

Área de Conocimiento de Tecnología de la  
Información y Comunicación

# **Desarrollo de un sistema web y aplicación móvil para la gestión eficiente de puntos de venta loto para Grupo Atenea.**

**Trabajo Monográfico para optar al título de  
Ingeniero en Computación**

**Elaborado por:**

Br. Joe Daniel Méndez  
Avilés.

Carnet: 2017-0262U

Br. Mickey Anthony  
Gudiel Reyes.

Carnet: 2017-0881U

**Tutor:**

MSc. Luis Eduardo  
Chávez Mairena.





Área de Conocimiento de  
Tecnología de la Información  
y Comunicación

**SECRETARIA DE ÁREA ACADÉMICA**

**F-8: CARTA DE FINALIZADO PLAN DE ASIGNATURA**

El Suscrito Secretario del **ÁREA DEL CONOCIMIENTO DE TECNOLOGÍA DE LA INFORMACIÓN Y COMUNICACIÓN** hace constar que:

**MÉNDEZ AVILÉS JOE DANIEL**

Carné: **2017-0262U** Turno: **Diurno** Plan de Asignatura: **2015** de conformidad con el Reglamento Académico vigente en la Universidad, ha aprobado todas las asignaturas correspondientes a la carrera de **INGENIERÍA EN COMPUTACIÓN**, en el año 2022 y solo tiene pendiente la realización de una de las formas de culminación de estudio.

Se extiende la presente **CARTA DE FINALIZADO PLAN DE ASIGNATURA**, a solicitud del interesado en la ciudad de Managua, a los quince días del mes de abril del año dos mil veinte y cuatro.

Atentamente,

Ing. Cedrick Elksnherr DallaTorre  
SECRETARIO DE ÁREA ACADÉMICA



☎ Teléfono: (505) 2270 0723

📍 Recinto Universitario Simón Bolívar  
Avenida Universitaria,  
Managua, Nicaragua.  
Apdo: 5595

✉ cedrick.dallatorre@fec.uni.edu.ni  
www.fec.uni.edu.ni



Área de Conocimiento de  
Tecnología de la Información  
y Comunicación

**SECRETARIA DE ÁREA ACADÉMICA**

**F-8: CARTA DE FINALIZADO PLAN DE ASIGNATURA**

El Suscrito Secretario del **ÁREA DEL CONOCIMIENTO DE TECNOLOGÍA DE LA INFORMACIÓN Y COMUNICACIÓN** hace constar que:

**GUDIEL REYES MICKEY ANTHONY**

Carné: **2017-0881U** Turno: **Diurno** Plan de Asignatura: **2015** de conformidad con el Reglamento Académico vigente en la Universidad, ha aprobado todas las asignaturas correspondientes a la carrera de **INGENIERÍA EN COMPUTACIÓN**, en el año 2022 y solo tiene pendiente la realización de una de las formas de culminación de estudio.

Se extiende la presente **CARTA DE FINALIZADO PLAN DE ASIGNATURA**, a solicitud del interesado en la ciudad de Managua, a los quince días del mes de abril del año dos mil veinte y cuatro.

Atentamente,

Ing. Cedrick Elksnherr DallaTorre  
**SECRETARIO DE ÁREA ACADÉMICA**



Teléfono: (505) 2270 0723



Recinto Universitario Simón Bolívar  
Avenida Universitaria.  
Managua, Nicaragua.  
Apdo: 5595



cedrick.dallatorre@fec.uni.edu.ni  
www.fec.uni.edu.ni



Programa  
Académico de  
Ingeniería en  
Computación

Managua, 20 de mayo del 2024.

MSc. Claudia Lucia Benavidez Rugama.  
Directora DACTIC  
Sus manos.

Estimada MSc. Benavidez.

Me dirijo a usted de la manera más atenta para informarle que el trabajo monográfico que lleva título "**Desarrollo de un Sistema Web y Aplicación Móvil para la Gestión Eficiente de Puntos de Venta Loto para Grupo Atenea**". realizado por los bachilleres Br. Joe Daniel Méndez Avilés (Carné 2017-0262U), Br. Mickey Anthony Gudiel Reyes (Carné 2017-0881U), cuya tutoría fue asignada a mi persona, se encuentra revisado, completo y listo para someterse al proceso de calificación final.

Agradeciendo su amable atención, quedo a la espera de sus instrucciones aprovechando la ocasión para enviarle mis más sinceras muestras de consideración y respeto.

Atentamente,

MSc. Luis Eduardo Chávez Mairena  
Tutor

Cc. Archivo



Telefono : (505) 8277-0658



Recinto Universitario Simón Bolívar  
Avenida Universitaria.  
Managua, Nicaragua.



nelson.barrios@fec.uni.edu.ni



Managua, 30 de abril 2024

**Br. Joe Daniel Méndez Avilés**

**Br. Mickey Anthony Gudiel Reyes**

Egresados Programa académico Ingeniería en Computación  
Sus manos.-

Estimados Bachilleres:

Reciban cordiales saludos y éxito en sus actividades.

De acuerdo a carta recibida con fecha 30 de abril del presente año, en donde hacen solicitud de cambio del tema monográfico titulado: **“Sistema web para el control de los puntos de venta loto en Nicaragua, para la empresa “Grupo Atenea”**” al tema de trabajo monográfico titulado: **“Desarrollo de un sistema web y aplicación móvil para la gestión eficiente de puntos de venta loto para Grupo Atenea”**, tomando en consideración las justificaciones expuestas en la misma, tengo a bien comunicarles que se les autoriza dicho cambio de tema.

Esperando el cumplimiento con respecto al tiempo establecido, les reitero mis saludos.

Atentamente,



**Msc. Claudia Lucía Benavidez Rugama**  
**Directora Área de Conocimiento de**  
**Tecnología de la Información y Comunicación**

CC MSc. Luis Eduardo Chávez Mairena– Tutor  
MSc. Cedrick DallaTorre Parrales – Secretario Académico  
Archivo 2024.



Móvil: (505) 8588 8333



Recinto Universitario Simón Bolívar  
Avenida Universitaria.  
Managua, Nicaragua.  
Apdo: 5595



[www.uni.edu.ni](http://www.uni.edu.ni)

## **DEDICATORIA**

Primero a Dios, porque sin Él nada de esto sería posible. Luego, a mis padres por estar siempre para mí, darme ánimos y consejos en el momento preciso, y por sus grandes esfuerzos para sacarme adelante. Sus palabras de orgullo hacia mí funcionaron como un motor para luchar con todas mis fuerzas hasta cumplir la meta. A mis familiares, especialmente a mis hermanos, quienes siempre me han transmitido la satisfacción que sienten al ver mis logros, considerando mis éxitos como si fueran propios. A mis amigos y compañeros de clase, que siempre me brindaron una sonrisa y me apoyaron cuando lo necesité, confiando en que podía lograr mis objetivos. A los docentes que formaron parte de mi vida universitaria, por transmitir su vasto conocimiento con esmero y pasión para formar grandes profesionales. Por último, y no menos importante, al ministerio Nicaragua Resource Network (NRN), quienes me apoyaron en mi carrera universitaria desde mi primer año hasta culminar mi plan de estudios.

**Joe Daniel Méndez Avilés**

Dedico a Dios mi tesis por brindarme la oportunidad de vivir este momento, por las metas alcanzadas y por proporcionarme la fortaleza necesaria en los momentos en que estuve a punto de rendirme. A mis padres, quienes me motivaron a emprender el camino del estudio y respaldaron cada una de mis decisiones hasta el día de hoy. A mi abuela Rafaela Sequeira, por su inmenso amor incondicional y por nunca dejarme solo en la travesía de mi vida hasta este instante. A mi tía, la Profesora Melba Gudiel, y a su esposo, el Profesor Enrique Cerda (que en paz descanse), por inculcarme el valor del estudio, extendiendo esta dedicatoria a toda mi familia por su incondicional apoyo. A mis amigos, por su confianza inquebrantable y su fe en mis capacidades, les ofrezco mi más sincero reconocimiento. A mis profesores, por su inagotable paciencia para enseñar y el tiempo dedicado a guiarme, especialmente a Janine Mairena, Jaqueline López Alvarado, Luis Eduardo Chávez Mairena y Adilson Gonzalez, cuya orientación ha sido fundamental en mi formación.

**Mickey Anthony Gudiel Reyes.**

## **AGRADECIMIENTOS**

Elevamos nuestra gratitud a Dios, que nos ha guiado a través tanto, ya sea en nuestros logros como en las adversidades.

Nuestro más profundo agradecimiento a nuestros padres, cuyo esfuerzo y apoyo incansables han sido fundamentales para alcanzar este logro.

A nuestro tutor, Máster Luis Eduardo Chávez Mairena por apoyarnos en el desarrollo de este trabajo con dedicación y por ayudarnos a crecer profesionalmente.

Al Licenciado Juan Anthony Gutiérrez H propietario de empresa Grupo Atenea, nuestra gratitud por confiar en nosotros y por su apoyo esencial en la realización de este proyecto.

A nuestros maestros, les estamos inmensamente agradecidos por impartirnos un conocimiento excepcional y por su compromiso con la excelencia en la enseñanza.

## RESUMEN

El trabajo monográfico titulado "Desarrollo de un sistema web y aplicación móvil para la gestión eficiente de puntos de venta loto para Grupo Atenea" aborda el diseño, implementación y evaluación de un sistema tecnológico integral destinado a optimizar la administración de los puntos de venta de loto de la empresa Grupo Atenea en Nicaragua.

Este proyecto surge de la necesidad de mejorar la eficiencia operativa y el control administrativo en los puntos de venta, los cuales anteriormente dependían de procesos manuales y herramientas limitadas. La implementación de este sistema busca automatizar tareas críticas como la gestión de ventas, control de inventarios y generación de reportes, permitiendo una administración más precisa y efectiva.

El sistema desarrollado incluye una aplicación web y una aplicación móvil. La aplicación web, diseñada con Node.js y MariaDB, permite la gestión centralizada de todas las operaciones, mientras que la aplicación móvil, creada con Flutter, facilita la movilidad y el acceso a las funciones principales desde cualquier ubicación. Ambas aplicaciones están integradas y sincronizadas, garantizando la coherencia y la disponibilidad de los datos en tiempo real.

Entre las principales ventajas del sistema se destacan:

- **Eficiencia Operativa:** Automatización de procesos, reducción de errores humanos y mejora en la velocidad de atención al cliente.
- **Control Administrativo:** Monitoreo en tiempo real de ventas, inventarios y generación de reportes detallados para la toma de decisiones informadas.
- **Seguridad:** Implementación de medidas robustas para proteger los datos y asegurar la integridad de las transacciones.

No obstante, también se identificaron algunas desventajas:

- **Costo Inicial:** La inversión en desarrollo e implementación puede ser significativa.

- Adaptación al Cambio: Los usuarios deben adaptarse a la nueva tecnología y procedimientos, lo cual puede requerir capacitación adicional.
- En conclusión, este proyecto no solo proporciona una solución efectiva a las necesidades actuales de Grupo Atenea, sino que también establece una base sólida para futuras mejoras y expansiones, posicionando a la empresa a la vanguardia de la tecnología en la gestión de puntos de venta de loto.

## ÍNDICE

<b>1. Introducción</b> .....	<b>1</b>
<b>2. Antecedentes</b> .....	<b>3</b>
<b>3. Justificación</b> .....	<b>5</b>
<b>4. Objetivos</b> .....	<b>6</b>
<b>4.1 Objetivo General</b> .....	<b>6</b>
<b>4.2 Objetivos Específicos</b> .....	<b>6</b>
<b>5. Marco Teórico</b> .....	<b>7</b>
<b>5.1 Métodos ágiles de desarrollo de software</b> .....	<b>7</b>
<b>5.2 Sistemas de información para la gestión empresarial</b> .....	<b>7</b>
<b>5.3 Desarrollo de software</b> .....	<b>8</b>
<b>5.4 Insomnia</b> .....	<b>8</b>
<b>5.5 Diseño de interfaz de usuario (UI)</b> .....	<b>9</b>
<b>5.6 Responsive web design</b> .....	<b>9</b>
<b>5.7 El modelo en espiral</b> .....	<b>10</b>
<b>5.8 Framework</b> .....	<b>10</b>
<b>5.9 MariaDB</b> .....	<b>11</b>
<b>5.10 Node.js</b> .....	<b>11</b>
<b>5.11 NestJS</b> .....	<b>12</b>
<b>5.12 Arquitectura REST</b> .....	<b>12</b>
<b>5.13 Flutter</b> .....	<b>13</b>
<b>5.14 Integración continua</b> .....	<b>13</b>
<b>5.15 UX design</b> .....	<b>14</b>
<b>5.16 PlantUML</b> .....	<b>14</b>
<b>5.17 Aplicación móvil</b> .....	<b>15</b>

5.18 Sistema web .....	15
5.19 Seguridad en Aplicaciones Móviles y Web .....	16
5.20 Digital Ocean.....	16
5.21 Servidor Debian Linux.....	16
5.22 PM2 .....	17
5.23 Certificación SSL .....	17
5.24 Apache HTTP Server Project .....	18
5.25 Socket.io.....	18
5.26 Swagger.....	18
5.27 Clean Architecture.....	19
5.28 Inyección de dependencias .....	19
5.29 Manejador de estado (MobX).....	20
5.30 DTO .....	20
<b>6. Diseño Metodológico .....</b>	<b>21</b>
<b>6.1 Modelo Espiral .....</b>	<b>21</b>
6.1.1 Determinación de objetivos y requerimientos.....	23
6.1.2 Desarrollo de prototipos .....	24
6.1.3 Análisis de riesgos .....	24
6.1.4 Evaluación .....	25
6.1.5 Planificación.....	25
6.1.6 Implementación.....	26
6.1.7 Validación y Preparación del Sistema Implementado .....	26
6.1.8 Mantenimiento y Actualizaciones .....	27
<b>7. Resultados .....</b>	<b>28</b>
<b>7.1 Primera Etapa: Determinación de Objetivos y requerimientos .....</b>	<b>28</b>
7.1.1 Requerimientos Funcionales del Sistema .....	28
7.1.2 Resultados de los Requerimientos No Funcionales del Sistema .....	28
7.1.3 Resultados de los casos de uso.....	32

7.1.4 Detalle de casos de uso modelados. ....	37
7.1.5 Resultados de los Diagramas de Actividades. ....	49
7.1.6 Resultados de los Diagrama de Secuencia. ....	51
<b>7.2 Diseño .....</b>	<b>52</b>
7.2.1 Diseño de Interfaz de Usuario.....	52
7.2.2 Diseño de Contenido.....	52
7.2.3 Diseño Estético .....	52
7.2.4 Análisis y Diseño de Componentes y Despliegue .....	53
<b>7.3 Codificación .....</b>	<b>54</b>
7.3.1 Sistema Web:.....	54
7.3.2 Aplicación Móvil: .....	55
<b>7.4 Pruebas o validación.....</b>	<b>57</b>
7.4.1 Establecimiento de Puntos de Interrupción:.....	57
7.4.2 Ejecución y Observación:.....	57
7.4.3 Inspección de Variables y Flujo:.....	58
7.4.4 Uso de Insomnia para Testeo de Endpoints: .....	58
<b>8. Estudio de Factibilidad .....</b>	<b>60</b>
<b>8.1 Factibilidad Técnica .....</b>	<b>60</b>
8.1.1 Alternativa A:.....	60
8.1.2 Alternativa B:.....	62
<b>8.2 Factibilidad Operativa .....</b>	<b>64</b>
8.2.1 Alternativa A:.....	64
8.2.2 Alternativa B:.....	65
<b>8.3 Factibilidad Económica.....</b>	<b>66</b>
8.3.1 Alternativa A:.....	67
8.3.2 Alternativa B:.....	70
<b>8.4 Factibilidad Legal .....</b>	<b>73</b>
<b>9 Conclusión.....</b>	<b>75</b>
<b>10 Bibliografía .....</b>	<b>77</b>
<b>11 Anexos .....</b>	<b>78</b>

# I. INTRODUCCIÓN

En la actualidad, la tecnología ha revolucionado la forma en que se llevan a cabo los procesos empresariales en todo el mundo. Los avances tecnológicos en las últimas décadas han dado lugar a un cambio en la manera en que las empresas realizan sus operaciones, y, en consecuencia, se ha creado una nueva demanda para la implementación de soluciones tecnológicas en diversos ámbitos empresariales. En este sentido, uno de los sectores que ha venido experimentando un auge en la implementación de tecnologías es el de los puntos de lote, en donde la empresa Grupo Atenea ha logrado una posición líder en el mercado nicaragüense.

La gestión de los puntos de lote es un proceso que demanda tiempo y esfuerzo, especialmente en lo que respecta a la recolección de datos y la administración de las ventas. Es por ello que la empresa Grupo Atenea se ha planteado la necesidad de desarrollar un sistema que permita optimizar estos procesos y hacerlos más eficientes. La solución propuesta es el desarrollo de un sistema web y una aplicación móvil que permita una gestión más eficiente y un control adecuado de las operaciones de los puntos de lote.

Este trabajo monográfico tiene como objetivo presentar el proceso de diseño e implementación de un sistema web y aplicación móvil para el control de los puntos de lote, administrados por la empresa Grupo Atenea. Se pretende desarrollar una aplicación que tenga como principales funciones la optimización de los procesos manuales y escritos, la generación de respaldos de las ventas diarias, la realización de predicciones a medida del histórico de cada día y la mejora del control de ingresos y egresos, todo ello para asegurar una mejor administración y disminuir el margen de error. Además, se busca brindar seguridad y protección a los usuarios que utilicen la aplicación, evitando la manipulación de datos.

Este trabajo monográfico se estructura en once capítulos, cada uno dedicado a aspectos específicos del proyecto. Comienza con una introducción que brinda una visión general, seguida de un análisis exhaustivo de los antecedentes y la

justificación del proyecto. Los objetivos, tanto generales como específicos, se establecen con claridad para guiar el desarrollo del proyecto. El marco teórico abarca una amplia gama de temas relevantes, desde métodos ágiles de desarrollo de software hasta tecnologías específicas como Flutter y Node.js. En el diseño metodológico se detallan los pasos a seguir para la ejecución del proyecto, seguido de la presentación de los resultados obtenidos y un análisis de su viabilidad. La conclusión resume las principales contribuciones del trabajo, seguida de una bibliografía que detalla las fuentes consultadas y los anexos que complementan la investigación.

## II. ANTECEDENTES

El presente trabajo monográfico se enfoca en el desarrollo de un sistema web y una aplicación móvil para optimizar la eficiencia y control en los puntos loto, administrados por la empresa Grupo Atenea. Para entender la necesidad de este sistema, es necesario considerar el contexto en el que se encuentra la empresa y los puntos loto en Nicaragua.

Grupo Atenea es una empresa dedicada a la administración y operación de 35 puntos loto en Nicaragua, además de ser distribuidores de productos de ferretería y la compra y venta de motos. Grupo Atenea cuenta con un sistema ya desarrollado para la administración de los puntos loto, sin embargo, este presenta limitaciones que no cumplen con las necesidades de la empresa. Debido a estas limitaciones, la empresa se ve obligada a recurrir a procesos manuales, lo que a menudo resulta en pérdidas significativas, además de que aumenta la posibilidad de cometer errores humanos, lo que puede impactar en la gestión y administración del negocio.

Los antecedentes en este ámbito incluyen el desarrollo de sistemas similares en otras empresas que administran juegos de azar, como casinos o loterías. Por ejemplo, el Sistema de Control de Casino (SCC) implementado por la empresa Casinos Austria International en sus operaciones en todo el mundo. Otro ejemplo es el sistema de gestión de loterías desarrollado por la empresa española INTEGRAL, que ofrece una solución completa para la gestión de juegos de azar y loterías. Es importante mencionar que estos antecedentes no son específicos de Nicaragua, pero pueden servir como referencia para el desarrollo de un sistema similar en los puntos loto de Grupo Atenea. Además, en la entrevista realizada al propietario de la empresa Lic. Juan Anthony Gutiérrez, expresaba que existen otros sistemas y herramientas disponibles en el mercado los cuales no pueden ser adaptados a las necesidades específicas de la empresa debido a que son muy limitadas y no llenan los requerimientos esperados. Dos ejemplos son:

**El PuntoMóvil** es una aplicación móvil utilizada para la venta de rifas para la gestión y registro ventas, con numeración del 00-99.

