

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

**CARRERA
INGENIERÍA DE SISTEMAS**

**Trabajo de titulación previo a la obtención del título de:
Ingenieros de Sistemas**

TEMA:

**“DESARROLLO DE UN SOFTWARE PARA LA EMPRESA “LETERAGO
DEL ECUADOR S.A.” QUE PERMITIRÁ LA GESTION DE INVENTARIO DE
LICENCIAS PARA EL DEPARTAMENTO DE TI”**

AUTORES:

**JONATHAN ANDRÉS ÁLVAREZ FLORES
ALEXIS MARCELO OLMEDO PACHACAMA**

TUTOR:

JOSÉ LUIS AGUAYO MORALES

Quito, julio del 2021

CESIÓN DE DERECHOS DE AUTOR

Nosotros Jonathan Andrés Álvarez Flores, Alexis Marcelo Olmedo Pachacama, con documento de identificación N° 1721529707 y N° 1725983447, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación intitulado: “DESARROLLO DE UN SOFTWARE PARA LA EMPRESA “LETERAGO DEL ECUADOR S.A.” QUE PERMITIRÁ LA GESTION DE INVENTARIO DE LICENCIAS PARA EL DEPARTAMENTO DE TI”, mismo que ha sido desarrollado para optar por el título de: INGENIEROS DE SISTEMAS, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribo este documento en el momento que hago entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Jonathan Andrés Álvarez Flores
1721529707

Alexis Marcelo Olmedo Pachacama
1725983447

Quito, julio del 2021

DECLARATORIA DE COAUTORÍA DEL TUTOR

Yo declaro que bajo mi dirección y asesoría fue desarrollado el Proyecto Técnico, “DESARROLLO DE UN SOFTWARE PARA LA EMPRESA “LETERAGO DEL ECUADOR S.A.” QUE PERMITIRÁ LA GESTION DE INVENTARIO DE LICENCIAS PARA EL DEPARTAMENTO DE TI”, realizado por Jonathan Andrés Álvarez Flores y Alexis Marcelo Olmedo Pachacama, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerado como trabajo final de titulación.

Quito, julio del 2021

A handwritten signature in blue ink, appearing to be 'José Luis Aguayo Morales', written over a circular stamp or seal.

José Luis Aguayo Morales
C.I.: 1709562597

DEDICATORIA

Este trabajo primeramente lo dedico a Dios, quien me ha dado la vida, salud, la fuerza y voluntad para salir adelante. Levantándome día a día con las ganas de superarme en el ámbito académico, aun cuando el camino se puso difícil, recibí la fuerza para seguir y no desmayar en el transcurso de mi carrera. También agradezco a mi madre Jenny Flores por siempre ser mi apoyo y darme siempre los mejores consejos para ser una mejor persona, de igual manera a mi padre Andrés Álvarez por el gran esfuerzo realizado durante toda mi carrera en la universidad y ser un ejemplo para mí y en general a los dos por su esfuerzo, amor, sacrificio y aliento para que continúe estudiando, atentos siempre a lo que necesitaba.

A mis abuelos, especialmente a Alonso Flores y Matilde Cahueñas que han sido como mis padres durante toda mi vida, a mis hermanas Jennifer, Raquel y Melanie, de igual manera a Marco García por haberme aceptado como un hijo más y apoyarme siempre, y en general a toda mi familia que siempre confió en mí y en mis capacidades.

A mi novia Tania por cada uno de los consejos y apoyarme en los momentos difíciles y confiar en mí, ayudándome a superarme tanto como persona y profesionalmente.

Jonathan Andrés Álvarez Flores

DEDICATORIA

El presente trabajo lo dedico primero a Dios ya que él ha sabido darme fuerzas para seguir adelante en los momentos complicados, a mis padres Patricio Cadena y Sylvia Pachacama ya que sin ellos esto no habría sido posible en especial a mi madre Sylvia Pachacama la cual ha tenido la certeza para saber guiarme por un buen camino, a mi hermana Mayra Olmedo a mis sobrinos Matías Pilacuán y Sophie Tamayo que me impulsan a seguir mejorando tanto el ámbito educativo como en lo personal, a mi familia que ha sabido aconsejarme de buena manera para llegar a cumplir con este objetivo. Del mismo modo dedico este trabajo a mis seres queridos que por situaciones de la vida ya no se encuentran presentes en especial a mi abuelito quien supo dejarme bien claro el camino que se debe seguir en la vida.

Alexis Marcelo Olmedo Pachacama

AGRADECIMIENTO

Agradecemos de manera conjunta a nuestro ingeniero tutor José Luis Aguayo Morales por guiarnos y hacer posible el desarrollo de este proyecto, del mismo modo agradecemos a cada uno de nuestros profesores que supieron compartirnos sus conocimientos y también agradecemos a la Universidad Politécnica Salesiana porque aparte de formarnos profesionalmente nos ha sabido inculcar valores personales durante el transcurso de nuestra vida estudiantil en la institución formándonos como profesional y como personas, así como agradecemos a nuestros amigos los cuales estuvieron presentes en cada aventura a lo largo de este proceso.

Jonathan Andrés Álvarez Flores y Alexis Marcelo Olmedo Pachacama

ÍNDICE GENERAL

INTRODUCCIÓN.....	1
ANTECEDENTES	1
PROBLEMA	2
JUSTIFICACIÓN.....	3
OBJETIVOS.....	3
Objetivo general	3
Objetivos específicos.....	3
METODOLOGÍA.....	4
CAPÍTULO 1	7
SITUACIÓN ACTUAL	7
CAPÍTULO 2	9
2.1 FUNDAMENTOS TEÓRICOS	9
2.1.1 Antecedentes.....	9
2.1.2 Software.....	9
2.1.3 Licencias de software	10
2.1.4 Software propietario	10
2.1.5 Software libre	10
2.1.6 Programación de computadores	10
2.1.7 Programación en lenguaje de alto nivel.....	10

2.1.8 Base de datos	11
2.1.9 MYSQL	11
2.1.10 HTML.....	11
2.1.11 CSS.....	11
2.1.12 JavaScript	12
2.1.13 Node	12
2.1.14 Express	13
2.1.15 Node Cron	13
2.1.16 Nodemailer	13
2.1.17 React.....	13
2.1.18 Librería multer.....	14
2.1.19 Librerías de software	14
2.1.20 Modelo Vista Controlador.....	14
2.1.21 Lenguaje de modelado UML.....	14
2.1.22 Diagrama de casos de uso.....	15
2.1.23 Diagrama de clases	17
2.1.24 Servidor	18
2.1.25 Windows server	19
CAPÍTULO 3	20
3.1 DISEÑO DE LA SOLUCIÓN.....	20
3.1.1 Diagrama de caso de usos del administrador	20
3.1.2 Plantilla de descripción del administrador	21

3.1.3	Diagrama de caso de usos del usuario consulta.....	24
3.1.4	Plantilla de descripción del usuario consulta.....	25
3.1.5	Diagrama de clases	26
3.2	BOCETO DE LA BASE DE DATOS.....	27
3.2.1	Descripción de tablas	27
3.3	DISEÑO DEL SISTEMA.....	30
3.3.1	Preparación del entorno de desarrollo	30
CAPÍTULO 4		46
4.1	ANÁLISIS DE RESULTADOS.....	46
4.1.1	Pruebas de caja negra.....	46
4.1.2	Pruebas de seguridad	53
CONCLUSIONES.....		57
RECOMENDACIONES		58
LISTA DE REFERENCIAS.....		59

ÍNDICE DE TABLAS

Tabla 1 Documentación de caso de uso del administrador	21
Tabla 2 Documentación de caso de uso del usuario consulta.	25

ÍNDICE DE FIGURAS

Figura 1 Flujo de proceso SCRUM.....	4
Figura 2 Características de SCRUM	5
Figura 3 Roles de SCRUM.....	6
Figura 4 Misión	8
Figura 5 Visión.....	8
Figura 6 Ventajas de Node	12
Figura 7 Representación gráfica de la relación de asociación.....	16
Figura 8 Representación gráfica de la relación de generalización	16
Figura 9 Gráfico de la dependencia extend	17
Figura 10 Gráfico de la dependencia include	17
Figura 11 Ejemplo de clase	18
Figura 12 Diagrama de caso de uso del administrador.....	20
Figura 13 Diagrama de caso de uso del usuario consulta.....	24
Figura 14 Diagrama de clases.....	26
Figura 15 Esquema de la base de datos	27
Figura 16 Página de inicio para la descarga de Visual Studio Code	31
Figura 17 Pantalla de inicio Visual Studio Code.....	32
Figura 18 Página de inicio para descargar MYSQL.....	33
Figura 19 Ventana de inicio del instalador de MYSQL	34
Figura 20 Chequeo de requerimientos MYSQL.....	35
Figura 21 Instalación de requerimientos	36
Figura 22 Tipo y red de MYSQL	37
Figura 23 Método de autenticación MYSQL	38
Figura 24 Asignación de clave MYSQL	39

Figura 25 Finalización de la instalación de MYSQL	40
Figura 26 Ventana de inicio de MYSQL.....	41
Figura 27 Página de descarga de Node.js	42
Figura 28 Inicio de instalación de Node.js	43
Figura 29 Licencia de Node.js.....	44
Figura 30 Instalación de Node.js	45
Figura 31 Final del proceso de instalación de Node.js.....	45
Figura 32 Ventana de registro de licencias con errores.....	47
Figura 33 Ingreso de licencias correcto	48
Figura 34 Primera parte de los errores de la ventana agregar usuarios	49
Figura 35 Segunda parte de los errores de la ventana agregar usuarios	49
Figura 36 Registro de usuario exitoso	50
Figura 37 Registro de proveedor con error.....	51
Figura 38 Registro exitoso de proveedor.....	51
Figura 39 Registro de ayuda fallido	52
Figura 40 Registro de ayuda exitoso	52
Figura 41 Inyección SQL al login del sistema	53
Figura 42 Inyección SQL al campo contraseña.....	54
Figura 43 Página de login del sistema.....	54
Figura 44 Inyección JS al sistema	55
Figura 45 Inyección JS fallida.....	55
Figura 46 Inyección JS para intentar cambiar el nombre del usuario	56
Figura 47 Inyección JS para editar usuario fallida	56

Resumen

Este proyecto tuvo como objetivo desarrollar un software que facilite el cumplimiento de la gestión de inventarios de licencias en el departamento de TI de la empresa "Leterago del Ecuador SA", apoyado por la metodología ágil SCRUM, ya que en el proceso se involucró a todos los interesados y desarrolladores.

Este proyecto fue respuesta a la necesidad de automatizar el proceso de gestión de licencias ya que actualmente el departamento de TI de la empresa cuenta con un control manual de licencias, el cual puede estar sujeto a errores o pérdida de información.

El documento se ha dividido en cuatro capítulos. El capítulo uno explica la situación actual de la empresa, el capítulo dos trae algunos conceptos teóricos de los componentes utilizados, el capítulo tres muestra el diseño de la solución y el capítulo cuatro reporta los resultados de las pruebas de software.

Esta aplicación para la administración del control de licencias de software ha supuesto un gran cambio ya que el producto final reduce y automatiza el trabajo respecto a cómo se hacía anteriormente.

Abstract

This project was having the aim of developing software that facilitates compliance with license inventory management in the IT department of the company "Leterago del Ecuador SA", aided by the agile SCRUM methodology, since throughout stakeholders and developers were involved in the process.

This project was answer to the need to automate license management process since currently the IT department of the company has a license manual control, which may be subject to errors or loss of information.

The document has been divided into four chapters. Chapter one explains the current situation of the company, chapter two brings some theoretical concepts of the components used, chapter three shows the design of the solution and chapter four report the results of the software tests.

This application to the administration of the control of software licenses has made a great change since the final product reduces and automated the work compared to how it was done previously.

INTRODUCCIÓN

ANTECEDENTES

La empresa “LETERAGO DEL ECUADOR S.A.” al dedicarse a la exportación, compra, venta, distribución y fabricación de medicamentos de uso humano y veterinario, es decir a la comercialización de productos farmacéuticos. Para llevar a cabo con éxito las actividades mencionadas usan herramientas tecnológicas es decir diferentes softwares en los cuales actualmente no se está llevando un control riguroso sobre su licenciamiento lo cual puede ocasionar problemas como la pérdida de información entre otras, es por esto por lo que se necesita crear un software el cual esté únicamente encargado de llevar un control riguroso sobre cada una de las licencias de los softwares utilizados en esta empresa.

Conociendo estos antecedentes el presente trabajo está enfocado en el desarrollo de un software cuya función es facilitar el control de las licencias de software utilizadas en el departamento de TI de la empresa mencionada, este software será capaz de llevar a cabo una correcta administración de todas las herramientas tecnológicas así como proveerá la facilidad de la generación de reportes para que la información no se encuentre solamente de manera digital sino de manera física en caso de que sea necesario.

Para el desarrollo de este software se utilizará el lenguaje de programación de alto nivel JavaScript ayudado del framework Node.js y del mismo modo la utilización de ciertas librerías las cuales facilitaran cumplir con los requerimientos planteados, estos conceptos están detallados en el capítulo 2 del presente documento.

En el capítulo 3 se detalla la funcionalidad total del software ayudado de los distintos diagramas UML. Estos diagramas cumplirán con el objetivo de facilitar a los usuarios la correcta utilización del software detallando paso a paso su uso de manera gráfica.

PROBLEMA

Dentro de la empresa “LETERAGO DEL ECUADOR S.A.” en su departamento de TI a pesar de contar con un control manual de sus licencias de software el cual consiste en el registro de manera manual de sus licencias en una plantilla ya específica realizada en Microsoft Word la cual cuenta de los siguientes campos: número, descripción del sistema, fabricante/proveedor, licenciado (si/no), responsable T.I y observaciones. Al ser un control manual se está propenso a errores humanos es decir a no tener una información integra, a la acumulación innecesario de papel y a futuras pérdidas de registros ya que esta plantilla es la única evidencia del registro y control de las licencias de software. Debido a esto se han visto inconvenientes en el momento de manejar los diferentes softwares los cuales necesitan su respectivo licenciamiento, provocando que no se pueda tener un control riguroso y exacto, poniendo en riesgo tanto los programas como los datos de los usuarios al no estar alertados sobre la próxima caducidad de las diferentes licencias de softwares.

Con el presente proyecto técnico se pretende desarrollar e implementar un software que permita a la empresa “LETERAGO DEL ECUADOR S.A.” mantener el control de las licencias de software manejadas dentro del departamento de TI, obteniendo un mayor control y de esta manera asegurando que todos los softwares se encuentren funcionando de manera correcta.

Al mismo tiempo el desarrollo de este sistema adjudicara ciertas ventajas al departamento de TI. La principal ventaja de este sistema es la automatización del control de inventarios de licencias, y gracias a esto se podrán realizar búsquedas, realizar reportes, obtener alertas cuando una licencia esté a punto de caducar, crear modificar o eliminar licencias de manera dinámica con el objetivo de acelerar el trabajo del personal que manejará este sistema.

JUSTIFICACIÓN

Este proyecto técnico es factible ya que mientras más ayuda tecnológica se tiene, más se facilita el trabajo, en este caso el software encargado de controlar las licencias será dinámico y proveerá una ventaja tecnológica sobre las demás empresas que no cuentan con este servicio.

Al contar con una ventaja tecnológica sobre la competencia se verá beneficiada la empresa “LETERAGO DEL ECUADOR S.A.” en especial su departamento de TI, ya que se gestionarán las licencias de manera más eficaz y eficiente a como se lo estaba llevando a cabo.

En base a los requerimientos del proyecto el software desarrollado será capaz de gestionar las licencias de todos los programas que están registrados en el departamento de TI, teniendo en cuenta que el software desarrollado estará instalado de manera local.

OBJETIVOS

Objetivo general

Desarrollar un software para la empresa “LETERAGO DEL ECUADOR S.A.” que permitirá la gestión de inventario de licencias para el departamento de TI.

Objetivos específicos

- Diseñar una base de datos en la cual se pueda relacionar toda la información requerida de una manera apropiada para el correcto funcionamiento del software.
- Generar alertas por medio de correos notificando que una licencia de software está próxima a caducarse.
- Implementar el software dentro del departamento de TI de la empresa “LETERAGO DEL ECUADOR S.A.”.

