



UNIVERSIDAD NACIONAL DE INGENIERÍA  
DIRECCION DE AREA DE CONOCIMIENTOS DE  
TECNOLGIA DE LA INFORMACION Y LA COMUNICACION

Trabajo monográfico para optar al título de  
Ingeniero en Computación.

### Tema

Sistema auxiliar para control de información académica de estudiantes de medicina mientras se encuentran realizando el período de internado rotativo en los hospitales escuela a los que son asignados.

### Autores

Br. Camilo Ernesto Martínez Carrillo

Br. Luis Alberto Bravo

Br. Sami Khaled Al Masagedi Castro

### Tutor

Msc. Gloria Talía Flores Quintana

Febrero del 2024

Managua, Nicaragua

## **Dedicatoria**

Primeramente a Dios por haberme dado la sabiduría necesaria para poder llegar hasta este momento, por haberme acompañado cada día de mi vida protegiéndome y dándome salud, a mis padres por ser un ejemplo a seguir, ustedes son el pilar fundamental en mi educación, de mi vida y de todo lo que soy, esto no hubiera sido posible sin contar con su apoyo incondicional, por estar ahí motivándome a esforzarme más para alcanzar mis metas y siempre ser un soporte en los momentos difíciles, a los docentes que fueron parte fundamental en el aprendizaje mientras pase por esta alma mater y a mi esposa e hijos por estar a mi lado motivándome a seguir adelante y no darme por vencido hasta alcanzar este momento.

**Luis Alberto Bravo**

Me gustaría expresar mi más sincero agradecimiento a todos aquellos que han jugado un papel importante en la realización de este trabajo monográfico, agradezco a mi familia su inquebrantable apoyo y comprensión. Su confianza en mis capacidades me proporcionó la motivación que necesitaba para perseverar a través de los desafíos de este esfuerzo. Un gran agradecimiento a nuestra tutora, cuya guía, paciencia y experiencia fueron fundamentales para dar forma a este proyecto. También quiero extender mi agradecimiento a mis compañeros de equipo, por sus valiosos conocimientos y críticas constructivas a lo largo de este viaje académico. Esta tesis no habría sido posible sin el apoyo colectivo, el aliento y la inspiración de todos los mencionados anteriormente, y estoy profundamente agradecido por sus contribuciones.

**Sami Khaled Al Massagedi Castro.**

Quiero agradecer ante todo a Dios por darme fortaleza, sabiduría, las ganas de superación como persona, por ser pieza vital para afrontar los retos de la vida y acompañarme en cada uno de ellos. Quiero dedicar este trabajo de investigación a mi familia por ser el motor que me permite avanzar y querer mejorar académicamente cada día, de igual manera, dedicar este proyecto a mis padres que inculcaron excelentes valores de vida en mi persona y que han sido soporte vital en mi vida, logrando llegar y alcanzar las metas propuestas hasta el día de hoy, a mis compañeros de equipo y tutor quienes, con su conocimiento y virtudes, me apoyaron y fueron pieza vital para la culminación de este proyecto monográfico.

**Camilo Ernesto Martínez Carrillo.**

## **Agradecimientos**

Agradecemos a Dios por habernos dotado de sabiduría y entendimiento para poder comprender el contenido impartido por nuestros docentes y ponerlo en práctica día a día en nuestra vida profesional, lo que nos permitió llegar hasta este momento importante para nosotros.

Agradecemos a los docentes de nuestra Alma mater por haber tenido la paciencia y disposición para enseñarnos nuevas habilidades que nos son y serán de suma utilidad en nuestra vida profesional, gracias por estar siempre disponibles para aclarar dudas y responder a nuestras consultas.

Gracias especiales a nuestra tutora Msc. Gloria Talía Flores Quintana, por ayudarnos durante este proceso, por estar disponible para nuestras consultas todo el tiempo, por siempre ayudarnos a mejorar en este proceso y acompañarnos hasta el final, gracias por compartir parte de su experiencia y sus consejos con nosotros.

No podemos dejar de agradecer a nuestras familias que siempre estuvieron ahí para nosotros, brindándonos apoyo, palabras de aliento, compartiendo esas noches de desvelos y de una u otra forma ayudándonos a llegar hasta el final, ¡nada de esto habría sido posible sin tener un equipo de apoyo tan sólido como ustedes que son los pilares que nos sostienen en nuestra vida profesional y personal, gracias totales!

## Contenido

1. Introducción.....	1
2. Antecedentes. ....	2
3. Justificación.....	3
4. Objetivos. ....	5
Objetivo General. ....	5
Objetivos Específicos.....	5
5. Marco Teórico. ....	6
5.1 ¿Qué es una interfaz web? .....	6
5.2 ¿Que es una aplicación web? .....	7
5.3 Herramientas de desarrollo web recomendadas. ....	7
5.3.1 HTML.....	7
5.3.2 JavaScript.....	8
5.3.3 CSS .....	8
5.3.4 ASP.NET .....	8
5.3.5 C#.....	9
5.3.6 SQL Server.....	9
5.4 Las herramientas metodológicas a utilizar. ....	9
5.4.1 Scrum.....	9
5.4.2 El ciclo de vida del software .....	10
5.4.3 Lenguaje unificado de Modelado (UML).....	10
5.4.3.1 Diagramas de casos de uso. ....	11
5.4.3.2 Diagramas de clases. ....	11
5.4.3.3 Diagramas de secuencia.....	12
5.4.3.4 Diagramas de actividad.....	13
5.4.3.5 Diagramas de estado. ....	13
5.4.3.6 Diagramas de componente. ....	14
5.4.3.7 Diagramas de Despliegue. ....	15
5.4.3.8 Diagramas de Paquete.....	15

5.4.3.9 Diagramas de Estructura compuesta. ....	16
5.4.3.10 Diagramas de Comunicación. ....	17
5.5 Opciones seleccionadas .....	18
6. Diseño Metodológico.....	18
6.1 Recopilación de la información.....	18
6.1.1 Entrevistas.....	18
6.1.2 Documento de Archivo. ....	19
6.1.3 Observaciones. ....	19
6.1.4 Focus groups online. ....	19
6.2 Ciclo de vida de desarrollo del software. ....	19
6.3 Metodología scrum.....	20
7. Estimación de costos del Proyecto.....	22
7.1. Alternativa número 1. ....	23
7.1.1 Factibilidad técnica. ....	23
7.1.1.1 Recurso Persona.....	23
7.1.1.2 Recurso Hardware.....	23
7.1.1.3 Recurso Software.....	24
7.1.2 Factibilidad Económica.....	25
7.2 Alternativa número 2. ....	26
7.2.1 Factibilidad Técnica.....	26
7.2.1.1 Recurso Persona.....	26
7.2.1.2 Recurso Hardware. ....	26
7.2.1.3 Recurso Software.....	27
7.2.2 Factibilidad Económica. ....	27
7.3. Alternativa número 3. ....	28
7.3.1. Factibilidad Técnica.....	29
7.3.1.3. Recurso Persona.....	29
7.3.1.2. Recurso Hardware.....	29
7.3.1.3. Recurso Software.....	29
7.3.2. Factibilidad Económica.....	30

8. Análisis y selección de alternativas .....	31
9. Factibilidad operativa. ....	32
10. Factibilidad legal.....	35
11. Analisis de Riesgos del sistema. ....	36
12. Fase de inicio .....	61
12.1 Especificación del producto.....	61
12.1.1 Módulos:.....	61
12.1.2 Roles .....	63
12.1.3 Tablas.....	64
12.2 Requerimientos funcionales .....	66
12.2.1 Diagrama de paquetes .....	66
12.2.1.1 Funciones del catálogo docente .....	66
12.2.1.2 Funciones del catálogo Estudiante.....	66
12.2.1.3 Funciones del catálogo Notas .....	67
12.2.1.4 Funciones del catálogo Modulo .....	67
12.2.1.5 Funciones del catálogo Universidad.....	67
12.2.1.6 Funciones del catálogo Reportes .....	67
12.2.1.7 Funciones del catálogo hospital .....	67
12.2.2 Requerimientos funcionales .....	68
12.3 Permisos de usuarios .....	79
13. Fase de elaboración.....	82
13.1 Casos de uso .....	82
13.2. Diagrama Entidad-Relación.....	85
13.3. Diagrama de Actividades.....	86
13.4. Arquitectura de software.....	91
13.4.1 Cliente .....	91
13.4.2 Servidor Web.....	91
13.4.3 Gestor de Bases de Datos .....	92
14. Fase de construcción .....	92
14.1. Modelado de datos.....	92

14.1.1 Modulo Docente .....	92
14.1.2 Modulo Estudiante.....	93
14.1.3 Modulo Notas .....	93
14.1.4 Modulo Universidad.....	93
14.1.5 Modulo módulo.....	94
14.1.6 Modulo Reportes.....	94
14.1.6 Modulo Hospital.....	94
14.2 Interfaz de usuario.....	95
14.3. Casos de Prueba.....	100
14.4 Resumen de pruebas funcionales .....	103
15. Conclusiones.....	104
16. Recomendaciones.....	105
17. Bibliografía .....	106
18. Cronograma de ejecución .....	110
19. Anexos .....	111

### **Indice de ilustraciones**

Ilustración 1 - Ciclo de vida del software.....	21
Ilustración 2 - Proceso SCRUM.....	22
Ilustración 3 – Gestión de riesgos .....	37
Ilustración 4 – Diagrama de paquetes.....	67
Ilustración 5 – Agregar médico interno/docente .....	82
Ilustración 6 – Agregar universidad/hospital.....	82
Ilustración 7 – Agregar departamento/municipio .....	83
Ilustración 8 – Agregar modulo/Período.....	83
Ilustración 9 – Agregar notas .....	84
Ilustración 10 – Modificar notas.....	84
Ilustración 11 – Generar reporte de notas.....	84
Ilustración 12 – Diagrama entidad relación .....	85

Ilustración 13 – Diagrama actividad – Primer inicio de sesión .....	86
Ilustración 14 – Diagrama actividad – Crear médico interno .....	86
Ilustración 15 – Diagrama actividad – Crear médico docente .....	87
Ilustración 16 – Diagrama actividad – Crear departamento .....	87
Ilustración 17 – Diagrama actividad – Crear municipio .....	88
Ilustración 18 – Diagrama actividad – Crear Hospital.....	88
Ilustración 19 – Diagrama actividad – Crear Modulo.....	89
Ilustración 20 – Diagrama actividad – Crear universidad .....	89
Ilustración 21 – Diagrama actividad – Crear Período.....	90
Ilustración 22 – Diagrama actividad – ingresar notas.....	90
Ilustración 23 – Diagrama actividad – Visualizar notas .....	91
Ilustración 24 – Pantalla de inicio de sesión.....	95
Ilustración 25 – Pantalla principal del sistema.....	95
Ilustración 26 – Pantalla para crear estudiante .....	96
Ilustración 27 – Pantalla para crear Docente .....	96
Ilustración 28 – Pantalla para crear Universidad.....	97
Ilustración 29 – Pantalla para crear Modulo .....	97
Ilustración 30 – Pantalla para modificar información de usuarios .....	98
Ilustración 31 – Pantalla para Modificar / desactivar Universidad .....	98
Ilustración 32 – Pantalla para Modificar / desactivar Modulo .....	98
Ilustración 33 – Pantalla para ingresar / modificar Notas .....	99
Ilustración 34 – Pantalla donde el estudiante genera reporte de notas.....	99
Ilustración 35 – Pantalla donde el subdirector genera reporte de notas.....	99

### **Indice de Tablas**

Tabla 1 - Tabla de requerimientos factor humano alternativa 1. ....	24
Tabla 2 - Tabla de requerimientos hardware alternativa 1. ....	24
Tabla 3 - Tabla de requerimientos software alternativa 1.....	25
Tabla 4 - Tabla de factibilidad económica 1. ....	26

Tabla 5 - Tabla de requerimientos factor humano alternativa 2. ....	27
Tabla 6 - Tabla de requerimientos hardware alternativa 2. ....	27
Tabla 7 - Tabla de requerimientos software alternativa 2.....	28
Tabla 8 - Tabla de factibilidad económica 2. ....	28
Tabla 9 - Tabla de requerimientos factor humano alternativa 3. ....	30
Tabla 10 - Tabla de requerimientos hardware alternativa 3. ....	30
Tabla 11 - Tabla de requerimientos software alternativa 3.....	30
Tabla 12 - Tabla de factibilidad económica 3. ....	31
Tabla 13 – Tabla de factibilidad operativa. ....	34
Tabla 14 – Tipos de riesgos. ....	32
Tabla 15 – Niveles de probabilidad de riesgo. ....	38
Tabla 16 – Niveles de Impacto de riesgo. ....	39
Tabla 17 – Objetivo afectado por riesgo.....	40
Tabla 18 – Diseño Matriz de Riesgo. ....	40
Tabla 19 – Escala de Riesgo. ....	40
Tabla 20 – Matriz de Riesgo. ....	41
Tabla 21 – Matriz de Prevención y mitigación de riesgos ....	54
Tabla 22 – Lista de tablas usadas en el sistema.....	64
Tabla 23 – Matriz de requerimientos funcionales. ....	68
Tabla 24 – Permisos de usuarios.....	79
Tabla 25 – Caso de prueba 1.....	100
Tabla 26 – Caso de prueba 2.....	100
Tabla 27 – Caso de prueba 3.....	100
Tabla 28 – Caso de prueba 4.....	101
Tabla 29 – Caso de prueba 5.....	101
Tabla 30 – Caso de prueba 6.....	102
Tabla 31 – Caso de prueba 7.....	102
Tabla 32 – Caso de prueba 8.....	103
Tabla 33 – Caso de prueba 9.....	103

## **1. Introducción.**

El presente documento describe el desarrollo de un sistema auxiliar para los médicos-docentes a cargo del programa de internado rotativo que se lleva a cabo en los diferentes hospitales escuela a nivel nacional y que tienen la necesidad de poder llevar un control digital, de una manera sistematizada sobre la información tanto personal como académica de los estudiantes aspirantes a médicos que cursan sus módulos durante el tiempo de internado, permitiendo la generación de reportes históricos y en tiempo real, al mismo tiempo dicho sistema estandariza el proceso de recopilación, manejo y presentación de la información.

Este es un periodo crítico ya que incluye tanto aprendizaje por parte de los practicantes como evaluación del desempeño de ellos en las áreas a las que han sido asignados por parte de los médicos-docentes y esta información luego debe ser analizada, calificada y presentada por los médicos-docentes a la directiva de cada uno de estos hospitales y del SILAIS para realizar la toma de decisiones sobre la ubicación de cada uno de los futuros médicos, las áreas en las que deben mejorar, que tan bien han asimilado el aprendizaje teórico y donde serán ubicados para su servicio social que es la etapa final de sus estudios.

Este sistema es creado utilizando las metodologías ágiles relacionadas al desarrollo del software y al ciclo de vida del software, tales como SCRUM, se hizo uso del gestor de bases de datos SQL Server, así como ASP y Bootstrap 4.6 para la creación de interfaces gráficas, todo esto tomando en cuenta los estándares de creación de software y tomando en consideración los requerimientos especiales de los usuarios finales del sistema tanto académicos como de manejo de información.

## **2. Antecedentes.**

El ser humano siempre ha buscado medios que le permitan realizar una toma de decisiones efectiva, en cada una de las operaciones que realiza, la salud es un punto clave y tiene muchas ramas o áreas en las que una persona se puede especializar, para saber cuál es el área en la que mejor se desempeña un estudiante de medicina existe un internado rotativo en el cual los aspirantes a médicos deben cumplir luego de sus clases teóricas en la universidad, para esto se le asigna un hospital escuela en el cual deben pasar por los cuatro principales módulos de la medicina, en cada uno de estos se les evalúa y el desempeño tanto en las prácticas clínicas como en los seminarios teóricos que reciben y se les asigna una nota, que luego sirve de referente para decidir en qué hospital servirán su servicio social obligatorio para culminar su carrera.

La gestión, registro y almacenamiento de estas notas es un proceso clave ya que permite a las autoridades superiores del ministerio de salud poder cubrir la demanda de los usuarios del sistema de salud, de acuerdo con los recursos disponibles, sin embargo el proceso de recopilación de notas para los egresados de medicinas que están en el internado rotativo es un proceso extenso llevado a cabo de manera manual por cada uno de los docentes en los distintos hospitales escuelas del país, donde son asignados cada uno de los futuros doctores utilizando hojas manuales y luego transcribiendo esto en un formato de Excel que posteriormente es consolidado por una sola persona a nivel nacional.

Se realizó una revisión tanto a nivel nacional como internacional, no encontrándose un sistema que se acoplara a las necesidades que requiere el ministerio de salud para llevar a cabo el control de este periodo de internado rotativo.

### **3. Justificación.**

El año de internado rotativo, es un proceso que viene luego de la culminación de los estudios universitarios en las aulas de clase y antes del año de servicio social obligatorio, durante este tiempo se les imparten seminarios teóricos y realizan módulos prácticos, este proceso se repite cuatro veces durante el internado, tanto los seminarios como las prácticas generan notas que deben ser almacenadas, presentadas a las autoridades superiores de salud y los estudiantes mismo deben de tener acceso a poder ver sus notas.

El sistema de salud de Nicaragua carece de una herramienta informática dedicada al manejo y administración de las notas de los estudiantes de medicina en su internado rotativo, lo anterior trae como consecuencia retraso en la entrega de información ya que la misma es recopilada en distintas versiones de Word, Excel y hasta a través de procesos manuales y la generación de reportes históricos y en tiempo real, esto es algo que puede tornarse aún más difícil dependiendo del volumen de esta información.

La ejecución de esta tarea requiere conocer cada uno de los procesos que lleva a cabo el docente junto con los reportes que necesita para evaluar el desempeño de sus estudiantes. Este proceso es crítico para la dirección de cada uno de los hospitales escuela y para los docentes ya que les permite realizar una macro evaluación y una micro evaluación de cada una de las áreas en las que se desempeñan los futuros doctores y en base a eso tomar decisiones sobre la ubicación de ellos una vez egresados, por lo que esto demanda que el docente disponga de instrumentos auxiliares que le apoyen en la labor de evaluación y toma de decisiones.

Tomando en consideración el actual proceso donde se escribe a puño y letra toda la información, las demoras que esto causa y los inconvenientes que se presentan en este proceso, se detalla a continuación la creación de un sistema para realizar de manera estandarizada y que permita en tiempo real poder llevar el control de esta información, de manera que las notas puedan estar disponibles en línea para que las autoridades superiores del ministerio de salud puedan recibirlas sin demoras, a la vez esto permite que los estudiantes puedan ver las notas en tiempo real, también se disminuye la carga de trabajo de los docentes al darles una herramienta de última generación y un proceso estandarizado a seguir para la evaluación de los futuros médicos que actualmente están cursando el internado rotativo en los diferentes hospitales escuela en los cuales son asignados.

## **4. Objetivos.**

### **Objetivo General.**

- ❖ Desarrollar un sistema que facilite el control y monitoreo de información, así como la generación de reportes relacionado a los alumnos de los últimos años de medicina que cursan su internado rotativo en cada una de las áreas a las que son asignados en los diferentes hospitales escuela con los que cuenta el sistema de salud nicaragüense.

### **Objetivos Específicos.**

- ❖ Proporcionar a los docentes de medicina encargados del seguimiento de estudiantes internos de los últimos años de la carrera de medicina con un sistema para el control y monitoreo de las notas.
- ❖ Analizar los requerimientos del sistema para establecer el alcance puntual del proyecto.
- ❖ Diseñar la arquitectura del sistema acorde a los requerimientos identificados auxiliándose de las herramientas UML disponibles para esto.
- ❖ Codificar los distintos módulos del sistema según la arquitectura definida para este cumpliendo con cada uno de los requisitos previamente levantados.
- ❖ Validar el correcto funcionamiento del sistema desarrollado mediante las pruebas de software y estándares normados.

## 5. Marco Teórico.

El desarrollo de un sistema informático es un proceso complejo que implica la utilización de diferentes herramientas y tecnologías. Una combinación eficaz de HTML, JavaScript, CSS, ASP.NET, C# y SQL Server siempre aplicando la metodología de desarrollo del software correcto puede ser una opción poderosa para el desarrollo de aplicaciones web y de escritorio. A continuación, se presentan los conceptos asociados a las herramientas utilizadas en el desarrollo del sistema.

### 5.1 ¿Qué es una interfaz web?

Una interfaz web, también conocida como interfaz de usuario web o IU web, es la interfaz gráfica de usuario (GUI) que se presenta en una página web a través de la cual los usuarios interactúan con un sitio web o una aplicación web<sup>1</sup>. Permite a los usuarios ingresar información, solicitar servicios o datos y recibir comentarios del sistema.

Una interfaz web generalmente consta de elementos visuales como cuadros de texto, botones, menús, formularios e íconos que permiten a los usuarios realizar diversas acciones en el sitio web o la aplicación web. Está diseñado para ser intuitivo y fácil de usar, lo que permite a los usuarios navegar fácilmente por el sitio web y acceder a la información o los servicios que necesitan.

Las interfaces web se crean utilizando tecnologías de desarrollo web como HTML, CSS y JavaScript. El diseño y la funcionalidad de una interfaz web son consideraciones importantes en el desarrollo de cualquier sitio web o aplicación web, ya que pueden afectar significativamente la experiencia del usuario y la usabilidad general.

<sup>1</sup> MDN contributors, (Agosto 28, 2023). HTML Basics, Mozilla Developers Network Web Docs.

## 5.2 ¿Que es una aplicación web?

Una aplicación web o app web para abreviar, es una aplicación de software a la que se accede a través de Internet mediante un navegador web. A diferencia de las aplicaciones de escritorio tradicionales, las aplicaciones web se ejecutan completamente dentro de un navegador web y no requieren instalación ni descargas en el dispositivo del usuario. Las aplicaciones web generalmente se crean utilizando tecnologías web como HTML, CSS y JavaScript, y se alojan en un servidor web<sup>2</sup>. Se puede acceder a ellos en cualquier dispositivo con conexión a Internet y un navegador web, lo que los hace muy accesibles y convenientes para los usuarios.

Las aplicaciones web son una forma popular y poderosa de crear y entregar aplicaciones de software, con una amplia gama de usos y beneficios tanto para los usuarios finales como para los desarrolladores.

## 5.3 Herramientas de desarrollo web recomendadas para la elaboración de este proyecto.

**5.3.1 HTML** (Hypertext Markup Language) es un lenguaje basado en texto que es interpretado por navegadores web utilizado para crear y estructurar contenido en la web<sup>3</sup>, cuando un usuario visita una página web, su navegador lee el código HTML y lo utiliza para mostrar el contenido de la página en un formato visual, también proporciona un conjunto de etiquetas y atributos que definen los diversos elementos de una página web, como encabezados, párrafos, imágenes, enlaces, formularios y más. HTML es una tecnología fundamental de la World Wide Web, que permite a los desarrolladores crear y mostrar páginas web a las que pueden acceder usuarios de todo el mundo mediante un navegador web, aunque se usa a menudo junto con otras tecnologías web, como CSS (hojas de estilo en cascada) y JavaScript, para crear experiencias web más sofisticadas e interactivas.

<sup>2</sup> ESIC Business & Marketing School, (2023, Enero 12). ¿Qué son las aplicaciones web? ESIC.

<sup>3</sup> MDN contributors, (Julio 18, 2023). Conceptos Basicos de HTML, Mozilla Developers Network Web Docs.

**5.3.2 JavaScript** es un lenguaje de programación de alto nivel que se usa ampliamente para crear contenido web interactivo y dinámico<sup>4</sup>. Se introdujo por primera vez en 1995 y desde entonces se ha convertido en uno de los lenguajes de programación más populares del mundo, ya que es un lenguaje de programación dinámico y altamente versátil que en la mayoría de los casos se ejecuta en el lado del cliente. Permitiendo así agregar interactividad y funcionalidad a las páginas web por medio de características tales como validación de formularios, manejo de eventos y creación de animaciones, manipular el DOM (Modelo de objeto de documento), realizar solicitudes AJAX (JavaScript asíncrono y XML), también se usa en el lado del servidor con Node.js para crear aplicaciones y API del lado del servidor.

**5.3.3 CSS** (hojas de estilo en cascada) es un lenguaje de estilo que se utiliza para definir la apariencia visual de los documentos HTML y XML, incluidos los colores, los diseños, las fuentes y otros elementos de diseño<sup>5</sup>. Se introdujo por primera vez en 1996 y desde entonces se ha convertido en una parte esencial del desarrollo web ya que permite a los desarrolladores crear estilos reutilizables que se pueden aplicar a varias páginas. Esto facilita el mantenimiento de un diseño coherente en todo el sitio web y también ayuda a realizar actualizaciones y cambios en el diseño.

**5.3.4 ASP.NET** es un marco de aplicación web desarrollado por Microsoft que permite a los desarrolladores crear aplicaciones y servicios web dinámicos<sup>6</sup>. Se introdujo por primera vez en 2002 y desde entonces se ha convertido en una tecnología popular para crear aplicaciones web, esta proporciona un amplio conjunto de bibliotecas, componentes y herramientas que permiten a los desarrolladores crear aplicaciones web complejas con funciones como autenticación, autorización, acceso a bases de datos y más. Admite varios lenguajes de programación, incluidos C# y Visual Basic, y se puede utilizar con una variedad de tecnologías de desarrollo web, como HTML, CSS y JavaScript.

<sup>4</sup> MDN contributors, (Julio 24, 2023). JavaScript, Mozilla Developers Network Web Docs.

<sup>5</sup> MDN contributors, (Marzo 13, 2023). CSS, Mozilla Developers Network Web Docs.