**El sistema POS fuera de línea** es el software perfecto de POS (punto de venta) gratuito si tiene una tienda de comestibles, tienda minorista, cafetería, restaurante, bar, pizzería, panadería, cafetería, camión de comida, salón de belleza y spa, cualquier negocio de servicios, entre otros.

Este sistema de punto de venta le proporciona un excelente reemplazo para una caja registradora y realiza un seguimiento de las ventas y el inventario en tiempo real. Puede gestionar su venta y aumentarla con una buena gestión.

Ante las limitaciones presentadas por los recursos disponibles, surge la necesidad imperante de desarrollar un sistema web a medida que satisfaga todos los requisitos de la empresa. Asimismo, se requiere una aplicación móvil que facilite la realización de las operaciones más comunes y esenciales de este negocio, alimentando de manera sincrónica al sistema web. Esta solución integral garantizará una gestión eficiente y sin contratiempos de los puntos de venta de Grupo Atenea.

### III. JUSTIFICACIÓN

La justificación de este proyecto surge de la necesidad de mejorar la eficiencia y el control en los puntos loto, administrados por la empresa Grupo Atenea. En la actualidad, aunque Grupo Atenea cuenta con un sistema para el registro de ventas y el control de ingresos y egresos, su funcionamiento presenta limitaciones y fallas frecuentes, lo que obliga a recurrir a la realización de estos procesos de forma manual y escrita. Esto genera un elevado margen de error y pérdida de tiempo valioso.

Además, no existe un registro detallado de las ventas diarias, lo que dificulta la toma de decisiones y la realización de predicciones a medida del histórico de cada día. Esto se traduce en una falta de control y de organización en la administración del negocio, lo que a su vez afecta la eficiencia y la rentabilidad.

Por lo tanto, la implementación de un sistema web y una aplicación móvil para el control de los puestos loto en Nicaragua permitirá optimizar el tiempo y mejorar el control de ingresos y egresos, así como la administración del negocio. Asimismo, la posibilidad de realizar predicciones a medida del histórico de cada día permitirá una toma de decisiones más acertada. En este sentido, el desarrollo de un sistema web y una aplicación móvil para los puntos loto de Nicaragua es una solución práctica e innovadora que permitirá mejorar la eficiencia y el control en la administración de estos negocios. Por lo tanto, se justifica la realización de este proyecto para brindar una solución a la problemática existente y mejorar el rendimiento y la rentabilidad de los negocios de Grupo Atenea.

## **IV. OBJETIVOS**

### **4.1 Objetivo General**

- Desarrollar el sistema web y una aplicación móvil para el control de los puntos de venta loto para la empresa Grupo Atenea.

### **4.2 Objetivos Específicos**

- Identificar el alcance, requerimientos funcionales y no funcionales del sistema web para el control de los puntos Loto administrados por la empresa Grupo Atenea.
- Diseñar un modelo seleccionado que cumpla con los requerimientos obtenidos del sistema web, usando la metodología de desarrollo en espiral y la herramienta case PlantUML.
- Codificar una app móvil y una web para gestionar puntos loto. Flutter para la app móvil y web, Node.js para el backend web con MariaDB.
- Implantar el sistema web para el control de los puntos loto en los equipos y puesto de venta de la empresa grupo Atenea, estableciendo las medidas de seguridad y protección de los datos en el sistema.

## **V. MARCO TEÓRICO**

### **5.1 Métodos ágiles de desarrollo de software**

“Representan un enfoque dinámico y adaptable en el proceso de creación de software, distinguiéndose por su carácter iterativo e incremental. En contraposición a los modelos tradicionales que siguen planes rígidos y predefinidos, los métodos ágiles fomentan la flexibilidad y la capacidad de respuesta a los cambios. Este enfoque implica que los equipos de desarrollo trabajen en ciclos cortos y sucesivos, conocidos como iteraciones, donde desarrollan, prueban y ajustan iterativamente partes del software. Este proceso iterativo permite la detección temprana de errores y la adaptación rápida a los requisitos cambiantes del proyecto. Además, los métodos ágiles fomentan una comunicación y colaboración constantes entre los miembros del equipo de desarrollo, los usuarios y otras partes interesadas, promoviendo así un entendimiento común de los objetivos del proyecto y facilitando la toma de decisiones informadas. En esencia, los métodos ágiles no solo buscan la entrega continua y rápida de software funcional, sino que también promueven una cultura de mejora continua, adaptabilidad y colaboración en todo el proceso de desarrollo” (Beck et al., 2001).

### **5.2 Sistemas de información para la gestión empresarial**

"Se definen como herramientas tecnológicas diseñadas para facilitar la recopilación, almacenamiento, procesamiento y análisis de datos relacionados con las operaciones y procesos de una organización. Estos sistemas, están destinados a apoyar la toma de decisiones a todos los niveles de la empresa, desde la alta dirección hasta el personal operativo. En lugar de operar de manera aislada, estos sistemas integran diferentes funciones empresariales, como contabilidad, finanzas, recursos humanos, producción, ventas y marketing, en una única plataforma centralizada. Esto permite una mayor eficiencia y coordinación entre los diversos departamentos de la organización. Además, los sistemas de información para la gestión empresarial ofrecen funcionalidades como la generación de informes en tiempo real, la automatización de procesos comerciales, la gestión de relaciones con clientes (CRM) y la planificación de recursos empresariales (ERP), lo que ayuda

a mejorar la productividad y la competitividad de la empresa en el mercado. Estos sistemas desempeñan un papel fundamental en la optimización de los procesos empresariales, la toma de decisiones estratégicas y la mejora del rendimiento organizacional en su conjunto. " (Laudon & Laudon, 2017).

### **5.3 Desarrollo de software**

"El desarrollo de software, se define como un proceso multifacético que abarca desde la concepción hasta la entrega y el mantenimiento de aplicaciones informáticas. Este proceso comienza con la planificación, donde se definen los objetivos del software y se establecen los requisitos del sistema. Luego, entra en la fase de diseño, donde se elabora la arquitectura del software y se detallan sus componentes y funcionalidades. La implementación es la etapa en la que se escribe el código, se desarrollan los algoritmos y se realizan pruebas unitarias para garantizar el funcionamiento adecuado del programa. Por último, el mantenimiento implica la corrección de errores, la implementación de mejoras y la adaptación a cambios en el entorno. Para llevar a cabo este proceso de desarrollo de software de manera efectiva, se requiere no solo habilidades en programación, sino también un sólido conocimiento técnico en áreas como la ingeniería de software, la gestión de proyectos y la calidad del software. Es un proceso complejo que demanda un enfoque meticuloso y una combinación de habilidades técnicas y prácticas para garantizar la entrega exitosa de aplicaciones informáticas funcionales y de alta calidad." (Sommerville, 2016).

### **5.4 Insomnia**

" Insomnia, una herramienta esencial en el desarrollo de software y la integración de sistemas, ofrece un entorno completo y eficiente para interactuar con APIs y servicios web. Permite a los desarrolladores enviar solicitudes HTTP, gestionar entornos y variables, realizar pruebas de extremo a extremo y colaborar en proyectos de manera efectiva. Con su capacidad para visualizar y manipular datos en tiempo real, Insomnia facilita la toma de decisiones informadas y el análisis de información durante el proceso de desarrollo. En este contexto, Insomnia ha transformado la forma en que los equipos de desarrollo trabajan y se comunican,

desempeñando un papel crucial en la optimización de procesos y el intercambio de conocimientos en el ámbito del desarrollo de software." (Insomnia, 2024).

### **5.5 Diseño de interfaz de usuario (UI)**

"Representa un proceso esencial en el desarrollo de aplicaciones, puesto que se enfoca en la creación de la experiencia que los usuarios tienen al interactuar con el software. Este proceso implica no solo la disposición visual de elementos en la pantalla, sino también la creación de una experiencia intuitiva y satisfactoria para el usuario. Una interfaz efectiva debe ser fácil de usar, con una navegación clara y comprensible, y debe presentar información de manera ordenada y accesible. Además, el diseño de la interfaz de usuario busca también generar una impresión visual atractiva, utilizando principios de diseño como la coherencia, la legibilidad y la estética. Es importante destacar que una interfaz bien diseñada no solo mejora la experiencia del usuario, sino que también puede influir significativamente en la aceptación y el éxito de una aplicación. Por lo tanto, el diseño de interfaz de usuario no solo se trata de estética, sino de crear una experiencia completa y satisfactoria que impulse la adopción y el uso continuo del software por parte de los usuarios." (Norman & Nielsen, 2010).

### **5.6 Responsive web design**

"El diseño web adaptable, también conocido como Responsive Web Design (RWD), constituye una metodología esencial en el desarrollo de sitios web modernos. Esta técnica se centra en crear interfaces web que se ajusten de manera dinámica y fluida a diferentes tamaños de pantalla, desde dispositivos móviles hasta computadoras de escritorio. El objetivo principal del diseño web adaptable es garantizar una experiencia de navegación óptima y consistente para los usuarios, independientemente del dispositivo que estén utilizando. Para lograr esto, se emplean diversas técnicas de diseño y programación, como el uso de rejillas flexibles, imágenes y contenido adaptable, y consultas de medios (media queries) para adaptar el diseño y la disposición de los elementos en función del tamaño de la pantalla del dispositivo. En un mundo donde el acceso a internet se realiza cada vez más a través de dispositivos móviles, el diseño web adaptable se ha convertido

en una práctica fundamental para asegurar que los sitios web sean accesibles y funcionales para una amplia gama de usuarios” (W3C, 2018).

### 5.7 El modelo en espiral

“El modelo en espiral es un enfoque de desarrollo de software que combina modelos secuenciales con una filosofía iterativa y orientada a riesgos. El desarrollo avanza en ciclos iterativos, cada uno compuesto por actividades de planificación, análisis de riesgos, desarrollo, evaluación y revisión. Su característica distintiva es la gestión proactiva de riesgos desde las primeras etapas del proyecto. Este enfoque iterativo permite flexibilidad y adaptabilidad, ajustándose a medida que se recopila más información y se abordan los riesgos identificados. Al integrar elementos de modelos secuenciales y basados en prototipos, el modelo en espiral se adapta a una variedad de proyectos, gestionando eficazmente la complejidad del desarrollo de software.” (Boehm, 1986).

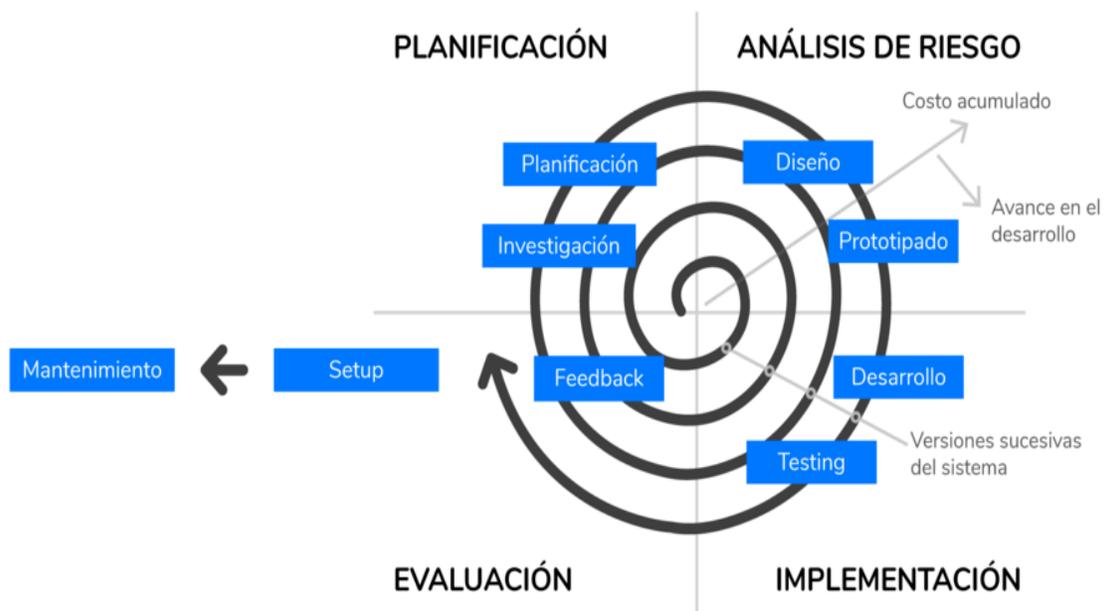


Imagen 1. Modelo en Espiral (Quizizz. s. f.).

### 5.8 Framework

"Es una estructura + software que sirve como base fundamental para el desarrollo de aplicaciones en diversos contextos. Este proporciona un conjunto de herramientas, bibliotecas y patrones de diseño que facilitan a los desarrolladores la creación de aplicaciones de manera más eficiente y rápida. Por lo general, un

framework se centra en un área específica, como el desarrollo web, y puede incluir una arquitectura predefinida junto con una serie de convenciones para estandarizar y agilizar el proceso de desarrollo. Esto permite a los desarrolladores enfocarse en la lógica de negocio de la aplicación, aprovechando las funcionalidades preexistentes del framework para tareas comunes y repetitivas. En esencia, los frameworks proporcionan una base sólida y coherente que promueve la consistencia y la eficiencia en el desarrollo de software." (Wikipedia, 2023).

### **5.9 MariaDB**

"Es un sistema de gestión de bases de datos de código abierto, basado en la tecnología de MySQL. Reconocido por su escalabilidad, rendimiento y seguridad, MariaDB se ha consolidado como una alternativa atractiva a las bases de datos comerciales costosas. Su compatibilidad con diversos lenguajes de programación facilita su integración en distintos entornos de desarrollo. La plataforma ofrece una arquitectura sólida capaz de manejar grandes volúmenes de datos de manera eficiente, garantizando la integridad y confidencialidad de la información almacenada. MariaDB emerge como una solución versátil y económica para satisfacer las necesidades de gestión de datos de organizaciones de todos los tamaños." (MariaDB Foundation, 2023).

### **5.10 Node.js**

"Es una plataforma de desarrollo de aplicaciones web que se fundamenta en el motor JavaScript V8 de Google. Este entorno se destaca por su capacidad para construir aplicaciones escalables y en tiempo real, siendo especialmente útil para aquellas que demandan numerosas interacciones en tiempo real con el servidor. Al aprovechar el lenguaje JavaScript tanto en el lado del cliente como del servidor, Node.js permite a los desarrolladores crear aplicaciones web de manera más coherente y eficiente. Su enfoque en el modelo de operaciones de entrada y salida sin bloqueo (non-blocking I/O) le confiere un rendimiento óptimo, lo que lo convierte en una opción atractiva para aplicaciones que manejan un gran volumen de solicitudes concurrentes. Node.js emerge como una plataforma versátil y potente para el desarrollo de aplicaciones web modernas, particularmente aquellas que

requieren una capacidad de escalamiento y una respuesta en tiempo real.” (Node.js, 2021).

### **5.11 NestJS**

“Es un framework altamente modular y eficiente basado en Node.js que brinda a los desarrolladores la capacidad de construir rápidamente aplicaciones empresariales sólidas y escalables. La combinación de características como TypeScript, arquitectura MVC (Modelo-Vista-Controlador), inyección de dependencias y soporte para bases de datos, entre otras, lo convierten en una herramienta poderosa y versátil para el desarrollo de aplicaciones empresariales. Al adoptar TypeScript, NestJS proporciona beneficios como la tipificación estática y un sistema de módulos robusto, lo que facilita la creación y mantenimiento de código limpio y organizado. La arquitectura MVC de NestJS promueve una separación clara de preocupaciones y facilita la escalabilidad y la reutilización del código. Además, la inyección de dependencias permite una gestión eficiente de las dependencias y la creación de componentes fácilmente intercambiables. NestJS emerge como un framework integral que ofrece todas las herramientas necesarias para el desarrollo rápido y eficiente de aplicaciones empresariales sólidas y escalables.” (Burke, B, 2018).

### **5.12 Arquitectura REST**

“Es un enfoque de diseño de software utilizado para desarrollar aplicaciones web escalables y distribuidas. Este modelo se basa en los principios fundamentales de la World Wide Web y utiliza los verbos estándar de HTTP, como GET, POST y DELETE, para acceder y manipular los datos en una aplicación web. La arquitectura REST promueve la construcción de sistemas que sean fáciles de entender, escalables y altamente interoperables. Al adoptar una arquitectura orientada a recursos, REST permite a los desarrolladores crear APIs (Interfaces de Programación de Aplicaciones) que son intuitivas y coherentes. Además, al aprovechar las características inherentes de HTTP, como la caché y la seguridad, la arquitectura REST facilita la creación de aplicaciones web robustas y de alto rendimiento. La arquitectura REST representa un enfoque poderoso y flexible para

el diseño de sistemas distribuidos en la web, promoviendo la interoperabilidad, la escalabilidad y la simplicidad en el desarrollo de aplicaciones” (Fielding, 2000).

### **5.13 Flutter**

“Es un framework de desarrollo de aplicaciones móviles respaldado por Google y basado en el lenguaje de programación Dart, ha ganado prominencia en la comunidad de desarrollo debido a su innovadora aproximación y sus destacadas características. Este framework destaca por su capacidad para crear interfaces de usuario rápidas y fluidas, lo que se logra a través de su enfoque en widgets personalizables y reutilizables que permiten a los desarrolladores diseñar interfaces altamente personalizadas y visualmente atractivas. La fuerza de Flutter radica también en su rendimiento excepcional, impulsado por su propio motor de renderizado, lo que garantiza una experiencia de usuario consistente y receptiva en una variedad de dispositivos y plataformas. Además, Flutter ofrece la ventaja de desarrollar aplicaciones móviles multiplataforma con un solo código base, lo que simplifica significativamente el proceso de desarrollo y reduce los costos asociados. Con estas características y su creciente comunidad de desarrolladores, Flutter continúa posicionándose como una opción líder para la construcción de aplicaciones móviles modernas y de alta calidad” (Flutter, 2021).

### **5.14 Integración continua**

“Es una práctica fundamental en el desarrollo de software que se centra en la automatización de los procesos de compilación, pruebas y despliegue de aplicaciones. Esta metodología permite a los equipos de desarrollo detectar y corregir errores de manera temprana en el ciclo de desarrollo, lo que conduce a una reducción significativa en el tiempo de entrega y una mejora en la calidad del software final. Al automatizar estos procesos, la integración continua fomenta una entrega más rápida y frecuente de nuevas funcionalidades, lo que a su vez aumenta la capacidad de adaptación del software a los cambios en los requisitos del usuario o del mercado. Además, al garantizar la estabilidad del código a lo largo del desarrollo, la integración continua promueve un ambiente de desarrollo más colaborativo y ágil, donde los desarrolladores pueden trabajar de manera eficiente

y segura en equipo. La integración continua se ha convertido en una práctica esencial en el desarrollo de software moderno, permitiendo a los equipos desarrollar y desplegar software de alta calidad de manera más rápida, eficiente y confiable” (Fowler & North, 2006).

### **5.15 UX design**

“El diseño de experiencia de usuario (UX design), se posiciona como una disciplina esencial en el desarrollo de sistemas web responsive, enfocándose en la creación de interfaces de usuario que brinden una experiencia intuitiva y fácil de usar, adaptándose a diversos tamaños de pantalla. Este enfoque se centra en comprender las necesidades y expectativas de los usuarios, así como en diseñar interfaces que optimicen la usabilidad y la accesibilidad. Al integrar principios de diseño centrados en el usuario, el UX design busca garantizar que las interacciones entre el usuario y el sistema sean eficientes, satisfactorias y significativas. Además, en el contexto del desarrollo de sistemas web responsive, el UX design juega un papel crucial en la adaptación de la interfaz de usuario a diferentes dispositivos y resoluciones de pantalla, asegurando una experiencia consistente y atractiva en todas las plataformas. El UX design se erige como un elemento fundamental para el éxito de los sistemas web responsive, al garantizar que las interfaces de usuario sean diseñadas de manera efectiva para satisfacer las necesidades y expectativas del usuario final” (Norman & Nielsen, 2010).