- Generar reportes en formato Excel y pdf con el objetivo de facilitar la información en caso de ser necesario

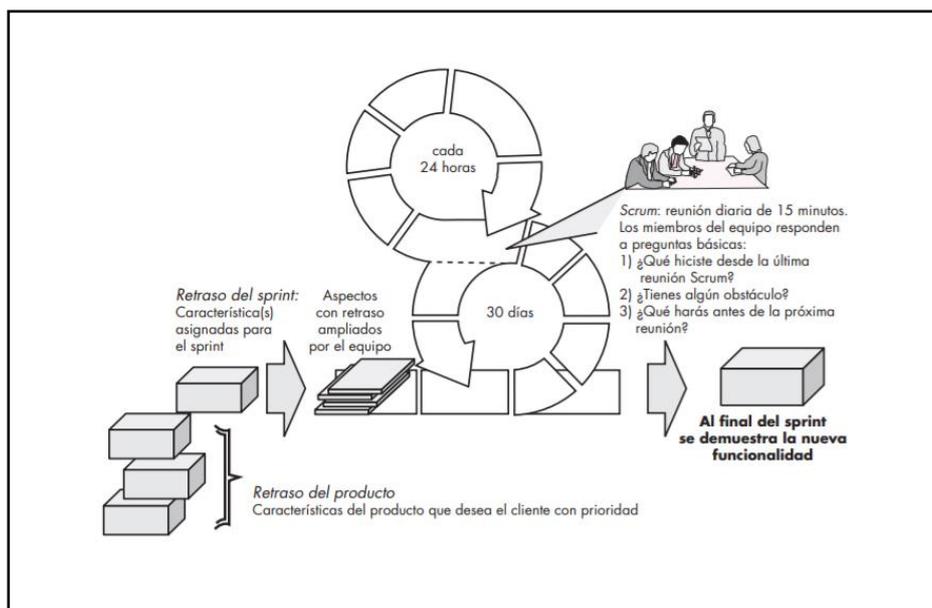
METODOLOGÍA

Se utilizará una metodología ágil en este caso SCRUM ya que es un método para trabajar en equipo, en otras palabras, durante el desarrollo del software se verán sumamente involucrados tanto los interesados como los desarrolladores. Al utilizar esta metodología se asume que los requisitos podrán ir variando con el fin de obtener el mejor resultado posible en la culminación del proyecto.

Los principios de SCRUM son consistentes con el marco de capacitación utilizado para guiar las actividades de desarrollo en el proceso de investigación, que incluye las funciones del proceso de demanda, investigación, diseño, evolución y entrega. En cada proceso, la carga de trabajo se genera en un proceso de implementación llamado sprint (Pressman R. , 2010).

Figura 1

Flujo de proceso SCRUM



Nota. La figura 1 muestra el flujo de proceso de SCRUM. Fuente: (Pressman R. , 2010).

Retraso: es una lista en la cual se encuentran las prioridades de los requerimientos o características del proyecto que dan al cliente un valor de negocio.

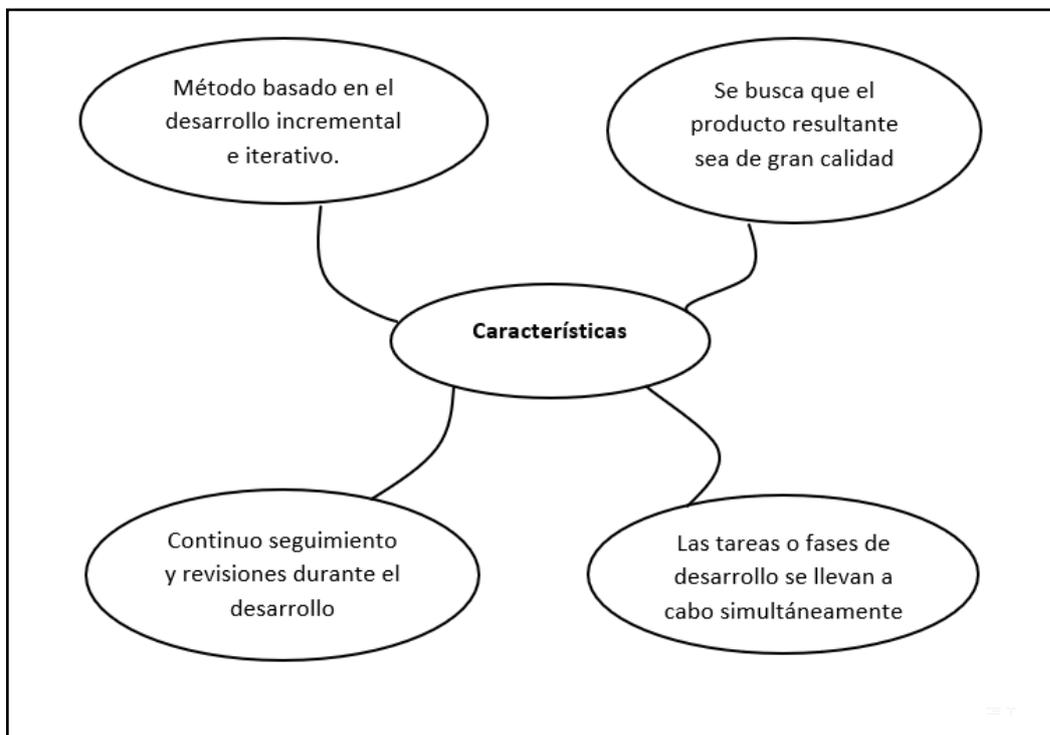
Sprint: son unidades de trabajo necesarias para cumplir un requerimiento causado por una demora por lo que debe ajustarse a una caja de tiempo antes establecida, además durante el sprint no se introducen cambios, permitiendo a los miembros del equipo trabajar en un ambiente estable aunque de corto plazo.

Reuniones Scrum: reuniones breves con un tiempo estimado de 15 minutos. Beedle y sus colegas [Bee99] presentan un análisis exhaustivo de estos patrones. Los patrones de proceso Scrum permiten que un equipo de software trabaje con éxito en un mundo en el que es imposible eliminar la incertidumbre.

Características

Figura 2

Características de SCRUM



Nota. En la figura 2 se muestran algunas de las características de SCRUM. Fuente: (Torrado, 2019).

Roles

Cada integrante del equipo cumple un rol al cual se le adjuntan responsabilidades que se deben cumplir en el transcurso del proyecto. Según (Torrado, 2019) los siguientes son los roles indispensables, que se muestran en la figura 3:

Figura 3

Roles de SCRUM

- **Product Owner o dueño del producto:** Persona que ocupa el rol principal en el proyecto. Entre sus responsabilidades se encuentra la optimización del proyecto, así como la inspección de cada avance del producto.
- **Scrum Master.** Persona líder del equipo de trabajo cuya función es inducir la metodología y supervisarla para que se cumplan de manera correcta sus principios. El Scrum Master también se encarga de organizar las reuniones.
- **Scrum Team:** Personas encargadas del desarrollo del producto final, este equipo puede estar formado por ingenieros, programadores entre otros. El equipo de desarrollo debe ser capaz de abastecer la organización del proyecto cumpliendo con las diferentes actividades fijadas.

Nota. En la figura 3 se especifican los roles que existen en SCRUM. Fuente: (Torrado, 2019).

CAPÍTULO 1

SITUACIÓN ACTUAL

Con el avance de la tecnología los procesos que se llevaban manualmente se han ido automatizando, es decir se han ido transformando para facilitar sus diferentes funciones. En el caso del departamento de TI de la empresa “LETERAGO DEL ECUADOR S.A.” no ha sucedido esto, ya que al momento de llevar a cabo un control sobre sus licencias de programas que son usados se los registra de manera manual, ya que actualmente cada nueva licencia de software es registrada en una hoja la cual lleva su respectivo formato.

La sede principal de la empresa se encuentra localizada en Quito en la Av. Manuel Córdova Galarza KM 7.5. En dicha sede su departamento de T.I actualmente realiza el control de las licencias de software de manera manual es decir mediante la utilización de una plantilla como se mencionó anteriormente en la cual se detalla el nombre del sistema, la descripción del sistema, el fabricante o el proveedor, si se encuentra o no licenciada, su responsable en el departamento de T.I y sus respectivas observaciones. Cabe recalcar que estas licencias no cuentan con ningún formato de orden en otras palabras no se encuentran ordenadas ni alfabéticamente ni obedecen a ninguna otra característica.

Como es un proceso realizado manualmente esta propenso a que se cometan gran cantidad de errores humanos, esto ocasiona que no se pueda tener información precisa acerca de los detalles de ciertas licencias.

Del mismo modo el modelo actual de la gestión de licencias de software conlleva un desgaste innecesario de hojas de papel bond, pero no solo esto ya que se corre el riesgo de que parte de la información o en el peor de los casos la información completa se pierda. Se debe tener en cuenta que esta información es importante y si se pierde tendrá una gran consecuencia en el departamento de TI.

El objetivo del software que se desarrolló es poder ayudar a cumplir de manera más eficiente la misión y la visión de la empresa detalladas a continuación.

Figura 4

Misión



Nota. En la figura 4 se muestra la misión de la empresa. Fuente: (Leterago).

Figura 5

Visión



Nota. En la figura 5 se muestra la visión de la empresa. Fuente: (Leterago).

CAPÍTULO 2

2.1 FUNDAMENTOS TEÓRICOS

Es de vital importancia conocer algunos conceptos teóricos para el correcto desarrollo del actual proyecto técnico, con el objetivo de facilitar los conceptos principales de las herramientas que se están utilizando para su desarrollo.

2.1.1 Antecedentes

Actualmente los avances tecnológicos proporcionan simplicidad al momento de realizar ciertas actividades diarias como es la administración de diferentes procesos llevados a cabo de manera manual. Una de las formas en la cual los avances tecnológicos proporcionan ayuda es mediante la utilización de procesos automatizados los cuales en su mayoría se los realiza por medio de la implementación de software el cual se encuentra destinado a cumplir con una actividad en específico (Pressman R. , 2010).

El software es considerado un activo en la empresa, ya que gracias a este se obtiene cierta ventaja tecnológica sobre otras. Teniendo en cuenta que actualmente el software es un componente dominante en las economías del ámbito tecnológico (Pressman R. , 2010). También se considera un producto por lo que cuenta con sus respectivas licencias las cuales son las encargadas de verificar si el software fue adquirido de manera legal.

2.1.2 Software

El software tiene varias definiciones, pero en este caso la que mejor lo describe es que el software es un grupo de instrucciones que cuando se ejecutan proporcionan las características, función y desempeño buscados, así como estructuras de datos que permiten que los programas manipulen en forma adecuada la información (Pressman R. , 2010).

2.1.3 Licencias de software

Son un acuerdo entre el fabricante y el usuario que establece los derechos y obligaciones de cada una de las partes, por tal motivo al instalar un software con su respectiva licencia, el usuario acepta las condiciones establecidas en los términos del contrato (Brocca, 2005).

2.1.4 Software propietario

Creado con el objetivo de obtener réditos económicos mediante la comercialización de las distintas licencias. El mercadeo de estas se realiza mediante la entrega de un programa ejecutable con su respectivo contrato (Brocca, 2005).

2.1.5 Software libre

Es una licencia que permite a los usuarios copiarlo, distribuirlo, estudiarlo incluso hasta modificar el código, teniendo en cuenta que software libre es diferente que software gratis por lo cual el software libre puede comercializarse proponiendo un sistema donde los réditos económicos son el resultado de un trabajo eficiente (Brocca, 2005).

2.1.6 Programación de computadores

Desarrollo de instrucciones las cuales son ejecutadas por el computador, estas instrucciones ejecutadas de manera ordenada son conocidas como programa. “En términos coloquiales, los programas suelen pedir algunos datos al usuario, procesar estos datos de algún modo y devolver al usuario el resultado” (Rodríguez y Llanos, 2010).

2.1.7 Programación en lenguaje de alto nivel

Es la escritura de un programa en un lenguaje simbólico que actúa de forma individual al grupo de instrucciones que el procesador es capaz de ejecutar. La programación de alto nivel se ayuda de un compilador siendo este el encargado verificar el programa y ejecutarlo en caso de que se encuentre escrito de manera correcta (Rodríguez y Llanos, 2010)

2.1.8 Base de datos

Según el autor (Cobo, s.f.) una base de datos es un grupo de datos almacenados en un solo lugar de manera única en un recurso informático, accesible para distintos usuarios y aplicaciones al mismo tiempo.

2.1.9 MYSQL

Considerado como uno de los sistemas de gestión de base de datos relacionados escrito en código abierto más utilizados en el mundo. Fue desarrollado en los lenguajes de programación C y C++, por lo que son compatibles con varias versiones de Windows y distribuciones de Linux (Stepantsov, 2018).

Según (Stepantsov, 2018) indica que las siguientes son algunas razones por las que MYSQL alcanzó un gran reconocimiento:

- Confiabilidad.
- Escalado.
- Seguridad.

2.1.10 HTML

Es un lenguaje formado mediante etiquetas definidas con un nombre específico envuelto en paréntesis angulares, la función de estos paréntesis es delimitar la etiqueta y el nombre es el encargado de identificar el tipo de contenido al cual se está haciendo referencia (Gauchat, 2012).

2.1.11 CSS

Lenguaje cuyo propósito es establecer la apariencia o estilos de los elementos HTML, como fuente, tamaño, entre otros. CSS es utilizado para establecer estilos únicos y personalizados al gusto de cada persona mediante el uso de sus propiedades y valores (Gauchat, 2012).

2.1.12 JavaScript

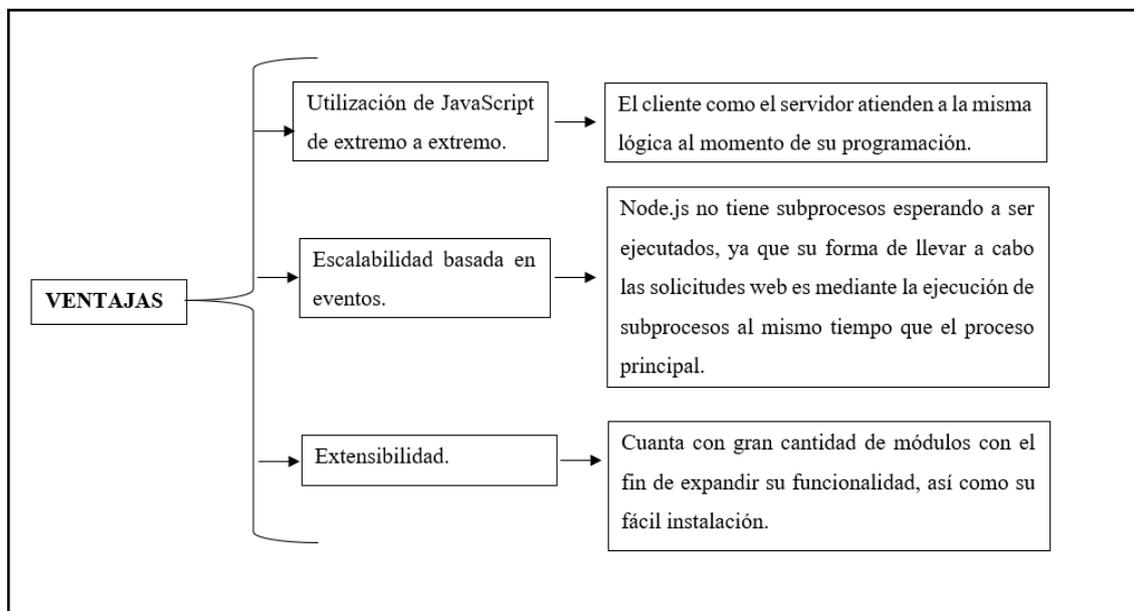
JavaScript es un lenguaje de programación por lo que se diferencia de HTML y CSS, en otras palabras es semejante a los diferentes lenguajes de programación de alto nivel como C++ o Java. La característica que diferencia a JavaScript de los demás lenguajes de programación es su capacidad de realizar tareas personalizadas al tratar su contenido dinámicamente (Gauchat, 2012).

2.1.13 Node

Igualmente conocido a modo de Node.js es un framework de desarrollo basado en el motor de Google JavaScript v8, el cual funciona como un compilador ya que el código de Node.js es desarrollado en JavaScript y posteriormente su V8 lo ejecuta. Dando como resultado que los scripts se compilen dentro la misma aplicación en la parte del servidor, permitiendo una rigurosa integración entre el servidor web y los scripts (Dayley, 2018).

Figura 6

Ventajas de Node



Nota. En la figura 6 se muestran algunas de las ventajas de Node. Fuente: (Dayley, 2018).

2.1.14 Express

Es un framework que trabaja en lado del servidor utilizado para la elaboración de aplicaciones web a partir de Node. Añade a Node una serie de características web las cuales facilitan métodos HTTP y funcionalidad de middleware añadiendo diferentes componentes para trabajar conjuntamente. Una de las características de este módulo es que no concreta requisitos ni aplica restricciones para la integración de cualquier tipo de base de datos, es decir se puede utilizar una base de datos relacional como PostgreSQL o una base de datos no SQL como MongoDB (Hoque, 2020).

2.1.15 Node Cron

Es un pequeño modulo programador de tareas desarrollado en JavaScript puro para que sea compatible con Node.js. La función principal de este módulo es permitir programar tareas automáticas en Node.js mediante el uso de una sintaxis crontab completa (Merencia, 2021).

2.1.16 Nodemailer

Módulo que se debe incorporar en Node.js para hacer posible el envío automático de emails del mismo modo puede ser utilizado para recibir mensajes, enviar correos de información (Uesteibar, 2015). La utilización de este módulo en este proyecto técnico es la de enviar mensajes.

2.1.17 React

React o React.js, pertenece a una librería de JavaScript cuyo objetivo es el desarrollo de interfaces de usuarios. React pretende facilitar la creación de aplicaciones web sofisticadas ya que únicamente se preocupa de la interfaz de usuario, debido a esto se lo puede combinar con otras librerías de JavaScript o frameworks como es el caso de Node.js (Jimenez, 2016).

