiencia del sistema de justicia.
- Mapeo de los procesos de negocio para visualizar los cuellos de botella y las ineficiencias del flujo manual.

El diagnóstico identificó tres áreas críticas de mejora:

- Flujo de solicitud y entrega de documentos: El proceso, que incluye el llenado de formularios (FUT), pagos en entidades externas y la espera para el recojo físico, es un punto central de ineficiencia y demora.
- Accesibilidad a la información del expediente: La dependencia de archivos físicos y la consulta presencial limitan el acceso a la información a horarios y ubicaciones específicas, impidiendo la gestión remota y en tiempo real.
- Automatización de la consulta y verificación de estado: La falta de un canal digital y automatizado para consultas simples obliga a los usuarios a invertir tiempo y recursos en tareas repetitivas que podrían ser gestionadas instantáneamente.

**Figura 4**

Diagrama del proceso de negocio de la gestión de solicitud de certificados



*Nota.* Elaboración propia.

## 6.1.2. Iniciación del proyecto de software

### a) Definición de Roles de Scrum y Stakeholders

**Tabla 1:***Roles Scrum del proyecto*

<b>Roles de Scrum</b>	<b>Nombres y Apellidos</b>
Product Owner	Moreno Saavedra, Marco Josué
Scrum Master	Moreno Saavedra, Marco Josué
Development Team	Moreno Saavedra, Marco Josué

*Nota. Elaboración Propia***b) Requisitos**

En el marco del presente proyecto, se desarrolló un análisis sistemático para identificar los requisitos esenciales que debe satisfacer el sistema. La metodología incluyó entrevistas con usuarios clave y análisis de los procesos existentes en el módulo penal. Posteriormente, se agrupó estos requerimientos y se transformó en historias de usuario (HU), buscando describir las funcionalidades desde la perspectiva operativa de los usuarios finales. Esta aproximación garantiza mayor claridad en la especificación de requisitos y facilita su posterior implementación técnica.

A continuación, se detalla los requisitos funcionales y no funcionales que se identificó durante el análisis:

**Tabla 2:***Requisitos funcionales del sistema.*

<b>Nro.</b>	<b>Requisito Funcional</b>
R01	El sistema debe permitir crear, modificar y desactivar usuarios.
R02	El sistema debe permitir asignar diferentes roles a los usuarios (administrador, fiscal, abogado libre, etc.).
R03	El sistema debe registrar información de contacto de los usuarios, incluyendo número telefónico.
R04	El sistema debe solicitar al usuario compartir su contacto al iniciar sesión.
R05	El sistema debe verificar si el usuario está registrado en la base de datos.

---

R06	El sistema debe redirigir al usuario al registro si no está dado de alta.
R07	El sistema debe detectar usuarios no registrados.
R08	El sistema debe proporcionar enlaces a los formatos de registro según el tipo de usuario.
R09	El sistema debe indicar los canales para enviar los formatos de registro completados.
R10	El sistema debe aceptar el formato correcto de número de expediente para consultas.
R11	El sistema debe permitir filtrar expedientes por especialidad (penal, familia, civil, laboral, etc.).
R12	El sistema debe mostrar resultados únicos o listas para selección al consultar expedientes.
R13	El sistema debe validar el formato del DNI al consultar expedientes por DNI del imputado.
R14	El sistema debe verificar el apersonamiento para abogados libres al consultar por DNI.
R15	El sistema debe permitir filtrar expedientes por especialidad al consultar por DNI.
R16	El sistema debe mostrar las actas de audiencia disponibles de un expediente.
R17	El sistema debe permitir seleccionar actas por fecha específica o la más reciente.
R18	El sistema debe facilitar la descarga de las actas de audiencia.
R19	El sistema debe mostrar las resoluciones disponibles de un expediente.
R20	El sistema debe permitir seleccionar resoluciones por fecha específica o la más reciente.
R21	El sistema debe facilitar la descarga de las resoluciones.
R22	El sistema debe mostrar los documentos digitalizados disponibles de un expediente.
R23	El sistema debe permitir seleccionar documentos digitalizados por fecha específica o el más reciente.
R24	El sistema debe facilitar la descarga de los documentos digitalizados.
R25	El sistema debe verificar si existen depósitos judiciales en el expediente.
R26	El sistema debe generar automáticamente un reporte con la información de depósitos judiciales.

---

---

R27	El sistema debe permitir descargar el reporte de depósitos judiciales.
R28	El sistema debe verificar las personas que han firmado en un expediente.
R29	El sistema debe mostrar una lista para selección si hay múltiples firmantes.
R30	El sistema debe permitir descargar el registro de firmas.
R31	El sistema debe ofrecer opciones de envío de documentos (chat o correo).
R32	El sistema debe enviar correctamente los documentos por el canal seleccionado.
R33	El sistema debe confirmar el envío exitoso de los documentos.
R34	El sistema debe solicitar el DNI del imputado para consultar información sobre principio de oportunidad.
R35	El sistema debe generar un reporte con la información disponible sobre principio de oportunidad.
R36	El sistema debe permitir descargar el documento generado sobre principio de oportunidad.
R37	El sistema debe ofrecer opciones para continuar con el mismo expediente o cambiar a otro expediente.
R38	El sistema debe mantener el contexto de la sesión del usuario al navegar entre expedientes.
R39	El sistema debe registrar las consultas realizadas por los usuarios.
R40	El sistema debe contabilizar los documentos descargados por tipo.
R41	El sistema debe generar reportes por tipo de usuario y especialidad.
R42	El sistema debe permitir redactar mensajes para todos los usuarios.
R43	El sistema debe enviar notificaciones a través del chat.
R44	El sistema debe confirmar el envío exitoso de los mensajes masivos.

---

*Nota.* Elaboración Propia

**Tabla 3:**

*Requisitos no funcionales del sistema.*

---

<b>Nro</b>	<b>Requisito No Funcional</b>
RNF01	El sistema debe estar disponible al menos el 99% del tiempo durante el horario laboral.
RNF02	El sistema debe responder a las consultas de expedientes en menos de 3 segundos bajo carga normal.

---

---

RNF03	El sistema debe proteger la información de los usuarios mediante autenticación y cifrado de datos.
RNF04	El sistema debe registrar logs de acceso y operaciones relevantes para auditoría y trazabilidad.
RNF05	El sistema debe ser compatible con los navegadores web modernos y dispositivos móviles.
RNF06	El sistema debe permitir la recuperación ante fallos y respaldar la información diariamente.
RNF07	El sistema debe ser escalable para soportar un aumento del 100% en la cantidad de usuarios simultáneos.
RNF08	El sistema debe contar con una interfaz accesible y usable para usuarios con diferentes perfiles.
RNF09	El sistema debe cumplir con la normativa vigente de protección de datos personales.
RNF10	El sistema debe permitir la integración con otros sistemas judiciales mediante APIs seguras.
RNF11	El sistema debe mostrar mensajes de error claros y amigables para el usuario final.
RNF12	El sistema debe permitir la actualización y mantenimiento sin afectar la disponibilidad de los servicios.

---

*Nota.* Elaboración Propia

### c) Creación del Backlog del Producto

Luego de realizar un proceso detallado de identificación y análisis de requisitos, se procedió a agruparlos para dar forma a las Historias de Usuario (HU). Esta transformación permitió representar de manera sencilla y comprensible lo que cada usuario espera del sistema, enfocándose siempre en generar valor real.

A continuación, todas estas historias se organizaron dentro del Backlog del Producto, que no solo actúa como un repositorio central, sino que también ofrece una visión clara y priorizada de lo que debe construirse. Gracias a ello, es posible planificar el trabajo en iteraciones (sprints) de forma ordenada, asegurando que el marco siempre esté enfocado en lo más importante.

**Tabla 4:**

*Backlog del producto e historias de usuario*

<b>ID</b>	<b>Rol</b>	<b>Deseo/ Funcionalidad</b>	<b>Beneficio/Para qué</b>
HU01	Como administrador	Quiero gestionar los usuarios y sus roles en el sistema	Para controlar quiénes pueden acceder al sistema y qué acciones pueden realizar
HU02	Como usuario	Quiero autenticarme en el sistema compartiendo mi contacto	Para acceder a las funcionalidades según mi rol y permisos asignados
HU03	Como usuario no registrado	Quiero recibir instrucciones sobre cómo registrarme	Para poder completar mi registro y acceder al sistema
HU04	Como usuario	Quiero consultar expedientes por número	Para acceder a la información específica de un caso
HU05	Como usuario	Quiero consultar expedientes por DNI del imputado	Para encontrar casos relacionados con una persona específica
HU06	Como usuario	Quiero acceder a actas de audiencia de un expediente	Para revisar el contenido y resoluciones de las audiencias
HU07	Como usuario	Quiero acceder a resoluciones de un expediente	Para conocer las decisiones judiciales sobre el caso

HU08	Como usuario	Quiero acceder a documentos digitalizados de un expediente	Para revisar escritos y otros documentos del caso
HU09	Como usuario	Quiero consultar información sobre depósitos judiciales	Para verificar pagos y consignaciones relacionadas con el caso
HU10	Como fiscal	Quiero consultar el registro de firmas en un expediente	Para verificar quiénes han firmado documentos en el caso
HU11	Como usuario	Quiero seleccionar el modo de recepción de documentos	Para recibir los archivos de la manera más conveniente
HU12	Como fiscal	Quiero consultar información sobre principio de oportunidad	Para verificar si un imputado tiene acuerdos previos
HU13	Como usuario	Quiero navegar entre diferentes expedientes	Para consultar información de múltiples casos sin reiniciar el sistema
HU14	Como administrador	Quiero acceder a estadísticas de uso del sistema	Para analizar patrones de consulta y mejorar el servicio
HU15	Como administrador	Quiero enviar mensajes masivos a los usuarios	Para comunicar novedades o cambios en el sistema

*Nota.* Elaboración Propia

### 6.1.3. Planificación y estimación de la Iteración (Sprint)

En esta etapa, se llevó a cabo una revisión minuciosa del Backlog del Producto con el objetivo de priorizar las historias de usuario (HU) y las tareas más relevantes para el desarrollo. Esta revisión permitió enfocar el trabajo en aquellas funcionalidades que aportan mayor valor al usuario y al sistema en general. Para realizar el proceso de priorización, se aplicó una fórmula sencilla, ideal para tomar decisiones rápidas y objetivas dentro del marco ágil. Esta técnica se basa en la relación entre dos factores clave: el valor de negocio y la complejidad técnica, y permite estimar qué historias de usuario deberían implementarse primero.

Como punto de partida, el valor de negocio se representa mediante una escala del 1 al 5 y es asignado directamente por el *Product Owner*, tomando como base su experiencia y conocimiento del contexto. A mayor puntuación, mayor es la contribución que dicha historia de usuario aporta al objetivo general del sistema.

En paralelo, los puntos de historia permiten estimar el esfuerzo requerido para completar cada funcionalidad. Esta valoración es realizada por el equipo técnico y considera tanto la complejidad como el tiempo necesario para su implementación. No se trata únicamente de contar tareas, sino de evaluar el grado de dificultad técnica y los posibles obstáculos involucrados.

Una vez definidos ambos factores, se procede a calcular el valor de prioridad, dividiendo el valor de negocio entre los puntos de historia. El resultado ofrece una medida del retorno de valor por unidad de esfuerzo. Bajo este criterio, una historia que combine un alto impacto en el negocio con una baja complejidad tendrá un valor de prioridad elevado, lo que sugiere que debe ser considerada con mayor urgencia dentro del ciclo de desarrollo

Posteriormente, las tareas fueron asignadas para su desarrollo definiendo criterios de aceptación claros y específicos para cada una permitiendo verificar de manera objetiva cuándo una funcionalidad se considera realmente terminada y lista para su validación. Estos criterios también sirven como guía para asegurar la calidad del producto entregado en cada iteración.

**Tabla 5:***Historias de usuario priorizadas*

ID	Deseo/ Funcionalidad	Criterio de Aceptación	Valor de Negocio	Puntos de Historia	Valor de Prioridad	Prioridad
HU01	Quiero gestionar los usuarios y sus roles en el sistema	✓ El sistema permite crear, modificar y desactivar usuarios	5	2	2.5	Alta
		✓ Se pueden asignar diferentes roles (administrador, fiscal, abogado libre, etc.)				
		✓ Se registra información de contacto incluyendo número telefónico				
HU02	Quiero autenticarme en el sistema compartiendo mi contacto	✓ El sistema solicita compartir contacto al iniciar	4	2	2	Alta
		✓ Verifica si el usuario está registrado en la base de datos				
		✓ Redirige a registro si el usuario no está dado de alta				
HU03	Quiero recibir instrucciones sobre cómo registrarme	✓ El sistema detecta usuarios no registrados	5	3	1.67	Media
		✓ Proporciona enlaces a los formatos de registro según el tipo de usuario				
		✓ Indica los canales para enviar los formatos completados				
HU04	Quiero consultar expedientes por número	✓ El sistema acepta el formato correcto de número de expediente	5	3	1.67	Alta
		✓ Permite filtrar por especialidad (penal, familia, civil, laboral, etc.)				
		✓ Muestra resultados únicos o listas para selección				

HU05	Quiero consultar expedientes por DNI del imputado	✓	El sistema valida el formato del DNI	5	3	1.67	Alta
		✓	Verifica el apersonamiento para abogados libres				
HU06	Quiero acceder a actas de audiencia de un expediente	✓	El sistema muestra las actas disponibles	3	2	1.5	Alta
		✓	Permite seleccionar por fecha específica o la más reciente				
HU07	Quiero acceder a resoluciones de un expediente	✓	El sistema muestra las resoluciones disponibles	4	3	1.33	Alta
		✓	Permite seleccionar por fecha específica o la más reciente				
HU08	Quiero acceder a documentos digitalizados de un expediente	✓	El sistema muestra los documentos disponibles	4	3	1.33	Alta
		✓	Permite seleccionar por fecha específica o el más reciente				
HU09	Quiero consultar información sobre depósitos judiciales	✓	El sistema verifica si existen depósitos en el expediente	4	3	1.33	Media
		✓	Genera automáticamente un reporte con la información				
HU10	Quiero consultar el registro de firmas en un expediente	✓	El sistema verifica las personas que han firmado	5	4	1.25	Media
		✓	Muestra lista para selección si hay múltiples firmantes				
HU11	Quiero seleccionar el modo de recepción de documentos	✓	El sistema ofrece opciones de envío (chat o correo)	5	4	1.25	Media
		✓	Envía correctamente los documentos por el canal seleccionado				
		✓	Confirma el envío exitoso				

HU12	Quiero consultar información sobre principio de oportunidad	✓ El sistema solicita el DNI del imputado	3	3	1	Media
		✓ Genera un reporte con la información disponible				
		✓ Permite descargar el documento				
HU13	Quiero navegar entre diferentes expedientes	✓ El sistema ofrece opciones para continuar con el mismo expediente	3	3	1	Baja
		✓ Permite cambiar a otro expediente				
		✓ Mantiene el contexto de la sesión del usuario				
HU14	Quiero acceder a estadísticas de uso del sistema	✓ El sistema registra las consultas realizadas	2	4	0.5	Baja
		✓ Contabiliza documentos descargados por tipo				
		✓ Genera reportes por tipo de usuario y especialidad				
HU15	Quiero enviar mensajes masivos a los usuarios	✓ El sistema permite redactar mensajes para todos los usuarios	2	4	0.5	Baja
		✓ Envía notificaciones a través del chat				
		✓ Confirma el envío exitoso				

*Nota. Elaboración Propia*

#### 6.1.4. Identificación de tareas por cada iteración (Sprint)

Durante la planificación de cada Sprint, los requisitos del sistema fueron descompuestos en tareas claras y específicas, facilitando su asignación y estimación de tiempo. Este proceso permitió organizar de manera efectiva el trabajo del equipo, asegurando un seguimiento preciso y equilibrando las cargas laborales. La implementación se inició con los diagramas de arquitectura por historia de usuario y continuó de forma secuencial con la base de datos, el backend, el frontend y los reportes PDF. La priorización de tareas se realizó considerando el valor de negocio frente a la complejidad técnica, lo cual permitió entregar funcionalidad valiosa desde las primeras iteraciones.