### **5.16 PlantUML**

“Es una herramienta de diagramación en línea que permite a los usuarios crear una variedad de diagramas, incluyendo diagramas UML, diagramas de flujo y diagramas de entidad-relación, entre otros. Lo que distingue a PlantUML es su enfoque basado en texto, donde los diagramas se describen utilizando un lenguaje de marcado sencillo y legible. Esto facilita la creación rápida y precisa de diagramas, sin la necesidad de interactuar directamente con una interfaz gráfica. PlantUML ofrece una solución flexible tanto para usuarios individuales como para equipos, y su enfoque basado en texto fomenta la colaboración y la gestión del código fuente de los diagramas de manera eficiente. Además, PlantUML se integra fácilmente con

otras herramientas y servicios, ampliando su utilidad y versatilidad en una variedad de contextos de trabajo y proyectos. En resumen, PlantUML se presenta como una herramienta valiosa y accesible para la creación y colaboración en diagramas en línea, ofreciendo una combinación de funcionalidades avanzadas y facilidad de uso que la hace atractiva para una amplia gama de usuarios.” (PlantUML, 2024)

### **5.17 Aplicación móvil**

“Una aplicación móvil es un tipo de software diseñado específicamente para dispositivos móviles como teléfonos inteligentes y tabletas, que ofrece una amplia gama de funcionalidades y servicios. Estas aplicaciones pueden incluir desde juegos y entretenimiento hasta herramientas de productividad, redes sociales y aplicaciones empresariales. Se descargan e instalan directamente en el dispositivo móvil a través de tiendas de aplicaciones como App Store para dispositivos iOS y Google Play Store para dispositivos Android. Su popularidad y utilidad han crecido significativamente en los últimos años, convirtiéndose en una parte integral de la vida cotidiana de millones de usuarios en todo el mundo” (Statista, 2022).

### **5.18 Sistema web**

“Un sistema web es una aplicación basada en la web que proporciona funcionalidades y servicios a través de internet. Estos sistemas se acceden a través de un navegador web y pueden incluir una amplia variedad de características, como procesamiento de datos, almacenamiento de información, interacción con bases de datos, y colaboración en tiempo real. Los sistemas web pueden ser desde simples sitios estáticos hasta complejas aplicaciones web interactivas. Su diseño y desarrollo requieren un enfoque integral que abarca aspectos como la arquitectura de software, la experiencia de usuario y la seguridad en línea. Con el crecimiento continuo de la conectividad a internet, los sistemas web se han vuelto indispensables en numerosos ámbitos, incluyendo el comercio electrónico, la educación en línea, la gestión de contenidos y la administración de empresas” (Mozilla Developer Network, 2022).

### **5.19 Seguridad en Aplicaciones Móviles y Web**

“La seguridad en aplicaciones móviles y web constituye un pilar fundamental para salvaguardar la información confidencial de los usuarios y garantizar la robustez del sistema en su totalidad. Este aspecto crucial abarca una amplia gama de medidas y prácticas destinadas a proteger los datos sensibles y la integridad del sistema. Entre estas se incluyen la implementación de sólidas políticas de autenticación de usuarios, el establecimiento de rigurosos controles de autorización de acceso, la adopción de medidas preventivas contra amenazas cibernéticas tales como la inyección de SQL y XSS, así como la aplicación de técnicas avanzadas de cifrado de datos para preservar su confidencialidad. Asimismo, es imprescindible cumplir con normativas de privacidad y regulaciones como el GDPR, garantizando así el adecuado manejo y protección de la información personal de los usuarios. Además, se hace indispensable asegurar la seguridad en la transmisión de datos mediante la implementación de protocolos seguros como HTTPS, que encriptan la información durante su transporte, fortaleciendo así la defensa contra posibles vulnerabilidades y ataques externos” (Mozilla Developer Network, 2022).

### **5.20 Digital Ocean**

"Digital Ocean es una plataforma en la nube que ofrece servicios de infraestructura como servicio (IaaS) para el despliegue y gestión de aplicaciones en línea. Esta plataforma proporciona una amplia gama de servicios, incluyendo servidores virtuales (droplets), almacenamiento en la nube, redes, bases de datos y herramientas de desarrollo. Los usuarios pueden acceder a los servicios de Digital Ocean a través de su panel de control basado en web, lo que facilita la configuración y administración de recursos en la nube. Digital Ocean es conocido por su simplicidad, fiabilidad y escalabilidad, lo que lo convierte en una opción popular para empresas y desarrolladores que buscan implementar y escalar aplicaciones en la nube de manera eficiente y rentable" (Digital Ocean, 2024).

### **5.21 Servidor Debian Linux**

"Debian es una distribución de sistema operativo basada en Linux ampliamente utilizada en entornos de servidor. Conocida por su estabilidad,

seguridad y amplio soporte de software, Debian es una opción popular para la implementación de servidores en una variedad de aplicaciones. Su sistema de gestión de paquetes, APT (Advanced Package Tool), facilita la instalación y actualización de software en los servidores Debian. Además, la comunidad de usuarios activa y el compromiso con los principios del software libre hacen de Debian una opción atractiva para aquellos que buscan un sistema operativo confiable y de código abierto para sus necesidades de servidor" (Debian, 2024).

## **5.22 PM2**

"PM2 (Process Manager 2) es una herramienta de gestión de procesos para aplicaciones Node.js que facilita el despliegue, monitoreo y escalado de aplicaciones en entornos de producción. PM2 permite administrar múltiples procesos Node.js en un solo servidor, ofreciendo características como el reinicio automático en caso de fallo, balanceo de carga y gestión de logs. Además, PM2 incluye una interfaz de línea de comandos intuitiva y una interfaz de usuario basada en web para monitorear el estado de las aplicaciones en tiempo real. Con su amplia gama de características y su facilidad de uso, PM2 es una herramienta popular entre los desarrolladores de Node.js para gestionar sus aplicaciones en entornos de producción" (PM2, 2024).

## **5.23 Certificación SSL**

"Una Certificación SSL (Secure Sockets Layer) es un protocolo de seguridad estándar que se utiliza para establecer conexiones cifradas entre un navegador web y un servidor. Esta certificación garantiza que los datos transmitidos entre el navegador y el servidor están protegidos y no pueden ser interceptados por terceros. La implementación de un certificado SSL en un sitio web permite la encriptación de la información confidencial, como datos de inicio de sesión, información de pago y otra información sensible, lo que ayuda a proteger la privacidad y la seguridad de los usuarios. Además, los certificados SSL también contribuyen a establecer la confianza del usuario al mostrar un candado o un indicador de conexión segura en el navegador web. La obtención y configuración de

un certificado SSL es un paso importante para garantizar la seguridad y la integridad de un sitio web en línea" (Kaspersky, 2024).

#### **5.24 Apache HTTP Server Project**

"El Proyecto Apache HTTP Server es una iniciativa de desarrollo de software de código abierto que se centra en la creación y mantenimiento del servidor web Apache HTTP Server. Apache HTTP Server es uno de los servidores web más populares y ampliamente utilizados en el mundo, conocido por su estabilidad, flexibilidad y seguridad. Este servidor web es compatible con una variedad de sistemas operativos, incluyendo Linux, Unix, Windows y otros, y ofrece una amplia gama de características y módulos que permiten la personalización y extensión según las necesidades del usuario. La comunidad de desarrollo activa detrás del Proyecto Apache HTTP Server continúa mejorando y actualizando el software para garantizar su rendimiento óptimo y su compatibilidad con las últimas tecnologías web" (Apache HTTP Server Project, 2024).

#### **5.25 Socket.io**

"Socket.io es una biblioteca de JavaScript de código abierto que facilita la comunicación bidireccional en tiempo real entre clientes web y servidores. Esta biblioteca permite la creación de aplicaciones web en tiempo real, como chats en línea, juegos multijugador y aplicaciones de colaboración, mediante el uso de conexiones WebSocket o alternativas basadas en polling para admitir una amplia gama de navegadores y entornos. Socket.io simplifica el manejo de eventos y la transmisión de datos en tiempo real, lo que la convierte en una opción popular entre los desarrolladores para construir aplicaciones interactivas y en tiempo real en la web" (Socket.io, 2024).

#### **5.26 Swagger**

"Swagger es un conjunto de herramientas de código abierto que permite a los desarrolladores diseñar, construir, documentar y consumir servicios web RESTful de manera eficiente. Con Swagger, los desarrolladores pueden definir la estructura de sus API en un formato legible por humanos, como YAML o JSON, utilizando la especificación OpenAPI (anteriormente conocida como Swagger Specification).

Esta especificación proporciona un estándar para describir los puntos finales de la API, los parámetros de entrada, las respuestas esperadas y otros detalles importantes. Además, Swagger ofrece herramientas para generar automáticamente documentación interactiva, SDKs cliente y servidores de pruebas basados en la especificación OpenAPI, lo que simplifica el proceso de desarrollo y consumo de servicios web RESTful" (Swagger, 2024).

### **5.27 Clean Architecture**

“Las arquitecturas limpias, también conocidas como "Clean Architectures", son un conjunto de principios y patrones de diseño de software propuestos por el desarrollador de software Robert C. Martin (también conocido como "Uncle Bob"). Clean Architecture se presentó por primera vez por Robert C. Martin en una serie de conferencias y charlas en la década de 2010. Uno de los artículos más influyentes sobre el tema es su publicación "The Clean Architecture" (La Arquitectura Limpia), que fue publicado en el blog de Uncle Bob en 2012. Desde entonces, el concepto ha ganado popularidad y se ha convertido en un enfoque preferido para desarrolladores y equipos que buscan crear software más mantenible y escalable. La idea principal detrás de las arquitecturas limpias es separar las preocupaciones en diferentes capas bien definidas, con reglas estrictas sobre cómo deben interactuar entre sí” (Diego Coder, 2024)

### **5.28 Inyección de dependencias**

“La Inyección de Dependencias es un patrón de diseño de software utilizado en el desarrollo de aplicaciones que busca desacoplar componentes y promover el modularidad y la reutilización de código. Este patrón consiste en suministrar a un componente las dependencias que necesita para realizar su trabajo, en lugar de que el componente las cree directamente. Esto se logra pasando las dependencias como parámetros del constructor, métodos o a través de métodos de configuración. La Inyección de Dependencias facilita la creación de componentes más flexibles, testeables y mantenibles, ya que permite cambiar las dependencias sin necesidad de modificar el código del componente, lo que promueve la cohesión y la separación de preocupaciones en el diseño de software” (Martin, R. C. 2002).

### **5.29 Manejador de estado (MobX)**

El Manejador de Estado (MobX) es una biblioteca de gestión de estado para aplicaciones web y móviles escritas en JavaScript o TypeScript. MobX proporciona una forma sencilla y eficiente de mantener el estado de la aplicación sincronizado con la interfaz de usuario y los datos en tiempo real. Utiliza un modelo de programación reactivo basado en observables, lo que significa que los componentes de la aplicación pueden observar los cambios en el estado y reaccionar automáticamente a ellos, actualizando la interfaz de usuario según sea necesario. MobX se integra fácilmente con frameworks populares como React y Angular, y ofrece características como la gestión automática de la derivación de datos, la capacidad de realizar cambios mutables en el estado de forma segura y la optimización del rendimiento a través de la minimización de las actualizaciones innecesarias de la interfaz de usuario. (MobX, 2024)

### **5.30 DTO**

El Objeto de Transferencia de Datos (DTO, por sus siglas en inglés) es un patrón de diseño utilizado en el desarrollo de software para facilitar la transferencia de datos entre subsistemas de una aplicación. Un DTO es una estructura de datos simple que encapsula múltiples atributos o campos relacionados, permitiendo la transferencia eficiente de información entre capas o componentes de un sistema sin acoplar directamente las clases de dominio con las capas de presentación o persistencia de datos. El DTO ayuda a evitar la exposición innecesaria de la estructura interna de los objetos de dominio y mejora la separación de preocupaciones en una arquitectura de software. Además, los DTO pueden ser serializados y deserializados fácilmente para su transporte a través de redes o almacenamiento en bases de datos, lo que los hace especialmente útiles en entornos distribuidos o sistemas que requieren interoperabilidad entre diferentes tecnologías o plataformas. (Gamma, 1994)

## VI. DISEÑO METODOLÓGICO

### 6.1 Modelo Espiral

Es una metodología de desarrollo de software que se enfoca en la flexibilidad y adaptabilidad a los cambios que puedan surgir en el proyecto. A diferencia de otras metodologías más rígidas, como el modelo en cascada, el modelo espiral permite que los equipos de desarrollo realicen iteraciones y ajustes continuos durante el proceso de desarrollo. En esta sección, se presentará un diseño metodológico para el desarrollo de un sistema web responsive para optimizar la eficiencia y control en los puntos loto, administrados por la empresa Grupo Atenea, utilizando el modelo espiral.

#### **Fase 1: Determinación de Objetivos y requerimientos**

En esta etapa fundamental de inicio, nos sumergiremos en una reunión estratégica con el representante de Grupo Atenea para no solo determinar los objetivos del proyecto, sino también para comprender a fondo las necesidades y expectativas de la empresa. Durante este encuentro, exploraremos en detalle los requisitos funcionales y no funcionales del sistema, asegurándonos de capturar todas las funcionalidades clave y considerar aspectos como la seguridad, la usabilidad y el rendimiento. Este proceso no solo nos permitirá identificar el alcance del proyecto, sino que también sentará las bases sólidas para un desarrollo efectivo y enfocado en las necesidades específicas de Grupo Atenea en la gestión de sus puntos de venta Loto.

#### **Fase 2: Análisis de riesgos**

En esta fase, se llevará a cabo un análisis de riesgos para identificar los posibles problemas que puedan surgir durante el proceso de desarrollo y se establecerá un plan de contingencia para abordar los riesgos más críticos. Además, se identificarán los requisitos que pueden cambiar a lo largo del proyecto. También se realizará una evaluación de impacto para entender cómo estos riesgos y cambios pueden afectar los plazos y el presupuesto. Se priorizarán las acciones correctivas y preventivas necesarias para mitigar cualquier impacto negativo en el desarrollo del sistema.

### **Fase 3: Desarrollo de prototipos**

En esta fase, se desarrollarán prototipos de la aplicación. Se va a diseñar un modelo seleccionado que cumpla con los requerimientos obtenidos del sistema web y los prototipos se utilizarán para obtener comentarios y opiniones de los usuarios sobre las características y funcionalidades del sistema. Los comentarios y opiniones de los usuarios serán utilizados para mejorar y adaptar el sistema de acuerdo con las necesidades y expectativas de los usuarios. Acá vamos a cumplir con uno de los objetivos planteados que fue diseñar un modelo seleccionado que cumpla con los requerimientos obtenidos del sistema web, usando la metodología de desarrollo en espiral y la herramienta case PlantUML. Posteriormente en esta misma fase comenzamos a codificar el sistema web para el control de los puntos loto usando el lenguaje de programación Dart utilizado para desarrollar aplicaciones con Flutter, Node.js y como gestor de bases de datos MariaDB. Para terminar, cumpliendo otro de los objetivos planteados.

### **Fase 4: Evaluación**

En esta fase, se llevará a cabo una evaluación de los prototipos desarrollados en la fase anterior. Se realizarán pruebas para garantizar la calidad y eficacia del sistema. Se llevarán a cabo pruebas de usabilidad, pruebas de rendimiento y pruebas de seguridad para garantizar que el sistema funcione correctamente.

### **Fase 5: Planificación**

En esta fase, se llevará a cabo la planificación del proyecto. Se elaborará un plan de trabajo que incluya las tareas a realizar, los plazos y los recursos necesarios para el desarrollo del sistema. Se elaborará un plan de comunicación para mantener a los miembros del equipo y los usuarios informados sobre el estado del proyecto.

### **Fase 6: Implementación**

En esta fase, se implementará el sistema desarrollado. Se realizarán pruebas adicionales para garantizar que el sistema funcione correctamente. Se realizarán pruebas de integración para garantizar que todas las partes del sistema funcionen juntas de manera efectiva. Y acá cumplimos uno de los últimos objetivos el cual era

implantar el sistema web para el control de los puntos loto en los equipos y puesto de venta de la empresa grupo Atenea.

### **Fase 7: Validación y Preparación del Sistema Implementado**

En esta fase, se llevará a cabo una evaluación final del sistema implementado. Se realizarán pruebas finales para garantizar que el sistema funcione correctamente. Se llevarán a cabo pruebas de usabilidad, pruebas de rendimiento y pruebas de seguridad para garantizar que el sistema funcione correctamente y como se menciona anteriormente se cumplen los últimos objetivos de nuestro proyecto que serían, establecer medidas de seguridad y protección de los datos en el sistema y evaluar la eficiencia y efectividad del sistema implementado en los puntos loto, administrados por la empresa Grupo Atenea.

### **Fase 8: Mantenimiento**

En esta fase, se llevará a cabo el mantenimiento del sistema. Se realizarán actualizaciones.

#### **6.1.1 Determinación de objetivos y requerimientos**

En esta etapa inicial del proyecto de desarrollo del sistema web y aplicación móvil para la gestión eficiente de puntos de venta de loto para Grupo Atenea, nos adentraremos en una fase crucial donde se establecerán los cimientos para el éxito del proyecto. En primer lugar, nos dedicaremos a definir de manera precisa y detallada los objetivos del proyecto, centrándonos en comprender a fondo las aspiraciones comerciales de Grupo Atenea y las necesidades específicas relacionadas con la gestión de sus puntos de venta de loto. Esto se llevará a cabo a través de una reunión estratégica con el representante clave de Grupo Atenea, donde se profundizará en los aspectos cruciales del negocio y se explorarán las posibles alternativas disponibles. Además, durante esta fase, se realizará un exhaustivo análisis para identificar tanto los requisitos funcionales como los no funcionales del sistema, abordando aspectos críticos como la gestión del inventario de boletos de loto, el seguimiento de ventas y ganancias. El objetivo principal de esta etapa es sentar una base sólida y comprensiva que guíe el desarrollo

subsiguiente del sistema, garantizando su alineación con las necesidades y expectativas de Grupo Atenea y su éxito a largo plazo.

### **6.1.2 Desarrollo de prototipos**

En esta etapa crucial del proyecto de desarrollo del sistema web y aplicación móvil para la gestión eficiente de puntos de venta loto para Grupo Atenea, nos dedicaremos a realizar un exhaustivo análisis de riesgos con el objetivo de anticipar y mitigar posibles problemas que puedan surgir durante el proceso de desarrollo. Se llevará a cabo una evaluación detallada de los riesgos potenciales, como posibles demoras en el desarrollo, cambios en los requisitos del cliente, problemas de compatibilidad entre plataformas y vulnerabilidades de seguridad. Una vez identificados estos riesgos, se elaborará un plan de contingencia sólido que incluya acciones específicas para abordar cada riesgo identificado y minimizar su impacto en el proyecto. Además, se establecerá un sistema de monitoreo continuo para supervisar la evolución de los riesgos a lo largo del tiempo y ajustar el plan de contingencia según sea necesario. Esta fase también se centrará en identificar requisitos que puedan cambiar a lo largo del proyecto, como nuevas funcionalidades solicitadas por el cliente o cambios en las prioridades comerciales. El objetivo principal de esta etapa es garantizar la preparación del equipo de desarrollo para hacer frente a cualquier desafío potencial y asegurar el éxito del proyecto en todas las etapas de su ejecución.

### **6.1.3 Análisis de riesgos**

En esta fase crucial del proyecto de desarrollo del sistema web y aplicación móvil para la gestión eficiente de puntos de venta loto para Grupo Atenea, nos enfocaremos en el desarrollo de prototipos de la aplicación que reflejen fielmente los requisitos obtenidos del sistema web. Utilizando la metodología de desarrollo en espiral y la herramienta Case PlantUML, se diseñará un modelo seleccionado que satisfaga las necesidades y expectativas de los usuarios finales. Estos prototipos serán sometidos a pruebas exhaustivas para obtener comentarios y opiniones de los usuarios sobre las características y funcionalidades del sistema. Los

comentarios recibidos serán cuidadosamente analizados y utilizados para mejorar y adaptar el sistema de acuerdo con las necesidades y expectativas de los usuarios.