2.1.18 Librería multer

Multer cumple la función de un middleware cuyo objetivo es enlazar distintas aplicaciones de software (Vaati, 2018). Para esta ocasión multer será utilizado como una ayuda para el proceso de carga de archivos.

2.1.19 Librerías de software

Son archivos escritos sobre un lenguaje de programación, cuyo propósito es añadir ciertas funcionalidades al lenguaje de programación con el objetivo de reducir el tiempo de desarrollo y aumentar la calidad de este (tiThink, 2018) .

2.1.20 Modelo Vista Controlador

Conocido también por sus iniciales MVC (Modelo Vista Controlador) es una arquitectura basada en capas, el cual es muy utilizado en la creación de software para aplicaciones web. Se conoce que el modelo (M) es el encargado de atender todo lo que se encuentra vinculado con los datos mientras que la vista (V) se encarga de comunicarlo con las distintas interfaces de usuario y por último el controlador (C) es el encargado de manipular el modelo (M) con el fin de mostrar información en la vista (V) (Ávila, 2019).

2.1.21 Lenguaje de modelado UML

Se debe tener claro el concepto de modelo. Según (Parga, 2015) un modelo es considerado una idealización de un inconveniente existente. Es por esto por lo que surge el concepto de modelar, entonces se dice que modelar es la acción de obtener las características principales de un inconveniente existente con una intención establecida. En este caso se construyen los modelos con el objetivo de asegurar el correcto funcionamiento del producto final.

Una vez se ha establecido el concepto sobre modelar, se dice que el lenguaje unificado de modelado (UML) es uno de los lenguajes de modelado definido por Object

Management Group (OMG) diseñado como un lenguaje gráfico cuyo objetivo es la construcción del modelado del sistema, este lenguaje es utilizado para modelar distintos tipos de aplicaciones y plataformas (Fuentes y Vallecillo , s.f).

2.1.22 Diagrama de casos de uso

Para entender que es un diagrama de casos de uso hay que contestarse algunas preguntas. La primera de ellas es: ¿Qué es un caso de uso?

Un caso de uso se puede considerar como una agrupación de escenarios que describen de forma exacta el proceso paso a paso de cómo utilizar un sistema, proporcionando una explicación precisa de las acciones que puede realizar cada actor del sistema (Gutierrez, 2011).

De la definición de caso de uso surge la necesidad de comprender el concepto de otras palabras, las cuales forman parte indispensable sobre los diagramas de caso de uso (Gutierrez, 2011):

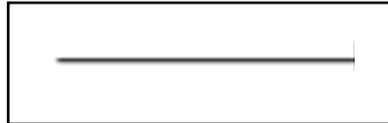
- **Escenario:** Considerado como una sucesión de operaciones entre los usuarios y el sistema a desarrollar.
- **Actor o rol:** Representa el rol asignado hacia una persona, mediante el cual interactúa con el sistema. Por lo general dentro de cada sistema existe más de un rol, pero los roles principales son los del administrador el cual podrá realizar todas las actividades sin ninguna restricción dentro del sistema y un usuario normal el cual tendrá ciertas restricciones según sea lo solicitado.

Para completar con los elementos de los diagramas de casos de uso se explicará acerca de las relaciones que pueden existir, según (Lázaro, 2009) las relaciones entre elementos son las siguientes:

- **Asociación:** Representada por una línea continua sin puntas en ninguna de los extremos, indica la participación de un actor sobre un caso de uso.

Figura 7

Representación gráfica de la relación de asociación

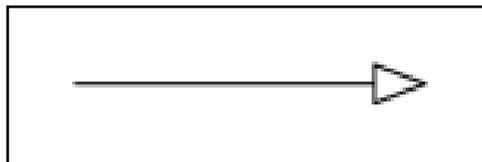


Nota. La figura 7 muestra la representación gráfica de la relación de asociación en el diagrama de casos de uso. Fuente: (Gutierrez, 2011).

- **Generalización:** Representada por una línea continua con una flecha en una de sus puntas, indica la relación entre un caso de uso general con uno específico.

Figura 8

Representación gráfica de la relación de generalización



Nota. La figura 8 muestra la representación gráfica de la relación de generalización que se usa dentro del diagrama de casos de uso. Fuente: (Gutierrez, 2011).

- **Dependencia:** Dentro de la dependencia existen dos condiciones.
 - **<<Extend>>:** Es una función opcional, la figura 9 muestra su representación gráfica.

Figura 9

Gráfico de la dependencia extend

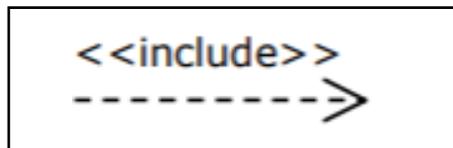


Nota. La figura 9 muestra la representación gráfica de la relación de dependencia extend que se usa dentro del diagrama de casos de uso. Fuente: (Gutierrez, 2011).

- **<<Include>>**: Ocurre cuando se hace una llamada obligatoria de un caso de uso, la siguiente figura 10 muestra su representación gráfica.

Figura 10

Gráfico de la dependencia include



Nota. La figura 10 muestra la representación gráfica de la relación de dependencia include que se usa dentro del diagrama de casos de uso. Fuente: (Gutierrez, 2011).

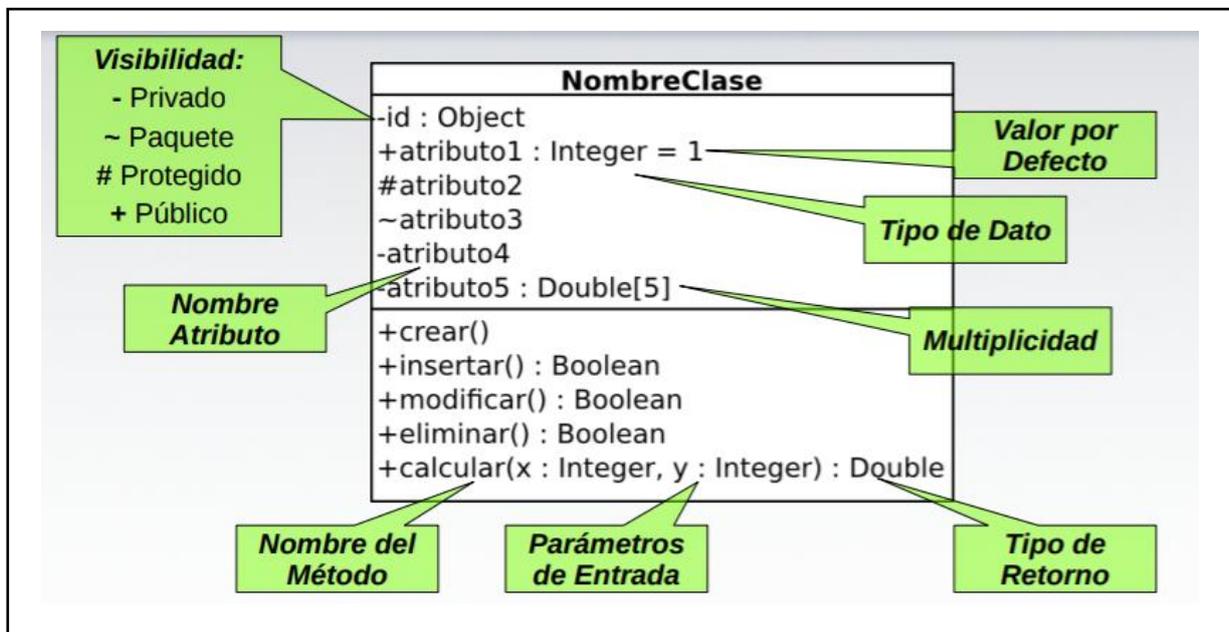
2.1.23 Diagrama de clases

El objetivo del diagrama de clases es mostrar la composición fija del sistema a modelar, del mismo modo mostrar las clases y objetos del sistema con sus respectivas relaciones. En otras palabras, el diagrama de clases es el encargado de justificar la organización del boceto que se realizará. El diagrama de clases está compuesto de diferentes componentes. (Gutierrez, 2011).

- **Clase:** Una clase determina la distribución y conducta de una agrupación de objetos, los cuales están formados de un patrón de distribución semejante y un comportamiento similar.
- **Atributos:** Los atributos son pertenencias notables de una clase, los cuales representan su organización. Los atributos pueden ser simples o compuestos.
- **Métodos:** Son las conductas enlazadas a una clase.

Figura 11

Ejemplo de clase



Nota. La figura 11 muestra el ejemplo de una clase en la cual se detalla los atributos y métodos.

Fuente: (Gutierrez, 2011).

2.1.24 Servidor

Recurso computacional cuyo propósito es atender las peticiones realizadas por los clientes, esencialmente un servidor es una computadora con gran capacidad tecnológica que ayudará con la gestión de la red, una página o a controlar los recursos compartidos (Medellín, 2016).

2.1.25 Windows server

Es un sistema operativo casi igual al Microsoft Windows normal con la diferencia de que Windows Server está enfocado en otorgar servicios de red de manera eficaz, vertiginosa e inequívoca hacia distintos equipos informáticos (León, 2019).

CAPÍTULO 3

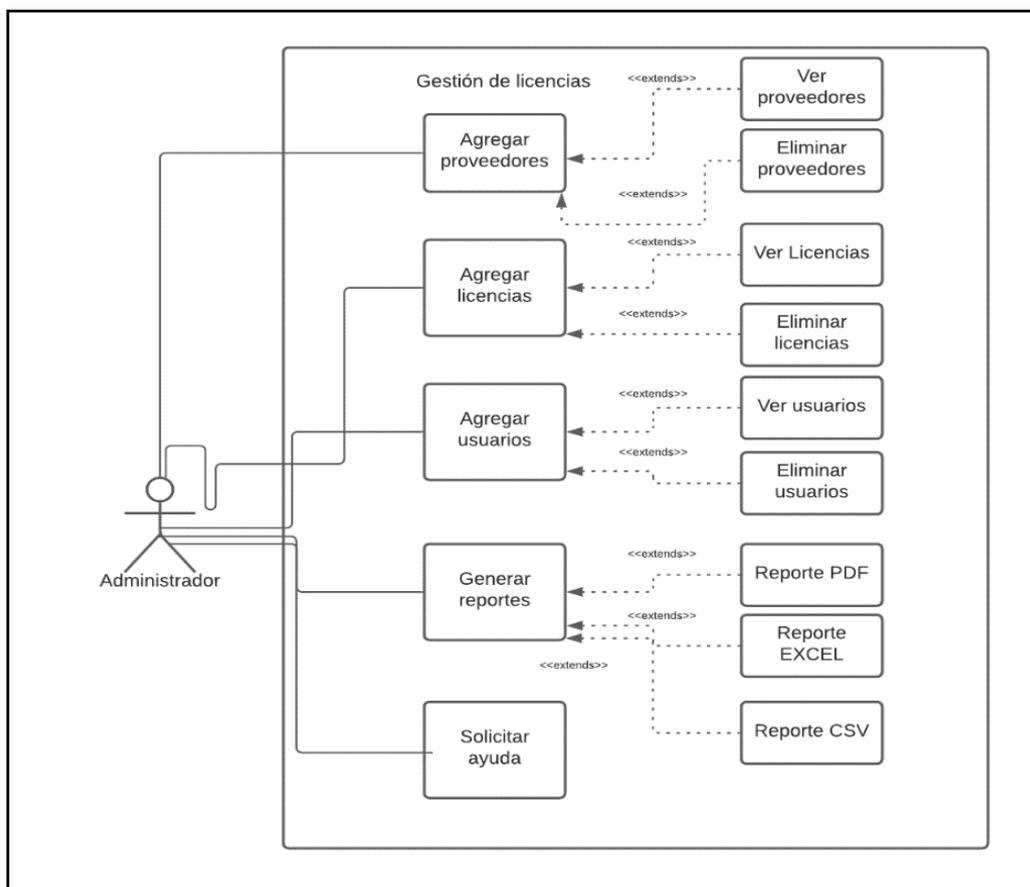
En este capítulo se detallará de manera gráfica el diseño de la solución hacia el problema que se ha encontrado, mostrando diferentes diagramas UML cuyo propósito será mostrar de manera gráfica y detallada el funcionamiento del software.

3.1 DISEÑO DE LA SOLUCIÓN

3.1.1 Diagrama de caso de usos del administrador

Figura 12

Diagrama de caso de uso del administrador



Nota. En la figura 12 se muestra el diagrama de casos de usos del administrador del software.

Elaborado por: Los autores.

3.1.2 Plantilla de descripción del administrador

Tabla 1

Documentación de caso de uso del administrador.

Caso de uso	Documentación
Agregar proveedores	<p>Detalle: Permite la creación de nuevos proveedores.</p> <p>Actor: Administrador.</p> <p>Proceso normal:</p> <ol style="list-style-type: none">1. Actor inicia sesión en el sistema.2. Actor navega al menú de proveedores.3. Actor llena los campos correspondientes.4. Actor crea un nuevo proveedor. <p>Proceso secundario:</p> <ol style="list-style-type: none">5. Actor no llena correctamente los campos.6. Actor observa un mensaje de error.7. Actor corrige los campos.8. Proveedor agregado. <p>Extiende: Este proceso principal extiende dos subprocesos que pueden o no ejecutarse, estos subprocesos son: ver proveedores y eliminar proveedores.</p>
Agregar licencias	<p>Detalle: Permite añadir nuevas licencias de software al sistema.</p> <p>Actor: Administrador.</p> <p>Proceso normal:</p> <ol style="list-style-type: none">1. Actor inicia sesión en el sistema.

	<ol style="list-style-type: none"> 2. Actor navega al menú licencias. 3. Actor llena los campos correspondientes. 4. Al llenar los datos correctamente se crea la nueva licencia. <p>Proceso secundario:</p> <ol style="list-style-type: none"> 5. Actor no completa los datos necesarios. 6. Actor corrige los campos. 7. Actor registra una nueva licencia. <p>Extiende: Este proceso principal extiende dos subprocesos que pueden o no ejecutarse, estos subprocesos son: ver licencias y eliminar licencias.</p>
<p>Agregar usuarios</p>	<p>Detalle: Este proceso permite la creación de nuevos usuarios con su rol respectivo.</p> <p>Actor: Administrador.</p> <p>Proceso normal:</p> <ol style="list-style-type: none"> 1. Actor ingresa al sistema. 2. Actor navega al menú de usuarios. 3. Actor llena los campos correctamente. 4. Actor crea un nuevo usuario. <p>Proceso secundario:</p> <ol style="list-style-type: none"> 5. Actor no llena los campos correctamente. 6. Se visualiza un mensaje de error. 7. Actor corrige los campos. 8. Usuario creado.

	<p>Extiende: Este proceso principal extiende dos subprocesos que pueden o no ejecutarse, estos subprocesos son: ver usuarios y eliminar usuarios.</p>
<p>Generar reportes</p>	<p>Detalle: Este proceso permite generar reportes sobre las licencias registradas de manera automática.</p> <p>Actor: Administrador.</p> <p>Proceso normal:</p> <ol style="list-style-type: none"> 1. Actor inicia sesión en el sistema. 2. Actor navega al menú licencia. 3. Actor escoge el tipo de reporte. 4. Actor descarga el reporte. <p>Proceso secundario:</p> <ol style="list-style-type: none"> 5. Actor no selecciona ningún reporte. 6. Actor no descarga ningún reporte. <p>Extiende: A este proceso se extiende tres subprocesos, estos son la generación de reportes en formatos pdf, excel y csv.</p>
<p>Solicitar ayuda</p>	<p>Detalle: En este proceso se puede solicitar ayuda en caso de que exista algún problema.</p> <p>Actor: Administrador.</p> <p>Proceso normal:</p> <ol style="list-style-type: none"> 1. Actor inicia sesión en el sistema.

