

**UNIVERSIDAD CENTRAL  
VICERRECTORÍA ACADÉMICA  
FACULTAD DE INGENIERÍA Y ARQUITECTURA**

**Desarrollo de una solución informática de escritorio para la gestión  
de citas veterinarias en la pyme Veterinaria Pulguitas**

**MODALIDAD DE TESIS PARA OPTAR POR EL GRADO DE BACHILLER EN INGENIERÍA  
INFORMÁTICA**

**ELABORADO POR**

**CRISTOPHER GERARDO ULLOA JIMÉNEZ**

**TUTOR: ING. MAURICIO TORRES**

**SEDE CENTRAL**

**JULIO, 2024**

**Tabla de contenido**

Figuras.....	VI
Tablas.....	VII
Dedicatoria.....	X
Agradecimiento.....	XI
Resumen.....	XII
Capítulo I. Problema.....	1
Planteamiento del problema.....	2
Objetivos.....	3
Objetivo general.....	3
Objetivos específicos.....	3
Justificación.....	3
Antecedentes.....	4
Antecedentes internacionales.....	4
Antecedentes nacionales.....	6
Proyecciones.....	9
Alcances.....	9
Limitaciones.....	10
Capítulo II. Marco teórico.....	11
Historia de la programación.....	12
Paradigmas de programación.....	12
Programación orientada a objetos.....	13
Ventajas de la programación orientada a objetos.....	14
Bases de datos.....	15
Tipos de bases de datos.....	15
SQL.....	16
DBMS.....	16
Visual Studio.....	17
C#.....	17

Arquitectura cliente-servidor .....	18
Ventajas de utilizar cliente-servidor .....	18
Scrum.....	19
Roles en Scrum .....	20
Artefactos de Scrum .....	21
Eventos Scrum .....	22
Valores Scrum .....	23
Historias de usuario .....	24
Beneficios claves de las historias de usuario.....	25
Factores por considerar en la confección de historias de usuario .....	25
Git.....	26
Sistemas de información.....	27
Tipos de sistemas de información .....	28
Windows Form .....	29
UML .....	29
Requerimientos funcionales y no funcionales .....	29
Metodologías de desarrollo de software .....	31
Modelo incremental.....	31
Modelo en cascada .....	32
Modelo XP .....	33
Tipos de prototipo.....	33
Prototipos funcionales .....	34
Prototipos de experiencia de usuario.....	35
Prototipos de procesos comerciales.....	35
Tipos de algoritmo .....	35
Pruebas unitarias, integrales y globales .....	37
Capítulo III. Marco metodológico.....	38
Marco metodológico .....	39
Enfoque de la investigación .....	40
Método de la investigación.....	41

Fuentes de información .....	42
Variables.....	42
Instrumentos .....	44
Metodología ágil.....	44
Adaptación de la metodología.....	45
Fases.....	46
¿Qué beneficios ofrece la utilización de esta metodología?.....	46
Capítulo IV. Análisis de resultados.....	47
Plan de gestión de riesgos.....	48
Identificación y control de riesgos .....	49
Clasificación de los riesgos .....	49
Ponderación de los riesgos .....	50
Probabilidad de ocurrencia.....	50
Impacto.....	51
Matriz de riesgos .....	51
Estudio de factibilidad.....	52
Análisis de factibilidad.....	53
Factibilidad técnica .....	54
Factibilidad económica.....	56
Factibilidad operativa .....	56
Capítulo V. Conclusiones y recomendaciones.....	57
Conclusiones.....	58
Recomendaciones .....	59
Capítulo VI. Propuesta .....	60
Análisis de requerimientos.....	61
Requerimientos de usuario .....	61
Requerimientos funcionales .....	66
Requerimientos no funcionales .....	71
Modelos UML .....	73
Casos de uso.....	73

Diagramas de secuencia .....	82
Diagramas de actividades del sistema.....	87
Modelo ER (Entidad Relación).....	92
Diccionarios de datos .....	92
Diseño de pantallas del sistema .....	97
Pantalla de Inicio de sesión .....	97
Pantalla del menú principal .....	97
Pantalla para el manejo de clientes .....	98
Pantalla para el manejo de pacientes.....	98
Pantalla para manejo de citas .....	99
Pantalla modificación datos de cita activa .....	100
Pantalla manejo de tratamientos.....	100
Pantalla para el manejo de valoraciones de citas.....	101
Pantalla para modificación de valoraciones .....	102
Pantalla para manejo de usuarios. ....	102
Referencias bibliográficas .....	104
Anexo 1. Carta de revisión filológica.....	108

## Figuras

Figura 1 .....	12
Figura 2 .....	14
Figura 3 .....	17
Figura 4 .....	18
Figura 5 .....	19
Figura 6 .....	21
Figura 6 .....	24
Figura 7 .....	26
Figura 8 .....	30
Figura 9 .....	34
Figura 10 .....	74
Figura 11 .....	75
Figura 12 .....	76
Figura 13 .....	77
Figura 14 .....	79
Figura 15 .....	80
Figura 16 .....	81
Figura 17 .....	82
Figura 18 .....	83
Figura 19 .....	84
Figura 20 .....	84
Figura 21 .....	85
Figura 22 .....	85
Figura 23 .....	86
Figura 24 .....	86
Figura 25 .....	87

Figura 26 .....	87
Figura 27 .....	88
Figura 28 .....	89
Figura 29 .....	89
Figura 30 .....	90
Figura 31 .....	91
Figura 32 .....	97
Figura 33 .....	97
Figura 34 .....	98
Figura 35 .....	98
Figura 36 .....	99
Figura 37 .....	100
Figura 38 .....	100
Figura 39 .....	101
Figura 40 .....	102
Figura 41 .....	102

## **Tablas**

Tabla 1 .....	42
Tabla 2 .....	45
Tabla 3 .....	50
Tabla 4 .....	50
Tabla 5 .....	51
Tabla 6 .....	52
Tabla 7 .....	54
Tabla 8 .....	54
Tabla 9 .....	55
Tabla 10 .....	56
Tabla 11 .....	61
Tabla 12 .....	62

Tabla 13.....	62
Tabla 14.....	63
Tabla 15.....	63
Tabla 16.....	64
Tabla 17.....	64
Tabla 18.....	65
Tabla 19.....	66
Tabla 20.....	66
Tabla 21.....	67
Tabla 22.....	67
Tabla 23.....	68
Tabla 24.....	69
Tabla 25.....	69
Tabla 26.....	70
Tabla 27.....	71
Tabla 28.....	72
Tabla 29.....	72
Tabla 30.....	73
Tabla 31.....	74
Tabla 32.....	75
Tabla 33.....	76
Tabla 34.....	78
Tabla 35.....	79
Tabla 36.....	80
Tabla 37.....	92
Tabla 38.....	93
Tabla 39.....	93
Tabla 40.....	93
Tabla 41.....	94
Tabla 42.....	94

Tabla 43.....	94
Tabla 44.....	95
Tabla 45.....	95
Tabla 46.....	96
Tabla 47.....	96

## **Dedicatoria**

A Dios, por ser mi pilar de fortaleza.

A mis padres, por su incondicional apoyo y creer en mí, ellos son y serán siempre mi motivación.

A mis hermanas, por siempre estar presentes en cada una de las etapas de mi vida.

**Cristopher Gerardo Ulloa Jiménez**

## **Agradecimiento**

Agradezco a mi familia por su apoyo en todo momento, por siempre darme motivación a lo largo de estos años de formación académica. Asimismo, extendo el agradecimiento a mi tutor por sus consejos y su atenta colaboración en la elaboración de esta investigación. Y finalmente a todos aquellos que de una u otra forma han aportado en mi proceso educativo para al fin concluir exitosamente esta etapa de mi vida.

## Resumen

La presente investigación está enfocada en el desarrollo de un sistema de citas para la Veterinaria Pulguitas. Este trabajo pretende agilizar el proceso por medio del cual manejan lo referente al servicio de citas, manejo de clientes y pacientes en la veterinaria. En aras de realizar un diagnóstico de la situación, este trabajo se fija desde un enfoque mixto, a pesar de que ciertamente hay más inclinación por el contexto cualitativo, no se deja de lado por completo el aspecto cuantitativo.

Se propone un análisis a partir de historias de usuario, casos de uso y Scrum, mediante los cuales se busca identificar las oportunidades de mejora dentro del establecimiento. En este sentido y de acuerdo con la información recolectada, se brinda una opción informática que incluye una base de datos junto con un sistema que no requiere licenciamiento, lo cual en este contexto calza con la problemática en cuestión.

De acuerdo con el problema, un sistema que incluya las diferentes pantallas de manejo de clientes, usuarios, pacientes y citas puede mejorar significativamente el proceso de recolección de datos, así como centralizar el proceso en una única herramienta que permita agilizar la labor de la veterinaria. Ciertamente, una vez que esta solución sea sometida a las pruebas oportunas para garantizar el funcionamiento esperado y tenga el aval, se espera que se considere la implementación en el local.

## **Capítulo I. Problema**

## **Planteamiento del problema**

Para cualquier tipo de organización dedicada a brindar servicios, resulta de mucha utilidad contar con herramientas que permitan la gestión y control de estos, pero un adecuado seguimiento depende de un eficiente registro de entrada y salida. En este sentido, una solución que brinde ayuda en dicho aspecto permite establecer un marco de referencia sobre la situación de la empresa.

Sucedo en muchos casos, que este tipo de empresas consideradas pymes deben abrir un poco la carta de servicios a ofrecer, principalmente, para hacer más atractivo al cliente el paso por su local. Para este caso específico, se tiene como principal al servicio de consulta veterinaria.

Ahora bien, en el caso de los locales que ofrecen el servicio de consulta veterinaria, es de suma relevancia contar con alguna solución que permita el registro y consulta de los pacientes veterinarios; principalmente, para agilizar el proceso de valoración y seguimiento del estado de cada uno de los que en algún momento visitan el local.

Independientemente de si la empresa es pequeña, mediana o grande, siempre es importante contar con una buena administración de los servicios que se ofrecen. En lo que concierne a este caso, para cumplir con esta máxima, lo oportuno es abordar de manera íntegra la ausencia de control y el problema de agilizar el registro de la consulta veterinaria.

## **Objetivos**

### ***Objetivo general***

Desarrollar una solución informática para la pyme Veterinaria Pulguitas a fin de buscar la optimización en el manejo de consultas veterinarias dentro del local implementando una solución de escritorio a la medida.

### ***Objetivos específicos***

1. Revisar los requerimientos necesarios para el desarrollo de la solución a través de requerimientos funcionales, no funcionales, historias de usuario y casos de uso.
2. Definir herramientas y metodologías analizando cuáles son las más convenientes para una solución efectiva y el óptimo desarrollo del sistema en cuestión.
3. Establecer el diseño de la estructura de base de datos para el manejo de información dentro de la solución tomando en cuenta la normalización y buenas prácticas.
4. Desarrollar los respectivos formularios para el cumplimiento de funcionalidades tomando en cuenta la administración del sistema.
5. Ejecutar pruebas en el sistema para verificar la correcta ejecución de las funcionalidades considerando las diferentes opciones para estos fines.

## **Justificación**

Al implementar esta solución se busca colaborar con la eficiencia respecto al control y gestión de las citas en la veterinaria. Este es un problema que ocurre en los locales que no han

tenido la oportunidad de implementar una solución tecnológica que aliviane la carga de trabajo del negocio. Por lo que esta investigación es útil en la medida en que analiza el caso del pequeño emprendimiento e intenta solucionar la necesidad específica de un mejor control. A partir de esto, se busca tener un impacto en la eficiencia del manejo de las citas según sea conveniente.

En este sentido, haciendo un análisis preliminar de la situación en general del negocio y contrastando los fines del proyecto, se considera que se cumple con las especificaciones necesarias para llevar a cabo la investigación planteada, tanto en infraestructura disponible como las herramientas de desarrollo, así como la disposición del local por mejorar los puntos anteriormente señalados.

Aunado a lo dicho hasta este punto, es importante señalar que esta investigación se erige como una donación que intenta apoyar a la pyme para que esta siga colaborando con el tratamiento veterinario de muchos animales. Con estos objetivos, la solución pretende colaborar con el manejo y centralización de la información a partir de un sistema intuitivo y amigable listo para su uso.

Finalmente, es de suma relevancia indicar que la veterinaria, además de los clientes y pacientes que recibe todos los días, también trata animales en situación de maltrato o que viven en las calles; de ahí que este trabajo y solución se plantea como colaboración de bien social.

## **Antecedentes**

### ***Antecedentes internacionales***

El primer antecedente que se toma en consideración es el desarrollado por Hidalgo (2016), el cual fue realizado para optar por el grado de ingeniero en Sistemas. El objeto de

investigación fue desarrollar una herramienta que permita la sistematización en el control de ingresos y egresos que se realicen diariamente. De este trabajo resulta oportuno tomar como ejemplo la forma en que aborda el problema por medio del levantamiento de requerimientos funcionales y no funcionales, así como la confección de casos de uso con sus respectivos diagramas. Asimismo, se considera la utilización de diagramas de clases, de datos, entre otros, los cuales son oportunos para un manejo íntegro de los datos.

Como segundo antecedente, se considera el trabajo sobre temática de recolección y manejo de documentos presentado por Cabrera (2018), para optar por el grado de ingeniero en Electrónica y Redes de Información. El objeto de la investigación de Cabrera (2018) es desarrollar un prototipo de aplicación que permita automatizar el proceso de recolección de documentos de los profesores del DETRI (Cabrera, 2018). Dentro de las herramientas utilizadas para el desarrollo de este y que son importantes de considerar en esta investigación, se encuentran: motor de base de datos SQL Server, además, toma como metodología Kanban.

En este trabajo, se hace uso de la metodología Kanban, cuyo principal objetivo es gestionar de manera general el desarrollo de las tareas, lo que permite efectuar entregas en cualquier instante; además, facilita el cambio de las prioridades de las tareas y cuenta con una visualización perfecta del flujo de trabajo.

Dicho proyecto es importante para esta investigación en la medida en que incorpora el uso de Microsoft Visual Studio como IDE. Lo cual es de relevancia a la hora de considerar una posible integración con alguna otra herramienta de Microsoft, por ejemplo, SQL Server. También es relevante señalar el uso de otra metodología ágil que puede ser de utilidad a la hora de establecer un flujo de actividades por realizar.

Como tercer antecedente se toma el trabajo realizado por Chacha (2022) para optar por el grado de ingeniero en Sistemas. El proyecto de investigación tiene como finalidad crear un sistema para el control de inventario y producción. Este se propone como solución, ya que el manejo de registros se hace de forma manual; en este contexto, el uso de herramientas tecnológicas busca abarcar la automatización y hacer más ágil las reglas esenciales del negocio.

En cuanto a la metodología, se hizo uso de *Extreme Programming* para mantener un correcto control en cada una de las actividades, con tiempo y organización de la mano de las historias de usuario. Dicho proyecto es relevante para la investigación, principalmente, por ser una solución que hace más ágil el manejo del negocio y por la inclusión de la metodología ágil *Extreme Programming*; tampoco se dejan de lado ni las herramientas de desarrollo ni la parte de base de datos que representan importantes bases para considerar. Otra arista que resalta es el empleo de módulos para el manejo de inventarios; esto por similitudes con el tema tratado en esta investigación.

### ***Antecedentes nacionales***

Como primer antecedente nacional se toma el trabajo de Álvarez y Ramírez (2013) realizado para optar por el grado académico de Licenciatura en Informática con énfasis en Sistemas de Información. Este proyecto planteó una propuesta para automatizar y agilizar los principales procesos de la clínica de odontología; es un intento de llevar a cabo una herramienta informática personalizada al caso, en este sentido, tiene el mismo enfoque que la investigación desarrollada en este documento.

El proyecto se realizó por medio de una interfaz fácil de usar, mejorando la organización de la información y las labores necesarias para un adecuado y ágil manejo de los datos. Este

trabajo significa un gran insumo en la medida en que incluye ciertos aspectos importantes que son considerados en la solución desarrollada a favor de la Veterinaria Pulguitas.

Como segundo antecedente nacional, se toma el trabajo de Gamboa y Herrera (2012) realizado para optar al grado de Licenciatura en Informática con énfasis en Sistemas de Información. Este proyecto consistió en la elaboración de un sistema que solventa la gestión y control de información correspondiente a los recursos humanos, procesamiento de datos e información que compete a la gestión del personal, sus actividades y procesos relacionados. La cual, previo a esta solución, se manejaba en archivos de Excel haciendo difícil y descentralizada la consulta, lo cual tiene similitudes con el tema del trabajo en estudio.

Para el desarrollo, se hizo uso del IDE Visual Studio y como lenguaje de programación se empleó C#, entre otras herramientas. Ahora bien, en lo referente al motor de base de datos, se utiliza Microsoft SQL Server.

Este proyecto resulta importante para la investigación en curso porque las herramientas de programación son de gran valor. Su uso se debe considerar para desarrollar una solución informática de esta índole, así mismo presenta una estructura modular para llevar a cabo las funciones correspondientes al mantenimiento y control de la información, de la cual puede echarse mano para hacer ágil la experiencia del usuario en la visualización de datos pertinentes.

Como tercer antecedente nacional, se toma el trabajo de Bustamante (2021) realizado para optar al grado de Licenciatura en Informática con énfasis en Sistemas de Información. Dicho proyecto se pensó como una solución informática para el área de audiovisuales en la Sede Regional Brunca de la UNA, ya que en ese momento se generaba todo manualmente, lo que

posibilita la generación de errores por parte de la administración del departamento (lo que igualmente sucede para el caso de la investigación en estudio).

El problema se centra en que la biblioteca cuenta con el mecanismo tradicional de utilizar boletas para el control de los equipos y salas que prestan. Por lo que el objetivo de la solución es que tanto los interesados en préstamos como los encargados puedan tener una visualización adecuada de los datos.

La investigación recurre al uso de la metodología en cascada, en la que no se puede avanzar a la siguiente fase, si la anterior no se encuentra totalmente terminada. Este tercer antecedente es útil para la investigación por la forma en que se aborda la solución a nivel de programación y de requerimientos. Por lo visto hasta este punto, las herramientas de desarrollo son muy similares, por lo que un análisis y revisión de experticia es el criterio más importante para tomar consideraciones sobre el tema.

## **Proyecciones**

### *Alcances*

Basado en los objetivos específicos, se busca entregar una solución que permita gestionar las citas en la pyme Veterinaria Pulguitas. Por lo tanto, a continuación, se enumeran los puntos importantes:

1. Es relevante señalar que la solución responde estrictamente a este caso particular, por lo que esta delimitación va de la mano con el objetivo específico uno, es decir, está sujeta a las necesidades del caso en cuestión.
2. Asimismo, se pretende entregar una solución tomando las consideraciones técnicas y metodológicas investigadas y planteadas desde el objetivo dos. Se busca crear una solución con una interfaz amigable y fácil de usar.
3. Como se indica en el objetivo específico tres, se entregará una base de datos que centralice la información referente a los clientes, pacientes, usuarios, tratamientos y citas de la veterinaria, por lo tanto, se hace la salvedad de que el manejo de esta información queda exclusivamente a cargo del negocio Veterinaria Pulguitas.
4. Según lo indicado en el objetivo cuatro, se pretende desarrollar un módulo de ingreso al sistema que autentique el rol del usuario, un módulo para agendar citas, un módulo para ingresar nuevos usuarios, un módulo para el manejo de clientes, un módulo para el manejo de pacientes, un módulo para agregar valoraciones de citas activas y para modificar valoraciones de citas cerradas que, por algún motivo, se hayan cerrado con datos erróneos, finalmente, un módulo para agregar tratamientos.

Finalmente, no se contempla realizar capacitación del personal de la veterinaria. También es importante indicar que el soporte de la aplicación no está dentro de los alcances y, una vez entregado, corre por cuenta del local.

### ***Limitaciones***

Dentro de las limitaciones, inicialmente se tendría que considerar la disponibilidad de infraestructura, ya que, con base en una revisión inicial, se puede indicar que se cuenta con poca infraestructura tecnológica, de ahí que en lo referente a recursos tecnológicos se encuentra la limitante.