Además, en esta misma fase, daremos inicio a la codificación del sistema web para el control de los puntos de venta loto, utilizando el lenguaje de programación Dart, ampliamente reconocido por su eficiencia y compatibilidad con herramientas de desarrollo como Flutter y Node.js. Además, se integrará el gestor de bases de datos MariaDB para garantizar un almacenamiento seguro y eficiente de los datos del sistema. Este enfoque nos permitirá cumplir con uno de los objetivos planteados, que es diseñar un modelo seleccionado que cumpla con los requisitos del sistema web, mientras avanzamos en el desarrollo y la implementación efectiva del sistema.

#### **6.1.4 Evaluación**

En esta etapa crucial del proyecto, nos enfocaremos en la evaluación exhaustiva de los prototipos desarrollados en la fase anterior. Se llevarán a cabo una serie de pruebas meticulosas para garantizar la calidad, eficacia y fiabilidad del sistema en su conjunto.

Las pruebas de usabilidad serán una prioridad, ya que nos permitirán evaluar la facilidad de uso y la experiencia del usuario en la interacción con el sistema. Se pondrá a prueba la navegación, la claridad de la interfaz y la accesibilidad de las funcionalidades clave para asegurar que el sistema sea intuitivo y fácil de usar para todos los usuarios finales.

Además, se llevarán a cabo pruebas de rendimiento para evaluar la velocidad, la eficiencia y la capacidad de respuesta del sistema bajo diferentes cargas de trabajo y condiciones. Esto nos permitirá identificar y abordar posibles cuellos de botella y optimizar el rendimiento del sistema para garantizar una experiencia fluida y sin interrupciones para los usuarios.

#### **6.1.5 Planificación**

En esta etapa crucial del proyecto, nos enfocaremos en la elaboración detallada de la planificación del mismo. Se desarrollará un plan de trabajo exhaustivo que incluirá todas las tareas necesarias para el desarrollo del sistema, así como los plazos

asociados y los recursos requeridos. Además, se diseñará un plan de comunicación integral para mantener a todos los miembros del equipo y a los usuarios informados sobre el progreso del proyecto, garantizando una colaboración efectiva y una transparencia en todas las etapas del proceso de desarrollo. Esta fase es fundamental para garantizar una ejecución eficiente y exitosa del proyecto, permitiendo una gestión efectiva de los recursos y una coordinación adecuada entre todos los involucrados.

### **6.1.6 Implementación**

En esta etapa crítica del proyecto, se llevará a cabo la implementación del sistema desarrollado con meticulosidad. Se realizarán pruebas adicionales para verificar la funcionalidad y estabilidad del sistema, incluyendo pruebas de integración para asegurar el correcto funcionamiento de todas las partes del sistema de manera cohesionada. Cumpliendo con uno de los últimos objetivos del proyecto, se procederá a implantar el sistema web para el control de los puntos de venta loto en los equipos y puestos de venta de la empresa Grupo Atenea. Esta fase representa un hito fundamental en el proyecto, marcando la transición del desarrollo a la puesta en marcha del sistema en un entorno operativo real.

### **6.1.7 Validación y Preparación del Sistema Implementado**

En el tramo final de este proyecto, nos adentramos en una fase de evaluación exhaustiva del sistema implementado para garantizar su pleno funcionamiento y cumplir con los últimos objetivos trazados. Durante esta etapa, se llevarán a cabo pruebas finales meticulosas, que abarcarán desde pruebas de usabilidad para asegurar una experiencia fluida para los usuarios, hasta pruebas de rendimiento destinadas a verificar la eficiencia del sistema bajo diversas condiciones de carga. Asimismo, se realizarán pruebas de seguridad para salvaguardar los datos y garantizar la integridad del sistema en su conjunto. Además de cumplir con estos criterios, esta fase culminará con la implementación de medidas de seguridad y protección de datos, así como la evaluación de la eficiencia y efectividad del sistema en los puntos de venta loto gestionados por Grupo Atenea. Con esta evaluación final, cerraremos con éxito este importante proyecto, asegurando que el sistema

esté completamente preparado para su despliegue y operación en un entorno operativo real.

### **6.1.8 Mantenimiento y Actualizaciones**

En esta etapa final del ciclo de vida del proyecto, nos adentramos en el período de mantenimiento continuo del sistema desarrollado. Durante esta fase, se llevará a cabo una serie de tareas destinadas a garantizar el funcionamiento óptimo y la fiabilidad del sistema a lo largo del tiempo. Esto incluirá la implementación de actualizaciones periódicas para corregir errores, mejorar la seguridad y agregar nuevas funcionalidades según sea necesario. El objetivo principal de esta fase es asegurar que el sistema siga siendo eficiente, seguro y adecuado para satisfacer las necesidades en constante evolución de los usuarios y del negocio.

## VII. RESULTADOS

### 7.1 Primera Etapa: Determinación de Objetivos y requerimientos

En la primera fase del proyecto de desarrollo de software para Grupo Atenea, nos centramos en recopilar los requisitos esenciales para avanzar. Este proceso fue guiado por la metodología en espiral, que nos permite una exploración iterativa y detallada de las necesidades del sistema. A través de entrevistas y reuniones con el propietario de la empresa Lic. Juan Anthony Gutiérrez, pudimos obtener una comprensión profunda de los procesos requeridos para el sistema. Ahora, estamos listos para presentar los resultados de nuestro análisis de requisitos, sentando así las bases sólidas para el desarrollo del proyecto.

#### 7.1.1 Requerimientos Funcionales del Sistema

Número	Requerimiento	Descripción
RF1	Login de usuario	Permite a los usuarios autenticarse en el sistema de gestión, asegurando que solo usuarios autorizados puedan acceder.
RF2	Alta de usuario	Crea nuevas cuentas de usuario en el sistema, permitiendo la incorporación de nuevos administradores y vendedores.
RF3	Editar usuario	Modifica la información de usuarios existentes en el sistema, manteniendo los datos actualizados y precisos.
RF4	Eliminar usuario	Elimina cuentas de usuario del sistema de gestión, garantizando la seguridad al remover accesos no autorizados.
RF5	Asignación y gestión de roles y permisos	Administra roles y permisos de los usuarios, asegurando que cada usuario tenga los accesos adecuados a sus funciones.
RF6	Alta de punto de venta	Registra nuevos puntos de venta en el sistema, expandiendo la red de distribución y venta de loto.

RF7	Editar punto de venta	Modifica detalles de puntos de venta existentes, asegurando que la información sea precisa y actualizada.
RF8	Eliminar punto de venta	Borra puntos de venta del sistema de gestión, manteniendo la base de datos limpia y segura.
RF9	Alta de vendedor	Añade nuevos vendedores al sistema, permitiendo que más personas puedan realizar ventas de loto por puntos específicos.
RF10	Editar vendedor	Actualiza información de vendedores registrados, manteniendo los datos precisos y relevantes.
RF11	Eliminar vendedor	Elimina registros de vendedores del sistema, asegurando que solo personal autorizado pueda realizar ventas.
RF12	Alta de juego	Introduce nuevos juegos al sistema de gestión, ampliando la oferta de loterías disponibles.
RF13	Editar juego	Modifica detalles de juegos existentes en el sistema, asegurando que la información de los juegos esté siempre actualizada.
RF14	Eliminar juego	Elimina juegos del sistema de gestión, retirando juegos que ya no están activos o disponibles.
RF15	Alta de sorteo	Crea nuevos sorteos en el sistema.
RF16	Editar sorteo	Modifica información de sorteos existentes, asegurando que los detalles de los sorteos sean precisos y actualizados.
RF17	Eliminar sorteo	Borra sorteos del sistema de gestión, retirando eventos que ya no son relevantes o necesarios.

RF18	Asignar techo a número	Establece límites de apuesta para números específicos, controlando el riesgo y la gestión financiera.
RF19	Editar techo de número	Modifica límites de apuesta existentes en el sistema, asegurando que los techos reflejen las políticas actuales.
RF20	Eliminar techo de número	Elimina límites de apuesta de números, permitiendo ajustes en la estrategia de gestión de riesgos.
RF21	Asignar número restringido	Marca números como no disponibles para apuestas, asegurando que ciertos números no puedan ser jugados.
RF22	Editar número restringido	Modifica números restringidos en el sistema, ajustando las restricciones según las necesidades del negocio.
RF23	Eliminar número restringido	Elimina números restringidos del sistema, permitiendo que estos números puedan ser jugados nuevamente.
RF24	Imprimir boleta	Genera e imprime boletas de apuestas, facilitando la entrega de comprobantes a los jugadores.
RF25	Reimprimir boleta	Reimprime boletas de apuestas previamente generadas, proporcionando duplicados cuando sea necesario.
RF26	Realizar venta de boleta	Procesa la venta de boletas de apuestas, registrando las transacciones en el sistema.
RF27	Registrar números ganadores	Documenta los números ganadores en el sistema, asegurando la transparencia y precisión en los resultados.

RF28	Agregar Factores a un monto	Asigna factores de pago a montos de apuesta, determinando cuánto se pagará a los ganadores.
RF29	Editar Factores a un monto	Modifica factores de pago existentes en el sistema, ajustando las políticas de pago según sea necesario.
RF30	Eliminar Factores a un monto	Elimina factores de pago del sistema, removiendo aquellos que ya no son aplicables.

Tabla 1: Requerimientos de sistema – Requerimientos Funcionales.

### 7.1.2 Resultados de los Requerimientos No Funcionales del Sistema

Además de los requisitos funcionales, se identificaron varios requisitos no funcionales:

Número	Requerimiento	Descripción
RNF1	Requerimientos del entorno	El sistema seguirá funcionando, aunque haya fallas eléctricas en las instalaciones.
RNF2	Vida Útil	El sistema funcionará de acá a 5 años sin requerir mantenimiento.
RNF3	Requerimientos de Adaptabilidad	La aplicación y el sistema deberán adaptarse a los cambios en el modelo de negocio.
RNF4	Facilidad de Uso	<ul style="list-style-type: none"> <li>• El sistema contará con mensajes de verificación por cada proceso que se haga.</li> <li>• El menú de la aplicación y del sistema será intuitivo para que cualquier persona lo pueda utilizar.</li> <li>• Los colores de la aplicación y del sistema tienen que ser iguales, según los requisitos del propietario.</li> </ul>

RNF5	Facilidad de Mantenimiento	El sistema se entregará al propietario de la empresa Lic. Juan Anthony Gutiérrez con un manual técnico y documentación relevante para mantenimiento.
RNF6	Soporte al Usuario	<ul style="list-style-type: none"> <li>• El sistema y la aplicación tendrán un Manual de Usuario para saber cómo funciona.</li> <li>• El sistema tendrá una sección de ayuda para poder descargar el manual de usuario.</li> </ul>
RNF7	Requerimientos de Desempeño	El sistema y la aplicación responderán con rapidez a las peticiones del usuario.
RNF8	Seguridad	Los permisos de acceso al sistema deben ser únicamente agregados o modificados por el administrador del sistema.
		Digital Ocean generará backups automáticos en nuestro droplet para que se realicen a intervalos regulares.

Tabla 2: Requerimientos de sistema – Requerimientos no Funcionales.

### 1.2.1 7.1.3 Resultados de los casos de uso.

En el desarrollo de la aplicación, se han realizado casos de uso para cubrir todos los escenarios y procesos posibles que los usuarios pueden enfrentar al usar las aplicaciones web y móvil. Estos casos de uso están diseñados con precisión para capturar las funcionalidades clave, asegurando que las necesidades del usuario final sean satisfechas. A través de estos casos de uso, hemos identificado y validado las interacciones críticas del usuario, tanto habituales como excepcionales, garantizando una experiencia coherente y eficiente. La documentación detallada de cada caso de uso proporciona una guía completa de las operaciones del sistema desde la perspectiva del usuario, ofreciendo información valiosa para el diseño y desarrollo continuo.