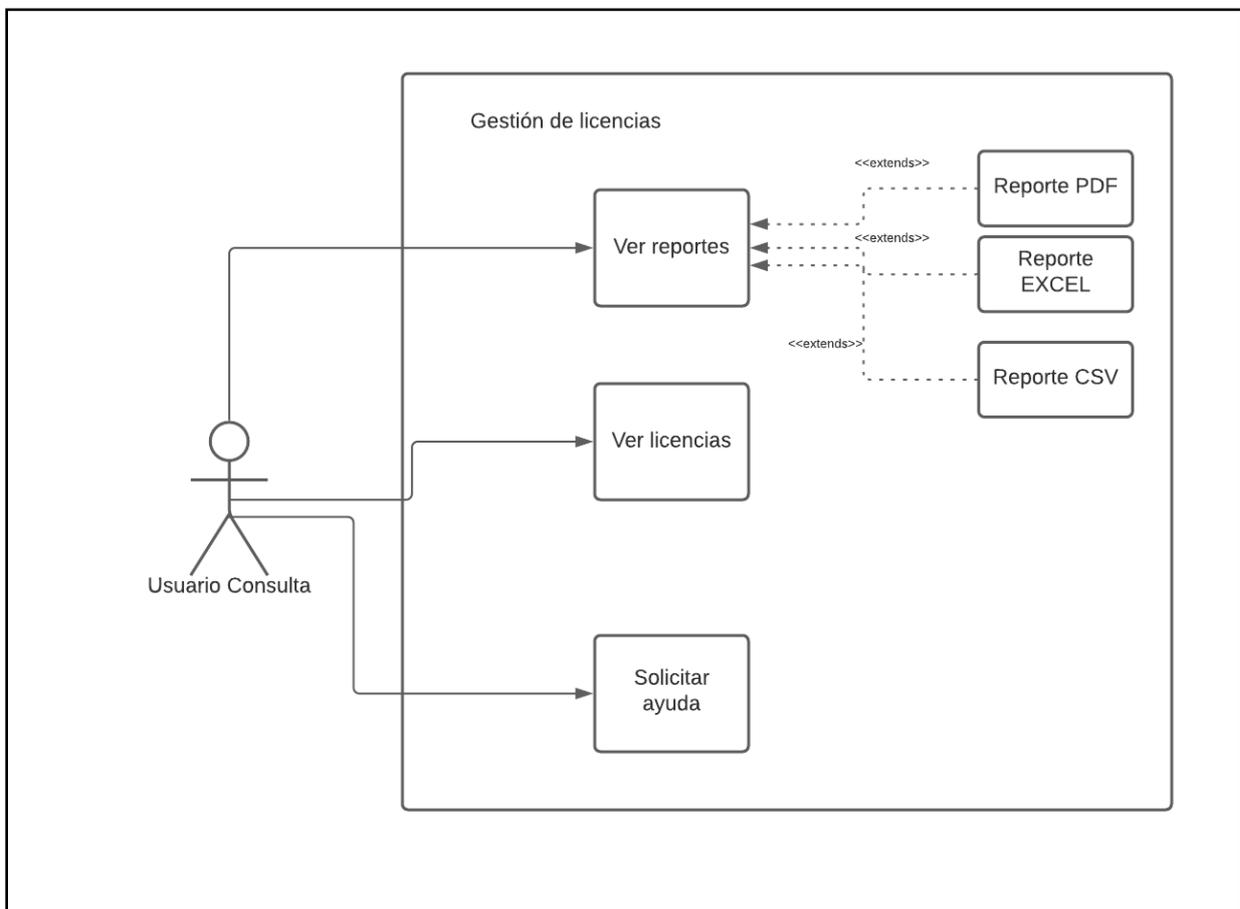
	<ol style="list-style-type: none"> 2. Actor navega al menú de ayuda. 3. Actor solicita ayuda.
--	---

Nota. En la tabla 1 se detalla el proceso de cada caso de uso que cumple en administrador del software. Elaborado por: Los autores.

3.1.3 Diagrama de caso de usos del usuario consulta

Figura 13

Diagrama de caso de uso del usuario consulta



Nota. En la figura 13 se muestra el diagrama de casos de usos del usuario consulta dentro del software. Elaborado por: Los autores.

3.1.4 Plantilla de descripción del usuario consulta

Tabla 2

Documentación de caso de uso del usuario consulta.

Caso de uso	Documentación
Ver reportes	<p>Detalle: En este proceso se puede visualizar los diferentes estilos de reportes que hay para ser descargados.</p> <p>Actor: Usuario consulta.</p> <p>Proceso normal:</p> <ol style="list-style-type: none">1. Actor ingresa al sistema con el rol de usuario de consulta.2. Actor se dirige a la ventana de licencias.3. Actor escoge el tipo de reporte.4. Actor descarga el reporte. <p>Extiende: A este proceso se extiende tres subprocesos, estos son la generación de reportes en formatos pdf, excel y csv.</p>
Ver licencias	<p>Detalle: En este proceso se puede visualizar todas las licencias que se encuentra registradas.</p> <p>Actor: Usuario consulta.</p> <p>Proceso normal:</p> <ol style="list-style-type: none">1. Actor ingresa al sistema con el rol de usuario consulta.2. Actor se dirige a la ventana de licencias.3. Actor visualiza las licencias.
	<p>Detalle: En este proceso se puede solicitar ayuda en caso de que ocurra algún problema.</p>

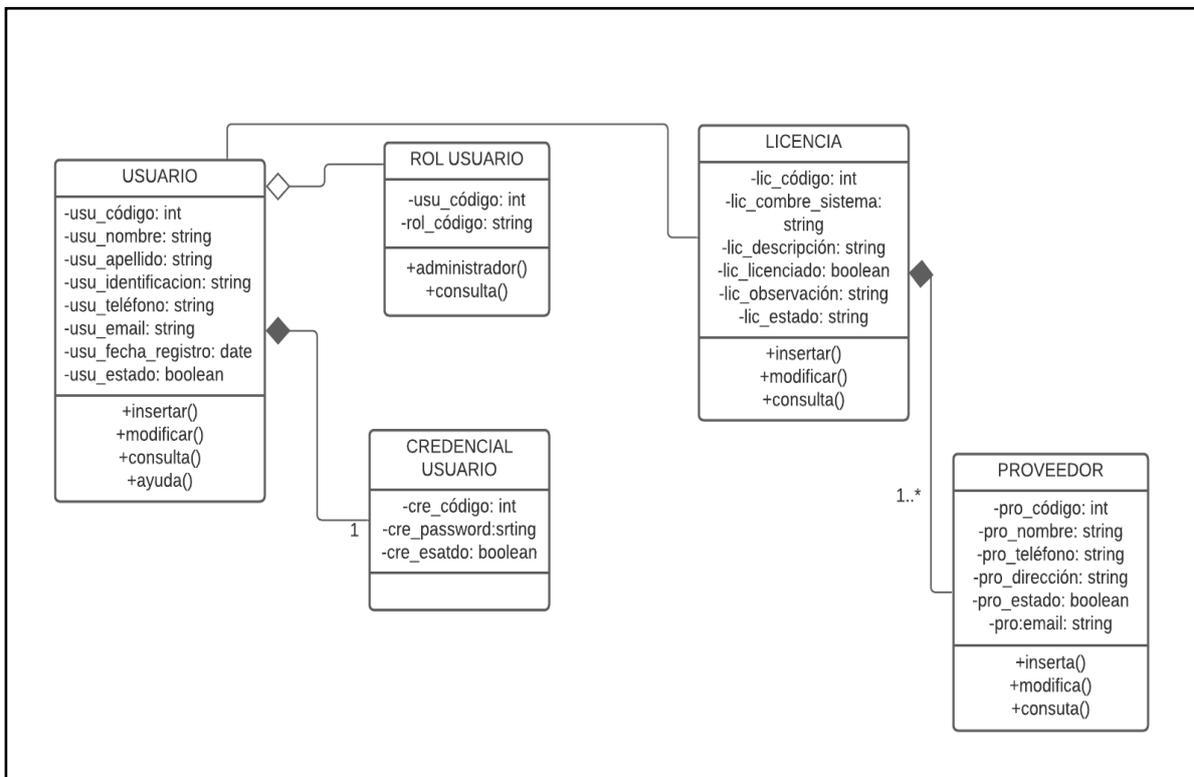
Solicitar ayuda	<p>Actor: Usuario consulta.</p> <p>Proceso normal:</p> <ol style="list-style-type: none"> 1. Usuario ingresa al sistema con el rol de usuario consulta. 2. Usuario navega al menú de ayuda. 3. Usuario solicita ayuda.
-----------------	---

Nota. En la tabla 2 se detalla el proceso de cada caso de uso que cumple el usuario consulta en el software. Elaborado por: Los autores.

3.1.5 Diagrama de clases

Figura 14

Diagrama de clases



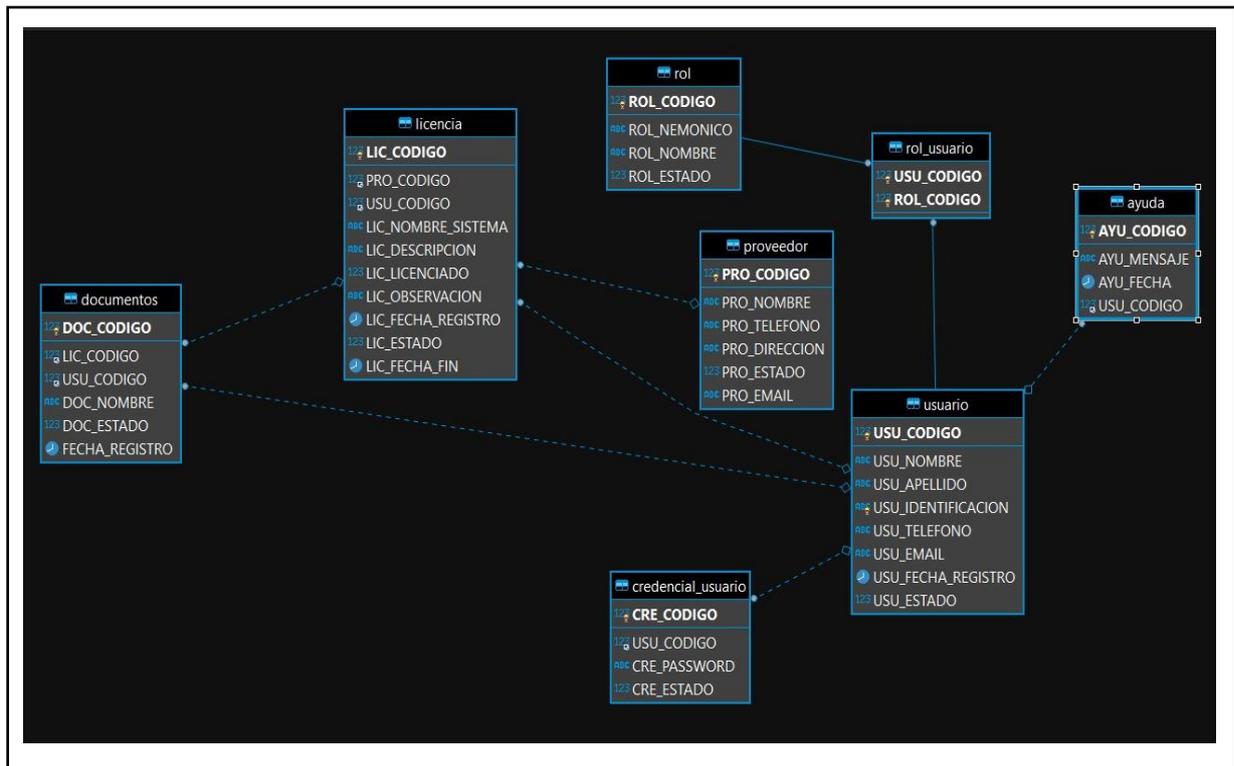
Nota. En la figura 14 se muestra el diagrama de clase del software. Elaborado por: Los autores.

3.2 BOCETO DE LA BASE DE DATOS

Después de haber realizado un análisis cuidadoso sobre los requerimientos solicitados, se creó una base de datos en la cual se guardará toda la información necesaria para poder cumplir con los requisitos que fueron solicitados (ver Figura 9).

Figura 15

Esquema de la base de datos



Nota. En la figura 15 se muestra el esquema de la base de datos que se emplea en el software desarrollado. Elaborado por: Los autores.

3.2.1 Descripción de tablas

- Tabla documentos.
 - Clave primaria: DOC_CODIGO.
 - Fila 1: LIC_CODIGO.

- Fila 2: USU_CODIGO.
- Fila 3: DOC_NOMBRE.
- Fila 4: FECHA_REGISTRO.
- Relación 1: uno a varios con la tabla licencia.
- Relación 2; uno a varios con la tabla usuarios.
- Tabla licencia.
 - Clave primaria: LIC_CODIGO.
 - Fila 1: PRO_CODIGO.
 - Fila 2: USU_CODIGO.
 - Fila 3: LIC_NOMBRE_SISTEMA.
 - Fila 4: LIC_DESCRIPCION.
 - Fila 5: LIC_LICENCIADO.
 - Fila 6: LIC_OBSERVACION.
 - Fila 7: LIC_FECHA_REGISTRO.
 - Fila 8: LIC_ESTADO.
 - Fila 9: LIC_FECHA_FIN.
 - Relación 1: varios a uno con la tabla documentos.
 - Relación 2: uno a varios con la tabla proveedor.
 - Relación 3: uno a varios con la tabla usuario.
- Tabla rol.
 - Clave primaria: ROL_CODIGO.
 - Fila 1: ROL_NEMONICO.
 - Fila 2: ROL_NOMBRE.
 - Fila 3: ROL_ESTADO
 - Relación 1: varios a uno con la tabla rol_usuario.

- Tabla proveedor.
 - Clave primaria: PRO_CODIGO.
 - Fila 1: PRO_NOMBRE,
 - Fila 2: PRO_TELEFONO.
 - Fila 3: PRO_DIRECCION.
 - Fila 4: PRO_ESTADO.
 - Fila 5: PRO_EMAIL.
 - Relación 1: varios a uno con la tabla licencia.

- Tabla credencial_usuario.
 - Clave primaria: CRE_CODIGO.
 - Fila 1: USU_CODIGO.
 - Fila 2: CRE_PASSWORD.
 - Fila 3: CRE_ESTADO.
 - Relación 1: uno a varios con la tabla usuario.

- Tabla rol_usuario.
 - Fila 1: USU_CODIGO.
 - Fila 2: ROL_CODIGO.
 - Relación 1: uno a varios con la tabla rol.
 - Relación 2: uno a varios con la tabla usuario.

- Tabla usuario.
 - Clave primaria: USU_CODIGO.
 - Fila 1: USU_NOMBRE.
 - Fila 2: USU_APELLIDO.
 - Fila 3: USU_IDENTIFICADOR.
 - Fila 4: USU_TELEFONO.

- Fila 5: USU_EMAIL.
- Fila 6: USU_FECHA_REGISTRO
- Fila 7: USU_ESTADO.
- Relación 1: varios a uno con la tabla documentos.
- Relación 2: varios a uno con la tabla licencia.
- Relación 3: varios a uno con la tabla credencial_usuario.
- Relación 4: varios a uno con la tabla ayuda.
- Tabla ayuda.
 - Clave primaria: AYU_CODIGO.
 - Fila 1: AYU_MENSAJE.
 - Fila 2: AYU_FECHA.
 - Fila 3: USU_CODIGO.
 - Relación 1: uno a varios con la tabla usuario.

3.3 DISEÑO DEL SISTEMA

Para el desarrollo de la aplicación se utilizó el entorno de desarrollo integrado (IDE) Visual Studio Code con el lenguaje de programación JavaScript con la integración del framework conocido como Node.js para el backend y React.js para el frontend, posee una modelo vista controlador y la información se guarda en una base de datos MYSQL.

3.3.1 Preparación del entorno de desarrollo

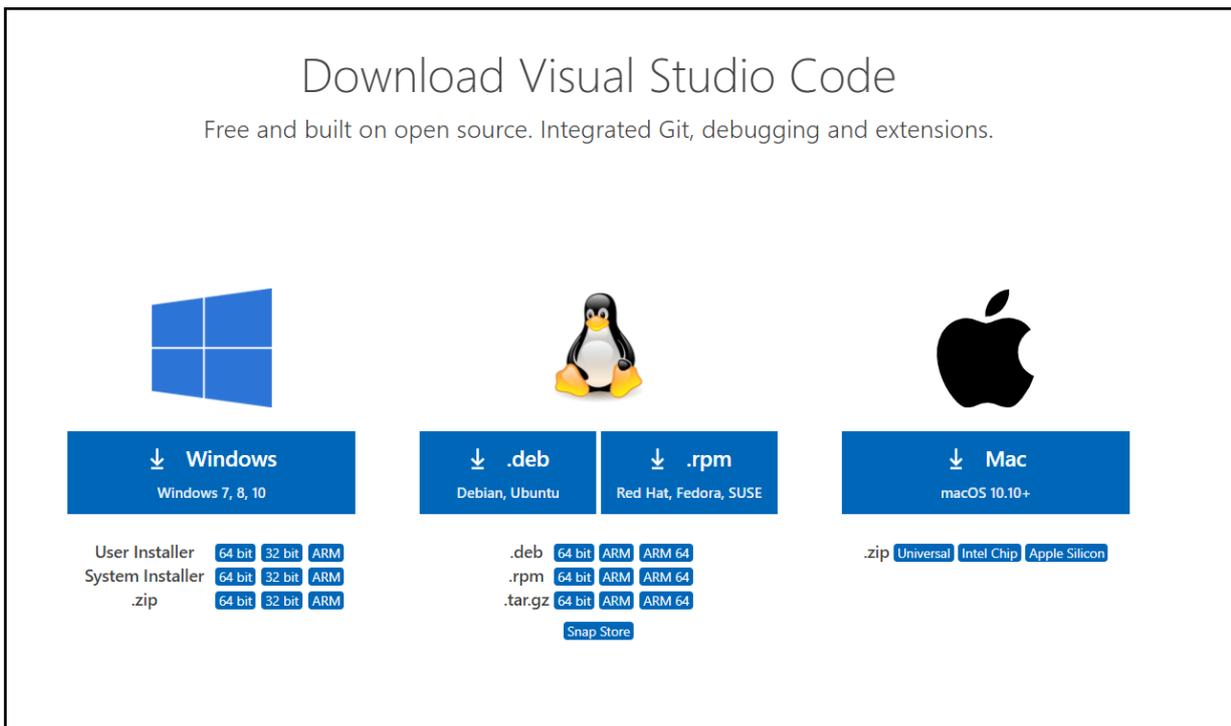
Con el fin de comenzar con el desarrollo del software solicitado se procedió con la instalación de todos los recursos que fueron utilizados.

