



INSTITUTO TECNOLÓGICO
"CORDILLERA"

CARRERA DE ANÁLISIS DE SISTEMAS

**SISTEMATIZACIÓN DEL PROCESO DE INVENTARIO MEDIANTE UN
APLICATIVO WEB PARA LA EMPRESA MUEBLES ELIZABETH
UBICADA EN LA CIUDAD DE QUITO**

**Proyecto de investigación previo a la obtención del título de Tecnólogo en
Análisis de Sistemas**

Autor: Jefferson Alexander Manzaba Castro

Tutor: Ing. Johnny Patricio Coronel Ordoñez

Quito, Octubre 2017



DECLARATORIA DE APROBACION DEL TUTOR Y LECTOR



DECLARATORIA

Yo Jefferson Alexander Manzaba Castro, declaro que la investigación es totalmente original y personal, he citado las fuentes correspondientes por lo cual en su ejecución se ha respetado las disposiciones legales que protegen los derechos de autor vigentes. Todas las ideas, resultados y conclusiones en las cuales he llegado son de mi absoluta responsabilidad.

Jefferson Alexander Manzaba Castro

C.I. 171966474-8

LICENCIA DE USO NO COMERCIAL

Yo, Jefferson Alexander Manzaba Castro portador de la cédula de ciudadanía signada con el **No 171966474-8** de conformidad con lo establecido en el Artículo 110 del Código de Economía Social de los Conocimientos, la Creatividad y la Innovación (INGENIOS) que dice: “En el caso de las obras creadas en centros educativos, universidades, escuelas politécnicas, institutos superiores técnicos, tecnológicos, pedagógicos, de artes y los conservatorios superiores, e institutos públicos de investigación como resultado de su actividad académica o de investigación tales como trabajos de titulación, proyectos de investigación o innovación, artículos académicos, u otros análogos, sin perjuicio de que pueda existir relación de dependencia, la titularidad de los derechos patrimoniales corresponderá a los autores. Sin embargo, el establecimiento tendrá una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra con fines académicos. Sin perjuicio de los derechos reconocidos en el párrafo precedente, el establecimiento podrá realizar un uso comercial de la obra previa autorización a los titulares y notificación a los autores en caso de que se traten de distintas personas. En cuyo caso corresponderá a los autores un porcentaje no inferior al cuarenta por ciento de los beneficios económicos resultantes de esta explotación. El mismo beneficio se aplicará a los autores que hayan transferido sus derechos a instituciones de educación superior o centros educativos.”, otorgo licencia gratuita, intransferible y no exclusiva para el uso no comercial del proyecto denominado **Sistematización del Proceso de Inventario mediante un Aplicativo Web para la Empresa Muebles Elizabeth ubicada en la Ciudad de Quito** con fines académicos al Instituto Tecnológico Superior Cordillera.

FIRMA

NOMBRE

Jefferson Alexander Manzaba Castro

CEDULA

171966474-8

Quito, Octubre 2017

AGRADECIMIENTO

En primer lugar agradezco a Dios por permitirme existir en este mundo tan bonito, tan maravilloso de tener una familia tan bella que me brinda su amor que están en las buenas y en las malas que son un pilar fundamental en mi vida les agradezco a mis hermanos y a mi hermana por darme su cariño por acoger los buenos consejos que les he dado por respetarme. A mis padres Ramón Manzaba y Jenny Castro por enseñarme muchas cosas buenas, por cuidarme, por darme su amor, la educación siempre han puesto mano duro y les agradezco por ser así conmigo gracias a ustedes soy un hombre honrado, trabajador, respetuoso y ahora culminando una etapa más de mi vida que es de ser un profesional y se vienen otras etapas más que las cumpliré con la ayuda de ustedes y la ayuda de Dios.

Agradezco a todos los Ingenieros y a las Ingenieras por inculcarme todos sus conocimientos en estos 3 años de preparación por ser ese pilar fundamental en mi carrera profesional los considero con todo cariño como mis amigos que me han sabido guiar que me han aconsejado que nunca me rinda, cumpla todos mis sueños y metas les agradezco de todo corazón bendiciones.



DEDICATORIA

Primeramente mis agradecimientos a Dios, por la vida, por las fuerzas que me ha
dado de nunca rendirme permitiéndome terminar una etapa tan grande en
mi vida profesional.

Dedico esto a mis padres que me aconsejaron siempre a terminar lo que empiezo en
mi vida de nunca rendirme y por el apoyo económico para poder terminar la
carrera.

Al Ing. Johnny Coronel que me supo dirigir como tutor que me inculco sus
conocimientos, me dio su apoyo y su voluntad para lograr realizar el proyecto y
culminarlo sin problemas.

A mi Familia, amigos, ingenieros, ingenieras gracias por todo su apoyo
incondicional que me ayudara a seguir adelante en mi vida profesional y personal.