<sup>6</sup> Application architecture for .NET: Designing applications and services, Microsoft Press, 2003,

**5.3.5 C#** (pronunciado "C-sharp") es un lenguaje de programación de alto nivel desarrollado por Microsoft que se usa ampliamente para desarrollar una variedad de aplicaciones, incluidas aplicaciones de escritorio, aplicaciones web, aplicaciones móviles, juegos y más<sup>7</sup>. Se introdujo por primera vez en 2000 y desde entonces se ha convertido en uno de los lenguajes de programación más populares del mundo.

**5.3.6 SQL Server** es un sistema de gestión de bases de datos relacionales (RDBMS) desarrollado por Microsoft. Es una plataforma de base de datos potente y escalable que se usa ampliamente para almacenar, administrar y analizar grandes cantidades de datos en entornos empresariales<sup>8</sup>.

## **5.4 Las herramientas metodológicas a seguir en la elaboración de esta aplicación.**

**5.4.1 Scrum** es un marco ágil para administrar y completar proyectos complejos. Se introdujo por primera vez en la década de 1990 y desde entonces se ha convertido en una metodología popular para el desarrollo de software, así como para otros tipos de proyectos. Se basa en un conjunto de principios y valores que enfatizan la colaboración, la flexibilidad y la mejora continua. Está diseñado para ser liviano y adaptable, con un enfoque en brindar valor al cliente de manera oportuna y eficiente<sup>9</sup>.

En Scrum, los proyectos se dividen en una serie de iteraciones de tiempo limitado denominadas sprints, que suelen durar entre una y cuatro semanas. Al comienzo de cada sprint, el equipo realiza una reunión de planificación para identificar el trabajo que debe realizarse y establecer un objetivo de sprint. A lo largo del sprint, el equipo trabaja en conjunto para completar el trabajo, con reuniones diarias para monitorear el progreso e identificar cualquier obstáculo.

<sup>7</sup> Arnaud, B (Febrero 2021) ASP.NET con C# en Visual Studio 2019

<sup>8</sup> AXELOS Limited, (Mayo 18, 2021)¿Que es el SQL? y ¿Para qué sirve?, Nextech.

<sup>9</sup> Menzinsky, A., López, G., Palacio, J (Julio 2016) SCRUM Manager

5.4.2 El ciclo de vida del software, también conocido como ciclo de vida de desarrollo de software (SDLC), es el proceso mediante el cual se desarrolla, prueba, implementa y mantiene el software. El ciclo de vida del software generalmente consta de varias fases, cada una de las cuales está diseñada para lograr objetivos específicos y producir artefactos específicos<sup>10</sup>. Las diferentes fases del ciclo de vida del software pueden variar según la metodología de desarrollo específica utilizada, pero las fases más utilizadas son:

- Planificación
- Análisis
- Diseño
- Implementación
- Prueba
- Despliegue
- Mantenimiento

#### 5.4.3 Lenguaje unificado de Modelado (UML).

UML significa "Lenguaje de Modelado Unificado" en inglés ("Unified Modeling Language" en inglés) y es un lenguaje visual utilizado en la ingeniería de software para diseñar, documentar y comunicar sistemas de software<sup>11</sup>.

UML proporciona un conjunto de diagramas gráficos estandarizados que permiten a los desarrolladores y otros profesionales de software representar visualmente aspectos clave de un sistema de software, como sus componentes, estructura, comportamiento y procesos. Estos diagramas pueden ser utilizados para describir los requisitos del sistema, diseñar soluciones, comunicar ideas a los equipos de desarrollo y documentar sistemas existentes.

Los diagramas UML más comunes incluyen

<sup>10</sup> Prezi. (21 de Mayo de 2014). Ciclo de vida de un Software según Pressman, prezi.com.

<sup>11</sup> Pérez, R. (2 de Abril de 2011). Metodologías para el Desarrollo de Aplicaciones Web, prezi.com.

#### 5.4.3.1 Diagramas de casos de uso.

Los diagramas de caso de uso son un tipo de diagrama UML utilizado para representar visualmente los requisitos funcionales de un sistema de software desde el punto de vista de los usuarios o actores del sistema.

Un caso de uso describe una secuencia de acciones o eventos que un actor (por ejemplo, un usuario, un sistema externo o un dispositivo) realiza para lograr un objetivo dentro del sistema. El objetivo puede ser realizar una acción, obtener una información o cumplir una tarea específica dentro del sistema.

En un diagrama de caso de uso, se representan los actores del sistema como círculos o figuras humanas y los casos de uso como óvalos. Las líneas que conectan los actores y los casos de uso representan las interacciones entre ellos. Los casos de uso también pueden tener relaciones de inclusión o extensión entre ellos, para indicar cómo se relacionan los casos de uso en el sistema.

Los diagramas de caso de uso son útiles para la identificación y comprensión de los requisitos del sistema desde la perspectiva del usuario. Son una herramienta útil para comunicar de manera efectiva con los interesados del proyecto y pueden ayudar a los equipos de desarrollo de software a desarrollar un sistema que satisfaga las necesidades y expectativas de los usuarios finales.

#### 5.4.3.2 Diagramas de clases.

Los diagramas de clase son un tipo de diagrama UML utilizado para representar la estructura estática de un sistema de software, mostrando las clases que componen el sistema, sus atributos, métodos y las relaciones entre ellas.

Una clase en un diagrama de clase representa una plantilla para la creación de objetos. Los atributos de la clase son las variables que definen el estado de los objetos creados a partir de la clase, mientras que los métodos son las funciones que definen el comportamiento de los objetos.

En un diagrama de clase, las clases se representan como rectángulos divididos en tres secciones:

- La primera sección indica el nombre de la clase.
- La segunda sección indica los atributos de la clase.
- La tercera sección indica los métodos de la clase.

Las relaciones entre las clases se representan mediante líneas y flechas, que indican el tipo de relación (por ejemplo, herencia, asociación, agregación o composición) y la multiplicidad de la relación.

Los diagramas de clase son útiles para modelar y diseñar la estructura de un sistema de software y para visualizar cómo se relacionan las clases entre sí. Son una herramienta valiosa para la comunicación entre los miembros del equipo de desarrollo de software y pueden ayudar a garantizar que el diseño del sistema sea sólido y coherente.

#### 5.4.3.3 Diagramas de secuencia.

Un diagrama de secuencia es un tipo de diagrama UML que se utiliza para representar visualmente la interacción entre objetos en un sistema de software.

Un diagrama de secuencia muestra cómo los objetos interactúan entre sí para lograr un objetivo específico. Cada objeto se representa como una línea vertical llamada "línea de vida", que muestra la duración de vida del objeto. Las interacciones entre los objetos se muestran como flechas horizontales que conectan las líneas de vida. Cada flecha representa un mensaje que un objeto envía o recibe de otro objeto.

En un diagrama de secuencia, se pueden mostrar diferentes tipos de mensajes, como mensajes sincrónicos, asincrónicos, de retorno y de excepción. Los mensajes sincrónicos son aquellos en los que el objeto que envía el mensaje espera una respuesta antes de continuar, mientras que los mensajes asincrónicos son aquellos en los que el objeto que envía el mensaje no espera una respuesta antes de continuar.

Los diagramas de secuencia son útiles para modelar y diseñar la interacción entre objetos en un sistema de software y para visualizar cómo se comporta el sistema en diferentes situaciones. Son una herramienta valiosa para la comunicación entre los miembros del equipo de desarrollo de software y pueden ayudar a garantizar que el diseño del sistema sea sólido y coherente.

#### 5.4.3.4 Diagramas de actividad.

Un diagrama de actividad es un tipo de diagrama UML utilizado para modelar el comportamiento dinámico de un proceso de negocio o de un flujo de trabajo en un sistema de software.

Un diagrama de actividad muestra la secuencia de actividades o tareas que se realizan en un proceso o flujo de trabajo. Cada actividad se representa como un rectángulo con bordes redondeados y se conecta con flechas que representan la secuencia de actividades. Las decisiones se representan mediante rombos y se conectan con flechas que representan los posibles resultados de la decisión.

Los diagramas de actividad pueden incluir también otros elementos, como bucles, ramificaciones, particiones y barras de sincronización. Los bucles se utilizan para representar actividades que se realizan repetidamente, mientras que las ramificaciones se utilizan para representar la toma de decisiones en el flujo de trabajo. Las particiones se utilizan para dividir las actividades en grupos lógicos, y las barras de sincronización se utilizan para mostrar la concurrencia de actividades.

Los diagramas de actividad son útiles para modelar y diseñar procesos de negocio y flujos de trabajo en un sistema de software. Son una herramienta valiosa para la comunicación entre los miembros del equipo de desarrollo de software y pueden ayudar a garantizar que el diseño del sistema sea sólido y coherente.

#### 5.4.3.5 Diagramas de estado.

Un diagrama de estado es un tipo de diagrama UML que se utiliza para modelar el comportamiento dinámico de un objeto o una clase en un sistema de software.

Un diagrama de estado muestra los diferentes estados que puede tener un objeto o una clase, así como las transiciones que ocurren entre los estados. Cada estado se representa como un rectángulo con bordes redondeados, y las transiciones se representan como flechas entre los estados.

Los eventos que ocurren en el sistema de software pueden activar transiciones entre los estados. Estos eventos se representan como etiquetas en las flechas que conectan los estados. También se pueden especificar las acciones que ocurren cuando se produce una transición.

Los diagramas de estado son útiles para modelar y diseñar el comportamiento de objetos o clases en un sistema de software. Son una herramienta valiosa para la comunicación entre los miembros del equipo de desarrollo de software y pueden ayudar a garantizar que el diseño del sistema sea sólido y coherente. Además, los diagramas de estado pueden ser utilizados para identificar posibles errores en el diseño del sistema y para optimizar su funcionamiento.

#### 5.4.3.6 Diagramas de componente.

Los diagramas de componentes son un tipo de diagrama en UML que se utilizan para modelar la estructura y las relaciones entre los componentes de un sistema de software.

Los componentes son módulos de software que encapsulan una funcionalidad específica del sistema y se pueden reutilizar en diferentes sistemas. Los diagramas de componentes se utilizan para describir cómo los componentes de un sistema interactúan entre sí y con el entorno en el que se ejecutan.

Un diagrama de componentes consta de componentes, interfaces, conectores y artefactos. Los componentes representan los módulos de software, las interfaces definen cómo los componentes interactúan entre sí y con el entorno, y los conectores representan las relaciones entre los componentes y las interfaces.

Además, los artefactos son elementos que se utilizan para representar los archivos de código fuente, bibliotecas y otros recursos necesarios para la implementación de los componentes.

Los diagramas de componentes son útiles para modelar la estructura de los componentes de un sistema de software, sus relaciones y dependencias, y cómo interactúan con otros componentes y con el entorno en el que se ejecutan.

#### 5.4.3.7 Diagramas de Despliegue.

Los diagramas de despliegue son un tipo de diagrama en UML que se utilizan para modelar la arquitectura física de un sistema de software y su entorno de ejecución.

Estos diagramas representan los componentes de hardware y software que componen el sistema y sus interacciones en un entorno de ejecución específico. Los diagramas de despliegue son especialmente útiles para describir cómo se instala, configura y ejecuta un sistema de software en un entorno específico.

Un diagrama de despliegue consta de nodos, componentes y relaciones. Los nodos representan el hardware o el entorno de software, como servidores, routers, bases de datos, etc. Los componentes representan el software que se ejecuta en los nodos, como servidores de aplicaciones, bases de datos, servicios web, etc. Las relaciones representan cómo se conectan los nodos y los componentes, como las conexiones de red, las conexiones de base de datos, las conexiones de servicios web, etc.

Además, los diagramas de despliegue también pueden mostrar la distribución de componentes en diferentes nodos y cómo se balancea la carga en el sistema.

Los diagramas de despliegue son útiles para modelar la arquitectura física de un sistema de software, los componentes de hardware y software que lo componen, sus interacciones en un entorno de ejecución específico, y cómo se distribuyen los componentes en diferentes nodos y se balancea la carga en el sistema.

#### 5.4.3.8 Diagramas de Paquete.

Los diagramas de paquetes son un tipo de diagrama en UML que se utilizan para organizar y mostrar la estructura de un sistema de software en términos de paquetes lógicos y las dependencias entre ellos.

Un paquete en UML es una colección de elementos de modelado, como clases, interfaces, casos de uso, etc. Los diagramas de paquetes se utilizan para mostrar cómo se organizan los elementos de modelado en paquetes lógicos y cómo se relacionan entre sí.

Los diagramas de paquetes se utilizan a menudo en la fase de diseño de un proyecto de software para organizar y estructurar el sistema. Estos diagramas permiten al equipo de desarrollo visualizar la estructura del sistema y comprender cómo los diferentes elementos de modelado se relacionan entre sí.

Además, los diagramas de paquetes también se pueden utilizar para identificar los paquetes que deben ser reutilizados en otros proyectos, y para analizar el impacto de los cambios en los elementos de modelado en otros paquetes.

Los diagramas de paquetes son útiles para organizar y mostrar la estructura de un sistema de software en términos de paquetes lógicos y las dependencias entre ellos, lo que facilita la comprensión de la estructura y la gestión de la complejidad del sistema.

#### 5.4.3.9 Diagramas de Estructura compuesta.

Un Diagrama de Estructura Compuesta es un tipo de diagrama en UML que describe la estructura interna de una clase o componente, mostrando cómo sus partes y componentes están interconectados y trabajan juntos para lograr un comportamiento específico.

Este diagrama se utiliza para modelar sistemas complejos y su funcionamiento interno al dividirlos en sus partes constituyentes, como objetos, componentes y conexiones, y luego mostrar cómo interactúan y colaboran para crear el comportamiento del sistema.

Un diagrama de estructura compuesta consta de partes, puertos, conectores y estructuras. Las partes son los componentes que forman el sistema, como objetos u otros componentes. Los puertos se utilizan para representar los puntos donde las partes se comunican entre sí o con el mundo exterior. Los conectores representan las relaciones entre partes y puertos, como asociaciones,

dependencias o comunicaciones. Las estructuras definen la organización interna de una pieza, mostrando sus atributos y operaciones.

Los diagramas de estructura compuesta son útiles para diseñar y comprender sistemas complejos que tienen múltiples niveles de abstracción e interdependencia, como sistemas de software, sistemas electrónicos o incluso sistemas biológicos. Permiten a los desarrolladores visualizar la estructura interna de un sistema y sus interacciones, ayudándoles a identificar posibles problemas de diseño, mejorar el rendimiento y optimizar el desarrollo.

#### 5.4.3.10 Diagramas de Comunicación.

Un diagrama de comunicación es un tipo de diagrama de interacción en UML que describe cómo los objetos intercambian mensajes entre sí para llevar a cabo una tarea o un escenario específico.

En este tipo de diagrama, los objetos se representan como cajas con su nombre, mientras que las líneas entre ellos representan los mensajes que se intercambian para comunicarse. Estos mensajes pueden ser sincrónicos o asincrónicos, y pueden llevar parámetros o valores de retorno.

Los diagramas de comunicación se utilizan a menudo para modelar escenarios complejos que implican múltiples objetos que interactúan entre sí. Estos diagramas son especialmente útiles para identificar los flujos de comunicación y los roles que desempeñan los diferentes objetos en un sistema, lo que puede ayudar a detectar errores o ineficiencias en el diseño.

Los diagramas de comunicación se pueden utilizar en conjunto con otros diagramas UML, como los diagramas de casos de uso o los diagramas de secuencia, para proporcionar una visión más completa de cómo funciona un sistema en diferentes situaciones.

Los diagramas de comunicación son útiles para modelar cómo los objetos intercambian mensajes entre sí para llevar a cabo una tarea o un escenario específico, y pueden ayudar a identificar los flujos de comunicación y los roles que desempeñan los diferentes objetos en un sistema.

Cada tipo de diagrama se utiliza para modelar diferentes aspectos del sistema de software y puede ser útil en diferentes etapas del ciclo de vida del desarrollo de software.

UML es una herramienta valiosa para el diseño y la documentación de sistemas de software porque permite una comunicación y colaboración más efectivas, mejora la claridad y comprensión del sistema, identifica problemas temprano, puede generar código automáticamente y sirve como documentación del sistema.

### 5.5. Opciones seleccionadas.

Luego de definir cada uno de los conceptos anteriores para el desarrollo de este proyecto se han seleccionado hacer uso de las siguientes tecnologías:

- En la interfaz gráfica: HTML, Bootstrap, Hojas de estilos CSS.
- Para la gestión de la base de datos: SQL Server 2019.

## 6. Diseño Metodológico.

Para el desarrollo de este sistema se tomaron en consideración principalmente los siguientes aspectos:

- Recopilación de la información.
- Ciclo de vida de desarrollo del Software.
- Metodología SCRUM para desarrollo de aplicaciones web.

### 6.1 Recopilación de la información.

Debido al carácter especial y todas las posibles variables que están involucradas en este proceso se utilizan los siguientes métodos de recopilación de información:

#### 6.1.1 Entrevistas.

Se llevaron a cabo entrevistas tanto presenciales como vía web con las personas involucradas en el proceso de recopilación, consolidación y análisis de la

información para así poder establecer los parámetros necesarios a tomarse en cuenta para la creación de los reportes.

#### 6.1.2 Documento de Archivo.

Se efectuó un análisis a los formatos que actualmente son utilizados por los docentes en el proceso manual para definir las interfaces de usuarios y cada uno de los tipos de datos del sistema.

#### 6.1.3 Observaciones.

Se agendaron y realizaron visitas para la realización de observaciones en el campo y entender a fondo cada una de las etapas del actual proceso al mismo tiempo que se delimitan los módulos que no forman parte del sistema pero que pueden ser agregados a futuro.

#### 6.1.4 Focus groups online.

Una vez obtenida y analizada la información inicial se realizaron una serie de reuniones tanto presenciales como en línea con las personas involucradas para exponer lo recopilado y validar que toda la información está de acuerdo con sus necesidades.

### 6.2 Ciclo de vida de desarrollo del software.

El ciclo de vida de desarrollo de software proporciona un enfoque estructurado para el desarrollo de software que ayuda a garantizar que los proyectos de software se completen con éxito, a tiempo y dentro del presupuesto. También proporciona un marco para mejorar la comunicación, garantizar la calidad, gestionar el riesgo y facilitar el mantenimiento y las actualizaciones a lo largo del tiempo<sup>12</sup>. El ciclo de vida del software se utiliza para:

- Proporcionar estructura y organización.
- Garantizar la calidad.
- Administrar el riesgo.
- Mejorar la comunicación.

<sup>12</sup> Granda, S. (2021), Diferentes modelos de ciclo de vida de un software, INLOGIQ

- Garantizar el cumplimiento.
- Facilite el mantenimiento y las actualizaciones.

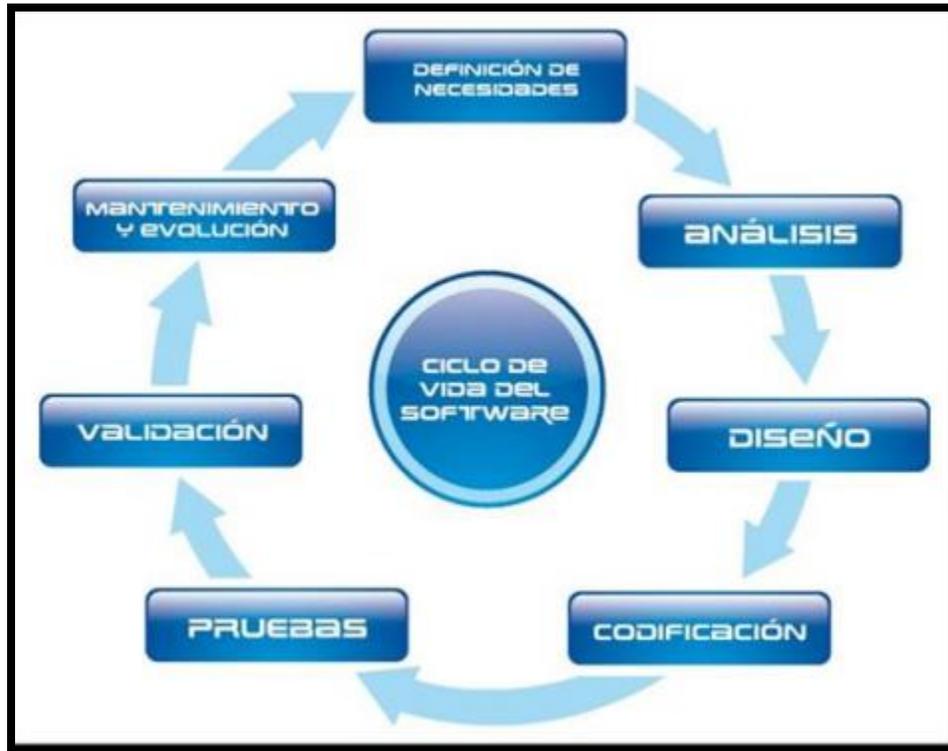


Ilustración 1 – Ciclo de vida del software

### 6.3 Metodología scrum.

El diseño metodológico utilizado en el desarrollo de este sistema informático es SCRUM debido a que ofrece<sup>13</sup>:

- Agilidad.
- Enfoque en el cliente.
- Transparencia.
- Colaboración.
- Mejora continua.

<sup>13</sup> Pursell, S. (20 enero, 2023), Metodología Agile: qué es y cómo aplicarla a tu proyecto

Tomando en consideración lo anteriormente mencionado, se propone utilizar sprints y seguir los pasos a continuación mencionado durante el desarrollo de este proyecto:

1. **Planificación del Sprint:** Durante la planificación del sprint, el equipo de desarrollo se reúne para definir los objetivos y prioridades del sprint. Se identifican las tareas necesarias para alcanzar estos objetivos y se asignan roles y responsabilidades a los miembros del equipo.
2. **Desarrollo del Sprint:** Durante el desarrollo del sprint, el equipo trabaja en las tareas asignadas y se comunica regularmente para asegurarse de que el proyecto está avanzando según lo previsto.
3. **Revisión del Sprint:** Al final del sprint, el equipo presenta el trabajo completado y se realiza una revisión para evaluar el progreso e identificar áreas de mejora.
4. **Retroalimentación y planificación:** Después de la revisión del sprint, se recopila retroalimentación y se utiliza para planificar el próximo sprint.
5. **Repetición:** Este proceso se repite hasta que se haya completado todo el trabajo necesario para desarrollar el sistema informático.

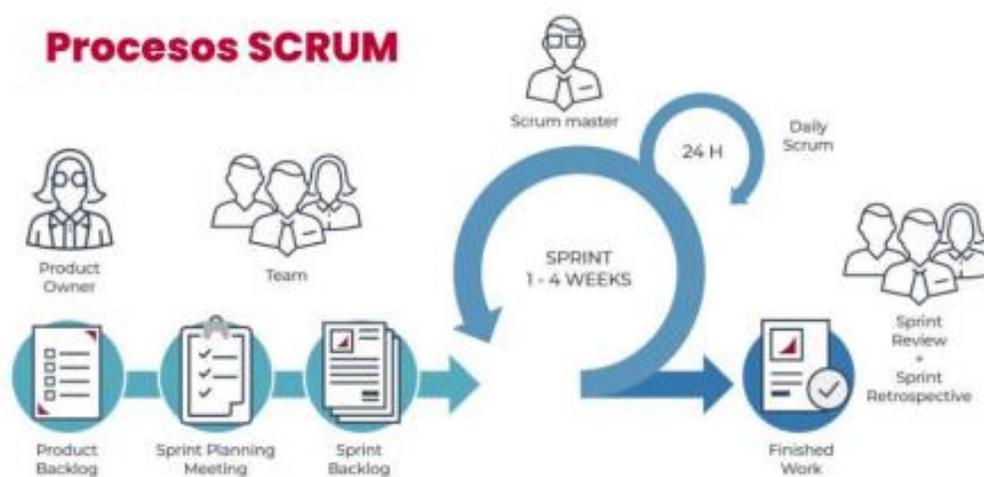


Ilustración 2 – Proceso SCRUM

## 7. Estimación de costos del Proyecto.

Fue utilizada con propósito de determinar si la propuesta de solución era viable, confiable y posible de realizar, por lo anterior se llevó a cabo un estudio de factibilidad que, abordando diferentes aspectos importantes del proceso desde una perspectiva operativa, técnica, financiera (económica) y legal ayudo a determinar la viabilidad del proyecto. Se tomaron en consideración la factibilidad técnica que se enfoca en las necesidades del usuario y todo lo que conlleva el proceso de desarrollo del software<sup>14</sup>; la factibilidad operativa, encargada de confirmar que se tengan las capacidades necesarias para lograr de manera eficiente la operación de los diferentes módulos y herramientas que permiten el acceso de las aplicaciones<sup>15</sup>; la factibilidad financiera, que se enfoca en los costos e insumos, que conlleva al desarrollo del software tanto desde el punto de vista del personal como del costo de los equipos<sup>16</sup>; por último, pero no menos importante, la factibilidad legal que ayuda a validar que el sistema cumpla con lo contemplado en el marco de las leyes, que establecen el desarrollo del software y los limitantes legales que de alguna manera pueden afectar el alcance del registro de la información<sup>17</sup> de cada uno de los médicos en su internado rotativo por medio de la aplicación.

Se efectuó la evaluación de las distintas tecnologías existentes para el desarrollo de este software. Se obtuvo la información de los distintos componentes que están en manos de la organización, los cuales son importantes para el desarrollo y correcto funcionamiento del software y se presentan 3 alternativas.

14 [TUY2007] Tuya, J., Ramos, I., Dolado, J. (2007). Técnicas cuantitativas para la gestión en la ingeniería del software. España: Netbiblo, S. L.

15, 16 INGENIERÍA DE SOFTWARE. Novena edición. Ian Sommerville. Traducción: Víctor Campos Olgún. Traductor especialista en Sistemas Computacionales

17 [SOM2005] Sommerville, I. (2005). Ingeniería del software. Madrid: Pearson Educación, S.A.