Por último, en lo que se refiere a licencias, es importante señalar que no se cuenta con el presupuesto para adquirir *software* propietario, por lo que, para la solución en estudio, es necesario recurrir a opciones gratuitas.

## **Capítulo II. Marco teórico**

## Historia de la programación

En el artículo de Engloba Tech (n.d), se explica que los inicios de la programación datan de inicios del siglo XIX, esto gracias a la creación, por parte de Joseph Marie Jacquard, de tarjetas perforadas utilizadas principalmente para controles telares mecánicos.

Sin embargo, el salto más grande inició en 1936 con la invención de la célebre Máquina de Turing, con el objetivo de resolver cualquier problema matemático. Ahora bien, existen muchos debates sobre cuándo surgió el primer lenguaje de programación real, empero, en aras de seguir la línea del tiempo, se tiene que el primer gran esfuerzo fue FORTRAN, el cual tuvo su nacimiento en 1957 creado por la empresa IBM y John Backus. Dos años después nace COBOL diseñado por CODASYL.

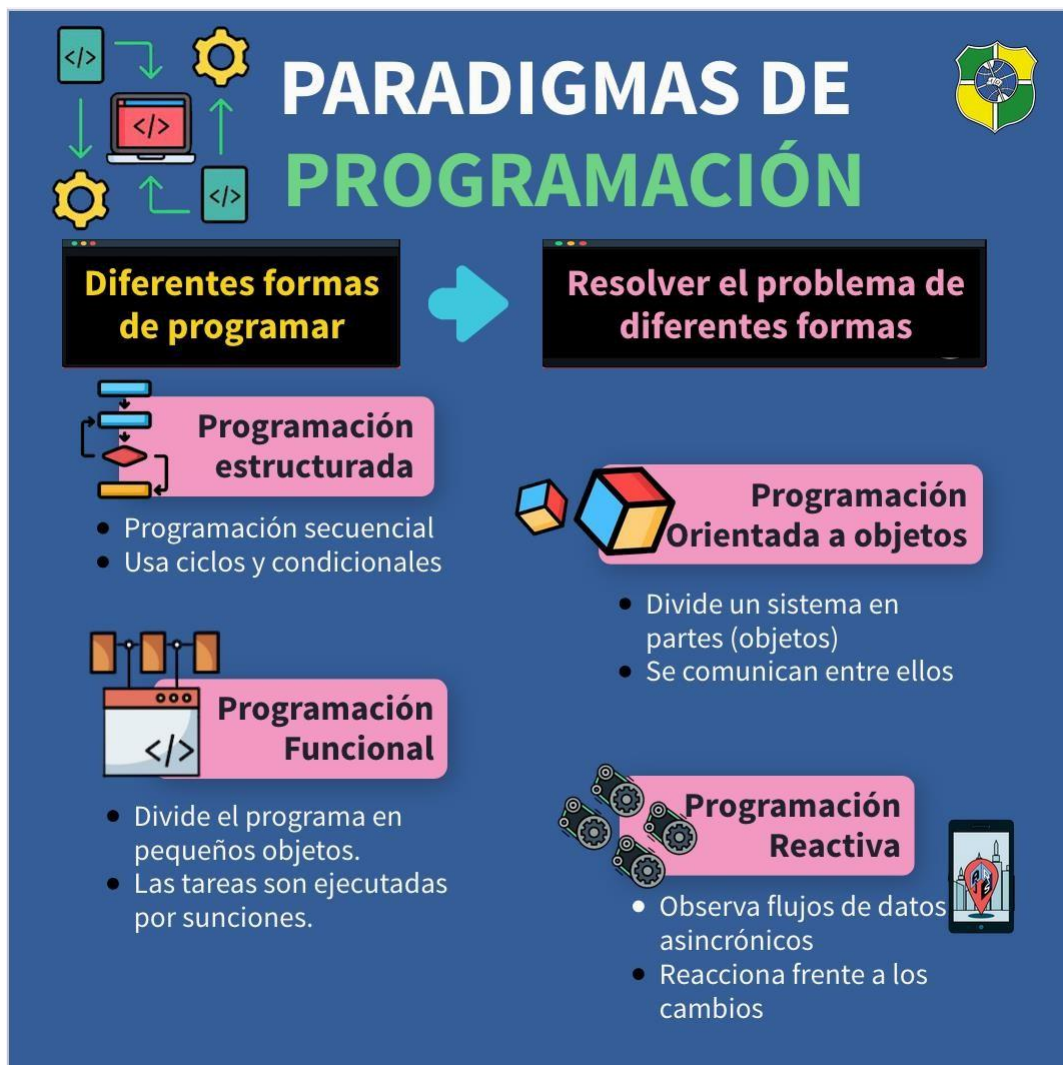
Ya para el año 1969, gracias a Dennis Ritchie, aparece C, lo cual representó un gran cambio. Seguidamente, Bjarne Stroustrup basado en C confecciona C++, cuyo uso es amplio desde el día de su creación; ya para el año 1987 aparece Perl, en 1991 Python y unos cuantos años más tarde empiezan a aparecer los lenguajes con más renombre: PHP, Java y JavaScript. Ya para el nuevo milenio nace C#, entre otros más que han permitido grandes avances a nivel de programación y, por ende, de la tecnología que se conoce ahora.

## Paradigmas de programación

Según Martínez (n.d), los paradigmas de programación ofrecen distintas metodologías para abordar diversos problemas computacionales. La figura 1 muestra una breve explicación sobre los paradigmas más conocidos.

### Figura 1

*Paradigmas de programación.*



Fuente: (Martínez, n.d)

### Programación orientada a objetos

Según UNIR (2023), la programación orientada a objetos (POO) es una forma abstracta que organiza código asemejándose a cómo las personas piensan y entienden el mundo real. La POO permite agrupar datos y operaciones relacionadas entre las diferentes entidades, estas mismas entidades son las que se denominan objetos.

Otro concepto importante por considerar es el de clase, cuya principal característica es la de ser un modelo abstracto a partir del cual se crean los objetos de acuerdo con su especificidad.

La figura 2 permite la comprensión de este paradigma.

**Figura 2.**

*Programación orientada a objetos.*



Fuente (Tecsify, n.d)

**Ventajas de la programación orientada a objetos.**

Algunas de las principales ventajas de la POO son las siguientes:

1. Reutilización de objetos en diferentes programas.
2. Modelado sencillo de estructuras simples.
3. Polimorfismo.
4. Facilidad de depuración.
5. Implementación segura de acceso por medio de encapsulamiento.
6. Herencia de atributos entre objetos.

### **Bases de datos**

Según describe Microsoft (n.d), una base de datos es una herramienta para recopilar y organizar información. Las bases de datos son contenedores de objetos, brindan una mejor visibilidad y manejo de datos para los casos donde ya es difícil la comprensión, el listado, las búsquedas o la extracción de subconjuntos de estos.

#### ***Tipos de bases de datos***

##### **Bases de datos relacionales**

Son aquellas que almacenan y organizan datos con relaciones definidas para un acceso rápido. Los datos se organizan en tablas que contienen información y son representadas mediante filas y columnas.

##### **Bases de datos distribuidas**

Son múltiples bases de datos lógicamente relacionadas, pero que se encuentran en diferentes espacios lógicos y geográficos.

### **Bases de datos NoSQL**

Son almacenes de datos que no usan SQL para las consultas, utilizan otros lenguajes para su construcción y acceso.

### **Bases de datos orientadas a grafos**

Usan un modelo basado en nodos y bordes, lo cual ayuda en la representación de datos interconectados similar a lo que hacen las redes sociales.

### **Bases de datos de autogestión**

Utilizan tecnología basada en nube y *machine learning* para automatizar muchas de las tareas rutinarias necesarias para gestionar bases de datos (ORACLE, n.d).

### **SQL**

SQL, por sus siglas, lenguaje de consulta estructurada, es un lenguaje de programación que sirve para almacenar y procesar información en base de datos relacional. Se inventó en la década de los setenta con base en el modelo de datos relacional. Oracle fue el primer proveedor en ofrecer un sistema comercial basado en SQL. (AWS, n.d)

### **DBMS**

DBMS, por sus siglas, sistema de gestión de base de datos, es un sistema que permite gestionar o administrar los datos en un sistema computacional. Su función se centra en permitir a los usuarios la gestión, almacenamiento y uso de los datos para que puedan ser accedidos y usados según convenga (Rouse, 2024).

## Visual Studio











Microsoft Visual Studio IDE es una plataforma que se puede utilizar para editar, depurar y compilar código; incluye compiladores, herramientas de completado de código y muchas más funciones para mejorar el proceso de desarrollo de *software* (Microsoft, n.d). Además, soporta lenguajes como C#, C++, C, visual Basic, entre otros.

## C#

Es un lenguaje de programación moderno, innovador, de código abierto y multiplataforma orientado a objetos (Microsoft, n.d). A continuación, en la figura 3, se muestra un *ranking* de los lenguajes de programación más utilizados según TIOBE (2024).

### Figura 3

#### *Ranking de lenguajes de programación 2024*

Jun 2024	Jun 2023	Change	Programming Language	Ratings	Change
1	1		 Python	15.39%	+2.93%
2	3	▲	 C++	10.03%	-1.33%
3	2	▼	 C	9.23%	-3.14%
4	4		 Java	8.40%	-2.88%
5	5		 C#	6.65%	-0.06%
6	7	▲	 JavaScript	3.32%	+0.51%
7	14	▲▲	 Go	1.93%	+0.93%
8	9	▲	 SQL	1.75%	+0.28%
9	6	▼	 Visual Basic	1.66%	-1.67%
10	15	▲▲	 Fortran	1.53%	+0.53%

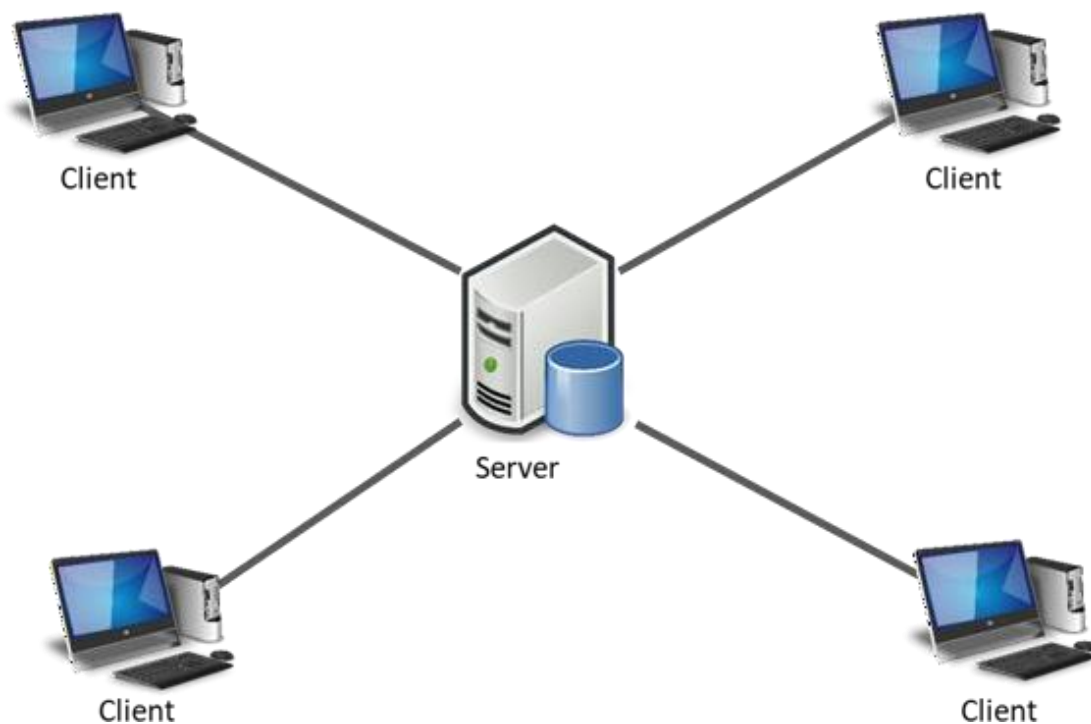
Fuente: (TIOBE, 2024)

## Arquitectura cliente-servidor

La arquitectura cliente-servidor es uno de los estilos arquitectónicos más distribuidos (Blancarte, n.d). Está conformado por dos componentes principales: el proveedor y el consumidor; en resumidas cuentas, el proveedor es un servidor que brinda un servicio y el cliente es quien consume dicho servicio.

### Figura 4

*Ejemplo de cliente-servidor*



Fuente: (Blancarte, n.d)

### Ventajas de utilizar cliente-servidor

1. Fácil de instalar el cliente, por lo general, es una aplicación simple.

2. Hay separación de responsabilidades, la lógica de negocio se encuentra de forma separada (Arsys, n.d).
3. Mejora la escalabilidad, ambos entornos pueden ser actualizados por separado.
4. Facilita la administración de los datos y la seguridad.

## **Scrum**

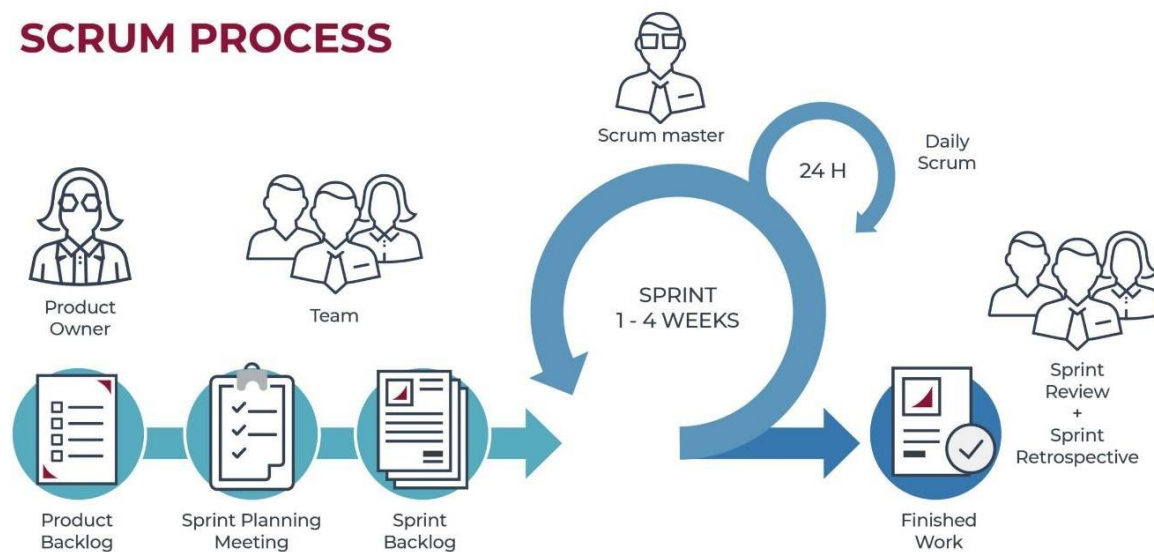
En el siguiente apartado, se define lo que es Scrum, así como algunos de los elementos importantes que la conforman y la hacen una de las metodologías más influyentes en la administración de proyectos. Según Atlassian (n.d), Scrum es un marco de gestión de proyectos de metodología ágil que ayuda a los equipos a estructurar y gestionar el trabajo mediante un conjunto de valores, principios y prácticas.

Scrum es tan versátil que no solo es posible aplicarlo a equipos de desarrollo de *software*, sino también aplicarlo a diferentes tipos de trabajo en equipo. Incluye un conjunto de reuniones, herramientas y funciones que de manera coordinada ayudan a los equipos a estructurar y gestionar el trabajo.

## **Figura 5**

*Proceso de Scrum en los proyectos*

## SCRUM PROCESS



Fuente: (Departamento de Comunicación UEMC Business School, 2022)

### *Roles en Scrum*

Es importante que los miembros sean capaces de organizarse de forma multifuncional y autodidacta, cada uno de ellos es responsable de ciertas tareas y se espera que las finalice según los tiempos acordados. Los tres roles principales en un equipo de trabajo Scrum son:

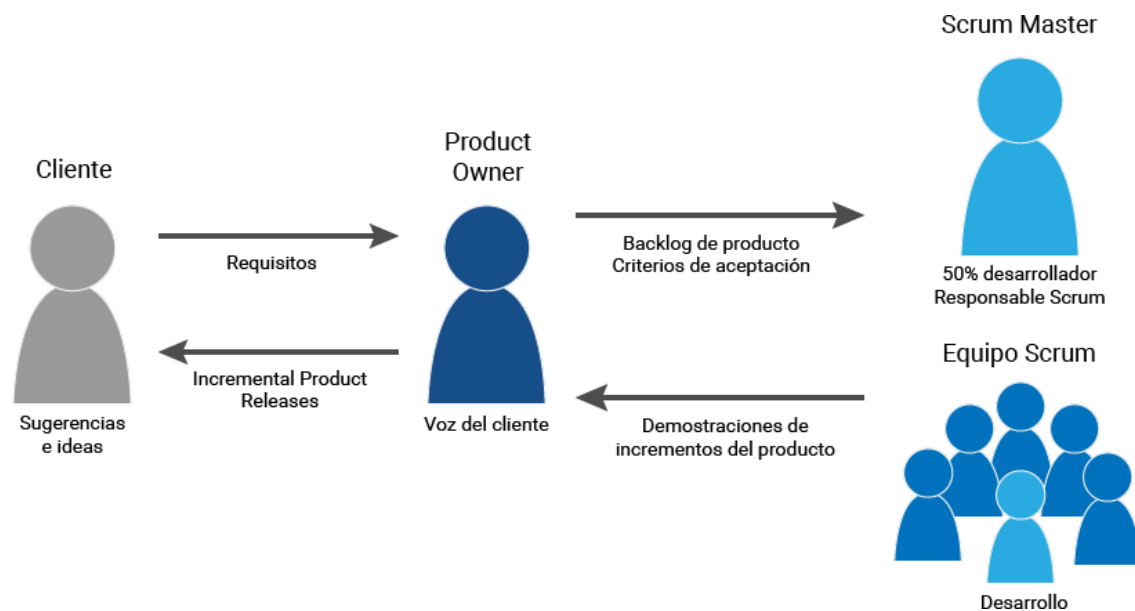
Propietario de producto o *product owner*: se encarga de que todos los miembros del equipo estén alineados desde la perspectiva del negocio. Es el responsable de comunicarse continuamente con el cliente, debe conocer el negocio y el mercado lo mejor posible.

Facilitador o Scrum máster: se encarga del cumplimiento y entendimiento de las reglas del marco Scrum, en este sentido, es el especialista en la metodología dentro del equipo.

Equipo de desarrolladores: lo conforman los profesionales, lo ideal es que lo integren de 3 a 9 personas, las cuales se ocupan de realizar las tareas de forma autoorganizada y multifuncional, quienes disponen de las habilidades para realizar diseños, pruebas, análisis, documentación, etc.

## Figura 6

### *Roles principales en Scrum*



Fuente: (Desire, n.d)

### *Artefactos de Scrum*

#### ***Product backlog (agenda del producto)***

Es básicamente la lista completa de funcionalidades que quedan por agregar al producto.

El *product owner* es quien lo prioriza para que el equipo trabaje en las características más importantes en cada momento. Además, va de la mano de las historias de usuario.

### ***Sprint backlog***

En el primer día de un *sprint*, durante la reunión de planificación, los miembros del equipo crean el *sprint backlog*; el cual secciona las tareas que el equipo realizará en el *sprint* en cuestión que va de la mano con lo especificado en el *product backlog*.

### ***Burndown chart***

Este se divide en la del *sprint* y la de las entregas, muestran el trabajo que queda en un *sprint* o bien cuánto queda para finalizar una entrega. Así mismo, ayuda a verificar si se está cumpliendo la planificación.

### ***Eventos Scrum***

Según Deloitte (n.d), Scrum se maneja a través de cinco eventos marcados para el control de sus procesos. La razón de ello es mantener los puntos mínimos necesarios para facilitar que el control empírico de procesos funciona.