### 7.1.3.1 Gestionar techos

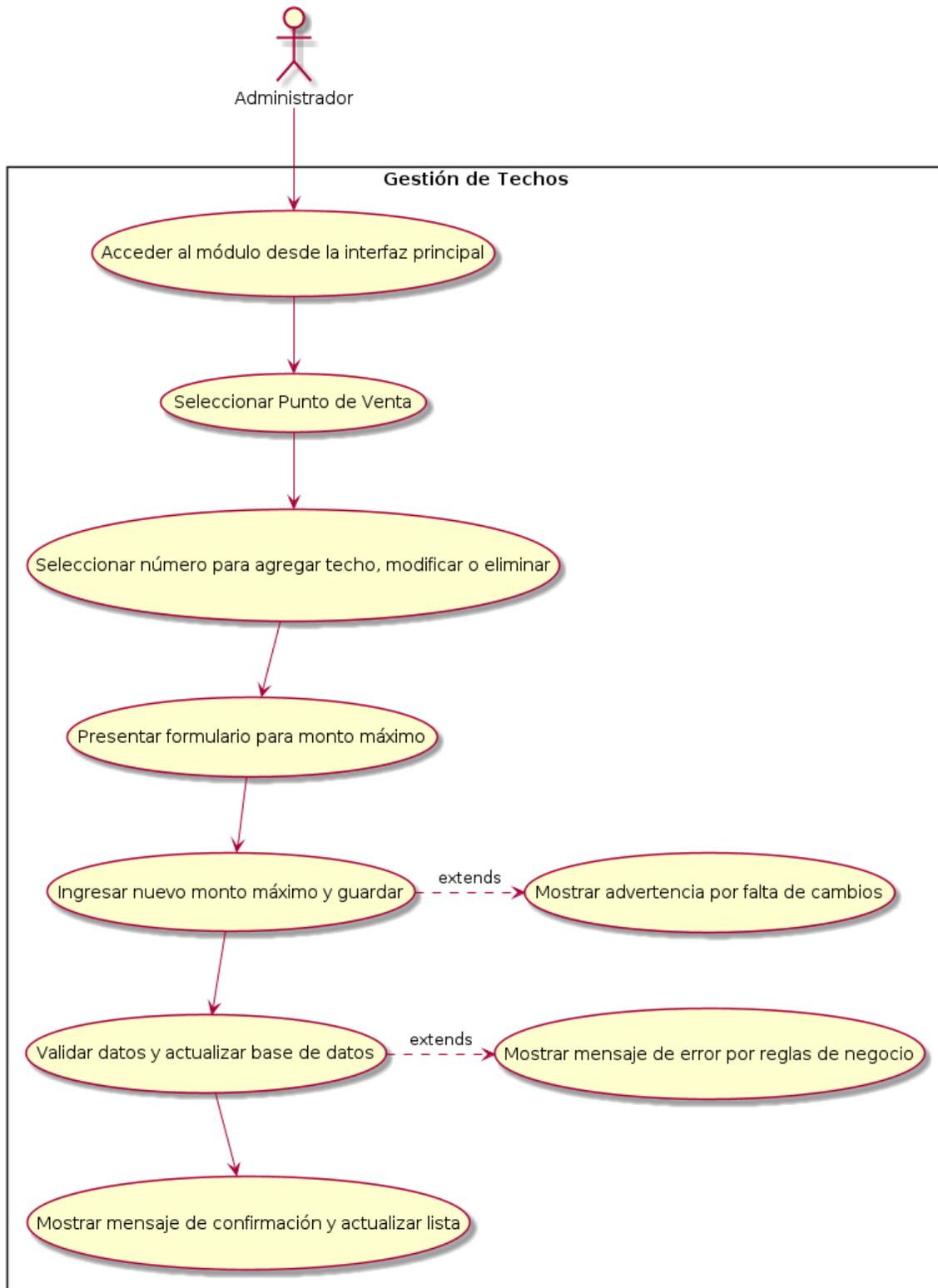


Diagrama 1: caso de uso – Gestionar techos

### 7.1.3.2 Autenticación y Gestión de Sesiones de Usuario

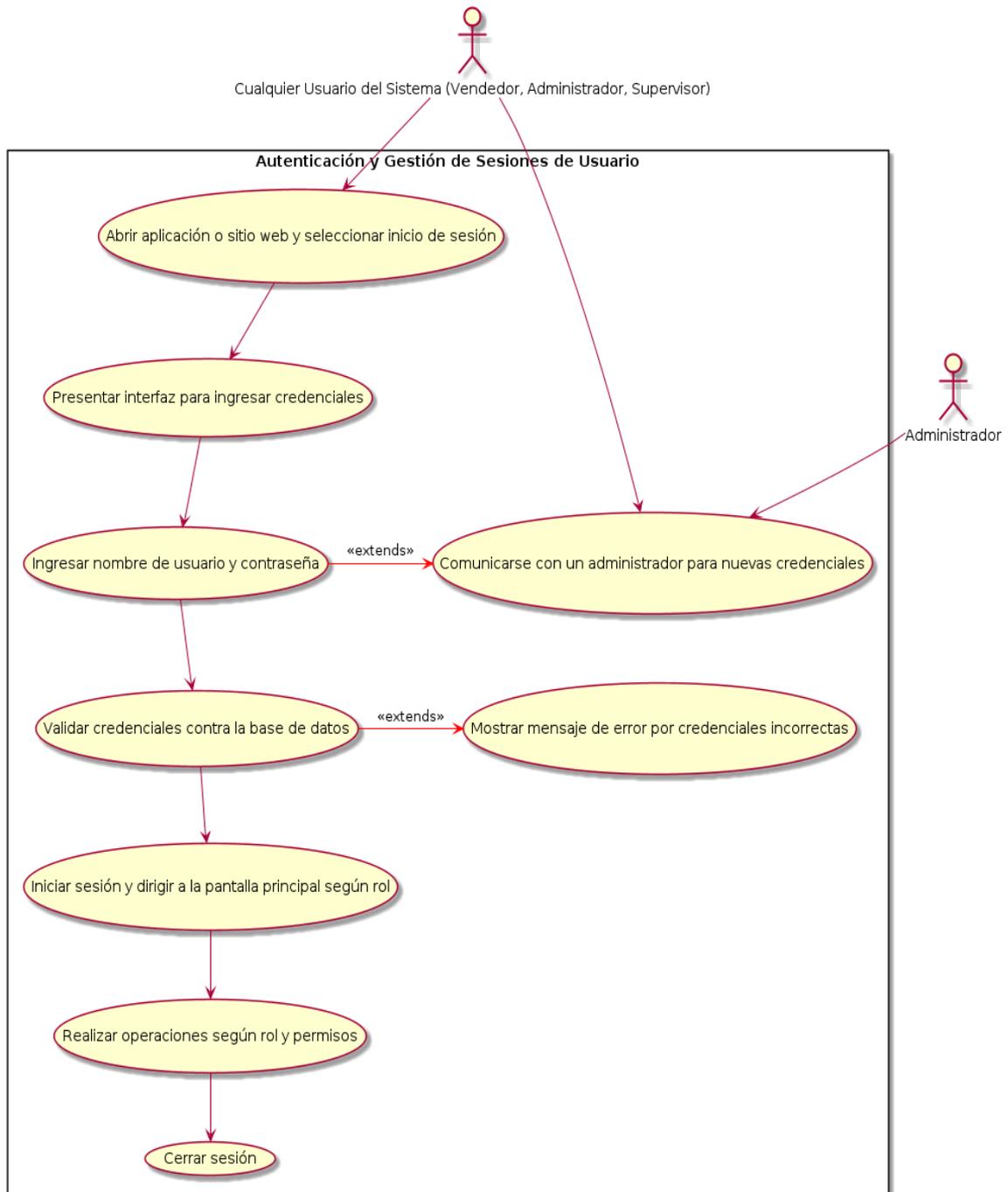


Diagrama 2: caso de uso – Autenticación y Gestión de Sesiones de Usuario

### 7.1.3.3 Gestión de Ventas de Boletas

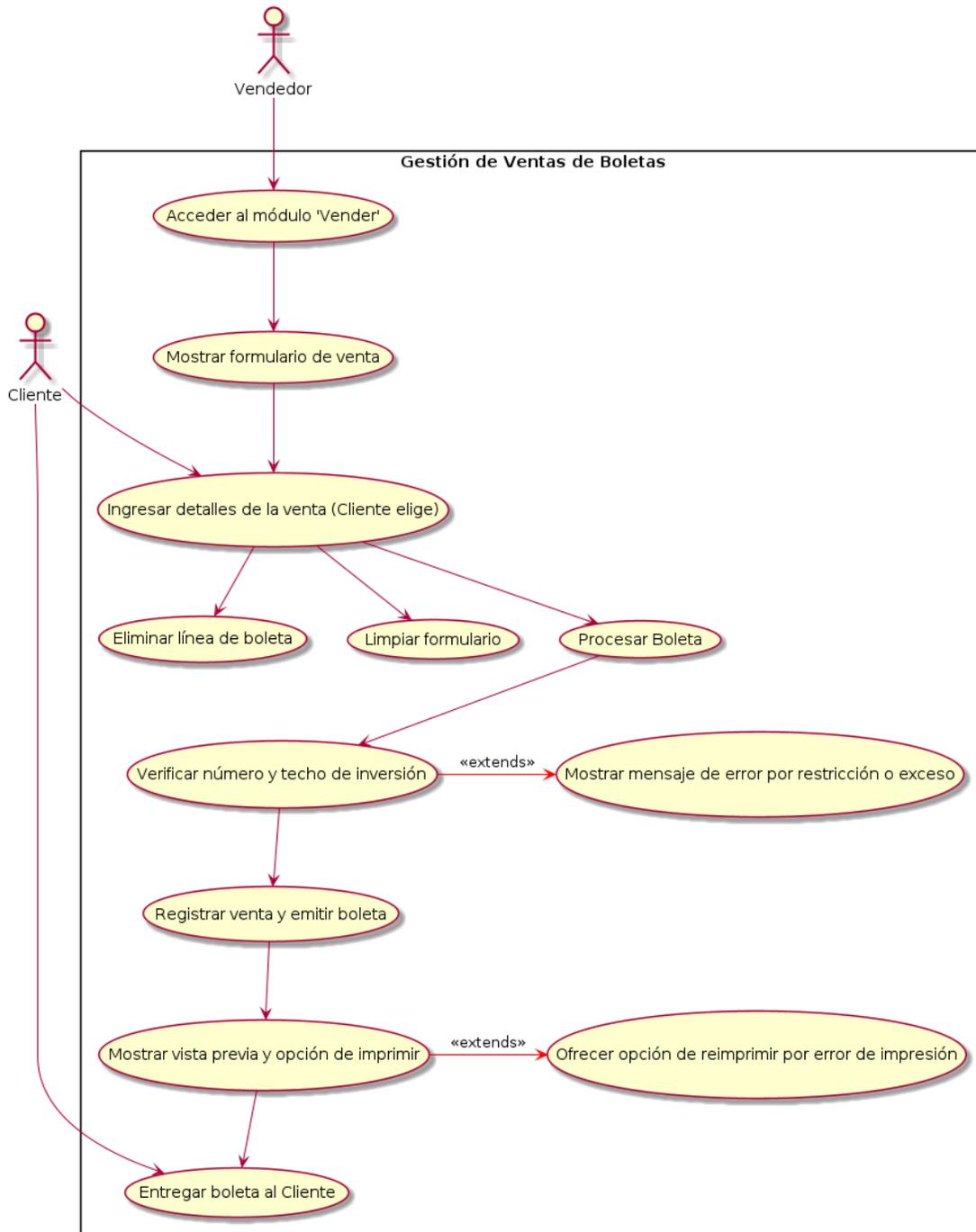


Diagrama 3: caso de uso – Gestión de Ventas de Boletas

### 7.1.3.4 Gestión de Factores

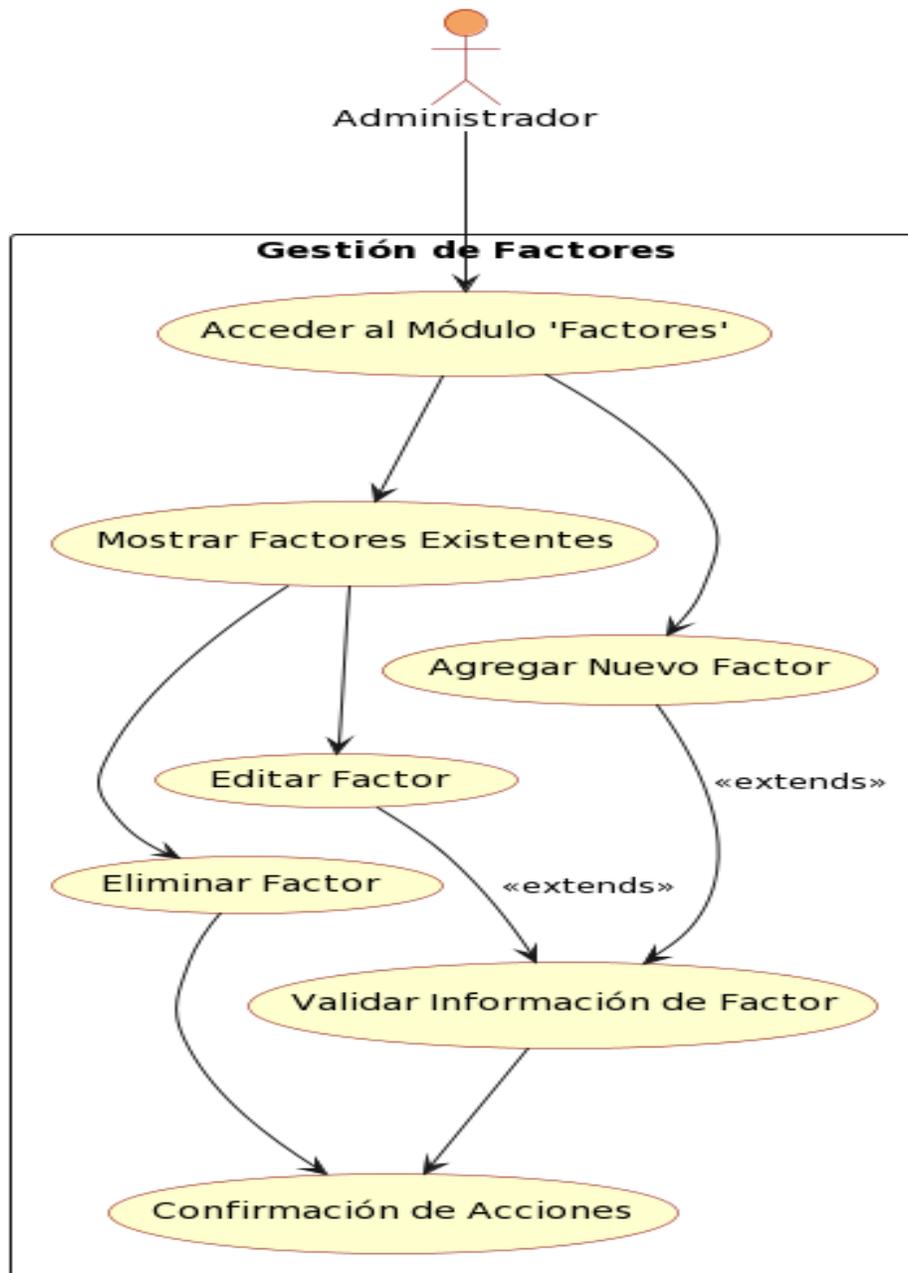


Diagrama 4: caso de uso – Gestión de Gestión de Factores

## 7.1.4 Detalle de casos de uso modelados.

### 7.1.4.1 Autenticación y Gestión de Sesiones de Usuario

#### 7.1.4.1.1 Login de usuario.

Caso de Uso	<b>Login de usuario</b>
Requerimientos Funcionales	RF01
Objetivo en Contexto	Permitir a los usuarios acceder a sus cuentas y habilitar el uso de funcionalidades específicas basadas en sus roles.
Precondiciones	El usuario debe tener una cuenta válida creada en el sistema.
Final Exitoso	El usuario ingresa al sistema y es dirigido a la página de inicio correspondiente a su rol.
Final Fallido	Mensaje de error por credenciales incorrectas o fallo en el proceso de autenticación.
Actores Principales	Administrador, Supervisor y Vendedor
Actores Secundarios	Sistema de gestión de bases de datos para la autenticación.
Evento de Inicio	El usuario selecciona la opción para iniciar sesión en la aplicación o sitio web.
Flujo Principal	<ol style="list-style-type: none"><li>1. El usuario abre la aplicación o sitio web y selecciona la opción para iniciar sesión.</li><li>2. El sistema muestra el formulario de Login.</li><li>3. El usuario introduce su nombre de usuario y contraseña.</li><li>4. El sistema verifica las credenciales contra la base de datos</li><li>5. Si las credenciales son correctas, el sistema concede acceso al usuario y muestra la página principal.</li><li>6. Si las credenciales son incorrectas, el sistema niega el acceso y muestra un mensaje de error.</li></ol>

	<p>7. El usuario tiene la opción de intentar iniciar sesión nuevamente.</p> <p>8. El caso de uso finaliza cuando el usuario cierra la sesión o cierra la aplicación/sitio web.</p>
--	--

Tabla 3: caso de uso – Login de usuario.

#### **7.1.4.1.2 Alta de usuario.**

Caso de Uso	<b>Alta de usuario</b>
Requerimientos Funcionales	RF02
Objetivo en Contexto	Crear una nueva cuenta de usuario en el sistema para permitir el acceso y la asignación de roles específicos.
Precondiciones	El administrador ha iniciado sesión en el sistema con permisos para crear usuarios.
Final Exitoso	Una nueva cuenta de usuario es creada exitosamente y está lista para ser utilizada.
Final Fallido	Mensaje de error debido a la entrada de datos incorrectos o a problemas técnicos que impiden la creación de la cuenta.
Actores Principales	Administrador
Actores Secundarios	Sistema de gestión de bases de datos para la creación de cuentas.
Evento de Inicio	El administrador decide crear una nueva cuenta de usuario.

Flujo Principal	<ol style="list-style-type: none"> <li>1. El administrador accede al módulo de gestión de usuarios.</li> <li>2. El administrador selecciona la opción para crear un nuevo usuario.</li> <li>3. El sistema presenta un formulario para ingresar la información del nuevo usuario.</li> <li>4. El administrador completa el formulario con los datos requeridos.</li> <li>5. El sistema valida los datos ingresados.</li> <li>6. Si los datos son válidos, el sistema registra el nuevo usuario en la base de datos.</li> <li>7. Se muestra un mensaje de confirmación al administrador de que la cuenta ha sido creada.</li> <li>8. El caso de uso finaliza con la creación exitosa de la cuenta.</li> </ol>
-----------------	---

Tabla 4: caso de uso – Alta de usuario

#### **7.1.4.1.3 Editar usuario.**

Caso de Uso	<b>Editar usuario</b>
Requerimientos Funcionales	RF03
Objetivo en Contexto	Actualizar los datos de una cuenta de usuario existente en el sistema.
Precondiciones	El administrador con permisos ha iniciado sesión y accede al perfil que desea editar.

Final Exitoso	Los datos de la cuenta de usuario se actualizan correctamente en el sistema.
Final Fallido	No se pueden actualizar los datos debido a información incorrecta o fallas en la conexión a la base de datos, y se muestra un mensaje de error.
Actores Principales	Administrador
Actores Secundarios	Sistema de gestión de bases de datos para la actualización de cuentas.
Evento de Inicio	El administrador con permisos decide editar una cuenta de usuario.
Flujo Principal	<ol style="list-style-type: none"> <li>1. El actor accede al módulo de gestión de usuarios.</li> <li>2. Selecciona la cuenta de usuario a editar.</li> <li>3. El sistema muestra el formulario con los datos actuales del usuario.</li> <li>4. El actor actualiza los datos necesarios y envía la información.</li> <li>5. El sistema valida y aplica los cambios.</li> <li>6. Se muestra un mensaje de confirmación de la actualización exitosa</li> <li>7. El caso de uso finaliza con la cuenta de usuario actualizada.</li> </ol>

Tabla 5: caso de uso – Editar usuario

#### **7.1.4.1.4 Eliminar usuario.**

Caso de Uso	<b>Eliminar usuario</b>
Requerimientos Funcionales	RF04
Objetivo en Contexto	Remover una cuenta de usuario existente del sistema para evitar el acceso futuro y eliminar su presencia en la base de datos.
Precondiciones	El administrador ha iniciado sesión con éxito y tiene los permisos necesarios para eliminar cuentas de usuario.

Final Exitoso	La cuenta de usuario es eliminada del sistema y no hay referencias futuras a ella.
Final Fallido	El sistema no puede eliminar la cuenta debido a problemas técnicos o porque la cuenta está vinculada a operaciones pendientes o históricas, mostrando un mensaje de error.
Actores Principales	Administrador
Actores Secundarios	Sistema de gestión de bases de datos para la eliminación de cuentas.
Evento de Inicio	El administrador decide eliminar una cuenta de usuario del sistema.
Flujo Principal	<ol style="list-style-type: none"> <li>1. El administrador navega al módulo de gestión de usuarios.</li> <li>2. Localiza la cuenta de usuario específica que necesita eliminar.</li> <li>3. El administrador selecciona la opción de eliminar la cuenta de usuario.</li> <li>4. El sistema solicita confirmación para proceder con la eliminación.</li> <li>5. Tras confirmar, el sistema procede a eliminar la cuenta y todas sus dependencias.</li> <li>6. Se muestra un mensaje que confirma la eliminación exitosa de la cuenta.</li> <li>7. El caso de uso concluye con la base de datos actualizada para reflejar la eliminación.</li> </ol>

Tabla 6: caso de uso – Eliminar usuario

#### **7.1.4.2 Gestionar techos.**

##### **7.1.4.2.1 Asignar techo a un número.**

Caso de Uso	<b>Asignar techo a un número</b>
Requerimientos Funcionales	RF18
Objetivo en Contexto	Establecer un límite máximo de apuesta que puede ser vendida para un número específico dentro de un juego determinado.

Precondiciones	El usuario ha iniciado sesión con éxito y tiene los permisos necesarios para modificar los techos de apuesta.
Final Exitoso	El techo asignado es guardado en el sistema, aplicándose a las ventas futuras del número específico en el juego seleccionado.
Final Fallido	Se muestra un mensaje de error si los datos son incorrectos, si el techo no cumple con las políticas del juego, o si hay un problema al guardar los datos en la base de datos.
Actores Principales	Administrador
Actores Secundarios	Sistema de base de datos para registro de techos
Evento de Inicio	El administrador elige asignar un nuevo techo a un número.
Flujo Principal	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción para gestionar techos desde la interfaz principal.</li> <li>2. El usuario ingresa al menú de configuración de techos.</li> <li>3. El usuario selecciona el juego y el número al cual desea asignar el techo.</li> <li>4. El usuario ingresa el valor del techo y lo somete para su registro.</li> <li>5. El sistema valida y registra el nuevo techo en la base de datos.</li> <li>6. Se muestra una confirmación del registro exitoso.</li> <li>7. El sistema actualiza la visualización de los techos con la nueva información.</li> <li>8. El caso de uso concluye con el techo establecido para el número.</li> </ol>

Tabla 7: caso de uso – Asignar techo a un numero

#### **7.1.4.2.2 Editar techo a un número.**

Caso de Uso	<b>Editar techo a un número</b>
Requerimientos Funcionales	RF19
Objetivo en Contexto	Modificar el límite de venta existente para un número específico dentro de un juego.

Precondiciones	El usuario ha iniciado sesión con éxito y tiene los permisos necesarios para editar los techos de apuesta.
Final Exitoso	El techo modificado es actualizado en el sistema y se aplica a las ventas futuras del número específico en el juego seleccionado.
Final Fallido	Se muestra un mensaje de error si los datos son incorrectos, si la modificación no cumple con las políticas del juego, o si hay un problema al actualizar los datos en la base de datos.
Actores Principales	Administrador
Actores Secundarios	Sistema de base de datos para registro de techos
Evento de Inicio	El administrador elige editar el techo de un número.
Flujo Principal	<ol style="list-style-type: none"> <li>1. El usuario accede al módulo de gestión de techos de apuesta.</li> <li>2. El usuario selecciona el juego y el número cuyo techo desea editar.</li> <li>3. El usuario modifica el valor del techo en el formulario proporcionado.</li> <li>4. El usuario envía la solicitud de actualización.</li> <li>5. El sistema valida y aplica los cambios en la base de datos.</li> <li>6. Se muestra una confirmación de la actualización exitosa.</li> <li>7. El sistema refleja el cambio en la interfaz de usuario.</li> <li>8. El caso de uso concluye con la actualización exitosa del techo.</li> </ol>

Tabla 8: caso de uso – Editar techo a un numero

#### **7.1.4.2.3 Eliminar techo a un número.**

Caso de Uso	<b>Eliminar techo a un número</b>
Requerimientos Funcionales	RF20

Objetivo en Contexto	Retirar el límite máximo de apuesta establecido para un número específico dentro de un juego.
Precondiciones	El usuario ha iniciado sesión con éxito y tiene los permisos necesarios para eliminar techos de apuesta.
Final Exitoso	El techo es eliminado del sistema, removiendo cualquier restricción de apuesta para el número específico en el juego seleccionado.
Final Fallido	Se muestra un mensaje de error si no se encuentra el techo, si los datos son incorrectos o si hay un problema al eliminar los datos en la base de datos.
Actores Principales	Administrador.
Actores Secundarios	Sistema de base de datos para registro de techos
Evento de Inicio	El actor decide eliminar el techo de un número.
Flujo Principal	<ol style="list-style-type: none"> <li>1. El usuario accede al módulo de gestión de techos de apuesta.</li> <li>2. El usuario localiza el número cuyo techo desea eliminar.</li> <li>3. El usuario selecciona la opción para eliminar el techo asignado.</li> <li>4. El sistema solicita confirmación antes de proceder con la eliminación.</li> <li>5. Tras la confirmación, el sistema elimina el techo de la base de datos.</li> <li>6. Se muestra una confirmación de la eliminación exitosa.</li> <li>7. El caso de uso concluye con la actualización de la interfaz de usuario para reflejar que el techo ha sido eliminado.</li> </ol>

Tabla 9: caso de uso – Eliminar techo a un numero

### 7.1.4.3 Gestión de ventas de boletas

#### 7.1.4.3.1 Realizar venta de boleta.

Caso de Uso	<b>Realizar venta de boleta</b>
Requerimientos Funcionales	RF26
Objetivo en Contexto	Procesar la transacción de una boleta incluyendo la validación contra números restringidos y cumplimiento de techos de inversión.
Precondiciones	El vendedor ha iniciado sesión con éxito y está operando desde un punto de venta autorizado. Debe tener acceso al módulo de ventas y la información del cliente y el juego debe estar disponible.
Final Exitoso	La boleta se vende, cumple con todas las restricciones y techos establecidos, y se entrega al cliente.
Final Fallido	La venta no se completa debido a la selección de un número restringido, exceder el techo de inversión permitido o errores en el sistema, y se informa al vendedor mediante un mensaje de error.
Actores Principales	Vendedor
Actores Secundarios	Cliente, Sistema de gestión de bases de datos
Evento de Inicio	El vendedor selecciona la función 'Vender' para iniciar el proceso de venta de una boleta.
Flujo Principal	<ol style="list-style-type: none"><li>1. El vendedor accede al módulo 'Vender' en el sistema.</li><li>2. El sistema muestra el formulario de venta.</li><li>3. El vendedor ingresa los detalles de la venta según la elección del cliente.</li><li>4. El vendedor procesa la boleta, lo cual incluye la verificación de techos de inversión y números restringidos.</li><li>5. Si la boleta cumple con las restricciones, el sistema registra la venta y emite la boleta.</li></ol>

	6. Se ofrece la boleta al cliente con una vista previa y la opción de imprimir. 7. La boleta se entrega al cliente, finalizando el proceso de venta.
--	---