### 7.1. Alternativa número 1.

La primera opción propone la participación de 3 desarrolladores/analistas y la obtención del servicio de alojamiento de un servidor en la nube utilizando los servicios de Microsoft Azure. Esta opción permite obtener un equipo de cómputo virtual, para el alojamiento de la aplicación en la nube y la realización de las diferentes pruebas en el aplicativo a manos de los desarrolladores o analistas programadores, para cumplir con el aseguramiento de la calidad del software, el mantenimiento y desarrollo de este.

#### 7.1.1 Factibilidad técnica.

##### 7.1.1.1 Recurso Persona.

#	Recurso	Descripción Requerimiento	Cantidad Requerida
1	Desarrollador/Programador	Programador y Analista de sistema encargado del desarrollo del código fuente. De igual manera, encargado de mantenimiento de funcionalidades.	3

Tabla 1 – Tabla de requerimientos factor humano alternativa 1.

Fuente: Elaboración propia.

##### 7.1.1.2 Recurso Hardware.

#	Recurso	Descripción requerimiento	Cantidad Requerida
1	Máquina Virtual - Servidor de desarrollo	<ul style="list-style-type: none"><li>• <b>Máquina Virtual A1 v2 Servidor:</b><ul style="list-style-type: none"><li>o 2 núcleos.</li><li>o 4 GB RAM.</li><li>o 20 GB de almacenamiento temporal.</li><li>o SSD estándar 16 GB de almacenamiento</li></ul></li></ul>	1

Tabla 2 – Tabla de requerimientos hardware alternativa 2.

Fuente: Elaboración propia.

#	Recurso	• Descripción requerimiento	Cantidad Requerida
2	Unidad de Cómputo	<ul style="list-style-type: none"> <li>• Processor Intel Core I3-1115G4.</li> <li>• CPU 2.5 GHz.</li> <li>• 8 GB de Memoria RAM DDR4.</li> <li>• SSD 256 GB de almacenamiento.</li> </ul>	1

Tabla 2 – Tabla de requerimientos hardware alternativa 2 (Cont).

Fuente: Elaboración propia.

### 7.1.1.3 Recurso Software.

#	Recurso	Descripción requerimiento	Cantidad Requerida
1	Internet Information Service (IIS Server)	<ul style="list-style-type: none"> <li>• 2 GB de RAM.</li> <li>• 50 GB espacio en disco disponible.</li> </ul>	1
2	SQL Server - Gestor de Bases de Datos	<ul style="list-style-type: none"> <li>• Versión 2019.</li> <li>• 4 GB de espacio en disco disponible.</li> <li>• 2 GB de RAM</li> </ul>	1
3	Visual Studio Community Edition (ASP.NET y C#)	<ul style="list-style-type: none"> <li>• Versión 2022.</li> <li>• 2 GB de RAM.</li> <li>• 20 - 50 GB de espacio en disco disponible</li> </ul>	1

Tabla 3 –Tabla de requerimientos software alternativa 1

Fuente: Elaboración propia.

### 7.1.2 Factibilidad Económica.

En la siguiente tabla se muestra el valor económico de todos los recursos mencionados anteriormente y que comprende la alternativa número 1, para así poder poner en marcha esta alternativa propuesta.

#	Recurso	Valor Económico (Dólares)	
<i>Hardware</i>			
1	Máquina Virtual con licencia de SQL Server	\$37.10 (Costo Mensual)	\$222.06 (Costo semi anual)
2	Unidad de Cómputo	\$421.04 (Costo unico)	\$421.04 (Costo unico)
<i>Software</i>			
3	Internet Information Service (IIS Server)	\$0.0 (Incluido en el pago de Servidor virtual)	
4	SQL Server - Gestor de Bases de Datos	\$0.0 (Versión Desarrollador)	
5	Visual Studio Community Edition (ASP.NET y C#)	\$0.0 (Versión Desarrollador)	
<i>Persona</i>			
6	3 desarrolladores	\$750 por desarrollador (Mensual)	\$2,150 (Costo total de desarrollo – 2 meses)
Total		\$4,943.10	

Tabla 4 – Tabla factibilidad económica 1.

Fuente: Elaboración propia basado en los costos de máquinas virtuales Azure.

## 7.2 Alternativa número 2.

La segunda opción propone la participación de 3 desarrolladores/analistas, la adquisición o compra de un servidor dedicado, el cual se dedicará al alojamiento del sistema de manera local, así como, la adquisición de una computadora dedicada para la creación, prueba y ejecución del software por parte de los desarrolladores o analistas de sistema encargados del proyecto.

### 7.2.1 Factibilidad Técnica.

#### 7.2.1.1 Recurso Persona.

#	Recurso	Descripción Requerimiento	Cantidad Requerida
1	Desarrollador/Programador	Programador y Analista de sistema encargado del desarrollo del código fuente. De igual manera, encargado de mantenimiento de funcionalidades.	3

Tabla 5 – Tabla de requerimientos factor humano alternativa 2.

Fuente: Elaboración propia.

#### 7.2.1.2 Recurso Hardware.

#	Recurso	Descripción Requerimiento	Cantidad Requerida
1	Servidor Físico	<ul style="list-style-type: none"><li>• Servidor Dell POWEREDGE T40.</li><li>• Intel Xeon E-2224G.</li><li>• 8 GB de RAM.</li><li>• 1 TB de almacenamiento</li></ul>	1
2	Unidad de Cómputo	<ul style="list-style-type: none"><li>• Processor Intel Core I5-1135G7.</li><li>• PC Dell Optiplex.</li><li>• 8 GB de RAM.</li><li>• HDD 256 GB de almacenamiento.</li></ul>	1

Tabla 6 – Tabla de requerimientos hardware alternativa 2.

Fuente: Elaboración propia.

### 7.2.1.3 Recurso Software.

#	Recurso	Descripción Requerimiento	Cantidad Requerida
1	Internet Information Service (IIS Server)	<ul style="list-style-type: none"> <li>• 2 GB de RAM.</li> <li>• 50 GB espacio en disco disponible.</li> </ul>	1
2	SQL Server - Gestor de Bases de Datos	<ul style="list-style-type: none"> <li>• Versión 2019.</li> <li>• 4 GB de espacio en disco disponible.</li> <li>• 2 GB de RAM</li> </ul>	1
3	Visual Studio Community Edition (ASP.NET y C#)	<ul style="list-style-type: none"> <li>• Versión 2022.</li> <li>• 2 GB de RAM.</li> <li>• 20 - 50 GB de espacio en disco disponible</li> </ul>	1

Tabla 7 – Tabla de requerimientos software alternativa 2.

Fuente: Elaboración propia.

### 7.2.2 Factibilidad Económica.

En la siguiente tabla se muestra el valor económico de todos los recursos mencionados anteriormente y que comprende la alternativa número 2, para así poder poner en marcha esta alternativa propuesta.

#	Recurso	Valor Económico (Dólares)	
<i>Hardware</i>			
1	Máquina Virtual - Servidor de desarrollo	\$1,162.34 (Costo Mensual)	\$6,974.04 (Costo semi anual)
2	Unidad de Cómputo	\$785.00 (Costo unico)	\$785.00 (Costo unico)

Tabla 8 – Tabla de factibilidad económica 2.

Fuente: Elaboración propia.

#	Recurso	Valor Económico (Dólares)	
<i>Software</i>			
3	Internet Information Service (IIS Server)	\$0.0 (Incluido en el pago de Servidor virtual)	
4	SQL Server - Gestor de Bases de Datos	\$0.0 (Versión Desarrollador)	
5	Visual Studio Community Edition (ASP.NET y C#)	\$0.0 (Versión Desarrollador)	
<i>Persona</i>			
6	3 desarrolladores	\$750 por desarrollador (Mensual)	\$2,150 (Costo total de desarrollo – 2 meses)
Total		\$9,909.04	

Tabla 8 – Tabla de factibilidad económica 2 (cont).

Fuente: Elaboración propia.

### 7.3. Alternativa número 3.

Para esta tercera opción se reutilizan los servidores locales con los que cuenta la organización actualmente para en estos alojar el aplicativo de manera nativa, la adquisición de un equipo de trabajo donde se procederá al desarrollo interno del aplicativo, ejecución de la fase de implementación, ejecución de la fase de prueba y lanzamiento. Para asegurar el éxito de este proceso, se necesitará control completo de los desarrolladores para cada una de las etapas, por lo que se requiere la participación de 3 desarrolladores/analistas.

### 7.3.1. Factibilidad Técnica.

#### 7.3.1.3. Recurso Persona.

#	Recurso	Descripción	Cantidad Requerida
1	Desarrollador/Programador	Programador y Analista de sistema encargado del desarrollo del código fuente. De igual manera, encargado de mantenimiento de funcionalidades.	3

Tabla 9 – Tabla de requerimientos factor humano alternativa 3.

Fuente: Elaboración propia.

#### 7.3.1.2. Recurso Hardware.

#	Recurso	Descripción	Cantidad Requerida
1	Servidor Físico	<ul style="list-style-type: none"><li>• Procesador Intel Xeon E-2224G.</li><li>• 16 GB de RAM.</li><li>• SSD 500 GB de almacenamiento</li></ul>	1
2	Unidad de Cómputo	<ul style="list-style-type: none"><li>• Dell OptiPlex Processor Intel Core I5-1135G7.</li><li>• 8 GB de RAM.</li><li>• HDD 256 GB de almacenamiento.</li></ul>	1

Tabla 10 – Tabla de requerimientos hardware alternativa 3.

Fuente: Elaboración propia.

#### 7.3.1.3. Recurso Software.

#	Recurso	Descripción	Cantidad Requerida
1	Internet Information Service (IIS Server)	<ul style="list-style-type: none"><li>• 2 GB de RAM.</li><li>• 50 GB espacio en disco disponible.</li></ul>	1

Tabla 11 – Tabla de requerimientos software alternativa 3.

Fuente: Elaboración propia.

#	Recurso	Descripción	Cantidad Requerida
2	SQL Server - Gestor de Bases de Datos	<ul style="list-style-type: none"> <li>• Versión 2019.</li> <li>• 4 GB de espacio en disco disponible.</li> <li>• 2 GB de RAM.</li> </ul>	1
3	Visual Studio Community Edition (ASP.NET y C#)	<ul style="list-style-type: none"> <li>• Versión 2022.</li> <li>• 2 GB de RAM.</li> <li>• 20 - 50 GB de espacio en disco disponible</li> </ul>	1

Tabla 11 – Tabla de requerimientos software alternativa 3 (cont).

Fuente: Elaboración propia.

### 7.3.2. Factibilidad Económica.

En la siguiente tabla se muestra el valor económico de todos los recursos mencionados en anteriormente en la alternativa 3, de manera de poder implementar y poner en marcha esta alternativa propuesta.

#	Recurso	Valor Económico (Dólares)
<i>Hardware</i>		
1	Máquina Virtual - Servidor de desarrollo	\$0.0 (Existente en la organización)
2	Unidad de Cómputo	\$0.0 (Existente en la organización)
<i>Software</i>		
3	Internet Information Service (IIS Server)	\$0.0 (Existente en la organización)
4	SQL Server - Gestor de Bases de Datos	\$0.0 (Existente en la organización)
5	Visual Studio Community Edition (ASP.NET y C#)	\$0.0 (Existente en la organización)

Tabla 12 – Tabla de factibilidad económica alternativa 3.

Fuente: Elaboración propia basado en la infraestructura existente de la organización.

#	Recurso	Valor Económico (Dólares)	
<i>Persona</i>			
6	3 desarrolladores	\$750 por desarrollador (Mensual)	\$2,150 (Costo total de desarrollo – 2 meses)
Total		\$2,150.0	

Tabla 12 – Tabla de factibilidad económica alternativa 3.

Fuente: Elaboración propia basado en la infraestructura existente de la organización (cont).

## 8. Análisis y selección de alternativas

Una vez realizado el análisis de las tres alternativas para el desarrollo e implementación del sistema, en conjunto con la parte involucrada se validó que debido a requerimientos internos y regulaciones especiales de manejo de la información al tratarse de un ente gubernamental la mejor opción costo/beneficio para la organización, es la numero 3, cabe mencionar que el costo de desarrollo arriba mencionado se reduce a \$0 al tratarse este de un trabajo monográfico y por lo tanto una retribución a la sociedad por parte de los desarrolladores de la aplicación.

En la factibilidad técnica de todas las alternativas se dictamino que el hardware tanto de equipo servidor y unidad de cómputo tienen muchas similitudes en características, las 3 opciones son apropiadas en el desarrollo y soporte del sistema, sin embargo, al tratarse de una entidad gubernamental existen ciertos protocolos y aplicaciones específicas para protección de la información que ya están en los equipos existentes en la organización y que serán usados para desarrollar la aplicación.

En la factibilidad económica se deja detallado que todos los costos de requerimientos a nivel de servidor y unidad de trabajo del desarrollador son casi semejantes, pero al usar el equipo existente dentro de la organización este costo se reduce a cero, por lo que esta se confirma que la mejor opción y la seleccionada para este proyecto es la numero 3.

## 9. Factibilidad operativa.

El sistema auxiliar para control de información académica de estudiantes de medicina mientras se encuentra realizando el período de internado rotativo en los hospitales escuela a los que son asignados, está diseñado para ser un sistema web, al que se puede acceder desde cualquier ordenador o dispositivo móvil con acceso a internet, ya que se encuentra alojado en los servidores internos de la organización y bajo su propio dominio.

Al tratarse de una organización gubernamental, esta tiene su propio dominio, el cual es [minsa.gob.ni](https://controlmedicosinternos.minsa.gob.ni) y se acordó que para acceder al sistema se use la dirección <https://controlmedicosinternos.minsa.gob.ni> en cualquier navegador web. Al usar los recursos existentes de la organización el alojamiento y configuraciones de red necesarias para poder publicar el sistema al igual que los protocolos de seguridad de red para el acceso al mismo son establecidos por el equipo de seguridad interno de acuerdo con las regulaciones existentes.

En lo que se refiere a los accesos para los diferentes tipos de usuarios del sistema estos son restringidos por una capa de seguridad adicional que está directamente en la aplicación y que les permite tener solamente ciertos niveles de accesos dependiendo del tipo de usuario y los privilegios asignados, a continuación, se muestran dichos accesos y niveles de permisos:

- Invitado
- Miembro
- Administrador

Las funciones y roles de estos se definen en la siguiente tabla:

Usuario	Descripción	Permisos
Invitado	Estudiantes que verificaran sus notas en línea. Este usuario tiene mayormente permisos de lectura.	<ul style="list-style-type: none"> <li>• Modificar su información personal.</li> <li>• Ver notas del módulo actual.</li> <li>• Ver historial de notas.</li> <li>• Cambiar contraseña de acceso propia.</li> </ul>
Miembro	Los docentes que subirán notas al sistema. Este usuario tiene algunos permisos de modificación y de administración para ciertos catálogos.	<ul style="list-style-type: none"> <li>• Agregar nuevos usuarios invitados al sistema.</li> <li>• Modificar la información de los usuarios invitados.</li> <li>• Crear las notas de los módulos.</li> <li>• Modificar las notas de los módulos.</li> <li>• Eliminar las notas de los módulos.</li> <li>• Generar reporte de notas.</li> <li>• Generar reportes de estudiantes.</li> </ul>
Administrador	Las autoridades del ministerio de salud. Este usuario tiene permisos de control total y administración sobre todo el sistema.	<ul style="list-style-type: none"> <li>• Control total.</li> <li>• Crear usuarios Invitados.</li> <li>• Modificar información de usuarios invitados.</li> <li>• Eliminar usuarios invitados.</li> <li>• Crear usuarios Miembros.</li> <li>• Actualizar información de miembros.</li> </ul>

Tabla 13 – Tabla de factibilidad Operativa.

Fuente: Elaboración propia.

Usuario	Descripción	Permisos
Administrador	Las autoridades del ministerio de salud. Este usuario tiene permisos de control total y administración sobre todo el sistema.	<ul style="list-style-type: none"> <li>• Eliminar usuarios miembros.</li> <li>• Asignar usuarios invitados a grupos de clases.</li> <li>• Crear grupos de clases.</li> <li>• Actualizar grupos de clases.</li> <li>• Eliminar grupos de clases.</li> <li>• Actualizar módulos de clases.</li> <li>• Eliminar módulos de clases.</li> <li>• Generar reportes.</li> <li>• Cambiar nivel de permisos de usuarios miembros e invitados.</li> </ul>

Tabla 13 – Tabla de factibilidad Operativa.

Fuente: Elaboración propia (cont).

Cabe mencionar que el sistema admite la creación de nuevos roles de usuarios que pueden ser definidos y agregados posteriormente a la puesta en funcionamiento de este, estos otros usuarios pueden ser definidos con diferentes tipos de accesos en caso de ser necesario en el futuro según sean las necesidades de la organización, a corto, mediano y a largo plazo permitiendo que el sistema sea operativamente factible.

## **10. Factibilidad legal**

Las herramientas de desarrollo y los sistemas operativos seleccionados en la realización de este proyecto, así como los motores de bases de datos y herramientas de creación de las interfaces graficas que se utilizaron para el diseño y desarrollo del aplicativo poseen licencias de uso emitidas por el fabricante de cada uno de estos y cumplen con las normas estipuladas en la ley N° 787 de la Republica de Nicaragua, Ley de Protección de Datos Personales (Anexo 5).

Al tratarse de una entidad gubernamental, además de la Ley N° 787 existen normas específicas e internacionales tales como la GDPR (Reglamento General de Protección de Datos) asegurando el consentimiento explicito, derechos de los usuarios, protección de datos y responsabilidad de institución para su resguardo.

Debido a que la organización cuenta con servidores internos tanto de Bases de Datos como servidores IIS y estos han sido seleccionados para el alojamiento de la aplicación, la licencia para el gestor de bases de datos SQL Server 2019 ya están previamente adquiridas en la organización, lo que nos exime de todas las obligaciones contractuales y legales durante diseño y desarrollo de la aplicación.

Teniendo, en conclusión, no se corren dificultades con penalidades legales o retrasos ejecutivos que pudieran afectar de manera directa al diseño y desarrollo del software en su totalidad.

## 11. Analisis de Riesgos del sistema.

Para la gestión de riesgos del proyecto se utiliza la siguiente metodología con los procesos de identificación, análisis cualitativo, plan de respuesta y control y seguimiento de los riesgos<sup>18</sup>.



Ilustración 3 – Gestión de Riesgos.

Fuente: Elaboración propia.

Lo primero que se realiza es definir los tipos de riesgos a tomar en consideración, estos se dividen en las categorías que se mencionan a continuación<sup>19</sup>.

Tipo de Riesgo	Descripción
<b>Tecnológico</b>	Desconocimiento de las tecnologías base del proyecto, necesidad tecnológica inmadura. Amenazan la calidad del software y la planificación temporal del proyecto. La implementación puede llegar a ser difícil o imposible
<b>Persona</b>	Errores humanos que pueden ser ortográficos, de compilación, de ejecución, Falta de experiencia del líder del equipo, rotación de personal, perfiles inadecuados en el equipo, incumplimiento de fecha límite.

Tabla 14 – Tipos de Riesgos.

Fuente: Elaboración propia.

<sup>18</sup> Webcindario.com. (2021). Riesgo del Software, Webcindario.com

<sup>19</sup> Retos en Supply Chain. (2020, mayo 14). Guía PMBOK: definición, estructura y tips de estudio. Retos en Supply Chain - EAE Business School Barcelona

Tipo de Riesgo	Descripción
<b>Organizacional</b>	Asociados a las restricciones o limitaciones impuestas por la organización que pueden afectar al producto, conflictos internos entre los departamentos involucrados, cambios en el entorno organizacional.
<b>Herramientas</b>	Asociados a problemas con las herramientas necesarias a utilizar para poder construir un producto final de calidad.
<b>Requerimientos</b>	Errores en el diseño, mala planificación, de programación, falta de pruebas al sistema, bajo o mala comunicación con el cliente final.
<b>Estimación</b>	Asociados a riesgos con el tamaño del software a construir o desarrollar y las estimaciones del alcance de este. Integraciones con sistemas externos desconocidos, número de personas que usarán el sistema.

Tabla 14 – Tipos de Riesgos.

Fuente: Elaboración propia.

Una vez definidas las categorías de los riesgos, se procede al siguiente paso que consiste en crea una tabla en la cual se definen las probabilidades de que un riesgo suceda, así como el impacto que tendría el riesgo si llegara a suceder en el proyecto<sup>20</sup>.

Nivel	Probabilidad de riesgo	Descripción
<b>1</b>	Raro	El evento tiene muy pocas probabilidades de ocurrir.
<b>2</b>	Improbable	El evento puede ocurrir en determinado momento y si se cumplen ciertas condiciones

Tabla 15 – Niveles de probabilidad de riesgo.

Fuente: Elaboración propia.

<sup>20</sup> Retos en Supply Chain. (2020, mayo 14). Guía PMBOK: definición, estructura y tips de estudio. Retos en Supply Chain - EAE Business School Barcelona

Nivel	Probabilidad de riesgo	Descripción
3	Moderado	El evento puede ocurrir en determinado momento en cualquier tipo de condiciones.
4	Probable	El evento puede ocurrir independientemente de las condiciones.
5	Casi seguro	El evento ocurrirá independientemente de las condiciones.

Tabla 15 – Niveles de probabilidad de riesgo (cont.).

Fuente: Elaboración propia.

Nivel	Factor de riesgo	Descripción
1	Insignificante	Si llegara a suceder el impacto serían mínimos o insignificantes.
2	Menor	Si llegara a suceder el impacto sería bajo.
3	Moderado	Si llegara a suceder el impacto sería medio.
4	Mayor	Si llegara a suceder el impacto sería alto.
5	Catastrófico	Si llegara a suceder el impacto tendría consecuencias desastrosas.

Tabla 16 – Niveles de Impacto de riesgo.

Fuente: Elaboración propia.

Para calificar cada uno de los riesgos identificados se realizó la matriz de probabilidad e impacto, multiplicando el valor numérico del nivel de probabilidad de ocurrencia del riesgo por el nivel numérico del impacto del riesgo, este resultado es el que determina el nivel del riesgo.

La fórmula para utilizar es la siguiente:

$$ER= P \times I$$

Donde:

**P** es la probabilidad de que un riesgo ocurra.

**I** es el impacto que tendría el riesgo en el proyecto.

Para el desarrollo de este proyecto se evaluaron los efectos de los riesgos anteriormente clasificados en los cuatro ejes que se describen a continuación<sup>21</sup>:

Tipo de Riesgo	Descripción
<b>Alcance</b>	Afectación en el alcance y los objetivos del proyecto si sucediera este evento.
<b>Tiempo</b>	Nivel de afectación en el tiempo de desarrollo del proyecto al suceder este evento.
<b>Costo</b>	Costo adicional que puede significar para el proyecto si el evento sucede.
<b>Calidad</b>	Afectación a la calidad en el producto final que significaría si este evento sucede.

Tabla 17 – Objetivo afectado por riesgo.

Fuente: Elaboración propia.

Con todo lo anteriormente definido se procede a realizar la matriz de estimación de probabilidad de riesgo que se muestra a continuación<sup>22</sup>:

Probabilidad	Impacto				
	Insignificante - 1	Menor - 2	Moderado - 3	Mayor - 4	Catastrófico - 5
Raro - 1					
Improbable - 2					

Tabla 18 – Diseño de Matriz de riesgo (Probabilidad e impacto).

Fuente: Elaboración propia.

<sup>21, 22</sup> Anyconnector.com (marzo 12, 2021). ¿Qué es el riesgo tecnológico? Como administrar las amenazas empresariales. Anyconnector.com.

Probabilidad	Impacto				
	Insignificante - 1	Menor - 2	Moderado - 3	Mayor - 4	Catastrófico - 5
Moderado - 3					
Probable - 4					
Casi seguro - 5					

Tabla 18 – Diseño de Matriz de riesgo (Probabilidad e impacto) (Cont.).

Fuente: Elaboración propia.

Una vez se tiene la matriz de probabilidad e impacto, se propuso una escala numérica la cual permitió realizar la clasificación del riesgo según su nivel, estos pueden ser muy bajo, bajo, medio, alto, y muy alto<sup>23</sup>.

Nivel de riesgo	Probabilidad x Impacto
Muy alto	> 80
Alto	51 - 80
Medio	31 - 50
Bajo	11 - 30
Muy Bajo	< 10

Tabla 19 – Escala de Riesgo.

Fuente: Elaboración propia.

Durante cada una de las fases de desarrollo de software se realizó un análisis detallado de cómo funcionaría éstas para luego proceder a listar los riesgos más significativos y que en la realidad se pudieron presentar, teniendo en cuenta diversos factores tales como la lógica del negocio, el compromiso por parte del cliente, los requerimientos, la calidad al realizar el diseño, codificación, pruebas, conocimiento y experiencia por parte del personal, manejo de herramientas y metodologías entre otros, lo cual dio pauta para proponer el listado de riesgos para cada fase y posteriormente se aplicó paso a paso la respectiva gestión de riesgos anteriormente descrita<sup>24</sup>, a continuación la matriz de riesgos obtenida:

<sup>23</sup> Torres, O. P. (s/f). Gestión de riesgos en proyectos de software. Piranirisk.com.

<sup>24</sup> (que es la guía PMBOK y cómo influye en la administración de proyectos)

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
Fase de análisis del proyecto							
R001	Requerimientos incompletos o ambiguos.	Los requerimientos no se definieron de manera clara y completa durante las reuniones iniciales.	5	Alcance	4	20	<b>90</b>
				Tiempo	5	25	
				Costo	5	25	
				Calidad	4	20	
				Riesgo total		90	
R002	Retrasos en las especificaciones de requerimientos.	Las reuniones con el cliente para el levantamiento de requerimientos se cancelan continuamente.	3	Alcance	3	9	<b>42</b>
				Tiempo	4	12	
				Costo	4	12	
				Calidad	3	9	
				Riesgo total		42	
R003	Incorporación continua de nuevos requerimientos.	El cliente no tiene claridad de lo que desea, surgen necesidades nuevas.	3	Alcance	4	12	<b>45</b>
				Tiempo	4	12	
				Costo	4	12	
				Calidad	3	9	
				Riesgo total		45	
R004	Modificación continua de requerimientos.	Actualizaciones necesarias debido a una deficiente definición de requerimientos inicialmente.	4	Alcance	3	12	<b>48</b>
				Tiempo	3	12	
				Costo	3	12	
				Calidad	3	12	
				Riesgo total		48	

Tabla 20 – Matriz de Riesgo.

