



unl

Universidad
Nacional
de Loja

Universidad Nacional de Loja

Facultad de la Energía, las Industrias y los Recursos Naturales

No Renovables

Carrera de Ingeniería en Sistemas

Sistematización de Fichas Médicas Ocupacionales y Test Psicológicos

Trabajo de Titulación
previo a la obtención
del título de Ingeniero
en Sistemas

AUTORES:

Yannela Mishelle Castro Valarezo

Josué Fernando Ortega Jaramillo

DIRECTOR:

Ing. Valeria del Rosario Herrera Salazar, Mg. Sc.

Loja – Ecuador

2022

Dedicatoria

Agradecimiento

Índice de Contenidos

1. Título	9
2. Resumen	10
2.1. Abstract.....	10
3. Introducción.....	11
4. Marco Teórico.....	12
4.1. Antecedentes	12
4.1.1. Salud Ocupacional.....	12
4.1.2. Ficha Ocupacional	12
4.1.3. CIE-10.....	12
4.2. Sistematización de la Información	13
4.2.1. Principios básicos de un Sistema de información.....	13
4.3. Tecnologías para Ingeniería de Software	14
4.3.1. Arquitectura 4 + 1	14
4.3.2. Framework Django	15
4.3.3. Lenguaje De Programación.....	15
4.3.4. Base De Datos.....	16
4.4. Metodologías ágiles para desarrollo de Software	16
4.4.1. Marco de trabajo SCRUM	16
4.4.2 Metodología Ágil eXtreme Programming	17
4.4.3 Metodología Híbrida SXP	17
4.5. Trabajos Relacionados	18
4.5.1. Primer Trabajo Relacionado.....	18
4.5.2. Segundo Trabajo Relacionado	18
4.5.3. Tercer Trabajo Relacionado	19
4.5.4. Cuarto Trabajo Relacionado.....	19
4.5.5. Quinto Trabajo Relacionado.....	20
5. Metodología	21
5.1. Área de estudio	21
5.2. Procedimiento.....	22
6. Resultados	25
6.1. OBJETIVO 1: Analizar la situación actual del manejo de la información dentro de la Subdirección de Seguridad y Salud Ocupacional.....	25
6.1.1. Análisis de los requerimientos con los miembros de la Subdirección de Seguridad y Salud Ocupacional.	25
6.1.2. Definir el procedimiento para la recolección de información.	27
6.2. OBJETIVO 2: Desarrollar un software que facilite la sistematización de fichas médicas ocupacionales y test psicológicos.	29

6.2.1. Análisis y diseño de diagramas e historias de usuario, a partir de los requerimientos obtenidos.	29
6.2.2. Desarrollo e implementación de las funciones del software, en base a las tecnologías y metodologías establecidas por el departamento de TI.	40
6.3. OBJETIVO 3: Evaluar el funcionamiento del software a través de pruebas aplicadas por la Dirección de Tecnologías de Información.....	47
6.3.1. Elección de herramientas para la aplicación de pruebas del software desarrollado.	47
6.3.2. Realizar pruebas conjuntamente con los usuarios que serán beneficiados con el software implementado.....	50
7. Discusión	51
Objetivo 1: Analizar la situación actual del manejo de la información dentro de la Subdirección de Seguridad y Salud Ocupacional.....	51
Objetivo 2: Desarrollar un software que facilite la sistematización de fichas médicas ocupacionales y test psicológicos.	51
Objetivo 3: Evaluar el funcionamiento del software a través de pruebas aplicadas por la Dirección de Tecnologías de Información.....	52
8. Conclusiones	54
9. Recomendaciones	55
10. Bibliografía.....	56
11. Anexos.....	59

Índice de Tablas

Tabla 1. Usuario Administrador.	25
Tabla 2. Usuario Médico.	25
Tabla 3. Requerimientos Funcionales.	26
Tabla 4. Requerimientos No Funcionales.	26
Tabla 5. Información recolectada con entrevistas.	27
Tabla 6. Planificación de Historias de Usuario para cada iteración.	29
Tabla 7. Arquitectura 4 + 1.	30
Tabla 8. Valoración económica Recursos Humanos	52
Tabla 9. Valoración económica Recursos y Servicios	53
Tabla 10. Costo Total del Trabajo de Titulación.	53
Tabla 11. Referencias Arquitectura De Software	68
Tabla 12. Vistas del Modelo 4 + 1	69
Tabla 13. Descripción del Caso de Uso Iniciar Sesión	71
Tabla 14. Descripción del Caso de Uso Obtener datos SIAAF	71
Tabla 15. Descripción del Caso de Uso Registrar Ficha Médica Inicial	72
Tabla 16. Descripción del Caso de Uso Registrar Ficha Médica Periódica	73
Tabla 17. Descripción del Caso de Uso Actualizar Ficha Médica Periódica	74
Tabla 18. Descripción del Caso de Uso Registrar Hoja de Evolución	75
Tabla 19. Descripción del Caso de Uso Actualizar Hoja de Evolución.	76
Tabla 20. Descripción del Caso de Uso Registrar Ficha Médica de Reintegro.	77
Tabla 21. Descripción del Caso de Uso Registrar Ficha Médica de Salida	77
Tabla 22. Descripción del Caso de Uso Obtener Código CIE	78
Tabla 23. Descripción del Caso de Uso Generar Certificado de Aptitud Médica	79
Tabla 24. Descripción del Caso de Uso Receptar Certificado	80
Tabla 25. Historias de Usuario para la Iteración 1	108
Tabla 26. Historias de Usuario para la Iteración 2	112
Tabla 27. Historias de Usuario para la Iteración 3	118
Tabla 28. Clase Ficha_medica_inicio	124
Tabla 29. Clase Ficha_medica_periodica	125
Tabla 30. Clase Ficha_medica_reintegro.	126
Tabla 31. Clase Ficha_medica_salida.	126
Tabla 32. Clase Certificado_medico	127

Índice de Figuras

Figura 1. Ubicación de Bienestar Universitario UNL, en donde se encuentra la Subdirección de Seguridad y Salud Ocupacional	21
Figura 2. Diagrama de Caso de Uso General	31
Figura 3. Modelo Conceptual	32
Figura 4. Diagrama de Clases	33
Figura 5. Diagrama de Actividad Obtener Datos SIAAF	34
Figura 6. Diagrama de Actividad Registrar Ficha Inicial	35
Figura 7. Diagrama de Actividad Registrar Hoja de Evolución	36
Figura 8. Diagrama de Actividad Obtener Código CIE.....	36
Figura 9. Diagrama de Actividad Generar Certificado de Aptitud Médica	37
Figura 10. Diagrama de Componentes.....	38
Figura 11. Diagrama de Despliegue	39
Figura 12. Creación del Módulo-StartApp	40
Figura 13. Función Crear Ficha Médica al Paciente.	41
Figura 14. Crear Ficha Médica de Inicio.....	42
Figura 15. Función Crear Ficha Médica Periódica.	43
Figura 16. Función Crear Hoja de Evolución.....	44
Figura 17. Función Crear Ficha Médica de Reintegro.	45
Figura 18. Función Crear Ficha Médica de Salida	46
Figura 19. Función Editar Ficha Médica.....	46
Figura 20. Crear Certificado Médico	47
Figura 21. Resultado test_crear_ficha_medica().	48
Figura 22. Resultado test_crear_ficha_medica_inicio().	48
Figura 23. Resultado test_crear_ficha_medica_reintegro().	48
Figura 24. Resultado test_crear_ficha_medica_salida().	49
Figura 25. Resultado test_crear_ficha_medica_periodica().	49
Figura 26. Resultado test_crear_hoja_evolucion().	49
Figura 27. Resultado test_crear_certificado_medico().	49
Figura 28. Diagrama de Caso de Uso General	70
Figura 29. Caso de Uso Gestionar Cuenta	70
Figura 30. Caso de Uso Gestionar Ficha Médica	72
Figura 31. Caso de Uso Gestionar Certificado.....	79
Figura 32. Modelo Conceptual.....	81
Figura 33. Diagrama de Clases	82
Figura 34. Diagrama de Actividad Iniciar Sesión	83
Figura 35. Diagrama de Actividad Obtener Datos SIAAF	84
Figura 36. Diagrama de Actividad Registrar Fichas Médicas	85
Figura 37. Diagrama de Actividad Actualizar Ficha Médica Periódica	86
Figura 38. Diagrama de Actividad Registrar Hoja de Evolución	87
Figura 39. Diagrama de Actividad Actualizar Hoja de Evolución.....	88
Figura 40. Diagrama de Actividad Obtener Código CIE	89
Figura 41. Diagrama de Actividad Generar Certificado de Aptitud Médica	90
Figura 42. Diagrama de Actividad Receptar Certificado	91
Figura 43. Diagrama de Componentes.....	92
Figura 44. Diagrama de Despliegue	93
Figura 45. Diagrama de Clases - Iteración 1	109
Figura 46. Función Crear Ficha Médica al Paciente.	110
Figura 47. Crear Ficha Médica de Inicio.....	111
Figura 48. Test Crear Ficha Médica	111

Figura 49. Test Crear Ficha Médica de Inicio.....	112
Figura 50. Diagrama de Clases - Iteración 2.....	114
Figura 51. Función Crear Ficha Médica Periódica.	115
Figura 52. Función Crear Hoja de Evolución.....	115
Figura 53. Función Crear Ficha Médica de Reintegro.	116
Figura 54. Función Editar Ficha Médica.....	116
Figura 55. Test Crear Ficha Médica de Reintegro	117
Figura 56. Test Crear Ficha Médica Periódica	117
Figura 57. Test Crear Hoja de Evolución.....	117
Figura 58. Diagrama de Clases - Iteración 3.....	120
Figura 59. Función Crear Ficha Médica de Salida	121
Figura 60. Función Crear Certificado Médico.....	121
Figura 61. Test Crear Ficha Médica de Salida	122
Figura 62. Test Crear Certificado Médico.....	122

Índice de Anexos

Anexo I. Entrevista 1	59
Anexo II. Entrevista 2	61
Anexo III. Entrevista 3	63
Anexo IV. Documento de Especificación de Requisitos de Software	66
Anexo V. Documento de Arquitectura de Software	67
Anexo VI. Formatos de Fichas Médicas Ocupacionales	94
Anexo VII. Formato Certificado Aptitud Médica	103
Anexo VIII. Hoja de Evolución	105
Anexo IX. Receta Médica	106
Anexo X. Desarrollo de la Metodología SXP	107
Anexo XI. Clases y Atributos	123

1. Título

**Sistematización de Fichas Médicas Ocupacionales y Test
Psicológicos**

**Systematisation of Occupational Medical Records and
Psychological Tests**

2. Resumen

2.1. Abstract

3. Introducción

4. Marco Teórico

4.1. Antecedentes

4.1.1. Salud Ocupacional

A nivel mundial, la Salud Ocupacional es considerada como un pilar fundamental en el desarrollo de un país, sus acciones están dirigidas a la promoción y protección de la salud de los trabajadores y la prevención de accidentes de trabajo y enfermedades ocupacionales causadas por las condiciones de trabajo y riesgos ocupacionales. La misión de los servicios de salud en el trabajo debe tener como meta el poder aportar a los trabajadores las herramientas y los recursos necesarios para crear, mantener y mejorar el ambiente óptimo en la relación persona-trabajo, garantizando un nivel óptimo de bienestar y salud integral del individuo, dentro del contexto social en el cual se desenvuelve. Todo servicio básico de salud ocupacional debe ser accesible, estar a disposición de cualquier trabajador o empresa, ser capaz de responder a las solicitudes del interesado, tener capacidad de control de calidad y retroalimentación del proceso y resultado del servicio prestado [1].

4.1.2. Ficha Ocupacional

Es un análisis de la carga de trabajo mediante la elaboración de Fichas Ocupacionales o herramientas de similares características que permitiesen disponer de información suficiente y ordenada sobre las cargas y requerimientos de las diferentes ocupaciones y grupos ocupacionales, al menos de las más frecuentes [2].

Según [3] , la información ocupacional orienta la valoración médica en los distintos momentos de evaluación de caso, además, determina la capacidad del trabajador para afrontar los requerimientos de la ocupación y su posibilidad de retorno al puesto de trabajo.

4.1.2.1. Historia Clínica Electrónica

El artículo [4] indica que es la que reside en un sistema electrónico diseñando para recolectar, almacenar, manipular y dar soporte a los usuarios en cuanto a proveer accesibilidad a datos seguros y completos, brindando información clínica importante para el cuidado de los pacientes. Además, es importante entender que esto no es una simple digitalización del registro médico en papel, ya que implica la integración de múltiples sistemas existentes (o componentes) que brinden información en un repositorio clínico común y contemple además el resto de los procesos y actores de la organización.

4.1.3. CIE-10

La Clasificación Internacional de Enfermedades ha sido, durante varios años, el instrumento esencial que permitía la comparación nacional e internacional en el campo de la salud pública. Una clasificación de enfermedades es un complejo sistema de asignación de categorías (alfabéticas, numéricas o alfanuméricas) a enfermedades o situaciones relacionadas con la

enfermedad de acuerdo a una serie de criterios preestablecidos. La codificación médica según la CIE 10ma edición, consiste en la asignación de códigos estandarizados a informes médicos, en representación de diagnósticos y procedimientos, y su finalidad es la generación de estadísticas de morbilidad y mortalidad, es decir, permite el registro sistemático, el análisis, la interpretación y la comparación de los datos de mortalidad y morbilidad recogidos en diferentes países o áreas, y en diferentes épocas. Con el uso de una clasificación se consigue convertir los términos diagnósticos, los terapéuticos y otros términos relacionados con la salud en códigos preestablecidos, para favorecer su archivado y posterior recuperación para el análisis de la información. [5], [6]

El CIE-10 es una clasificación alfanumérica jerárquica que contiene entre 3 y 7 dígitos, y aporta una información clínica muy detallada. El número de diagnósticos va de 69.099, y el número de procedimientos es de 72.000. [6]

Por tanto, el CIE se ocupa a nivel internacional para fines estadísticos relacionados con morbilidad y mortalidad, los sistemas de reintegro y soportes de decisión automática en medicina.

4.2. Sistematización de la Información

Para el autor del libro [7], los sistemas de información (SI) basados en computadoras, recopilan, almacenan y transforman los datos en información, según las instrucciones que definen las personas mediante programas para la computadora. Es decir que, al menos, todo sistema cuenta con un proceso para generar información [8].

El autor [9], establece que la sistematización de información hace referencia al ordenamiento y clasificación de datos e informaciones, bajo determinados criterios y estructurando de manera precisa categorías, relaciones, posibilitando la constitución de bases de datos organizados, etc.

4.2.1. Principios básicos de un Sistema de información

En el libro [10], el autor señala algunos de los requisitos básicos que se deben considerar en un SI:

- Separación del contenido y la presentación
- Reutilización del contenido
- Utilización de metadatos
- Abundancia de vínculos
- Edición no técnica
- Facilidad de empleo y eficiencia

Sobre la gestión del contenido:

- Historial de las versiones y almacenamiento
- Flujo de trabajo
- Seguridad
- Integración con sistemas externos
- Generación de informes

Sobre la publicación:

- Empleo de hojas de estilo
- Soporte para múltiples formatos
- Indexación extensiva.

4.3. Tecnologías para Ingeniería de Software

La ingeniería de Software permite ampliar la visión del desarrollo del software, ya que contempla actividades de análisis y diseño previos, y de integración y verificación posteriores. Para el desarrollo de software es necesario conseguir una buena organización de la producción del mismo, a través del trabajo en equipo y empleando técnicas y herramientas apropiadas que agilicen cada una de las actividades [11].

4.3.1. Arquitectura 4 + 1

El artículo [12] señala que, el modelo 4+1 describe la arquitectura del software, la misma que usa cinco vistas concurrentes, dentro de las cuales existe un conjunto de intereses de diferentes stakeholders del sistema.

La arquitectura del software se trata de abstracciones, además, describe como un sistema es descompuesto en componentes, de estilos y estética, los cuales se comunican e interactúan entre sí. También tiene relación con el diseño y la implementación de la estructura de alto nivel del software. Los diseñadores construyen la arquitectura haciendo uso de varios elementos que, en su mayoría, satisfacen los requisitos de funcionalidad, así como también los requisitos no funcionales del sistema. [12], [13]

Las cinco vista que propone Kruchten [12] son:

- **Vista Lógica:** La arquitectura lógica se enfoca principalmente en los requisitos funcionales. Aquí se aplican principios de abstracción, encapsulamiento y herencia. Esta vista es representada a través de diagrama de clases, el cual muestra un conjunto de clases y sus relaciones lógicas.
- **Vista de Procesos:** Toma en cuenta algunos requisitos no funcionales, se tratan los aspectos de concurrencia y distribución, integridad del sistema, y tolerancia a fallos.

Esta arquitectura se describe en varios niveles de abstracción y sus diferentes intereses. La vista de procesos se puede representar mediante diagrama de actividades.

- **Vista de Desarrollo o Despliegue:** En esta vista se muestra el sistema desde la perspectiva de un programador y se ocupa de la gestión del software, es decir, se centra en la organización real de los módulos de software en el ambiente de desarrollo. Se pueden incluir los diagramas de componentes y de paquetes.
- **Vista Física:** La arquitectura física toma en cuenta los requisitos no funcionales del sistema, en esta vista se especifican varias configuraciones físicas como las conexiones entre los componentes que conforman la solución. Se puede representar a través del diagrama de despliegue.
- **Vista de Escenarios:** En esta vista los elementos de las cuatro vistas trabajan conjuntamente, usando instancias de casos de uso más generales, con lo que tendremos una trazabilidad de componentes, clases, equipos, paquetes, etc. Para su representación se puede incluir el diagrama de casos de uso.

4.3.2. Framework Django

Django se desarrolla con la necesidad de realizar aplicaciones en un plazo de días o incluso horas. Dos años después de empezar a crear un marco de trabajo para aplicaciones web se publicó como código abierto. Debido a que Django es un marco de aplicación web basado en Python, se necesita Python para utilizarlo; soporta oficialmente las bases de datos PostgreSQL, MySQL, Oracle y SQLite. Django presenta una estructura que no requiere la edición de muchos archivos, lo que a su vez ahorra tiempo [14].

Este framework se ha mantenido fiel a sus orígenes como un Modelo-Vista-Controlador (MVC) diseñado para operar con bases de datos relacionales. Sin embargo, sigue la mayoría de las tendencias de desarrollo web (a través de paquetes de terceros) para operar junto a tecnologías como las bases de datos no relacionales (NoSQL). Esto hace que Django sea ahora el marco de desarrollo web elegido por una amplia gama de organizaciones [15].

4.3.3. Lenguaje De Programación

- **Python**

Es un lenguaje de programación que a través del tiempo se ha adaptado por dos razones principales: la primera es que es un lenguaje de alto nivel, muy fácil de usar, y la segunda, que es de código libre, la ventaja de usar Python es que es independiente de la plataforma que se utilice, ya sea Windows, Mac o UNIX. [16]

Python ha llegado a ser un lenguaje de propósito general y ha ganado popularidad en varios ámbitos como el desarrollo rápido de aplicaciones web, administración de sistemas, ciencia de datos, computación científica, inteligencia artificial, internet de las cosas, etc. [17]

4.3.4. Base De Datos

- **PostgreSQL**

Es un sistema de bases de datos relacional, admite un modelo de datos que consta de un conjunto de relaciones, sustituyendo modelos previos por su “simplicidad espartana”, esto hace más difícil en la implementación de ciertas aplicaciones que se realizan [18]. Postgres permite a los usuarios ampliar fácilmente al sistema en los objetos de código abierto en el desarrollo validando mayor confiabilidad, robustez en sus funciones y en el rendimiento [19].

Además, PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos para garantizar la estabilidad del sistema, es decir que, un fallo en uno de los procesos no afectará a los demás y el sistema continuará en funcionamiento [20].

4.4. Metodologías ágiles para desarrollo de Software

Las metodologías ágiles han revolucionado la forma en que se produce software, ya que le dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas. Esto ha demostrado su efectividad en proyectos con requisitos que van cambiando a lo largo de su desarrollo, y cuando se exige reducir drásticamente los tiempos sigue manteniendo una alta calidad. “Las metodologías ágiles proporcionan una serie de pautas y principios junto a técnicas pragmáticas que hacen que la entrega del proyecto sea menos complicada y más satisfactoria tanto para los clientes como para los equipos de trabajo” [21].

4.4.1. Marco de trabajo SCRUM

Se compone de una serie de reglas, que definen roles que integran los equipos, artefactos necesarios para los procesos, bloques de tiempo preestablecidos y ceremonias que deben respetarse. Se caracteriza por adoptar una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del producto, basar la calidad del resultado más en el conocimiento tácito de las personas en equipos autoorganizados, que, en la calidad de los procesos empleados, y el solapamiento de las diferentes fases del desarrollo, en lugar de realizarlas una tras otra en un ciclo secuencial o de cascada [22], [23].

Dentro del marco de trabajo de Scrum se definen tres roles [24]:

- Scrum Master, asegura los procesos
- Dueño de Producto, maximiza el valor del producto
- Equipo, realiza el trabajo.

La definición de bloques de tiempo iterativos se basa en seis reuniones para cumplir los objetivos [22], [24]:

- Reunión de Planificación de la Entrega
- Reunión de Planificación del Sprint
- El Sprint
- Reunión diaria
- Reunión de revisión
- Reunión de Retrospectiva

4.4.2 Metodología Ágil eXtreme Programming

Extreme Programming (XP) propone una metodología basada esencialmente en la simplicidad y agilidad, además enfatiza el trabajo en equipo, los gerentes, los clientes y los desarrolladores son todos socios iguales [25]. Al ser una metodología de desarrollo rápido y ágil maneja un conjunto de prácticas técnicas que pretenden enfatizar los efectos positivos de en un proyecto de desarrollo de Software, si se las aplica simultáneamente [24].

En XP la organización del equipo de trabajos se divide en los siguientes roles [26]:

- Programador, desarrolla el código del sistema de acuerdo a los requerimientos.
- Cliente, define los requerimientos y las pruebas funcionales.
- Tester, ejecuta todas las pruebas funcionales.
- Tracker, hace el seguimiento del proceso global del proyecto.

Esta metodología se divide en cuatro fases:

- Planificación
- Diseño
- Codificación
- Pruebas

4.4.3 Metodología Híbrida SXP

Esta metodología parte de la combinación de SCRUM y XP, se basa fundamentalmente en la implementación de procesos ágiles, promueve la creatividad, acepta el cambio y exige un alto nivel de responsabilidad por parte de los miembros del equipo de desarrollo, además trabaja en 3 fases fundamentales [27]:

- Planificación-Definición, se elabora la ficha de costo del proyecto con la planificación inicial, se establece la visión del proyecto.

- Desarrollo, se implementa el producto hasta que cumpla con todas las funcionalidades, realizando pruebas unitarias, y al finalizar cada iteración se realizan pruebas al sistema.
- Entrega y mantenimiento, durante este periodo se da soporte al cliente según lo pactado inicialmente.

4.5. Trabajos Relacionados

Para el desarrollo del TT se encontraron diferentes temas y proyectos que están bastante relacionados con el objeto de estudio, dentro de ellos se listan los siguientes.

4.5.1. Primer Trabajo Relacionado

Desarrollo de un Sub-Módulo de Historia Clínica Laboral dentro del ERP para la Empresa SRF Consultores

El proyecto de tesis propone crear el módulo de historia clínica laboral para la empresa SRF Consultores, en donde se podrá registrar cada consulta del trabajador con el médico interno de la empresa, realizar consultas informativas y seguimientos respecto al estado de salud ocupacional de los empleados, todo esto integrando información relevante de otros módulos como recursos humanos y HSEQ. En el momento de implementar un módulo de historia clínica en el ERP de la empresa SRF Consultores se lograrían cambios en los procesos actuales de atención médica, implementando mejoras de innovación con un producto que integraría áreas como recursos humanos y HSE, ofreciendo un nuevo y más amplio manejo de la información referente al estado de salud de los trabajadores, impactando áreas de gestión como SGSST. Por lo tanto, tiene como objetivos a cumplir, el lograr una transformación en el proceso actual de manejo de información de la historia clínica laboral, entregando información relevante para la prevención de enfermedades laborales, aplicar las técnicas y procedimientos adecuados para el manejo de la historia clínica laboral según el marco legal, construir una solución que contemple el diseño, las mejores prácticas y patrones relevantes basado en la arquitectura de software por capas, seguir lineamientos de calidad para entornos de desarrollo de pruebas, validar la funcionalidad del submódulo de historia clínica en ambientes de pruebas de SRF Consultores.[28]

4.5.2. Segundo Trabajo Relacionado

Análisis y Diseño de Prototipo de Software para la Automatización de Historias Clínicas del Policlínico UDEP

Este trabajo trata de buscar una alternativa de solución al diseñar un prototipo de software para Policlínico UDEP que pueda permitir una gestión automática de las historias clínicas de este centro médico para mejorar el servicio de atención a sus pacientes y que brinde las funciones que satisfagan sus necesidades; tiene como objetivo el Análisis y diseño de un

prototipo de un software que permita tener una aplicación compatible con el sistema operativo (Windows) para la creación de historias clínicas de manera digital, para facilitarle la gestión de información al personal administrativo y médico. Además, indica que los beneficios que se podrían obtener con este sistema son muchos, como: eficiencia, eficacia e información a disposición inmediata de todas las áreas involucradas; además que permitirá reducir el número de reclamos por parte de los pacientes, aumentando así sus niveles de satisfacción por la atención recibida y, pensando a gran escala, esta idea podría aplicarse a otras entidades de salud. [29]

4.5.3. Tercer Trabajo Relacionado

Automatización del Proceso de Salud Ocupacional del Distrito 17D04 “Puengasí a Itchimbía”

En este trabajo de tesis se plantea como problemática que, el área de salud ocupacional del Distrito 17D04 Puengasí a Itchimbía, carece de una historia clínica ocupacional en forma digital de fácil acceso a la nube que le permita al experto de salud ocupacional acceder a la información del personal del Distrito desde cualquier unidad de salud que visite para requerir atención, a partir de esto propone una solución informática que facilitará el seguimiento de salud del personal en el tiempo, tomar decisiones para mejorar su ambiente de trabajo, así como prevenir enfermedades o dolencias que puede traer un trabajador al momento de ingresar al Distrito o quizá llevar un control de principales áreas que necesitan modificaciones ya que en estas el personal contrae enfermedades o lesiones continuamente. Por ello, tiene como objetivo: Analizar, diseñar, desarrollar e implementar un sistema para la automatización del proceso de salud ocupacional del Distrito 17D04 Puengasí a Itchimbía. [30]

4.5.4. Cuarto Trabajo Relacionado

Sistema de fichas de control médico ocupacional y periódico para el análisis de riesgo laboral acumulado de los docentes y servidores de la Universidad Técnica de Cotopaxi.

El presente proyecto trata sobre el desarrollo e implementación de un sistema informático, que permite sistematizar los procesos de valoración médica que en la actualidad se lleva a cabo de forma electrónica en el Departamento Médico Ocupacional de la Universidad Técnica de Cotopaxi. Este sistema está dirigido al Dr. Richard Pérez, Médico Ocupacional de la institución, permitiéndole optimizar varios de sus procesos de análisis y predicción de riesgos laborales acumulados, facilita la comunicación y colaboración con los pacientes, desarrollado mediante el lenguaje de programación C#, con un motor de base de datos SQL SERVER 2012, tiene un estándar muy utilizado para el análisis, como resultado, se tiene un software de fichas médicas de control periódico para el análisis y predicción de riesgo laboral acumulado compuesto por los siguientes módulos: Perfiles de Usuario para la autenticación,

Gestionar Pacientes, Registro de Datos Para el Análisis de Riesgo Laboral Acumulado, Registrar Examen Físico, Registrar Hábitos, Generar Preguntas ASSIST, Gestionar Tablas de Mantenimiento de Datos, Consultar Resumen Clínico, Generar Certificado Médico. El sistema actualmente está instalado y se encuentra funcionando eficazmente, razón por la que se cuenta con el aval de implementación respectivo. [31]

4.5.5. Quinto Trabajo Relacionado

Implementación del Sistema Web SISEMO, para Optimizar el proceso de Evaluaciones Medicas Ocupacionales de La Clínica Servisalud-Serviperu, Lima 2018.

El autor de la presente tesis propone como objetivo “Implementar un sistema informático web de Salud Ocupacional, para optimizar el proceso de evaluaciones médicas ocupacionales de la Clínica Servisalud”. La implementación del sistema web se acredita por las varias razones, una de ellas es mejorar la búsqueda de las Historias clínicas, esto permitirá la gestión de la información contenida en la historia clínica logrando emitir informes en tiempo real, así mismo el acceso a la información de la Evaluación medicas Ocupacionales en tiempo real, permitiendo acceder desde fuera de la clínica con sólo tener acceso a Internet. A partir de esto obtuvieron los siguientes resultados: El sistema SSIEMO disminuye los tiempos en los registro de las evaluaciones médicas, desde la apertura de la Historia Clínica hasta la entrega del Certificado de Aptitud a la Empresa e Informe Médico al Trabajador, debido que se pusieron campos predefinidos que ayudan al personal de salud al seleccionar solo el predefinido, reducción cantidad horas/hombre empleadas para la búsqueda, proceso de atención y reportes de la atención médica del paciente, Reducción del espacio físico empleado para almacenar las historias clínicas de los pacientes, entre otros. [1]

5. Metodología

Para un lograr correcto desarrollo del TT se hizo uso de métodos y técnicas útiles para llevar una buena organización de las actividades que se realizaron en cada fase del proyecto. La metodología para desarrollo de software empleada en el proyecto fue la combinación de Scrum con XP, obteniendo así la metodología híbrida SXP, en la que se dividieron las actividades según las 3 fases establecidas en dicha metodología, complementando cada una de ellas con entrevistas y el método de experimentación, que fueron fundamentales para determinar si la implementación solucionaría el problema dentro de la unidad y para la culminación exitosa del TT.

5.1. Área de estudio

El presente TT se realizó en la Subdirección de Seguridad y Salud Ocupacional de la Universidad Nacional de Loja, en colaboración con el subdirector, el Licenciado Diego Falconí Espinosa y demás personal de dicha unidad, misma que se encuentra en el Departamento de Bienestar Universitario, siendo esta unidad elegida debido a que no cuenta con un software que automatice el proceso de gestión de las fichas médicas ocupacionales de docentes, empleados y trabajadores; esto impide que se lleve de manera rápida una revisión ya que toda la información se encuentra en documentos físicos y existe una gran cantidad de la misma, además, en caso de existir algún percance, se puede perder toda la información.



Figura 1. Ubicación de Bienestar Universitario UNL, en donde se encuentra la Subdirección de Seguridad y Salud Ocupacional

5.2. Procedimiento

Para lograr el objetivo general del presente trabajo de titulación se realizó el proceso que se muestra a continuación:

Fase 1: Analizar la situación actual del manejo de la información dentro de la Subdirección de Seguridad y Salud Ocupacional.

En esta fase se aplicó la técnica de la entrevista, en la que se establece una comunicación interpersonal el investigador y el sujeto o de estudio y así obtener respuestas verbales a las interrogantes planteadas sobre el problema [32].

La entrevista se realizó al Licenciado Diego Paúl Falconí Espinosa, Subdirector de Seguridad y Salud Ocupacional, (ver Anexo I, Anexoll), de igual manera, se realizó una entrevista a la Doctora Iliana Voneth Figueroa Delgado, Médico Ocupacional de la Universidad Nacional de Loja (ver Anexo III), esto con el fin de determinar la situación actual que existe dentro de esta unidad.

Una vez analizados los resultados obtenidos de las entrevistas se procedió a especificar los requerimientos (ver Anexo IV), además, se consideraron las preguntas y respuestas más relevantes de las entrevistas, en las cuales se recolectó la información necesaria para llevar a cabo la construcción del software, a partir de la fase 2.

Fase 2: Desarrollar un software que facilite la sistematización de fichas médicas ocupacionales y test psicológicos.

El cumplimiento de esta fase se llevó a cabo a través de la metodología híbrida SXP, que se compone de 3 fases:

- **Planificación-Definición:** Se realizaron reuniones en las que se definen las historias de usuario (ver Anexo IV, Anexo X), además se diseñaron los diagramas establecidos dentro de la arquitectura 4 +1 (ver Anexo V), los que complementaron la fase de desarrollo.
- **Desarrollo:** Durante esta fase se realizó la codificación del software con el lenguaje de programación Python, dentro del framework Django, se desarrolló cada una de las funciones establecidas a través de los requerimientos y se procedió a realizar las pruebas unitarias de cada una de ellas, y al finalizar cada iteración se realizaron las correspondientes pruebas del sistema. (ver Anexo X).
- **Entrega y mantenimiento:** Al finalizar la etapa de desarrollo del sistema, se procedió a realizar una capacitación a los usuarios finales de este proyecto, con la finalidad de hacer un uso correcto del software. Esto se complementó con la entrega de la documentación técnica correspondiente.

Fase 3: Evaluar el funcionamiento del software a través de pruebas aplicadas por la Dirección de Tecnologías de Información.

En esta última fase se realizó una selección de herramientas que permitieron hacer las pruebas correspondientes al sistema, con el fin de corregir errores y así entregar un producto de calidad.

Finalmente, se realizaron las pruebas de aceptación de usuario, dentro del entorno en el que se hace uso del sistema.

Recursos

- **Recursos Científicos:**

- **Estudios de Caso:** Con esta técnica se realizó la exploración de los diferentes casos existentes, (ver sección 4.5. Trabajos Relacionados), lo que permitió obtener un amplio conocimiento del problema detectado y dar una solución al mismo. A partir de esto se hizo la delimitación del objeto de estudio y se obtuvieron los requerimientos para el desarrollo del software.
- **Investigación Bibliográfica:** Esta técnica fue la base para el fundamento teórico del TT, realizada a través de consultas de libros, artículos, tesis, etc. confiables en diferentes bases de datos.
- **Experimentación:** A través de dicha técnica se realizó la evaluación del software dentro de la Subdirección de Seguridad y Salud Ocupacional en conjunto con personal de DTI y el personal que labora en la unidad, esto con el fin de determinar que las funciones del software son las correctas y satisfacen las necesidades basadas en los requerimientos establecidos.