1. Instalación del entorno de desarrollo integrado (IDE)

- Se puede descargar Visual Studio Code en el siguiente enlace <https://code.visualstudio.com/download>. En dicho enlace se puede seleccionar el sistema operativo sobre el cual se va a instalar el IDE.

Figura 16

Página de inicio para la descarga de Visual Studio Code



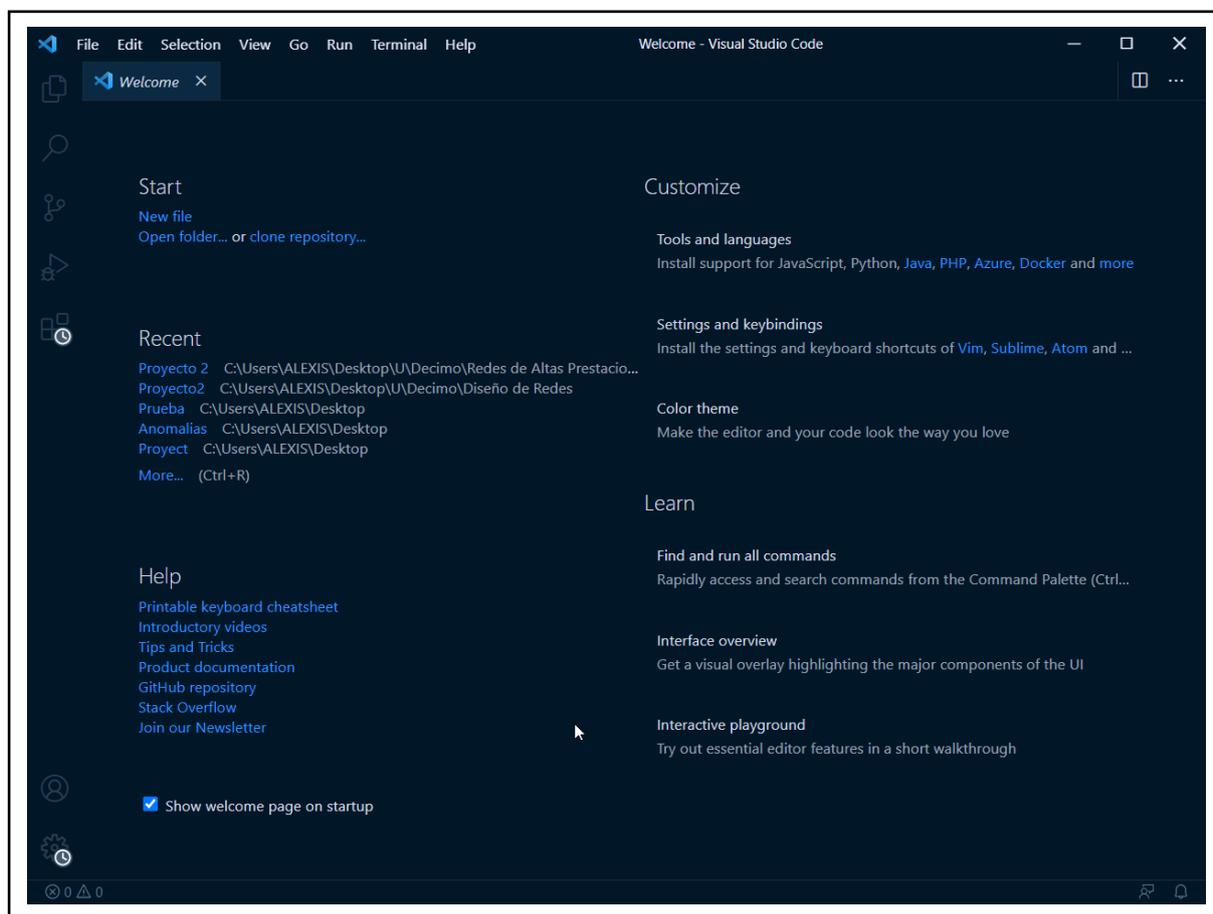
Nota. La figura 16 hace referencia a la página de descarga del IDE en la cual se puede seleccionar el sistema operativo sobre el cual se va a instalar el IDE de trabajo. Fuente: (Microsoft, 2021).

- Una vez se haya seleccionado el sistema operativo se procede a abrir el archivo descargado.
- Se procede con la instalación, el proceso es presionar sobre el botón siguiente hasta que aparezca el botón finalizar.

- Al abrir el IDE se mostrará una pantalla como la de la figura 17.

Figura 17

Pantalla de inicio Visual Studio Code



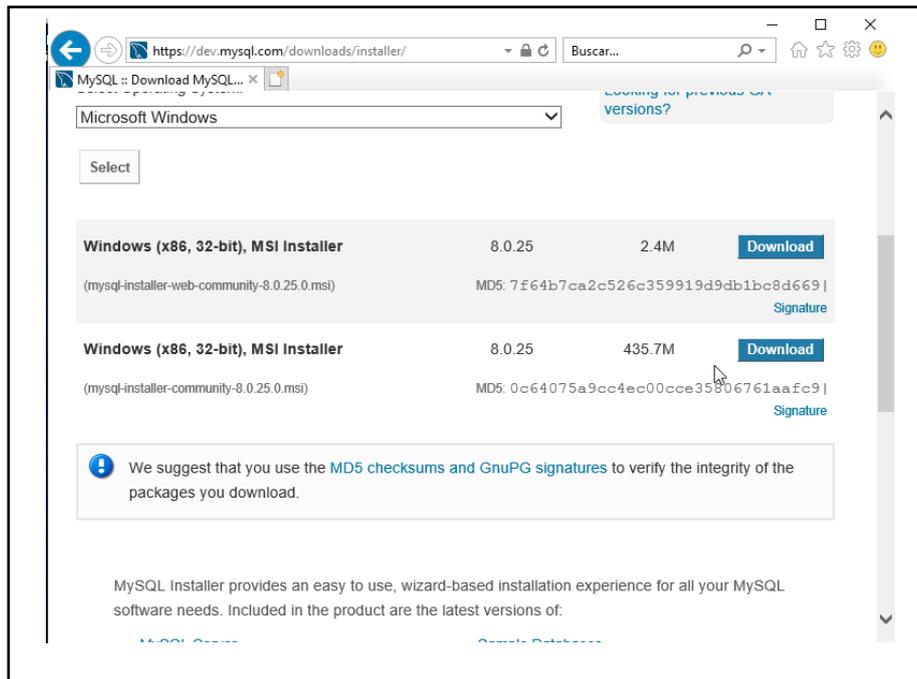
Nota. En la figura 17 se muestra la pantalla de inicio del IDE Visual Studio Code. Elaborado por: Los autores.

2. Instalación de MYSQL

- Dirigirse a la página <https://dev.mysql.com/downloads/> y seleccionar el sistema operativo sobre el cual se instalará.

Figura 18

Página de inicio para descargar MYSQL

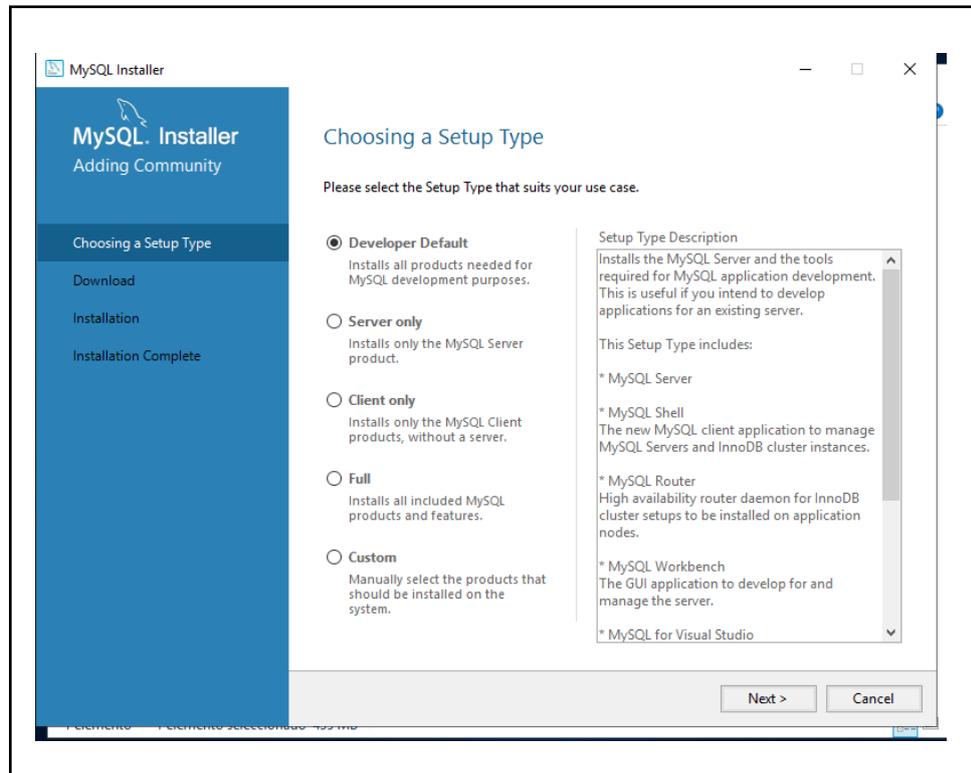


Nota. La figura 18 hace referencia a la página de descarga de MYSQL, en la cual se puede seleccionar el tipo de sistema operativo sobre el cual se va a trabajar. Fuente: (Oracle, 2021).

- Una vez se haya seleccionado el sistema operativo empezará la descarga, al terminar la descarga se abre el archivo de instalación y se mostrará una ventana como la siguiente.

Figura 19

Ventana de inicio del instalador de MYSQL

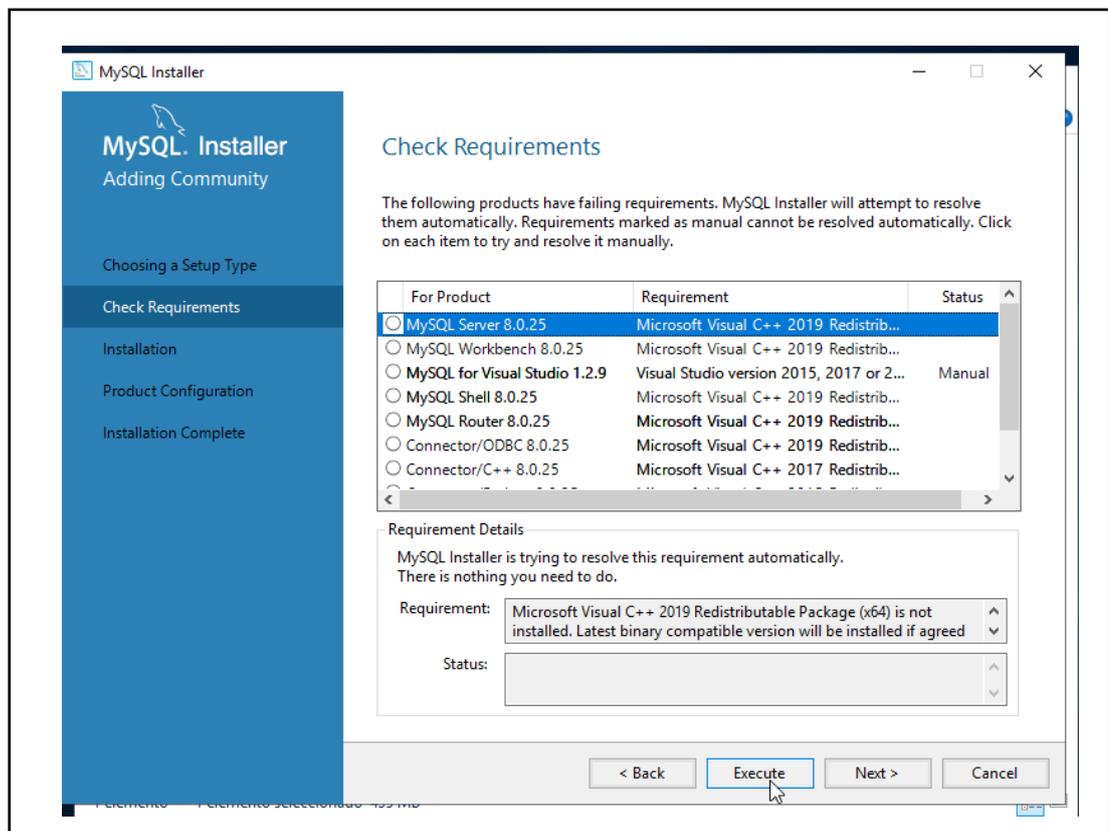


Nota. En la figura 19 se muestra la pantalla de inicio del instalador de MYSQL en la cual se escoge el tipo de instalación a implementar. Elaborado por: Los autores.

- Una vez se haya escogido el tipo de instalación se presiona el botón siguiente, lo cual llevará a una nueva ventana en la cual se chequean los requerimientos para continuar con la instalación.

Figura 20

Chequeo de requerimientos MYSQL

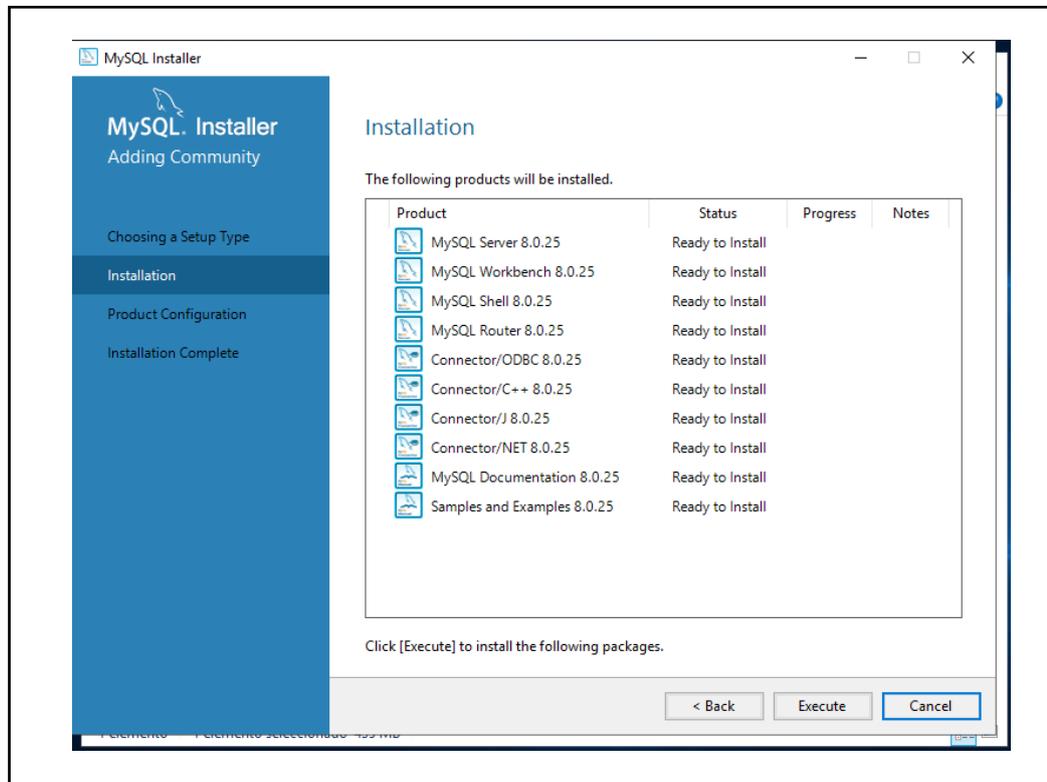


Nota. La figura 20 hace referencia al chequeo de los requerimientos con los que debe contar el sistema para continuar con la instalación. Elaborado por: Los autores.

- Al terminar con la verificación de los requerimientos se procede a instalar una serie de componentes. Para esta instalación se debe dar un clic sobre el botón execute.

Figura 21

Instalación de requerimientos

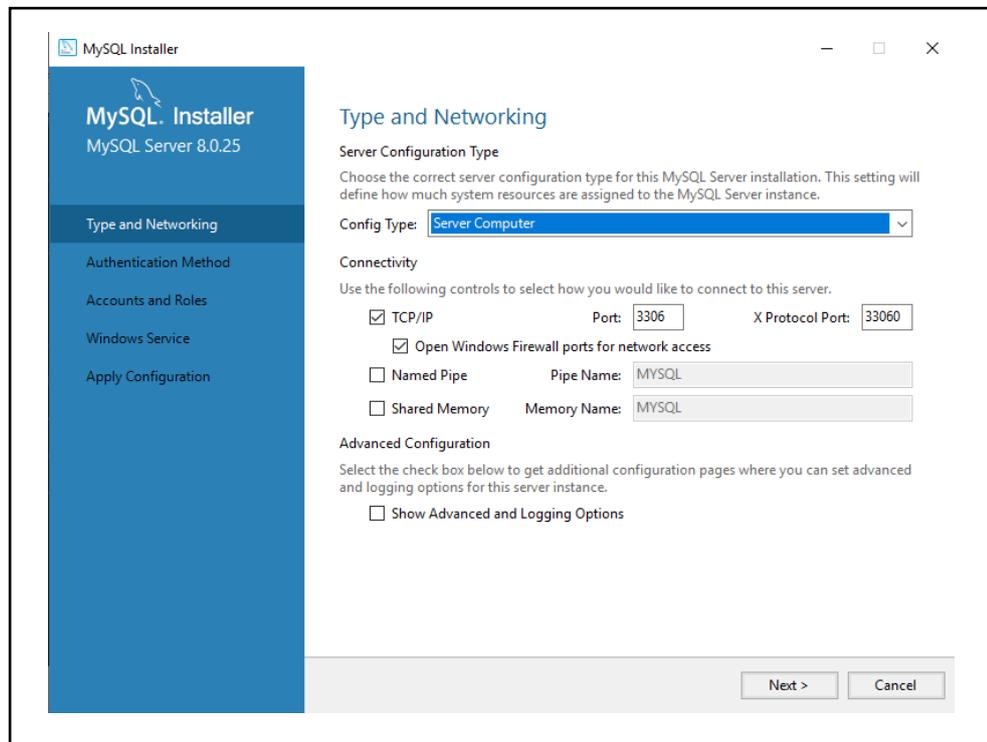


Nota. La figura 21 muestra una serie de componentes con los que debe contar MySQL para continuar su instalación. Elaborado por: Los autores.

