



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

**“DESARROLLO DE UNA APLICACIÓN DE GESTIÓN DE COMPRAS Y
PAGOS EN EL ALMACÉN DE MUEBLES “CACHA” UTILIZANDO EL
MICRO-FRAMEWORK LUMEN”**

TRABAJO DE TITULACIÓN
TIPO: PROYECTO TÉCNICO

Presentado para optar al grado académico de:
INGENIERO EN SISTEMAS INFORMÁTICOS

AUTOR: KLEBER EDUARDO HIPO MOROCHO

TUTOR: Ing. DANILO MAURICIO PASTOR RAMÍREZ.

Riobamba – Ecuador

2019

©2019, Kleber Eduardo Hipo Morocho

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el derecho del autor.

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA EN SISTEMAS

El tribunal de Trabajo de Titulación certifica que: El proyecto Técnico “DESARROLLO DE UNA APLICACIÓN DE GESTIÓN DE COMPRAS Y PAGOS EN EL ALMACÉN DE MUEBLES “CACHA” UTILIZANDO EL MICRO-FRAMEWORK LUMEN”, de responsabilidad del Sr. KLEBER EDUARDO HIPO MOROCHO, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación, quedando autorizada su presentación.

NOMBRE	FIRMA	FECHA
Ing. Washington Luna DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA	_____	_____
Ing. Patricio Moreno DIRECTOR DE LA ESCUELA DE INGENIERÍA EN SISTEMAS	_____	_____
Ing. Danilo Pastor DIRECTOR DE TRABAJO DE TITULACIÓN	_____	_____
Ing. Alejandra Oñate MIEMBRO DE TRABAJO DE TITULACIÓN	_____	_____

Yo, KLEBER EDUARDO HIPO MOROCHO soy responsable de las ideas, doctrinas y resultados expuestos en esta Tesis y el patrimonio intelectual de la Tesis de Grado pertenece a la Escuela Superior Politécnica de Chimborazo.

Kleber Eduardo Hipo Morocho

DEDICATORIA

El presente trabajo es dedicado a mis padres Francisco y María que con su trabajo y esfuerzo han apoyado para llegar hasta esta etapa de mi vida, que han sido un pilar fundamental en mi vida universitaria que con sus consejos y mensajes de motivación enseñando que con esfuerzo y perseverancia se puede alcanzar las metas propuestas. A mi hermano Alex que estuvo presente para apoyarme en los momentos más difíciles con sus mensajes de motivación un mil gracias.

A todas las personas que me apoyaron moral y económicamente durante mi vida universitaria.

Un mil gracias.

Kleber Eduardo

AGRADECIMIENTO

Agradecer en primera instancia a DIOS por brindarme salud y vida. A mis padres que es lo más valioso que me dio la vida, por su apoyo moral y económicamente.

A la Escuela Superior Politécnica de Chimborazo que me dio la oportunidad de preparar para mi vida profesional. También al Ing. Danilo Pastor que supo guiar para el desarrollo del presente trabajo de titulación y así terminar mi carrera universitaria.

TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	x
ÍNDICE DE FIGURAS.....	xii
ÍNDICE DE GRÁFICOS.....	xv
ÍNDICE DE ANEXOS.....	xvi
ÍNDICE DE ABREVIATURAS.....	xvii
RESUMEN.....	xviii
ABSTRACT.....	xix
INTRODUCCIÓN.....	1

CAPÍTULO I

1. MARCO TEÓRICO.....	6
1.1. Aplicaciones Web.....	6
1.1.1. <i>Definición.</i>	6
1.1.2. <i>Clasificación.</i>	6
1.2. API.....	7
1.2.1. <i>Definición</i>	7
1.2.2. <i>Clasificación</i>	7
1.2.3. <i>Ventaja de REST Sobre SOAP</i>	7
1.3. API REST	8
1.3.1. <i>Definición</i>	8
1.3.2. <i>Características de la Arquitectura API REST.</i>	9
1.3.3. <i>Modelo de Autenticación</i>	10
1.4. Lenguajes de Programación Orientados a la Web	11
1.4.1. <i>Definición.</i>	11
1.4.2. <i>Comparación lenguaje de programación orientado a la Web</i>	11
1.5. Lenguaje Programación PHP.....	12
1.5.1. <i>Definición.</i>	12
1.5.2. <i>Características</i>	12
1.5.3. <i>Ventajas y Desventajas</i>	13
1.6. Framework.	13

1.6.1. <i>Definición.</i>	13
1.6.2. <i>Ventajas y desventajas</i>	13
1.7. Micro-Framework	14
1.7.1. <i>Definición</i>	14
1.7.2. <i>Tipos de Micro-Framework</i>	14
1.8. Lumen	15
1.8.1. <i>Definición.</i>	15
1.8.2. <i>Arquitectura</i>	16
1.8.3. <i>Funcionamiento</i>	16
1.8.4. <i>Lumen Vs Laravel.</i>	17
1.9. Mysql	18
1.9.1. <i>Definición.</i>	18
1.9.2. <i>Características</i>	19
1.10. Aplicación de Página Única (SPA)	19
1.11. VUE.JS	21
1.12. Metodología Scrum	22
1.12.1. <i>Roles en Scrum</i>	22
1.12.2. <i>El Sprint - ¿Dónde? ¿Cuándo?</i>	23
1.13. Calidad de Software	23
1.13.1. Estándar ISO 9126	24
1.13.2. Eficiencia	24

CAPITULO II

2. MARCO METODOLÓGICO	25
2.1 Introducción	25
2.2 Tipo de Investigación	25
2.3 Técnicas	25
2.4 Aplicación de la Metodología Ágil Scrum	25
2.4.1 <i>Estudio Preliminar</i>	25
2.4.2 <i>Fase planificación</i>	26
2.4.3 <i>Fase Desarrollo</i>	32
2.4.4 <i>Fase de Cierre</i>	53
2.5 Método de Evaluación	54
2.5.1. Planificación	54
2.5.2. Población	54

2.5.3. Muestra.....	54
2.5.4. Determinación del escenario de prueba.	55

CAPITULO III

3. MARCO DE RESULTADOS, DISCUSIÓN Y ANÁLISIS

3.1. Tiempo de Respuesta.....	57
3.1.1. Análisis de los Proceso en la actualidad en el Almacén	57
3.1.2. Análisis de los Proceso utilizando el sistema web	58
3.1.3. Comparación de Resultados.....	58
3.1.4. Análisis de Datos con t-student	59
3.2. Utilización de Recursos.....	63
3.2.1. Densidad del mensaje de uso de entrada/salida	64
3.2.2. Métrica Uso de memoria RAM	71
3.2.3. Métrica uso de procesador	75
3.3. Resultados Obtenidos	80
3.3.1. Tiempos de respuesta	80
3.3.2. Utilización de recursos.....	81
CONCLUSIONES	86
RECOMENDACIONES	87

BIBLIOGRAFÍA

ANEXOS

ÍNDICE DE TABLAS

Tabla 1-1:	Tabla de los métodos http (verbos)	8
Tabla 2-1:	Tabla comparativa de los diferentes Lenguajes de Programación.	11
Tabla 3-1:	Tabla de ventajas y desventajas de php	13
Tabla 4-1:	Tabla de las ventajas y desventajas de usar un framework.....	14
Tabla 5-1:	Comparación de Laravel vs Lumen	18
Tabla 1-2:	Tabla de puntos estimados por el método T-Shirt.....	26
Tabla 2-2:	Tabla Product Backlog - Historias de Usuario.....	27
Tabla 3-2:	Tabla Product Backlog Historias Técnicas	28
Tabla 4-2:	Tabla de las personas y su rol del proyecto	29
Tabla 5-2:	Tipos de usuario y roles de usuario del sistema	29
Tabla 6-2:	Tabla del Sprint Backlog con las Historias Técnicas y Usuarios.....	30
Tabla 7-2:	Historia de usuario del registro de compras.....	44
Tabla 8-2:	Prueba de aceptación 1, Registro de Compra	44
Tabla 9-2:	Prueba de aceptación 2, Registro de Compra	45
Tabla 10-2:	Prueba de aceptación 3, Registro de compra	45
Tabla 11-2:	Tarea de ingeniería 1, Registro de compra	46
Tabla 12-2:	Prueba de aceptación 1, Tarea de ingeniería 1, Registro de compra.....	46
Tabla 13-2:	Tarea de ingeniería 2, Registro de compra	47
Tabla 14-2:	Prueba de aceptación 1, Tarea de ingeniería 2, Registro de compra	47
Tabla 15-2:	Tarea de ingeniería 2, prueba de aceptación 2, Registro de compra	47
Tabla 16-2:	Tarea de ingeniería 3, Registro de compra	48
Tabla 17-2:	Prueba de aceptación 1, Tarea de ingeniería 3, Registro de compra	48
Tabla 18-2:	Tarea de ingeniería 3, prueba de aceptación 2, Registro de compra	49
Tabla 19-2:	Tarea de ingeniería 3, prueba de aceptación 3, Registro de compra	49
Tabla 20-2:	Tarea de ingeniería 4, Registro de compra	50
Tabla 21-2:	Prueba de aceptación 1, Tarea de ingeniería 4 del Registro de Compra	50
Tabla 22-2:	Tarea de ingeniería 4, prueba de aceptación 2, Registro de compra	51
Tabla 23-2:	Tarea de ingeniería 5, Registro de compra	51
Tabla 24-2:	Prueba de aceptación 12, Tarea de ingeniería 5 del Registro de Compra	52
Tabla 25-2:	Prueba de aceptación 13, Tarea de ingeniería 5, Registro de compra	52
Tabla 26-2:	Lista de las Historias de Usuario para el caso de estudio	55
Tabla 1-3:	Tiempos de respuesta de los procesos de forma manual	57
Tabla 2-3:	Tabla de tiempos de Respuesta con el Sistema	58

Tabla 3-3: Comparación de los tiempos de Respuesta con y sin el sistema	59
Tabla 4-3: Prueba t para muestras emparejadas.....	61
Tabla 5-3: Regla de decisión de los datos	62
Tabla 6-3: Tabla de características de utilización de recursos densidad de mensaje	64
Tabla 7-3: Resultados de la métrica densidad del mensaje I/O	70
Tabla 8-3: Evaluación de la métrica de uso de memoria RAM.....	71
Tabla 9-3: Resultados de las tareas de utilización de memoria	75
Tabla 10-3: Descripción para la evaluación de la métrica uso del procesador	76
Tabla 11-3: Resultados de la evaluación de la métrica uso del procesador	79
Tabla 12-3: Criterio Evaluación de la Eficiencia.....	80
Tabla 13-3: Indicador de evaluación de los tiempos de respuesta	80
Tabla 14-3: Resultados de los tiempos de respuesta obtenidos con el sistema	80
Tabla 15-3: Indicador de evaluación de la métrica densidad de mensaje I/O	81
Tabla 16-3: Resultados obtenidos densidad de mensaje I/O	81
Tabla 17-3: Indicador de evaluación del uso de memoria.....	82
Tabla 18-3: Resultados obtenidos del uso de memoria RAM	82
Tabla 19-3: Indicador de la métrica evaluación del uso de Procesador	83
Tabla 20-3: Resultado Obtenidos del uso del procesador	83
Tabla 21-3: Ponderación de las métricas de evaluación del sistema.....	84
Tabla 22-3: Resultados finales de las métricas de la eficiencia	84
Tabla 23-3: Criterio para la evaluación final de la eficiencia.....	85

ÍNDICE DE FIGURAS

Figura 1-1: Solicitud de los API.....	8
Figura 2-1: Autenticación para API REST	10
Figura 3-1: Acceso a las páginas por token.....	10
Figura 4-2: Arquitectura Micro-Framework Lumen.....	16
Figura 5-2: Funcionamiento del Micro-Framework Lumen	16
Figura 6-2: Activación de Eloquent Lumen	17
Figura 7-1: Web Tradicional vs SPA.....	20
Figura 8-1: Ecosistema de Vue.....	22
Figura 1-2: Arquitectura del sistema.....	32
Figura 2-2: Autenticación de los usuarios	34
Figura 3-2: Registro de Información.....	34
Figura 4-2: Página de la Descarga de Composer	35
Figura 5-2: Instalación de composer	36
Figura 6-2: Instalación del micro-Framework Lumen	36
Figura 7-2: Proceso de instalación de Lumen.....	36
Figura 8-2: comando de ejecución del micro-Framework	37
Figura 9-2: Interfaz de inicio del micro-Framework	37
Figura 10-2: Pagina de la descarga de nodejs	37
Figura 11-2: Instalación de Vue.js.....	37
Figura 12-2: Versión del Vue.Js Instalado	38
Figura 13-2: Creación de un nuevo proyecto de vue	38
Figura 14-2: Finalización de la instalación del proyecto	38
Figura 15-2: Instalación de las dependencias para vue.....	39
Figura 16-2: Ejecución del proyecto en modo desarrollador Vue.js.....	39
Figura 17-2: Interfaz principal del Framework Vue.js	39
Figura 19-2: Diagrama Lógico de la Base de Datos	41
Figura 20-2: Página de descarga de plugins datadict_grt.py	42
Figura 21-2: Instalación del plugins datadict_grt.py	42
Figura 22-2: Selección del archivo datadict_grt.py	43
Figura 23-2: Generación del diccionario de datos	43
Figura 24-2: Diccionario datos tabla factura.....	43
Figura 1-3: Tiempos Promedios Automatizados en Shapiro-wilk.....	60
Figura 2-3: Tiempos Promedios Manuales en Shapiro-wilk.....	60

Figura 1-3: Error Controlado que los datos sean Objeto JSON.....	65
Figura 2-3: Error Controlado cuando envié datos vacíos del cliente	65
Figura 3-3: Error Controlado cuando envié sin productos comprados	65
Figura 4-3: Error Controlado cuando los datos no este correctos del producto	65
Figura 5-3: Error Controlado cuando trate de modificar compra no existente	65
Figura 6-3: Error Controlado cuando no envié el formato	66
Figura 7-3: Error Controlado cuando no haya seleccionado el cliente	66
Figura 8-3: Error Controlado cuando no haya agregado productos al pedido cliente.....	66
Figura 9-3: Error Controlado cuando no haya agregado productos al pedido cliente.....	66
Figura 10-3: Error Controlado cuando el precio de venta sea menor que al de compra	66
Figura 11-3: Error Controlado cuando no haya Cantidad disponible en el Almacén	67
Figura 12-3: Error Controlado no exista el pedido a modificar.....	67
Figura 13-3: Error Controlado cuando los datos no son de tipo JSON.....	67
Figura 14-3: Error Controlado cuando elimine un pedido no existente	67
Figura 15-3: Error Controlado de cedula duplicada	68
Figura 16-3: Error Controlado cuando no se suba la foto	68
Figura 17-3: Error Controlado Campos vacíos.	68
Figura 18-3: Error Controlado de modificación de clientes no existentes.	68
Figura 19-3: Error Controlado, envié datos no JSON para generar Factura	68
Figura 20-3: Error Controlado, en él envié datos no JSON para la venta a crédito	69
Figura 21-3: Error Controlado en él envié de datos sin la tabla de amortización.....	69
Figura 22-3: Error Controlado cuando los datos no tiene formato JSON Para Crédito	69
Figura 23-3: Error Controlado, monto mayor que el crédito a pagar	69
Figura 24-3: Error Controlado, datos recibidos no es JSON para Reporte de Compras	69
Figura 25-3: Error Controlado, datos recibidos no es JSON para Reporte de pedido	70
Figura 26-3: Error Controlado, datos recibidos no es JSON para Reporte del stock mínimo....	70
Figura 27-3: Utilización de la memoria en la tarea de Registrar Compra.....	71
Figura 28-3: Utilización de la memoria en la tarea de modificar Compra.....	72
Figura 29-3: Utilización de la memoria en la tarea de Registrar Pedido	72
Figura 30-3: Utilización de la memoria en la tarea de Modificar Compra	72
Figura 31-3: Utilización de la memoria para la tarea de eliminar Pedido.....	72
Figura 32-3: Utilización de la memoria en la tarea de Registrar Cliente	73
Figura 33-3: Utilización de la memoria en la tarea de Modificar Cliente.....	73
Figura 34-3: Utilización de la memoria en la generación de una factura	73
Figura 35-3: Utilización de la memoria en generar un crédito de una factura	74
Figura 36-3: Utilización de la memoria en el cobro del crédito	74
Figura 37-3: Utilización de la memoria en el reporte de las compras	74

Figura 38-3: Utilización de la memoria en el reporte del listado de pedidos.....	74
Figura 39-3: Utilización de la memoria en el reporte de los productos con stock mínimo.....	75
Figura 40-3: Utilización del Procesador en la tarea de Registrar Compra.....	76
Figura 41-3: Utilización del Procesador en la tarea de modificar Compra.....	76
Figura 42-3: Utilización del Procesador en la tarea de Registrar Pedido.....	76
Figura 43-3: Utilización del Procesador en la tarea de Modificar Compra.....	77
Figura 44-3: Utilización del Procesador para la tarea de eliminar Pedido.....	77
Figura 45-3: Utilización del Procesador en la tarea de Registrar Cliente.....	77
Figura 46-3: Utilización del Procesador en la tarea de Modificar Cliente.....	77
Figura 47-3: Utilización del Procesador en la generación de una factura.....	78
Figura 48-3: Utilización del Procesador en generar un crédito de una factura.....	78
Figura 49-3: Utilización del Procesador en el cobro del crédito.....	78
Figura 50-3: Utilización del Procesador en el reporte de las compras.....	78
Figura 51-3: Utilización del Procesador en el reporte del listado de pedidos.....	79
Figura 52-3: Utilización del Procesador en el reporte de los productos con stock mínimo.....	79

ÍNDICE DE GRÁFICOS

Gráfico 1-2: Gráfico Burndown Chart del proyecto	53
Gráfico 1-3: Gráfico de comparación de los tiempos de respuesta.....	63
Gráfico 2-3: Porcentaje de tiempos de Respuestas	63

ÍNDICE DE ANEXOS

Anexo A: Diagrama de Gantt

Anexo B: Diccionario de datos

Anexo C: Historias de Usuario Y técnicas

Anexo D: Manual de Usuario

Anexo E: Tiempos de Respuesta con el sistema

ÍNDICE DE ABREVIATURAS

JSON	Notación de Objetos de JavaScript.
HTML	Lenguaje de marcas de hipertexto.
CSS	Hojas de estilo en cascada.
PHP	Preprocesador de hipertexto.
API	Interfaz de programación de aplicaciones
REST	Transferencia de estado representacional
CORS	Intercambio de Recursos de Origen Cruzado
PDF	Formato de documento portátil
ISO	Organización Internacional de Normalización
SQL	Lenguaje de consulta estructurada
HTTP	Protocolo de transferencia de hipertexto
ISO	Organización Internacional de Normalización

RESUMEN

En el presente trabajo de titulación se desarrolló una aplicación de gestión de compras y ventas para el Almacén de Muebles Cacha, utilizando el micro-Framework Lumen más el gestor de base de datos MySQL. Para la recolección de la información se utilizó la técnica de la entrevista, observación y la revisión de documentos, con el fin de conocer las necesidades y la prioridad para el correcto desempeño de las actividades en el negocio. Durante el desarrollo se aplicó la metodología Ágil SCRUM, se realizaron 9 Sprints con una duración de 120 horas, en estos se desarrollaron los requerimientos definidos, como diseño y las funcionalidades del sistema, obteniendo un total de 45 historias de usuario y 7 historias técnicas, posteriormente, en el desarrollo se derivaron 161 tareas de ingeniería y 444 pruebas de aceptación. Para la evaluación de la eficiencia del software se aplicó la norma ISO 9126, en base a la cual se analizó el comportamiento del tiempo y utilización de recursos, en el primero se analizaron los tiempos de respuesta, en el segundo se analizó la densidad de mensajes de entrada y salida (I/O), uso de memoria y procesador. Los datos recopilados fueron: tiempos de respuesta 50%, densidad de mensajes I/O 6.67%, memoria 20% y procesador 3.33%; obteniendo un total del 80%, siendo esta una eficiencia satisfactoria de acuerdo al análisis realizado. Se analizaron los tiempos de respuesta de forma tradicional con respecto a la forma automatizada, en donde aplicando la prueba T-student se constató que los tiempos para completar las tareas disminuyeron, de forma automatizada con respecto a la forma manual en un 87.86%. Se recomienda, en caso de que el microframework Lumen no posea algunas características, este sea migrado al framework completo Laravel.

PALABRAS CLAVES: <INGENIERÍA DE SOFTWARE>, <DESARROLLO DE SOFTWARE>, <FACTURACIÓN>, <INVENTARIO>, <MICRO FRAMEWORK-LUMEN>, <NORMA ISO /IEC 9126>, < EFICIENCIA>

ABSTRACT

In the present titling work can be an application of management of purchases and sales for the warehouse of furniture Cacha, using the micro-frame Lumen plus the manager of data of MySQL, For the recolection of the information was used the technique of Interview, observation and review of documents, in order to know the needs and the priority for the correct performance of the activities in the business. During the development the Agile SCRUM methodology was applied, 9 sprints were carried out with a duration of 120 hours, the elements used were developed, such as the design and system functionalities, which are obtained from a total of 45 user stories and 7 technical histories, Subsequently, 161 engineering tasks and 444 acceptance tests were developed. For the evaluation of software efficiency, the ISO 9126 standard is applied, based on which the behavior of time and the use of resources is analyzed. The first one analyzes the response times, the second analyzes the density of the Input and output (I / O) messages, memory usage and processor. The data collected were: response times 50%, message density I / O 6.67%, memory 20% and processor 3.33%; A total of 80% is being obtained. The response times will be analyzed in a traditional way with respect to the automated form, where the T-student test will be applied to complete the reduced tasks, automatically with respect to the manual form in 87.86%. It is recommended, in the case that the microframewok Lumen does not have some characteristics, this should be migrated to the complete framework of Laravel.

KEY WORDS: <SOFTWARE ENGINEERING>, <SOFTWARE DEVELOPMENT>, <BILLING>, <INVENTORY>, <MICRO FRAMEWORK-LUMEN>, <ISO / IEC 9126>, <EFFICIENCY>

INTRODUCCIÓN

Antecedentes

Hoy en día el desarrollo de las aplicaciones para el internet han ganado su espacio estos ayudan a las empresas en los diferentes procesos que realicen como ventas hacer conocer los productos o servicios que ofertan a los clientes, finanzas hacer conocer la situación actual de la empresa como llevar contabilidad, recursos económicos conocer las utilidades generadas durante un tiempo de trabajo, realizar la distribución de cargas horarias para sus funcionarios y a su vez registrar el ingreso y salida del trabajo. Entre las principales áreas y sus respectivas aplicaciones tenemos la de gestión de compras (fullstep), ventas (farandsoft), las finanzas (Alegra), facturación (factuonline), el marketing (Mention), la contabilidad (Alegra), producción y muchas otras áreas importantes de la empresa pueden apoyarse en el uso de herramientas web y así elevar los índices de productividad mientras se reducen sus costos. Dichas aplicaciones son creadas mediante estructuras o marcos ya prediseñados de código que automatizan y que ayudan a optimizan el tiempo durante el desarrollo de aplicaciones, estos son los llamados framework.

En la actualidad la mayoría de los desarrolladores usan un framework para poder desarrollar una aplicación esto hace ya que la mayoría de los framework se basan en la arquitectura del patrón de diseño MVC (Modelo, Vista y Controlador), permitiendo así tener una aplicación más robusta (Colaboradores, 2016). Para poder utilizarlo se tiene que incorporar todos sus componentes aunque no se lo utilice alguno de ellos en la aplicación. Se debe configurar antes de empezar a trabajar porque no vienen configuradas y su curva de aprendizaje por lo que es necesario aprender todo para su correcto funcionamiento.

Actualmente los frameworks completos tiene sus ventajas como optimizar el tiempo a sus desarrolladores, sin embargo no ofrecen ninguna flexibilidad, mientras que los micro frameworks tienen muchas opciones flexibles (R2b Solutions, 2016). Las principales diferencias son: La curva de aprendizaje aumentará exponencialmente a un límite, La huella del código es pequeña, lo que significa que el mantenimiento tiende a ser bajo. Estos ayudan en el desarrollo de sitios web simples y usando recursos específicos proporcionando esquemas de desarrollo más rápido y también mejora las pruebas de calidad.

Existen una gran cantidad de micro-framework como: Silex, Slim, Lumen, Phalcon, Limonade. **SÍLEX** que “un micro framework que utiliza componentes de Symfony permitiéndole heredar las buenas prácticas y profesionalismo de Symfony pero manteniendo la agilidad que da una aplicación sencilla de PHP” (Enzo, 2014), **SLIM** es un excelente framework MVC que nos permite crear aplicaciones rápidamente sin levantar una infraestructura llena de módulos y opciones que

no vamos a utilizar. Es liviano, la sintaxis es clara y hace uso extensivo de todas las cualidades de PHP 5.3 (latincloud, 2013), **LUMEN** es la solución perfecta para la construcción de **micro-servicios** basados en Laravel y Blazing Fast API. Lumen es uno de los micro-framework más rápidos disponibles en la actualidad (Chacaltana, 2015).

Para este trabajo se inclinó por usar el micro-framework lumen por las siguientes razones:

La capacidad de procesar más de mil peticiones, la mayor parte de Lumen viene ya pre configurado y en caso de necesitarlo lo hace en el archivo .env por lo cual no existe el directorio config como lo posee su hermano Laravel, posee un diferente enrutador y esta es la principal razón por la que es más veloz usa FastRoute una implementación de enrutamiento ligera desarrollada por Nikita Popov (Lewis, 2015).

Además Lumen permite definir rutas que son gestionadas automáticamente. También posibilita definir middleware, que es un tipo de componente intermedio que se ejecuta antes y/o después de una petición HTTP para modificarla. En este caso se utiliza para autenticación, para limitar la cantidad de peticiones por unidad de tiempo y para gestionar el mecanismo de seguridad Cross-Origin Resource Sharing (CORS), el cual es obligatorio en las peticiones de tipo XMLHttpRequest (Marco, 2015).

Como alternativa para el desarrollo de sistema en el almacén Cacha queremos automatizar el proceso de pedidos y ventas de tal forma que se pueda hacer registro de clientes, tener un catálogo de los artículos disponibles en stock especificando su categoría a la que pertenece, almacenamiento de los datos de los empleados.

El Almacén de muebles “**Cacha**” se encuentra ubicado en la ciudad de Riobamba en las se dedica a la venta de los muebles a la comunidad Riobambeña y de la Provincia.

El proceso actual para realizar un pedido, los clientes deben acercarse al local y solicitar lo que desean comprar al ejecutivo de ventas, el cual debe constatar en el inventario la existencia del mismo y posterior recoge los datos del solicitante en una solicitud del pedido, en caso que no exista se debe enviarle el pedido a la fábrica y la entrega se tarda de 3 a 5 días.

En caso de que el pedido sea a crédito se registra en una tarjeta kardex especificando el monto del pedido el tiempo para su cancelación y de la misma forma la frecuencia de pagos (diarios, semanales, quincenales o mensuales).

Por lo tanto se plantea desarrollar el sistema para almacén de muebles Cacha, tal que permita reservar una cantidad de artículos (Pedidos) que posteriormente será cancelado a crédito o a contado (venta), la venta a crédito se tendrá que especificar el tiempo y la frecuencia de pagos.

Formulación del Problema

¿Se puede desarrollar el sistema para la automatización de almacenes Cacha orientada a la web usando el micro-framework lumen para evaluar la eficiencia de software?

Justificación

Justificación Teórica

Cuando se construyen aplicaciones web en algunos casos no se requiere de toda la estructura robusta que ofrece un framework, razón por la cual surgió la necesidad de utilizar un micro-Framework, estas son versiones sumamente ligeras a las otras ya existentes (Rodríguez, 2007).

El micro-framework tiene las capacidades de realizar más de mil peticiones por segundo, viene ya pre configurada los principales módulos a utilizar lo que no sucede con los framework completos (Lewis, 2015). La ventaja de utilizar un micro-framework son: más sencillo, más liviano, una mayor velocidad de ejecución en las peticiones y con características más específicas y que puede generar micro servicios(Santos Hernández and Serrano Parreño, 2017).

Al usar un micro-framework se puede implementar los micros servicios en un sistema, lo que tienen menos funciones ya que se utilizan lo suficientemente necesario, sino que también llevan las ventajas de ser ligero, rápido y ágil. (Habib, 2015). Para lo cual Lumen es la solución perfecta para la construcción de micro-servicios basados en Laravel y REST APIS. De hecho, es uno de los micro-frameworks más rápidos disponible (Lumen, 2019). Lumen nos ayuda con la creación inmediata de servicios y el desarrollo de API's de gran rendimiento para nuestro proyecto, basados en Laravel (PHP), actualmente es una de las micro-estructura más rápidas y de fácil código que se encuentran disponibles (Rodríguez, 2007).