Gracias



ÍNDICE GENERAL

Contenido	Página
DECLARATORIA DE APROBACION DEL TUTOR Y LECTOR.....	ii
DECLARATORIA.....	iii
LICENCIA DE USO NO COMERCIAL.....	iv
AGRADECIMIENTO	v
DEDICATORIA.....	vi
INTRODUCCION	xvii
Resumen Ejecutivo.....	xviii
Abstrac.....	xx
Capítulo I: Antecedentes	1
1.01 Contexto.....	1
1.02 Justificación	2
1.03 Definición del Problema Central.	3
Capítulo II: Análisis De Involucrados.....	5
2.01 Requerimientos	5
2.01.01 Descripción del sistema actual	5
2.01.02 Visión y alcance	5
2.01.03 Entrevistas	6
2.01.04 Matriz de requerimientos	7



	Página
2.01.05 Descripción detallada requerimiento.....	9
2.02 Mapeo De Involucrados.....	20
2.03 Matriz De Involucrados	21
Capítulo III: Problemas y Objetivos	22
3.01 Árbol de Problemas	22
3.02 Árbol de Objetivos.....	23
3.03.01 Casos de uso individual	25
3.04 Especificación de casos de uso	27
3.05 Casos de uso de realización	30
3.05.01 Diagrama de secuencia del sistema.....	37
Capítulo IV: Análisis de Alternativas	40
4.01 Matriz de Análisis de Alternativas.....	40
4.02 Matriz de Impacto de los Objetivos	41
4.03 Estándares para el Diseño de Clases.....	42
4.04 Diagrama de Clases.....	45
4.05 Modelo Lógico-Físico.....	45
4.06 Diagrama de Componentes	48
4.07 Diagrama de Estrategias	49
4.08 Matriz de Marco Lógico	50
4.09 Vistas Arquitectónicas	51
4.09.01 Vista Lógica	51
4.09.02 Vista Física.....	52
4.09.03 Vista de Desarrollo.....	52



	Página
4.09.04 Vista de Procesos	53
Capítulo V: Propuesta	54
5.01 Especificación de estándares de programación.....	54
5.01.01 Prefijos aplicados para controladores	55
5.02 Diseño de interfaces de usuario	56
5.03 Especificación de pruebas de unidad	66
5.04 Especificación de pruebas de aceptación.....	68
5.05 Especificación de pruebas de carga	69
5.06 Configuración del Ambiente mínimo/ideal.....	71
Capítulo VI: Aspectos Administrativos	73
6.01 Recursos.....	73
6.02 Presupuesto	75
6.03 Cronograma	76
Capítulo VII: Conclusiones y Recomendaciones.....	78
7.01 Conclusiones	78
7.02 Recomendaciones	78
Anexos	80
A.01 Manual de Instalación	81
A.02 Manual Usuario.....	94
A.03 Manual Técnico	10909

ÍNDICE DE TABLAS

Título	Página
Tabla 1 <i>Análisis de la Matriz de Fuerzas T</i>	3
Tabla 2 <i>Entrevista E001</i>	6
Tabla 3 <i>Matriz de requerimientos Funcionales</i>	7
Tabla 4 <i>Matriz de requerimientos No Funcionales</i>	8
Tabla 5 <i>Descripción del requerimiento RF001</i>	9
Tabla 6 <i>Descripción del requerimiento RF002</i>	10
Tabla 7 <i>Descripción del requerimiento RF003</i>	11
Tabla 8 <i>Descripción del requerimiento RF004</i>	12
Tabla 9 <i>Descripción del requerimiento RF005</i>	13
Tabla 10 <i>Descripción del requerimiento RF006</i>	14
Tabla 11 <i>Descripción del requerimiento RF007</i>	15
Tabla 12 <i>Descripción del requerimiento RF008</i>	16
Tabla 13 <i>Descripción del requerimiento RNF001</i>	17
Tabla 14 <i>Descripción del requerimiento RNF002</i>	18
Tabla 15 <i>Descripción del requerimiento RNF003</i>	19
Tabla 16 <i>Matriz de Análisis de Involucrados</i>	21

Página

Tabla 17	<i>Especificación de Casos de Uso UC002</i>	27
Tabla 18	<i>Especificación de Casos de Uso UC003</i>	28
Tabla 19	<i>Especificación de Casos de Uso UC004</i>	28
Tabla 20	<i>Especificación de Casos de Uso UC005</i>	29
Tabla 21	<i>Especificación de Casos de Uso UC006</i>	29
Tabla 22	<i>Caso de Uso de Realización UCR002</i>	30
Tabla 23	<i>Caso de Uso de Realización UCR003</i>	31
Tabla 24	<i>Caso de Uso de Realización UCR004</i>	33
Tabla 25	<i>Caso de Uso de Realización UCR005</i>	34
Tabla 26	<i>Caso de Uso de Realización UCR006</i>	36
Tabla 27	<i>Matriz de Análisis de alternativas</i>	40
Tabla 28	<i>Matriz de Análisis de impacto de los objetivos</i>	41
Tabla 29	<i>Parámetros para el diagrama de clases</i>	44
Tabla 30	<i>Matriz de Marco Lógico</i>	50
Tabla 31	<i>Descripción tipo de Variables</i>	55
Tabla 32	<i>Descripción tipo de control</i>	55

