



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN SISTEMAS E INFORMÁTICA**

**TEMA: “SIVENLAC” SISTEMA WEB DE INVENTARIO PARA CONTROL
DE EXISTENCIAS EN LA BODEGA DE MATERIALES, DE LA EMPRESA
“PASTOLAC”**

AUTOR: VERGARA BUSTILLOS, PAUL ALBERTO

DIRECTOR: ING. NAVAS MOYA, MILTON PATRICIO

LATACUNGA

2019



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

CERTIFICADO DEL DIRECTOR

Certifico que el trabajo de titulación, **“SIVENLAC” SISTEMA WEB DE INVENTARIO PARA CONTROL DE EXISTENCIAS EN LA BODEGA DE MATERIALES, DE LA EMPRESA “PASTOLAC”**, fue realizado por el señor **VERGARA BUSTILLOS, PAÚL ALBERTO**, el mismo que ha sido revisado en su totalidad, analizado por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de las Fuerzas Armadas - ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Latacunga, 28 Junio del 2019


.....
Ing. Navas Moya, Milton Patricio
C.C.: 0502029275



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

AUTORÍA DE RESPONSABILIDAD

Yo, **VERGARA BUSTILLOS, PAÚL ALBERTO**, declaro que el contenido, ideas y criterios del trabajo de titulación **“SIVENLAC” SISTEMA WEB DE INVENTARIO PARA CONTROL DE EXISTENCIAS EN LA BODEGA DE MATERIALES, DE LA EMPRESA “PASTOLAC”**, es de mi autoría y responsabilidad, cumpliendo con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de las fuerzas armadas – ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Consecuentemente el contenido de investigación mencionada es veraz.

Latacunga, 28 Junio del 2019

Vergara Bustillos, Paúl Alberto

C.C.: 0502832827



DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

AUTORIZACIÓN

Yo, **VERGARA BUSTILLOS, PAÚL ALBERTO**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **“SIVENLAC” SISTEMA WEB DE INVENTARIO PARA CONTROL DE EXISTENCIAS EN LA BODEGA DE MATERIALES, DE LA EMPRESA “PASTOLAC”**, en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y autoría.

Latacunga, 28 Junio del 2019

.....
Vergara Bustillos, Paúl Alberto

C.C.: 0502832827

DEDICATORIA

A toda mi familia por ellos soy quien soy.

A mi esposa Mimí por su amor, dedicación, confianza y por ser la persona que junto a mis hijos José Miguel y Matías impulsan mi vida cada día para seguir adelante y ser una mejor persona.

A mi padre Alberto por haberme apoyado en toda mi vida estudiantil, por brindarme los recursos necesarios para poder llevar a cabo mis estudios.

A mi madre Sara por su amor incondicional, por su apoyo, confianza y esfuerzo para brindarme una vida llena de alegría y satisfacción.

A mi abuelito Huguito eje principal de mi familia mi segundo padre por todo su amor, confianza, apoyo y enseñanzas en toda mi vida.

A mis queridas tías mis segundas madres por su apoyo incondicional en todo momento de mi vida.

A mis hermanas por el apoyo brindado día a día.

Paúl

AGRADECIMIENTO

Agradezco a toda mi familia por toda la confianza depositada en mí, por sus palabras de aliento y por el apoyo que me brindan día tras día.

A mi tutor Ing. Patricio Navas por haberme brindado su tiempo, orientación y esfuerzo para poder finalizar este proyecto.

A al Ing. Javier Montaluisa por impartir sus conocimientos durante mi formación como profesional.

A la Ing. Ximena López por su apoyo y palabras de aliento brindadas en los momentos más difíciles de la realización de este proyecto.

Al Ing. Lucas Garcés por la ayuda brindada para la realización de este proyecto.

Paúl

ÍNDICE DE CONTENIDO

CARÁTULA

CERTIFICADO DEL DIRECTOR.....	i
AUTORÍA DE RESPONSABILIDAD.....	ii
AUTORIZACIÓN.....	iii
DEDICATORIA.....	iv
AGRADECIMIENTO.....	v
ÍNDICE DE CONTENIDOS	vi
ÍNDICE DE FIGURAS.....	xi
ÍNDICE DE TABLAS.....	xiii
RESUMEN.....	xiv
ABSTRACT.....	xv

CAPÍTULO I

DEFINICIÓN DEL PROBLEMA

1.1	Introducción.....	1
1.2	Planteamiento del problema.....	1
1.3	Formulación del Problema.....	2
1.4	Antecedentes.....	2
1.5	Justificación elmportancia.....	3
1.5.1	Justificación técnica.....	3
1.5.2	Justificación social.....	3
1.6	Objetivos.....	4

1.6.1	Objetivo General.....	4
1.6.2	Objetivos Específicos.....	4
1.7	Hipótesis.....	4
1.8	Operación de variables de Investigación.....	5
1.9	Alcances y limitaciones de la investigación.....	6
1.9.1	Alcances.....	6
1.9.2	Limitaciones.....	7

CAPÍTULO II

MARCO TEÓRICO

2.1	Introducción.....	8
2.2	Fundamentos Teóricos.....	8
2.2.1	Descripción general de la empresa.....	8
a.	Ubicación.....	10
b.	Productos elaborados en la empresa.....	10
2.2.2	Inventario.....	11
2.2.3	Metodología de desarrollo de software.....	12
a.	Metodología XP.....	12
b.	Características de la Metodología XP.....	13
c.	Proceso XP.....	15
d.	Ciclo de vida de XP.....	15

e.	Prácticas XP.....	18
2.2.4	Visual Studio Code.....	22
2.2.5	Sistema Web.....	22
a.	Características de un Sistema Web.....	23
b.	Sistema web de Inventarios.....	24
2.2.6	Base de Datos.....	24
a.	Base de datos NoSQL.....	24
b.	Base de Datos MongoDB.....	25
2.2.7	Servidor Web.....	26
a.	Node.JS.....	26
b.	Servidor Apache.....	27

CAPÍTULO III

DESARROLLO DEL SISTEMA WEB "SIVENLAC"

3.1	Introducción.....	29
3.2	Métodos y técnicas.....	29
3.2.1	Diseño de la investigación.....	29
3.2.2	Tipo de investigación.....	29
3.2.3	Métodos.....	29
3.3	Aplicación de la Metodología XP.....	30
3.3.1	Fase 1: Exploración.....	30

a.	Contenido del sistema web “SIVENLAC”.....	30
b.	Condiciones de uso.....	37
c.	Recursos.....	37
d.	Factibilidad.....	38
3.3.2	Fase 2: Diseño.....	39
a.	Diseño Instruccional.....	39
b.	Diseño de Pantallas.....	40
3.3.3	Fase 3: Iteraciones.....	47
a.	Arquitectura.....	47
3.3.4	Fase IV: Producción.....	61
a.	Descripción de Interfaces “SIVENLAC”	61
b.	Pantalla de Inicio.....	61
c.	Menú Principal “SIVENLAC”	63
d.	Modulo Iniciar Sesión.....	64
e.	Módulo Listado de Documentos.....	64
f.	Módulo Stock Actual.....	68
g.	Módulo Bodegas.....	69
h.	Módulo Ítems.....	71
i.	Módulo Terceros.....	73

j. Módulo Usuarios.....	75
-------------------------	----

3.3.5 Fase V: Mantenimiento.....	77
----------------------------------	----

3.3.6 Fase VI: Muerte del Proyecto.....	77
---	----

CAPÍTULO IV

PRUEBAS DE VALIDACIÓN

3.4 Introducción.....	78
-----------------------	----

3.4.1 Inicio del servidor de la base de datos MongoDB.....	78
--	----

3.4.2 Restauración de la Base de Datos INVENDB utilizada en el sistema web “SIVENLAC”.....	79
---	----

3.4.3 Inicio del servidor Node.JS.....	81
--	----

3.4.4 Inicio de “SIVENLAC”.....	83
---------------------------------	----

3.4.5 Pruebas operaciones CRUD.....	84
-------------------------------------	----

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones.....	87
-----------------------	----

5.2 Recomendaciones.....	87
--------------------------	----

REFERENCIAS BIBLIOGRÁFICAS.....	89
---------------------------------	----

ANEXOS.....	110
-------------	-----

ÍNDICE DE FIGURAS

Figura 1. Productos Elaborados PASTOLAC	11
Figura 2. Proceso XP.....	22
Figura 3. Característica Sistema Web.	24
Figura 4. Logo MongoDB.....	26
Figura 5. Logo Node.JS.....	27
Figura 6. Logo Apache.....	28
Figura 7. Mapa Interacción "SIVENLAC"	39
Figura 8. Diseño Pantalla Principal "SIVENLAC".....	40
Figura 9. Diseño Listado de Documentos.....	41
Figura 10. Diseño Agregar Producto.	41
Figura 11. Diseño Crear Documento.	42
Figura 12. Diseño Stock Actual.....	42
Figura 13. Diseño Bodegas.	43
Figura 14. Diseño Crear Bodega.	43
Figura 15. Diseño Ítems.....	44
Figura 16. Diseño Crear Ítem.....	44
Figura 17. Diseño Terceros.	45
Figura 18. Diseño Crear Terceros.	45
Figura 19. Diseño Usuarios.	46
Figura 20. Diseño Crear Usuarios.	46
Figura 21. Tiempo de Conexión con Node.JS.	47
Figura 22. Tiempo de Conexión con Apache.....	48
Figura 23. Servidor y Ancho de banda del usuario con Node.JS.....	49
Figura 24. Servidor y Ancho de banda del usuario con Apache.	49
Figura 25. Consumo de Memoria con Node.JS.	50
Figura 26. Consumo de Memoria con Apache.....	51
Figura 27. Transferencia de datos con Node.JS.....	52
Figura 28. Transferencia de datos con Apache.	53
Figura 29. Tiempo de conexión en la consulta de datos con Node.JS.	54
Figura 30. Tiempo de conexión en la consulta de datos con Apache.....	55
Figura 31. Servidor y Ancho de banda de usuario en la consulta de datos con Node.JS.....	56
Figura 32. Servidor y Ancho de banda de usuario en la consulta de datos con Apache.....	57
Figura 33. Consumo de memoria en la consulta de datos con Node.JS	58
Figura 34. Consumo de memoria en la consulta de datos con Apache.....	58

Figura 35. Transferencia de datos en la consulta de información con Node.JS.	59
Figura 36. Transferencia de datos en la consulta de información con Apache.....	60
Figura 37. Pantalla de Inicio "SIVENLAC".	62
Figura 38. Menú Principal "SIVENLAC".....	63
Figura 39. Iniciar Sesión "SIVENLAC".	64
Figura 40. Modulo Listado de Documentos "SIVENLAC".	64
Figura 41. Elaboración Documento.	65
Figura 42. Agregar Producto.....	66
Figura 43. Elaboración Documento 1.	67
Figura 44. Modulo Stock Actual "SIVENLAC".	68
Figura 45. Modulo Bodegas "SIVENLAC".	69
Figura 46. Creación Bodega.	70
Figura 47. Módulo Ítems "SIVENLAC".	71
Figura 48. Creación Ítem.	72
Figura 49. Modulo Terceros "SIVENLAC".	73
Figura 50. Creación Tercero.	74
Figura 51. Modulo Usuarios "SIVENLAC".	75
Figura 52. Creación Usuario.	76
Figura 53. Inicio Servidor MongoDB.	78
Figura 54. Servidor MongoDB iniciado.	79
Figura 55. Restore MongoDB.	80
Figura 56. Base de Datos MongoDB restaurada.	80
Figura 57. Inicio Server.js.	81
Figura 58. Server.js iniciado.	82
Figura 59. Iniciando "SIVENLAC".	83
Figura 60. "SIVENLAC".	83
Figura 61. CRUD - Create	84
Figura 62. CRUD - Read.....	85
Figura 63. CRUD - Update.....	85
Figura 64. CRUD - Delete.....	86

ÍNDICE DE TABLAS

Tabla 1. <i>Variables de la Investigación</i>	5
Tabla 2. <i>Contenido técnico "SIVENLAC"</i>	30
Tabla 3. <i>Descripción Pantalla Principal "SIVENLAC"</i>	31
Tabla 4. <i>Descripción Módulo Listado de Documentos</i>	32
Tabla 5. <i>Descripción Ventana Crear Documento</i>	32
Tabla 6. <i>Descripción Módulo Crear Documento</i>	33
Tabla 7. <i>Descripción Módulo Stock Actual</i>	33
Tabla 8. <i>Descripción Módulo Bodegas</i>	34
Tabla 9. <i>Descripción ventana crear bodega</i>	34
Tabla 10. <i>Descripción Módulo Ítems</i>	35
Tabla 11. <i>Descripción ventana crear ítem</i>	35
Tabla 12. <i>Descripción Módulo Terceros</i>	35
Tabla 13. <i>Descripción ventana crear tercero</i>	36
Tabla 14. <i>Descripción Módulo Usuarios</i>	36
Tabla 15. <i>Descripción ventana creación usuario</i>	37
Tabla 16. <i>Servidor y Ancho de banda del usuario</i>	49
Tabla 17. <i>Consumo de Memoria</i>	50
Tabla 18. <i>Transferencia de Datos</i>	52
Tabla 19. <i>Tiempo de conexión en la consulta de datos</i>	54
Tabla 20. <i>Servidor y Ancho de banda de usuario en la consulta de datos</i>	55
Tabla 21. <i>Consumo de memoria en la consulta de datos</i>	58
Tabla 22. <i>Transferencia de datos en la consulta de información</i>	59
Tabla 23. <i>Pruebas CRUD</i>	84

RESUMEN

El proyecto de titulación tiene como finalidad el desarrollo e implementación de un sistema web de inventarios “SIVENLAC” para la empresa “PASTOLAC” de la parroquia San Juan de Pastocalle, mismo que será capaz de dar soluciones al ingreso, egreso y consulta de productos en stock; será implementado con la base de datos NoSQL Mongo DB; y ejecutado por el servidor web que se acople a las características del sistema; “SIVENLAC” está desarrollado con herramientas de última generación, en la fase de desarrollo se encuentran cosas relevantes que son motivo de investigación para lograr encontrar el sentido de su funcionamiento y desempeño dentro de algunos procesos particularmente en la infraestructura, como parte del desarrollo del sistema en la estructura se utiliza Node.js por su robustez y agilidad en algunos de los procesos y partiendo de la utilización surge la necesidad de medir rendimientos entre Apache vs Node.js para determinar la arquitectura más óptima y cual puede tener mejor interacción con una base de datos NoSQL en el manejo del inventario; en estas pruebas se midió rendimiento, carga y stress, como producto del desarrollo de “SIVENLAC” se pudo demostrar que la utilización de una base de datos NoSQL en un sistema web de inventarios es óptimo y que en algunos casos mejoró el desempeño con relación a una base de datos relacional.

PALABRAS CLAVE:

- **SERVIDORES WEB**
- **MONGO DB**
- **EMPRESA PASTOLAC**

ABSTRACT

The title project aims to develop and implement a web inventory system "SIVENLAC" for the company "PASTOLAC" of the parish of San Juan de Pastocalle, which will be able to provide solutions to the entry, exit and consultation of products in stock, will be implemented with the database NoSQL Mongo DB, and executed by the web server that is coupled to the characteristics of the system; "SIVENLAC" is developed with last generation tools, in the development phase there are relevant things that are reason for research to find the sense of its operation and performance within some processes particularly in the infrastructure, as part of the development of the system in the structure Node is used.js for its robustness and agility in some of the processes and from the use arises the need to measure performance between Apache vs Node.js to determine the most optimal architecture and which can have better interaction with a NoSQL database in inventory management, in these tests measured performance, load and stress, as a product of the development of "SIVENLAC" could demonstrate that the use of a NoSQL database in a web inventory system is optimal and that in some cases improved performance in relation to a relational database.

KEYWORDS:

- **WEB SERVERS**
- **MONGO DB**
- **COMPANY PASTOLAC**

CAPÍTULO I

Definición del Problema

1.1 Introducción

En el presente capítulo se expone una parte del proyecto de investigación, en donde se resalta los contenidos generales del problema actual, su justificación, importancia, y los objetivos a alcanzar dentro de la investigación.

Dentro del resultado final se obtendrá un software que permitirá agilizar el proceso de registro de productos existentes en stock de “PASTOLAC”, haciendo buen uso de la tecnología como una herramienta que facilitara el control y actualización de datos de productos.

1.2 Planteamiento del problema

La empresa procesadora de lácteos “PASTOLAC” al momento no cuenta con un sistema digital que controle el inventario de productos en stock; esta investigación tiene como objetivo la creación e implementación de un Sistema Web utilizando una base de datos no relacional Mongo DB; la misma que nos permitirá definir el servidor web más apropiado, para las operaciones básicas de una base de datos inserción, consulta, actualización y borrado.

1.3 Formulación del Problema

En base a lo anteriormente expuesto se formula el siguiente problema: ¿Cómo facilitar el control y actualización de existencias en la bodega de producto terminado, de la empresa “PASTOLAC”?

1.4 Antecedentes

PASTOLAC, es una empresa procesadora de lácteos ubicada en la parroquia San Juan de Pastocalle, creada por un grupo organizado de pequeños productores de leche de la parroquia, quienes hicieron un estudio socio económico de la población y crearon al principio un centro de acopio de leche, para ser distribuida en las grandes empresas procesadoras de lácteos del Cantón Latacunga; sin embargo el bajo costo de entrega del producto a estas empresas los obligo a formar su propio emprendimiento de procesamiento de productos lácteos.

En el año 2014 se inicia el procesamiento de leche, con la producción de quesos, ampliándose posteriormente a yogurt y en la actualidad helados de crema; los mismos que poco a poco han ido ganando mercado no sólo a nivel local, sino también a nivel provincial y nacional; PASTOLAC cuenta con todos los permisos, normas y registros sanitarios como lo estipula la ley orgánica de nuestro país.

PASTOLAC no cuenta con un sistema digital que ayude a agilizar el proceso de registro de productos existentes en stock, lo que permite que se pueda aplicar nuestra investigación en la empresa.

1.5 Justificación e Importancia

Un sistema de inventarios es una herramienta principal de control, que establecerá una relación detallada, ordenada y valorada dentro de su almacenamiento.

La procesadora de lácteos PASTOLAC no cuenta actualmente con un sistema digital de inventario que permita controlar los productos procesados en stock de forma eficiente y oportuna.

El presente proyecto busca facilitar el control y actualización de datos de productos existentes en las bodegas de PASTOLAC, generando mayores recursos económicos a la empresa, optimización de registros, recursos, tiempo y mejorando la calidad de vida de los pequeños productores.

1.5.1 Justificación técnica

“SIVENLAC” es una herramienta software útil y fiable para ayudar a la empresa “PASTOLAC” a gestionar el proceso de control de su inventario de productos elaborados, además de brindar información en tiempo real que se encuentra almacenada en una base de datos no relacional MongoDB, que sirve para manejar la información de una manera rápida, eficaz y escalable, permitiendo a los usuarios acceder a la información disponible desde cualquier departamento de la empresa.

1.5.2 Justificación social

Con la implementación de “SIVENLAC”, se permite el ingreso y registro de los productos elaborados diariamente, obteniendo grandes beneficios tales como generar mayores recursos económicos a la empresa, optimización de registros,

recursos, tiempo, mejora en la calidad de vida de los pequeños productores, y brindar un mejor servicio a sus clientes.

1.6 Objetivos

1.6.1 Objetivo General

Diseñar, desarrollar e implementar un sistema web “SIVENLAC” que actualice y agilite el manejo del inventario de productos en stock en PASTOLAC.

1.6.2 Objetivos Específicos

- Analizar la información existente para el desarrollo del sistema web “SIVENLAC” a través de metodologías orientadas a la creación de aplicaciones web.
- Diseño e implementación de una base de datos No Relacional Mongo DB para “SIVENLAC” orientada a la pequeña y mediana empresa.
- Implementar una aplicación web “SIVENLAC” para el inventario de productos de la empresa PASTOLAC.
- Realizar la comparativa de “SIVENLAC” en los servidores web Apache y Node.JS utilizando las herramientas para pruebas de rendimiento, carga y stress.

1.7 Hipótesis

Con la implementación del sistema web de inventario, se emplearán mejor los recursos disponibles en PASTOLAC, generando un mejor registro de los

productos por medio del proceso ágil y oportuno de ingresos, egresos y actualización de datos de los mismos.

1.8 Operación de variables de Investigación

Tabla 1

Variables de la Investigación.

Variables	Concepto	Indicadores	Instrumentos
Independiente: Sistema web de Inventarios	Aplicación desarrollada con el propósito de mejorar el control de productos elaborados por la empresa "PASTOLAC"	Manejo y comprensión del sistema web por parte de los usuarios.	Herramienta Software
Variable Dependiente: Control del inventario en stock de productos elaborados.	Se optimizan los recursos en el proceso actualización de los productos en stock-control de inventarios.	<ul style="list-style-type: none"> • Actualización de datos • Ingreso de Datos • Transferencia de datos • Visualización de datos 	Pruebas de Rendimiento

1.9 Alcances y limitaciones de la investigación

1.9.1 Alcances

El proyecto de titulación consiste en desarrollar un sistema de control de inventario para facilitar y agilizar el modo de almacenamiento y control de los productos elaborados, con la finalidad de mejorar el desempeño de “PASTOLAC” para brindar un mejor servicio a sus clientes.

El sistema web “SIVENLAC”, está diseñado con los siguientes módulos:

- **Modulo Principal**

Es la pantalla principal del sistema donde se encuentra los accesos a todas las funcionalidades del sistema.

- **Listado de Documentos**

Mediante este enlace se pueden ver los documentos elaborados.

- **Crear Documento**

Mediante este enlace se pueden elaborar documentos..

- **Stock Actual**

Mediante este enlace se puede observar el stock actual de los ítems en las bodegas.

- **Movimiento Inventario**

Este enlace muestra el movimiento de inventario en las bodegas.

- **Bodegas**

Administración de las bodegas en las cuales se puede realizar ingresos y egresos.

- **Ítems**

Administración de los ítems que se utilizarán para generar una factura.

- **Terceros**

Administración de clientes o proveedores.

- **Usuarios**

Administración de los usuarios que inician sesión.

1.9.2 Limitaciones

Entre las limitaciones que se encontraron tenemos las siguientes:

- La información para el desarrollo e implementación del sistema web fue escasa ya que los empleados no están acostumbrados a manejar sistemas digitales, todo el proceso de almacenamiento se lo realiza de forma manual, se lleva la contabilidad de los productos terminados en cuadernos, por lo que no cuentan con un inventario claro de los productos que mantienen en stock.
- Al ser una empresa joven la introducción de los datos en “SIVENLAC” se dará a partir de la implementación del mismo, para las pruebas de funcionamiento fue necesario el uso de datos ficticios.

CAPÍTULO II

MARCO TEÓRICO

2.1 Introducción

En este capítulo se dan a conocer cada uno de los temas que abarcan el desarrollo e implementación del sistema web “SIVENLAC”; definiciones, características, funciones, estructuras, etc.

2.2 Fundamentos Teóricos

2.2.1 Descripción general de la empresa

- **Nombre de la asociación:** Asociación Agroecológicos Pastocalle “AGROPAS”
- **Nombre comercial:** “PASTOLAC” Lácteos de Pastocalle

AGROPAS se constituye con la participación de 25 mujeres y es reconocido jurídicamente por el Consejo de Desarrollo de las Nacionalidades y Pueblos del Ecuador “CODENPE”, cuyo objetivo es trabajar organizadamente en búsqueda de nuevas alternativas de producción, transformación y comercialización de productos alimenticios con enfoque agroecológico.

Desde diciembre del 2014 AGROPAS ingresa a una nueva línea de trabajo que es la recolección, acopio, enfriamiento y venta de leche con el apoyo económico de HEIFER y dotación de equipos por parte del MAGAP, desde marzo del 2015 se inicia elaborando Queso Fresco de forma casera,

mientras que a partir de Agosto del 2015 AGROPAS recibe recursos económicos del Instituto de Economía Popular y Solidaria para equipamiento de transformación de lácteos.

AGROPAS mantiene una línea de trabajo muy cercano con la Cooperativa de Ahorro y crédito Pucara Ltda., fruto de ese trabajo en conjunto la cooperativa entrega en comodato un lote de terreno para la construcción del centro de Acopio.

Bajo el nombre comercial de "PASTOLAC" comienza la producción de leche, acopio de la misma, transformación y comercialización de lácteos como: queso fresco, yogurt, helados, crema de leche y queso mozzarella; para esta actividad la empresa cuenta con tanque de enfriamiento, equipos de laboratorio, tanques de transportación en acero inoxidable, bidones y baldes de acero inoxidable para cada uno de los socios/as, también cuenta con un caldero, meza de desuerado, un cuarto frío, una prensa manual, un fechador, un sellador, bombas de succión, los productos como queso fresco, mozzarella, y yogurt cuenta con certificación sanitaria y trabaja con dos trabajadores directos y 22 indirectos, toda transacción se realiza a través de la cooperativa, es decir que los pagos a los socios proveedores de leche es transferido a las cuentas individuales a la libreta de ahorro de cada uno de los socios.

a. Ubicación

“PASTOLAC” está ubicada en la:

- **Provincia:** Cotopaxi
- **Cantón:** Latacunga
- **Parroquia:** San Juan de Pastocalle
- **Barrio:** Pucara
- **Calle:** Sucre a 50 metros antes del CVB Pulgarcito, junto a la Cooperativa de Ahorro y Crédito Pucara Ltda.

b. Productos elaborados en la empresa

Actualmente “PASTOLAC” elabora 6 productos principalmente los cuales son:

- Yogurt de diferentes sabores y presentaciones.
 - Yogurt de 100 cc
 - Yogurt de 250 cc
 - Yogurt de 500 cc
 - Yogurt de 1000 cc
 - Yogurt de 2000 cc
 - Yogurt de 4000 cc
- Queso fresco de diferentes presentaciones.
 - Queso Fresco de 500 gramos
 - Queso Fresco de 140 gramos
- Queso mozzarella de diferentes presentaciones.
 - Queso Mozzarella de 500 gramos

- Queso Mozzarella de 140 gramos
- Helados de diferentes sabores y presentaciones.
 - Helados 100 gramos
 - Helados 50 gramos
- Crema de leche.
- Leche cruda.