Scrum es una metodología ágil que ayuda a desarrollar un software minimizando sus riesgos durante su desarrollo y así poder obtener calidad hacer y control de los avances que se van ido desarrollando durante la jornada laboral. Las razón es que tiene beneficios en tres niveles, en el nivel plantilla permitiendo el trabajo en equipo esto incrementara la motivación a los miembros del equipo, nivel proceso y el nivel de resultados hacen las entregas de las iteraciones llamadas sprint en un tiempo mínimo de dos semanas (Querald, 2014).

Justificación Aplicativa

Con la finalidad de colaborar en la resolución del problema planteado del almacén se busca realizar el sistema propuesto ya que el sistema apoyara a la gestión de la información respecto a cada cliente registrado en el sistema, permitirá al cliente registrar un pedido de los artículos deseados. A los empleados del Almacén les permitirá manejar de modo más eficiente la información de los clientes, artículos disponibles, la lista de los clientes con créditos y su respectivo tiempo de pago posteriormente lista que servirá ya ser emitida al buró de crédito.

El dueño del Almacén podrá revisar la cantidad y el monto vendida durante el día, semana o al mes y así poder obtener las utilidades generadas durante el periodo o la fecha seleccionada. Para mejorar los procesos mencionados anteriormente el sistema contará con una base de datos en la cual se podrá almacenar la información perdurable en el tiempo para los empleados.

El sistema contendrá los siguientes módulos:

Módulo de Clientes.- el sistema presenta el ingreso con la información requerida de los mismos.

Módulo Venta a Contado.- Esta opción permitirá sin realizar un pedido una venta directa, especificando la cedula del cliente.

Módulo de crédito.- esto nos permitirá generar un venta a crédito especificando la el periodo de pagos y el tiempo.

Módulo de pedido.- permite realizar o reservar un producto que se encuentra en el almacén, la duración tendrá un máximo de 3 días, posterior a los 3 días se dará de baja ese pedido.

Reportes.- el administrador del sistema tendrá acceso a todos los reportes que genere el sistema para la toma de decisiones por parte del área administrativa del Almacén.

Módulo de Autenticación.- este nos permitirá crear un módulo seguridad de ingreso al sistema para los clientes como para los empleados del Almacén.

Respecto a las líneas de investigación de la EIS – ESPOCH –PNBV se encuentra: Dentro del

Programa para el desarrollo de aplicaciones de software, hardware y telecomunicaciones.

En la línea de investigación de la EIS está clasificado como **Proceso de Desarrollo de Software.**

Objetivos

Objetivo General

Desarrollar e Implementar el Sistema web para la automatización de compras y ventas del Almacén de muebles CACHA, analizando la eficiencia del software con la norma ISO 9126.

Objetivos Específicos

- Analizar la arquitectura y el funcionamiento del micro-framework Lumen.
- Aplicar la metodología ágil Scrum durante su desarrollo.
- Desarrollar el sistema con los módulos que sean necesarios para la automatización de los procesos en el almacén de muebles Cacha.

CAPÍTULO I

1. MARCO TEÓRICO

La finalidad de este capítulo es tener conocimientos claros como conceptos, características, funcionamiento y ventajas y desventajas de las herramientas y tecnologías que se van a utilizar.

1.1. Aplicaciones Web.

1.1.1. Definición.

Se llama aplicación web a cualquier aplicación que pueda acceder mediante una red (Internet) ,para su correcto funcionamiento utiliza un servidor web, el usuario para poder consumir estas aplicaciones debe acceder desde un ordenador mediante un browser (Maldonado Guerrero, 2016,p. 12).

1.1.2. Clasificación.

Las aplicaciones web según (Yeeply, 2018) se clasifican en:

Aplicaciones Web Estáticas: Este tipo de aplicación nos proporciona poca información y por lo general está desarrollado con HTML y CSS y no suelen cambiar frecuentemente. Para poder modificar la información contenida no es tan sencilla, para hacer se tiene que descargar y a posterior subir con los nuevos cambios realizados.

Aplicaciones Web Dinámicas: Son un poco más complejas su contenido cambia frecuentemente, cuentan con un administrador que pueden agregar, modificar y/o eliminar contenidos, para guardar esta información poseen bases de datos y son programados ya en un lenguaje de programación. Estas ya cuentan una autenticación de usuarios.

Tienda Virtual o Comercio Electrónico: A más de ser una aplicación dinámicas están se dedican a realizar un comercio digital ofertando productos o servicios, esas aplicaciones poseen un medio de pago como tarjetas, PayPal entre otros. A más de ofertar sus servicios cuentan con un panel de administrador que pueda gestionar sus servicios o Productos.

Portal de Aplicaciones Web: Esta aplicación en su página principal cuentan con accesos a diversas sesiones como un buscador, acceso a una categoría, Autenticación de usuarios.

Aplicaciones web Animadas: En estas aplicaciones sus contenidos se los presenta con unos efectos especiales (animaciones) sus diseños son actuales pero esto es un inconveniente para el posicionamiento web puesto que los buscadores no pueden interpretar correctamente.

Aplicaciones de gestor de contenidos: En este tipo de aplicaciones el contenido cambia continuamente y para lo cual se puede instalar un CMS (Sistema de Gestión de Contenidos) como WordPress, Joomla o Drupal.

1.2. API

1.2.1. Definición

Una API su acrónimo (Application Programming Interface) es un servicio que contribuye un software de ciertas funcionalidades y que estas puedan ser utilizadas por otro software, estas son utilizadas de enlace entre un software con otro por esta razón esta comunicación de los APIS es de máquina a máquina (software-to-software) (Plaza Estévez, Ramírez Lamela and Acosta Morales, 2015).

1.2.2. Clasificación

Según (Alvarado, 2018) la clasificación de los API son las siguientes:

- API de Servicio Web.
- API de Código Fuente.
- API Heredadas.
- API de Producción.
- API de Desarrollo.

Los API de Servicio Web están basados en los estándares SOAP, XML-RPC, JSON-RPC y REST estas son empleadas en las arquitecturas orientado a servicios.

1.2.3. Ventaja de REST Sobre SOAP

En la actualidad en el mundo de los servicios web de las API las tecnologías que más se han utilizado son las de SOAP y REST. A partir del año 2000 la tecnología REST han evolucionado y en la actualidad muchos de las empresas apuestan por esta arquitectura, por los cual (Cano, 2017) menciona algunas de las principales ventajas del REST con respecto al SOAP.

- Los REST son diseñados para que se puedan ser usados en el internet.
- SOAP utiliza para sus mensajes XML lo cual hace que el tamaño sea mayor en cambio REST utiliza formatos Pequeños, por esto REST es más rápido a la hora de realizar peticiones.
- Los Servicios SOAP en comparación con los de REST son más difíciles de escalar. Las grandes empresas del mundo empiezan a utilizar los servicios web REST.
- Los servicios SOAP consumen más ancho de banda con respecto a los de REST.

1.3. API REST

1.3.1. Definición

Un REST (Representational State Transfer) “es conjunto de Restricciones con las que podemos crear un estilo de arquitectura software, la cual podremos usar para crear aplicaciones web respetando HTTP”(Ordoñez, 2018).

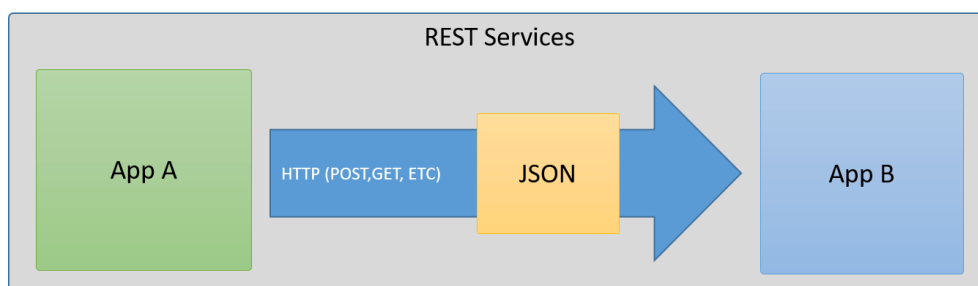


Figura 1-1: Solicitud de los API

Fuente: (Blancarte, 2017)

Los servicios REST empieza por una solicitud que realiza una aplicación A esta debe utilizar los métodos http para su comunicación, estas son receptadas por otra aplicación B lo cual procesa toda la información necesario y emite una respuesta con un estado en el protocolo http. Http para su comunicación utilizan verbos

Tabla 1-1: Tabla de los métodos http (verbos)

Método	Descripción
Get	Solicita Datos
Post	Registra nuevos datos en la base de datos
Delete	Elimina los Datos de la base de datos
Put	Actualiza los datos y tienen cambios en la base de datos

Realizado por: Hipo, Kleber, 2019

El cliente es el cual empieza con una petición utilizando uno de los métodos o verbos de http. Esta es receptada por el servidor quien las procesa los datos y emite una respuesta indicado si se acepta o rechazada la petición realizada, esta para su verificación es emitida en un formato de código de estados http.

Haldar (Haldar, 2017) en su artículo menciona que los principales a los tipos de estados, los códigos 2xx representan el estado de éxito, 3xx de re direccionamiento, 4xx error del cliente y 5xx que son error del servidor. A continuación se muestra los principales estados y su representación.

- 200 Ok: Representa un éxito para cualquiera de los métodos http.
- 201 Created: Se ha creado un recurso con éxito.
- 204 No Content: Este estado representa que la petición se realizó con éxito pero no ha devuelto ningún contenido.
- 304 Not Modified: este estado indica que el cliente ya cuenta con la información en la cache.
- 400 Bad Request: Indica que la solicitud que envió el cliente no pudo ser procesada.
- 401 Unauthorized: No tiene los permisos para acceder a este recurso.
- 404 Not Found: Indica que este recurso no está disponible
- 500 Internal Server Error: Indica que la petición es correcta pero que el servidor está confundido.
- 503 Service Unavailable: El servidor está inactivo.

1.3.2. Características de la Arquitectura API REST.

Las características para cumplir una arquitectura REST Fielding deben cumplir las siguientes condiciones

- Cliente-Servidor
- Staleless (Sin Estado)
- Cacheable
- Interfaz de Uniforme
- Sistema de Capas.

Cliente-Servidor: Esta es una de las arquitecturas que tanto el cliente y el servidor deben estar separados, esto permite que una aplicación sea escalable razón por la cual el cliente (FrontEnd) y la del servidor (BackEnd) son desarrollados desacoplados para posteriormente ser unido por una API.

BackEnd es la parte del servidor donde se procesa la petición que realiza un cliente con la ayuda de un lenguaje de programación (Php, Java) y retorna los resultado (Fernandes Fernandes, 2015).

FrontEnd: Es el lugar donde se ejecuta la aplicación en si un Navegador donde el cliente realiza las peticiones y recibe el resultado de la misma, este ultimo los procesa para mostrar al usuario (HTML, Css, JavaScript) (Fernandes Fernandes, 2015).

Peticiones Stateless (Sin Estado): Este es uno de los puntos más importantes donde que un servidor no guarda ningún estado como variables de sesión por esta razón las peticiones se realizan sin ningún estado.

Cacheable: en la web los clientes pueden cachear las respuestas de un servidor. Los datos pueden ser utilizadas si en caso que ya realizo una petición anteriormente.

1.3.3. Modelo de Autenticación

Uno de los principales modelos de autenticación es el mediante Token.

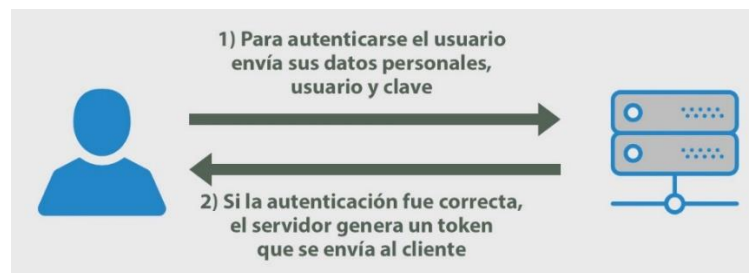


Figura 2-1: Autenticación para API REST

Fuente: (Baquero Garcia, 2018)

En las aplicaciones tradicionales por lo general se usan variables de sesión, pero en las aplicaciones API REST la autenticación es diferente. Como uno de las características de esta arquitectura es el del cliente-servidor y el sin estado, este proceso empieza por un cliente (FrontEnd) donde envía un usuario y la contraseña mediante la utilización de los servicios API REST, estas son receptadas por un servidor(BackEnd) donde los datos son validados y en caso que fuese éxito retorna un token al cliente utilizando los estados Http (Baquero Garcia, 2018).

Los tokens son almacenados en el lado del cliente (browser), lo cual puede ser almacenado en una Sesión Storage o Local Storage por esta razón se dice aplicación sin estado además se puede realizar que los tokens sean válidos hasta un cierto tiempo para luego ser caducados (Azaustre, 2015).



Figura 3-1: Acceso a las páginas por token

Fuente: (Baquero Garcia, 2018)

Cuanto un token se haya almacenado el cliente en las futuras peticiones deberá incluir en la cabecera http el token de autenticación, la cual es receptada por un servidor y posteriormente descriptada y verifica, si existe o no, este token en caso que no existe retorna con un estado http al cliente y si fuese el caso que existiera permite acceder al recurso solicitada (Baquero Garcia, 2018).

1.4. Lenguajes de Programación Orientados a la Web

1.4.1. Definición.

Un lenguaje de programación es un conjunto de reglas, palabras o símbolos, estas son asociadas a un programa y esta es llevada a un ordenador para que pueda realizar los cálculos necesarios. Cada lenguaje de programación posee su propia sintaxis y estas son usadas para formar un programa, estas no son aplicaciones sino una herramienta que permite construir aplicaciones dependiendo de las necesidades que tenga (Almagro, 2012, p. 2).

1.4.2. Comparación lenguaje de programación orientado a la Web

Tabla 2-1: Tabla comparativa de los diferentes Lenguajes de Programación.

Lenguaje	Características	Fortalezas	Debilidades
PHP	<ul style="list-style-type: none"> * Utilizado para generar paginas Dinámicas. * Se ejecuta en el Servidor. * Nuevas versiones permiten la programación de objetos. * No se necesita instalar php en el cliente. 	<ul style="list-style-type: none"> * Sintaxis similar a otros lenguajes. * Libre y gratuito. * Puede combinarse con HTML. * Posee una extensa documentación. * Multiplataforma. * Manejo de base de datos. * Posee una gran variedad de Frameworks para su desarrollo. * Lenguaje muy popular y por ente tiene una extensa comunidad. No requiere definición de variables * Existe una gran variedad de alojamientos en la web. 	<ul style="list-style-type: none"> * La Programación Orientado a objetos es deficiente en las aplicaciones grandes. * Necesita de un servidor para funcionar. * Toda la información se los realiza en el lado del servidor.
	<ul style="list-style-type: none"> * Es Orientado Objetos. * Multiplataforma 	<ul style="list-style-type: none"> * Permite realizar modularización por ser Orientado a Objetos. 	<ul style="list-style-type: none"> * Relativamente lento por ser un lenguaje interpretado con respeto a

JAVA		* Permite la creación de aplicaciones de escritorio y soporte para aplicaciones web y móvil.	los otros lenguajes de programación.
JSP (JAVA SERVER PAGES)	* Lenguaje para la creación de sitios web dinámicos. * Necesita de un servidor tomcat. * Motor basado en servlets de java * Multiplataforma	* Código bien Estructurado. * Integridad con módulos de java. * Ejecución rápida de servlets. La parte dinámica está escrita en java.	* Complejidad de aprendizaje.

Fuente: (Rosada, 2015)

Realizado por: Hipo, Kleber, 2019

Después de analizar los principales lenguajes de programación orientado a web se decidió utilizar el lenguaje de programación php por razones de facilidad de uso, ser un lenguaje muy popular y por ende tiene una comunidad muy extensa y para realizar su despliegue cuenta en el mercado una gran variedad de hosting.

1.5. Lenguaje Programación PHP.

1.5.1. Definición.

Es una lenguaje programación su acrónimo PHP significa “Pre-Procesador Hipertexto” se ejecuta del lado del servidor. Es un lenguaje de intérprete libre usado para la creación de sitios web dinámicos en la World Wide Web, se caracteriza por su potencia, versatilidad, robustez y modularidad además es de los primeros lenguajes en permitir incorporar HTML en su contenido (Arias, 2017,p. 13).

Se puede instalar en la mayoría de los sistemas operativos, la mayoría de las aplicaciones están programadas en este lenguaje de programación como: Joomla, Drupal, WordPress, MediaWike, Magento entre otros

1.5.2. Características

En la documentación oficial de php menciona que las principales características las cuales son:

- Amplio soporte de conexiones a las diferentes bases de datos como Mysql, Postgresql, Sqlite entre otros.

- Lenguaje de script de código abierto.
- Soporte para la instalación de los objetos de Java y emplearlos de forma transparente como objetos de PHP.
- Libertad de elegir el sistema operativo y el servidor web.
- Soporta programación por procedimientos o la programación Orientada a objetos.
- Capacidad de crear imágenes, ficheros PDF e incluso películas Flash (usando libswf y Ming)

1.5.3. *Ventajas y Desventajas*

Las principales ventajas y desventajas del lenguaje programación php son las siguientes.

Tabla 3-1: Tabla de ventajas y desventajas de php

Ventajas	Desventajas
* Lenguaje de Multiplataforma (Aguayo, 2016). * El código php no se visualiza al cliente, el servidor al momento de ejecutar envía el resultado en HTML al navegador (Aguayo, 2016). * Está en constante desarrollo y puede soportar una gran cantidad de funcionalidades (Perez, 2018). * Existe una extensa documentación en el Internet por lo cual será más fácil resolver los problemas que tenga durante su desarrollo (Perez, 2018).	*.Por ser un lenguaje interpretado por parte del servidor puede ser colapsado al momento de que las peticiones aumente extremadamente.

Realizado Por: Hipo, Kleber ,2019

1.6. Framework.

1.6.1. *Definición.*

Se define un framework como un esqueleto o un patrón que sirve para el desarrollo de una aplicación, contiene un conjunto de archivos en la cual consta de toda una estructura necesaria para el desarrollo de una aplicación (Sierra *et al.*, 2004,p.3)

Al momento de realizar una aplicación empezamos por codificar desde cero, esto no sería lo más adecuado, para esto se tiene los frameworks que nos ayudan a optimizar el trabajo y tiempo.

1.6.2. *Ventajas y desventajas*

Según (Ortiz, 2018) las ventajas y desventajas de utilizar un framework son las siguientes:

Tabla 4-1: Tabla de las ventajas y desventajas de usar un framework

Ventajas	Desventajas
<p>* Desarrollo Rápido: La mayoría de los framework proporcionan la arquitectura modelo, vista y controlador, esta ayuda en vez de planificar una arquitectura empezar en el desarrollo. Muchos vienen con herramientas en línea la cual facilita a genera código y optimizar aún más el tiempo de desarrollo.</p> <p>* Hace que una aplicación sea más segura, las soluciones incluidas en un framework son validadas por múltiples desarrolladores que utilizan el framework, los framework son de código abierto por lo cual si se pasa por alto algunas seguridades son corregidas rápidamente</p> <p>* Mantenimiento más fácil al tener una estructura de programación intuitiva, la mayoría de frameworks están orientado a objetos por lo cual facilita la realización del mantenimiento como leer, mantener y desarrollar</p> <p>* Trabajo en equipo más fuerte: ayuda a los nuevos desarrolladores a comprender y empezar a trabajar en un proyecto en ejecución.</p> <p>* Dispone de una comunidad muy activa en las cuales se tienen grupos, libros materiales audiovisuales entre otros.</p>	<p>*.Ejecución Lenta los framework siempre agrega complejidad y sobrecarga en forma de clases y bibliotecas cargadas antes de que su código sea llamado.</p> <p>* Un framework requiere de un tiempo para poder aprender y dominarla.</p> <p>* Posee una dependencia externa para su código de aplicación controlado por alguien que no sea el que utiliza o el equipo.</p>

Realizado Por: Hipo, Kleber ,2019

1.7. Micro-Framework

1.7.1. Definición

Los Micro-Frameworks son una sub-clasificación de los frameworks popular de aplicaciones web, generalmente se derivan de los frameworks Completos o llamados frameworks full Stack. Son útiles para aplicaciones pequeñas que envían y reciben solicitudes HTTP, en rutan solicitudes HTTP a un controlador específico y envían y devuelven la respuesta del controlador. Se usan comúnmente para crear aplicaciones API (Nawaz, 2016).

1.7.2. Tipos de Micro-Framework

En la actualidad en el mercado se puede encontrar una gran cantidad de micro-framework:

- Lumen
- Sílex

- Slim
- Phalcon
- Wave Framework

Los frameworks completos ayudan a los programadores durante todo el desarrollo desde la creación de las interfaces hasta el almacenamiento de los datos en un gestor de base de datos, de ese grupo de frameworks, si el framework y las bibliotecas son más pequeñas que 5000 líneas de código, se les llama como un micro-Framework (Palupi, 2019) .

Los micro-framework omiten muchos de los componentes de una configuración de un framework completo como:

- Motor de plantillas web
- Validación de entrada
- Abstracción de la base de datos
- Roles, cuentas y autenticación.

En un estudio realizado por (Tarvainen, 2016) realiza las pruebas a los diferentes micro-frameworks entre los cuales se encuentra Lumen, Silix, Slim la cual consta en realizar en serializar un objeto JSON establecer Cabeceras y enviar salidas las cuales muestran que lumen puede procesar 2254 solicitudes por segundo , Sílex 1266 solicitudes por segundo y Slim alcanzado 1638 solicitudes por segundo. Puesto que Taylor Otwell en una entrevista menciona que el Mirco-Framework lumen es el más rápido dentro de este grupo, en caso que no fuese así no lo hubiere lanzado al mercado.

1.8. Lumen

1.8.1. Definición.

Lumen es un micro-framework para PHP creado por Taylor Otwell creador de Laravel, comparte algunos de los componentes de su “hermano mayor” Laravel, pero Lumen es una versión más liviana de Laravel y orientado más a la creación de APIS y micro servicios, aunque de igual manera se puede usar para la creación de sitios web (Palacios, 2015).

1.8.2. Arquitectura

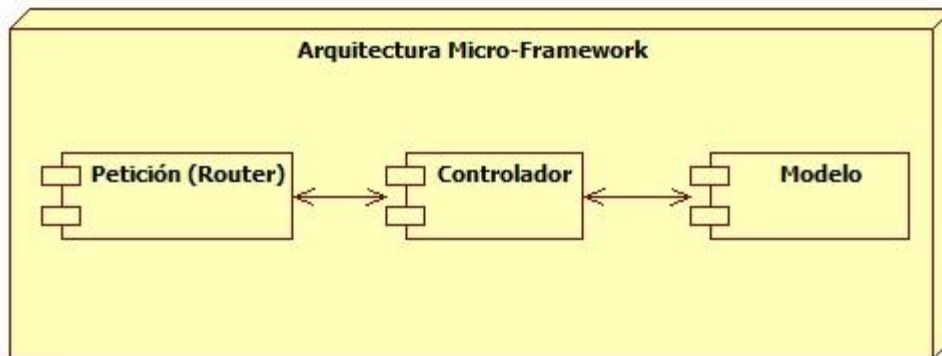


Figura 4-2: Arquitectura Micro-Framework Lumen
Realizado Por: Hipo, Kleber ,2019

Como Lumen es un micro-framework creado por el mismo desarrollador de Laravel hereda algunas características del framework completo, entre esos esta su arquitectura, Laravel posee el patrón de diseño de Modelo, Vista, Controlador (MVC) y una petición (Cárdenas Paladines, 2015, p.8), Lumen de igual forma hereda la misma estructura solo con una simple modificación excluyendo la capa de la vista esta no es necesario al igual que las plantillas blade, ya que en vez de esta capa retorna una respuesta de texto plano por tal razón se puede llamar como un micro-framework de modelo controlador.

En ninguna parte se menciona que una vista debe ser exactamente una interfaz gráfica, las vistas se puede tomar como un texto plano (JSON,XML), como los datos que se devuelven mediante la petición es un texto plano se puede considerar de la misma forma como una vista (Arriaga, 2017)

1.8.3. Funcionamiento

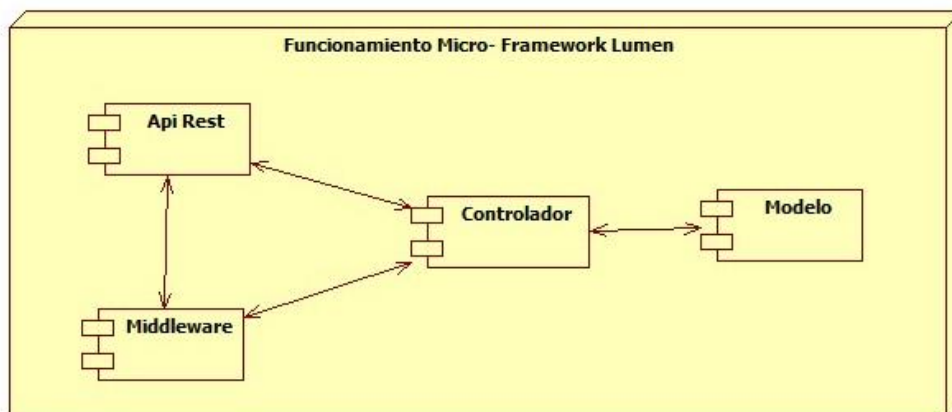


Figura 5-2: Funcionamiento del Micro-Framework Lumen
Realizado Por: Hipo, Kleber ,2019

Lumen para su correcto funcionamiento posee capas, las cuales deben permitir para poder acceder al recurso que se solicite, una petición realizada por el cliente llega a la capa de los API REST los cuales tienen un enrutador con la ruta de acceso al recurso, la cual es asociado con un controlador, para que la solicitud pueda acceder al recurso del controlador, existe una capa intermedia llamada middleware.

La Capa del Middleware son aquellos que permiten o niega la solicitud de acceso al recurso, a su vez existen rutas que no necesariamente deben acceder por esta capa ya que tienen un acceso directo al controlador como por ejemplo para la autenticación de usuarios que no necesitaría validar datos, las rutas que deben pasar por la capa anteriormente mencionada realizan la validación, algunas rutas que necesariamente deben verificar que el usuario que este accediendo este autenticado, esto se lo realiza mediante la utilización de tokens, en cada una de las peticiones que se realiza en la cabecera (header) de la petición HTTP deben siempre deben enviar su token del usuario esta a su vez verifica en la base de datos de que el dato enviado pertenezca a un usuario registrado si existe permite que el acceso a la siguiente capa que son los controladores caso contrario rechaza la petición con un estado y una respuesta con el error generado.

Cabe mencionar que los API REST no tiene problemas cuando se realizan solicitudes que estén en el mismo dominio, cuando realiza la petición de diferentes dominios se debe utilizar los CORS (Intercambio de Recursos de Origen Cruzado) los cuales nos permite acceder de dominios distintos, esto ayuda en la seguridad de la información (Chiemela, 2017). En si este es un Middleware de igual forma, se debe registrar en el archivo app.php.

En los modelos es en donde se accede a la base datos para lo cual Lumen utilizan los paquete de Eloquent, estos permite el acceso para la gestión de la información. En este micro-framework esto viene por defecto desactivado para lo cual en el archivo app.php se debe des comentar las fachadas.

```
Dotenv :: load (__ DIR __. '/ .. /');  
...  
$ app-> withFacades ();  
$ app-> withEloquent ();
```

Figura 6-2: Activación de Eloquent Lumen

Fuente: (Houston, 2015)

1.8.4. Lumen Vs Laravel.

Una de las cosas que se tiene en común es que ambos son creados por Taylor Otwell, como laravel sigue el patrón MVC lumen igual sigue la misma estructura Modelo, Controlador esta es una de las diferencias ya que no posee la capa de la vista ya que no lo necesita, más estos micro-

frameworks son para los micro-servicios o API REST. Laravel utiliza un enrutador de symfony mientras que lumen posee otro tipo de enrutador lo cual sea más rápido que su padre.

Lumen sacrifica algunas características para ser flexible como bootstrap de autenticación, sesiones, vistas, pero si en caso que fuese necesario se podrá ir agregando. No poseen archivos de configuración separados como en el framework completo todo su configuración se lo realiza en el archivo .env.

Las razones para escoger este micro-Framework es: cuando un proyecto crece se podrá migrar fácilmente un framework completo como es Laravel, Laravel cuentan con una comunidad muy extensa como lumen utiliza una gran cantidad de paquetes de su padre se puede encontrar ayuda en ellos, y la última es su rendimiento a la hora de procesar solicitudes (Ahmad, 2017, p. 103).