Página

Tabla 33 <i>Diseño de interfaces de usuario</i>	56
Tabla 34 <i>Especificación de Tipos de Componentes utilizados</i>	57
Tabla 35 <i>Especificación de Tipos de Componentes utilizados</i>	59
Tabla 36 <i>Especificación de Tipos de Componentes utilizados</i>	60
Tabla 37 <i>Especificación de Tipos de Componentes utilizados</i>	62
Tabla 38 <i>Especificación de Tipos de Componentes utilizados</i>	63
Tabla 39 <i>Especificación de Tipos de Componentes utilizados</i>	65
Tabla 40 <i>Especificación de Pruebas de Unidad EPDU001</i>	66
Tabla 41 <i>Especificación de Pruebas de Unidad EPDU002</i>	66
Tabla 42 <i>Especificación de Pruebas de Unidad EPDU003</i>	67
Tabla 43 <i>Especificación de Pruebas de Unidad EPDU004</i>	67
Tabla 44 <i>Especificación de Pruebas de Aceptación EPDA001</i>	68
Tabla 45 <i>Especificación de Pruebas de Aceptación EPDA002</i>	68
Tabla 46 <i>Especificación de Pruebas de Carga EPDC001</i>	69
Tabla 47 <i>Especificación de Pruebas de Carga EPDC002</i>	70
Tabla 48 <i>Especificación de Pruebas de Carga EPDC003</i>	70



Página

Tabla 49 <i>Especificación de requerimientos de Hardware</i>	71
Tabla 50 <i>Especificación de requerimientos de Software</i>	72
Tabla 51 <i>Recursos Humanos utilizados en la construcción del proyecto</i>	73
Tabla 52 <i>Recursos Económicos utilizados en la construcción del proyecto</i>	73
Tabla 53 <i>Recurso de Hardware usado en el desarrollo</i>	74
Tabla 54 <i>Recurso de Software usado en el desarrollo</i>	74
Tabla 55 <i>Presupuesto</i>	75

ÍNDICE DE FIGURAS

Título	Página
<i>Figura 1.</i> Mapeo de involucrados	20
<i>Figura 2.</i> Árbol de Problemas	22
<i>Figura 3.</i> Árbol de Objetivos	23
<i>Figura 4.</i> Diagrama de caso de uso General UC001.....	24
<i>Figura 5.</i> Diagrama de caso de uso Administrador UC002.....	25
<i>Figura 6.</i> Diagrama de caso de uso Personal de bodega UC003.....	25
<i>Figura 7.</i> Diagrama de caso de uso Cliente UC004.....	26
<i>Figura 8.</i> Diagrama de caso de uso Contador UC005.....	26
<i>Figura 9.</i> Diagrama de caso de uso Jefe de Bodega UC006.....	27
<i>Figura 10.</i> Diagrama de caso de realización Administrador UCR002.....	30
<i>Figura 11.</i> Diagrama de caso de realización Personal de bodega UCR003.....	31
<i>Figura 12.</i> Diagrama de realización Cliente UCR004.....	32
<i>Figura 13.</i> Diagrama de realización Contador UCR005.....	34
<i>Figura 14.</i> Diagrama de realización Jefe de bodega UCR006.....	35
<i>Figura 15.</i> Diagrama de Secuencia Administrador.....	37
<i>Figura 16.</i> Diagrama de Secuencia de Personal.....	38

Página

<i>Figura 17.</i> Diagrama de Secuencia Cliente.....	38
<i>Figura 18.</i> Diagrama de Secuencia Contador.....	39
<i>Figura 19.</i> Diagrama de Secuencia Jefe de bodega.....	39
<i>Figura 20.</i> Diagrama de Clases	45
<i>Figura 21.</i> Modelo Lógico.....	46
<i>Figura 22.</i> Modelo Físico.....	47
<i>Figura 23.</i> Diagrama de Componentes	48
<i>Figura 24.</i> Diagrama de Estrategias	49
<i>Figura 25.</i> Vista Lógica	51
<i>Figura 26.</i> Vista Física	52
<i>Figura 27.</i> Vista de Desarrollo	52
<i>Figura 28.</i> Vista de Procesos.....	53
<i>Figura 29.</i> Interfaz inicialización del aplicativo.....	56
<i>Figura 30.</i> Interfaz de Login.....	57
<i>Figura 31.</i> Interfaz del Usuario.....	58
<i>Figura 32.</i> Interfaz de Ingreso de Proveedores.....	60



Página

<i>Figura 33.</i> Interfaz de Ingreso de Clientes.....	61
<i>Figura 34.</i> Interfaz de Categoría de productos.....	62
<i>Figura 35.</i> Interfaz de Ingreso de productos.....	64
<i>Figura 36.</i> Interfaz de Ingreso de venta.....	65
<i>Figura 37.</i> Cronograma mediante procesos por fechas.....	76
<i>Figura 38.</i> Cronograma Esquema grafico.....	77

INTRODUCCIÓN

El objetivo del siguiente trabajo es encontrar la solución a los problemas que existe en el área de bodega en la Empresa Muebles Elizabeth, mediante un sistema de inventario y ventas. Se va a detallar análisis, diseño y la implementación del aplicativo web para mejorar el ingreso de información al inventario que el personal realice el proceso de venta sea más rápido y que el cliente este satisfecho al momento que haga su compra y obtenga su producto de manera rápida y eficiente.

Con este proyecto se mostrara como un aplicativo web mejorará todos los procesos de ingreso de información de productos, categorías, proveedores, clientes, ventas etc.

Toda la información ingresada será alojada en una base de datos para que genere todos los reportes necesarios que necesite la Empresa y la información siempre se encuentre actualizada y no se pierda.

Resumen Ejecutivo

Este proyecto de tesis tiene el objetivo de implementar la sistematización del proceso de inventario mediante un aplicativo web para la Empresa Muebles Elizabeth en la provincia de Pichincha canto Quito. Empresa dedicada a la elaboración de mueblería como muebles de sala, comedores, dormitorios etc. El objetivo primordial de la Empresa es elaborar los mejores muebles de alta calidad para sus clientes y que estén satisfechos del producto que adquieren.