Estos productos son comercializados dentro y fuera de la parroquia.



Figura 1. Productos Elaborados PASTOLAC

2.2.2 Inventario

Los inventarios es la cantidad de bienes que una empresa mantiene en existencia en un momento determinado, estos bienes pueden ser para ventas o también para ser utilizados en la producción constituye el nexo entre la producción y la venta de un producto y representa una inversión considerable para la empresa. (Duran, 2012).

El inventario en una empresa manufacturera está conformado por la materia prima, insumos o productos en proceso necesarios para la fabricación y

posterior comercialización en productos terminados, los inventarios permiten asegurar la subsistencia del negocio y el desarrollo de sus actividades operativas. (Duran, 2012).

La administración del inventario es un tema central para evitar problemas financieros en las organizaciones, es un componente fundamental en la productividad de una empresa, ya que es el activo corriente de menor liquidez que manejan y que además contribuye a generar rentabilidad. Es el motor que mueve a la organización, pues es la base para la comercialización de la empresa que le permite obtener ganancias. (Duran, 2012).

2.2.3 Metodología de desarrollo de software.

Las metodologías de desarrollo de software son un marco de trabajo eficiente que surgió en la década de los años 70 ya que ofrecían una respuesta a los problemas que surgían con los antiguos métodos de desarrollo, los cuales se enfocaban en la creación de software sin el control apropiado de las actividades del grupo de trabajo, lo que provocaba un producto lleno de deficiencias y problemas resultando en la insatisfacción del cliente, pues se le ofrecía un software que no cumplía con sus necesidades. (Garcés & Egas, 2015).

a. Metodología XP

XP se encuentra en el grupo de las conocidas metodologías ágiles, está centrada básicamente en el trabajo en equipo, se basa en que con unas buenas relaciones interpersonales se conlleva a obtener el éxito en el desarrollo del proyecto de software, es una metodología que se preocupa en gran parte del

aprendizaje de las personas encargadas del proceso trata de crear el mejor ambiente de trabajo para llegar al objetivo en el menor tiempo posible. XP busca siempre la constante participación del cliente con los desarrolladores, está enfocada en obtener una comunicación confiable entre todos los miembros del proyecto incluido el cliente, trata de buscar soluciones rápidas y eficientes en cualquier cambio que se presente dentro del desarrollo del proyecto. XP está diseñado y adecuado para aquellos proyectos que tienen unos requisitos imprecisos, cambiantes o poco entendibles, todo esto provoca que exista un riesgo técnico en la elaboración del proyecto. (Letelier & Penadés, 2012).

b. Características de la Metodología XP

- **Historias de Usuarios**

Son utilizadas para especificar los requisitos del software, el cliente escribe sobre que le gustaría que haga el software, que características debe poseer, estos requisitos expuestos por el usuario pueden ser una función específica que debe realizar el software o simplemente algún requisito que el usuario exponga y que no tenga relevancia para el desarrollo, las historias de usuario son cambiantes por el hecho de que en cualquier momento pueden romperse y reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. (Letelier & Penadés, 2012).

- **Roles XP**

- **Programador:** Es quien se encarga de escribir o desarrollar el código del sistema.
- **Cliente:** Es el encargado de escribir las historias de usuario, y de realizar las pruebas de funcionalidad del sistema, escoge y da prioridad a cada una de las historias de usuario para que sean implementadas. (Letelier & Penadés, 2012).
- **Encargado de pruebas:** Es el encargo de realizar las pruebas al software conjuntamente con el usuario, expone los resultados obtenidos a los miembros del equipo.
- **Encargado de seguimiento:** Es quien se encarga de verificar que el proceso de desarrollo de software se esté cumpliendo según el cronograma establecido, está pendiente de los resultados obtenidos, comunica las novedades ocurridas en el desarrollo para poder mejorar el tiempo en futuros procesos, está encargado de que los objetivos en el proceso de desarrollo se estén alcanzando y de ser necesario puede realizar cambios para que estos se cumplan.
- **Entrenador:** Es el encargado del proceso global, provee guías para que se aplique cada una de las fases y prácticas de XP, se encarga de que todo el proceso vaya realizándose correctamente.
- **Consultor:** Es un miembro fuera del equipo que puede ayudar a dar solución a un problema específico, por tal razón el consultor debe tener

conocimientos específicamente en el tema ya que deberá guiar al equipo para resolver dicho problema. (Letelier & Penadés, 2012).

- **Gestor:** Es el encargado de la coordinación, es el vínculo entre clientes y programadores, crea las condiciones adecuadas para que el equipo trabaje efectivamente.

c. Proceso XP

Un proyecto XP tiene éxito cuando el cliente selecciona el valor de negocio a implementar basado en la habilidad del equipo para medir la funcionalidad que puede entregar a través del tiempo. El ciclo de desarrollo consiste (a grandes rasgos) en los siguientes pasos: (Letelier & Penadés, 2012).

1. El cliente define el valor de negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador construye ese valor de negocio.
5. Vuelve al paso 1.

d. Ciclo de vida de XP

Consiste de seis fases:

Fase I: Exploración

- Los clientes plantean las historias de usuario.

- El equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto.
- Se prueba la tecnología y se explora la posible arquitectura del sistema a utilizar.

Fase II: Planificación de la Entrega

- El cliente da prioridad a cada una de las historias de usuario elaboradas.
- Los programadores deben realizar una evaluación del tiempo y esfuerzo que será necesario para la implementación de cada una de las historias de uso..
- Se realiza y determina el cronograma de entrega del primer producto el cual debería ser en no más de 3 meses, esto se lo realiza conjuntamente con el cliente.
- Los programadores establecen las estimaciones de esfuerzo asociado a la implementación de las historias.

Fase III: Iteraciones

Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado, el Plan de Entrega está compuesto por iteraciones de no más de tres semanas, al final de la última iteración el sistema estará listo para entrar en producción.

Los elementos que deben tomarse en cuenta durante la elaboración del Plan de la Iteración son: historias de usuario no abordadas, velocidad del

proyecto, pruebas de aceptación no superadas en la iteración anterior y tareas no terminadas en la iteración anterior. Todo el trabajo de la iteración es expresado en tareas de programación, cada una de ellas es asignada a un programador como responsable, pero llevadas a cabo por parejas de programadores. (Letelier & Penadés, 2012).

Fase IV: Producción

- Se requiere de pruebas adicionales y revisiones de rendimiento antes de que el sistema sea trasladado al entorno del cliente.
- Se deben tomar decisiones sobre la inclusión de nuevas características a la versión actual.

Fase V: Mantenimiento

Mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones. Para realizar esto se requiere de tareas de soporte para el cliente. De esta forma, la velocidad de desarrollo puede bajar después de la puesta del sistema en producción. La fase de mantenimiento puede requerir nuevo personal dentro del equipo y cambios en su estructura. (Letelier & Penadés, 2012).

Fase VI: Muerte del Proyecto

- El cliente no tiene más historias para ser incluidas en el sistema.
- Se satisfacen las necesidades del cliente.

- Se genera la documentación final del sistema y no se realizan más cambios.
- Cuando el sistema no brinda los beneficios que el cliente buscaba o ya no se cuenta con el presupuesto para el mantenimiento del mismo también se considera como una muerte del proyecto.

e. Prácticas XP

Dentro de XP siempre debe existir primordialmente la posibilidad de disminuir el costo que conlleva el realizar cambios en el desarrollo del software a lo largo de todo el proyecto, para obtener siempre una evolución que funcione satisfactoriamente para el cliente.

Con el uso adecuado de tecnologías para el desarrollo de software, XP trata de que el costo de elaboración e implementación crezca de forma lenta, también se ayuda adicionalmente de prácticas disciplinadas que mostramos a continuación:

- **El juego de la planificación**
 - Comunicación habitual entre cliente y programadores.
 - Estimación de esfuerzo y tiempo necesario para la implementación de cada una de las historias de usuario, participación de los clientes para decidir el ámbito y tiempo de entrega conjuntamente con el equipo técnico de cada iteración. (Letelier & Penadés, 2012).
 - Se da prioridad y esfuerzo a las historias de usuario, se establece el contenido de la entrega y/o iteración.

- Se realizan entregas pequeñas las cuales son versiones no completas del sistema pero si funcionales y operativas que pueden ayudar al negocio.¹

- **Metáfora**

- En XP se podría asumir que tiene una arquitectura evolutiva ya que se ayuda de una metáfora para determinar posibles inconvenientes explícitamente en el comienzo del proyecto. (Letelier & Penadés, 2012).
- Se plantea una metáfora o grupo de metáforas entre cliente y el equipo de desarrollo, en una metáfora se puede decir que se describe el cómo debería operar el sistema. (Letelier & Penadés, 2012).

- **Diseño simple**

- Con XP se pretende que el diseño sea lo más simple posible para evitar la complejidad innecesaria.

- **Pruebas**

- Las pruebas deben ser constantes y unitarias sobre todo en cada modificación del sistema, los clientes ayudan con las pruebas para poder validar cada historia de usuario escrita, se debe apoyar siempre esta actividad ya que es crucial en la implementación del sistema. (Letelier & Penadés, 2012).

- **Refactorización**

- Es una actividad que ayuda con la reestructuración del código con lo que se logra eliminar la duplicación de código, tiene que ser constante por que

puede mejorar significativamente la legibilidad, y hacerlo más flexible. (Letelier & Penadés, 2012).

- **Programación en parejas**

- Es aconsejable que el desarrollo del código se realice en parejas de programadores ya que nos puede brindar las siguientes ventajas:
 - Detección de errores
 - El producto final puede presentar una tasa menor de errores.
 - Diseño mejorado
 - Menos líneas de código
 - Problemas resueltos en menor tiempo.
 - Compartición de conocimientos.
 - Más personas conocen como fue desarrollado el código del software.
 - Mejora la socialización del grupo de programadores.

- **Propiedad colectiva del código**

- Cualquier miembro del grupo de desarrolladores puede modificar el código con esto se logra lo siguiente:
 - Apoyar con nuevas ideas para cada segmento del proyecto.
 - Que un programador no sea indispensable.

- **Integración continúa**

- Para que un código sea incorporado siempre se deber realizar pruebas en primera instancia con esto se reduce los esfuerzos de los desarrolladores al momento de reutilizar un código o compartir el mismo. (Letelier & Penadés, 2012).

- **40 horas por semana**
 - En lo posible siempre es necesario que se trabaje como máximo 40 horas ya que si se sobrepasa de este tiempo es posible que el grupo de trabajo se desmotive y empiece a crearse problemas innecesarios si esto ocurre hay que corregir inmediatamente el problema. (Letelier & Penadés, 2012).
- **Cliente in-situ**
 - El cliente la parte más importante del proyecto siempre debe estar disponible para el grupo de programadores..
 - El cliente conduce el trabajo para lograr el éxito del proyecto.
 - La comunicación oral siempre va a ser mejor entendida que la escrita.
- **Estándares de programación**
 - XP en todo momento da prioridad a la comunicación de los programadores por medio del código, es necesario que se sigan los siguientes estándares. (Letelier & Penadés, 2012).
 - Código legible para los miembros del equipo, para lograr la facilidad en los cambios.
 - Siempre comentar las prácticas que se van desarrollando para que sea una aplicación conjunta y equilibrada.



Figura 2. Proceso XP.

2.2.4 Visual Studio Code

Visual Studio Code es un editor de código fuente ligero pero potente que se ejecuta en su escritorio y está disponible para Windows, macOS y Linux. Viene con soporte incorporado para JavaScript, TypeScript y Node.js y tiene un rico ecosistema de extensiones para otros idiomas (como C ++, C #, Java, Python, PHP, Go) y tiempos de ejecución (como .NET y Unity). (Visual Studio Code).

2.2.5 Sistema Web

Un sistema web es básicamente una manera de facilitar el logro de una tarea específica en la Web, se han convertido en pocos años en complejos

sistemas con interfaces de usuario cada vez más parecidas a las aplicaciones de escritorio, por lo tanto permite al usuario interactuar directamente con los datos, todo en forma personalizada, dando servicio a procesos de negocio de considerable envergadura y estableciéndose sobre ellas requisitos estrictos de accesibilidad y respuesta. (Castejón Garrido, 2004).

a. Características de un Sistema Web

El vertiginoso avance tecnológico está encaminado con el despegue de la web y al progresivo manejo de grandes cantidades de datos, apoyados siempre de nuevos equipos tecnológicos que han obligado a las empresas a implementar nuevas arquitecturas que se adapten a sus necesidades; adicional el uso de intranets corporativas hacen que los sistemas web sean una gran herramienta de trabajo ya que permiten:

- Que la información pueda ser accesible desde cualquier punto dentro de la empresa.
- Si es necesario la información puede ser accesible desde fuera de la empresa.
- La información puede ser compartida en forma total o según corresponda a cada área de la empresa.

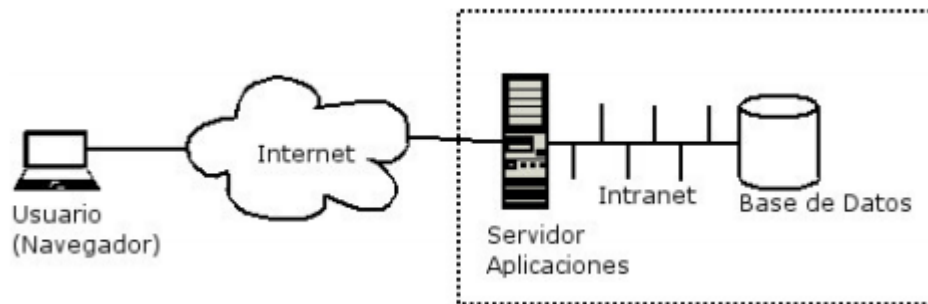


Figura 3. Característica Sistema Web.

Fuente: (Castejón Garrido, 2004).

b. Sistema web de Inventarios

Es aquel sistema que ha sido desarrollado para ayudar a la empresa a tener control sobre el ingreso y egreso de los productos que elabora, así también a tener un registro de la materia prima que tiene disponible en stock, también puede ser usado para controlar los bienes que posee una empresa; siendo una herramienta muy importante para regular el equilibrio de los flujos de entrada y salida.

2.2.6 Base de Datos

Una base de datos se considera como un “almacén” el cual nos permite guardar gran cantidad de datos e información de una manera más organizada y automática para que luego se pueda encontrar dicha información de una manera más fácil y exacta. (Valdés Pérez, 2007).

a. Base de datos NoSQL

Con los constantes cambios en las sociedades, se impulsan la aparición de nuevos modelos que satisfagan las nuevas necesidades, uno de estos es NoSQL, que se define como un conjunto de tecnologías que se apartan del modelo

tradicional de bases relacionales; aunque algunos autores lo describen como “un movimiento más que como una tecnología” (Castro Romero, Gonzáles Sanabria, & Callejas Cuervo, 2012).

El paradigma de bases de datos NoSQL surge a causa del cambio que se ha dado en el manejo de la información durante los últimos años, los volúmenes de información crecen a un ritmo sin precedentes, y cada vez se hace más compleja su administración; las empresas no solo desean almacenar esta información, sino quieren sacarle el mayor provecho; los usuarios piden cada vez más velocidad en las consultas, y la arquitectura de los sistemas ha tenido un cambio considerable. (Castro Romero, Gonzáles Sanabria, & Callejas Cuervo, 2012).

b. Base de Datos MongoDB

Es una base de datos basada en documentos y open source (código abierto) y se posiciona como la principal base de datos NoSQL, está escrita en el lenguaje c++ y ofrece a sus usuarios, escalabilidad y particionamiento, hablando en términos prácticos, MongoDB utiliza documentos para hacerla flexible, escalable y rápida. (Okman, Gal-Oz, Gonen, Gudes, & Abramov, 2011).

Guarda estructuras de datos en documentos similares a JSON con un esquema dinámico, MongoDB utiliza una especificación llamada BSON que es JSON binario; para almacenar sus documentos, provocando una integración de los datos con mayor rapidez y facilidad en algunas aplicaciones, en lugar de una columna que defina lo que debería estar en la línea, cada línea es un documento,

y este documento define y contiene los datos en sí. (Gupta, Bangare, Dalal, & Inamdar, 2016).



Figura 4. Logo MongoDB.

Fuente: (Ernesto Valero , 2016).

2.2.7 Servidor Web

Es aquel que se encarga de hacer funcionar las aplicaciones desarrolladas en un entorno web, haciéndolas accesibles a los clientes, debe encargarse de la creación y de la carga en memoria de las instancias de los componentes, es capaz de gestionar la disponibilidad de las aplicaciones. (Aumaille, 2002).

a. Node.JS

Node.js es un entorno de ejecución de Entrada y Salida sobre el intérprete de JavaScript que se ejecuta del lado del servidor, está basado en la implementación de Google en tiempo de ejecución con el motor V8, está desarrollado principalmente en lenguaje C y C++, centrándose en el rendimiento y en el bajo consumo de memoria. (Tilkov & Vinoski, 2010). La elección de la base de datos es muy importante a la hora de considerar un servidor web, debe ir con el framework del servidor seleccionado, Node.js puede sincronizarse perfectamente y trabajar en conjunto con MongoDB, ya

que almacena documentos como BSON, que no es más que JSON binario, en resumen, JSON es una forma JavaScript de mantener los datos, de ahí por qué MongoDB encaja tan bien en nuestro marco de trabajo centrado en JavaScript. (Gupta, Bangare, Dalal, & Inamdar, 2016).



Figura 5. Logo Node.JS.

Fuente: (Node.JS).

b. Servidor Apache

El servidor web Apache es un servidor Web gratuito y desarrollado por el Apache Server Project, cuyo objetivo es la creación de un servidor web fiable, eficiente y fácilmente extensible con código fuente abierto, es conjuntamente manejado por el “Grupo Apache” que es un grupo de voluntarios localizados alrededor del mundo que planean, desarrollan y producen toda la documentación relacionada con el servidor. (Márquez Díaz, Sampedro, & Vargas, 2002).

Apache tiene otras características como la indexación de directorios, uso de sobrenombres con las carpetas de negociación de contenidos, informes configurables sobre los errores HTTP, ejecución SetUID o Programas CGI,

administración de recursos para los procesos emparentados, mapas de imagen para los servidores de reescritura de URL, corrección de URL, etc. (Kabir, 2003).



Figura 6. Logo Apache.

Fuente: (García González, 2016).

CAPÍTULO III

DESARROLLO DEL SISTEMA WEB “SIVENLAC”

3.1 Introducción

En este capítulo se presentan los métodos y técnicas de investigación que han sido utilizadas para el proceso de desarrollo del proyecto, que tiene como objetivo agilizar el control de entradas y salidas de productos de la empresa “PASTOLAC”.

3.2 Métodos y técnicas

3.2.1 Diseño de la investigación

El presente proyecto de titulación ha sido estructurado de la siguiente manera:

3.2.2 Tipo de investigación

Investigación experimental: Mediante un software externo se realizaron pruebas a los servidores web Node.JS y Apache para poder determinar por medio de gráficas estadísticas cual es el más apropiado para ser implementado en el sistema web “SIVENLAC”, las pruebas se realizaron con un conjunto de 500 datos ficticios los cuales fueron insertados de manera automática por el software WebServer Streets Tool para los módulos de inserción y consulta de datos para cada uno de los servidores.

3.2.3 Métodos

Para el cumplimiento de las tareas se utilizaron los siguientes métodos de investigación:

- **Histórico – Lógico:** Para determinar los antecedentes históricos de conceptualización.
- **Inductivo – Deductivo:** Para la estructuración del marco teórico.
- **Hipotético – Deductivo:** Para conceptualización de la hipótesis.
- **Modelación:** Para determinar el objeto de la investigación.
- **Método Lógico:** Fue aplicado para determinar el servidor web que mejor se acoplo al sistema web.
- **Método Experimental:** Mediante un software externo se puso a prueba las arquitecturas de los servidores web.

3.3 Aplicación de la Metodología XP

3.3.1 Fase 1: Exploración

Se plantea a grandes rasgos como va estructurado el sistema web, el desarrollador se familiariza con las herramientas tecnológicas usadas para el proyecto, se prueba tecnología y se exploran opciones para la posible arquitectura del sistema.

a. Contenido del sistema web “SIVENLAC”

Tabla 2.

Contenido técnico "SIVENLAC".

Recurso	Propósito
Ventanas	Muestra cada una de las interacciones del sistema web con el usuario.
Menú	Se muestra cada uno de los módulos a los que el usuario tiene acceso.

CONTINÚA 

Iconos	Identificación de cada uno de los módulos del sistema web.
Texto	Información sobre cada uno de los módulos del sistema web.
Botones	Permite la interacción del sistema con el usuario.

Tabla 3

Descripción Pantalla Principal "SIVENLAC".

Elementos	Descripción
Encabezado	Nombre del Sistema
Texto	Breve descripción del sistema
Botón / Iniciar Sesión	Permite el ingreso al sistema
Botón / Icono / Listado de Documento	Mediante este enlace se pueden revisar diferentes tipos de documentos de ingresos o egresos.
Botón / Icono / Crear Documento	Mediante este enlace se pueden crear documentos de ingreso y egreso de productos.
Botón / Icono / Stock Actual	Mediante este enlace se puede observar el stock actual de los ítems en las bodegas.
Botón / Icono / Bodegas	Administración de las bodegas en las cuales se puede realizar ingresos y egresos.
Botón / Icono / Ítems	Administración de los ítems que se utilizarán para generar una factura.

CONTINÚA 

Botón / Icono / Terceros	Administración de clientes o proveedores.
Botón / Icono / Usuarios	Administración de los usuarios que inician sesión.

Tabla 4

Descripción Módulo Listado de Documentos.

Elementos	Descripción
Encabezado	Nombre de la Ventana
Lista / Documentos Generados	Lista de los Documentos que se han generado.
Botón / Crear Documento	Agrega un nuevo tipo de documento.

Tabla 5

Descripción Ventana Crear Documento.

Elementos	Descripción
Encabezado	Nombre de la Ventana
Texto / Caja de Texto / Ingreso # Orden	Se ingresa el número de orden
Texto / Lista / Ingreso – Egreso - Factura	Se selecciona el tipo de documento a generar.
Botón / Agregar Ítem	Agrega un nuevo producto al documento.
Lista / Productos Ingresados	Lista de los productos que se han ingresado al documento

CONTINÚA 

Botón / Guardar	Generar Documento
Botón / Ver Ordenes	Muestra todas las órdenes generadas.

Tabla 6

Descripción Módulo Crear Documento.

Elementos	Descripción
Encabezado	Nombre de la Ventana
Texto / Caja de Texto / Ingreso # Orden	Se ingresa el número de orden
Texto / Lista / Ingreso – Egreso - Factura	Se selecciona el tipo de documento a generar.
Botón / Agregar Ítem	Agrega un nuevo producto al documento.
Lista / Productos Ingresados	Lista de los productos que se han ingresado al documento
Botón / Guardar	Generar Documento
Botón / Ver Ordenes	Muestra todas las órdenes generadas.

Tabla 7

Descripción Módulo Stock Actual.

Elementos	Descripción
Encabezado	Nombre de la Ventana
Lista / Productos en stock	Listado de los productos existentes en cada una de las bodegas

Tabla 8*Descripción Módulo Movimiento Inventario*

Elementos	Descripción
Encabezado	Nombre de la Ventana
Lista	Lista de los movimientos realizados en el inventario

Tabla 9*Descripción Módulo Bodegas.*

Elementos	Descripción
Encabezado	Nombre de la Ventana
Lista	Lista de las bodegas creadas
Botón / Crear Bodega	Crear una nueva bodega para el ingreso de productos.

Tabla 10*Descripción ventana crear bodega*

Elementos	Descripción
Encabezado	Nombre de la Ventana.
Texto / Caja de Texto / Código Bodega	Ingresar un código para una nueva bodega.
Texto / Caja de Texto / Nombre Bodega	Ingresar nombre de la bodega.
Texto / Caja de Texto / Dirección	Ingresar dirección de la bodega.
Botón / Enviar	Crea una nueva bodega.
Botón / Ver Todos	Listado de todas las bodegas ingresadas.