En la **tabla 5-1** se puede visualizar las principales diferencias entre el Framework Laravel con el Micro Framework Lumen

Tabla 5-1: Comparación de Laravel vs Lumen

Características	Laravel	Lumen
Uso	Es un Framework completo para el desarrollo web bajo el lenguaje de programación php.	Es la versión minificado del Framework Lumen para la creación de micro servicios y APIS.
Plataforma	Es compatible con todos los sistemas operativos.	Como es parte de Laravel es compatible de igual forma con todos los sistemas Operativos.
Peticiones	Procesa menos solicitudes por segundo en comparación con Lumen.	Procesa más solicitudes por segundo.
Comunidad	Tiene una comunidad más pequeña en comparación con Angular.	Tiene una comunidad más grande y muchos contribuyentes individuales.
Licencia	Fue licenciado bajo licencia MIT.	También fue autorizado bajo MIT y se derivó de las características de Laravel.
Tiempo de respuesta	El tiempo de respuesta es alto para Laravel en comparación con Lumen.	El tiempo de respuesta es menor en comparación con Laravel.
Velocidad	Laravel trabaja a menor velocidad en comparación con Lumen.	Lumen como procesa más solicitudes por segundo es muy Veloz.

Realizado por: Kleber Hipo

La **tabla 5-1** muestra la comparación entre el micro-Framework Lumen y el Framework completo Laravel donde se puede apreciar que los dos pueden crear los APIS REST, se ejecutan en cualquier sistema operativo, la principal diferencia son los tiempos de respuesta. Lumen posee una velocidad mayor en comparación con Laravel y por ende los tiempos de respuesta son menores con el Framework completo.

1.9. Mysql

1.9.1. Definición.

Mysql es un sistema de gestor de base de datos de código abierto y basado en el lenguaje de consultas SQL. su uso es en todas las aplicaciones generalmente en las web destaca por su gran

adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como Php, Perl y Java y además puede ejecutarse en los diferentes sistemas operativos como Linux, Windows Unix (Rouse, 2015)

1.9.2. Características

Según Perez Garcia (Pérez García, 2007,p.14) menciona las siguientes características de Mysql.

- Velocidad. MySQL es rápido.
- Facilidad de uso.
- Es un sistema de base de datos de alto rendimiento pero relativamente simple y es mucho menos complejo de configurar y administrar que sistemas más grandes.
- Coste. Es gratuito.
- Capacidad de gestión de lenguajes de consulta. MySQL comprende SQL, el lenguaje elegido para todos los sistemas de bases de datos modernos.
- Capacidad. Pueden conectarse muchos clientes simultáneamente al servidor. Los clientes pueden utilizar varias bases de datos simultáneamente. Además, está disponible una amplia variedad de interfaces de programación para lenguajes como C, Perl, Java, PHP y Python.
- Conectividad y seguridad. MySQL está completamente preparado para el trabajo en red y las bases de datos pueden ser accedidas desde cualquier lugar de Internet. Dispone de control de acceso. - Portabilidad. MySQL se puede utilizar en una gran cantidad de sistemas Unix diferentes así como bajo Microsoft Windows.
- Distribución abierta. Puede obtener y modificar el código fuente de MySQL

1.10. Aplicación de Página Única (SPA)

Un SPA es una aplicación que actúa en el navegador, la cual no necesita recargar la página durante su uso. Al igual que todas las aplicaciones, está pensado para ayudar al usuario a completar una tarea, como "escribir un documento" o "administrar un servidor web". Podemos pensar en un SPA como un gran cliente que se carga desde un servidor web (Mikowski, Powell and Benson, 2014,p.4).



Figura 7-1: Web Tradicional vs SPA
Fuente: (De La Fuente, 2016).

En la **Figura 7-1** se puede visualizar que las aplicaciones tradicionales realizan una petición al servidor, esta es receptada y en ella se obtiene el código HTML para posterior retornar al navegador, el cual es quien las procesa y visualiza al cliente refrescando toda la página, esta acción se repite en todas las peticiones lo cual genera largos tiempos de espera, en las aplicaciones SPA esta tecnología cambia puesto que en la primera llamada los archivos HTML, CSS Y JavaScript son cargados por una única vez en las posteriores llamadas cuando se invoca a un recurso esta se modifica dinámicamente, esto dependerá de la interacción del usuario con el sistema; la interacción del cliente con el servidor se realiza mediante las API que son devueltos en un texto plano como JSON, aquellas son consumidas con el lenguaje JavaScript y Ajax en el lado del cliente (De La Fuente, 2016 , p.29).

Para decir que es una aplicación de SPA según (Brehm, 2013) deben solucionar los siguientes:

- **Enrutamiento:** Permita a los usuarios navegar mediante Urls, estas deben ser enlazadas con una vista para generar cambios en la pantalla.
- **Interfaz de Usuario Dinámica:** Aquellas que permitan actualizar la información con el uso de las APIS
- **Acceso a Datos:** Permita acceder a los datos almacenados en el lado del cliente, el navegador nos proporciona alguno de ellos como LocalStorage o SesionStorage.

Entre las principales tecnologías para el desarrollo de esta arquitectura son HTML, CSS, JavaScript de igual manera en el mercado se puede encontrar como React, Vue, Angular, Polimer,

Ember; todas estas librerías o Frameworks de JavaScript, entre las más usadas las tres primeras, en nuestro caso se utilizara Vue por gusto del desarrollador.

Algunas aplicaciones con esta arquitectura que se pueden encontrar en la web son: YouTube, Gmail, Netflix, entre otros.

1.11. VUE.JS

Es un Framework progresivo el cual nos ayuda a construir interfaces de usuarios amigables, simple de utilizar y/o a su vez permite integrar con proyectos o con librerías existentes, de la misma forma soporta aplicación de una sola página SPA (Single Page Application) (Vuejs).

Características

Según (Rosa Moncayo, 2017) menciona las siguientes características:

- **Accesible.**
- **Versátil:** Su núcleo es bastante pequeño y se escala a través de plugins, con lo cual existe mucho muchas librerías para
- Escalable por el mismo tema de la versatilidad.
- **Reactivo:**
- **Optimizado:** Su core ocupa **74KB**, es sumamente liviano.
- **Comunidad:** Posee una comunidad muy grande con más 66500 estrellas en GitHub y 130 personas contribuyendo al core cada día.
- **Licencia MIT:** se publicó bajo el amparo de esta licencia.

Para su uso se puede realizar como si fuera una librería agregando mediante las etiquetas Script, cuando se desea realizar un proyecto mediano se puede utilizar el gestor de paquetes de node.js NPM.

Los principales ecosistemas que se pueden abarcar este framework con la ayuda de los plugins:

- **Vue-Router:** Gestionar las Urls.
- **Conectar a servicios externos:** mediante la utilización de Vue-axios ya que esta es la recomendada por los desarrolladores de vue.js
- **Test:** realizar pruebas unitarias y funcionales
- **Vuex:** gestionar los estados en una aplicación con el flujo de los datos.

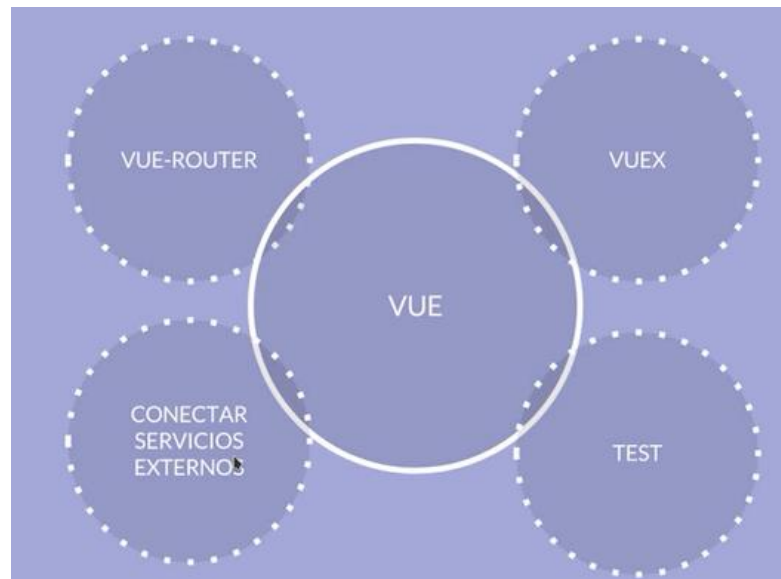


Figura 8-1: Ecosistema de Vue
Fuente: (Rosa Moncayo, 2017)

1.12. Metodología Scrum

Scrum es un framework el cual permite trabajar con un equipo de forma colaborativa para obtener un mejor resultado en el Proyecto en un conjunto de iteraciones (Lara, 2016). Scrum divide un proceso y por lo cual se debe responder a las siguientes preguntas:

- El ¿Quién? y el ¿Qué?: Identifica los roles de cada uno de los miembros del equipo y define su responsabilidad en el proyecto.
- El ¿Donde? y el ¿Cuándo?: que representan el Sprint
- El ¿Por Qué? y el ¿Cómo?: representan las herramientas que utilizan los miembros de Scrum

1.12.1. Roles en Scrum

Se identifica a entre los principales roles son los siguientes

- El Product Owner/Dueño del producto es la “voz del cliente” y el responsable de desarrollar, mantener y priorizar las tareas en el backlog (Schwaber, 2017).
- El Scrum Master es responsable de asegurarse que el trabajo del equipo vaya bien siguiendo las bases de Scrum. Además, se encarga de remover cualquier obstáculo que pueda encontrar el equipo de desarrollo (Schwaber, 2017).
- Los Development Team Members o llamados Miembros del Equipo de desarrollo son los encargados de desarrollar o escribir y probar el código (Schwaber, 2017).

1.12.2. El Sprint - ¿Dónde? ¿Cuándo?

Deberá responder a las preguntas de ¿Dónde? Y ¿Cuándo?

Sprint es la unidad básica de trabajo para un equipo Scrum. Es una simple iteración llevada a cabo por los miembros del equipo. Un equipo puede completar varios sprints durante el desarrollo del proyecto. Un Sprint inicia con un equipo con un conjunto de tareas y finaliza con la demostración de un entregable. El tiempo mínimo para un Sprint es de una semana y el máximo es de 4 semanas (Lara, 2016).

Backlog de Producto/Product Backlog

Esto puede referirse a todo elemento que sea parte del proyecto. Puede ser un bug, una referencia o parte de un requerimiento. Brindan información muy general del proyecto y muchas veces no son tomados como requerimientos oficiales.

Historias de Usuario /User Stories

Es un elemento especial del product Backlog. Son llamados Historias porque en ellos se proporciona información sobre cómo debe ser el comportamiento del requerimiento que se está trabajando. De igual manera, proporciona información directa del cliente en caso de existir algún cambio. Generalmente estos sí son tomados como requerimientos oficiales.

Backlog del Sprint/Sprint Backlog

Es el conjunto de elementos tomados del Product Backlog que fueron priorizados, medidos y aceptados en las reuniones de Sprint Planning. Estos, en conjunto con sus respectivos User Stories, forman oficialmente los requerimientos a elaborar en cada uno de los Sprints que tendrá el proyecto (Lara, 2016).

1.13. Calidad de Software

“La totalidad de características de un producto de software que tienen como habilidad, satisfacer necesidades explícitas o implícitas” (Prieto, 2008).

Se define como la calidad de software cuando un componente o procesos cumplen los requerimientos especificados de acuerdo a las necesidades de un cliente. (López Echeverry, 2008)

1.13.1. Estándar ISO 9126

La ISO, bajo la norma ISO-9126, ha establecido un estándar internacional para la evaluación de la calidad de productos de software el cual fue publicado en 1992 con el nombre de “Information technology –Software product evaluation: Quality characteristics and guidelines for their use”, en el cual se establecen las características de calidad para productos de software (Abud Figueroa, 2012,p.1).

Según Esta parte de la ISO 9126 describe el modelo de calidad del producto de software. La primera parte del modelo especifica 6 características de calidad interna y externa, las cuales están divididas en sub-características, son manifestadas externamente cuando el software es utilizado como parte de un sistema, y son un resultado de atributos internos del software (Scalone, 2006) .

1.13.2. Eficiencia

La capacidad que tiene un producto de software para proveer el desempeño apropiado relacionado a la cantidad de recursos usados, bajo condiciones determinadas. Está compuesta por las siguientes sub-características: comportamiento en el tiempo, utilización de recursos (Ruiz, Peña and Arturo , Castro, 2006,p.27)

Según Pressman menciona que la eficiencia es “Grado en el que el software emplea óptimamente los recursos del sistema” (Pressman, 2010,p.343).

La eficiencia según (Pressman, 2010,p.343) cuenta con dos características las cuales son:

- Comportamiento del tiempo
- Comportamiento de los recursos

Comportamiento con respecto al Tiempo. Atributos del software relativos a los tiempos de respuesta y de procesamiento de los datos (Abud Figueroa, 2012,p.2)

Comportamiento con respecto a Recursos. Atributos del software relativos a la cantidad de recursos usados y la duración de su uso en la realización de sus funciones (Abud Figueroa, 2012,p.2)

CAPITULO II

2. MARCO METODOLÓGICO

2.1 Introducción

Para que un proyecto sea exitoso en la actualidad se debe considerar como la base fundamental la metodología de desarrollo, por lo cual en el presente capítulo se indica los pasos que se ejecuta durante el desarrollo del sistema web de gestión compras y ventas para el Almacén de muebles “Cacha” de la ciudad de Riobamba, para lo cual se aplica la metodología ágil SCRUM, a continuación se detalla cada una de las actividades para obtener un producto de calidad.

2.2 Tipo de Investigación

Para el desarrollo del presente proyecto se realiza una investigación aplicada, por medio de la cual permite utilizar herramientas tecnológicas para la solución de un problema, permitiendo poner en práctica los conocimientos que se ha adquirido durante la vida universitaria.

2.3 Técnicas

Las técnicas empleadas para el desarrollo son entrevista y la observación. La Entrevista se emplea al propietario del almacén el Sr. Mario Morocho para obtener las necesidades que poseen en la actualidad y la forma de cómo lleva el control de la información de los productos hasta la fecha. La observación, por medio de esta técnica se conoce los procesos que se necesitan ser automatizados, además se procede a realizar la revisión de los documentos del control de los créditos las cuales son: tarjeta Kardex, recibo de pagos; con esta información se obtiene de un enfoque claro del sistema a desarrollar.

2.4 Aplicación de la Metodología Ágil Scrum

2.4.1 Estudio Preliminar

El almacén de muebles “CACHA” está ubicado en la ciudad de Riobamba en las Calles Juan Montalvo y Colombia (Esquina) la cual se dedica a ofrecer una gran variedad de muebles a la ciudadanía como son: camas, colchones, comedores, semaneros, roperos, muebles de sala, cocina entre otros, la persona que está a cargo de esta empresa es el Sr. Mario Morocho.

La empresa en la actualidad no dispone de un sistema que les pueda ayudar a llevar un control detallado de las ventas a créditos y contado, registro de los créditos que están retrasado y por ende la cantidad de mora que se debe cancelar, el cobro de interés dependiendo de cuánto tiempo ha consumido su capital, dichos procesos son llevados de una forma manual en una tarjeta de crédito, cuando un cliente se acerca a cancelar una cuota, se realiza un cálculo manual de las moras en caso que este retrasado en el pago, además no disponen de un control de cuáles son los productos y cantidad que disponen, tampoco con un registro de cuáles son los productos que se compraron, no posee un inventario mensual de los productos, todo esto dificulta para agilizar en brindar una información severa y certera. Para agilizar y optimizar los procesos de la empresa es necesario la automatización de los procesos de las compras y ventas con un sistema informático dependiendo de los requerimientos presentados el negocio.

2.4.2 Fase planificación

En esta fase se realiza una planificación de acuerdo a los requerimientos presentados por la empresa y estas son desarrolladas de acuerdo a la prioridad del negocio además los requerimientos se agrupan en un tiempo de entrega los cuales se llaman Sprint, en esta planificación se toma en cuenta de igual forma a las metáforas las cual no son solicitadas por el cliente pero que son necesarias para el desarrollo del sistema. La planificación se los representa en un diagrama de GANTT. **ANEXO A.**

2.4.2.1 Requerimientos (Product backlog)

Para el Product backlog, se detalla el conjunto de requerimientos como historias de usuario, para cada miembro del equipo se define un número de actividades a realizar, de igual manera se determina un número de iteraciones (Sprints) del proyecto en las que tiene que trabajar.

Para realizar la estimación de cada tarea se utiliza el método de la talla de camiseta o T-shirt. Las estimadas son S, M, L y XL, como se lo presenta en la **tabla 1-2.**

Tabla 1-2: Tabla de puntos estimados por el método T-Shirt

Talla	Puntos estimados	Horas de trabajo
XS	8	8
S	16	16
M	20	20
L	32	40
XL	40	80

Realizado por: Kleber Hipo, 2019

Se toma como 8 horas laborables, para cada punto estimado representa a una hora de trabajo del desarrollador, las tallas L representan 40 puntos estimados que son 40 horas laborables toda una semana, las talla M representa 32 puntos estimados que es de 4 días, la XM representa 20 puntos

estimados que es de 3 días laborables, la S de 16 puntos estimados que es 2 días laborables y la talla XS de 8 puntos estimados siendo un día laborable.

Cada iteración o llamado en Scrum como Sprint, cuenta con un número de historias técnicas y de usuario, estas actividades se desarrolla en orden, cada sprint tiene una duración de 3 semanas con 120 puntos estimados lo que representa 120 horas laborables.

Los requerimientos son las necesidades que tiene el cliente las cuales se obtiene en las reuniones con el mismo y estas es priorizada dependiendo de acuerdo a las necesidades, las cuales son utilizadas por la herramienta Product Backlog en la cual donde a cada uno se estima el tiempo que se tiene para completar dicha tarea, a continuación se detalla el listado de requerimientos.

Tabla 2-2: Tabla Product Backlog - Historias de Usuario

ID	TAREA	PRIORIDAD	ESTIMACIÓN
HU-01	Autenticación de usuarios	Alta	16
HU-02	Registrar datos de la Empresa	Alta	16
HU-03	Modificar Datos Empresa	Medio	16
HU-04	Registrar categoría	Alta	8
HU-05	Modificar Categoría	Media	8
HU-06	Eliminar una categoría	Baja	8
HU-07	Registrar un color	Media	8
HU-08	Modificar un color	Media	8
HU-09	Eliminar Color	Baja	8
HU-10	Registrar Empleado	Alta	20
HU-11	Modificar empleado	Media	20
HU-12	Eliminar Empleado	Baja	8
HU-13	Registrar Sucursal	Alta	16
HU-14	Modificar sucursal	Media	16
HU-15	Eliminar sucursal	Baja	8
HU-16	Registrar Material	Media	16
HU-17	Modificar Material	Media	16
HU-18	Eliminar Material	Baja	8
HU-19	Registrar Proveedor	Alta	20
HU-20	Modificar Proveedor	Media	20
HU-21	Eliminar Proveedor	Baja	8
HU-22	Generar factura	Alta	20

HU-23	Generar venta a Crédito	Alta	40
HU-24	Registrar Cliente	Alta	20
HU-25	Modificar Cliente	Media	20
HU-26	Eliminar Cliente	Baja	8
HU-27	Registrar Compra	Alta	40
HU-28	Modificar Compra	Alta	40
HU-29	Eliminar Compra	Media	16
HU-30	Registrar Pedido	Alta	40
HU-31	Modificar Pedido	Alta	40
HU-32	Eliminar Pedido	Medio	16
HU-33	Registrar Producto	Alta	16
HU-34	Modificar Producto	Media	16
HU-35	Registrar Inventario	Alta	16
HU-36	Catalogo productos	Alta	40
HU-37	Reporte de los clientes con más pedidos	Media	20
HU-38	Listado de las compras entre fechas	Media	20
HU-39	Listado de los Pedidos entre fechas	Media	20
HU-40	Listado de los productos más vendidos	Media	20
HU-41	Listado de los productos con un stock mínimo	Media	20
HU-42	Listado de los clientes con créditos	Media	20
HU-43	Cobrar Crédito	Alta	20
HU-44	Listado de las facturas a Contado	Media	20
HU-45	Listado de las facturas a Crédito	Media	20

Realizado por: Kleber Hipo, 2019

Tabla 3-2: Tabla Product Backlog Historias Técnicas

ID	TAREA	PRIORIDAD	ESTIMACIÓN
HT-01	Definir estándar de codificación	Media	8
HT-02	Definir Arquitectura del sistema	Alta	20

HT-03	Definir Estándar de interfaz de usuario	Alta	40
HT-04	Instalar los frameworks	Alta	20
HT-05	Diseño de la base de datos	Alta	40
HT-06	Capacitación de los usuarios	Media	16
HT-07	manual de Usuario	Alta	40

Realizado por: Kleber Hipo, 2019

2.4.2.2 Identificación de personas y roles involucrados

En la tabla 4-2 se presenta los roles de las personas implicadas en el desarrollo del presente proyecto.

Tabla 4-2: Tabla de las personas y su rol del proyecto

PERSONA	ROL	CONTACTO	INSTITUCIÓN
Ing. Danilo Pastor	SCRUM Master	danilo.pastor@esPOCH.edu.ec	ESPOCH-FIE
Sr Mario Morocho	Product Owner	Almacenescacha@hotmail.com	Gerente almacenes Cacha
Kleber Hipo	Development Team	hipochimborazo1@hotmail.com	ESPOCH-EIS

Realizado por: Kleber Hipo. 2019

2.4.2.3 Tipos de roles y usuarios del Sistema

A continuación, se muestra los tipos de usuarios y el rol que desempeñan en el presente proyecto, los cuales están implicados directamente en el manejo del sistema los cuales se los describe a continuación:

Tabla 5-2: Tipos de usuario y roles de usuario del sistema

TIPO DE USUARIO	DESCRIPCIÓN	ROL	RESPONSABLE
Administrador	Es el responsable de administrar el sistema	Tiene el control total de las funcionalidades del sistema web (gestionar Categorías, Material, Proveedor, Clientes, Productos, Pedidos, Compras y Reportes)	Sra. Mario Ramiro Morocho

Jefe de Agencia	Usuarios que accederán a la aplicación web	Tiene el control total del sistema pero de una sucursal específico	Mario Morocho
Empleado	Persona que es va realizar las compras y ventas al almacén	Administración de clientes, Confirmación de pedidos, gestión de compras y ventas	Sandra Paucar

Realizado por: Kleber Hipo, 2019

2.4.2.4 Plan de Entrega (Sprint Backlog)

Para el desarrollo y cumplimiento de los Sprints del proyecto, se establece un cronograma de actividades estimando una duración de 26 semanas laborables, teniendo en cuenta que la fecha del inicio del proyecto es el 02/04/2018 y su culminación el 26/09/2018, se trabaja 8 horas diarias de lunes a viernes, bajo la siguiente planificación.

Tabla 6-2: Tabla del Sprint Backlog con las Historias Técnicas y Usuarios

N° Sprint	Id	Tarea	N° Horas	Fecha Inicio	Fecha Fin	N° horas
1	HT-02	Definir Arquitectura del sistema	20	02/04/2018	04/04/2018	120
	HT-03	Definir Estándar de interfaz de usuario	40	04/04/2018	10/04/2018	
	HT-04	Instalar los frameworks	20	11/04/2018	13/04/2018	
	HT-05	Diseño e Implementación de la base de datos	40	16/04/2018	20/04/2018	
2	HT-01	Definir estándar de codificación	8	23/04/2018	23/04/2018	120
	HU-01	Autenticación de usuarios	16	24/04/2018	25/04/2018	
	HU-02	Registrar datos de la Empresa	16	26/04/2018	27/04/2018	
	HU-04	Registrar categoría	8	30/04/2018	30/04/2018	
	HU-33	Registrar Producto	16	01/05/2018	02/05/2018	
	HU-13	Registrar Sucursal	16	03/05/2018	04/05/2018	
	HU-19	Registrar Proveedor	20	07/05/2018	09/05/2018	
3	HU-10	Registrar Empleado	20	09/05/2018	11/05/2018	120
	HU-24	Registrar Cliente	20	14/05/2018	16/05/2018	
	HU-11	Modificar empleado	20	16/05/2018	18/05/2018	
	HU-27	Registrar Compra	40	21/05/2018	25/05/2018	
4	HU-28	Modificar Compra	40	28/05/2018	01/06/2018	120
	HU-30	Registrar Pedido	40	04/06/2018	08/06/2018	
	HU-31	Modificar Pedido	40	11/06/2018	15/06/2018	
	HU-22	Generar factura	20	18/06/2018	20/06/2018	

	HU-20	Modificar Proveedor	20	20/06/2018	22/06/2018	
5	HU-23	Generar venta a Crédito	40	25/06/2018	29/06/2018	120
	HU-36	Catalogo productos	40	02/07/2018	06/07/2018	
	HU-43	Cobrar Crédito	20	09/07/2018	11/09/2018	
	HU-25	Modificar Cliente	20	11/07/2018	13/07/2018	
6	HU-35	Registrar Inventario	16	16/07/2018	17/07/2018	120
	HU-03	Modificar Datos Empresa	16	18/07/2018	19/07/2018	
	HU-05	Modificar Categoría	8	20/07/2018	20/07/2018	
	HU-07	Registrar un color	8	23/07/2018	23/07/2018	
	HU-08	Modificar un color	8	24/07/2018	24/07/2018	
	HU-14	Modificar sucursal	16	25/07/2018	26/07/2018	
	HU-16	Registrar Material	16	27/07/2018	30/07/2018	
	HU-17	Modificar Material	16	31/07/2018	01/08/2018	
7	HU-37	Reporte de los clientes con más pedidos	20	06/08/2018	08/08/2018	120
	HU-38	Listado de las compras entre fechas	20	08/08/2018	10/08/2018	
	HU-39	Listado de los Pedidos entre fechas	20	13/08/2018	15/08/2018	
	HU-40	Listado de los productos más vendidos	20	15/08/2018	17/08/2018	
	HU-41	Listado de los productos con un stock mínimo	20	20/08/2018	22/08/2018	
	HU-42	Listado de los clientes con créditos	20	22/08/2018	24/08/2018	
8	HU-44	Listado de las facturas a Contado	20	27/08/2018	29/08/2018	120
	HU-45	Listado de las facturas a Crédito	20	29/08/2018	31/08/2018	
	HU-06	Eliminar una categoría	8	03/09/2018	03/09/2018	
	HU-09	Eliminar Color	8	04/09/2018	04/09/2018	
	HU-12	Eliminar empleado	8	05/09/2018	05/09/2018	
	HU-15	Eliminar sucursal	8	06/09/2018	06/09/2018	
	HU-21	Eliminar Proveedor	8	07/09/2018	07/09/2018	
	HU-26	Eliminar Cliente	8	10/09/2018	10/09/2018	
9	HU-32	Eliminar Pedido	16	11/09/2018	12/09/2018	64
	HU-34	Modificar Producto	16	13/09/2018	14/09/2018	
	HU-18	Eliminar Material	8	17/09/2018	17/09/2018	
	HT-07	manual de Usuario	40	18/09/2018	24/09/2018	
	HT-06	Capacitación de los usuarios	16	25/09/2018	26/09/2018	

2.4.2.5 Reuniones Scrum

Al momento de culminar un Sprint, es necesario realizar reuniones de cierre de cada uno de los mismos, para realizar la verificación de las historias de usuario culminadas y entregadas hasta la fecha.

En esta reunión el equipo discute de los problemas que se tuvieron con las historias de usuario, estos errores son analizados y corregidos en equipo antes de empezar con el siguiente sprint.

2.4.3 Fase Desarrollo

2.4.3.1 Arquitectura del sistema

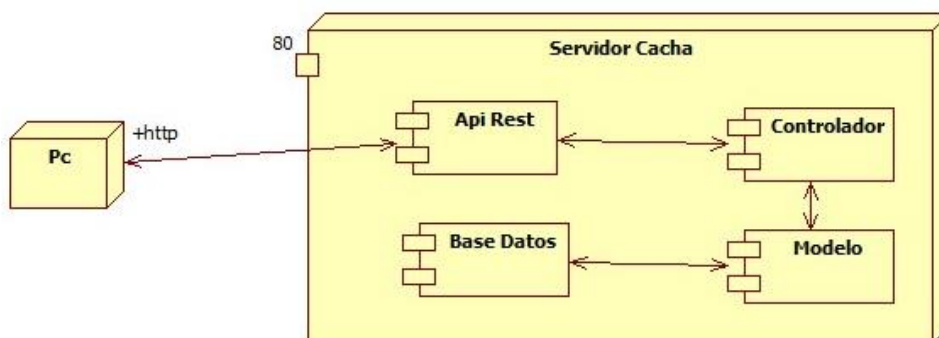


Figura 1-2: Arquitectura del sistema

Realizado por: Kleber Hipo, 2019

Para el desarrollo de la aplicación web se define la arquitectura Modelo Controlador (MC), La capa de la vista este Framework no los posee.