**Tabla 6:**

*Tareas y descripción de cada historia de Usuario*

N°	Historia De Usuario	Tarea	Descripción	T.E días
HU01	Gestionar usuarios y roles del sistema	T01	Diseño de Capas-MVC: Crear diagrama de clases para Contact, Type y Estado con atributos y métodos según arquitectura MVC.	2
		T02	Base de datos: Implementar tablas para usuarios (Contact) y roles (Type) con relaciones, índices y restricciones necesarias.	1
		T03	Backend: Desarrollar IContactService e implementación para operaciones CRUD de usuarios y gestión de roles.	2
		T04	Frontend: Implementar menús en el bot para gestión de usuarios y asignación de roles.	2
HU02	Autenticarse compartiendo contacto	T05	Diseño de Capas-MVC: Crear diagrama de clases para autenticación y validación de contactos.	1

		T06	Base de datos: Implementar campos para almacenar información de contacto y estado de autenticación.	1
		T07	Backend: Desarrollar lógica de verificación de usuarios registrados en la base de datos.	2
		T08	Frontend: Implementar flujo de solicitud de contacto y validación en el bot.	1
		T09	Diseño de Capas-MVC: Crear diagrama de clases para el proceso de registro.	2
HU03	Recibir instrucciones de registro	T10	Backend: Desarrollar lógica para detectar usuarios no registrados y proporcionar enlaces.	2
		T11	Frontend: Implementar mensajes informativos y enlaces a formatos de registro.	2
		T12	Diseño de Capas-MVC: Crear diagrama de clases para Expediente y Expediente_Aux.	2
	Consultar expedientes por número	T13	Base de datos: Implementar tabla expediente_aux para almacenar resultados de búsqueda.	1
HU04		T14	Backend: Desarrollar IExpedienteAuxService e implementación para consultas por número.	2
		T15	Frontend: Implementar menú de filtro por especialidad y validación de formato.	1
	Consultar expedientes por DNI del imputado	T16	Diseño de Capas-MVC: Ampliar diagrama para incluir relación Expediente-Persona.	1
HU05		T17	Backend: Desarrollar lógica para validar DNI y verificar apersonamiento.	2
		T18	Frontend: Implementar interfaz para ingreso de DNI y filtros por especialidad.	1
	Acceder a actas de audiencia	T19	Diseño de Capas-MVC: Crear diagrama de clases para Acta y Acta_Aux.	1
HU06		T20	Base de datos: Implementar tabla acta_aux para almacenar resultados de búsqueda.	1

		T21	Backend: Desarrollar IActaAuxService, ActaImpl y lógica de descarga FTP.	2
		T22	Frontend: Implementar menú de selección por fecha y descarga de actas.	1
		T23	Diseño de Capas-MVC: Crear diagrama de clases para Resolucion y Resolucion_Aux.	2
	Acceder a resoluciones de expedientes	T24	Base de datos: Implementar tabla resolucion_aux para almacenar resultados.	2
HU07		T25	Backend: Desarrollar IResolucionAuxService e implementación para consultas.	2
		T26	Frontend: Implementar menú de selección por fecha y descarga de resoluciones.	1
		T27	Diseño de Capas-MVC: Crear diagrama de clases para Documento_Digitalizado y DDigitalizado_Aux.	2
	Acceder a documentos digitalizados	T28	Base de datos: Implementar tabla para documentos digitalizados.	2
HU08		T29	Backend: Desarrollar IDDigitalizadoService e implementación para consultas.	2
		T30	Frontend: Implementar menú de selección por fecha y descarga de documentos.	1
		T31	Diseño de Capas-MVC: Crear diagrama de clases para Deposito y Deposito_Aux.	2
	Consultar información de depósitos judiciales	T32	Base de datos: Implementar tabla deposito_aux para almacenar resultados.	1
HU09		T33	Backend: Desarrollar IDepositoAuxService e implementación para consultas.	2
		T34	Frontend: Implementar generación automática de reportes de depósitos.	1
	Consultar registro de firmas	T35	Diseño de Capas-MVC: Crear diagrama de clases para Firma y su relación con Persona.	2
HU10		T36	Backend: Desarrollar lógica para verificar firmantes en expedientes.	2

		T37	Frontend: Implementar lista de selección para múltiples firmantes.	1
		T38	Diseño de Capas-MVC: Crear diagrama de flujo para opciones de envío.	2
HU11	Seleccionar modo de recepción de documentos	T39	Backend: Desarrollar lógica para envío por diferentes canales (chat/correo).	2
		T40	Frontend: Implementar menú opcionesEnvioMenu con confirmación de envío.	1
		T41	Crear diagrama de clases y flujo para la consulta de principios de oportunidad.	1
		T42	Agregar/ajustar tabla o campo en la base de datos para almacenar información sobre principios de oportunidad si aplica.	1
HU12	Consultar información sobre principio de oportunidad	T43	Desarrollar servicio IPrincipioOportunidadService e implementación para consulta; exponer endpoint o método para obtener la información	2
		T44	Implementar interfaz para mostrar la información sobre principio de oportunidad al usuario.	2
		T45	Implementar interfaz para mostrar la información sobre principio de oportunidad al usuario.	1
HU13	Navegar entre diferentes expedientes	T46	Verificar que la estructura de expedientes permita la navegación (relaciones, índices, etc.).	1
		T47	Desarrollar lógica en el backend para listar, seleccionar y navegar entre expedientes; exponer métodos en el servicio correspondiente.	2

		Implementar controles de navegación (botones, listas, etc.) para que el usuario pueda moverse entre expedientes de forma intuitiva.	2
		Diseño de Capas-MVC: Crear diagrama de clases para Estadística.	1
HU14	Acceder a estadísticas de uso	Base de datos: Implementar tabla estadística con campos para diferentes métricas.	1
		Backend: Desarrollar IEstadisticaService e implementación para registro y consulta.	2
		Frontend: Implementar generación de reportes por tipo de usuario y especialidad.	1
		Diseño de Capas-MVC: Crear diagrama de flujo para envío de mensajes masivos.	1
HU15	Enviar mensajes masivos	Backend: Desarrollar lógica para obtener lista de chats (getChatsforPersonalMessage).	1
		Frontend: Implementar interfaz para redacción y envío de mensajes masivos.	1

*Nota.* Elaboración Propia

### 6.1.5. Planificación de las iteraciones (Sprint)

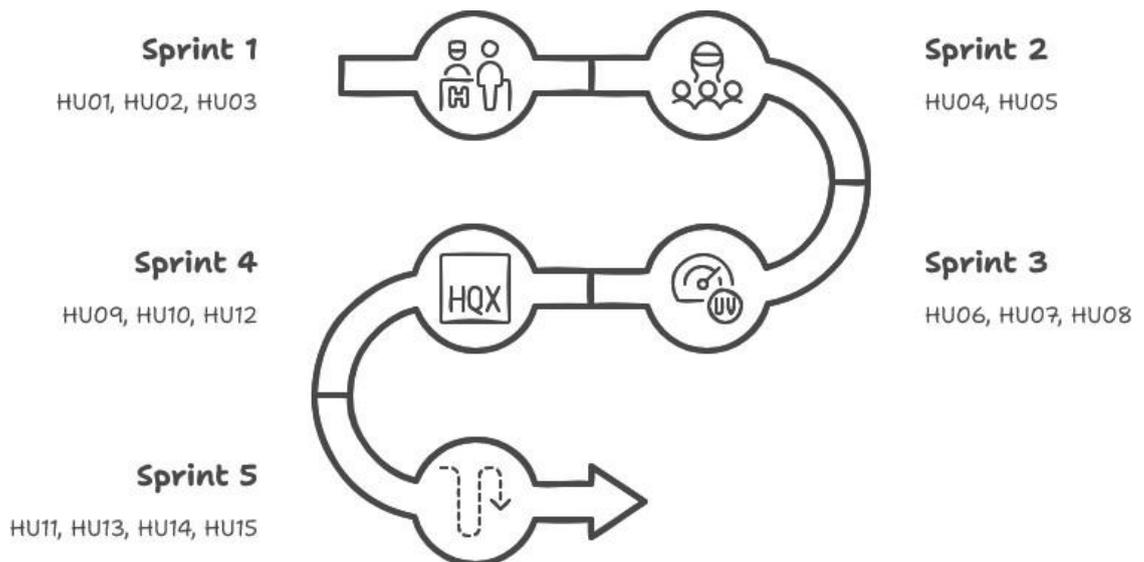
La Planificación de los Sprint es realizado a través de reuniones colaborativas donde el equipo seleccionó y organizó las tareas prioritarias del Product Backlog que se comprometieron a completar durante el siguiente ciclo de trabajo, estableciendo objetivos claros y definiendo cómo alcanzarlos en los tiempos establecidos.

**Tabla 7:**  
*Planificación de las iteraciones (sprints)*

SPRINT	HISTORIAS DE USUARIO
1	HU01 ⇔ HU02 ⇔ HU03
2	HU04 ⇔ HU05
3	HU06 ⇔ HU07 ⇔ HU08
4	HU09 ⇔ HU10 ⇔ HU12
5	HU11 ⇔ HU13 ⇔ HU14 ⇔ HU15

*Nota.* Elaboración propia

**Figura 5**  
*Secuencia de ejecución de los Sprints*



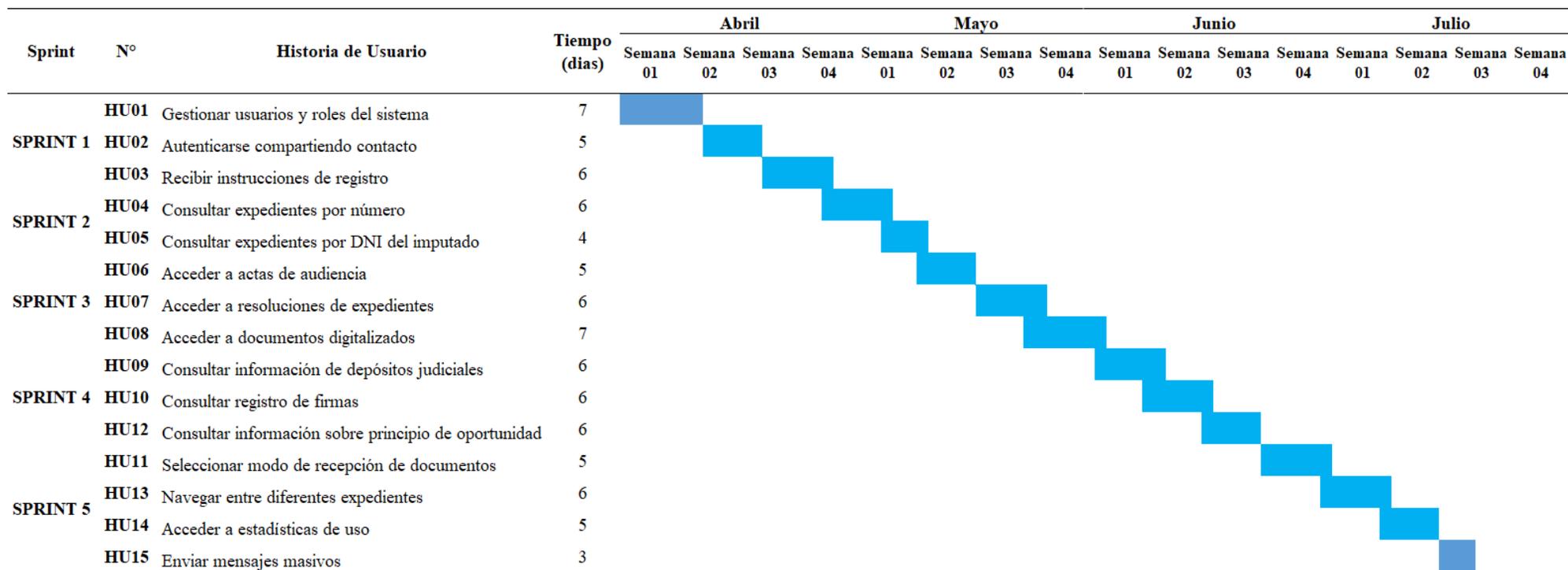
*Nota.* Elaboración propia

#### **6.1.6. Cronograma de las Iteraciones (Sprints)**

Se estableció un plan temporal detallado para las iteraciones (Sprints) mediante un calendario organizado que especifica el orden cronológico y extensión de cada fase de desarrollo, determinando los momentos de comienzo y conclusión para cada Sprint. Este enfoque facilitó la estructuración y representación gráfica de la distribución temporal del proyecto, segmentándolo en períodos uniformes y administrables. Seguidamente, se presenta el detalle calendarizado para cada uno de los sprints que conforman el proyecto.

**Tabla 8:**

*Cronograma para los Sprints(Año 2019)*



*Nota.* Elaboración propia

### **6.1.7. Ejecución de la Iteración (Sprint).**

Durante esta etapa se puso en marcha la construcción práctica del sistema. Esto incluyó actividades como el diseño general, la organización de sus partes, la escritura del código, la realización de pruebas, la elaboración de documentación, la revisión del trabajo hecho y la integración de todo en un solo conjunto funcional. Todo el proceso fue organizado y monitoreado de forma visual y dinámica a través de tableros de trabajo colaborativo, lo que permitió tener un control claro y actualizado del avance del proyecto.

La estructura del sistema se definió a partir de los elementos principales identificados en las fases anteriores. A partir de esta base, se fue desarrollando cada parte del sistema paso a paso, asegurando que todo se integrara de manera lógica y coherente. Además, se trabajó con la metodología ágil SCRUM que permitió dividir el trabajo en ciclos cortos, entregar avances de forma regular y adaptarse a los cambios que fueron surgiendo en los requisitos. Gracias al uso de los tableros Scrum, fue posible visualizar en qué etapa se encontraba cada tarea, detectar posibles retrasos y ajustar prioridades según la importancia y el esfuerzo que requería cada una.

En resumen, esta forma de trabajo permitió que el sistema se desarrollara de manera ordenada y eficiente, cumpliendo tanto con los objetivos planteados como con las expectativas del equipo y de quienes usarán el sistema.

**Tabla 9:***Especificación de la HU01: Gestionar usuarios y roles del sistema*

<b>N° Historia</b>	<b>Usuario</b>
HU01	Administrador
<b>Sprint</b>	<b>Prioridad</b>
01	Alta
<b>Nombre</b>	<b>T. E</b>
Gestionar usuarios y roles del sistema	7
<b>Responsable</b>	Moreno Saavedra, Marco

**Criterios de Aceptación:**

- El sistema permite crear, modificar y desactivar usuarios
- Se pueden asignar diferentes roles (administrador, fiscal, abogados, etc.)
- Se registra información de contacto incluyendo número telefónico

**Tareas:**

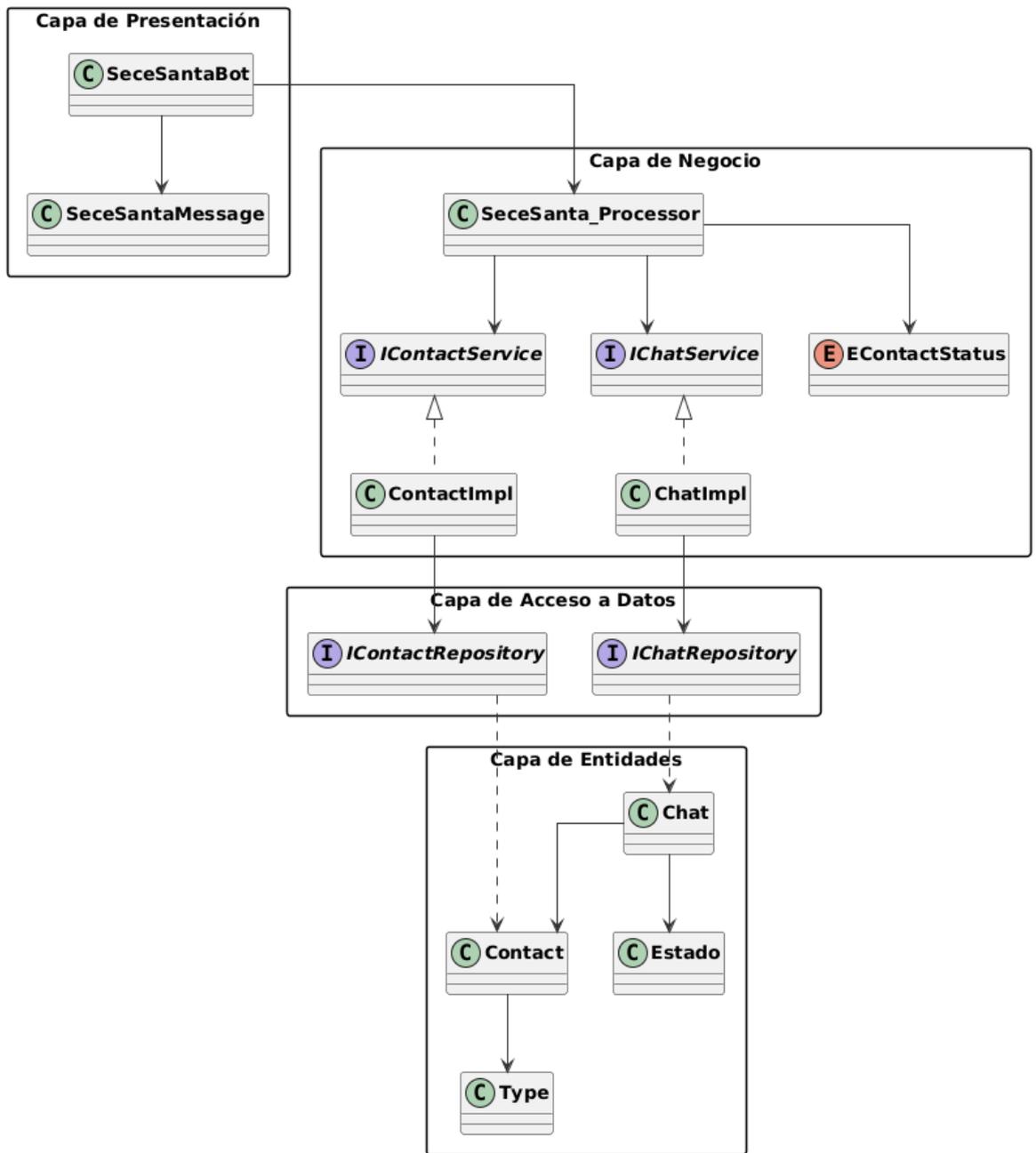
- Diseño Capas-MVC: Crear diagrama de clases para Contact, Type y Estado con atributos y métodos según arquitectura MVC.
- Base de datos: Implementar tablas para usuarios (Contact) y roles (Type) con relaciones, índices y restricciones necesarias.
- Backend: Desarrollar IContactService e implementación para operaciones CRUD de usuarios y gestión de roles.
- Frontend: Implementar menús en el bot para gestión de usuarios y asignación de roles.

---

*Nota.* Elaboración propia

**Figura 6**

*Diagrama de Capas-MVC para la HU01*



*Nota.* Elaboración propia

## Figura 7

UI para la HU01: Gestionar usuarios y roles del sistema

SECE\_Santa      Dashboard   Personal   Reportes   Configuración      ?   