Tabla 11*Descripción Módulo Ítems*

Elementos	Descripción
Encabezado	Nombre de la Ventana
Lista	Lista de los productos ingresados
Botón / Crear Ítem	Crear el ingreso de un nuevo producto

Tabla 12*Descripción ventana crear ítem*

Elementos	Descripción
Encabezado	Nombre de la Ventana.
Texto / Caja de Texto / Código Ítem	Ingresar un código para un nuevo ingreso de ítem.
Texto / Caja de Texto / Nombre	Ingresar nombre del ítem.
Texto/ Check / Grava I.V.A	Seleccionar si el ítem grava I.V.A.
Botón / Enviar	Crea el nuevo ítem.
Botón / Ver Todos	Muestra el listado de todos los ítems ingresados.

Tabla 13*Descripción Módulo Terceros.*

Elementos	Descripción
Encabezado	Nombre de la Ventana
Lista	Lista de terceros ingresados
Botón / Crear Tercero	Crear el ingreso de un nuevo proveedor o cliente

Tabla 14*Descripción ventana crear tercero*

Elementos	Descripción
Encabezado	Nombre de la Ventana.
Texto / Caja de Texto / Código Tercero	Ingresar un código para un nuevo ingreso de tercero.
Texto / Lista / Tipo	Seleccionar Cliente o Proveedor.
Texto / Caja de Texto / Razón Social	Ingresar una razón social.
Texto / Caja de Texto / Documento Identificación	Ingresar una documento de identificación.
Texto / Caja de Texto / Dirección	Ingresar una dirección.
Texto / Caja de Texto / Teléfono 1	Ingresar teléfono 1.
Texto / Caja de Texto / Teléfono 2	Ingresar teléfono 2.
Texto / Caja de Texto / Correo Electrónico	Ingresar correo electrónico.
Botón / Enviar	Crea el nuevo cliente o proveedor.
botón / Ver Todos	Muestra el listado de todos los clientes y proveedores.

Tabla 15*Descripción Módulo Usuarios*

Elementos	Descripción
Encabezado	Nombre de la Ventana
Lista	Lista de terceros ingresados
Botón / Crear Usuario	Crear el ingreso de un nuevo usuario

Tabla 16

Descripción ventana creación usuario

Elementos	Descripción
Encabezado	Nombre de la Ventana.
Texto / Caja de Texto / Código Usuario	Ingresar un código para un nuevo usuario del sistema.
Texto / Caja de Texto / Clave	Ingresar una clave para un nuevo usuario del sistema.
Botón / Enviar	Crea el nuevo usuario.
botón / Ver Todos	Muestra el listado de todos los usuarios ingresados.

b. Condiciones de uso.

“SIVENLAC” va hacer utilizado por la persona que esté a cargo del manejo del ingreso y egreso de los productos elaborados; y por aquellas personas que la dirección de la empresa crea que sea necesario la utilización y acceso al sistema web.

c. Recursos

Para la investigación se utilizaron los siguientes recursos:

- Técnicos
- Humanos
- Financieros
- Logísticos

➤ **Talento Humano**

- Investigador y desarrollador del proyecto
- Director del proyecto

➤ **Recurso Material**

- Computadora
- Impresora
- Carpetas
- Papel

➤ **Recurso Intangible**

- Internet
- Software
- Tiempo de investigación y desarrollo.

d. Factibilidad

- **Técnica:** La implementación del software se lo realizara con una base de datos MongoDB y con el servidor web Node.js.
- **Operativa:** El sistema está desarrollado de forma sencilla para que el usuario no tenga complicaciones al momento de interactuar con el sistema.
- **Económica:** El proyecto en su totalidad está financiado por la persona responsable del proyecto.

3.3.2 Fase 2: Diseño

Dentro de la fase de diseño encontraremos:

- Diseño Instruccional
- Diseño de Pantallas

a. Diseño Instruccional

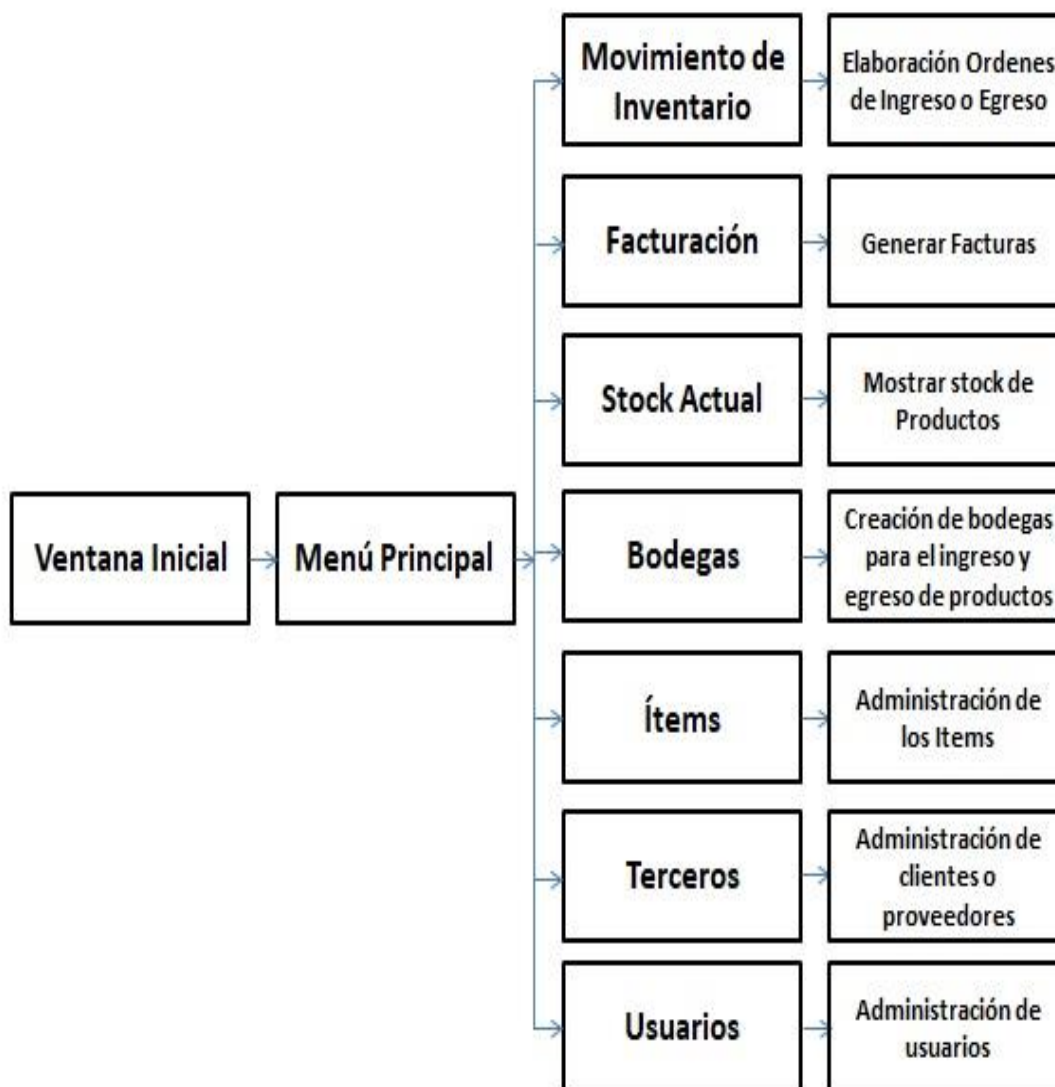


Figura 7. Mapa Interacción "SIVENLAC".

b. Diseño de Pantallas

Se presenta un diseño simple para cada uno de los módulos que va a contener el sistema web.

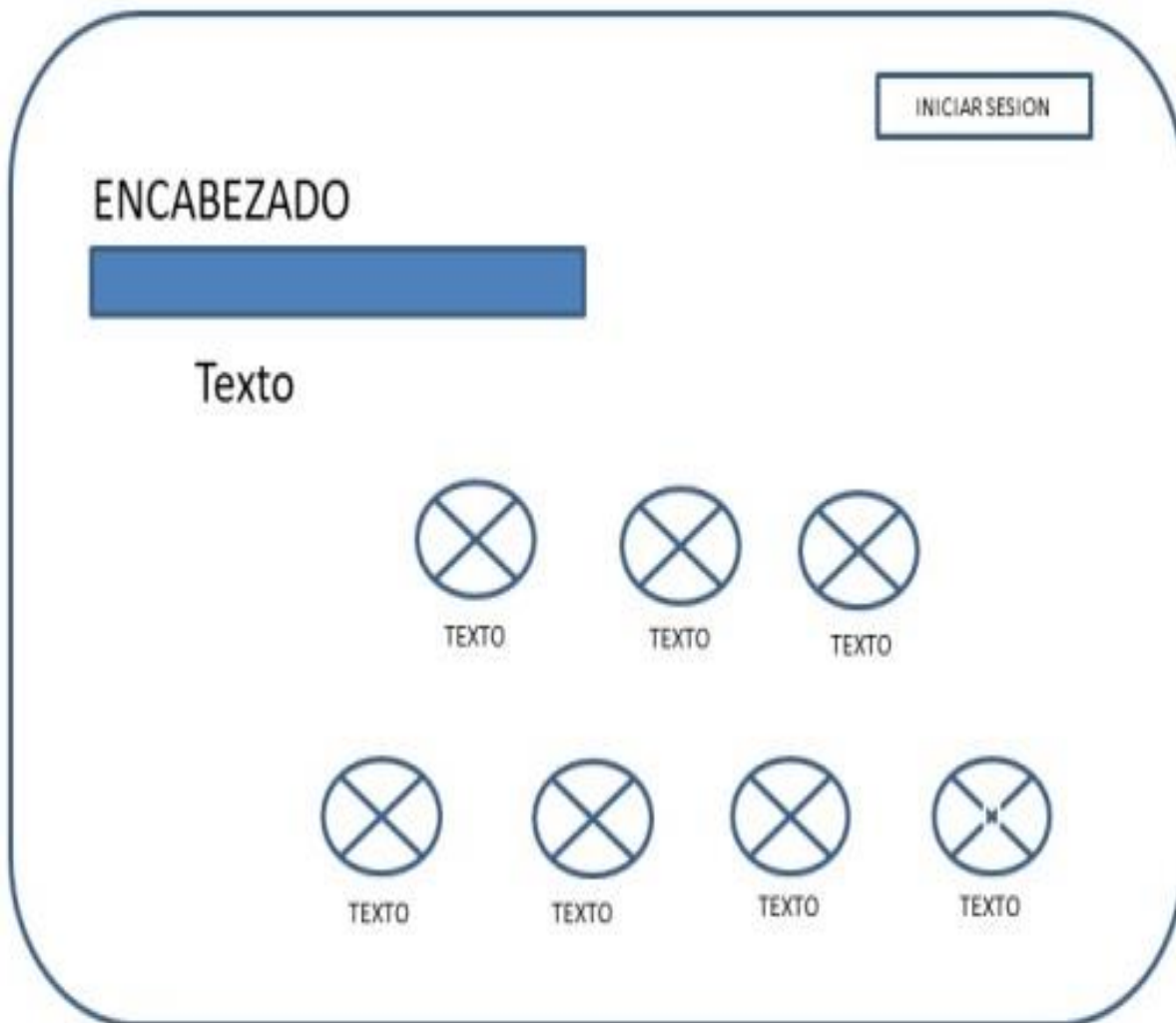


Figura 8. Diseño Pantalla Principal "SIVENLAC".



Figura 9. Diseño Listado de Documentos.



Figura 10. Diseño Agregar Producto.

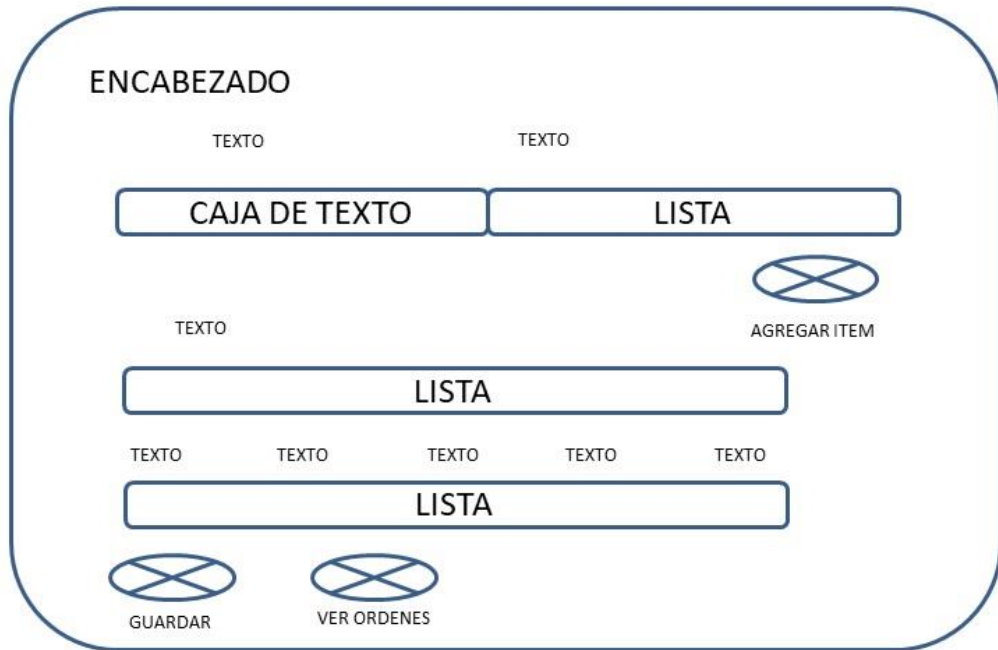


Figura 11. Diseño Crear Documento.



Figura 12. Diseño Stock Actual.



Figura 13. Diseño Bodegas.



Figura 14. Diseño Crear Bodega.



Figura 15. Diseño Ítems.

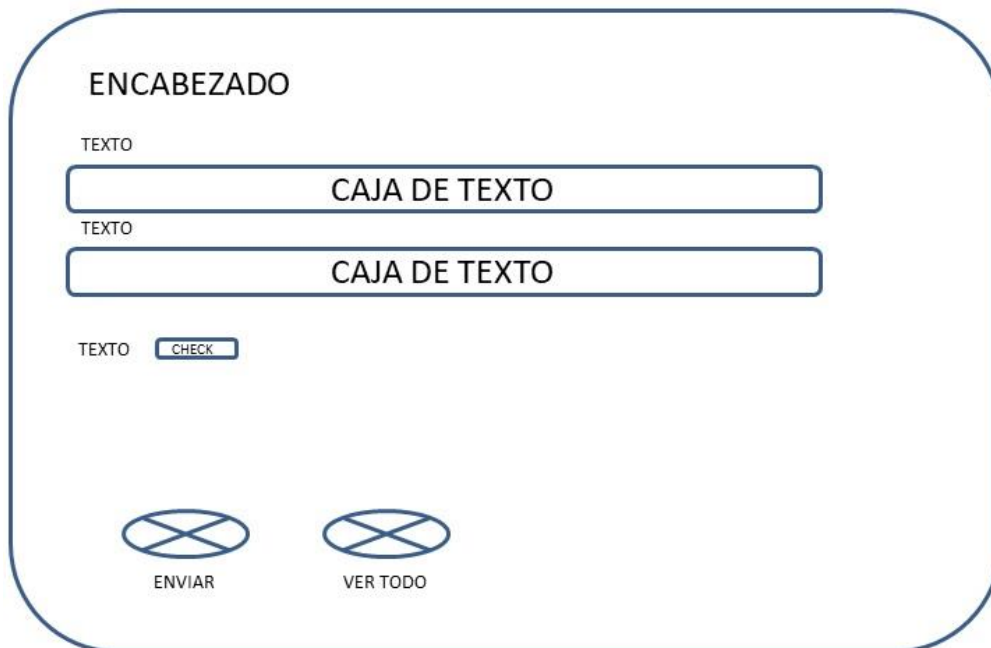


Figura 16. Diseño Crear Ítem.



Figura 17. Diseño Terceros.

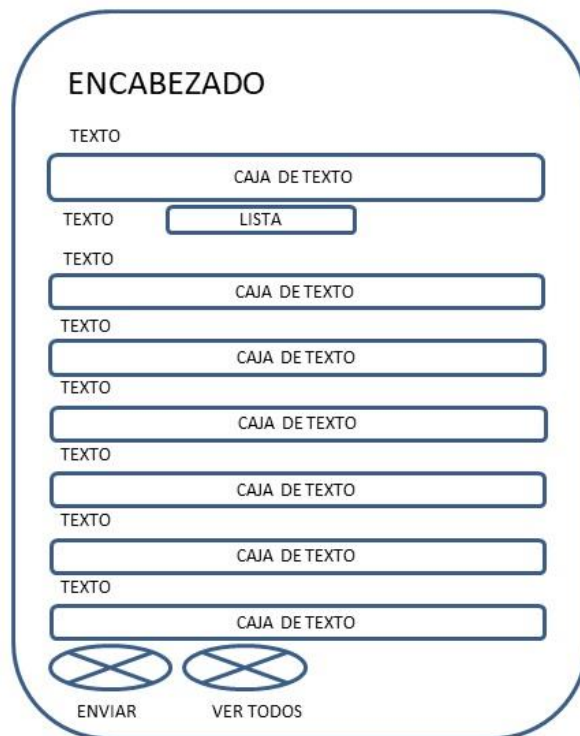


Figura 18. Diseño Crear Terceros.

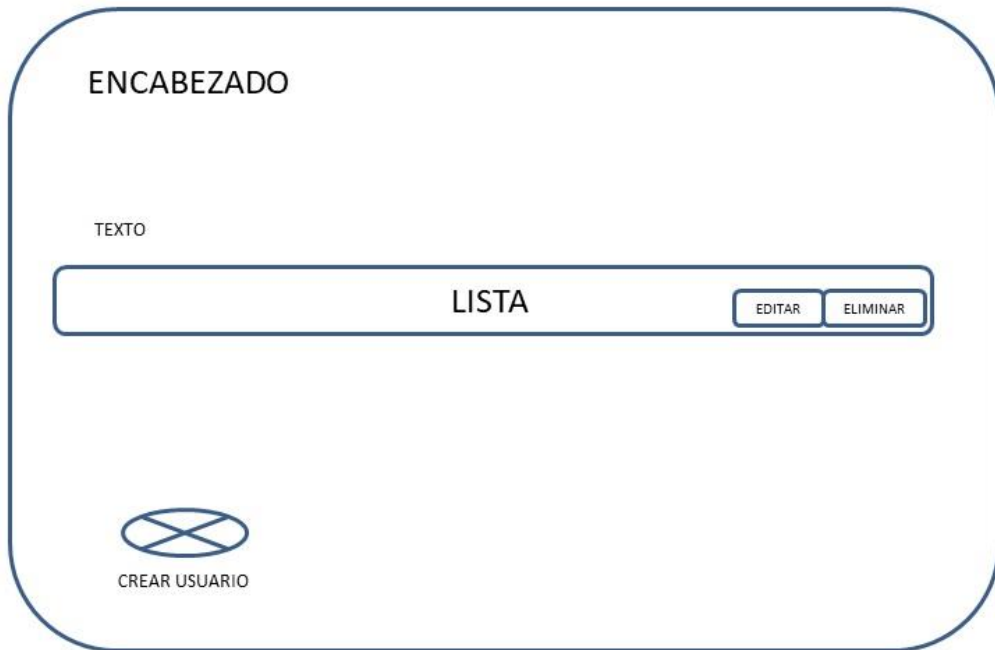


Figura 19. Diseño Usuarios.



Figura 20. Diseño Crear Usuarios.

3.3.3 Fase 3: Iteraciones

Se comienza con las primeras pruebas al sistema, de ser necesario se siguen agregando nuevas funcionalidades que el cliente requiera, se establece la arquitectura más adecuada para el sistema.

a. Arquitectura

Para determinar una de las partes de la arquitectura que mejor se acople al sistema se realizaron pruebas a los servidores web Apache y Node.JS, para comprobar su rendimiento, a continuación se muestra una serie de tablas que contienen graficas estadísticas que muestran el desempeño de cada uno de ellos.

- Pruebas en el Módulo Ítems.

Tabla 17

Tiempo de Conexión.

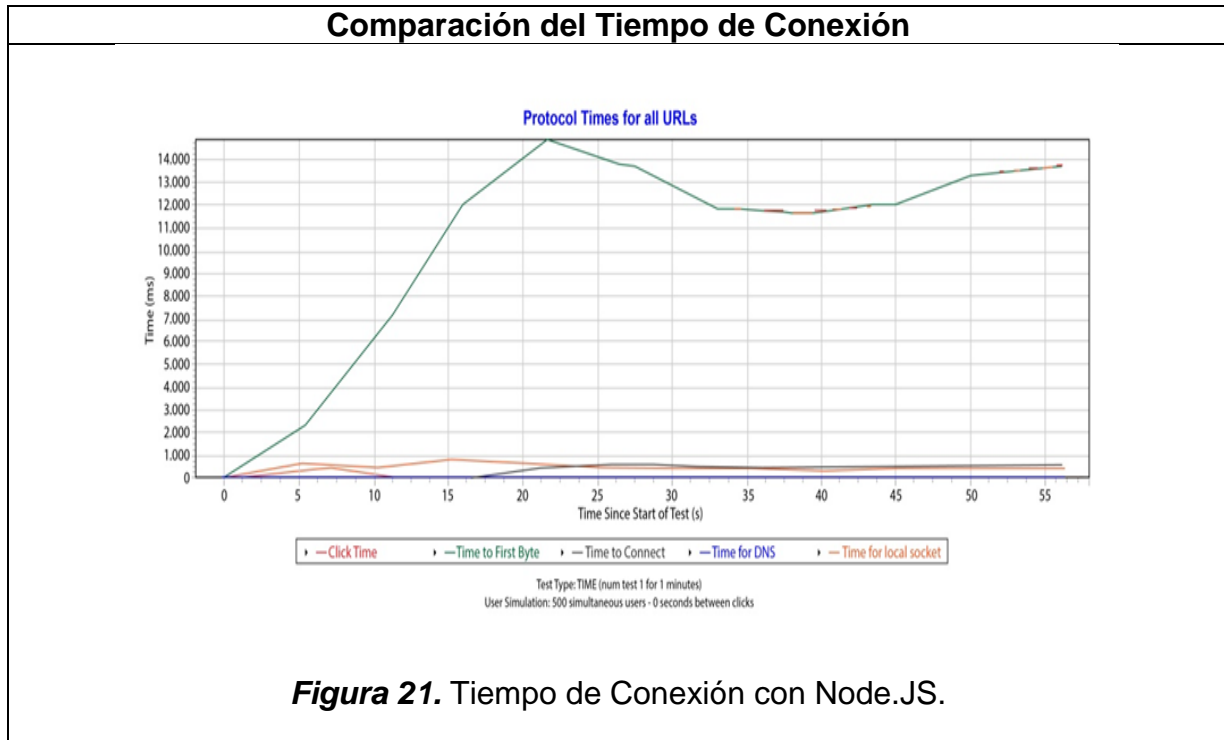
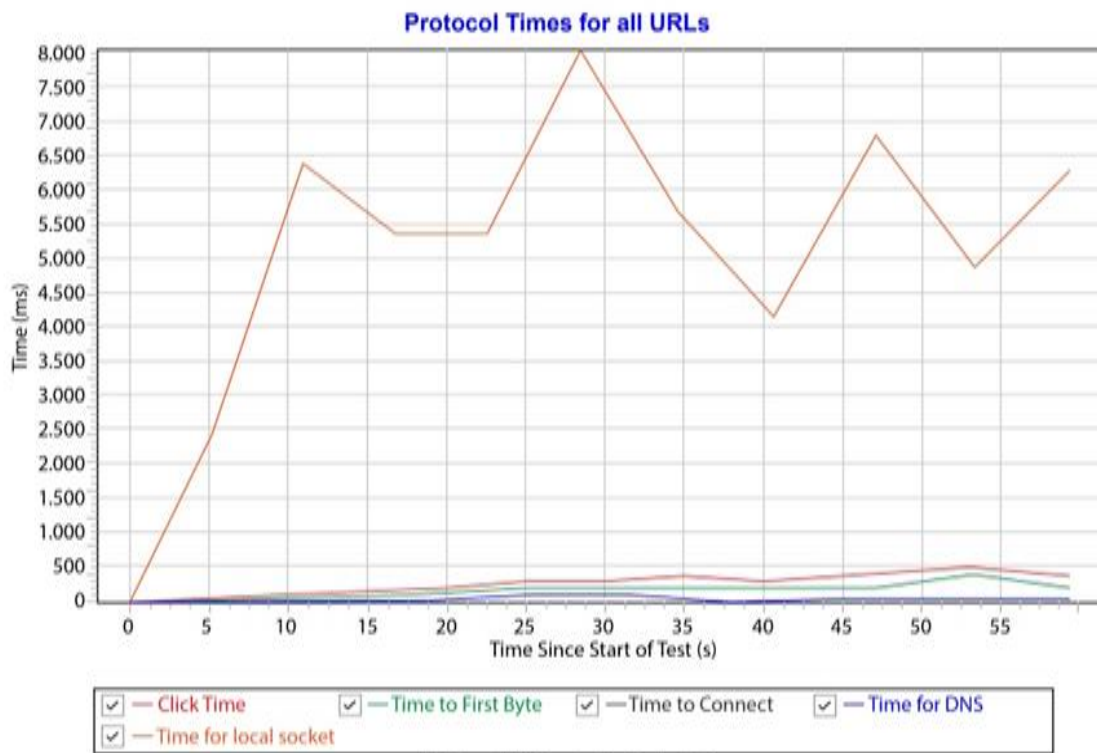


Figura 21. Tiempo de Conexión con Node.JS.