- **Recursos Técnicos:**

- **Entrevista:** Esta técnica se aplicó al Lic. Diego Falconí y a la Dra. Iliana Figueroa, con ella se pudo obtener información sobre la forma en que se lleva la gestión de las fichas médicas ocupacionales en la Subdirección de Seguridad y Salud Ocupacional de la UNL, y con ello se logró desarrollar el software como solución para el problema de investigación planteado. (ver Anexo 11.1)
- **Hardware:** Se hace uso de los equipos de los tesisas para desarrollar cada una de las actividades (Laptop Hp).
- **Software:** Se utilizaron varias herramientas que sirvieron para completar eficazmente el presente TT.
- **Django:** Dentro de este framework de desarrollo de código abierto, se realizó la codificación del software en lenguaje Python.

- **Metodología SXP:** Se empleó esta metodología de desarrollo de software a partir del segundo objetivo del TT, de acuerdo a cada una de las fases que la componen.

Participantes

El presente TT enfocado en la línea de investigación de Ingeniería de Software, fue desarrollado por los siguientes participantes:

- Yannela Mishelle Castro Valarezo y Josué Fernando Ortega Jaramillo, como estudiantes autores del TT. Empezando sus actividades desde el planteamiento del tema del Proyecto de Trabajo de Titulación (PTT), hasta la ejecución y cumplimiento de los objetivos planteados.
- Ing. Valeria del Rosario Herrera Salazar, como directora del TT, quien revisa y supervisa los avances realizados por los autores.
- Lic. Diego Falconí Espinosa, subdirector de la unidad de Seguridad y Salud Ocupacional, y Dra. Iliana Figueroa, médico ocupacional, quienes colaboraron con entrevistas que sirvieron para obtener los requerimientos que se deben satisfacer con la implementación del software.
- Ing. Danny Muñoz, jefe de desarrollo del Departamento de Tecnología de la Información, quien es responsable de establecer la forma en que se lleva el desarrollo del TT, además aplicará las pruebas para verificar que el software funciona correctamente.

6. Resultados

Dentro de esta sección se describe cada uno de los resultados obtenidos de acuerdo a las actividades establecidas para cada objetivo que se cumple en el TT.

6.1. OBJETIVO 1: Analizar la situación actual del manejo de la información dentro de la Subdirección de Seguridad y Salud Ocupacional.

Para el cumplimiento del primer objetivo, primeramente, se realizó una entrevista a la Dra. Iliana Figueroa, médico ocupacional de la UNL (ver sección 11.3. Anexo III), en dicha entrevista se pudo obtener a mejor detalle las necesidades que se desea cumplir con el desarrollo de la solución informática.

6.1.1. Análisis de los requerimientos con los miembros de la Subdirección de Seguridad y Salud Ocupacional.

A partir de la entrevista realizada se realizó la documentación de Especificación de Requisitos (ver Anexo IV), en la cual se determinó los usuarios que intervienen en el sistema y las actividades que realizará cada uno. (Tabla 1, Tabla 2)

Tabla 1. Usuario Administrador.

Usuario	Administrador
Formación	Persona con conocimientos básicos en la manipulación de sistemas para gestionar información
Habilidades	Administrar sistema
Actividades	<ul style="list-style-type: none">• Receptar Certificados Generados• Enviar Notificación

Tabla 2. Usuario Médico.

Usuario	Médico
Formación	Persona con conocimientos básicos en la manipulación de sistemas para gestionar información
Habilidades	Administrar sistema
Actividades	<ul style="list-style-type: none">• Registrar Fichas Médicas• Buscar Fichas Médicas• Actualizar Fichas Médicas• Registrar Hoja de Evolución• Actualizar Hoja de Evolución• Generar Certificados• Enviar Notificación

Los requisitos del sistema se basan en los reglamentos, tanto de la Universidad Nacional, como de la Subdirección de Seguridad y Salud Ocupacional, los cuales fueron aprobados por el Subdirector de Seguridad y Salud Ocupacional, Lic. Diego Falconi, y por el Médico Ocupacional de la UNL, Dra. Iliana Figueroa. A continuación, en la Tabla 3, se muestran los requerimientos funcionales establecidos. (ver Anexo IV)

Tabla 3. Requerimientos Funcionales.

REQUERIMIENTOS FUNCIONALES	
RF001	Iniciar Sesión
RF002	Obtener datos del SIAAF
RF003	Registrar Ficha Médica Inicial
RF004	Buscar Ficha Médica Inicial
RF005	Registrar Ficha Médica Periódica
RF006	Registrar Hoja de Evolución
RF007	Actualizar Ficha Médica Periódica
RF008	Actualizar Hoja de Evolución
RF009	Buscar Ficha Médica Periódica
RF010	Registrar Ficha Médica de Reintegro
RF011	Buscar Ficha Médica de Reintegro
RF012	Registrar Ficha Médica de Salida
RF013	Buscar Ficha Médica de Salida
RF014	Obtener Código CIE
RF015	Generar Certificado
RF016	Receptar Certificado
RF017	Enviar Notificación

Así mismo, en la Tabla 4 se indican los Requerimientos No Funcionales, los cuales se detallan en el documento de Especificación de Requisitos (ver Anexo IV).

Tabla 4. Requerimientos No Funcionales.

REQUERIMIENTOS NO FUNCIONALES	
Requerimiento	Descripción
Seguridad	Cada usuario tendrá sus propios privilegios y funciones
Fiabilidad	La información del sistema será respaldada automáticamente.

Disponibilidad	El sistema deberá estar disponible los 365 días del año.
Portabilidad	El sistema deberá funcionar en cualquier navegador web.
Usabilidad	El sistema deberá tener una interfaz amigable e intuitiva con los usuarios.
Tiempo de Respuesta	El sistema deberá tener un tiempo de respuesta rápida, entre 2 a 5 segundos como máximo.
Arquitectura	El sistema deberá ser fácilmente adaptable y escalable.
Producción y Mantenimiento	El sistema deberá pasar por un control de calidad. El sistema estará disponible para DTI, que será el encargado de darle mantenimiento al módulo desarrollado.

6.1.2. Definir el procedimiento para la recolección de información.

Para completar el proceso de recolección de información, se implementó la técnica de la entrevista, de modo que, se realizaron 3 entrevistas (ver sección 11. Anexos), en las cuales se logró obtener información relevante, y una vez establecidos los requerimientos del sistema, se logró recolectar los datos específicos y necesarios para desarrollar el sistema.

A continuación, en la Tabla 5, se listan las preguntas de las entrevistas que fueron más relevantes al momento de obtener la información.

Tabla 5. Información recolectada con entrevistas.

Pregunta	Información Recolectada
¿Qué tipo de información requiere guardar en el Sistema?	Información relacionada con las fichas médicas ocupacionales, que contienen datos básicos y la historia médica laboral de la persona que ingresa, se mantiene laborando y posteriormente sale de la institución.
¿Qué datos son necesarios para hacer el registro inicial de una ficha médica ocupacional?	Actividades que va a cumplir ante la institución, se debe conocer situaciones personales, antecedentes, situaciones laborales, si está apto/a para desarrollar las funciones o si necesita algún tratamiento médico para solucionar la emergencia al momento que ingresa.
¿Qué información se guarda cuando un paciente acude a una consulta?	Sintomatología presentada, si se automedicó, novedades de los síntomas, si han avanzado, aminoraron, y también su evolución.
¿Existen diferentes tipos de fichas médicas? ¿Cuáles serían estos tipos?	<ul style="list-style-type: none"> ○ Fichas médicas de inicio ○ Fichas médicas periódicas ○ Fichas médicas de salida

	<ul style="list-style-type: none"> ○ Fichas médicas de reintegro
¿Qué tipos de certificados se emiten para un paciente?	<ul style="list-style-type: none"> ○ Certificado de Aptitud Médica ○ Certificado de Ausencias

6.2. OBJETIVO 2: Desarrollar un software que facilite la sistematización de fichas médicas ocupacionales y test psicológicos.

6.2.1. Análisis y diseño de diagramas e historias de usuario, a partir de los requerimientos obtenidos.

Para la creación de historias de usuario y diagramas se dividieron las actividades en tres iteraciones, con las fases definidas en la metodología SXP, obteniendo así 10 historias de usuario (ver Anexo IV), y el documento de arquitectura 4 + 1, en donde se modelaron los diagramas necesarios (ver Anexo V).

A continuación, en la Tabla 6 se indica las historias de usuarios establecidas para cada iteración:

Tabla 6. Planificación de Historias de Usuario para cada iteración.

Iteración	Fecha Inicio	Fecha Fin	Historia de Usuario
1	20/06/2022	21/06/2022	<i>Como:</i> Usuario <i>Quiero:</i> Iniciar sesión <i>Para:</i> Acceder al sistema
	22/06/2022	24/06/2022	<i>Como:</i> Médico <i>Quiero:</i> Buscar los datos de un paciente <i>Para:</i> obtener los datos personales del SIAAF-UNL y hacer el registro de los datos personales del paciente en las fichas médicas ocupacionales.
	25/06/2022	27/06/2022	<i>Como:</i> Médico <i>Quiero:</i> Registrar fichas médicas iniciales <i>Para:</i> Ingresar los datos médicos iniciales y generales del paciente.
2	29/06/2022	05/07/2022	<i>Como:</i> Médico <i>Quiero:</i> Registrar ficha periódica <i>Para:</i> Llevar un control del estado de los pacientes en su entorno de trabajo.
	06/07/2022	09/07/2022	<i>Como:</i> Médico <i>Quiero:</i> Actualizar los datos de la ficha periódica <i>Para:</i> Mantener al día los datos médicos del paciente.
	11/07/2022	13/07/2022	<i>Como:</i> Médico <i>Quiero:</i> Registrar fichas médicas de reintegro <i>Para:</i> Poder controlar el reintegro de un servidor a sus labores, en caso de haberse ausentado por más de 15 días.
3	14/07/2022	15/07/2022	<i>Como:</i> Médico <i>Quiero:</i> Registrar ficha médica de salida <i>Para:</i> Notificar y controlar la salida de un servidor de la institución.
	18/07/2022	21/07/2022	<i>Como:</i> Médico <i>Quiero:</i> Obtener los códigos CIE,

			<i>Para:</i> Clasificar la o las patologías diagnosticadas en los pacientes, dentro de la ficha médica.
	25/07/2022	26/07/2022	<i>Como:</i> Médico <i>Quiero:</i> Buscar un paciente <i>Para:</i> Poder visualizar los datos registrados anteriormente en las fichas médicas ocupacionales.
	27/07/2022	29/07/2022	<i>Como:</i> Médico <i>Quiero:</i> Generar certificados <i>Para:</i> enviarlos al Administrador de Salud Ocupacional y tramitar el proceso que corresponda.
	01/08/2022	03/08/2022	<i>Como:</i> Administrador <i>Quiero:</i> Listar los certificados <i>Para:</i> Visualizar los certificados generados y enviarlos a las personas que corresponda.

La planificación detallada de las iteraciones se puede observar en el Anexo X: Desarrollo de la Metodología Híbrida SXP.

- **Arquitectura de Software**

Para el diseño de software se empleó la arquitectura 4 +1 de Kruchten, en donde se incluye los distintos diagramas UML para la documentación del proyecto (ver Anexo V). Esta arquitectura se compone de 5 vistas, siendo estas la vista de escenarios, representada con el diagrama de casos de uso, la vista lógica que incluye el modelo conceptual y el diagrama de clases, la vista de procesos que indica la interacción entre los mismos a través de los diagramas de actividades, la vista de despliegue que se representa por medio del diagrama de componentes y finalmente la vista física en donde se muestran los componentes físicos y su conexión, a través del diagrama de despliegue, todo lo anteriormente mencionado se refleja en la tabla 7:

Tabla 7. Arquitectura 4 + 1.

Vista	Elemento Modelado	Descripción
Vista de Escenarios	Casos de Uso	Interacción entre los stakeholders y el sistema.
Vista Lógica	Modelo Conceptual Diagrama de Clases	Servicios que brinda el sistema a sus stakeholders.
Vista de Procesos	Diagrama de Actividades	Indica los procesos que realizan las funcionalidades del sistema.
Vista de Despliegue	Diagrama de Componentes	Detalla la relación de los componentes del sistema.
Vista Física	Diagrama de Despliegue	Muestra los componentes físicos del sistema y su conexión.

○ Vista de Escenarios

Los actores principales, como se muestra en el diagrama (ver Figura 1), son Administrador de Salud Ocupacional y Médico, cada uno de ellos realiza diferentes funciones que serán atendidas en el sistema a desarrollar.

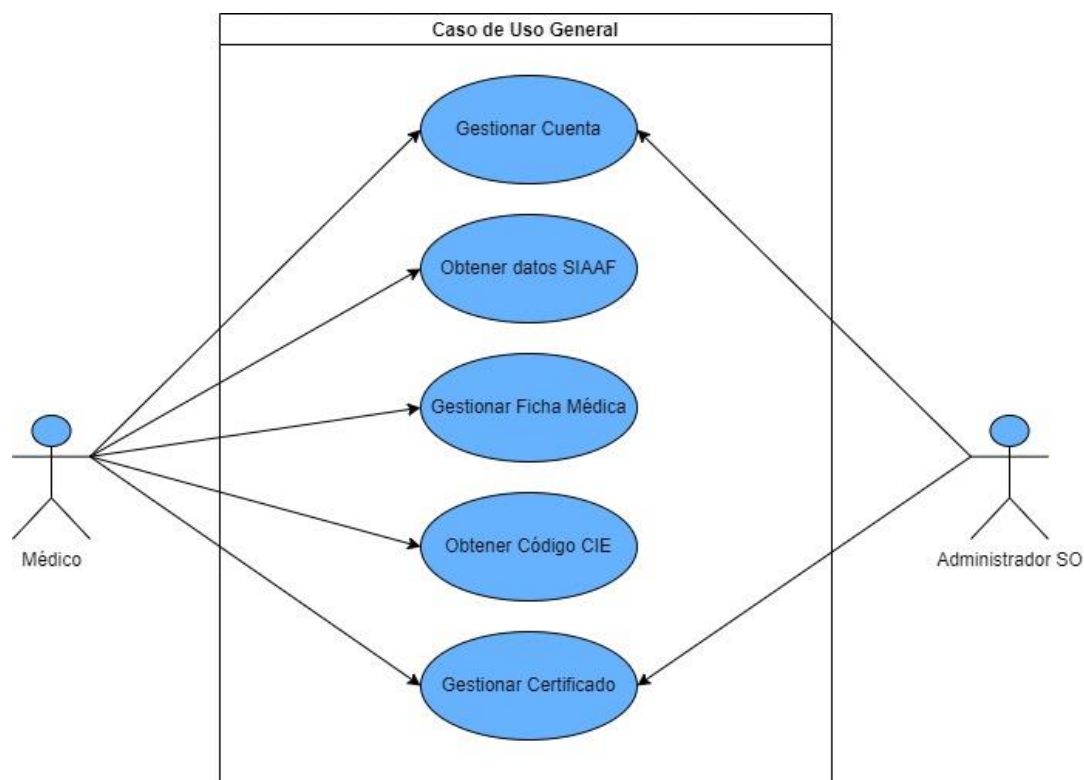


Figura 2. Diagrama de Caso de Uso General

○ Vista Lógica

Dentro de esta vista se muestran los requisitos funcionales del sistema, a través del modelo conceptual y diagrama de clases. (Figura 2 y 3)

Se debe tener en cuenta el siguiente uso de colores para las clases definidas:

- Clases existentes en el SIAAF
- Clases Nuevas, creadas para el sistema

Para observar todos los atributos de las clases creadas ver Anexo XI.

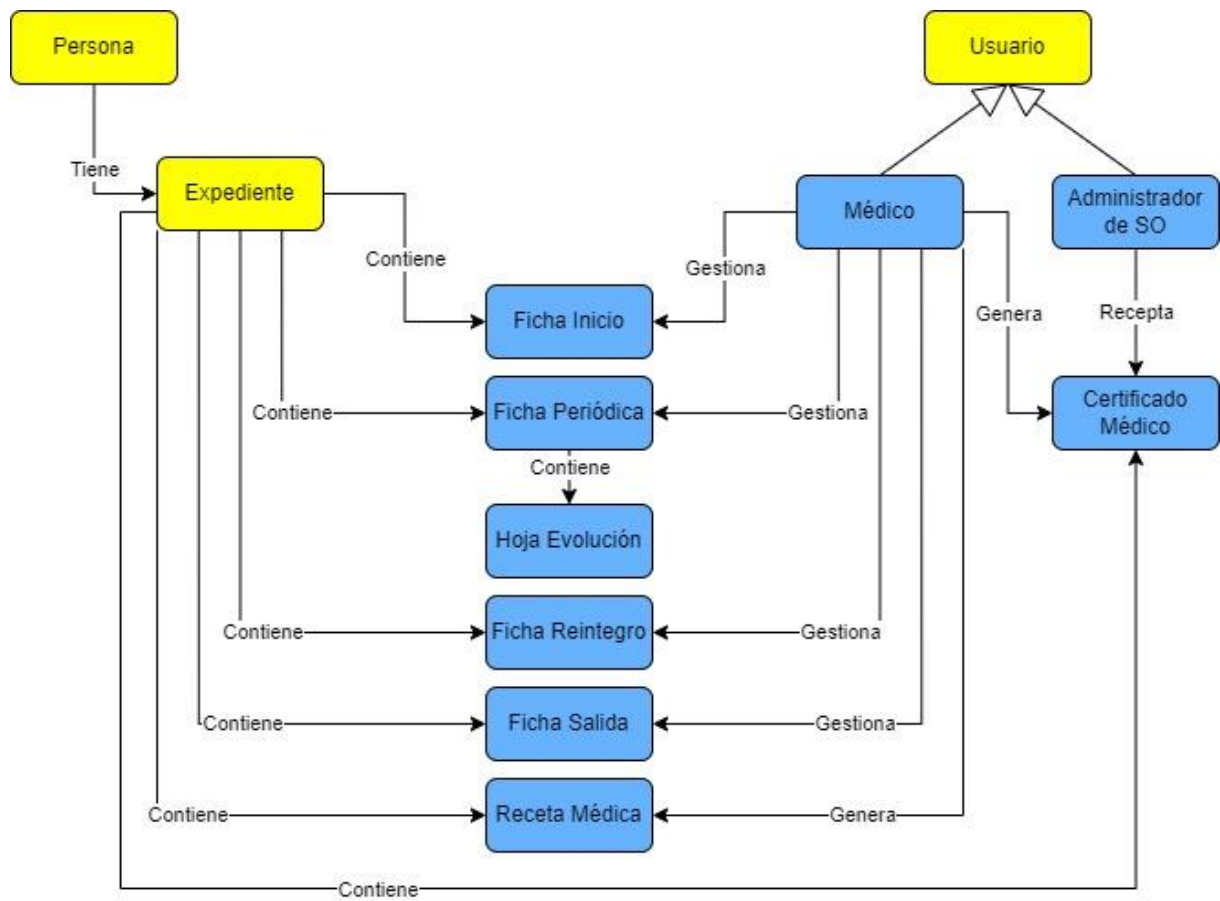


Figura 3. Modelo Conceptual

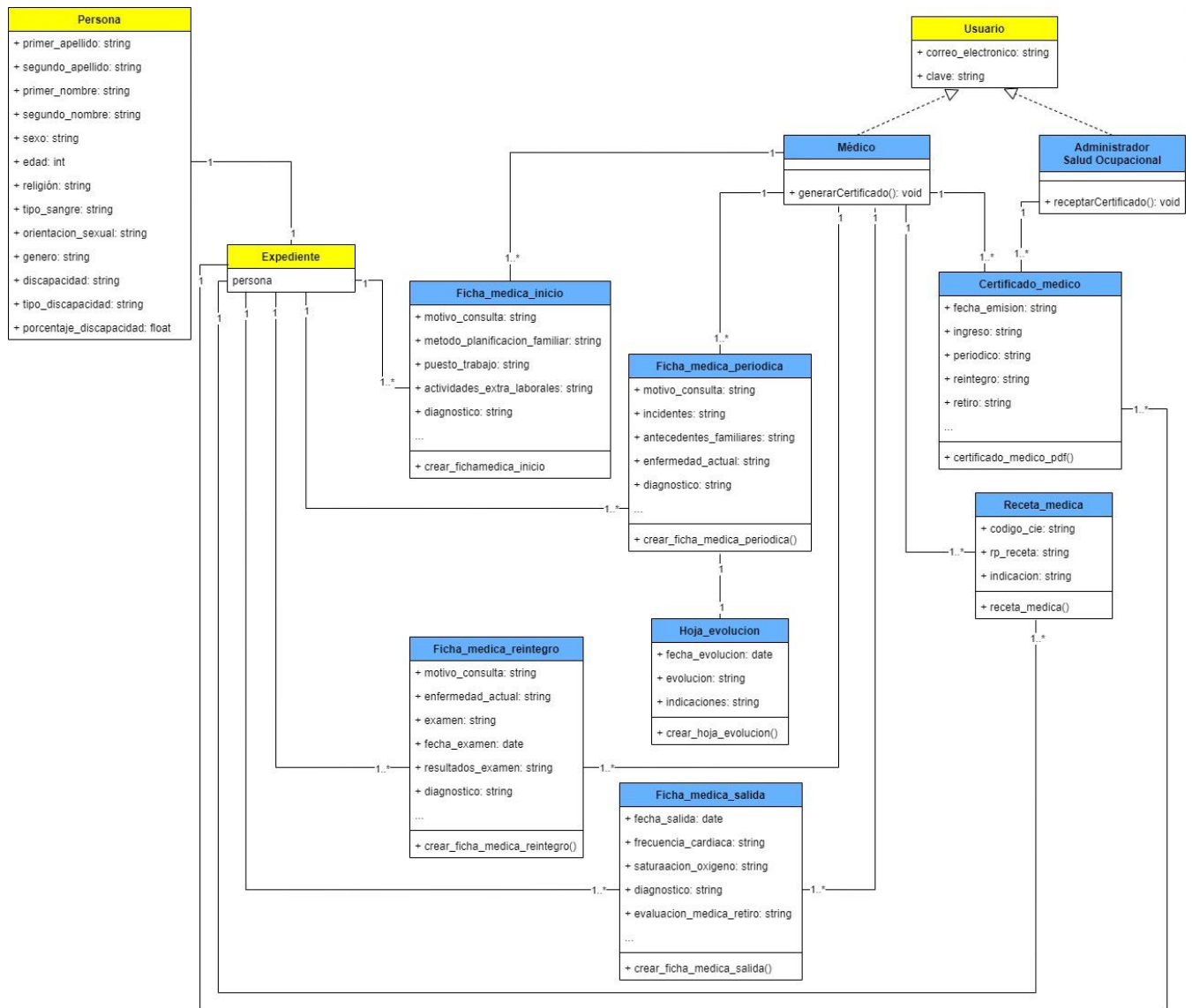


Figura 4. Diagrama de Clases

○ Vista de Procesos

Los diagramas de actividades representan la forma en que se comporta el sistema y sus procesos de una manera simple, a continuación, se pueden observar los diagramas más relevantes (Figura 4-8):

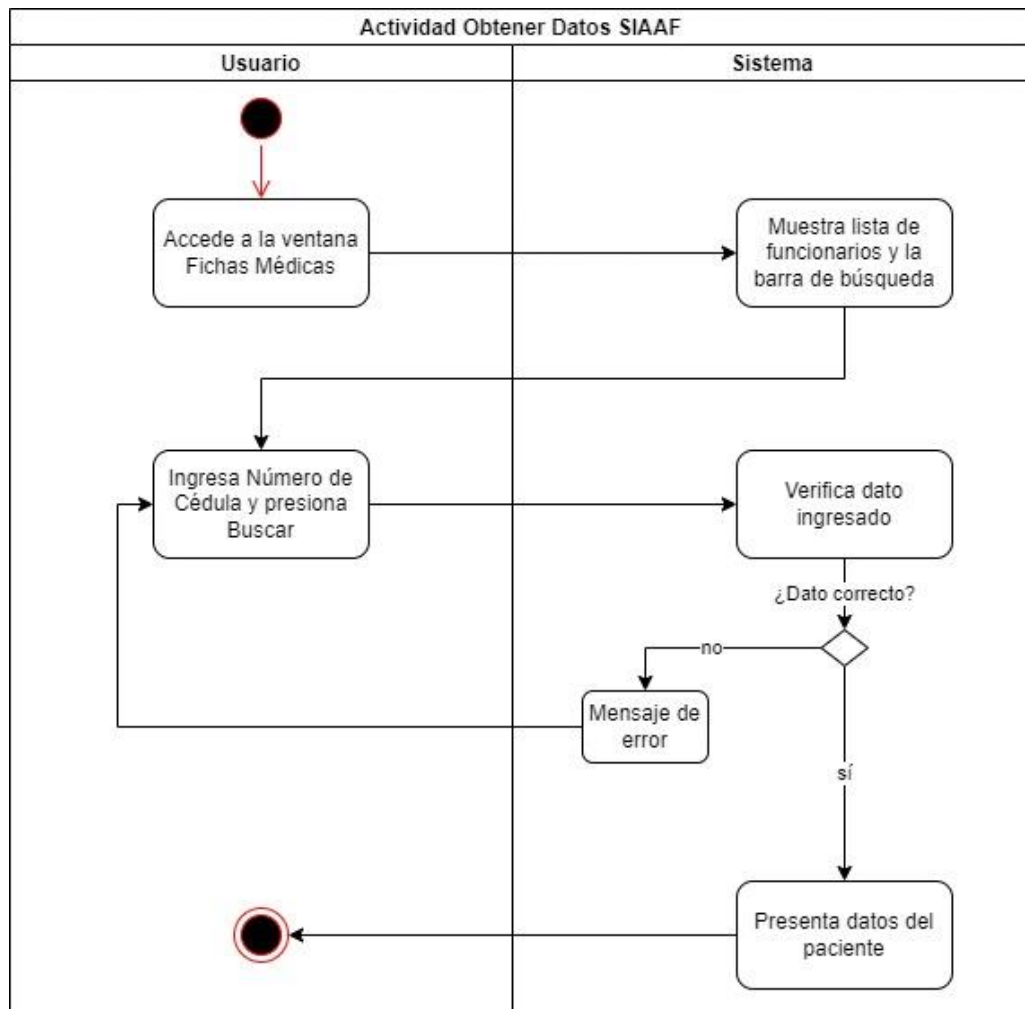


Figura 5. Diagrama de Actividad Obtener Datos SIAAF

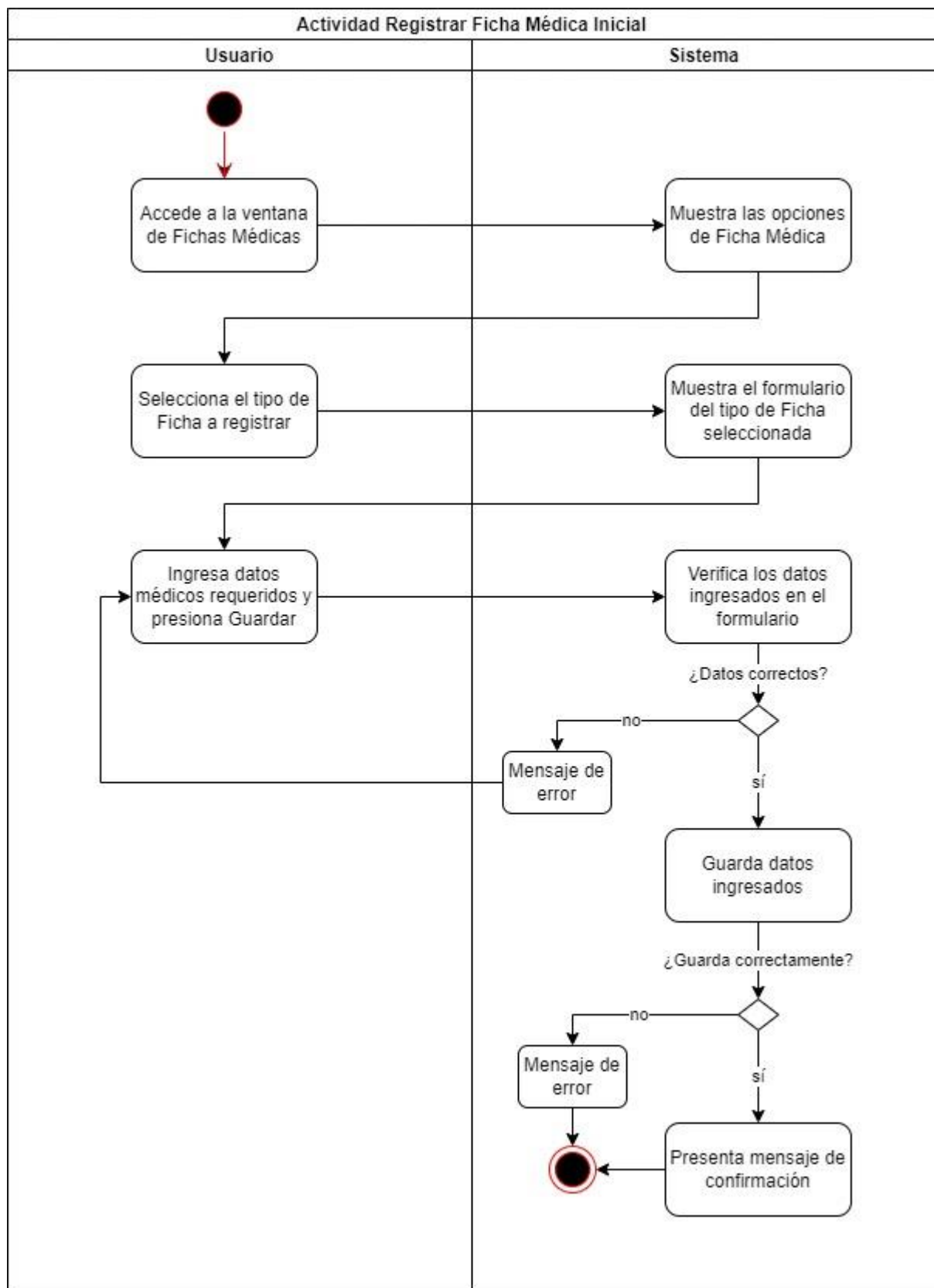


Figura 6. Diagrama de Actividad Registrar Ficha Inicial

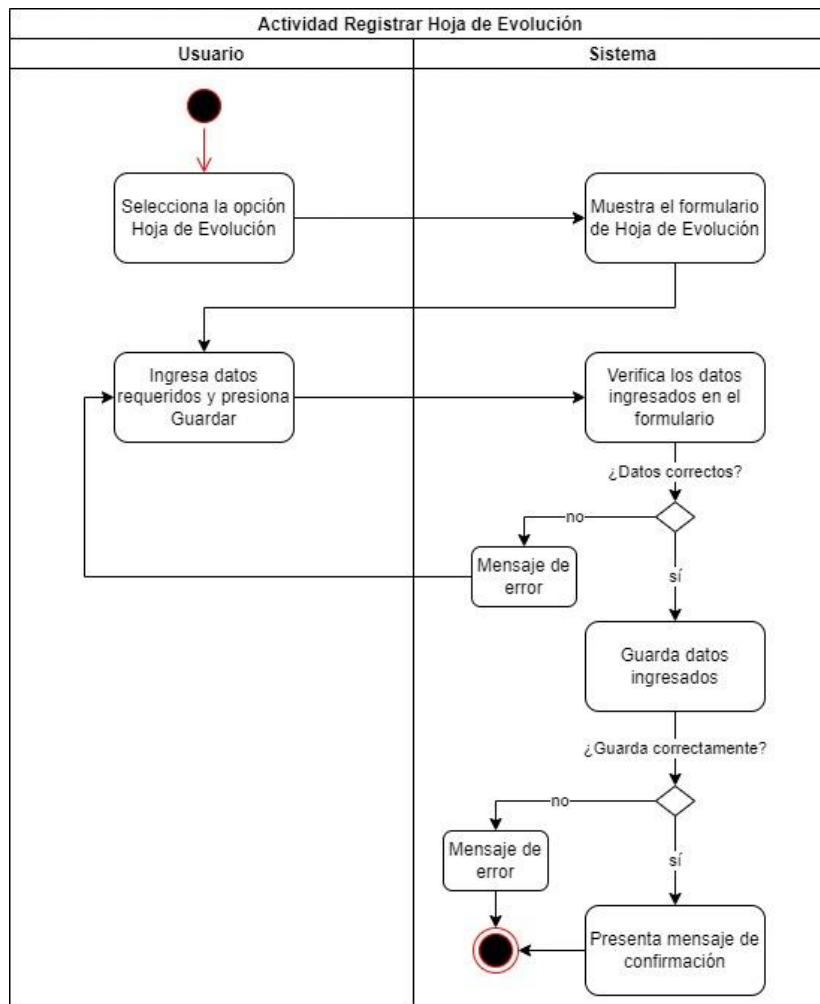


Figura 7. Diagrama de Actividad Registrar Hoja de Evolución

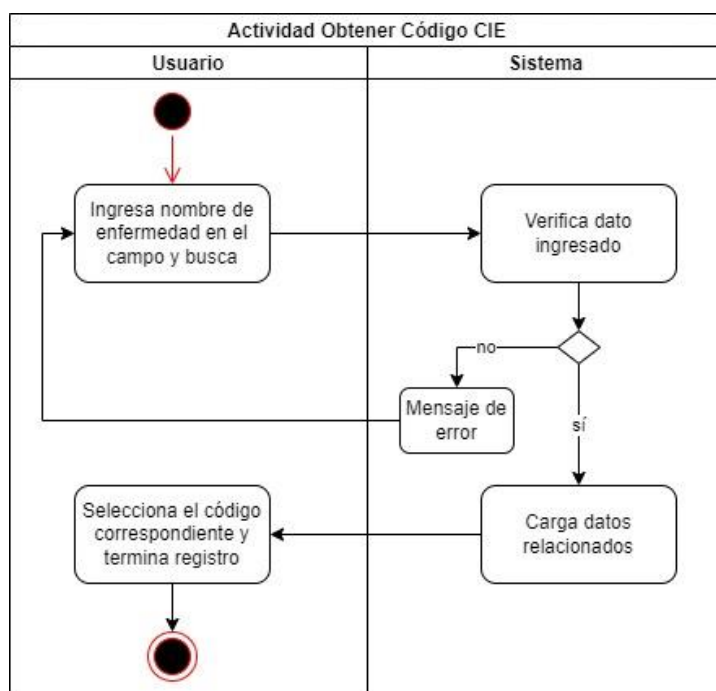


Figura 8. Diagrama de Actividad Obtener Código CIE

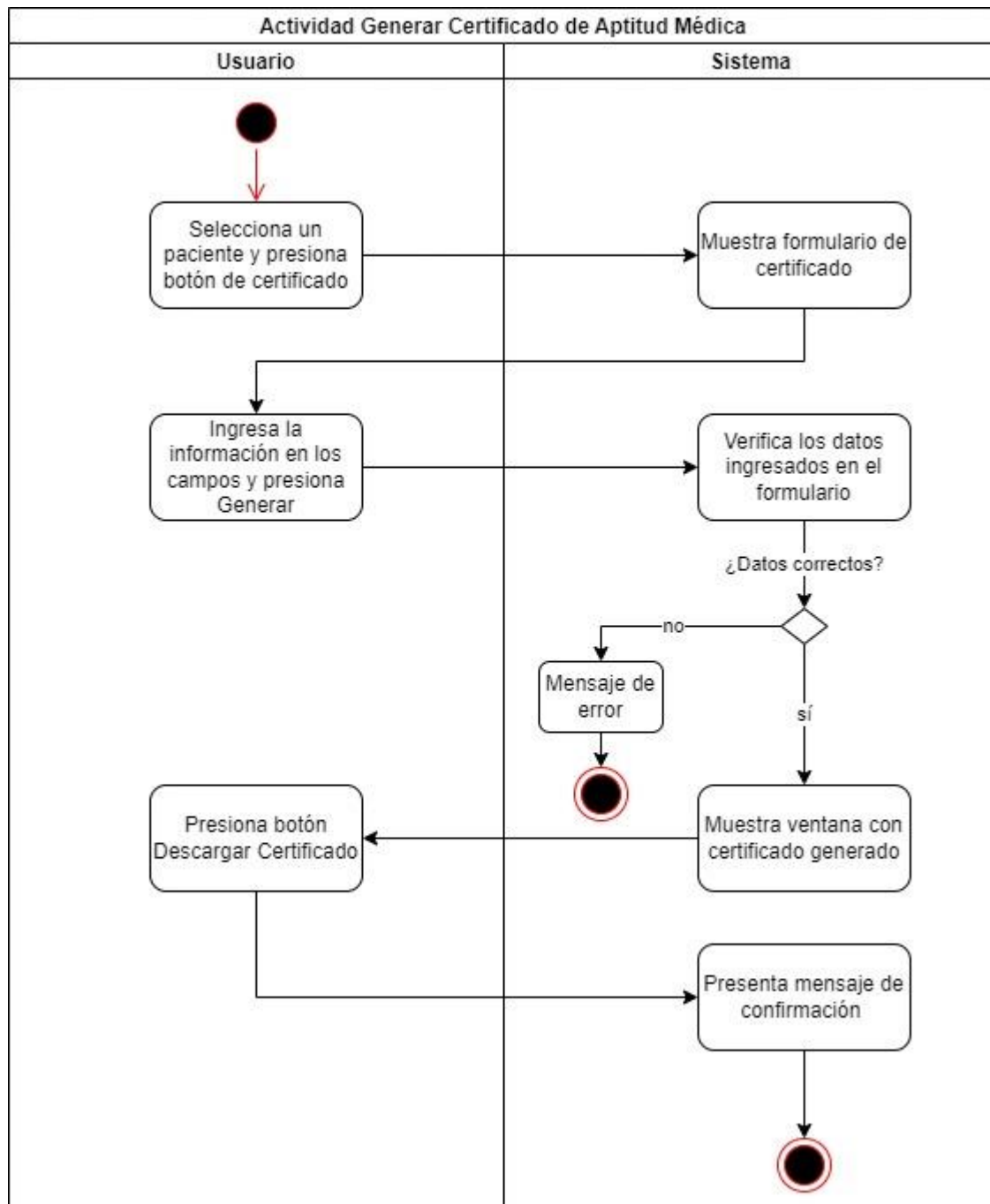


Figura 9. Diagrama de Actividad Generar Certificado de Aptitud Médica

- **Vista de Despliegue**

En esta vista se organiza los módulos o componentes del sistema con la siguiente estructura, como se muestra en la figura 9:

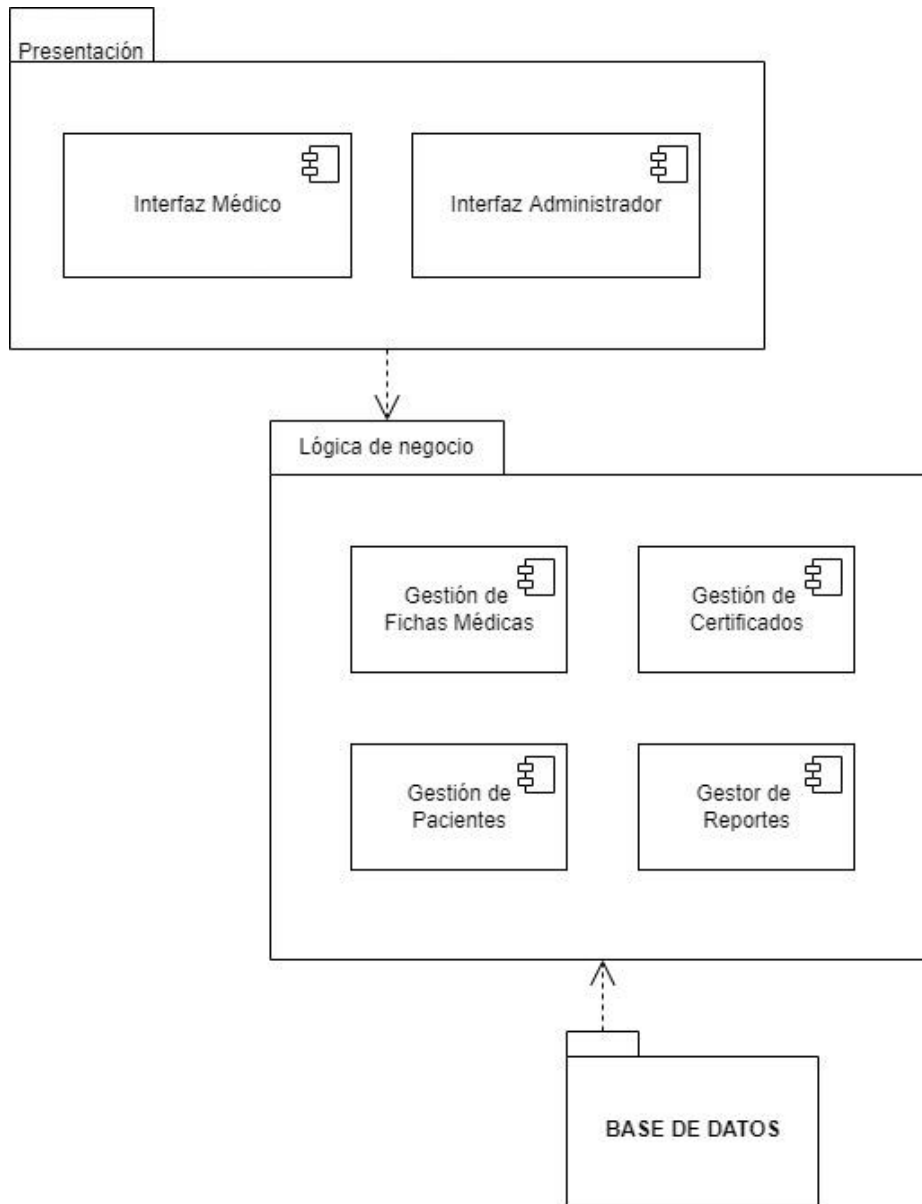


Figura 10. Diagrama de Componentes

- **Vista Física**

Se representa por medio del diagrama de despliegue (Figura 10), que muestra los nodos finales del sistema, usando la arquitectura cliente-servidor.

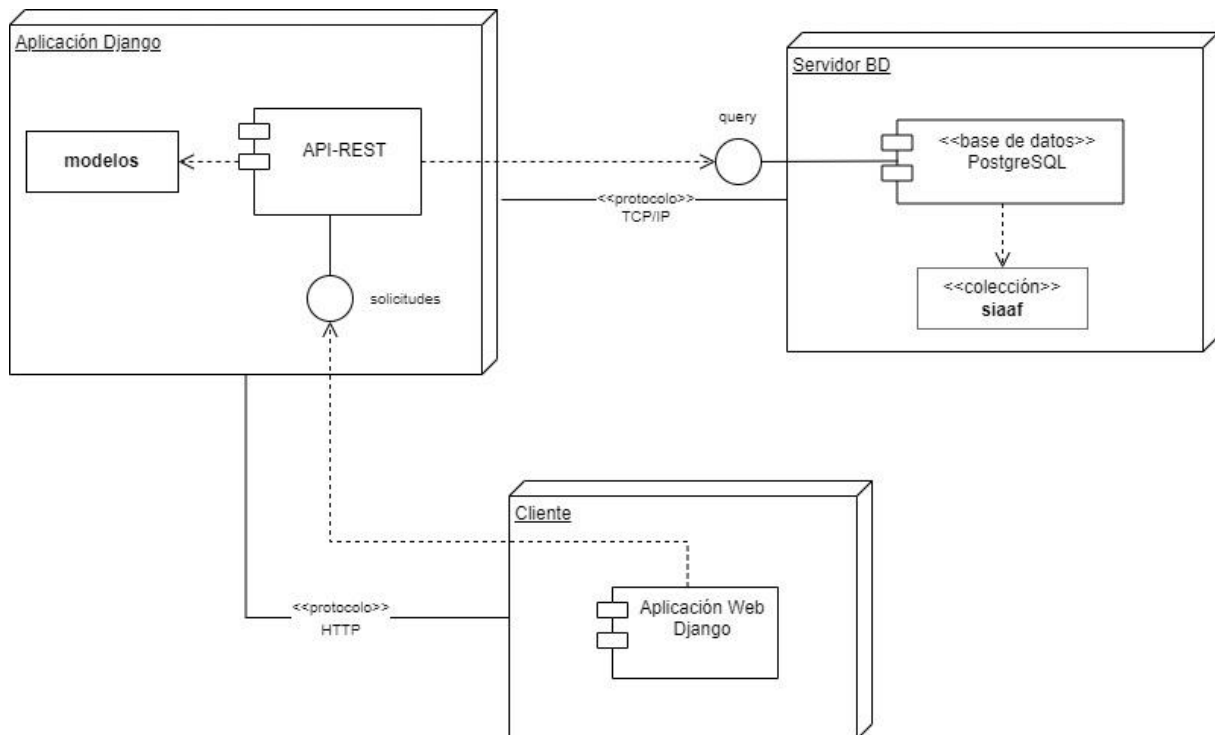


Figura 11. Diagrama de Despliegue

6.2.2. Desarrollo e implementación de las funciones del software, en base a las tecnologías y metodologías establecidas por el departamento de TI.

Una vez completada la primera fase de la metodología, se realiza la fase de Desarrollo, de acuerdo a cada iteración, haciendo uso del framework Django y el lenguaje Python para la codificación del sistema, con el fin de satisfacer cada requisito establecido anteriormente.

A continuación, en las figuras se muestra la codificación del sistema:

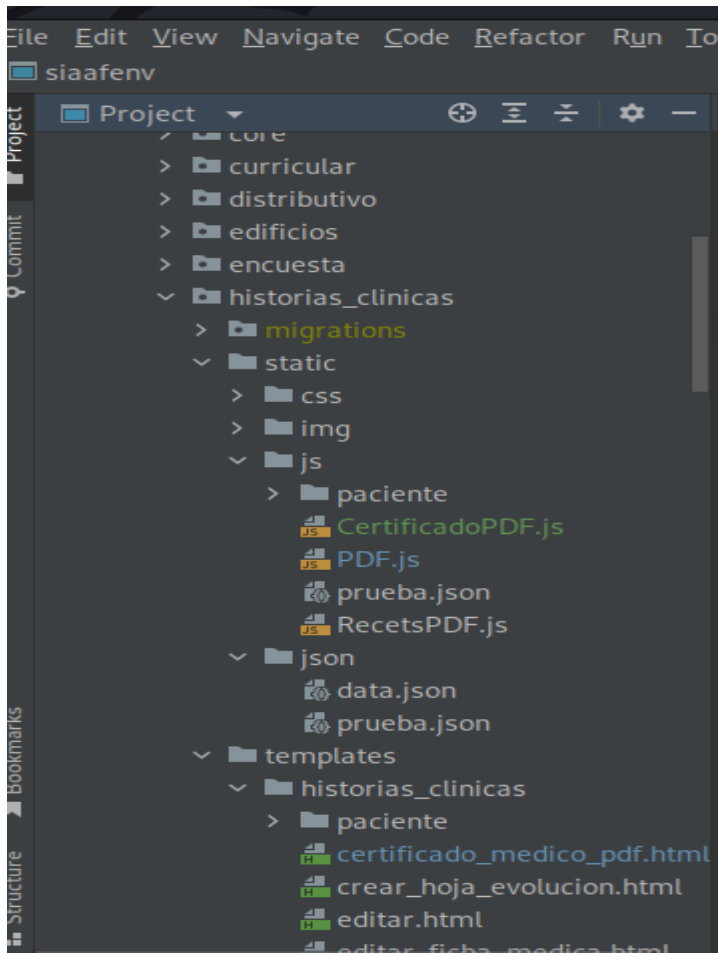


Figura 12. Creación del Módulo-StartApp


```

@permission_required('configuracion.view_crear_paciente', raise_exception=True, )
def crear_paciente(request):
    formulario_ficha = Ficha_medicaForm(request.POST)
    url_id = request.GET['id']
    print('aqui esta la cedula de la url', url_id)
    buscar_funcionario = Funcionario.objects.get(id=url_id)
    edad = buscar_funcionario.usuario.persona.get_edad()
    print('edad', edad)
    context = {
        'funcionario': buscar_funcionario,
    }
    print(' encuentro la persona con cedula', buscar_funcionario.usuario.persona.numero_documento)
    if request.method == 'POST':
        datos = request.POST.dict()
        fm = Ficha_medica()
        fm.ciiu = datos.get('ciiu')
        fm.ruc = buscar_funcionario.usuario.persona.numero_documento
        fm.nro_archivo = datos.get('nro_archivo')
        fm.establecimiento_salud = datos.get('establecimiento_salud')
        fm.institucion_sistema_nompreempresa = datos.get('institucion_sistema_o_nompre_empresa')
        fm.num_historia_clinica = datos.get('num_historia_clinica')
        fm.primer_apellido = buscar_funcionario.usuario.persona.primer_apellido
        fm.segundo_apellido = buscar_funcionario.usuario.persona.segundo_apellido
        fm.primer_nombre = buscar_funcionario.usuario.persona.primer_nombre
        fm.segundo_nombre = buscar_funcionario.usuario.persona.segundo_nombre
        fm.sexo = buscar_funcionario.usuario.persona.sexo
        fm.religion = buscar_funcionario.usuario.persona.religion
        fm.edad = edad
        fm.fecha_ingreso_trabajo = datos.get('fecha_ingreso_trabajo')
        fm.puesto_trabajo = datos.get('puesto_trabajo')
        fm.area_trabajo = datos.get('area_trabajo')
        fm.actividad_relevante = datos.get('actividad_relevante')

        fm.grupo_sanguineo = buscar_funcionario.usuario.persona.tipo_sangre
        fm.porcentaje_discapacidad = buscar_funcionario.usuario.persona.porcentaje_discapacidad
        fm.tipo_discapacidad = buscar_funcionario.usuario.persona.tipo_discapacidad
        fm.discapacidad = buscar_funcionario.usuario.persona.discapacidad
        fm.orientacion_sexual = buscar_funcionario.usuario.persona.orientacion_sexual
        fm.identidad_genero = datos.get('identidad_genero')
        fm.lateralidad = datos.get('actividad_relevante')
        if (fm.save() != False):
            messages.success(request, 'Ficha medica registrada correctamente')
            print('Ficha medica registrada existosamente')
            return redirect('historias_clinicas:listar_Paciente')

    return render(request, 'historias_clinicas/index_ficha_medica_inicio.html', context)

```

Figura 13. Función Crear Ficha Médica al Paciente.

```


def crear_fichamedica_inicio(request):
    url_ruc = request.GET['id']
    print('crear_fichamedica_inicio y el id es:', url_ruc)
    paciente = Ficha_medica.objects.get(id=url_ruc)
    context2 = {
        'paciente': paciente,
    }
    print('paciente', paciente.edad)
    print(' este es el paciente', paciente.ruc)
    try:
        print('entro al try')
        ficha_medica_id = Ficha_medica.objects.get(id=url_ruc)
        listarFMI = Ficha_medica_inicio.objects.get(ficha_medica=url_ruc)
        if request.method == 'POST':
            print('formulario fichamedica_inicio')
            datos = request.POST.dict()
            crear_ficha = Ficha_medica_inicio()
            crear_ficha.motivo_consulta = datos.get('motivo_consulta')
            crear_ficha.antecedentes_clinicos_quirurgicos = datos.get('antecedentes_clinicos_')
            crear_ficha.menarquia = datos.get('menarquia')
            crear_ficha.ciclos = datos.get('ciclos')
            crear_ficha.fecha_ultima_mens = datos.get('fecha_ultima_mens')
            crear_ficha.gestas = datos.get('gestas')

            messages.success(request, 'Ficha medica inicio registrada correctamente')
            return redirect('/historias_clinicas/fichasmedicas/lista?id=' + url_ruc)

    print('edad', paciente.edad)
    return render(request, 'historias_clinicas/ficha_medica_inicio.html', context2)

```

Figura 14. Crear Ficha Médica de Inicio.

 56  766

```

def crear_ficha_medica_periodica(request):
    url_ruc = request.GET['id']
    print('crear_fichamedica_inicio y el id es:', url_ruc)
    paciente = Ficha_medica.objects.get(id=url_ruc)
    context2 = {
        'paciente': paciente,
    }
    print('paciente', paciente.edad)
    print('este es el paciente', paciente.ruc)
    try:
        print('entro al try')

        ficha_medica_id = Ficha_medica.objects.get(id=url_ruc)
        listarFMI = Ficha_medica_periodica.objects.get(ficha_medica=url_ruc)
        print('datos', listarFMI.temperatura)
        if request.method == 'POST':
            print('formulario fichamedica_periodica')
            datos = request.POST.dict()
            crear_ficha = Ficha_medica_periodica()
            crear_ficha.motivo_consulta = datos.get('motivo_consulta')
            crear_ficha.antecedentes_clinicos_quirurgicos = datos.get('antecedentes_clinicos_quirurgicos')
            crear_ficha.tabaco = datos.get('tabaco')
            crear_ficha.tiempo_consumo_tabaco = datos.get('tiempo_consumo_tabaco')
            crear_ficha.cantida_tabaco = datos.get('cantida_tabaco')
            if (crear_ficha.save() != False):
                print('Ficha medica inico registrada existosamente')
                messages.success(request, 'Ficha medica periodica registrada correctamente')
                return redirect('/historias_clinicas/fichasmedicas/lista?id=' + url_ruc)

    print('edad', paciente.edad)
    return render(request, 'historias_clinicas/ficha_medica_periodica.html', context2)
  
```

Figura 15. Función Crear Ficha Médica Periódica.

```

def crear_hoja_evolucion(request):
    url_ruc = request.GET['id']
    print('hoja evolucion:', url_ruc)
    paciente = Ficha_medica.objects.get(id=url_ruc)
    lista = Hoja_evolucion.objects.filter(ficha_medica=url_ruc).order_by('id')
    print('nombres y apellidos', paciente.primer_nombre)
    if request.method == 'POST':
        print('formulario fichamedica_inicio')
        datos = request.POST.dict()
        crear_ficha = Hoja_evolucion()
        crear_ficha.fecha_evolucion = datos.get('fecha_evolucion')
        crear_ficha.evolucion = datos.get('evolucion')
        crear_ficha.indicaciones = datos.get('indicaciones')
        crear_ficha.ficha_medica = paciente
        if (crear_ficha.save() != False):
            print('Ficha medica inico registrada existosamente')
            messages.success(request, 'Hoja evolucion registrada correctamente')
            return redirect('/historias_clinicas/ficha_medica/hoja_evolucion?id=' + url_ruc)
            # return redirect('/historias_clinicas/fichasmedicas/lista?id=' + url_ruc)
    context = {
        'paciente': paciente,
        'lista': lista,
    }
    return render(request, 'historias_clinicas/hoja_evolucion.html', context)

```

Figura 16. Función Crear Hoja de Evolución

```

def crear_ficha_medica_reintegro(request):
    url_ruc = request.GET['id']
    print('crear_fichamedica_inicio y el id es:', url_ruc)
    paciente = Ficha_medica.objects.get(id=url_ruc)
    context2 = {
        'paciente': paciente,
    }
    print('paciente', paciente.edad)
    print(' este es el paciente', paciente.ruc)
    try:
        print('entro al try')
        ficha_medica_id = Ficha_medica.objects.get(id=url_ruc)
        listarFMI = Ficha_medica_reintegro.objects.get(ficha_medica=url_ruc)
        if request.method == 'POST':
            print('formulario fichamedica_inicio')
            datos = request.POST.dict()
            crear_ficha = Ficha_medica_reintegro()
            crear_ficha.motivo_consulta = datos.get('motivo_consulta')
            crear_ficha.observaciones_accidentestrabajo_calificadoiess = datos.get(
                'observaciones_accidentestrabajo_calificadoiess')
            crear_ficha.enfermedad_actual = datos.get('enfermedad_actual')
            crear_ficha.presion_alta = datos.get('presion_alta')
            crear_ficha.temperatura = datos.get('temperatura')
            if (crear_ficha.save() != False):
                print('Ficha medica reintegro registrada exitosamente')
                messages.success(request, 'Ficha medica reintegro registrada correctamente')
                return redirect('/historias_clinicas/fichasmedicas/lista?id=' + url_ruc)
    return render(request, 'historias_clinicas/ficha_medica_reintegro.html', context2)

```

Figura 17. Función Crear Ficha Médica de Reintegro.

```

def crear_ficha_medica_salida(request):
    url_ruc = request.GET['id']
    print('crear_fichamedica_inicio y el id es:', url_ruc)
    paciente = Ficha_medica.objects.get(id=url_ruc)
    context2 = {
        'paciente': paciente,
    }
    print('paciente', paciente.edad)
    print('este es el paciente', paciente.ruc)
    try:
        print('entro al try')
        ficha_medica_id = Ficha_medica.objects.get(id=url_ruc)
        listarFMI = Ficha_medica_salida.objects.get(ficha_medica=url_ruc)
        if request.method == 'POST':
            datos = request.POST.dict()
            crear_ficha = Ficha_medica_salida()
            crear_ficha.fecha_salida = datos.get('fecha_salida')
            crear_ficha.antecedentes_clinicos_quirurgicos = datos.get('antecedentes_clinicos_quirurgicos')
            crear_ficha.accidentestrabajo_calificadoiess = datos.get('accidentestrabajo_calificadoiess')
        print('edad', paciente.edad)

    return render(request, 'historias_clinicas/ficha_medica_salida.html', context2)

```

Figura 18. Función Crear Ficha Médica de Salida

```

def editar_ficha_medica(request):
    url_ruc = request.GET['ruc']
    print('el ruc es:', url_ruc)
    fm = Ficha_medica.objects.get(ruc=url_ruc)
    if request.method == 'POST':
        datos = request.POST.dict()
        fm.ciiu = datos.get('ciiu')
        fm.nro_archivo = datos.get('nro_archivo')
        fm.establishcimiento_salud = datos.get('establishcimiento_salud')
        fm.institucion_sistema_nompreempresa = datos.get('institucion_sistema_o_nompre_empresa')
        fm.num_historia_clinica = datos.get('num_historia_clinica')
        fm.fecha_ingreso_trabajo = datos.get('fecha_ingreso_trabajo')
        fm.puesto_trabajo = datos.get('puesto_trabajo')
        fm.area_trabajo = datos.get('area_trabajo')
        fm.actividad_relevante = datos.get('actividad_relevante')

        fm.identidad_genero = datos.get('identidad_genero')
        fm.lateralidad = datos.get('actividad_relevante')
        if (fm.save() != False):
            messages.success(request, 'Paciente actualizado existosamente')
            print('Paciente actualizado existosamente')
            return redirect('historias_clinicas:listar_Paciente')
        else:

```

Figura 19. Función Editar Ficha Médica.

```

def certificado_medico_pdf(request):
    certificado_medico = Certificado_medicoForm(request.POST)
    url_ruc = request.GET['ruc']
    print('certificado medico:', url_ruc)
    paciente = Ficha_medica.objects.get(ruc=url_ruc)
    fecha = datetime.today().strftime('%Y-%m-%d')
    context = {
        'paciente': paciente,
        'fecha': fecha,
    }
    if request.method == 'POST':
        datos = request.POST.dict()
        crear_certificado = Certificado_medico()
        crear_certificado.ingreso = datos.get('ingreso')
        crear_certificado.fecha_emision = datos.get('fecha_emision')
        crear_certificado.periodico = datos.get('periodico')
        crear_certificado.reintegro = datos.get('reintegro')
        crear_certificado.retiro = datos.get('retiro')
        crear_certificado.apto = datos.get('apto')
        crear_certificado.apto_en_observacion = datos.get('apto_en_observacion')
        crear_certificado.apto_con_limitaciones = datos.get('apto_con_limitaciones')
        crear_certificado.no_apto = datos.get('no_apto')
        crear_certificado.detalle_observacion = datos.get('detalle_observacion')
        crear_certificado.codigo_medico = datos.get('codigo_medico')
        crear_certificado.nombre_apellido_medico = datos.get('nombre_apellido_medico')
        crear_certificado.descripcion_recomendacion = datos.get('descripcion_recomendacion')
        crear_certificado.realizacion_evaluacion_medica_si = datos.get('realizacion_evaluacion_medica_si')
        crear_certificado.realizacion_evaluacion_medica_no = datos.get('realizacion_evaluacion_medica_no')
        crear_certificado.condicion_presuntiva = datos.get('condicion_presuntiva')
        crear_certificado.condicion_definitiva = datos.get('condicion_definitiva')
        crear_certificado.condicion_no_aplica = datos.get('condicion_no_aplica')
        crear_certificado.condicion_relacionada_si = datos.get('condicion_relacionada_si')
        crear_certificado.condicion_relacionada_no = datos.get('condicion_relacionada_no')
        crear_certificado.condicion_relacionada_no_aplica = datos.get('condicion_relacionada_no_aplica')
        crear_certificado.ficha_medica = paciente
        #crear_certificado.ficha_medica = paciente
        if (crear_certificado.save() != False):
            messages.success(request, 'Certificado registrado correctamente')
            print('Certificado registrado existosamente')
            return redirect('historias_clinicas:listar_Paciente')
        # return redirect('/historias_clinicas/fichasmedicas/lista?id=' + url_ruc)
    return render(request, 'historias_clinicas/nueva_certificado.html', context)

```

Figura 20. Crear Certificado Médico

6.3. OBJETIVO 3: Evaluar el funcionamiento del software a través de pruebas aplicadas por la Dirección de Tecnologías de Información.

6.3.1. Elección de herramientas para la aplicación de pruebas del software desarrollado.

Para realizar las pruebas unitarias se eligió la herramienta Pytest, con la que se comprobó que cada uno de los métodos creados funcionan de manera correcta, obteniendo así resultados favorables, tal como se muestra en Figura 20-25


```

(ffercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
└─$ pytest test_metodos.py
===== test session starts =====
platform linux -- Python 3.10.5, pytest-7.1.2, pluggy-1.0.0
django: settings: ProyectoPruebas.settings (from env)
rootdir: /home/fercho/Imágenes/PROYECTO/ProyectoPruebas/ProyectoPruebas, configfile: pytest.ini
plugins: django-4.5.2, anyio-3.6.1
collected 1 item

test_metodos.py . [100%]

===== 1 passed in 1.05s =====

(ffercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
└─$

```

Figura 21. Resultado test_crear_ficha_medica().

```

(ffercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
└─$ pytest test_metodos.py
===== test session starts =====
platform linux -- Python 3.10.5, pytest-7.1.2, pluggy-1.0.0
django: settings: ProyectoPruebas.settings (from env)
rootdir: /home/fercho/Imágenes/PROYECTO/ProyectoPruebas/ProyectoPruebas, configfile: pytest.ini
plugins: django-4.5.2, anyio-3.6.1
collected 2 items

test_metodos.py .. [100%]

===== 2 passed in 1.04s =====

(ffercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
└─$

```

Figura 22. Resultado test_crear_ficha_medica_inicio().

```

(venv)-(fercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
└─$ pytest test_metodos.py
===== test session starts =====
platform linux -- Python 3.10.5, pytest-7.1.2, pluggy-1.0.0
django: settings: ProyectoPruebas.settings (from ini)
rootdir: /home/fercho/Videos/PROYECTO/ProyectoPruebas/ProyectoPruebas, configfile: pytest.ini
plugins: django-4.5.2, anyio-3.6.1
collected 3 items

test_metodos.py ... [100%]

===== 3 passed in 2.41s =====

(venv)-(fercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
└─$

```

Figura 23. Resultado test_crear_ficha_medica_reintegro().


```

(venv)-(fercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
$ pytest test_metodos.py
===== test session starts =====
platform linux -- Python 3.10.5, pytest-7.1.2, pluggy-1.0.0
django: settings: ProyectoPruebas.settings (from ini)
rootdir: /home/fercho/Videos/PROYECTO/ProyectoPruebas/ProyectoPruebas, configfile: pytest.ini
plugins: django-4.5.2, anyio-3.6.1
collected 4 items

test_metodos.py .... [100%]

===== 4 passed in 2.44s =====
(venv)-(fercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
$

```

Figura 24. Resultado test_crear_ficha_medica_salida().

```

(venv)-(fercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
$ pytest test_metodos.py
===== test session starts =====
platform linux -- Python 3.10.5, pytest-7.1.2, pluggy-1.0.0
django: settings: ProyectoPruebas.settings (from ini)
rootdir: /home/fercho/Videos/PROYECTO/ProyectoPruebas/ProyectoPruebas, configfile: pytest.ini
plugins: django-4.5.2, anyio-3.6.1
collected 5 items

test_metodos.py ..... [100%]

===== 5 passed in 2.41s =====
(venv)-(fercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
$

```

Figura 25. Resultado test_crear_ficha_medica_periodica().

```

(fercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/prueba/Pruebas]
$ pytest test_metodos.py
===== test session starts =====
platform linux -- Python 3.10.5, pytest-7.1.2, pluggy-1.0.0
django: settings: ProyectoPruebas.settings (from ini)
rootdir: /home/fercho/Videos/PROYECTO/ProyectoPruebas/ProyectoPruebas, configfile: pytest.ini
plugins: django-4.5.2, anyio-3.6.1
collected 6 items

test_metodos.py ..... [100%]

===== 6 passed in 2.44s =====
(fercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/prueba/Pruebas]
$

```

Figura 26. Resultado test_crear_hoja_evolucion().

```

(venv)-(fercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
$ pytest test_metodos.py
===== test session starts =====
platform linux -- Python 3.10.5, pytest-7.1.2, pluggy-1.0.0
django: settings: ProyectoPruebas.settings (from ini)
rootdir: /home/fercho/Videos/PROYECTO/ProyectoPruebas/ProyectoPruebas, configfile: pytest.ini
plugins: django-4.5.2, anyio-3.6.1
collected 7 items

test_metodos.py ..... [100%]

===== 7 passed in 2.64s =====
(venv)-(fercho@fercho)-[~/../ProyectoPruebas/ProyectoPruebas/pruebas_realizadas/Pruebas]
$

```

Figura 27. Resultado test_crear_certificado_medico().

6.3.2. Realizar pruebas conjuntamente con los usuarios que serán beneficiados con el software implementado.

Resultados tarea 2 Objetivo 3

7. Discusión

Para el desarrollo del trabajo de titulación se definieron 3 fases, las cuales corresponden a los objetivos planteados para dar cumplimiento al proyecto. Estas fases se detallan a continuación:

Objetivo 1: Analizar la situación actual del manejo de la información dentro de la Subdirección de Seguridad y Salud Ocupacional.

En la primera fase, correspondiente al primer objetivo, se aplicó la técnica de la entrevista que permitió conocer la situación en la que se encontraba la Subdirección de Seguridad y Salud Ocupacional, dado que el número de usuarios finales es bajo, esta técnica, dentro del proyecto, ayudó de mejor manera a obtener una amplia información en cuanto a las necesidades que se debían satisfacer al momento de construir el sistema. Además, se determinó que la falta de un sistema para la gestión de fichas médicas ha hecho que el proceso sea lento y toda la información se encuentre en riesgo de ser vulnerada. A partir de los resultados obtenidos de las entrevistas se realizó el documento de Especificación de Requerimientos de Software (ver Anexo IV), de acuerdo al formato establecido por la Dirección de Tecnologías de Información, en este documento se identificaron 17 requerimientos funcionales y 8 requerimientos no funcionales.

Objetivo 2: Desarrollar un software que facilite la sistematización de fichas médicas ocupacionales y test psicológicos.

Para la segunda fase se combinaron el marco de trabajo Scrum y la metodología ágil XP, obteniendo la metodología híbrida ScrumXP (SXP), para llevar a cabo el desarrollo del producto para la Sistematización de Fichas Médicas Ocupacionales. Las tareas y actividades se dividieron en 3 iteraciones, y de acuerdo a SXP, se componen de 3 fases que son Planificación-Definición, Desarrollo y Entrega y Mantenimiento, sin embargo, se debe tener en cuenta que la última fase de la metodología únicamente se cumple en la última iteración (ver Anexo X).

En la fase de Planificación-Definición se especificaron las historias de usuario de manera eficiente, siendo estas aprobadas por todos los participantes, especialmente por los clientes interesados en el sistema, así mismo, se realizó la documentación del Modelo de Arquitectura 4+1 en donde se incluyeron los diagramas necesarios para obtener una mejor visión de la composición del sistemas (ver Anexo V), cabe mencionar que, para este proyecto no fue indispensable la realización de dicho documento ya que los diagramas más relevantes fueron el diagrama de clases, diagrama de despliegue y el diagrama de componentes, sin embargo esta documentación sirve como base para fundamentar de mejor manera el desarrollo del TT.