### Gestión de Usuarios

[+ Agregar Usuario](#)

Buscar por nombre o correo

NOMBRE	CORREO	ROL	ESTADO	ACCIONES
Juan Pérez	juan.perez@example.com	Administrador	Activo	Editar Desactivar
María García	maria.garcia@example.com	Fiscal Provincial	Activo	Editar Desactivar
Carlos Rodríguez	carlos.rodriguez@example.com	Defensor Público	Activo	Editar Desactivar
Ana López	ana.lopez@example.com	Procuraduría	Desactivado	Editar Activar
Luis Martínez	luis.martinez@example.com	Abogados	Activo	Editar Desactivar

*Nota.* Elaboración propia

**Tabla 10:***Especificación de la HU02: Autenticar usuario en el Sistema*

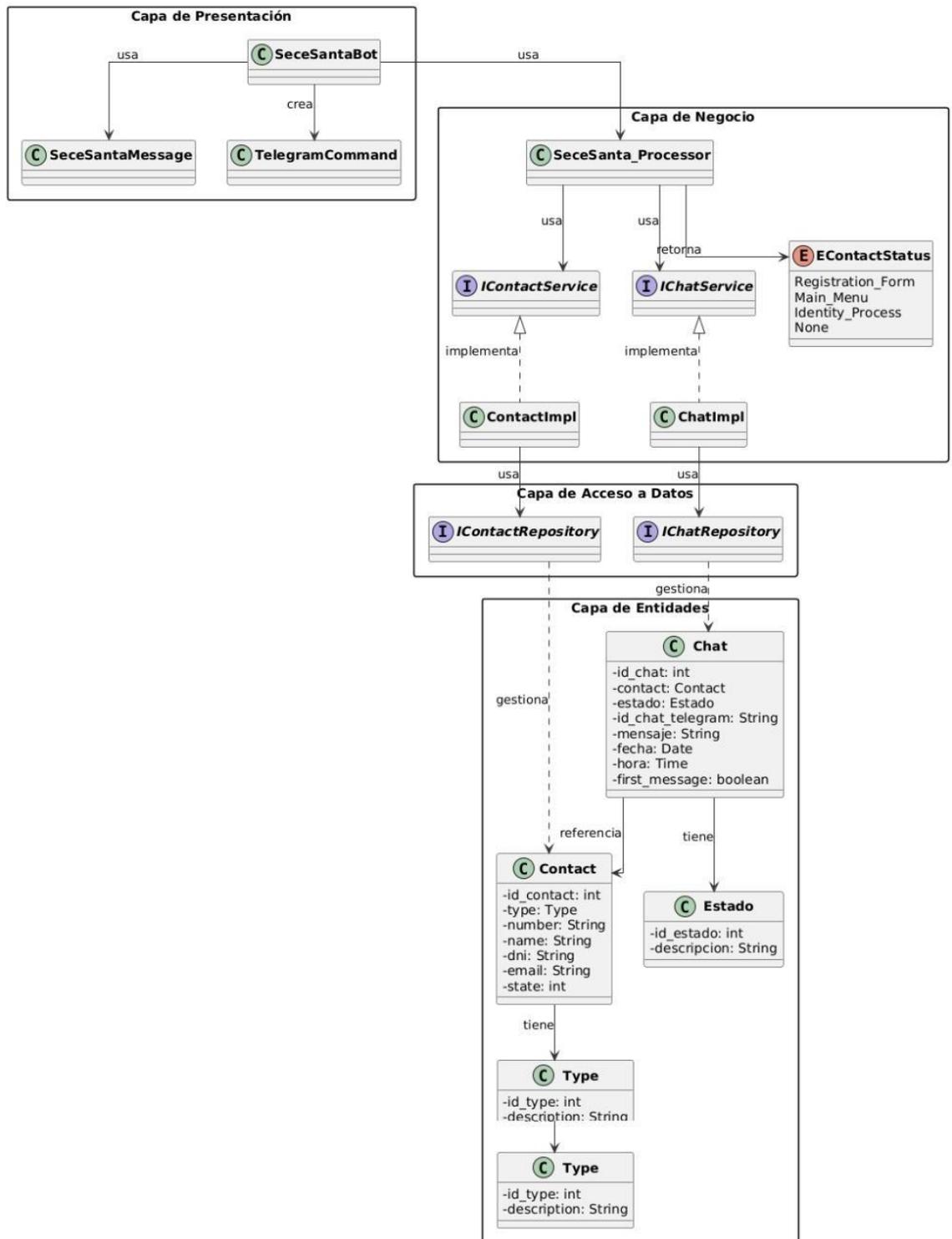
<b>N° Historia</b>	<b>Usuario</b>
HU02	Usuario
<b>Sprint</b>	<b>Prioridad</b>
01	Alta
<b>Nombre</b>	<b>T.E</b>
Autenticarse en el sistema compartiendo mi contacto	5
<b>Responsable</b>	Moreno Saavedra, Marco
<b>Criterios de Aceptación:</b>	
El sistema solicita compartir contacto al iniciar	
<ul style="list-style-type: none"><li>• Verifica si el usuario está registrado en la base de datos</li><li>• Redirige a registro si el usuario no está dado de alta</li></ul>	
<b>Tareas:</b>	
<ul style="list-style-type: none"><li>• Diseño Capas-MVC: Crear diagrama de clases para autenticación y validación de contactos.</li><li>• Base de datos: Implementar campos para almacenar información de contacto y estado de autenticación.</li><li>• Backend: Desarrollar lógica de verificación de usuarios registrados en la base de datos.</li><li>• Frontend: Implementar flujo de solicitud de contacto y validación en el bot.</li></ul>	

---

*Nota.* Elaboración propia

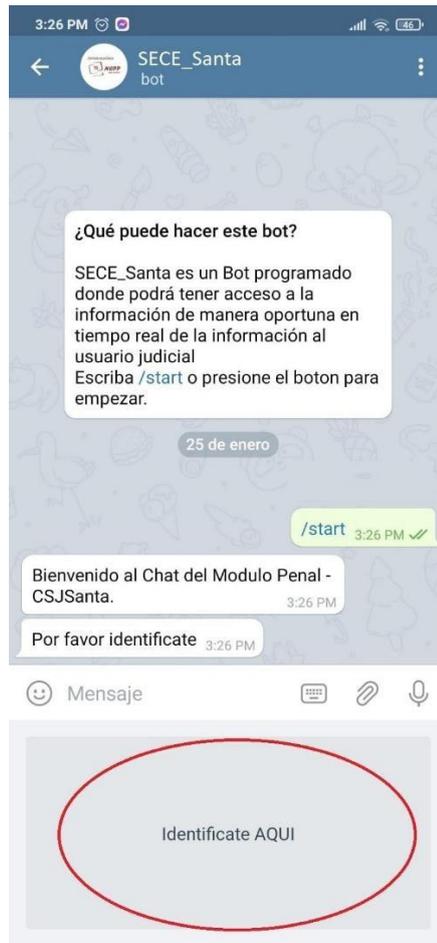
**Figura 8**

*Diagrama de Capas-MVC para la HU02*



*Nota.* Elaboración propia

**Figura 9**  
*UI para la HU02: Autenticar Usuarios*



*Nota.* Elaboración propia

**Tabla 11:**

*Especificación de la HU03: Recibir instrucciones de registro*

Nº Historia	Usuario
HU03	Usuario no registrado
<b>Sprint</b>	<b>Prioridad</b>
01	Media
<b>Nombre</b>	<b>T.E</b>
Recibir instrucciones sobre cómo registrarme	6
<b>Responsable</b>	Moreno Saavedra, Marco

**Criterios de Aceptación:**

- El sistema detecta usuarios no registrados
- Proporciona enlaces a los formatos de registro según el tipo de usuario
- Indica los canales para enviar los formatos completados

---

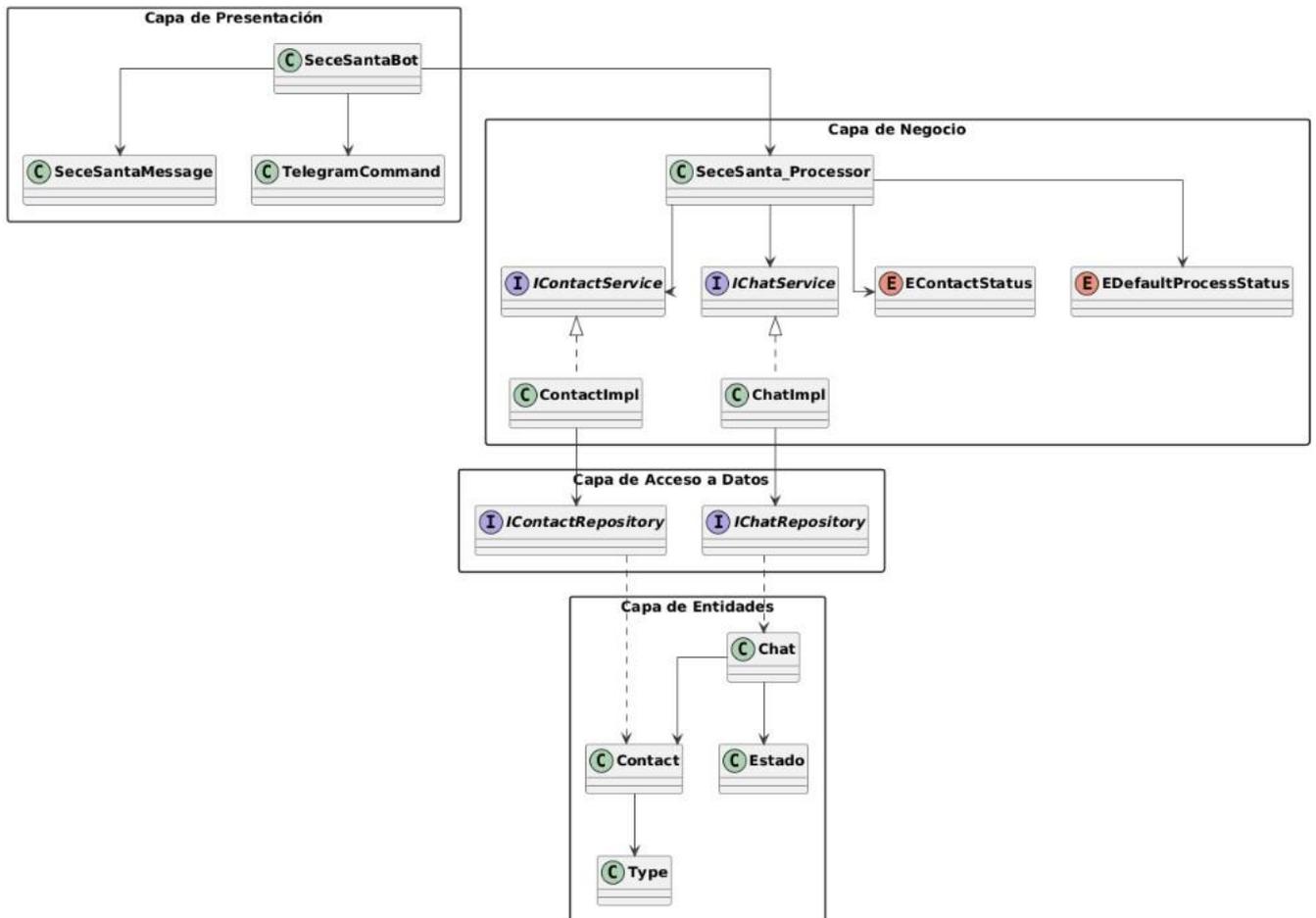
**Tareas:**

- Diseño Capas -MVC: Crear diagrama de flujo para el proceso de registro.
  - Backend: Desarrollar lógica para detectar usuarios no registrados y proporcionar enlaces.
  - Frontend: Implementar mensajes informativos y enlaces a formatos de registro.
- 

*Nota.* Elaboración propia

**Figura 10**

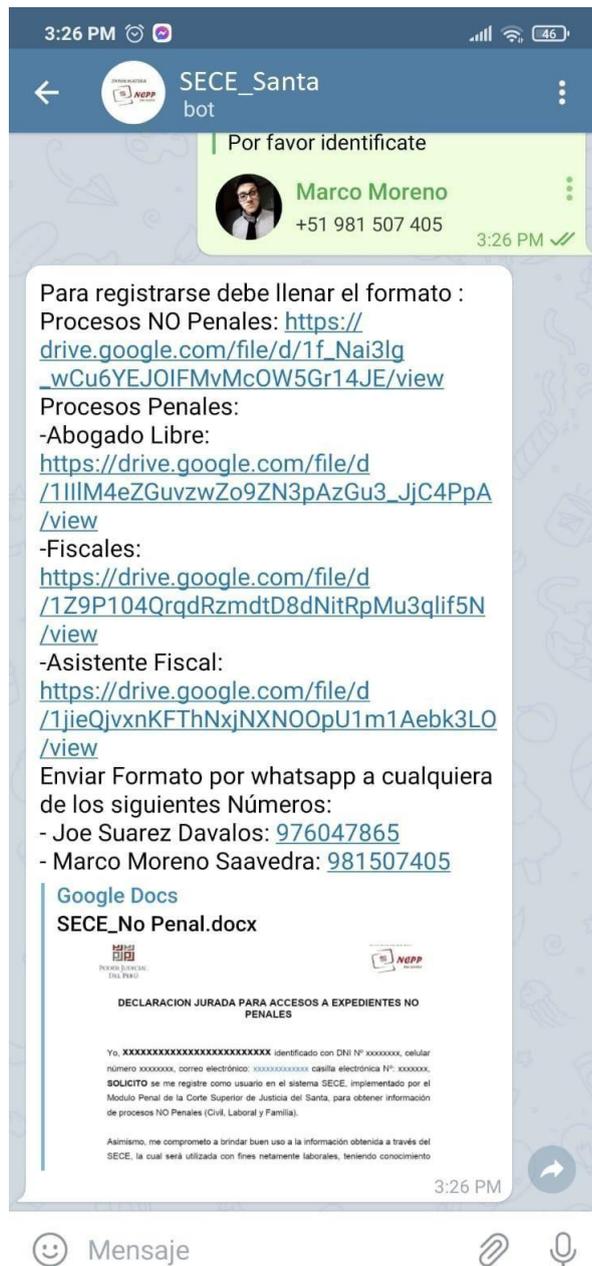
*Diagrama de Capas-MVC para la HU3*



*Nota.* Elaboración propia

**Figura 11**

*UI para la HU03: Recibir instrucciones de registro*



*Nota. Elaboración propia*

**Tabla 12:**

*Especificación de la HU04: Consultar expedientes por número*

<b>N° Historia</b>	<b>Usuario</b>
HU04	Usuario registrado
<b>Sprint</b>	<b>Prioridad</b>
02	Alta
<b>Nombre</b>	<b>T.E</b>
Consultar expedientes por número	6
<b>Responsable</b>	Moreno Saavedra, Marco

**Criterios de Aceptación:**

- El sistema acepta el formato correcto de número de expediente
- Permite filtrar por especialidad (penal, familia, civil, laboral, etc.)
- Muestra resultados únicos o listas para selección

**Tareas:**

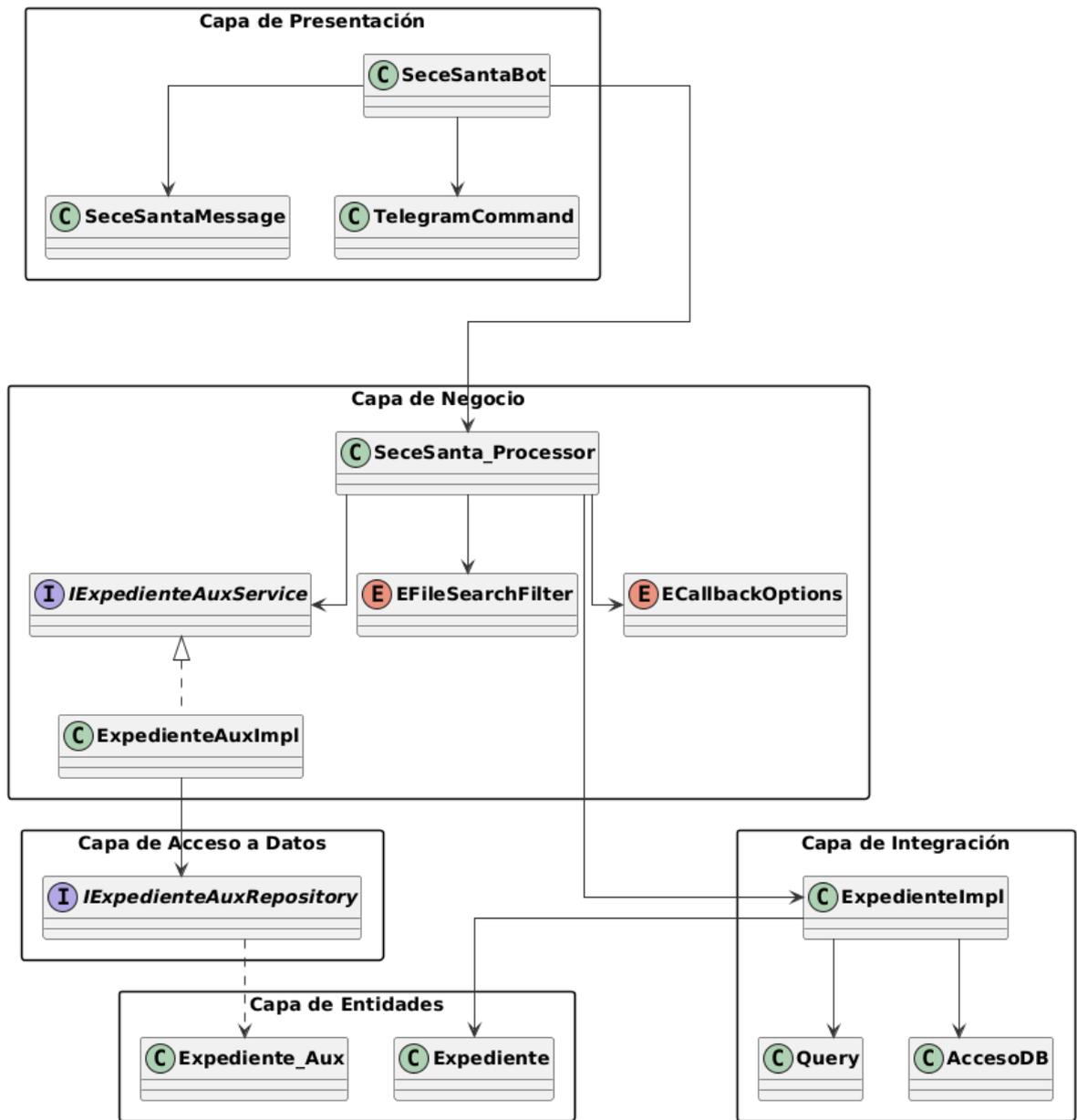
- Diseño de Capas-MVC: Crear diagrama de clases para Expediente y Expediente\_Aux.
- Base de datos: Implementar tabla expediente\_aux para almacenar resultados de búsqueda.
- Backend: Desarrollar IExpedienteAuxService e implementación para consultas por número.
- Frontend: Implementar menú de filtro por especialidad y validación de formato.