El problema principal que se tiene, es que no existe un sistema que realice el proceso de sistematización de inventario esto ocasiona que exista perdida de información de la materia prima que ingresa y sale de bodega también al momento de realizar la venta se demora en el proceso, por lo cual el cliente que va a adquirir el producto se sienta descontento por la demora en la obtención de su producto.

Siendo así mediante los conocimientos adquiridos en todos los semestres de la carrera de Sistemas se procederá a realizar el aplicativo web para la Empresa Muebles Elizabeth mediante reuniones con el administrador y personal de la empresa. Por lo cual se ha podido ver el problema que tenían y se ha llegado a una solución la cual es la sistematización del proceso de inventario mediante el aplicativo web que hará el ingreso, salida y venta de materia prima de bodega para que la información sea más clara, no allá perdida de información al momento de realizar la venta del producto al

cliente y este se encuentre satisfecho de la buena atención y de lo rápido que se genera la venta para obtener su producto.

El proyecto que se ha desarrollado contara con siete capítulos donde se analizó la problemática de la empresa que está dedica al diseño y elaboración de muebles realizando una mejora y avance a nivel de la empresa dejando atrás los problemas que tenían en el proceso de inventario y así cumplir las metas y objetivos que tiene la empresa mediante un buen procesamiento de información, confiabilidad para el usuario con ayuda del aplicativo web y así terminar con el problema que la empresa tenía.

Se ha realizado manuales del desarrollo de todo el aplicativo web, en el cual se describe el correcto uso y manejo de las herramientas utilizadas para la buena utilización del sistema. Se presenta un buen análisis al presupuesto económico y a los recursos humanos utilizados, también se presenta la realización del cronograma de actividades donde se detalla fechas y tareas para el desarrollo del sistema y al final de todo obtenemos las conclusiones y recomendaciones del aplicativo web.

Abstract

This thesis project has the objective of implementing the systematization of the inventory process through a web application for Elizabeth Furniture Company in the province of Pichincha canton Quito. Company dedicated to the manufacture of furniture as living room furniture, dining rooms, bedrooms etc. The Company's primary objective is to produce the best high quality furniture for its customers and to be satisfied with the product they purchase.

The main problem is that there is no system that performs the process of systematization of inventory that causes loss of information on the raw material that enters and leaves the warehouse also at the time of making the sale is delayed in the process , So that the customer who will purchase the product will feel dissatisfied by the delay in obtaining their product.

Being thus through the knowledge acquired in all the semesters of the Systems career will proceed to realize the web application for the Elizabeth Furniture Company through meetings with the administrator and personnel of the company. So we have been able to see the problem they had and we have reached a solution which is the systematization of the inventory process through the web application that will make the entrance, exit and sale of raw material from the warehouse to make the information more Clear, there is lost of information at the time of the sale of the product to the customer and he is satisfied with the good care and how fast the sale is generated to get his product.



The project that has been developed will have seven chapters that analyzed the problem of the company that is dedicated to the design and elaboration of furniture making an improvement and progress at the level of the company leaving behind the problems they had in the inventory process and thus Meet the goals and objectives of the company through good information processing, reliability for the user with the help of the web application and thus end the problem that the company had.

Manuals have been developed for the development of the entire web application, which describes the correct use and management of the tools used for the good use of the system. A good analysis is presented to the economic budget and the human resources used. It is also presented the realization of the schedule of activities which details dates and tasks for the development of the system and at the end of everything we obtain the conclusions and recommendations of the web application.

Capítulo I: Antecedentes

1.01 Contexto

La Empresa Muebles Elizabeth dedicada a diseño y elaboración de muebles es la mejor en el sector artesanal de muebles, realizan la elaboración de mueblería por categorías (como muebles de sala hechos por los mejores tapiceros, dormitorios y comedores hecho por los mejores carpinteros y talladores) para el cliente que desea tener un mueble de categoría y excelencia con garantía. Por lo cual los tapiceros realizan los mejores muebles con textiles finos para las personas que se permiten pagar un precio más elevado de lo habitual. Al igual que los carpinteros elaboran los dormitorios y comedores más finos con la mejor madera y tallado a mano para un mejor acabado para las personas que les gusta la elegancia y que pueden pagar un valor más alto.

Existen muchas empresas que elaboran muebles de sala, otras que sólo arman dormitorios y otras que se dedican sólo a elaborar comedores y lo cual no entregan sus pedidos a tiempo, al cliente le quedan mal es ahí en donde la Empresa Muebles Elizabeth les brinda una mejor atención a sus clientes ya que ellos fabrican todo, como son muebles de sala, comedores, dormitorios, retapizados y relacados. La empresa siempre entrega su mercadería al cliente a tiempo sin atrasos a la fecha que se pide los muebles de sala, comedor o dormitorio. La empresa lo realiza sin pérdida de tiempo para que el cliente este satisfecho y feliz al momento de adquirir su mercadería.

1.02 Justificación

La Empresa Muebles Elizabeth está entrando con fuerza al mercado artesanal en la elaboración de Muebles por lo cual han venido trabajando arduamente para alcanzar sus objetivos y metas que son en fabricar los mejores muebles de sala, comedores y dormitorios para todo el público que le gusta tener unos muebles más finos y elegantes y que se permiten pagar un precio más elevado.