Además, desde el punto de vista de los autores, la combinación del marco de trabajo Scrum y la metodología XP fue lo ideal para lograr desarrollar el sistema, esto debido a que permitió organizar las actividades, de acuerdo al tiempo de cada iteración y así realizar un buen trabajo por parte del equipo implicado en el presente TT.

Objetivo 3: Evaluar el funcionamiento del software a través de pruebas aplicadas por la Dirección de Tecnologías de Información.

En la tercera fase se realizaron los test de cada método implementado en el desarrollo del código fuente del producto, la herramienta seleccionada para la elaboración de las pruebas de software fue el framework de Pytest, permitiendo aplicar los test de cada método o funcionalidades creadas del módulo y así verificar que funcione correctamente, esto con la finalidad de validar que los usuarios de la Subdirección de Seguridad y Salud Ocupacional de la UNL al momento de interactuar con el sistema no se presenten inconvenientes. Los resultados obtenidos evidencian que se puede hacer uso del sistema sin fallos en las funcionalidades empleadas. (ver sección 6.3. OBJETIVO 3)

Valoración Técnica Económica Ambiental

El uso de recursos técnicos para el desarrollo del TT fue de gran importancia debido a que facilitaron varias tareas, optimizando así el tiempo. También se toma en cuenta la participación de diferentes actores que intervinieron para el desarrollo del proyecto.

En el aspecto ambiental, dado que existe gran número de funcionarios laborando dentro de la Universidad Nacional de Loja y que cada uno de ellos tiene registro de su ficha médica, al hacer uso del sistema para gestionar esta información, se reduce en gran medida el uso de papel. Por otro lado, al ser un sistema manejado solamente por el médico ocupacional la necesidad de recursos tecnológicos es mínima, por tanto, se puede evitar la utilización de varios equipos informáticos, los cuales emiten cantidades considerables de químicos dañinos para el medio ambiente.

Tabla 8. Valoración económica Recursos Humanos

Talento Humano	Nro. Horas	Valor por Hora	Valor Total
Autor 1 del TT	400	\$ 4.00	\$ 1,600.00
Autor 2 del TT	400	\$ 4.00	\$ 1,600.00
Directora TT	100	\$ 15.00	\$ 1,500.00
Total			\$ 4,700.00

Tabla 9. Valoración económica Recursos y Servicios

Recurso – Servicio	Nro. Meses	Cantidad	Valor Unitario	Valor Total
Servicio de Internet	6	2	\$ 30.00	\$ 360.00
Laptop	6	2	\$ 1000.00	\$ 2,000.00
Django	6	1	\$ 0.00	\$ 0.00
Pycharm	5	1	\$ 0.00	\$ 0.00
PostgreSQL	5	1	\$ 0.00	\$ 0.00
Total				\$ 2,360.00

Una vez estimados los valores económicos de los recursos, se obtiene una inversión total de \$ 7,766.00 para el desarrollo del TT, como se detalla en la Tabla 19:

Tabla 10. Costo Total del Trabajo de Titulación

Recurso	Subtotal
Recurso Humano	\$ 4,700.00
Recursos – Servicios	\$ 2,360.00
Subtotal	\$ 7,060.00
Imprevistos (10% del subtotal)	\$ 706.00
TOTAL	\$ 7,766.00

8. Conclusiones

Una vez culminado el desarrollo del Trabajo de Titulación podemos concluir lo siguiente:

- Al haber desarrollado el software para la Sistematización de Fichas Médicas Ocupacionales se logró optimizar el tiempo que empleaba el personal al hacer el registro de la información médica, ya que al hacer este proceso de forma manual se empleaba un tiempo bastante largo, en cambio al hacer uso del sistema, el tiempo se ha logrado reducir considerablemente para completar los datos, así mismo se reduce el riesgo de vulnerabilidad que tenía dicha información al guardarse de forma física.
- Mediante el uso de la metodología híbrida SXP, fue posible dividir el trabajo en tres iteraciones que tenían como objetivo desarrollar características comunes y correlacionadas, ayudar a organizar las tareas y centrarse más en el cumplimiento de los requisitos del software, ya que cada miembro del equipo involucrado proporcionó información y desde diferentes perspectivas, lo que permitió una validación constante.
- El diseño de diagramas permite obtener una visión más amplia de los requisitos del sistema antes de su codificación, logrando así, observar características que no fueron consideradas inicialmente y realizar las correcciones correspondientes para implementarlas dentro del sistema ya que fueron de importancia para cumplir con un producto de calidad.

9. Recomendaciones

Con la finalización del Trabajo de Titulación y todo el proceso realizado para su desarrollo, se recomienda lo siguiente:

- Emplear el sistema desarrollado de manera continua, para que así exista una mejora en cuanto a gestión de la información en el ámbito médico, ~~además de que~~ permitirá una buena optimización de tiempo ~~de cada usuario~~, ya que en sus ~~cargos~~ el tiempo es bastante limitado
- Aplicar la arquitectura 4+1, para proyectos de software, sólo con los artefactos más relevantes a explorar, ya que no es necesario desarrollar todos los diagramas que ofrece dicha arquitectura, porque existen diagramas que tienen un propósito similar a otros en cada una de las vistas, esto permitirá ahorrar tiempo y esfuerzo en el desarrollo del proyecto, sin embargo, se logrará el mismo resultado.
- **Al realizar un proyecto de software** dentro del ámbito educativo, aún más como trabajo de titulación, es aconsejable seguir e implementar metodologías ágiles, porque se adaptan mejor al tamaño de dichos proyectos, los plazos cortos y la retroalimentación constante que se necesita, incluso en algunos casos, se puede combinar dos o más metodologías, lo que aportará una mejor organización para desarrollar el proyecto.

9.1. Trabajos Futuros

- **Implementar más funcionalidades al sistema**, que tengan que ver con las tareas que realiza la Subdirección de Seguridad y Salud Ocupacional, y así ofrecer un producto que satisfaga de manera completa a esta subdirección, en todos sus ámbitos.
- **Implementar, en el sistema, la función que permitan obtener datos estadísticos sobre los pacientes atendidos por el Médico ocupacional, de acuerdo a un intervalo de tiempo determinado.**

10. Bibliografía

- [1] J. R. Sucuitana Gomero, "Implementación del Sistema Web SISEMO, para Optimizar el proceso de Evaluaciones Medicas Ocupacionales de La Clínica Servisalud-Serviperu, Lima 2018.," UNIVERSIDAD PRIVADA DEL NORTE, Lima, 2018.
- [2] A. S. Intriago Bayas and G. A. Valenzuela Franco, "IMPLEMENTACIÓN DE APLICACIÓN WEB PARA EL AGENDAMIENTO DE CITAS MÉDICAS, FICHAS OCUPACIONALES Y FACTURACIÓN ELECTRÓNICA PARA EL CENTRO MÉDICO GUAYAS OF3 DE LA CIUDAD DE GUAYAQUIL," UNIVERSIDAD POLITÉCNICA SALESIANA SEDE GUAYAQUIL, Guayaquil, 2019.
- [3] R. Ramos Muñoz, "Fichas ocupacionales: Ibermutuamur," *Med. Secur. Trab. (Madr)*., vol. 60, pp. 98–109, 2014, Accessed: May 31, 2022. [Online]. Available: https://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S0465-546X2014000500016.
- [4] S. D. Montenegro, J. M. Almada, and A. R. Palejko, "Beneficios de una Historia Clínica Electrónica en el ámbito de la Salud Ocupacional," 2020, Accessed: May 11, 2022. [Online]. Available: http://sedici.unlp.edu.ar/bitstream/handle/10915/115778/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y.
- [5] A. J. Ramos Martín-Vegue, J. L. Vázquez-Barquero, and S. Herrea Castanedo, "CIE-10 (I): Introducción, historia y estructura general," *Pápeles Médicos*, vol. 11, no. 1, pp. 24–35, 2002.
- [6] M. Almagro, R. Martínez, V. Fresno, and S. Montalvo, "Estudio preliminar de la anotación automática de códigos CIE-10 en informes de alta hospitalarios - Preliminary Study of the Automatic Annotation of Hospital Discharge Report with ICD-10 codes," *Proces. Leng. Nat.*, 2018, doi: 10.26342/2018-60-5.
- [7] E. Oz, *Administración de los Sistemas de Información*, 5th ed. 2008.
- [8] P. G. Padilla Huamantincó, "SISTEMATIZACIÓN DE LA REFORMA DE LOS SISTEMAS DE INFORMACIÓN EN SALUD DEL MINISTERIO DE SALUD DEL PERÚ (MINSA) EN EL PERIODO 2016-2017," 2018.
- [9] O. Jara, H. Cep, C. De Estudios, P. Alforja, and C. Rica, "DILEMAS Y DESAFIOS DE LA SISTEMATIZACION DE EXPERIENCIAS 1."
- [10] X. López, *Sistemas digitales de información*. 2006.
- [11] S. R. Gómez Palomo and E. Gil Moraleda, *Aproximación a la ingeniería del software*. Centro de Estudios Ramon Areces S.A., 2020.
- [12] P. Kruchten, "Planos Arquitectónicos: El Modelo de '4+1' Vistas de la Arquitectura del Software *."
- [13] O. Gómez, "Documentando la arquitectura de software," Accessed: Jun. 27, 2022. [Online]. Available: www.osgg.net.
- [14] D. Ghimire, "Comparative study on Python web frameworks: Flask and Django," May 2020.
- [15] D. Rubio, "Introduction to the Django Framework," *Begin. Django*, pp. 1–29, 2017, doi: 10.1007/978-1-4842-2787-9_1.
- [16] O. Cervantes, D. Báez López, and E. Castillo Juárez, *Python con aplicaciones a las matemáticas, ingeniería y finanzas*. 2017.
- [17] J. C. García Monsálvez, "Python como primer lenguaje de programación textual en la

Enseñanza Secundaria = Python as First Textual Programming Language in Secondary Education,” *Python como Prim. Leng. Program. textual en la Enseñanza Secund. = Python as First Textual Program. Lang. Second. Educ.*, vol. 18, no. 2, pp. 147–162, Aug. 2017, doi: 10.14201/EKS2017182147162.

- [18] Thomas Lockhart, “Tutorial de PostgreSQL,” *Development*, p. 1, 2014.
- [19] “PostgreSQL: The world’s most advanced open source database.” <https://www.postgresql.org/> (accessed May 12, 2022).
- [20] M. P. Zea Ordóñez, J. R. Molina Ríos, and F. F. Redrován Castillo, *ADMINISTRACIÓN DE BASES DE DATOS CON POSTGRESQL*, vol. 19. 2017.
- [21] E. G. Maida and J. Pacienza, “Metodologías de desarrollo de software,” 2015, Accessed: May 10, 2022. [Online]. Available: <https://repositorio.uca.edu.ar/bitstream/123456789/522/1/metodologias-desarrollo-software.pdf>.
- [22] A. Menzinsky, G. López, and J. Palacio, “Scrum Manager Guía de formación,” 2016. <https://openlibra.com/es/book/download/scrum-manager-guia-de-formacion> (accessed May 04, 2022).
- [23] A. Menzinsky, C. Ruata, J. Palacio, M. Ariza, N. Monzón, and R. García, “Scrum Level,” 2016, Accessed: May 09, 2022. [Online]. Available: <https://openlibra.com/es/book/download/scrum-level-guia-para-evaluacion-de-la-agilidad>.
- [24] E. Bahir, *Scrum y eXtreme Programming para programadores*. 2012.
- [25] H. Baumeister and H. Lichter, “Agile Processes in Software Engineering and Extreme Programming,” 2017. <https://openlibra.com/es/book/download/agile-processes-in-software-engineering-and-extreme-programming> (accessed May 04, 2022).
- [26] C. J. Cárdenas Tuttillo and E. F. Quimbata Quingaluiza, “Análisis, Diseño y Construcción de un Prototipo de una Red Social orientada a la Seguridad para la Empresa CEFOSEG,” 2017.
- [27] Y. Bagarotti, A. Abad, and Y. Arias, “Experiencias durante la gestión de la calidad en proyectos que usan metodologías ágiles,” *Ing. UC*, 2013, Accessed: May 11, 2022. [Online]. Available: <https://www.redalyc.org/pdf/707/70732641006.pdf>.
- [28] D. L. Huertano Ávila and L. E. Plazas Jiménez, “DESARROLLO DE UN SUB-MÓDULO DE HISTORIA CLÍNICA LABORAL DENTRO DEL ERP PARA LA EMPRESA SRF CONSULTORES,” UNIVERSIDAD ANTONIO NARIÑO, Bogotá, 2021.
- [29] Y. Benites, K. Albújar, E. Arámbulo, J. Mantilla, and D. Torres, “ANÁLISIS Y DISEÑO DE PROTOTIPO DE SOFTWARE PARA LA AUTOMATIZACIÓN DE HISTORIAS CLÍNICAS DEL POLICLÍNICO UDEP,” 2017.
- [30] A. C. Vicuña Pilataxi and Á. J. Caranqui Corella, “AUTOMATIZACIÓN DEL PROCESO DE SALUD OCUPACIONAL DEL DISTRITO 17D04 ‘PUENGASÍ A ITCHIMBÍA,’” UNIVERSIDAD POLITÉCNICA SALESIANA SEDE QUITO, Quito, 2018.
- [31] M. N. Cuyo Montes and W. J. Quimbata Molina, “SISTEMA DE FICHAS DE CONTROL MÉDICO OCUPACIONAL Y PERIÓDICO PARA EL ANÁLISIS DE RIESGO LABORAL ACUMULADO DE LOS DOCENTES Y SERVIDORES DE LA UNIVERSIDAD TÉCNICA DE COTOPAXI,” UNIVERSIDAD TÉCNICA DE COTOPAXI, Latacunga, 2018.
- [32] H. Feria Avila, M. Matilla González, and S. Mantecón Licea, “LA ENTREVISTA Y LA ENCUESTA: ¿MÉTODOS O TÉCNICAS DE INDAGACIÓN EMPÍRICA?,” *Didasc@lia*

didáctica y Educ. ISSN 2224-2643, vol. 11, no. 3, pp. 62–79, Sep. 2020, Accessed:
Aug. 28, 2022. [Online]. Available:
<https://revistas.ult.edu.cu/index.php/didascalia/article/view/992>.

11. Anexos

Anexo I. Entrevista 1

Entrevista 1

Entrevistado: Dr. Diego Falconi Espinoza

Cargo: Subdirector de Seguridad y Salud

Día: 18/01/2022

Hora: 12H30

Duración: 9 minutos

Lugar de entrevista: Plataforma ZOOM

PREGUNTAS:

1. ¿Está de acuerdo en que se debe hacer una inspección de seguridad dentro del departamento de salud ocupacional para prevenir los riesgos que pueden presentarse?

Es muy importante que nosotros permanentemente estemos realizando inspecciones de seguridad en los diferentes ámbitos de trabajo de las personas y además de eso observaciones de seguridad a las personas en el cumplimiento de las normas de seguridad y una vigilancia de la salud a la comunidad universitaria con protocolos estrictos, y desarrollar procedimientos protocolos o mecanismos que permitan mantener una vida saludable dentro del transcurso de la vida de trabajo de una persona dentro de la Institución.

2. ¿Usted considera que se tiene conocimientos amplios en cuanto a Seguridad de la Información?

Es un tema que tenemos que manejarlo desde el punto de vista informático. Aquí manejamos datos confidenciales, en el tema de salud, con respecto a patologías o presencia de enfermedades de los docentes, servidores y trabajadores. Ahí si necesitamos hacer un control de la información de acuerdo a lo que establece el acuerdo 14.04 del Ministerio de Salud.

3. ¿Considera que es importante saber sobre el concepto de seguridad y para el desempeño de sus labores dentro del departamento?

Sí, desde luego, nosotros no podemos hacer seguridad sin tener conocimientos, si hablamos de seguridad informática el tema va por el lado, solamente de la situación médica de las personas, ahí es donde necesitamos cuidar esa información.

4. ¿Conoce cuáles son los factores de riesgo que se toman en cuenta dentro de un sistema de información en el departamento de salud ocupacional?

Si hablamos de factores de riesgo propiamente dichos en parte de Seguridad y Salud Ocupacional, los factores de riesgo son seis. Pero si hablamos de factores de riesgo en la Seguridad Informática desconozco mucho en particular.

5. ¿Cree usted que sería beneficiosa la implementación de un Plan de Gestión de Riesgos en el Departamento de Salud Ocupacional?

Nosotros tenemos un Plan de gestión de riesgos institucional, que se refiere a los factores de riesgos en el ámbito laboral. Si hablamos del tema Seguridad informática es necesario manejar un sistema de información, primero desarrollar un sistema de registro de fichas médicas ocupacionales, que es el tema de seguridad informática, y después buscar las formas y métodos convenientes para poder implementar las seguridades y que esta información no sea intervenida, hackeada, no sea difundida mal intencionadamente por otras personas.

Link de acceso a video:

<https://drive.google.com/file/d/1fqQxZ7s1wfYgLy9UmPe4P1WC7bhdVTso/view?usp=sharing>

Anexo II. Entrevista 2

Entrevista 2

Entrevistado: Dr. Diego Falconí Espinoza

Día: 21/01/2022

Hora: 12H30

Duración: 13 minutos

Lugar de entrevista: Plataforma ZOOM

PREGUNTAS:

1. ¿Actualmente cuenta con un Sistema que le permita guardar la información dentro del Departamento de Salud Ocupacional?

No contamos con un sistema que nos permita guardar información de las actividades que se realiza dentro de la unidad, solamente contamos con un software para análisis de riesgos ergonómicos.

2. ¿Considera usted que un Sistema de Información para el departamento es necesario para un mejor control de la información?

Definitivamente si es necesario, para ir alimentando todas las actividades que nosotros realizamos, principalmente dentro del área médica, que es donde tenemos que guardar información delicada y se la debe tener de fácil disponibilidad en caso de emergencia de las personas que laboran en la institución, es decir, docentes, servidores y trabajadores.

3. ¿Qué tipo de información requiere guardar en el Sistema?

Información referente a sistema de gestión de seguridad, que tiene que ver con inspecciones de seguridad, con capacitaciones de seguridad, entregas de equipo de protección, procesos internos de compras, subastas, que realizamos dentro de la unidad. Por otro lado, en la parte médica, la información que tiene que ver con las fichas médicas ocupacionales, en donde se guarda datos básicos y la historia médica laboral de la persona que ingresa y se mantiene laborando y posteriormente sale de la institución.

4. ¿Qué beneficios estima que se obtendría al implementar un Sistema?

En el área médica, tener una información oportuna de las condiciones de salud de las personas, actualmente el médico revisa la historia clínica laboral en una hoja Excel, pero muchas veces la hoja Excel es muy limitada o los conocimientos no están profundizados en el manejo de la herramienta Excel, por lo tanto, limita su accionar. Al tener un software, podríamos almacenar toda la información de la persona en una sola digitación, se digitaría el número de cedula y obtendríamos todos los datos y su historial médico.

5. ¿Cree usted conveniente que un Sistema Propuesto debe ser amigable con el usuario?

Definitivamente tiene que ser amigable, de lo contrario no mejoraría la situación actual.

Link de acceso a video:

<https://drive.google.com/file/d/1fqQxZ7s1wfYgLy9UmPe4P1WC7bhdVTso/view?usp=sharing>

Anexo III. Entrevista 3

Entrevista 3

Entrevistada: Dra. Iliana Figueroa

Día: 19/05/2022

Hora: 09H30

Duración: 8 minutos

Lugar de entrevista: Subdirección de Seguridad y Salud Ocupacional

PREGUNTAS:

1. ¿Cómo lleva el registro de las fichas médicas ocupacionales y test psicológicos de los pacientes?

Tenemos dos modalidades, la modalidad digital y la modalidad de empresa, si un registro o institución y de todo el país toda Historia Clínica Ocupacional debe permanecer por el lapso de veinte años la ficha física, así el colaborador se haya jubilado o dejado sus funciones o haya salido de la Institución. El Ministerio de Trabajo nos pide la ficha impresa, nos dan un nombre cualquiera que están en los registros, la ficha o historia clínica impresa que nos solicitan, por eso nosotros pedíamos como requerimiento lo que ustedes nos pueden ayudar a hacer en digital para tener un archivo de data, para tenerlo claro como una alternativa de solución manejarlo como el Ministerio de Trabajo, si nos piden alguna información algún colaborador, entonces digitamos y subimos la ficha, ahí es más preciso y más exacto.

2. ¿Qué datos son necesarios para hacer el registro inicial de una ficha médica ocupacional?

La Ficha Historia Clínica de Inicio, tenemos datos puntuales o dentro de los ejes como son la ubicación carga horaria, los riesgos que va a estar sometido la persona, que actividades va a cumplir, qué actividades va a desarrollar ante la institución, y toda actividad tienen riesgos, cada uno de los riesgos son fundamentales, los que se conocen entre ellos: Físicos, Químicos, Mecánicos, los Psicosociales, se debe conocer situaciones personales como antecedentes, situaciones laborales, es necesario conocer los tres últimos empleos que haya tenido la persona para tener una idea clara para ver si tuvo o no tuvo accidentes o incidentes en los anteriores trabajos, tener cuanto tiempo duro en cada empresa, si trabaja en esta empresa o no trabaja, si está apto para desarrollar las funciones o si necesita algún tratamiento médico para solucionar la emergencia al momento de cuando ingresa, entonces yo tengo por ejemplo: apto su descripción, apto como observación, entonces yo decido cómo le voy a poner en el historial médico de acuerdo a la ficha que yo he llenado.

3. ¿Qué información se guarda cuando un paciente acude a una consulta?

Primero, la sintomatología presentada en esa situación, si es que se automedico o ha utilizado por algún otro profesional anteriormente si es que lo ha estado, y las novedades que los síntomas han avanzado, aminoraron, se han hecho, se han incrementado y sobre todo qué novedades tiene como antecedentes.

4. ¿Existen diferentes tipos de fichas médicas? ¿Cuáles serían estos tipos?

Fichas médicas de inicio lo que pide la ley, Fichas médicas periódicas, Fichas médicas de salida y las Fichas médicas de reintegro, porque hay personas que se van por un largo periodo de tiempo, salen por alguna situación por más de quince días.

5. En el caso de los test psicológicos, ¿cómo es el registro de los pacientes o para qué sirve?

Hay diferentes situaciones para lo cual se debe aplicar Test Psicológicos, queríamos ver si estandarizamos en la institución para todo el personal porque yo hasta ahora se venía haciendo los test psicológicos para determinados grupos de personas que generalmente hasta la presente fecha y aplicado el test de Lucy Reidl que es para el autoestima, el larga familiar para ver como está el comportamiento de la Familia, el test de Beck para la depresión, dependiendo de las condiciones o adicciones podemos aplicar el test de alcohol dependiendo de los pacientes, haciendo la historia clínica, que novedades tenemos, que eventualidades presenta en base a ello se le hace la variante si le hace algún otro test.

6. En caso de que un paciente tenga una enfermedad, ¿cómo se lleva a cabo el seguimiento de su evolución y tratamiento?

Cuando se atiende a un paciente se le da dos días de reposo o descanso, para su reintegro tiene que venir por aquí, en otros de los casos cuando no se da permisos médicos se le ve como se le hace un seguimiento para que venga en las próximas 48 horas, porque siempre la efectividad de las medicinas siempre hay que considerar las 48 a 72 horas de haber sido administradas.

7. ¿Quién tiene acceso a la información de las fichas médicas ocupacionales y test psicológicos?

En este caso solamente mi persona, esto lo que dice en la 584 en el tema de Confidencialidad; salvo que el paciente o colaborador autorice se puede permitir talento humano si hay alguna demanda de alguna cosa y sepan de ahí es el tema de confidencialidad. El ingreso de la información sería para mi persona, salvo para ciertas condiciones como un accidente ahí sí les pediría que ingrese también el técnico el Ing. Alberto Coronel y también el jefe, porque los

accidentes son del reporte del IESS, ellos también deben conocer la situación y sobre todo para un acceso y manejar estos aplicativos, por eso es que yo les decía cómo tener ahí en cuatro ejes para poder ver y verificar, en el eje de historias clínicas yo tendría ingreso y en las otras tres era para compartir los accesos.

8. ¿Qué tipos de certificados se emiten para un paciente?

Certificado de Aptitud Médica

Certificado de Ausencias

9. ¿Cuáles son los motivos por los que se emite un certificado?

Tenemos el Certificado de Aptitud Médica que se suele manejar, Certificado de Ausencias son aquellos con los que dependiendo de la patología como certificado médico de empresa solo se puede dar tres días de permiso no más.

Link de acceso a audio:

https://drive.google.com/file/d/174hiRemb9kT16Kj_Lp3XTtmlAz4-kpXv/view?usp=sharing

Anexo IV. Documento de Especificación de Requisitos de Software

Proyecto:

Sistematización de Fichas Médicas Ocupacionales y Test Psicológicos

Especificación de Requisitos de Software

<https://drive.google.com/file/d/1m6akHfOVJLIqWKRbm9aFm2sLHBEvFfI7/view?usp=sharing>

Anexo V. Documento de Arquitectura de Software

Proyecto:

Sistematización de Fichas Médicas Ocupacionales y Test Psicológicos

Documento de Arquitectura de Software

1. Introducción

Dado que la arquitectura es la parte más importante dentro del desarrollo de software, la siguiente documentación detalla las vistas que componen esta arquitectura del modelo 4+1, siendo estos: vista lógica, vista de despliegue, vista de escenarios, vista física y vista de procesos. A través de esta arquitectura se obtiene una visión más clara y específica de los procesos y funciones que se implementaron en el sistema.

2. Propósito

Este documento pretende describir de forma completa el diseño de la arquitectura del modelo 4+1 de Kruchten, que detalla la lógica y el comportamiento del sistema que será desarrollado.

3. Alcance

A través del presente documento se describirá la arquitectura de software que fue utilizada para el desarrollo del sistema, esto permitirá entender de mejor manera el funcionamiento del mismo, a través de la arquitectura 4+1, misma que comprende la vista lógica, vista de despliegue, vista de escenarios, vista física y vista de procesos.

4. Referencias

Tabla 11. Referencias Arquitectura De Software

Referencia	Título del Documento
Anexo de especificación	Especificación de Requerimientos de Software
Modelo	Arquitectura de software [12]

5. Vista Global

En este documento se muestra de forma organizada la arquitectura empleada para el desarrollo del sistema; esta arquitectura 4 + 1 describe al software por medio de vistas, las cuales son:

- **Vista Lógica:** Comprende la estructura y funcionalidades del sistema y las prestaciones o servicios que va a ofrecer como el manejo de datos, del servicio web API-REST.
- **Vista de Despliegue:** Muestra cómo está dividido el software en componentes y sus dependencias.
- **Vista de Escenarios:** Representada por los casos de uso, une y relaciona la vista lógica, de despliegue, física y de procesos.
- **Vista Física:** Se muestran los componentes físicos que interactúan en el sistema.

- **Vista de Procesos:** Muestra los procesos y la forma en que estos se comunican, dentro del sistema.

6. Representación de la Arquitectura

La arquitectura del sistema está construida a partir de las vistas del modelo 4 + 1 de Kruchten, basada en UML, a continuación, se describe las 5 vistas con sus respectivos diagramas.

Tabla 12. Vistas del Modelo 4 + 1

Vista	Elemento Modelado	Descripción
Vista de Escenarios	Casos de Uso	Interacción entre los stakeholders y el sistema.
Vista Lógica	Modelo Conceptual Diagrama de Clases	Servicios que brinda el sistema a sus stakeholders.
Vista de Procesos	Diagrama de Actividades	Indica los procesos que realizan las funcionalidades del sistema.
Vista de Despliegue	Diagrama de Componentes	Detalla la relación de los componentes del sistema.
Vista Física	Diagrama de Despliegue	Muestra los componentes físicos del sistema y su conexión.

7. Objetivos de la Arquitectura

El sistema debe cumplir con los siguientes objetivos:

- **Calidad:** El sistema debe cumplir con todo lo establecido en el documento de Especificación de Requisitos.
- **Rendimiento:** Las peticiones al servidor deberán cumplirse de forma eficiente.
- **Disponibilidad:** El sistema estará disponible las 24 horas, 365 días del año, exceptuando días en que entrará en mantenimiento.
- **Portabilidad:** El sistema será una aplicación web, que estará disponible en todas las plataformas (Navegador Web).

8. Vista de Escenarios

En esta sección se muestran los diagramas de caso de uso, en los cuales se indica la interacción entre los stakeholders y el sistema.

8.1. Diagrama de Casos de Uso

Los actores principales, como se muestra en el diagrama, son Administrador de Salud Ocupacional y Médico, cada uno de ellos realiza diferentes funciones que serán atendidas en el sistema a desarrollar.

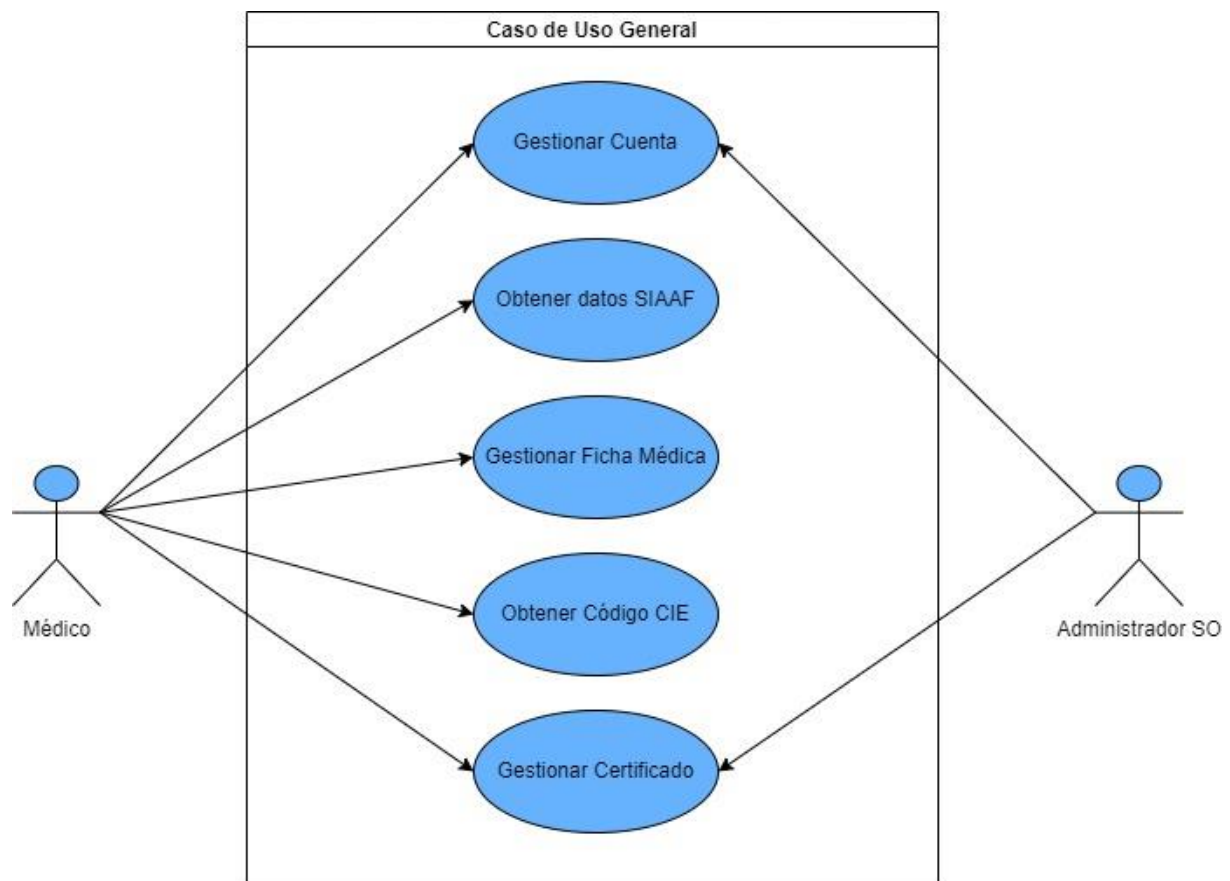


Figura 28. Diagrama de Caso de Uso General

8.2. Especificación de Casos de Uso

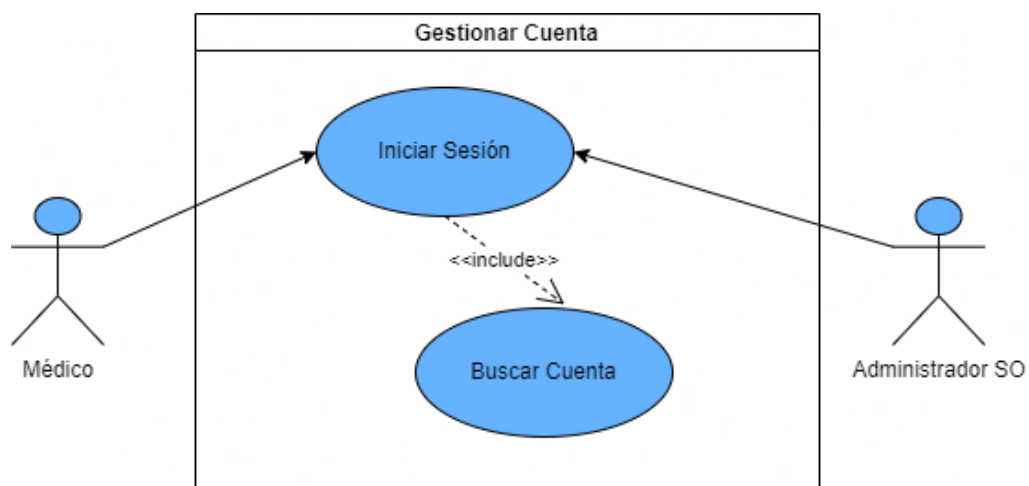


Figura 29. Caso de Uso Gestionar Cuenta

Tabla 13. Descripción del Caso de Uso Iniciar Sesión

Nombre:		Iniciar Sesión	
Descripción: El UC permite al Administrador y Médico iniciar sesión y acceder al sistema.			
Actores: Administrador de Salud Ocupacional Médico			
Precondiciones: Estar registrado en el sistema.			
FLUJO NORMAL			
Acciones del Usuario		Acciones del Sistema	
1. Accede a la ventana de Inicio de Sesión.		2. Muestra el formulario con los campos.	
3. Ingresa las credenciales institucionales (Correo y Contraseña)			
4. Presiona botón [Iniciar Sesión]		5. Verifica las credenciales ingresadas	
		6. Busca los datos en la BD	
		7. Redirecciona a la pantalla principal	
		8. Fin de UC	
FLUJO ALTERNATIVO			
		A. Credenciales Incorrectas	
		A.6. Presenta mensaje de error “Las credenciales son incorrectas”.	
		A.7. El UC continúa en el paso 3 del flujo normal de eventos	
		A.8. Fin de UC	
Observación:			