Tabla 10: caso de uso – Realizar venta de boleta

#### 7.1.4.4 Gestión de Factores

##### 7.1.4.4.1 Asignar Factor a un monto

Caso de Uso	<b>Asignar Factor a un monto</b>
Requerimientos Funcionales	RF28
Objetivo en Contexto	Permitir al administrador asignar un factor de pago específico a un monto para una cantidad dentro de un juego y sorteo.
Precondiciones	El administrador ha iniciado sesión con éxito y tiene permisos para modificar los factores de pago.
Final Exitoso	Un nuevo factor de pago se asigna al monto especificado y queda registrado en el sistema para su utilización en cálculos de premios.
Final Fallido	La asignación del factor falla debido a una validación de datos incorrecta o problemas técnicos, y se muestra un mensaje de error al administrador.
Actores Principales	Administrador
Actores Secundarios	Sistema de gestión de base de datos
Evento de Inicio	El administrador selecciona la opción para asignar un nuevo factor a un monto en el módulo de gestión de factores.
Flujo Principal	<ol style="list-style-type: none"> <li>1. El administrador navega al módulo Factores</li> <li>2. Selecciona la opción para asignar un nuevo factor.</li> <li>3. Introduce el monto de la apuesta y el factor de pago correspondiente.</li> <li>4. El sistema valida la información y verifica la coherencia con las reglas del negocio.</li> </ol>

	<p>5. Si la validación es exitosa, el sistema guarda el nuevo factor en la base de datos.</p> <p>6. Se muestra una confirmación de que el factor ha sido asignado.</p> <p>7. El caso de uso finaliza satisfactoriamente.</p>
--	--

Tabla 11: caso de uso – Asignar Factor a un monto

#### **7.1.4.4.2 Editar Factor a un monto**

Caso de Uso	<b>Editar Factor a un monto</b>
Requerimientos Funcionales	RF29
Objetivo en Contexto	Facilitar al administrador la actualización de los factores de pago existentes para montos de cantidades específicas.
Precondiciones	El administrador debe haber iniciado sesión correctamente y tener acceso al módulo de factores. Debe existir al menos un factor de pago previamente asignado a un monto.
Final Exitoso	El factor de pago para un monto específico se actualiza correctamente en la base de datos y se refleja en el cálculo de pagos futuros.
Final Fallido	No se puede actualizar el factor debido a errores de validación de datos o problemas de conexión con la base de datos. Se notifica al administrador del fallo.
Actores Principales	Administrador
Actores Secundarios	Sistema de gestión de base de datos
Evento de Inicio	El administrador elige la opción de editar un factor de pago existente.
Flujo Principal	<ol style="list-style-type: none"> <li>1. El administrador accede al módulo de Factores.</li> <li>2. Busca y selecciona el factor de pago que desea editar.</li> <li>3. El administrador modifica los detalles del factor de pago.</li> </ol>

	<p>4. El sistema valida los cambios para asegurar que se adhieren a las políticas de pago.</p> <p>5. Una vez validado, el sistema actualiza el factor en la base de datos.</p> <p>6. El administrador recibe una confirmación de la actualización exitosa.</p> <p>7. Se concluye el proceso de edición.</p>
--	---

Tabla 12: caso de uso – Editar Factor a un monto

#### **7.1.4.4.3 Eliminar Factor a un monto**

Caso de Uso	<b>Editar Factor a un monto</b>
Requerimientos Funcionales	RF30
Objetivo en Contexto	Permitir al administrador eliminar un factor de pago asignado a un monto de una cantidad específica para evitar su uso en cálculos de pago futuros.
Precondiciones	El administrador debe estar autenticado y tener permisos necesarios para gestionar factores. Debe existir un factor de pago que esté asignado a un monto.
Final Exitoso	El factor de pago asignado al monto específico es eliminado con éxito de la base de datos y no se utilizará para cálculos futuros.
Final Fallido	La eliminación del factor falla debido a un error en la validación o en la base de datos, y se informa al administrador del error.
Actores Principales	Administrador
Actores Secundarios	Sistema de gestión de base de datos
Evento de Inicio	El administrador decide eliminar un factor de pago y selecciona la opción correspondiente en el sistema.

Flujo Principal	<ol style="list-style-type: none"> <li>1. El administrador ingresa al módulo de Factores.</li> <li>2. Localiza el factor de pago que desea eliminar y selecciona la opción de eliminación.</li> <li>3. Se solicita confirmación antes de proceder con la eliminación para prevenir acciones no intencionadas.</li> <li>4. Una vez confirmado, el sistema elimina el factor de la base de datos.</li> <li>5. Se muestra al administrador una notificación sobre la eliminación exitosa del factor.</li> <li>6. Fin del caso de uso tras la eliminación exitosa.</li> </ol>
-----------------	---

Tabla 13: caso de uso – Eliminar Factor a un monto

## 7.1.5 Resultados de los Diagramas de Actividades.

### 7.1.5.1 Gestionar techos

#### 7.1.5.1.1 Asignar techo a un numero

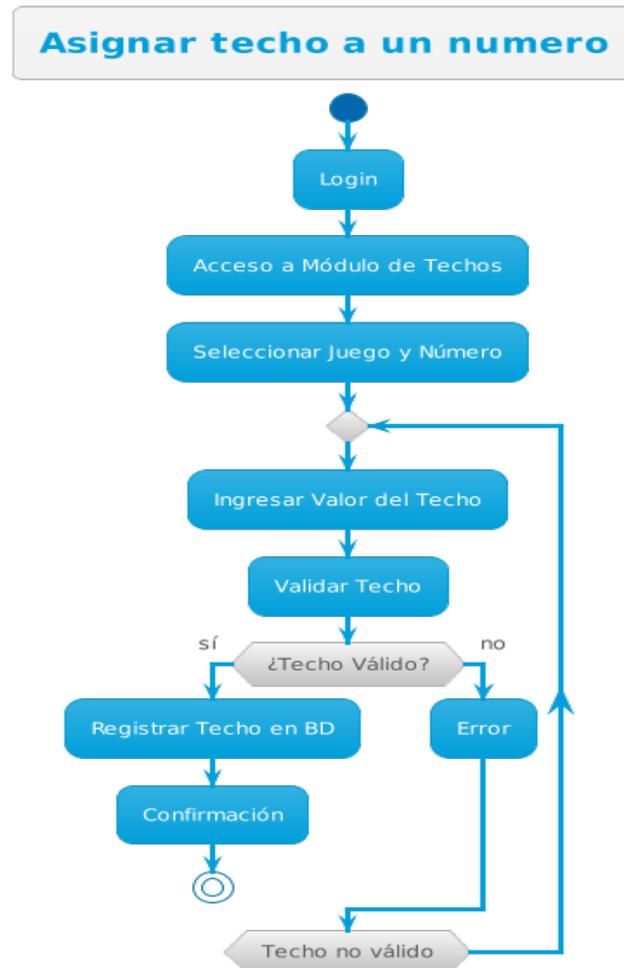


Diagrama 5 Diagrama de actividades – Asignar techo a un número

### 7.1.5.1.2 Editar techo a un numero

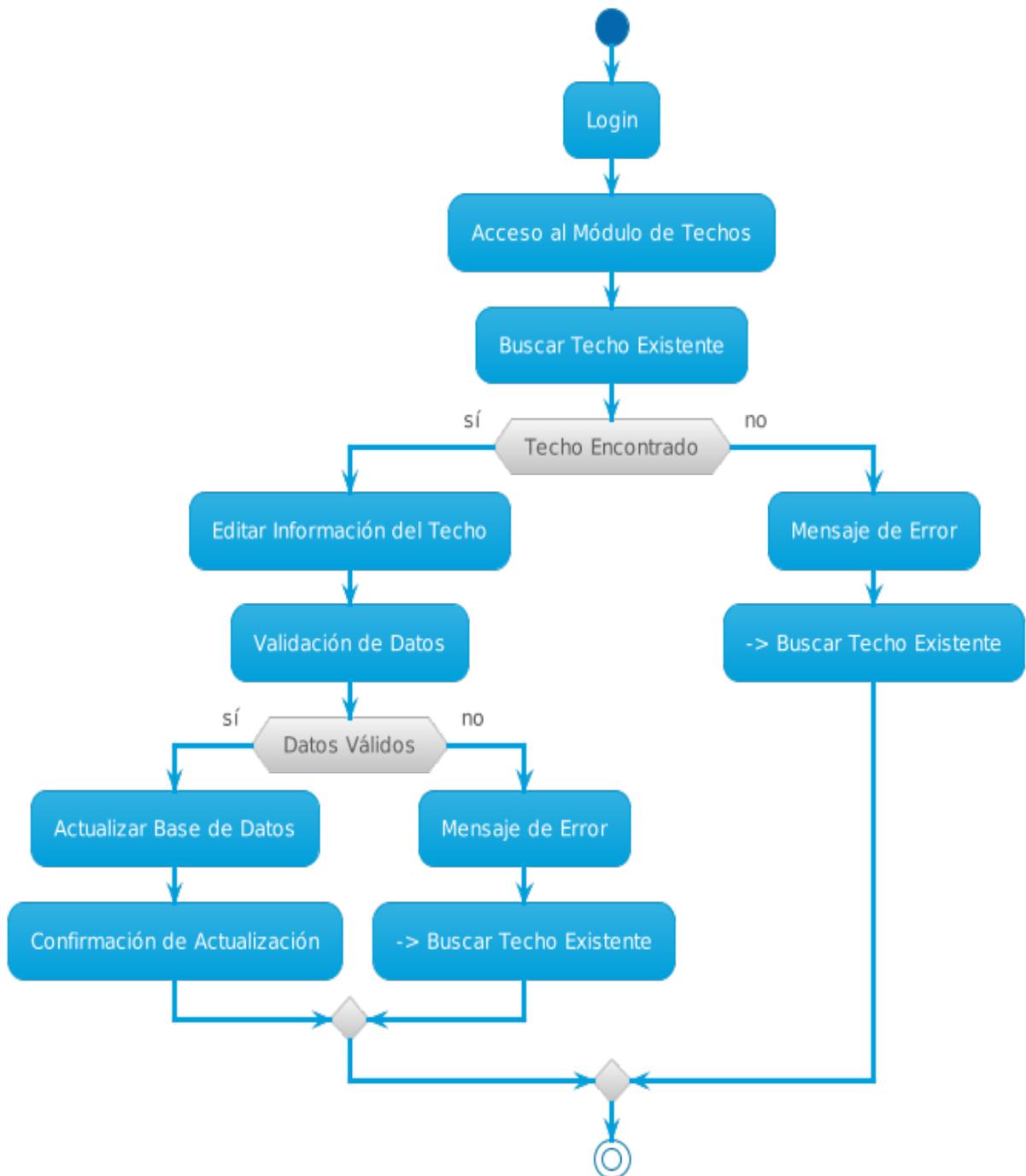


Diagrama 6 Diagrama de actividades – Editar techo a un número

## 7.1.6 Resultados de los Diagrama de Secuencia.

### 7.1.6.1 Gestionar techos

#### 7.1.6.1.1 Asignar techo a un numero

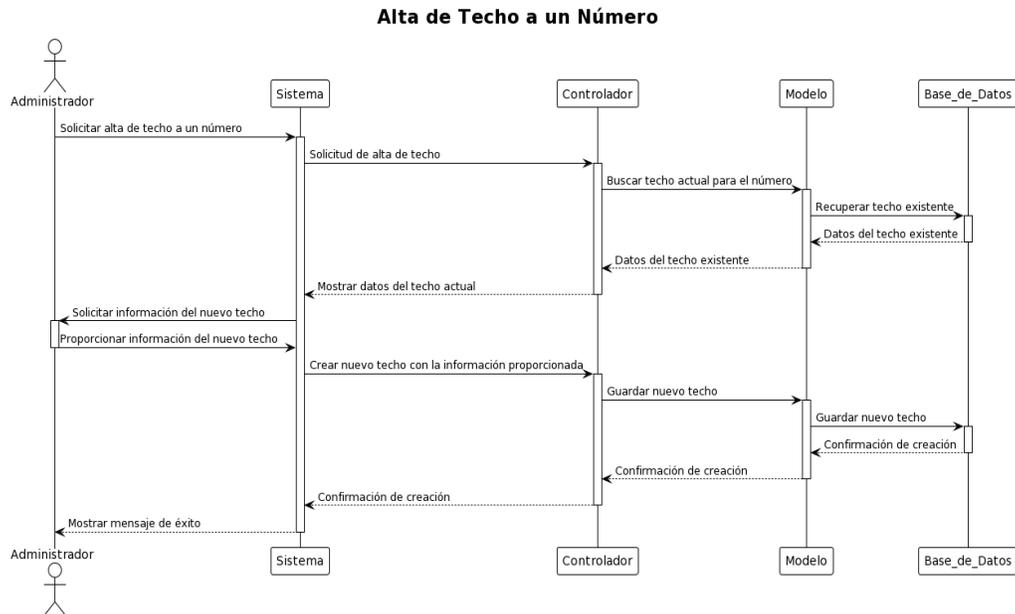


Diagrama 16 Diagrama de Secuencia – Asignar techo a un numero

#### 7.1.6.1.2 Editar techo a un numero

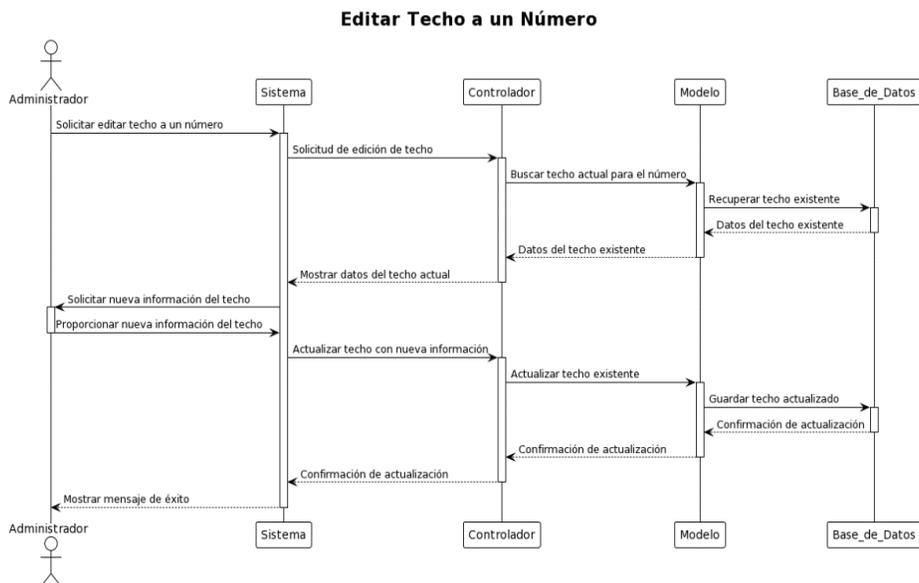


Diagrama 17 Diagrama de Secuencia – Editar techo a un numero

## **7.2 Diseño**

### **7.2.1 Diseño de Interfaz de Usuario.**

En nuestro sistema, la interfaz de usuario se centra en la accesibilidad y personalización según roles. Hay dos paneles principales: uno para el Usuario Administrador con acceso completo y otro para usuarios con accesos personalizados según tareas asignadas. Esto garantiza que cada usuario tenga la funcionalidad necesaria, mejorando la eficiencia y la experiencia.

Véase las ilustraciones 1, 5 y 6 en los Anexos en CD sección 11.2 Interfaces de sistema.

### **7.2.2 Diseño de Contenido**

En nuestro sistema, el diseño de contenido se enfoca en ofrecer una experiencia de usuario clara y eficiente. La página de inicio actúa como un panel principal intuitivo, proporcionando acceso directo a módulos esenciales como gestión de ventas, puntos de venta, juegos y premios. Cada módulo tiene iconografía y etiquetas claras para asegurar su comprensión inmediata. Los administradores tienen un panel para gestionar dinámicamente el contenido, manteniendo el sistema actualizado. El diseño es adaptable a diferentes dispositivos, manteniendo claridad y coherencia visual.

Véase la ilustración 6 en los Anexos en CD sección 11.2 Interfaces de sistema.

### **7.2.3 Diseño Estético**

En nuestro sistema, el diseño estético es crucial para crear una experiencia de usuario positiva y fortalecer la identidad de la marca. Se emplean formas suaves y colores que reflejan la marca, facilitando la navegación y ofreciendo una experiencia visual atractiva. Ilustraciones y elementos visuales relacionados con la loto enriquecen la interfaz y mejoran la comprensión del usuario. Véase ilustraciones 1 y 2 en los Anexos en CD, sección 11.2 Interfaces de sistema.

La organización de contenidos utiliza formas estratégicas como divisores para facilitar la diferenciación visual y la localización rápida de la información. Los elementos interactivos se destacan con un diseño que invita a la acción y ofrece feedback visual inmediato para una experiencia fluida y satisfactoria.

El diseño es adaptable y consistente en diferentes dispositivos, asegurando una experiencia positiva tanto en la versión web como en la aplicación móvil. Esto garantiza que la estética y funcionalidad se mantengan coherentes, fortaleciendo el reconocimiento de la marca Grupo Atenea. Véase ilustraciones 1 y 2 en los Anexos en CD, sección 11.2 Interfaces de sistema.

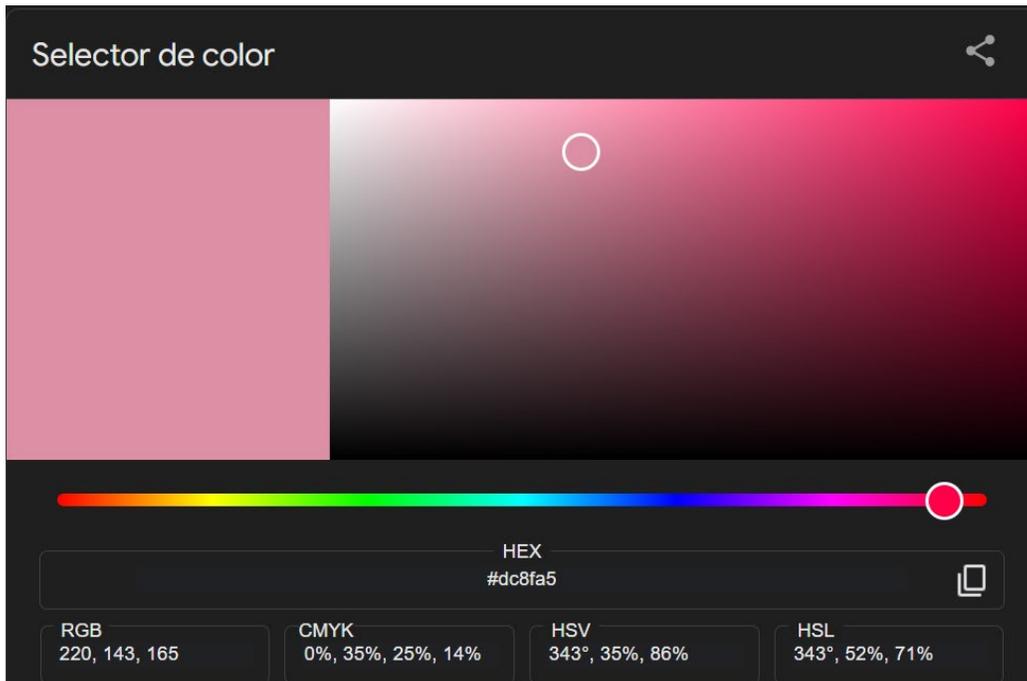


Imagen 2. Paleta de Colores Utilizados

## 7.2.4 Análisis y Diseño de Componentes y Despliegue

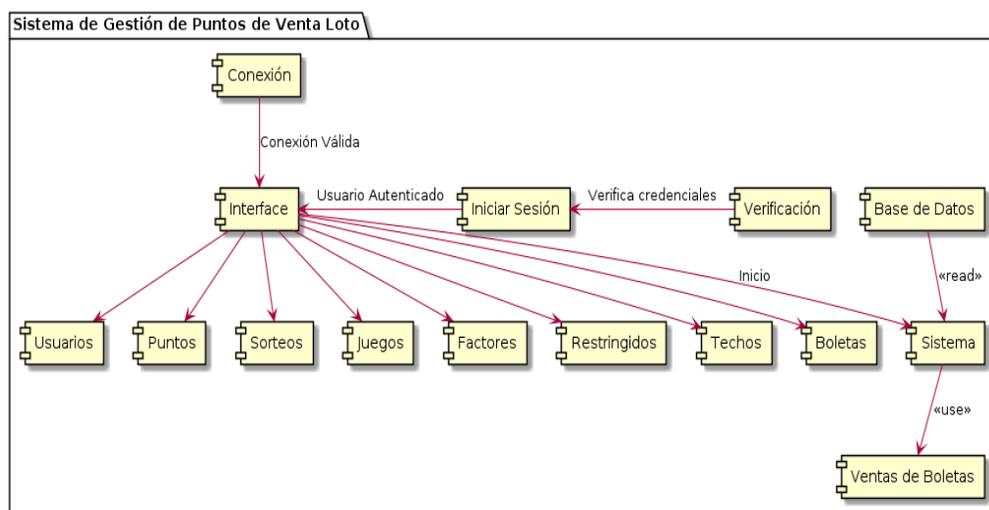


Diagrama 27: Diseño de Componentes

Representa la arquitectura del Sistema de Gestión de Puntos de Venta Loto, ilustrando la interacción entre los componentes principales del sistema. Central a la operación del sistema es la Interface, que valida la conexión y autentica a los usuarios a través de los componentes de Conexión, Iniciar Sesión y Verificación. Una vez autenticado, el usuario puede interactuar con el Sistema, el cual gestiona las Ventas de Boletas y el acceso a diferentes catálogos y acciones como Usuarios, Puntos, Sorteos, Juegos, Factores, Restringidos, Techos y la emisión de Boletas. Todos estos componentes se apoyan en una robusta Base de Datos, que almacena y proporciona la información necesaria para el funcionamiento integral del sistema. Véase ilustración 1 y 6 en los Anexos en CD sección 11.2 Interfaces de sistema.