Test Type: TIME (run test for 1 minutes)
 User Simulation: 500 simultaneous users - 0 seconds between clicks

Figura 22. Tiempo de Conexión con Apache.

En la Figura 21 se observa la transición del tiempo de respuesta en el servidor web del Node.JS donde el tiempo de conexión local (Time for local socket) oscila entre 0 y 1000 ms, considerados como tiempo promedio para tener un buen servicio, cuando se ingresa un conjunto de 500 columnas/documentos a la bases de datos, mientras tanto en la Figura 22 se observa que con Apache el tiempo oscila entre los 6000 y 8000ms.

Tabla 18

Servidor y Ancho de banda del usuario.

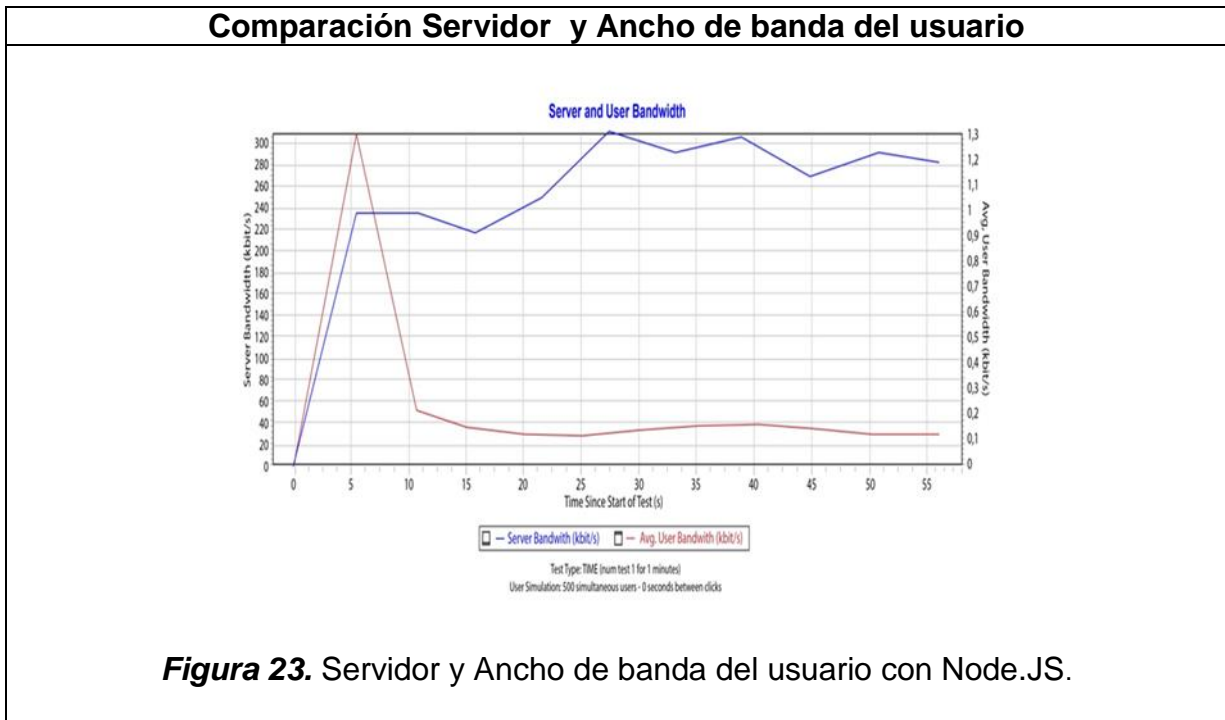


Figura 23. Servidor y Ancho de banda del usuario con Node.JS.

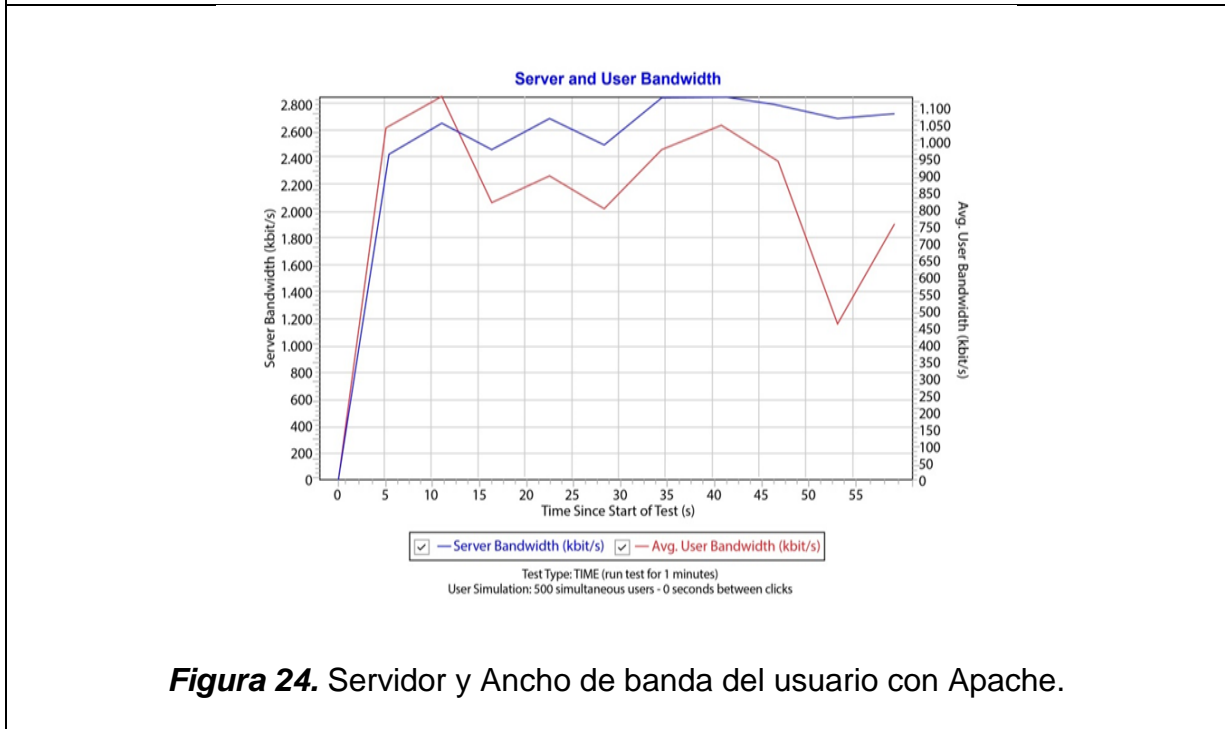
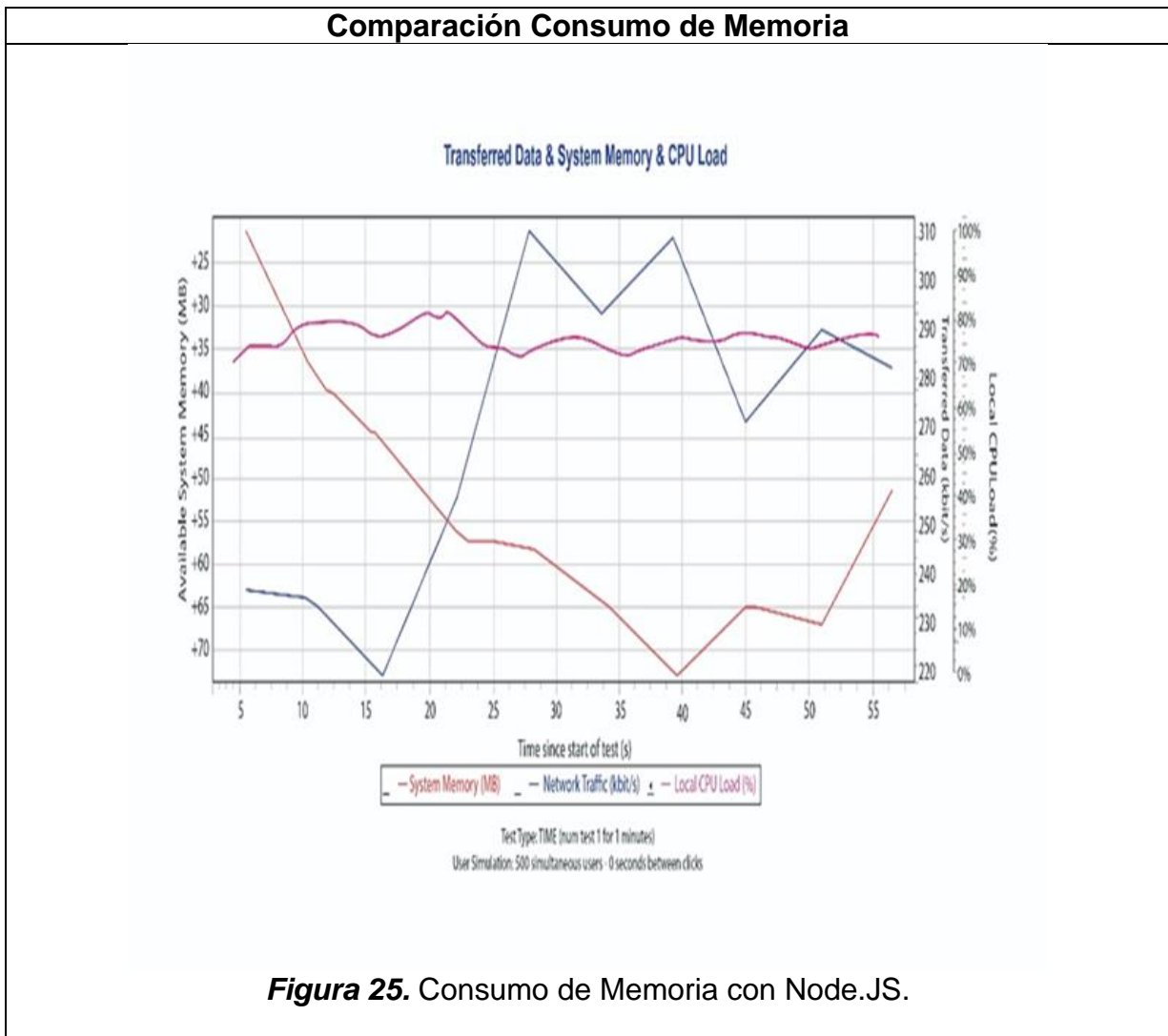


Figura 24. Servidor y Ancho de banda del usuario con Apache.

En la Figura 23 se puede apreciar que el consumo del ancho de banda (Server Bandwidth) es menor con el servidor web Node.JS que oscila entre los 200 y 300 kbits/s mientras que con Apache es mucho mayor superando los 2400 kbit/s como nos muestra la Figura 24.

Tabla 19

Consumo de Memoria.



CONTINÚA

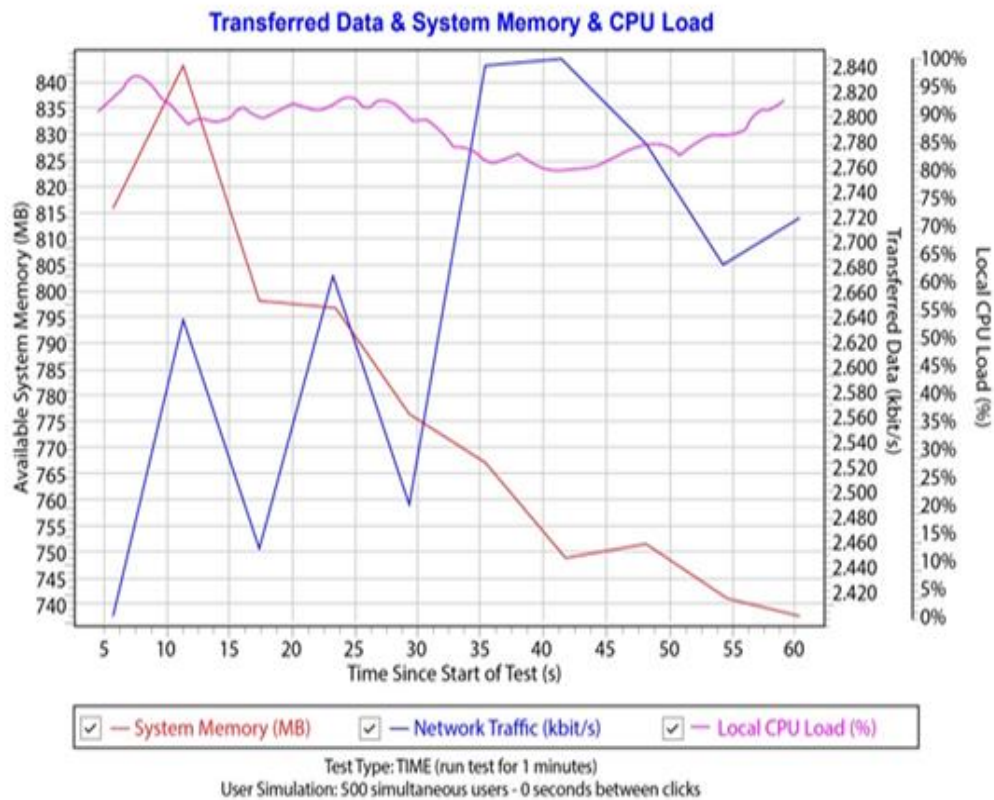


Figura 26. Consumo de Memoria con Apache.

El consumo de memoria (System Memory) con el servidor web Node.JS oscila entre 20 y 80 mb como se nos muestra en la Figura 25 mientras que con Apache es mucho mayor la cual oscila entre los 740 y 840 mb de consumo **Figura 26**, de igual manera la transferencia de datos (Network Traffic) y carga de CPU (Local CPU Load) por parte de Node.JS es menor con respecto al servidor Apache.

Tabla 20

Transferencia de Datos

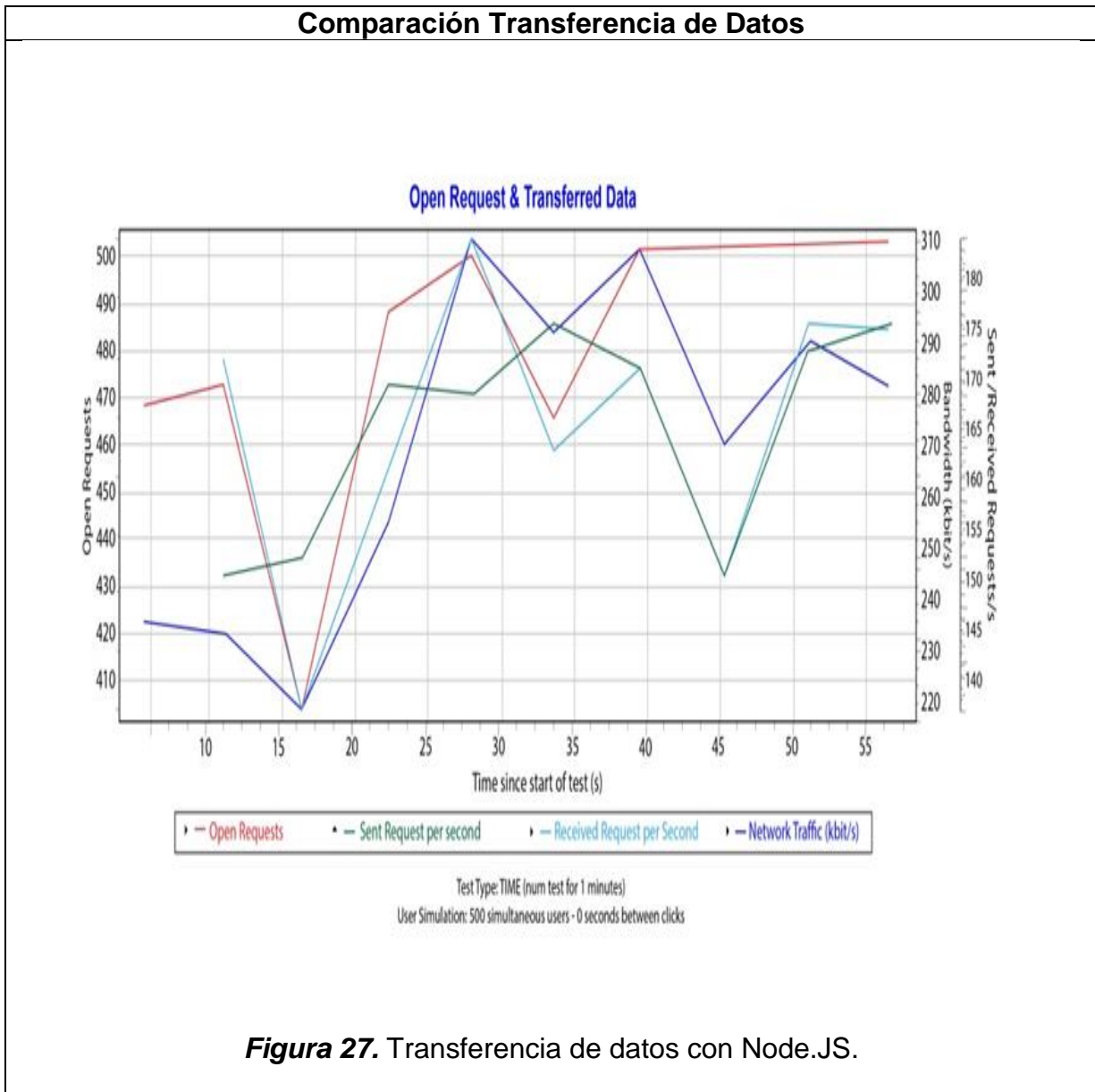


Figura 27. Transferencia de datos con Node.JS.

CONTINÚA 

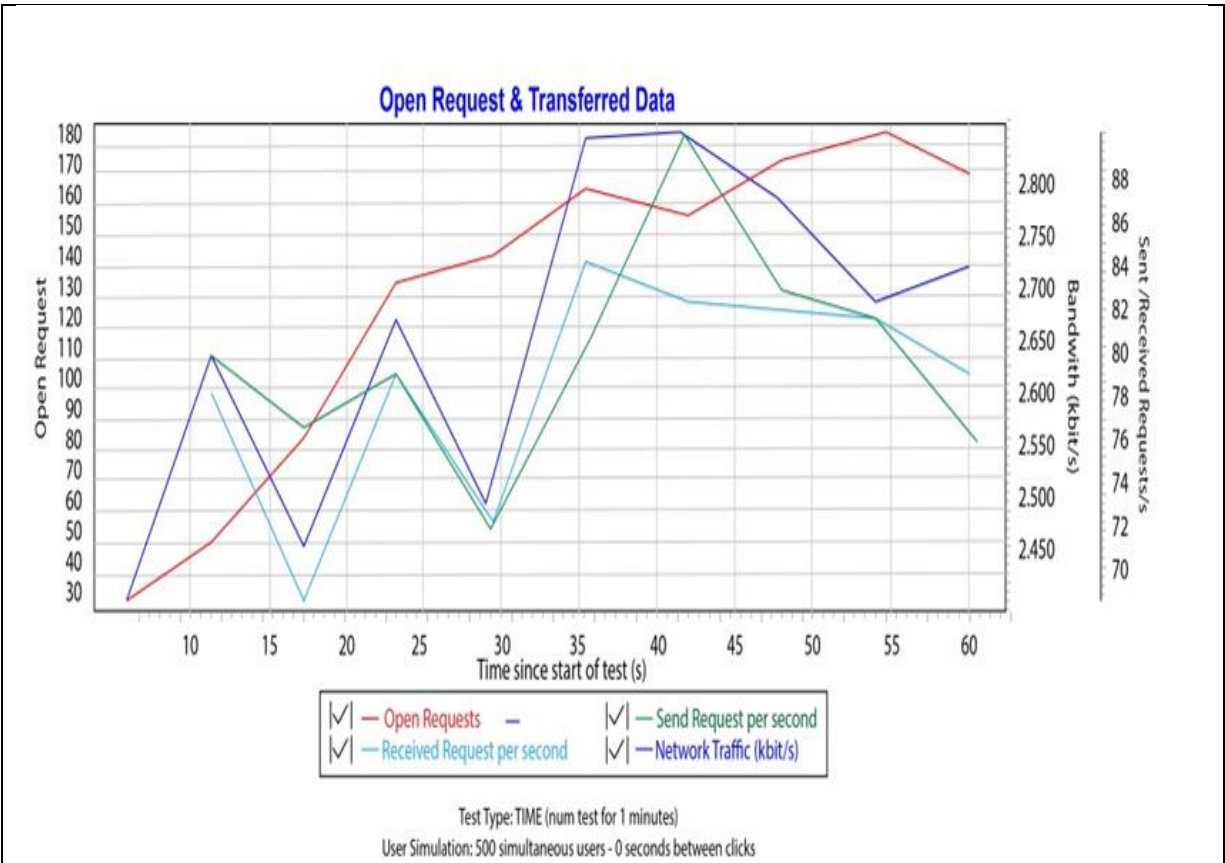


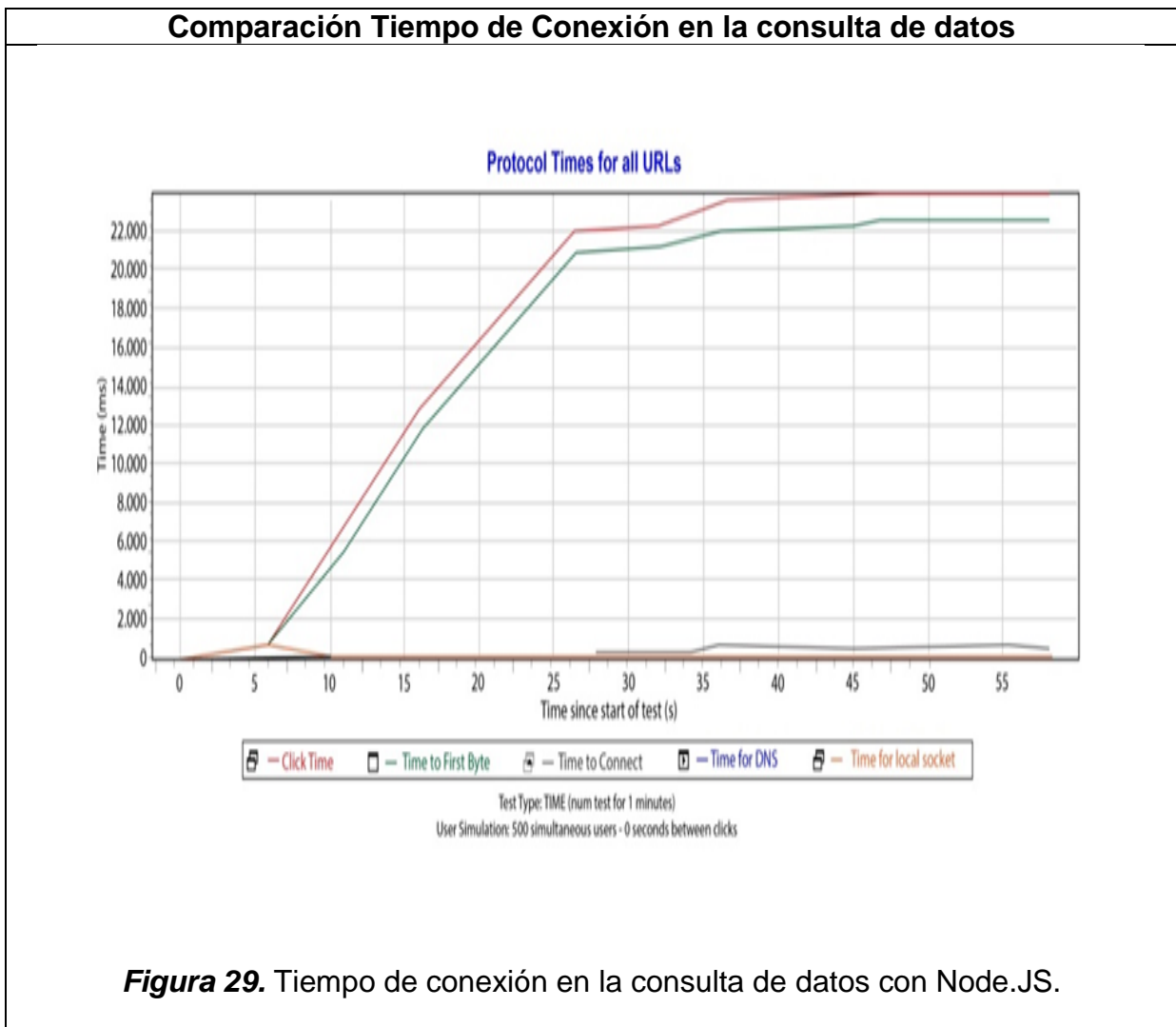
Figura 28. Transferencia de datos con Apache.

La transferencia de datos por segundo, y el tráfico en la red (Network Traffic) al enviar y recibir respuestas (Received / Sent Request) en Node.JS **Figura 27**, tomando en cuenta las solicitudes abiertas es mucho menor que en el servidor **Figura 28**.

- Pruebas en el módulo Stock Actual.

Tabla 21

Tiempo de conexión en la consulta de datos.