La mayoría de Empresas en el ámbito artesanal en la fabricación de muebles venden sus muebles a un precio muy elevado y sus muebles tienen ese acabado que desea el cliente es ahí que la Empresa Muebles Elizabeth da un mejor precio a sus clientes no tan elevados y sus muebles son de un gran acabado, elegancia y confort para que sus clientes estén satisfechos de los muebles de sala, comedor o dormitorio que van a adquirir.

Por lo cual la Empresa necesita tener todo su inventario de bodega perfectamente en orden de todo lo que adquieren como textiles, madera, esponja y otros varios materiales para realización de tan sofisticados y elegantes muebles; pero hay un inconveniente que la empresa viene realizando su inventario de bodega de la forma tradicional en hojas de papel por lo cual se demoran más en registrar la mercadería y no tienen un buen control del inventario por lo cual se va a optimizar el sistema de inventario ingresos y egresos de bodega mediante un aplicativo web.

1.03 Definición del Problema Central.

El proceso de inventario de ingresos y egresos de bodega en la empresa lo están realizando de la manera tradicional en hojas lo cual les hace demorar al momento de ingresar la información de la materia prima al registro y no les permite tener un buen control de la materia prima que existe y de la que se acaba.

Por este motivo se va a realizar la siguiente sistematización del inventario de ingreso, salida y venta de materia prima de bodega en la Empresa Muebles Elizabeth.

Tabla 1

Matriz de Fuerzas T

SITUACIÓN EMPEORADA	SITUACIÓN ACTUAL				SITUACIÓN MEJORADA
Demasiada pérdida de materia prima.	Por la falta de coordinación en la entrega de la materia prima a los tapiceros y carpinteros				Optimizar agilizar y mejorar el proceso de ingreso y salida de la materia prima a bodega.
FUERZAS IMPULSADORAS	I	PC	I	P C	FUERZAS BLOQUEADORAS
Personal responsable del control de bodega y mercadería	3	2	4	3	El personal no está capacitado en el manejo del área de fabricación.
Registro adecuado de información del personal	3	3	5	3	No existe una utilización de herramientas informáticas.
Registro de ingreso de entrada y salida de materia prima a bodega.	3	2	4	3	Están desorganizados al momento de la búsqueda de información de las entradas y salidas del producto.
Registro adecuado de información de los clientes	3	3	5	3	No existe un registro de los clientes.
Registro de ingreso de proveedores	3	2	4	3	No existe un registro de proveedores.

Registro de ventas realizadas en el día	4	3	4	3	No existe un registro de ventas.
--	---	---	---	---	----------------------------------

Nota: pc: potencial de cambio

I: Intensidad

Escala: 1= Bajo 2=Medio Bajo 3=Malo 4=Medio Alto 5=Alto

Capítulo II: Análisis De Involucrados

2.01 Requerimientos

2.01.01 Descripción del sistema actual

El proceso de control de ingreso, salida y venta de materia prima de bodega en la empresa Muebles Elizabeth se lo realiza de forma tradicional escrito a mano, donde el empleado registra el ingreso y salida de materia prima a bodega y también las ventas que se realizó en el día a diferentes clientes en hojas de papel las cuales lleva el control el Administrador de la Empresa.

2.01.02 Visión y alcance

La visión de la Empresa Muebles Elizabeth es en ser la más grande empresa en la fabricación de muebles a nivel nacional dando a sus clientes muebles de calidad que cumplan todas las expectativas que el cliente desea.

Por lo cual la empresa necesita agilizar sus procesos de inventario y venta para dar un mejor servicio a los clientes sin tardanza se va a realizar la sistematización del inventario de bodega mediante un sistema web que hará el ingreso y salida de materia prima nueva y existente que está en bodega el sistema también realizara ventas pero no va a realizar facturación electrónica.

2.01.03 Entrevistas

Tabla 2

Entrevista

identificador: E001		
Preguntas	Objetivos	Análisis Posterior
¿Qué proceso interviene en el control de la materia prima en bodega?	No existe coordinación en el ingreso de materia prima	Se realiza el ingreso de la información de la materia prima en hojas físicas. -Procedimiento de Ingresos de Datos de la materia prima -Proceso de mantenimiento -Proceso de ordenes -Proceso de productos -Proceso de personal
¿Qué persona está encargada del inventario de bodega?	Hay que analizar el personal que ingresa a la plataforma.	Ver que Personal manejara la plataforma los cuales son el Administrador, Bodeguero, Contador.
¿Cómo se registra el encargado de bodega?	Registro del personal por medio de hojas.	No hay un buen control del registro del personal se encuentra desordenado.
¿Cuáles son los problemas que presenta la empresa en el inventario de bodega?	Se encuentra la información de materia prima de bodega en desorden.	Por la mala infraestructura que existe en el almacenamiento de información de la empresa.