---

*Nota.* Elaboración propia

**Figura 12**

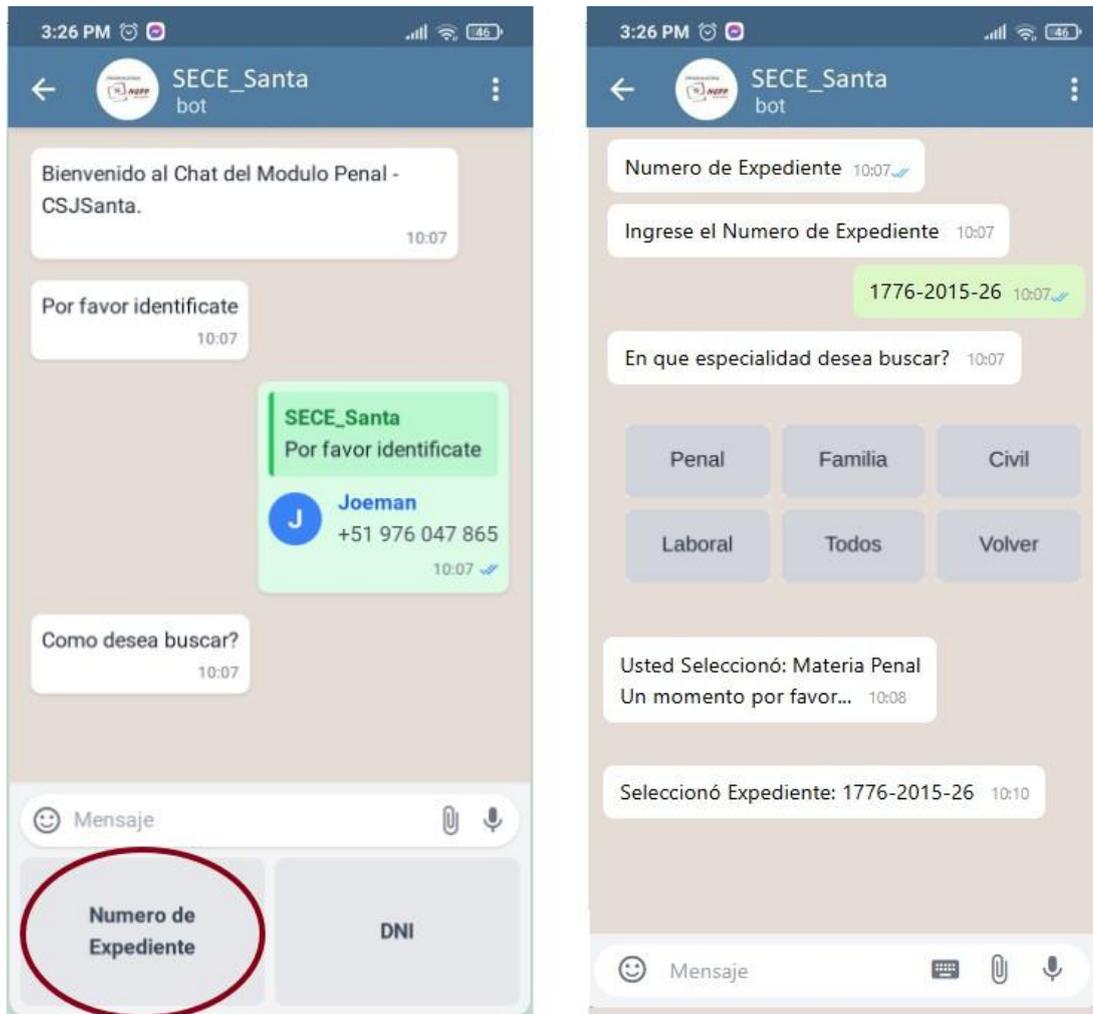
*Diagrama de Capas-MVC para la HU04*



*Nota.* Elaboración propia

**Figura 13**

*UI para la HU4: Consultar expedientes por número*



*Nota.* Elaboración propia

**Tabla 13:**

*Especificación de la HU05: Consultar expedientes por DNI del imputado*

<b>Nº Historia</b>	<b>Usuario</b>
HU05	Usuario registrado
<b>Sprint</b>	<b>Prioridad</b>
02	Alta
<b>Nombre</b>	<b>T.E</b>
Consultar expedientes por DNI del imputado	4
<b>Responsable</b>	Moreno Saavedra, Marco
<b>Criterios de Aceptación:</b>	
<ul style="list-style-type: none"><li>• El sistema valida el formato del DNI</li><li>• Verifica el apersonamiento para abogados libres</li></ul>	

- Permite filtrar por especialidad

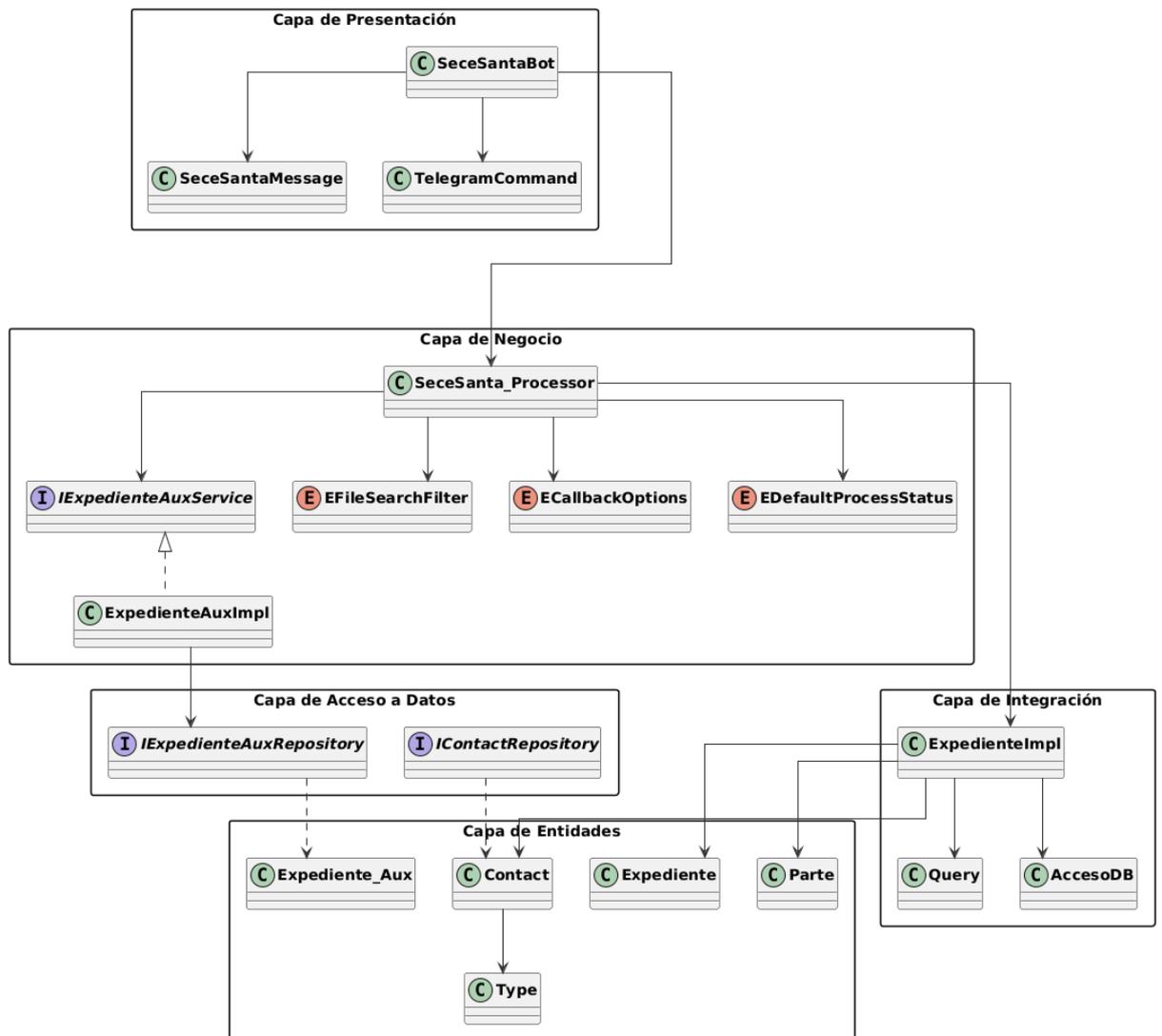
**Tareas:**

- Diseño de Capas-MVC: Ampliar diagrama para incluir relación Expediente-Persona.
- Backend: Desarrollar lógica para validar DNI y verificar apersonamiento.
- Frontend: Implementar interfaz para ingreso de DNI y filtros por especialidad.

*Nota.* Elaboración propia

**Figura 14**

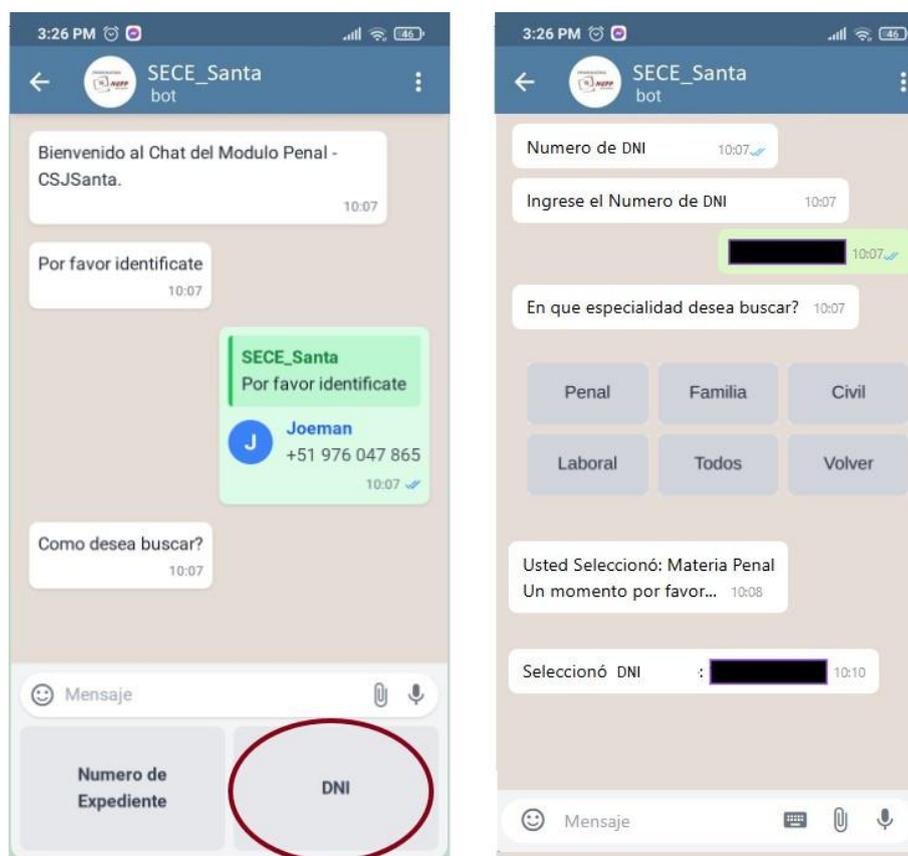
*Diagrama Capas-MVC para la HU05*



*Nota.* Elaboración propia

**Figura 15**

UI para la HU05: Consultar expedientes por DNI del imputado



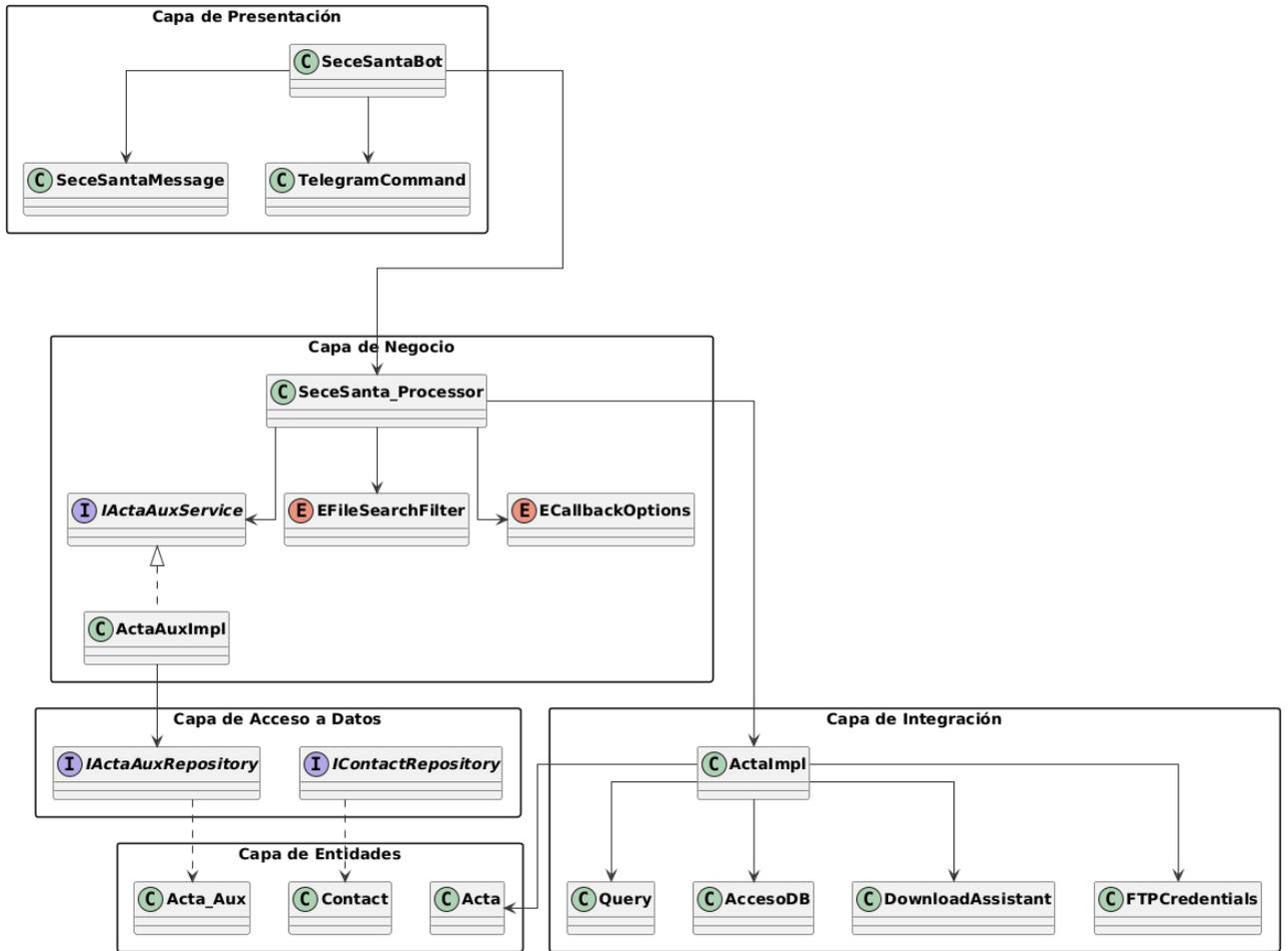
**Tabla 14:**

Especificación de la HU06: Actas de audiencia de un expediente

N° Historia	Usuario
HU06	Usuario registrado
<b>Sprint</b>	<b>Prioridad</b>
03	Alta
<b>Nombre</b>	<b>T.E</b>
Acceder a actas de audiencia de un expediente	5
<b>Responsable</b>	Moreno Saavedra, Marco
<b>Criterios de Aceptación:</b>	
<ul style="list-style-type: none"> <li>• El sistema muestra las actas disponibles</li> <li>• Permite seleccionar por fecha específica o la más reciente</li> <li>• Facilita la descarga del documento</li> </ul>	
<b>Tareas:</b>	
<ul style="list-style-type: none"> <li>• Diseño de Capas-MVC: Crear diagrama de clases para Acta y Acta_Aux.</li> <li>• Base de datos: Implementar tabla acta_aux para almacenar resultados de búsqueda.</li> <li>• Backend: Desarrollar IActaAuxService, ActaImpl y lógica de descarga FTP.</li> <li>• Frontend: Implementar menú de selección por fecha y descarga de actas.</li> </ul>	

Nota. Elaboración propia

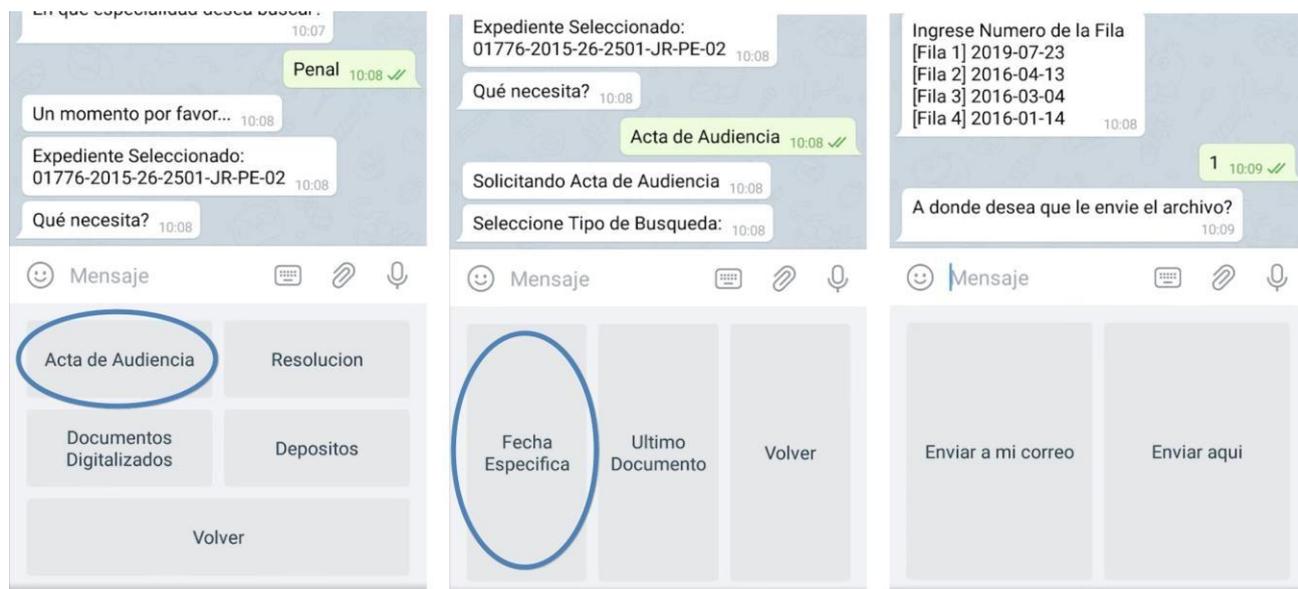
**Figura 16**  
*Diagrama de Capas-MVC para la HU06*



*Nota.* Elaboración propia

**Figura 17**

*UI para la HU06: Acceder a actas de audiencia de un expediente*



*Nota.* Elaboración propia

**Tabla 15:**

*Especificación de la HU07: Acceder a resoluciones de un expediente*

N° Historia	Usuario
HU07	Usuario registrado
<b>Sprint</b>	<b>Prioridad</b>
03	Alta
<b>Nombre</b>	<b>T.E</b>
Acceder a resoluciones de un expediente	6
<b>Responsable</b>	Moreno Saavedra, Marco