1. ***Sprint planning***: es una reunión que se realiza al comienzo de cada *sprint* donde participa todo el equipo. El objetivo es revisar el *product backlog* donde el quipo selecciona los ítems que componen el *sprint backlog*.
2. ***Daily Scrum***: es una reunión diaria de 15 minutos en la que se participa exclusivamente el equipo de desarrolladores. Cada una de las personas del equipo responden las siguientes preguntas:
  - a. ¿Qué hice ayer para contribuir al *sprint goal*?

- b. ¿Qué hice ayer para contribuir al *sprint goal*?
  - c. ¿Qué hice ayer para contribuir al *sprint goal*?
3. ***Sprint review***: es la reunión que tiene lugar al final del *sprint*, el *product owner* y el equipo desarrollador presentan a los *stakeholders* el incremento terminado para que lo inspeccionen.
  4. ***Sprint retrospective***: la retrospectiva ocurre al final del *sprint*, precisamente después del *sprint review*. El objetivo es reflexionar sobre el último *sprint* e identificar posibles mejoras para el próximo.
  5. ***Sprint grooming o refinement***: es una práctica recomendada para verificar que el *product backlog* esté preparado. Antes de la reunión, es importante que todos conozcan los requerimientos o historias de usuario que van a tratarse.

### ***Valores Scrum***

Según Scrum México Expertos en Agilidad (n.d), estos son los valores que Scrum toma como pilares a la hora de desarrollarse como metodología.

#### **Foco**

Todos se centran en el trabajo del *sprint* y los objetivos del equipo de desarrollo.

#### **Apertura**

El equipo de desarrollo y sus partes interesadas acuerdan ser abiertos con todo el trabajo y con los desafíos de su realización.

**Respeto**

Los miembros del equipo se respetan mutuamente para ser personas capaces e independientes.

**Valor**

Los miembros del Scrum Team tienen el coraje de hacer lo correcto y trabajar en problemas difíciles.

**Compromiso**

Las personas se comprometen personalmente a alcanzar los objetivos del equipo de desarrollo.

**Historias de usuario**

Según Rehkopf (n.d), una historia de usuario es, en resumen, una explicación general de una función de *software* escrita desde la perspectiva del usuario final. Ayudan a proporcionar un marco centrado en el usuario para el trabajo diario, lo que impulsa la colaboración y mejora el producto en general.

**Figura 6**

*Ejemplo de historia de usuario.*

Historias de usuario				
ID. de la	Rol	Característica /	Razón /Resultado	Criterio de aceptación
	Como <Rol>	Quiero <evento>	Para <finalidad>	
	Como <Rol>	Quiero <evento>	Para <finalidad>	
	Como <Rol>	Quiero <evento>	Para <finalidad>	

### ***Beneficios claves de las historias de usuario***

1. **Acuerdo mutuo:** permiten entender y averiguar las necesidades del usuario además de entender todos los puntos de vista dentro del proyecto.
2. **Recoger el interés de un grupo:** es una representación de interés de un gran número de personas en una sola historia, además, una historia de usuario puede recoger el interés de un grupo.
3. **Todos en una misma dirección:** facilita el entendimiento entre personas, simplifica en gran medida el proceso de toma de decisiones.
4. **Agiliza procesos en el desarrollo de Scrum:** en la tarea de agilizar procesos es viable valerse de las historias de usuario precisamente con el fin de agilizar el desarrollo de los proyectos.

### ***Factores por considerar en la confección de historias de usuario***

Según Miro (n.d), los siguientes puntos son importantes de considerar a la hora de diseñar las historias de usuario:

1. Identificar al usuario final.
2. Define la meta u objetivo del usuario final.
3. Describe el resultado deseado para el usuario final.
4. Colaborar con otros equipos implicados.
5. Priorizar las historias de usuario en *backlog* del producto.

## Git

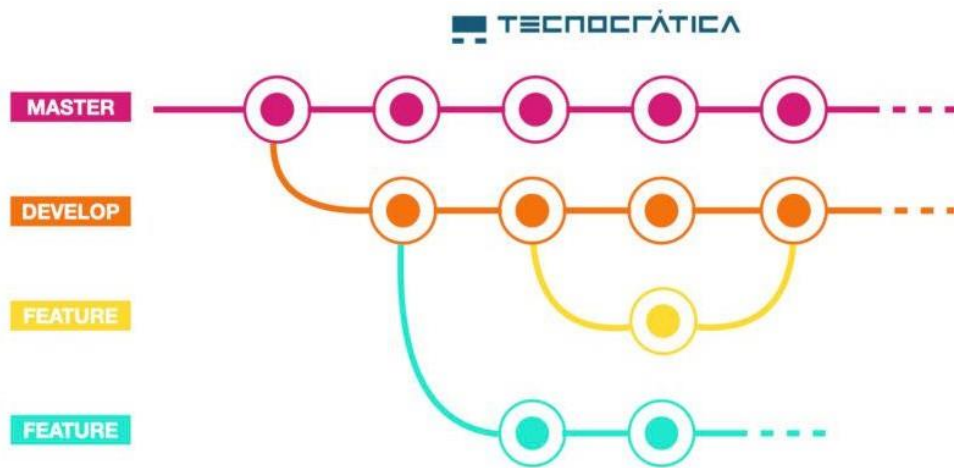
De acuerdo con Atlassian (n.d), Git es un proyecto de código abierto maduro y con un mantenimiento activo que desarrolló originalmente Linus Torvalds, el famoso creador del Kernel del sistema operativo Linux, en 2005.

Git presenta una arquitectura distribuida, es un ejemplo de DVCS (sistema de control de versiones distribuido, por sus siglas en inglés). En lugar de tener un único espacio para todo el historial de versiones del *software*, como sucede de manera habitual en los sistemas de control de versiones antaño populares, como CVS o Subversion (también conocido como SVN); en Git, la copia de trabajo del código de cada desarrollador es también un repositorio que puede albergar el historial completo de todos los cambios.

Estos espacios de trabajos son los que se conocen como ramas, donde cada uno de los equipos o involucrados trabajan. Una vez se tiene una versión definitiva, pueden hacer una solicitud *merge* para incorporarla al *branch* principal denominada máster.

## Figura 7

## Distribución de trabajo en ramas



Fuente (Tecnocrática, 2024)

## Sistemas de información

De acuerdo con la descripción hecha por Mesquita (2019), el objetivo de los sistemas de información es entender y analizar cómo ocurre el impacto de la adopción de las tecnologías de información en los procesos de decisión gerenciales y administrativos de las empresas.

El sistema de información puede trabajar con diversos elementos. Entre ellos están *software*, *hardware*, base de datos, sistemas especialistas, sistemas de apoyo a la gerencia, entre otros. Algunas de las características a la hora de integrar un sistema de información y que deben ser consideradas son las siguientes:

1. **Relevancia:** el sistema debe generar información relevante y necesaria según sea su finalidad.

2. **Integración:** es necesario considerar la integración entre el sistema información y la estructura del negocio.
3. **Control:** los sistemas de información pueden tener herramientas de control interno para garantizar que las informaciones generadas son confiables y los datos son controlados.
4. **Directrices:** se garantiza que los objetivos del negocio serán solucionados de manera objetiva, eficiente y directa.

### *Tipos de sistemas de información*

**ERP:** también llamados planeamiento de recursos de la empresa, son sistemas que integran diferentes procesos y datos conjuntándolos en solo lugar. Ayudan a traer más agilidad a los procesos.

**CRM:** también llamados gestión de relación con el cliente, son aquellos que automatizan todas las funciones relativas al contacto con los clientes y las preferencias de estos. El objetivo es responder a las necesidades de los consumidores de forma anticipada.

**SCM:** también llamados administración de la cadena de suministros, son aquellos que integran los diferentes procesos relativos a los proveedores de servicio.

**SIG:** conocidos como sistemas de información gerenciales, son aquellos dirigidos hacia el apoyo, a la toma de decisiones y que actúan a nivel estratégico, operacional y táctico.

## Windows Form

Según Microsoft (2023), Windows Form es un marco de interfaz de usuario para compilar aplicaciones de escritorio de Windows. Facilita la creación de aplicaciones de escritorio basadas en el diseño proporcionado por Visual Studio. Con Windows Form se puede desarrollar aplicaciones enriquecidas gráficamente que son fáciles de implementar, actualizar y con las que se puede trabajar con o sin conexión a internet.

## UML

De acuerdo con Lucidchart (n.d), UML fue creado para forjar un lenguaje de modelado visual común y semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de *software* complejos, tanto en estructura como en comportamiento. Con UML se describen los límites, la estructura y el comportamiento del sistema y los objetos que contiene; de igual manera, guarda relación directa con el análisis y el diseño orientados a objetos.

## Requerimientos funcionales y no funcionales

**Requerimientos funcionales:** según Medium (n.d), son declaraciones de los servicios que brindará el sistema, la forma que responderá a determinadas entradas, estos también pueden ser formados de forma negativa. Esto, siempre y cuando, sean una respuesta funcional al usuario u otro sistema. La especificación de los requerimientos funcionales ha de ser completa y coherente, en otras palabras, que los servicios solicitados estén bien definidos y que no sean contradictorios.

**Requerimientos no funcionales:** de acuerdo con lo dicho por Medium (n.d), son aquellos requisitos que se refieren directamente a las funciones específicas suministradas por el sistema,

por ejemplo, rendimiento, seguridad y disponibilidad. En resumen, no se refieren a lo que hace el sistema, sino al cómo lo hace. Estos se pueden dividir en los siguientes:

1. **Requerimientos del producto:** especifican el comportamiento del producto, por ejemplo, cantidad de memoria, requisitos de rendimientos, requisitos de fiabilidad, etc.
2. **Requerimientos organizativos:** están sujetos a políticas y procedimientos en la organización del cliente, por ejemplo: estándares en los procesos por utilizar, requisitos de implementación como lenguajes de programación y requisitos de entrega.
3. **Necesidades externas:** derivados de factores externos al sistema y al proceso de desarrollo,; ejemplo de esto puede ser requisitos de interoperabilidad, requisitos legales e incluso éticos.

## Figura 8

*Ejemplo de plantilla de requerimientos funcionales y no funcionales*

ID. de la historia	Descripción del requisito	Requerimientos funcionales y no funcionales			Entregables (EDT)	Interesado	Nivel d prioridad
		Criterios de aceptación	Nivel de complejidad	Nececidad, oportunidades u objetivos de negocio			
	Disponición di requisito	Criterio: Aceptación:	Nivel « Complejidad:	Digitalización de los porocesos de negocio			
	Como «Rob»	Quiero «evento»	Para «finalidad»				
	Como «Rob»	Quiero «evento»	Para «finalidad»				

## **Metodologías de desarrollo de software**

### ***Modelo incremental***

Este modelo, según plantea Pérez (2016), tiene como objetivo un incremento progresivo de la funcionalidad, en otras palabras, el producto va evolucionando con cada entrega hasta llegar al punto en que se amolda a lo que requiere el cliente. Se establecen entregas parciales mediante plazos; el punto es que, en cada una de las entregas, se debe mostrar una evolución respecto de la entrega anterior. Además, los responsables del proyecto deben estar constantemente analizando si los resultados son los esperados.

Dentro de esta metodología, hay iteraciones que no son independientes, sino que se encuentran vinculadas respecto de anteriores entregas. Así mismo, en esta metodología se pueden identificar al menos siete fases:

1. Requerimientos.
2. Definición de las tareas y las iteraciones.
3. Diseño de los incrementos.
4. Desarrollo del incremento.
5. Validación de incrementos.
6. Integración de incrementos.
7. Entrega del producto.

### ***Modelo en cascada***

Este modelo, de acuerdo con lo que propone Crehana (2022), se centra en un enfoque metodológico que consiste en ordenar de forma lineal las distintas etapas que se deben seguir al momento de desarrollar *software*.

Cada una de las tareas en esta metodología se dividen y se llevan a cabo de acuerdo con las fases de su proceso; esto se hace de manera escalonada, siguiendo una secuencia ordenada desde la primera hasta la última etapa. Dentro de este modelo, no se puede iniciar la fase n hasta que se haya concluido la fase n-1, además, los posibles errores se corrigen antes de avanzar.

En esta metodología, originalmente, se pueden identificar al menos siete fases:

1. Análisis de requisitos.
2. Diseño de sistema.
3. Diseño del programa.
4. Codificación.
5. Pruebas.
6. Implementación o verificación del programa
7. Mantenimiento.

### ***Modelo XP***

Este modelo denominado *Extreme Programming* (Sydle, 2023) es una metodología, cuyo fin es crear sistemas de alta calidad, basados en una estrecha interacción con los clientes, pruebas constantes y ciclos de desarrollo cortos.

Lo que se busca son ciclos de entrega rápidos donde hay continuas entregas que pretenden conseguir los resultados esperados por el cliente. Se basa en buenas prácticas y etapas predefinidas para que el proceso tenga máxima efectividad; esto con ciclos semanales, reuniones periódicas entre el cliente y el equipo de desarrollo.

XP se caracteriza por contar con los siguientes valores:

1. Comunicación
2. Simplicidad
3. *Feedback*
4. Coraje
5. Respeto

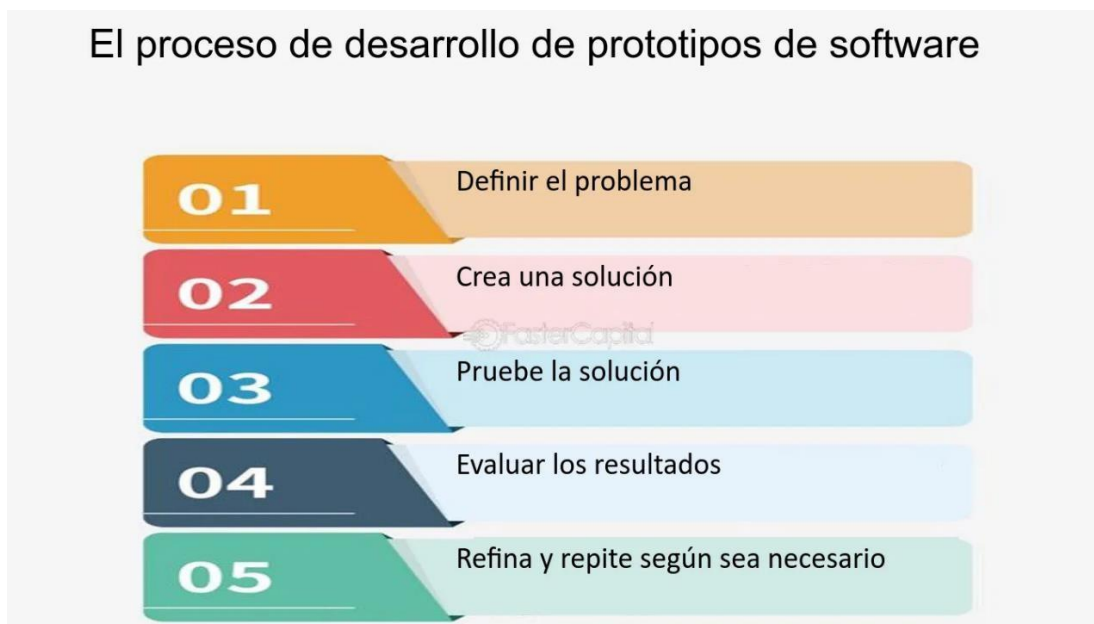
### **Tipos de prototipo**

Desde el campo de la ingeniería de *software*, hablar de prototipos es algo cotidiano, también esencial a la hora de tenerlo en cuenta en el proceso de desarrollo. Y esto debido a que, ciertamente, un prototipo de *software* es un modelo de preproducción de una aplicación o sistema de *software* (Faster Capital, 2024). Esto ayuda en la labor de probar viabilidad, medir aprobación

y reafirmar o confirmar lo esperado en el diseño. Igualmente, un prototipo permite la experimentación y posibilita la evaluación sobre cómo se está abarcando un problema y si es o no la forma más oportuna de hacerlo.

### Figura 9

*El proceso de desarrollo de prototipos de software*



Fuente (Faster Capital, 2024)

### ***Prototipos funcionales***

A menudo, son representaciones sencillas de una idea o concepto, pueden contener unas pocas pantallas con el fin de ver cómo está la funcionalidad de la interfaz o algún otro tipo de comportamiento. Este tipo de prototipos permiten probar las bases antes de gastar más tiempo en algo que, de acuerdo con la perspectiva del usuario, puede no satisfacer su necesidad.

### ***Prototipos de experiencia de usuario***

Este tipo de prototipo funciona para que el usuario interactúe con el producto. Utilizan *wireframes* y maquetas que ejemplifican cómo los usuarios navegan por la interfaz y qué tipo de interacción están proponiendo. Se espera que sea lo suficientemente realista para mostrar una experiencia prácticamente real.

### ***Prototipos de procesos comerciales***

Este tipo de prototipo involucra más la aprehensión del flujo de tareas para llegar a completar una tarea específica, es decir, conocer los procedimientos para llegar a un objetivo. Esta tarea puede involucrar el diseño de diagramas de flujo o cualquier otro similar que muestre cuál secuencia es la que se debe seguir para completar un objetivo o tareas. De acuerdo con lo dicho, puede notarse que facilitan el entendimiento necesario para completar determinada tarea.

### **Tipos de algoritmo**

Según lo caracteriza Maluenda (n.d), es un conjunto de instrucciones definidas, ordenadas para resolver un problema, en otras palabras, es un procedimiento que, dada una serie de orden de pasos, busca conseguir un fin.

Si bien los algoritmos están muy ligados a la computación, lógica o matemáticas, existen numerosos ejemplos cotidianos que pueden ser interpretados bajo la definición anterior; por ejemplo, una receta de cocina, manuales de instrucciones, etc.

Un algoritmo está conformado por tres partes:

1. ***Input*** (entrada): la información que inicialmente se brinda para trabajar.

2. **Proceso:** conjunto de pasos que, a partir de los *inputs*, busca llegar a una solución.
3. **Output (salida):** resultado final, a partir del manejo que se hizo de la información ingresada.

Todo algoritmo debe contar con las siguientes características para ser denominado como tal: debe ser preciso, sin ambigüedad; ha de ser ordenado, una secuencia clara y concisa que busca llegar a un fin. Deben ser finitos, es decir, un número determinado de pasos; deben ser concretos, o sea, son confeccionados para una situación específica y, finalmente, han de ser definidos, la misma respuesta para las mismas entradas.

Según Hostingplus (n.d), en programación existen tipos de algoritmos que pueden ser clasificados de acuerdo con cuatro categorías:

1. **Computacionales:** en este tipo la resolución depende de un cálculo matemático, por lo que puede ser resuelto por una computadora o calculadora.
2. **No computacionales:** son aquellos que no pueden ser resueltos por una computadora y requieren la intervención humana para dar con la solución.
3. **Cualitativos:** en estos no se realizan cálculos numéricos, sino secuencias lógicas o formales.
4. **Cuantitativos:** en estos sí se requiere de un cálculo numérico para poder llegar a una salida.

## Pruebas unitarias, integrales y globales

En todo proceso que lleve a cabo un desarrollo de *software*, lo oportuno es siempre realizar las pruebas pertinentes para garantizar que determinado producto hace lo que se supone debe hacer. Esto con el objetivo de prevenir errores y tener presente siempre la mejora, el rendimiento, etc.

Según IBM (n.d), algunas de los tipos de prueba de *software* son los siguientes:

1. **Pruebas de aceptación:** donde se verifica que el sistema funciona según lo que se espera.
2. **Revisión de código:** reafirmar que el *software* nuevo cumpla con los estándares de determinada organización y sigue las buenas prácticas.
3. Pruebas de integración se aseguran que los componentes del *software* funcionan en conjunto.
4. **Unit tests:** se asegura que las unidades de *software* funcionan de acuerdo con lo esperado.
5. **Pruebas funcionales:** emulan escenarios y se comprueban las funciones de acuerdo con estos escenarios.
6. **Pruebas de rendimiento:** comprobación del *software* de acuerdo con cargas de trabajo.