2.01.04 Matriz de requerimientos

Tabla 3

Matriz de Requerimientos Funcionales

Identificador	Descripción	Fuente	Prioridad	Tipo	Estado	Usuarios Involucrados
Requerimientos Funcionales						
RF001	El Sistema permitirá el ingreso de información del personal.	Bodeguero Admin Contador	Alta	Sistema	En revisión	Administrador Contador
RF002	El sistema permitirá registrar los datos del personal.	Admin y Personal	Alta	Sistema	En revisión	Administrador Contador
RF003	El sistema permitirá el registro de clientes.	Admin. Bodeguero	Alta	Sistema	En revisión	Administrador Bodeguero
RF004	La plataforma permitirá el ingreso de proveedores.	Admin Bodeguero	Alta	Sistema	En revisión	Administrador Bodeguero.
RF005	El sistema permitirá ingresar la información de la materia prima nueva y existente.	Admin Bodeguero	Alta	Sistema	En revisión	Administrador Bodeguero
RF006	El sistema permitirá eliminar la materia prima que se agotó.	Admin Bodeguero	Alta	Sistema	En revisión	Administrador Bodeguero
RF007	El sistema permitirá hacer ventas sin facturación electrónica.	Administrador Bodeguero	Alta	Sistema	En revisión	Administrador Bodeguero.
RF008	El sistema dará reportes de las ventas.	Admin Bodeguero Contador	Alta	Sistema	En revisión	Admin, Contador, Bodeguero

Tabla 4

Matriz de Requerimientos No Funcionales

Identificador	Descripción	Fuente	Prioridad	Tipo	Estado	Usuarios Involucrados
Requerimientos No Funcionales						
RNF001	El sistema debe mantener accesibilidad del 99% del ingreso que realice el personal.	Admin Bodeguero Contador	Medio Baja	Usuario	En revisión	Administrador Contador Bodeguero
RNF002	EL sistema se realizara de acuerdo al presupuesto que la empresa ha aprobado.	Admin Contador	Alta	Usuario	En revisión	Administrador Contador
RNF003	El sistema contara con manuales de usuarios para el manejo adecuado.	Admin Contador Bodeguero	Baja	Usuario	En revisión	Administrador Contador Bodeguero

2.01.05 Descripción detallada requerimiento

Tabla 5

Descripción del Requerimiento RF001 registro de datos del Personal

El sistema permitirá ingresar la información del usuario y validar los privilegios.		Estado	En ejecución
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RF001		
Tipo de requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Principales: usuario, contraseña		
Descripción	Registro de Personal nuevo y mantenimiento de la información.		
Datos de Salida	Reporte de datos del Usuario .		
Resultados Esperados	Información del Usuario.		
Origen	Entrevista		
Dirigido a	Administrador Contador		
Prioridad	5		
Requerimientos Asociados	RF002, RF003, RF004, RF006, RF007, RF008		
ESPECIFICACIÓN			
Precondiciones	Deberá tener completa la información donde el sistema pueda devolver información olvidada.		
Poscondiciones	Sin registro no se tendrá acceso al sistema.		
Criterios de Aceptación	Deberá registrar información adicional de referencia.		

Tabla 6

Descripción del Requerimiento RF002

El sistema permitirá el registro Personal		Estado	En ejecución
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RF002		
Tipo de requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	datos: nombres, apellidos, nombre de usuario, email, contraseña.		
Descripción	Permitirá registrar los datos del trabajador.		
Datos de Salida	Informe de trabajador.		
Resultados Esperados	Ingreso del Personal individual.		
Origen	Entrevista		
Dirigido a	Administrador Contador		
Prioridad	5		
Requerimientos Asociados	RF003, RF004, RF008		
ESPECIFICACIÓN			
Precondiciones	Los datos del Personal encargados de la empresa.		
Criterios de Aceptación	Información actualizada del Personal podrá ser visualizada.		

Tabla 7

Descripción del Requerimiento RF003

El sistema permitirá registrar la información obtenida por los clientes		Estado	En ejecución
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RF003		
Tipo de requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Registros como : nombres, apellidos, telefono, dirección,email.		
Descripción	Registro de clientes mediante la plataforma administrando así su información.		
Datos de Salida	Realización de reportes y documentos con datos del cliente.		
Resultados Esperados	Interactuar con el producto a través de la web.		
Origen	Entrevista		
Dirigido a	Administrador Bodeguero		
Prioridad	5		
Requerimientos Asociados	RF006, RF007, RF008		
ESPECIFICACIÓN			
Precondiciones	Organizar la información de los clientes.		
Pos condiciones	El administrador deberá validar todos sus pedidos.		
Criterios de Aceptación	Cumplir con los requisito para poder generar los documentos.		

Tabla 8

Descripción del Requerimiento RF004

El sistema registrara la información de los proveedores.		Estado	En ejecución
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RF004		
Tipo de requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Datos: nombres, apellidos, dirección, email, teléfono.		
Descripción	Registrará los datos de los proveedores.		
Datos de Salida	Reporte de proveedores.		
Resultados Esperados	Registro de los proveedores.		
Origen	Entrevista		
Dirigido a	Administrador Bodeguero		
Prioridad	5		
Requerimientos Asociados	RF001, RF002, RF003		
ESPECIFICACIÓN			
Precondiciones	El Administrador podrá ver el registro de los proveedores.		
Poscondiciones	Llenar los campos exigidos por el sistema.		

Tabla 9

Descripción del Requerimiento RF005 Ingreso de la Materia Prima

El sistema permitirá el registro por el personal de bodega de la materia prima nueva y existente.		Estado	En ejecución
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RF005		
Tipo de requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Datos: imagen, código del producto, nombre, categoría, descripción, precio de entrada, precio de salida, unidad, presentación, mínima en inventario, inventario inicial.		
Descripción	Podrá ingresar toda la información de la materia prima.		
Datos de Salida	Información detallada de la materia prima.		
Resultados Esperados	Despliegue de datos solicitados .		
Origen	Entrevista		
Dirigido a	Administrador Bodeguero		
Prioridad	5		
Requerimientos Asociados	RF006, RF007,RF008		
ESPECIFICACIÓN			
Precondiciones	El personal de bodega podrá ingresar los productos pero solo el administrador podrá editar.		
Poscondiciones	Ingresar información exacta.		