- Una vez completada la instalación de los componentes es el momento de establecer el tipo de configuración de MySQL.

Figura 22

Tipo y red de MYSQL

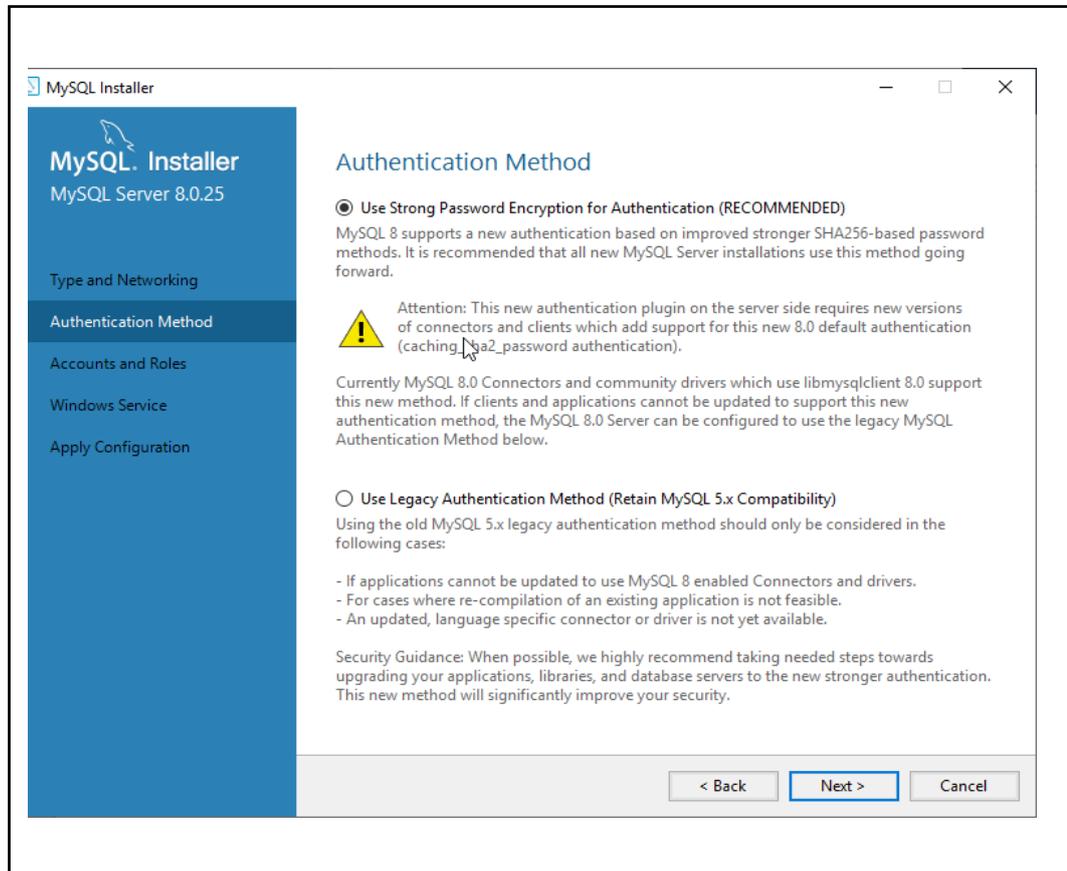


Nota. La figura 22 muestra la ventana de configuración del tipo de red de MYSQL en la cual se deben escoger los parámetros como: tipo de conectividad, puerto de comunicación, entre otros. Elaborado por: Los autores.

- Una vez presionado el botón siguiente en la ventana anterior hay que escoger método de autenticación para acceder a MYSQL, como se muestra en las figuras 23, 24, 25 y 26.

Figura 23

Método de autenticación MYSQL

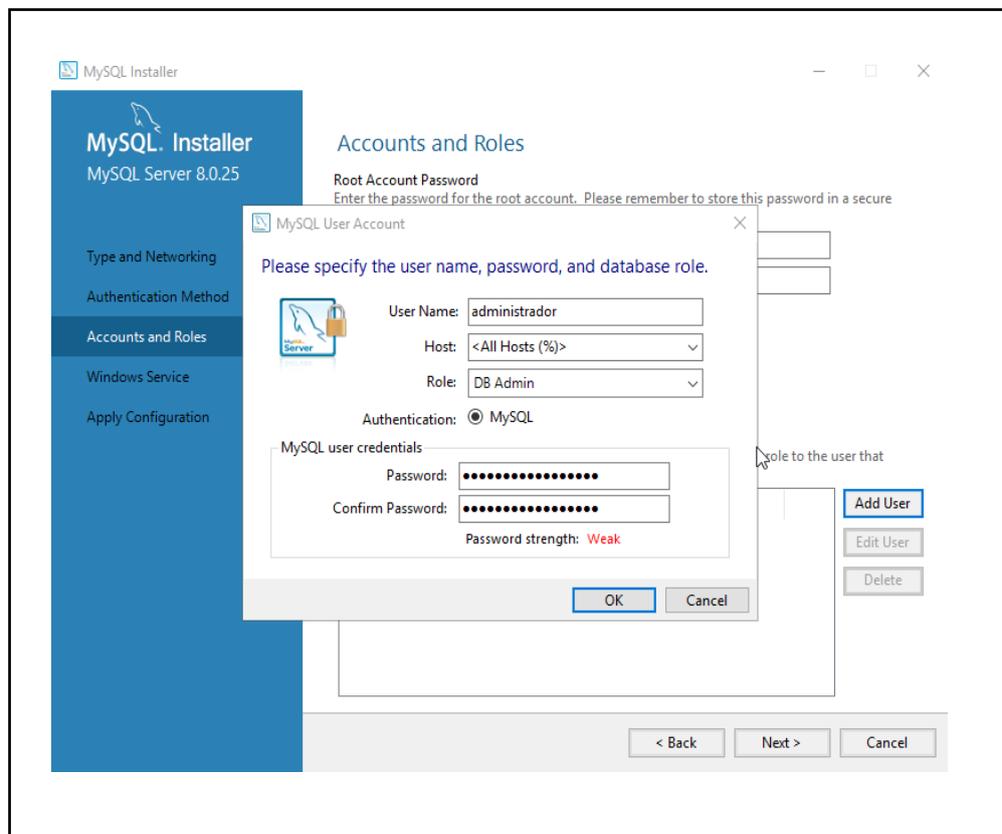


Nota. En la figura 23 se debe escoger el método de autenticación para ingresar a MYSQL.

Elaborado por: Los autores.

Figura 24

Asignación de clave MYSQL

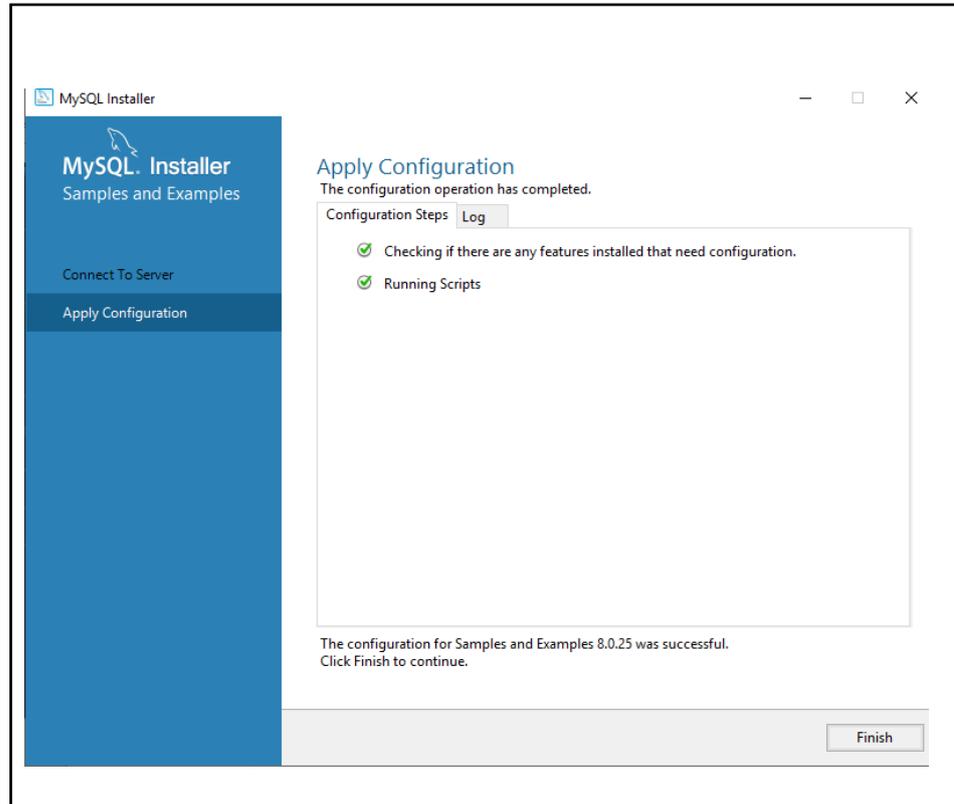


Nota. En la figura 24 se asigna la clave para acceder a MYSQL. Elaborado por: Los autores.

- Una vez se termina con la configuración de las claves de MYSQL se aplican los cambios y se finaliza con la instalación.

Figura 25

Finalización de la instalación de MYSQL



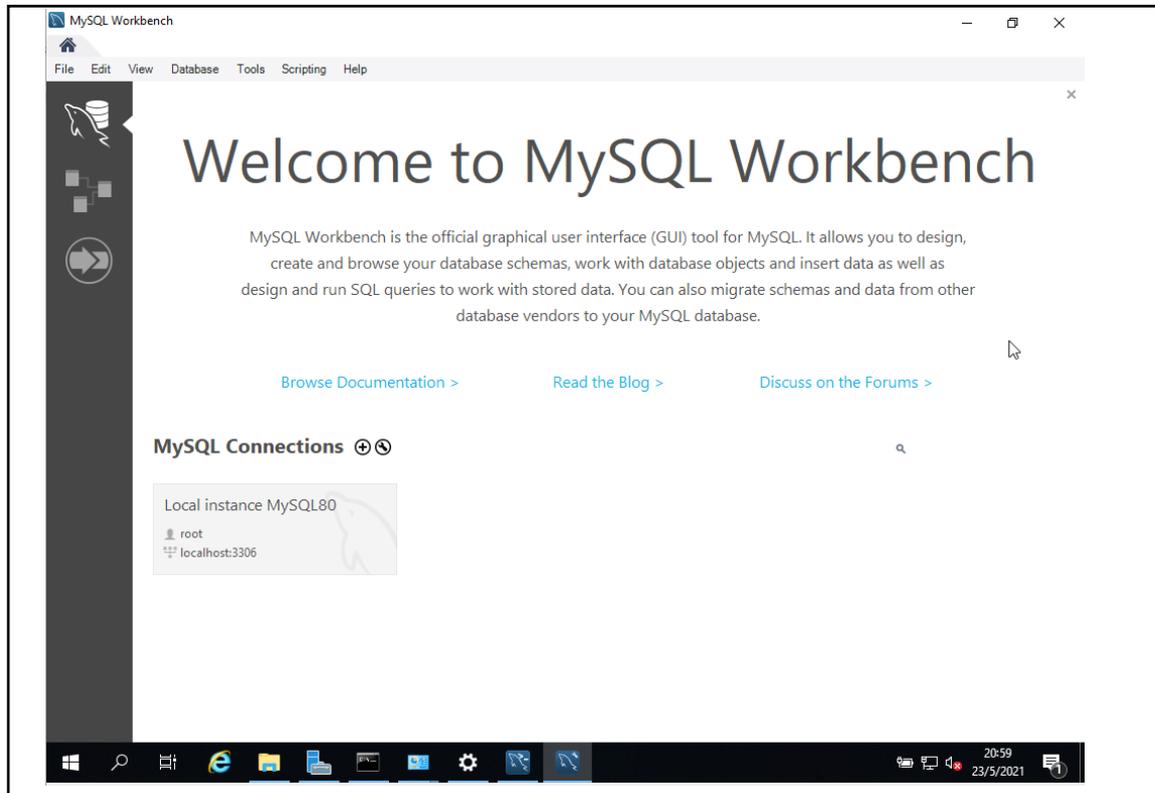
Nota. La figura 25 muestra la ventana para finalizar el proceso de instalación de MYSQL.

Elaborado por: Los autores.

- Una vez completada la instalación se abrirá una ventana como la que se muestra en la figura 26.

Figura 26

Ventana de inicio de MYSQL



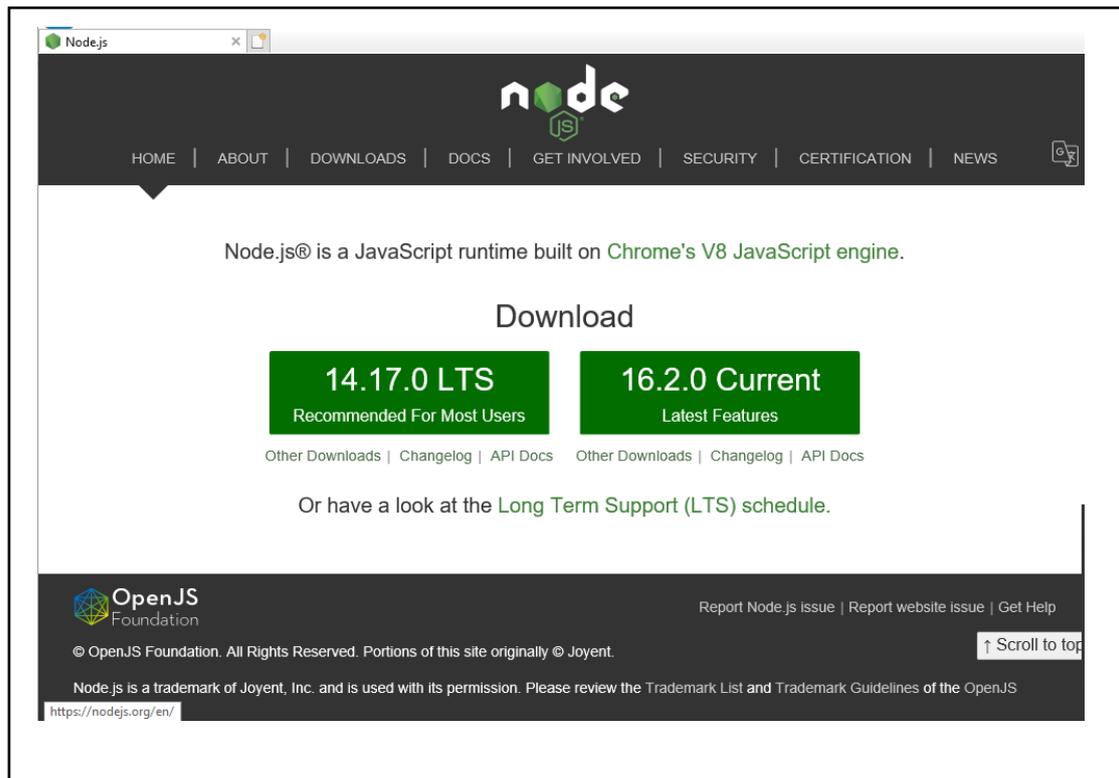
Nota. La figura 26 muestra la ventana de inicio de MYSQL. Elaborado por: Los autores.

3. Instalación de Node.js

- Como primer paso hay que dirigirse a la página oficial de Node.js <https://nodejs.org/es/download/> en la cual se escoge la versión de Node.js y el sistema operativo en cual se va a instalar.