CONTINÚA

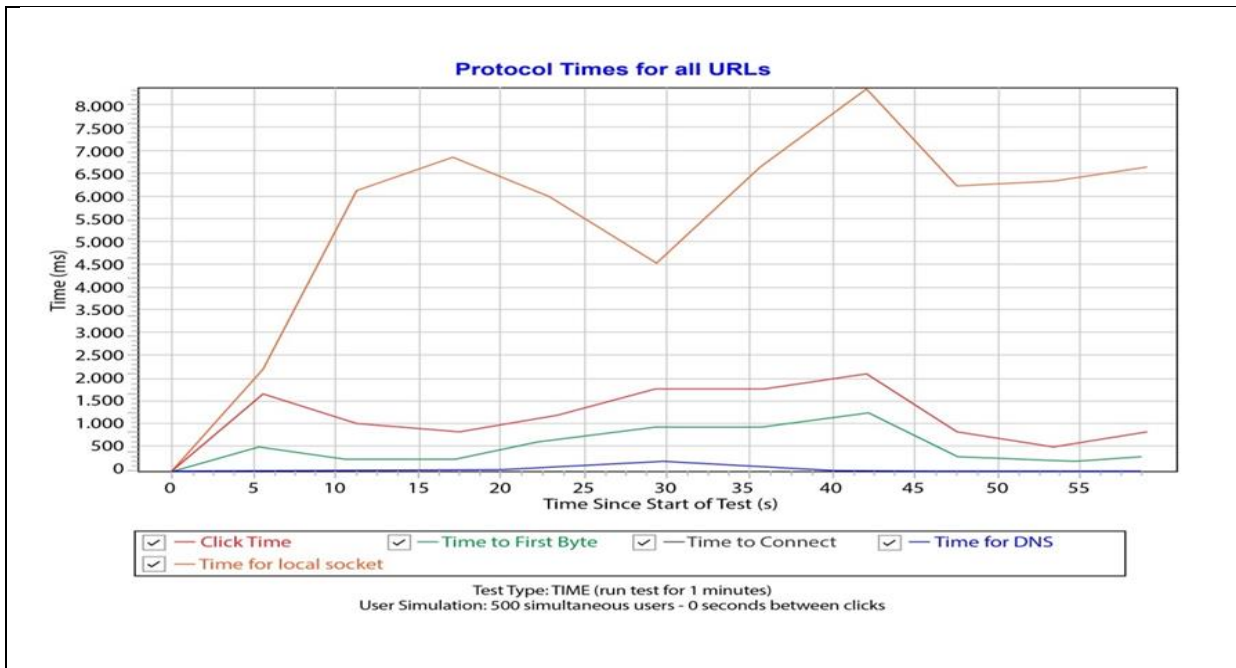


Figura 30. Tiempo de conexión en la consulta de datos con Apache

Al igual que en la función insertar la transición del tiempo de respuesta en el servidor web Node.JS en la función listar con respecto al tiempo de conexión local (Time for local socket) oscila entre 0 y 1000 ms **Figura 29**, mientras que con Apache **Figura 30** se observa que el tiempo oscila entre los 6000 y 8000ms.

Tabla 22

Servidor y Ancho de banda de usuario en la consulta de datos.

Comparación Servidor y Ancho de banda de usuario en la consulta de datos

CONTINUÁ 

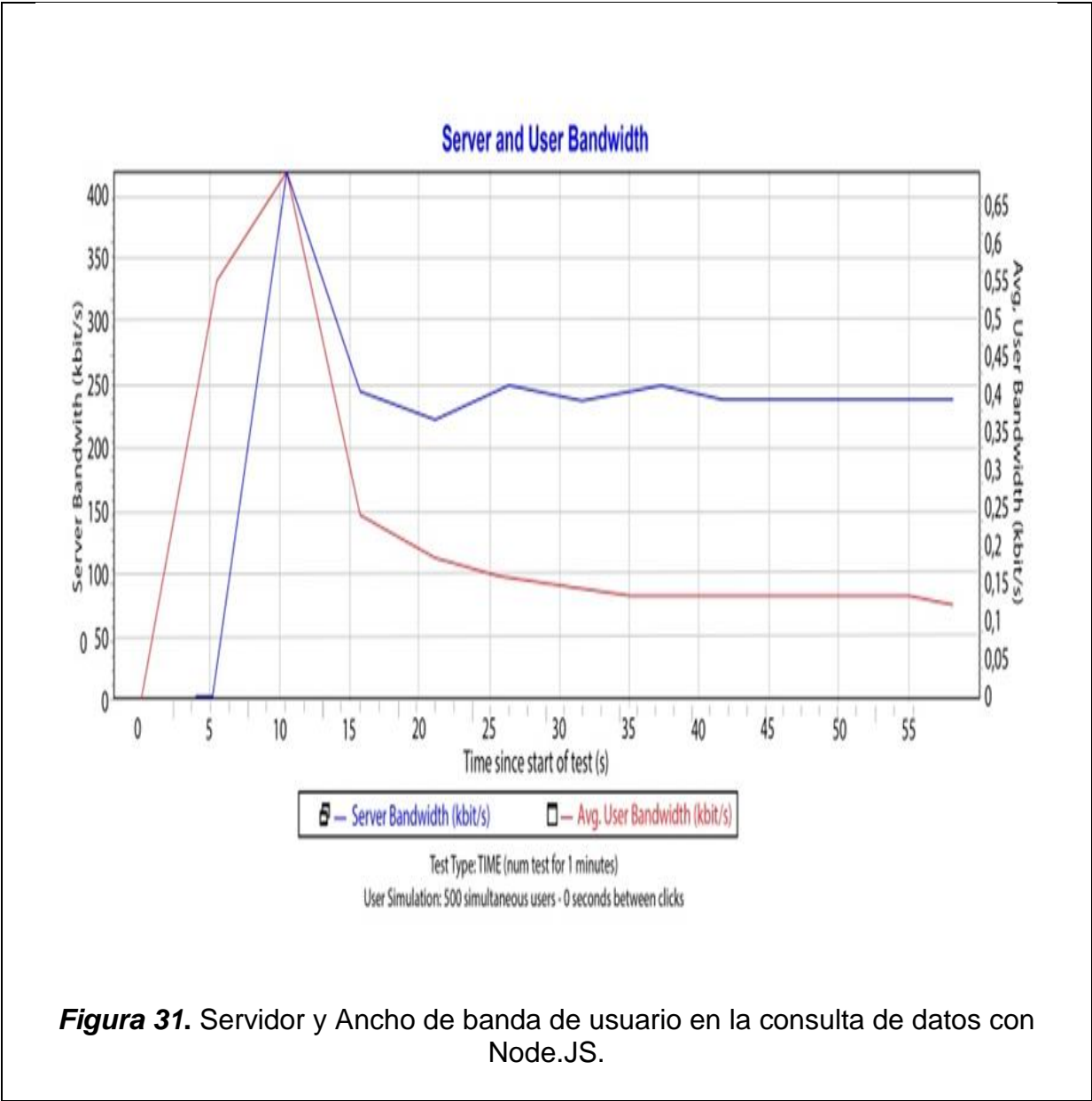


Figura 31. Servidor y Ancho de banda de usuario en la consulta de datos con Node.JS.

CONTINÚA

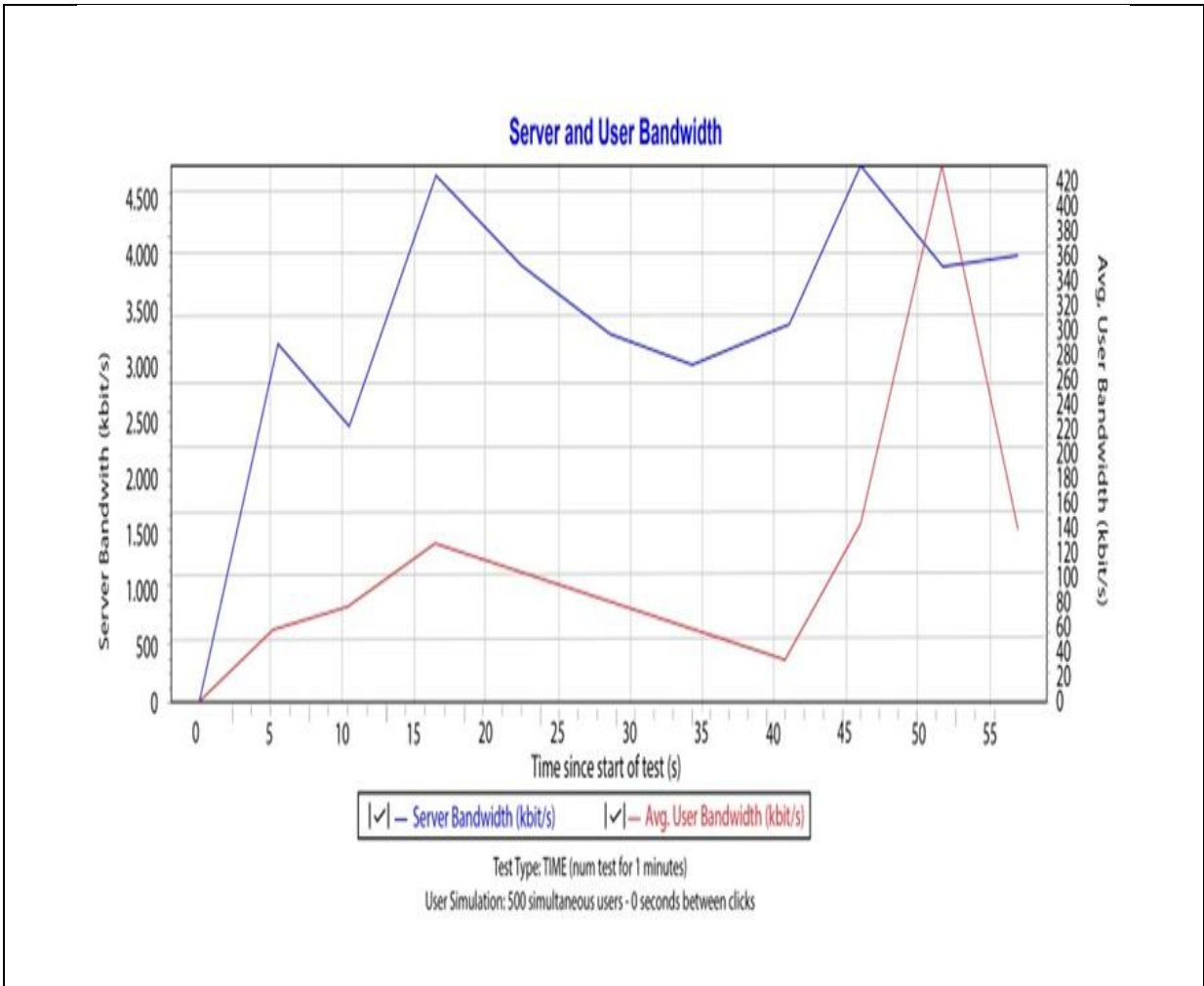
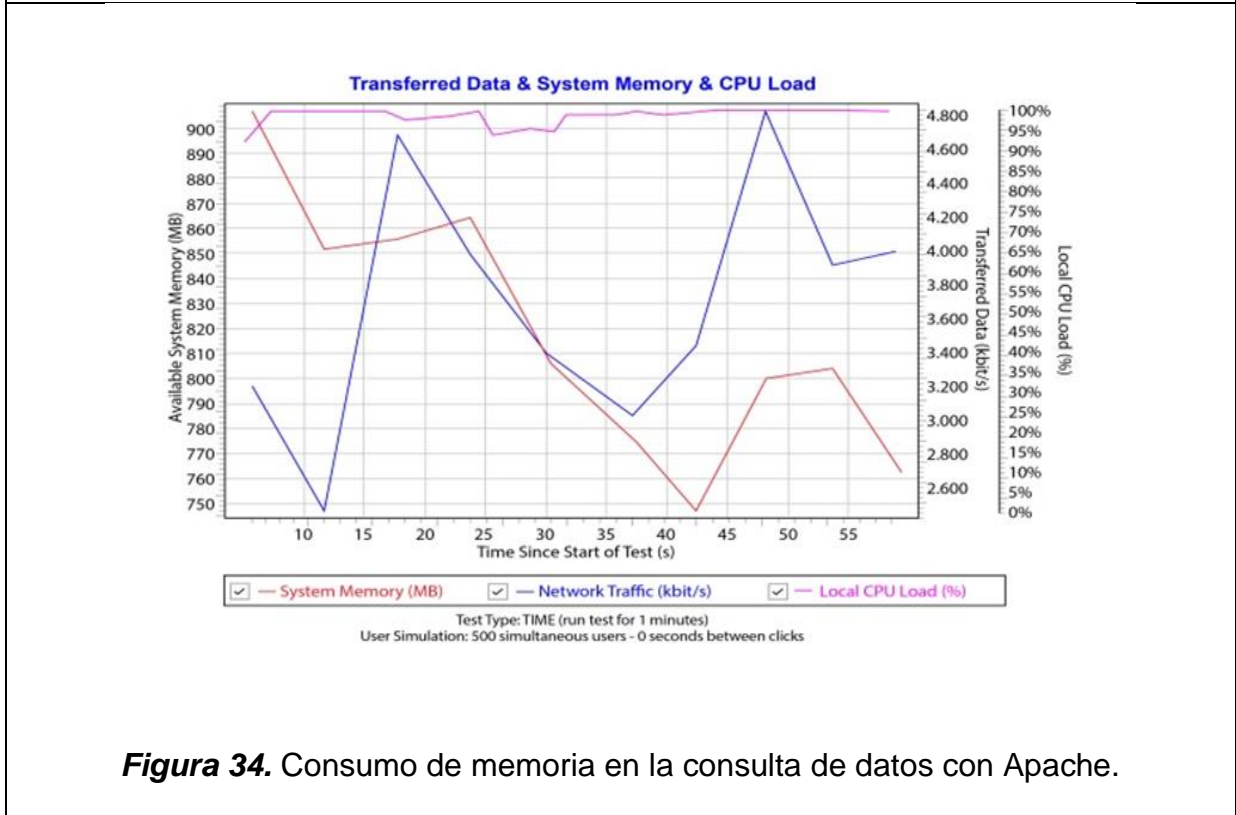
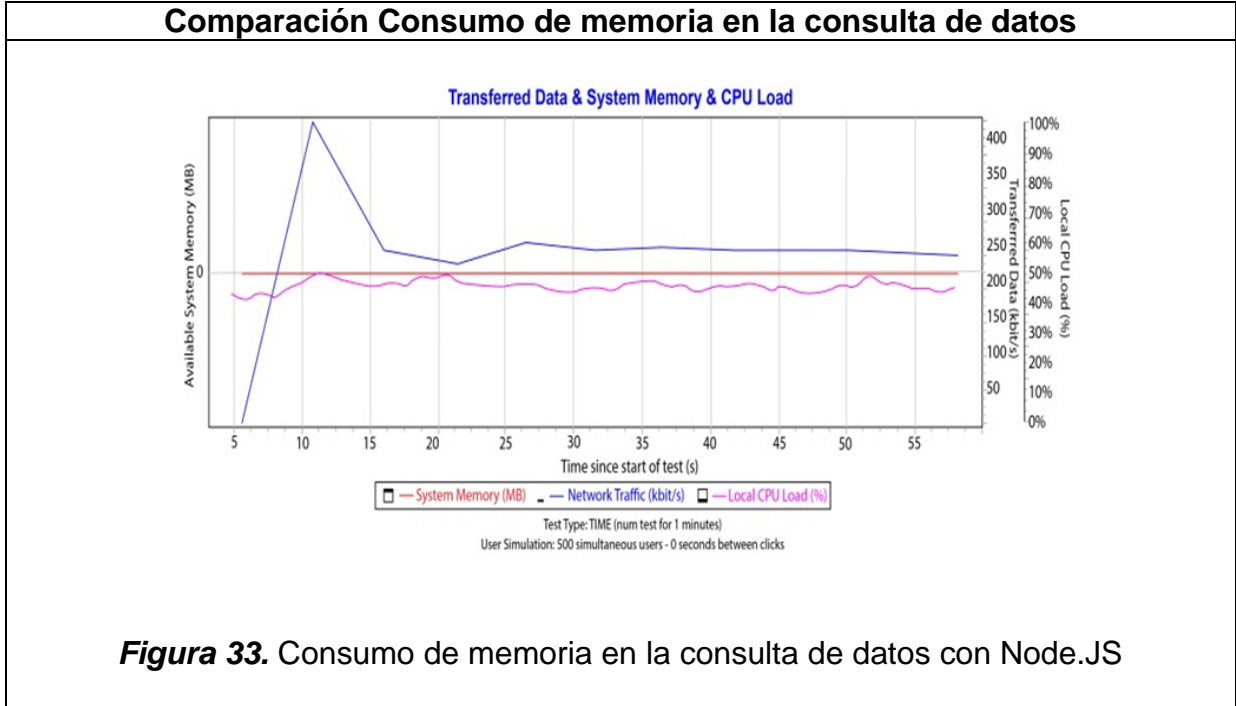


Figura 32. Servidor y Ancho de banda de usuario en la consulta de datos con Apache.

Se puede observar en la **Figura 31** que el consumo de ancho de banda (Server Bandwidth) de Node.JS es significativamente menor al del servidor Apache mostrado en la **Figura 32**, ya que Node.JS tiene la capacidad de gestionar las conexiones y solicitudes con una mayor eficiencia.

Tabla 23

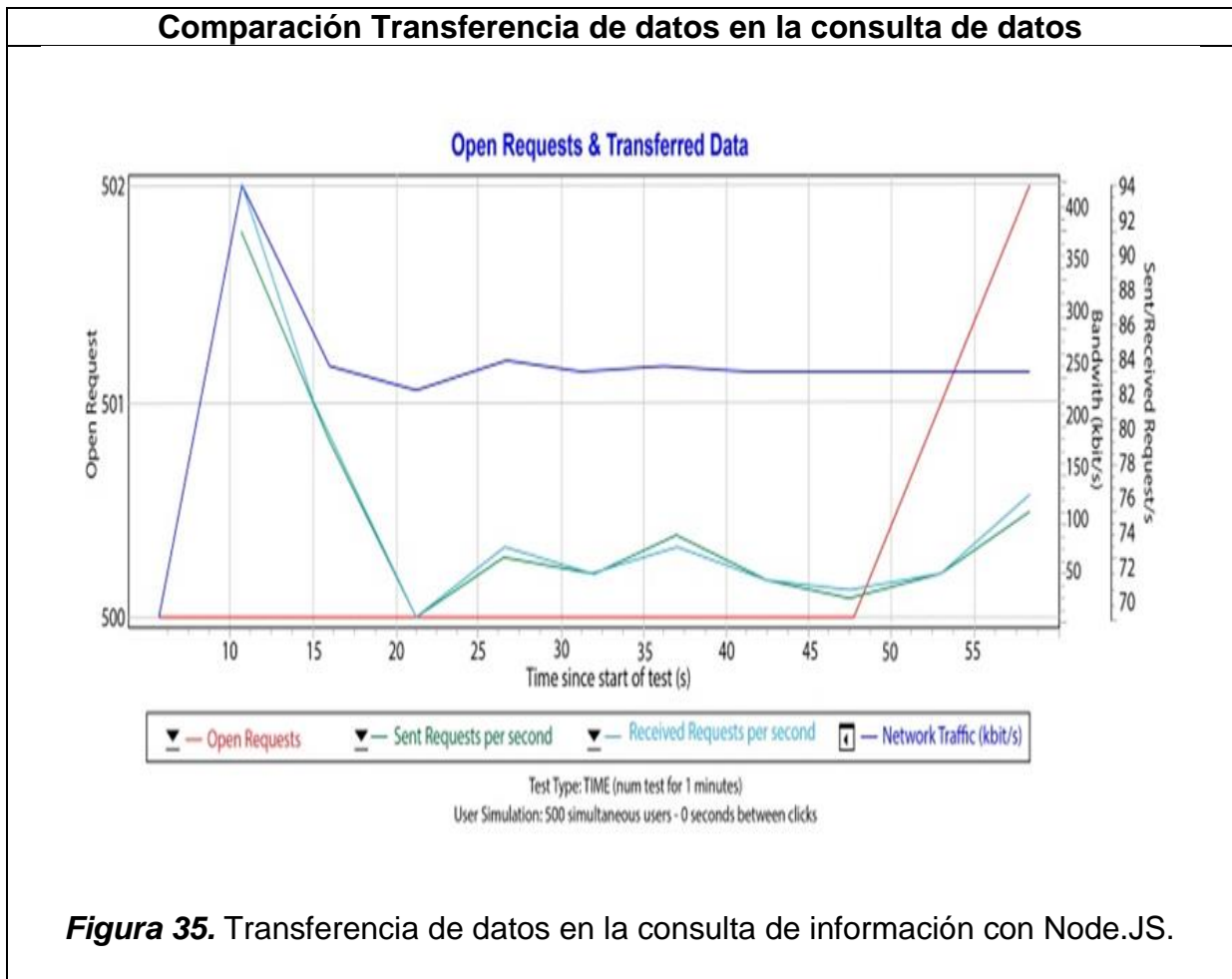
Consumo de memoria en la consulta de datos.



El servidor web Apache tiende a consumir más memoria como se puede observar en la **Figura 33** mientras que Node.JS optimiza el consumo de memoria **Figura 34**.

Tabla 24.

Transferencia de datos en la consulta de información.



CONTINÚA 

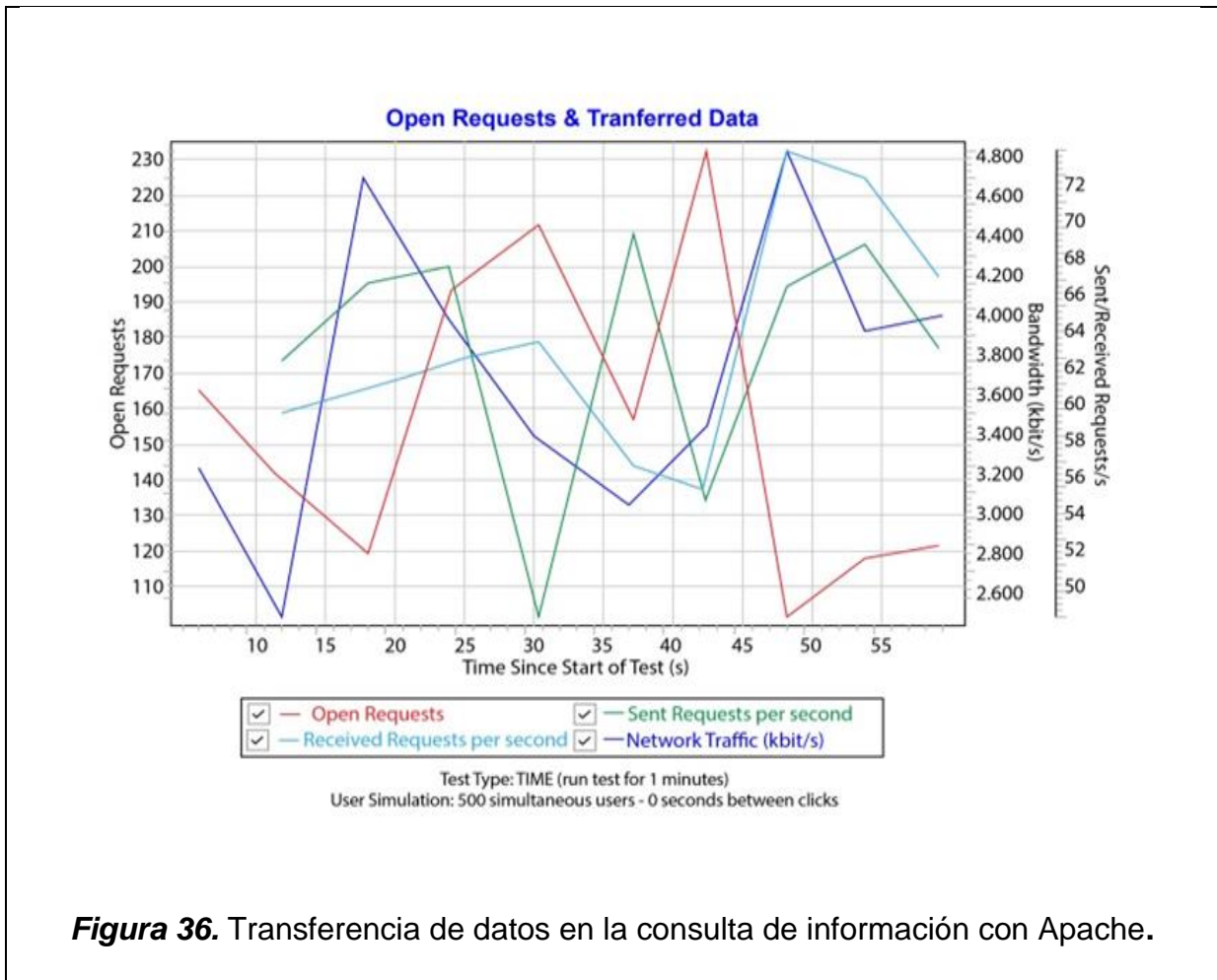


Figura 36. Transferencia de datos en la consulta de información con Apache.

La transferencia de datos con Apache mostrados en la Figura 35 nos permite visualizar que el consumo de ancho de banda (Network Traffic) es mayor, y que el envío y recepción de solicitudes (Sent/Received Request) se demora más con respecto al servidor Node.JS mostrado en la **Figura 36**, tomando en cuenta el número de solicitudes abiertas (Open Request) para cada uno de los servidores.