- **Modelo:** es aquel que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.
- **Controlador:** actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno.
- **Vista:** es la interacción entre el usuario final y el sistema estas vistas, estas pueden ser un formulario o un texto plano.

Como se menciona que una vista específicamente no debe ser una interfaz gráfica por lo cual se considera que la respuesta del API REST es una vista de texto plano, para la comprensión y una

mejor visualización del usuario final esta es procesada por un framework de JavaScript Vue.js de forma gráfica.

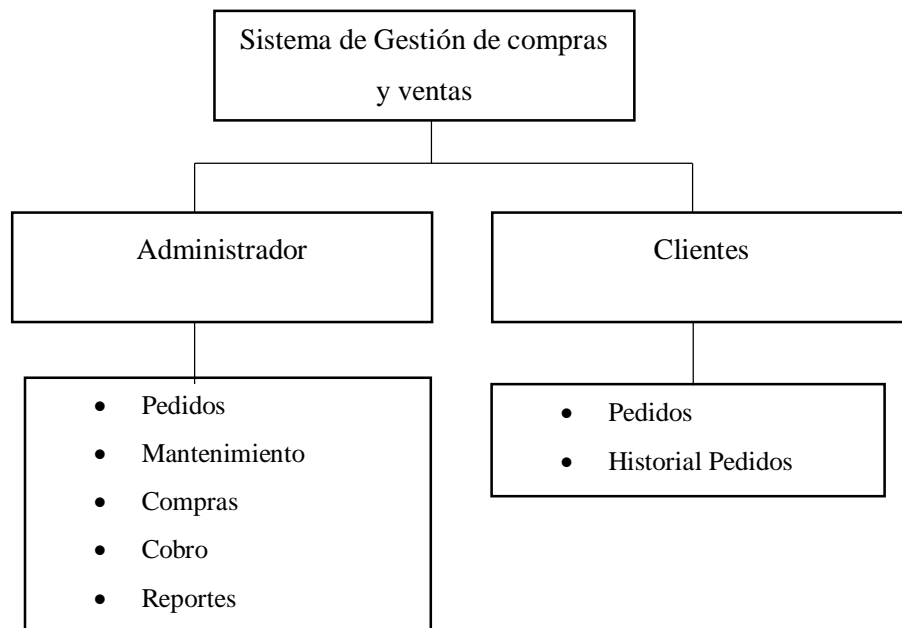


Figura 2-2: Diseño del sistema de Gestión de Compras
Realizado por: Hipo Kleber

El Módulo Administrador contiene los siguientes:

Módulo de Pedidos: En este módulo el usuario puede realizar la gestión de clientes, pedidos, facturas estas pueden ser a créditos o a contados, y créditos.

Módulo Mantenimiento: En este módulo el usuario podrá realizar la gestión de las sucursales, empleados, usuarios y los datos generales de la empresa.

Módulo de Compras: Se puede gestionar toda la información de un proveedor y las compras que ha realizado el almacén.

Módulo de Cobro.- Permite que se realice el cobro de un crédito.

Reportes: El administrador del sistema tiene el acceso a todos los reportes que genere el sistema, entre las cuales constan reportes de los productos con un stock mínimo, reporte de la compras y pedidos entre fechas, reporte de los clientes más frecuentes, reporte de las facturas a crédito y/o contado.

El Módulo Administrador contiene los siguientes:

Módulo de Pedidos: En este módulo el cliente realiza el ingreso de datos de su Pedido.

Historial Pedidos: Almacenas los datos los datos de los diferentes pedidos que realiza un Cliente

2.4.3.2 Estándar de interfaz de usuario

Este tipo de actividad tiene como finalidad llegar a un diseño previo del cuál será el aspecto que tendrá el front-end del sistema, se realiza la estructuración de los diferentes bosquejos de pantalla, los cuales se detallan a continuación, considerando la compañía a la cual va dirigido el sistema, y los requerimientos del personal involucrado en el problema

Las interfaces mostradas a continuación varían de acuerdo a la petición que realice el usuario ya sea de entrada o salida de datos, habiendo similitud entre ellas. Por esta razón se ha tomado un ejemplar de cada tipo de acción, detallando de la mejor forma como está estructurado cada una de las pantallas que forma el sistema.

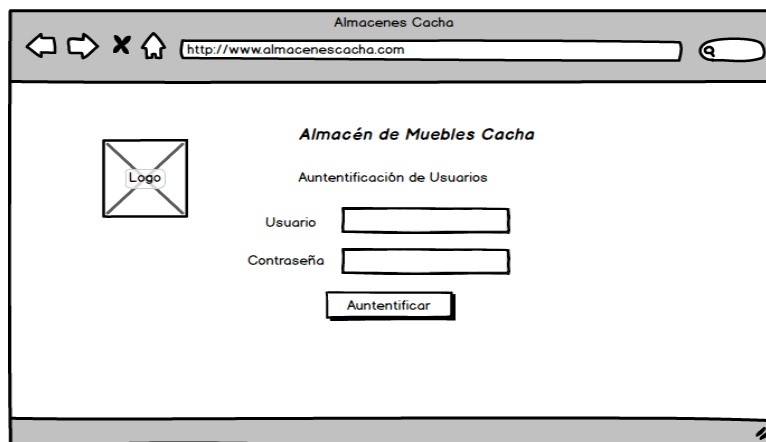


Figura 3-2: Autenticación de los usuarios
Realizado por: Kleber Hipo, 2019

Para la interfaz del sistema se define que en la interfaz tenga un logo, un menú principal de opciones en la parte izquierda y en la parte central derecha es en donde se visualiza la funcionalidad de la aplicación como listado y formulario.

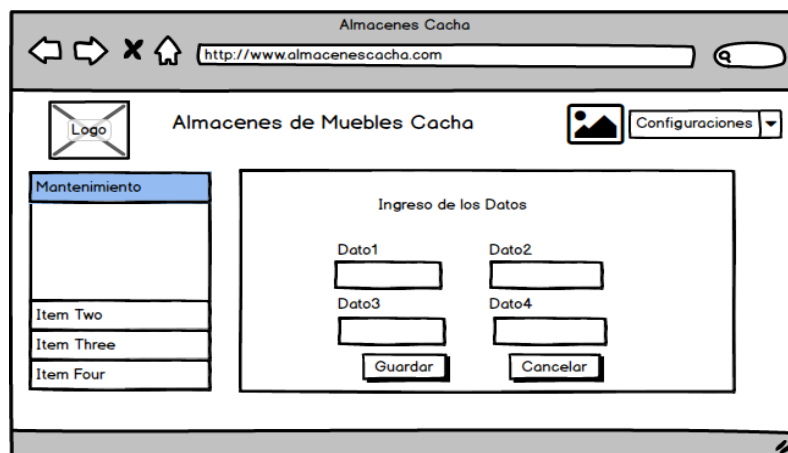


Figura 4-2: Registro de Información
Realizado por: Kleber Hipo, 2019

En esta sesión del documento se describe cada una de las actividades realizadas en el desarrollo del sistema web para la mueblería “Cacha” utilizando el Micro-framework Lumen y Mysql.

2.4.3.3 Instalación de los frameworks

Instalar el Micro-Framework Lumen

Para proceder a instalar el Micro-Framework Lumen en nuestro ordenador se debe tener instalado el lenguaje de programación php (XAMPP), y los pasos que se debe seguir son los siguientes:

- Dirigir a la página de composer que se encuentra en la URL <https://getcomposer.org/download/>.
- En la página oficial se descarga para Windows el archivo ejecutable .exe.

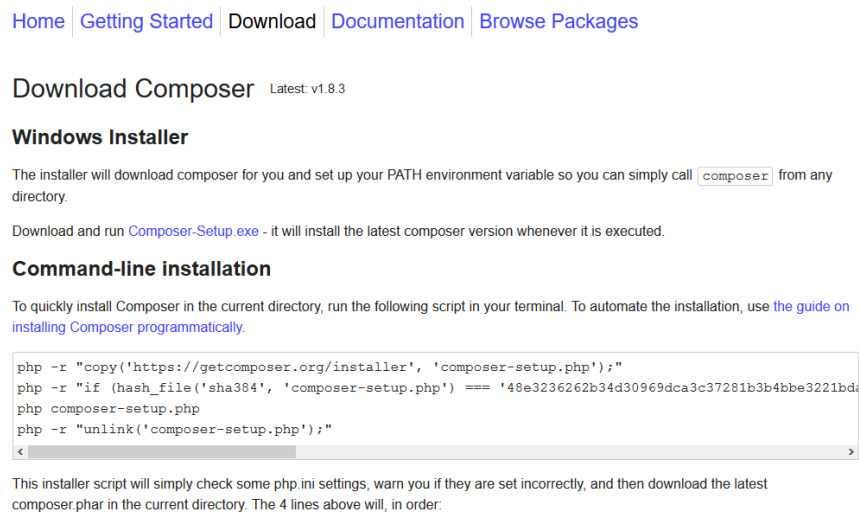


Figura 5-2: Página de la Descarga de Composer
Realizado por: Kleber Hipo, 2019

- Se ejecuta el archivo descargado como administrador y se indica la dirección en donde está instalado php como muestra la figura.

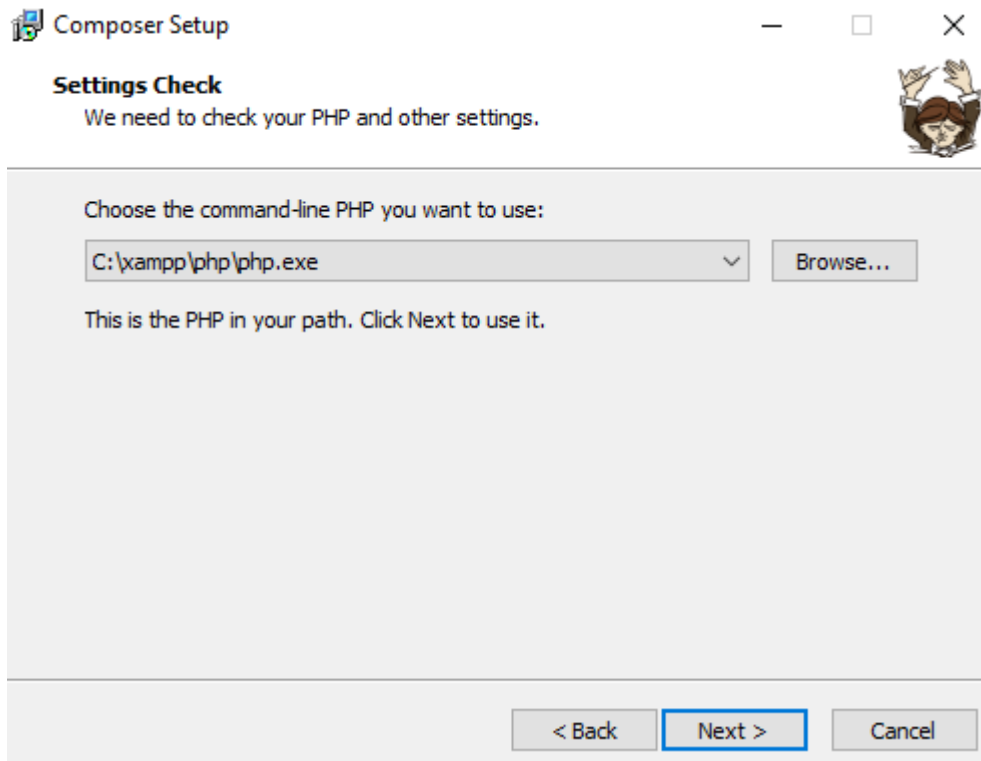


Figura 6-2: Instalación de composer

Realizado por: Kleber Hipo, 2019

- Se Crea una carpeta con el nombre del proyecto que se va a realizar en la dirección de C:\xampp\htdocs, y accedemos a este directorio en el símbolo del sistema (cmd).
- Ya en la consola de Windows se procede a ejecutar el comando *composer create-project --prefer-dist laravel/lumen AlmacenesCacha*, posterior empezara el proceso de la instalación.

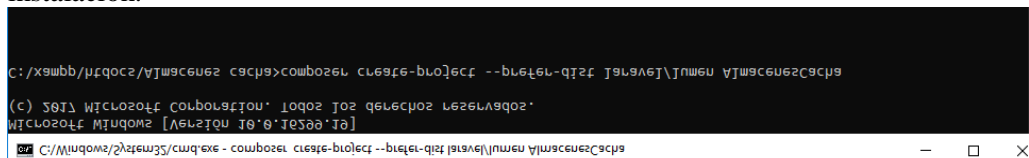


Figura 7-2: Instalación del micro-Framework Lumen

Realizado por: Kleber Hipo, 2019

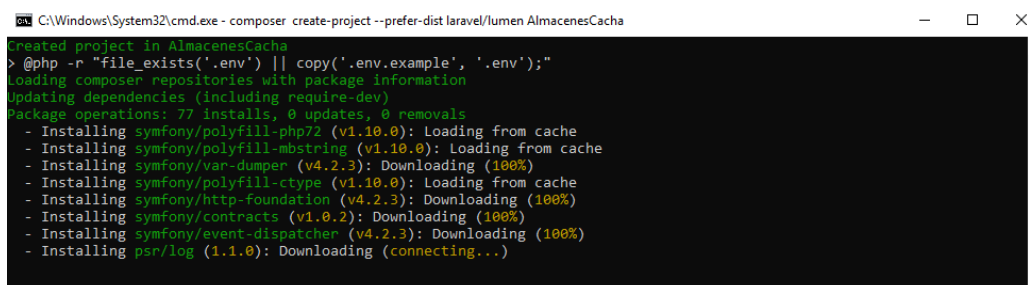


Figura 8-2: Proceso de instalación de Lumen

Realizado por: Kleber Hipo, 2019

- Cuando termine de instalar se ejecuta en el símbolo del sistema el comando ***php -S localhost:8000 -t public*** como en la **figura 19-2**, abrimos un navegador web y se escribe localhost:8000, se visualizara lumen y su versión como se puede ver en la figura 10-2 esta nos indica que nuestro micro-framework se ha instalado correctamente.

```

C:\Windows\System32\cmd.exe - php -S localhost:8000 -t public
C:\xampp\htdocs\Almacenes\AlmacenesCacha> php -S localhost:8000 -t public
PHP 7.2.11 Development Server started at Wed Feb 6 10:57:50 2019
Listening on http://localhost:8000
Document root is C:\xampp\htdocs\Almacenes\AlmacenesCacha\public
Press Ctrl-C to quit.
[Wed Feb 6 10:58:37 2019] ::1:54349 [200]: /
[Wed Feb 6 10:59:00 2019] ::1:54352 [404]: /favicon.ico - No such file or directory
  
```

Figura 9-2: comando de ejecución del micro-Framework
 Realizado por: Kleber Hipo, 2019

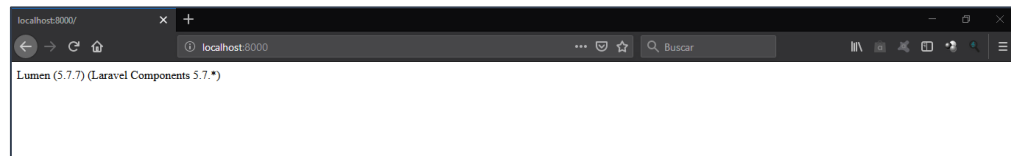


Figura 10-2: Interfaz de inicio del micro-Framework
 Realizado por: Kleber Hipo, 2019

Instalación de Framework Vue Js.

1. Descargar node.js para lo cual se debe dirigir a la url <https://nodejs.org/en/> y descargar.

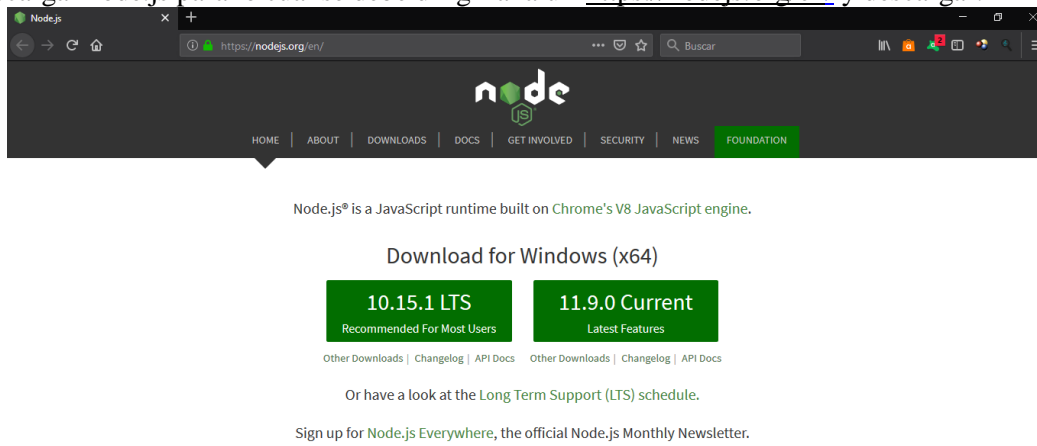


Figura 11-2: Pagina de la descarga de nodejs
 Realizado por: Kleber Hipo, 2019

2. Instalar el archivo descargado como un programa cualquiera de Windows, se puede instalar de forma global para la pc o para un directorio específico.
3. Se crea una carpeta la cual contendrá el Framework con el nombre del proyecto (C:\xampp\htdocs).
4. En el símbolo del sistema se accede al directorio creado en el paso anterior y ejecuta el comando ***npm install vue***

```

C:\xampp\htdocs\Almacenes\AlmacenesCacha> npm install vue
[.....] / rollbackFailedOptional: verb npm-session 77326edc4a4139ea
  
```

Figura 12-2: Instalación de Vue.js
 Realizado por: Kleber Hipo, 2019

Posterior de haber instalado en el mismo terminal ejecute el comando *vue --version* y si en caso que se instaló de forma correcta indicara la versión de vue como muestra en la figura.

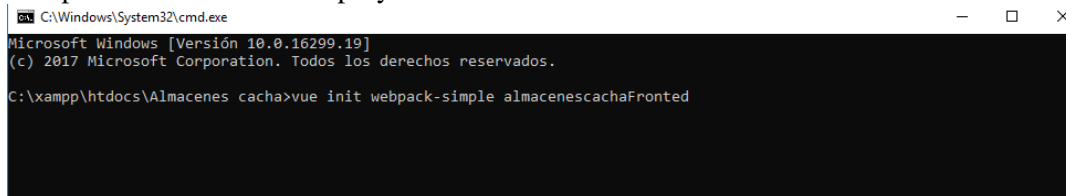


```
C:\Windows\System32\cmd.exe - cmd
C:\xampp\htdocs\Almacenes_cacha>vue --version
2.9.3
C:\xampp\htdocs\Almacenes_cacha>
```

Figura 13-2: Versión del Vue.Js Instalado

Realizado por: Kleber Hipo, 2019

5. Creación de un nuevo proyecto para lo cual se ejecuta el comando *vue init webpack-simple* acompañado del nombre del proyecto.

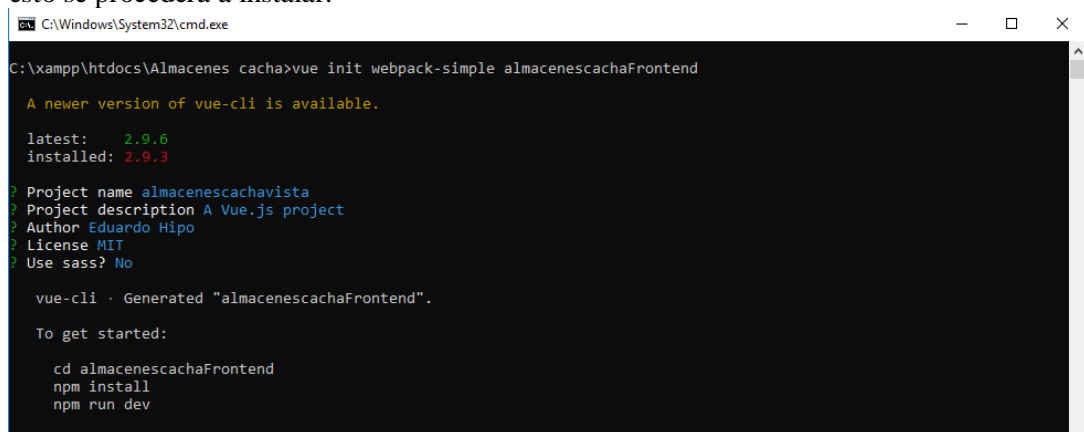


```
C:\Windows\System32\cmd.exe
Microsoft Windows [Versión 10.0.16299.19]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.
C:\xampp\htdocs\Almacenes_cacha>vue init webpack-simple almacenescachaFronted
```

Figura 14-2: Creación de un nuevo proyecto de vue

Realizado por: Kleber Hipo, 2019

A continuación se solicita datos como nombre del proyecto el autor y una descripción dado esto se procederá a instalar.



```
C:\Windows\System32\cmd.exe
C:\xampp\htdocs\Almacenes_cacha>vue init webpack-simple almacenescachaFrontend
A newer version of vue-cli is available.
latest: 2.9.6
installed: 2.9.3
? Project name almacenescachavista
? Project description A Vue.js project
? Author Eduardo Hipo
? License MIT
? Use sass? No
vue-cli · Generated "almacenescachaFrontend".
To get started:
  cd almacenescachaFrontend
  npm install
  npm run dev
```

Figura 15-2: Finalización de la instalación del proyecto

Realizado por: Kleber Hipo, 2019

6. Cuando se haga terminado de instalar se debe acceder al directorio donde se instalado (almacenescachafrentend) y ejecutar el comando *npm install*.

```
C:\Windows\System32\cmd.exe
C:\xampp\htdocs\Almacenes_cacha\almacenes_cachaFrontend>npm install
> uglifyjs-webpack-plugin@0.4.6 postinstall C:\xampp\htdocs\Almacenes_cacha\almacenes_cachaFrontend\node_modules\uglifyjs-
-webkit-plugin
> node lib/post_install.js
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.7 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.7: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
added 803 packages in 287.781s
C:\xampp\htdocs\Almacenes_cacha\almacenes_cachaFrontend>
```

Figura 16-2: Instalación de las dependencias para vue
Realizado por: Kleber Hipo, 2019

7. Vue trae un servidor para desarrollo lo cual se ejecuta el comando *npm run dev*. Como muestra la **figura 17-2**.

```
npm
C:\xampp\htdocs\Almacenes_cacha\almacenes_cachaFrontend>npm run dev
> almacenes_cachavista@1.0.0 dev C:\xampp\htdocs\Almacenes_cacha\almacenes_cachaFrontend
> cross-env NODE_ENV=development webpack-dev-server --open --hot
Project is running at http://localhost:8080/
webpack output is served from /dist/
404s will fallback to /index.html
```

Figura 17-2: Ejecución del proyecto en modo desarrollador Vue.js
Realizado por: Kleber Hipo, 2019

8. En una navegador web se accede a la url **localhost:8080**. En la cual nos visualiza la interfaz de vue.

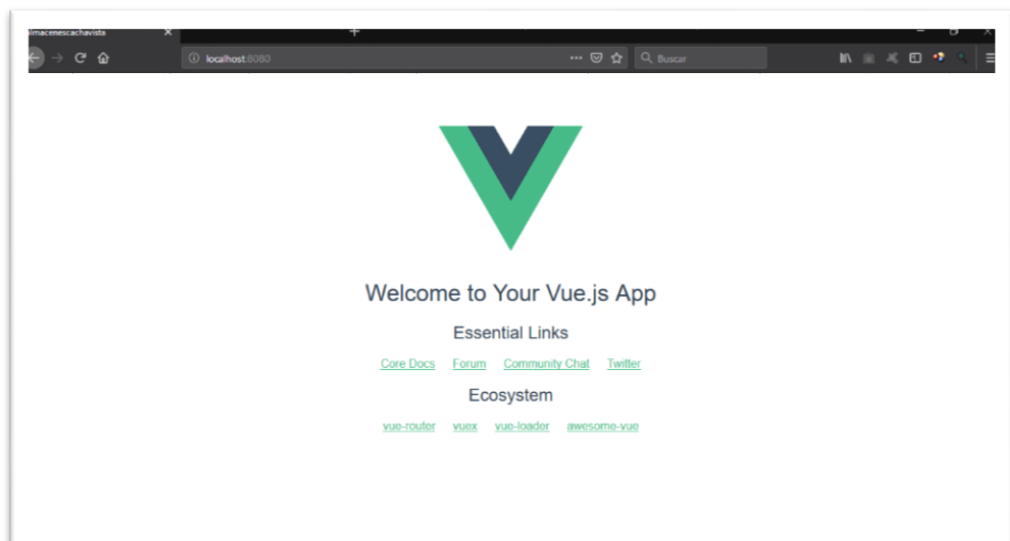


Figura 18-2: Interfaz principal del Framework Vue.js
Realizado por: Kleber Hipo, 2019

2.4.3.4 Diseño e Implementación de la base de datos

Diseño de la base de datos

Para el diseño físico de la base de datos se utiliza el gestor de base de datos Mysql en la cual consta de un total de 23 tablas donde se visualiza las diferentes relaciones con sus respectivas llaves foráneas como no indica la siguiente **Figura 19-2**.

Los tipos de datos que se usaron son:

- Integer: este tipo de datos que almacenar números enteros (Ejemplo: Cantidad)
- Numeric: Este dato se utilizó para almacenar los precios de los productos.
- Char: para una cadena de caracteres cuando se conoce su longitud (Ejemplo Cédula)
- Varchar: Para cadenas de caracteres que no se conocen su longitud.
- Date: Para el almacenamiento de las fechas.
- Timestamp: Tipo de dato para almacenar la fecha y hora de cuando se ejecuta una operación

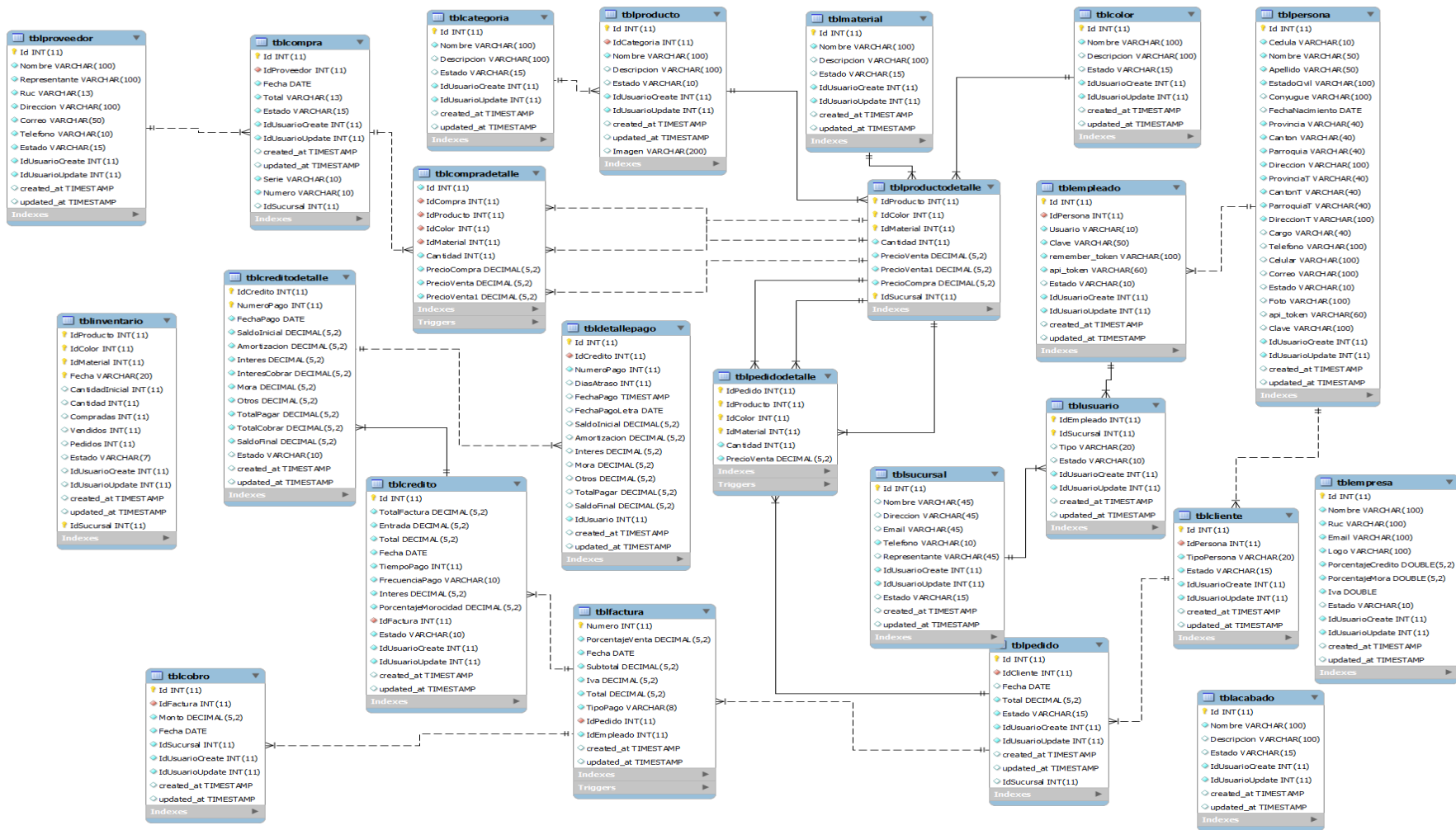


Figura 19-2: Diagrama Lógico de la Base de Datos
 Realizado por: Kleber Hipo, 2019

2.4.3.5 Diccionario de datos

Un diccionario de datos es el cual nos indica de cómo están estructurados las tablas, sus relaciones, además indican que tipo de datos son sus campos y la longitud.