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R005	Modificaciones incorrectas de las especificaciones.	Análisis incorrecto de los requerimientos.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	
R006	Entendimiento errado de los requerimientos.	Las necesidades expuestas por el cliente son entendidas y documentadas de manera equivocada.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	
<b>Fase de diseño del proyecto</b>							
R007	Incorrecta definición y estructuración de los datos establecidos.	Pobre definición de tipos de datos e integridad, y poco entendimiento sobre la relación o dependencia de estos.	3	Alcance	3	9	36
				Tiempo	3	9	
				Costo	3	9	
				Calidad	3	9	
				Riesgo total		36	
R008	Diseño de interfaces incompleto.	Desconocimiento de todas las interfases que pueden afectar la solución.	3	Alcance	4	12	45
				Tiempo	4	12	
				Costo	4	12	
				Calidad	3	9	
				Riesgo total		45	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R009	Subestimación del tamaño de la aplicación.	Al realizar el diseño se subestima el software con respecto a las necesidades del cliente.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	5	15	
				Calidad	3	9	
				Riesgo total		48	
R010	Falta de Especificación de la arquitectura lógica.	No se Define adecuadamente las interconexiones y recursos lógicos entre módulos del sistema de manera apropiada para su diseño detallado y administración.	5	Alcance	4	20	90
				Tiempo	5	25	
				Costo	5	25	
				Calidad	4	20	
				Riesgo total		90	
R011	Falta de Especificación de la arquitectura física.	No se define correctamente el conjunto de dispositivos físicos que se va a utilizar para que la arquitectura lógica funcione correctamente.	3	Alcance	4	12	51
				Tiempo	4	12	
				Costo	5	15	
				Calidad	4	12	
				Riesgo total		51	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R012	Desconocimiento de la lógica de negocio.	Mala interpretación o interpretación superficial de los requisitos para hacer el diseño detallado del sistema.	3	Alcance	5	15	54
				Tiempo	4	12	
				Costo	5	15	
				Calidad	4	12	
				Riesgo total		54	
Fase de codificación del proyecto							
R013	Bajo rendimiento de la herramienta CASE.	Las herramientas CASE que se utilizan como apoyo no tienen el rendimiento y las funcionalidades esperadas.	3	Alcance	3	9	42
				Tiempo	4	12	
				Costo	3	9	
				Calidad	4	12	
				Riesgo total		42	
R014	Manejo inadecuado en liberación de versiones.	Despliegue incompleto de versión de la aplicación, despliegue con direccionamiento equivocada a bases de datos.	3	Alcance	3	9	42
				Tiempo	4	12	
				Costo	4	12	
				Calidad	3	9	
				Riesgo total		42	
R015	Falta de documentación en código fuente.	limitación del tiempo. Aplicación de malas prácticas de desarrollo y ausencia de revisiones.	3	Alcance	2	6	30
				Tiempo	3	9	
				Costo	2	6	
				Calidad	3	9	
				Riesgo total		30	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R016	Modificación cronograma actividades.	Actividades no contempladas, adición de nuevas actividades. Complejidad del desarrollo de actividades no estimadas. Retrasos en la ejecución de actividades por imprevistos.	4	Alcance	3	12	56
				Tiempo	4	16	
				Costo	4	16	
				Calidad	3	12	
				Riesgo total		56	
R017	No disponibilidad de hardware o software.	El hardware o software esencial no es entregado a tiempo.	4	Alcance	5	20	72
				Tiempo	5	20	
				Costo	4	16	
				Calidad	4	16	
				Riesgo total		72	
R018	El Software es complejo de implementar.	El desarrollo de la aplicación tiene un nivel alto de complejidad. El modelado realizado en la fase de diseño no fue tan claro ni específico.	4	Alcance	4	16	64
				Tiempo	4	16	
				Costo	4	16	
				Calidad	4	16	
				Riesgo total		64	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R019	Compleja la integración de módulos del software.	Al codificar y comenzar la integración se hace evidente que la especificación está incompleta, contiene requisitos contradictorios o hay fallas en el diseño del software.	5	Alcance	4	20	<b>90</b>
				Tiempo	5	25	
				Costo	5	25	
				Calidad	4	20	
				Riesgo total		90	
R020	Retiro de personal con conocimiento y experiencia.	Al ser las únicas personas que maneja ciertos temas específicos y complejos al irse generan retraso en el curso de las tareas.	4	Alcance	4	16	<b>64</b>
				Tiempo	5	20	
				Costo	3	12	
				Calidad	4	16	
				Riesgo total		64	
R021	No hay buena comunicación o sinergia en el equipo.	La comunicación entre el personal de desarrollo no es el más óptimo y así mismo su sinergia no es la más eficaz para el cumplimiento de objetivos en común.	3	Alcance	4	12	<b>48</b>
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R022	Falta de conocimiento y experiencia sobre las tareas asignadas y las herramientas a utilizar.	El personal no es idóneo o no tiene la experiencia necesaria para el rol asignado.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	
R023	Pérdida de respaldos.	Perdida de la copia de seguridad de la versión de software actual causado por virus o por remplazo de versión sin sacar la copia previamente.	3	Alcance	4	12	48
				Tiempo	5	15	
				Costo	4	12	
				Calidad	3	9	
				Riesgo total		48	
<b>Fase de Pruebas del proyecto</b>							
R024	Alcance de las pruebas no definido completamente.	No se definió desde el inicio de la fase el alcance debido a que no se tenía documentación y la que existía era muy superficial o estaba desactualizada.	5	Alcance	4	20	95
				Tiempo	5	25	
				Costo	5	25	
				Calidad	5	25	
				Riesgo total		95	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R025	Documentación de requisitos insuficiente, desactualizada, contradictoria o ambigua.	Los casos de prueba no quedan cubiertos en su totalidad, debido a que pueden existir cambios o mejoras que no se encuentran actualizados a la fecha.	5	Alcance	5	25	<b>100</b>
				Tiempo	5	25	
				Costo	5	25	
				Calidad	5	25	
				Riesgo total		100	
R026	Realizar pruebas en ambiente desarrollo.	Alta Inestabilidad del ambiente, funcionalidades probadas, pero no certificadas al 100%.	4	Alcance	4	16	<b>72</b>
				Tiempo	4	16	
				Costo	5	20	
				Calidad	5	20	
				Riesgo total		72	
R027	No se realizan las pruebas completamente.	El conjunto de pruebas realizadas no son lo suficientes para garantizar la calidad del software esto sucede por omisión o por falta de tiempo.	4	Alcance	4	16	<b>64</b>
				Tiempo	4	16	
				Costo	4	16	
				Calidad	4	16	
				Riesgo total		64	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R028	No se realiza priorización en la ejecución de las pruebas.	No se le da prioridad para probar las funcionalidades más importantes y complejas del software, al final se descubre defectos bloqueantes los cuales necesitan tiempo para ser solucionados.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	
R029	Demoras excesivas en la reparación de defectos encontrados en las pruebas.	solución de defectos no priorizada por parte de los desarrolladores lo cual retrasa las pruebas.	3	Alcance	3	9	42
				Tiempo	4	12	
				Costo	4	12	
				Calidad	3	9	
				Riesgo total		42	
R030	Problemas de disponibilidad con el ambiente de pruebas.	Problemas con la disponibilidad, preparación y estabilización del ambiente donde se ejecutan las pruebas, afectando cronogramas y retrasando el inicio de cada ciclo.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R031	Retraso Testing debido a nuevos errores después de despliegues.	Doble trabajo causado por despliegues que dañan funcionalidades ya existentes, generando así nuevos defectos.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	
R032	Pobre Productividad.	Tiempos muertos en sub-fases iniciales de la fase de pruebas que no se pueden recuperar por entregas tardías de desarrollo.	3	Alcance	2	6	27
				Tiempo	3	9	
				Costo	2	6	
				Calidad	2	6	
				Riesgo total		27	
R033	No hay suficientes recursos o ingresan demasiado tarde.	Ingreso tardío del personal a los roles necesitados. Personal reducido donde se necesitan más de los asignados.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	
<b>Fase de entrega del proyecto</b>							
R034	capacitación superficial a usuarios finales.	Por limitación o subestimación del tiempo se realiza una capacitación incompleta sobre el uso de la aplicación.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R035	La aplicación no procesa transacciones por segundo como se esperaba.	La capacidad a nivel de hardware no es acorde al número de transacciones solicitadas. La codificación del procedimiento de la transacción es poco eficiente en el tiempo de respuesta.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	
R036	Fallas del hardware limitan la funcionalidad del software.	Inestabilidad de la red o Internet, caída o afectación por virus de los servidores, falla eléctrica.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	
R037	Arquitectura inadecuada por parte del cliente.	especificación superficial de los requisitos básicos para la arquitectura hardware y software. modificación de la arquitectura con respecto a la definida inicialmente.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R038	Documentación sobre el uso de la aplicación.	Generación pobre de documentos necesarios para la instalación y uso efectivo de la aplicación.	3	Alcance	2	6	30
				Tiempo	3	9	
				Costo	2	6	
				Calidad	3	9	
				Riesgo total		30	
R039	Vulnerabilidades del software presentadas en producción.	Omisión de validaciones en la fase de pruebas. Ambiente producción como es un ambiente real se pueden presentar defectos que no se presentaron en el ambiente de pruebas.	3	Alcance	3	9	48
				Tiempo	3	9	
				Costo	5	15	
				Calidad	5	15	
				Riesgo total		48	
R040	Resistencia del personal para cambiar las prácticas del pasado.	El personal que va a utilizar el nuevo software presenta miedo al cambio debido a la costumbre de utilizar el anterior software.	3	Alcance	2	6	27
				Tiempo	3	9	
				Costo	2	6	
				Calidad	2	6	
				Riesgo total		27	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Código de Riesgo	Descripción del Riesgo	Causa Raíz	estimación Probabilidad	Objetivo afectado	estimación impacto	Probabilidad x Impacto	Nivel de Riesgo
R041	Software contiene numerosos errores cuando se entrega al cliente.	El cliente por cumplir con fechas de salida a producción toma el riesgo de salir con defectos existentes en la aplicación que aún no se han solucionado por parte del equipo de desarrollo.	3	Alcance	3	9	48
				Tiempo	3	9	
				Costo	5	15	
				Calidad	5	15	
				Riesgo total		48	
R042	presentación de defectos en ambiente producción.	Hallazgo de defectos que no se detectaron previamente o que no se presentaron en el ambiente de pruebas	5	Alcance	4	20	90
				Tiempo	4	20	
				Costo	5	25	
				Calidad	5	25	
				Riesgo total		90	
R043	Fallas del software que limitan la funcionalidad de la aplicación.	Base de datos fuera de línea, servidor de aplicaciones fuera de línea, falla en el sistema operativo.	3	Alcance	4	12	48
				Tiempo	4	12	
				Costo	4	12	
				Calidad	4	12	
				Riesgo total		48	

Tabla 20 – Matriz de Riesgo (Cont.).

Fuente: Elaboración propia.

Una vez se ha definido la matriz de riesgos procedemos al siguiente paso que es el plan de prevención y mitigación de los riesgos identificados, presentando las acciones en caso de que el evento descrito ocurra a como se detalla en la siguiente tabla:

<b>Código de Riesgo</b>	<b>descripción del Riesgo</b>	<b>Nivel de Riesgo</b>	<b>Medida de prevención</b>	<b>Plan de mitigación</b>
R001	Requerimientos incompletos o ambiguos.	Muy Alto	Asegurarse que el usuario tiene claro lo que desea que el sistema haga y tener listas preguntas sobre temas poco claros.	Incorporar los requerimientos o cambio de forma clara y completa para que se cumpla con la función que se está solicitando.
R002	Retrasos en las especificaciones de requerimientos.	Medio	Compromiso y responsabilidad de parte de los usuarios involucrados para proporcionar información necesaria de requerimientos.	Tener mecanismos para reunirse virtualmente, listas de preguntas que se pueden enviar por correo, realizar llamadas telefónicas.
R003	Incorporación continua de nuevos requerimientos.	Medio	Hacer un levantamiento completo de los requisitos del sistema y validarlo con el cliente antes de iniciar la etapa de codificación.	Agregar horas adicionales para reuniones con el cliente y lograr definir todos los requerimientos del sistema.
R004	Modificación continua de requerimientos.	Medio	Reuniones periódicas, listado de temas poco claros, dar al cliente opciones para sus requerimientos.	Definir alcance del sistema y obtener aprobación del cliente para trabajar en base a esos requerimientos.
R005	Modificaciones incorrectas de las especificaciones.	Medio	Garantizar que las especificaciones son las correctas antes de realizar modificaciones a la aplicación.	Contar con una versión anterior a la que se pueda regresar en caso de que las últimas modificaciones realizadas dañen la aplicación

Tabla 21 – Matriz de Prevención y Mitigación de Riesgo.

Fuente: Elaboración propia.

Código de Riesgo	descripción del Riesgo	Nivel de Riesgo	Medida de prevención	Plan de mitigación
R006	Entendimiento errado de los requerimientos.	Medio	Documentar la información y validar con el cliente que los cambios solicitados fueron entendidos de manera correcta.	Asignar personal con experiencia para realizar nuevamente el levantamiento de los requisitos.
R007	Incorrecta definición y estructuración de los datos establecidos.	Medio	Utilizar herramientas de software que aseguren el modelado de datos correcto.	Tener un plan de contingencia y un software de apoyo para realizar los diagramas necesarios en el sistema.
R008	Diseño de interfaces incompleto.	Medio	Seguir el cronograma de actividades en cada una de sus etapas y asegurar el cumplimiento de estas.	Contar con personal adicional para apoyar en el diseño de las interfaces.
R009	subestimación del tamaño de la aplicación.	Medio	Definir correctamente cual es el alcance de la aplicación, cuales con los requerimientos y objetivos del sistema.	Entender los requerimientos del cliente, comunicarse con el equipo de desarrollo y hacer una estimación realista del tiempo y recursos necesarios para la aplicación.
R010	Falta de Especificación de la arquitectura lógica.	Muy Alto	Definir la arquitectura lógica correcta y más eficiente utilizando modelos, vistas y diagramas con base a las especificaciones de los requerimientos.	Realizar el diseño de la arquitectura lógica teniendo en cuenta la arquitectura que posee el cliente.
R011	Falta de Especificación de la arquitectura física.	Alto	Definir la arquitectura física correcta y más eficiente con base a las especificaciones de los requerimientos.	Realizar el diseño de la arquitectura física teniendo en cuenta los equipos que posee el cliente.

Tabla 21 – Matriz de Prevención y Mitigación de Riesgo (Cont.).

Fuente: Elaboración propia.

<b>Código de Riesgo</b>	<b>descripción del Riesgo</b>	<b>Nivel de Riesgo</b>	<b>Medida de prevención</b>	<b>Plan de mitigación</b>
R012	Desconocimiento de la lógica de negocio.	Alto	Reuniones periódicas con el cliente, participación de usuarios finales, facilitación de documentos sobre procesos.	Reuniones para aclaración de dudas de temas puntuales, asignar personal proactivo y con experiencia.
R013	Bajo rendimiento de la herramienta CASE.	Medio	Investigar las herramientas disponibles y seleccionar la que sea más factible basándose en los requerimientos.	Tener una opción secundaria para utilizar en caso de que la herramienta solicitada no cumpla con los requisitos necesarios.
R014	Manejo inadecuado en liberación de versiones.	Medio	Liberar las versiones una vez que se completen las pruebas y validaciones para asegurar su correcto funcionamiento.	Utilizar mecanismos que permitan regresar a una versión anterior de la aplicación o a la última versión estable.
R015	Falta de documentación en código fuente.	Bajo	Documentar el código fuente mientras se va generando, utilizar programación fácil de entender.	Designar a una persona para realizar la documentación del código fuente junto con el desarrollador.
R016	Modificación cronograma actividades.	Alto	Incluir en los cronogramas tiempos racionales, dejando espacio para imprevistos, tener claro el alcance de la aplicación.	Recibir apoyo de parte de todos los miembros del equipo para realizar modificaciones en la asignación de recursos, tareas y optimizar los tiempos

Tabla 21 – Matriz de Prevención y Mitigación de Riesgo (Cont.).

Fuente: Elaboración propia.

<b>Código de Riesgo</b>	<b>descripción del Riesgo</b>	<b>Nivel de Riesgo</b>	<b>Medida de prevención</b>	<b>Plan de mitigación</b>
R017	No disponibilidad de hardware o software.	Alto	Definir y solicitar los equipos físicos como herramientas software necesarios para la codificación de la aplicación.	Tener un plan alternativo en caso de que no se tenga disponible un recurso de hardware o software.
R018	El Software es complejo de implementar.	Alto	Utilizar un modelo de desarrollo de software de acuerdo con el tamaño de la aplicación, tiempos, documentación, etc.	Realizar reutilización de software, realizar pruebas unitarias, apoyo de parte de un experto en este tipo de desarrollos.
R019	Compleja la integración de módulos del software.	Muy Alto	Utilizar independencia modular, crear módulos relativamente pequeños para minimizar el impacto al hacer un cambio.	Realizar programación en equipo, desarrollar módulos paralelamente y realizar pruebas de componentes.
R020	Retiro de personal con conocimiento y experiencia.	Alto	Contratar personal con experiencia y ofrecer las condiciones necesarias para la realización de sus tareas.	Tener preparado personal de respaldo que este calificado para todas las áreas involucradas en el proyecto.
R021	No hay buena comunicación o sinergia en el equipo.	Medio	Utilizar un repositorio compartido, tener llamadas y reuniones periódicas para discutir avances y bloqueos.	Sostener reuniones para que cada equipo exprese sus bloqueos y proponer soluciones a estos.
R022	Falta de conocimiento y experiencia sobre las tareas asignadas y las herramientas a utilizar.	Medio	Asignar personal con conocimientos y experiencia tanto en proyectos como en el uso de las herramientas.	Brindar capacitaciones al personal sobre el uso de las herramientas y la gestión del proyecto.

Tabla 21 – Matriz de Prevención y Mitigación de Riesgo (Cont.).

Fuente: Elaboración propia.

<b>Código de Riesgo</b>	<b>descripción del Riesgo</b>	<b>Nivel de Riesgo</b>	<b>Medida de prevención</b>	<b>Plan de mitigación</b>
R023	Pérdida de respaldos.	Medio	Tener mecanismos de respaldo adecuados y realizar respaldos periódicos.	Tener mecanismos de respaldo secundarios a manera de contingencias.
R024	Alcance de las pruebas no definido completamente.	Muy Alto	Realizar reuniones para definir objetivos, pruebas, validaciones necesarias y realizar un cronograma de ejecución de pruebas.	reunión con desarrolladores para aclarar dudas, contextualizar y obtener su apoyo en el seguimiento del cronograma de pruebas.
R025	Documentación de requisitos insuficiente, desactualizada, contradictoria o ambigua.	Muy Alto	Documentar, informar y explicar todos los cambios que se hagan al producto en su ciclo de desarrollo.	Designar a una persona para realizar las actualizaciones en la documentación de acuerdo con la última versión de la aplicación.
R026	Realizar pruebas en ambiente desarrollo.	Alto	Tener preparado un ambiente con características similares al ambiente de producción, pero aislado.	Crear un ambiente con las condiciones necesarias para llevar a cabo las pruebas del software.
R027	No se realizan las pruebas completamente.	Alto	Realizar una óptima planeación y ejecución de actividades de prueba, asignar analistas de calidad con experiencia en el tema.	Utilizar técnicas de prueba y buenas prácticas para cubrir todas las pruebas necesarias.
R028	No se realiza priorización en la ejecución de las pruebas.	Medio	Contar con el apoyo de un especialista de control de calidad para priorizar las pruebas a realizarse.	Contrata a un especialista de control de calidad para que apoye en la realización de las pruebas.

Tabla 21 – Matriz de Prevención y Mitigación de Riesgo (Cont.).

Fuente: Elaboración propia.

<b>Código de Riesgo</b>	<b>descripción del Riesgo</b>	<b>Nivel de Riesgo</b>	<b>Medida de prevención</b>	<b>Plan de mitigación</b>
R029	Demoras excesivas en la reparación de defectos encontrados en las pruebas.	Medio	Corregir los errores encontrados durante la etapa de pruebas rápidamente.	Contar con personal adicional para apoyar en las correcciones del software.
R030	Problemas de disponibilidad con el ambiente de pruebas.	Medio	Tener preparado un ambiente con características similares al ambiente de producción, pero aislado.	Crear un ambiente con las condiciones necesarias para llevar a cabo las pruebas del software.
R031	Retraso Testing debido a nuevos errores después de despliegues.	Medio	Asegurarse de todos los errores detectados y documentados han sido modificados antes de realizar las pruebas.	Solicitar apoyo a todo el equipo para poder corregir los errores detectados y poder realizar las pruebas.
R032	Pobre Productividad.	Bajo	Definir tiempos y objetivos de manera que sean alcanzables.	Ajustar los tiempos de entrega, analizar problemas y bloqueos en productividad para proponer soluciones a estos.
R033	No hay suficientes recursos o ingresan demasiado tarde.	Medio	Contar con los recursos de software, hardware y personal suficiente para la ejecución del proyecto.	Tener un plan de respaldo para adquisición de equipo hardware, software y contratación de personal.
R034	capacitación superficial a usuarios finales.	Medio	Realizar capacitaciones constantes a los usuarios finales y hacer un proceso de acompañamiento en la adaptación de uso del sistema.	Capacitaciones adicionales a petición del cliente, proporcionar videos de entrenamiento y material de autoconsumo en el uso del sistema.

Tabla 21 – Matriz de Prevención y Mitigación de Riesgo (Cont.).

Fuente: Elaboración propia.

<b>Código de Riesgo</b>	<b>descripción del Riesgo</b>	<b>Nivel de Riesgo</b>	<b>Medida de prevención</b>	<b>Plan de mitigación</b>
R035	La aplicación no procesa transacciones por segundo como se esperaba.	Medio	Hacer pruebas con cargar masiva de tareas y así llevar la aplicación más allá de los límites para los cuales fue diseñada en la etapa de pruebas.	Realizar las modificaciones necesarias al código para asegurar que la aplicación procesa las transacciones necesarias.
R036	Fallas del hardware limitan la funcionalidad del software.	Medio	Adquirir hardware con certificación técnica y validar las condiciones del lugar donde se encuentra el hardware.	Hacer uso de la garantía proporcionada por el proveedor de hardware y utilizar hardware de reemplazo para asegurar la continuidad del funcionamiento del sistema.
R037	Arquitectura inadecuada por parte del cliente.	Medio	Asegurarse que el cliente tiene la arquitectura adecuada para el correcto funcionamiento de la aplicación.	Tener una propuesta de arquitectura sugerida o una versión ligera del sistema que se adecue a la arquitectura del cliente.
R038	Documentación sobre el uso de la aplicación.	Bajo	Asegurarse que la documentación sea fácil de entender por parte de los usuarios finales y realizarla a la par del desarrollo.	Crear un manual de usuario para mostrar el uso de la aplicación.
R039	Vulnerabilidades del software presentadas en producción.	Medio	Cumplir con todas las pruebas necesarias para asegurar la seguridad de la aplicación en sus etapas de desarrollo y pruebas.	Realizar las modificaciones necesarias y tener establecidos mecanismos de seguridad para el sistema.
R040	Resistencia del personal para cambiar las prácticas del pasado.	Bajo	Mostrar a los usuarios las ventajas que tiene la aplicación en el trabajo diario y como los puede beneficiar.	Realizar demostraciones de la funcionalidad de la aplicación, sostener grupos de enfoque con los usuarios para conocer su punto de vista.

Tabla 21 – Matriz de Prevención y Mitigación de Riesgo (Cont.).

Fuente: Elaboración propia.

<b>Código de Riesgo</b>	<b>descripción del Riesgo</b>	<b>Nivel de Riesgo</b>	<b>Medida de prevención</b>	<b>Plan de mitigación</b>
R041	Software contiene numerosos errores cuando se entrega al cliente.	Medio	Realizar pruebas exhaustivas durante el proceso de construcción para asegurar la calidad del software.	Establecer mecanismos para atender las incidencias y tener personal disponible para realizar mantenimiento al software según sea requerido.
R042	presentación de defectos en ambiente producción.	Muy Alto	Realizar las pruebas en ambiente preproducción, realiza pruebas piloto en ambiente de producción.	Tener un plan para regresar a la última versión estable de la aplicación que cumple con la funcionalidad total, si se presentaran fallas una vez está en producción.
R043	Fallas del software que limitan la funcionalidad de la aplicación.	Medio	Mantener sistema de respaldo de base de datos, actualizaciones de sistema operativo al día.	Tener un sistema de respaldo de base de datos y de servidor de aplicaciones.

Tabla 21 – Matriz de Prevención y Mitigación de Riesgo (Cont.).

Fuente: Elaboración propia.

## 12. Fase de inicio

### 12.1 Especificación del producto

El Sistema auxiliar para control de información académica de estudiantes de medicina mientras se encuentran realizando el período de internado rotativo en los hospitales escuela a los que son asignados cuenta con diferentes módulos que permitieron garantizar que sus objetivos se cumplieran según el diseño planteado en las primeras etapas de este.