7. **Pruebas de seguridad:** validan que el *software* no está expuesto o si tiene algún tipo de vulnerabilidad.
8. **Pruebas de estrés:** se prueba cuál es el nivel de “estrés” que soporta efectivamente el sistema antes de fallar.
9. **Pruebas de usabilidad:** se prueba qué tan bien responde un sistema completando su tarea de acuerdo con las peticiones del cliente.

Es bien sabido que, si se cuenta con un plan de gestión de pruebas, se avala cuáles pruebas pueden aportar más en el proceso en general y, por ende, optimiza la entrega final garantizando un mejor producto.

### **Capítulo III. Marco metodológico**

**Marco metodológico**

De acuerdo con Azuero (2018), un marco metodológico refiere a un conjunto de acciones que tienen como principal objetivo describir y analizar a fondo el problema en cuestión. Esto se llega a realizar a través de diferentes métodos, algunos de ellos pueden ser observación y recolección de datos.

También se puede añadir, según Rivas (2022), que es la parte del escrito donde se busca argumentar, según métodos, procedimientos y limitaciones, cuál es el problema en específico y la manera en que es oportuno abordarlo. En este sentido, consta de una serie de pasos, técnicas y procedimientos que son utilizados para formular y resolver problemas; podría también añadirse que es el resultado de la aplicación de cada uno de los puntos expuesto en el marco teórico.

### ***Enfoque de la investigación***

En el enfoque de la investigación, tiene lugar un proceso sistemático, disciplinado y controlado. Se relaciona directamente con alguno de los dos métodos más comúnmente conocidos en todo proceso investigativo: cualitativo y cuantitativo (Eumed.net, n.d).

En el enfoque cualitativo, lo que se hace es recopilar y analizar datos no numéricos, para comprender opiniones o experiencias (Santander Universidades, 2021). Este tipo de investigación se centra en obtener una comprensión más profunda del tema de estudio de la mano de los sujetos.

Así mismo, la investigación cualitativa, de acuerdo con sus características, es más flexible e interpretativo que la investigación cuantitativa. Ofrece mucha información y su enfoque es descriptivo, por tanto, es un proceso inductivo.

Ahora bien, en la investigación cuantitativa hay más un enfoque hacia la recolección, análisis e interpretación de datos numéricos (León, n.d). Esto se produce a través de un proceso deductivo.

Dentro de este enfoque hay mediciones objetivas, análisis estadísticos, generalización y objetividad, como puede verse, la caracteriza la rigurosidad y, en este sentido, hay una conexión muy estrecha entre la observación empírica y expresiones matemáticas.

Es importante añadir que, además de los enfoques señalados, también existe el denominado como mixto. En algunas ocasiones, debido a las premisas de la investigación, es oportuno combinar ambas y mezclar elementos importantes de cada una; esto ciertamente favorece a una mejor comprensión del problema.

Este trabajo, en aras de una mejor comprensión del problema, se enfoca en un método mixto, pues ciertamente permite aprovechar las fortalezas de cada enfoque proporcionando una visión más provechosa. Además, ofrece un panorama más amplio para un análisis completo de los requerimientos del sistema en cuestión.

### ***Método de la investigación***

El método de investigación, según define Sánchez (n.d), es el conjunto de técnicas escogidas de acuerdo con la orientación del proyecto y que, según el uso de determinadas herramientas, permitirán la obtención de un producto.

En el caso particular de la Veterinaria, el insumo más importante es la recopilación de datos respecto del flujo de labores, por ejemplo: entrevistas sobre cómo manejan los registros, así como la experiencia de los colaboradores en este proceso. Asimismo, la observación es un método por medio del cual se espera recopilar información.

Para el proyecto en cuestión, se aplican todos los conocimientos obtenidos a través de la formación universitaria para confeccionar una herramienta programada capaz de llenar el problema que afronta la pequeña empresa. Así mismo, es importante añadir que se considera la utilización de un marco de trabajo ágil, en este caso, Scrum.

### ***Fuentes de información***

En cuanto a las fuentes de información, es más que evidente que son utilizadas para la localización e identificación que responde al problema en estudio. En este sentido, son cualquier instrumento o recurso que satisface la necesidad informativa (Valencia, n.d).

Ahora bien, como fuentes primarias, se toman tesis, libros, artículos, sitios web, documentación sobre implementaciones similares y también la información recolectada en la veterinaria sobre el manejo del problema. Por su parte, como fuentes secundarias, lo más útil para este caso es el uso de diversos sitios web que enriquecen el estudio y que son pertinentes. Es importante aclarar que cada una de las fuentes es contrastada, son fuentes relevantes y de credibilidad.

### ***Variables***

**Tabla 1**

Cuadro de variables

<b>Objetivo general</b>	<b>Objetivo específico</b>	<b>Variable</b>	<b>Dimensión</b>	<b>Indicador</b>	<b>Definición conceptual</b>
Desarrollar una solución	Realizar una revisión sobre los requerimientos	Requerimientos funcionales y no funcionales, historias de	Conocer cómo debe operar el sistema de	Manejo de clientes, pacientes, usuarios y	Reconocer el proceso de manejo de clientes,

<p>informática para la pyme</p> <p>Veterinaria</p> <p>Pulguitas a fin de buscar la optimización en</p>	<p>necesarios para el desarrollo de la solución a través de requerimientos funcionales, no funcionales, historias de usuario y casos de uso.</p>	<p>usuario y casos de uso.</p>	<p>acuerdo con las expectativas.</p>	<p>citas en la veterinaria.</p>	<p>pacientes, usuarios y citas.</p>
<p>el manejo de la consulta veterinaria dentro del local implementando una solución de escritorio a la medida.</p>	<p>Definir herramientas y metodologías analizando cuáles son las más convenientes para una solución efectiva y el óptimo desarrollo del sistema en cuestión.</p>	<p>Metodología ágil.</p>	<p>Reconocer cuál metodología se adecua mejor a las expectativas del proyecto.</p>	<p>Manejo de tareas y labores dentro del proyecto.</p>	<p>Reconocer de acuerdo con el proyecto cuál es la mejor de las opciones a nivel de metodología de proyectos.</p>
	<p>Establecer el diseño de la estructura de base de datos para el manejo de información dentro de la solución. tomando en cuenta normalización y buenas prácticas.</p>	<p>Diseño de base de datos.</p>	<p>Reconocer cuál es la estructura de información que maneja la veterinaria y adaptarla a una base de datos.</p>	<p>Manejo de la información de los clientes, pacientes, usuarios y citas.</p>	<p>Conocer la estructura y forma en que se maneja la información dentro de la veterinaria.</p>
	<p>Desarrollar los respectivos formularios para el</p>	<p>Desarrollo de formularios en</p>	<p>Reconocer a nivel técnico el flujo de acciones</p>	<p>Manejo del flujo de información</p>	<p>Reconocer los procesos involucrados a nivel de</p>

	cumplimiento de funcionalidades tomando en consideración la administración del sistema.	Windows Form con C#.	visualizado para el sistema.	entre los formularios.	funcionalidad entre formularios.
	Ejecutar pruebas en el sistema para la verificación de la correcta ejecución de las funcionalidades considerando las diferentes opciones para estos fines.	Plan de pruebas.	Reconocer los aspectos necesarios para medir el funcionamiento esperado.	Respuesta del sistema ante eventos esperados.	Evaluar un plan de pruebas para garantizar la calidad por medio de la ejecución de pruebas exhaustivas y efectivas antes de entrega final o lanzamiento.

### ***Instrumentos***

Para el caso de este proyecto, se contempla principalmente la entrevista como instrumento de medición, a fin de obtener la información requerida sobre la tarea de registro de citas en la veterinaria. Seguidamente, se considera la creación de un documento para cada requerimiento identificado, con el fin de cumplir con los objetivos del sistema.

### **Metodología ágil**

Según lo planteado por Garrido (2023), la metodología ágil es aquella que permite adaptar la forma de trabajo a las condiciones del proyecto, esto principalmente con el objetivo de encontrar flexibilidad y acomodar el proyecto a las circunstancias específicas del entorno.

### *Adaptación de la metodología*

Es sabido que la metodología Scrum está dividida en fases, donde ciertamente hay una división de actividades que posibilitan el desarrollo de determinado trabajo. Para los fines de este proyecto, se contempla la siguiente tabla con las fases de Scrum que se siguen en el desarrollo del sistema de citas para la veterinaria.

**Tabla 2**

#### *Fases Scrum*

FASE	PROCESO	ENTREGABLE
<b>Análisis</b>	Crear idea general del proyecto	Idea general del proyecto
	Identificar al Scrum Máster y <i>stakeholder</i>	Equipo Scrum
	Formar equipo Scrum	
	Realizar planificación del <i>sprint</i>	Sprint “0”
	Definir el modelado del sistema	
	Confeccionar historias de usuario	Historia de usuario
	Estimar historias de usuario	
<b>Desarrollo</b>	Crear <i>sprint backlog</i>	<i>Product backlog</i>
	Crear entregables	
	Realizar reuniones del <i>sprint</i>	

	Descripción de artefactos	
--	---------------------------	--

### ***Fases***

El desarrollo de este proyecto intenta otorgar una herramienta por medio de la cual sea más ágil y fácil de almacenar la información de los clientes que visitan la Veterinaria Pulguitas. Esta solución informática se propone como un sistema que centralice la información de los clientes, pacientes veterinarios, usuarios de sistema, tratamientos y el manejo de citas. Estas informaciones deben estar disponibles para cuando sea requerido.

Para lograr lo anteriormente dicho, es preciso lo siguiente:

1. Conocer cómo se recolecta la información, así como qué tipo de información de los pacientes y clientes es almacenada, todo esto con el fin de identificar información clave y reconocer una visión general del manejo de esta, y así contemplarlo en la solución informática.
2. Entrevistar al encargado de realizar las citas para saber cómo se recolecta la información de los clientes y pacientes.
3. Desarrollar la solución informática y definir la estructura de base de datos de acuerdo con la necesidad específica de la veterinaria.

### ***¿Qué beneficios ofrece la utilización de esta metodología?***

Scrum plantea un marco de trabajo flexible y adaptable, en este sentido, el proyecto puede ser redirigido de acuerdo con nuevas prioridades y con esto dar margen a una gestión

regular de las expectativas del cliente. Asimismo, desde las primeras iteraciones, se puede manejar los problemas, ya que posibilita una mitigación de riesgos de forma anticipada (Proyectos agiles.org, 2022). Dicho lo anterior, es palpable que la utilización de una metodología ágil como Scrum se da pie a un mejor escenario de productividad y calidad.

## **Capítulo IV. Análisis de resultados**

**Plan de gestión de riesgos**

En un plan de riesgos se busca identificar, analizar y mitigar los riesgos potenciales que podrían afectar el desarrollo y el éxito de un proyecto. Es importante señalar que este debe ser viable y debe ayudar a anticipar problemas, tomar decisiones informadas y aumentar las

posibilidades de alcanzar los objetivos establecidos. En los siguientes puntos, se muestran los riesgos existentes en el proyecto en cuestión.

### ***Identificación y control de riesgos***

En la identificación de riesgos se hace uso de una matriz de riesgos con el fin de clasificar cada uno de ellos de acuerdo con una descripción, clasificación, ponderación y probabilidad e impacto en el proyecto.

### ***Clasificación de los riesgos***

Se plantean bajo la siguiente clasificación:

- **Recurso:** cuando un recurso puede ser afectado.
- **Tiempo planificado:** cuando el proyecto puede verse afectado por alguna actividad que no sale de acuerdo con la estimación de tiempo, lo cual produce un aumento en el tiempo de la actividad.
- **Requerimientos:** cuando se puede afectar el desarrollo del proyecto debido a que es necesario para el avance.
- **Equipo de trabajo:** cuando miembros del equipo impiden el correcto flujo de trabajo.
- **Aceptación:** cuando la aceptación de alguna parte del proyecto se ve amenazada por algún factor en medio.

***Ponderación de los riesgos***

Los riesgos pueden recibir los niveles expuestos en la tabla 3.

**Tabla 3*****Ponderación de riesgos***

<b>Puntaje</b>	<b>Calificación</b>
<b>81-100</b>	Muy alto
<b>61-80</b>	Alto
<b>41-60</b>	Moderado
<b>21-40</b>	Bajo
<b>0-20</b>	Muy bajo

***Probabilidad de ocurrencia***

La probabilidad de ocurrencia se medirá de acuerdo con los puntos expuestos en la tabla 4.

**Tabla 4*****Probabilidad de ocurrencia***

<b>Calificación</b>	<b>Puntos</b>	<b>Descripción</b>
<b>Muy alto</b>	100	Riesgo muy altamente probable de ocurrir, entre 80% y 99%.
<b>Alto</b>	80	Riesgo altamente probable de ocurrir, entre 60% y 80%.
<b>Mediano</b>	60	Riesgo medianamente probable de ocurrir, entre 40% y 60%.

<b>Bajo</b>	40	Riesgo poco probable de ocurrir, entre 20% y 40%. El impacto al proyecto es bajo.
<b>Muy bajo</b>	20	Riesgo muy poco probable. El impacto al proyecto es casi nulo.

### *Impacto*

**Tabla 5**

#### *Calificación del impacto*

<b>Calificación</b>	<b>Puntos</b>	<b>Descripción</b>
<b>Muy alto</b>	100	Impacto en el proyecto sobre la mitad de los recursos planeados en el área que se encuentre el riesgo.
<b>Alto</b>	80	Impacto alto en el proyecto, puede ocupar un 30% o 40% más de los recursos planeados en el área que se encuentre el riesgo.
<b>Mediano</b>	60	Impacto remarcado en el proyecto, puede ocupar un 20% más de los recursos planeados en el área que se encuentre el riesgo.
<b>Bajo</b>	40	Impacto muy pequeño en el proyecto, puede ocupar un 10% más de los recursos planeados en el área que se encuentre el riesgo.
<b>Muy bajo</b>	20	Impacto prácticamente insignificante para el proyecto. La magnitud es muy pequeña para ser cuantificada.

### *Matriz de riesgos*

Una matriz de riesgos es una herramienta de análisis utilizada para la visualización de riesgos, pero principalmente su uso se centra en la clasificación de los riesgos, identificación y priorización en función de su ocurrencia e impacto potencial dentro del proyecto. A continuación, en la tabla 6, se incluye la matriz de riesgos para el proyecto en cuestión.

**Tabla 6***Matriz de riesgos*

<b>Código</b>	<b>Riesgos</b>	<b>Clasificación</b>	<b>Impacto</b>	<b>Probabilidad</b>	<b>Acciones para mitigar el riesgo</b>
<b>RSK01</b>	Datos de los usuarios comprometidos	Requerimientos y equipo de trabajo	Muy alto	10.00%	Limitar acceso a la base de datos y permitir acceso restringido únicamente ligado al sistema.
<b>RSK02</b>	Fallos en la infraestructura tecnológica que podrían llevar a un mal funcionamiento del sistema	Recursos y requerimientos	Alto	30.00%	Realizar pruebas exhaustivas del sistema, considerar elegir tecnologías confiables.
<b>RSK03</b>	Tiempo de entrega del proyecto	Tiempo planificado	Modera do	30.00%	Utilizar metodología ágil para la planificación de actividades dentro del proyecto.

**Estudio de factibilidad**

En el proyecto en cuestión, se tiene como objetivo crear un sistema que permita registrar las citas veterinarias de los clientes, gestionar valoraciones sobre las citas, gestionar los tratamientos que se le pueden asignar a los animales y manejar todo lo referentes a los dueños de los animales, los pacientes veterinarios y los usuarios encargados de la veterinaria. Se busca agilizar el manejo de la información de la veterinaria que hasta el momento se ha manejado ya sea en Excel o de forma manual.

Antes que cualquier ejecución y análisis de un proyecto, es importante realizar un estudio de factibilidad para determinar la viabilidad y demás factores relacionados. De acuerdo con

Martins (2024), los estudios de factibilidad o también denominados estudios de viabilidad sirven para evaluar si el plan de proyecto puede salir bien o mal. En este sentido, permiten evaluar el proyecto a fin de determinar si se puede avanzar o no el proyecto en cuestión.

### *Análisis de factibilidad*

En este momento, el inconveniente principal de la veterinaria en cuanto a las citas es el no contar con una herramienta para llevar un control de registro de citas para todos los clientes y sus animales. Por lo que es necesario desarrollar un sistema que permita gestionar las citas de forma ágil, donde se almacena la información de estas, junto con los mantenimientos de los clientes y pacientes, además, que esté disponible para su revisión según sea conveniente.

El sistema propuesto resulta de beneficio en la medida en que ayuda a un manejo más sencillo, al tener la información centralizada con una interfaz intuitiva y sencilla de utilizar, contrario a llevarla de forma manual como actualmente se hace. Como se ha mencionado antes, dentro de los aportes del desarrollo en cuestión, se tiene lo siguiente:

- Centralización de la información.
- Sistema con interfaz intuitiva.
- Mantenimientos sobre clientes, pacientes, citas y tratamientos.
- Notificaciones por medio de correo electrónico de las citas agendadas.

### ***Factibilidad técnica***

Seguidamente, se exponen los recursos tecnológicos que son tomados en cuenta para la realización del proyecto.

#### **Base de datos SQL Server Express Edition**

**Tabla 7**

*Requerimientos de instalación SQL Server Express Edition*

<b>Recursos requeridos</b>		<b>Recursos de equipo de la Veterinaria Pulguitas</b>	
<b>Windows</b>	10, 11 o Windows Server 2016 64 bit	<b>Windows</b>	11
<b>Espacio en Disco</b>	6 GB	<b>Espacio en Disco</b>	1 TB
<b>Memoria RAM</b>	1 GB	<b>Memoria RAM</b>	16 GB

Se sugiere la base de datos SQL Server, principalmente, por ser un motor de base de datos de uso reconocido con respaldo de Microsoft y por la familiaridad para el equipo desarrollador.

Según DB-Engines, para el año 2024, se cuenta con la siguiente información:

**Tabla 8**

*Ranking de bases de datos*

Rank			DBMS	Database Model	Score		
Jun 2024	May 2024	Jun 2023			Jun 2024	May 2024	Jun 2023
1.	1.	1.	Oracle +	Relational, Multi-model	1244.08	+7.79	+12.61
2.	2.	2.	MySQL +	Relational, Multi-model	1061.34	-22.39	-102.59
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-model	821.56	-2.73	-108.50
4.	4.	4.	PostgreSQL +	Relational, Multi-model	636.25	-9.30	+23.43
5.	5.	5.	MongoDB +	Document, Multi-model	421.08	-0.58	-4.29
6.	6.	6.	Redis +	Key-value, Multi-model	155.94	-1.86	-11.41
7.	7.	↑ 8.	Elasticsearch	Search engine, Multi-model	132.83	-2.52	-10.92
8.	↑ 9.	↑ 11.	Snowflake +	Relational	130.36	+9.03	+16.23
9.	↓ 8.	↓ 7.	IBM Db2	Relational, Multi-model	125.90	-2.56	-18.99
10.	10.	10.	SQLite +	Relational	111.41	-2.91	-19.81

Fuente: (DB-Engines, 2024)

### Software de desarrollo Visual Studio Community 2022 17.3.4

Tabla 9

*Requerimientos de instalación Visual Studio Community.*

Recursos requeridos		Recursos de equipo de la Veterinaria Pulguitas	
<b>Windows</b>	10, 11	<b>Windows</b>	11
<b>Espacio en Disco</b>	850 MB- hasta 210 GB	<b>Espacio en Disco</b>	1 TB
<b>Memoria RAM</b>	4 GB	<b>Memoria RAM</b>	16 GB

Al ser esta una solución de índole cliente-servidor, para el desarrollo de esta solución, se va a utilizar C# como lenguaje de programación implementado a través de Windows Form, debido a la familiaridad del equipo desarrollador y porque C# se reconoce por ser un lenguaje robusto y confiable. Este se encuentra en el top 5 de lenguajes más utilizados a nivel mundial según se vio anteriormente.

### Factibilidad económica

El proyecto desarrollado se presenta como una solución para la Veterinaria Pulguitas y se erige como una donación, por lo que no conlleva gastos de mano de obra. En la realización de este se hace uso de *software* gratuito. En la tabla 10 se muestran algunos puntos importantes.