**Criterios de Aceptación:**

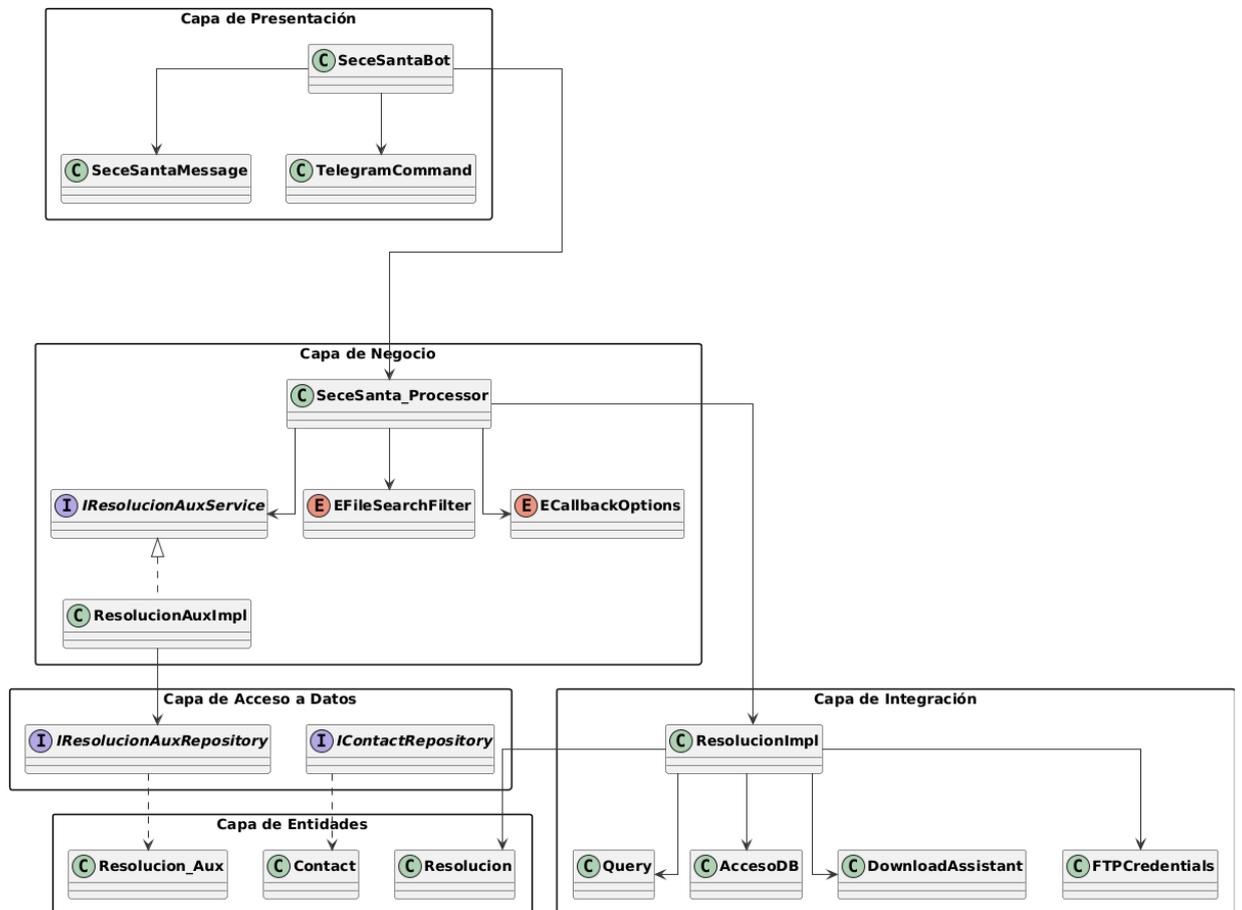
- El sistema muestra las resoluciones disponibles
- Permite seleccionar por fecha específica o la más reciente
- Facilita la descarga del documento

**Tareas:**

- Diseño de Capas-MVC: Crear diagrama de clases para Resolucion y Resolucion\_Aux.
- Base de datos: Implementar tabla resolucion\_aux para almacenar resultados.
- Backend: Desarrollar IResolucionAuxService e implementación para consultas.
- Frontend: Implementar menú de selección por fecha y descarga de resoluciones.

*Nota.* Elaboración propia

**Figura 18**  
 Diagrama de Capas-MVC para la HU07



Nota. Elaboración propia

**Figura 19**  
 UI para la HU07: Acceder a resoluciones de un expediente



Nota. Elaboración propia

**Tabla 16:**

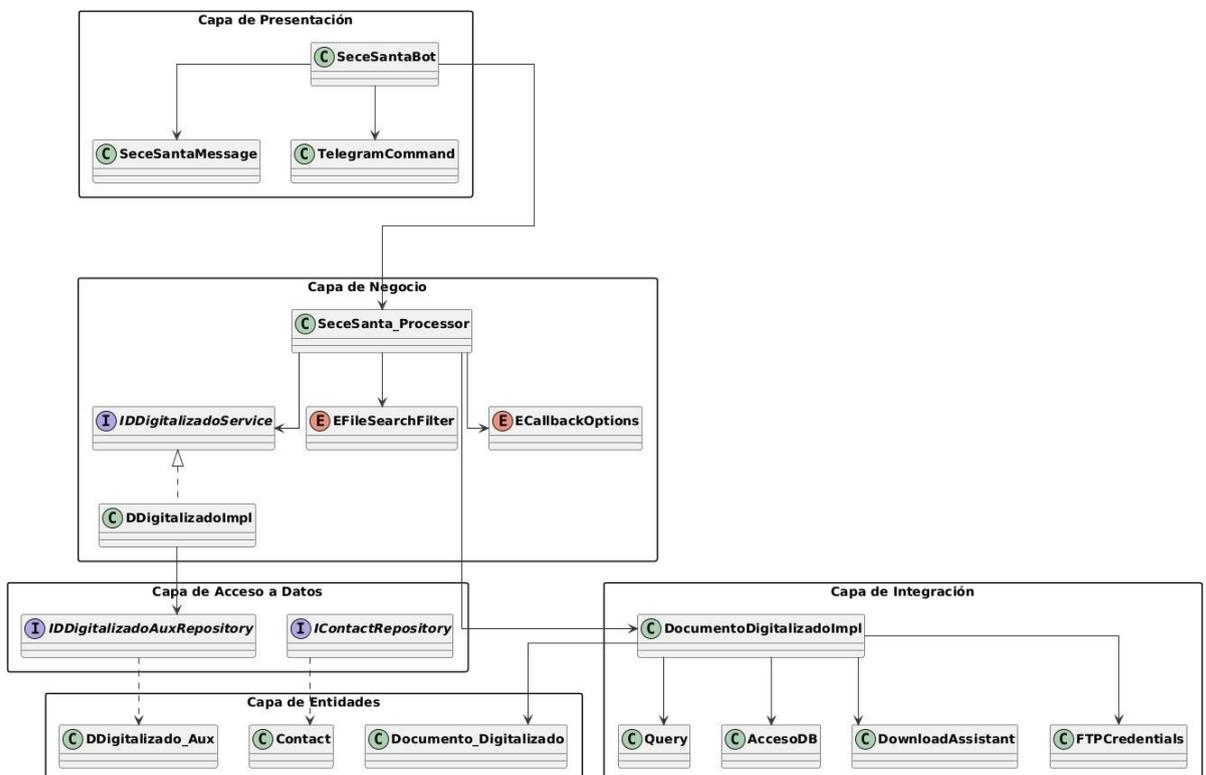
*Especificación de la HU08: Acceder a digitalizados de un expediente*

N° Historia	Usuario
HU08	Usuario registrado
<b>Sprint</b>	<b>Prioridad</b>
03	Alta
<b>Nombre</b>	<b>T.E</b>
Acceder a documentos digitalizados de un expediente	7
<b>Responsable</b>	Moreno Saavedra, Marco
<b>Criterios de Aceptación:</b>	
<ul style="list-style-type: none"> <li>• El sistema muestra los documentos disponibles</li> <li>• Permite seleccionar por fecha específica o el más reciente</li> <li>• Facilita la descarga del documento</li> </ul>	
<b>Tareas:</b>	
<ul style="list-style-type: none"> <li>• Diseño de Capas-MVC: Crear diagrama de clases para Documento_Digitalizado y DDigitalizado_Aux.</li> <li>• Base de datos: Implementar tabla para documentos digitalizados.</li> <li>• Backend: Desarrollar IDDigitalizadoService e implementación para consultas.</li> <li>• Frontend: Implementar menú de selección por fecha y descarga de documentos.</li> </ul>	

*Nota.* Elaboración propia

**Figura 20**

*Diagrama Capas-MVC para la HU08*



*Nota.* Elaboración propia

**Figura 21**

*UI para la HU08: Acceder a documentos digitalizados de un expediente*



*Nota. Elaboración propia*

**Tabla 17:**

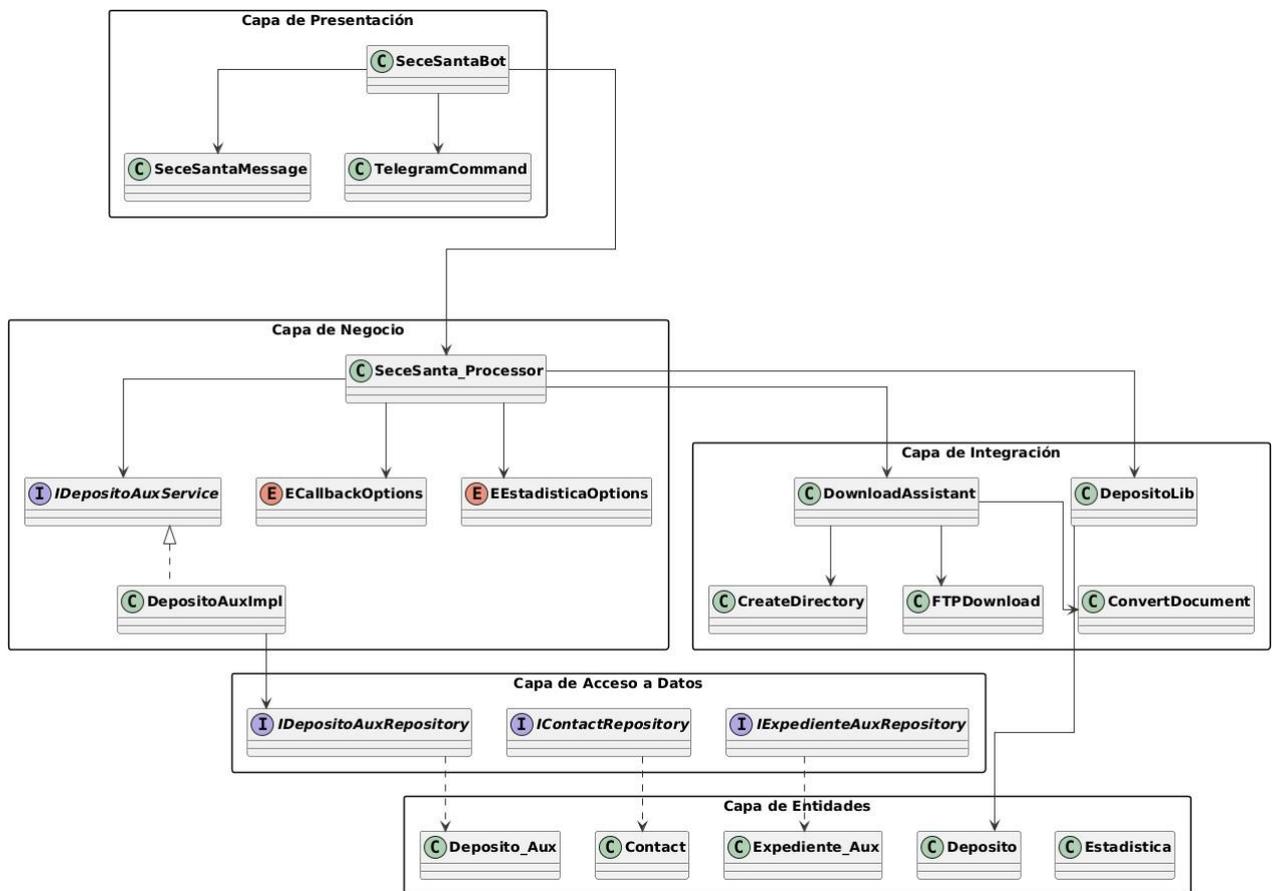
*Especificación de la HU09: Consultar depósitos judiciales*

N° Historia	Usuario
HU09	Usuario registrado
<b>Sprint</b>	<b>Prioridad</b>
04	Media
<b>Nombre</b>	<b>T.E</b>
Consultar información sobre depósitos judiciales	6
<b>Responsable</b>	Moreno Saavedra, Marco
<b>Criterios de Aceptación:</b>	
<ul style="list-style-type: none"> <li>• El sistema verifica si existen depósitos en el expediente</li> <li>• Genera automáticamente un reporte con la información</li> <li>• Permite descargar el reporte</li> </ul>	
<b>Tareas:</b>	
<ul style="list-style-type: none"> <li>• Diseño de Capas-MVC: Crear diagrama de clases para Deposito y Deposito_Aux.</li> <li>• Base de datos: Implementar tabla deposito_aux para almacenar resultados.</li> <li>• Backend: Desarrollar IDepositoAuxService e implementación para consultas.</li> <li>• Frontend: Implementar generación automática de reportes de depósitos.</li> </ul>	

*Nota. Elaboración propia*

**Figura 22**

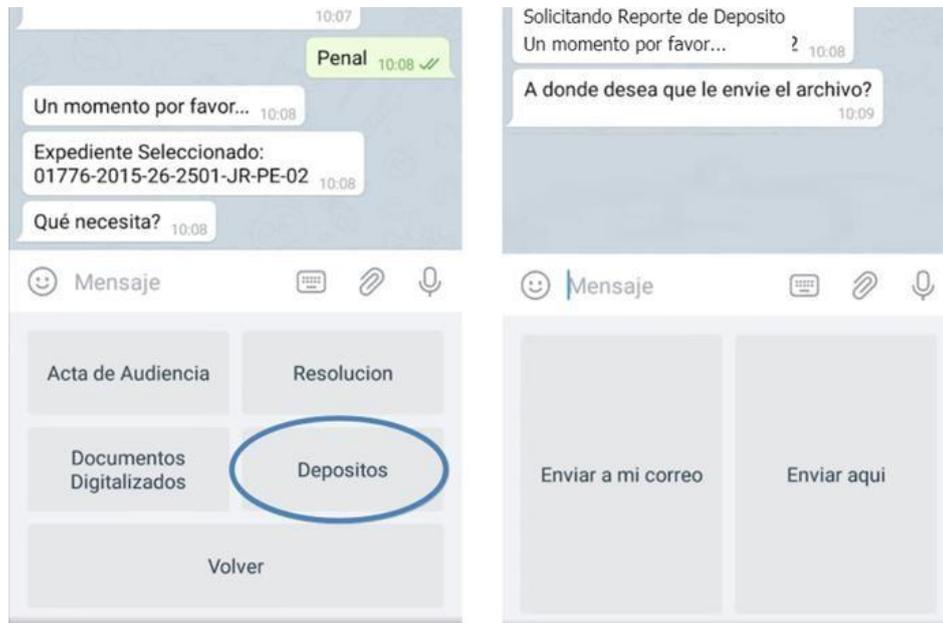
Diagrama de Capas-MVC para la HU09



*Nota. Elaboración propia*

### Figura 23

UI para la HU09: Consultar información sobre depósitos judiciales



Nota. Elaboración propia

### Tabla 18:

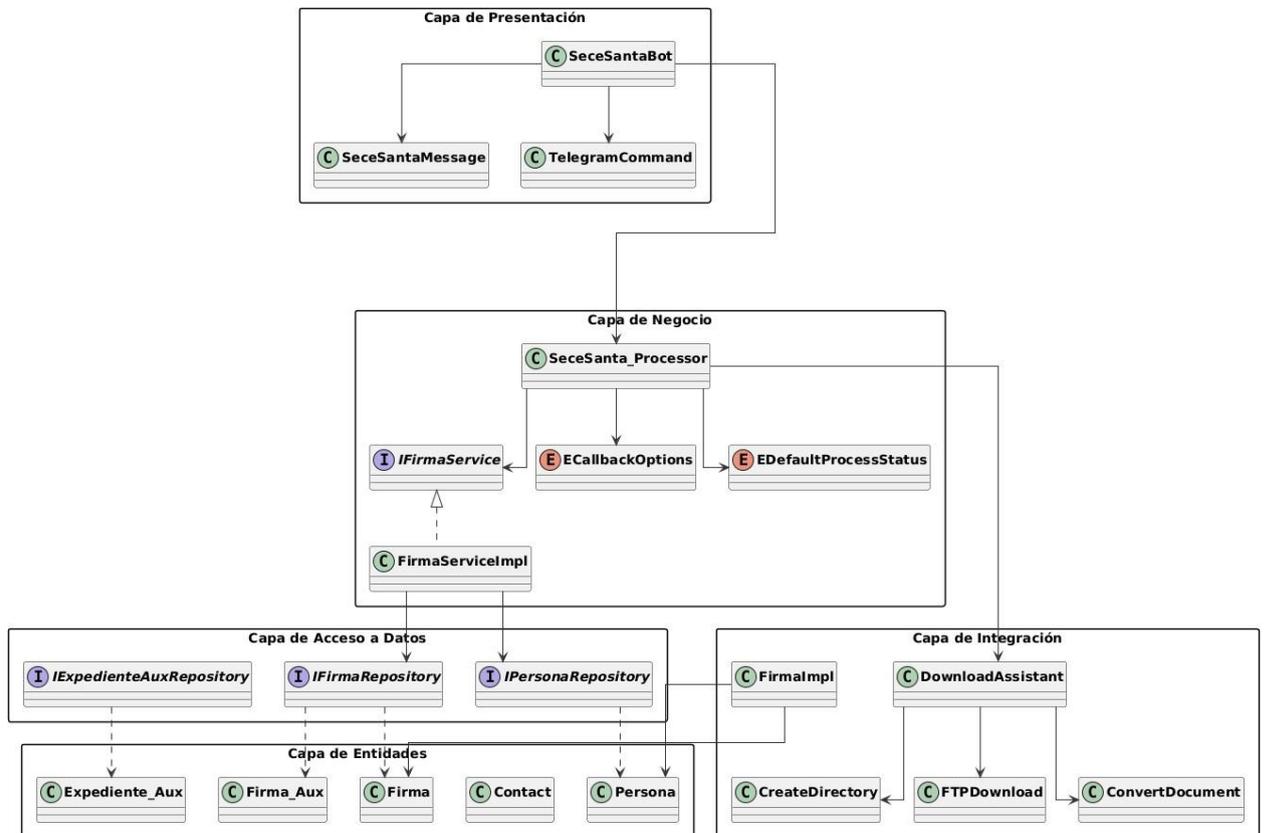
Especificación de la HU10: Consultar registros de firmas.