En base a lo expuesto anteriormente, y determinando que el servidor web Node.JS fue el que mejor desempeño tuvo en las pruebas se decide implementar el sistema

web “SIVENLAC” con dicho servidor, por lo tanto “SIVENLAC” será desarrollado utilizando:

- **Desarrollo:** Visual Studio Code
- **Servidor Web:** Node.JS
- **Base de datos:** MongoDB

3.3.4 Fase IV: Producción

Se implementa el software y se comienza a realizar pruebas adicionales de funcionamiento, se concreta los bosquejos realizados en la fase de diseño, el prototipo es revisado y de ser necesario se realizan correcciones que el usuario crea sean necesarias, a continuación se presenta una descripción de cada una de las interfaces del sistema.

- a. Descripción de Interfaces “SIVENLAC”
- b. Pantalla de Inicio

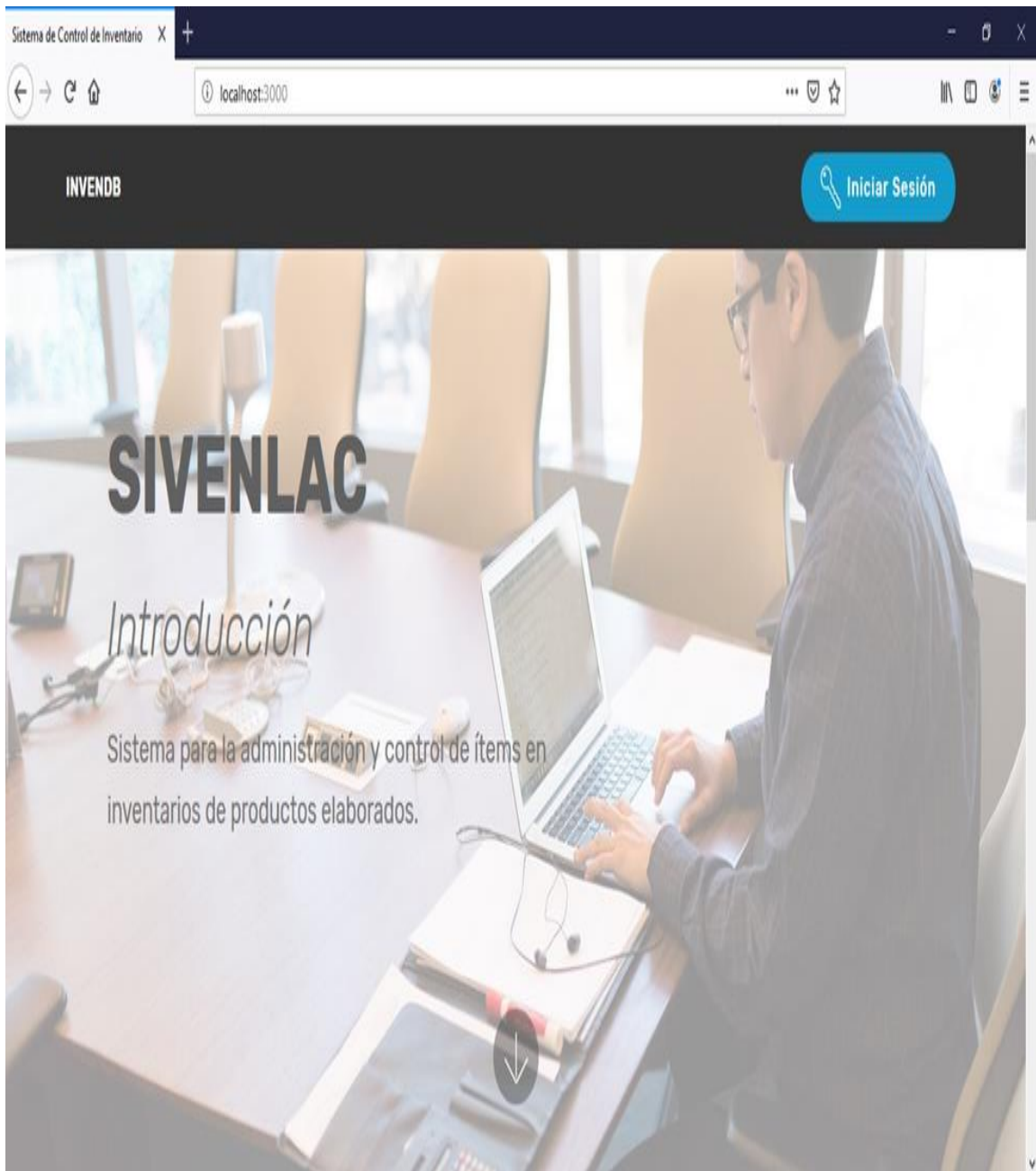


Figura 37. Pantalla de Inicio "SIVENLAC".

Pantalla principal "SIVENLAC" se muestra una pequeña introducción de lo que efectúa el sistema, en la parte superior se muestra el Botón "Iniciar Sesión" con el cual vamos a poder acceder a cada uno de los módulos del sistema.

c. Menú Principal "SIVENLAC"

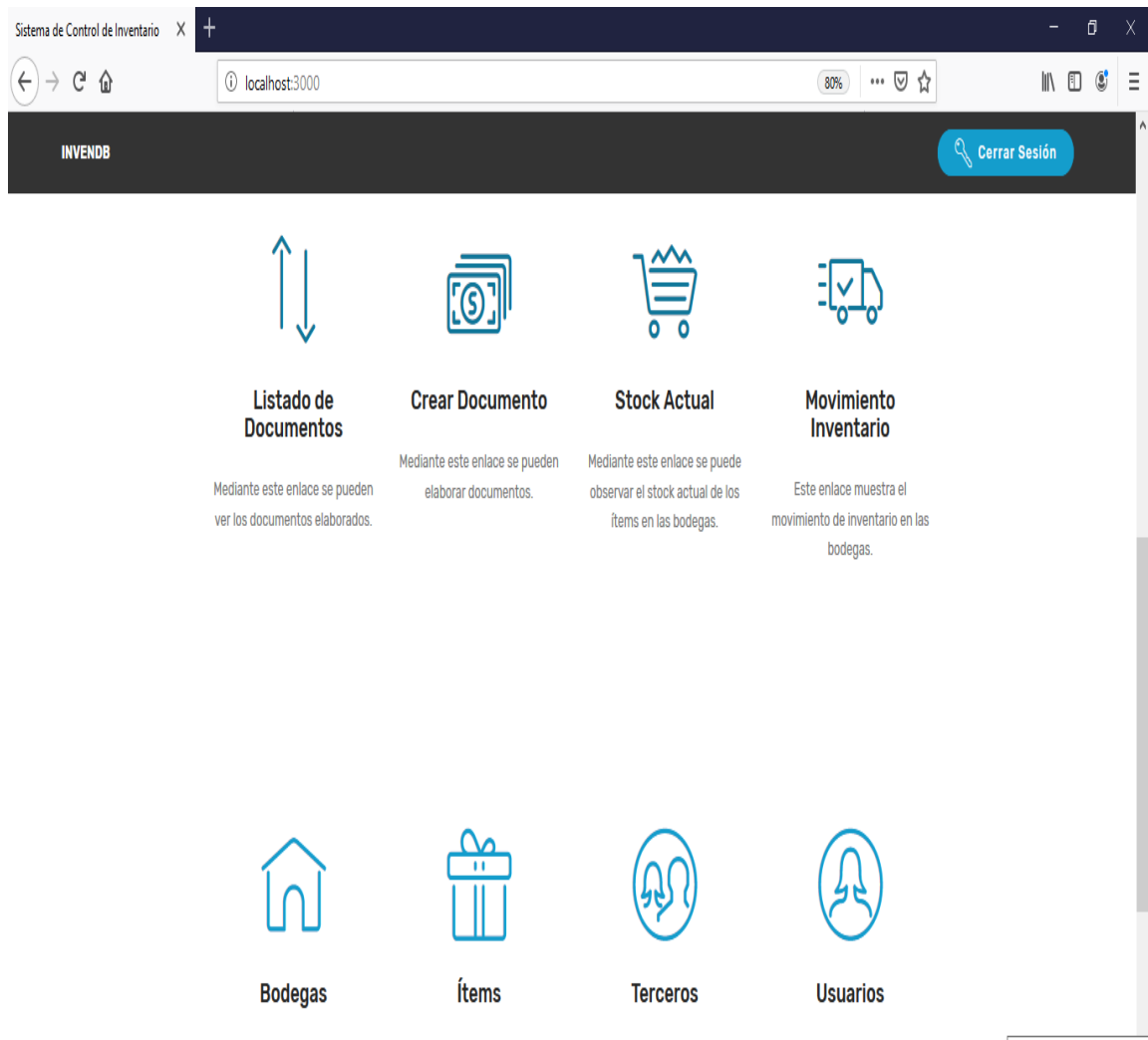


Figura 38. Menú Principal "SIVENLAC".

En el menú principal de "SIVENLAC", se muestran los 8 módulos que lo conforman, cada uno de estos serán accesibles solo a los usuarios que tenga un usuario y clave de acceso, una vez iniciada sesión se podrá interactuar con cada uno de ellos.

d. Modulo Iniciar Sesión

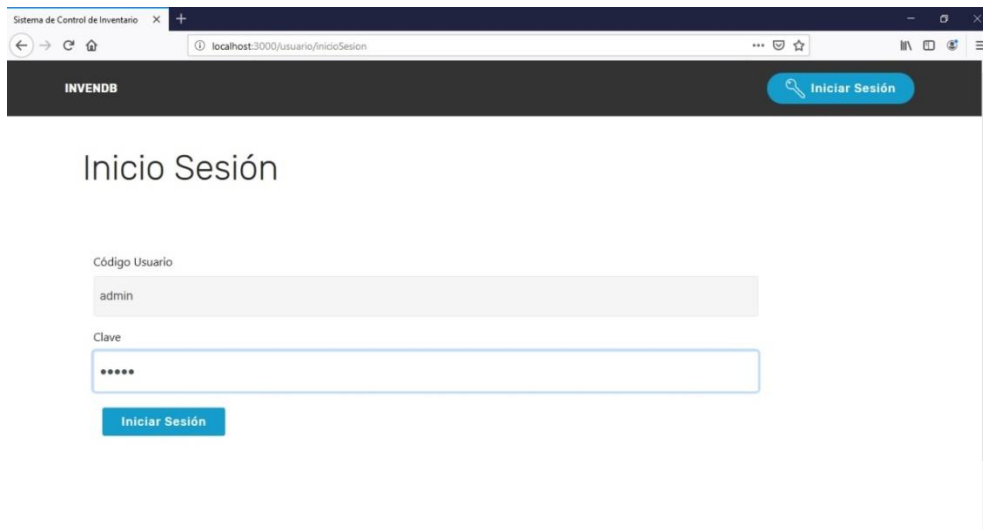


Figura 39. Iniciar Sesión "SIVENLAC".

En el módulo de Inicio de Sesión nos aparece la interface para ingresar el usuario y la clave para poder acceder a los módulos del sistema.

e. Módulo Listado de Documentos

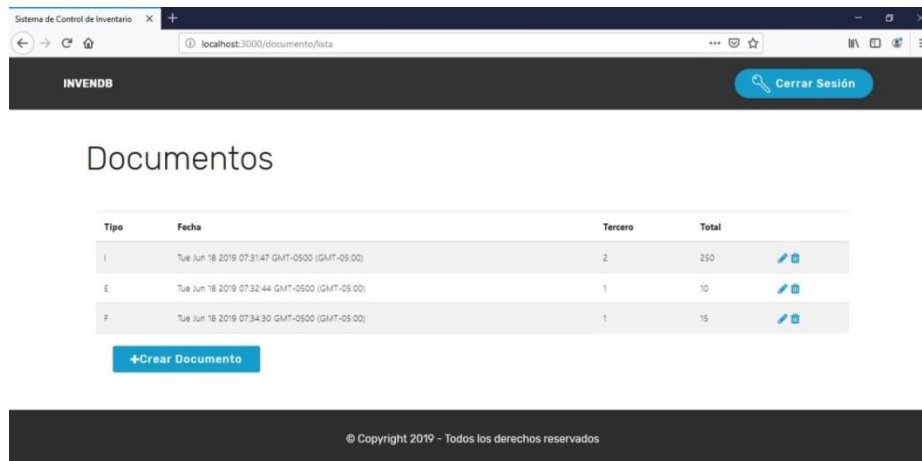


Figura 40. Modulo Listado de Documentos "SIVENLAC".

En el módulo Listado de Documentos podemos observar cada uno de los documentos que se han generado y que afectan a la stock de la empresa estos pueden ser de tres tipos Ingreso, Egreso y Factura, adicional contiene un botón “Crear Documento” con el cual se puede generar un nuevo documento.

The screenshot shows a web browser window with the URL 'localhost:3000/documento'. The page title is 'Sistema de Control de Inventario'. The header contains 'INVENDB' and a 'Cerrar Sesión' button. The main content area is titled 'Elaboración Documento'. It features three input fields: 'Orden No.' with the value '4', 'Tipo Documento' with a dropdown menu showing 'Ingreso', and 'Tercero' with a dropdown menu showing 'PASTOLAC'. A yellow button labeled '+Agregar Item' is positioned to the right of the 'Tercero' field. Below these fields is a table with the following columns: 'Código', 'Nombre', 'Cantidad', 'Precio', and 'Total'. At the bottom of the form are two blue buttons: 'Guardar' and 'Ver Ordenes'.

Figura 41. Elaboración Documento.

En la ventana Elaboración Documento podemos generar una nueva orden sea de ingreso, egreso o factura, contiene un botón “Agregar Ítem” el cual nos permitirá ingresar productos a nuestra orden se muestra a continuación:

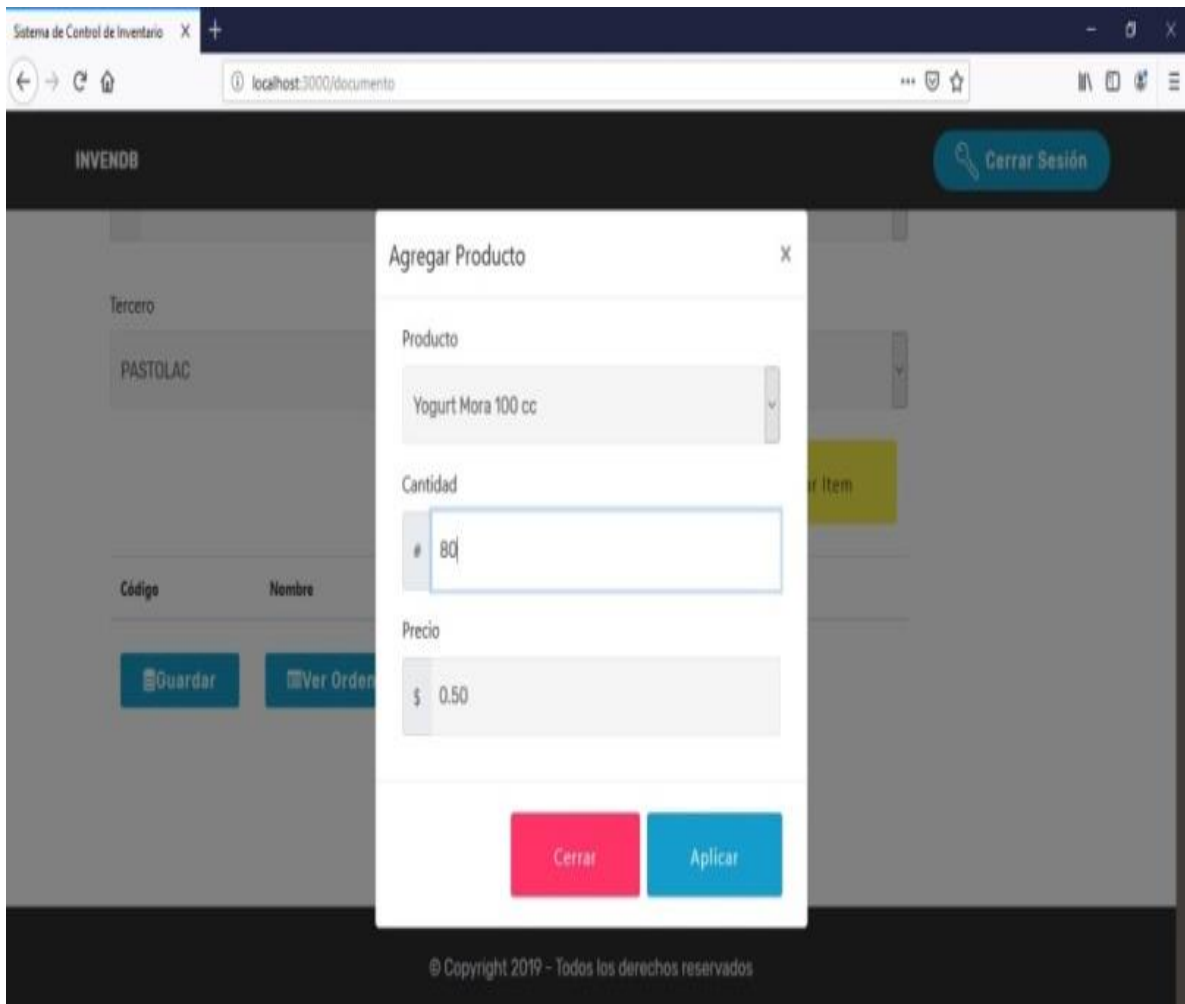


Figura 42. Agregar Producto.

En la ventana “Agregar Producto” se puede elegir una serie de productos previamente ingresados en el módulo ítems (se explicara a detalle cuando se llegue al apartado del Módulo Ítems), una cantidad y el precio del producto, contiene el botón “Cerrar” el cual cerrara la ventana sin realizar cambios en la orden y el botón “Aplicar” el cual ingresara el producto a la orden, se muestra a continuación:

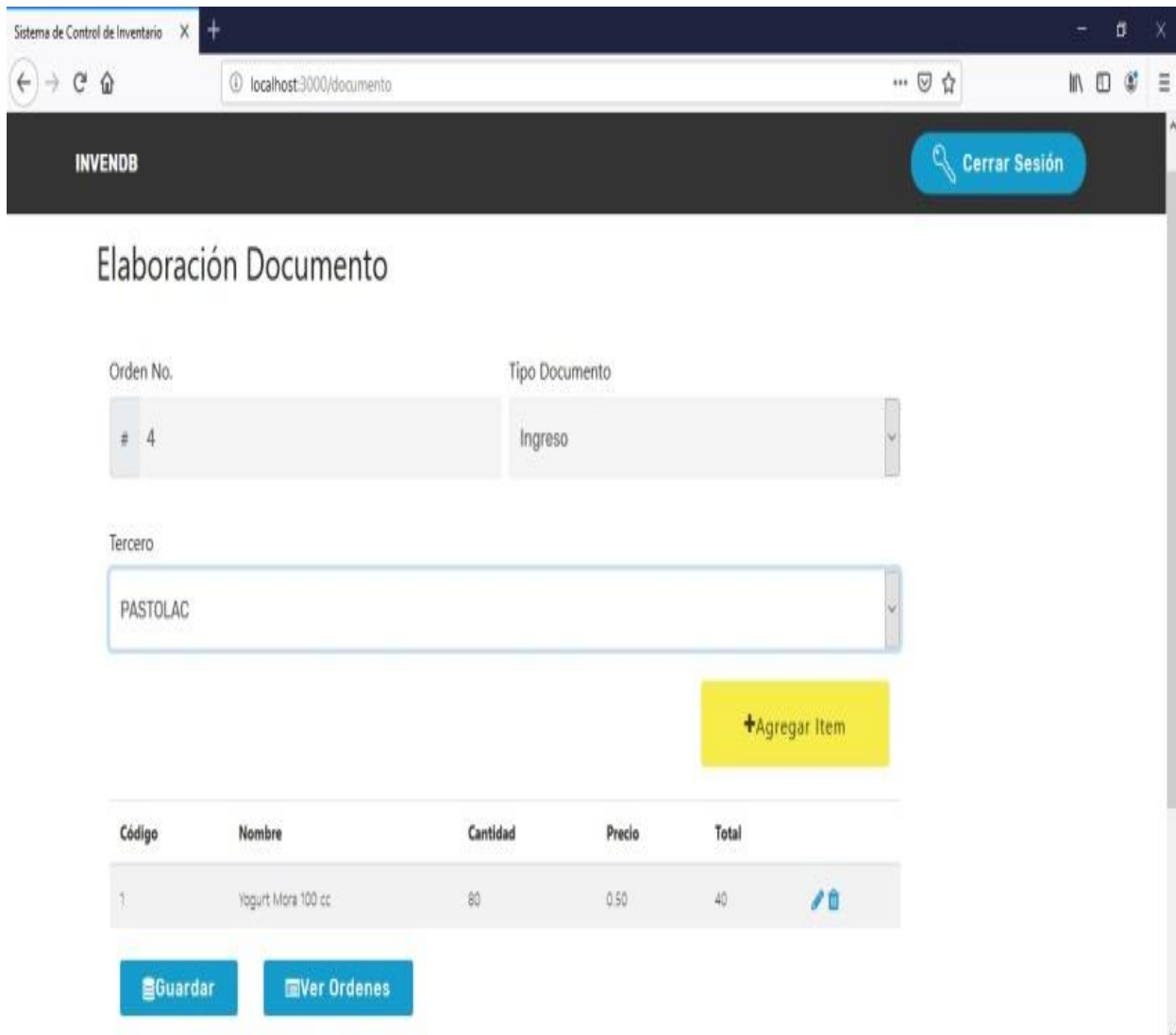


Figura 43. Elaboración Documento 1.

Una vez ingresados todos los datos de la orden; la ventana Elaboración Documento contiene un botón “Guardar” el mismo que guardara la orden, y el botón “Ver Ordenes” el cual nos dirigirá al Módulo Listado Documentos *Figura 40* para poder visualizar las ordenes elaboradas.

f. Módulo Stock Actual

En el módulo Stock Actual vamos a poder ver el estado actual de los productos en las bodegas.

INVENDB [Cerrar Sesión](#)

Stock Actual

Código Bodega	Código Item	Total
1	3	200
1	2	1000
1	1	718

© Copyright 2019 - Todos los derechos reservados

Figura 44. Modulo Stock Actual “SIVENLAC”.

g. Módulo Bodegas

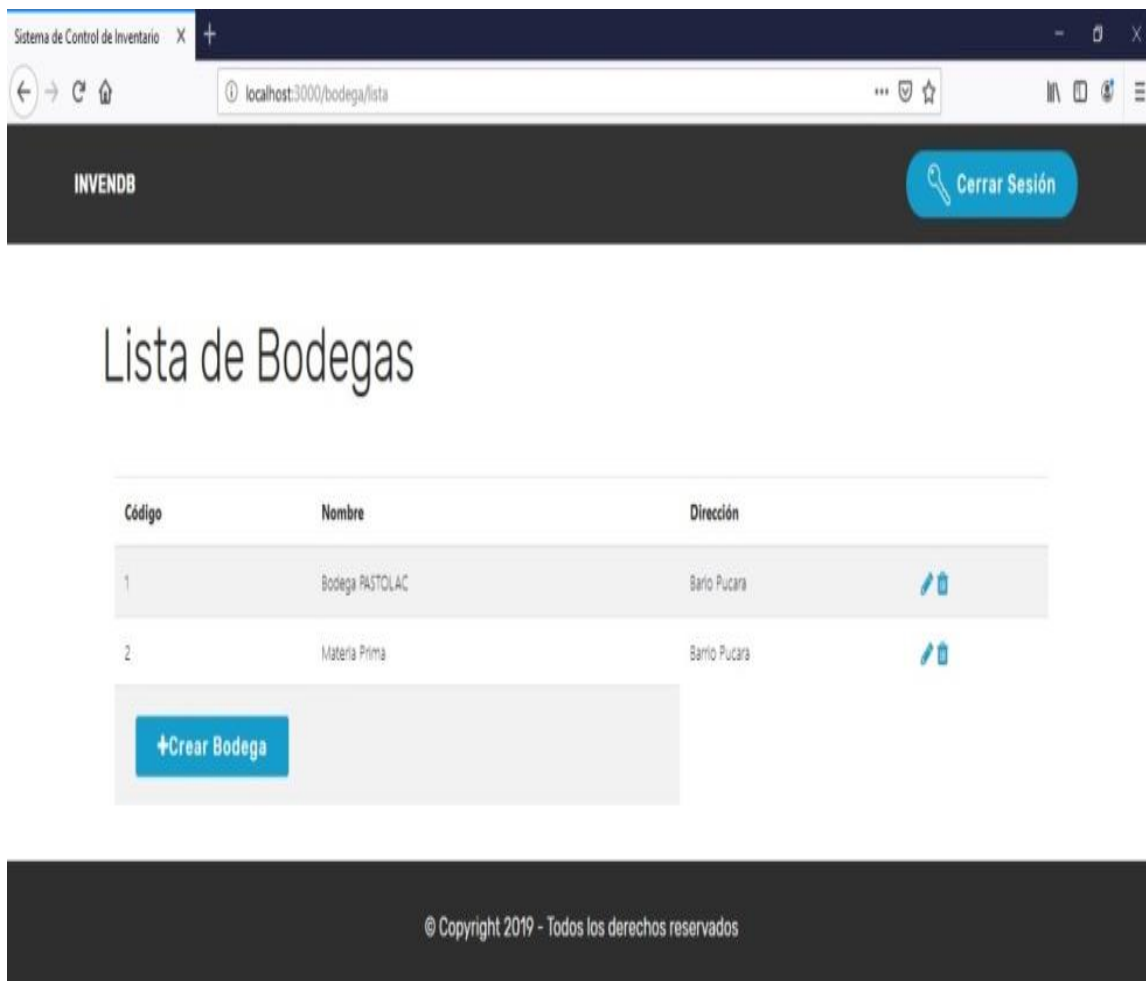


Figura 45. Modulo Bodegas "SIVENLAC".