Figura 27

Página de descarga de Node.js

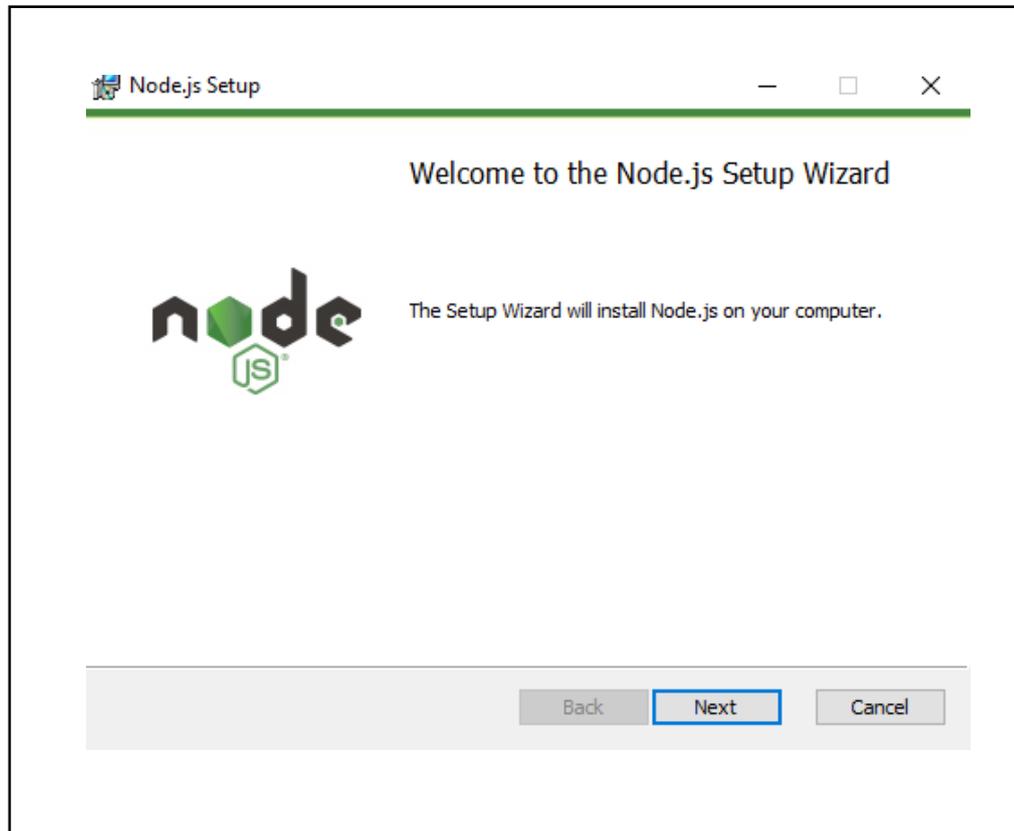


Nota. La figura 27 muestra la página en la que se puede descargar Node.js en sus diferentes versiones y para diferentes sistemas operativos. Fuente: (Foundation, s.f).

- Una vez descargada la versión, en este caso 14.17.0 LTS se procede a abrir el archivo de instalación, el cual mostrará una ventana como la de la figura 28.

Figura 28

Inicio de instalación de Node.js

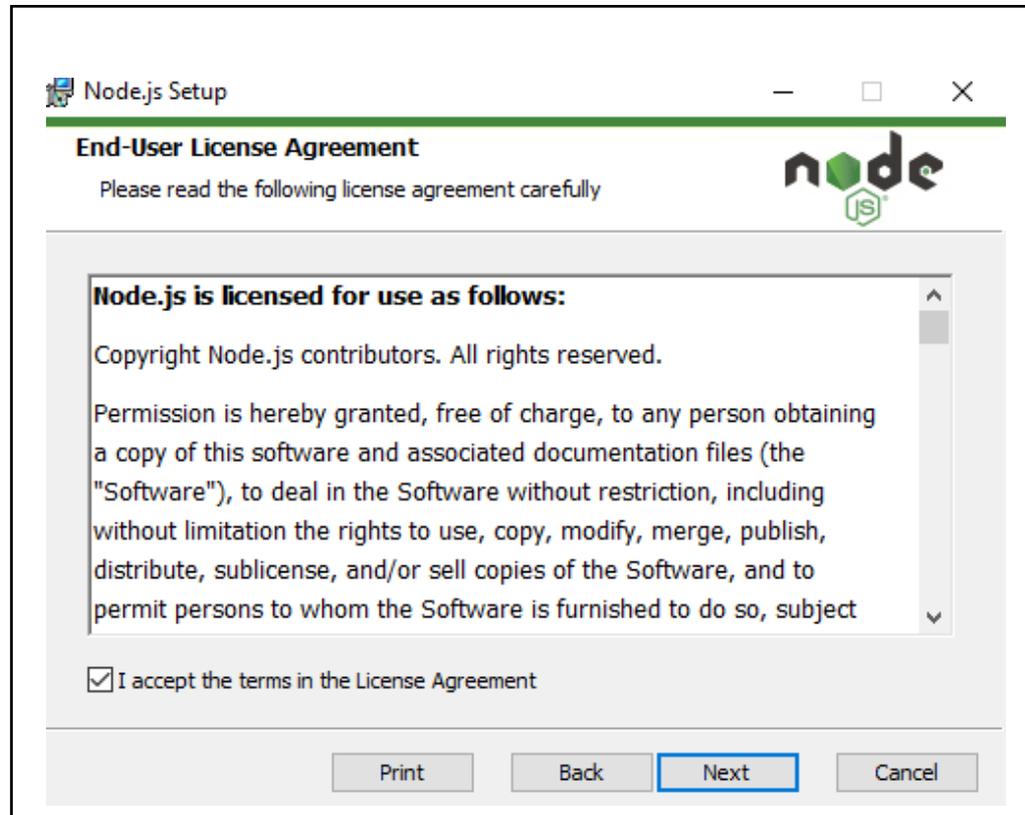


Nota. La figura 28 muestra la ventana de inicio para la instalación de Node.js. Elaborado por: Los autores.

- Para continuar con la instalación se debe aceptar la licencia y presionar el botón next.

Figura 29

Licencia de Node.js

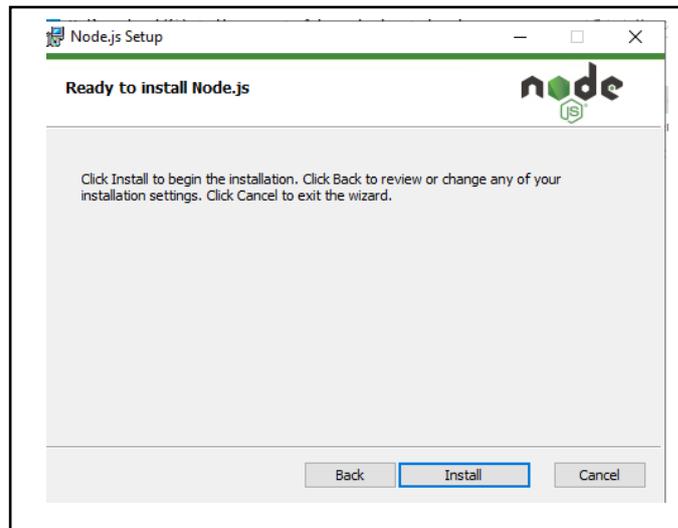


Nota. La figura 29 muestra la licencia de uso de Node.js. Elaborado por: Los autores.

- Una vez aceptada la licencia se procede con la instalación de Node.js como se muestra en las figuras 30 y 31.

Figura 30

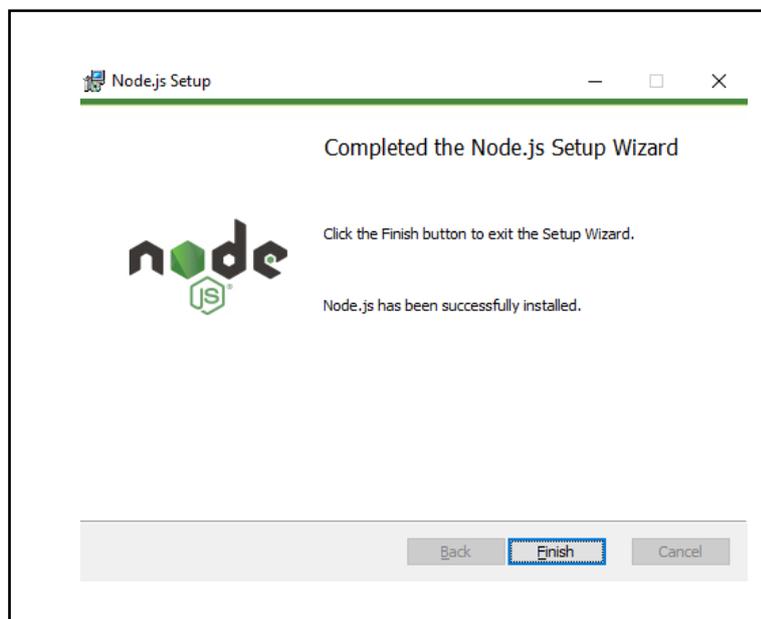
Instalación de Node.js



Nota. La figura 30 muestra la ventana para instalar Node.js. Elaborado por: Los autores.

Figura 31

Final del proceso de instalación de Node.js



Nota. La figura 31 muestra que el proceso de instalación de Node.js ha finalizado. Elaborado por: Los autores.

CAPÍTULO 4

4.1 ANÁLISIS DE RESULTADOS

Con el propósito de analizar los resultados obtenidos se han realizado únicamente pruebas de caja negra en el software, con el fin de verificar el correcto ingreso de datos en las diferentes interfaces del software.

El motivo por el cual no se realizaron pruebas como las de carga o estrés fue debido a que el software funciona únicamente de manera local, por lo que existirán máximo 5 personas conectadas de manera simultánea, es por esto que específicamente para este software estas pruebas no otorgarían información precisa.

4.1.1 *Pruebas de caja negra*

Teniendo en cuenta que en el software posee únicamente dos roles como lo es el administrador y un usuario común, se debe ingresar como administrador para realizar las distintas pruebas.

- **Ventana de registro de licencias.**

Para comprobar que los datos sean ingresados de manera correcta en esta ventana el software cuenta con sus respectivas validaciones. Como la validación que no permite que se dejen campos vacíos en los siguientes parámetros: nombre del sistema, proveedor y licenciado.

Del mismo modo cuenta con la validación de archivos la cual solo permite que se suban archivos en formato pdf y jpg.

Figura 32

Ventana de registro de licencias con errores

The screenshot shows a web form titled "Registro de licencias" with the following fields and errors:

- Nombre del sistema:** A text input field with a red border and the error message "Ingrese un nombre válido" below it.
- Proveedor:** A dropdown menu with a red border and the error message "Seleccione un prestador" below it.
- Licenciado:** A dropdown menu with a red border and the error message "Seleccione" below it.
- Descripción del sistema:** A text input field.
- Observaciones:** A text input field.
- Fecha de vencimiento:** A date input field showing "25/05/2021" with a calendar icon.
- Activo:** A checked checkbox.
- Archivo:** A file upload area with a dashed border. It contains a message: "Seleccione el archivo pdf o imagen jpg. Arrastre el archivo o buscalo en tu máquina haciendo click [AQUÍ](#)". Below this, a file named "Acuerdos.docx" (1.3 MB) is shown with a close icon. At the bottom right of the upload area is a button labeled "ELIMINAR TODOS".

At the bottom of the form, there is a blue "GUARDAR" button and a red error banner that reads: "Solo se permite archivos en pdf o imagenes jpg".

Nota. La figura 32 muestra los errores que se producen en los distintos campos de texto que conforman la ventana de registro de licencias. Elaborado por: Los autores.

Para que se pueda ingresar una licencia de manera correcta se deben llenar los campos que son obligatorios mostrados en la figura 32, así como se debe seleccionar un correcto formato de archivo a subir.

Figura 33

Ingreso de licencias correcto

Registro de licencias

Registro

Nombre del sistema: Office Descripción del sistema: []

Proveedor: IBM Observaciones: []

Licenciado: Sí Fecha de vencimiento: 25/05/2021

Activo:

Archivo:

Seleccione el archivo pdf o imagen jpg
Arrastre el archivo o buscalo en tu máquina haciendo click [AQUÍ](#)

AcuerdosAlvarezAndres.pdf
180.53 KB

GUARDAR

ELIMINAR TODOS

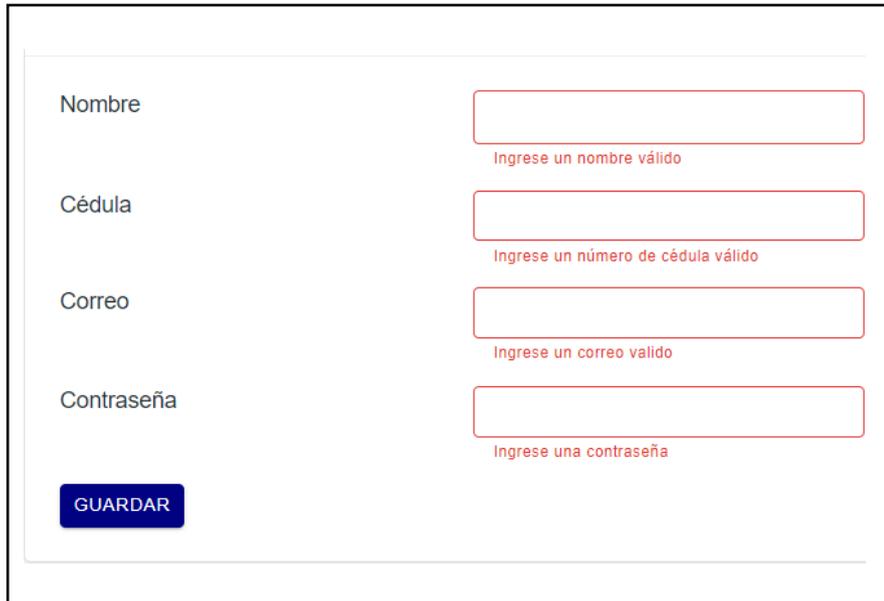
Nota. La figura 33 muestra un ingreso exitoso de una licencia. Elaborado por: Los autores.

- **Ventana de registro de usuario.**

La ventana de registro de usuarios cuenta con más validaciones ya que en esta todos los campos son obligatorios y algunos de estos poseen validaciones especiales, como la validación que el número de cédula sea correcto y la verificación del campo correo.

Figura 34

Primera parte de los errores de la ventana agregar usuarios



The screenshot shows a registration form with four input fields, each with a red border and a red error message below it. The fields are: 'Nombre' (Name) with the error 'Ingrese un nombre válido'; 'Cédula' (ID) with the error 'Ingrese un número de cédula válido'; 'Correo' (Email) with the error 'Ingrese un correo valido'; and 'Contraseña' (Password) with the error 'Ingrese una contraseña'. A blue 'GUARDAR' (Save) button is located at the bottom left of the form.

Nota. La figura 34 muestra los errores en los campos de nombre, cédula el cual indica que se debe ingresar una cédula valida, correo el cual verifica el formato del correo y la contraseña que es un campo obligatorio. Elaborado por: Los autores.

Figura 35

Segunda parte de los errores de la ventana agregar usuarios



The screenshot shows a registration form with three input fields, each with a red border and a red error message below it. The fields are: 'Apellido' (Last Name) with the error 'Revise este campo solo letras'; 'Teléfono' (Phone) with the error 'Solo se permiten número'; and 'Rol' (Role) with a dropdown menu showing 'Seleccione...' and the error 'Seleccione un rol'.

Nota. La figura 35 hace referencia a los errores producidos en el apellido, en el teléfono ya que se permite que se ingresen solo números y el rol no debe ser un campo en blanco. Elaborado por: Los autores.

Para que el proceso de ingresar un usuario sea ejecutado de manera correcta se deben llenar los campos teniendo en cuenta las consideraciones mencionadas.

Figura 36

Registro de usuario exitoso

Registro de usuario

Registro

Nombre	José Luis	Apellido	Aguayo
Cédula	1721529707	Teléfono	098456325
Correo	jose@gmail.com	Rol	ADMINISTRADOR ▼
Contraseña	123		

GUARDAR

Nota. La figura 36 muestra el ingreso exitoso de un nuevo usuario al sistema. Elaborado por:
Los autores.

- **Ventana de registro de proveedores.**

La ventana de registro de proveedores es la más flexible en cuanto a validaciones en este sistema ya que únicamente se necesita que el nombre del proveedor quede registrado.

Figura 37

Registro de proveedor con error



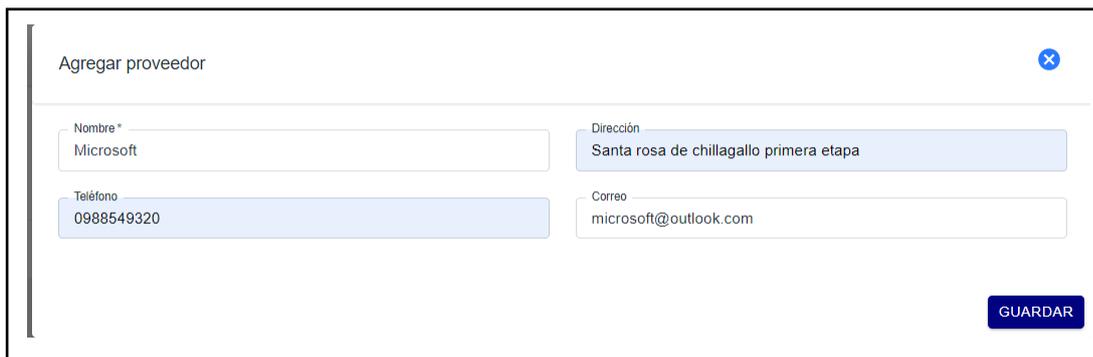
The screenshot shows a web form titled "Agregar proveedor" with a close button (X) in the top right corner. The form contains four input fields: "Nombre *", "Dirección", "Teléfono", and "Correo". The "Nombre *" field is highlighted with a red border, and a red error message "Campo obligatorio" is displayed below it. The "Dirección" field contains the text "Santa rosa de chillagallo primera etapa". The "Teléfono" field contains "0988549320". The "Correo" field contains "microsoft@outlook.com". A blue "GUARDAR" button is located at the bottom right of the form.