N° Historia	Usuario
HU10	Usuario registrado
<b>Sprint</b>	<b>Prioridad</b>
04	Media
<b>Nombre</b>	<b>T.E</b>
Consultar el registro de firmas en un expediente	6
<b>Responsable</b>	Moreno Saavedra, Marco
<b>Criterios de Aceptación:</b>	
<ul style="list-style-type: none"><li>• El sistema verifica las personas que han firmado</li><li>• Muestra lista para selección si hay múltiples firmantes</li><li>• Permite descargar el registro de firmas</li></ul>	
<b>Tareas:</b>	
<ul style="list-style-type: none"><li>• Diseño de Capas-MVC: Crear diagrama de clases para Firma y su relación con Persona.</li><li>• Backend: Desarrollar lógica para verificar firmantes en expedientes.</li><li>• Frontend: Implementar lista de selección para múltiples firmantes.</li></ul>	

Nota. Elaboración propia

**Figura 24**

*Diagrama de Capas-MVC para la HU10*



*Nota. Elaboración propia*

**Figura 25**

*UI para la HU10: Consultar el registro de firmas en un expediente*



*Nota. Elaboración propia*

**Tabla 19:**

*Especificación de la HU11: Seleccionar recepción de documentos*

<b>N° Historia</b>	<b>Usuario</b>
HU11	Usuario registrado
<b>Sprint</b>	<b>Prioridad</b>
04	Media
<b>Nombre</b>	<b>T.E</b>
Seleccionar el modo de recepción de documentos	5
<b>Responsable</b>	Moreno Saavedra, Marco

**Criterios de Aceptación:**

- El sistema ofrece opciones de envío (chat o correo)
- Envía correctamente los documentos por el canal seleccionado
- Confirma el envío exitoso

**Tareas:**

- Diseño de Capas-MVC: Crear diagrama de flujo para opciones de envío.
- Backend: Desarrollar lógica para envío por diferentes canales (chat/correo).
- Frontend: Implementar menú opcionesEnvioMenu con confirmación de envío.

---

*Nota.* Elaboración propia

**Figura 26**

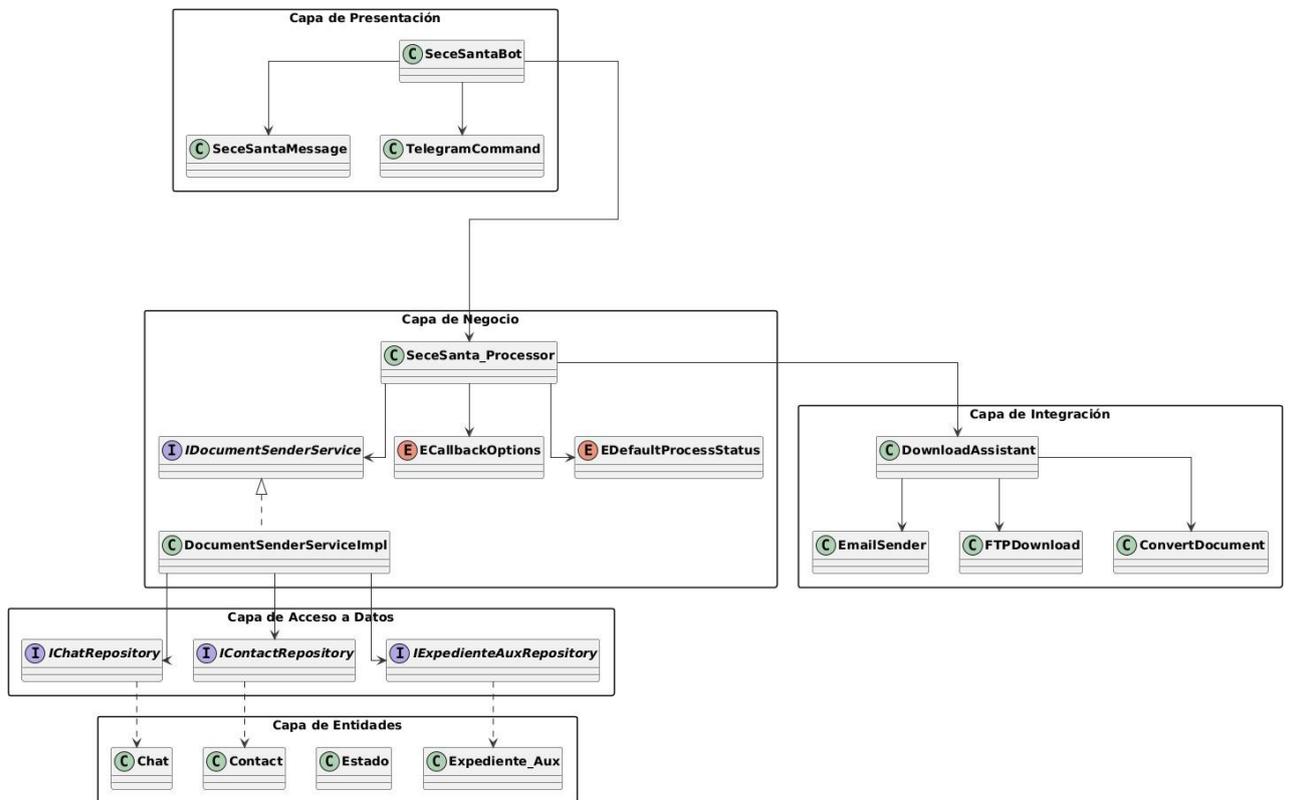
*UI para la HU11: Seleccionar el modo de recepción de documentos*



*Nota.* Elaboración propia

**Figura 27**

*Diagrama de Capas-MVC para la HU11*



*Nota.* Elaboración propia

**Tabla 20:**

*Especificación de la HU12: Consultar principio de oportunidad*

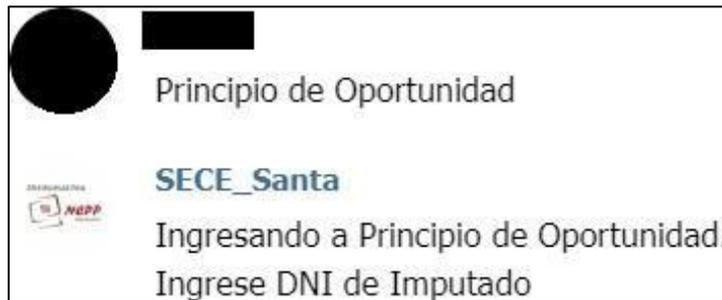
NºHistoria	Usuario
HU12	Usuario registrado
<b>Sprint</b>	<b>Prioridad</b>
04	Media
<b>Nombre</b>	<b>T.E</b>
Consultar información sobre principio de oportunidad	6
<b>Responsable</b>	Moreno Saavedra, Marco
<b>Criterios de Aceptación:</b>	
<ul style="list-style-type: none"> <li>• El sistema solicita el DNI del imputado</li> <li>• Genera un reporte con la información disponible</li> <li>• Permite descargar el documento</li> </ul>	
<b>Tareas:</b>	

- 
- Diseño de Capas-MVC: Crear diagrama de clases para la entidad de principio de oportunidad.
  - Backend: Desarrollar lógica para consultar información por DNI del imputado.
  - Frontend: Implementar interfaz para solicitud de DNI y descarga de reportes.
- 

*Nota.* Elaboración propia

### **Figura 28**

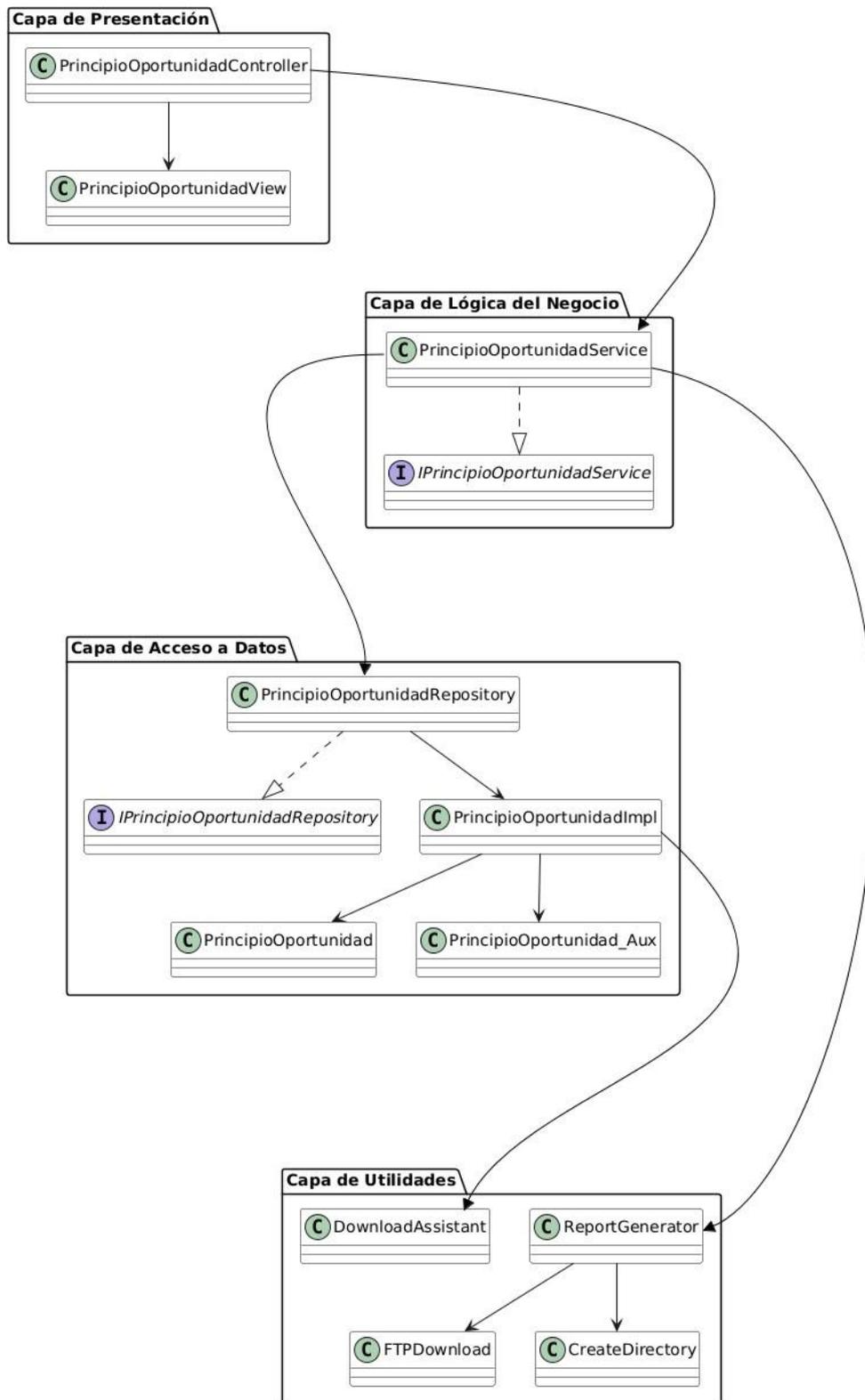
*UI para la HUI2: Consultar información sobre principio de oportunidad*



*Nota.* Elaboración propia

**Figura 29**

Diagrama de Capas-MVC para la HU12



*Nota.* Elaboración propia

**Tabla 21:**

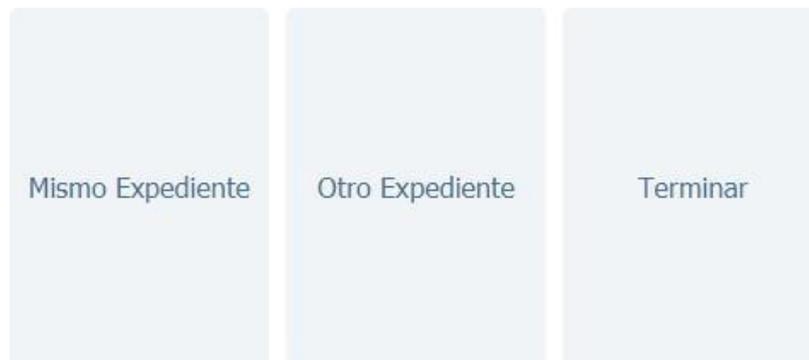
*Especificación de la HU13: Navegar entre diferentes expedientes*

<b>N° Historia</b>	<b>Usuario</b>
HU13	Usuario registrado
<b>Sprint</b>	<b>Prioridad</b>
05	Baja
<b>Nombre</b>	<b>T.E</b>
Navegar entre diferentes expedientes	6
<b>Responsable</b>	Moreno Saavedra, Marco
<b>Criterios de Aceptación:</b>	
<ul style="list-style-type: none"><li>• El sistema ofrece opciones para continuar con el mismo expediente</li><li>• Permite cambiar a otro expediente</li><li>• Mantiene el contexto de la sesión del usuario</li></ul>	
<b>Tareas:</b>	
<ul style="list-style-type: none"><li>• Diseño de Capas-MVC: Crear diagrama de flujo para navegación entre expedientes.</li><li>• Backend: Desarrollar lógica para mantener el contexto de sesión y expedientes.</li><li>• Frontend: Implementar menú opcionesUltimoMenu con opciones de navegación.</li></ul>	

*Nota. Elaboración propia*

**Figura 30**

*UI para la HU13: Navegar entre diferentes expedientes*



*Nota. Elaboración propia*

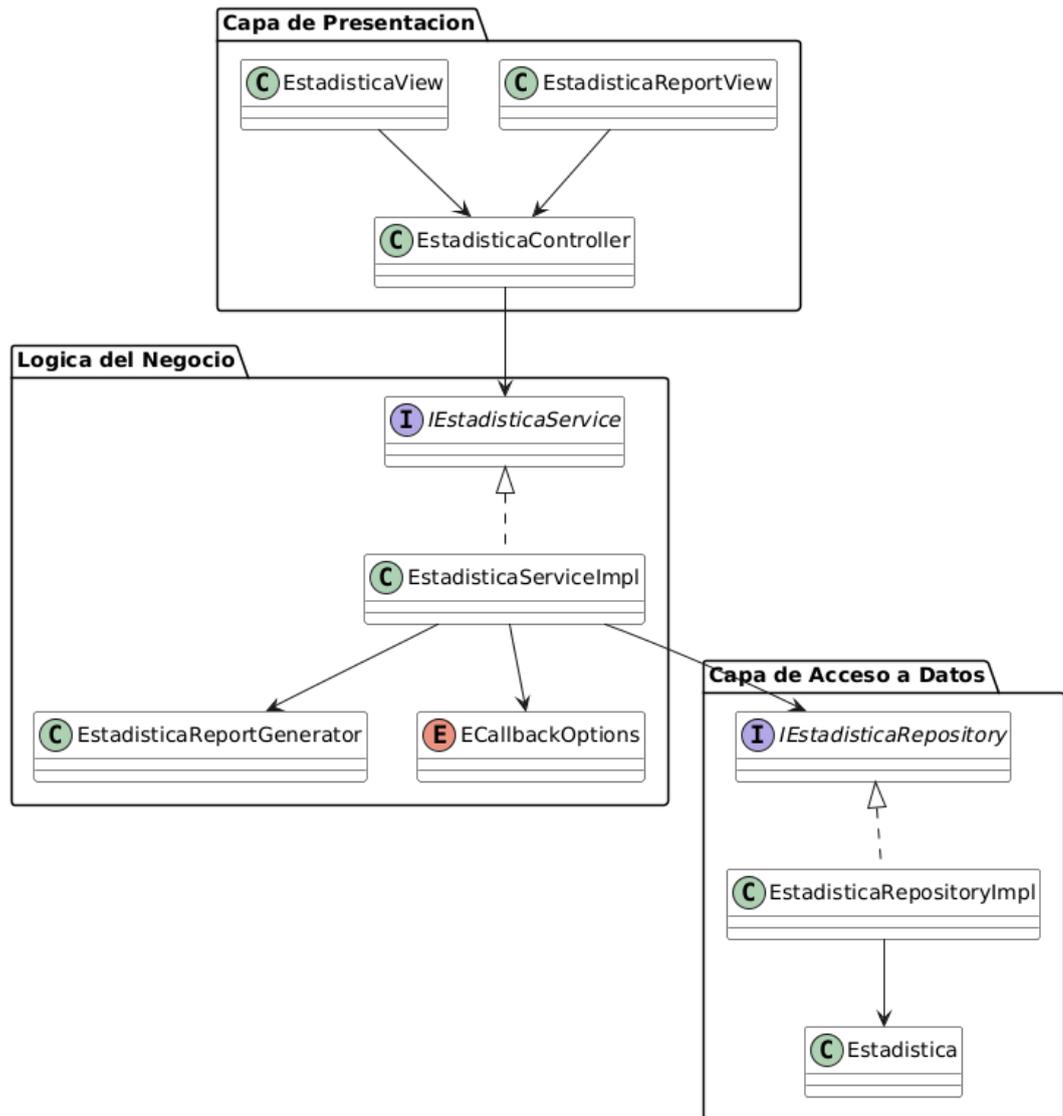


- Backend: Desarrollar IEstadisticaService e implementación para registro y consulta.
- Frontend: Implementar generación de reportes por tipo de usuario y especialidad.