Tabla 10

Descripción del Requerimiento RF006

El sistema podrá eliminar la materia prima que se ha agotado.		Estado	En ejecución
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RF006		
Tipo de requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	El registro de la materia prima existente que se ha agotado será eliminado.		
Descripción	Permitirá al personal y administrador ver que un producto se ha agotado.		
Datos de Salida	Se eliminara la materia prima que ya no existe.		
Resultados Esperados	Despliegue de datos solicitados.		
Origen	Entrevista		
Dirigido a	Administrador Bodeguero		
Prioridad	5		
Requerimientos Asociados	RF007, RF008		
ESPECIFICACIÓN			
Precondiciones	Personal autorizado tendrá el privilegio de ingreso de productos.		
Poscondiciones	Tener valores exactos del stock existente.		

Tabla 11

Descripción del Requerimiento RF007 Ventas sin Facturación Electrónica.

El sistema permitirá hacer ventas sin facturación electrónica.		Estado	En ejecución
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RF007		
Tipo de requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Datos: código, cantidad, unidad, producto, precio unitario, precio total.		
Descripción	El sistema ingresa información del producto para la venta.		
Datos de Salida	Se procesara la venta.		
Resultados Esperados	Venta exitosa detallada.		
Origen	Entrevista		
Dirigido a	Administrador Bodeguero		
Prioridad	5		
Requerimientos Asociados	RF006, RF008		
ESPECIFICACIÓN			
Precondiciones	El personal debe llenar bien los datos al momento de realizar la venta.		
Poscondiciones	Llenar todos los campos exigidos por el sistema.		

Tabla 12

Descripción del Requerimiento RF008.

El sistema dará el reporte de las ventas hechas		Estado	En ejecución
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RF008		
Tipo de requerimiento	Crítico	Tipo de Requerimiento	Funcional
Datos de Entrada	Información del cliente, atendido por, código, cantidad, nombre del producto, precio unitario descuento, sub total, total.		
Descripción	El sistema pide que toda la información se llene no puede quedar ningún campo vacío.		
Datos de Salida	Información detallada de la venta hecha.		
Resultados Esperados	Verificación por parte del Administrador.		
Origen	Entrevista		
Dirigido a	Administrador Contador Bodeguero		
Prioridad	5		
Requerimientos Asociados	RF006, RF007		
ESPECIFICACIÓN			
Precondiciones	Los descuentos se los realiza solo a clientes frecuentes.		
Poscondiciones	Llenar todos los campos exigidos.		

Tabla 13

Descripción del Requerimiento RNF001

El sistema deber tener accesibilidad del 99 % las veces que el personal, contador o administrador ingrese.		Estado	Aprobado
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RNF001		
Tipo de requerimiento	No crítico	Tipo de Requerimiento	No funcional
Datos de Entrada	Ninguno		
Descripción	Ninguno		
Datos de Salida	El sistema estará disponible las 24 horas del día los 7 días de la semana.		
Resultados Esperados	El aplicativo web deberá tener la disponibilidad del 99% en línea.		
Origen	Entrevista		
Dirigido a	Administrador Contador Bodeguero		
Prioridad	2		
Requerimientos Asociados	Ninguno		
ESPECIFICACIÓN			
Precondiciones	Ninguno		
Poscondiciones	Ninguno		
Criterios de Aceptación	Funcionando sin inconvenientes.		

Tabla 14

Descripción del Requerimiento RNF002 Costo del Software

El sistema deberá ser implementado al presupuesto asignado por la empresa.		Estado	Aprobado
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RNF002		
Tipo de requerimiento	No crítico	Tipo de Requerimiento	No funcional
Datos de Entrada	Ninguno		
Descripción	Ninguno		
Datos de Salida	Herramientas necesitadas para el desarrollo.		
Resultados Esperados	El sistema podrá visualizarse sin la intervención de grandes cantidades de dinero.		
Origen	Entrevista		
Dirigido a	Administrador Contador		
Prioridad	2		
Requerimientos Asociados	Ninguno		
ESPECIFICACIÓN			
Precondiciones	Presupuesto asignado por la empresa para el desarrollo del sistema web.		
Poscondiciones	Ninguno		
Criterios de Aceptación	El sistema estará diseñado y desarrollado con la herramientas necesarias para su implementación.		

Tabla 15

Descripción del Requerimiento RNF003 Registro del respectivo manual de usuario para una adecuada manipulación del software

El sistema contará con manuales de usuarios para un buen manejo.		Estado	Aprobado
Creado por	Jefferson Manzaba	Actualizado por	Jefferson Manzaba
Fecha de Creación	20/04/2017	Fecha de Actualización	21/04/2017
Identificador	RNF003		
Tipo de requerimiento	No crítico	Tipo de Requerimiento	No funcional
Datos de Entrada	Ninguno		
Descripción	Ninguno		
Datos de Salida	Correcta Manipulación del software de todo su funcionamiento.		
Resultados Esperados	Todo el mantenimiento del software podrá ser manejado con la visualización y el correcto estudio de los manuales.		
Origen	Entrevista		
Dirigido a	Administrador Contador Bodeguero		
Prioridad	3		
Requerimientos Asociados	Ninguno		
ESPECIFICACIÓN			
Precondiciones	Tener conocimientos técnicos para la actualización del software.		
Poscondiciones	Ninguno		
Criterios de Aceptación	Permitirá una adecuada manipulación del sistema.		