Los siguientes catálogos son los que forman parte del sistema:

#### 12.1.1 Módulos:

- **Estudiante**

A este módulo tiene acceso todos los usuarios, aquí se encuentra contenida toda la información de los estudiantes.

- **Reportes**

A este módulo tienen acceso todos los usuarios, pero dependiendo del rol que posean los permisos pueden ser de edición o de solo lectura.

- **Notas**

A este módulo tienen acceso solamente de lectura los estudiantes, pero los docentes y Administradores tiene permisos de edición y control total.

- **Universidad**

A este módulo solamente tienen acceso los administradores, ya que es aquí donde se definen las universidades a las que pertenecen los estudiantes y las cuales tendrán acceso al sistema.

- **Modulo**

Aquí es donde se guarda la información de cada uno de los módulos, solamente los administradores tendrán acceso a este módulo con permisos de control total.

- **Docente**

En este módulo se guarda la información de los docentes, a este tendrán acceso de lectura los docentes y de control total los Administradores, que son los únicos que pueden crear docentes nuevos, los estudiantes no tendrán acceso a este módulo.

- **Hospital**

En este módulo se guarda la información de los hospitales en los cuales se lleva a cabo el internado, a este tendrán acceso de lectura los docentes y estudiantes, pero el control total lo tendrán los Administradores, que son los únicos que pueden agregar o dar de baja hospitales nuevos.

#### 12.1.2 Roles

- **Invitado:** Los estudiantes tendrán este rol, que les permitirá acceso de lectura a la mayoría de los módulos del sistema, podrán generar algunos reportes de notas únicamente sobre su usuario, pero su acceso será limitado.
- **Miembro:** Los docentes tendrán este rol, con el que pueden acceder a la mayoría de los catálogos, modificarlos, crear nuevos registros y tendrán acceso a la mayoría de los reportes, con este rol se podrán generar reportes de notas para los usuarios, para los módulos, divididos por universidades y otros.
- **Administrador:** Este rol lo tendrá el subdirector docente de cada facultad junto con las autoridades superiores del ministerio de salud, con este tendrán acceso total a todos los módulos del sistema, incluyendo la

creación y modificación de nuevas universidades, docentes, alumnos, etc y podrán generar todos los reportes deseados.

### 12.1.3 Tablas

El sistema cuenta con las tablas que se describen a continuación junto con el uso que se da a cada una de ellas y la respectiva descripción.

Uso	Nombre	Descripción
<b>Procesos usuarios</b>	acadNotaModuloPersona	Tabla para el almacenamiento de las notas que obtiene en cada módulo un médico interno
<b>Configuración</b>	acadPeriodoInternado	Tabla para el almacenamiento de los periodos en que se evalúan los módulos para cada interno
<b>Procesos usuarios</b>	grlBeneficiarioSeguroPersona	Tabla para el almacenamiento de contactos que se beneficiarían del seguro en cada de muerte de un interno
<b>Procesos usuarios</b>	grlContactoEmergenciaPersona	Tabla para el almacenamiento de contactos de emergencia de los médicos internos

Tabla 22 – Lista de tablas usadas en el sistema.

Fuente: Elaboración propia.

Uso	Nombre	Descripción
<b>Catalogo</b>	grlDepartamento	Tabla para el almacenamiento de los departamentos de Nicaragua
<b>Catalogo</b>	grlEstado	Tabla para el almacenamiento de los tipos de estado que pueden tener los registros de las tablas
<b>Catalogo</b>	grlEstadoCivil	Tabla para el almacenamiento de los estados civiles que contempla la ley en Nicaragua
<b>Catalogo</b>	grlHospital	Tabla para el almacenamiento de los hospitales de Nicaragua
<b>Configuración</b>	grlHospitalModulo	Tabla para el almacenamiento de que modulo evalúa cada hospital
<b>Catalogo</b>	grlModulo	Tabla para el almacenamiento de los módulos o programas con que evalúan los hospitales a sus internos
<b>Catalogo</b>	grlMunicipio	Tabla para el almacenamiento de los municipios de Nicaragua
<b>Catalogo</b>	grlParentesco	Tabla para el almacenamiento de los tipos de parentesco
<b>Procesos usuarios</b>	grlPersona	Tabla para el almacenamiento de los datos personales
<b>Catalogo</b>	grlSexo	Tabla para el almacenamiento de
<b>Catalogo</b>	grlUniversidad	Tabla para el almacenamiento de las universidades

Tabla 22 – Lista de tablas usadas en el sistema (Cont.).

Fuente: Elaboración propia.

## 12.2 Requerimientos funcionales

### 12.2.1 Diagrama de paquetes

A continuación, se puede observar el diagrama de paquetes del sistema.

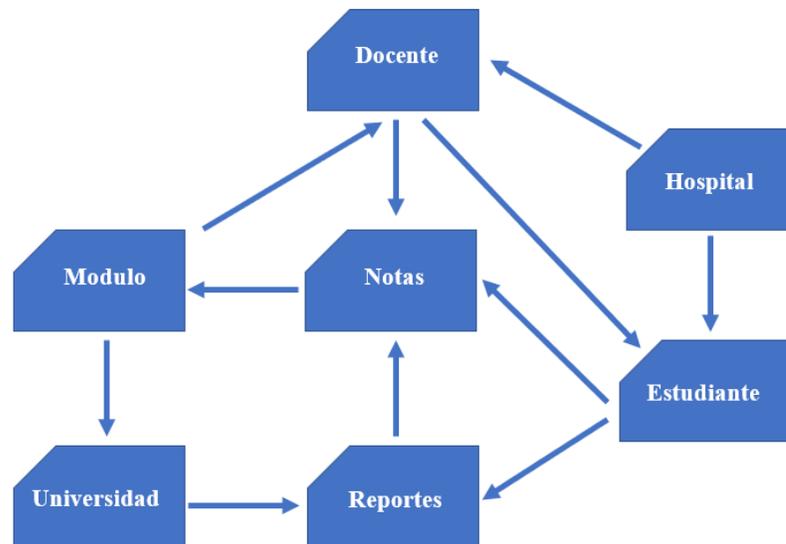


Ilustración 4 – Diagrama de paquetes.

Fuente: Elaboración propia.

#### 12.2.1.1 Funciones del catálogo docente

Este módulo permite la administración de los docentes, asignación de estos a cada uno de los módulos, a cada uno de los hospitales escuela, modificación de su información personal, en caso de que sea necesario se permite el movimiento de docentes entre módulos o entre hospitales, además de la actualización o eliminación de usuarios docentes.

#### 12.2.1.2 Funciones del catálogo Estudiante

Este módulo permite la administración de los estudiantes, creación de nuevos estudiantes cada vez que sea necesario, actualización de datos, asignación de estos a un hospital escuela, modulo o universidad, asignación de los estudiantes a un docente y el movimiento de estos en caso de que se cambien de universidad o no pasen el módulo y deban tomarlo nuevamente con otro grupo.

#### *12.2.1.3 Funciones del catálogo Notas*

Este módulo permite el ingreso, modificación o eliminación de notas para todos los estudiantes, en todos los módulos que ellos van cursando, las notas se van guardando según ellos van realizando actividades y según el docente las ingresa.

#### *12.2.1.4 Funciones del catálogo Modulo*

Este módulo llamado “Modulo” contiene la información del módulo al cual son asignados los estudiantes, los detalles de este, tales como en qué hospital se cursa este módulo, cuantos estudiantes hay asignados, si es primera vez que un estudiante toma el módulo o si es parte de otro grupo y está tomando el módulo nuevamente, etc.

#### *12.2.1.5 Funciones del catálogo Universidad*

Este módulo contiene la información sobre las universidades de las cuales proceden los estudiantes, además de la información de la universidad misma y permite la administración de nuevas universidades que ingresen al sistema o universidades que ya no son parte del sistema y deben darse de baja.

#### *12.2.1.6 Funciones del catálogo Reportes*

Este módulo contiene la reportaría total sobre las notas de los estudiantes, aquí se pueden generar diferentes reportes para obtener información tal como cantidad de estudiantes de una universidad en específico que aprobaron, cual fue el módulo con más estudiantes aprobados o reprobados, la universidad que aporó más alumnos, etc.

#### *12.2.1.7 Funciones del catálogo hospital*

Este módulo contiene la información de los hospitales escuela en los cuales los estudiantes y docentes realizan el internado, permite la actualización de hospitales que están en el sistema, agregar nuevos hospitales y la desactivación de hospitales que ya no forman parte del programa por lo que deben desactivarse en el sistema.

## 12.2.2 Requerimientos funcionales

ID	Enunciado	Tareas
RF1	Gestión de estudiantes.	<ul style="list-style-type: none"><li>• Modelar la clase y realizar los diagramas UML necesarios.</li><li>• Hacer las modificaciones necesarias a la base de datos para incluir una nueva tabla.</li><li>• Validar que los datos necesarios del usuario estudiante son correctos.</li><li>• Validar que el tipo de datos utilizado es el adecuado.</li><li>• Realizar el diseño de la interfaz gráfica para usuarios estudiantes.</li><li>• Validar que la conexión a la base de datos funciona y las entradas se guardan, modifican o eliminan con éxito.</li><li>• Realizar pruebas</li></ul>

Ilustración 23 – Matriz de requerimientos funcionales.

Fuente: Elaboración propia.

ID	Enunciado	Tareas
RF2	Gestión de docentes.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir una nueva tabla.</li> <li>• Validar que los datos necesarios del usuario docente son correctos.</li> <li>• Validar que el tipo de datos utilizado es el adecuado.</li> <li>• Validar que el usuario docente puede ingresar notas en el sistema.</li> <li>• Validar que las notas se asignan a un estudiante.</li> <li>• Realizar el diseño de la interfaz gráfica para usuarios docentes.</li> <li>• Validar que la conexión a la base de datos funciona y las entradas se guardan, modifican o eliminan con éxito.</li> <li>• Realizar pruebas</li> </ul>
RF3	Gestión de Universidades.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir una nueva tabla.</li> <li>• Validar que los datos necesarios de la universidad son correctos.</li> <li>• Validar que el tipo de datos utilizado es el adecuado.</li> <li>• Validar que los módulos, docentes y estudiantes pueden ser asignados a la universidad.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la información de las universidades.</li> <li>• Validar que la conexión a la base de datos funciona y las entradas se guardan, modifican o eliminan con éxito.</li> <li>• Realizar pruebas</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Tareas
RF4	Gestión de módulos.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir una nueva tabla.</li> <li>• Validar que los datos necesarios del módulo son correctos.</li> <li>• Validar que el tipo de datos utilizado es el adecuado.</li> <li>• Validar que las notas, docentes y estudiantes se asignan al módulo.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la información de los módulos.</li> <li>• Validar que la conexión a la base de datos funciona y las entradas se guardan, modifican o eliminan con éxito.</li> <li>• Realizar pruebas</li> </ul>
RF5	Gestión de notas.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir una nueva tabla.</li> <li>• Validar que los datos necesarios de las notas son correctos.</li> <li>• Validar que el tipo de datos utilizado es el adecuado.</li> <li>• Validar que las notas pueden ser asignadas a un estudiante.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la información de las notas a todos los usuarios según su nivel de permisos.</li> <li>• Validar que la conexión a la base de datos funciona y las entradas se guardan, modifican o eliminan con éxito.</li> <li>• Realizar pruebas</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Tareas
RF6	Gestión de Hospitales	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir una nueva tabla.</li> <li>• Validar que los datos necesarios de las notas son correctos.</li> <li>• Validar que el tipo de datos utilizado es el adecuado.</li> <li>• Validar que los estudiantes pueden ser asignados a un hospital.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la información de los hospitales a todos los usuarios según su nivel de permisos.</li> <li>• Validar que la conexión a la base de datos funciona y las entradas se guardan, modifican o eliminan con éxito.</li> <li>• Realizar pruebas</li> </ul>
RF7	Asignación de Estudiantes.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir o actualizar una nueva tabla.</li> <li>• Validar que los datos necesarios de los estudiantes son correctos.</li> <li>• Validar que las relaciones en la base de datos funcionan correctamente.</li> <li>• Validar que a los estudiantes se pueden asignar a un periodo de internado en un hospital.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la asignación de los estudiantes.</li> <li>• Validar que la conexión a la base de datos funciona y las entradas se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Tareas
RF8	Asignación de módulos.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir o actualizar una nueva tabla.</li> <li>• Validar que los datos necesarios de los módulos son correctos.</li> <li>• Validar que las relaciones en la base de datos funcionan correctamente.</li> <li>• Validar que los módulos pueden ser asignados a un docente.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la asignación de los módulos.</li> <li>• Validar que la conexión a la base de datos funciona y las entradas se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>
RF9	Asignación de docentes.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir o actualizar una nueva tabla.</li> <li>• Validar que los datos necesarios de los docentes son correctos.</li> <li>• Validar que las relaciones en la base de datos funcionan correctamente.</li> <li>• Validar que a los docentes les pueden ser asignados estudiantes o modulo.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la asignación de los docentes.</li> <li>• Validar que la conexión a la base de datos funciona y las entradas se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Tareas
RF10	Asignación de periodos.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir o actualizar una nueva tabla.</li> <li>• Validar que los datos necesarios de los periodos son correctos.</li> <li>• Validar que las relaciones en la base de datos funcionan correctamente.</li> <li>• Validar que las fechas de inicio y final de los periodos están definidas.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la asignación de los docentes.</li> <li>• Validar que la conexión a la base de datos funciona y las entradas se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>
RF11	Actualización de información de estudiante realizada por el estudiante.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para actualizar la información en la tabla.</li> <li>• Validar que los datos necesarios del estudiante son correctos.</li> <li>• Validar que las entradas cumplen con el formato establecido.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la información del estudiante al estudiante.</li> <li>• Validar que el estudiante puede recibir correos del sistema.</li> <li>• Validar que el estudiante puede usar el servicio de reinicio de contraseña.</li> <li>• Validar que la conexión a la base de datos funciona y las modificaciones se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Tareas
RF12	Actualización de información de estudiante hecha por otro usuario administrador.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para actualizar la información en la tabla.</li> <li>• Validar que los datos necesarios del estudiante son correctos.</li> <li>• Validar que las entradas cumplen con el formato establecido.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la información del estudiante al administrador.</li> <li>• Validar que el usuario administrador puede reiniciar contraseña para otros usuarios.</li> <li>• Validar que el usuario administrador puede cambiar el alias de otros usuarios.</li> <li>• Validar que el usuario administrador puede cambiar información personal de otros usuarios.</li> <li>• Validar que la conexión a la base de datos funciona y las modificaciones se guardan.</li> <li>• Realizar pruebas</li> </ul>
RF13	Actualización de información de docente realizada por un usuario administrador.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para actualizar la información.</li> <li>• Validar que los datos necesarios del docente son correctos.</li> <li>• Validar que las entradas cumplen con el formato establecido.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la información del docente al administrador.</li> <li>• Validar que el usuario administrador puede cambiar el alias del docente.</li> <li>• Validar que el usuario administrador puede cambiar información personal del docente.</li> <li>• Validar que el usuario administrador puede reasignar al docente a otro modulo.</li> <li>• Validar que la conexión a la base de datos funciona y las modificaciones se guardan.</li> <li>• Realizar pruebas</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Tareas
RF14	Actualización de información de universidad.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para actualizar la información en la tabla.</li> <li>• Validar que los datos necesarios de la universidad son correctos.</li> <li>• Validar que las entradas cumplen con el formato establecido.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la información de la universidad al administrador.</li> <li>• Validar que la información de la universidad se puede modificar y/o eliminar.</li> <li>• Validar que la conexión a la base de datos funciona y las modificaciones se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>
RF15	Actualización de información de modulo.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para actualizar la información en la tabla.</li> <li>• Validar que los datos necesarios del módulo son correctos.</li> <li>• Validar que las entradas cumplen con el formato establecido.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra la información del módulo al administrador.</li> <li>• Validar que al módulo se le puede asignar otro docente.</li> <li>• Validar que a un módulo se le pueda asignar más de un docente.</li> <li>• Validar que la conexión a la base de datos funciona y las modificaciones se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Tareas
RF16	Corrección de notas a un estudiante realizadas por un usuario docente.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para actualizar la información en la tabla.</li> <li>• Validar que los datos necesarios del estudiante y de las notas son correctos.</li> <li>• Validar que las entradas cumplen con el formato establecido.</li> <li>• Realizar el diseño de la interfaz gráfica que permita realizar la corrección de notas a un estudiante por parte de un docente.</li> <li>• Validar que la conexión a la base de datos funciona y las modificaciones se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>
RF17	Corrección de notas a un estudiante realizadas por un usuario administrador.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para actualizar la información en la tabla.</li> <li>• Validar que los datos necesarios del estudiante y de las notas son correctos.</li> <li>• Validar que las entradas cumplen con el formato establecido.</li> <li>• Realizar el diseño de la interfaz gráfica que permita realizar la corrección de notas a un estudiante por parte de un administrador.</li> <li>• Validar que la conexión a la base de datos funciona y las modificaciones se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Tareas
RF18	Desactivación de un usuario por parte de un administrador.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para actualizar la información en la tabla.</li> <li>• Validar que los datos necesarios del usuario son correctos.</li> <li>• Validar que las entradas cumplen con el formato establecido.</li> <li>• Realizar el diseño de la interfaz gráfica que permita realizar la eliminación de un usuario estudiante o docente por parte de un administrador.</li> <li>• Validar que la conexión a la base de datos funciona y las modificaciones se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>
RF19	Desactivación de una universidad por parte de un administrador.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para actualizar la información en la tabla.</li> <li>• Validar que los datos necesarios de la universidad son correctos.</li> <li>• Validar que las entradas cumplen con el formato establecido.</li> <li>• Realizar el diseño de la interfaz gráfica que permita realizar la eliminación de una universidad por parte de un administrador.</li> <li>• Validar que la conexión a la base de datos funciona y las modificaciones se guardan con éxito.</li> <li>• Realizar pruebas</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Tareas
<b>RF20</b>	Generar reporte de notas de un estudiante por parte del docente.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir una nueva tabla.</li> <li>• Validar que las credenciales del estudiante utilizadas para inicio de sesión traen únicamente las notas de este estudiante.</li> <li>• Validar que el tipo de datos ingresado es el adecuado.</li> <li>• Validar que el estudiante puede ver un histórico de sus notas.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra el reporte de notas por estudiante.</li> <li>• Realizar búsquedas en la base de datos.</li> <li>• Realizar pruebas.</li> </ul>
<b>RF21</b>	Generar reporte de notas de un módulo por parte de un docente.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir una nueva tabla.</li> <li>• Validar que el periodo en el que se hace la búsqueda de notas es el actual.</li> <li>• Validar que un docente pueda filtrar las notas por estudiante.</li> <li>• Validar que el tipo de datos utilizado es el adecuado.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra el reporte de notas por estudiante.</li> <li>• Realizar búsquedas en la base de datos.</li> <li>• Realizar pruebas.</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Tareas
RF22	Generar reporte de notas de un módulo por parte de un usuario administrador.	<ul style="list-style-type: none"> <li>• Modelar la clase y realizar los diagramas UML necesarios.</li> <li>• Hacer las modificaciones necesarias a la base de datos para incluir una nueva tabla.</li> <li>• Validar que la búsqueda de notas se puede hacer en cualquier periodo.</li> <li>• Validar que el administrador pueda filtrar las notas por estudiante.</li> <li>• Validar que el tipo de datos utilizado es el adecuado.</li> <li>• Realizar el diseño de la interfaz gráfica que muestra el reporte de notas por estudiante.</li> <li>• Realizar búsquedas en la base de datos.</li> <li>• Realizar pruebas.</li> </ul>

Tabla 23 – Matriz de requerimientos funcionales (Cont.).

Fuente: Elaboración propia.

### 12.3 Permisos de usuarios

ID	Enunciado	Invitado (Estudiante)	Miembro (Docente)	Administrador (Subdirector docente)	Desarrollador
RF1	Gestión de estudiantes.	RU	RU	CRUD	CRUD
RF2	Gestión de docentes.	R	RU	CRUD	CRUD
RF3	Gestión de Universidades.	R	R	CRUD	CRUD
RF4	Gestión de módulos.	R	R	CRUD	CRUD
RF5	Gestión de notas.	R	RU	CRUD	CRUD
RF6	Gestión de Hospitales.	R	R	CRUD	CRUD

Tabla 24 – Permisos de usuarios.

Fuente: Elaboración propia.

ID	Enunciado	Invitado (Estudiante)	Miembro (Docente)	Administrador (Subdirector docente)	Desarrollador
<b>RF7</b>	Asignación de estudiantes.	R	R	CRUD	CRUD
<b>RF8</b>	Asignación de módulos.	R	R	CRUD	CRUD
<b>RF9</b>	asignación de docentes.	R	R	CRUD	CRUD
<b>RF10</b>	Asignación de períodos.	R	R	CRUD	CRUD
<b>RF11</b>	Actualización de información de estudiante realizada por el estudiante.	RU	R	CRUD	CRUD
<b>RF12</b>	Actualización de información de estudiante hecha por otro usuario administrador.	R	RU	CRUD	CRUD
<b>RF13</b>	Actualización de información de docente realizada por un usuario administrador.	R	RU	CRUD	CRUD
<b>RF14</b>	Actualización de información de universidad.	R	R	CRUD	CRUD
<b>RF15</b>	Actualización de información de modulo.	R	R	CRUD	CRUD
<b>RF16</b>	Corrección de notas a un estudiante realizadas por un usuario docente.	R	RU	CRUD	CRUD

Tabla 24 – Permisos de usuarios (Cont.).

Fuente: Elaboración propia.

ID	Enunciado	Invitado (Estudiante)	Miembro (Docente)	Administrador (Subdirector docente)	Desarrollador
RF17	Corrección de notas a un estudiante realizadas por un usuario administrador.	R	R	CRUD	CRUD
RF18	Desactivación de un usuario por parte de un administrador.	R	R	CRUD	CRUD
RF19	Desactivación de una universidad por parte de un administrador.	R	R	CRUD	CRUD
RF20	Generar reporte de notas de un estudiante por parte del docente.	R	R	CRUD	CRUD
RF21	Generar reporte de notas de un módulo por parte de un docente.	R	R	CRUD	CRUD
RF22	Generar reporte de notas de un módulo por parte de un usuario administrador.	R	R	CRUD	CRUD

Tabla 24 – Permisos de usuarios (Cont.).

Fuente: Elaboración propia.

### 13. Fase de elaboración

Con toda la información obtenida en las etapas anteriores, se procede a realizar validaciones por medio de casos de uso y la elaboración de diagramas de actividad.

#### 13.1 Casos de uso

##### 13.1.1 Agregar Médico interno/docente

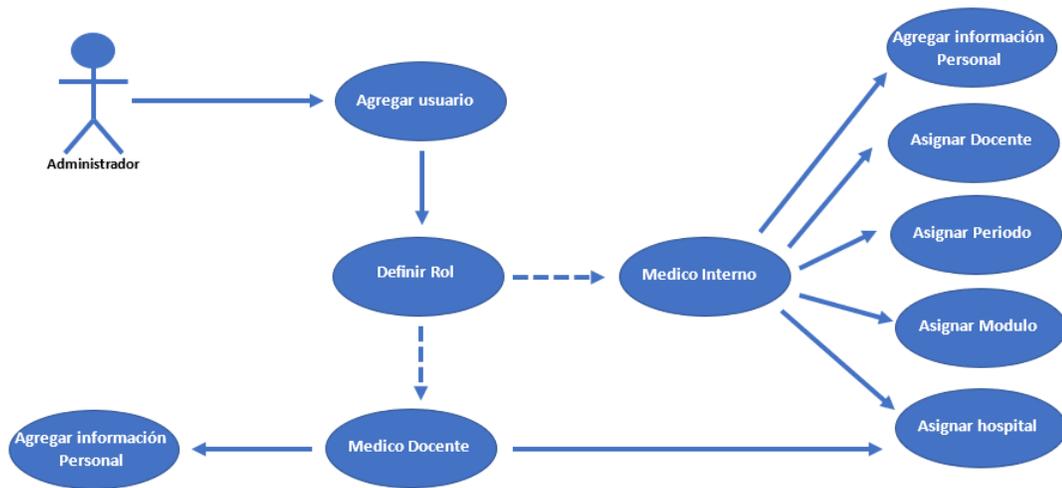


Ilustración 5 – Agregar Médico interno/docente.