Para obtener el diccionario de datos de nuestra base de datos primero se descarga un Plugin llamado **datadict_grt.py** para Mysql Workbench desde la dirección [http://wiki.flat.cl/index.php?title=Generar diccionario de datos con MySQL Workbench](http://wiki.flat.cl/index.php?title=Generar_diccionario_de_datos_con_MySQL_Workbench) .

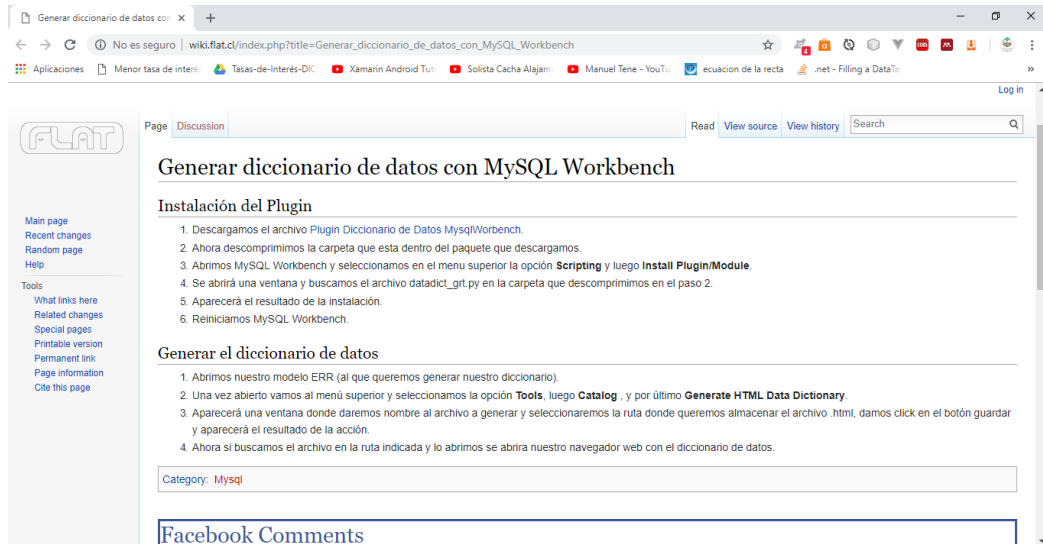


Figura 20-2: Página de descarga de plugins datadict_grt.py
Realizado por: Kleber Hipo, 2019

Posterior a eso se procede a instalarlo en nuestra aplicación ingresando en el menú de la barra Superior, **Scripting >Install Plugin/Module** para lo cual nos solicita que se seleccione el plugin, posterior a eso se cierra MySQL Workbench para que tome los nuevos cambios.

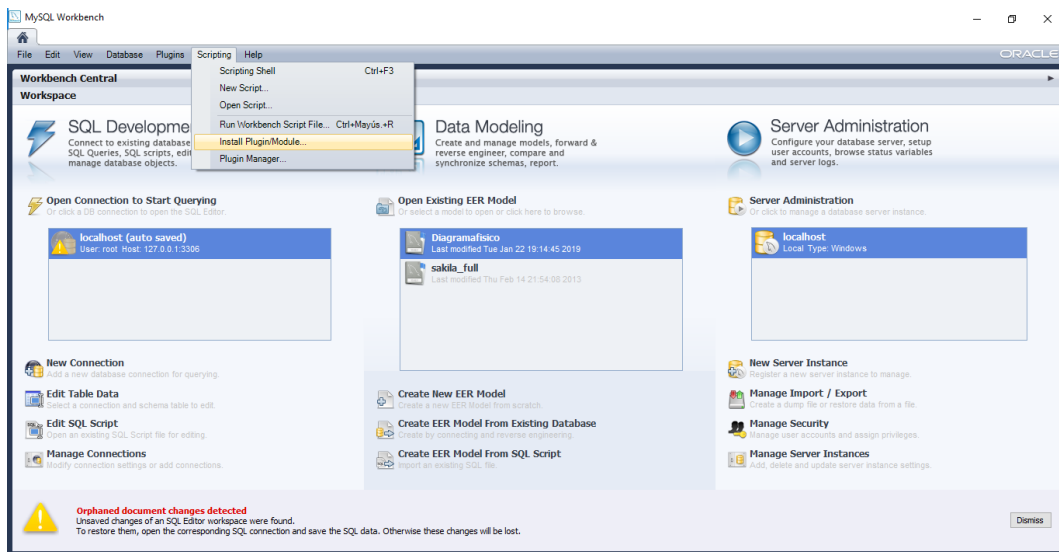


Figura 21-2: Instalación del plugins datadict_grt.py
Realizado por: Kleber Hipo, 2019

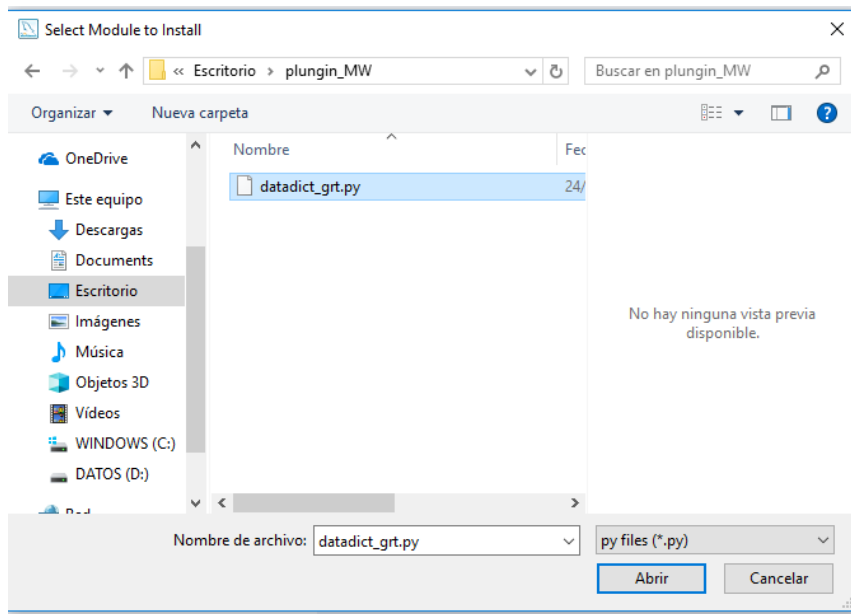


Figura 22-2: Selección del archivo datadict_grt.py
Realizado por: Kleber Hipo, 2019

Cuando se ejecute de nuevo mysql Workbench se selecciona el diseño físico de la base de datos y se redirige en el menú superior a **Plugins>Catalog>Generate HTML Data Dictionary** se asigna un nombre del archivo con la extensión.html

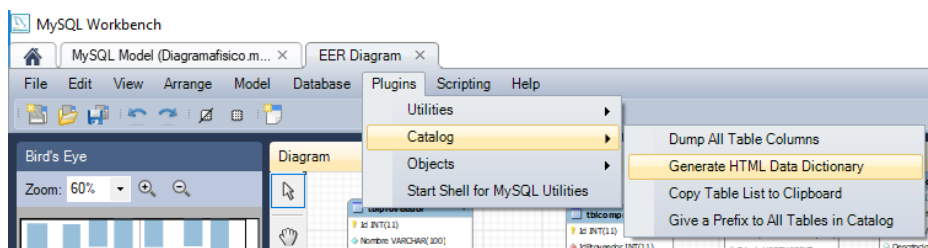


Figura 23-2: Generación del diccionario de datos
Realizado por: Kleber Hipo, 2019

Se elige el directorio en donde se desea guardar el archivo generado.

Diccionario de datos tabla tblsucursal

tblcompra										
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
Id	INT(11)	✓	✓					✓		
IdProveedor	INT(11)		✓							
Fecha	DATE		✓							
Total	VARCHAR(13)		✓							
Estado	VARCHAR(15)		✓						'Activo'	
IdUsuarioCreate	INT(11)		✓							
IdUsuarioUpdate	INT(11)		✓							
created_at	TIMESTAMP								NULL	
updated_at	TIMESTAMP								NULL	
Serie	VARCHAR(10)		✓							
Numero	VARCHAR(10)		✓							
IdSucursal	INT(11)								NULL	

Figura 24-2: Diccionario datos tabla factura
Realizado por: Kleber Hipo, 2019

En la **Figura 24-2** se puede visualizar de cual el la clave primaria los campos que son obligatorios y las cuales no son obligatorios.

Para poder visualizar de forma detallada el diccionario de datos revisar el **Anexo B.**

2.4.3.6 Documentación de las historias de Usuario y Técnicas

Registro de Compras

HU-27 Registrar Compra: Esta historia de usuario tiene una duración de 40 puntos estimados dando inicio el 21/05/2018 hasta el 25/05/2018.

Como nuestra aplicación tiene la parte de FrontEnd y Backend por lo cual se debe hacer tareas de ingeniería en las dos partes a continuación las diferentes tareas de esta historia de usuario.

Tabla 7-2: Historia de usuario del registro de compras

HISTORIA DE USUARIO	
Número: HU-27	Nombre: Registrar Compra
Modificación historia de usuario:	
Usuario: Administrador	Sprint Asignada: 3
Prioridad en Negocio: Alta	Puntos Estimados: 40
Riesgo en desarrollo: Alto	Puntos Reales: 50
Descripción: Como Administrador quiero registrar los datos de una compra para tener un control detallado de los productos que ingresan al Almacén	
Observaciones:	
Pruebas de Aceptación:	
<ul style="list-style-type: none"> • Emitir un mensaje de éxito cuando se haya registrado los datos de la Compra. • Emitir un mensaje cuando se trate de enviar formularios vacíos. • Emitir el mensaje cuando no se haya registrado los datos de la compra. 	

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 1, HU-27 Registrar Compra

Tabla 8-2: Prueba de aceptación 1, Registro de Compra

Prueba de aceptación	
Código: PA-01	Historia de Usuario: HU-27 Registrar Compra
Nombre: Emitir un mensaje de éxito cuando se haya registrado los datos de la Compra.	
Responsable: Kleber Hipo	Fecha: 25-05-2018
Descripción: Verificar que se emita un mensaje de éxito cuando los datos estén registrados en la base de datos	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • La funcionalidad del registro de compra debe estar implementado. • Debe estar conectada con la base de datos • El Usuario debe estar autenticado 	

Pasos de ejecución:	
<ul style="list-style-type: none"> • Clic en Compra del menú izquierdo. • Clic en el submenú Compra • Clic en el botón Nuevo. • Ingrese los datos requeridos en el formulario • Clic en el botón Guardar. 	
Resultado esperado:	
Mensaje del sistema indicando que se ha registrado los datos correctamente.	
Evaluación: Exitosa.	

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 2, HU-27 Registrar Compra

Tabla 9-2: Prueba de aceptación 2, Registro de Compra

Prueba de aceptación	
Código: PA-02	Historia de Usuario: HU-27 Registrar Compra
Nombre: Emitir un mensaje cuando se trate de enviar formularios vacíos	
Responsable: Kleber Hipo	Fecha: 25-05-2018
Descripción: No se permita el registro cuando existan formularios vacíos.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • La funcionalidad de registro de compra debe estar implementado. • Debe estar conectada con la base de datos • El Usuario debe estar autenticado 	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Clic en Compra del menú Izquierdo • Clic en el submenú Compra • Clic en el botón Nuevo. • Dar clic en el botón Guardar. 	
Resultado esperado:	
Emitir un mensaje indicado que hay datos vacíos.	
Evaluación: Exitosa.	

Realizado por: Kleber Hipo, 2019

Tabla 10-2: Prueba de aceptación 3, Registro de compra

Prueba de aceptación	
Código: PA-03	Historia de Usuario: HU-27 Registrar Compra
Nombre: Emitir el mensaje cuando no se haya registrado los datos de la compra.	
Responsable: Kleber Hipo	Fecha: 25-05-2018
Descripción: Comprobar que se emita un mensaje indicando que los datos ingresados no se almacenaron en la base de datos.	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • La funcionalidad de registro de compra debe estar implementado. • El Usuario debe estar autenticado. • Estar desconectado de la base de datos 	

<p>Pasos de ejecución:</p> <ul style="list-style-type: none"> • Clic en Compra del menú Izquierdo • Clic en el submenú Compra • Clic en Nuevo. • Ingresar los datos que se solicita el formulario • Clic en el botón Guardar.
<p>Resultado esperado: Visualizar el mensaje indicando que no se pudo guardar los datos de la compra y el error correspondiente</p>
<p>Evaluación: Exitosa.</p>

Realizado por: Kleber Hipo, 2019

Tarea de ingeniería Backend

Tarea de Ingeniería 1, HU-27 Registrar Compra

Tabla 11-2: Tarea de ingeniería 1, Registro de compra

TAREA DE INGENIERÍA	
Historia de Usuario: HU-27 Registrar Compra	
Número de Tarea: TI-01	Nombre de Tarea: Crear el controlador de la Compra con su método registrar
Tipo de Tarea: Desarrollo	Puntos Estimados: 6
Fecha Inicio: 21-05-2018	Fecha Fin: 21-05-2018
Programador Responsable: Kleber Hipo	
Descripción: Se debe crear el controlador de la compra con sus respectivo método Registrar	
Pruebas de Aceptación	
<ul style="list-style-type: none"> • Verificar que el controlador este en la estructura correcta del proyecto 	

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 1 de la Tarea de Ingeniería 1 de HU-27 Registrar Compra

Tabla 12-2: Prueba de aceptacion1, Tarea de ingeniería 1, Registro de compra

Prueba de aceptación	
Código: PA-04	Historia de Usuario: HU-27 Registrar Compra
Nombre: Verificar que el controlador este en la estructura correcta del proyecto	
Responsable: Kleber Hipo	Fecha: 21-05-2018
Descripción: Comprobar que el controlador exista en el proyecto para su funcionamiento	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • El controlador de Compra debe estar creado con su respectivo método registrar. 	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Abrir el directorio de la aplicación • Abrir la carpeta app. • Abrir la carpeta Http. • Abrir la carpeta controller. • Clic sobre el archivo CompraController.php 	
Resultado esperado:	
Debe estar el controlador de la compra con su respectivo método Registrar.	

Evaluación: Exitosa.

Realizado por: Kleber Hipo, 2019

Tarea de Ingeniería 2, HU-27 Registrar Compra

Tabla 13-2: Tarea de ingeniería 2, Registro de compra

TAREA DE INGENIERÍA	
Historia de Usuario: HU-27 Registrar Compra	
Número de Tarea: TI-02	Nombre de Tarea: Crear el modelo de la Compra
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Fecha Inicio: 21-05-2018	Fecha Fin: 21-05-2018
Programador Responsable: Kleber Hipo	
Descripción: Crear el modelo de la compra en el directorio correspondiente.	
Pruebas de Aceptación	
<ul style="list-style-type: none"> • Verificar que el modelo este en la estructura correcta. • Comprobar que este enlazado con la tabla tblcompra de la base de datos 	

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 1 de la tarea de Ingeniería 2, HU-27 Registrar Compra

Tabla 14-2: Prueba de aceptación 1, Tarea de ingeniería 2, Registro de compra

Prueba de aceptación	
Código: PA-05	Historia de Usuario HU-27 Registrar Compra
Nombre: Verificar que el modelo este en la estructura correcta	
Responsable: Kleber Hipo	Fecha: 21-05-2018
Descripción: Comprobar que el modelo este creado en el proyecto para su correcto funcionamiento con el controlador	
Condiciones de Ejecución:	
<ul style="list-style-type: none"> • El modelo de Compra debe estar creado anteriormente. 	
Pasos de ejecución:	
<ul style="list-style-type: none"> • Abrir el directorio de la aplicación • Abrir en la carpeta app. 	
Resultado esperado:	
Visualizar el modelo de la Compra.	
Evaluación: Exitosa.	

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 2 de la Tarea de Ingeniería 2: HU-27 Registrar Compra

Tabla 15-2: Tarea de ingeniería 2, prueba de aceptación 2, Registro de compra

Prueba de aceptación	
Código: PA-06	Historia de Usuario: HU-27 Registrar Compra
Nombre: Comprobar que este enlazado con la tabla tblcompra de la base de datos	
Responsable: Kleber Hipo	Fecha: 21-05-2018

Descripción: Comprobar que el modelo exista en el proyecto y esté relacionado con su tabla correspondiente de la base de datos
Condiciones de Ejecución: <ul style="list-style-type: none"> • El modelo de Compra debe existir. • La tabla tblcompra debe estar creado en la base de datos
Pasos de ejecución: <ul style="list-style-type: none"> • Abrir el directorio de la aplicación • Abrir la carpeta app. • Clic sobre el archivo Compra.php. • Verificar que el atributo \$table tenga el nombre de la tabla tbcompra. • Verificar que los atributos de la tabla tblcompra estén en el array \$fillable
Resultado esperado: Debe estar el modelo Compra con sus respectivos atributos de la tabla y estar enlazada con la misma.
Evaluación: Exitosa.

Realizado por: Kleber Hipo, 2019

Tarea de Ingeniería 3, HU-27 Registrar Compra

Tabla 16-2: Tarea de ingeniería 3, Registro de compra

TAREA DE INGENIERÍA	
Historia de Usuario: HU-27 Registrar Compra	
Número de Tarea: TI-03	Nombre de Tarea: Crear API REST del registro la Compra
Tipo de Tarea: Desarrollo	Puntos Estimados: 8
Fecha Inicio: 22-05-2018	Fecha Fin: 22-05-2018
Programador Responsable: Kleber Hipo	
Descripción: Crear el servicio API REST y este enlazado con su respectivo controlador y método	
Pruebas de Aceptación <ul style="list-style-type: none"> • Verificar que exista la ruta de Compra/create este en el archivo de los servicios web. • Verificar que el servicio web esté conectado con el controlador correspondiente. • Realizar testing del servicio web creado 	

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 1, Tarea de Ingeniería 3, HU-27 Registrar Compra

Tabla 17-2: Prueba de aceptación 1, Tarea de ingeniería 3, Registro de compra

Prueba de aceptación	
Código: PA-07	Historia de Usuario: HU-27 Registrar Compra
Nombre: Verificar que exista la ruta de Compra/creáte este en el archivo de los servicios web.	
Responsable: Kleber Hipo	Fecha: 22-05-2018
Descripción: Crear el servicio API REST para que una aplicación externa pueda consumir los datos.	

Condiciones de Ejecución: <ul style="list-style-type: none"> • Estar creado el servicio REST en el archivo web.php
Pasos de ejecución: <ul style="list-style-type: none"> • Abrir el directorio de la aplicación • Abrir en la carpeta routes. • Clic sobre el archivo web.php.
Resultado esperado: Estar el método el servicio REST API con la ruta Compra/create.
Evaluación: Exitosa.

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 2, Tarea de Ingeniería 3, HU-27 Registrar Compra

Tabla 18-2: Tarea de ingeniería 3, prueba de aceptación 2, Registro de compra

Prueba de aceptación	
Código: PA-08	Historia de Usuario: HU-27 Registrar Compra
Nombre: Verificar que el servicio web esté conectado con el controlador correspondiente	
Responsable: Kleber Hipo	Fecha: 22-05-2018
Descripción: Verificar que se haya una comunicación entre API REST y el controlador Compra con su respectivo método create	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Estar creado el servicio web API REST y este enlazado con el controlador Compra con su respectivo método 	
Pasos de ejecución: <ul style="list-style-type: none"> • Abrir el directorio de la aplicación • Abrir en la carpeta routes. • Clic sobre el archivo web.php. • Visualizar que la API REST Compra/create con su respectivo verbo Http Post. 	
Resultado esperado: El servicio web debe estar enlazado con su controlador e invocando el método create del controlador Compra	
Evaluación: Exitosa.	

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 3, Tarea de Ingeniería 3, HU-27 Registrar Compra

Tabla 19-2: Tarea de ingeniería 3, prueba de aceptación 3, Registro de compra

Prueba de aceptación	
Código: PA-09	Historia de Usuario: HU-27 Registrar Compra
Nombre: Realizar testing del servicio web creado	
Responsable: Kleber Hipo	Fecha: 22-05-2018
Descripción: Verificar que el la api retorne algún valor del recurso solicitado.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Estar creado el servicio API REST. 	

<ul style="list-style-type: none"> • Tener Instalado Postman. • Estar conectado a la base de datos • Poseer un token de un Usuario. • Poseer la ruta Compra/create
Pasos de ejecución: <ul style="list-style-type: none"> • Ejecutar Postman • Especificar la url de la autenticación con su respectivo método Http localhost:90/Compra/create. • Clic en Headers • Agregar API_token= 1dc58d73-9b74-11e8-9b63-9cad97d7ad3c • Clic en body. <pre>{ "IdProveedor":1, "Serie":"001", "Número":"001", "Total":"34.50", "Fecha":"2018-09-04", "NumElementos":4, "IdUsuarioCreate":1, "CompraDetalle":[{ "Id":"1", "Nombre":"Cama de dos plazas", "Color":"1", "Material":"1", "Cantidad":"8", "PrecioVenta":"6", "PrecioCompra":"3.10"}, {" "Id":"21", "Nombre":"cama matrimoniales para personas", "Color":"1", "Material":"2", "Cantidad":"4", "PrecioVenta":"4", "PrecioCompra":"1.10"}]}</pre> • Clic en Send.
Resultado esperado: Retornar los datos de los datos enviados en formato JSON y su estado de 201 created.
Evaluación: Exitosa.

Realizado por: Kleber Hipo, 2019

Tarea de Ingeniería 4, HU-27 Registrar Compra

Tabla 20-2: Tarea de ingeniería 4, Registro de compra

TAREA DE INGENIERÍA	
Historia de Usuario: HU-27 Registrar Compra	
Número de Tarea: TI-04	Nombre de Tarea: crear triggers en la tabla detalle compra
Tipo de Tarea: Desarrollo	Puntos Estimados: 8
Fecha Inicio: 23-05-2018	Fecha Fin: 23-05-2018
Programador Responsable: Kleber Hipo	
Descripción: Crear triggers que cuando realice una compra sea capaz de aumentar la cantidad de productos que posee el almacén	
Pruebas de Aceptación	
<ul style="list-style-type: none"> • Verificar que exista el triggers en la tabla tblcompradetalle. • Verificar que actualice la columna cantidad de la tabla tblproductodetalle. 	

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 1, Tarea de Ingeniería 4, HU-27 Registrar Compra

Tabla 21-2: Prueba de aceptación 1, Tarea de ingeniería 4 del Registro de Compra

Prueba de aceptación	
Código: PA-10	Historia de Usuario: HU-27 Registrar Compra
Nombre: Verificar que exista el triggers en la tabla tblcompradetalle	
Responsable: Kleber Hipo	Fecha: 23-05-2018
Descripción: visualizar el funcionamiento del trigers.	

Condiciones de Ejecución: <ul style="list-style-type: none"> • Tener creado la tabla tblcompradetalle • Tener creado la tabla tblproductodetalle
Pasos de ejecución: <ul style="list-style-type: none"> • Abrir un navegador con la url http://localhost:90/phpmyadmin • Clic en la base de datos bdcacha1. • Clic en el símbolo + de la tabla tblcompradetalle. • Clic en disparadores
Resultado esperado: Visualizar el trigger ActualizarCantidad.
Evaluación: Exitosa.

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 2, Tarea de Ingeniería 4, HU-27 Registrar Compra

Tabla 22-2: Tarea de ingeniería 4, prueba de aceptación 2, Registro de compra

Prueba de aceptación	
Código: PA-11	Historia de Usuario: HU-27 Registrar Compra
Nombre: Verificar que actualice la columna cantidad de la tabla tblproductodetalle.	
Responsable: Kleber Hipo	Fecha: 23-05-2018
Descripción: Este trigger será la encargada de tener un control de la cantidad de productos y si en caso que no posee el producto será la encargada de crear registros en la tabla tblproductodetalle.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Tener creado la tabla tblcompradetalle 	
Pasos de ejecución: <ul style="list-style-type: none"> • Realizar el testing de servicio en postman como la prueba de aceptación 09 	
Resultado esperado: Visualizar que la columna cantidad se aumente de la tabla tblproductodetalle	
Evaluación: Exitosa.	

Realizado por: Kleber Hipo, 2019

Tarea de ingeniería Frondent

Tarea de Ingeniería 5, HU-27 Registrar Compra

Tabla 23-2: Tarea de ingeniería 5, Registro de compra

TAREA DE INGENIERÍA
Historia de Usuario: HU-27 Registrar Compra

Número de Tarea: TI-05	Nombre de Tarea: Desarrollo del Frontend para el registro del Compra
Tipo de Tarea: Desarrollo	Puntos Estimados: 40
Fecha Inicio: 23-05-2018	Fecha Fin: 23-05-2018
Programador Responsable: Kleber Hipo	
Descripción: Crear la interfaz gráfica de registro de datos de la Compra	
Pruebas de Aceptación <ul style="list-style-type: none"> • Verificar que este creado la interfaz gráfica del registro de datos de la Compra. • Verificar la conexión con el backend Compra/create. 	

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 12, Tarea de Ingeniería 5 de la HU-27 Registrar Compra

Tabla 24-2: Prueba de aceptación 12, Tarea de ingeniería 5 del Registro de Compra

Prueba de aceptación	
Código: PA-12	Historia de Usuario: HU-27 Registrar Compra
Nombre: Verificar que este creado la interfaz gráfica del registro de datos de la Compra.	
Responsable: Kleber Hipo	Fecha: 24-05-2018
Descripción: Visualizar que este creado el formulario para el ingreso de los datos del Compra.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Tener creado el formulario del registro de Compra. 	
Pasos de ejecución: <ul style="list-style-type: none"> • Estar Autenticado en el Sistema • Clic en Compra y clic en el submenú Compra • Clic en Nuevo 	
Resultado esperado:	
Debe Visualizar el formulario de registro de los datos de la Compra.	
Evaluación: Exitosa.	

Realizado por: Kleber Hipo, 2019

Prueba de Aceptación 13, Tarea de Ingeniería 5 de la HU-27 Registrar Compra

Tabla 25-2: Prueba de aceptación 13, Tarea de ingeniería 5, Registro de compra

Prueba de aceptación	
Código: PA-13	Historia de Usuario: HU-27 Registrar Compra
Nombre: Verificar la conexión con el backend.	
Responsable: Kleber Hipo	Fecha: 25-05-2018
Descripción: comprobar que este realizado la conexión con la API REST Compra/créate tomando los datos que se ingresa el usuario por medio del formulario.	
Condiciones de Ejecución: <ul style="list-style-type: none"> • Haber creado la interfaz de usuario. • tener creado la conexión con backend con la ruta Compra/create. 	