En el módulo Bodegas podemos crear y visualizar las bodegas que la empresa ha creado con el fin de almacenar el stock de los productos terminados, bienes, materia prima, etc.

Contiene el botón "Crear Bodega" el cual permitirá crear una bodega nueva dentro del sistema de control se muestra a continuación:

Sistema de Control de Inventario X +

localhost:3000/bodega

INVENDB Cerrar Sesión

Creación Bodega

Código

Nombre

Dirección

Enviar Ver Todos







Figura 46. Creación Bodega.

El botón “Enviar” guardara la nueva bodega, y el botón “Ver Todos” mostrara todas las bodegas creadas Figura 45.

h. Módulo Ítems



Lista de Items

Código	Nombre	Grava I.V.A.	
1	Yogurt Mora 100 cc.	true	
2	Yogurt Mora 250 cc.	true	
3	Yogurt Mora 500 cc.	true	
4	Yogurt Mora 1000 cc.	true	
5	Yogurt Mora 2000 cc.	true	
6	Yogurt Mora 4000 cc.	true	

[+Crear Item](#)

Figura 47. Módulo Ítems "SIVENLAC".

En el módulo ítems podemos crear y visualizar los productos que la empresa ha creado con el fin de almacenar el stock de los productos terminados, bienes, materia prima, etc.

Contiene el botón “Crear Ítem” el cual permitirá crear un producto, bien, materia prima, etc., que serán controlados por “SIVENLAC”, se muestra a continuación:

The screenshot shows a web browser window with the title "Sistema de Control de Inventario". The address bar displays "localhost:3000/item". The page header is dark with the text "INVENDB" on the left and a blue "Cerrar Sesión" button on the right. The main content area is titled "Creación Item". Below the title, there are three input fields: "Código" containing the number "7", "Nombre" containing "Yogurt Durazno 100 cc", and "Grava I.V.A." with a checked checkbox. At the bottom of the form, there are two blue buttons: "Enviar" and "Ver Todos".

Figura 48. Creación Ítem.

El botón “Enviar” guardara el nuevo ítem, y el botón “Ver Todos” mostrara todas los ítems creados *Figura 47*.

i. Módulo Terceros.

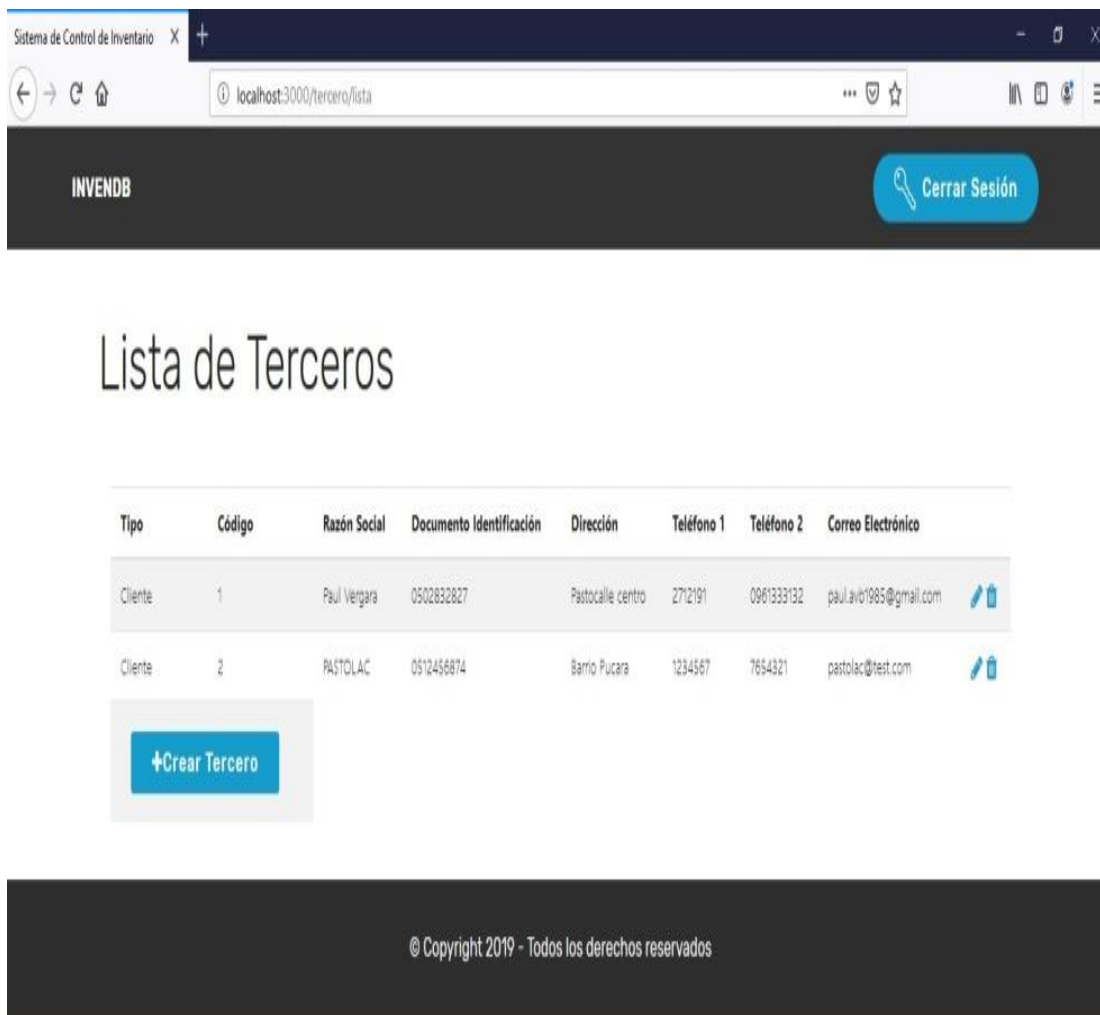


Figura 49. Modulo Terceros "SIVENLAC"

Dentro de “SIVENLAC” se considera como Terceros a las Razones Sociales que interactúan de forma directa con el stock, estos pueden ser un cliente o un proveedor, dentro del Módulo Terceros se puede observar los clientes y proveedores que la empresa tenga ingresados

adicional contiene el botón “Crear Tercero” el cual nos permitirá el ingreso de un nuevo tercero al sistema, se muestra a continuación:

The screenshot shows a web browser window with the title "Sistema de Control de Inventario". The address bar displays "localhost:3000/tercero". The page content includes a header with "INVENTARIO" and a "Cerrar Sesión" button. The main heading is "Creación Tercero". The form contains the following fields and values:

Field	Value
Código	3
Tipo	Proveedor
Razón Social	Luga Mens
Documento Identificación	569821476
Dirección	Barrio Otullo
Teléfono 1	258749
Teléfono 2	962654
Correo Electrónico	lumen@luz.com

At the bottom of the form are two buttons: "Enviar" and "Ver Todos".

Figura 50. Creación Tercero.

El botón “Enviar” guardara el nuevo tercero, y el botón “Ver Todos” mostrara todos los terceros creados Figura 49.

j. Módulo Usuarios

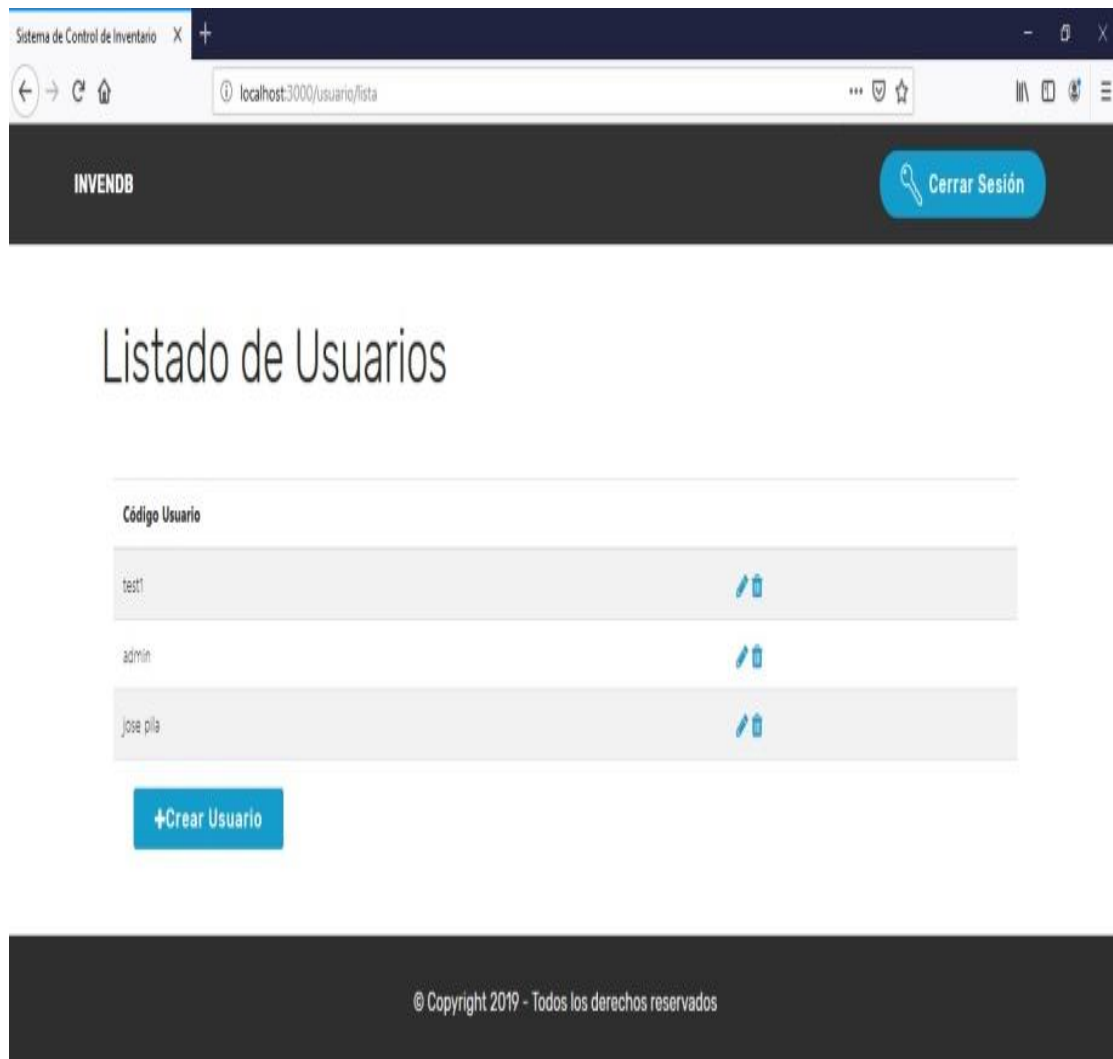


Figura 51. Modulo Usuarios "SIVENLAC".

En el módulo Usuarios podemos crear y visualizar los usuarios que la empresa ha creado con el fin de tener acceso al sistema.

Contiene el botón “Crear Usuario” el cual permitirá crear un usuario nuevo y darle acceso al sistema se muestra a continuación:

Sistema de Control de Inventario X +

localhost:3000/usuario

INVENDB Cerrar Sesión

Creación Usuario

Código Usuario

marlene caiza

Clave

Enviar Ver Todos

Figura 52. Creación Usuario.

El botón “Enviar” guardara el nuevo usuario, y el botón “Ver Todos” mostrara todos los usuarios creados Figura 51.

3.3.5 Fase V: Mantenimiento

Se realizan varias pruebas adicionales al sistema, el Cliente interactúa con el sistema y se realizan algunos cambios en los módulos.

3.3.6 Fase VI: Muerte del Proyecto

Se considera como muerte del proyecto cuando el cliente no tiene más cambios que realizar al sistema, sin embargo en la arquitectura con la que está implementado “SIVENLAC” no aplica ya que se trata de una arquitectura escalable la misma que permite realizar cambios al sistema según nuevas necesidades del cliente.

CAPÍTULO IV

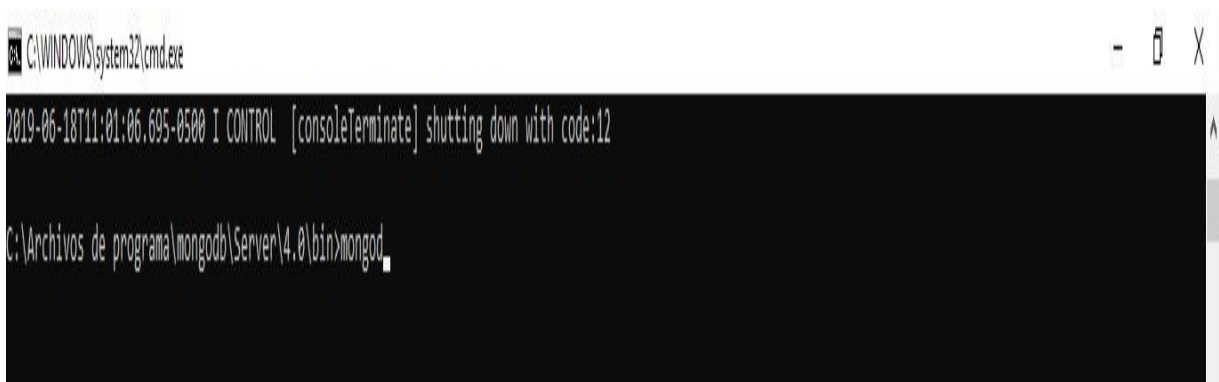
PRUEBAS DE VALIDACIÓN

3.4 Introducción

En el presente capítulo se realizan las siguientes pruebas de validación de “SIVENLAC”.

3.4.1 Inicio del servidor de la base de datos MongoDB.

Se inicia el servidor de la base de datos MongoDB utilizando el comando “Mongod” dentro de la ventana de comandos de Windows en el directorio donde se encuentra instalada la base de datos.



```
C:\WINDOWS\system32\cmd.exe
2019-06-18T11:01:06.695-0500 I CONTROL [consoleTerminate] shutting down with code:12
C:\Archivos de programa\mongodb\Server\4.0\bin>mongod
```

Figura 53. Inicio Servidor MongoDB.

Resultado:

```

C:\WINDOWS\system32\cmd.exe - mongod
2019-06-26T18:55:17.837-0700 I CONTROL [initandlisten] MongoDB starting : pid=6268 port=27017 dbpath=C:\data\db\ 64-bit host=DESKTOP-J5F5CPI
2019-06-26T18:55:17.837-0700 I CONTROL [initandlisten] targetMinOS: Windows 7/Windows Server 2008 R2
2019-06-26T18:55:17.837-0700 I CONTROL [initandlisten] db version v4.0.8
2019-06-26T18:55:17.838-0700 I CONTROL [initandlisten] git version: 9b00696ed75f65e1ebc8d635593bed79b290cfbb
2019-06-26T18:55:17.838-0700 I CONTROL [initandlisten] allocator: tcmalloc
2019-06-26T18:55:17.838-0700 I CONTROL [initandlisten] modules: none
2019-06-26T18:55:17.838-0700 I CONTROL [initandlisten] build environment:
2019-06-26T18:55:17.838-0700 I CONTROL [initandlisten]   distmod: 2008plus-ssl
2019-06-26T18:55:17.838-0700 I CONTROL [initandlisten]   distarch: x86_64
2019-06-26T18:55:17.838-0700 I CONTROL [initandlisten]   target_arch: x86_64
2019-06-26T18:55:17.839-0700 I CONTROL [initandlisten] options: {}
2019-06-26T18:55:17.843-0700 I STORAGE [initandlisten] Detected data files in C:\data\db\ created by the 'wiredTiger' storage engine, so setting the active storage engine to 'wiredTiger'.
2019-06-26T18:55:17.846-0700 I STORAGE [initandlisten] wiredtiger_open config: create,cache_size=2504M,session_max=20000,eviction=(threads_min=4,threads_max=4),config_base=false,statistics=(fast),log=(enabled=true,archive=true,path=journal,compressor=snappy),file_manager=(close_idle_time=100000),statistics_log=(wait=0),verbose=(recovery_progress),
2019-06-26T18:55:18.417-0700 I STORAGE [initandlisten] WiredTiger message [1561600518:416872][6268:140710526012496], txn-recover: Main recovery loop: starting at 14/86912 to 15/256
2019-06-26T18:55:18.777-0700 I STORAGE [initandlisten] WiredTiger message [1561600518:776652][6268:140710526012496], txn-recover: Recovering log 14 through 15
2019-06-26T18:55:18.991-0700 I STORAGE [initandlisten] WiredTiger message [1561600518:991643][6268:140710526012496], txn-recover: Recovering log 15 through 15
2019-06-26T18:55:19.163-0700 I STORAGE [initandlisten] WiredTiger message [1561600519:163546][6268:140710526012496], txn-recover: Set global recovery timestamp: 0
2019-06-26T18:55:19.393-0700 I RECOVERY [initandlisten] WiredTiger recoveryTimestamp. Ts: Timestamp(0, 0)
2019-06-26T18:55:19.736-0700 I CONTROL [initandlisten]
2019-06-26T18:55:19.736-0700 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.
2019-06-26T18:55:19.740-0700 I CONTROL [initandlisten] **   Read and write access to data and configuration is unrestricted.
2019-06-26T18:55:19.743-0700 I CONTROL [initandlisten]
2019-06-26T18:55:19.746-0700 I CONTROL [initandlisten] ** WARNING: This server is bound to localhost.
2019-06-26T18:55:19.749-0700 I CONTROL [initandlisten] **   Remote systems will be unable to connect to this server.
2019-06-26T18:55:19.751-0700 I CONTROL [initandlisten] **   Start the server with --bind_ip <address> to specify which IP
2019-06-26T18:55:19.758-0700 I CONTROL [initandlisten] **   addresses it should serve responses from, or with --bind_ip_all to
2019-06-26T18:55:19.761-0700 I CONTROL [initandlisten] **   bind to all interfaces. If this behavior is desired, start the
2019-06-26T18:55:19.762-0700 I CONTROL [initandlisten] **   server with --bind_ip 127.0.0.1 to disable this warning.
2019-06-26T18:55:19.763-0700 I CONTROL [initandlisten]
2019-06-26T20:55:20.819-0500 W FTDC [initandlisten] Failed to initialize Performance Counters for FTDC: WindowsPdhError: PdhExpandCounterPathW failed with 'El objeto especificado no se encontró en el equipo.' for counter '\\Memory\\Available Bytes'
2019-06-26T20:55:20.820-0500 I FTDC [initandlisten] Initializing full-time diagnostic data capture with directory 'C:/data/db/diagnostic.data'
2019-06-26T20:55:20.840-0500 I NETWORK [initandlisten] waiting for connections on port 27017
2019-06-26T20:56:05.423-0500 I NETWORK [listener] connection accepted from 127.0.0.1:64502 #1 (1 connection now open)
2019-06-26T20:56:05.458-0500 I NETWORK [conn1] received client metadata from 127.0.0.1:64502 conn1: { driver: { name: "nodejs", version: "3.2.7" }, os: { type: "Windows_NT", name: "win32", architecture: "x64", version: "10.0.17134" }, platform: "Node.js v10.16.0, LE, mongod-core: 3.2.7" }
2019-06-26T23:03:00.600-0500 I NETWORK [conn1] end connection 127.0.0.1:64502 (0 connections now open)
2019-06-26T23:04:43.522-0500 I NETWORK [listener] connection accepted from 127.0.0.1:64932 #2 (1 connection now open)
2019-06-26T23:04:43.542-0500 I NETWORK [conn2] received client metadata from 127.0.0.1:64932 conn2: { driver: { name: "nodejs", version: "3.2.7" }, os: { type: "Windows_NT", name: "win32", architecture: "x64", version: "10.0.17134" }, platform: "Node.js v10.16.0, LE, mongod-core: 3.2.7" }

```

Figura 54. Servidor MongoDB iniciado.

3.4.2 Restauración de la Base de Datos INVENDB utilizada en el sistema web

“SIVENLAC”

Se restaura la base de datos INVENDB utilizando el comando “mongorestore –db INVENDB dump/INVENDB” dentro de la ventana de comandos de Windows en el directorio donde se encuentra instalada la base de datos.

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Versión 10.0.17134.029]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Polo>CD..
C:\Users>CD..
C:\>CD ARCHIVOS DE PROGRAMA
C:\Archivos de programa>cd mongodb
C:\Archivos de programa\mongodb>cd server
C:\Archivos de programa\mongodb\Server>cd 4.0
C:\Archivos de programa\mongodb\Server\4.0>cd bin
C:\Archivos de programa\mongodb\Server\4.0\bin>mongorestore --db INVENDB dump/INVENDB
```

Figura 55. Restore MongoDB.

Resultado:

```
C:\WINDOWS\system32\cmd.exe
C:\Archivos de programa\mongodb\Server\4.0>cd bin
C:\Archivos de programa\mongodb\Server\4.0\bin>mongorestore --db INVENDB dump/INVENDB
2019-06-18T11:20:49.374-0500 the --db and --collection args should only be used when restoring from a BSON file. Other uses are deprecated and will not exist in the
Future; use --include instead
2019-06-18T11:20:49.929-0500 building a list of collections to restore from dump\INVENDB\dir
2019-06-18T11:20:49.962-0500 reading metadata for INVENDB.terceros from dump\INVENDB\terceros.metadata.json
2019-06-18T11:20:49.962-0500 reading metadata for INVENDB.items from dump\INVENDB\items.metadata.json
2019-06-18T11:20:49.963-0500 reading metadata for INVENDB.documentos from dump\INVENDB\documentos.metadata.json
2019-06-18T11:20:49.963-0500 restoring INVENDB.terceros from dump\INVENDB\terceros.bson
2019-06-18T11:20:49.963-0500 reading metadata for INVENDB.stocks from dump\INVENDB\stocks.metadata.json
2019-06-18T11:20:49.964-0500 restoring INVENDB.items from dump\INVENDB\items.bson
2019-06-18T11:20:49.965-0500 restoring INVENDB.documentos from dump\INVENDB\documentos.bson
2019-06-18T11:20:49.965-0500 restoring INVENDB.stocks from dump\INVENDB\stocks.bson
2019-06-18T11:20:49.970-0500 no indexes to restore
2019-06-18T11:20:49.970-0500 finished restoring INVENDB.terceros (2 documents)
2019-06-18T11:20:49.971-0500 reading metadata for INVENDB.usuarios from dump\INVENDB\usuarios.metadata.json
2019-06-18T11:20:49.973-0500 restoring INVENDB.usuarios from dump\INVENDB\usuarios.bson
2019-06-18T11:20:49.975-0500 no indexes to restore
2019-06-18T11:20:49.975-0500 finished restoring INVENDB.documentos (3 documents)
2019-06-18T11:20:49.976-0500 reading metadata for INVENDB.bodegas from dump\INVENDB\bodegas.metadata.json
2019-06-18T11:20:49.977-0500 restoring INVENDB.bodegas from dump\INVENDB\bodegas.bson
2019-06-18T11:20:49.979-0500 error: E11000 duplicate key error collection: INVENDB.usuarios index: _id_dup key: { : ObjectId('5cfff5e6cb9b995d143b9d66') }
2019-06-18T11:20:49.979-0500 no indexes to restore
2019-06-18T11:20:49.979-0500 finished restoring INVENDB.usuarios (2 documents)
2019-06-18T11:20:49.997-0500 error: E11000 duplicate key error collection: INVENDB.bodegas index: _id_dup key: { : ObjectId('5d027170bcf2b32b81075e8') }
2019-06-18T11:20:49.997-0500 no indexes to restore
2019-06-18T11:20:49.998-0500 finished restoring INVENDB.bodegas (1 document)
2019-06-18T11:20:50.077-0500 error: multiple errors in bulk operation:
- E11000 duplicate key error collection: INVENDB.stocks index: _id_dup key: { : ObjectId('5d07fbbf4496e952d8aba772') }
- E11000 duplicate key error collection: INVENDB.stocks index: _id_dup key: { : ObjectId('5d07fbd74496e952d8aba779') }
- E11000 duplicate key error collection: INVENDB.stocks index: _id_dup key: { : ObjectId('5d07fbd74496e952d8aba77a') }
- E11000 duplicate key error collection: INVENDB.stocks index: _id_dup key: { : ObjectId('5d0804359bec715a6854aa88') }
2019-06-18T11:20:50.077-0500 no indexes to restore
2019-06-18T11:20:50.985-0500 finished restoring INVENDB.stocks (4 documents)
2019-06-18T11:20:50.986-0500 no indexes to restore
2019-06-18T11:20:50.989-0500 finished restoring INVENDB.items (3 documents)
2019-06-18T11:20:50.991-0500 done
C:\Archivos de programa\mongodb\Server\4.0\bin>
```

Figura 56. Base de Datos MongoDB restaurada.

3.4.3 Inicio del servidor Node.JS

Se inicia el servidor server.js desde Visual Studio Code.

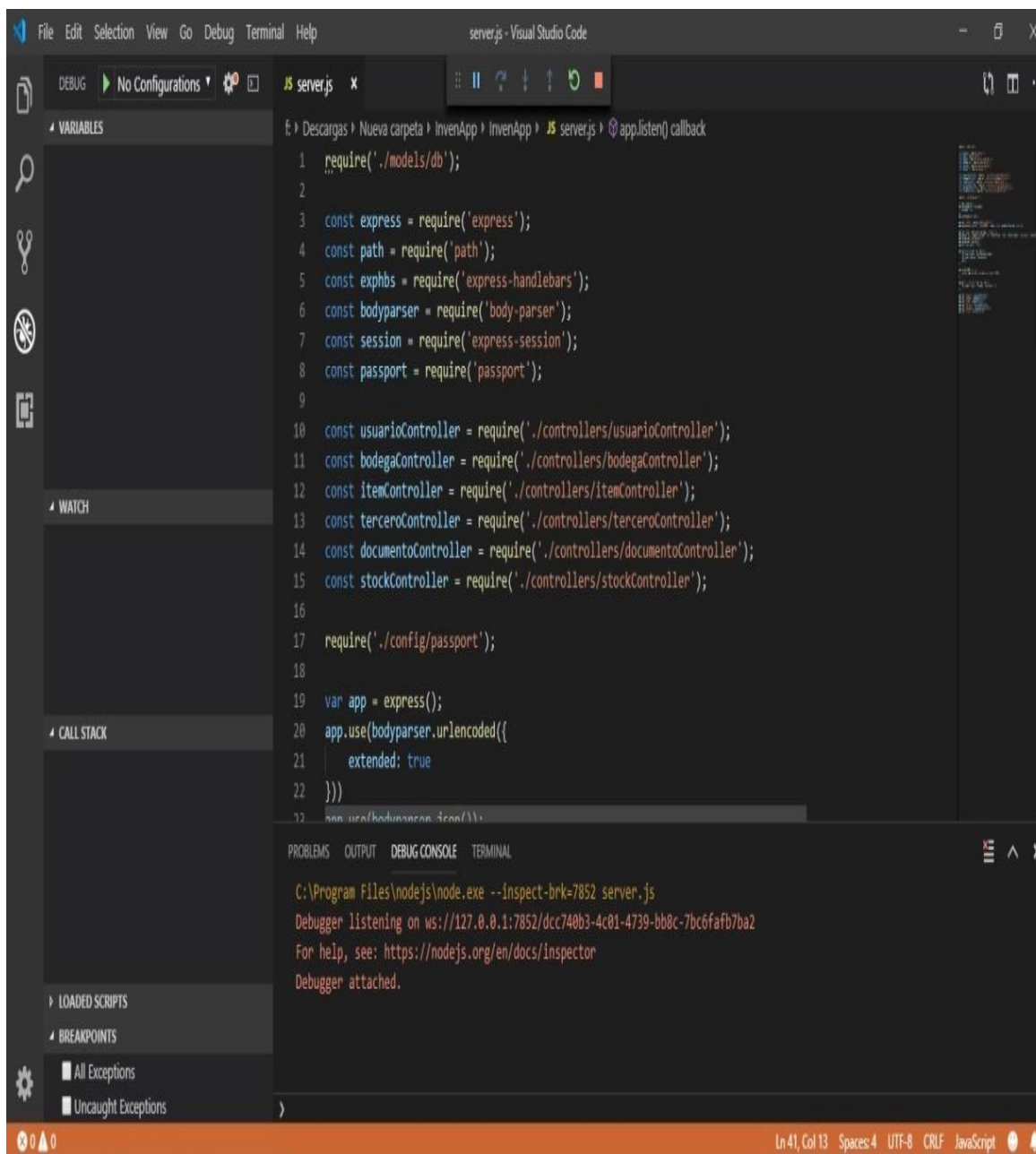


Figura 57. Inicio Server.js.

Resultado:

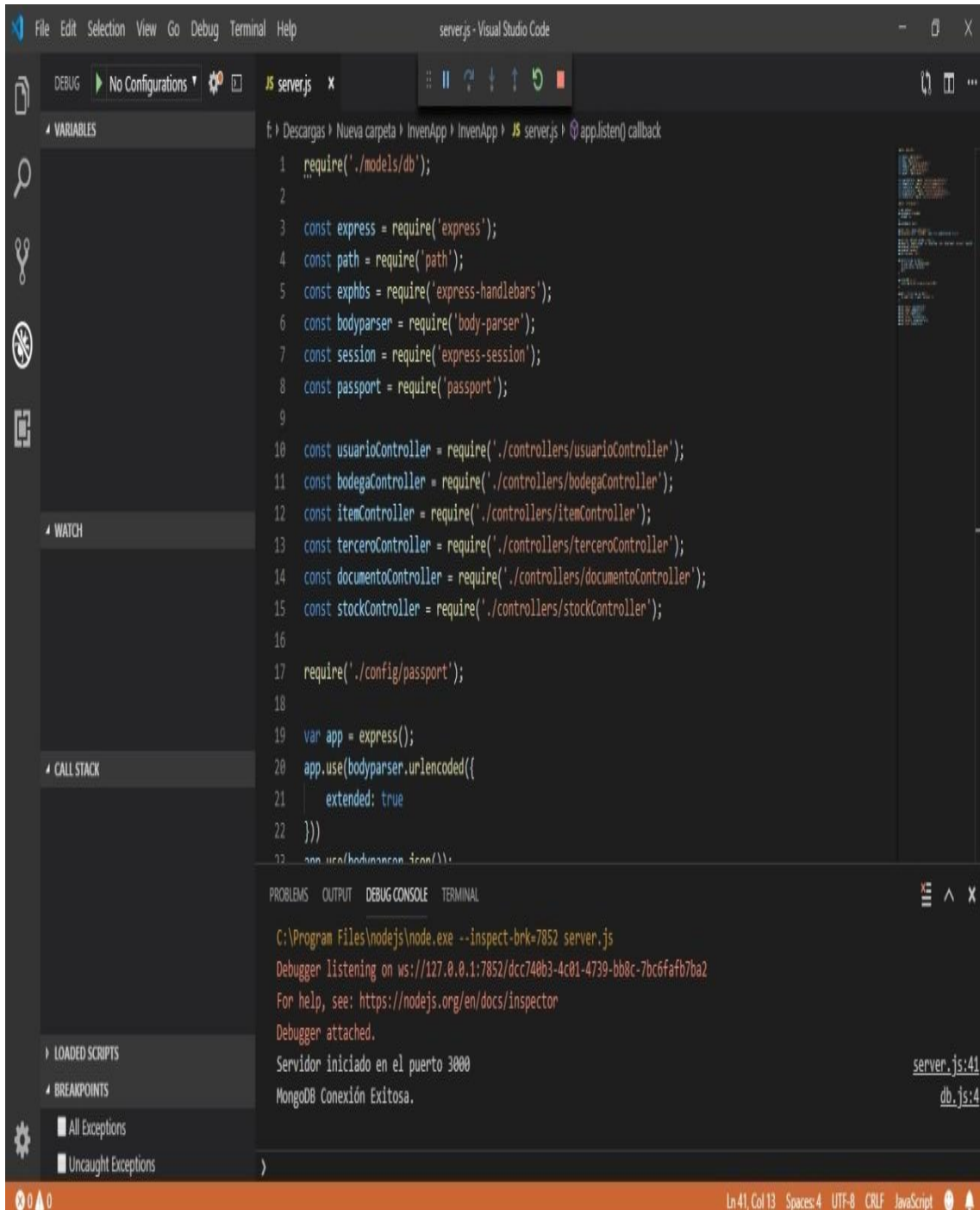


Figura 58. Server.js iniciado.

El servidor se inicia correctamente en el puerto 3000, y la conexión con la base de datos es exitosa.

3.4.4 Inicio de “SIVENLAC”

Se inicia el sistema web “SIVENLAC” desde un navegador web en el puerto 3000 utilizando “localhost:3000/” como dirección.

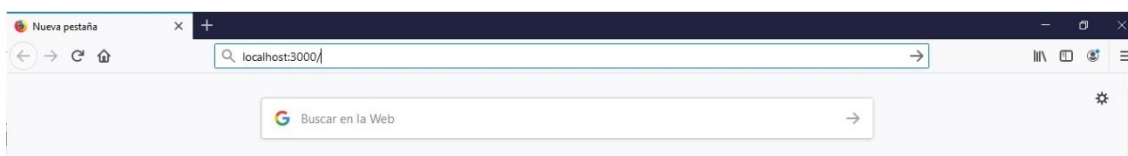


Figura 59. Iniciando "SIVENLAC".

Resultado:



Figura 60. "SIVENLAC".

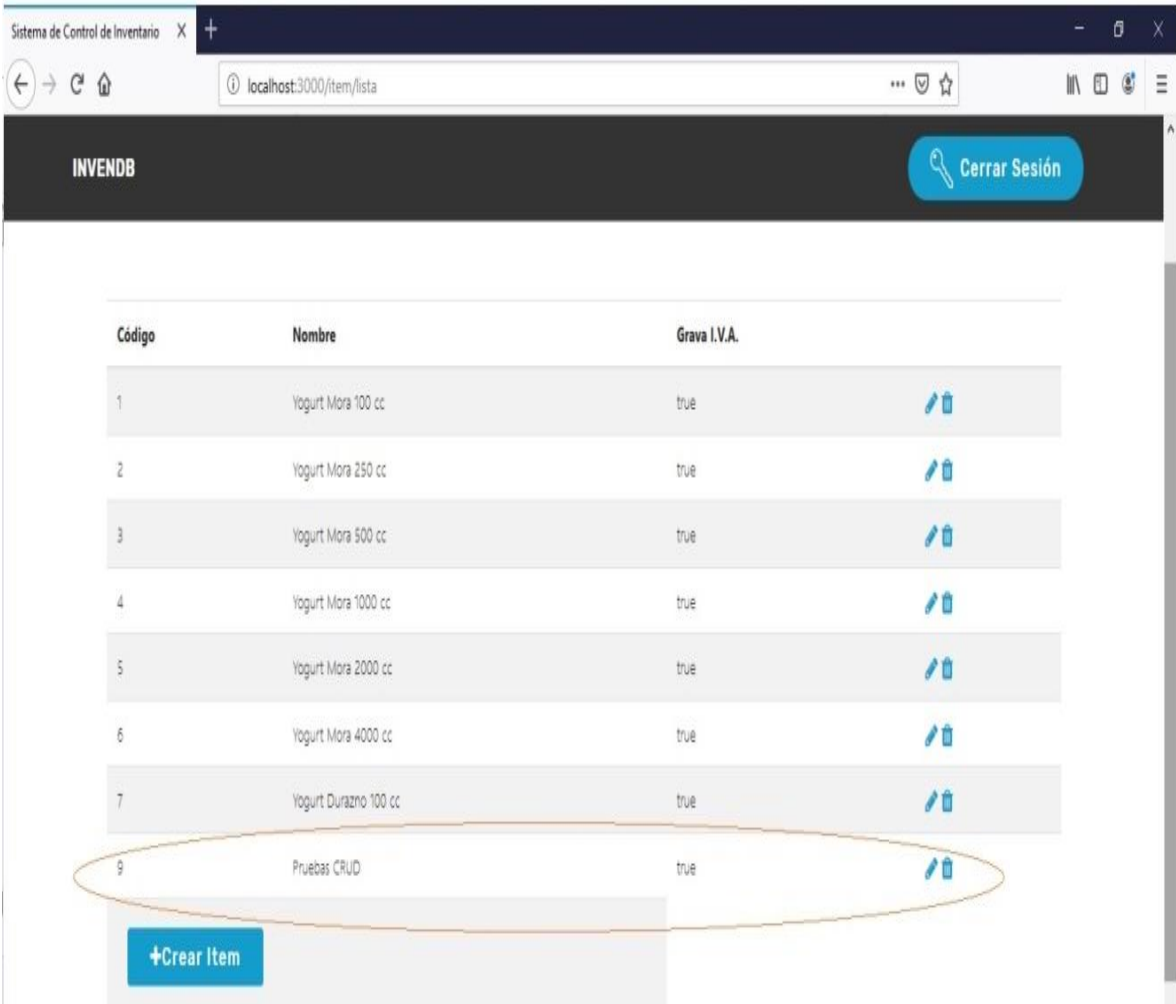
Se inicia el sistema web “SIVENLAC” correctamente.

3.4.5 Pruebas operaciones CRUD.

Tabla 25

Pruebas CRUD.

CREATE



The screenshot shows a web browser window with the URL `localhost:3000/item/lista`. The page title is 'Sistema de Control de Inventario'. The main content area is titled 'INVENDB' and contains a table of inventory items. The table has three columns: 'Código', 'Nombre', and 'Grava I.V.A.'. The items listed are:

Código	Nombre	Grava I.V.A.
1	Yogurt Mora 100 cc	true
2	Yogurt Mora 250 cc	true
3	Yogurt Mora 500 cc	true
4	Yogurt Mora 1000 cc	true
5	Yogurt Mora 2000 cc	true
6	Yogurt Mora 4000 cc	true
7	Yogurt Durazno 100 cc	true
9	Pruebas CRUD	true

The row for 'Pruebas CRUD' is circled in red. Below the table is a '+Crear Item' button. In the top right corner, there is a 'Cerrar Sesión' button. The browser's address bar shows the URL `localhost:3000/item/lista`.

Figura 61. CRUD - Create

READ

CONTINÚA



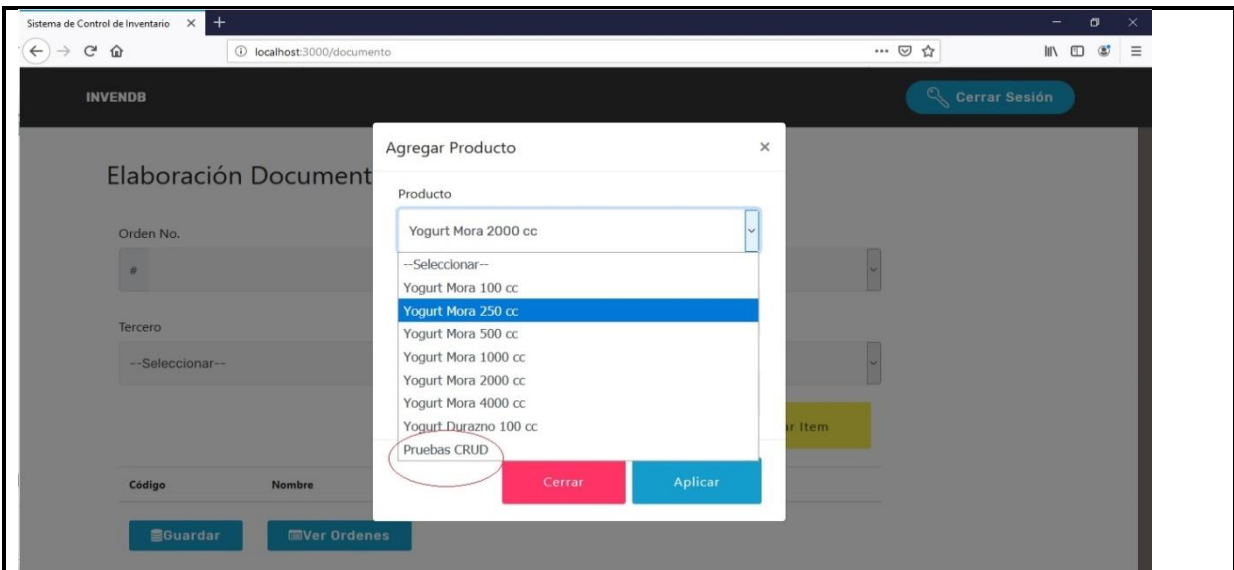


Figura 62. CRUD - Read.

UPDATE

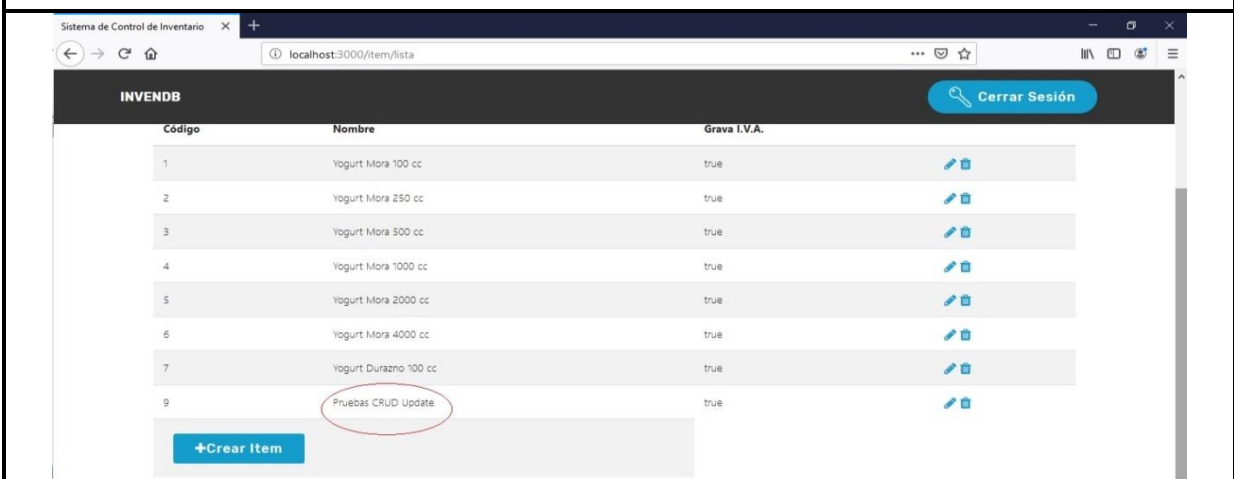


Figura 63. CRUD - Update.

DELETE

CONTINUÍA 

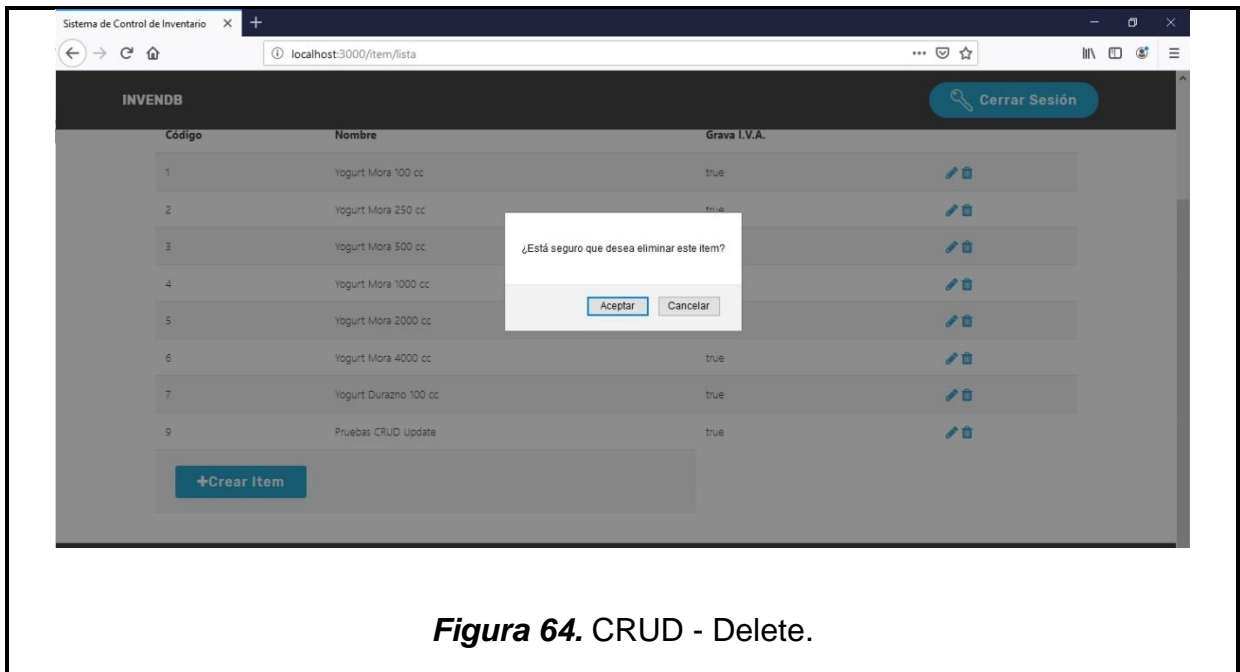


Figura 64. CRUD - Delete.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- El desarrollo de la Web necesita servidores capaces de responder de una manera rápida y efectiva ante la posible sobre carga de usuarios o de datos, los servicios Web depende gran parte de la respuesta de los servidores.
- MongoDB, resulta ser una de las bases de datos más eficientes y rápida con el manejo de grandes cantidades de información, debido a la rapidez de ejecución de la misma.
- En los experimentos realizados, los tiempos registrados por la aplicación Webserver Stress Tool para las operaciones CRUD nos dieron como resultado que el servidor Node.js es más eficiente que el servidor tradicional Apache; Node.js mostró mejores resultados en las pruebas realizadas por lo cual, puede brindar mayor rendimiento al momento de ser utilizado en un servicio Web.

5.2 Recomendaciones

- Se recomienda el uso de “SIVENLAC” ya que un sistema web de inventario es muy importante dentro de una organización ya que permite ahorrar recursos, tiempo y dinero y se puede tener un mejor control del stock.

- Se recomienda romper con los paradigmas tradicionales sobre el uso de base de datos no relacionales en sistemas de poca envergadura ya que son muy adaptables a cualquier tipo de sistema y puede mejorar en gran medida la funcionabilidad del sistema.

REFERENCIAS BIBLIOGRÁFICAS

- Aumaille, B. (2002). *J2EE Desarrollo de aplicaciones web*. Barcelona: Ediciones ENI.
- Castejón Garrido, J. S. (2004). Arquitectura y diseño de sistemas web modernos. Murcia: *Revista de Ingeniería Informática del CIIRM*.
- Castro Romero, A., Gonzáles Sanabria, J. S., & Callejas Cuervo, M. (2012). Utilidad y funcionamiento de las bases de datos NoSQL. Bogota: *Revista Facultad de Ingeniería, UPTC*, 21-32.
- Cevallos, K. (8 de Mayo de 2015). *Ingeniería del Software*. Obtenido de <https://ingsoftwarekarlacevallos.wordpress.com/2015/05/08/metodologia-de-desarrollo-agil-xp-y-scrum/>, recuperado el 19 de Abril del 2019.
- Duran, Y. (2012). Administración del inventario: elemento clave para la optimización de las utilidades en las. Mérida: *Visión Gerencial*, 55-78.
- Ernesto Valero , M. (12 de Mayo de 2016). *Paradigma Digital*. Obtenido de <https://www.paradigmadigital.com/dev/validacion-nativa-documentos-mongodb-3-2/>, recuperado el 21 de Abril del 2019.
- Garcés, L., & Egas, L. (2015). Evolución de las Metodologías de Desarrollo de la Ingeniería de Sistemas. Santa Elena: *Revista Científica y Tecnológica UPSE*.
- García Gonzáles, A. (29 de Junio de 2016). *PANAMAHITEK*. Obtenido de <http://panamahitek.com/montaje-servidor-apache-nuestra-pc/>, recuperado el 20 de Abril del 2019, recuperado el 29 de Abril del 2019.
- Gupta, M., Bangare, S. L., Dalal, M., & Inamdar, A. (2016). Using Node.js to Build High Speed and Scalable. Ravi Nagar: *International Journal of Research in Advent Technology*.
- Kabir, M. (2003). *La biblia del servidor apache*. Madrid: Anaya Multimedia.
- Letelier, P., & Penadés, M. C. (2012). Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). Valencia: *Técnica Administrativa*.
- Márquez Díaz, J., Sampedro, L., & Vargas, F. (2002). Instalación y configuración de Apache, un servidor Web gratis. Madrid: *Redalyc*, 10-23.
- Node.JS. (s.f.). Obtenido de <https://nodejs.org/es/>, recuperado el 07 de Junio del 2019.

Okman, L., Gal-Oz, N., Gonen, Y., Gudes, E., & Abramov, j. (2011). Security Issues in NoSQL Databases. *2011 International Joint Conference of IEEE TrustCom-11/IEEE ICESS-11/FCST-11*. Changsha, Pekín.

Tilkov, S., & Vinoski, S. (2010). Node.js: Using JavaScript to Build High-Performance Network Programs. Washington: *IEEE Internet Computing*, 80-83.

Valdés Pérez. (26 de Octubre de 2007). *Maestros del Web*. Obtenido de <http://www.maestrosdelweb.com/que-son-las-bases-de-datos/>, recuperado el 25 de Abril del 2019.

Visual Studio Code. (s.f.). Obtenido de <https://code.visualstudio.com/docs>, recuperado el 05 de Junio del 2019.

ANEXOS



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

CERTIFICACIÓN

Se certifica que el presente trabajo de titulación fue desarrollado por el señor:
VERGARA BUSTILLOS, PAÚL ALBERTO.

Ing. Navas Moya, Milton Patricio
Director de Proyecto

Ing. Garcés Guayta, Lucas Rogelio
Director de Carrera



Abg. Albán Yáñez, Darwin Santiago
Secretario Académico