##### 13.1.2 Agregar universidad/hospital

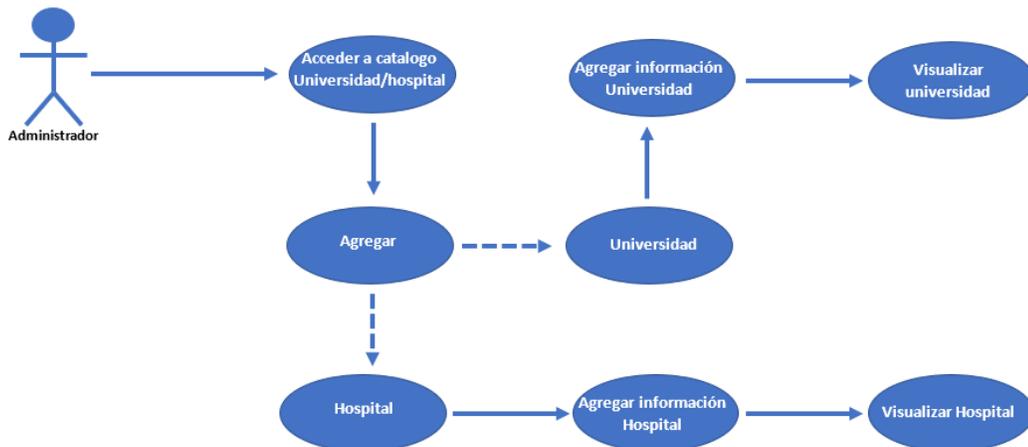


Ilustración 6 – Agregar universidad/hospital

### 13.1.3 Agregar departamento/municipio

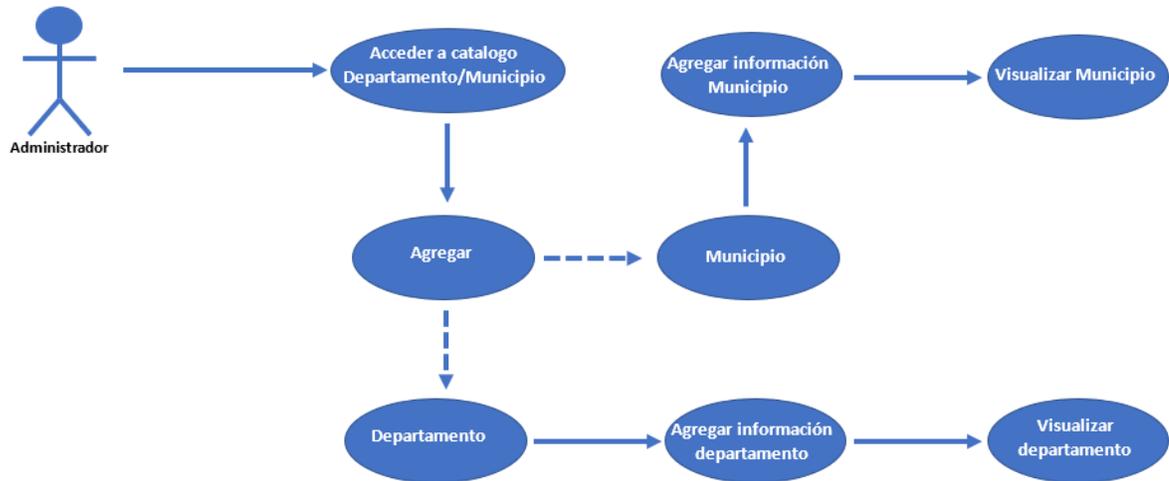


Ilustración 7 – Agregar departamento/municipio.

### 13.1.4 Agregar modulo

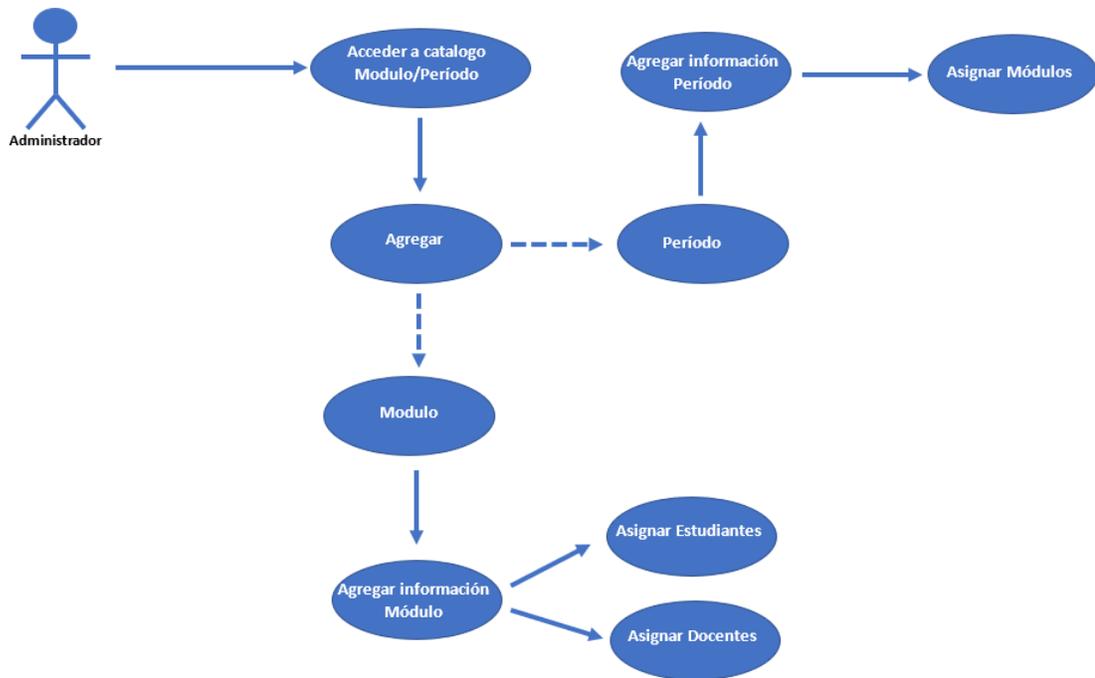


Ilustración 8 – Agregar modulo/Período.

### 13.1.5 Agregar notas



Ilustración 9 – Agregar notas.

### 13.1.6 Modificar notas



Ilustración 10 – Modificar notas.

### 13.1.7 Generar reporte de notas

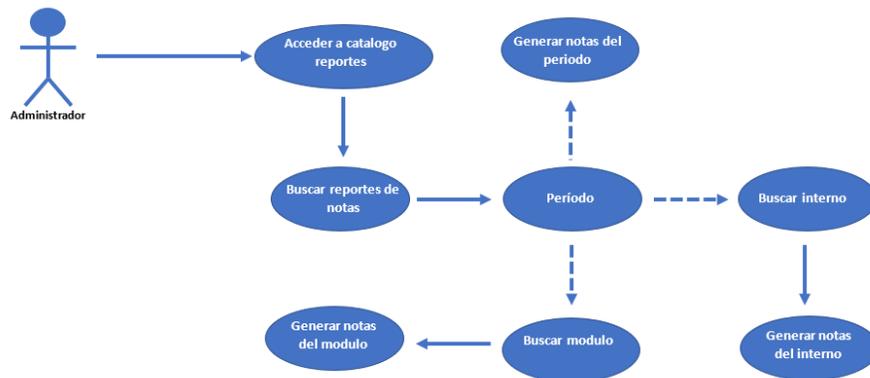


Ilustración 11 – Generar reporte de notas.

## 13.2. Diagrama Entidad-Relación

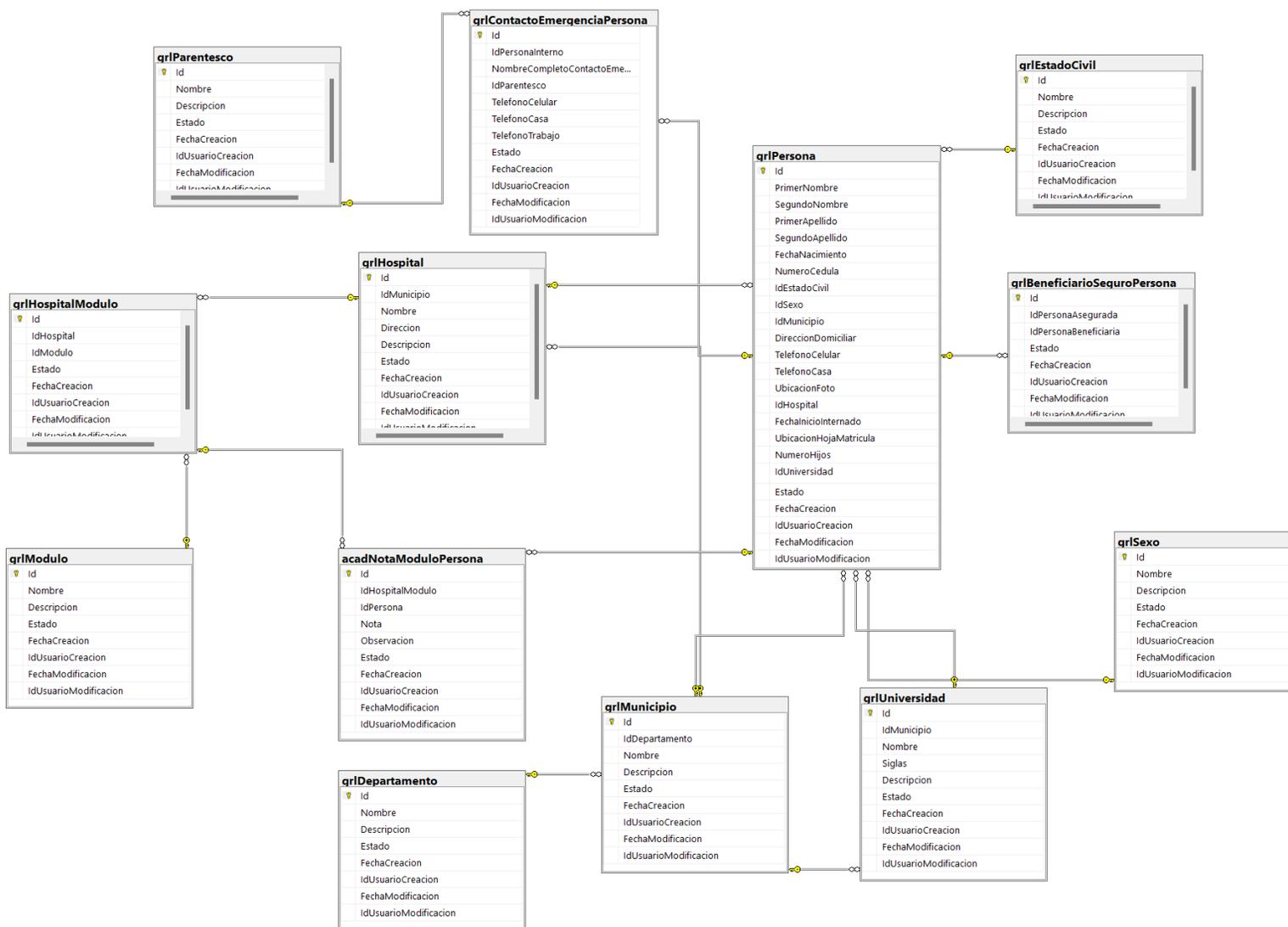


Ilustración 12 – Diagrama entidad relación.

### 13.3. Diagrama de Actividades

#### Inicio de sesión por primera vez

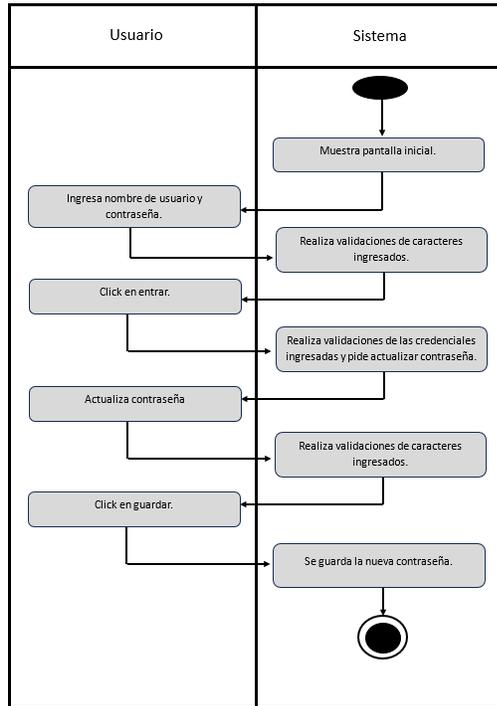


Ilustración 13 – Diagrama actividad – Primer inicio de sesión

#### Crear Medico interno

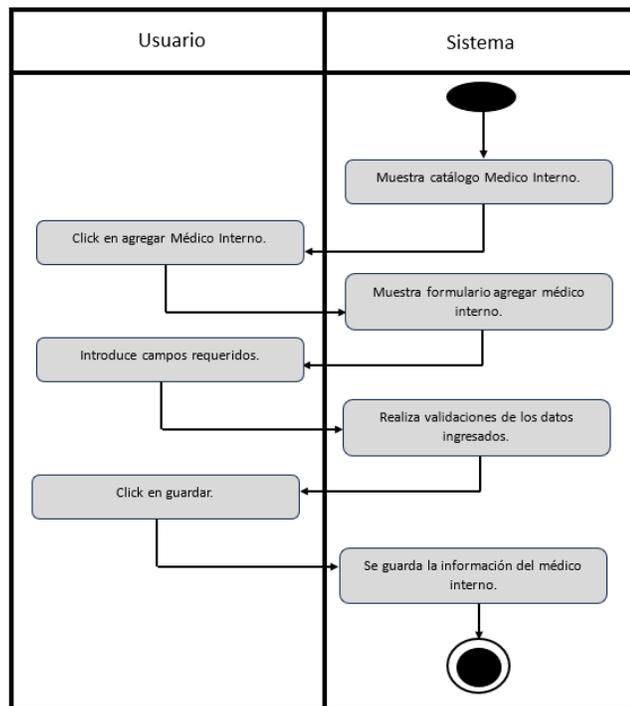


Ilustración 14 – Diagrama actividad – Crear médico interno.

## Crear medico docente

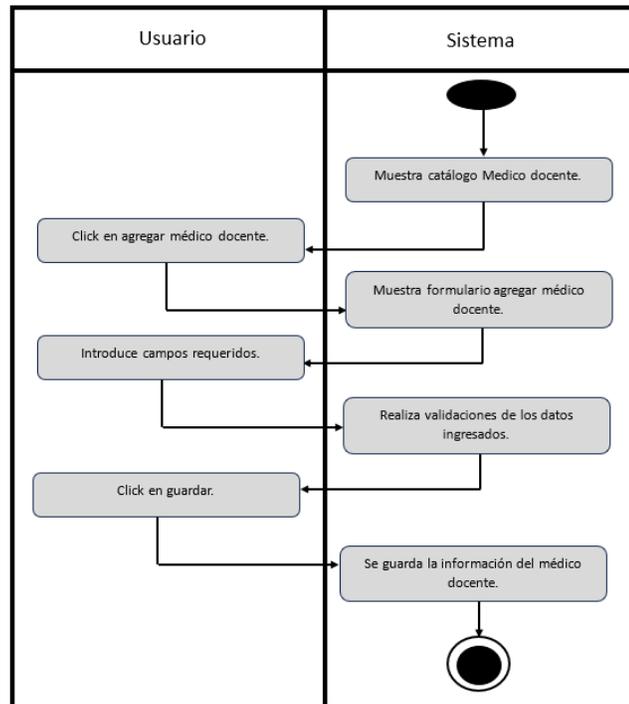


Ilustración 15 – Diagrama actividad – Crear médico docente.

## Crear departamento

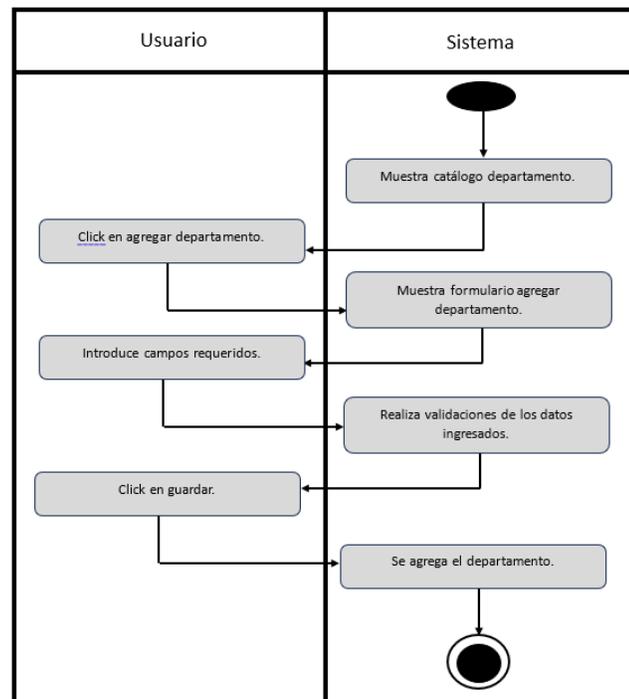


Ilustración 16 – Diagrama actividad – Crear departamento.

## Crear municipio

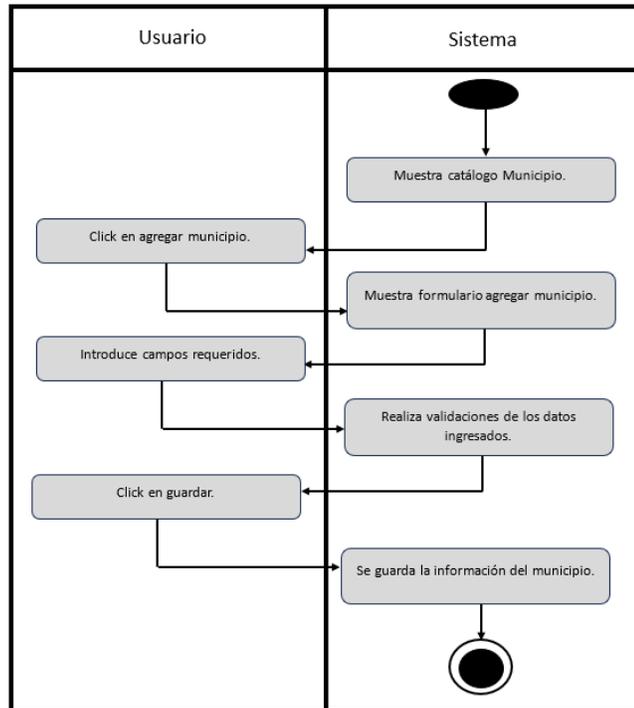


Ilustración 17 – Diagrama actividad – Crear municipio.

## Crear Hospital

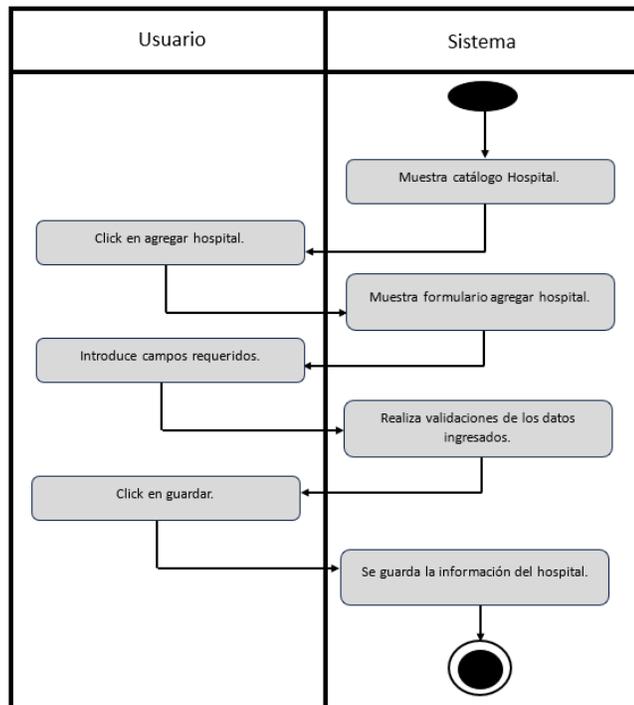


Ilustración 18 – Diagrama actividad – Crear Hospital.

## Crear módulo

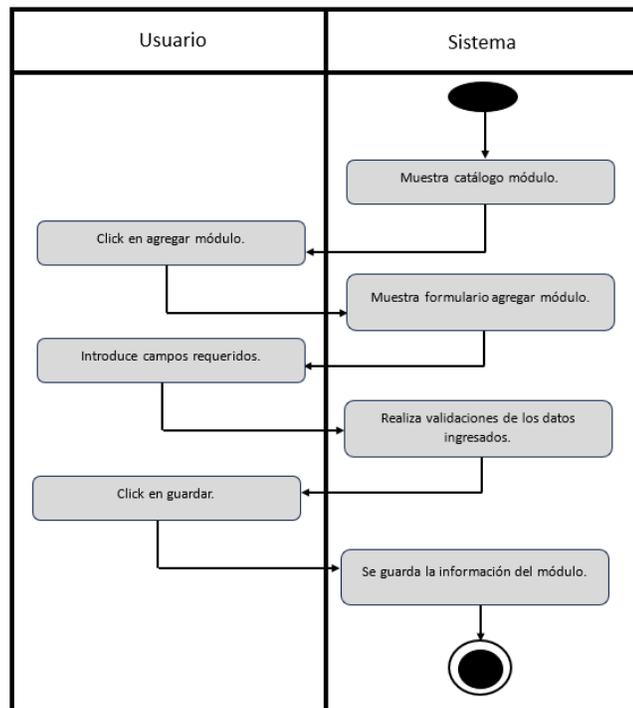


Ilustración 19 – Diagrama actividad – Crear Modulo.

## Crear universidad

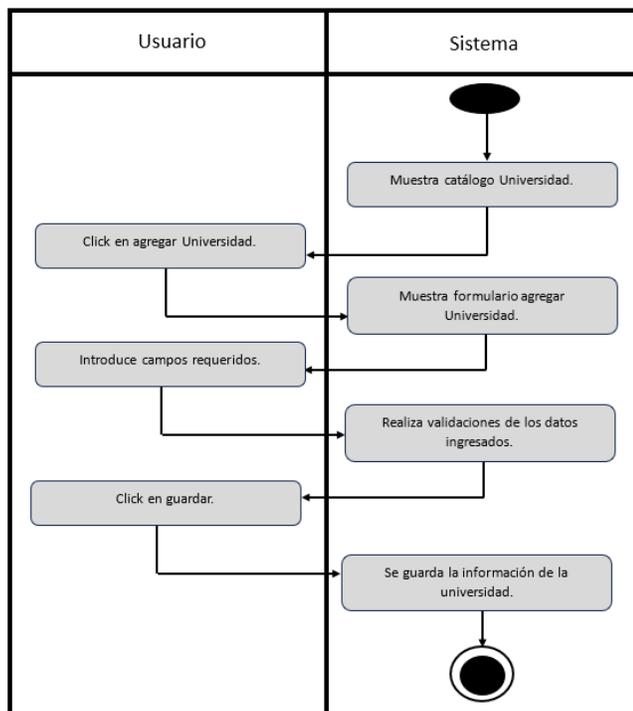


Ilustración 20 – Diagrama actividad – Crear universidad.

## Crear período

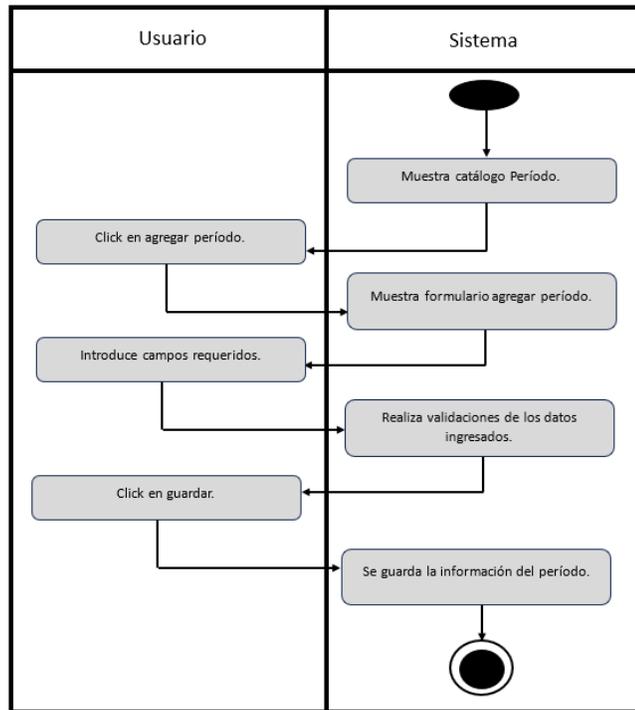


Ilustración 21 – Diagrama actividad – Crear Período.

## Ingresar notas

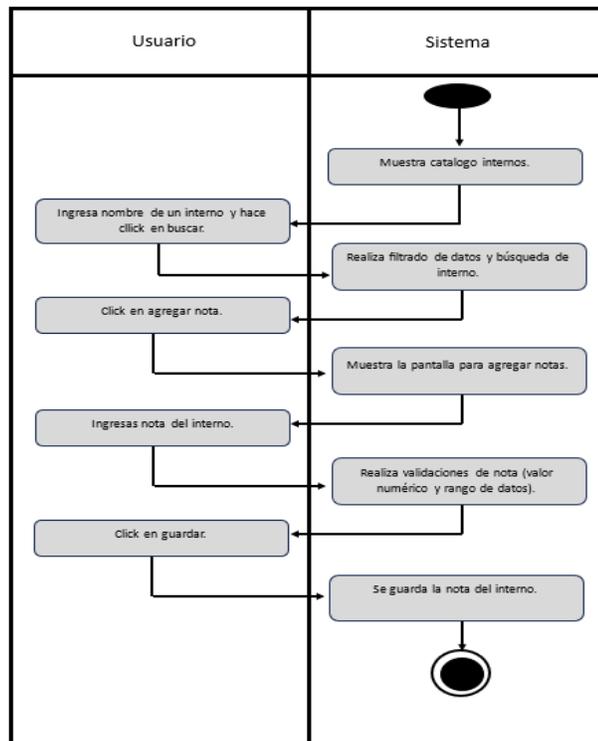


Ilustración 22 – Diagrama actividad – ingresar notas.

## Visualizar reporte de notas

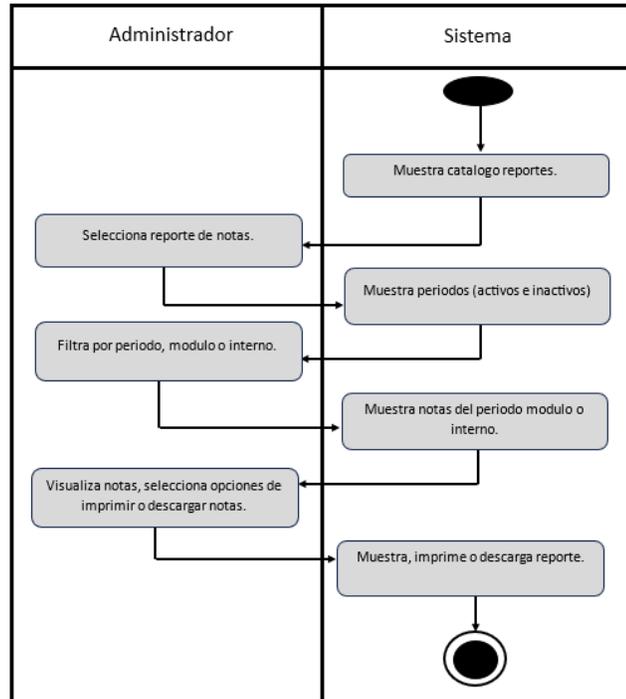


Ilustración 23 – Diagrama actividad – Visualizar notas.