Tabla 14. Descripción del Caso de Uso Obtener datos SIAAF

Nombre:		Obtener datos SIAAF
Descripción: El UC permite al Médico obtener los datos personales del paciente, registrados en la base de datos del SIAAF.		
Actores: Médico		
Precondiciones: Haber iniciado sesión, como Médico, en el sistema.		
FLUJO NORMAL		
Acciones del Usuario	Acciones del Sistema	
1. Presiona botón Fichas Médicas	2. Muestra lista de funcionarios registrados en el SIAAF	
3. Ingresa un número de cédula en el campo de búsqueda		
4. Presiona botón [Buscar]	5. Verifica el dato ingresado	
	6. Busca los datos en la BD del SIAAF	

	7. Retorna los datos personales de paciente
	8. Fin de UC
FLUJO ALTERNATIVO	
A. Dato ingresado no existe.	
	A.6. Presenta mensaje de error “Número de cédula no existe”
	A.7. El UC continúa en el paso 3 del flujo normal de eventos.
	A.8. Fin de UC
Observación:	

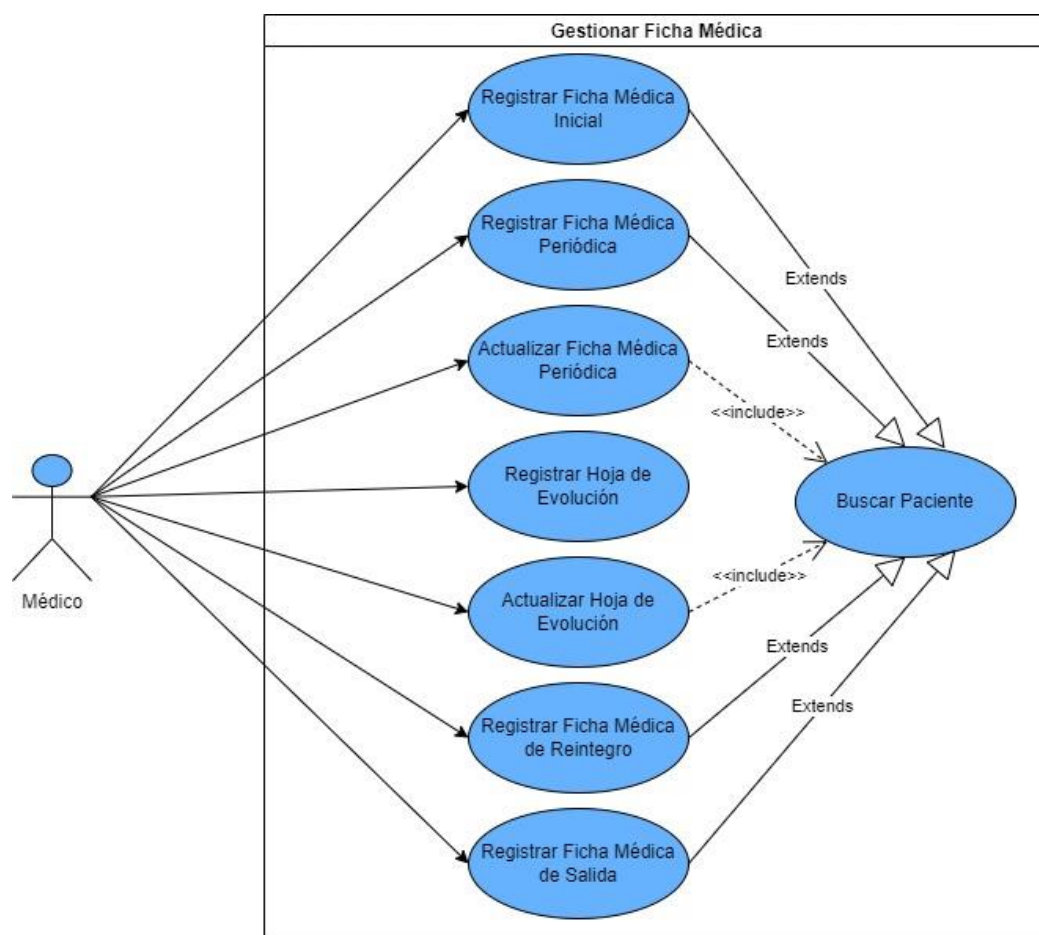


Figura 30. Caso de Uso Gestionar Ficha Médica

Tabla 15. Descripción del Caso de Uso Registrar Ficha Médica Inicial

Nombre:	Registrar Ficha Médica Inicial
Descripción: El UC permite al Médico registrar la ficha inicial de un paciente.	
Actores: Médico	

Precondiciones: Haber iniciado sesión, como Médico, en el sistema. Seleccionar un paciente.	
FLUJO NORMAL	
Acciones del Usuario	Acciones del Sistema
1. Accede a la ventana de Fichas Médicas.	2. Muestra las opciones de Ficha Médica
3. Selecciona la opción “Ficha Inicial”	4. Muestra el formulario de ficha inicial
5. Ingresa los datos requeridos en cada campo (ver todos los datos en 11.6. Anexo VI)	
6. Presiona el botón [Guardar]	7. Verifica que los datos ingresados estén correctos
	8. Muestra mensaje “Ficha Inicial registrada correctamente”
	9. Redirecciona a la pantalla principal
	10. Fin de UC
FLUJO ALTERNATIVO	
A. Datos incompletos	
A.6. Ingresa datos requeridos incompletos	A.7. Presenta mensaje de error “Complete los campos correctamente”
	A.8. El UC continúa en el paso 5 del flujo normal de eventos
	A.9. Fin de UC
B. Cancela registro	
B.7. Presiona botón [Cancelar]	B.8. Muestra mensaje “¿Seguro que desea cancelar registro?”
B.9. Presiona botón [Aceptar]	B.10. Redirecciona a la pantalla principal
	B.11. Fin de UC
Observación:	

Tabla 16. Descripción del Caso de Uso Registrar Ficha Médica Periódica

Nombre:	Registrar Ficha Médica Periódica
Descripción: El UC permite al Médico registrar los datos médicos de un paciente en la ficha periódica.	
Actores: Médico	
Precondiciones: Haber iniciado sesión, como Médico, en el sistema. Haber seleccionado un paciente.	
FLUJO NORMAL	
Acciones del Usuario	Acciones del Sistema
1. Accede a la ventana de Fichas Médicas.	2. Muestra las opciones de Ficha Médica
3. Selecciona la opción “Ficha Periódica”	4. Muestra el formulario de ficha periódica

5. Ingresa los datos requeridos en cada campo (ver todos los datos en 11.6. Anexo VI)	
6. Presiona el botón [Guardar]	7. Verifica que los datos ingresados estén correctos
	8. Muestra mensaje “Ficha Periódica guardada correctamente”
	9. Redirecciona a la pantalla principal
	10. Fin de UC
FLUJO ALTERNATIVO	
A. Datos incompletos	
A.6. Ingresa datos requeridos incompletos	A.7. Presenta mensaje de error “Complete los campos correctamente”
	A.8. El UC continúa en el paso 5 del flujo normal de eventos
	A.9. Fin de UC
B. Cancela registro	
B.7. Presiona botón [Cancelar]	B.8. Muestra mensaje “¿Seguro que desea cancelar registro?”
B.9. Presiona botón [Aceptar]	B.10. Redirecciona a la pantalla principal
	B.11. Fin de UC
Observación:	

Tabla 17. Descripción del Caso de Uso Actualizar Ficha Médica Periódica

Nombre:	Actualizar Ficha Médica Periódica
Descripción: El UC permite al Médico actualizar los datos médicos de la ficha periódica.	
Actores: Médico	
Precondiciones: Haber iniciado sesión, como Médico, en el sistema. Haber buscado un paciente.	
FLUJO NORMAL	
Acciones del Usuario	Acciones del Sistema
1. Accede a la ventana de Fichas Médicas	2. Carga las opciones de fichas médicas del paciente
3. Selecciona la opción “Ficha Periódica”	4. Muestra la ficha periódica con los datos cargados
5. Ingresa los nuevos datos en cada campo del formulario (ver todos los datos en 11.6. Anexo VI)	
6. Presiona el botón [Guardar]	7. Verifica que los datos ingresados estén correctos
	8. Muestra mensaje “Ficha Periódica actualizada correctamente”

	9. Redirecciona a la pantalla principal
	10. Fin de UC
FLUJO ALTERNATIVO	
A. Datos incorrectos	
A.6. Ingresar datos requeridos incompletos	A.7. Presenta mensaje de error "Complete los campos correctamente"
	A.8. El UC continúa en el paso 5 del flujo normal de eventos
	A.9. Fin de UC
B. Cancela actualización	
B.7. Presiona botón [Cancelar]	B.8. Muestra mensaje "¿Seguro que desea cancelar actualización?"
B.9. Presiona botón [Aceptar]	B.10. Redirecciona a la pantalla principal
	B.11. Fin de UC
Observación:	

Tabla 18. Descripción del Caso de Uso Registrar Hoja de Evolución

Nombre:	Registrar Hoja de Evolución
Descripción: El UC permite al Médico registrar la evolución e indicaciones para un paciente.	
Actores: Médico	
Precondiciones: Haber iniciado sesión, como Médico, en el sistema. Estar en la ventana de Ficha Periódica.	
FLUJO NORMAL	
Acciones del Usuario	Acciones del Sistema
1. Selecciona la opción "Hoja de Evolución"	2. Muestra el formulario de hoja de evolución
3. Ingresar los datos en los campos correspondientes (Fecha, Evolución, Indicaciones)	
5. Presiona el botón [Guardar]	6. Verifica que los datos ingresados estén correctos
	7. Muestra mensaje "Hoja de Evolución registrada correctamente"
	8. Redirecciona a la pantalla principal
	9. Fin de UC
FLUJO ALTERNATIVO	
A. Datos incompletos	
A.6. Ingresar datos requeridos incompletos	A.7. Presenta mensaje de error "Complete los campos correctamente"
	A.8. El UC continúa en el paso 5 del flujo normal de eventos
	A.9. Fin de UC
B. Cancela registro	

B.7. Presiona botón [Cancelar]	B.8. Muestra mensaje “¿Seguro que desea cancelar registro?”
B.9. Presiona botón [Aceptar]	B.10. Redirecciona a la pantalla principal
	B.11. Fin de UC
Observación:	

Tabla 19. Descripción del Caso de Uso Actualizar Hoja de Evolución

Nombre:		Actualizar Hoja de Evolución
Descripción: El UC permite al Médico actualizar la información de la hoja de evolución de un paciente.		
Actores: Médico		
Precondiciones: Haber iniciado sesión, como Médico, en el sistema. Haber buscado las fichas médicas del paciente.		
FLUJO NORMAL		
Acciones del Usuario		Acciones del Sistema
1. Selecciona la opción “Ficha Periódica”		2. Muestra la ficha periódica con los datos cargados
3. Selecciona la opción “Hoja de Evolución”		4. Retorna la información en el formulario de hoja de evolución
5. Ingresar los nuevos datos en cada campo del formulario (Fecha, Evolución, Indicaciones)		
6. Presiona el botón [Guardar]		7. Verifica que los datos ingresados estén correctos
		8. Muestra mensaje “Hoja de Evolución actualizada correctamente”
		9. Redirecciona a la pantalla principal
		10. Fin de UC
FLUJO ALTERNATIVO		
A. Datos incorrectos		
A.6. Ingresar datos requeridos incompletos		A.7. Presenta mensaje de error “Complete los campos correctamente”
		A.8. El UC continúa en el paso 5 del flujo normal de eventos
		A.9. Fin de UC
B. Cancela actualización		
B.7. Presiona botón [Cancelar]		B.8. Muestra mensaje “¿Seguro que desea cancelar actualización?”
B.9. Presiona botón [Aceptar]		B.10. Redirecciona a la pantalla principal
		B.11. Fin de UC
Observación:		

Tabla 20. Descripción del Caso de Uso Registrar Ficha Médica de Reintegro

Nombre:		Registrar Ficha Médica de Reintegro
Descripción: El UC permite al Médico registrar la ficha de reintegro de un paciente.		
Actores: Médico		
Precondiciones: Haber iniciado sesión, como Médico, en el sistema. Haber seleccionado un paciente.		
FLUJO NORMAL		
Acciones del Usuario		Acciones del Sistema
1. Accede a la ventana de Fichas Médicas.		2. Muestra las opciones de Ficha Médica
3. Selecciona la opción “Ficha de Reintegro”		4. Muestra el formulario de ficha de reintegro
5. Ingresa los datos requeridos en cada campo (ver los datos en 11.6. Anexo VI)		
6. Presiona el botón [Guardar]		7. Verifica que los datos ingresados estén correctos
		8. Muestra mensaje “Ficha de Reintegro guardada correctamente”
		9. Redirecciona a la pantalla principal
		10. Fin de UC
FLUJO ALTERNATIVO		
A. Datos incompletos		
A.6. Ingresa datos requeridos incompletos		A.7. Presenta mensaje de error “Complete los campos correctamente”
		A.8. El UC continúa en el paso 5 del flujo normal de eventos
		A.9. Fin de UC
B. Cancela registro		
B.7. Presiona botón [Cancelar]		B.8. Muestra mensaje “¿Seguro que desea cancelar registro?”
B.9. Presiona botón [Aceptar]		B.10. Redirecciona a la pantalla principal
		B.11. Fin de UC
Observación: El registro de la ficha de reintegro se realiza, únicamente, en el caso que el paciente se ha ausentado, de su puesto de trabajo, por más de 15 días		

Tabla 21. Descripción del Caso de Uso Registrar Ficha Médica de Salida

Nombre:		Registrar Ficha Médica de Salida
Descripción: El UC permite al Médico registrar la ficha de salida de un paciente		
Actores: Médico		
Precondiciones: Haber iniciado sesión, como Médico, en el sistema.		

Haber seleccionado un paciente.	
FLUJO NORMAL	
Acciones del Usuario	Acciones del Sistema
1. Accede a la ventana de Fichas Médicas.	2. Muestra las opciones de Ficha Médica
3. Selecciona la opción “Ficha de Salida”	4. Muestra el formulario de ficha de reintegro
5. Ingresa los datos requeridos en cada campo (ver los datos en 11.6. Anexo VI)	
6. Presiona el botón [Guardar]	7. Verifica que los datos ingresados estén correctos
	8. Muestra mensaje “Ficha de Reintegro guardada correctamente”
	9. Redirecciona a la pantalla principal
	10. Fin de UC
FLUJO ALTERNATIVO	
A. Datos incompletos	
A.6. Ingresa datos requeridos incompletos	A.7. Presenta mensaje de error “Complete los campos correctamente”
	A.8. El UC continúa en el paso 5 del flujo normal de eventos
	A.9. Fin de UC
B. Cancela registro	
B.7. Presiona botón [Cancelar]	B.8. Muestra mensaje “¿Seguro que desea cancelar registro?”
B.9. Presiona botón [Aceptar]	B.10. Redirecciona a la pantalla principal
	B.11. Fin de UC
Observación:	

Tabla 22. Descripción del Caso de Uso Obtener Código CIE

Nombre:	Obtener Código CIE
Descripción: El UC permite al Médico buscar el código CIE, de acuerdo a la enfermedad registrada en el diagnóstico de cada ficha médica.	
Actores: Médico	
Precondiciones: Haber iniciado sesión, como Médico, en el sistema. Haber buscado las fichas médicas del paciente. Haber registrado los datos en el formulario de ficha médica	
FLUJO NORMAL	
Acciones del Usuario	Acciones del Sistema
1. Ingresa nombre de enfermedad en el campo (Diagnóstico)	
2. Presiona botón [Buscar CIE]	3. Verifica la información ingresada en el campo

	4. Muestra los códigos relacionados con la búsqueda
5. Selecciona el código correspondiente	
6. Termina registro de datos en ficha médica	
	7. Redirecciona a la pantalla principal
	8. Fin de UC
FLUJO ALTERNATIVO	
A. Campo vacío	
A.3. presiona botón [Buscar] con el campo vacío	A.4. Presenta mensaje de error “Ingrese diagnóstico”
	A.5. El UC continúa en el paso 1 del flujo normal de eventos
	A.6. Fin de UC
Observación:	

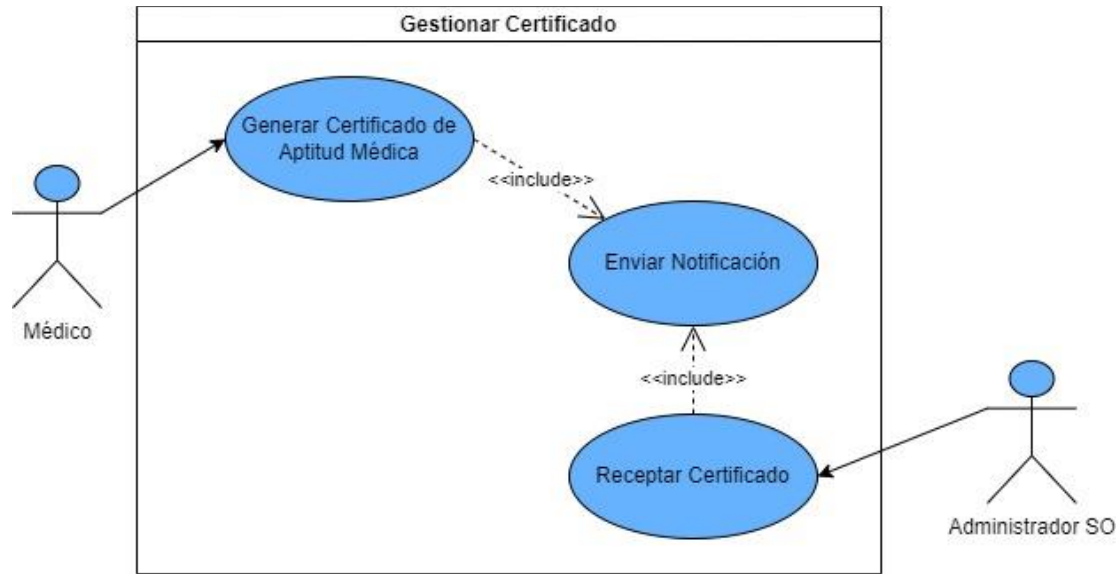


Figura 31. Caso de Uso Gestionar Certificado

Tabla 23. Descripción del Caso de Uso Generar Certificado de Aptitud Médica

Nombre:	Generar Certificado de Aptitud Médica
Descripción: El UC permite al Médico generar el certificado de aptitud médica, de acuerdo al diagnóstico registrado en la ficha inicial del paciente.	
Actores: Médico	
Precondiciones: Haber iniciado sesión, como Médico, en el sistema.	
FLUJO NORMAL	
Acciones del Usuario	Acciones del Sistema
1. Selecciona un paciente.	

2. Selecciona botón para generar certificado	4. Muestra una ventana con el formulario de Certificado de Aptitud Médica
5. Ingresa la información necesaria en cada campo del formulario (Observación: Apto, Apto en Observación, Apto con Limitación, No apto, Código Médico, Nombres y Apellidos del Médico, Firma y Sello)	
6. Presiona el botón [Generar]	7. Verifica que los datos ingresados estén correctos
	8. Muestra mensaje "Certificado generado correctamente"
	9. Muestra ventana emergente con el certificado generado
10. Presiona el botón Descargar certificado	
	11. Muestra mensaje "Certificado guardado"
	12. Redirecciona a la pantalla principal
	13. Fin de UC
FLUJO ALTERNATIVO	
A. Datos incompletos	
A.6. Ingresa datos requeridos incompletos	A.7. Presenta mensaje de error "Complete los campos correctamente"
	A.8. El UC continúa en el paso 5 del flujo normal de eventos
	A.9. Fin de UC
B. Cancela acción	
B.7. Presiona botón [Cancelar]	B.8. Muestra mensaje "¿Seguro que desea cancelar?"
B.9. Presiona botón [Aceptar]	B.10. Redirecciona a la pantalla principal
	B.11. Fin de UC
Observación:	

Tabla 24. Descripción del Caso de Uso Receptar Certificado

Nombre:	Receptar Certificado
Descripción: El UC permite al Administrador visualizar el certificado de una paciente, que ha sido generado por el médico.	
Actores: Administrador de Salud Ocupacional	
Precondiciones: Haber iniciado sesión, como Administrador, en el sistema.	
FLUJO NORMAL	
Acciones del Usuario	Acciones del Sistema
1. Ingresa a la ventana de Notificaciones	2. Muestra las notificaciones recibidas
3. Selecciona la notificación correspondiente a Certificado	4. Carga una ventana con la información contenida en la notificación.

5. Presiona el botón Ver Certificado	6. Muestra el certificado generado.
7. Revisa y verifica la información.	
8. Presiona botón Aceptar	9. Redirecciona a la pantalla principal
	10. Fin de UC
Observación:	

9. Vista Lógica

Dentro de esta vista se muestran los requisitos funcionales del sistema, a través del modelo conceptual y diagrama de clases.

Se debe tener en cuenta el siguiente uso de colores para las clases definidas:

- Clases existentes en el SIAAF
- Clases Nuevas, creadas para el sistema

9.1. Modelo Conceptual

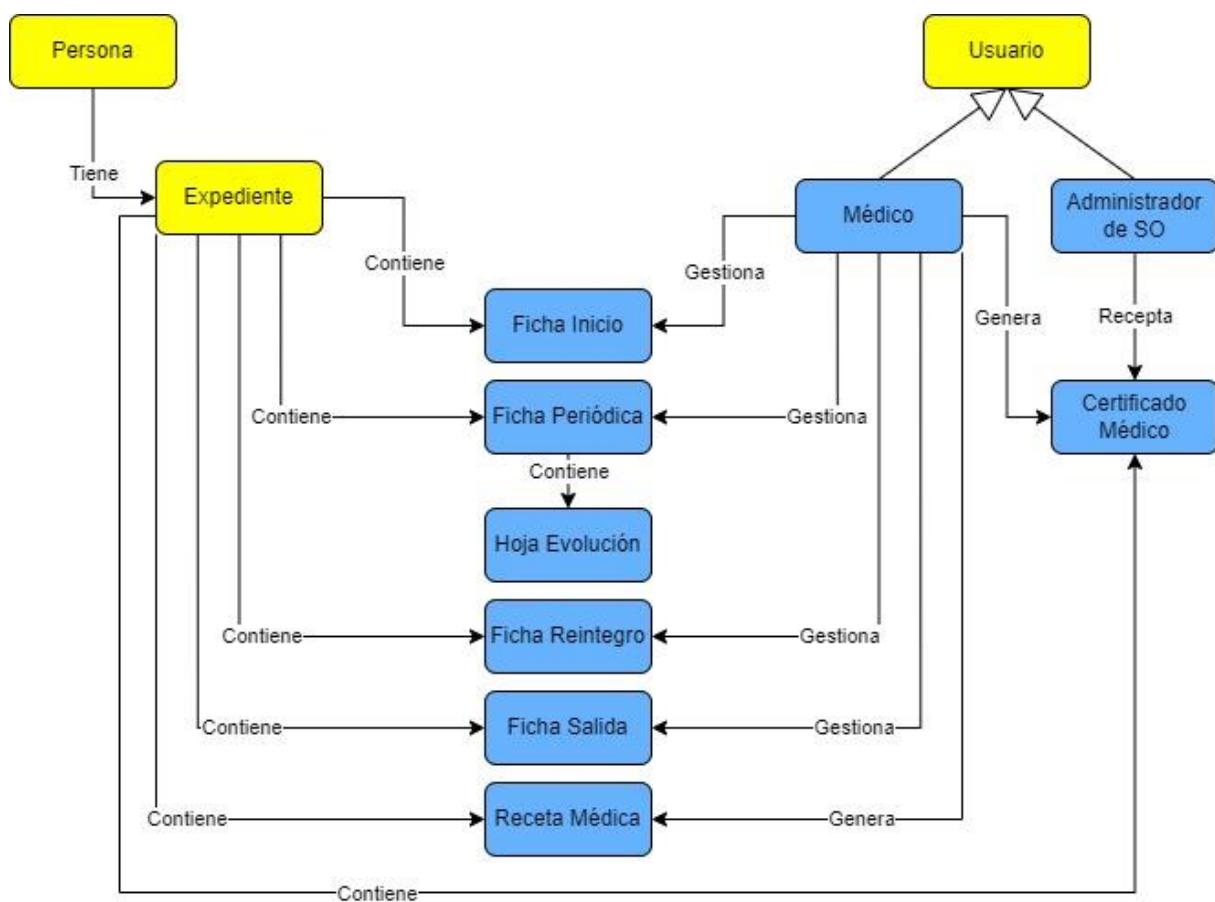


Figura 32. Modelo Conceptual

9.2. Diagrama de Clases

Para observar el diagrama de mejor manera acceder a:
<https://drive.google.com/drive/folders/1QqtrVI2XIqtsmXpwNOWQUY4wgvKp94eC?usp=sharing>

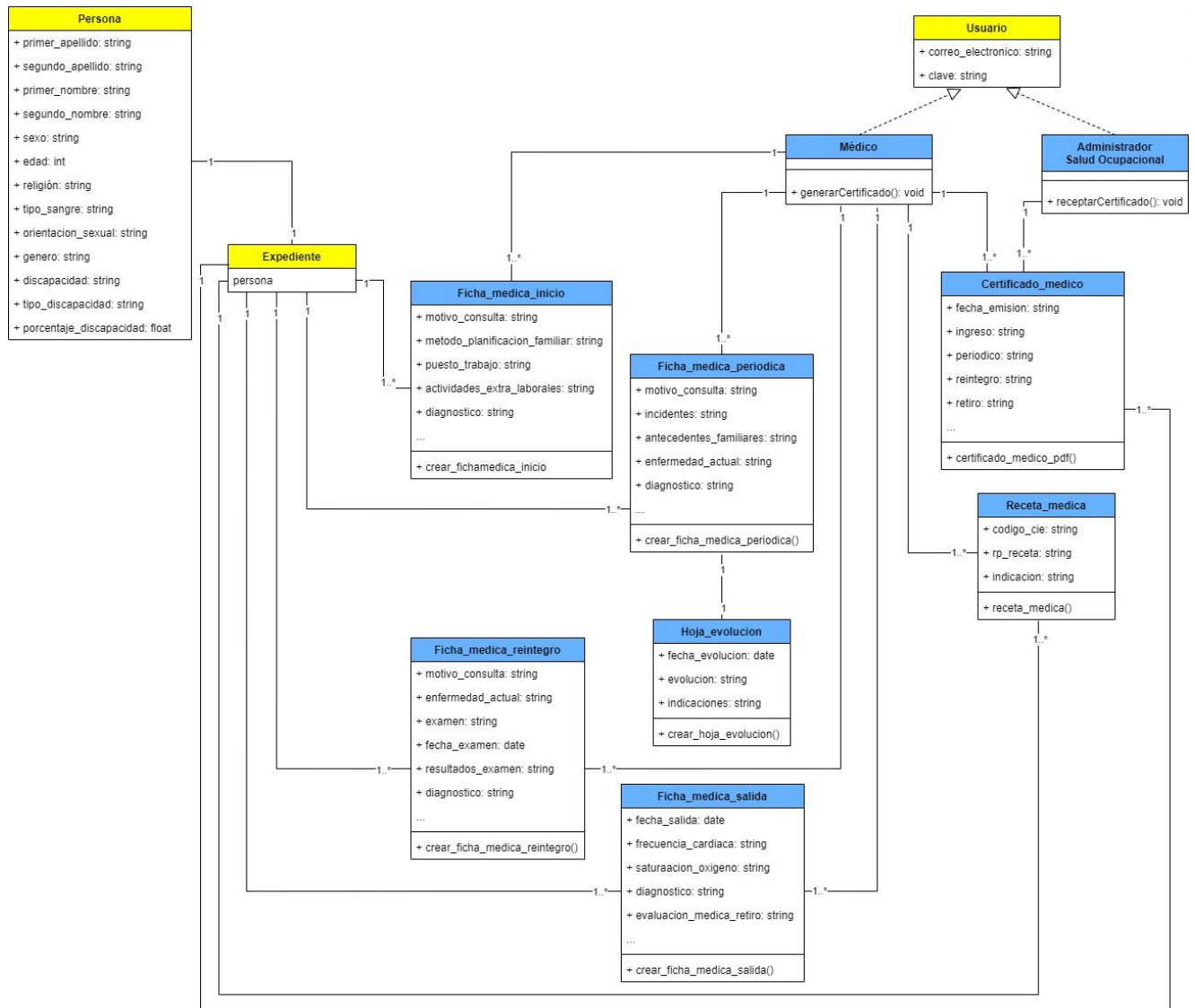


Figura 33. Diagrama de Clases

10. Vista de Procesos

Los diagramas de actividades representan esta vista de procesos del sistema.