**Tabla 10**

*Ítems de costos del proyecto*

Ítem	Detalle	Justificación	Monto
1	Equipos de computo	La Veterinaria Pulguitas cuenta con los equipos necesarios.	0
2	Licenciamiento de aplicaciones	Para el desarrollo se utiliza <i>software</i> gratuito.	0
3	Mantenimiento del proyecto	No incumbe al desarrollo del proyecto.	0
4	Horas profesionales de desarrollo de solución informática.	Al ser una donación no incurre en gastos de programación.	0

### Factibilidad operativa

Con el fin de garantizar el éxito del proyecto, es de importancia seguir los siguientes puntos a nivel operativo para el correcto desarrollo de la solución:

- Plan de comunicación para informar a los *stakeholders* sobre los beneficios que se esperan con el sistema desarrollado.

- Evaluación de infraestructura tecnológica disponible para la correcta ejecución del sistema realizado.
- Explicar que, al ser una donación y al hacer uso de *software* gratis, no hay inversión financiera por parte de la veterinaria.
- Pruebas en sitio del sistema realizado.

## **Capítulo V. Conclusiones y recomendaciones**

## **Conclusiones**

- A partir de la problemática detectada en la veterinaria y los objetivos planteados al inicio de este trabajo, se ha logrado desarrollar un sistema informático funcional que abarca las necesidades definidas y esperadas en cuanto a la necesidad primaria.
- En la elaboración de este proyecto, se aplicaron y superaron aspectos que van desde lo técnico hasta metodológico e investigativo, los cuales en cierta parte del trabajo resultaron ser desafíos, pero que, gracias a la dedicación y el estudio, pudieron ser sobrepuestos y aplicados de la mejor manera. Esto ciertamente permitió obtener un producto acorde a las expectativas y de calidad.

- El sistema desarrollado en este proyecto representa al final un aporte al contexto inicial según fue planteado: manejo de clientes, pacientes, citas, valoraciones y tratamientos. En este sentido, ha representado de valor a nivel operacional de la veterinaria, ya que, gracias a esto, puede ver agilizados los trámites.
- El sistema desarrollado ha sido probado y validado de acuerdo con la funcionalidad esperada por el beneficiario, por lo que se garantiza que cumple con los objetivos establecidos en el proyecto en cuestión.
- Gracias a las enseñanzas obtenidas a lo largo del camino en la universidad, junto con el tutor de este proyecto, se hace entrega de un trabajo con el cual se está en completo grado de satisfacción por parte del autor.

### **Recomendaciones**

- Como primera recomendación, se encuentra generar copias de seguridad y respaldos de la base de datos de forma periódica para evitar posibles pérdidas de información.
- En caso de ser requerido, se recomienda también ampliar las funcionalidades necesarias para aumentar las capacidades del sistema, la solución entregada permite realizar esta labor en caso de ser necesario.
- Como medida alterna, se recomienda implementar medidas de seguridad adicionales para el sistema.
- Seguidamente, se recomienda establecer un plan de mantenimiento preventivo y correctivo para garantizar la correcta ejecución del sistema pensado a largo plazo.

- Se recomienda realizar una capacitación interna de administradores y colaboradores de la veterinaria para un correcto y efectivo uso del sistema.

## **Capítulo VI. Propuesta**

## Análisis de requerimientos

### *Requerimientos de usuario*

A continuación, se presenta cada uno de los requerimientos de usuario recolectados a la hora de establecer cuáles son las necesidades específicas de la veterinaria.

**Tabla 11**

#### *Requerimiento de usuario R01*

<b>Responsable: Christopher Ulloa Jiménez</b>	
<b>Id: R01</b>	Prioridad: 1
<b>Descripción:</b>	Diseñar un sistema que permita gestionar las citas veterinarias, clientes, pacientes, usuarios y tratamientos en la Veterinaria Pulguitas.
<b>Fuente:</b>	Encargados de la veterinaria.

<b>Dependencias:</b>	Ninguna.
----------------------	----------

En la tabla 11, se describe el requerimiento sobre el sistema en general por parte de los encargados de la veterinaria.

### Tabla 12

#### *Requerimiento de usuario R02*

<b>Responsable: Christopher Ulloa Jiménez</b>	
<b>Id: R02</b>	<b>Prioridad: 1</b>
<b>Descripción:</b>	Desarrollar un formulario para el control de inicio de sesión.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 12, se observa el requerimiento de inicio de sesión solicitado por parte de los encargados de la veterinaria.

### Tabla 13

#### *Requerimiento de usuario R03*

<b>Responsable: Christopher Ulloa Jiménez</b>	
<b>Id: R03</b>	Prioridad: 1
<b>Descripción:</b>	Desarrollar un formulario para el control del registro de citas veterinarias.
<b>Fuente:</b>	Encargados de la veterinaria.

<b>Dependencias:</b>	Ninguna.
----------------------	----------

En la tabla 13, se observa el requerimiento de registro de citas veterinarias solicitado por parte de los encargados de la veterinaria.

#### Tabla 14

##### *Requerimiento de usuario R04*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: R04</b>	Prioridad: 1
<b>Descripción:</b>	Desarrollar un formulario para el control del registro o modificación de clientes de la veterinaria.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 14, se observa el requerimiento de registro o modificación de los clientes de la veterinaria solicitado por parte de los encargados de esta.

#### Tabla 15

##### *Requerimiento de usuario R05*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: R05</b>	Prioridad: 1
<b>Descripción:</b>	Desarrollar un formulario para el control del registro o modificación de pacientes de la veterinaria.

<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 15, se observa el requerimiento de registro o modificación de los pacientes de la veterinaria solicitado por parte de los encargados de esta.

### Tabla 16

#### *Requerimiento de usuario R06*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: R06</b>	Prioridad: 1
<b>Descripción:</b>	Desarrollar un formulario para el control del registro o modificación de los usuarios del sistema a desarrollar para la veterinaria.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 16, se observa el requerimiento de registro o modificación de los usuarios que van a utilizar el sistema de la veterinaria, solicitado por parte de los encargados de la veterinaria.

### Tabla 17

#### *Requerimiento de usuario R07*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: R07</b>	Prioridad: 1

<b>Descripción:</b>	Desarrollar un formulario para el control del registro o modificación de las valoraciones de las citas dentro del sistema a desarrollar para la veterinaria.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 17, se observa el requerimiento de registro o modificación de las valoraciones de las citas activas dentro del sistema de la veterinaria, solicitado por parte de los encargados de esta.

### Tabla 18

#### *Requerimiento de usuario R08*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: R08</b>	Prioridad: 1
<b>Descripción:</b>	Desarrollar un formulario para el control del registro o modificación de tratamientos veterinarios para asignar de la veterinaria.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 18, se observa el requerimiento de registro o modificación de los tratamientos para asignar de la veterinaria, solicitado por parte de los encargados de esta.

### ***Requerimientos funcionales***

Como se ha mencionado anteriormente, los requerimientos funcionales son las funciones específicas que se espera que el sistema pueda ejecutar. A continuación, se presenta la lista de requerimientos funcionales para el sistema de citas de la veterinaria pulguitas.

**Tabla 19**

#### *Requerimiento funcional RF01*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: RF01</b>	Prioridad: 1
<b>Descripción:</b>	Inicio de sesión de usuarios.
<b>Criterio de Aceptación:</b>	Los usuarios del sistema ya registrados deben poder iniciar sesión en el sistema con sus credenciales.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 19, se observa el requerimiento funcional de ingreso de sistema de los usuarios, en donde se permite el acceso a los usuarios.

**Tabla 20**

#### *Requerimiento funcional RF02*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: RF02</b>	Prioridad: 1

<b>Descripción:</b>	Registro de usuarios.
<b>Criterio de Aceptación:</b>	El usuario debe poder registrarse en el sistema proporcionando su información básica.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 20, se observa el requerimiento funcional del registro en sistema de los usuarios, en donde solicita cierta información relevante del mismo.

### Tabla 21

#### *Requerimiento funcional RF03*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: RF03</b>	Prioridad: 1
<b>Descripción:</b>	Gestión de usuarios.
<b>Criterio de Aceptación:</b>	El administrador ha de ser capaz de administrar las cuentas de los usuarios, manipulación de datos y desactivación de cuentas.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 21, se observa el requerimiento funcional del registro en sistema de los usuarios, donde solicita cierta información relevante del mismo.

### Tabla 22

#### *Requerimiento funcional RF04*

<b>Responsable: Christopher Ulloa Jiménez</b>	
<b>Id: RF04</b>	Prioridad: 1
<b>Descripción:</b>	Gestión de citas.
<b>Criterio de Aceptación:</b>	El usuario del sistema ha de ser capaz de registrar las citas de los pacientes veterinarios, asimismo, se espera que, con cada cita, se envíe una notificación por medio de correo electrónico sobre la cita.  También se espera que las citas puedan ser canceladas por medio del sistema.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 22, se observa el requerimiento funcional del registro en sistema de las citas por parte de los usuarios, al mismo tiempo se aclara que también puede darse el chance de cancelar la cita de ser necesario.

### Tabla 23

#### *Requerimiento funcional RF05*

<b>Responsable: Christopher Ulloa Jiménez</b>	
<b>Id: RF05</b>	Prioridad: 1
<b>Descripción:</b>	Gestión de clientes.
<b>Criterio de Aceptación:</b>	El usuario del sistema ha de ser capaz de registrar nuevos clientes (dueños de los animales); asimismo, se espera que la información de estos se pueda modificar según sea conveniente.

<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 23, se observa el requerimiento funcional del registro en sistema del cliente por parte de los usuarios, al mismo tiempo se aclara que también puede darse el chance de modificar la información de estos cuando sea necesario.

#### **Tabla 24**

##### *Requerimiento funcional RF06*

<b>Responsable: Christopher Ulloa Jiménez</b>	
<b>Id: RF06</b>	Prioridad: 1
<b>Descripción:</b>	Gestión de pacientes.
<b>Criterio de Aceptación:</b>	El usuario del sistema ha de ser capaz de registrar nuevos pacientes, asimismo, se espera que la información de estos se pueda modificar según sea conveniente.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 24, se observa el requerimiento funcional del registro en sistema de los pacientes por parte de los usuarios; al mismo tiempo, se aclara que también puede darse el chance de modificar la información de estos cuando sea necesario.

#### **Tabla 25**

##### *Requerimiento funcional RF07*

<b>Responsable: Christopher Ulloa Jiménez</b>	
<b>Id: RF07</b>	Prioridad: 1
<b>Descripción:</b>	Gestión de tratamientos.
<b>Criterio de Aceptación:</b>	El usuario administrador del sistema ha de ser capaz de registrar nuevos tratamientos; asimismo, se espera que la información de estos se pueda modificar según sea conveniente.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 25, se observa el requerimiento funcional del registro en sistema de los tratamientos por parte del usuario administrador, al mismo tiempo se aclara que también puede darse el chance de modificar la información de estos cuando sea necesario.

**Tabla 26**

*Requerimiento funcional RF08*

<b>Responsable: Christopher Ulloa Jiménez</b>	
<b>Id: RF08</b>	Prioridad: 1
<b>Descripción:</b>	Gestión de valoraciones de citas.
<b>Criterio de Aceptación:</b>	El usuario del sistema ha de ser capaz de registrar valoraciones sobre las citas activas; asimismo, se espera que la información de las citas que están en estado CERRADO que, por algún motivo o por error han sido cerradas, se pueda modificar según sea conveniente.
<b>Fuente:</b>	Encargados de la veterinaria.

<b>Dependencias:</b>	Ninguna.
----------------------	----------

En la tabla 26, se observa el requerimiento funcional del registro en sistema de los tratamientos por parte de los usuarios; al mismo tiempo, se aclara que también puede darse el chance de modificar la información de estos cuando sea necesario.

### ***Requerimientos no funcionales***

Como se ha mencionado anteriormente, los requerimientos no funcionales son aquellos que garantizan la correcta operación del sistema.

### **Tabla 27**

#### *Requerimiento funcional RNF01*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: RNF01</b>	Prioridad: 1
<b>Descripción:</b>	Seguridad.
<b>Criterio de Aceptación:</b>	El sistema debe implementar medidas de seguridad para garantizar la integridad de los datos, validará credenciales y roles de los usuarios.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 27, se observa el requerimiento no funcional de seguridad, en el sistema se tendrá dos roles: administrador y normal. Cada uno de ellos tendrá funcionalidades que pueden realizar.

**Tabla 28***Requerimiento funcional RNF02*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: RNF02</b>	Prioridad: 1
<b>Descripción:</b>	Usabilidad.
<b>Criterio de Aceptación:</b>	El sistema debe implementar una interfaz intuitiva y fácil de usar para el usuario.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 28, se observa el requerimiento no funcional de usabilidad, en el sistema se busca tener una interfaz que sea práctica y entendible para el usuario que utilice la herramienta.

**Tabla 29***Requerimiento funcional RNF03*

<b>Responsable: Cristopher Ulloa Jiménez</b>	
<b>Id: RNF03</b>	Prioridad: 1
<b>Descripción:</b>	Disponibilidad.
<b>Criterio de Aceptación:</b>	Una vez el sistema esté instalado, este debe tener disponibilidad de uso en los horarios operativos de la veterinaria.
<b>Fuente:</b>	Encargados de la veterinaria.

<b>Dependencias:</b>	Ninguna.
----------------------	----------

En la tabla 29, se observa el requerimiento no funcional de disponibilidad, se espera que la disponibilidad del sistema esté una vez está instalado sin ningún inconveniente.

### Tabla 30

#### *Requerimiento funcional RNF04*

<b>Responsable: Christopher Ulloa Jiménez</b>	
<b>Id: RNF04</b>	Prioridad: 1
<b>Descripción:</b>	Utilidad.
<b>Criterio de Aceptación:</b>	El sistema se espera que registre las citas y la información tanto de los usuarios como de los pacientes, sin ningún inconveniente.
<b>Fuente:</b>	Encargados de la veterinaria.
<b>Dependencias:</b>	Ninguna.

En la tabla 30, se observa el requerimiento no funcional de utilidad, se espera que el sistema sea de completa utilidad para la veterinaria y llene las necesidades requeridas.

### Modelos UML

#### *Casos de uso*

Seguidamente, se detallan los actores del sistema:

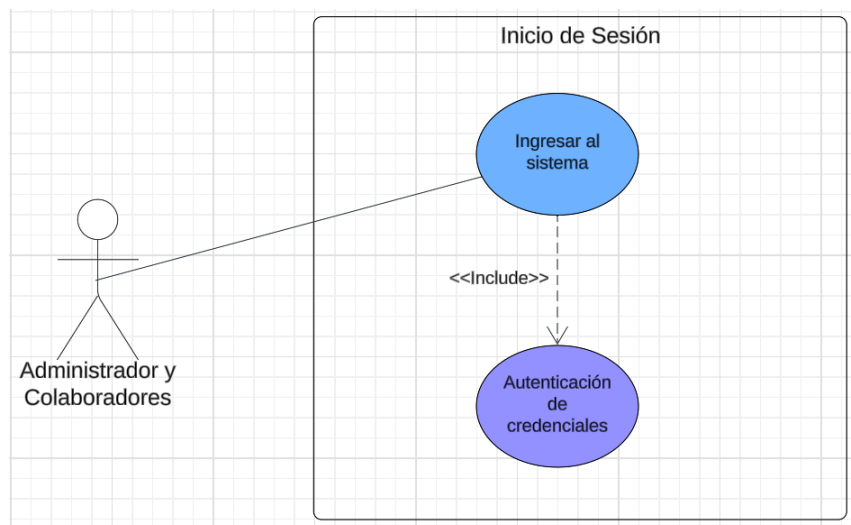
**Administrador:** persona que puede realizar todas las funcionalidades del sistema y acceder a la información relacionada a los clientes, pacientes, usuarios, citas, valoraciones de citas activas y tratamientos.

**Colaborador:** es la persona que puede registrar clientes, registrar pacientes, registrar y modificar citas, así como agregar valoraciones de citas activas.

### Caso de uso 1. Inicio de sesión

**Figura 10**

*Diagrama de caso de uso. Inicio de sesión.*



**Tabla 31**

*Caso de uso – Inicio de sesión*

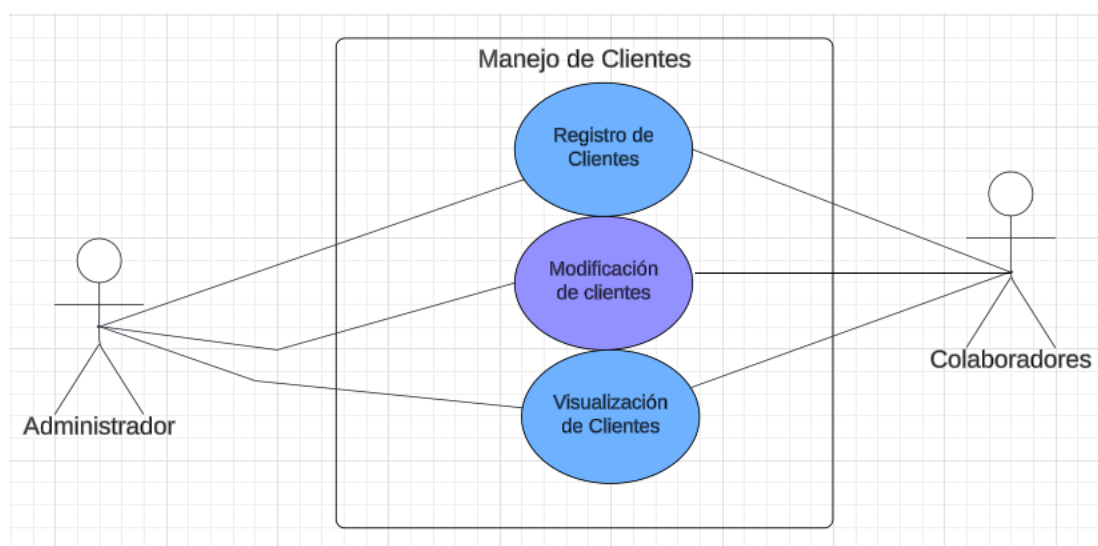
<b>Caso de uso. Inicio de sesión</b>	
<b>Actores</b>	Administrador y colaboradores.
<b>Propósito:</b>	Ingreso y autenticación del usuario en el sistema.
<b>Curso Normal:</b>	<ol style="list-style-type: none"> <li>1. El caso inicia cuando el actor digita su correo y contraseña en el sistema.</li> <li>2. El sistema procesa la información y realiza la validación de credenciales contra la base de datos.</li> </ol>

	3. Si el sistema ubica las credenciales, le permite el ingreso y el caso de uso finaliza.
<b>Curso Alterno:</b>	Si el actor olvida las credenciales debe solicitar a un administrador una nueva contraseña.
<b>Dependencias:</b>	Ninguna.

### Caso de uso 2. Manejo de clientes

**Figura 11**

*Diagrama de caso de uso. Manejo de clientes.*



**Tabla 32**

*Caso de uso – Manejo de clientes*

<b>Caso de uso. Manejo de clientes.</b>	
<b>Actores</b>	Administrador y colaborador.

<b>Propósito:</b>	Registro, modificación y visualización de clientes en el sistema.
<b>Curso Normal:</b>	<ol style="list-style-type: none"> <li>1. El caso inicia cuando el actor se autentica con su correo y contraseña en el sistema.</li> <li>2. El actor ingresa al formulario de registro o modificación del cliente.</li> <li>3. El actor ingresa nuevo cliente, modifica cliente o si lo desea visualiza la información de todos los clientes.</li> </ol>
<b>Curso Alternativo:</b>	Si el actor olvida las credenciales, debe solicitar a un administrador una nueva contraseña.
<b>Dependencias:</b>	Ninguna.