Nota. La figura 37 hace referencia al error en el campo del nombre del proveedor Elaborado por: Los autores.

Como se explicó anteriormente en esta ventana el único campo obligatorio es el nombre del proveedor, por esta razón al llenar este campo se producirá un ingreso exitoso de proveedor.

Figura 38

Registro exitoso de proveedor



The screenshot shows the same "Agregar proveedor" form as in Figure 37, but with all fields filled and highlighted in light blue. The "Nombre *" field contains "Microsoft". The "Dirección" field contains "Santa rosa de chillagallo primera etapa". The "Teléfono" field contains "0988549320". The "Correo" field contains "microsoft@outlook.com". The "GUARDAR" button is still present at the bottom right.

Nota. La figura 38 muestra un registro exitoso de un nuevo proveedor. Elaborado por: Los autores.

- **Ventana de ayuda**

Esta ventana es la más simple ya que únicamente está conformada de un editor de texto el cual debe ser llenado obligatoriamente.

Figura 39

Registro de ayuda fallido



Nota. La figura 39 muestra el registro fallido de la solicitud de ayuda ya que el campo está vacío. Elaborado por: Los autores.

Figura 40

Registro de ayuda exitoso



Nota. La figura 40 muestra el registro exitoso de la solicitud de ayuda. Elaborado por: Los autores.

4.1.2 Pruebas de seguridad

- Como primera prueba de seguridad se realiza la petición en el login del sistema ejecutando un comando de inyección SQL cuyo propósito es entrar al sistema con el primer usuario que se encuentre en la base.

Figura 41

Inyección SQL al login del sistema

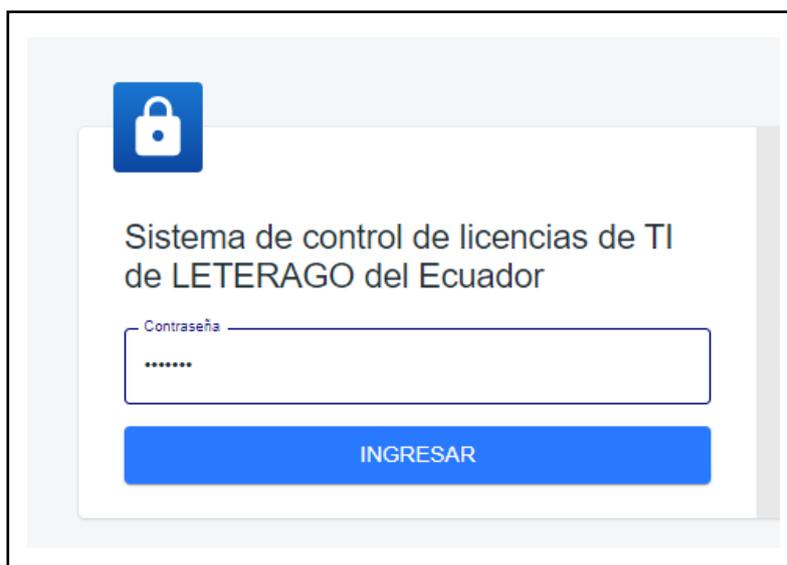


Nota. La figura 41 muestra el intento de acceder al sistema mediante inyección SQL. Elaborado por: Los autores.

Al momento de ingresar la clave se ingresa la misma inyección SQL ya que dicho comando indica que la contraseña es verdadera intentando vulnerar la seguridad del campo de la contraseña en la base de datos.

Figura 42

Inyección SQL al campo contraseña



Nota. La figura 42 muestra el intento de acceder al sistema mediante inyección SQL en el campo de la contraseña. Elaborado por: Los autores.

En el instante de intentar ingresar al sistema mediante inyección SQL fallará por lo que la página se redireccionará hacia la ventana inicial.

Figura 43

Página de login del sistema

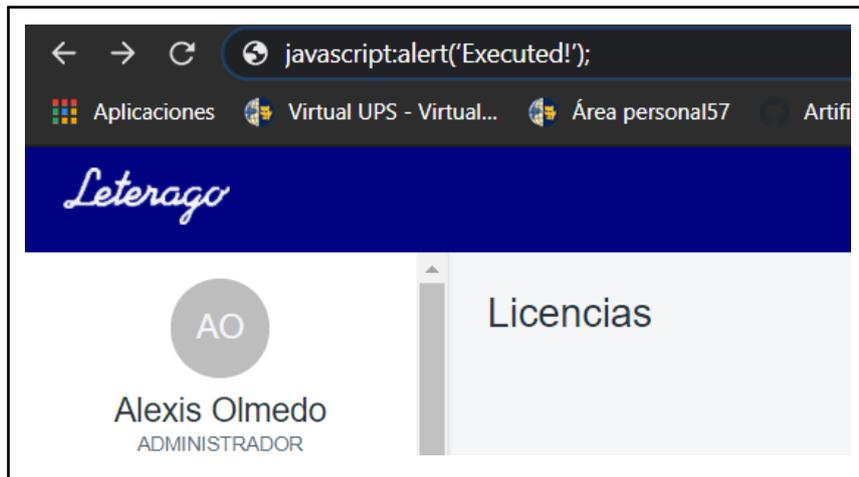


Nota. La figura 43 muestra la página principal del login del sistema después de un intento fallido de ingresar al sistema mediante inyección SQL. Elaborado por: Los autores.

- La segunda prueba de seguridad está basada a detectar si el software es vulnerable a comandos de inyección JS.

Figura 44

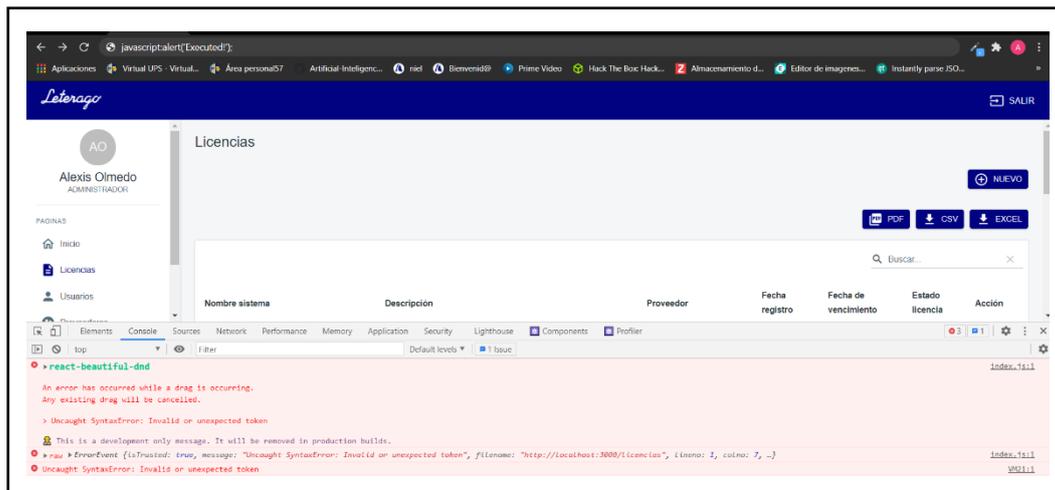
Inyección JS al sistema



Nota. La figura 44 muestra la ejecución de una inyección JS en el sistema. Elaborado por: Los autores.

Figura 45

Inyección JS fallida

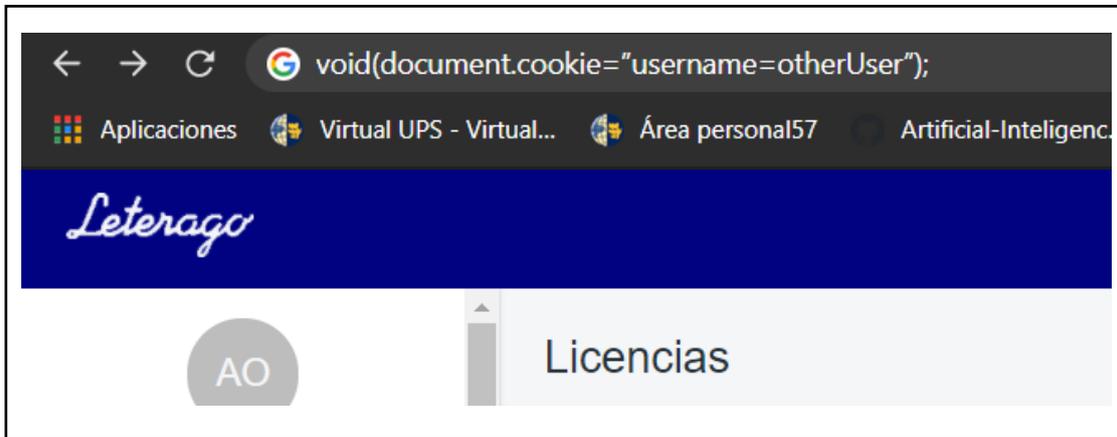


Nota. La figura 45 indica que el software no puede ser atacado mediante inyección JS. Elaborado por: Los autores.

- En la tercera prueba de seguridad se trata de alterar el nombre del usuario mediante inyección JS.

Figura 46

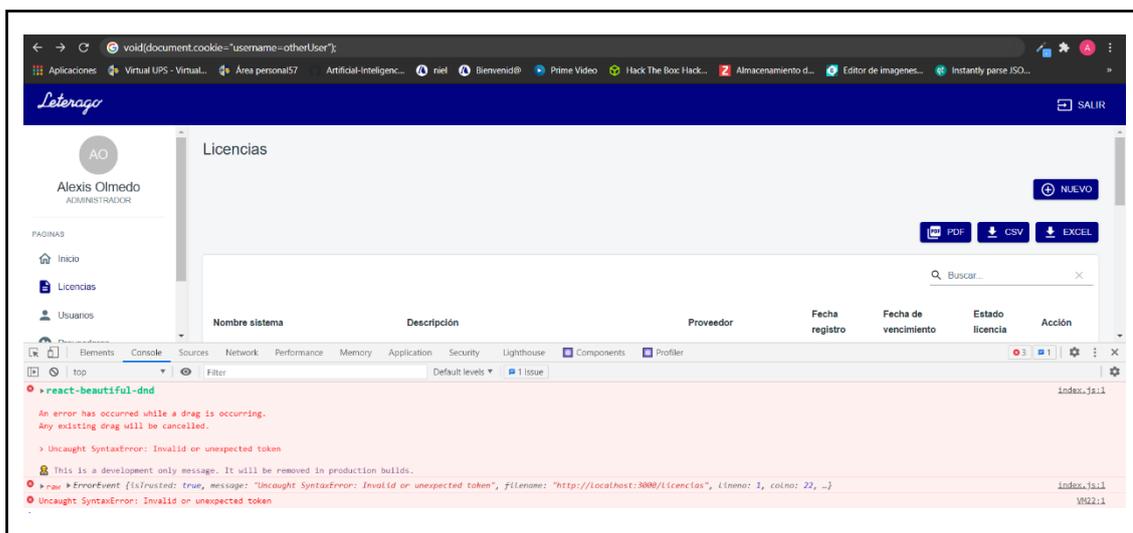
Inyección JS para intentar cambiar el nombre del usuario



Nota. La figura 46 muestra la inyección JS que se utiliza para intentar cambiar el nombre del usuario. Elaborado por: Los autores.

Figura 47

Inyección JS para editar usuario fallida



Nota. La figura 47 la inyección JS ha fallado ya que el sistema rechaza la petición. Elaborado por: Los autores.

CONCLUSIONES

El diagrama de base de datos diseñado cumple con todas las especificaciones para almacenar toda la información necesaria permitiendo que el software funcione de manera correcta.

Mediante la ayuda de la librería Nodemailer se pudo automatizar el proceso de creación y envío de correos electrónicos los cuales notifican si una licencia de software esta próxima a caducarse satisfaciendo totalmente este requerimiento.

Debido a que el software diseñado funciona de manera local en la red del departiendo de TI de la empresa “LETERAGO DEL ECUADOR S.A.” su implementación fue exitosa mediante la agregación de tareas automatizadas que se ejecutan al momento de iniciar el sistema operativo, proporcionando que el software diseñado se ejecute de manera automática cumpliendo las necesidades en un 100%.

Los reportes en formato pdf y excel cuentan con todos los campos que fueron solicitados en los requerimientos iniciales, permitiendo una mejor organización de los datos almacenados cumpliendo totalmente con el objetivo planteado en el proyecto.

RECOMENDACIONES

Habilitar un dominio público en el software creado, para que de esta manera se pueda facilitar el ingreso de las licencias de software desde cualquier parte teniendo en cuenta las medidas de ciberseguridad necesarias.

Se recomienda el desarrollo de una aplicación móvil, la cual pueda sincronizarse con el software creado con el fin de facilitar el registro de las licencias de software sin necesidad de utilizar una computadora conectada a la red local.

Se sugiere dar nuevas funcionalidades al software implementado de acuerdo con los nuevos requerimientos que vayan surgiendo en la empresa con el transcurso del tiempo y el marco legal que se encuentre vigente.

LISTA DE REFERENCIAS

Artículos de revista

Brocca, J. C. (2005). LAS LICENCIAS DE SOFTWARE DESDE LA PERSPECTIVA DEL USUARIO FINAL. *Pilquen*, 1-10.

Documentos de sitio web

Ávila, C. (12 de 11 de 2019). *Konred Lorenz*. Obtenido de <https://repositorio.konradlorenz.edu.co/handle/001/1528>

Stepantsov, A. (05 de 2018). *XAMIK*. Obtenido de https://www.theseus.fi/bitstream/handle/10024/148347/Stepantsov_Aleksandr.pdf?sequence=1

Informes

Fuentes, L., & Vallecillo, A. (s.f). *Una Introducción a los Perfiles UML*. Malaga: Depto. de Lenguajes y Ciencias de la Computación, Universidad de Málaga.

Gutierrez, D. (2011). *Casos de Uso Diagrama de Casos de Uso*. Universidad de los Andes.

Gutierrez, D. (2011). *UML Diagramas de Clases UML ilustrado*. Universidad de los Andes.

Lázaro, J. C. (2009). *UML Diagramas de Clases y Casos de Uso*. Madrid: Universidad Complutense Madrid.

Medellín, O. (2016). *Servidores en cómputo*. México.

Torrado, M. (2019). *Estudio de metodologías ágiles. Aplicación SCRUM*. Sevilla.

Libros

Cobo, A. (s.f.). *Diseño y programación de bases de datos*. Madrid: Vision Libros.

Dayley, B. (2018). *Node.js, MongoDB, and AngularJS Web Development*. Estados Unidos: Pearson Education.

Gauchat, J. (2012). *El gran libro de HTML5, CSS3 Javascript*. Barcelona.

Hoque, S. (2020). *Full-Stack React Projects*. Birmigham: Packt Publishing Ltd.

Jimenez, J. (2016). *Descubre React*. Lean Publishing.

Parga, C. J. (2015). *UML. Aplicaciones en Java y C++*. Grupo Editorial RA-MA.

Pressman, R. (2010). *Ingeniería del software un enfoque práctico*. México, D. F: María Teresa Zapata Terrazas.

Pressman, R. (2010). *Ingeniería del software, un enfoque práctico*. México: McGRAW-HILL.

Rodriguez, D. , & Llanos, J.(2010). *Fundamentos de informática y programación en C*. Paraninfo.

Sitios Web

Foundation., O. (s.f.). *node js*. Obtenido de s.f: <https://nodejs.org/es/download/>

Foundation, O. (s.f). *NodeJS*. Obtenido de <https://nodejs.org/es/download/>

León, A. d. (22 de 05 de 2019). *Hosting Diario*. Obtenido de <https://hostingdiario.com/windows-server/>

Merencia, L. (02 de 2021). *npm*. Obtenido de <https://www.npmjs.com/package/node-cron>

Microsoft. (2021). *Visual Studio Code*. Obtenido de <https://code.visualstudio.com/download>

Oracle. (2021). *MySQL Community Downloads*. Obtenido de <https://dev.mysql.com/downloads/>

tiThink. (29 de 08 de 2018). Obtenido de <https://www.tithink.com/es/2018/08/29/framework-o-librerias-ventajas-y-desventajas/>

Uesteibar. (03 de 03 de 2015). *Software Developer*. Obtenido de <https://medium.com/@uesteibar/env%C3%ADa-emails-desde-node-js-con-nodemailer-178cacf5cf6b#:~:text=Nodemailer%20es%20un%20m%C3%B3dulo%20que,para%20hacerlo%20con%20diversos%20servicios.>

Vaati, E. (11 de 09 de 2018). *envatotuts+*. Obtenido de <https://code.tutsplus.com/es/tutorials/file-upload-with-multer-in-node--cms-32088>