10.1. Diagrama de Actividades

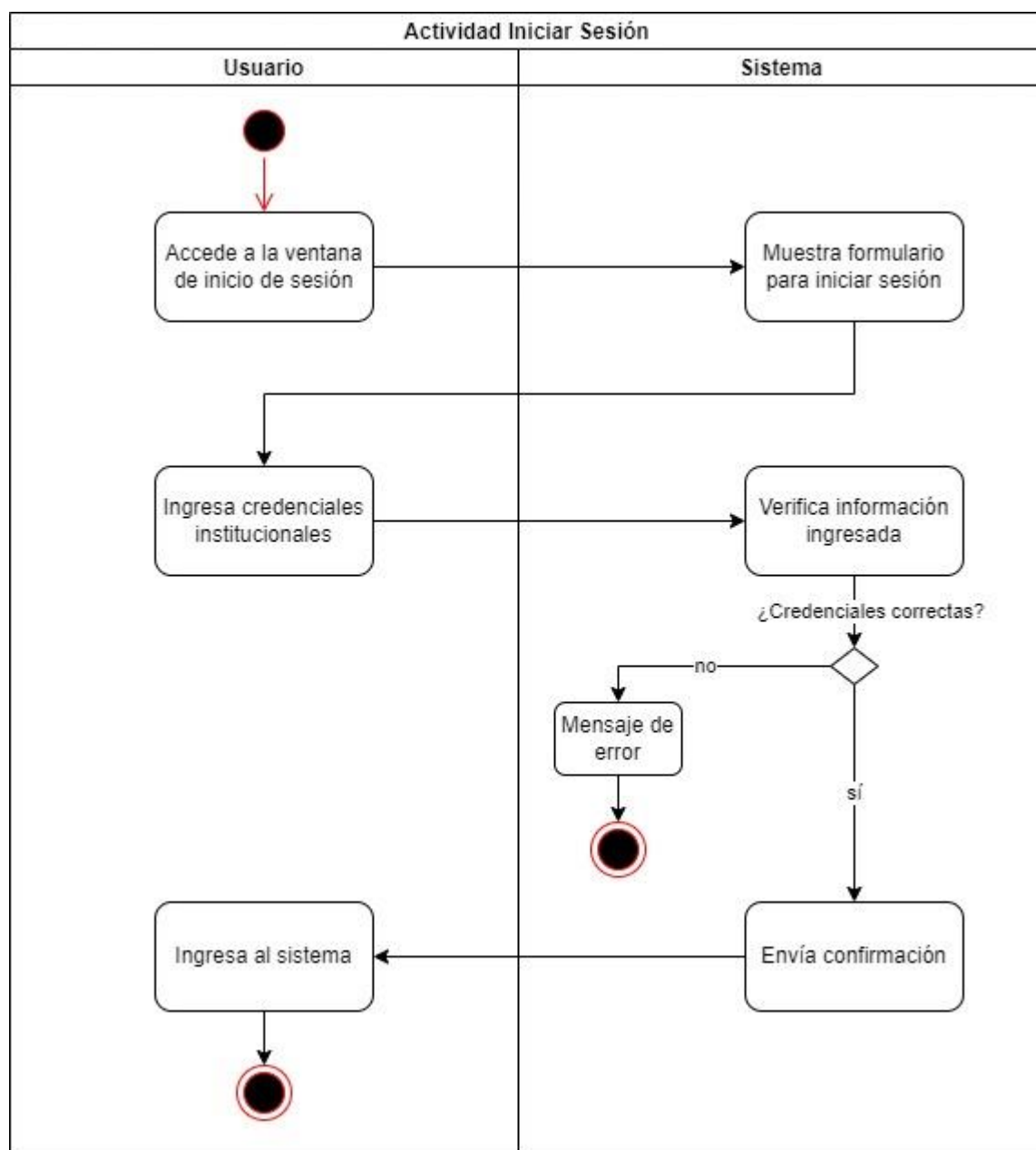


Figura 34. Diagrama de Actividad Iniciar Sesión

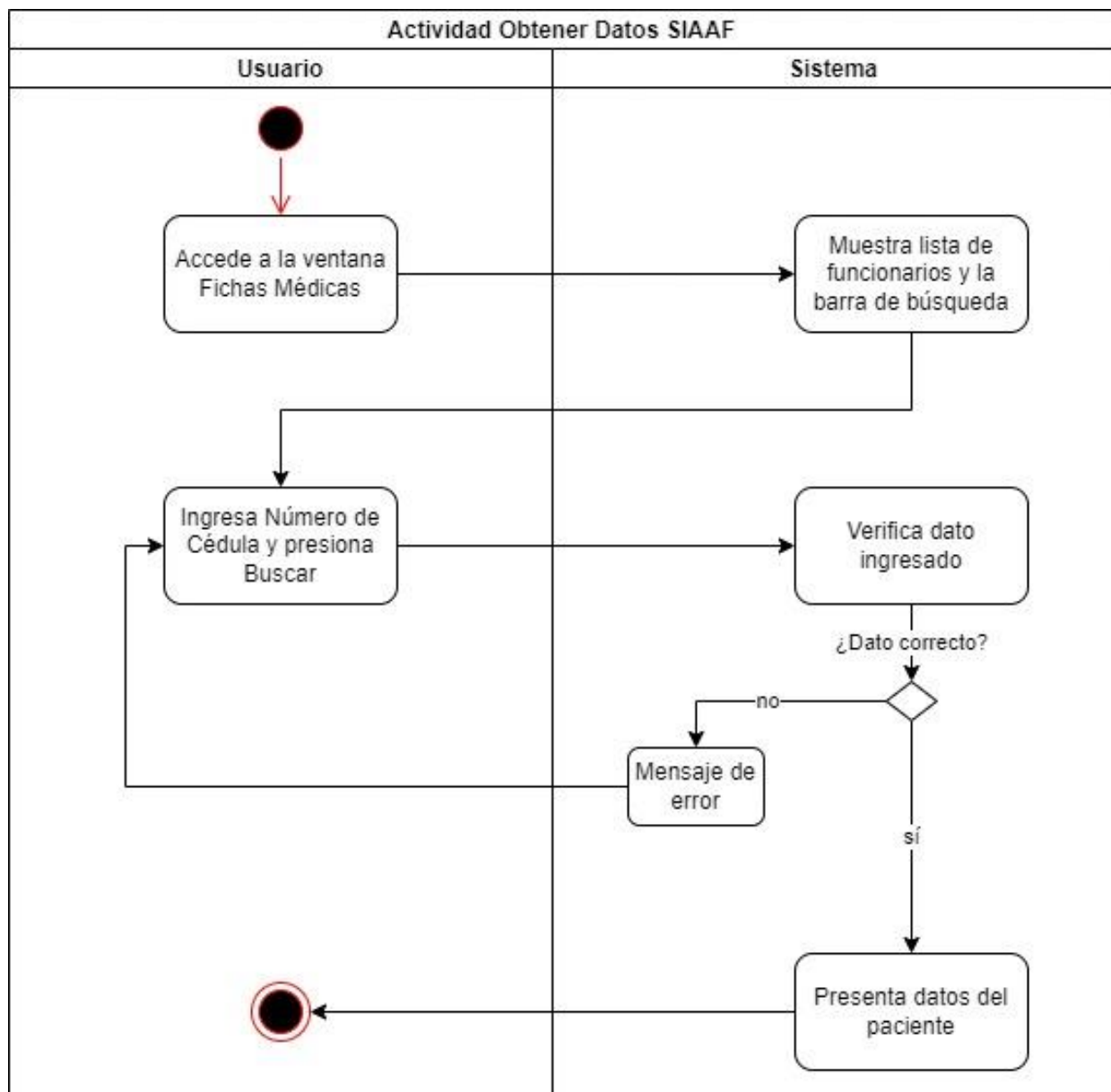


Figura 35. Diagrama de Actividad Obtener Datos SIAAF

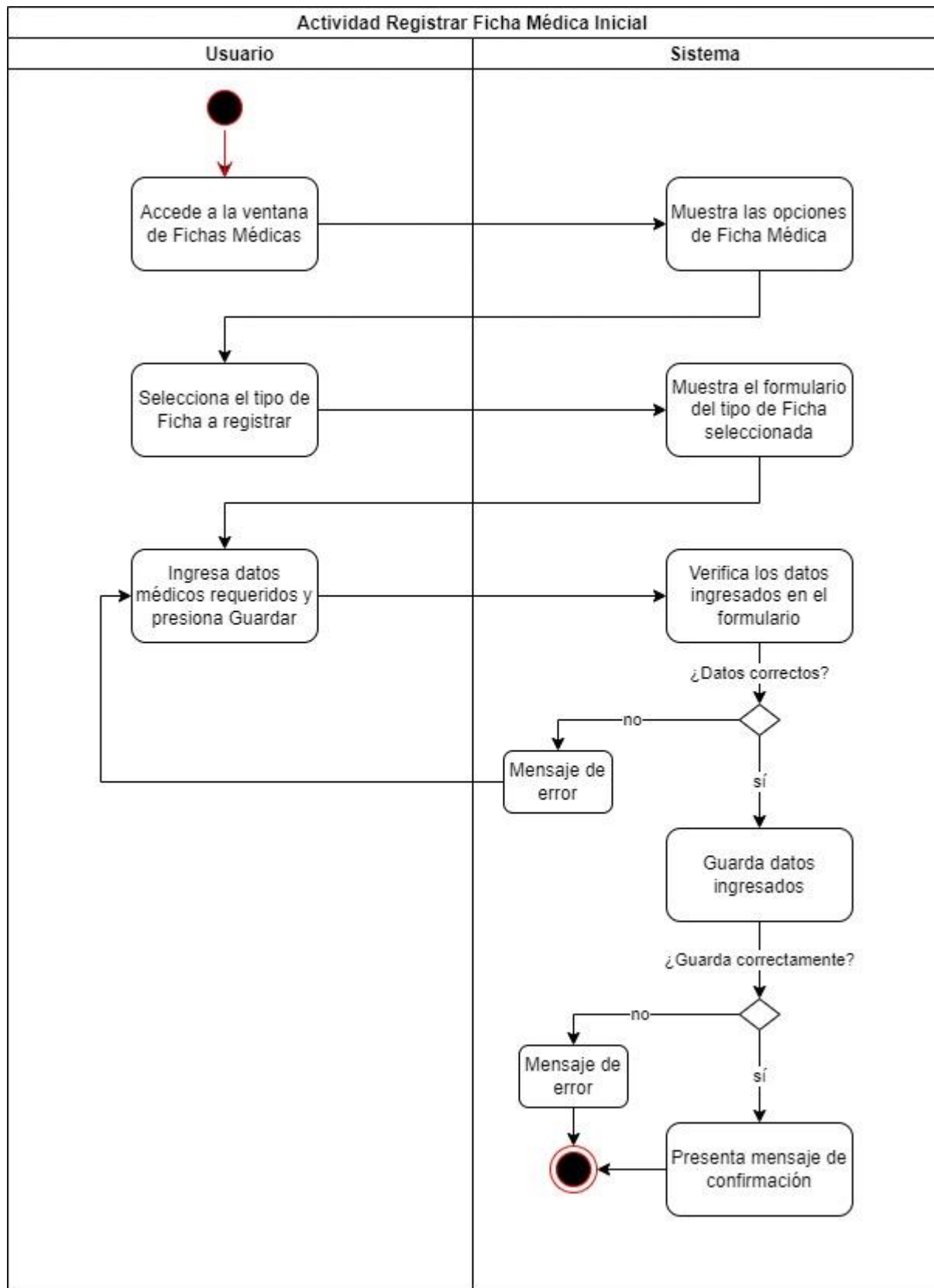


Figura 36. Diagrama de Actividad Registrar Fichas Médicas

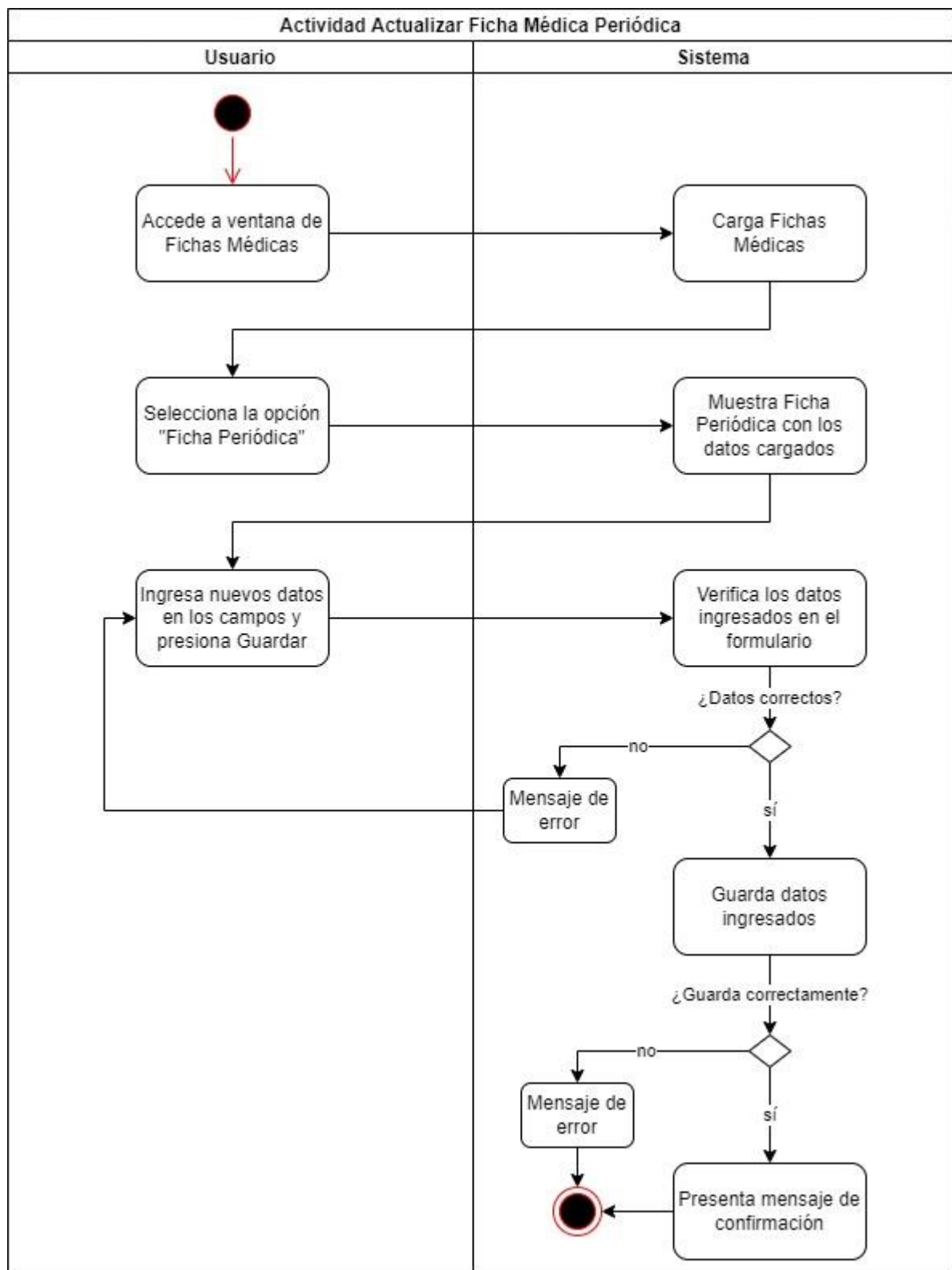


Figura 37. Diagrama de Actividad Actualizar Ficha Médica Periódica

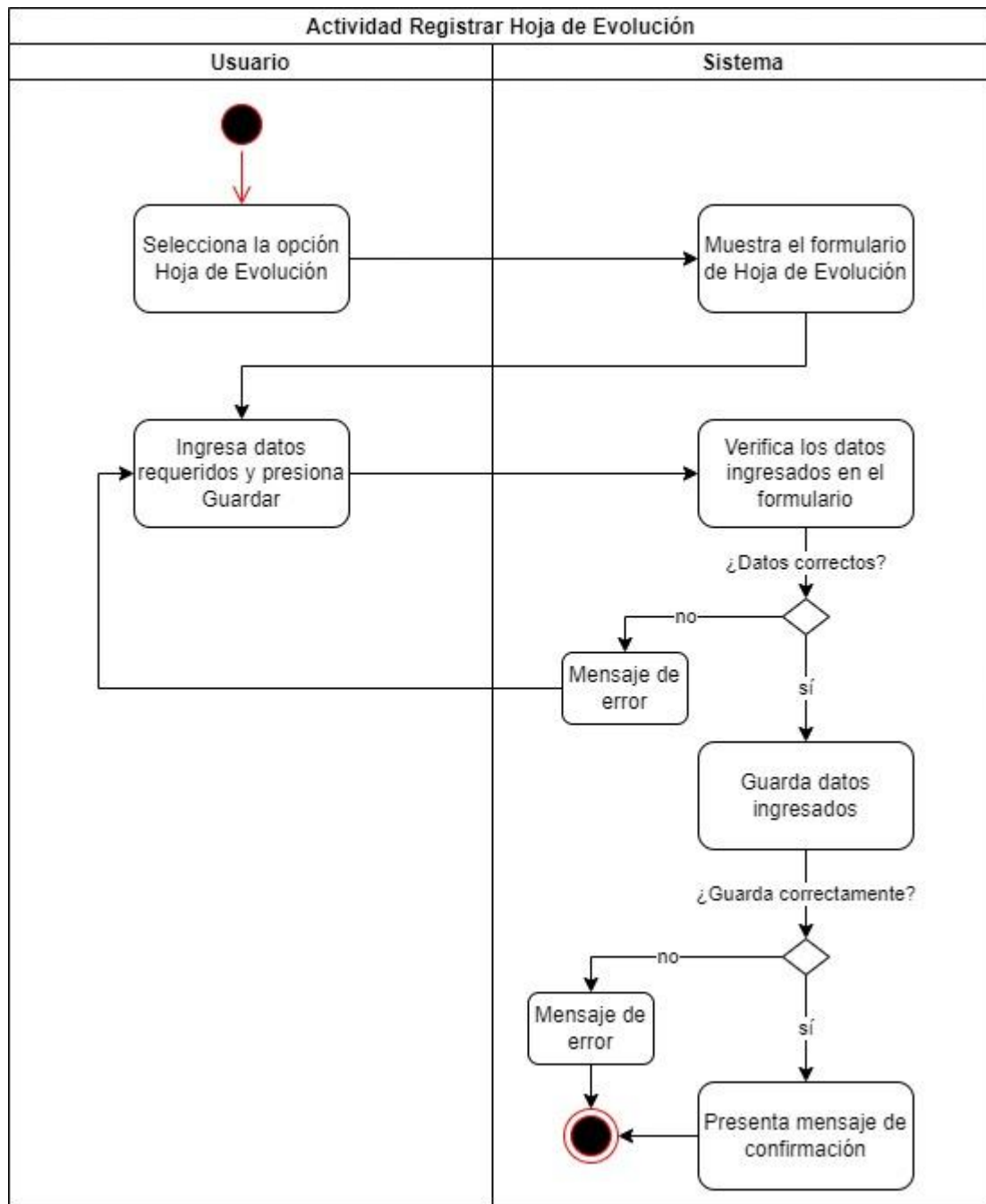


Figura 38. Diagrama de Actividad Registrar Hoja de Evolución

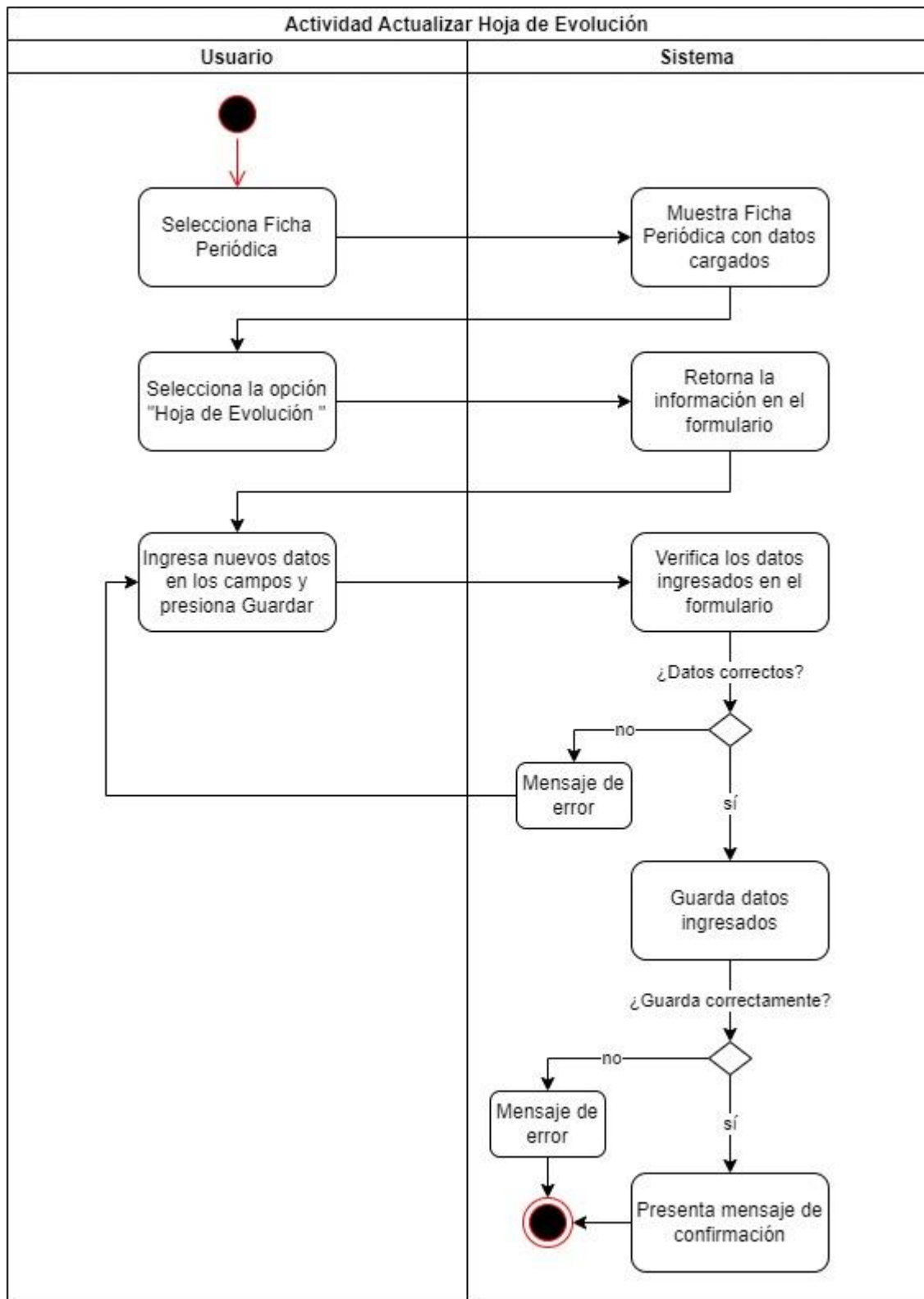


Figura 39. Diagrama de Actividad Actualizar Hoja de Evolución

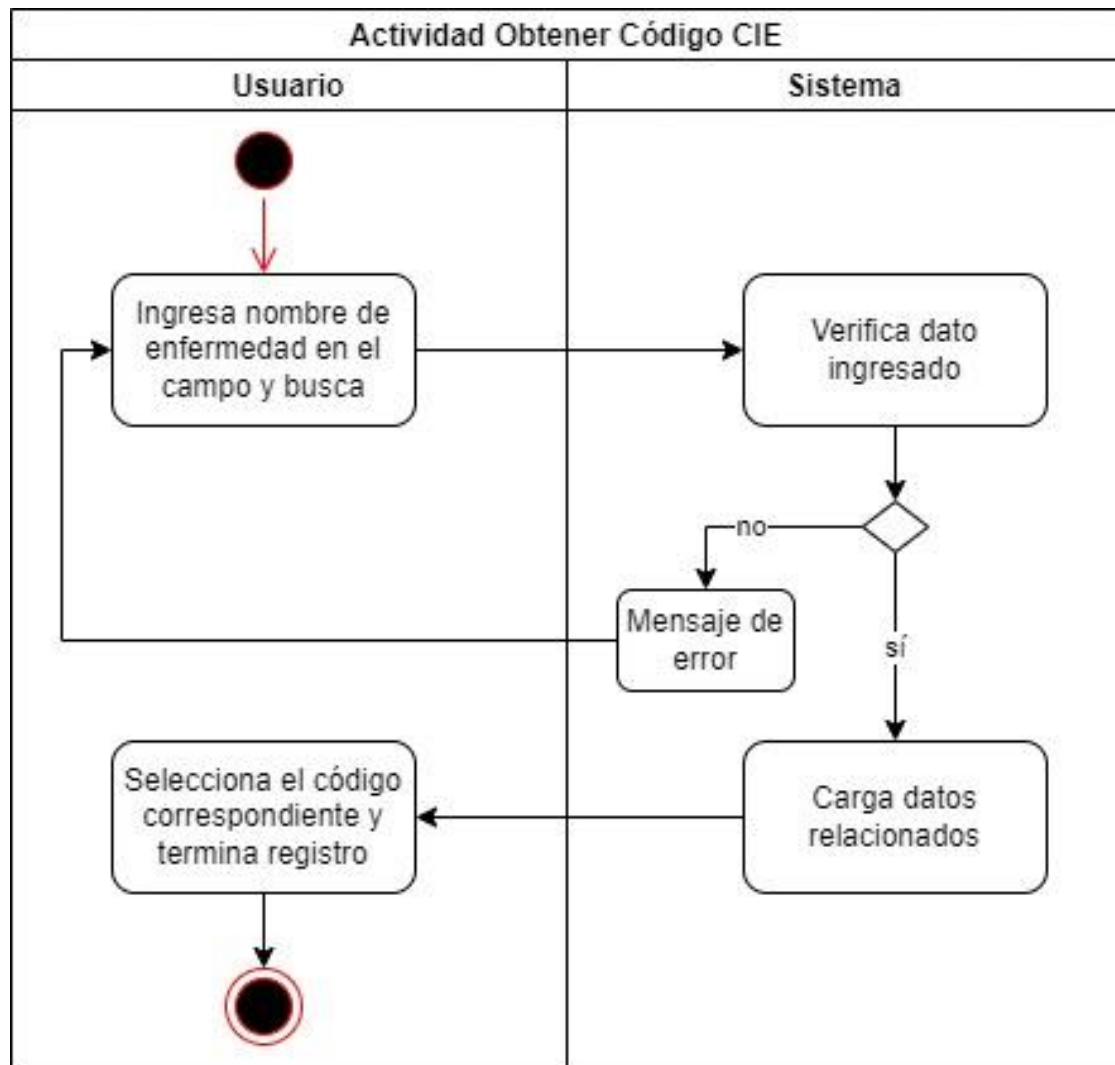


Figura 40. Diagrama de Actividad Obtener Código CIE

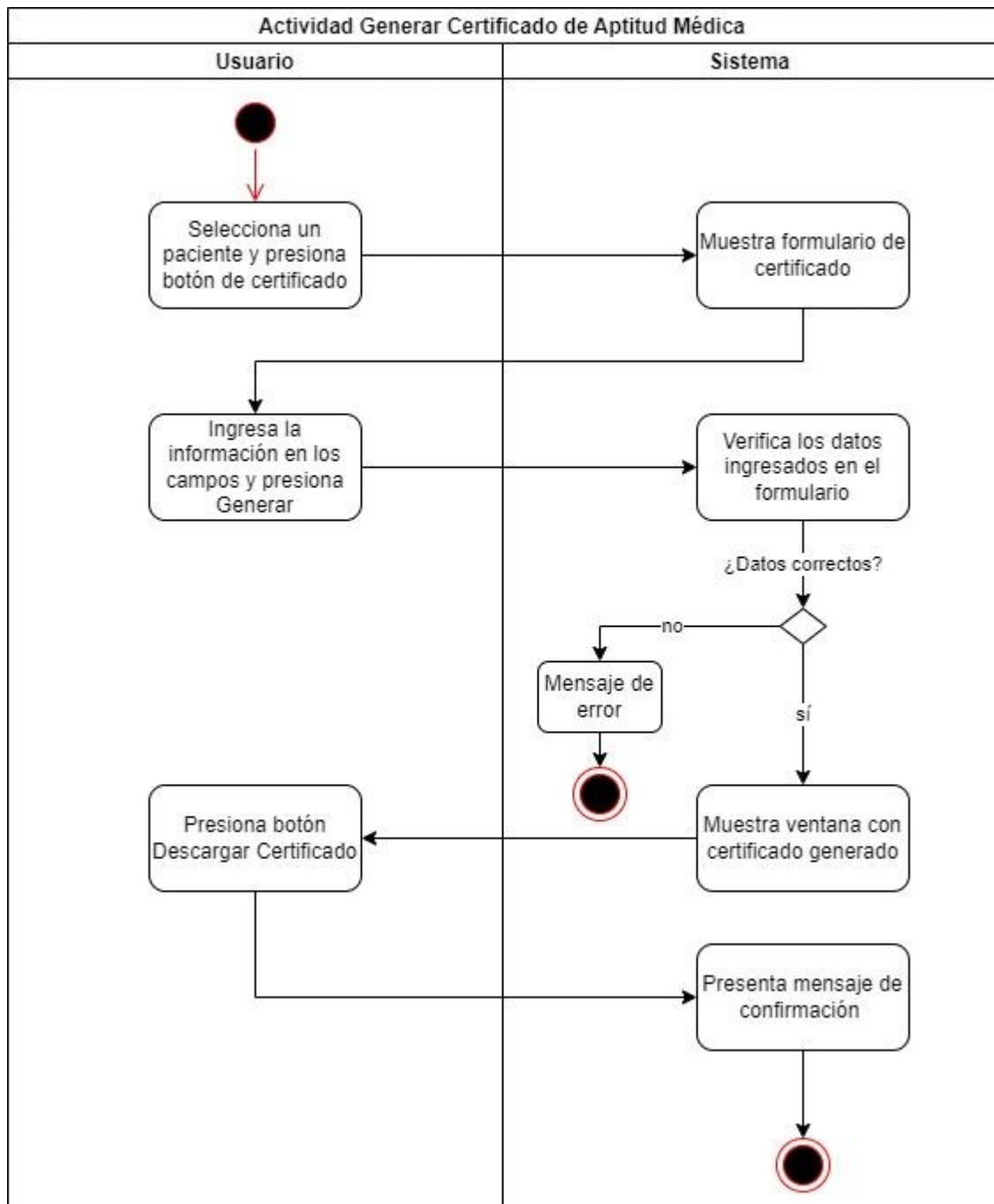


Figura 41. Diagrama de Actividad Generar Certificado de Aptitud Médica

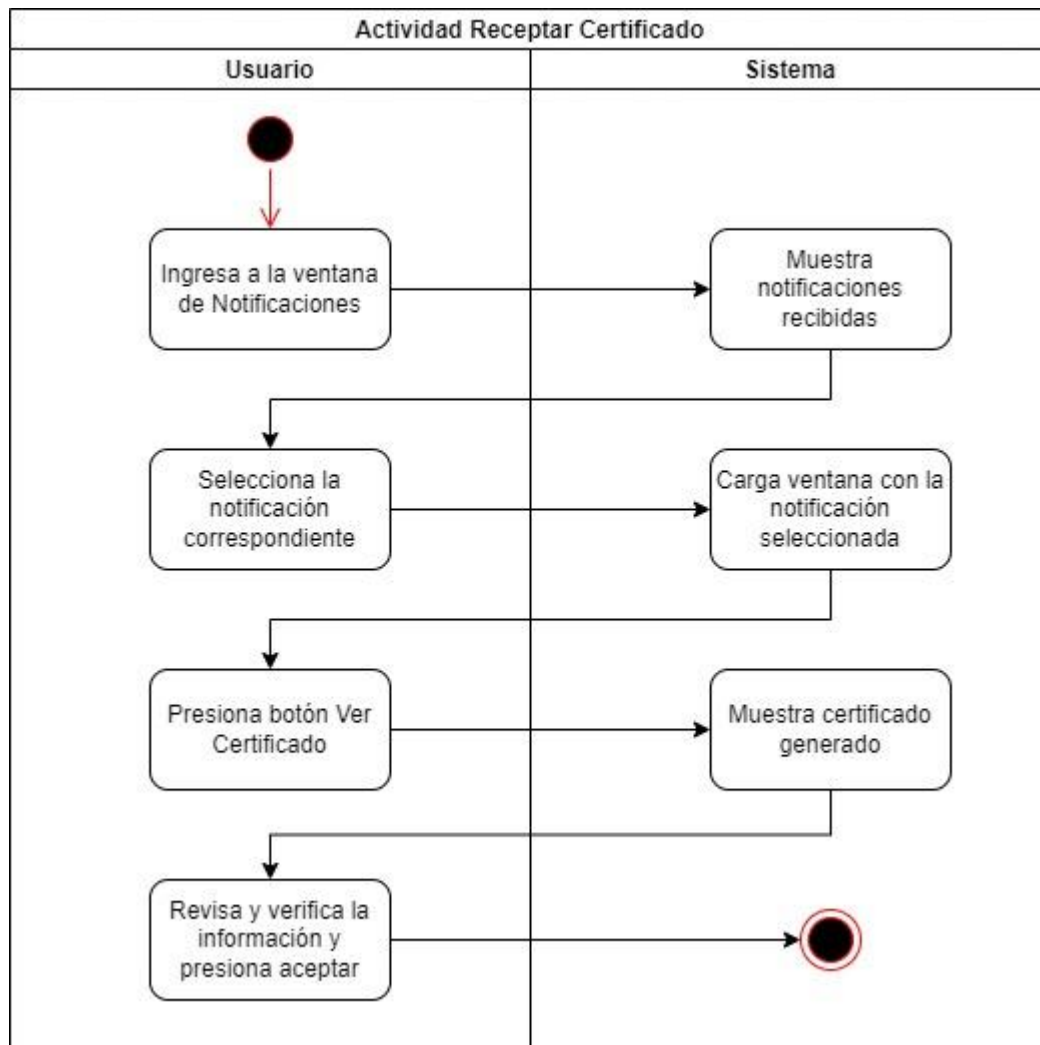


Figura 42. Diagrama de Actividad Receptar Certificado

11. Vista de Despliegue

En esta vista se organiza los módulos o componentes del sistema con la siguiente estructura:

- **Servidor de Aplicaciones:** Dentro de esta primera parte se define el desarrollo del Back-End, lógica del negocio, a través del framework Django, usando el lenguaje de programación Python, en este se establecerán las rutas y métodos que interactuarán con la base de datos PostgreSQL.
- **Cliente**
 - **Aplicación Web:** El sistema desarrollado en Django se visualizará a través de las peticiones con el protocolo HTTP al servicio web (API-REST). La aplicación está dirigida únicamente al médico ocupacional y al subdirector de Seguridad y Salud Ocupacional, quienes son encargados de la gestión de las fichas médicas y certificados de pacientes, quienes trabajan dentro de la institución.

- **Base de Datos:** La base de datos, en PostgreSQL, se comunicará directamente con el servicio web (API-REST).

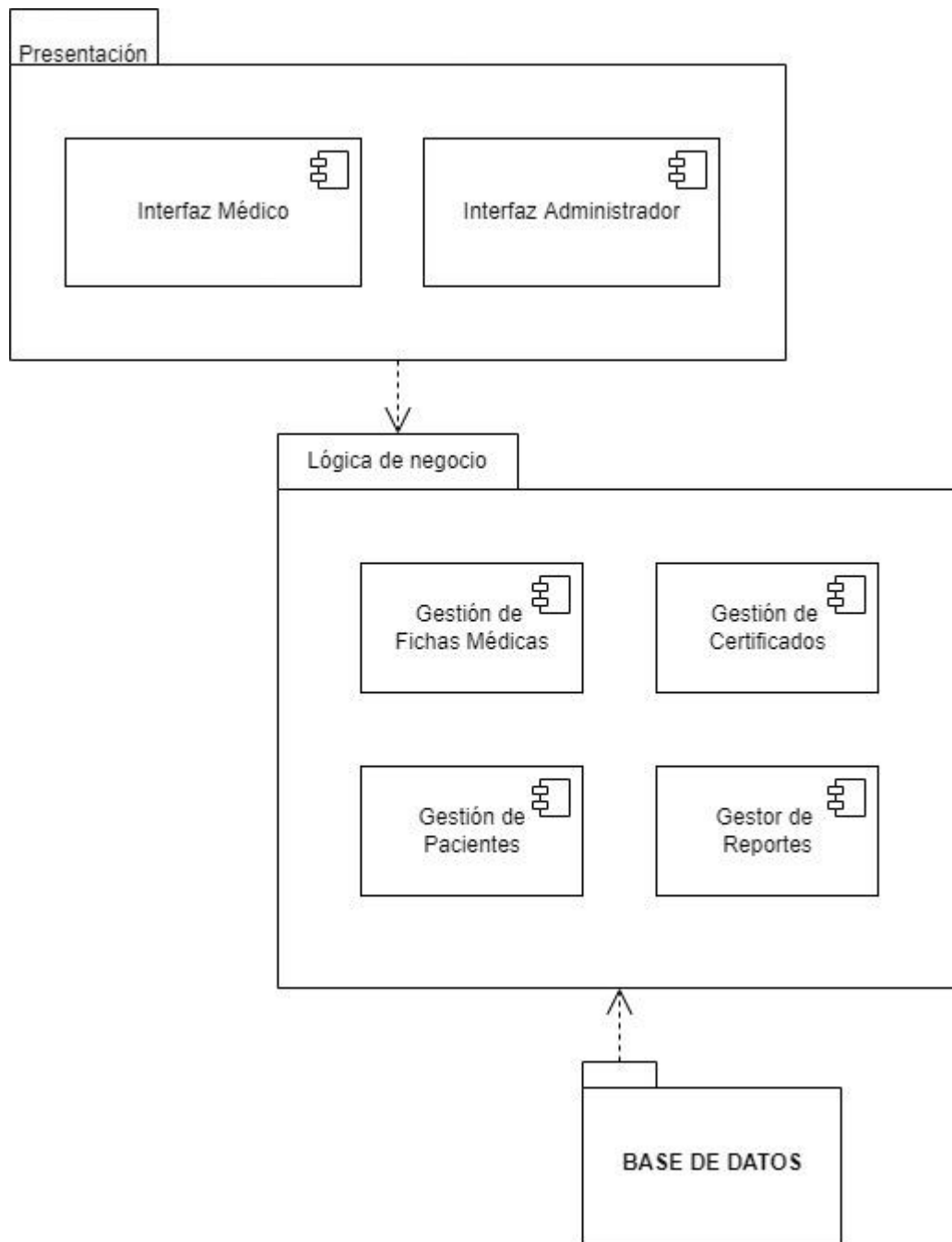


Figura 43. Diagrama de Componentes

12. Vista Física

Esta vista se representa por medio del diagrama de despliegue, que muestra los nodos finales del sistema, usando la arquitectura cliente-servidor.