### **7.3 Codificación**

#### **7.3.1 Sistema Web:**

##### **7.3.1.1 Módulo Factores:**

El Módulo de Factores consta de 4 Funciones Principales: Visualizar, Agregar, Editar y Eliminar.

##### **7.3.1.2 Ver Factores:**

La funcionalidad de "Ver Factores" permite mostrar los factores activos del sistema. La interfaz de usuario tanto en web como en móvil presenta una lista coherente y sincronizada de factores, la cual puede ser visualizada en las Ilustraciones 12 y 21 en los Anexos en CD. La obtención de estos factores activos es manejada en el backend, como se muestra en la Ilustración 33, donde se configura la API de NestJs para servir los datos requeridos.

##### **7.3.1.3 Agregar o Editar Factor:**

Para añadir nuevos factores o modificar los existentes, se utilizan los endpoints detallados en la Ilustración 32 en los Anexos en CD. Estos endpoints son consumidos por funciones en el servicio de factores en el backend, como se observa en las Ilustraciones 29 y 34 en los Anexos en CD, donde se muestran los métodos para crear y actualizar los factores respectivamente. Detrás de esta interfaz, el backend juega un papel crucial, comunicándose con la base de datos mediante endpoints RESTful diseñados para manejar la creación y actualización de factores

(Ilustración 34). El servicio de factores en NestJs (Ilustraciones 29 y 34 en los Anexos en CD) implementa métodos como *saveMany* y *update*, que respectivamente, añaden nuevos registros o modifican los existentes. Estas operaciones se ejecutan dentro de transacciones gestionadas por el repositorio de factores (Ilustración 21), asegurando la integridad y la consistencia de los datos.

#### **7.3.1.4 Eliminar Factor.**

La funcionalidad para eliminar factores está cuidadosamente implementada para asegurar que la operación sea segura y que los cambios se sincronicen adecuadamente en todas las interfaces. En el backend, se ha establecido un endpoint RESTful dedicado que atiende las solicitudes de eliminación (Ilustración 35 en los Anexos en CD). Cuando se invoca este endpoint, el servicio de factores procede a eliminar el registro específico de la base de datos. Se maneja una confirmación de la acción en la interfaz, lo que previene borrados accidentales y mejora la usabilidad del sistema. La eliminación es registrada en los logs del sistema a través del servicio de logger integrado, proporcionando un rastro de auditoría y aumentando la fiabilidad del proceso. Una vez completada la eliminación, la interfaz de usuario, tanto en la versión web como en la aplicación móvil, refleja la actualización de forma inmediata, manteniendo al usuario informado del estado actual de los datos.

### **7.3.2 Aplicación Móvil:**

#### **7.3.2.1 Módulo Factores:**

La gestión de factores en la aplicación móvil proporciona las mismas funcionalidades clave que la versión web y se mantiene sincronizada con ella. Esto garantiza una experiencia de usuario consistente y permite a los usuarios acceder a sus datos y realizar acciones desde cualquier dispositivo.

#### **7.3.2.2 Visualizar Factores:**

Utiliza la misma base de código que la versión web para mostrar los factores (Ilustraciones 12 y 21 en los Anexos en CD), asegurando una experiencia de usuario consistente a través de plataformas.

### **7.3.2.3 Agregar Factores:**

El formulario adaptado para dispositivos móviles permite añadir nuevos factores, utilizando los mismos métodos de la API que la versión web (Ilustración 32 en los Anexos en CD).

### **7.3.2.4 Editar Factores:**

La interfaz para editar factores en dispositivos móviles aprovecha las mismas capacidades de la API (Ilustraciones 29 y 34 en los Anexos en CD) para modificar la información de los factores existentes.

### **7.3.2.5 Eliminar Factores:**

Similar a la versión web, la aplicación móvil utiliza el servicio de factores para eliminar factores (Ilustración 35 en los Anexos en CD), con las necesarias confirmaciones para evitar eliminaciones accidentales.

### **7.3.2.6 Sincronización con el Sistema Web:**

La sincronización entre la aplicación móvil y el sistema web es esencial para garantizar la integridad de datos en el módulo de factores. Utilizando tecnologías modernas, como se muestra en las Ilustraciones 33 y 34 en los Anexos en CD, cualquier acción realizada en la aplicación móvil, como la adición, edición o eliminación de factores, se refleja inmediatamente en el sistema web. Esto se logra mediante el uso de WebSockets y servicios RESTful que mantienen un flujo bidireccional de información actualizado. Los usuarios pueden confiar en que verán la misma información relevante y actualizada independientemente de la plataforma que utilicen.

### **7.3.2.7 Interfaz de Usuario Adaptativa:**

La aplicación móvil utiliza Flutter para una interfaz adaptable, como se muestra en las Ilustraciones 12 y 21 en los Anexos en CD. Flutter permite que la aplicación responda y se ajuste automáticamente a diferentes tamaños de pantalla y orientaciones, asegurando que la funcionalidad no se vea comprometida en dispositivos más pequeños o al cambiar entre vistas de retrato y paisaje. Este diseño responsivo garantiza una experiencia coherente y accesible para todos los usuarios.

## 7.4 Pruebas o validación

### 7.4.1 Establecimiento de Puntos de Interrupción:

Se proporcionan instrucciones claras para colocar breakpoints efectivos que ayuden a inspeccionar el flujo de la aplicación, el estado de las variables y la pila de llamadas en tiempo real. Esto incluye ejemplos de cómo seleccionar líneas de código clave en los módulos de factores y otros componentes relevantes donde los errores son más propensos a surgir o donde se requiere una validación detallada del comportamiento de la aplicación. Además, se interpreta la información que se muestra en la ventana de depuración al alcanzar un breakpoint, como el valor de las variables y el estado actual de la ejecución, para diagnosticar problemas y verificar la lógica de la aplicación.



```
src > modules > factores > factores.service.ts > FactoresService
79   export class FactoresService {
80   }
81   async findAll(loggerId: LoggerIdentDto) {
82   }
83   }
84   }
85   }
86   }
87   }
88   }
89   async findOne(id: number, loggerId: LoggerIdentDto) {
90     const { requestId } = loggerId;
91     const factor: Factor = await this.factorRepository.findOne({
92       where: { factorId: id },
93     });
94     // Haga clic para agregar un punto de interrupción.
95     this.logger.warn(
96       `${requestId} - ${Mensajes.EL_FACTOR_CON_ID} ${id} ${Mensajes.NO_EXISTE}.`,
97     );
98     throw new BadRequestException(Mensajes.FACTOR_INEXISTENTE);
99   }
}
```

Imagen 3 Puntos de depuración en código

### 7.4.2 Ejecución y Observación:

Al utilizar la depuración en Visual Studio Code, confirmamos el correcto funcionamiento de los endpoints del módulo de factores. Los puntos de interrupción revelaron un flujo de ejecución adecuado y la inspección de variables demostró que los datos se manejan como se espera. Las respuestas de la API examinadas en Insomnia coincidieron con los casos de uso planeados, incluyendo la gestión correcta de errores. Los registros generados durante el proceso proporcionaron una trazabilidad valiosa que facilita la revisión y el mantenimiento continuo del código.

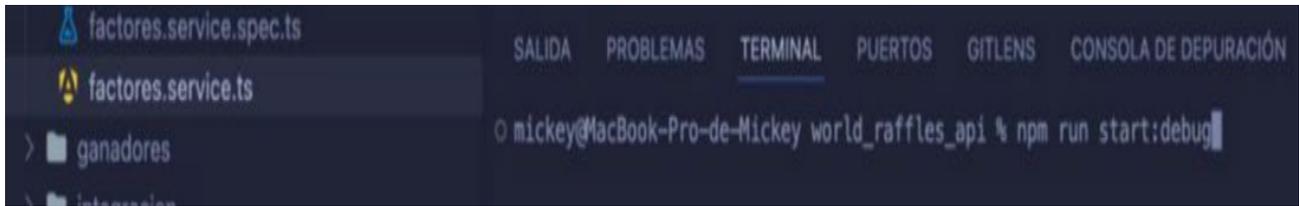


Imagen 4 Ejecución y observación

### 7.4.3 Inspección de Variables y Flujo:

En la fase de inspección de variables y flujo, la depuración nos permitió evaluar en tiempo real los valores y estados dentro de la aplicación, confirmando así la lógica de negocio y la integridad de los datos procesados. Este paso es crucial para asegurar que las operaciones, especialmente las relacionadas con la gestión de factores, se ejecuten según lo previsto, y para identificar posibles incongruencias o errores en las etapas tempranas del ciclo de vida del desarrollo.

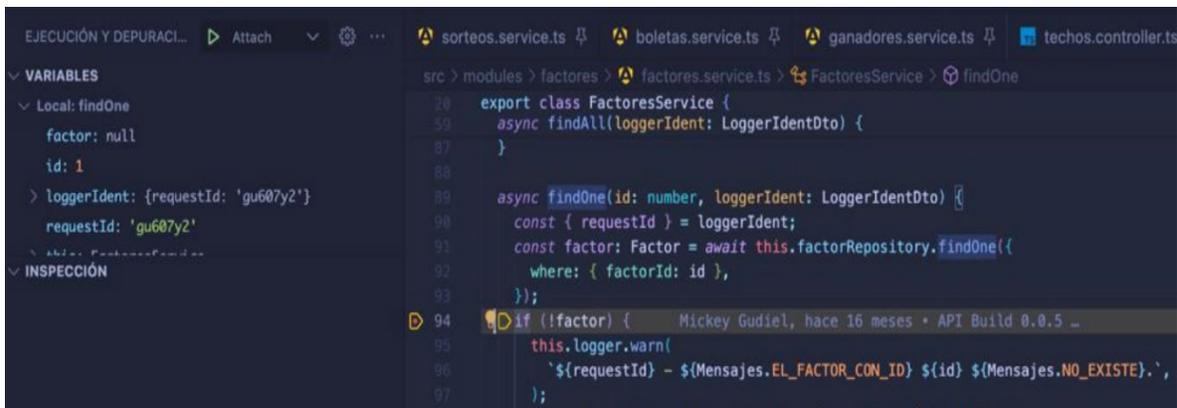


Imagen 5 Inspección de variables y flujo

### 1.2.2 7.4.4 Uso de Insomnia para Testeo de Endpoints:

El uso de Insomnia para el testeo de endpoints demostró ser un método eficaz para validar la funcionalidad de la API del módulo de factores. Al realizar peticiones específicas, pudimos verificar que los endpoints responden con los códigos y mensajes de estado adecuados, y que la API maneja correctamente los errores, asegurando así que los factores se gestionen de manera confiable en el sistema.

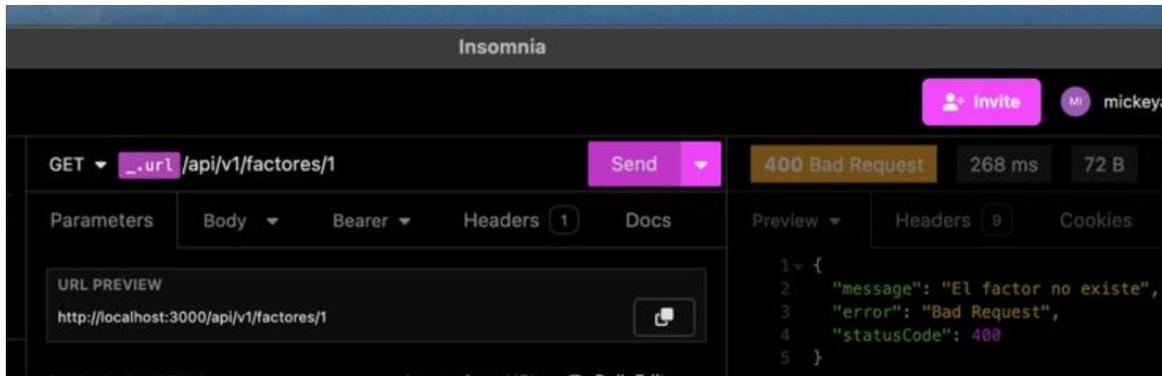


Imagen 6 Testeando endpoints con Insomnia

## VIII. ESTUDIO DE FACTIBILIDAD

### 8.1 Factibilidad Técnica

**8.1.1 Alternativa A:** Sistema y aplicación móvil para la gestión eficiente de puntos de venta loto para Grupo Atenea.

En el marco del análisis de factibilidad técnica, específicamente en la Alternativa A, que implica el desarrollo de un sistema y una aplicación móvil para la gestión eficiente de puntos de venta de loto para Grupo Atenea, se detallan las especificaciones del equipo de cómputo y del dispositivo móvil destinados a las pruebas. Se contempla la adquisición de un servicio de alojamiento en Digital Ocean como parte integral de la infraestructura requerida para el desarrollo y posterior implementación del proyecto.

Dell Inspiron 15 3000

Hardware	
Descripción	Precio
Procesador Core i3-6006U 2.00 GHz	
Disco Duro Solido de 256 GB	
8 GB de RAM	
Teclado numérico, Carga rápida	
Total:	US\$ 300

Tabla 14: costo y descripción de hardware / fuente: propia

Samsung Galaxy A14

Hardware	
Descripción	Precio
Mediatek MT6769 Helio G80 (12 nm) - Version A	
Android 13, One UI Core 5	
64GB ROM 4GB RAM	
Li-Po 5000 mAh, non-removable	

Total:	US\$ 140
--------	----------

Tabla 15: costo y descripción de hardware / fuente: propia

Impresora

Hardware	
Descripción	Precio
MP58-01, interfaz USB, inalámbrica, 203 dpi	US\$ 40
Soporta: Android, Windows, Mac, Linux, IOS	
Papel 58MM	US\$ 2
Total:	US\$ 42

Tabla 16: costo y descripción de hardware / fuente: propia

Alojamiento en Digital Ocean

Hosting	
Descripción	Precio
Compute: Virtual Machine (Droplet) - Basic Nodes - Regular CPU 2 GiB: 2 CPUs / 60 GiB SSD Disk / 2000 GiB Transfer, OS Debian Linux	\$192/año
Dominio Namecheap	\$2.98/año
Total:	\$194.98/año

Tabla 17: costo y descripción de recursos / fuente: propia

Recomendaciones por parte del equipo de trabajo:

- Extensión fiable: Opta por extensiones de dominio confiables y populares, como .com, .net o .org, que inspiren confianza y sean reconocidas a nivel mundial.
- Dominio corto y conocido: Elige un nombre de dominio corto y fácil de pronunciar que sea relevante para el negocio y fácil de recordar para los usuarios.
- Buscar alternativas: Explora diferentes opciones de proveedores de dominio y alojamiento web para obtener las mejores tarifas y características que se adapten a las necesidades del proyecto.
- Separar la base de datos a un servidor aparte para dividir la carga, para que el servidor que aloje la API solo este recibiendo peticiones, y sea más fluido el tráfico de datos entre los recursos asignados.
- Realizar un desarrollo que permita realizar ventas de manera offline y que cuando se establezca conexión de internet se envíe la información a la BD.

La infraestructura propuesta satisfará las necesidades de implementación del sistema, ya que se requiere que sea altamente portátil entre dispositivos y accesible desde múltiples ubicaciones. Dado que la empresa cuenta con 35 puntos de venta, es fundamental que la infraestructura de alojamiento web permita el acceso remoto y la colaboración en tiempo real entre los usuarios, lo que evitará la necesidad de inversiones adicionales en infraestructura física en cada ubicación.

**8.1.2 Alternativa B:** Sistema basado en la nube para la gestión eficiente de puntos de venta loto para Grupo Atenea.

Dentro del análisis de factibilidad técnica, se presenta la Alternativa B, la cual consiste en desarrollar un sistema basado en la nube para la gestión eficiente de puntos de venta loto para Grupo Atenea. En esta alternativa, se enfatiza en utilizar la infraestructura de la nube para alojar tanto el sistema como la aplicación móvil,

eliminando la necesidad de adquirir y mantener equipos de cómputo y dispositivos móviles específicos para las pruebas.

Las especificaciones detalladas del equipo de cómputo y del dispositivo móvil no son necesarias en esta alternativa, ya que el acceso al sistema se realizará a través de navegadores web y dispositivos móviles estándar con acceso a internet.

Recomendaciones por parte del equipo de trabajo:

- Selección del proveedor de servicios en la nube: Se recomienda investigar y seleccionar un proveedor de servicios en la nube confiable y con experiencia en alojamiento de aplicaciones empresariales. Se deben considerar aspectos como la seguridad, la disponibilidad, el rendimiento y el costo del servicio ofrecido.
- Configuración de la infraestructura en la nube: Es fundamental configurar adecuadamente la infraestructura en la nube para garantizar un rendimiento óptimo del sistema y la aplicación móvil. Esto incluye la asignación de recursos computacionales, el establecimiento de medidas de seguridad y la implementación de copias de seguridad regulares.
- Desarrollo de la aplicación móvil compatible con la nube: Se debe desarrollar la aplicación móvil de manera que pueda integrarse de forma transparente con la infraestructura en la nube. Esto garantizará una experiencia de usuario fluida y consistente en todos los dispositivos.
- Pruebas exhaustivas de rendimiento y seguridad: Antes de la implementación final, es crucial realizar pruebas exhaustivas de rendimiento y seguridad para asegurar que el sistema en la nube cumpla con los estándares de calidad y sea capaz de manejar la carga de trabajo esperada.

La infraestructura basada en la nube ofrecerá flexibilidad y escalabilidad, lo que permitirá adaptarse fácilmente a las necesidades cambiantes del negocio. Además,

al utilizar la nube, se reducen los costos de mantenimiento y se simplifica la gestión de la infraestructura, lo que resulta beneficioso para la empresa Grupo Atenea.

En resumen: **La alternativa “A”** es más viable pues proporciona una solución completa y bien definida que cumple con los objetivos del proyecto y garantiza una implementación exitosa del sistema de gestión de puntos de venta loto para Grupo Atenea. Su enfoque centrado en la personalización y la eficiencia lo convierte en la opción más indicada para satisfacer las necesidades específicas de la empresa.