## 13.4. Arquitectura de software

### 13.4.1 Cliente

Al ser un sistema web, el acceso al mismo es por medio de un navegador web, el sistema está optimizado para funcionar en cualquier navegador compatible con HTML ya que este es el principal componente de las interfaces de usuario además de las hojas de estilo CSS, de igual manera el sistema puede ser accedido desde dispositivos móviles con sistema operativo Android, iOS o Chromium.

### 13.4.2 Servidor Web

Para el desarrollo de este proyecto se seleccionó trabajar con la infraestructura ya existente por parte del cliente, ya que al tratarse de un ente gubernamental hay ciertos requerimientos de seguridad adicionales que debieron ser tomados en consideración y que por ley son requeridos, debido a esto la aplicación se alojara en un servidor IIS que la organización ya tiene configurado y desde el cual están

funcionando otras aplicaciones internas de la organización, la configuración, mantenimiento y recuperación a fallos de este servicio ya están previamente establecidos y son manejados por un equipo interno de la organización dedicado a eso específicamente, por motivos de seguridad y debido a estar fuera del alcance de este sistema estos no nos son revelados y por ende no son tomados en cuenta para esta etapa.

#### 13.4.3 Gestor de Bases de Datos

De igual manera que en inciso anterior debido a regulaciones de ley se ha seleccionado un servidor de SQL 2016 que actualmente está en uso en la organización, este se encargara de todo el manejo y almacenamiento de la información del sistema. Al ser un servidor en función el equipo dedicado a Bases de Datos de la organización es el encargado de realizar acciones tales como el respaldo y recuperación en caso de fallo, por medio de mecanismos preestablecidos que al estar fuera del alcance de este sistema no son tomados en consideración en esta etapa.

## 14. Fase de construcción

### 14.1. Modelado de datos

Trabajando sobre el modelo de datos depurado que resulto luego de aplicar la metodología de análisis al sistema se agregaron tablas y relaciones para mejorar y agregar funcionalidad a los módulos:

#### 14.1.1 Modulo Docente

En este módulo a como el mismo nombre lo indica se guarda toda la información de los docentes, tanto personal como académica que luego es utilizada en otros módulos dentro de la aplicación, si la información agregada no es correcta los usuarios docentes no podrán acceder al sistema, no se les podrán asignar a los docentes módulos, alumnos o universidades y por ende no podrán agregar notas.

#### 14.1.2 Modulo Estudiante

En este módulo es donde se almacena la información personal de estudiante y posteriormente se almacenarán las notas de esta una vez que los docentes las ingresen al sistema, el estudiante por medio de este mismo modulo podrá acceder y realizar consultas sobre sus notas y los módulos a los que este asignado, de igual manera ver a los docentes, por eso es importante que las relaciones entre las tablas funcionen de manera adecuada y la información sea correcta.

#### 14.1.3 Modulo Notas

Este es el módulo principal del sistema, ya que el objetivo primario de este sistema es el almacenamiento de la notas para que posteriormente se puedan generar reportes sobre estas, este módulo se encuentra en el corazón del sistema y es el único que interactúa con todos los demás módulos del sistema, todos los usuarios tienen acceso a este módulo con diferentes permisos que les son garantizados por medio del rol que les es asignado, si este módulo no funciona de manera correcta el sistema entero se vería comprometido, es por eso que es muy importante la validación de datos y relaciones en este módulo clave.

#### 14.1.4 Modulo Universidad

En este módulo se almacena la información de las diferentes universidades de las cuales vienen los estudiantes, también es utilizado para la generación de reportes que son enviados directamente a la administración del SILAIS y que sirven para tomar decisiones críticas sobre los estudiantes, docentes y módulos de acuerdo con lineamientos y políticas internas.

#### 14.1.5 Modulo módulo

Este módulo que lleva por nombre modulo es el equivalente a una materia o asignatura que los estudiantes tienen que cursar, en el sistema de educación medico docente se conoce como modulo y son 4 diferentes módulos que un estudiante debe cursar en una año para poder culminar el internado rotativo, este módulo tiene asignados docentes, estudiantes, notas, universidades y es después del módulo de notas el segundo más importante para el funcionamiento del sistema por lo que la integridad en su funcionamiento es un componente clave del sistema.

#### 14.1.6 Modulo Reportes

El módulo de reportes a como su nombre lo indica es donde se generan todos los reportes del sistema, todos los usuarios tienen acceso a generar diferentes reportes dependiendo de sus roles, los estudiantes que pueden generar su reporte individual de notas ya sea de uno o varios módulos, los docente pueden generar reportes de notas de uno o varios estudiantes y las autoridades superiores que pueden generar reportes ya sea a nivel de estudiantes, docentes, módulos o universidades para así poder obtener estadísticas que luego serán usada en la toma de decisiones sobre el futuro de cada uno de los estudiantes dentro del sistema de salud nacional y para estadísticas internas a nivel de SILAS.

#### 14.1.6 Modulo Hospital

Este módulo hospital contiene a como su nombre lo indica la información de los hospitales escuela en los cuales se lleva a cabo el internado, estos hospitales son importantes ya que en estos se llevan a cabo los módulos del internado, estos módulos tienen asignados docentes, que a su vez tiene asignados un grupo de estudiantes, a los cuales se les ingresan notas, por lo que es de vital importancia para asegurar el correcto funcionamiento del sistema.

## 14.2 Interfaz de usuario

Los módulos que tendrá el sistema están diseñados de acuerdo con los siguientes estándares para brindar consistencia en todas las páginas. El diseño se adaptará a diferentes dimensiones de pantallas, utilizando Foundation Framework, el rango soportado va de  $320 <$  a  $\geq 1200px$ , existe compatibilidad con navegadores escritorio y móviles gracias a las hojas de estilo que contiene el Framework.

### 14.2.1 Pantalla de inicio de sesión

Pantalla de inicio de sesión permite el acceso al sistema a los usuarios que previamente obtuvieron credenciales para hacer uso de este.



Ilustración 24 – Pantalla de inicio de sesión.

### 14.2.2 Pantalla principal

Una vez el usuario accede al sistema se mostrará la pantalla principal en la cual se encuentran un menú de navegación para los diferentes módulos del sistema.

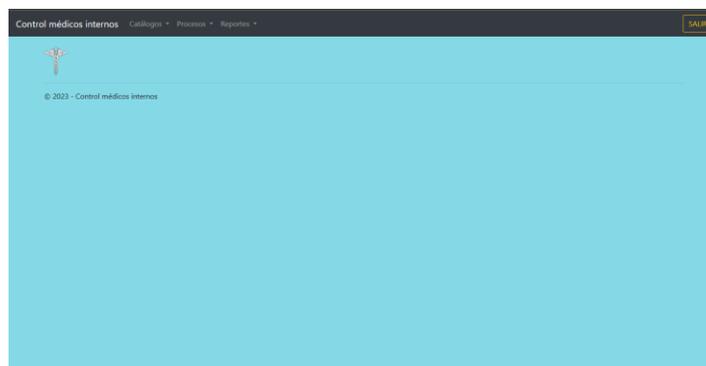


Ilustración 25 – Pantalla principal del sistema.

### 14.2.3 Pantalla para crear médico interno (estudiante)

Pantalla para poder crear nuevo estudiante llenando todos los campos requeridos en el formulario de registro.

The screenshot shows a web form titled "Agregar médico interno" with a light blue background. It is divided into three main sections: "Datos personales", "Datos de contacto", and "Datos académicos".

- Datos personales:** Includes fields for "Primer nombre:", "Segundo nombre:", "Primer apellido:", "Segundo apellido:", "Fecha nacimiento:" (with a date picker), "Número cédula:" (with a text input containing "0010101230002U"), "Estado civil:" (dropdown), "Sexo:" (dropdown), and "Numero hijos:" (text input with "0").
- Datos de contacto:** Includes "Departamento domicilio:" (dropdown), "Municipio domicilio:" (dropdown), "Teléfono celular:" (text input), "Teléfono domicilio:" (text input), and "Dirección domiciliar:" (text area).
- Datos académicos:** Includes "Hospital:" (dropdown), "Fecha inicia internado:" (date picker), and "Universidad:" (dropdown).

At the bottom left is a blue "Atras" button, and at the bottom right is a green "GUARDAR" button.

Ilustración 26 – Pantalla para crear estudiante.

### 14.2.4 Pantalla para crear docente

Pantalla para poder crear nuevo docente llenando todos los campos requeridos en el formulario de registro.

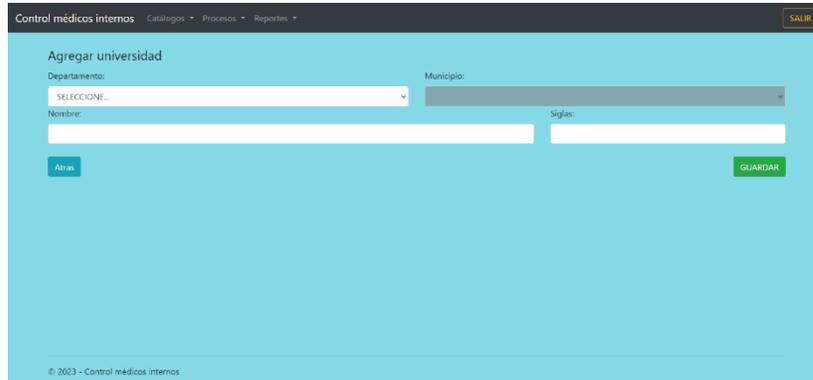
The screenshot shows a web form titled "Agregar credenciales" with a light blue background. It is part of a system called "Control médicos internos", as indicated by the header. The form includes fields for "Tipo usuario:" (dropdown), "Primer nombre:", "Primer apellido:", "Número cédula:", "Usuario:", "Contraseña:", and "Correo electrónico:".

At the bottom left is a blue "Atras" button, and at the bottom right is a green "GUARDAR" button. A yellow "SALIR" button is visible in the top right corner of the page header.

Ilustración 27 – Pantalla para crear Docente.

### 14.2.5 Pantalla para crear Universidad

Pantalla para poder crear nueva universidad llenando todos los campos requeridos en el formulario de registro.

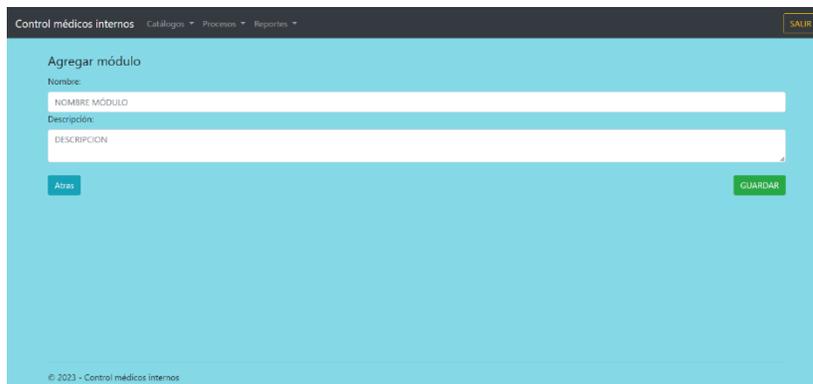


The screenshot shows a web interface for adding a university. The header includes 'Control médicos internos' and navigation links for 'Catalogos', 'Procesos', and 'Reportes'. A 'SALIR' button is in the top right. The main form is titled 'Agregar universidad' and contains the following fields: 'Departamento' (a dropdown menu with 'SELECCIONE...' selected), 'Municipio' (a dropdown menu), 'Nombre' (a text input field), and 'Siglas' (a text input field). There are 'Atras' and 'GUARDAR' buttons at the bottom of the form. A copyright notice '© 2023 - Control médicos internos' is visible at the bottom left.

Ilustración 28 – Pantalla para crear Universidad.

### 14.2.6 Pantalla para crear Modulo

Pantalla para poder crear nuevo módulo llenando todos los campos requeridos en el formulario de registro.



The screenshot shows a web interface for adding a module. The header is identical to the previous form. The main form is titled 'Agregar módulo' and contains the following fields: 'Nombre' (a text input field with 'NOMBRE MODULO' entered), and 'Descripción' (a text input field with 'DESCRIPCION' entered). There are 'Atras' and 'GUARDAR' buttons at the bottom of the form. A copyright notice '© 2023 - Control médicos internos' is visible at the bottom left.

Ilustración 29 – Pantalla para crear Modulo.

### 14.2.7 Pantalla para modificar información de usuarios (Estudiante o docente)

La pantalla muestra una interfaz para listar y poder editar los datos de los estudiantes previamente registrados o desactivarlo.

Catálogo credenciales					
CAMILO MARTINEZ			BUSCAR	DESHACER	Agregar credencial
Tipo usuario	Nombre completo	Cedula	Correo electrónico	Estado	
ESTUDIANTE	SAMI AL MASAGEDI	8882811890002U		ACTIVO	
ESTUDIANTE	KATHERINE DUMAS	0010410900005H	kdumas@gmail.com	ACTIVO	
DOCENTE	TANIA CATRO	0421206650002U	tania.castro@gmail.com	ACTIVO	
ADMINISTRADOR	CAMILO MARTINEZ	0011705880002L		ACTIVO	

Ilustración 30 – Pantalla para modificar información de usuarios.

### 14.2.8 Pantalla para modificar / desactivar universidad

La pantalla muestra una interfaz para listar y poder editar los datos de las universidades previamente registradas o desactivarlas.

Control médicos internos | Catálogos | Procesos | Reportes | SALIR

**Editar universidad**

Nombre:  Siglas:

[Atras](#) [GUARDAR](#)

© 2023 - Control médicos internos

Ilustración 31 – Pantalla para modificar/desactivar universidad.

### 14.2.9 Pantalla para modificar / desactivar modulo

La pantalla muestra una interfaz para listar y poder editar los datos de los módulos previamente registrados o desactivarlos.

Control médicos internos | Catálogos | Procesos | Reportes | SALIR

**Editar módulo**

Nombre:  Descripción:

[Atras](#) [GUARDAR](#)

© 2023 - Control médicos internos

Ilustración 32 – Pantalla para modificar/desactivar modulo.

#### 14.2.10 Pantalla para ingresar / modificar notas

La pantalla muestra una interfaz para ingresar las notas de un estudiante previamente registrado.

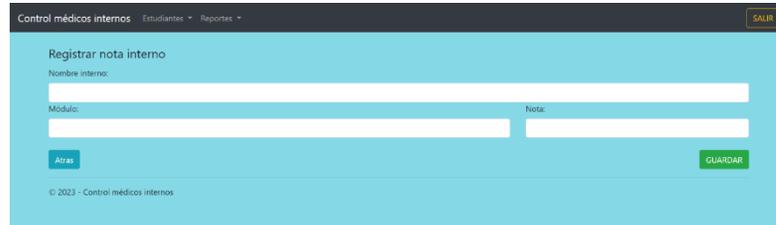


Ilustración 33 – Pantalla para ingresar / modificar notas.

#### 14.2.11 Pantalla donde el estudiante genera reportes

La pantalla muestra una interfaz desde la cual un estudiante previamente registrado puede visualizar sus notas.

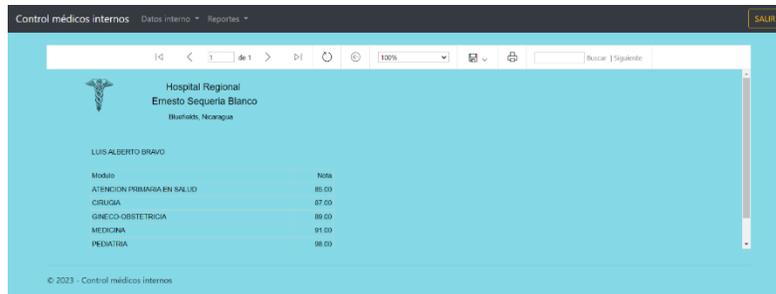


Ilustración 34 – Pantalla para que un estudiante genere su reporte de notas.

#### 14.2.12 Pantalla donde el subdirector docente genera reportes

La pantalla muestra una interfaz desde la cual un docente puede visualizar un reporte de las notas de un estudiante o de un módulo.

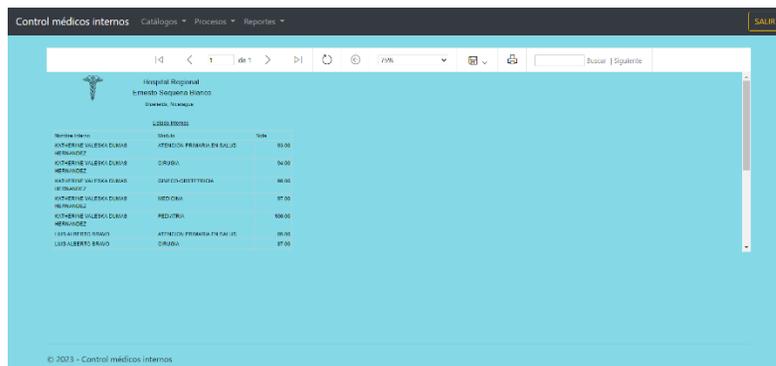


Ilustración 35 – Pantalla para que el subdirector docente genere reporte de notas de estudiante o modulo.

### 14.3. Casos de Prueba

En los siguientes casos de pruebas evaluaremos los procesos más importantes y fundamentales del sistema ya que sin estos el sistema como tal no cumpliría su objetivo ni alcance.

ID	CP01
Caso de Prueba	Primer inicio de sesión
Descripción	Valida que las credenciales del usuario sean correctas y se modifique la contraseña.
Precondiciones	Ingresar nombre de usuario y contraseña.
Valor de entrada	Nombre de usuario: salmasagedi2 Contraseña inicial: XXXX
Pasos de entrada	Ingresar campos requerido en el formulario. Dar click en botón entrar.
Resultado esperado	Cambio de contraseña.
Resultado obtenido	El sistema muestra la ventana de cambio de credenciales.

Tabla 25 – Caso de prueba 1.

ID	CP02
Caso de Prueba	Cambio de contraseña inicial.
Descripción	Solicitud para que el usuario cambie la contraseña inicial que le fue asignada.
Precondiciones	Se debe de introducir una contraseña que cumpla con los criterios previamente definidos para ser validada por el sistema.
Valor de entrada	Nueva contraseña
Pasos de entrada	Ingresar campos requeridos en el formulario. Dar click en Guardar.
Resultado esperado	Pantalla principal del sistema.
Resultado obtenido	El sistema muestra la pantalla principal.

Tabla 26 – Caso de prueba 2.

ID	CP03
Caso de Prueba	Creación de médico interno.
Descripción	Ingreso de todos los datos de un médico interno para ser usados en el sistema.
Precondiciones	Ingresar todos los datos solicitados en el formulario.

Tabla 27 – Caso de prueba 3.

ID		CP03
<b>Valor de entrada</b>		Ingresamos Primer nombre, segundo nombre, primer apellido, segundo apellido, fecha de nacimiento, número de cedula, estado civil, sexo, número de hijos, departamento de domicilio, municipio de domicilio, teléfono de domicilio, dirección de domicilio, departamento de la universidad de donde viene, municipio de la universidad de donde viene, nombre de la universidad, departamento en el que está el hospital, municipio en el que está el hospital y nombre del hospital.
<b>Pasos de entrada</b>	<b>de</b>	Ingresar campos requeridos en el formulario. Dar click en Guardar.
<b>Resultado esperado</b>		Mensaje exitoso.
<b>Resultado obtenido</b>		El sistema muestra el mensaje esperado.

Tabla 27 – Tabla de caso de prueba 3 (Cont.).

ID		CP04
<b>Caso de Prueba</b>		Creación de medico docente.
<b>Descripción</b>		Ingreso de los datos personales del médico docente para ser usados en el sistema.
<b>Precondiciones</b>		Ingresar todos los datos solicitados en el formulario.
<b>Valor de entrada</b>		Ingresamos primer nombre, segundo nombre, primer apellido, segundo apellido, numero de cedula, fecha de nacimiento.
<b>Pasos de entrada</b>	<b>de</b>	Ingresar campos requeridos en el formulario. Dar click en Guardar.
<b>Resultado esperado</b>		Mensaje exitoso.
<b>Resultado obtenido</b>		El sistema muestra el mensaje esperado.

Tabla 28 – Caso de prueba 4.

ID		CP05
<b>Caso de Prueba</b>		Agregar Hospital.
<b>Descripción</b>		Agregar un nuevo hospital escuela para ser utilizado en el sistema.
<b>Precondiciones</b>		Ingresar todos los datos solicitados en el formulario.

Tabla 29 – Caso de prueba 5.

ID		CP05
Valor de entrada		Ingresamos departamento en el que se encuentra el hospital, municipio en el que se encuentra el hospital, nombre del hospital, dirección del hospital y opcionalmente se puede agregar una descripción del hospital.
Pasos de entrada	de	Ingresar campos requeridos en el formulario. Dar click en Guardar.
Resultado esperado		Mensaje exitoso.
Resultado obtenido		El sistema muestra el mensaje esperado.

Tabla 29 – Caso de prueba 5(Cont.).

ID		CP06
Caso de Prueba		Agregar Modulo.
Descripción		Agregar un nuevo módulo (clase) para ser utilizado en el sistema.
Precondiciones		Ingresar todos los datos solicitados en el formulario.
Valor de entrada		Ingresamos el nombre del modulo y una breve descripción sobre este.
Pasos de entrada		Ingresar campos requeridos en el formulario. Dar click en Guardar.
Resultado esperado		Mensaje exitoso.
Resultado obtenido		El sistema muestra el mensaje esperado.

Tabla 30 – Caso de prueba 6.

ID		CP07
Caso de Prueba		Generar reporte de notas por el interno.
Descripción		Generación de un reporte de notas obtenidas por parte de un médico interno.
Precondiciones		Iniciar sesión como un usuario con rol de Estudiante.
Valor de entrada		Credenciales de usuario estudiante.
Pasos de entrada		Acceder al catálogo de reportes, seleccionar el reporte de notas.
Resultado esperado		Generación de reporte de notas.
Resultado obtenido		Se genera reporte de notas.

Tabla 31 – Caso de prueba 7.

ID CP08	
<b>Caso de Prueba</b>	Generar reporte de notas por el docente.
<b>Descripción</b>	Generación de un reporte de notas para todos los médicos docente en el periodo actual.
<b>Precondiciones</b>	Iniciar sesión como usuario con rol de docente.
<b>Valor de entrada</b>	Nombre del médico interno a buscar.
<b>Pasos de entrada</b>	Acceder al catálogo de reportes, seleccionar el reporte de notas, introducir el nombre de un médico interno, click en buscar.
<b>Resultado esperado</b>	Generación de reporte de notas.
<b>Resultado obtenido</b>	Se genera el reporte de notas.

Tabla 32 – Caso de prueba 8.

ID CP09	
<b>Caso de Prueba</b>	Modificar notas.
<b>Descripción</b>	Modificación de la nota de un médico interno por parte de un docente o del subdirector docente.
<b>Precondiciones</b>	Iniciar sesión como un usuario con rol de docente / administrador.
<b>Valor de entrada</b>	Nombre del médico interno a buscar.
<b>Pasos de entrada</b>	Acceder al catálogo de reportes, seleccionar el reporte de notas, introducir el nombre de un médico interno, click en buscar, seleccionar nota a modificar, modificarla y luego dar click en guardar.
<b>Resultado esperado</b>	Modificación de la nota.
<b>Resultado obtenido</b>	Se modifica la nota con éxito.

Tabla 33 – Caso de prueba 9.

## Resumen de pruebas funcionales

Se realizaron las pruebas funcionales de los procesos más importantes realizados por el sistema y no se encontraron problemas de desempeño ni del lado del cliente, ni de lado de los servidores, el procesamiento de datos no representa sobrecarga en el servidor de aplicaciones ni en el servidor de bases de datos y la

aplicación cliente responde a como se esperaba. De igual manera se realizaron pruebas con varios usuarios en diferentes ubicaciones geográficas del país y los resultados fueron similares, por lo que la aplicación se comporta a como se espera y es totalmente funcional.

## **15. Conclusiones**

Mediante la elaboración de este proyecto se ha podido dotar a los docentes con una herramienta informática con las capacidades de ayudar en el manejo de notas y generar todos los reportes necesarios para que estos puedan llevar el control del desempeño de los médicos internos mientras estos realizan su internado rotativo en los diferentes hospitales escuela del país, permitiendo que esto sea un proceso estandarizado y sistematizado.

Esta herramienta facilitara no solo el seguimiento de los médicos internos en tiempo real, mientras estén en el internado, sino que también el llevar un control histórico de sus notas luego de que finalice este periodo, lo cual es de gran beneficio para las autoridades del SILAIS al agilizar el proceso de toma de decisiones y evaluación del aprendizaje de los médicos internos.

Esto ha sido posible luego de realizar las validaciones necesarias con todas las partes involucradas, definir el alcance de este proyecto partiendo de los requerimientos, luego pasando al diseñando la arquitectura necesaria y utilizando las herramientas CAD disponibles, para posteriormente realizar la codificación de cada uno de los módulos a ser utilizados por los diferentes tipos de usuarios.