La aplicación web es el cliente que se comunica con el servidor, que almacena el servicio web (API-REST) el cual interactúa con la base de datos para poder procesar la información.

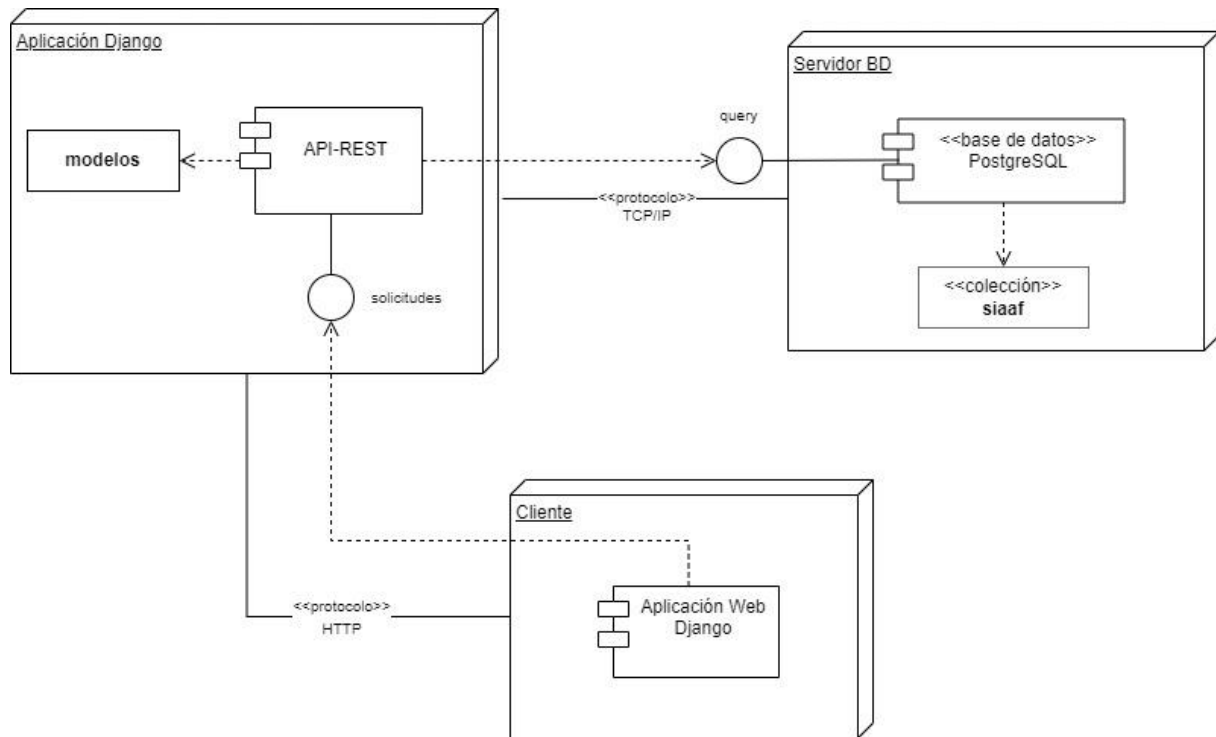


Figura 44. Diagrama de Despliegue

Anexo VI. Formatos de Fichas Médicas Ocupacionales

Proyecto:

Sistematización de Fichas Médicas Ocupacionales y Test Psicológicos

Fichas Médicas Ocupacionales

<https://drive.google.com/file/d/1qozoN9j-Dq5M3luaOTU7UhNJTZW20Mj2/view?usp=sharing>

A. DATOS DEL ESTABLECIMIENTO - EMPRESA Y USUARIO																																																																
INSTITUCIÓN DEL SISTEMA O NOMBRE DE LA EMPRESA										RUC					OSU					ESTABLECIMIENTO DE SALUD					NÚMERO DE HISTORIA CLÍNICA					NÚMERO DE ARCHIVO																																		
PRIMER APELLIDO					SEGUNDO APELLIDO					PRIMER NOMBRE					SEGUNDO NOMBRE					SEXO		EDAD (años)		RELIGIÓN		GRUPO SANGÜÍNEO		LATERALIDAD																																				
ORIENTACIÓN SEXUAL					IDENTIDAD DE GÉNERO					DISCAPACIDAD					FECHA DE INGRESO AL TRABAJO (año/mes/día)					PUESTO DE TRABAJO (CUI)					ÁREA DE TRABAJO					ACTIVIDADES RELEVANTES AL PUESTO DE TRABAJO A OCUPAR																																		
B. MOTIVO DE CONSULTA																																																																
DESCRIPCIÓN																																																																
ANOTAR LA CAUSA DEL PROBLEMA EN LA VERSIÓN DEL INFORMANTE																																																																
C. ANTECEDENTES PERSONALES																																																																
ANTECEDENTES CLÍNICOS Y QUIRÚRGICOS																																																																
DESCRIPCIÓN																																																																
ANTECEDENTES GINECO OBSTÉTRICOS																																																																
MENARQUÍA					CICLOS					FECHA DE ÚLTIMA MENSTRUACIÓN (año/mes/día)					GESTAS					PARTOS					CESÁREAS					ABORTOS					HIJOS					VIDA SEXUAL ACTIVA					MÉTODO DE PLANIFICACIÓN FAMILIAR																			
EXÁMENES REALIZADOS					SI					NO					TIEMPO (años)					RESULTADO					EXÁMENES REALIZADOS					SI					NO					TIEMPO (años)					RESULTADO																			
PAPANICOLAOU																				ECOMINARIO																																												
COLPOSCOPIA																				MAMOGRAFÍA																																												
ANTECEDENTES REPRODUCTIVOS MASCULINOS																																																																
EXÁMENES REALIZADOS					SI					NO					TIEMPO (años)					RESULTADO					MÉTODO DE PLANIFICACIÓN FAMILIAR					SI					NO					TIPO					HIJOS					VIVOS					MUERTOS									
ANTÍGENO PROSTÁTICO																																																																
ECO PROSTÁTICO																																																																
HÁBITOS TÓXICOS																																																																
CONSUMOS NOCIVOS					SI					NO					TIEMPO DE CONSUMO (años)					CANTIDAD					EX CONSUMIDOR					TIEMPO DE ABSTINENCIA (años)					ESTILO					SI					NO					¿CUAL?					TIEMPO / CANTIDAD									
TABACO																																																																
ALCOHOL																																																																
OTRAS DROGAS:																																																																
D. ANTECEDENTES DE TRABAJO																																																																
ANTECEDENTES DE EMPLEOS ANTERIORES																																																																
EMPRESA					PUESTO DE TRABAJO					ACTIVIDADES QUE DESEMPEÑABA					TIEMPO DE TRABAJO (años)					RIESGO					OBSERVACIONES																																							
																				FÍSICO																																												
																				MÉCANICO																																												
																				QUÍMICO																																												
																				BIOLÓGICO																																												
																				ELECTRÓNICO																																												
																				PROCESAMIENTO																																												
																				PROFESIONAL																																												
ACCIDENTES DE TRABAJO (DESCRIPCIÓN)																																																																
FUE CALIFICADO POR EL INSTITUTO DE SEGURIDAD SOCIAL CORRESPONDIENTE: SI <input type="checkbox"/> ESPECIFICAR: _____ NO <input type="checkbox"/> FECHA: _____																																																																
DESCRIPCIÓN:																																																																
ENFERMEDADES PROFESIONALES																																																																
FUE CALIFICADO POR EL INSTITUTO DE SEGURIDAD SOCIAL CORRESPONDIENTE: SI <input type="checkbox"/> ESPECIFICAR: _____ NO <input type="checkbox"/> FECHA: _____																																																																
DESCRIPCIÓN:																																																																

K. EXAMEN FÍSICO REGIONAL														
REGIONES														
1. Piel	a. Cicatrices		3. Oído	a. C. auditivo externo		5. Nariz	a. Tabique		6. Tórax	a. Pulmones		11. Pelve	a. Pélvis	
	b. Tetuajes			b. Pabellón			b. Cornetes			b. Paredes Costal			b. Genitales	
	c. Piel y Faneras			c. Timpanos			c. Mucosas			a. Visceras			a. Vascular	
	a. Párpados			a. Labios			d. Senos paranasales			b. Pared abdominal			b. Miembros superiores	
	b. Conjuntivas			b. Lengua			a. Tiroides / masas			a. Flexibilidad			c. Miembros inferiores	
2. Ojos	c. Pupilas		4. Oído interno	c. Faringe		6. Codo	b. Movilidad		10. Codo	b. Desviación		12. Miembros inferiores	a. Fuerza	
	d. Córneas			d. Amígdalas			a. Mamas			c. Dolor			b. Sensibilidad	
	a. Motilidad			a. Dentadura			b. Conozón						c. Marcha	
													d. Reflejos	
SI EXISTE EVIDENCIA DE PATOLOGÍA MARCAR CON "X" Y DESCRIBIR EN LA SIGUIENTE SECCIÓN COLOCANDO EL NUMERAL														
Observaciones:														
L. RESULTADOS DE EXÁMENES GENERALES Y ESPECÍFICOS DE ACUERDO AL RIESGO Y PUESTO DE TRABAJO (IMAGEN, LABORATORIO Y OTROS)														
EXAMEN		FECHA	RESULTADOS											
OBSERVACIONES:														
M. DIAGNÓSTICO														
PRE= PRESUNTIVO		DEF= DEFINITIVO		CIE		PRE		DEF						
1	Descripción													
2														
3														
N. APTITUD MÉDICA PARA EL TRABAJO														
APTO		APTO EN OBSERVACIÓN		APTO CON LIMITACIONES		NO APTO								
Observación														
Limitación														
O. RECOMENDACIONES Y/O TRATAMIENTO														
Descripción														
CERTIFICO QUE LO ANTERIORMENTE EXPRESADO EN RELACIÓN A MI ESTADO DE SALUD ES VERDAD. SE ME HA INFORMADO LAS MEDIDAS PREVENTIVAS A TOMAR PARA DISMINUIR O MITIGAR LOS RIESGOS RELACIONADOS CON MI ACTIVIDAD LABORAL.														
P. DATOS DEL PROFESIONAL								Q. FIRMA DEL USUARIO						
FECHA		HORA		NOMBRES Y APELLIDOS		CÓDIGO		FIRMA Y SELLO						

[illegible]

F. ENFERMEDAD ACTUAL									
Descripción									
G. REVISIÓN DE ÓRGANOS Y SISTEMAS									
EN CASO DE EXISTIR PATOLOGÍA, MARCAR CON "X" Y DESCRIBIR ABAJO ANOTANDO EL NUMERAL									
1. PIEL - ANEXOS	3. RESPIRATORIO	5. DIGESTIVO	7. MÚSCULO ESQUELÉTICO	9. HEMO LINFÁTICO					
2. ÓRGANOS DE LOS SENTIDOS	4. CARDIO-VASCULAR	6. GENITO - URINARIO	8. ENDOCRINO	10. NERVIOSO					
Descripción									
H. CONSTANTES VITALES Y ANTROPOMETRÍA									
PRESIÓN ARTERIAL (mmHg)	TEMPERATURA (°C)	FRECUENCIA CARDÍACA (b/min)	SATURACIÓN DE OXÍGENO (2%)	FRECUENCIA RESPIRATORIA (b/min)	PESO (kg)	TALLA (cm)	ÍNDICE DE MASA CORPORAL (kg/m²)	PERÍMETRO ABDOMINAL (cm)	
I. EXAMEN FÍSICO REGIONAL									
REGIONES									
1. Piel	a. Cicatrices	2. Oído	a. C. auditivo externo	3. Nariz	a. Tabique	4. Tórax	a. Pulmones	5. Pélvis	a. Pélvis
	b. Tatuajes		b. Pabellón		b. Concha		b. Pared costal		b. Genitales
	c. Piel y anexos		c. Timpanos		c. Mucosas		c. Várices		c. Vascular
	d. Párpados		d. Labios		d. Senos paranasales		d. Pared abdominal		d. Miembros superiores
	e. Conjuntivas		e. Lengua		e. Tiroidea / masas		e. Flexibilidad		e. Miembros inferiores
2. Ojos	f. Pupilas	3. Oído interno	f. Faringe	4. Cuello	f. Movilidad	5. Abdomen	f. Desviación	6. Extremidades	f. Fuerza
	g. Córneas		g. Amígdalas		g. Mamas		g. Desviación		g. Sensibilidad
	h. Motilidad		h. Dentadura		h. Conciencia		h. Dolor		h. Marcha
SI EXISTE EVIDENCIA DE PATOLOGÍA MARCAR CON "X" Y DESCRIBIR EN LA SIGUIENTE SECCIÓN COLOCANDO EL NUMERAL									
Observaciones:									
J. RESULTADOS DE EXÁMENES GENERALES Y ESPECÍFICOS DE ACUERDO AL RIESGO Y PUESTO DE TRABAJO (IMAGEN, LABORATORIO Y OTROS)									
EXÁMEN	FECHA	RESULTADO							
Observaciones:									
K. DIAGNÓSTICO									
		PRE= PRESUNTIVO	DEF= DEFINITIVO	CE	PRE	DEF			
1.	Descripción								
2.									
3.									
L. APTITUD MÉDICA PARA EL TRABAJO									
APTO		APTO EN OBSERVACIÓN		APTO CON LIMITACIONES		NO APTO			
Observación									
Limitación									
M. RECOMENDACIONES Y/O TRATAMIENTO									
Descripción									
CERTIFICO QUE LO ANTERIORMENTE EXPRESADO EN RELACIÓN A MI ESTADO DE SALUD ES VERDAD. SE ME HA INFORMADO LAS MEDIDAS PREVENTIVAS A TOMAR PARA DISMINUIR O MITIGAR LOS RIESGOS RELACIONADOS CON MI ACTIVIDAD LABORAL.									
N. DATOS DEL PROFESIONAL						O. FIRMA DEL USUARIO			
FECHA	HORA	NOMBRES Y APELLIDOS	CÓDIGO	FIRMA Y SELLO					

A. DATOS DEL ESTABLECIMIENTO - EMPRESA Y USUARIO										
INSTITUCIÓN DEL SISTEMA O NOMBRE DE LA EMPRESA			RUC	OSU	ESTABLECIMIENTO DE SALUD		NÚMERO DE HISTORIA CLÍNICA	NÚMERO DE ARCHIVO		
PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	SEXO	EDAD años	PUESTO DE TRABAJO (CUIUC)	FECHA DEL ÚLTIMO DÍA LABORAL	FECHA DE REINGRESO	TOTAL años	CAUSA DE SALIDA
B. MOTIVO DE CONSULTA / CONDICIÓN DE REINGRESO										
Descripción										
C. ENFERMEDAD ACTUAL										
Descripción										
D. CONSTANTES VITALES Y ANTROPOMETRÍA										
FRASEO ANTROPOMÉTRICO	TEMPERATURA °C	FRECUENCIA CARDÍACA /min	SATURACIÓN DE OXÍGENO %	FRECUENCIA RESPIRATORIA /min	PSICO /kg	TALLA cm	ÍNDICE DE MASA CORPORAL kg/m²	PERÍMETRO ANTRÓPICO cm		
/										
E. EXAMEN FÍSICO REGIONAL										
PSICÓLOGO										
1. Piel	a. Coloración		3. Oído	a. C. auditivo externo		5. Nariz	a. Tórax	a. Pulmones		11. Piel
	b. Tatuajes			b. Posición			b. Corriente	b. Piel de Costal		b. Genitales
	c. Pel y Femenos			c. Timbre			c. Mucosa	c. Visión		c. Vascular
2. Ojos	a. Párpados		4. Oído	a. Latido		6. Tórax	a. Alveolar	a. Piel abdominal		d. Membre superiores
	b. Conjuntivas			b. Lengua			b. Tórax / mama	b. Flexibilidad		e. Membre inferiores
	c. Pupila			c. Rango		8. Oído	b. Movilidad	b. Desecación		f. Piel
	d. Córnea			d. Amigdalas			a. Mamas	c. Dolor		g. Sensibilidad
	e. Motilidad			e. Dentadura		7. Tórax	b. Corazón			h. Marcha
OP = CON INDICACIÓN DE PROTOCOLO, MARCAR "X" Y DESCRIBIR EN LA SIGUIENTE SECCIÓN RP = SIN INDICACIÓN DE PROTOCOLO, MARCAR "X" Y NO DESCRIBIR										
Observaciones:										
F. RESULTADOS DE EXÁMENES (IMAGEN, LABORATORIO Y OTROS)										
EXAMEN	FECHA	RESULTADO								
Observaciones:										
G. DIAGNÓSTICO										
PRE-PRONÓSTICO	OP-PRONÓSTICO	CE	PRE	DISP						
1. Descripción										
2.										
3.										
H. APTITUD MÉDICA PARA EL TRABAJO										
APTO	APTO EN OBSERVACIÓN	APTO CON LIMITACIONES	NO APTO							
Observación										
Limitación										
Reubicación										
I. RECOMENDACIONES Y/O TRATAMIENTO										
Descripción										
CERTIFICO QUE LO ANTERIORMENTE EXPRESADO EN RELACIÓN A MI ESTADO DE SALUD ES VERDAD. SE ME HA INFORMADO LAS MEDIDAS PREVENTIVAS A TOMAR PARA DISMINUIR O MITIGAR LOS RIESGOS RELACIONADOS CON MI ACTIVIDAD LABORAL.										
J. DATOS DEL PROFESIONAL				K. FIRMA DEL USUARIO						
FECHA	HORA	NOMBRE Y APELLIDO	CÓDIGO	FIRMA Y SELLO						

A. DATOS DEL ESTABLECIMIENTO - EMPRESA Y USUARIO									
INSTITUCIÓN DEL SISTEMA O NOMBRE DE LA EMPRESA		RUC	CIU	ESTABLECIMIENTO DE SALUD		NÚMERO DE HISTORIA CLÍNICA		NÚMERO DE ARCHIVO	
PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	SEXO	FECHA DE INICIO DE LABORES	FECHA DE SALIDA	TIEMPO (años)	PUESTO DE TRABAJO (CIUC)	
					aaaa / mm / dd	aaaa / mm / dd			
ACTIVIDADES			FACTORES DE RIESGO						
B. ANTECEDENTES PERSONALES									
ANTECEDENTES CLÍNICOS Y QUIRÚRGICOS									
ACCIDENTES DE TRABAJO (DESCRIPCIÓN)									
FUE CALIFICADO POR EL INSTITUTO DE SEGURIDAD SOCIAL CORRESPONDIENTE: SI <input type="checkbox"/> ESPECIFICAR: _____ NO <input type="checkbox"/> FECHA: <input type="text"/> / <input type="text"/> / <input type="text"/>									
Observaciones:									
Detallar aquí en caso se presuma de algún accidente de trabajo que no haya sido reportado o calificado:									
ENFERMEDADES PROFESIONALES									
FUE CALIFICADO POR EL INSTITUTO DE SEGURIDAD SOCIAL CORRESPONDIENTE: SI <input type="checkbox"/> ESPECIFICAR: _____ NO <input type="checkbox"/> FECHA: <input type="text"/> / <input type="text"/> / <input type="text"/>									
Observaciones:									
Detallar aquí en caso de que se presuma de alguna enfermedad relacionada con el trabajo que no haya sido reportada o calificada:									
C. CONSTANTES VITALES Y ANTROPOMETRÍA									
PRESIÓN ARTERIAL (mmHg)	TEMPERATURA (°C)	FRECUENCIA CARDÍACA (bpm)	SATURACIÓN DE OXÍGENO (%)	FRECUENCIA RESPIRATORIA (bpm)	PESO (kg)	TALLA (m)	ÍNDICE DE MASA CORPORAL (kg/m²)	PERÍMETRO ABDOMINAL (cm)	
/									
D. EXAMEN FÍSICO REGIONAL									
REGIONES									
1. Piel		3. Oído		5. Nariz		7. Tórax		9. Abdomen	
a. Clorosis		a. C. auditivo externo		a. Tabique		a. Pulmones		a. Pélvis	
b. Tatuajes		b. Pabellón		b. Cornetes		b. Pared costal		b. Genitales	
c. Piel y uñas		c. Timpanos		c. Mucosas		a. Visceras		a. Vascular	
2. Ojos		4. Orofaringe		6. Cuello		8. Pared abdominal		10. Membros superiores	
a. Párpados		a. Labios		d. Senos paranasales		b. Pared abdominal		b. Membros inferiores	
b. Conjuntivas		b. Lengua		a. Tiroides / masas		c. Flexibilidad		a. Fuerza	
c. Pupilas		c. Faringe		b. Movilidad		a. Dentición		b. Sensibilidad	
d. Córneas		d. Amígdalas		a. Mamas		b. Dolor		c. Marcha	
e. Motilidad		e. Dentadura		1. Tórax		10. Columna		d. Reflejos	
CON EVIDENCIA DE PATOLOGÍA MARCAR CON "X" Y DESCRIBIR EN LA SIGUIENTE SECCIÓN ANOTANDO EL NUMERAL.									
Observaciones									

E. RESULTADOS DE EXÁMENES GENERALES Y ESPECÍFICOS DE ACUERDO AL RIESGO Y PUESTO DE TRABAJO (IMAGEN, LABORATORIO Y OTROS)					
EXAMEN	FECHA <small>dd/mm/aaaa</small>	RESULTADO			

Observaciones:

F. DIAGNÓSTICO		PRE+ PRESUNTIVO	DEF+ DEFINITIVO	CIE	PRE	DEF
1	Descripción					
2						
3						

G. EVALUACIÓN MÉDICA DE RETIRO	
SE REALIZÓ LA EVALUACIÓN	<input type="checkbox"/> SI <input type="checkbox"/> NO
Observaciones:	

H. RECOMENDACIONES Y/O TRATAMIENTO
Descripción

CERTIFICO QUE LO ANTERIORMENTE EXPRESADO EN RELACIÓN A MI ESTADO DE SALUD ES VERDAD. SE ME HA INFORMADO MI ESTADO ACTUAL DE SALUD Y LAS RECOMENDACIONES PERTINENTES.

I. DATOS DEL PROFESIONAL					J. FIRMA DEL USUARIO	
FECHA <small>dd/mm/aaaa</small>		HORA		NOMBRES Y APELLIDOS	CÓDIGO	FIRMA Y SELLO

Anexo VII. Formato Certificado Aptitud Médica

Proyecto:

Sistematización de Fichas Médicas Ocupacionales y Test Psicológicos

Certificado de Aptitud Médica

<https://drive.google.com/file/d/1RJuY02p2ckWjf2mQ-s66qGGRcHQvmdk1/view?usp=sharing>

A. DATOS DEL ESTABLECIMIENTO - EMPRESA Y USUARIO					
INSTITUCIÓN DEL SISTEMA O NOMBRE DE LA EMPRESA	RUC	CIIU	ESTABLECIMIENTO DE SALUD	NÚMERO DE HISTORIA CLÍNICA	NÚMERO DE ARCHIVO
PRIMER APELLIDO	SEGUNDO APELLIDO	PRIMER NOMBRE	SEGUNDO NOMBRE	SEXO	PUESTO DE TRABAJO (CIJO)

B. DATOS GENERALES	
FECHA DE EMISIÓN:	<div> <div>aaaa</div> <div>mm</div> <div>dd</div> </div>
EVALUACIÓN:	INGRESO <input type="checkbox"/> PERIÓDICO <input type="checkbox"/> REINTEGRO <input type="checkbox"/> RETIRO <input type="checkbox"/>

C. APTITUD MÉDICA LABORAL	
Después de la valoración médica ocupacional se certifica que la persona en mención, es calificada como:	
<div> <div>APTO</div> <div>APTO EN OBSERVACIÓN</div> <div>APTO CON LIMITACIONES</div> <div>NO APTO</div> </div>	
DETALLE DE OBSERVACIONES:	

D. EVALUACIÓN MÉDICA DE RETIRO	
El usuario se realizó la evaluación médica de retiro	<div> <div>SI</div> <div>NO</div> </div>
Condición del diagnóstico	<div> <div>Presuntiva</div> <div>Definitiva</div> <div>No aplica</div> </div>
La condición de salud esta relacionada con el trabajo	<div> <div>SI</div> <div>NO</div> <div>No aplica</div> </div>

E. RECOMENDACIONES	
Descripción	

Con este documento certifico que el trabajador se ha sometido a la evaluación médica requerida para (el ingreso /la ejecución/ el reintegro y retiro) al puesto laboral y se ha informado sobre los riesgos relacionados con el trabajo emitiendo recomendaciones relacionadas con su estado de salud.



La presente certificación se expide con base en la historia ocupacional del usuario (a), la cual tiene carácter de confidencial.

F. DATOS DEL PROFESIONAL DE SALUD		G. FIRMA DEL USUARIO
<div> <div>NOMBRE Y APELLIDO</div> <div>CÓDIGO</div> <div>FIRMA Y SELLO</div> </div>		

Anexo VIII. Hoja de Evolución

[illegible]

Anexo IX. Receta Médica

 1859		Universidad Nacional de Loja	Subdirección de Seguridad y Salud Ocupacional
Loja - Ecuador			
Paciente: Edad			
Loja, de de CIE -10			
Rp:			
INDICACIONES:			

Anexo X. Desarrollo de la Metodología SXP

Proyecto:

Sistematización de Fichas Médicas Ocupacionales y Test Psicológicos

Metodología Híbrida SXP

Desarrollo de la Metodología Híbrida SXP

Para el correcto desarrollo del sistema, fue necesario dividir las actividades en varias iteraciones, que se componen de dos fases principales que son la planificación y desarrollo, dentro de esta última se realizan las pruebas unitarias y pruebas al sistema al terminar cada iteración. La siguiente fase de esta metodología es la de entrega y mantenimiento que se realiza en la iteración final.

A continuación, se detallan cada una de las iteraciones realizadas en el TT.

1. Iteración 1

1.1. Planificación-Definición

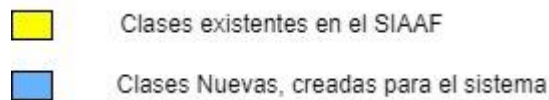
En esta primera fase se diseñaron las historias de usuario iniciales para detallar las funciones que se implementaron en el sistema, como se muestra a continuación:

Tabla 25. Historias de Usuario para la Iteración 1

ID	Historia de Usuario	Escenario	Criterios de Aceptación	Responsable
H001	<i>Como:</i> Usuario <i>Quiero:</i> Iniciar sesión <i>Para:</i> Acceder al sistema	Inicio de sesión	El sistema mostrará el formulario para Inicio de sesión. El sistema validará las credenciales a través del SAC	Josué Ortega
H002	<i>Como:</i> Médico <i>Quiero:</i> Buscar los datos de un paciente <i>Para:</i> obtener los datos personales del SIAAF-UNL y hacer el registro de los datos personales del paciente en las fichas médicas ocupacionales.	Ingresar número de cédula	El sistema muestra la lista de funcionarios registrados en el SIAAF-UNL. El sistema verifica si la información ingresada es correcta. El sistema se conectará a la Base de datos (BD) del SIAAF-UNL y se retornarán los datos personales del paciente.	Josué Ortega
H003	<i>Como:</i> Médico <i>Quiero:</i> Registrar fichas médicas iniciales <i>Para:</i> Ingresar los datos médicos	Registro de Datos Iniciales	El sistema muestra el formulario de ficha inicial para ingresar los datos necesarios. Todos los campos requeridos serán validados según el tipo	Yannela Castro

	iniciales y generales del paciente.		de dato que debe ser ingresado.	
--	-------------------------------------	--	---------------------------------	--

Para llevar a cabo la codificación del sistema, se realizó el diagrama de clases, en el que se consideraron solamente las clases contempladas en las historias de usuario descritas inicialmente, se debe tener en cuenta lo siguiente:



Así se obtiene el siguiente diagrama:

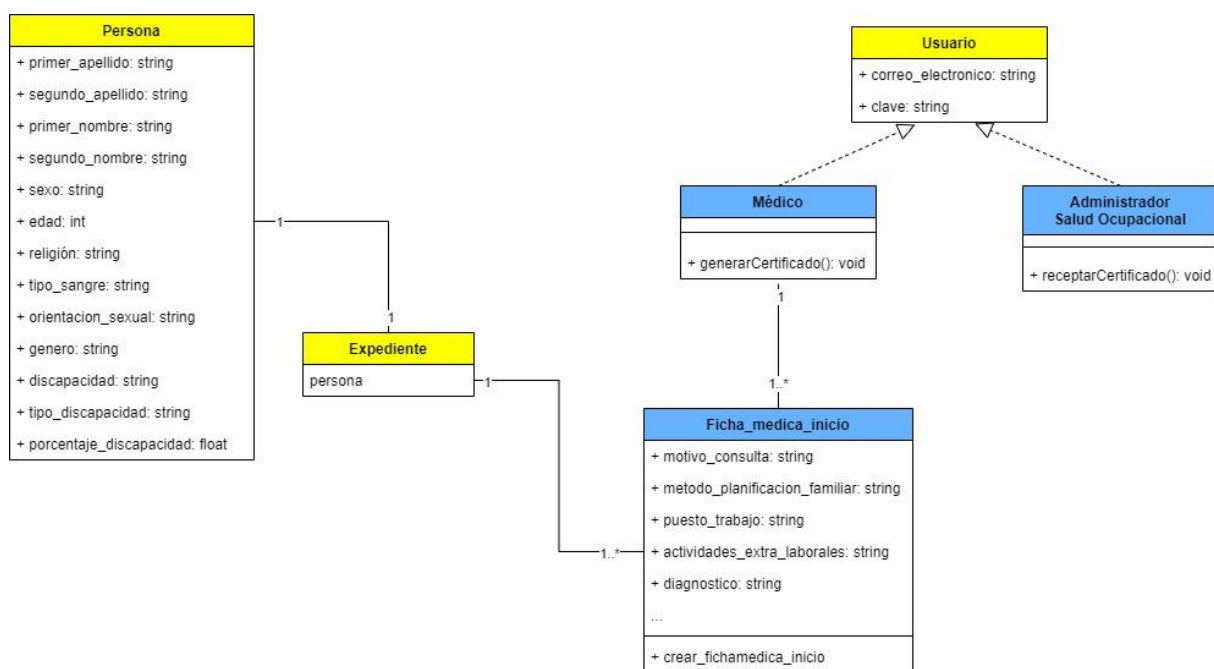


Figura 45. Diagrama de Clases - Iteración 1

Todos los atributos de la clase Ficha_medica_inicio se pueden observar en el *Anexo XI: Clases y atributos del sistema*.

1.2. Desarrollo

Para iniciar con esta fase se implementaron las funcionalidades descritas en las historias de usuario de la presente iteración, para el inicio de sesión de los usuarios se tomaron en cuenta las credenciales ya creadas dentro del SIAAF, con esto se dieron los permisos correspondientes de acuerdo a las funciones que realiza tanto Médico, como Administrador, dentro del sistema.

Posteriormente, para el usuario Médico, se realizó la función de obtener los datos necesarios de los funcionarios registrados en el SIAAF para empezar así con el registro de los datos médicos, y a partir de esto se procedió a implementar la función para registrar la ficha médica inicial de cada paciente.

A continuación, se muestran los métodos implementados dentro de esta primera iteración:

```
@permission_required('configuracion.view_crear_paciente', raise_exception=True, )
def crear_paciente(request):
    formulario_ficha = Ficha medicaForm(request.POST)
    url_id = request.GET['id']
    print('aquí esta la cedula de la url', url_id)
    buscar_funcionario = Funcionario.objects.get(id=url_id)
    edad = buscar_funcionario.usuario.persona.get_edad()
    print('edad', edad)
    context = {
        'funcionario': buscar_funcionario,
    }
    print(' encuentro la persona con cedula', buscar_funcionario.usuario.persona.numero_documento)
    if request.method == 'POST':
        datos = request.POST.dict()
        fm = Ficha medica()
        fm.ciiu = datos.get('ciiu')
        fm.ruc = buscar_funcionario.usuario.persona.numero_documento
        fm.nro_archivo = datos.get('nro_archivo')
        fm.establishcimiento_salud = datos.get('establecimiento_salud')
        fm.institucion_sistema_nompreempresa = datos.get('institucion_sistema_o_nompre_empresa')
        fm.num_historia_clinica = datos.get('num_historia_clinica')
        fm.primer_apellido = buscar_funcionario.usuario.persona.primer_apellido
        fm.segundo_apellido = buscar_funcionario.usuario.persona.segundo_apellido
        fm.primer_nombre = buscar_funcionario.usuario.persona.primer_nombre
        fm.grupo_sanguineo = buscar_funcionario.usuario.persona.tipo_sangre
        fm.porcentaje_discapacidad = buscar_funcionario.usuario.persona.porcentaje_discapacidad
        fm.tipo_discapacidad = buscar_funcionario.usuario.persona.tipo_discapacidad
        fm.discapacidad = buscar_funcionario.usuario.persona.discapacidad
        fm.orientacion_sexual = buscar_funcionario.usuario.persona.orientacion_sexual
        fm.identidad_genero = datos.get('identidad_genero')
        fm.lateralidad = datos.get('actividad_relevante')
        if (fm.save() != False):
            messages.success(request, 'Ficha medica registrada correctamente')
            print('Ficha medica registrada existosamente')
            return redirect('historias_clinicas:listar_Paciente')
    return render(request, 'historias_clinicas/index_ficha_medica_inicio.html', context)
```

Figura 46. Función Crear Ficha Médica al Paciente.

```

def crear_fichamedica_inicio(request):
    url_ruc = request.GET['id']
    print('crear_fichamedica_inicio y el id es:', url_ruc)
    paciente = Ficha_medica.objects.get(id=url_ruc)
    context2 = {
        'paciente': paciente,
    }
    print('paciente', paciente.edad)
    print(' este es el paciente', paciente.ruc)
    try:
        print('entro al try')
        ficha_medica_id = Ficha_medica.objects.get(id=url_ruc)
        listarFMI = Ficha_medica_inicio.objects.get(ficha_medica=url_ruc)
        if request.method == 'POST':
            print('formulario fichamedica_inicio')
            datos = request.POST.dict()
            crear_ficha = Ficha_medica_inicio()
            crear_ficha.motivo_consulta = datos.get('motivo_consulta')
            crear_ficha.antecedentes_clinicos_quirurgicos = datos.get('antecedentes_clinicos_')
            crear_ficha.menarquia = datos.get('menarquia')
            crear_ficha.ciclos = datos.get('ciclos')
            crear_ficha.fecha_ultima_mens = datos.get('fecha_ultima_mens')
            crear_ficha.gestas = datos.get('gestas')

            messages.success(request, 'Ficha medica inicio registrada correctamente')
            return redirect('/historias_clinicas/fichasmedicas/lista?id=' + url_ruc)

    print('edad', paciente.edad)
    return render(request, 'historias_clinicas/ficha_medica_inicio.html', context2)

```

Figura 47. Crear Ficha Médica de Inicio.