### Caso de uso 3. Manejo de pacientes

Figura 12

Diagrama de caso de uso – Manejo de pacientes

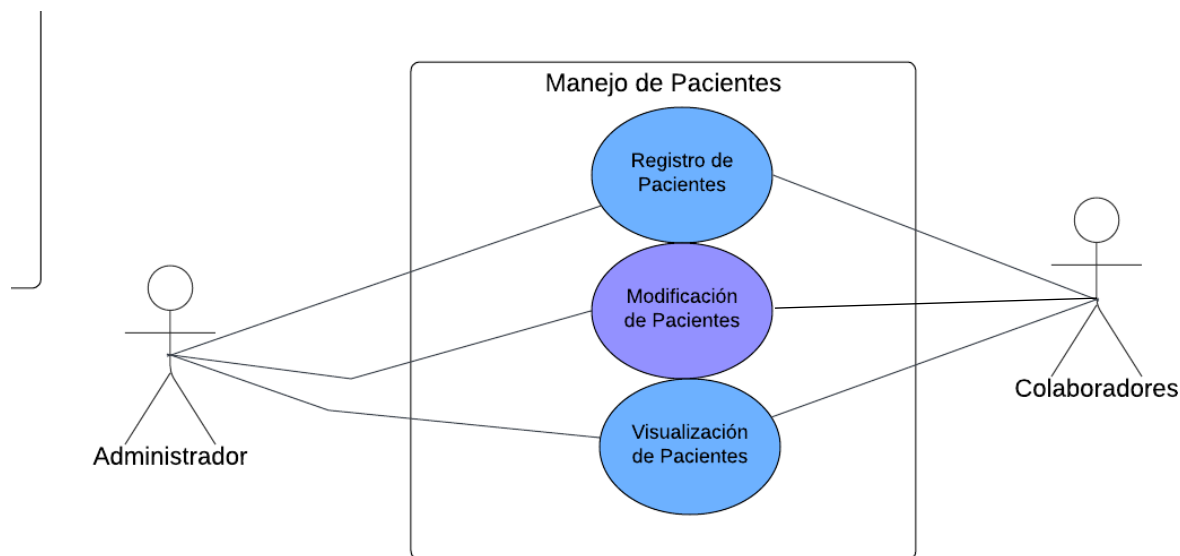


Tabla 33

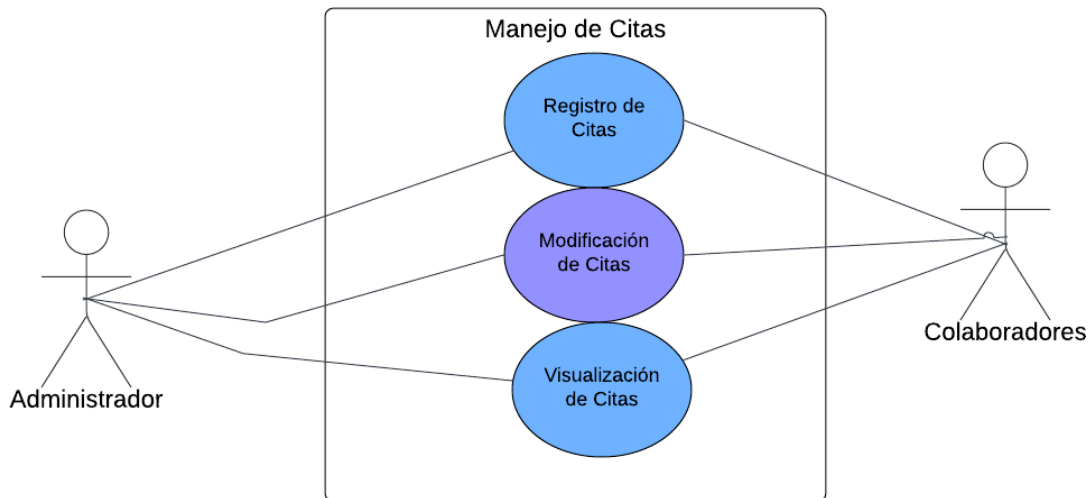
*Caso de uso – Manejo de pacientes*

<b>Caso de uso. Manejo de pacientes.</b>	
<b>Actores</b>	Administrador y colaborador.
<b>Propósito:</b>	Registro, modificación y visualización de pacientes en el sistema.
<b>Curso Normal:</b>	<ol style="list-style-type: none"> <li>1. El caso inicia cuando el actor se autentica con su correo y contraseña en el sistema.</li> <li>2. El actor ingresa al formulario de registro o modificación del paciente.</li> <li>3. El actor ingresa nuevo paciente, modifica paciente o si lo desea visualiza la información de todos los pacientes.</li> </ol>
<b>Curso Alterno:</b>	Si el actor olvida las credenciales, debe solicitar a un administrador una nueva contraseña.
<b>Dependencias:</b>	Ninguna.

**Caso de uso 4. Manejo de citas.**

**Figura 13**

*Diagrama de caso de uso – Manejo de citas.*



**Tabla 34**

*Caso de uso – Manejo de citas*

<b>Caso de uso. Manejo de citas.</b>	
<b>Actores</b>	Administrador y colaborador.
<b>Propósito:</b>	Registro, modificación y visualización de citas en el sistema.
<b>Curso Normal:</b>	<ol style="list-style-type: none"> <li>1. El caso inicia cuando el actor se autentica con su correo y contraseña en el sistema.</li> <li>2. El actor ingresa al formulario de manejo de citas.</li> <li>3. El actor ingresa nueva cita para el paciente, modifica citas ya registradas o si lo desea visualiza la información de las citas en sistema.</li> </ol>
<b>Curso Alternativo:</b>	Si el actor olvida las credenciales, debe solicitar a un administrador una nueva contraseña.
<b>Dependencias:</b>	Ninguna.

## Caso de uso 5. Manejo de valoraciones

Figura 14

Diagrama de caso de uso – Manejo de valoraciones

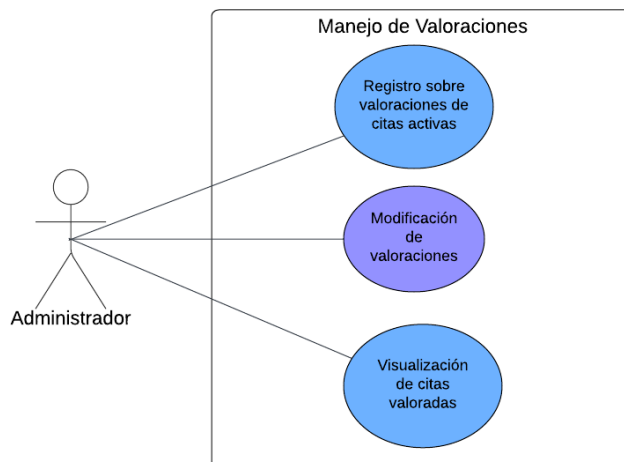


Tabla 35

Caso de uso – Manejo de valoraciones

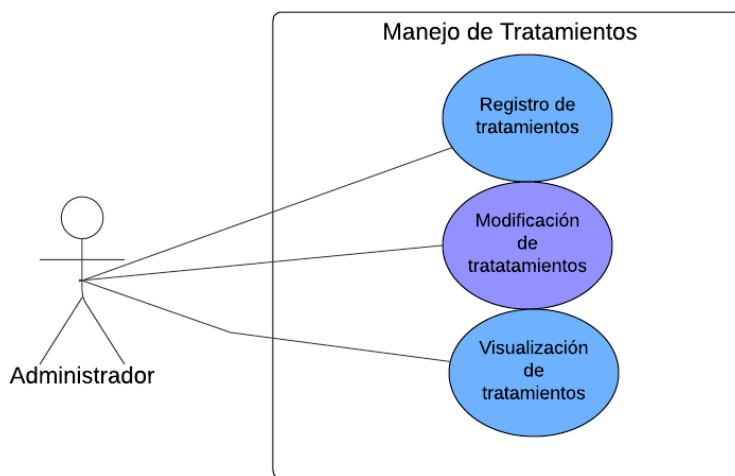
Caso de uso. Manejo de valoraciones.	
<b>Actores</b>	Administrador.
<b>Propósito:</b>	Registro, modificación y visualización de valoración de citas en el sistema.
<b>Curso Normal:</b>	<ol style="list-style-type: none"> <li>1. El caso inicia cuando el actor se autentica con su correo y contraseña en el sistema.</li> <li>2. El actor ingresa al formulario de manejo de valoraciones.</li> <li>3. El actor ingresa nueva valoración para el paciente que tiene una cita activa, modifica citas ya valoradas en caso de que sea necesario modificarla o si lo desea visualiza la información de las citas ya valoradas anteriormente.</li> </ol>

<b>Curso Alterno:</b>	Si el actor olvida las credenciales debe solicitar a un administrador una nueva contraseña.
<b>Dependencias:</b>	Ninguna.

### Caso de uso 6. Manejo de tratamientos

**Figura 15**

*Diagrama de caso de uso- Manejo de tratamientos.*



**Tabla 36**

*Caso de uso – Manejo de tratamientos*

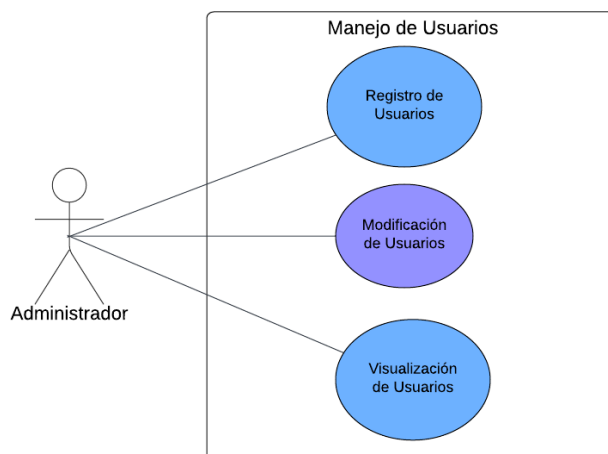
<b>Caso de uso. Manejo de tratamientos.</b>	
<b>Actores</b>	Administrador.
<b>Propósito:</b>	Registro, modificación y visualización de tratamientos en el sistema.

<b>Curso Normal:</b>	<ol style="list-style-type: none"> <li>1. El caso inicia cuando el actor se autentica con su correo y contraseña en el sistema.</li> <li>2. El actor ingresa al formulario de manejo de tratamientos.</li> <li>3. El actor ingresa nuevo tratamiento, puede modificar el tratamiento y si lo desea, visualiza la información de los tratamientos registrados anteriormente.</li> </ol>
<b>Curso Alternativo:</b>	Si el actor olvida las credenciales, debe solicitar a un administrador una nueva contraseña.
<b>Dependencias:</b>	Ninguna.

### Caso de uso 7. Manejo de usuarios

**Figura 16**

*Diagrama de caso de uso- Manejo de usuarios.*



**Tabla 37**

*Caso de uso – Manejo de usuarios*

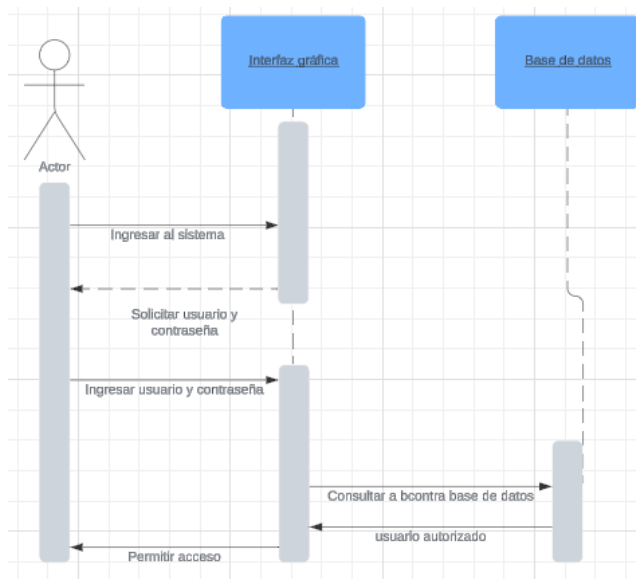
<b>Caso de uso. Manejo de usuarios.</b>	
<b>Actores</b>	Administrador.
<b>Propósito:</b>	Registro, modificación y visualización de usuarios en el sistema.
<b>Curso Normal:</b>	<ol style="list-style-type: none"> <li>1. El caso inicia cuando el actor se autentica con su correo y contraseña en el sistema.</li> <li>2. El actor ingresa al formulario de manejo de usuarios.</li> <li>3. El actor ingresa al nuevo usuario, puede modificar al usuario y si lo desea visualiza la información de los tratamientos registrados anteriormente.</li> </ol>
<b>Curso Alterno:</b>	Si el actor olvida las credenciales, debe solicitar a un administrador una nueva contraseña.
<b>Dependencias:</b>	Ninguna.

### *Diagramas de secuencia*

#### **Diagrama de secuencia 1. Inicio de sesión.**

#### **Figura 17**

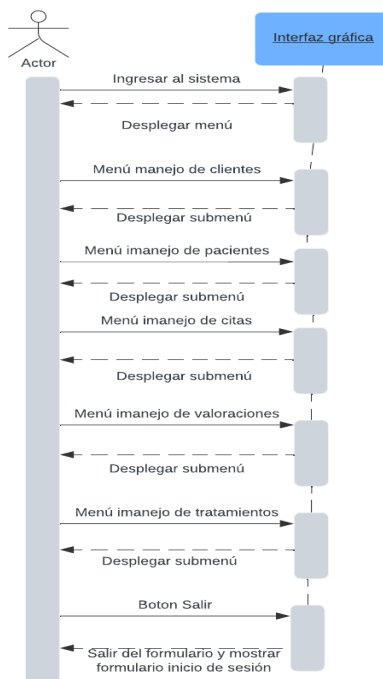
*Diagrama de secuencia – Inicio de sesión.*



**Diagrama de secuencia 2 – Visualización de menú principal**

**Figura 18**

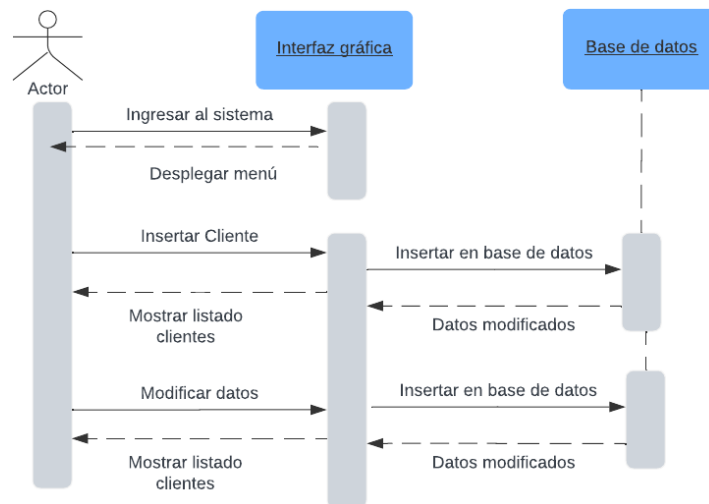
*Diagrama de secuencia – Visualización de menú principal*



### Diagrama de secuencia 3. Manejo de clientes

Figura 19

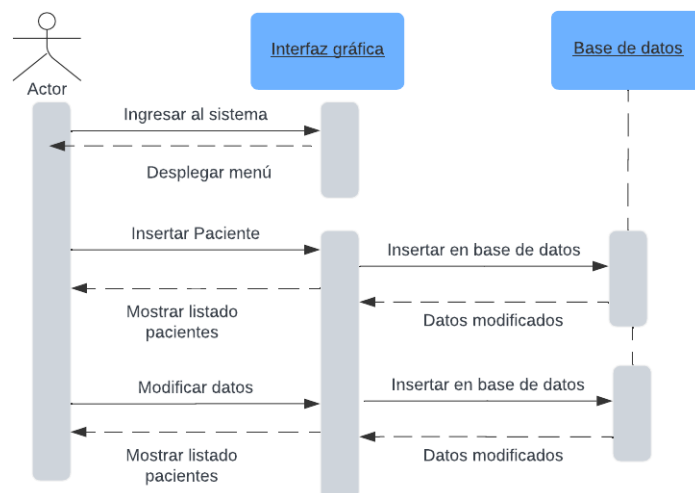
*Diagrama de secuencia- Manejo de clientes.*



### Diagrama de secuencia 4. Manejo de pacientes

Figura 20

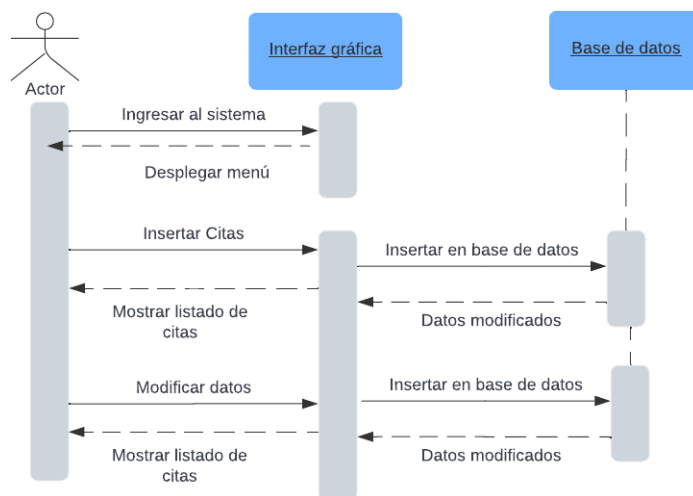
*Diagrama de secuencia 4. Manejo de pacientes*



### Diagrama de secuencia 5. Manejo de citas.

Figura 21

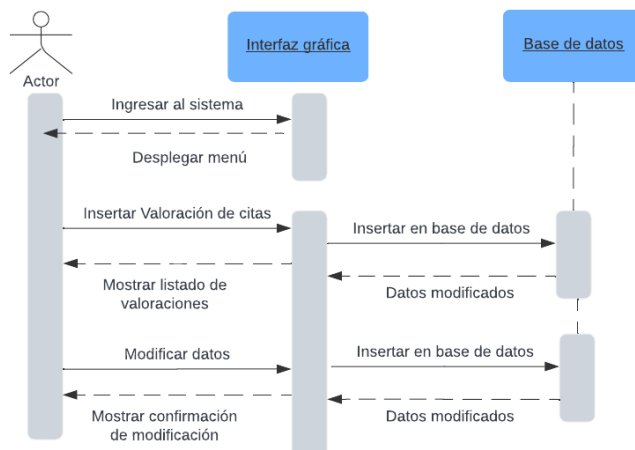
*Diagrama de secuencia 4. Manejo de citas*



### Diagrama de secuencia 6. Manejo de valoraciones.

Figura 22

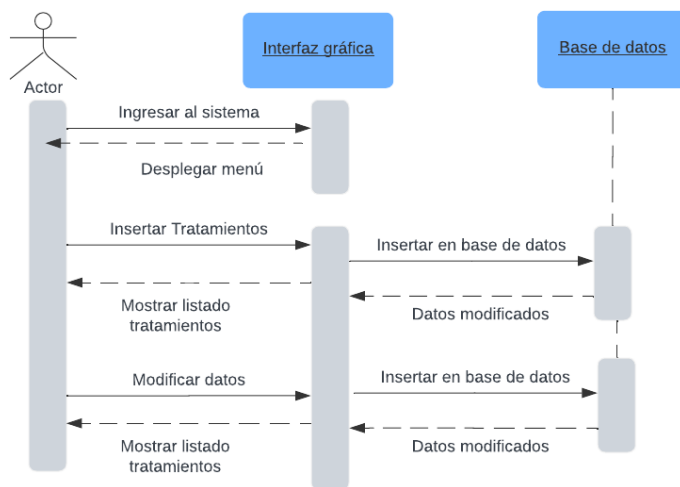
*Diagrama de secuencia 4. Manejo de valoraciones*



### Diagrama de secuencia 7. Manejo de tratamientos

Figura 23

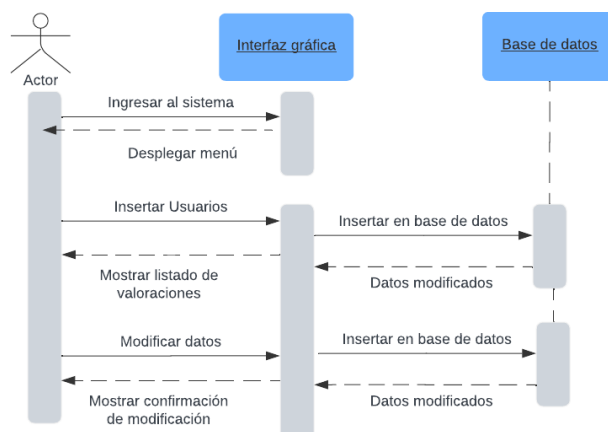
*Diagrama de secuencia 4. Manejo de tratamientos*



### Diagrama de secuencia 8. Manejo de usuarios

Figura 24

*Diagrama de secuencia 4. Manejo de usuarios*

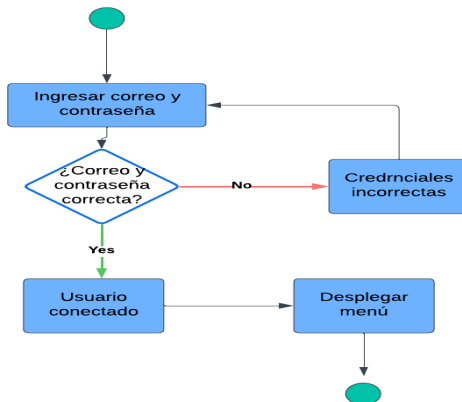


### Diagramas de actividades del sistema

Diagrama de actividad 1. Inicio de sesión.