## **8.2 Factibilidad Operativa**

El sistema web y la aplicación móvil estarán diseñados para facilitar la gestión diaria de las ventas de loto, techos y sorteos, asegurando una experiencia de usuario intuitiva y eficaz, lo cual permitirá a los vendedores y administradores realizar múltiples tareas de forma simultánea. El sistema propuesto será flexible, permitiendo actualizaciones y cambios según se requieran, garantizando así su mantenibilidad. Se hará hincapié en la seguridad y la fiabilidad para proteger los datos de transacciones y clientes, así como para asegurar la integridad en la comunicación entre los puntos de venta y el servidor central.

Con esta herramienta, los usuarios podrán gestionar la venta de números, el establecimiento de techos, la administración de números restringidos y la generación de reportes detallados de ventas y estadísticas de sorteos, todo de acuerdo con los permisos asociados a su perfil de usuario. El sistema ofrecerá un control eficiente y organizado de todas las operaciones relacionadas con los sorteos y ventas, facilitando la impresión de boletas y el seguimiento en tiempo real de las ventas y pagos.

### **8.2.1 Alternativa A: Sistema y Aplicación Móvil de Gestión de Puntos de Venta Loto.**

#### **Procesos:**

- Optimización en la atención al cliente mediante una interfaz de usuario que simplifica la interacción con los servicios ofrecidos.

- Organización y accesibilidad rápida de la información relacionada con las ventas y gestiones de los sorteos.
- Registro instantáneo de operaciones gracias a la conectividad en tiempo real a través de internet.

***Personal:***

Disminución del esfuerzo manual del personal en el área de ventas y administración gracias a la automatización.

El equipo de desarrollo del sistema estará compuesto por:

1. Analista de Sistemas: Encargado de analizar y definir los requerimientos del sistema desde su concepción hasta su implementación.
2. Programador: Responsable de codificar las funcionalidades definidas por el analista de sistemas en el software.
3. Especialista en Documentación: Documentará el sistema de forma clara y precisa, incluyendo manuales de usuario y especificaciones técnicas.

Este enfoque garantizará un sistema adaptable y fácil de usar que respalde las operaciones de la empresa, mejorando la eficiencia general y contribuyendo al éxito del negocio en el competitivo mercado de la venta loto."

**8.2.2 Alternativa B:** Prototipo de Servidor del Sistema y Aplicación Móvil de Gestión y Venta loto.

**Beneficios Tangibles**

- Los beneficios tangibles son aquellos que se pueden medir en términos de mejora en el rendimiento o en ahorros económicos directos. Los beneficios que experimentará la red de puntos de venta Loto con la implementación de este sistema son:
- Reducción de los tiempos de espera para los clientes al comprar números o recibir sus boletas.

- Incremento de la efectividad de los empleados, ya que los procesos manuales serán minimizados, permitiendo así que el personal dedique tiempo a tareas de mayor valor.
- Precisión mejorada en el registro y seguimiento de las ventas y los pagos de premios, reduciendo errores en las transacciones.
- Registro simplificado y eficiente de techos y números restringidos, facilitando la adaptación a las fluctuaciones del mercado.
- Ahorro de recursos físicos y de espacio al reducir la dependencia de la documentación en papel, al almacenar registros digitales seguros.

### **Beneficios Intangibles**

Los beneficios intangibles, aunque no se miden fácilmente, contribuyen significativamente al valor y al crecimiento sostenible de la organización. Estos incluyen:

- Mejora en la satisfacción del cliente debido a un servicio más rápido y personalizado.
- Fortalecimiento de la imagen de marca de la empresa, ya que la adopción de tecnología avanzada comunica modernidad y eficiencia.
- Mayor facilidad y agilidad en la toma de decisiones estratégicas gracias a la disponibilidad de análisis y reportes en tiempo real.
- Fomento de un ambiente de trabajo más ágil y menos cargado de tareas repetitivas, lo que puede contribuir a un mayor bienestar y satisfacción del personal.

### **8.3 Factibilidad Económica**

Al implementar el sistema de gestión y venta loto, mediante el uso de tecnologías y plataformas de código abierto para el desarrollo de la aplicación móvil y el sistema web, la empresa logrará un ahorro significativo al evitar los costos asociados con la adquisición de licencias de software propietario. Este enfoque no solo reduce los

costos iniciales de implementación, sino que también facilita la escalabilidad y la personalización del sistema para adaptarse a las necesidades específicas de la empresa sin incurrir en gastos adicionales. Además, la automatización de las ventas y la gestión de sorteos y números promete mejorar la eficiencia operativa, reduciendo los errores manuales y aumentando la satisfacción del cliente, lo que se traduce en un incremento en las ventas y, por ende, en la rentabilidad de la empresa.

### 8.3.1 Alternativa A: Sistema web y aplicación móvil para la gestión eficiente de puntos de venta loto para Grupo Atenea

- Costos de Inversión

Dell Inspiron 15 3000

Descripción	Precio
Procesador Core i3-6006U 2.00 GHz	
Disco Duro Solido de 256 GB	
8 GB de RAM	
Teclado numérico, Carga rápida	
Total:	US\$ 300

Tabla 18: costo y descripción de hardware / fuente: propia

Samsung Galaxy A14

Hardware	
Descripción	Precio
Mediatek MT6769 Helio G80 (12 nm) - Version A	
Android 13, One UI Core 5	
64GB	
4GB	
Li-Po 5000 mAh, non-removable	
Total:	US\$ 140

Tabla 19: costo y descripción de hardware / fuente: propia

## Impresora

Hardware	
Descripción	Precio
MP58-01, interfaz USB, inalámbrica, 203 dpi	US\$ 40
Soporta: Android, Windows, Mac, Linux, IOS	
Papel 58MM	US\$ 2
Total:	US\$ 42

Tabla 20: costo y descripción de hardware / fuente: propia

- Costos de desarrollo

Software	Precio
<b>Compute: Virtual Machine (Droplet) - Basic Nodes - Regular CPU</b>	\$192/año
<b>2 GiB: 2 CPUs / 60 GiB SSD Disk / 2000 GiB Transfer, OS Debian Linux</b>	
<b>Namecheap</b>	\$2.98/año
<b>Google Chrome y Mozilla Firefox.</b>	\$0
<b>TOTAL</b>	<b>\$194.98</b>

Recursos Humanos			
N.º	Cargo	Costo Individual	Costo Total
2	Desarrolladores	\$ 2500	\$ 5000
		Total	\$ 5000
Recursos Tecnológicos			
Hardware			
Cantidad	Descripción	Costo/Hora	Total

<b>2</b>	150 horas Computadora	\$ 0,80	\$ 240
			Total \$ 240
<b>Recursos Materiales</b>			
<b>Cantidad</b>	<b>Descripción</b>	<b>Costo</b>	<b>Total</b>
<b>25</b>	Transporte a reuniones	\$ 3	\$ 75
<b>20</b>	Viáticos	\$ 5	\$ 100
		Total	\$ 175

Tabla 21: costo y descripción de hardware / fuente: propia

<b>Flujo de Pago</b>	
<b>Recursos</b>	<b>Costos</b>
<b>Recursos Humanos</b>	\$ 5,000
<b>Recursos Tecnológicos</b>	\$ 240
<b>Recursos Materiales</b>	\$ 175
<b>Imprevistos (10%)</b>	\$ 200
<b>Total</b>	\$ 5,615

Tabla 22: flujo de pago / fuente: propia

### **Costos Complementarios:**

En el siguiente apartado se consideran los gastos de servicios como el agua, la luz, transporte y el servicio de Internet.

<b>servicio</b>	<b>Cantidad/Mes</b>	<b>Costo Estimado al mes</b>	<b>Total</b>
<b>Luz %20</b>	6	U\$ 19.60	\$ 117.60
<b>Agua%20</b>	6	U\$ 5.00	\$ 30.00
<b>Internet%40</b>	6	U\$ 12	\$ 72.00
		Total	\$ 229.00

Tabla 23: costos complementarios / fuente: propia

## Costo Concurrente

<b>COSTOS</b>	<b>VALOR</b>
<b>Costo soporte y actualización software de mantenimiento / año</b>	\$ 200
<b>Consultoría para mantenimiento / año</b>	\$ 200
<b>Capacitación para mantenimiento / por año</b>	\$ 50
<b>Total</b>	\$ 450

Tabla 24: costo concurrente / fuente: propia

## Costo Total

<b>Costo de Inversión</b>	\$ 482
<b>Costo de desarrollo</b>	\$ 5,615
<b>Costo Concurrentes</b>	\$ 450
<b>Total</b>	\$ 6,547

Tabla 25: costo total / fuente: propia

### 8.3.2 Alternativa B: Sistema web y aplicación móvil para la gestión eficiente de puntos de venta loto para Grupo Atenea

- Costos de inversión

<b>Software</b>	<b>Precio</b>
<b>Hosting en OrangeHost</b>	\$16/año
<b>Google Chrome y Mozilla Firefox.</b>	\$0
<b>TOTAL</b>	\$ 16

Tabla 26: costo de inversión / fuente: propia

- Costos de desarrollo

<b>Recursos Humanos</b>			
<b>N.º</b>	<b>Cargo</b>	<b>Costo Individual</b>	<b>Costo Total</b>
<b>1</b>	Desarrollador	\$ 1500	\$ 1500
		Total	\$ 1500
<b>Recursos Tecnológicos</b>			

<b>Hardware</b>			
<b>Cantidad</b>	<b>Descripción</b>	<b>Costo/Hora</b>	<b>Total</b>
<b>2</b>	140 horas Computadora	\$ 0,80	\$ 224
		<b>Total</b>	<b>\$ 224</b>
<b>Recursos Materiales</b>			
<b>Cantidad</b>	<b>Descripción</b>	<b>Costo</b>	<b>Total</b>
<b>25</b>	Transporte a reuniones	\$ 3	\$ 75
<b>20</b>	Viáticos	\$ 5	\$ 100
		<b>Total</b>	<b>\$ 175</b>

Tabla 27: costos de desarrollo / fuente: propia

<b>Flujo de Pago</b>	
<b>Recursos</b>	<b>Costos</b>
<b>Recursos Humanos</b>	\$ 1500
<b>Recursos Tecnológicos</b>	\$ 224
<b>Recursos Materiales</b>	\$ 175
<b>Imprevistos (10%)</b>	\$ 110
<b>Total</b>	<b>\$ 2,009</b>

Tabla 28: flujo de pago / fuente: propia

### **Costos Complementarios:**

En el siguiente apartado se consideran los gastos de servicios como el agua, la luz, transporte y el servicio de Internet.

<b>servicio</b>	<b>Cantidad/Mes</b>	<b>Costo Estimado al mes</b>	<b>Total</b>
<b>Luz %20</b>	6	U\$ 19.60	\$ 117.60
<b>Agua%20</b>	6	U\$ 5.00	\$ 30.00

<b>Internet%40</b>	6	U\$ 12	\$ 72.00
		Total	\$ 229.00

Tabla 29: costos complementarios / fuente: propia

### Costo Concurrente

<b>COSTOS</b>	<b>VALOR</b>
<b>Costo soporte y actualización software de mantenimiento / año</b>	\$ 100
<b>Consultoría para mantenimiento / año</b>	\$ 100
<b>Capacitación para mantenimiento / por año</b>	\$ 50
<b>Total</b>	\$ 250

Tabla 30: costo concurrente / fuente: propia

### Costo Total

<b>Costo de Inversión</b>	\$ 16
<b>Costo de desarrollo</b>	\$ 2,009
<b>Costo Concurrentes</b>	\$ 250
<b>Total</b>	\$ 2,275

Tabla 31: costo total / fuente: propia

En conclusión, nuestra preferencia se orienta hacia la **Alternativa A**, dado que presenta una solución más integral, brindando una infraestructura en la nube superior con un abanico de servicios más extenso y versátil en comparación con la opción subsiguiente. Esta alternativa asegura una plataforma robusta y escalable que se adapta eficientemente a nuestras necesidades operativas y de crecimiento futuro.

## **8.4 Factibilidad Legal**

En el marco de la factibilidad legal para nuestro proyecto, se llevará a cabo un período de prueba de un mes, destinado a realizar un análisis exhaustivo del software. El objetivo principal es identificar y resolver cualquier posible incidencia, garantizando así la estabilidad y la conformidad con las regulaciones legales pertinentes.

Es importante destacar que nuestro enfoque se centra en la utilización exclusiva de software de código abierto, eliminando la necesidad de adquirir licencias propietarias. Además, se contempla el uso de soluciones de pago únicamente para el hosting, optando por servicios confiables como Digital Ocean. Este enfoque no solo fortalece la sostenibilidad y escalabilidad del proyecto, sino que también asegura la conformidad legal en el uso de tecnologías de código abierto.

### **Marco Jurídico:**

- Cumplimiento con la Ley General de Protección de Datos Personales, asegurando un manejo de información de clientes y transacciones conforme a la legislación.

### **Comunicación y Tratamiento de Datos:**

- La responsabilidad del tratamiento de datos recae tanto en los desarrolladores del sistema como en la empresa, asegurando que el manejo de datos personales cumpla con las normativas vigentes. Esto incluye la obtención del consentimiento explícito de los usuarios para el procesamiento de sus datos personales, enfatizando la transparencia en el uso de esta información.

### **Consentimiento de Datos:**

- Se requerirá el consentimiento inequívoco de los usuarios para el tratamiento de sus datos personales, a excepción de lo que establezca la ley.

- Se evitará solicitar datos sensibles no esenciales para el acceso al sistema.

**Deber de Información:**

- Los usuarios serán informados de manera clara sobre los detalles de la recolección de datos, incluido el propósito del tratamiento de sus datos personales y sus derechos de acuerdo con la legislación aplicable.

**Importancia de la Protección de Datos en el Sistema:**

- El sistema se beneficiará significativamente de una gestión de seguridad de la información robusta, garantizando la accesibilidad, integridad y confidencialidad de los datos. La adopción de protocolos de seguridad avanzados permitirá una distribución segura de documentos, la prevención del robo de datos y asegurará la privacidad de la información manejada por el sistema.

Este enfoque integral hacia la protección de datos y el cumplimiento legal no solo aumentará la confianza de los usuarios en el sistema, sino que también protegerá a la empresa contra posibles sanciones y riesgos reputacionales.

## IX. CONCLUSIÓN

El desarrollo de este proyecto monográfico se centró en la creación de un sistema web y una aplicación móvil destinados a la gestión eficiente de puntos de venta Loto para el Grupo Atenea. Con la finalización de este trabajo, nos complace concluir con la satisfactoria aceptación del sistema por parte de la dirección de Grupo Atenea, destacando el logro de los objetivos propuestos y la implementación exitosa de las funcionalidades acordadas al inicio del proyecto.

A lo largo de la implementación del sistema web y la aplicación móvil, se demostró su efectividad en la optimización de procesos, logrando un significativo ahorro de tiempo en las operaciones diarias y una mejora notable en la ejecución de tareas administrativas y de gestión de ventas. Además, el proyecto ha contribuido a la reducción de costos relacionados con el uso de papel y ha eliminado casi por completo las pérdidas de información, asegurando una eficiencia del 100% en el sistema desarrollado. Este éxito se sustenta en la implementación de métodos de control y gestión operacional avanzados, así como en la cuidadosa codificación de la plataforma en un entorno web y móvil seguro y accesible.

### **Recomendaciones:**

Para fortalecer aún más el sistema y expandir sus capacidades, sugerimos las siguientes mejoras futuras:

- **Enriquecimiento de Datos:** Incorporar campos adicionales en los registros de usuarios, puntos de venta y vendedores para ofrecer una gestión más detallada y personalizada.
- **Expansión de Funcionalidades:** Añadir herramientas que permitan una gestión más exhaustiva de sorteos.
- **Integración de Análisis Avanzados:** Implementar módulos de análisis de ventas y patrones de compra para optimizar las estrategias de ventas.

- Mejoras en la Seguridad de Datos: Reforzar los protocolos de seguridad para la protección de datos personales y transaccionales, cumpliendo con las normativas más estrictas en materia de privacidad.
- Crear un módulo que posibilite ventas tanto en línea como fuera de línea. Los datos se almacenarán localmente y se enviarán a la base de datos cuando se recupere la conexión a internet.

El trabajo realizado sienta las bases para una gestión moderna y eficaz de los puntos de venta Loto, ofreciendo al Grupo Atenea una herramienta poderosa para liderar en el sector. Confiamos en que las recomendaciones propuestas contribuirán al crecimiento continuo y al éxito sostenido del sistema en el futuro.

## X. BIBLIOGRAFÍA

- Apache HTTP Server Project. (2024). Acerca del Proyecto Apache HTTP Server. Recuperado de [https://httpd.apache.org/ABOUT\\_APACHE.html](https://httpd.apache.org/ABOUT_APACHE.html)
- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Thomas, D. (2001). Manifesto for agile software development. Agile Alliance.
- Beizer, B. (2003). Software testing techniques. Dreamtech Press.
- Boehm, B. (1986). A spiral model of software development and enhancement. Computer, 21(5), 61-72. <https://doi.org/10.1109/MC.1988.1663532>
- Chen, H., Zhang, J., & Zhang, Y. (2012). Mobile applications: Whats next? IEEE Computer Society.
- Debian. (2024). Acerca de Debian. Recuperado de <https://www.debian.org/intro/about>
- Diego Coder 2024, recuperado de: <https://medium.com/@diego.coder/introducci%C3%B3n-a-las-clean-architectures-723fe9fe17fa>
- Digital Ocean 2024, recuperado el 20 de enero de 2024 de: <https://www.digitalocean.com/about>
- Fielding, R. T. (2000). Architectural styles and the design of network-based software architectures. Doctoral dissertation, University of California, Irvine.
- Flutter. (2021). What is Flutter? <https://flutter.dev/docs/get-started/learn-more>
- Fowler, M., & North, D. (2006). Continuous integration. ThoughtWorks.
- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). Patrones de Diseño: Elementos de Reutilización de Software Orientado a Objetos. Addison-Wesley.

- Insomnia. (2024), recuperado el 25 de enero de 2024 de: <https://insomnia.rest/products/insomnia>
- Kaspersky. (2024). Recuperado de <https://latam.kaspersky.com/resource-center/definitions/what-is-a-ssl-certificate>
- Kim, G., Humble, J., & Debois, J. (2018). The DevOps handbook: How to create world-class agility, reliability, and security in technology organizations. IT Revolution.
- Laudon, K. C., & Laudon, J. P. (2017). Sistemas de información gerencial. Pearson.
- Martin, R. C. (2002). Agile Software Development, Principles, Patterns, and Practices. Prentice Hall.
- MobX. (s.f.). MobX Documentation. Recuperado de <https://mobx.js.org/>
- Mozilla Developer (2022) Recuperado el 15 de enero de 2024, de <https://developer.mozilla.org/es/docs/Web/Security>
- Newman, S. (2015). Building microservices: Designing fine-grained systems. OReilly Media, Inc.
- Node.js. (2021). About Node.js. <https://nodejs.org/en/about/>
- Norman, D. A., & Nielsen, J. (2010). The definition of user experience. Nielsen Norman Group.
- PlantUML. (s.f.). Recuperado el 25 de enero de 2024, de <https://plantuml.com/>
- PM2. (2024). PM2 - Advanced, production process manager for Node.js. Recuperado de <https://pm2.keymetrics.io/docs/usage/quick-start/>
- Quizizz. (s. f.). Modelo en espiral [Imagen]. Recuperado el 23 de febrero de 2023, de

<https://quizizz.com/admin/quiz/6079a7907b10f0001ff9ce99/modelo-en-espiral>

- Socket.io. (2024). Acerca de Socket.io. Recuperado de <https://socket.io/docs/v4/>
- Sommerville, I. (2016). Software engineering. Pearson Education Limited.
- Swagger. (2024). Acerca de Swagger. Recuperado de <https://swagger.io/>
- Statista, 16 de febrero de 2022 en Tendencias en el desarrollo de aplicaciones móviles en 2022
- <https://es.linkedin.com/pulse/tendencias-en-el-desarrollo-de-aplicaciones-m%C3%B3viles-2022-v2msoft>
- World Wide Web Consortium (W3C). (2018). Responsive web design. <https://www.w3.org/standards/webdesign/responsive-design>

## **XI. ANEXOS**

Se puede revisar a fondo todo el contenido de los anexos en el Disco Adjunto.