1.2.1. Pruebas Unitarias

En la fase de desarrollo se realizaron las pruebas unitarias de las funciones más relevantes, con las que se comprobó su correcto funcionamiento. Estas pruebas se muestran a continuación:

```

import pytest
from ProyectoPruebas.pruebas_realizadas.models import *

@pytest.mark.django_db
def test_crear_ficha_medica():
    ficha_medica = Ficha_medica()
    ficha_medica.num_historia_clinica = '1105899634'
    ficha_medica.establishment_salud = 'UNL'
    ficha_medica.grupo_sanguineo = 'ORH+'
    ficha_medica.save()
    return ficha_medica
assert ficha_medica.num_historia_clinica == "1105899634"

```

Figura 48. Test Crear Ficha Médica

```

@pytest.mark.django_db
def test_crear_ficha_medica_inicio():
    ficha_medica_inicio = Ficha_medica_inicio()
    ficha_medica_inicio.motivo_consulta = 'tos seca'
    ficha_medica_inicio.puesto_trabajo = 'Enfermería'
    ficha_medica_inicio.actividades_desempena = 'Atender Docentes, Empleados y Trabajadores'
    ficha_medica_inicio.antecedentes_familiares = 'Rosa Ortiz:Mamá'
    ficha_medica_inicio.ficha_medica = test_crear_ficha_medica()
    ficha_medica_inicio.save()
    return ficha_medica_inicio
assert ficha_medica_inicio.motivo_consulta == "tos seca"

```

Figura 49. Test Crear Ficha Médica de Inicio

2. Iteración 2

2.1. Planificación-Definición

En esta primera fase, de la segunda iteración, se diseñaron las historias de usuario que tienen que ver con la gestión de las fichas médicas ocupacionales, como se muestra a continuación:

Tabla 26. Historias de Usuario para la Iteración 2

ID	Historia de Usuario	Escenario	Criterios de Aceptación	Responsable
H004	<i>Como:</i> Médico <i>Quiero:</i> Registrar ficha periódica <i>Para:</i> Llevar un control del estado de los pacientes en su entorno de trabajo.	Registro de Datos de Ficha Periódica	El sistema mostrará la ventana de ficha periódica. El sistema retorna los datos personales. El sistema valida cada uno de los campos con los datos ingresados.	Yannela Castro
		Registrar Hoja de evolución	El sistema mostrará en la ventana de ficha periódica, la opción para acceder a la ventana de Hoja de Evolución. El sistema mostrará el formulario de hoja de evolución. El sistema valida cada uno de los campos con los datos ingresados.	

H005	<i>Como:</i> Médico <i>Quiero:</i> Actualizar los datos de la ficha periódica <i>Para:</i> Mantener al día los datos médicos del paciente.	Actualizar datos de Ficha Periódica	Una vez cargados los datos en el formulario de ficha periódica, el médico procede a actualizar los campos. El sistema valida cada uno de los campos con los datos ingresados.	Josué Ortega
		Actualizar Hoja de Evolución	El sistema carga los datos en el formulario de hoja de evolución. El sistema valida cada uno de los campos.	
H006	<i>Como:</i> Médico <i>Quiero:</i> Registrar fichas médicas de reintegro <i>Para:</i> Poder controlar el reintegro de un servidor a sus labores, en caso de haberse ausentado por más de 15 días.	Ingresar datos a la ficha de reintegro	El sistema mostrará la ventana con el formulario para la ficha de reintegro. El sistema carga los datos personales del paciente. Todos los campos serán validados según el tipo de dato que debe ser ingresado.	Yannela Castro

Una vez establecidas las historias de usuario para esta iteración, se realizaron ajustes en el diagrama de clases anterior, agregando las clases relacionadas a esta nueva iteración, obteniendo el siguiente diagrama:

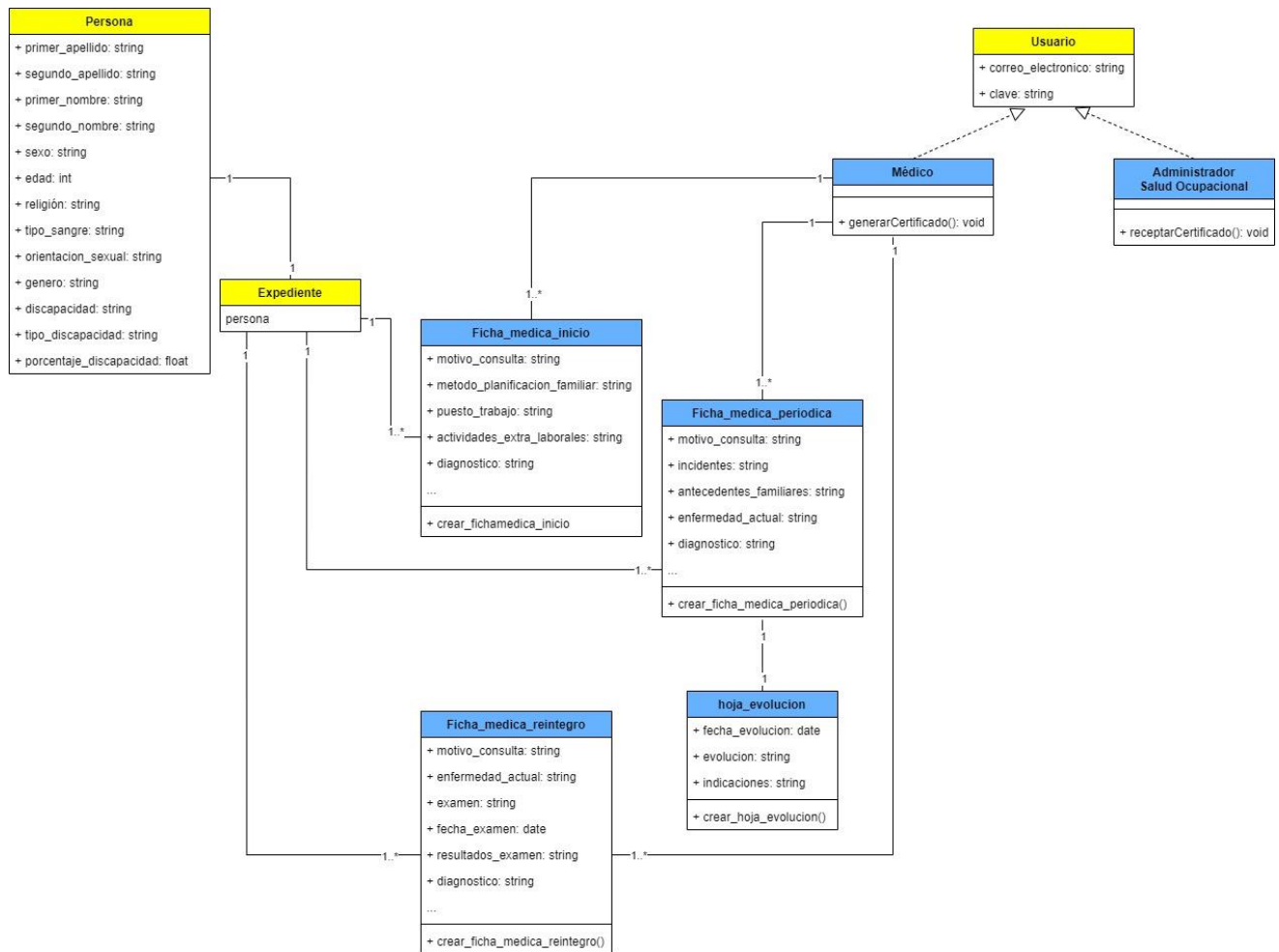


Figura 50. Diagrama de Clases - Iteración 2

Todos los atributos de las clases se pueden observar en el *Anexo XI: Clases y atributos del sistema*.

2.2. Desarrollo

Las funcionalidades implementadas en la segunda iteración tienen que ver con la gestión de las fichas médicas que debe registrar cada paciente, pues una vez obtenidos los datos personales del SIAAF, realizado en la iteración anterior, se procede con el registro de cada tipo de ficha médica, según sea el caso, además se implementó la función para poder actualizar datos necesarios.

A continuación, se muestran los métodos creados en esta iteración:

```

def crear_ficha_medica_periodica(request):
    url_ruc = request.GET['id']
    print('crear_fichamedica_inicio y el id es:', url_ruc)
    paciente = Ficha_medica.objects.get(id=url_ruc)
    context2 = {
        'paciente': paciente,
    }

    print('paciente', paciente.edad)
    print('este es el paciente', paciente.ruc)
    try:
        print('entro al try')

        ficha_medica_id = Ficha_medica.objects.get(id=url_ruc)
        listarFMI = Ficha_medica_periodica.objects.get(ficha_medica=url_ruc)
        print('datos', listarFMI.temperatura)
        if request.method == 'POST':
            print('formulario fichamedica_periodica')
            datos = request.POST.dict()
            crear_ficha = Ficha_medica_periodica()
            crear_ficha.motivo_consulta = datos.get('motivo_consulta')
            if (crear_ficha.save() != False):
                print('Ficha medica inico registrada existosamente')
                messages.success(request, 'Ficha medica periodica registrada correctamente')
                return redirect('/historias_clinicas/fichasmedicas/lista?id=' + url_ruc)

    print('edad', paciente.edad)
    return render(request, 'historias_clinicas/ficha_medica_periodica.html', context2)

```

Figura 51. Función Crear Ficha Médica Periódica.

```

def crear_hoja_evolucion(request):
    url_ruc = request.GET['id']
    print('hoja evolucion:', url_ruc)
    paciente = Ficha_medica.objects.get(id=url_ruc)
    lista = Hoja_evolucion.objects.filter(ficha_medica=url_ruc).order_by('id')
    print('nombres y apellidos', paciente.primer_nombre)
    if request.method == 'POST':
        print('formulario fichamedica_inicio')
        datos = request.POST.dict()
        crear_ficha = Hoja_evolucion()
        crear_ficha.fecha_evolucion = datos.get('fecha_evolucion')
        crear_ficha.evolucion = datos.get('evolucion')
        crear_ficha.indicaciones = datos.get('indicaciones')
        crear_ficha.ficha_medica = paciente
        if (crear_ficha.save() != False):
            print('Ficha medica inico registrada existosamente')
            messages.success(request, 'Hoja evolucion registrada correctamente')
            return redirect('/historias_clinicas/ficha_medica/hoja_evolucion?id=' + url_ruc)
            # return redirect('/historias_clinicas/fichasmedicas/lista?id=' + url_ruc)

    context = {
        'paciente': paciente,
        'lista': lista,
    }

    return render(request, 'historias_clinicas/hoja_evolucion.html', context)

```

Figura 52. Función Crear Hoja de Evolución

```

def crear_ficha_medica_reintegro(request):
    url_ruc = request.GET['id']
    print('crear_fichamedica_inicio y el id es:', url_ruc)
    paciente = Ficha_medica.objects.get(id=url_ruc)
    context2 = {
        'paciente': paciente,
    }
    print('paciente', paciente.edad)
    print(' este es el paciente', paciente.ruc)
    try:
        print('entro al try')
        ficha_medica_id = Ficha_medica.objects.get(id=url_ruc)
        listarFMI = Ficha_medica_reintegro.objects.get(ficha_medica=url_ruc)
        if request.method == 'POST':
            print('formulario fichamedica_inicio')
            datos = request.POST.dict()
            crear_ficha = Ficha_medica_reintegro()
            crear_ficha.motivo_consulta = datos.get('motivo_consulta')

            if (crear_ficha.save() != False):
                print('Ficha medica reintegro registrada existosamente')
                messages.success(request, 'Ficha medica reintegro registrada correctamente')
                return redirect('/historias_clinicas/fichasmedicas/lista?id=' + url_ruc)

    return render(request, 'historias_clinicas/ficha_medica_reintegro.html', context2)

```

Figura 53. Función Crear Ficha Médica de Reintegro.

```

def editar_ficha_medica(request):
    url_ruc = request.GET['ruc']
    print('el ruc es:', url_ruc)
    fm = Ficha_medica.objects.get(ruc=url_ruc)
    if request.method == 'POST':
        datos = request.POST.dict()
        fm.ciiu = datos.get('ciiu')
        fm.nro_archivo = datos.get('nro_archivo')
        fm.establishcimiento_salud = datos.get('establecimiento_salud')
        fm.institucion_sistema_nompreempresa = datos.get('institucion_sistema_o_nompre_empresa')
        fm.num_historia_clinica = datos.get('num_historia_clinica')
        fm.fecha_ingreso_trabajo = datos.get('fecha_ingreso_trabajo')
        fm.puesto_trabajo = datos.get('puesto_trabajo')
        fm.area_trabajo = datos.get('area_trabajo')
        fm.actividad_relevante = datos.get('actividad_relevante')

        fm.identidad_genero = datos.get('identidad_genero')
        fm.lateralidad = datos.get('actividad_relevante')
        if (fm.save() != False):
            messages.success(request, 'Paciente actualizado existosamente')
            print('Paciente actualizado existosamente')
            return redirect('historias_clinicas:listar_Paciente')
        else:

```

Figura 54. Función Editar Ficha Médica.

2.2.1. Pruebas Unitarias

Tal como la primera iteración, se realizaron las pruebas unitarias a cada método para comprobar que tienen un correcto funcionamiento, como se muestra a continuación:

```
@pytest.mark.django_db
def test_crear_ficha_medica_reintegro():
    ficha_medica_reintegro = Ficha_medica_reintegro()
    ficha_medica_reintegro.enfermedad_actual = 'Sinusitis'
    ficha_medica_reintegro.temperatura = '33 °C'
    ficha_medica_reintegro.descripcion_examen_fisico_regional = 'piel seca'
    ficha_medica_reintegro.ficha_medica = test_crear_ficha_medica()
    ficha_medica_reintegro.save()
    return ficha_medica_reintegro
assert ficha_medica_reintegro.enfermedad_actual == "Sinusitis"
```

Figura 55. Test Crear Ficha Médica de Reintegro

```
@pytest.mark.django_db
def test_crear_ficha_medica_periodica():
    ficha_medica_periodica = Ficha_medica_periodica()
    ficha_medica_periodica.cantidad_medificacion = '20 miligramos'
    ficha_medica_periodica.observaciones_accidentestrabajo_calificadoiess = 'Dolor en su pierna derecha por golpe'
    ficha_medica_periodica.diagnostico = 'Fractura de Ligamentos'
    ficha_medica_periodica.ficha_medica = test_crear_ficha_medica()
    ficha_medica_periodica.save()
    return ficha_medica_periodica
assert ficha_medica_periodica.diagnostico == "Fractura de Ligamentos"
assert ficha_medica_periodica.cantidad_medificacion == "20 miligramos"
```

Figura 56. Test Crear Ficha Médica Periódica

```
@pytest.mark.django_db
def test_crear_hoja_evolucion():
    hoja_evolucion = Hoja_evolucion()
    hoja_evolucion.fecha_evolucion = '20-07-2022'
    hoja_evolucion.evolucion = 'Presenta un resfriado leve'
    hoja_evolucion.indicaciones = 'No tomar cosas heladas'
    hoja_evolucion.ficha_medica = test_crear_ficha_medica()
    hoja_evolucion.save()
    return hoja_evolucion
assert hoja_evolucion.evolucion == "Presenta un resfriado leve"
```

Figura 57. Test Crear Hoja de Evolución

3. Iteración 3

3.1. Planificación-Definición

Para la última iteración, se planificaron las historias de usuario que complementan las funcionalidades del sistema, tales como la búsqueda del paciente, búsqueda del código CIE, gestión de certificados. Estas historias se describen a continuación:

Tabla 27. Historias de Usuario para la Iteración 3

ID	Historia de Usuario	Escenario	Criterios de Aceptación	Responsable
H007	<i>Como:</i> Médico <i>Quiero:</i> Registrar ficha médica de salida <i>Para:</i> Notificar y controlar la salida de un servidor de la institución.	Registro de datos en ficha de salida	El sistema mostrará la ventana con el formulario para la ficha de reintegro. El sistema carga los datos personales del paciente. Todos los campos serán validados según el tipo de dato que debe ser ingresado.	Yannela Castro
H008	<i>Como:</i> Médico <i>Quiero:</i> Obtener los códigos CIE, <i>Para:</i> Clasificar la o las patologías diagnosticadas en los pacientes, dentro de la ficha médica.	Obtener código CIE	El sistema muestra el formulario de ficha médica. El sistema carga los códigos relacionados con la nomenclatura de la patología ingresada. El sistema verifica que los campos estén completos.	Josué Ortega
H009	<i>Como:</i> Médico <i>Quiero:</i> Buscar un paciente <i>Para:</i> Poder visualizar los datos registrados anteriormente en las fichas médicas ocupacionales.	Búsqueda de paciente	El sistema mostrará la ventana para realizar la búsqueda por número de ficha médica. El sistema mostrará el campo para ingresar el número de ficha médica (número de cédula) de un paciente. El campo ingresado será validado por el sistema.	Yannela Castro

			El sistema se conectará a la base de datos del SIAAF. El sistema cargará los datos del paciente.	
H010	<i>Como:</i> Médico <i>Quiero:</i> Generar certificados <i>Para:</i> enviarlos al Administrador de Salud Ocupacional y tramitar el proceso que corresponda.	Generar Certificado de Aptitud Médica	El sistema mostrará la ventana con la lista de pacientes atendidos. El sistema muestra una ventana con el formulario de certificado de aptitud médica. El sistema verifica la información ingresada.	Josué Ortega
	<i>Como:</i> Administrador <i>Quiero:</i> Listar los certificados <i>Para:</i> Visualizar los certificados generados y enviarlos a las personas que corresponda.	Receptar Certificado	El sistema mostrará la ventana de notificaciones. El sistema carga el certificado generado.	

Una vez establecidas las historias de usuario para esta iteración se procedió a agregar las clases necesarias para completar el diagrama de clases, que se muestra a continuación:

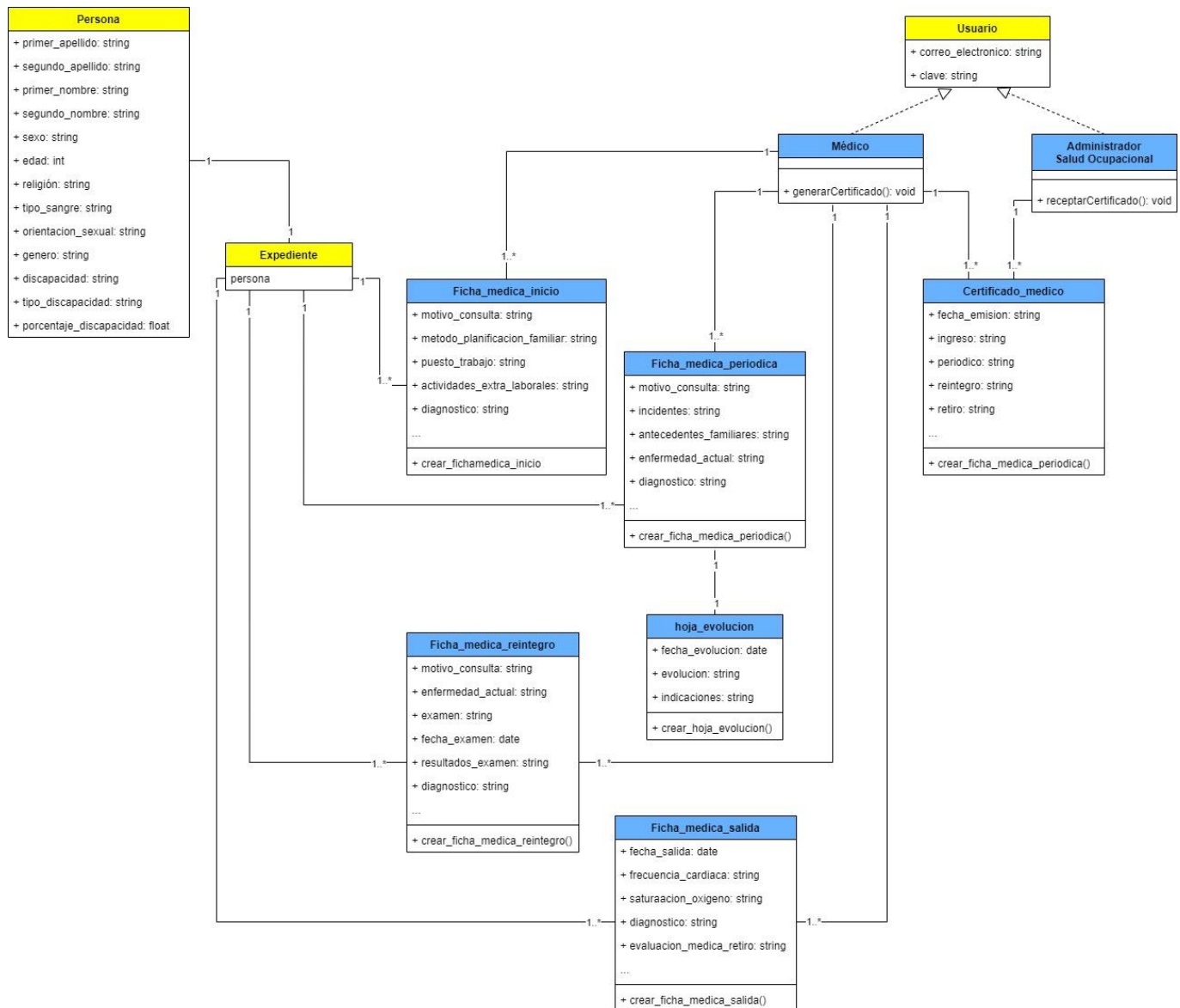


Figura 58. Diagrama de Clases - Iteración 3

Todos los atributos de las clases se pueden observar en el *Anexo XI: Clases y atributos del sistema*.

3.2. Desarrollo

Dentro de la tercera y última iteración se implementaron las funciones relacionadas con la gestión de certificados, es decir, el Médico tendrá la opción de generar el certificado médico del paciente, para luego ser enviado al Administrador, quien se encarga de revisar la información y enviar al departamento correspondiente, esto se lleva a cabo a través de notificaciones dentro del sistema. Además de esto, también se realizó la función para buscar un paciente a través de su número de ficha médica, que es el mismo número de cédula, con el fin de encontrar su información médica de manera rápida. Estas funciones se muestran a continuación:


```

def crear_ficha_medica_salida(request):
    url_ruc = request.GET['id']
    print('crear_fichamedica_inicio y el id es:', url_ruc)
    paciente = Ficha_medica.objects.get(id=url_ruc)
    context2 = {
        'paciente': paciente,
    }
    print('paciente', paciente.edad)
    print('este es el paciente', paciente.ruc)
    try:
        print('entro al try')
        ficha_medica_id = Ficha_medica.objects.get(id=url_ruc)
        listarFMI = Ficha_medica_salida.objects.get(ficha_medica=url_ruc)
        if request.method == 'POST':
            datos = request.POST.dict()
            crear_ficha = Ficha_medica_salida()
            crear_ficha.fecha_salida = datos.get('fecha_salida')
            crear_ficha.antecedentes_clinicos_quirurgicos = datos.get('antecedentes_clinicos_quirurgicos')
            crear_ficha.accidentestrabajo_calificadoiess = datos.get('accidentestrabajo_calificadoiess')
        print('edad', paciente.edad)

    return render(request, 'historias_clinicas/ficha_medica_salida.html', context2)

```

Figura 59. Función Crear Ficha Médica de Salida

```

def certificado_medico_pdf(request):
    certificado_medico = Certificado_medicoForm(request.POST)
    url_ruc = request.GET['ruc']
    print('certificado medico:', url_ruc)
    paciente = Ficha_medica.objects.get(ruc=url_ruc)
    fecha = datetime.today().strftime('%Y-%m-%d')
    context = {
        'paciente': paciente,
        'fecha': fecha,
    }
    if request.method == 'POST':
        datos = request.POST.dict()
        crear_certificado = Certificado_medico()
        crear_certificado.ingreso = datos.get('ingreso')
        crear_certificado.fecha_emision = datos.get('fecha_emision')
        crear_certificado.periodico = datos.get('periodico')
        crear_certificado.reintegro = datos.get('reintegro')
        crear_certificado.condicion_relacionada_si = datos.get('condicion_relacionada_si')
        crear_certificado.condicion_relacionada_no = datos.get('condicion_relacionada_no')
        crear_certificado.condicion_relacionada_no_aplica = datos.get('condicion_relacionada_no_aplica')
        crear_certificado.ficha_medica = paciente
        #crear_certificado.ficha_medica = paciente
        if (crear_certificado.save() != False):
            messages.success(request, 'Certificado registrado correctamente')
            print('Certificado registrado exitosamente')
            return redirect('historias_clinicas:listar_Paciente')
    # return redirect('/historias_clinicas/fichasmedicas/lista?id=' + url_ruc)
    return render(request, 'historias_clinicas/nueva_certificado.html', context)

```

Figura 60. Función Crear Certificado Médico

3.2.1. Pruebas Unitarias

A continuación, se muestran las pruebas unitarias que se realizaron también a las funciones de la tercera iteración:

```
@pytest.mark.django_db
def test_crear_ficha_medica_salida():
    ficha_medica_salida = Ficha_medica_salida()
    ficha_medica_salida.observaciones_enfermedadesprofesionales = 'Tiene cancer'
    ficha_medica_salida.talla = '1.72'
    ficha_medica_salida.fecha_examen = '29-08-2022'
    ficha_medica_salida.ficha_medica = test_crear_ficha_medica()
    ficha_medica_salida.save()
    return ficha_medica_salida
assert ficha_medica_salida.fecha_examen == "29-08-2022"
```

Figura 61. Test Crear Ficha Médica de Salida

```
@pytest.mark.django_db
def test_crear_certificado_medico():
    certificado_medico = Certificado_medico()
    certificado_medico.fecha_emision = '30-03-2022'
    certificado_medico.apto = 'SI'
    certificado_medico.apto_en_observacion = 'Debe evitar comer comidas saturadas'
    certificado_medico.nombre_apellido_medico = 'Dra. Iliana Voneth Figueroa Delgado'
    certificado_medico.descripcion_recomendacion = 'Llevar una comida balanceada durante las 3 comidas diarias'
    certificado_medico.ficha_medica = test_crear_ficha_medica()
    certificado_medico.save()
    return certificado_medico
assert certificado_medico.fecha_emision == "30-03-2022"
assert certificado_medico.apto == "SI"
```

Figura 62. Test Crear Certificado Médico

3.3. Entrega y Mantenimiento

Esta fase se llevó a cabo una vez que se culminó con todas las iteraciones y todas las funcionalidades del sistema han sido implementadas, por tanto, se entrega el producto final funcionando y cumpliendo con las necesidades de los usuarios. Además, se dan capacitaciones a los clientes para que hagan un uso correcto del sistema, complementando esto con el Manual de Usuario.

Anexo XI. Clases y Atributos

Proyecto:

Sistematización de Fichas Médicas Ocupacionales y Test Psicológicos

Clases y atributos implementados en el sistema

A continuación, en las siguientes tablas se describen todos los atributos que están contenidos dentro de sus respectivas clases:

Tabla 28. Clase Ficha_medica_inicio

Nombre de la clase: Ficha_medica_inicio	
Atributos	
motivo_consulta	antecedentes_clinicos_quirurgicos
menarquia	fecha_ultima_mens
gestas	partos
cesareas	abortos
papanicolaou	tiempo_papanicolaou
resultado_papanicolaou	colposcopia
tiempo_colposcopia	resultado_colposcopia
ecomamario	tiempo_ecomamario
resultado_eco_mamario	mamografía
tiempo_mamografia	resultado_mamografia
eco_prostatico	tiempo_eco_prostatico
resultado_eco_prostatico	antigeno_prostatico
tiempo_antigeno_prostatico	resultado_antigeno_prostatico
hijos_vivos	hijos_muertos
vida_sexual	metodo_planificacion_familiar
tipo_metodo_planificacion_familiar	tabaco
tiempo_consumo_tabaco	cantidad_tabaco
exconsumidor_tabaco	abstinencia_tabaco
alcohol	tiempo_consumo_alcohol
cantidad_alcohol	exconsumidor_alcohol
abstinencia_alcohol	otra_droga
tiempo_otra_droga	cantidad_otra_droga
abstinencia_otra_droga	exconsumidor_otra_droga
actividad_fisica	tipo_actividad_fisica
tiempo_actividad_fisica	medicación
cantidad_medicacion	empresa
puesto_trabajo	actividades_desempena
tiempo_trabajo	riesgo
observaciones_trabajo	accidentestrabajo_calificadoiess
observaciones_accidentestrabajo_calificadoiess	fecha_accidentestrabajo_calificadoiess
enfermedadesprofesionales_calificadoiess	observaciones_enfermedadesprofesionales
fecha_enfermedadesprofesionales	antecedentes_familiares
descripcion_antecedentes_familiares	factor_riesgo_puesto_trabajo
factor_riesgo_actividad	riesgo_fisico
riesgo_mecanico	riesgo_quimico
riesgo_biologico	riesgo_ergonomico
riesgo_sicosocial	riesgo_medidas
actividades_extra_laborales	enfermedad_actual
revision_organos_sistemas	descripcion_revision_organos_sistemas
presion_alta	temperatura
frecuencia_cardiaca	saturaacion_oxigeno
frecuencia_respiratoria	talla
peso	indice_masa_corporal
perimetro_abdominal	examen_fisico_regional

descripcion_examen_fisico_regional	piel
ojos	faringe
oído	nariz
cuello	torax
abdomen	columna
pelvis	enfermedades
neurologico	examen
fecha_examen	resultados_examen
diagnostico	apto
apto_observacion	apto_limitaciones
no_apto	tratamiento
Operaciones	
+ crear_fichamedica_inicio()	

Tabla 29. Clase Ficha_medica_periodica

Nombre de la clase: Ficha_medica_periodica	
Atributos	
motivo_consulta	antecedentes_clinicos_quirurgicos
tabaco	tiempo_consumo_tabaco
cantidad_tabaco	exconsumidor_tabaco
abstinencia_tabaco	alcohol
tiempo_consumo_alcohol	cantidad_alcohol
exconsumidor_alcohol	abstinencia_alcohol
otra_droga	tiempo_otra_droga
cantidad_otra_droga	abstinencia_otra_droga
exconsumidor_otra_droga	actividad_fisica
tipo_actividad_fisica	tiempo_actividad_fisica
medicación	cantidad_medicacion
incidentes	accidentestrabajo_calificadoiess
observaciones_accidentestrabajo_calificadoiess	fecha_accidentestrabajo_calificadoiess
enfermedadesprofesionales_calificadoiess	observaciones_enfermedadesprofesionales
fecha_enfermedadesprofesionales	antecedentes_familiares
descripcion_antecedentes_familiares	factor_riesgo_puesto_trabajo
factor_riesgo_actividad	riesgo_fisico
riesgo_mecanico	riesgo_quimico
riesgo_biologico	riesgo_ergonomico
riesgo_sicosocial	riesgo_medidas
enfermedad_actual	revision_organos_sistemas
descripcion_revision_organos_sistemas	presion_alta
temperatura	frecuencia_cardiaca
saturaacion_oxigeno	frecuencia_respiratoria
talla	peso
indice_masa_corporal	perimetro_abdominal
examen_fisico_regional	descripcion_examen_fisico_regional
piel	ojos
faringe	oído
nariz	cuello
torax	abdomen
columna	pelvis

enfermedades	neurologico
examen	fecha_examen
resultados_examen	diagnostico
apto	apto_observacion
apto_limitaciones	no_apto
tratamiento	
Operaciones	
+ crear_ficha_medica_periodica()	

Tabla 30. Clase Ficha_medica_reintegro

Nombre de la clase: Ficha_medica_reintegro	
Atributos	
motivo_consulta	enfermedad_actual
presion_alta	temperatura
frecuencia_cardiaca	saturation_oxigeno
frecuencia_respiratoria	talla
peso	indice_masa_corporal
perimetro_abdominal	examen_fisico_regional
descripcion_examen_fisico_regional	piel
ojos	faringe
oido	nariz
cuello	torax
abdomen	columna
pelvis	enfermedades
neurologico	examen
fecha_examen	resultados_examen
diagnostico	apto
apto_observacion	apto_limitaciones
no_apto	tratamiento
Operaciones	
+ crear_ficha_medica_reintegro()	

Tabla 31. Clase Ficha_medica_salida

Nombre de la clase: Ficha_medica_salida	
Atributos	
fecha_salida	antecedentes_clinicos_quirurgicos
accidentestrabajo_calificadoiess	observaciones_accidentestrabajo_calificadoiess
fecha_accidentestrabajo_calificadoiess	enfermedadesprofesionales_calificadoiess
observaciones_enfermedadesprofesionales	fecha_enfermedadesprofesionales
presion_alta	temperatura
frecuencia_cardiaca	saturation_oxigeno
frecuencia_respiratoria	talla
peso	indice_masa_corporal
perimetro_abdominal	examen_fisico_regional
descripcion_examen_fisico_regional	piel
ojos	faringe
oido	nariz

cuello	torax
abdomen	columna
pelvis	enfermedades
neurologico	examen
fecha_examen	resultados_examen
diagnostico	evaluacion_medica_retiro
observacion_evaluacion_medica_retiro	tratamiento
Operaciones	
+ crear_ficha_medica_salida()	

Tabla 32. Clase Certificado_medico

Nombre de la clase: Certificado_medico	
Atributos	
fecha_emision	ingreso
periodico	reintegro
retiro	apto
apto_en_observacion	apto_con_limitaciones
no_apto	detalle_observacion
codigo_medico	nombre_apellido_medico
descripcion_recomendacion	realizacion_evaluacion_medica_si
realizacion_evaluacion_medica_no	condicion_presuntiva
condicion_definitiva	condicion_no_aplica
condicion_relacionada_si	condicion_relacionada_no
condicion_relacionada_no_aplica	
Operaciones	
+ certificado_medico_pdf()	