**Figura 25**

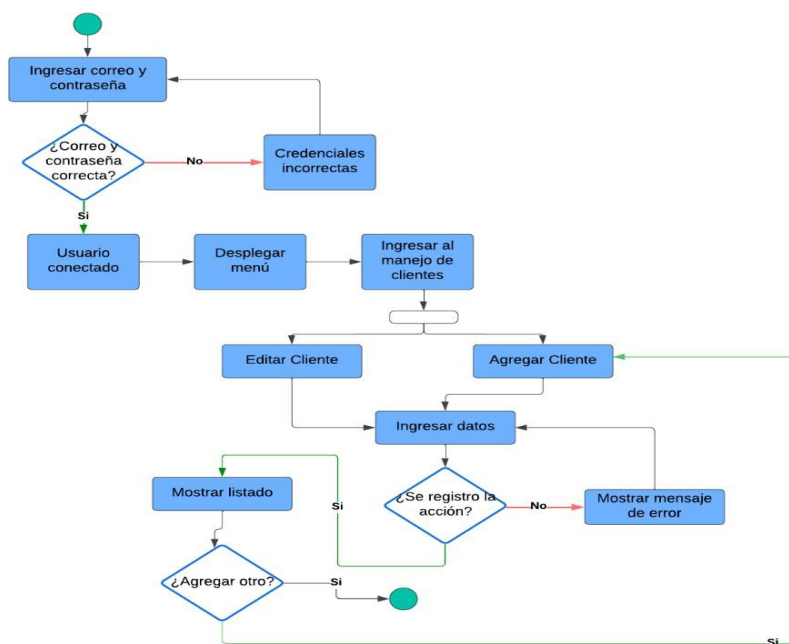
*Diagrama de actividad – Inicio de sesión.*



**Diagrama de actividad 2. Manejo de clientes.**

**Figura 26**

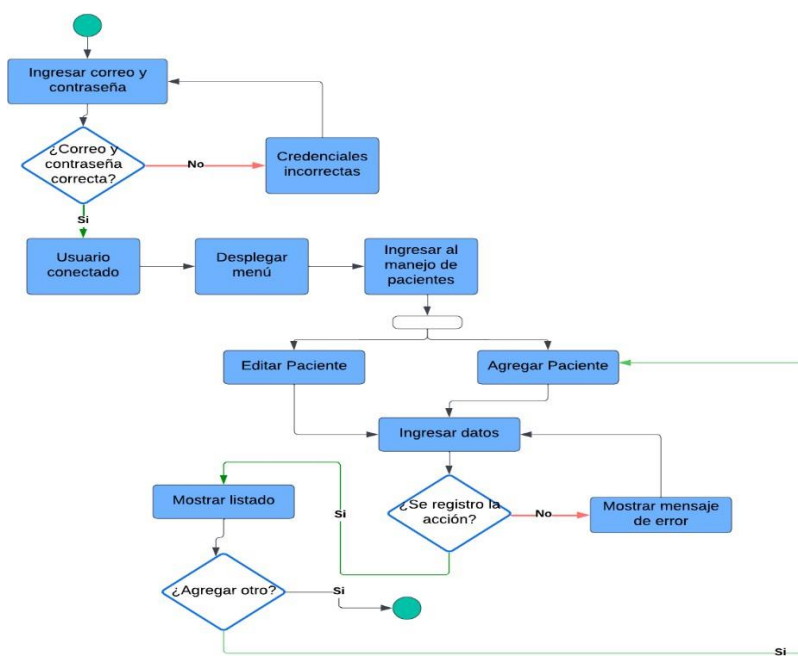
*Diagrama de actividad. Manejo de clientes.*



**Diagrama de actividad 3. Manejo de pacientes.**

**Figura 27**

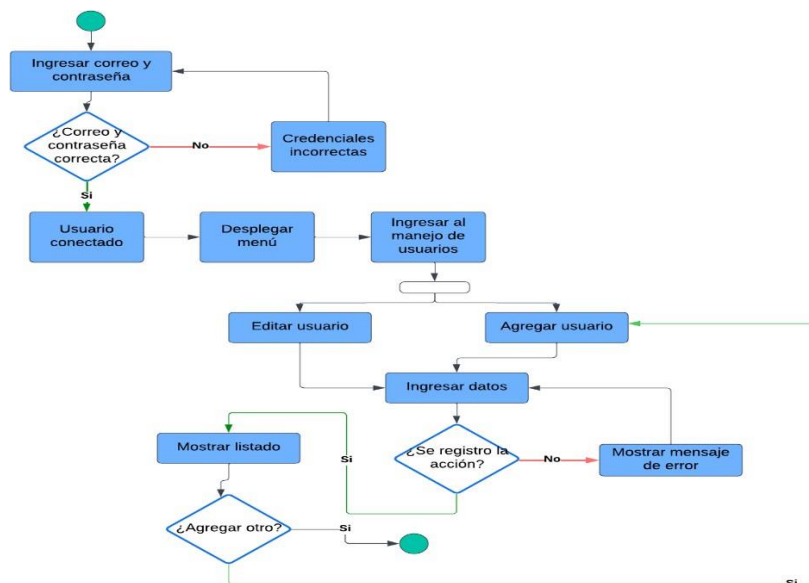
*Diagrama de actividad. Manejo de pacientes.*



### Diagrama de actividad 4. Manejo de usuarios.

Figura 28

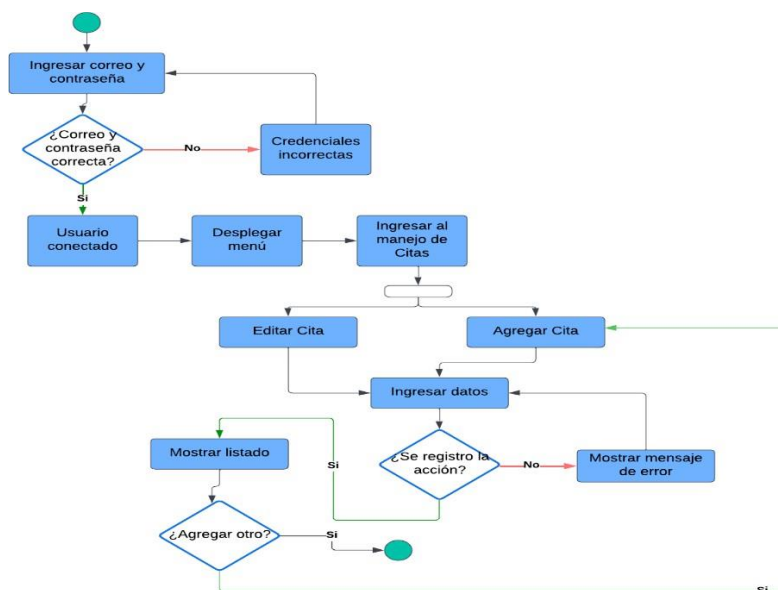
Diagrama de actividad. Manejo de usuarios.



### Diagrama de actividad 5. Manejo de citas.

Figura 29

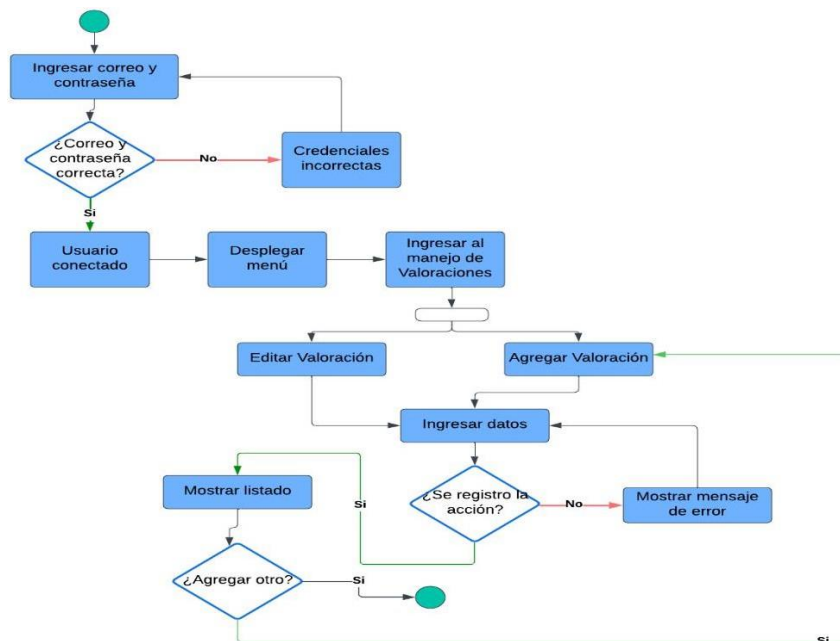
Diagrama de actividad. Manejo de citas.



**Diagrama de actividad 6. Manejo de valoraciones.**

**Figura 30**

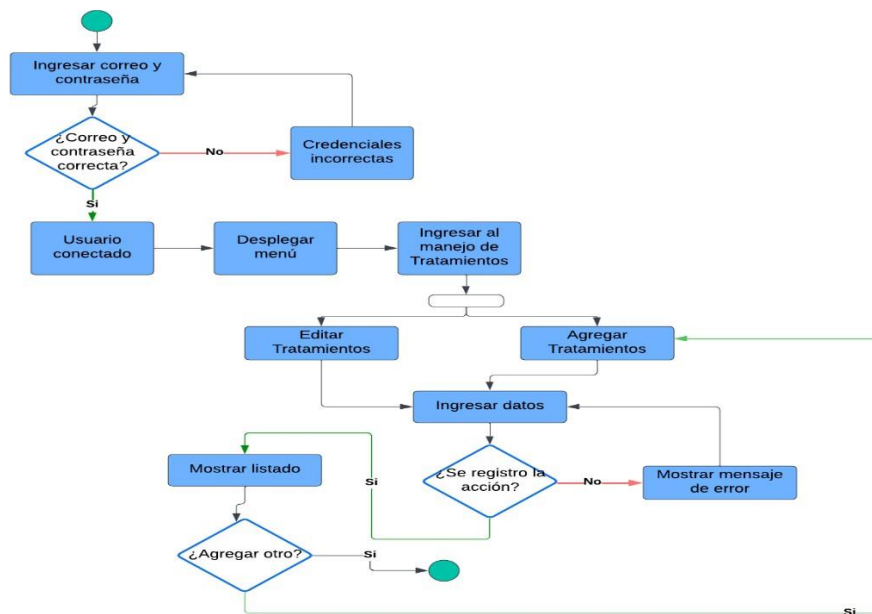
*Diagrama de actividad. Manejo de valoraciones.*



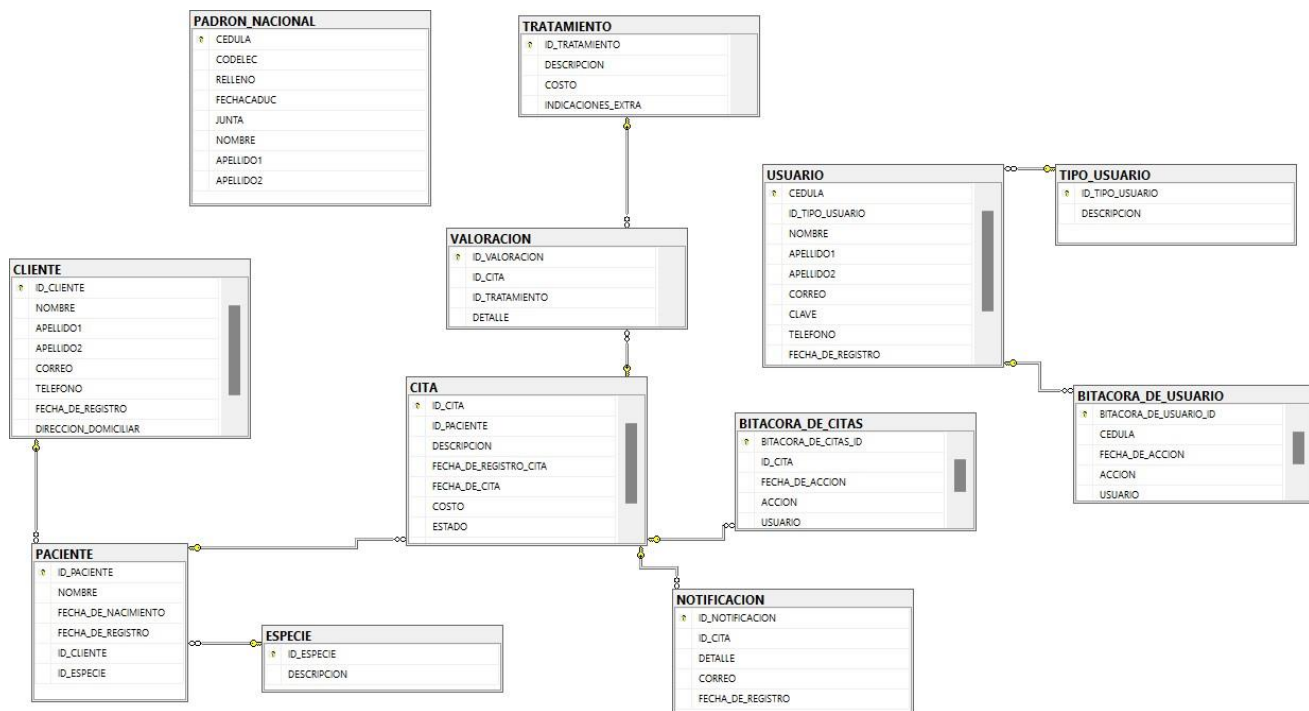
## Diagrama de actividad 7. Manejo de tratamientos.

Figura 31

Diagrama de actividad. Manejo de tratamientos.



### Modelo ER (Entidad Relación)



### Diccionarios de datos

Tabla 38

Diccionario de datos tabla CLIENTE

CLIENTE				
Tabla de parámetros del sistema				
Campo	Tipo de dato	Llave primaria	Nulo	
ID_CLIENTE	INT	PK	No	
NOMBRE	VARCHAR(100)		No	
APELLIDO1	VARCHAR(100)		No	
APELLIDO2	VARCHAR(100)		No	
CORREO	VARCHAR(100)		No	
TELEFONO	INT		No	
FECHA_DE_REGISTRO	DATETIME		No	
DIRECCION_DOMICILIAR	VARCHAR(100)		No	

Tabla 39

Diccionario de datos tabla ESPECIE

ESPECIE			
Tabla de parámetros del sistema			
Campo	Tipo de dato	Llave primaria	Nulo
ID_CLIENTE	INT	PK	No
DESCRIPCION	VARCHAR(100)		No

Tabla 40

Diccionario de datos tabla NOTIFICACION

NOTIFICACION			
Tabla de parámetros del sistema			
Campo	Tipo de dato	Llave primaria	Nulo
ID_NOTIFICACION	INT	PK	No
ID_CITA	INT		
DETALLE	VARCHAR(100)		No
CORREO	VARCHAR(100)		
FECHA_DE_REGISTRO	DATETIME		

Tabla 41

Diccionario de datos tabla PACIENTE

PACIENTE			
Tabla de parámetros del sistema			
Campo	Tipo de dato	Llave primaria	Nulo
ID_PACIENTE	INT	PK	No
NOMBRE	VARCHAR(100)		No
FECHA_DE_NACIMIENTO	DATETIME		No
FECHA_DE_REGISTRO	DATETIME		No

<b>ID_CLIENTE</b>	INT	No
<b>ID_ESPECIE</b>	INT	No

**Tabla 42**

*Diccionario de datos tabla PADRON\_NACIONAL*

<b>PADRON_NACIONAL</b>			
<b>Tabla de parámetros del sistema</b>			
<b>Campo</b>	<b>Tipo de dato</b>	<b>Llave primaria</b>	<b>Nulo</b>
<b>CEDULA</b>	INT	PK	No
<b>CODELEC</b>	INT		No
<b>APELLIDO1</b>	VARCHAR(100)		No
<b>APELLIDO2</b>	VARCHAR(100)		No
<b>NOMBRE</b>	VARCHAR(100)		No
<b>JUNTA</b>	INT		No
<b>FECHACADUC</b>	INT		No
<b>RELLENO</b>	VARCHAR(5)		No

**Tabla 43**

*Diccionario de datos tabla TRATAMIENTO*

<b>TRATAMIENTO</b>			
<b>Tabla de parámetros del sistema</b>			
<b>Campo</b>	<b>Tipo de dato</b>	<b>Llave primaria</b>	<b>Nulo</b>
<b>ID_TIPO_USUARIO</b>	INT	PK	No
<b>DESCRIPCION</b>	VARCHAR(100)		No
<b>COSTO</b>	INT		No
<b>INDICACIONES_EXTRA</b>	VARCHAR(100)		Si

**Tabla 44**

*Diccionario de datos tabla TIPO\_USUARIO*

<b>TIPO_USUARIO</b>			
<b>Tabla de parámetros del sistema</b>			
<b>Campo</b>	<b>Tipo de dato</b>	<b>Llave primaria</b>	<b>Nulo</b>
<b>ID_TIPO_USUARIO</b>	INT	PK	No
<b>DESCRIPCION</b>	VARCHAR(100)		No

**Tabla 45***Diccionario de datos tabla USUARIO*

<b>USUARIO</b>			
<b>Tabla de parámetros del sistema</b>			
<b>Campo</b>	<b>Tipo de dato</b>	<b>Llave primaria</b>	<b>Nulo</b>
<b>ID_USUARIO</b>	INT	PK	No
<b>ID_TIPO_USUARIO</b>	INT		No
<b>NOMBRE</b>	VARCHAR(100)		No
<b>APELLIDO1</b>	VARCHAR(100)		No
<b>APELLIDO2</b>	VARCHAR(100)		No
<b>CORREO</b>	VARCHAR(100)		No
<b>TELEFONO</b>	INT		No
<b>FECHA_DE_REGISTRO</b>	DATETIME		No
<b>ESTADO</b>	VARCHAR(40)		No
<b>CLAVE</b>	VARCHAR(60)		No

**Tabla 46***Diccionario de datos tabla VALORACION*

<b>VALORACION</b>			
<b>Tabla de parámetros del sistema</b>			
<b>Campo</b>	<b>Tipo de dato</b>	<b>Llave primaria</b>	<b>Nulo</b>

<b>ID_VALORACION</b>	INT	PK	No
<b>ID_CITA</b>	INT		No
<b>ID_TRATAMIENTO</b>	INT		No
<b>DETALLE</b>	VARCHAR(100)		No

**Tabla 47**

*Diccionario de datos tabla BITACORA\_DE\_CITAS*

<b>BITACORA_DE_CITAS</b>			
<b>Tabla de parámetros del sistema</b>			
<b>Campo</b>	<b>Tipo de dato</b>	<b>Llave primaria</b>	<b>Nulo</b>
<b>ID_BITACORA_DE_CITAS</b>	INT	PK	No
<b>ID_CITA</b>	INT		No
<b>FECHA_DE_ACCION</b>	DATETIME		No
<b>ACCION</b>	VARCHAR(25)		
<b>USUARIO</b>	VARCHAR(70)		No

**Tabla 48**

*Diccionario de datos tabla BITACORA\_DE\_USUARIO*

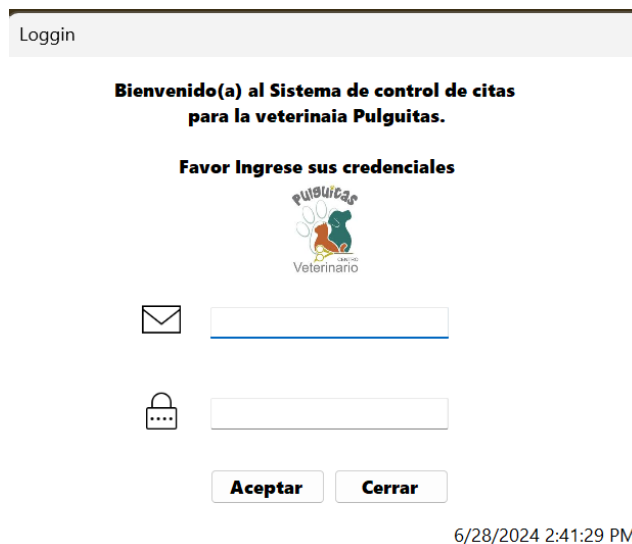
<b>BITACORA_DE_USUARIO</b>			
<b>Tabla de parámetros del sistema</b>			
<b>Campo</b>	<b>Tipo de dato</b>	<b>Llave primaria</b>	<b>Nulo</b>
<b>ID_BITACORA_DE_USUARIO</b>	INT	PK	No
<b>CEDULA</b>	INT		No
<b>FECHA_DE_ACCION</b>	DATETIME		No
<b>ACCION</b>	VARCHAR(25)		
<b>USUARIO</b>	VARCHAR(70)		No

## Diseño de pantallas del sistema

### *Pantalla de Inicio de sesión*

#### Figura 32

*Pantalla por medio de la que se validan las credenciales del usuario*



The screenshot shows a login interface with the following elements:

- A header bar with the text "Loggin".
- A welcome message: "Bienvenido(a) al Sistema de control de citas para la veterinaria Pulguitas."
- A prompt: "Favor Ingrese sus credenciales".
- A logo for "Pulguitas Veterinario" featuring a green bear and a red dog.
- Two input fields: the first is preceded by an envelope icon (email) and the second by a padlock icon (password).
- Two buttons at the bottom: "Aceptar" and "Cerrar".
- A timestamp in the bottom right corner: "6/28/2024 2:41:29 PM".