<p>Pasos de ejecución:</p> <ul style="list-style-type: none"> • Abrir el directorio del backend • Clic en views. • Clic en Compra • Clic en Compra.vue
<p>Resultado esperado: Debe estar el método enviar especificando la ruta del API REST más los datos del Compra en formato JSON.</p>
<p>Evaluación: Exitosa.</p>

Realizado por: Kleber Hipo, 2019

Se documenta cada una de las historias de Usuario técnicas y pruebas de aceptación las cuales fueron desarrolladas por el correcto funcionamiento del sistema. **Anexo C.**

2.4.3.7 Manual de Usuario.

Este documento es de suma importancia es en donde se explica de forma detallada el uso del sistema realizado, aquí se detalla los pasos que se deben seguir para registro de las informaciones, mantenimiento y consultas e indicando los posibles mensajes que se le puede presentar al usuario en el momento cuando esté utilizando el sistema. Este material se encuentra disponible en el **Anexo D**

2.4.4 Fase de Cierre

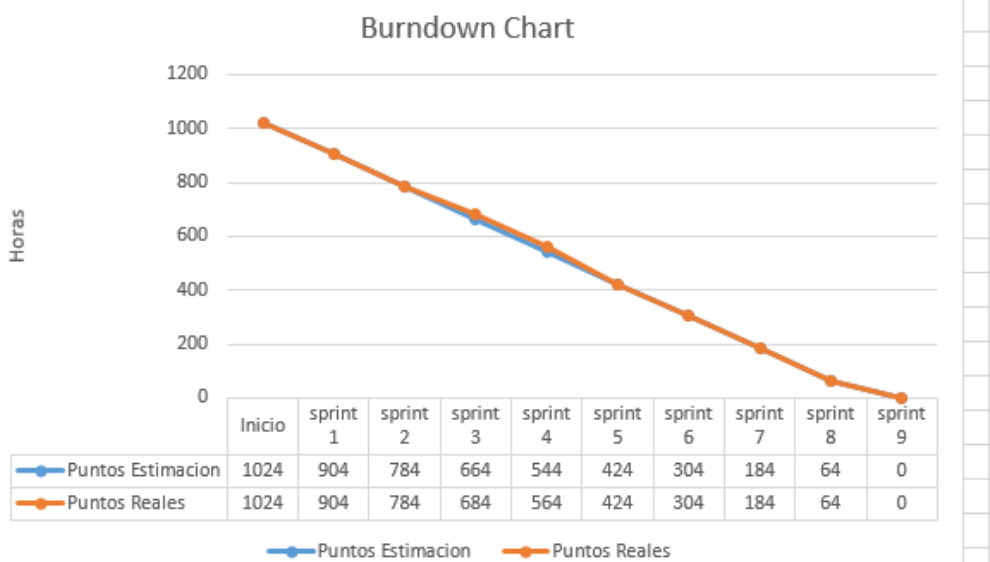


Gráfico 1-2: Gráfico Burndown Chart del proyecto

Realizado por: Kleber Hipo, 2019

Cada sprint tiene una duración de 120 puntos estimados lo cual representa 120 horas laborables (3 semanas), en el **Gráfico 1:2** se puede visualizar el avance del proyecto en cada una de las iteraciones, en las iteraciones 3 y 4 son las que tuvo mayor problema ya que en estas iteraciones se planifico la gestión de compras y pedidos, estas llevaron más tiempo de lo planificado en el

sprint 7 de los reportes se subestimo por lo tanto los 20 puntos estimados se los logro recompensar en este sprint ademas en el 8 de la misma forma hubo una subestimación lo cual del sprint 4 se recompensa de la misma manera con este sprint.

2.5 Método de Evaluación

2.5.1. Planificación

Para analizar la eficiencia del software se utiliza la norma ISO 9126, se analiza el comportamiento de tiempo y la utilización de recursos. En el comportamiento del tiempo se realiza tres mediciones de tiempo de los 13 procesos de la forma manual y sistematizada, estos datos se los toma con un cronómetro inicializado en 0 en ambos casos, posterior se realiza una comparación de los datos promedios de forma sistematizada y manual. Para la utilización de los recursos en la métrica de densidad de mensajes Entrada y salida I/O para cada tarea se cuenta de forma manual las líneas de código generados para la solución del problema y las condiciones que fueron necesarias para la solución. En uso del Procesador y de Memoria se realizan las 3 mediciones con el uso del sistema. Para las 13 tareas se toma los datos, estos se realiza con la ayuda del Administrador de tareas de nuestro sistema operativo.

2.5.2. Población

Para medir la eficiencia del software al sistema web del Almacén de muebles “Cacha” se toma como una población los 45 requerimientos funcionales las cuales son definidas como historias de usuario en el desarrollo del mismo.

2.5.3. Muestra

Para la muestra se aplica el muestreo no aleatorio o también llamado no probabilísticos de conveniencia, lo cual consiste en seleccionar un grupo de muestras que permite ahorrar tiempo, recursos y que son fáciles de poder acceder a dichos datos (Universidad Autónoma de Baja California, 2017), la principal desventaja es este tipo de muestreo es que los resultados obtenidos tienen una validez solo para el grupo del campo de estudio seleccionado (Nepo Villano, 2017).

Para seleccionar las muestras por conveniencia se tomó como base las tareas más frecuentes que se realizan a diario, lo cuales están en la **tabla 26-3**.

Tabla 26-2: Lista de las Historias de Usuario para el caso de estudio

Módulo	Tarea
Compra	<ul style="list-style-type: none">➤ Registrar Compra➤ Modificar Compra
Pedido	<ul style="list-style-type: none">➤ Registrar Pedido➤ Modificar un Pedido➤ Eliminar un Pedido➤ Registrar Cliente➤ Modificar Cliente➤ Generar Factura➤ Generar venta a crédito
Cobro	<ul style="list-style-type: none">➤ Cobro de un crédito
Reportes	<ul style="list-style-type: none">➤ Listado de las compras entre fechas➤ Listado de los Pedidos entre fechas➤ Listado de los productos con un stock mínimo

Realizado por: Kleber Hipo, 2019

2.5.4. Determinación del escenario de prueba.

Los escenarios que se tomaron en cuenta para el los tiempos de respuesta, uso del procesador y memoria fueron las siguientes:

- 1. Registrar Compra:** se realizó el registro de tres compras con un detalle de compras de 3 productos, en cada uno de ellos se fueron capturando los tiempos de respuesta y a su vez capturando el uso de la memoria y del procesador que se visualiza el administrador de tareas
- 2. Modificar Compra:** Anteriormente se debe estar registrado las 3 compras, se realizó la modificación de tres compras con un detalle de compras de 3 productos, en cada uno de ellos se fueron capturando los tiempos de respuesta con el cronometro y a su vez capturando el uso de la memoria y del procesador que se visualiza el administrador de tareas
- 3. Registrar Pedido:** Se realizó el registro de tres Pedido con un detalle de pedido de 3 productos, en cada uno de ellos se fueron capturando los tiempos de respuesta y a su vez capturando el uso de la memoria y del procesador que se visualiza el administrador de tareas
- 4. Modificar Pedido:** Anteriormente se debe estar registrado los 3 pedido, se realizó la modificación de tres pedidos con un detalle de compras de 3 productos, en cada uno de ellos se fueron capturando los tiempos de respuesta con el cronometro y a su vez capturando el uso de la memoria y del procesador que se visualiza el administrador de tareas.
- 5. Eliminar un Pedido:** Anteriormente se debe estar registrado los 3 pedidos, se realizó la eliminación de tres compras con un detalle de compras de 3 productos, en cada uno de ellos se fueron capturando los tiempos de respuesta con el cronometro y a su vez capturando el uso de la memoria y del procesador que se visualiza el administrador de tareas
- 6. Registrar cliente:** para esta tarea se registró 6 datos obligatorios que se requiere para un cliente, y con este proceso se realizó una repetición de 3 veces con diferentes datos, en cada

repetición se fueron capturando los tiempos de respuesta con el cronometro y a su vez capturando el uso de la memoria y del procesador que se visualiza el administrador de tareas

7. Modificar Cliente: En esta tarea se realiza la modificación de 6 datos del formulario, y con este proceso se realizó una repetición de 3 veces con diferentes datos, en cada repetición se fueron capturando los tiempos de respuesta con el cronometro y a su vez capturando el uso de la memoria y del procesador que se visualiza el administrador de tareas

8. Generar Factura: para generar una factura se realizó un pedido el cual contiene 3 productos, esta tarea se realiza 3 veces con pedidos de diferentes clientes. en cada repetición se fueron capturando los tiempos de respuesta con el cronometro y a su vez capturando el uso de la memoria y del procesador que se visualiza el administrador de tareas

9. Generar venta a crédito: para la generación de ventas a credito se realizó para un plazo de 1 mes días equivalente a 4 meses con pagos mensuales, este proceso se repite durante 3 veces y en cada repetición se fueron capturando los tiempos de respuesta con el cronometro y a su vez capturando el uso de la memoria y del procesador que se visualiza el administrador de tareas

10. Cobro de un crédito: se han realizado 3 repeticiones de cobros de cuotas de pagos mensuales en la cual se ha capturado el tiempo de respuesta y el uso del procesado y memoria estas dos últimas se visualiza en el administrador de tareas.

11. Listado de los productos con un stock mínimo: la consulta consto listado de los productos con un stock mínimo de con repeticiones de 3 veces

12. Listado de los Pedidos entre fechas: tres repeticiones en la consulta con los pedidos que se han realizado el último mes.

13. Listado de las Compras entre fechas: tres repeticiones en la consulta con las compras que se han realizado el último mes.

CAPITULO III

3. MARCO DE RESULTADOS, DISCUSIÓN Y ANÁLISIS

En el presente capítulo se muestra el análisis realizado de la eficiencia del software aplicando el estándar ISO 9126 al sistema desarrollado del almacén de muebles Cacha, estableciendo indicadores en base de la situación actual de la empresa con el fin realizar cada una de las actividades requeridas para el cumplimiento de los objetivos planteados en el presente trabajo de titulación.

3.1. Tiempo de Respuesta

3.1.1. Análisis de los Procesos en la actualidad en el Almacén

En el almacén de muebles Cacha en la actualidad toda la información y los procesos son llevados de forma manual lo cual esto genera inconvenientes a la hora de brindar una información adecuada al cliente. Para la recopilación de los tiempos se tomaron tareas que se desarrollan con mayor frecuencia en el almacén las cuales se detallaron en la **tabla 26-2**. Los tiempos de cada tarea se detalla en la **tabla 1-3**.

Tabla 1-3: Tiempos de respuesta de los procesos de forma manual

Módulo	Tarea	Tiempo 1	Tiempo 2	Tiempo 3	Promedio (Min)
Compra	Registrar Compra	5.57	6.55	5.55	5.89
	Modificar Compra	6.57	4.00	5.55	5.37
Pedido	Registrar Pedido	5.57	5.22	4.55	5.11
	Modificar un Pedido	6.57	6.75	5.55	6.29
	Eliminar un Pedido	5.57	4.60	3.55	4.57
	Registrar Cliente	4.57	3.72	3.55	3.94
	Modificar Cliente	3.57	2.50	3.55	3.21
	Generar Factura	5.57	3.55	4.55	4.56
	Generar venta a crédito	7.57	8.98	6.55	7.70
Cobro	Cobro de un crédito	5.57	3.37	4.55	4.49
Reportes	Listado de las compras entre fechas	6.57	5.98	6.55	6.37

	Listado de los Pedidos entre fechas	7.57	6.87	5.55	6.66
	Listado de los productos con un stock mínimo	7.57	8.72	6.55	7.61
PROMEDIO GENERAL					5.52

Realizado por: Kleber Hipo, 2019

En la **tabla 1-3** se puede observar los tiempos de respuesta que fueron tomados para cada una de los procesos, donde el proceso de generación de ventas a crédito son las que llevan más tiempo en la actualidad.

3.1.2. Análisis de los Proceso utilizando el sistema web

Los tiempos de respuesta de lo realiza de la misma forma pero con el uso del sistema, en algunos tareas los tiempos fueron en minutos y segundos, para lo cual se tuvo que convertir los segundos a minutos dividiendo por 60 el cual da el tiempo de respuesta en minutos, estos datos se puede visualizar en la **tabla 2-3**.

Tabla 2-3: Tabla de tiempos de Respuesta con el Sistema

Tarea	Tiempo 1 (min)	Tiempo 2 (min)	Tiempo 3 (Min)	Tiempos Promedios
Registrar Compra	1.85	1.65	1.72	1.74
Modificar Compra	1.28	0.88	0.90	1.02
Registrar Pedido	1.25	0.98	1.05	1.09
Modificar un Pedido	1.33	0.75	0.77	0.95
Eliminar un Pedido	0.22	0.18	0.22	0.21
Registrar Cliente	0.78	1.10	1.02	0.97
Modificar Cliente	0.70	0.72	0.75	0.72
Generar Factura	0.40	0.38	0.33	0.37
Generar venta a crédito	0.63	0.60	0.68	0.64
Cobro de un crédito	0.35	0.40	0.33	0.36
Listado de las compras entre fechas	0.22	0.20	0.20	0.21
Listado de los Pedidos entre fechas	0.27	0.27	0.27	0.27
Listado de los productos con un stock mínimo	0.23	0.23	0.22	0.23
Promedio	0.73	0.64	0.65	0.67

Realizado por: Kleber Hipo, 2019

3.1.3. Comparación de Resultados

Para un análisis de los datos se procedieron a comparar los tiempos de respuesta entre los procesos que se desarrollan de forma manual y con el uso del sistema lo cual se muestra en la **tabla: 3-3**.

Tabla 3-3: Comparación de los tiempos de Respuesta con y sin el sistema

Tarea	Sin Sistema (Min)	Con Sistema (Min)	Tiempo Reducción del sin con el sistema (Min)
Registrar Compra	5.89	1.74	4.15
Modificar Compra	5.37	1.02	4.35
Registrar Pedido	5.11	1.09	4.02
Modificar un Pedido	6.29	0.85	5.34
Eliminar un Pedido	4.57	0.21	4.36
Registrar Cliente	3.94	0.97	2.97
Modificar Cliente	3.21	0.72	2.49
Generar Factura	4.56	0.37	4.19
Generar venta a crédito	7.70	0.64	7.06
Cobro de un crédito	4.49	0.36	4.13
Listado de las compras entre fechas	6.37	0.21	6.16
Listado de los Pedidos entre fechas	6.66	0.27	6.39
Listado de los productos con un stock mínimo	7.61	0.23	7.38
Promedio	5.52	0.67	4.85

Realizado por: Kleber Hipo, 2019

En la **tabla 3-3** se puede visualizar que con el uso del sistema los tiempo de respuesta se ha logrado disminuir notablemente, se tomaron en cuenta el tiempo en minutos que se demora para completar la tarea de forma tradicional y este valor es tomado como el 100%, y el de uso del sistema dependiendo del valor anterior se ha logrado obtener su porcentaje, esta nos indica que la forma tradicional se demora de 5.52 minutos en promedio y con los procesos automatizados tiene una duración de 0.67 minutos en promedio, se tiene una reducción de 4.85 minutos lo cual representa un 87.86% de reducción.

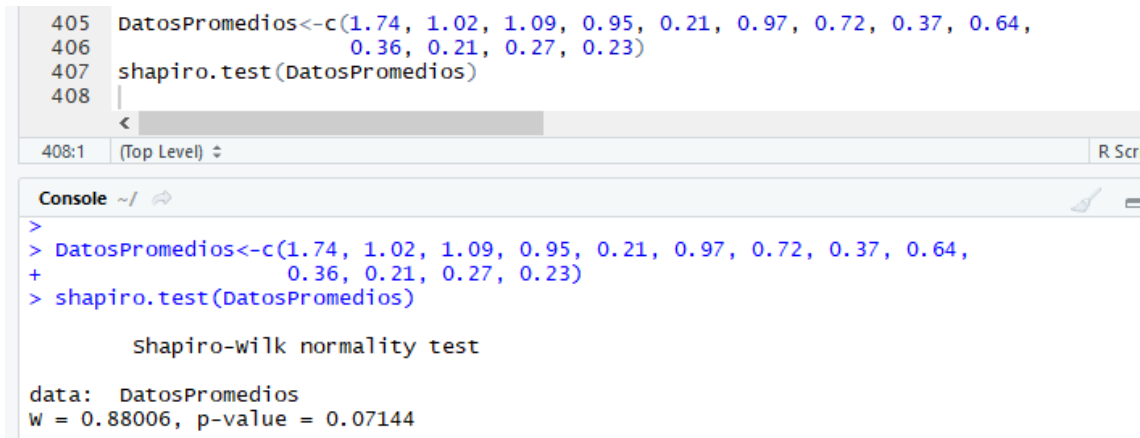
3.1.4. Análisis de Datos con t-student

Para aplicar la estadística descriptiva inicialmente se debe evaluar la normalidad de los datos recopilados de los requerimientos expuestos, siendo estos los tiempos manuales y automatizados, para lo cual son tomados los tiempos promedios de la forma tradicional y con el uso del sistema.

Para evaluar la normalidad de los datos promedios obtenidos con el uso de la forma tradicional se utilizó el Software estadístico RStudio.

1. Estadístico de Prueba

Se aplicó Shapiro-Wilk test para verificar si los datos recopilados son normales o no. A continuación, se puede observar en las figuras de acuerdo a los datos promedios.

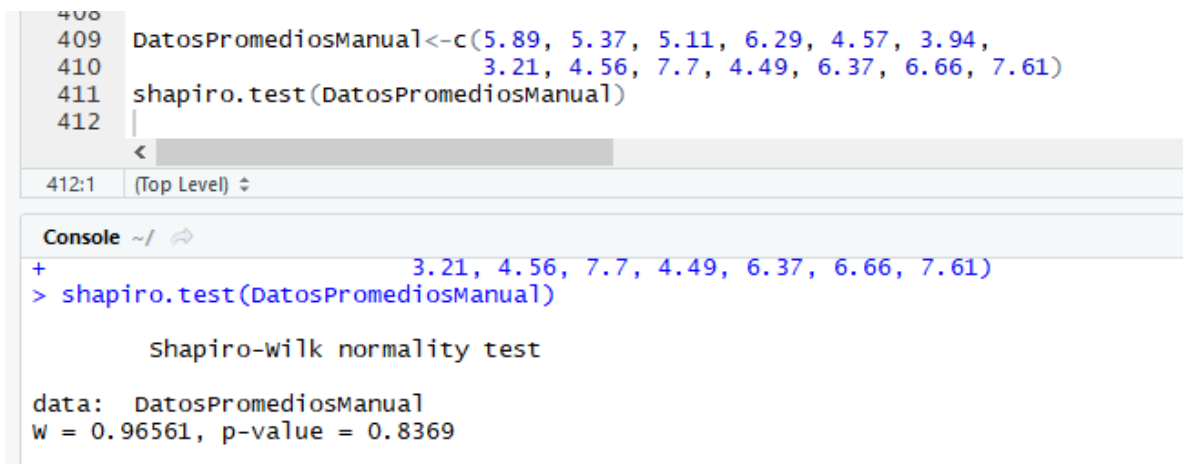


```
405 DatosPromedios<-c(1.74, 1.02, 1.09, 0.95, 0.21, 0.97, 0.72, 0.37, 0.64,  
406                    0.36, 0.21, 0.27, 0.23)  
407 shapiro.test(DatosPromedios)  
408 |  
< [Progress bar]  
408:1 (Top Level) [R Scr]  
  
Console ~/ [?]  
>  
> DatosPromedios<-c(1.74, 1.02, 1.09, 0.95, 0.21, 0.97, 0.72, 0.37, 0.64,  
+                   0.36, 0.21, 0.27, 0.23)  
> shapiro.test(DatosPromedios)  
  
      Shapiro-wilk normality test  
  
data:  DatosPromedios  
w = 0.88006, p-value = 0.07144
```

Figura 1-3: Tiempos Promedios Automatizados en Shapiro-wilk

Realizado por: Kléber Hipo, 2019.

En la **Figura 1-3** se observa que los datos obtenidos corresponden que vienen de una distribución normal puesto que p-value es 0.07144 es mayor al porcentaje de error 0.05.



```
408  
409 DatosPromediosManual<-c(5.89, 5.37, 5.11, 6.29, 4.57, 3.94,  
410                        3.21, 4.56, 7.7, 4.49, 6.37, 6.66, 7.61)  
411 shapiro.test(DatosPromediosManual)  
412 |  
< [Progress bar]  
412:1 (Top Level) [R Scr]  
  
Console ~/ [?]  
+ 3.21, 4.56, 7.7, 4.49, 6.37, 6.66, 7.61)  
> shapiro.test(DatosPromediosManual)  
  
      Shapiro-wilk normality test  
  
data:  DatosPromediosManual  
w = 0.96561, p-value = 0.8369
```

Figura 2-3: Tiempos Promedios Manuales en Shapiro-wilk

Realizado por: Kléber Hipo, 2019.

En la **Figura 1-3** se observa que los datos obtenidos corresponden que vienen de una distribución normal puesto que p-value 0.8369 es mayor al porcentaje de error 0.05.

Se comprueba que los datos del Sin y Con el uso del sistema viene de una distribución normal por lo cual se procede a aplicar la prueba T.

Planteamiento de la hipótesis:

Hipótesis Nula (H_0) = El tiempo promedio de la aplicación desarrollada es igual al tiempo promedio de los procesos realizados manualmente de los requerimientos evaluados

Hipótesis Alterna (H_1) = El tiempo promedio automatizado es diferente al tiempo promedio manual de los requerimientos evaluados

1. Nivel de Significancia

$$\alpha = 0.05$$

2. Estadístico t

Se aplicó la prueba t para medias de dos muestras emparejadas ya que los datos obtenidos fueron del antes y después, estos datos también son cuantitativos, normales y menores a 30, por tanto, el estadístico t en este caso de estudio es de 11,57.

Tabla 4-3: Prueba t para muestras emparejadas

Prueba t-para dos muestras emparejadas	
Grados de libertad	12
Estadístico t	11.57
p-value dos colas	7.25549E-08
Valor crítico dos colas	2.18

Realizado por: Kléber Hipo, 2019.

En la **tabla 4-3** se observa de se tiene los grados de libertad de 12, el valor estadístico t de 11.57, el valor de p-value de dos colas 7.25.549E-08 y un valor crítico para dos colas de 2.18

3. Establecer regla de decisión

En el **Gráfico 1-3**, se puede visualizar la distribución t-Student para medias de dos muestras emparejadas, obteniendo el estadístico de prueba, la zona de aceptación y la zona de rechazo marcada de color rojo, siendo esta en la cola derecha e izquierda dado que la condición de comprobación es diferente en la Hipótesis Alternativa (H_1).

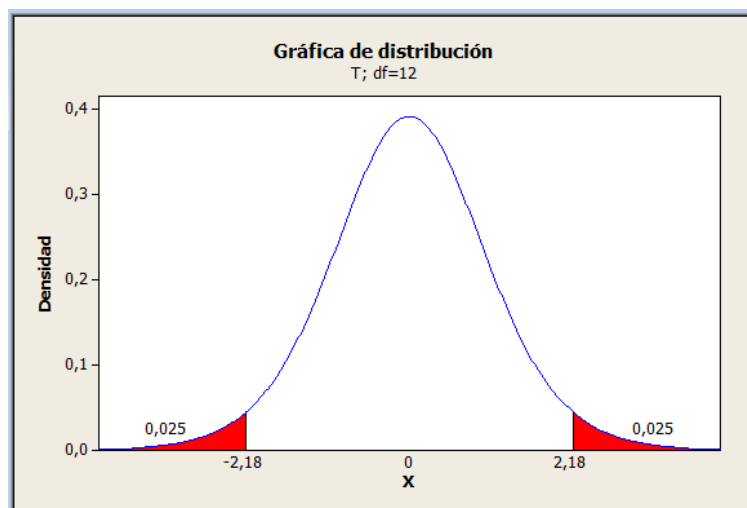


Gráfico 1-3: Gráfico de distribución t de los requerimientos evaluados
Realizado por: Kléber Hipo, 2019.

A continuación, en la **Tabla 4-3**, se puede observar los datos que se utilizó en la regla de decisión para los requerimientos evaluados, el (t_c) es el valor estadístico t la misma que se puede visualizar en la **Tabla 4-3**, el (tt) es el valor que se puede observar en el **Gráfico 1-3**.

Tabla 5-3: Regla de decisión de los datos

T calculado (t_c)	>	T tratado (tt)	
11,57	>	-2,18	Rechazo H_0
(p)	<	(α)	
7.25549E-08	<	0,05	Rechazo H_0

Realizado por: Kléber Hipo, 2019.

4. Toma de decisiones

Dado el valor t_c de 11,57 indica que es mayor al valor crítico de la cola izquierda (-2,18) rechazando la H_0 ; es decir que el tiempo promedio sistematizado es diferente al tiempo promedio tradicional de los requerimientos evaluados denotando un nivel de significancia de 0,05, como se puede observar en el **Gráfico 1.3**.

Para encontrar los tiempos de reducción se trabajó con las medias de los promedios del sin y con el sistema en la cual son 5.52 y 0.67 minutos respectivamente.

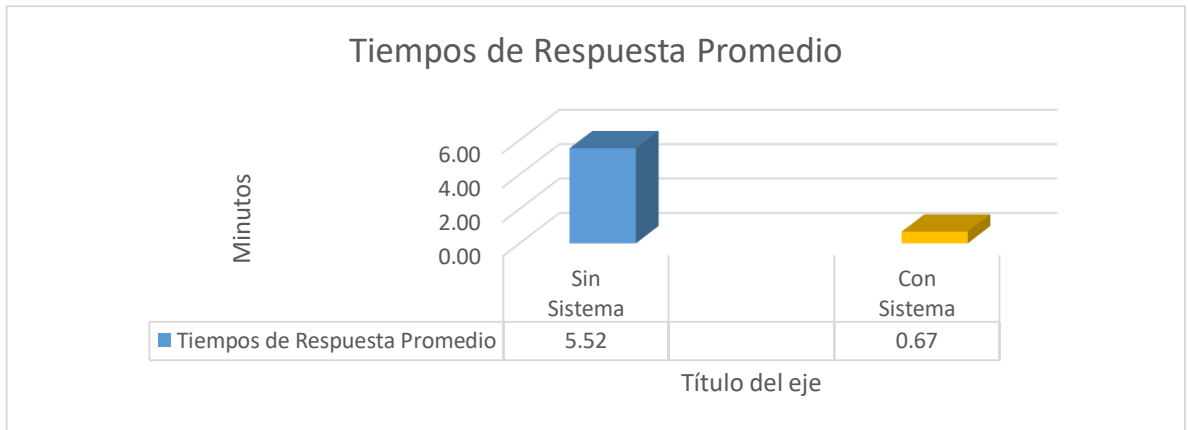


Gráfico 1-3: Gráfico de comparación de los tiempos de respuesta
Realizado por: Kleber Hipo, 2019

En el **Gráfico 1-3** se puede observar que para las 13 tareas tienen una media de 5.52 minutos de forma manual y con el uso del sistema de una media de 0.67 minutos esto en segundos tendría una aproximación de 40.20 segundos. En el **Gráfico 2-3** se representa el 100% de los tiempos de la forma tradicional y el uso del sistema se requiere de 12.14% del tiempo utilizado de la forma manual para completar las tareas, esto representa que se ha disminuido el tiempo en un 87.86% con respecto de la forma tradicional.

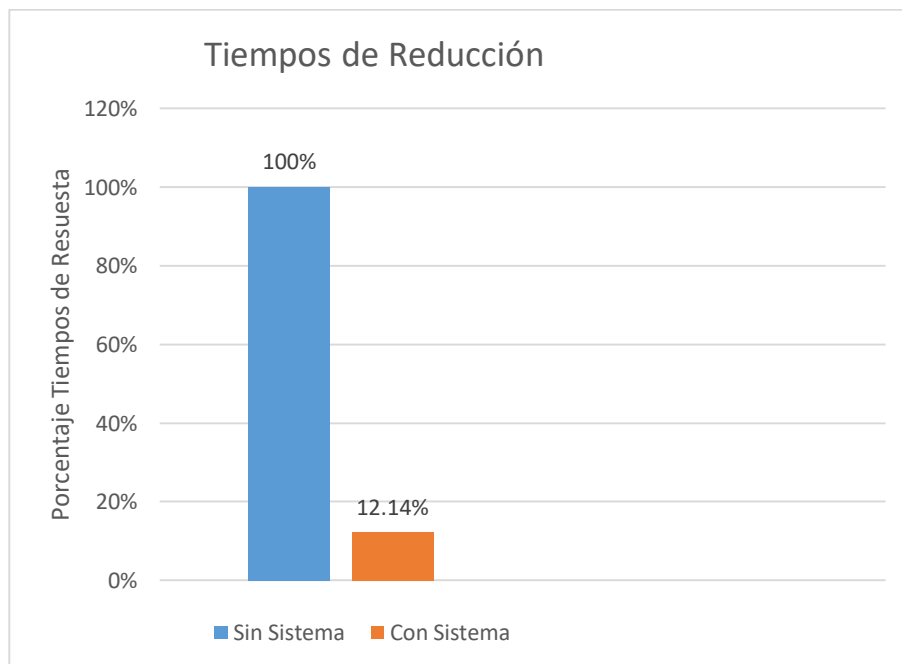


Gráfico 2-3: Porcentaje de tiempos de Respuestas
Realizado por: Kleber Hipo, 2019

3.2. Utilización de Recursos

La utilización de recursos es otro de los parámetros que se debe evaluar para la eficiencia según la norma ISO 9126 para lo cual se ha evaluado los siguientes.