*Nota.* Elaboración propia

**Figura 32**

*Diagrama de Capas-MVC para la HUI4*



*Nota.* Elaboración propia

**Tabla 23:**

*Especificación de la HU15: Enviar mensajes masivos a los usuarios*

<b>N° Historia</b>	<b>Usuario</b>
HU15	Administrador
<b>Sprint</b>	<b>Prioridad</b>
05	Baja
<b>Nombre</b>	<b>T.E</b>
Enviar mensajes masivos a los usuarios	5
<b>Responsable</b>	Moreno Saavedra, Marco
<b>Criterios de Aceptación:</b>	
<ul style="list-style-type: none"><li>• El sistema permite redactar mensajes para todos los usuarios</li><li>• Envía notificaciones a través del chat</li><li>• Confirma el envío exitoso</li></ul>	
<b>Tareas:</b>	
<ul style="list-style-type: none"><li>• Diseño de Capas-MVC: Crear diagrama de flujo para envío de mensajes masivos.</li><li>• Backend: Desarrollar lógica para obtener lista de chats (getChatsforPersonalMessage).</li><li>• Frontend: Implementar interfaz para redacción y envío de mensajes masivos.</li></ul>	

---

*Nota.* Elaboración propia

**Figura 33**

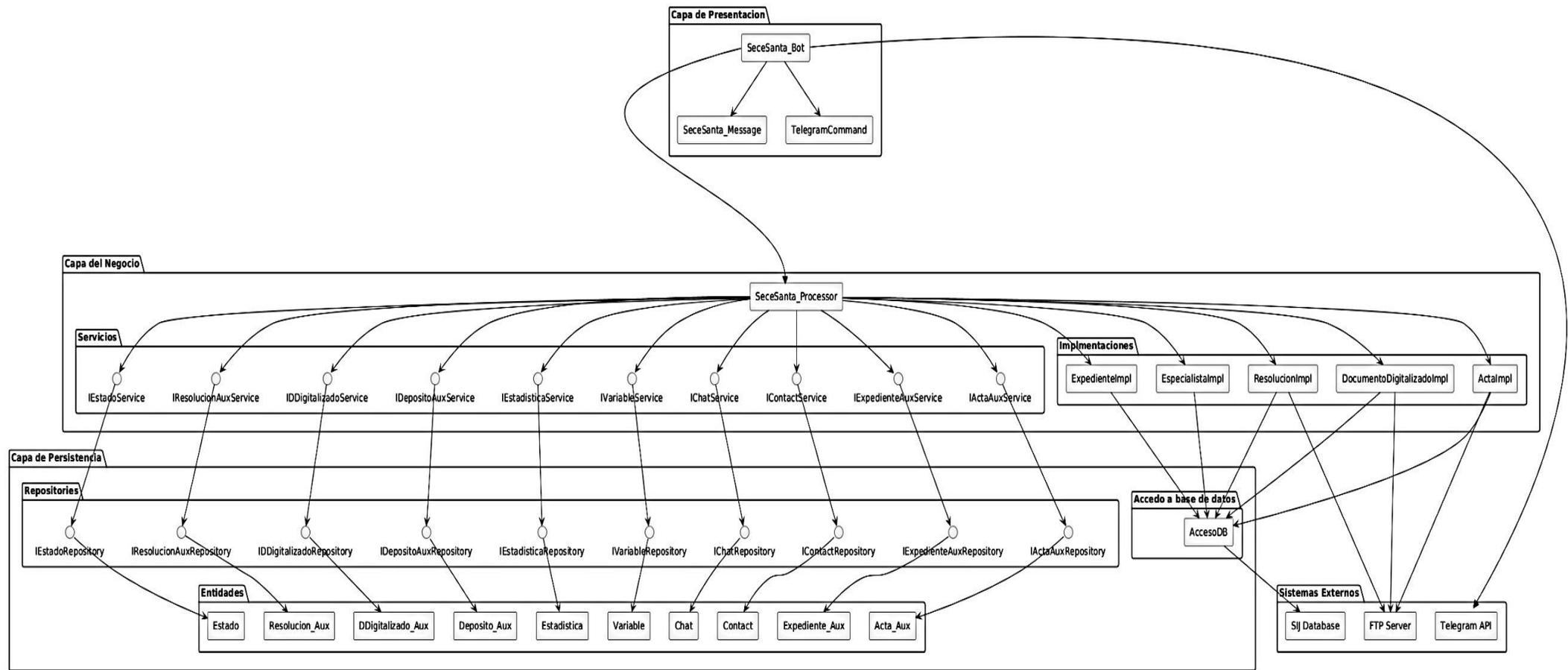
Diagrama Capas-MVC para la HU15



*Nota.* Elaboración propia

**Figura 34**

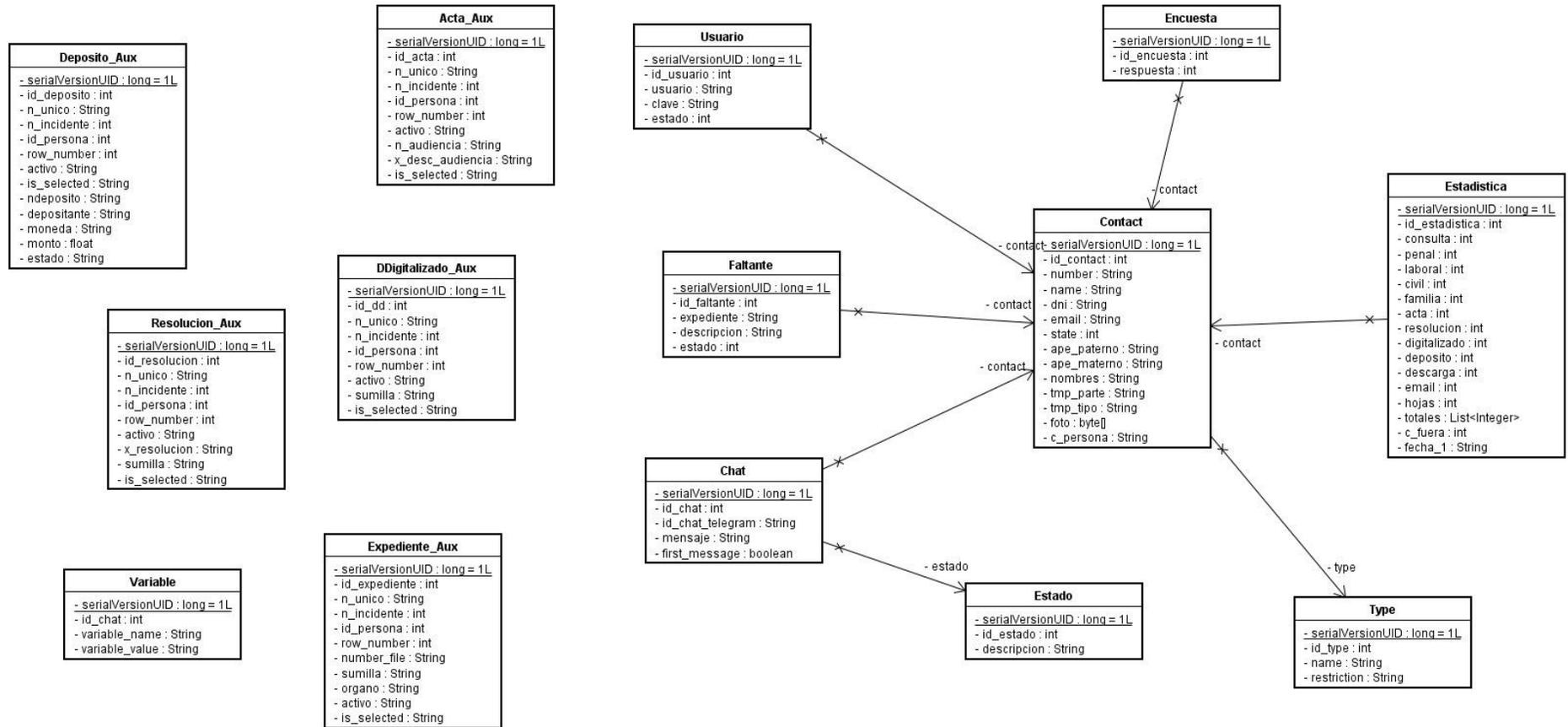
*Arquitectura de la aplicación*



*Nota.* Elaboración propia

**Figura 35**

*Arquitectura del dominio de la aplicación*

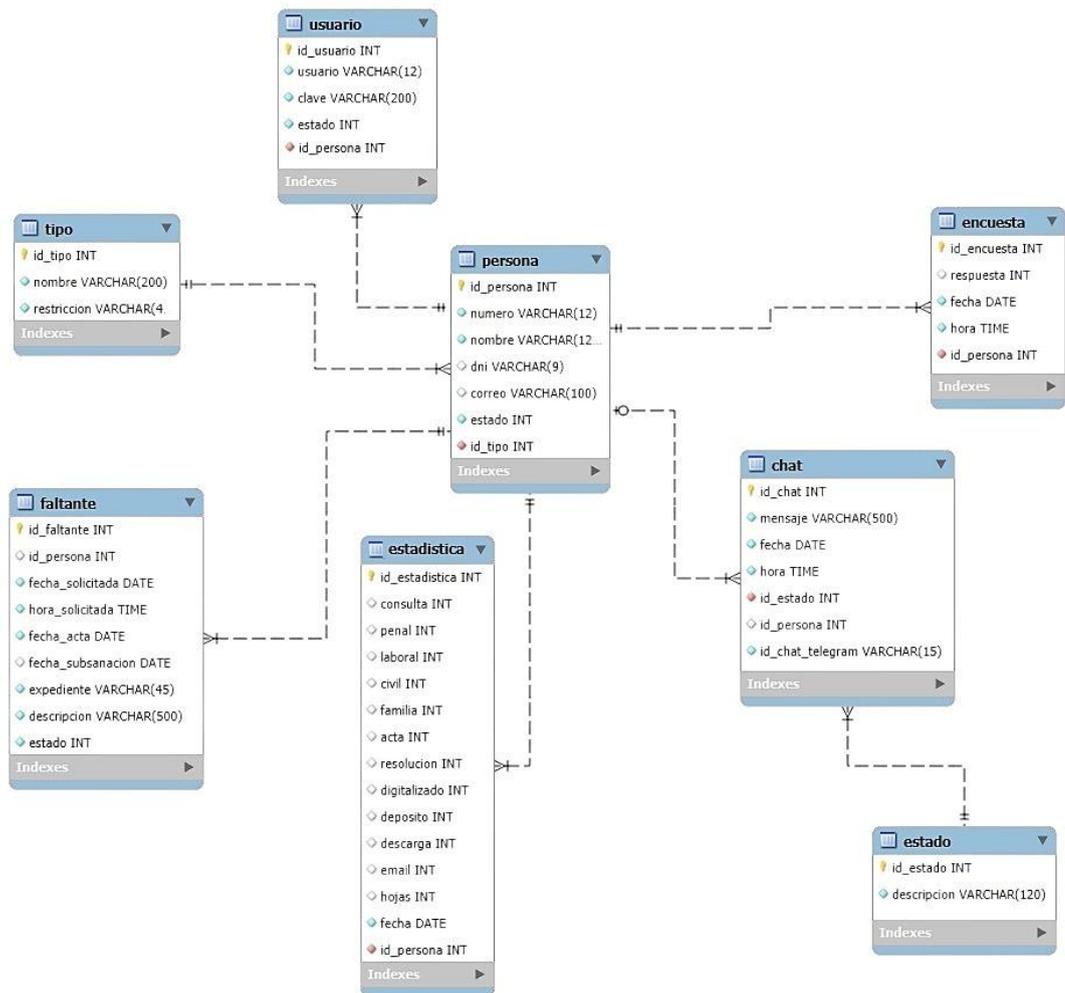


*Nota.* Elaboración propia

## 6.2. Diagrama de Base de Datos

Figura 36

Diagrama de Base de Datos de la aplicación



Nota. Elaboración propia

## **VII. Ubicación de las experiencias en el marco del sustento teórico**

Contar con una comprensión sólida y fundamentada del marco teórico relacionado con el problema de investigación resultó esencial para el desarrollo exitoso del presente proyecto. A lo largo del desarrollo del proyecto, se buscó que los conceptos teóricos aprendidos se integraran de manera concreta en cada etapa del trabajo práctico. Esta conexión entre la teoría y la implementación real fue clave para dar forma a una solución funcional, centrada en las necesidades de los usuarios.

### **API Rest:**

Se eligió una arquitectura que permitiera al sistema comunicarse fácilmente con otros servicios: API REST. Esta decisión facilitó que el chatbot pudiera intercambiar información de manera clara y ordenada, sin depender de procesos complicados. Gracias a ello, fue posible integrar nuevas funciones con mayor facilidad y asegurar que el sistema pudiera crecer y adaptarse con el tiempo.

### **Chatbot:**

El desarrollo del chatbot fue uno de los ejes centrales del proyecto. Su propósito fue brindar una atención automatizada, accesible y clara a los usuarios. Se crearon distintos flujos de conversación que permitieran resolver consultas de forma natural. Esto no solo agilizó la atención, sino que también mejoró la experiencia de quienes interactúan con el sistema, haciéndola más directa y eficiente.

### **API Chatbot Telegram:**

Para acercar el sistema a los usuarios, se optó por usar Telegram como canal de comunicación. Esta plataforma, bastante conocida y fácil de usar, permitió que el chatbot estuviera disponible en distintos dispositivos sin necesidad de instalar nada adicional. Durante la integración, se resolvieron aspectos importantes como

el control del acceso y la entrega oportuna de mensajes, lo que aseguró una interacción segura y sin interrupciones.

### **Scrum:**

La organización del trabajo se realizó utilizando esta metodología ágil que dividió todo el proceso en ciclos cortos (sprints), con metas claras y entregas frecuentes. Esto permitió mantener un ritmo de trabajo constante, adaptarse a los cambios y tomar decisiones en función de lo que realmente aportaba valor al proyecto. Los tableros de seguimiento ayudaron a tener siempre una visión clara del avance y a distribuir mejor las tareas entre los integrantes del equipo.

### **Spring Framework:**

Para dar soporte al funcionamiento del sistema, se trabajó con una herramienta que facilitó la organización interna del código y la cohesión de sus diferentes componentes. Esta elección permitió construir un sistema ordenado, modular y estable y preparado para responder bien ante una posible escalabilidad de las funcionalidades. Además, la amplia comunidad y documentación disponible contribuyeron a resolver posibles problemas técnicos de manera eficiente.

### **PostgreSQL:**

Toda la información que el sistema genera —como los registros de conversaciones o los datos de los usuarios— fue almacenada usando una base de datos confiable y segura: PostgreSQL. Esta herramienta permitió mantener los datos bien organizados, siempre disponibles y protegidos contra pérdidas, lo cual fue fundamental para el correcto funcionamiento del chatbot.

En conjunto, la aplicación práctica de estos conceptos no solo permitió alcanzar los objetivos del proyecto, sino también adquirir experiencia valiosa en el uso de tecnologías actuales y en el desarrollo de soluciones centradas en el usuario. Este proceso confirmó la importancia de unir el conocimiento teórico con la práctica, adaptándolo a contextos reales que exigen respuestas eficientes, accesibles y sostenibles.

### **VIII. Aportes logrados para el desarrollo del centro laboral**

Los aportes del proyecto han permitido generar mejoras importantes para el Módulo Penal CSJ Santa. Se ha logrado disminuir considerablemente los tiempos de respuesta para consultas de expedientes judiciales, eliminando las demoras causadas por búsquedas manuales en archivos físicos y reduciendo la carga de trabajo del personal administrativo, especialmente durante horarios de alta demanda de consultas por parte de fiscales y abogados.

Se ha centralizado en un solo sistema la funcionalidad de consulta y entrega de documentos judiciales mediante múltiples canales, lo cual ha permitido desacoplar esta funcionalidad de los procesos administrativos tradicionales y facilitar el trabajo de mantenimiento al equipo técnico del área de informática. Esta centralización ha mejorado la eficiencia operativa y ha reducido la duplicidad de esfuerzos en la atención a usuarios.

El uso del asistente virtual ha permitido al Módulo Penal de la Corte Superior de Justicia del Santa fortalecer sus canales digitales, facilitando que los usuarios se mantengan informados sobre el estado de sus consultas, la disponibilidad de documentos y cualquier actualización relacionada con sus expedientes. Esta mejora ha generado un mayor nivel de confianza entre fiscales, abogados y personal administrativo, quienes han empezado a utilizar con más frecuencia estas herramientas tecnológicas como apoyo en sus actividades diarias.

Además, se ha logrado ampliar el acceso a la información judicial, permitiendo que los usuarios autorizados puedan hacer consultas desde cualquier lugar y en cualquier momento, sin necesidad de acudir físicamente a las oficinas del módulo. Esta facilidad de acceso no solo agiliza los trámites, sino que representa un paso importante hacia la modernización de los procesos institucionales. De esta forma,

la institución reafirma su compromiso con la innovación tecnológica como una vía para mejorar la eficiencia y la calidad del servicio de justicia.

#### **IX. Aportes para la formación profesional**

A partir de la experiencia ganada en la implementación de este proyecto y el recorrido profesional desarrollado, se han logrado importantes contribuciones para mi formación en Ingeniería de Sistemas. Se ha diseñado una arquitectura que ha implementado un sistema de consultas automatizadas mediante la integración de chatbots con APIs REST, lo cual ha mejorado de forma notable la eficiencia en el acceso a información judicial, evitando demoras en las consultas y asegurando la entrega efectiva de documentos a los usuarios autorizados.

Se ha diseñado una API REST escalable utilizando Spring Boot, mediante el uso de controladores modulares, servicios optimizados y repositorios eficientes que han permitido el procesamiento rápido de consultas de expedientes judiciales. Esta arquitectura ha demostrado su capacidad para manejar múltiples usuarios simultáneos sin comprometer el rendimiento del sistema.

Se ha trabajado con bases de datos relacionales optimizadas para el manejo de información judicial, diseñadas para realizar consultas rápidas y eficientes sobre grandes volúmenes de expedientes y documentos digitalizados. Esta experiencia ha permitido comprender la importancia de la optimización de consultas y el diseño adecuado de índices para sistemas de alta demanda.

Se ha implementado la integración con la API de Telegram para el desarrollo del chatbot, permitiendo que los usuarios accedan al sistema desde cualquier dispositivo móvil o computadora. Esta integración ha facilitado la comunicación bidireccional entre usuarios y sistema, mejorando significativamente la experiencia de consulta de información judicial.

Se ha implementado un sistema de entrega de documentos mediante múltiples canales, incluyendo correo electrónico y chat directo, utilizando servicios de mensajería seguros que garantizan la confidencialidad de la información judicial. Esta implementación ha demostrado la importancia de ofrecer opciones flexibles de entrega que se adapten a las necesidades específicas de cada usuario del sistema judicial.

## **X. Conclusiones y recomendaciones**

### **10.1. Conclusiones**

- Se logró realizar un diagnóstico claro y bien fundamentado de los requisitos funcionales y no funcionales del sistema, a partir del análisis detallado de los procesos de consulta judicial que ya se venían utilizando en el entorno institucional.
- Se desarrolló e implementó el chatbot SECE\_Santa como un Producto Viable Mínimo (MVP), empleando la metodología ágil Scrum. Este enfoque permitió entregar avances de forma progresiva, poniendo como prioridad la funcionalidad de consulta en tiempo real, dado su impacto directo en la productividad de fiscales y abogados.
- Se concretó de forma exitosa la integración del chatbot con los sistemas de información principales del Módulo Penal de la Corte Superior de Justicia del Santa. La elección de Telegram como plataforma de despliegue aseguró una alta tasa de adopción por parte de los usuarios sin necesidad de desarrollar una aplicación nativa desde cero.
- Se logró democratizar el acceso a la información judicial, eliminando las barreras geográficas y económicas que antes limitaban las consultas
- Se logró una mejora notable en la experiencia de usuario (UX) para un proceso tradicionalmente complejo y burocrático.
- Se logró optimizar el uso de los recursos humanos en Módulo Penal CSJ Santa. Al automatizar las tareas repetitivas de consulta y entrega de documentos digitales, el sistema libera tiempo del personal administrativo y judicial.