### *Pantalla del menú principal*

#### Figura 33

*Pantalla del menú principal*



*Pantalla para el manejo de clientes*

**Figura 34**

*Pantalla para registrar, modificar y visualizar los clientes en sistema.*

ID_CLIENTE	NOMBRE	APELLIDO1	APELLIDO2	CORREO	TELEFC

*Pantalla para el manejo de pacientes*

**Figura 35**

*Pantalla para registrar, modificar y visualizar pacientes en el sistema.*

Manejo de Pacientes

6/28/2024 1:51:46 PM

**Registro y/o modificación de datos de Paciente:**

Cédula del Dueño:  Especie:

Nombre:  Fecha de Nacimiento:

Búsqueda Filtrada:

Ingrese el Nombre:

Ver todos los Pacientes

Limpiar Campos

Guardar Cambios

Salir

*Pantalla para manejo de citas*

**Figura 36**

*Pantalla para registro y visualización de las citas próximas en el sistema.*

Manejo de Citas

6/28/2024 1:54:32 PM

**Registro y/o modificación de datos de cita:**

Cédula Cliente:  Paciente:

Descripción:  Costo:

Hora:

Fecha de Cita:

Búsqueda Filtrada:

Ingrese # cita:

Ver todas las citas próximas

Limpiar Campos

Guardar Cambios

Salir

*Pantalla modificación datos de cita activa*

**Figura 37**

*Pantalla modificación de cita*

Modificación de cita

**Modificación de datos de cita:**

ID Cita:  ID Paciente:

Costo:  Paciente:

Hora:  Fecha de Cita:

Descripción:

Guardar Cambios

Cancelar Cita


Salir

*Pantalla manejo de tratamientos*

**Figura 38**

*Pantalla registro, modificación y visualización de tratamientos veterinarios.*

Manejo de Tratamientos 6/28/2024 2:04:13 PM



Ver todos los Tratamientos

Limpiar Campos

Guardar cambios

Salir

**Registro y/o modificación de tratamientos:**

**Descripción:**

**Costo:**

**Otras indicaciones:**

**Búsqueda Filtrada:**


Ingrese la descripción:

*Pantalla para el manejo de valoraciones de citas*

**Figura 39**

*Pantalla para el manejo de valoraciones de citas activas en sistema, se realizan valoraciones de citas activas en el sistema.*

Manejo de Valoraciones 6/28/2024 2:05:14 PM



Ver todas las citas valoradas a la fecha

Limpiar Campos

Guardar cambios

Salir

**Información para agregar valoración veterinaria:**

**Citas Activas que requieren valoración:**

**Tratamiento seleccionado:**

**Detalle:**

**Búsqueda Filtrada:**

Ingrese el nombre de la mascota:

### *Pantalla para modificación de valoraciones*

#### **Figura 40**

*Pantalla donde se hacen las modificaciones de las valoraciones de citas anteriormente valoradas y que requieren un cambio.*


Modificación de Valoración

**Información para la modificación de la valoración**

**Id valoración:** 6 **Tratamiento:**

**Descripción Cita:** Cita rutinaria **Paciente:** mister Ed

**Detalle:** ANIMAL ENFERMO

 Guardar Salir

### *Pantalla para manejo de usuarios.*

#### **Figura 41**

*Pantalla donde se maneja el registro, modificación y visualización de los usuarios del sistema.*

Manejo de Usuarios

6/28/2024 2:12:20 PM

**Registro y/o modificación de Usuarios:**

<b>Cédula:</b>	<input type="text"/>	<b>Nombre:</b>	<input type="text"/>
<b>Primer Apellido:</b>	<input type="text"/>	<b>Segundo Apellido:</b>	<input type="text"/>
<b>Correo:</b>	<input type="text"/>	<b>Teléfono:</b>	<input type="text"/>
<b>Estado:</b>	<input type="text"/>	<b>Generar Nueva Clave:</b>	<input type="text"/>
<b>Tipo de Usuario:</b>	<input type="text"/>		<input type="text"/>

**Búsqueda Filtrada:**

Ingrese la cédula:

Ver todos los usuarios

Limpiar Campos

Guardar Cambios

Salir

## Referencias bibliográficas

- Álvarez., D., & Ramírez, O. (2013). *Sistema Web de odontología general y ortodoncia para la gestión de expedientes, agenda, control de sesiones, cierres de caja y recordatorios de citas vía mensajes de texto SMS y correo electrónico*. UNA, Heredia, Costa Rica.
- Arsys. (n.d). *Todo sobre la arquitectura cliente-servidor*. <https://www.arsys.es/blog/todo-sobre-la-arquitectura-cliente-servidor#:~:text=Ventajas%20del%20modelo%20cliente%2Dservidor&text=Mejora%20a%20escalabilidad%2C%20ya%20que,tenemos%20m%C3%A1s%20capacidad%20de%20intervenci%C3%B3n>.
- Atlassian. (n.d). *Qué es Git*. <https://www.atlassian.com/es/git/tutorials/what-is-git>
- Atlassian. (n.d). *Qué es scrum y cómo empezar*. <https://www.atlassian.com/es/agile/scrum>
- AWS. (n.d). *¿Qué es SQL (lenguaje de consulta estructurada)?* <https://aws.amazon.com/es/what-is/sql/>
- Azuero, E. (2018). Significatividad del marco metodológico en el desarrollo de proyectos de investigación. *Revista Arbitrada Interdisciplinaria KOINONIA, IV*, 110-127.
- Blancarte, O. (n.d). *Arquitectura Cliente-Servidor*. <https://reactiveprogramming.io/blog/es/estilos-arquitectonicos/cliente-servidor>
- Bustamante, R. (2021). Sistema web para la gestión de recursos audiovisuales, servicios de mantenimiento, control administrativo, recopilación y publicación de investigaciones académicas para la Biblioteca de la Universidad Nacional, Sede Regional Brunca, Campus Pérez Zeledón. [Tesis para obtener el título de Licenciatura en Informática con énfasis en Sistemas de Información] Repositorio Universidad Nacional. Universidad Nacional .
- Cabrera, D. (2018). Automatización de la recolección de documentos de los profesores del DETRI para acreditación de carreras. [Tesis de titulación previo a la obtención del título de ingeniero en electrónica y redes de información] Repositorio Digital EPN. Escuela Politécnica Nacional.
- Chacha, K. (2022). Sistema Web aplicando arquitectura modelo vista controlador (MVC) para el control de inventario y producción en la pasteurizadora J.S del cantón salcedo. [Tesis de titulación previo a la obtención del título de ingeniero en sistemas] Repositorio Universidad técnica de Ambato. Universidad técnica de Ambato.
- Chaves, J. (2018). Sistema web de muestreo, trazabilidad, y análisis de ensayos, del Instituto de Investigaciones en Salud. [Tesis para obtener el título de Licenciatura en Informática con énfasis en Sistemas de Información]. Repositorio Universidad Nacional. Universidad Nacional.
- Cipsa Net. (2023). *Historia de la Programación*. <https://cipsa.net/historia-de-la-programacion/>
- Crehana. (2022). *Domina el modelo en cascada y potencia al máximo tus proyectos de software*. <https://www.crehana.com/blog/transformacion-digital/modelo-en-cascada/>
- DB-Engines. (2024). *DB-Engines Ranking*. <https://db-engines.com/en/ranking>

- Deloitte. (n.d). *Las 5 ceremonias Scrum: claves para la gestión de procesos*.  
<https://www2.deloitte.com/es/es/pages/technology/articles/ceremonias-scrum.html>
- Departamento de Comunicación UEMC Business School. (2022). *¿QUÉ ES SCRUM? CONOCE EL FRAMEWORK QUE AGILIZA EL TRABAJO EN EQUIPO*.  
<https://www.escueladenegociosydireccion.com/revista/business/scrum-framework-agiliza-trabajo-equipo/>
- Desire. (n.d). *Scrum*. <https://desire.webs.uvigo.es/contenidos/scrum/>
- Engloba Tech. (n.d). *La historia y la evolución del mundo de la programación*.  
<https://englobatech.com/la-historia-y-la-evolucion-del-mundo-de-la-programacion/>
- Eumed.net. (n.d). *Enfoques cuantitativo, cualitativo y mixto*. [https://www.eumed.net/tesis-doctorales/2012/mirm/cualitativo\\_cuantitativo\\_mixto.html](https://www.eumed.net/tesis-doctorales/2012/mirm/cualitativo_cuantitativo_mixto.html)
- Faster Capital. (2024). *Prototipo de software*. <https://fastercapital.com/es/contenido/Prototipo-de-software.html>
- Gamboa, A., & Herrera, D. (2012). Desarrollo e implementación del sistema para la gestión y control de información del recurso humano del Ministerio de Agricultura y ganadería en la dirección regional Brunca. [Tesis para obtener el título de Licenciatura en Informática con énfasis en Sistemas de Información] Repositorio Universidad Nacional. Universidad Nacional.
- Garrido, S. (2023). *Las metodologías ágiles más utilizadas y sus ventajas dentro de la empresa*.  
<https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>
- Hidalgo, C. (2016). *Desarrollo e implementación de aplicación de escritorio para inventario y facturación en el restaurante la cabaña típica*. Universidad Politécnica Salesiana de Ecuador, Guayaquil.
- Hostingplus. (n.d). *Algoritmos de programación: conoce los diferentes tipos*.  
<https://www.hostingplus.cl/blog/algoritmos-de-programacion-conoce-los-diferentes-tipos/>
- IBM. (n.d). *¿Qué son las pruebas de software?* <https://www.ibm.com/es-es/topics/software-testing>
- IBM. (n.d). *Definición de casos de uso*. <https://www.ibm.com/docs/es/product-master/12.0.0?topic=processes-defining-use-cases>
- Icaria Techonology. (2023). *Plan de pruebas de software: 8 pasos para realizarlas*.  
<https://icariatechology.com/plan-de-pruebas-de-software/>
- León, M. (n.d). *Investigación cuantitativa vs. cualitativa: una rápida mirada analítica*.  
<https://blogposgrado.ucontinental.edu.pe/investigacion-cuantitativa-vs-cualitativa-rapida-mirada-analitica>
- Lucidchart. (n.d). *Qué es el lenguaje unificado de modelado (UML)*.  
<https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml>
- Maluenda, R. (n.d). *Qué es un algoritmo informático características, tipo y ejemplos*.  
<https://profile.es/blog/que-es-un-algoritmo-informatico/>

- Martínez, M. (n.d). *¿QUÉ SON LOS PARADIGMAS DE PROGRAMACIÓN?*  
<https://profile.es/blog/que-son-los-paradigmas-de-programacion/>
- Martins, J. (2024). *Cómo aplicar un estudio de viabilidad en la gestión de proyectos.*  
<https://asana.com/es/resources/feasibility-study>
- Medium. (n.d). *Requerimientos Funcionales y No Funcionales, ejemplos y tips.*  
<https://medium.com/@requeridosblog/requerimientos-funcionales-y-no-funcionales-ejemplos-y-tips-aa31cb59b22a>
- Mesquita, R. (2019). *¿Qué es un Sistema de Información y cuáles son sus características?*  
 rockcontent: <https://rockcontent.com/es/blog/que-es-un-sistema-de-informacion/>
- Microsoft. (2023). *Guía de escritorio (.NET para Windows Forms).*  
<https://learn.microsoft.com/es-es/dotnet/desktop/winforms/overview/?view=netdesktop-8.0>
- Microsoft. (n.d). *C#.* <https://dotnet.microsoft.com/es-es/languages/csharp>
- Microsoft. (n.d). *Conceptos básicos sobre bases de datos.* <https://support.microsoft.com/es-es/topic/conceptos-b%C3%A1sicos-sobre-bases-de-datos-a849ac16-07c7-4a31-9948-3c8c94a7c204#:~:text=Una%20base%20de%20datos%20es,programa%20de%20procesamiento%20de%20texto.>
- Microsoft. (n.d). *Conozca la familia Visual Studio.* <https://visualstudio.microsoft.com/es/>
- Miro. (n.d). *Cómo escribir buenas historias de usuario en el método Agile.*  
<https://miro.com/es/agile/que-es-historia-usuario/>
- Northware. (n.d). *Requerimientos en el desarrollo de software y aplicaciones.*  
<https://www.northware.mx/blog/requerimientos-en-el-desarrollo-de-software-y-aplicaciones/#:~:text=Generalmente%20hay%20dos%20tipos%20de,c%C3%B3mo%20debe%20comportarse%20el%20sistema.>
- ORACLE. (n.d). *¿Qué es una base de datos?* <https://www.oracle.com/mx/database/what-is-database/#:~:text=Las%20bases%20de%20datos%20de%20autogesti%C3%B3n%20utilizan%20una%20tecnolog%C3%ADa%20basada,otras%20tareas%20de%20gesti%C3%B3n%20rutinarias.>
- Pérez, A. (2016). *Características y fases del modelo incremental.*  
<https://www.obsbusiness.school/blog/caracteristicas-y-fases-del-modelo-incremental>
- Proyectos ágiles.org. (2022). *Beneficios de Scrum.* <https://proyectosagiles.org/beneficios-de-scrum/>
- Rehkopf, M. (n.d). *Historias de usuario con ejemplos y plantilla.*  
<https://www.atlassian.com/es/agile/project-management/user-stories>
- Rivas, A. (2022). *Marco metodológico: ¿Cómo redactar y cuál es su estructura?*  
<https://normasapa.in/marco-metodologico/>
- Rouse, M. (2024). *Sistema de gestión de bases de datos (DBMS).*  
<https://www.techopedia.com/es/definicion/sistema-gestion-bases-datos-dbms>

- Sánchez, P. (n.d). *Métodos de investigación: Qué son y cómo elegirlos*.  
<https://www.questionpro.com/blog/es/metodos-de-investigacion/>
- Santander Universidades. (2021). *Investigación cualitativa y cuantitativa: características, ventajas y limitaciones*. <https://www.santanderopenacademy.com/es/blog/cualitativa-y-cuantitativa.html>
- Scrum México Expertos en Agilidad. (n.d). *Los Valores de Scrum*.  
<https://scrum.mx/informate/valores-de-scrum>
- StarkCloud. (2024). *¿Qué es el Desarrollo de Software?*  
[https://www.google.com/search?q=desarrollo+de+software+qu%C3%A9+es&oq=desarrollo+de+software+qu%C3%A9+es&gs\\_lcrp=EgZjaHJvbWUqBwgAEAAAYgAQyBwgAEAAAYgAQyCAGBEAAAYFhgeMgoIAhAAGA8YFhgeMgoIAxAAGA8YFhgeMgoIBBAA GA8YFhgeMggIBRAAGBYHjIICAYQABgWGB4yCAGHEAAAYFhgeMggICB](https://www.google.com/search?q=desarrollo+de+software+qu%C3%A9+es&oq=desarrollo+de+software+qu%C3%A9+es&gs_lcrp=EgZjaHJvbWUqBwgAEAAAYgAQyBwgAEAAAYgAQyCAGBEAAAYFhgeMgoIAhAAGA8YFhgeMgoIAxAAGA8YFhgeMgoIBBAA GA8YFhgeMggIBRAAGBYHjIICAYQABgWGB4yCAGHEAAAYFhgeMggICB)
- Sydle. (2023). *Extreme Programming: ¿qué es y cómo funciona?*  
<https://www.sydle.com/es/blog/extreme-programming-602ee205da4d096809438c9c>
- Tecnocrática. (2024). *Git en el desarrollo de software*. <https://tecnocratica.net/git-desarrollo-software/>
- Tecsify. (n.d). *Programación Orientada a Objetos*. <https://tecsify.com/blog/infografia/poo/>
- Tecsify. (n.d). *Programación Orientada a Objetos*. <https://tecsify.com/blog/infografia/poo/>
- TIOBE. (2024). *TIOBE Index for June 2024*. <https://www.tiobe.com/tiobe-index/>
- Torresburriel Estudio. (2018). *Beneficios de las user stories*.  
<https://torresburriel.com/weblog/beneficios-de-las-user-stories/>
- UNIR. (2023). *¿Qué es la programación orientada a objetos?*  
<https://unirfp.unir.net/revista/ingenieria-y-tecnologia/programacion-orientada-objetos/>
- Valencia, U. d. (n.d). *¿Qué son las fuentes de información?*  
[https://www.uv.es/cibisoc/tutoriales/trabajo\\_social/22\\_las\\_fuentes\\_de\\_informacin.html#:~:text=Por%20fuente%20de%20informaci%C3%B3n%20se,para%20satisfacer%20una%20necesidad%20informativa.](https://www.uv.es/cibisoc/tutoriales/trabajo_social/22_las_fuentes_de_informacin.html#:~:text=Por%20fuente%20de%20informaci%C3%B3n%20se,para%20satisfacer%20una%20necesidad%20informativa.)
- Vega, H. (2021). *Curso de Historia de la Programación: Lenguajes y Paradigmas*.  
<https://platzi.com/clases/2211-historia-programacion/35073-cuales-son-los-principales-paradigmas-de-programac/>
- Velázquez, A. (2018). *Aplicación web basada en el patrón MVC para mejorar la gestión de almacén de la empresa CISESAC. [Tesis para obtener el título profesional de ingeniero de sistemas] Repositorio de la UCV*. Universidad César Vallejo.

## **Anexo 1. Carta de revisión filológica**