En el artículo científico de (Patiño Rosado and Erika Paola, 2018,p.67) menciona que dentro de la métrica de la utilización de recursos en la eficiencia externa e interna existen sub-características , en la eficiencia interna son:

- Uso del procesador
- Uso de la memoria RAM

A su vez en la calidad externa menciona que existen:

- Utilización I/O Densidad de mensaje
- Utilización de Memoria Densidad de mensaje

Para nuestro caso de estudio se logró seleccionar las siguientes sub-características

- Uso de Memoria
- Uno del Procesador
- Densidad de mensajes de uso entrada /salida

3.2.1. *Densidad del mensaje de uso de entrada/salida*

Para los controles de los posibles errores que pueden tener el sistema se realizó en control en la parte de Backend en el controlador y en el Frondend en la vista con el lenguaje JavaScript

Tabla 6-3: Tabla de características de utilización de recursos densidad de mensaje

Características	Eficiencia Interna
Sub- característica	Utilización de recursos
Métrica	Densidad del mensaje de uso entrada y salida
Propósito de la métrica	Esto permite contar el número de errores que pertenece de I/O, la advertencia y comparar el número estimado de líneas de código.
Tipo de Medida	x=contable/contable A= contable B=contable
Interpretación	El mayor es mejor
Tipo de escala	Absoluto
Fuente de medición	Código fuente

Realizado por: Kleber Hipo, 2019

Fuente: (Vivanco Villamar, 2011)

Para el control de errores en todas las tareas siempre se validan en el lado del servidor que los datos recibidos este en formato JSON a excepción de las tareas inserción y modificación del cliente de igual manera se validó en el cliente utilizando JavaScript.

Para el valor de A se tomó en cuenta la cantidad de controles que se puso en cada uno de los procesos y para el Valor de B las cantidades de Línea de código que fueron utilizadas para el

completar el desarrollo de dicha tarea, para ese procesos se debe realizar en el código fuente de la aplicación.

1. Registrar Compra

Número de errores controlados en la tarea =4

1.1. Los datos que se reciben sea de objeto JSON caso contrario rechace la petición.

```
255     $data = $request->json()->all();
256     if ($request->isJson()) {
257
258         $Compra = Compra::create([
259             'IdProveedor' => $data['IdProveedor'],
```

Figura 1-3: Error Controlado que los datos sean Objeto JSON

Realizado por: Kleber Hipo, 2019

1.2. Enviar datos vacíos de la factura de venta

```
{ 788     } else {
789         this.$swal("Modulo Compras!", "Hay campos del formulario incorrecto revise y vuelva a enviarlo.",
790             "error");
791     }
```

Figura 2-3: Error Controlado cuando envié datos vacíos del cliente

Realizado por: Kleber Hipo, 2019

1.3. Enviar compra vacía.

```
784     }
{ 785     } else {
786         this.$swal("Modulo Compras!", "Agregue Productos a la Compra ", "error");
787     }
788     } else {
```

Figura 3-3: Error Controlado cuando envié sin productos comprados

Realizado por: Kleber Hipo, 2019

1.4 cuando los datos del producto comprado son erróneos.

```
781     } else {
782         this.$swal("Modulo Compras!", "Revise los Datos en Detalle de los Productos y vuelva a enviar ",
783             "error");
784     }
```

Figura 4-3: Error Controlado cuando los datos no este correctos del producto

Realizado por: Kleber Hipo, 2019

2. Modificar Compra

Número de errores controlados en la tarea =5.

Los errores controlados e son los mismos que en la tarea de registro de Compras.

2.1 cuando se trate de modificar una compra no existente

```
118     $data = $request->json()->all();
119     $Compra = Compra::find($data['Id']);
120
121     if ($Compra) {
122         $Compra->Id = $data['Id'];
```

Figura 5-3: Error Controlado cuando trate de modificar compra no existente

Realizado por: Kleber Hipo, 2019

3. Registrar Pedido

Número de errores controlados en la tarea =6

3.1 Error cuando envié datos que no sea formato JSON

```
262     try {
263         $data = $request->json()->all();
264         .if ($request->isJson()) {
265
266             $Pedido = Pedido::create([
```

Figura 6-3: Error Controlado cuando no envié el formato

Realizado por: Kleber Hipo, 2019

3.2 Error cuando no se envié los datos del cliente

```
        band=true;
    }else{
        this.$swal("Modulo de Pedidos!", "Seleccione un Cliente", "error");
        band=false;
    }
}
```

Figura 7-3: Error Controlado cuando no haya seleccionado el cliente

Realizado por: Kleber Hipo, 2019

3.3 Error cuando haya agregado ningún producto

```
    }
    }else{
        this.$swal("Modulo de Pedidos!", "Agregue algun Producto al Pedido", "error");
        band=false;
    }
}
```

Figura 8-3: Error Controlado cuando no haya agregado productos al pedido cliente

Realizado por: Kleber Hipo, 2019

3.4 Error cuando no haya agregado ningún producto al pedido

```
    }
    }else{
        this.$swal("Modulo de Pedidos!", "Agregue algun Producto al Pedido", "error");
        band=false;
    }
}
```

Figura 9-3: Error Controlado cuando no haya agregado productos al pedido cliente

Realizado por: Kleber Hipo

3.5 Error cuando no el precio de venta no es menor al precio de compra

```
    }else{
        this.$swal("Modulo de Pedidos!", "El Precio de venta "+self.PrecioVenta+" debe estar entre "+self.PrecioMin+" "+self.PrecioMax, "warning");
        self.PrecioVenta= self.PrecioMin;
    }
}
```

Figura 10-3: Error Controlado cuando el precio de venta sea menor que al de compra

Realizado por: Kleber Hipo, 2019

3.6 Error cuando no haya el stock suficiente para la cantidad solicitada


```

..... $res['success'] .= false;
..... $res['message'] .= 'Persona con esta cedula ya existe';
..... $res['data'] .= $Persona;
..... return $res;

```

Figura 15-3: Error Controlado de cedula duplicada

Realizado por: Kleber Hipo, 2019

6.2. Error cuando la foto no pueda subir al servidor

```

} else {
..... $res['success'] .= false;
..... $res['message'] .= 'Hubo un error subiendo el archivo, por favor inténtalo de nuevo!';
..... return $res;
}

```

Figura 16-3: Error Controlado cuando no se suba la foto

Realizado por: Kleber Hipo, 2019

6.3. Error cuando envié formulario en blanco

```

..... } else {
..... this.$swal("Envío Datos Cliente!", "Hay campos del formulario incorrecto revise y vuelva a
..... enviarlo", "error");
..... }
},

```

Figura 17-3: Error Controlado Campos vacíos.

Realizado por: Kleber Hipo, 2019

7. Modificar Cliente

Número de errores controlados en la tarea =4

Contiene los mismos errores de la tarea de registro de clientes

7.1. Error modifique cliente no existente

```

..... $Pers = Persona::where('Cedula', '=', $_POST['Cedula'])->get();
..... if ($Pers->count() == 1) {
.....     $target_path = "Persona/";
.....     $NombreFoto = "";
..... }

```

Figura 18-3: Error Controlado de modificación de clientes no existentes.

Realizado por: Kleber Hipo, 2019

8. Generar Factura

Número de errores controlados en la tarea =1

8.1. Error de envío de datos en el formato diferente de JSON para generar una factura

```

{
    try {
        $data = $request->json()->all();
        if ($request->isJson()) {
            $Factura = Factura::create([

```

Figura 19-3: Error Controlado, envío datos no JSON para generar Factura

Realizado por: Kleber Hipo, 2019

9. Generar venta a crédito

Número de errores controlados en la tarea =2

9.1. Error en el envío de datos en el formato diferente de JSON para generar una Venta a crédito de una factura

```
$data = $request->json()->all();  
if ($request->isJson()) {  
    $Credito = Credito::create([  
        'TotalFactura' => $data['Total'],
```

Figura 20-3: Error Controlado, en el envío de datos no JSON para la venta a crédito

Realizado por: Kleber Hipo, 2019

9.2. Error cuando trate de enviar datos antes de haber generado la tabla de amortizaciones

```
let band=true;  
if (self.TablaAmortizacion1.lengt == 0) {  
    this.$swal("Modulo Factura!", "Debe Generar la tabla de Amortizacion", "error");  
    band=false;
```

Figura 21-3: Error Controlado en el envío de datos sin la tabla de amortización

Realizado por: Kleber Hipo, 2019

10. Cobro de un crédito

Número de errores controlados en la tarea =2

10.1. Error cuando los datos enviados no contienen el formato JSON

```
try {  
    if ($request->isJson()) {  
        $now = new \DateTime();  
        $HoraActual = $now->format('Y-m-d H:i:s');  
        $data = $request->json()->all();
```

Figura 22-3: Error Controlado cuando los datos no tiene formato JSON Para Crédito

Realizado por: Kleber Hipo, 2019

10.2. Error cuando trate de enviar el monto mayor al crédito a cancelar

```
} else {  
    this.$swal("Modulo Cobros!", "La Cantidad No puede ser igual 0 y a su vez mayor al Total de  
    Credito "+ TotalAux + "", "error");  
}
```

Figura 23-3: Error Controlado, monto mayor que el crédito a pagar

Realizado por: Kleber Hipo, 2019

11. Listado de las compras entre fechas

Número de errores controlados en la tarea =1

11.1. Error cuando los datos enviados no contienen el formato JSON

```
try {  
    if ($request->isJson()) {  
        $data = $request->json()->all();  
        $FechaInicio = $data["FechaInicio"];
```

Figura 24-3: Error Controlado, datos recibidos no es JSON para Reporte de Compras

Realizado por: Kleber Hipo, 2019

12. Listado de los Pedidos entre fechas

Número de errores controlados en la tarea =1

12.1. Error cuando los datos enviados no contienen el formato JSON

```
try {
    .if-($request->isJson())-{
        .....$data.....=$request->json()->all();
        .....$IdProducto.....=$data["IdProducto"];
```

Figura 25-3: Error Controlado, datos recibidos no es JSON para Reporte de pedido

Realizado por: Kleber Hipo, 2019

13. Listado de los productos con un stock mínimo

Número de errores controlados en la tarea =1

13.1. Error cuando los datos enviados no contienen el formato JSON

```
try {
    .if-($request->isJson())-{
        .....$data.....=$request->json()->all();
        .....$Cantidad.....=$data["Cantidad"];
```

Figura 26-3: Error Controlado, datos recibidos no es JSON para Reporte del stock mínimo

Realizado por: Kleber Hipo, 2019

En la **tabla 6-3** constan todos los datos obteniendo en la métrica de la eficiencia interna.

Tabla 7-3: Resultados de la métrica densidad del mensaje I/O

Tarea	Medición, fórmula y cálculo de datos	Resultado
Registrar Compra	A=4; B = 150;	0.027
Modificar Compra	A=5; B =155;	0.032
Registrar Pedido	A=6; B =125;	0.048
Modificar un Pedido	A=7; B =108;	0.065
Eliminar un Pedido	A=2; B =56;	0.036
Registrar Cliente	A=3; B =101;	0.030
Modificar Cliente	A=4; B =94;	0.043
Generar Factura	A=1; B =110;	0.009
Generar venta a crédito	A=2; B =78;	0.026
Cobro de un crédito	A=2; B =75;	0.027
Listado de las compras entre fechas	A=1; B =58;	0.017
Listado de los Pedidos entre fechas	A=1; B =52;	0.019
Listado de los productos con un stock mínimo	A=1; B =56;	0.018
TOTAL		0.40

Realizado por: Kleber Hipo, 2019

3.2.2. Métrica Uso de memoria RAM

Para la evaluación de esta métrica se utilizó el administrador de tareas que nos proporciona nuestra pc (Windows), esta métrica es la cual nos indica que cantidad de memoria del computador utilizar para completar la tarea esto está en MB.

Según (López Yepes, 2013) menciona que para la medición de esta métrica se debe considerar lo siguiente.

Tabla 8-3: Evaluación de la métrica de uso de memoria RAM

Nombre:	Utilización de la Memoria
Propósito:	¿Cuál es el tamaño de la memoria estimado que el producto va a ocupar para completar una tarea específica?
Método de aplicación:	Estimar lo requisitos de memoria
Medición, Formula	X= tamaño en MB (calculada o simulada)
Interpretación	Cuando menor es mejor
Tipo de escala	Proporción
Tipo de medida	X=Tamaño
Fuente de Medición	Tamaño estimado de la medición
ISO/IEC 12207 SLCP:	Verificación
Audiencia:	Desarrolladores

Fuente: (López Yepes, 2013)

Realizado por: Kleber Hipo, 2019

En cada una de las pruebas se colocó la primera captura tanto del uso del memoria y del procesador las demás capturas 2 y 3 se pueden localizar en **el anexo E**.

Registrar Compra

Para la tarea de registrar una compra utiliza un promedio de 558.1 MB, para este procesos se los realizo 3 repeticiones, se utilizó el administrador de tareas he ir capturando las pantallas con un total de 3 productos en el detalle de compras.

Nombre	25% CPU	83% Memoria	75% Disco	0% Red	0% GPU	Motor de la
> Microsoft Word (32 bits)	0.3%	24.7 MB	0 MB/s	0 Mbps	0%	
> Google Chrome (17)	9.1%	558.1 MB	24.3 MB/s	0 Mbps	0%	GPU 0 -

Figura 27-3: Utilización de la memoria en la tarea de Registrar Compra

Realizado por: Kleber Hipo, 2019

Modificar Compra

Para la tarea de modificar una compra utiliza un promedio de 493.4 MB.

Nombre	17% CPU	83% Memoria	23% Disco	0% Red	2% GPU	Motor de la GPU
> Google Chrome (15)	5.0%	493.4 MB	0.1 MB/s	0 Mbps	0.2%	GPU 0 - 3D

Figura 28-3: Utilización de la memoria en la tarea de modificar Compra
Realizado por: Kleber Hipo, 2019

Registrar Pedido

Para la tarea de registrar los pedidos de los clientes utiliza un promedio de 327.5 MB.

Nombre	9% CPU	77% Memoria	27% Disco	0% Red	0% GPU	Motor de la GPU
> Google Chrome (13)	0.5%	327.5 MB	0.1 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 29-3: Utilización de la memoria en la tarea de Registrar Pedido
Realizado por: Kleber Hipo, 2019

Modificar pedido

Para la tarea de modificar un pedido utiliza un promedio de 347.8 MB.

Nombre	13% CPU	74% Memoria	14% Disco	0% Red	0% GPU	Motor de la GPU
> Google Chrome (14)	1.0%	347.8 MB	0.1 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 30-3: Utilización de la memoria en la tarea de Modificar Compra
Realizado por: Kleber Hipo, 2019

Eliminar un Pedido

Para la tarea de eliminar un pedido utiliza un promedio de 342.8 MB.

Nombre	9% CPU	76% Memoria	12% Disco	0% Red	1% GPU	Motor de la GPU
> Google Chrome (14)	0.2%	342.8 MB	0.1 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 31-3: Utilización de la memoria para la tarea de eliminar Pedido
Realizado por: Kleber Hipo, 2019

Registrar cliente

Para la tarea de registrar un Cliente utiliza un promedio de 330.2 MB.

Nombre	16% CPU	73% Memoria	14% Disco	0% Red	0% GPU	Motor de la GPU
> Google Chrome (14)	2.6%	330.2 MB	0.1 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 32-3: Utilización de la memoria en la tarea de Registrar Cliente

Realizado por: Kleber Hipo, 2019

Modificar Cliente

Para la tarea de Modificar un Cliente utiliza un promedio de 317.0 MB.

Nombre	15% CPU	73% Memoria	3% Disco	0% Red	1% GPU	Motor de la GPU
> Google Chrome (14)	2.2%	317.0 MB	0.1 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 33-3: Utilización de la memoria en la tarea de Modificar Cliente

Realizado por: Kleber Hipo, 2019

Generar Factura

Para la tarea de generar una factura utiliza un promedio de 354.6 MB.

Nombre	15% CPU	78% Memoria	18% Disco	0% Red	0% GPU	Motor de la GPU
> Google Chrome (16)	1.7%	354.6 MB	0.2 MB/s	0.1 Mbps	0%	GPU 0 - 3D

Figura 34-3: Utilización de la memoria en la generación de una factura

Realizado por: Kleber Hipo, 2019

Generar un Crédito

Para la tarea de generar un crédito de la factura a crédito utiliza un promedio de 358.9 MB.


Nombre	9% CPU	74% Memoria	22% Disco	0% Red	2% GPU	Motor de la GPU
>  Google Chrome (16)	0.9%	358.9 MB	0.1 MB/s	0 Mbps	0.2%	GPU 0 - 3D

Figura 35-3: Utilización de la memoria en generar un crédito de una factura
Realizado por: Kleber Hipo, 2019

Cobro de crédito

Para la tarea de cobro de las cuotas del crédito utiliza un promedio de 335.5 MB.


Nombre	19% CPU	79% Memoria	42% Disco	0% Red	4% GPU	Motor de la GPU
>  Google Chrome (16)	0.4%	335.5 MB	0.1 MB/s	0.1 Mbps	0%	GPU 0 - 3D

Figura 36-3: Utilización de la memoria en el cobro del crédito
Realizado por: Kleber Hipo, 2019

Lista de las Compras

Para el reporte de la compras utiliza un promedio de 342.2 MB.


Nombre	13% CPU	74% Memoria	10% Disco	0% Red	0% GPU	Motor de la GPU
>  Google Chrome (16)	4.8%	342.2 MB	0.2 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 37-3: Utilización de la memoria en el reporte de las compras
Realizado por: Kleber Hipo, 2019

Listado de los pedidos

Para el reporte del listado de los pedidos utiliza un promedio de 346.8 MB.


Nombre	18% CPU	74% Memoria	5% Disco	0% Red	2% GPU	Motor de la GPU
>  Google Chrome (16)	3.1%	346.8 MB	0.1 MB/s	0 Mbps	0.2%	GPU 0 - 3D

Figura 38-3: Utilización de la memoria en el reporte del listado de pedidos
Realizado por: Kleber Hipo, 2019

Listado de los productos con un stock Mínimo

Para el reporte de los productos con un stock mínimo utiliza un promedio de 337.7 MB.

Nombre	11% CPU	68% Memoria	2% Disco	0% Red	4% GPU	Motor de la GPU
> Google Chrome (17)	1.6%	337.7 MB	0.1 MB/s	0 Mbps	0.4%	GPU 0 - 3D

Figura 39-3: Utilización de la memoria en el reporte de los productos con stock mínimo
Realizado por: Kleber Hipo, 2019

A continuación en la siguiente **tabla 8-3** se detalla las diferentes tareas y su promedio de uso de memoria para completar dichas tareas.

Tabla 9-3: Resultados de las tareas de utilización de memoria

MÓDULO	TAREA	DATO 1 (MB)	DATO 2 (MB)	DATO 3 (MB)	PROMEDIO (MB)
Compra	Registrar Compra	488.3	558.1	547	531.13
	Modificar Compra	493.4	313.1	302.9	369.80
Pedido	Registrar Pedido	327.5	317.8	325.2	323.50
	Modificar un Pedido	347.8	338.4	339.6	341.93
	Eliminar un Pedido	242.8	341.2	320	301.33
	Registrar Cliente	330.2	307.5	306.9	314.87
	Modificar Cliente	317	320.6	306.9	314.83
	Generar Factura	254.6	377.2	396	342.60
	Generar venta a crédito	358.9	724.3	705.7	596.30
Cobro	Cobro de un crédito	335.5	474.6	472.2	427.43
Reportes	Listado de las compras entre fechas	342.2	381.8	383.4	369.13
	Listado de los Pedidos entre fechas	346.8	346.8	381.8	358.47
	Listado de los productos con un stock mínimo	337.7	391.7	391	373.47
Promedio					381.91

Realizado por: Kleber Hipo, 2019

3.2.3. Métrica uso de procesador

Para la evaluación de esta métrica del uso del procesador se tomó como variable X como el porcentaje que es la medida que se necesita para completar una tarea o actividad del sistema. Para

aquello se utilizó el administrador de tareas que viene incorporado por defecto en el sistema operativo Windows.

Tabla 10-3: Descripción para la evaluación de la métrica uso del procesador

Característica:	Eficiencia externa
Sub-característica:	Utilización de los recursos
Métrica:	Uso del procesador
Propósito de la métrica	Conocer cuánto de procesador utiliza en cada tarea.
Interpretación	Cuando menor es mejor
Tipo de escala	Radio
Tipo de medida	X=Porcentaje

Realizado por: Kleber Hipo, 2019

Fuente: (Vivanco Villamar, 2011)

Registrar Compra

El porcentaje del uso del procesador en promedio es de 1.8%

Nombre	8% CPU	80% Memoria	21% Disco	0% Red	1% GPU	Motor de la GPU
> Google Chrome (15)	1.8%	488.3 MB	0.1 MB/s	0 Mbps	0.6%	GPU 0 - 3D

Figura 40-3: Utilización del Procesador en la tarea de Registrar Compra

Realizado por: Kleber Hipo, 2019

Modificar Compra

El porcentaje del uso del procesador en promedio es de 5%

Nombre	17% CPU	83% Memoria	23% Disco	0% Red	2% GPU	Motor de la GPU
> Google Chrome (15)	5.0%	493.4 MB	0.1 MB/s	0 Mbps	0.2%	GPU 0 - 3D

Figura 41-3: Utilización del Procesador en la tarea de modificar Compra

Realizado por: Kleber Hipo, 2019

Registrar Pedido

El porcentaje del uso del procesador en promedio es de 0.5%

Nombre	9% CPU	77% Memoria	27% Disco	0% Red	0% GPU	Motor de la GPU
> Google Chrome (13)	0.5%	327.5 MB	0.1 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 42-3: Utilización del Procesador en la tarea de Registrar Pedido

Realizado por: Kleber Hipo, 2019

Modificar pedido

El porcentaje del uso del procesador en promedio es de 1%

Nombre	13% CPU	74% Memoria	14% Disco	0% Red	0% GPU	Motor de la GPU
> Google Chrome (14)	1.0%	347.8 MB	0.1 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 43-3: Utilización del Procesador en la tarea de Modificar Compra
Realizado por: Kleber Hipo, 2019

Eliminar un Pedido

El porcentaje del uso del procesador en promedio es de 0.2%

Nombre	9% CPU	76% Memoria	12% Disco	0% Red	1% GPU	Motor de la GPU
> Google Chrome (14)	0.2%	342.8 MB	0.1 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 44-3: Utilización del Procesador para la tarea de eliminar Pedido
Realizado por: Kleber Hipo, 2019

Registrar cliente

El porcentaje del uso del procesador en promedio es de 2.6%.

Nombre	16% CPU	73% Memoria	14% Disco	0% Red	0% GPU	Motor de la GPU
> Google Chrome (14)	2.6%	330.2 MB	0.1 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 45-3: Utilización del Procesador en la tarea de Registrar Cliente
Realizado por: Kleber Hipo, 2019

Modificar Cliente

El porcentaje del uso del procesador en promedio es de 2.2%

Nombre	15% CPU	73% Memoria	3% Disco	0% Red	1% GPU	Motor de la GPU
> Google Chrome (14)	2.2%	317.0 MB	0.1 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 46-3: Utilización del Procesador en la tarea de Modificar Cliente
Realizado por: Kleber Hipo, 2019

Generar Factura

El porcentaje del uso del procesador en promedio es de 1.7%

Nombre	15% CPU	78% Memoria	18% Disco	0% Red	0% GPU	Motor de la GPU
> Google Chrome (16)	1.7%	354.6 MB	0.2 MB/s	0.1 Mbps	0%	GPU 0 - 3D

Figura 47-3: Utilización del Procesador en la generación de una factura

Realizado por: Kleber Hipo, 2019

Generar un Crédito

El porcentaje del uso del procesador en promedio es de 0.9%.

Nombre	9% CPU	74% Memoria	22% Disco	0% Red	2% GPU	Motor de la GPU
> Google Chrome (16)	0.9%	358.9 MB	0.1 MB/s	0 Mbps	0.2%	GPU 0 - 3D

Figura 48-3: Utilización del Procesador en generar un crédito de una factura

Realizado por: Kleber Hipo, 2019

Cobro de crédito

El porcentaje del uso del procesador en promedio es de 0.4%

Nombre	19% CPU	79% Memoria	42% Disco	0% Red	4% GPU	Motor de la GPU
> Google Chrome (16)	0.4%	335.5 MB	0.1 MB/s	0.1 Mbps	0%	GPU 0 - 3D

Figura 49-3: Utilización del Procesador en el cobro del crédito

Realizado por: Kleber Hipo, 2019

Lista de las Compras

El porcentaje del uso del procesador en promedio es de 4.8%

Nombre	13% CPU	74% Memoria	10% Disco	0% Red	0% GPU	Motor de la GPU
> Google Chrome (16)	4.8%	342.2 MB	0.2 MB/s	0 Mbps	0%	GPU 0 - 3D

Figura 50-3: Utilización del Procesador en el reporte de las compras

Realizado por: Kleber Hipo, 2019

Listado de los pedidos

El porcentaje del uso del procesador en promedio es de 3.1%

Nombre	18% CPU	74% Memoria	5% Disco	0% Red	2% GPU	Motor de la GPU
> Google Chrome (16)	3.1%	346.8 MB	0.1 MB/s	0 Mbps	0.2%	GPU 0 - 3D

Figura 51-3: Utilización del Procesador en el reporte del listado de pedidos

Realizado por: Kleber Hipo, 2019

Listado de los productos con un stock Mínimo

El porcentaje del uso del procesador en promedio es de 1.6%

Nombre	11% CPU	68% Memoria	2% Disco	0% Red	4% GPU	Motor de la GPU
> Google Chrome (17)	1.6%	337.7 MB	0.1 MB/s	0 Mbps	0.4%	GPU 0 - 3D

Figura 52-3: Utilización del Procesador en el reporte de los productos con stock mínimo

Realizado por: Kleber Hipo, 2019

A continuación en la **tabla 10-3** el promedio de uso de los procesadores de las tareas seleccionadas.

Tabla 11-3: Resultados de la evaluación de la métrica uso del procesador

MÓDULO	TAREA	DATO 1 %	DATO 2 %	DATO 3 %	PROMEDIO
Compra	Registrar Compra	14.6 %	9.1 %	7.3 %	10.33 %
	Modificar Compra	2.5 %	0.9 %	0.8 %	1.40 %
Pedido	Registrar Pedido	0.4 %	6 %	3.1 %	3.17 %
	Modificar un Pedido	8.7 %	5.4 %	7.8 %	7.30 %
	Eliminar un Pedido	7.8 %	4.1 %	2.9 %	4.93 %
	Registrar Cliente	12.4 %	1.9 %	2.5 %	5.60 %
	Modificar Cliente	2.8 %	3.1 %	2 %	2.63 %
	Generar Factura	3 %	1.6 %	1.7 %	2.10 %
	Generar venta a crédito	2.5 %	4.10 %	3.10 %	3.23 %
Cobro	Cobro de un crédito	9.1 %	5.2 %	17.2 %	10.50 %
Reportes	Listado de las compras entre fechas	5.3 %	1.2 %	6.3 %	4.27 %
	Listado de los Pedidos entre fechas	3.1 %	3.5 %	4.2 %	3.60 %
	Listado de los productos con un stock mínimo	4.2 %	3.1 %	4.80 %	4.03 %

PROMEDIO	4.87 %
-----------------	---------------

Realizado por: Kleber Hipo, 2019

3.3. Resultados Obtenidos

Para evaluar la eficiencia del software desarrollado con la norma ISO 9126 se procedió a evaluar de acuerdo a la **tabla 12-3**. Asignando una valor cuantitativo y su valor Cualitativo

Tabla 12-3: Criterio de evaluación de los datos Parametrizados

Valor Cuantitativo	Valor Cualitativo
1	Regular
2	Malo
3	Aceptable
4	Bueno
5	Muy bueno
6	Excelente

Realizado por: Kleber Hipo. 2011

3.3.1. Tiempos de respuesta

Con la finalidad de medir el tiempo de respuesta que tanto son aceptables, se procede a evaluar los datos con la tabla de (VALENCIA ESPINOZA, 2018), cuando el tiempo de respuesta es menor esta tendrá una calificación alta, a continuación la tabla 13-3 de indicador de evaluaciones.