## 10.2. Recomendaciones

- Ampliar la cobertura de canales de comunicación. Considerando que la expansión a otras plataformas de mensajería masiva como WhatsApp para maximizar el alcance y la accesibilidad del chatbot para un mayor número de usuarios.
- Implementar más funcionalidades para permitir acciones más allá de la consulta, como realizar el pago de tasas judiciales o presentar escritos simples, convirtiéndolo en una herramienta de interacción bidireccional.
- Incorporar Analítica e Inteligencia Artificial a través del análisis de los datos de uso para detectar cuellos de botella en el sistema judicial y aplicar Procesamiento de Lenguaje Natural (PLN) para permitir que los usuarios realicen consultas de forma más conversacional e intuitiva.
- Adoptar una Arquitectura de Microservicios: Migrar la arquitectura del sistema a un modelo basado en microservicios. Esto permitiría que cada funcionalidad (consultas, pagos, notificaciones, IA) opere de forma independiente, facilitando las actualizaciones, mejorando la resistencia a fallos y permitiendo que el sistema escale de manera más eficiente.

## **XI. Referencias bibliográficas**

- Adamopoulou, E., & Moussiades, L. (2020). An Overview of Chatbot Technology. *Artificial Intelligence Applications and Innovations*.
- Al-Ayyoub, M., & Al-Zoubi, A. (). Generative vs Intent-based Chatbot for Judicial Advice. In *International Conference on Artificial Intelligence and Law (ICAIL)*. IEEE. <https://ieeexplore.ieee.org/document//>
- Castillo, R., (2022). Chatbot para mejorar la atención de consultas de los estudiantes en la institución educativa niño Jesús de Praga [Universidad Nacional Santiago Antúnez de Mayolo]. <https://hdl.handle.net/20.500.14593/5339>
- De, T., (2022). Implementar un chatbot basado en ibm watson para mejorar el servicio de información recurrente en la empresa inversiones Santa SRL, 2022 [Universidad Nacional Santiago Antúnez de Mayolo]. <https://hdl.handle.net/20.500.14593/5346>
- Fielding, R. T. (2000). *Architectural Styles and the Design of Network-based Software Architectures* (Doctoral dissertation, University of California, Irvine).
- Khandal, R., Meshram, M., & Mahatme, S. (2017). Study of Spring Framework. *International Conference On Emanations in Modern Engineering Science and Management (ICEMESM-2017)*
- Minaya, M., Saavedra, A. (2024). Implementación de chatbot logístico para mejorar la gestión en servicios farmacéuticos en la droguería Lidufarma, Santa Anita, Lima, Perú - 2023 [Universidad María Auxiliadora]. <https://hdl.handle.net/20.500.12970/2161>

- Morocco Sucapuca, A. L. (2023). El acceso a la justicia a través del uso del aplicativo Chatbot-Pj de los litigantes de la Corte Superior de Justicia de Puno – 2023 (Tesis de pregrado, Universidad Nacional del Altiplano). <https://repositorio.unap.edu.pe/handle/20.500.14082/20961>
- Quispe, C., (2018). EL EXPEDIENTE DIGITAL Y SU INCIDENCIA EN LA ADMINISTRACIÓN DE JUSTICIA EN EL PERÚ [Tesis, Universidad Señor de Sipán]. <https://hdl.handle.net/20.500.12802/5100>
- Schwaber, K., & Sutherland, J. (2020). La Guía Scrum: La Guía Definitiva de Scrum: Las Reglas del Juego. Scrum.org
- Sebastian, D., & Nugraha, K. A. (2021). Academic Customer Service Chatbot Development using TelegramBot API. 2021 2nd International Conference on Innovative and Creative Information Technology (ICITech)
- Stonebraker y Rowe (1986) describen el diseño de POSTGRES en su informe presentado al Departamento de Ingeniería Eléctrica y Ciencias de la Computación de la Universidad de California, Berkeley.
- Suryawanshi, S., & Suryawanshi, P. ( Montréal). .pdf ). ARTIFICIAL INTELLIGENCE FOR LEGAL CHATBOT. International Journal of Research and Development in Applied Science and Engineering, -V I .p
- Vassilopoulos, A. ( information? (Master's ). Can AI chatbots improve access to legal thesis, HEC [https://biblos.hec.ca/biblio/memoires/vassilopoulos\\_athena\\_m](https://biblos.hec.ca/biblio/memoires/vassilopoulos_athena_m) Montréal). .pdf

## **ANEXOS**



**DIRECTIVA N° 011-2019-P-CSJSA-PJ**

**DIRECTIVA QUE REGULA EL USO DEL SISTEMA EXPERTO DE  
ATENCION DE CONSULTAS DE EXPEDIENTES JUDICIALES  
SECE\_SANTA EN LA CORTE SUPERIOR DE JUSTICIA DEL  
SANTA**



**OBJETIVO**

Definir un procedimiento para el uso del Sistema Experto de Atención de Consultas de Expedientes Judiciales en la atención de consultas sobre el estado de los procesos judiciales, en todas las especialidades.

**II. FINALIDAD**

Normar el procedimiento de gestión del servicio de atención de consultas de expedientes judiciales a través del Sistema Experto de Atención de Consultas de Expedientes Judiciales - SECE\_SANTA, ampliando nuestro servicio al usuario judicial y otorgándole una atención celer, oportuna y en tiempo real, las 24 horas del día y los 365 días del año, generando confianza, bienestar y transparencia en la tramitación de sus procesos judiciales.



**III. ALCANCE**

Los lineamientos contenidos en la presente directiva, son de obligatorio cumplimiento en la Administración y Órganos Jurisdiccionales de todas las especialidades, de la Corte Superior de Justicia del Santa; así como de todos los usuarios judiciales registrados.



**IV. BASE LEGAL**

- 4.1. Constitución Política del Perú.
- 4.2. Decreto Supremo N° 017-93-JUS, Texto Único Ordenado de la Ley Orgánica del Poder Judicial.
- 4.3. Decreto Supremo N° 043-2003-PCM, que aprueba el Texto Único Ordenado de la Ley N° 27806 Ley de Transparencia y Acceso a la Información Pública.
- 4.4. Decreto Supremo N° 072-2003-PCM, que aprueba el Reglamento de Ley de Transparencia y Acceso a la Información Pública.
- 4.5. Ley N° 27444 Ley del Procedimiento Administrativo General.
- 4.6. Resolución Administrativa N°010-2004-CE-PJ, que aprueba el Reglamento Interno de Trabajo del Poder Judicial.





4.7. Resolución Administrativa N° 180-2004-CE-PJ, de fecha 06/10/2004, que aprueba la Directiva N° 007-2004-GG-PJ “Normas para efectuar Consulta de Expedientes Judiciales, a través de la Página Web del Poder Judicial”.



8. Resolución Administrativa N° 184-2007-CE-PJ, de fecha 07/08/2007, que modifica la Directiva N° 007-2004-GG-PJ “Normas para efectuar Consulta de Expedientes Judiciales, a través de la Página Web del Poder Judicial”.



4.9. Resolución Administrativa N° 623-2013-P-CSJSA/PJ, de fecha 03/06/2013, se aprueba el Manual de organización y funciones (MOF) de la Unidad Ejecutora “Corte Superior de Justicia del Santa”.

4.10. Resolución Administrativa N°090-2018-CE-PJ, de fecha 14/03/2018, que aprueba el Reglamento de Organización y Funciones de las Cortes Superiores de Justicia que operan como Unidades Ejecutoras.



### V. VIGENCIA

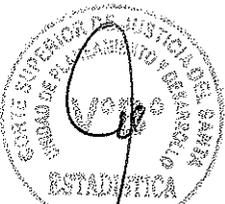
La presente Directiva tiene vigencia a partir del día siguiente de su aprobación por parte de la Presidencia de esta Corte Superior de Justicia.

### VI. DISPOSICIONES GENERALES

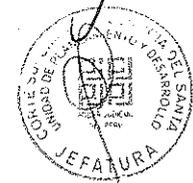


6.1. Es política de la Corte Superior de Justicia del Santa y del Poder Judicial como institución, brindar a los justiciables las máximas posibilidades para la obtención de información de sus procesos judiciales en curso, en la etapa que considere conveniente.

6.2. La Corte Superior de Justicia del Santa cuenta con los soportes tecnológicos necesarios que permiten brindar el servicio de consulta de Expedientes Judiciales a través del “**Sistema Experto de Atención de Consultas de Expedientes Judiciales**”, denominado para efectos de la presente Directiva: **SECE\_SANTA**.



6.3. El **SECE\_SANTA** funciona a través del aplicativo multiplataforma denominado “**TELEGRAM**”, el cual permitirá a los usuarios judiciales debidamente registrados (fiscales, procuradores, defensores públicos, abogados privados y partes del proceso), acceder a la información de sus procesos judiciales tramitados en la Corte Superior de Justicia del Santa, desde cualquier parte del mundo, de manera celer, oportuna y en tiempo real las 24 horas del día y los 365 días del año.



6.4. La información que se brindará corresponde a la información contenida en el Sistema Integrado Judicial – SIJ, referente a: actas de audiencia, resoluciones finales, depósitos judiciales y otros documentos digitalizados según correspondan, que se requieran de todas las especialidades de la Corte Superior de Justicia del Santa (Civil, Laboral, Penal<sup>1</sup> y Familia).

6.5. Solo se remitirá información de expedientes que se muestran en el Sistema Integrado Judicial; no se brindará información de procesos judiciales que se encuentren reservados o contemplen alguna indisponibilidad por prohibiciones o medidas de seguridad establecidas por el Magistrado.

6.6. Por seguridad de la información, toda consulta quedará registrada y almacenada de forma diaria en diferentes soportes informáticos, garantizando su preservación y recuperación a posterior.

## VII. DISPOSICIONES ESPECÍFICAS

7.1. Para acceder al **SECE\_SANTA**, los defensores públicos, abogados privados apersonados y partes del procesos judicial, deberán registrarse de manera personal en la Oficina de Informática del Módulo Procesal Penal, procediendo a llenar debidamente el formato del **Anexo N° 01**, denominado “**Registro de Usuario SECE\_SANTA**”, en donde consignará sus datos personales correspondientes, dentro de los cuales deberá figurar el número de celular desde donde accederá al aplicativo, además como requisito indispensable será contar con casilla electrónica<sup>2</sup>; y el **Anexo N° 02 Declaración Jurada de Apersonamiento**”.

Los fiscales y procuradores públicos, deberán registrarse de manera personal en la Oficina de Informática del Módulo Procesal Penal, procediendo a llenar solamente el formato del **Anexo N° 01**, denominado “**Registro de Usuario SECE\_SANTA**”, y aceptar el “Acuerdo de Confidencialidad de la información”<sup>3</sup>, contenida en este.

<sup>1</sup> Para recibir información de esta especialidad, los abogados libres deberán estar apersonados al proceso.

<sup>2</sup> Este requisito es indispensable para los abogados privados, así como para los fiscales, procuradores y defensores públicos.

<sup>3</sup> Este acuerdo también deberán firmarlo los abogados privados y las partes del proceso.



7.2. Una vez registrados, los usuarios deberán ingresar a **PLAY STORE**, y descargar e instalar el aplicativo “**TELEGRAM**”, mediante el cual podrán acceder al servicio que ofrece el **SECE\_SANTA**.



3. Al ingresar por primera vez al **SECE\_SANTA**, el usuario registrado deberá identificarse, compartiendo para ello su número de celular con el aplicativo.

7.4. Una vez confirmada su identidad, el usuario deberá seguir los pasos e ingresar los datos que indica el aplicativo para solicitar su información, referente a:

- Tipo de búsqueda (por N° de expediente o DNI).
- La especialidad a buscar (Penal, Familia, Civil o Laboral)
- Información que necesita consultar y descargar (Acta de Audiencia, Resolución Final, Documentos Digitalizados o Depósitos Judiciales).
- Fecha del documento a consultar (fecha específica o último documento).
- Lugar de recepción de la información en formato pdf (correo electrónico o celular).



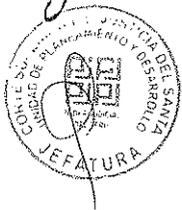
7.5. No se brindará información reservada o que contemple alguna prohibición por medidas de seguridad establecidas para su cumplimiento, tales como la emisión de una medida cautelar no ejecutada, levantamiento del secreto de las comunicaciones, entre otros.



7.6. Los usuarios de tipo fiscal y procurador público, tendrán acceso a la información de cualquier expediente judicial incluyendo los de tipo penal. Por el contrario, los usuarios tipo defensores públicos, abogados privados y partes del proceso, sólo tendrán acceso a la información de los expedientes en los cuales se encuentran debidamente apersonados y acreditados o de los cuales forman parte del proceso judicial, respectivamente.



7.7. Una vez finalizada su atención, se realizará al usuario la “**ENCUESTA DE SATISFACION EN LA ATENCION RECIBIDA**”, la cual reflejará en grados de Muy Malo, Malo, Regular, Bueno y Muy Bueno, el nivel de satisfacción de los usuarios, respecto al servicio brindado por el orientador.



- 7.8. El usuario bajo ningún caso podrá efectuar operaciones distintas a la naturaleza del servicio de consultas de expedientes judiciales.

### VIII. DISPOSICIONES COMPLEMENTARIAS

- 8.1. La Unidad de Servicios Judiciales a través de los Centros de Distribución General y Mesa de Partes, deben asegurar el correcto registro de la información (sobre todo el DNI o documento de identidad correspondiente) al momento de la presentación de la documentación por parte de los usuarios.



- 8.2. La Oficina de Informática del Módulo Penal será la responsable de velar por el óptimo funcionamiento del servicio de consulta de expedientes judiciales penales a través del aplicativo web, dando cuenta de manera mensual de su aplicación al Área de Informática de esta Corte Superior de Justicia.



- 8.3. El Área de Informática de la Corte Superior de Justicia del Santa deberá monitorear de manera permanente la ejecución del presente proyecto y brindar el soporte necesario a fin de garantizar la intangibilidad de la información contenida en la Base de Datos del Sistema de Expedientes Judiciales – SIJ, para lo cual, extraerá información necesaria y generará una Base de Datos Espejo; en el supuesto de que algún usuario pretenda manipular la información más allá de lo pactado en el servicio.



- 8.4. La Gerencia de Administración Distrital a través de la Administración del Módulo Penal, supervisarán lo dispuesto en la presente Directiva.

- 8.5. El Administrador del Módulo Penal deberá remitir, de manera mensual, información estadística relacionada a las atenciones realizadas a través del **SECE\_SANTA**, incluyendo el número de consultas por usuario y por tipo de usuario, ahorro de papel y el resultado de las encuestas; dicha información será enviada a través del formato correspondiente a la Gerencia de Administración Distrital y vía correo electrónico a la Presidencia de esta Corte Superior de Justicia, a la Gerencia de Administración Distrital y a la Jefatura de la Unidad de Servicios Judiciales.



- 8.6. El Responsable de la Oficina de Imagen Institucional de esta Corte Superior de Justicia se encargará de la difusión de la presente Directiva y del nuevo servicio, a través de los diversos medios de comunicación como: web institucional, correo electrónico institucional, radio, ferias informativas; así como la elaboración de los banner que serán colocados en lugares visibles para conocimiento de los interesados.





PODER JUDICIAL DEL PERU

DIRECTIVA QUE REGULA EL USO DEL SISTEMA EXPERTO DE ATENCION DE CONSULTAS DE EXPEDIENTES JUDICIALES SECE – SANTA EN LA CORTE SUPERIOR DE JUSTICIA DEL SANTA

D-011-2019-P-CSJSA-PJ-V01

8.7. Esta buena práctica podrá ser replicada por otras Cortes Superiores de Justicia del país.

IX. ANEXOS:

- Anexo N° 01: Registro de Usuario de SECE\_SANTA



PODER JUDICIAL DEL PERÚ  
CORTE SUPERIOR DE JUSTICIA DEL SANTA

CONSTANCIA

CUENTA DE ACCESO AL SISTEMA

Por medio de la presente se le hace entrega de la Cuenta y Clave de acceso al Sistema de Principio de Oportunidad.

DAMIÁN ACUÑA ZULMA RUTH

D.N.I.: Casilla Electrónica:

Quien tendrá el siguiente perfil de seguridad:

Perfil Asignado: Abogados (a)

Y a quien se le esta asignando la siguiente cuenta:

Usuario : ZDAMIAN

Clave: 123456



Haciendose responsable de su privacidad y resguardo, así como por cualquier uso indebido que se haga con las mismas.

Por lo tanto se le recuerda que ES OBLIGATORIO efectuar el cambio de su clave quedando bajo su responsabilidad el no hacerlo.

El Usuario se compromete a utilizar la información recibida exclusivamente para el desarrollo de las funciones propias de sus respectivos ámbitos de competencia. En ese sentido, están obligados a no comercializar, distribuir, reproducir, ceder a favor de terceros, y en general, utilizar para fines distintos tal información, en forma total o parcial, a través de medios propios o ajenos, magnéticos, escritos o de cualquier tipo.

SI

NO

DAMIÁN ACUÑA ZULMA RUTH

40583697

PODER JUDICIAL  
Corte Superior de Justicia del Santa

*José Suarez Aguales*  
INFORMÁTICO

NUOVO CODIGO PROCESAL PENAL



PODER JUDICIAL  
DEL PERU

DIRECTIVA QUE REGULA EL USO DEL SISTEMA EXPERTO DE ATENCION DE CONSULTAS DE  
EXPEDIENTES JUDICIALES SECE – SANTA EN LA CORTE SUPERIOR DE JUSTICIA DEL SANTA

D-011-2019-P-CSJSA-PJ-V01

ANEXO N° 02



PODER JUDICIAL  
DEL PERÚ

INFORMATICA



**NOPP**  
DEL SANTA

DECLARACION JURADA DE APERSONAMIENTO



\_\_\_\_\_ identificado con DNI N° \_\_\_\_\_ celular N° \_\_\_\_\_  
\_\_\_\_\_ correo electrónico: \_\_\_\_\_ casilla electrónica N°: \_\_\_\_\_ **DECLARO BAJO JURAMENTO**, estar apersonado y tener la autorización de mi patrocinados como abogado(a) en los siguientes expediente(s)

Nº EXPEDIENTE	FECHA	FIRMA	HUELLA

Para que se me registre como usuario en el sistema SECE implementado por el Modulo Penal de la Corte Superior de Justicia del Santa.

De ser el caso de que usted este brindando una información falsa y no se encuentre apersonado en el proceso que está solicitando información se le remitirá copias a su Colegio de Abogados o a la Procuraduría Pública, según corresponda, para que tome las medidas correctivas necesarias.

Chimbote, de del 20 .