## 16. Recomendaciones.

Para garantizar el buen desempeño del sistema se recomienda:

- Validar la posibilidad de anexar módulos adicionales, por ejemplo, para controlar asistencia tanto de los médicos docentes como de los médicos internos.
- Asignación de una persona para que periódicamente este dando mantenimiento al sistema, correcciones al código fuente y validando el correcto funcionamiento de este.
- Explorar la posibilidad de migrar las bases de datos y servidores de aplicaciones a la nube para así poder garantizar la alta disponibilidad del sistema.
- Crear un calendario para las actualizaciones periódicas de los servidores de aplicación y bases de datos, asegurarse así que están en óptimas condiciones para su funcionamiento.
- Realizar respaldos periódicos de la base de datos que está en uso y limpieza de la información no necesaria de esta para mantener el correcto funcionamiento de esta.

## 17. Bibliografía

<sup>1</sup> MDN contributors, (Agosto 28, 2023). HTML Basics, Mozilla Developers Network Web Docs. [https://developer.mozilla.org/en-US/docs/Learn/Getting\\_started\\_with\\_the\\_web/HTML\\_basics](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/HTML_basics)

<sup>2</sup> ESIC Business & Marketing School, (2023, enero 12). ¿Qué son las aplicaciones web? ESIC. <https://www.esic.edu/rethink/tecnologia/que-son-las-aplicaciones-web-c>

<sup>3</sup> MDN contributors, (Julio 18, 2023). Conceptos Basicos de HTML, Mozilla Developers Network Web Docs. [https://developer.mozilla.org/es/docs/Learn/Getting\\_started\\_with\\_the\\_web/HTML\\_basics](https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/HTML_basics)

<sup>4</sup> MDN contributors, (Julio 24, 2023). JavaScript, Mozilla Developers Network Web Docs. <https://developer.mozilla.org/es/docs/Web/JavaScript>

<sup>5</sup> MDN contributors, (Marzo 13, 2023). CSS, Mozilla Developers Network Web Docs. <https://developer.mozilla.org/es/docs/Web/CSS>

<sup>6</sup> Application architecture for .NET: Designing applications and services, Microsoft Press, 2003, ISBN0735618372

<sup>7</sup> Arnaud, B (Febrero 2021) ASP.NET con C# en Visual Studio 2019, <https://www.ediciones-eni.com/libro/asp-net-con-c-en-visual-studio-2019-diseno-y-desarrollo-de-aplicaciones-web-9782409029592.pdf>

<sup>8</sup> AXELOS Limited, (Mayo 18, 2021)¿Que es el SQL? y ¿Para qué sirve?, Nextech.

<https://nextech.pe/que-es-sql-server-y-para-que-sirve>

<sup>9</sup> Menzinsky, A., López, G., Palacio, J (Julio 2016) SCRUM Manager

[https://www.scrummanager.com/files/sm\\_proyecto.pdf](https://www.scrummanager.com/files/sm_proyecto.pdf)

<sup>10</sup> Prezi. (21 de Mayo de 2014). Ciclo de vida de un Software según Pressman, prezi.com.

[Ciclo de vida de un software, según Pressman. by \(prezi.com\)](https://prezi.com/gi4cghztosjg/metodologias-para-el-desarrollo-de-aplicaciones-web/)

<sup>11</sup> Pérez, R. (2 de Abril de 2011). Metodologías para el Desarrollo de Aplicaciones Web, prezi.com.

<https://prezi.com/gi4cghztosjg/metodologias-para-el-desarrollo-de-aplicaciones-web/>

<sup>12</sup> Granda, S. (2021), Diferentes modelos de ciclo de vida de un software,

INLOGIQ <https://inlogiq.com/diferentes-modelos-de-ciclos-de-vida-del-software/>

<sup>13</sup> Pursell, S. (20 enero, 2023), Metodología Agile: qué es y cómo aplicarla a tu proyecto <https://blog.hubspot.es/marketing/metodologia-agile>

<sup>14</sup> [TUY2007] Tuya, J., Ramos, I., Dolado, J. (2007). Técnicas cuantitativas para la gestión en la ingeniería del software. España: Netbiblo, S. L.

<sup>15</sup> INGENIERÍA DE SOFTWARE. Novena edición. Ian Sommerville. Traducción: Víctor Campos Olguín. Traductor especialista en Sistemas Computacionales.

<sup>16</sup> INGENIERÍA DE SOFTWARE. Novena edición. Ian Sommerville. Traducción: Víctor Campos Olgún. Traductor especialista en Sistemas Computacionales.

<sup>17</sup> [SOM2005] Sommerville, I. (2005). Ingeniería del software. Madrid: Pearson Educación, S.A.

<sup>18</sup> Webcindario.com. (2021). Riesgo del Software, Webcindario.com  
<https://ingenieriasoft.webcindario.com/gestion-y-planificacion-de-proyectos/planificacion-de-proyectos-de-software/riesgo-del-software.html>

<sup>19</sup> Retos en Supply Chain. (2020, mayo 14). *Guía PMBOK: definición, estructura y tips de estudio*. Retos en Supply Chain - EAE Business School Barcelona  
<https://retos-operaciones-logistica.eae.es/que-es-la-guia-pmbok-y-como-influye-en-la-administracion-de-proyectos/>

<sup>20</sup> Retos en Supply Chain. (2020, mayo 14). *Guía PMBOK: definición, estructura y tips de estudio*. Retos en Supply Chain - EAE Business School Barcelona  
<https://retos-operaciones-logistica.eae.es/que-es-la-guia-pmbok-y-como-influye-en-la-administracion-de-proyectos/>

<sup>21</sup> Anyconnector.com (Marzo 12, 2021). *Que es el riesgo tecnológico? Como administrar las amenazas empresariales*. Anyconnector.com.  
<https://anyconnector.com/es/digital-transformation-strategy/technology-risk.html>

<sup>22</sup> Anyconnector.com (Marzo 12, 2021). *Que es el riesgo tecnológico? Como administrar las amenazas empresariales*. Anyconnector.com.  
<https://anyconnector.com/es/digital-transformation-strategy/technology-risk.html>

<sup>23</sup> Torres, O. P. (s/f). *Gestión de riesgos en proyectos de software*.

Piranirisk.com. <https://www.piranirisk.com/es/blog/gestion-de-riesgos-proyectos-de-software>

<sup>24</sup> Retos en Supply Chain. (2020, mayo 14). *Guía PMBOK: definición, estructura y tips de estudio*. Retos en Supply Chain - EAE Business School Barcelona

<https://retos-operaciones-logistica.eae.es/que-es-la-guia-pmbok-y-como-influye-en-la-administracion-de-proyectos/>

## 18.Cronograma de ejecución

Actividades	Mes 1				Mes 2				Mes 3				Mes 4				Mes 5				Mes 6			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
<b>Etapa 1: Recopilacion de informacion / Necesidades</b>	█																							
Entrevistas.	█																							
Observaciones.		█																						
Revisión de documentos.			█																					
consolidado de informacion.				█																				
Validación con tutores.					█																			
<b>Etapa 2: Analisis, Especificaciones y Diseño</b>	█																							
Definición de la arquitectura del software.				█	█																			
Diseño de las bases de Datos.					█	█																		
Diseño de formularios e interfaces.						█	█	█																
Validación con tutores.							█	█	█															
Revisión.								█	█															
<b>Etapa 3: Codificación y pruebas del software</b>	█																							
Codificación de interfaces graficas.									█	█	█	█												
Codificación de formularios y reportes.										█	█	█	█											
Validación de entradas y salidas de datos.											█	█	█	█										
Elaboración de documentación de soporte.				█	█	█	█	█	█	█	█	█	█	█	█									
Pruebas del sistema.																█	█							
Reunión con los tutores.																	█							
Revisión.																		█	█					
<b>Etapa 4: Implementación y evaluación.</b>	█																							
Instalación del sistema.																						█		
Evaluación por parte de usuarios finales																						█	█	
Revisión.																								
<b>Etapa 5: Informes finales</b>	█																							
Redacción y revisión del informe final.																							█	
Aprobación de documento final.																							█	
Predefensa.																								█
Defensa																								█

## **19. Anexos**

1. Entrevista 03 de marzo.....	112
2. Entrevista 20 de marzo.....	116
3. Entrevista 10 de abril.....	120
4. Reglamento Internado.....	123
5. Publicación Ley 787 .....	151

Managua, viernes 3 de marzo de 2023.

Se realiza reunión vía videollamada utilizando Microsoft Teams entre los bachilleres Sami Khaled Al Masagedi Castro, Luis Alberto Bravo y Camilo Ernesto Martínez Carrillo contando con la participación de la Doctora Alma Rosa Castro Baltodano, subdirectora docente del Hospital Regional Doctor Ernesto Sequeira Blanco, ubicado en Bluefields, cabecera de la Región Autónoma Costa Caribe Sur. Videollamada inicia a las a las 07:00 PM, el objetivo de esta reunión es conocer cuál es el proceso que se tiene actualmente para el control de notas de los médicos mientras realizan su internado rotativo, conocer los requerimientos y establecer el alcance para la creación de un sistema que ayude a automatizar este proceso.

A continuación, se detallan las preguntas realizadas durante la entrevista:

- **¿Qué es el internado rotativo?**

El internado rotativo es un periodo de tiempo luego de que se finalizan todas las asignaturas de la carrera de medicina y antes de la realización del servicio social, el internado rotatorio es algo que se realiza a nivel internacional ya que es el momento en el que algunas escuelas de medicina perfilan alumnos hacia una especialidad en el transcurso de la carrera, además de que el internado es el año de practica para afianzar parte teórica de la carrera y hacer la transición a la parte práctica. Durante el internado hay una parte teórica y una parte práctica, ambas son calificadas.

- **¿Cuántos estudiantes se reciben en un año?**

Normalmente se reciben entre 40 y 50 internos al año que son divididos entre todos los hospitales escuela del país. Si son demasiado grande los grupos se dividen en 2 para la parte práctica, pero para seminarios teóricos van todos juntos.

- **¿El internado rotativo se realiza en todos los hospitales?**

MINSA tiene internado en aproximadamente 20 hospitales en el país.

- **¿De qué universidades vienen los alumnos para el internado?**

De todas las universidades que imparten la carrera de medicina, las más conocidas son UAM, URACAN, UNAN-León, UNAN-Managua y BICU. Es un requisito para poder tener el título final el haber completado el internado en alguno de los hospitales escuela.

- **¿Como se hace la división de los alumnos en los hospitales escuela?**

Hay una comisión que incluye Minsa, docencia del ministerio de salud, decanatura de medicina UNAN-León y decanatura de medicina UNAN-Managua que se encarga de asignar a los alumnos a cada hospital. Una vez realizado esto Minsa central solamente envía listado con números de cedula de todos los estudiantes y los hospitales a los cuales son asignados.

- **¿Cómo se lleva el control de las notas?**

Las calificaciones deben de estar en físico en el expediente de cada estudiante y en una tabla de Excel que es la que se envía al final del curso. La hoja final está dividida en una parte teórica y una parte práctica, ambas tienen calificaciones independientes, todos los formatos se suman y se ponderan al final del módulo para sacar la nota final.

- **¿Todos los hospitales escuela manejan el mismo sistema de calificaciones?**

Por regulación del Minsa así es, ya que no existe un sistema parecido a este en Nicaragua, ni a nivel centroamericano, todo se almacena de manera física.

- **¿Dónde se almacena esta información de los estudiantes?**

En cada hospital escuela se elabora un expediente con todos los datos, esto incluye información personal y notas del estudiante, la parte más importante son las notas, que se recopilan de manera manual en una hoja de Excel y luego se envían al SILAIS central.

- **¿Cuánto tiempo se deben de guardar los documentos de los estudiantes?**

Por regulación interna del MINSA los documentos deben de ser guardados por un período no menor de 10 años, esto incluye toda la documentación y hay bastante información, pero lo más importante son las calificaciones en caso de que se pierdan o suceda algo con estas se tiene que realizar una validación manual, lo que significa buscar en todas las cajas hasta encontrar el expediente del estudiante y esto es algo que sucede con bastante frecuencia.

- **¿Cuántas personas deben de tener acceso a la información de los estudiantes?**

Los docentes deben de tener acceso a la información de sus estudiantes, pero a nivel nacional hay una única persona encargada de recopilar todas las hojas de Excel en las que se suben las notas y unir las, esto luego se presenta al SILAIS central y la directora debe de tener acceso total a esta información.

- **¿Cómo se lleva a cabo el proceso actualmente?**

Cada docente califica a sus estudiantes de la manera que ellos deseen para acumular puntaje, siempre respetando que se califique tanto la parte teórica como la parte práctica, se hacen las ponderaciones necesarias para asegurarse que sea 40% teórico y 60% práctico, esto se debe de escribir en una hoja de Excel que es el formato utilizado a nivel nacional y además de enviarse al SILAS guardarse en el expediente de cada estudiante. (Una copia de esta hoja fue proporcionada por la doctora para que tengamos una idea de que es lo que se necesita)

- **¿Basándose en su experiencia cuales son las limitantes de este proceso?**

Las principales limitantes son el tiempo que se toma en buscar la información, el tiempo para la recopilación de esta y el depender solamente de una persona, de su trabajo manual y de su disponibilidad de agenda, a esto le podemos sumar que no hay información en tiempo real ya el consolidado se hace hasta el final del curso y no podemos dejar de lado el factor humano, la persona encargada del

consolidado es muy responsable pero siempre hay errores en el proceso que nos hacen validar dos y hasta tres veces una nota.

Debido a la disponibilidad de horario de las partes involucradas, se da por finalizada la entrevista y se agendará una segunda entrevista para entrar en más detalles sobre el proceso que se desea realizar una vez analizada la información y e presentará a la doctora Castro una propuesta del alcance que puede tener el sistema y lo que se puede hacer. La reunión se da por finalizada a las 08:39 PM.

---

Br. Sami Khaled Al Masagedi Castro

---

Br. Luis Alberto Bravo

---

Br. Camilo Ernesto Martínez Carrillo

---

Dra. Alma Rosa Castro Baltodano

Managua, lunes 20 de marzo de 2023.

Se realiza reunión vía videollamada utilizando Microsoft Teams entre los bachilleres Sami Khaled Al Masagedi Castro, Luis Alberto Bravo y Camilo Ernesto Martínez Carrillo contando con la participación de la Doctora Alma Rosa Castro Baltodano, subdirectora docente del Hospital Regional Doctor Ernesto Sequeira Blanco, ubicado en Bluefields, cabecera de la Región Autónoma Costa Caribe Sur. Videollamada inicia a las a las 06:00 PM, el objetivo de esta reunión es conocer cuáles son los procesos diferentes tipos de usuarios, los niveles de permisos que estos necesitan tener, los procesos clave y los datos que el sistema para el control de notas de los médicos internos debe de tener.

Primeramente, se hace una lectura de la transcripción de la última reunión, las preguntas hechas y las respuestas dadas por la doctora para asegurarnos que la información recopilada es la correcta y que todos estamos de acuerdo hasta este punto, se realizan algunas correcciones menores y se obtiene el aprobado de la doctora Castro.

A continuación, se detallan las preguntas realizadas durante la entrevista:

- **¿Cómo es el sistema de calificaciones actual?**

Este se divide en 2, una parte teórica y una parte práctica. La parte teórica 40% de evaluación - mitad equivale a seminarios, hojas de registro de calificaciones, 2 exámenes uno en la semana número 5 y otro en la semana número 10. Cada examen se pondera a 10 puntos, y cada seminario igual, los estudiantes necesitan 24 puntos = 60% para aprobar. La parte práctica 60% - calificación al finalizar cada turno de 24 horas, esta se da tomando en cuenta el desempeño del estudiante durante las prácticas de sala y un examen práctico con un paciente.

- **¿Cómo se dividen los periodos en los cuales se generan notas?**

Se dividen en 5 rotaciones durante el año, cada una dura entre 10 y 11 semanas y son en diferentes especialidades: pediatría, atención primaria, gineco-obstetricia, cirugía (ortopedia), medicina general y todos.

- **¿Qué sucede si un estudiante no obtiene la nota mínima para pasar un módulo?**

Ya sea que prueben o no el módulo los estudiantes siempre pueden iniciar el siguiente modulo, pero para poder finalizar el internado deben de tener aprobados todos los módulos.

- **¿Quiénes deben de tener acceso a la información de los estudiantes y sus notas?**

La directora docente del SILAIS monitorea actividades, desarrollo de clases y siempre se pone en copia cuando se mandan notas al MINSA, ella no modifica ni consolida la información, pero debe tener acceso para visualizarla, los docentes deben de tener acceso a subir las notas, modificarlas, crear usuarios y generar reportes. Los usuarios deben de tener acceso a subir su información personal y a ver sus notas. Además de ellos solamente se necesitaría el acceso de las personas de informática.

- **¿Qué información de los estudiantes es la que se guardará en el sistema?**

Lo principal son las notas y la información personal básica que incluye nombre, apellido, estado civil, dirección, teléfono, foto de cedula de identidad, foto de carné del INSS, foto actualizada de ellos el MINSA central nos pide agregar también la Universidad de la que vienen, adicionalmente sería ideal si se pudiera guardar en digital un soporte en caso de subsidios, ausencias, hojas de vacaciones, anecdotario, incidencias, faltas, etc. Esto no afecta las notas, pero es parte del expediente del estudiante.

- **¿Cómo funcionan las vacaciones?**

La primera semana de vacaciones es mientras hacen su rotación en medicina interna o atención primaria, se aplica el descuento de semana santa y luego el descuento de navidad según decreto gubernamental. Esto es algo adicional que no se llevaría en el sistema ya que el enfoque son las notas y nos las ausencias.

- **¿Con que frecuencia se deben de generar reportes?**

Por regulación del MINSA los reportes académicos son generados cada 10 semanas, al final de cada módulo y aquí solamente se necesitan el nombre del estudiante, la universidad de la que viene y las notas del módulo que actualmente se está cursando. Además de este hay un reporte final consolidado que contiene esta misma información, pero para todos los estudiantes del grupo, este reporte es el que se envía a las universidades para efectos de notas.

- **¿Hay algún otro reporte además de las notas que se desee generar?**

En un caso ideal se requeriría guardar el anecdotario del docente que son situaciones que ocurren con el medico interno en su área laboral, problemas, incidencias, faltas que cometieron los estudiantes, esto sería en formato PDF o en imágenes.

- **¿Existe algún tipo de filtrado o agrupación que se deba hacer con la información?**

El objetivo que se busca con el sistema es el de facilitar búsqueda de datos de los internos, anecdotarios, información de RRHH y no llenarse de papeles, por eso se sería ideal poder guardar los adjuntos. Los filtrados necesarios son básicos, agrupar a los alumnos por universidad, sexo, edad, ubicación, etc.

- **¿Se necesitaría que las universidades tengan acceso a la información de las notas?**

Si las bases de datos se pueden compartir esto agilizaría la integración entre entidades, pero esto es algo que primeramente se debe de consultar con el SILAIS central ya que por regulaciones gubernamentales hay algunas cosas que no se pueden hacer, aunque sean para facilitar o mejorar algún proceso.

- **¿Hay algún otro rol de se deba de tomar en consideración para permisos en el sistema?**

Debido a que los grupos deben de llevar estudiantes de cada universidad, cada grupo tiene un coordinador para facilitar organización y dar orientaciones, pero esta persona no necesita ningún tipo de acceso especial al sistema, ya que es solo un estudiante más, esto debido a que cada docente organiza los grupos de la manera que desea y no hay criterio para la organización.

Posiblemente el coordinador docente que es la persona encargada de cada módulo para hacer validaciones de notas deba de tener algún tipo de rol adicional que le permita además de subir las notas hacer otro tipo de correcciones a la información personal de los estudiantes. Para tener una mejor idea de los procesos y roles, la doctora Castro nos ha proporcionado una copia del reglamento del internado.

Debido a la disponibilidad de horario de las partes involucradas, se da por finalizada la entrevista y se agendará una tercera reunión en la cual se analizará la información recopilada hasta el momento y se presentará a la doctora Castro una propuesta de diseño para la base de Datos del sistema, los diferentes roles que tendrán los usuarios y los permisos de cada uno de estos. La reunión se da por finalizada a las 07:49 PM.

---

Br. Sami Khaled Al Masagedi Castro

---

Br. Luis Alberto Bravo

---

Br. Camilo Ernesto Martínez Carrillo

---

Dra. Alma Rosa Castro Baltodano

Managua, lunes 10 de abril de 2023.

Se realiza reunión presencial entre los bachilleres Sami Khaled Al Masagedi Castro, Luis Alberto Bravo y Camilo Ernesto Martínez Carrillo contando con la participación de la Doctora Alma Rosa Castro Baltodano, subdirectora docente del Hospital Regional Doctor Ernesto Sequeira Blanco, ubicado en Bluefields, cabecera de la Región Autónoma Costa Caribe Sur. La Reunión tiene lugar en la casa de habitación de la doctora Castro en Managua, ubicada en Carretera sur, la reunión inicia a las 07:13 PM, el objetivo de esta reunión es presentar a la doctora una propuesta inicial de la base de datos del sistema, la distribución de los usuarios y obtener el visto bueno para así iniciar la codificación de este.

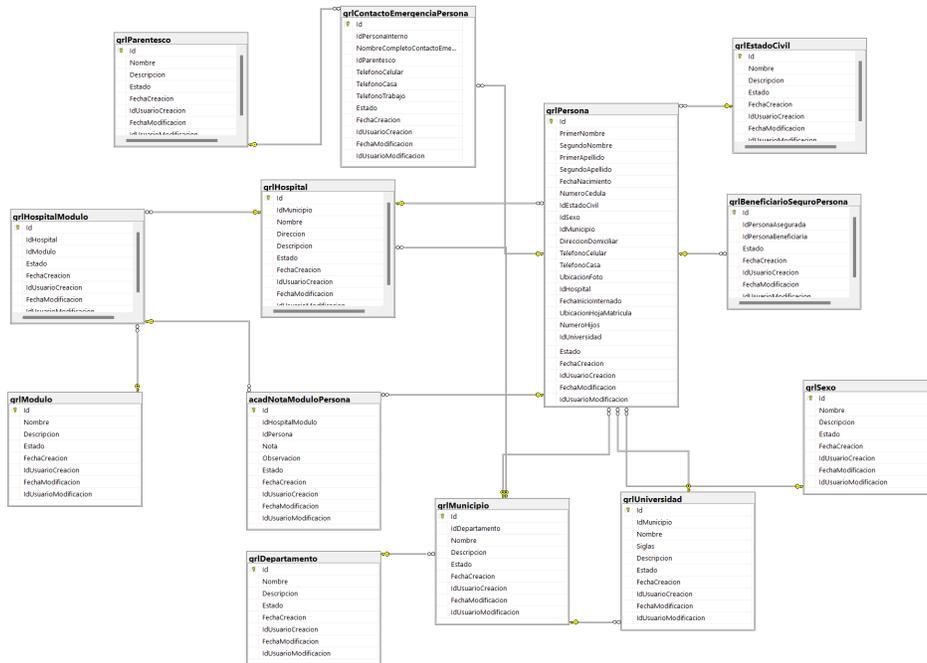
A continuación, se detallan lo discutido durante la reunión:

Tomando en consideración la información recopilada en las sesiones anteriores, se hace la propuesta de creación de las siguientes tablas para contener la información, cada una de estas fue explicada a la doctora Castro.

Uso	Nombre	Descripción
<b>Procesos usuarios</b>	acadNotaModuloPersona	Tabla para el almacenamiento de las notas que obtiene en cada módulo un médico interno
<b>Configuración</b>	acadPeriodoInternado	Tabla para el almacenamiento de los periodos en que se evalúan los módulos para cada interno
<b>Procesos usuarios</b>	grlBeneficiarioSeguroPersona	Tabla para el almacenamiento de contactos que se beneficiarían del seguro en cada de muerte de un interno
<b>Procesos usuarios</b>	grlContactoEmergenciaPersona	Tabla para el almacenamiento de contactos de emergencia de los médicos internos
<b>Catalogo</b>	grlDepartamento	Tabla para el almacenamiento de los departamentos de Nicaragua

Uso	Nombre	Descripción
<b>Catalogo</b>	grlEstado	Tabla para el almacenamiento de los tipos de estado que pueden tener los registros de las tablas
<b>Catalogo</b>	grlEstadoCivil	Tabla para el almacenamiento de los estados civiles que contempla la ley en Nicaragua
<b>Catalogo</b>	grlHospital	Tabla para el almacenamiento de los hospitales de Nicaragua
<b>Configuración</b>	grlHospitalModulo	Tabla para el almacenamiento de que modulo evalúa cada hospital
<b>Catalogo</b>	grlModulo	Tabla para el almacenamiento de los módulos o programas con que evalúan los hospitales a sus internos
<b>Catalogo</b>	grlMunicipio	Tabla para el almacenamiento de los municipios de Nicaragua
<b>Catalogo</b>	grlParentesco	Tabla para el almacenamiento de los tipos de parentesco
<b>Procesos usuarios</b>	grlPersona	Tabla para el almacenamiento de los datos personales
<b>Catalogo</b>	grlSexo	Tabla para el almacenamiento de
<b>Catalogo</b>	grlUniversidad	Tabla para el almacenamiento de las universidades

Una vez hecho esto se presentó y se explicó de manera breve el diagrama entidad relación para que la doctora entendiera como se relacionaban cada una de las tablas entre ellas.



De manera breve se discutieron detalles como la estructura que debe de llevar la pantalla de inicio de sesión, los logos a ser utilizados, la paleta de colores, la longitud de la contraseña entre otros. Debido a la disponibilidad de horario de las partes involucradas, se da por finalizada la reunión y se agendará una tercera reunión para mostrar avances en la codificación de sistema. La reunión se da por finalizada a las 09:22 PM, para hacer constar la veracidad de estas sesiones la doctora Castro firma y sella cada una de las transcripciones de reuniones anteriores y de la reunión del día de hoy.

---

Br. Sami Khaled Al Masagedi Castro

---

Br. Luis Alberto Bravo

---

Br. Camilo Ernesto Martínez Carrillo

---

Dra. Alma Rosa Castro Baltodano