Tabla 13-3: Indicador de evaluación de los tiempos de respuesta

Valor Cuantitativo	Rango
6	[0 – 4.2] minutos
5	[4.3 – 7.5] minutos
4	[7.6 – 10.9] minutos
3	[11.0 – 14.2] minutos
2	[14.3 – 17.5] minutos
1	[18.0 – ∞] minutos

Fuente: (VALENCIA ESPINOZA, 2018,p.71)

Realizado por: Hipo Kleber, 2019

En la **tabla 13-3** se observa de un rango de minutos y un valor cualitativo, donde al menor tiempo se le asignara una calificación alta de 6 y al mínimo de 1. En la **tabla 14.3** en cambio se observa los tiempos promedios de las 13 tareas seleccionadas

Tabla 14-3: Resultados de los tiempos de respuesta obtenidos con el sistema

Tarea	Con Sistema (Minutos)
Registrar Compra	1.74
Modificar Compra	1.02
Registrar Pedido	1.09
Modificar un Pedido	0.85
Eliminar un Pedido	0.21

Registrar Cliente	0.97
Modificar Cliente	0.72
Generar Factura	0.37
Generar venta a crédito	0.64
Cobro de un crédito	0.36
Listado de las compras entre fechas	0.21
Listado de los Pedidos entre fechas	0.27
Listado de los productos con un stock mínimo	0.23
Promedio	0.67

Realizado por: Kleber Hipo, 2019

Al analizar los datos obtenidos de los tiempos de respuesta de la **tabla 14-3** para las 13 tareas se puede visualizar que el tiempo promedio para las tareas es de 0.76 minutos lo cual nos indica en comparación con la **tabla 13-3** que se encuentra en el rango de 0 a 4.2 minutos, por lo tanto su valor cuantitativo es de 6 es decir que representa excelente de acuerdo a la **tabla 12-3**

3.3.2. Utilización de recursos

3.7.2.1 Densidad del mensaje de uso de entrada/salida

Para esta característica indican que mientras más errores sean controlados su puntuación será alta. Pará lo cual se analizó de acuerdo con la siguiente **tabla 15-3**, donde se están las categorías con su rango, valor cualitativo y valor cuántico sienta el valor cuantitativo 6 de excelencia.

Tabla 15-3: Indicador de evaluación de la métrica densidad de mensaje I/O

Valor Cuantitativo	Rango
1	[0 – 0.05]
2	[0.010 – 0.015]
3	[0.020 – 0.025]
4	[0.030 – 0.035]
5	[0.040 – 0.045]
6	[0.050 – ∞]

Fuente: (VALENCIA ESPINOZA, 2018,p.72)

Realizado Por: Hipo Kleber, 2019

Los resultados de que se obtuvieron para esta característica se detallan en la **tabla 16-3**

Tabla 16-3: Resultados obtenidos densidad de mensaje I/O

Tarea	Resultado
Registrar Compra	0.027
Modificar Compra	0.032
Registrar Pedido	0.048
Modificar un Pedido	0.065
Eliminar un Pedido	0.036
Registrar Cliente	0.030

Modificar Cliente	0.043
Generar Factura	0.009
Generar venta a crédito	0.026
Cobro de un crédito	0.027
Listado de las compras entre fechas	0.017
Listado de los Pedidos entre fechas	0.019
Listado de los productos con un stock mínimo	0.018
PROMEDIO	0.030

Realizado por: Kleber Hipo, 2019

Al analizar los datos obtenidos de la densidad de mensajes de entrada y salida I/O de la **tabla 16-3** de las 13 tareas se puede visualizar que el promedio para estas tarea es de 0.030 lo cual nos indica en comparación con la **tabla 15-3** se encuentra en el rango de 0.030 a 0.035, el cual nos indica que la eficiencia en esta sub-característica es de valor cuantitativo 4 significando el valor cualitativo bueno con respecto a la **tabla 12-3**.

3.7.2.2. Uso de memoria

Para esa sub-característica nos dice que mientras menos memoria utiliza la calificación que se le otorgue será más alta, siendo el valor 6 como la valoración máxima, en la **tabla 17-3** se muestra el indicador de evaluación para esta métrica.

Tabla 17-3: Indicador de parametrización para evaluación del uso de memoria

Valor Cuantitativo	Rango
6	[0 – 150] MB
5	[200 – 300] MB
4	[301 – 400] MB
3	[401 – 500] MB
2	[500 – 600] MB
1	[600 – ∞] MB

Fuente: (VALENCIA ESPINOZA, 2018,p. 73)

Realizado por: Kleber Hipo, 2019

A continuación en la **tabla 18-3** se muestra los valores de uso de memoria RAM para cada uno de las tareas seleccionadas anteriormente.

Tabla 18-3: Resultados obtenidos del uso de memoria RAM

Tarea	Memoria
Registrar Compra	531.13
Modificar Compra	369.80
Registrar Pedido	323.50
Modificar un Pedido	341.93
Eliminar un Pedido	301.33

Registrar Cliente	314.87
Modificar Cliente	314.83
Generar Factura	342.60
Generar venta a crédito	596.30
Cobro de un crédito	427.43
Listado de las compras entre fechas	369.13
Listado de los Pedidos entre fechas	358.47
Listado de los productos con un stock mínimo	373.47
Promedio	381.91

Realizado por: Kleber Hipo, 2019

Al analizar los datos obtenidos de la **tabla 18-3** en donde se presentan el uso de memoria RAM de las 13 tareas se puede visualizar que el promedio para estas tareas es de 381.91 MB lo cual nos indica en comparación con la **tabla 17-3** se encuentra en el rango de 300 MB a 400 MB con un valor cuantitativo 4 con un valor cualitativo de bueno de acuerdo con la **tabla 12-3**.

3.7.2.3. Uso del procesador

Esta característica nos indica que mientras menos procesador lo use para completar la tarea su puntuación que se le otorgue será alta de igual forma que los anteriores con una puntuación alta el valor 6 lo cual se presenta en la **tabla 19-3**.

Tabla 19-3: Indicador de la métrica evaluación del uso de Procesador

Valor Cuantitativo	Rango
6	0 – 1
5	1 – 2
4	2 – 3
3	3 – 4
2	4 – 5
1	5 – ∞

Realizado por: Kleber Hipo, 2019

Fuente: (VALENCIA ESPINOZA, 2018,p.74)

Con el propósito de presentar los resultados de las principales tareas para medir la utilización del uso del procesador de la aplicación desarrollada y así conocer de que tanto eficiente es en esta sub-característica los datos que se obtuvieron su muestra en la **tabla 20-3**

Tabla 20-3: Resultado Obtenidos del uso del procesador

Tarea	Procesador
Registrar Compra	10.33 %
Modificar Compra	1.40 %
Registrar Pedido	3.17 %
Modificar un Pedido	7.30 %
Eliminar un Pedido	4.93 %

Registrar Cliente	5.60 %
Modificar Cliente	2.63 %
Generar Factura	2.10 %
Generar venta a crédito	3.23 %
Cobro de un crédito	10.50 %
Listado de las compras entre fechas	4.27 %
Listado de los Pedidos entre fechas	3.60 %
Listado de los productos con un stock mínimo	4.03 %
Promedio	4.85 %

Realizado por: Kleber Hipo, 2019

Al analizar los datos obtenidos de la **tabla 20-3** en donde se presentan el uso del procesador de las 13 tareas se puede visualizar que el promedio para estas es de 4.85 % lo cual nos indica en comparación con la **tabla 19-3** se encuentra en el rango de 4 al 5 % con el valor cuantitativo de 2, por lo tanto esta nos indica que la eficiencia en esta sub-característica es de valor cualitativo malo de acuerdo con la **tabla 12-3**.

Para proceder a evaluar la eficiencia de cada una de las características se dio unas ponderaciones la cual se detalla a continuación:

Tabla 21-3: Ponderación de las métricas de evaluación del sistema

Características	Métrica	Ponderación
Tiempo de Respuesta	Tiempos de respuesta	50%
Utilización de recursos	Densidad de mensaje de uso I/O	10%
	Uso de la Memoria	30%
	Uso del Procesador	10%
Eficiencia Total		100%

Realizado por: Kleber Hipo, 2019

En la **tabla 21-3** se puede visualizar las ponderaciones que se le ha asignado, la norma ISO 9126 en la eficiencia tiene dos sub-características como son tiempos de respuesta y la utilización de recursos la cual es evaluado con el 50% a cada una, la segunda sub-característica posee otras métricas como uso de la memoria, procesador y densidad de mensaje de uso I/O a los cuales se le dio una ponderación de 30%,10% y 10% en el orden respectivo

A continuación en la **Tabla 22-3** se muestra las diferentes sub-características de la eficiencia como: tiempos de respuesta y la utilización de recursos del sistema desarrollado de los valores obtenidos.

Tabla 22-3: Resultados finales de las métricas de la eficiencia

Características	Métrica	Valor Cuantitativo	Ponderación Final
Tiempo de Respuesta	Tiempos de respuesta	6	50.00%

Utilización de recursos	Densidad de mensaje de uso I/O	4	6.67%
	Uso de la Memoria	4	20.00%
	Uso del Procesador	2	3.33%
Eficiencia Total			80.00%

Realizado por: Kleber Hipo, 2019

En el **tabla 22-3** se puede visualizar la eficiencia total del sistema que se obtuvo de acuerdo con la ISO/IEC 9621 en la cual se analizó las sub-características con sus respectivas métricas, como anteriormente se mencionó que el valor de 6 es la puntuación máxima entonces si tiene este valor en el valor cuantitativo por ende tiene valor máximo de la ponderación y si es otro valor se realiza la operación de conversión para obtener el valor de la ponderación final, dando como resultado de un 80% de eficiencia del sistema desarrollado y un 20% faltante del mismo, la razón por la cual no se logró obtener el 100% ya que en la sub-característica de Utilización de recursos en las métricas de uso de memoria y densidad de mensaje están en un valor cuantitativo de 4 correspondiente al valor cualitativo bueno de acuerdo con la **tabla 12-3**, y en el uso del procesador se encuentra en un valor cuantitativo de.

Para la realización de la **tabla 23-3** la cual nos sirvió para evaluar la eficiencia obtenida según (Vivanco Villamar, 2011,p.149) propone las escalas de medición en la cual valora del 1 al 10, en nuestro caso a los valores propuestos se tuvo que transformas dichos datos a porcentaje ya que nuestro valor lo tenemos en esa medida.

Tabla 23-3: Criterio para la evaluación final de la eficiencia

Escala de medición	Valor Cualitativo
0- 27.5%	Insatisfactorio
27.6% - 50%	Insatisfactorio
50.1% - 87.5%	Satisfactorio
87.6% - 100%	Muy Satisfactorio

Realizado por: Kleber Hipo, 2019

Fuente: (Vivanco Villamar, 2011, p.149)

En la **tabla 23-3** se puede visualizar que se tiene unos Escala de medición y el valor cualitativo en la cual nuestro dato de 80% se ubica en el rango de 50.1% - 87.6% lo cual nos indica que el sistema que se ha desarrollado para el almacén de muebles Cacha tiene un valor cualitativo de **Satisfactorio** por lo tanto es aceptado en La norma ISO 9126.

CONCLUSIONES

- En el desarrollo del FrontEnd se aplicó la arquitectura SPA (Aplicación de página Simple) la cual los CSS, JavaScript y HTML se cargan una sola vez, todo esto ayudando a los usuarios a tener una experiencia fluida como si fuese una aplicación de escritorio.
- El Micro-Framework Lumen posee la arquitectura de Modelo-Controlador por lo cual es necesario realizar la parte de la interfaz gráfica utilizando algún Framework de JavaScript como Vue.js, acompañado de las hojas de estilo CSS o de un Framework como Bootstrap.
- Lumen es un Micro-Framework para el desarrollo de API REST o Micro servicios las cuales son devueltas con un texto plano en formato JSON (BackEnd), estos datos deben ser consumidas por un FrontEnd utilizando una biblioteca de JavaScript Axios la cual es recomendada por los desarrolladores de Vue, ya que este se utilizó para el desarrollo de este proyecto.
- Para la resolución del sistema orientado a la web de almacén de Muebles Cacha se han identificado 45 Historias de Usuarios y 7 Tareas de Ingeniería, las cuales durante el desarrollo se derivaron en 161 Tareas de Ingeniería y 444 pruebas de aceptación.
- Se realizó un estudio comparativo de los tiempos de respuesta de la forma tradicional con la gestión automatizada con el sistema desarrollado, en la primera forma se obtuvo un total de 5.52 minutos en promedio equivalente al 100% y en la segunda forma con un tiempo de 0.67 minutos equivalente a un 12.14 %, esto nos indica que con el uso del sistema se ha reducido el tiempo de la empresa en un 87.86% en completar las mismas tareas.
- Para comprobar la eficiencia del software de acuerdo con la norma ISO/IEC 9126 se evaluó las sub-Características de Tiempos de respuesta contando con un 50% y la utilización de recursos con un 30%, la cual da como resultado de un 82% de eficiencia del sistema desarrollado para el almacén de muebles Cacha de la ciudad de Riobamba.

RECOMENDACIONES

- Se podría desarrollar un sistema con las mismas características utilizando el Framework Laravel en la parte del BackEnd y realizar la evaluación de las métricas de la eficiencia como se desarrolló en el presente proyecto técnico. La comprobación de la eficiencia del nuevo sistema sería un nuevo objetivo de la eficiencia entre el Framework Laravel con el Micro-Framework Lumen.
- En caso que la aplicación crezca y en la cual se necesite agregar nuevas características que no los posee el micro-Framework, se sugiere migrar la aplicación al Framework completo Laravel.
- La aplicación desarrollada contempla el proceso de gestión de compras y ventas, y puede ser escalable, por lo que se recomienda ampliar a futuro e incluir el módulo de pagos utilizando algunas plataformas que existen en el mercado como PayPal.
- En el desarrollo del sistema en la parte de FrontEnd se utilizó el Framework Vue.js, el cual es fácil de utilizar e integrar al framework Bootstrap, sin embargo se recomienda utilizar otras librerías como React o con el Framework Angular.

BIBLIOGRAFÍA

ABUD FIGUEROA, M. Calidad en la Industria del Software. La Norma ISO-9126. [En Línea], 2012. [Consulta: 27 Feberro 2019.] Disponible en <http://www.nacionmulticultural.unam.mx/empresasindigenas/docs/2094.pdf>

AGUAYO, A. Lenguaje PHP, [En Línea], 2016. [Consulta: 09 Enero 2019] Disponible en: <https://es.slideshare.net/JessAlfredoGonzlez/lenguaje-php-66879919>.

AHMAD, H. W. Building RESTful web services with PHP 7 : Lumen, Composer, API testing, microservices, and more. [En Línea], 2017. [Consulta: 21 Enero 2019.]. Disponible en: https://books.google.com.ec/books?id=xJIGDwAAQBAJ&pg=PP1&dq=Building+RESTful+web+services+with+PHP+7:+Lumen,+Composer,+API+testing,+microservices,+and+more.&hl=es&sa=X&ved=0ahUKEwiRwaaErP_gAhWK11kKHZ01AK8Q6AEIKDAA#v=onepage&q=Building+RESTful+web+service

ALMAGRO, C. U. Lenguajes de Programación [En Línea], 2012. [Consulta: 09 Enero 2019.]. Disponible en: <http://java.sun.com/docs/books/jls/>.

ALVARADO, I. Tipos de API según su función y tecnología. [En Línea], 2018. [Consulta: 13 Enero 2019]. Disponible en: <https://ceroideas.es/tipos-de-API-segun-su-funcion-y-tecnologia/>.

ARIAS, M. A. Aprende programación con PHP y MySQL. [En Línea], 2017. [Consulta: 09 Enero 2019.].ISBN 978-1544106007 Disponible en: https://books.google.es/books?hl=es&lr=lang_es&id=mP00DgAAQBAJ&oi=fnd&pg=PA13&dq=php&ots=DLOljB5JpY&sig=F9wEaDJ7PdwVxadEqBZT2ykomFo#v=onepage&q=php&f=false

ARRIAGA, A. Creación de un RESTful API con Laravel Lumen [En Línea], 2017. [Consulta: 12 Febrero 2019.]Disponible en: <https://www.youtube.com/watch?v=VmtOP5sEH68>.

AZAUSTRE, C. Qué es la autenticación basada en Token. [En Línea], 2015. [Consulta: 20 Enero 2019]. Disponible en: <https://carlosazaustre.es/que-es-la-autenticacion-con-token/>

BAQUERO GARCIA, J. Modelo de autenticación para API REST basado en token [Blog], 2018. [Consulta: 20 Enero 2019]. Disponible en: <https://www.arsys.es/blog/programacion/autenticacion-API-REST-token/> .

BLANCARTE, O. SOAP vs REST ¿cual es mejor?, [Blog], [Consulta: 01 Marzo 2019]. Disponible en: <https://www.oscarblancarteblog.com/2017/03/06/soap-vs-REST-2/>.

BREHM, S. The future of web apps is — ready? — isomorphic JavaScript. [En Línea], 2013. [Consulta: 01 Marzo 2019]. Disponible en: <https://venturebeat.com/2013/11/08/the-future-of-web-apps-is-ready-isomorphic-javascript/>

CANO, C. La Arquitectura REST. [Blog], 2017. [Consulta: 14 Enero 2019]. Disponible en: <http://www.tsgroup.com.co/wps/portal/tsg/blog/detalle-blog/la-arquitectura-REST>.

CARDENAS PALADINES, F. M. Desarrollo de prototipo de aplicación web para agendamiento de citas del consultorio odontológico Orthodont utilizando el framework Laravel de PHP.[En Línea] (Tesis)(PreGrado),2015,[Consulta: 13 Marzo 2019]. Disponible en: <http://dspace.ucacue.edu.ec/bitstream/reducacue/6472/1/TESIS%20MAURICIO.pdf>.

CHIEMELA, C. Steps to enable CORS on a Lumen API Backend [En Línea] ,2017,[Consulta: 13 Marzo 2019]. Disponible en: <https://www.codementor.io/chiemelachinedum/steps-to-enable-cors-on-a-lumen-API-backend-e5a0s1ecx>.

FERNANDES FERNANDES, A. P. Desarrollo de una Arquitectura REST para la interacción Social con objetos físicos [En Línea], (Tesis)(PreGrado),2015, [Consulta: 15 Enero 2019] Disponible en: https://e-archivo.uc3m.es/bitstream/handle/10016/23024/TFG_Pablo-Alberto_Fernandes_Fernandes.pdf).

HALDAR, M. RESTful API Designing guidelines—The best practices, [En Línea] , 2017,[Consulta: 14 Enero 2019] Disponible en: <https://hackernoon.com/RESTful-API-designing-guidelines-the-best-practices-60e1d954e7c9>.

HOUSTON, P. Enable Eloquent ORM in Laravel Lumen micro-framework, [En Línea], 2015, [Consulta: 13 Marzo 2019]. Disponible en: <https://medium.com/@petehouston/enable-eloquent-orm-in-laravel-lumen-micro-framework-7a4f2fbcaf5d>.

DE LA FUENTE, J. Integración de Single-Page Application en Liferay. [En Línea], 2016, [Consulta: 13 Marzo 2019]. Disponible en: <https://upcommons.upc.edu/bitstream/handle/2117/89976/118457.pdf>.

LARA, W. ¿Cómo funciona la metodología Scrum? [En Línea] ,2016,[Consulta: 01 Marzo 2019] Disponible en: <https://platzi.com/blog/metodologia-scrum-fases/>.

LÓPEZ ECHEVERRY, ANA MARIA CABRERA, C. AND VALENCIA AYALA, L. E. ‘Introducción a la calidad de software’, Scientia Et Technica, 2(39), pp. 326–331. ISSN: 0122-1701 [En Línea], 2008. Disponible en: <http://www.redalyc.org/articulo.oa?id=84920503058>.

LÓPEZ YEPES, J. F., Norma ISO 9126 - Eficiencia . [En Línea], 2013, [Consulta: 01 Febrero 2019] Disponible en: <https://prezi.com/xpg0h7trtnjg/norma-iso-9126-eficiencia/>.

LUMEN PHP Micro-Framework By Laravel, [En Línea] [Consulta: 21 Marzo 2019]. Disponible en: <https://lumen.laravel.com/>.

MALDONADO GUERRERO, J. R. DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA WEB DE SEGUIMIENTO Y EVALUACIÓN DE LAS PRÁCTICAS PRE-PROFESIONALES PARA LA FACULTAD DE INGENIERÍA ESCUELA CIVIL DE LA PUCE. [En Línea],2016, [Consulta: 08 Enero 2019]: http://repositorio.puce.edu.ec/bitstream/handle/22000/12562/Tesis_Teor%C3%ADa.pdf?sequence=1&isAllowed=y.

MIKOWSKI, M. S., POWELL, J. C. AND BENSON, G. D. Single Page Web Applications. [En Línea],2014, [Consulta: 01 Marzo 2019].Disponible en: <http://1.droppdf.com/files/A32RR/manning-single-page-web-applications-javascript-end-to-end-2014.pdf>.

NAWAZ, S. How to Install the Best PHP Micro Frameworks on a Cloud Server, [Blog] , 2016, [Consulta: 10 Enero 2019].Disponible en: <https://www.cloudways.com/blog/install-php-micro-frameworks-on-cloud-server/>.

NEPO VILLANO, J. C. Muestreo No Probabilístico, [En Línea] ,2017,[Consulta: 04 Junio 2019].Disponible en: <https://es.slideshare.net/jhonchristiannepovillano/muestreo-no-probabilstico-74683679>.

ORDOÑEZ, J.¿Qué es una API REST? [blog],2018, [Consulta: 13 Enero 2019].Disponible en: <https://www.idento.es/blog/desarrollo-web/que-es-una-API-REST/>.

ORTIZ, A. E. Para que usar un framework en PHP, pros y contras, ventajas y desventajas, porqué emplearlos.[blog],2018, [Consulta: 09 Enero 2019]. Disponible en: <http://blog.hostdime.com.co/para-que-usar-framework-php-pros-contras-ventajas-desventajas-porque-emplearlos/>.

PALACIOS, D. Lumen, la versión micro framework de Laravel [En Línea],2015, [Consulta: 20 Julio 2018].Disponible en: <https://styde.net/lumen-la-version-micro-framework-de-laravel-php/>.

PALUPI, H. PHP Microframework vs. Fullstack Framework. [En Línea] ,2019,[Consulta: 21 Enero 2019].Disponible en: <https://www.codepolitan.com/php-microframework-vs-full-stack-framework-58ac664739bdf>.

PATIÑO ROSADO, S. G. AND ERIKA PAOLA, R. G. ‘EVALUATION OF THE EFFICIENCY OF A BIOMETRIC CONTROL SYSTEM BASED ON ISO / IEC 9126-2 AND 9126-3’. doi: 10.17993/3ctic.2018.62.60-75.

PÉREZ GARCÍA, A. A. Desarrollo de herramientas web de gestión docente. [En Línea]

(Tesis)(Pregrado), UNIVERSIDAD POLITÉCNICA DE CARTAGENA, ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN, 2007, [Consulta: 09 Enero 2019]. Disponible en: <http://repositorio.upct.es/bitstream/handle/10317/179/pfc2475.pdf>.

PEREZ, J. L. Principales lenguajes de programación web, ventajas y desventajas, [En Línea]. 2018, [Consulta: 09 Enero 2019]. Disponible en: <https://www.registrodominiosinternet.es/2013/08/lenguajes-programacion-web-ventajas.html>.

PLAZA ESTÉVEZ, S., RAMÍREZ LAMELA, N. AND ACOSTA MORALES, C. API de servicios web orientados a accesibilidad. [En Línea] (Tesis)(Pregrado), Universidad Complutense de Madrid, Facultad de Informática, 2015, [Consulta: 13 Enero 2019]. Disponible en: http://eprints.ucm.es/38686/1/Memoria_API%20de%20servicios%20web%20de%20accesibilidad.pdf.

PRESSMAN, R. S. Ingeniería del software un enfoque práctico. [En Línea], SÉPTIMA EDICIÓN. Mexico, 2010 [Consulta: 27 Febrero 2019]. Disponible en: http://artemisa.unicauca.edu.co/~cardila/Libro_Pressman_7.pdf.

PRIETO, R., SUSANA, L. AND GARCÍA, A. ‘Mejores Prácticas Para El Establecimiento Y Aseguramiento De La Calidad Del Software’, Unidad Multidisciplinaria : CIET. 2008

R2B SOLUTIONS PHP Micro Framework - PHP Web Application Development, [En Línea], 2016, [Consulta: 21 Marzo 2019]. Disponible en: <https://www.k2bindia.com/know-why-the-php-micro-frameworks-are-rising-now/>.

ROSA MONCAYO, J. M. ¿Qué es Vue.js? [En Línea] , 2017, [Consulta: 01 Abril 2019]. Disponible en: <https://openwebinars.net/blog/que-es-vuejs/>.

ROSADA, S. *Desarrollo Web: Tabla comparativa de los lenguajes de programación*, [En Línea] , 2015, [Consulta: 08 Enero 2019]. Disponible en: <http://desarrollowebbydesarrolloweb.blogspot.com/2015/02/tabla-comparativa-de-los-lenguajes-de.html>.

ROUSE, M. ¿Qué es MySQL?. [En Línea], 2015, [Consulta: 21 Enero 2019]. Disponible en: <https://searchdatacenter.techtarget.com/es/definicion/MySQL>.

RUIZ, G. A., PEÑA, A. AND ARTURO, CASTRO, C. ‘Modelo de Evaluación de Calidad de Software Basado en Lógica Difusa, Aplicada a Métricas de Usabilidad de Acuerdo con la Norma ISO/IEC 9126’, 2006.

SANTOS HERNÁNDEZ, W. D. AND SERRANO PARREÑO, J. D. DESARROLLO DE UNA API REST CON SUS APLICACIONES WEB Y MÓVIL PARA LA VENTA DE ROPA ONLINE DE LA EMPRESA ROOSMAN, [En Línea] (Tesis) UNIVERSIDAD CENTRAL DEL

ECUADOR, FACULTAD DE INGENIERÍA, CIENCIAS FÍSICAS Y MATEMÁTICA, Ecuador, Quito, 2017. [Consulta: 14 Julio 2018]. Disponible en: <http://www.dspace.uce.edu.ec/bitstream/25000/9668/1/T-UCE-0011-312.pdf>.

SCALONE, F. ESTUDIO COMPARATIVO DE LOS MODELOS Y ESTANDARES DE CALIDAD DEL SOFTWARE. [En Línea], (Tesis) (Maestría) UNIVERSIDAD TECNOLÓGICA NACIONAL, FACULTAD REGIONAL BUENOS AIRES, Buenos Aires, Argentina, 2006. [Consulta: 27 Febrero 2019]. Disponible en: <http://laboratorios.fi.uba.ar/lsi/scalone-tesis-maestria-ingenieria-en-calidad.pdf>.

SCHWABER KEN, S. J. La Guía de Scrum TM. [En Línea], 2017, [Consulta: 01 Abril 2019]. Disponible en: <http://creativecommons.org/licenses/by-sa/4.0/legalcodeandalsodescribedinsummaryformat> <http://creativecommons.org/licenses/by-sa/4.0/>. By utilizing.

SIERRA, F. et al. Estudio y análisis de los framework en php basados en el modelo vista controlador para el desarrollo de software orientado a la web., *Revista Investigación y Desarrollo en TIC*, [En Línea], 2013, vol. 4, no. 2. [Consulta: 09 Enero 2019]. ISSN 2216-1570. Disponible en: <http://publicaciones.unisimonbolivar.edu.co/rdigital/ojs/index.php/identific/articulo/view/1517/1446>.

TARVAINEN, J. *Symfony Benchmarks: Symfony Microkernel, Lumen, Silex, Slim...*, [En Línea] 2016-01-06. [Consulta: 20 Julio 2018]. Disponible en: <https://symfony.fi/entry/symfony-benchmarks-microkernel-silex-lumen-and-slim>.

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA Muestreo no probabilístico, [En Línea], 2017. [Consulta: 04 Junio 2019]: Disponible en: <https://www.studocu.com/es/document/universidad-autonoma-de-baja-california/metodologia-de-la-investigacion/ejercicios-obligatorios/muestreo-no-probabilistico/3042921/view>.

VALENCIA ESPINOZA, E. C. DESARROLLO DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE LA INFORMACIÓN DE UN CENTRO ODONTOLÓGICO USANDO EL FRAMEWORK ZEND (Tesis) (PreGrado), Escuela Superior Politécnica de Chimboraz, Facultad de Informática y Electrónica, Escuela de Ingeniería en Sistemas, Riobamba, Ecuador, 2018 [Consulta: 04 Abril 2019]: Disponible en: <http://dspace.esPOCH.edu.ec/bitstream/123456789/9130/1/18T00754.pdf>

VIVANCO VILLAMAR, A. A. EVALUACIÓN DE CALIDAD DEL SISTEMA INTEGRADO PARA CASAS DE VALORES SICAV DE LA BOLSA DE VALORES DE QUITO UTILIZANDO LA NORMA ISO/IEC 14598, [En Línea] (Tesis) (PreGrado), Escuela Politécnica

Nacional, Escuela de Ingeniería. , Quito, Ecuador, 2011. [Consulta: 20 Febrero 2019] Disponible en: <http://bibdigital.epn.edu.ec/bitstream/15000/4329/1/CD-3948.pdf>.

VUEJS Introducción — Vue.js.[En Línea] [Consulta: 01 Abril 2019] Disponible en: <https://es-vuejs.github.io/vuejs.org/v2/guide/>.

YEEPLY 6 Tipos de desarrollo de aplicaciones web que puedes crear. [En Línea] , 2018, [Consulta: 08 Enero 2019] Disponible en: <https://www.yeeply.com/blog/6-tipos-desarrollo-de-aplicaciones-web/>.