2.02 Mapeo De Involucrados

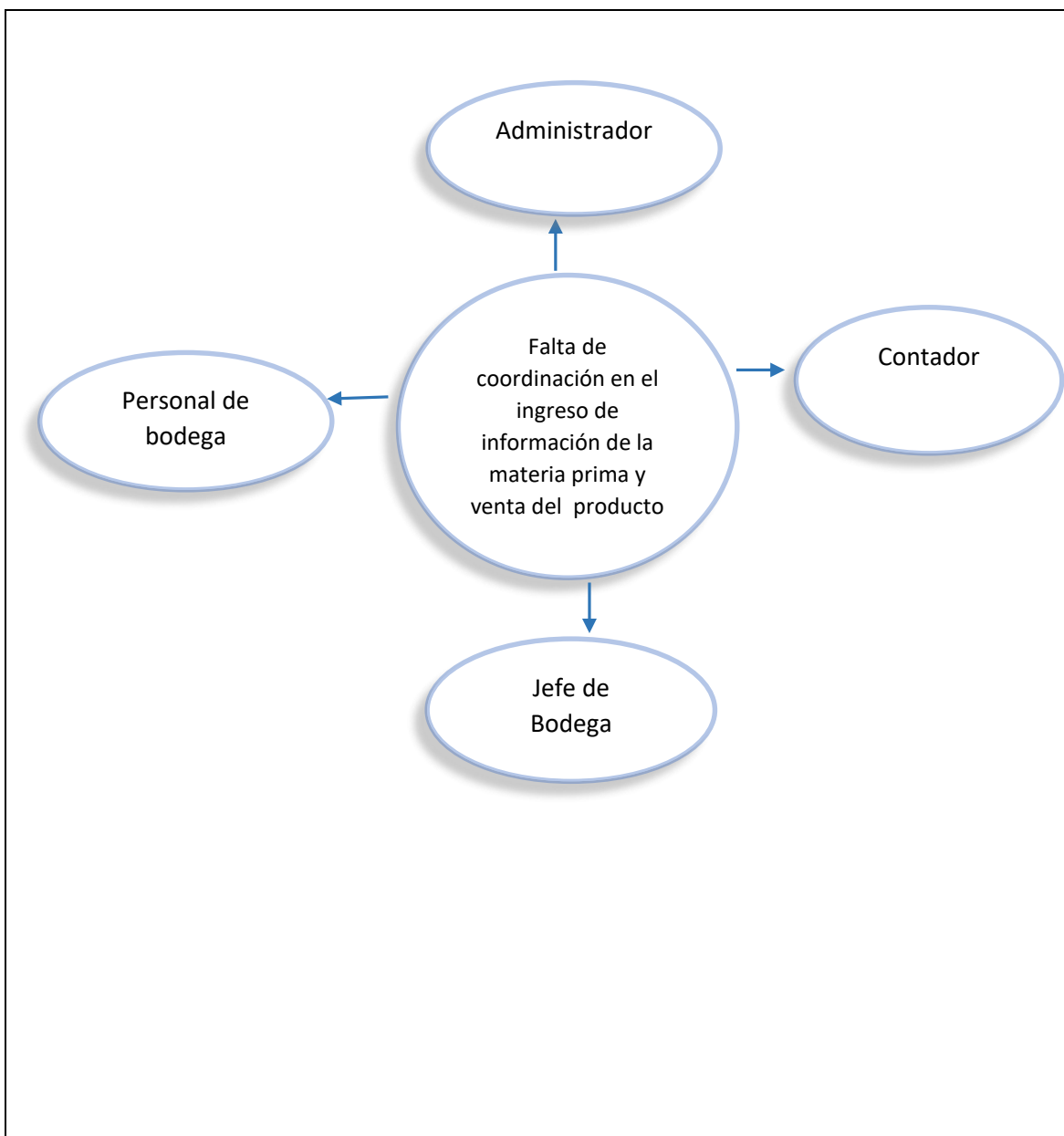


Figura 1. Mapeo de Involucrados: El siguiente diagrama podemos visualizar todo el personal que realiza el proceso de ingreso y venta de la materia prima.

2.03 Matriz De Involucrados

Tabla 16

Matriz

Actores Involucrados	Interés sobre el problema Central	Problemas Percibidos	Recursos, Mandos y Capacidades	Interés sobre el Proyecto	Conflictos Potenciales
Administrador	Organización y ventas en la empresa manejo financiero	Coordinación inadecuada en áreas de la empresa	Financiero, Producción, Ventas.	Coordinación entre personal ,proveedores y clientes en las ventas	Falta de coordinación en bodega y ventas de la materia prima.
Contador	Distribución y Organización del personal en la empresa	Mala administración en el momento de realizar ventas	Recursos Financieros, Tecnológicos Caja	Organización de las ventas vía web.	Falta de coordinación con las áreas de la empresa.
Jefe de bodega	Buena distribución en el área de bodega.	Demora en la entregas.	Producción y bodega.	Control de entradas y salidas de los productos.	Mala distribución de la materia prima.
Personal de bodega	Mejorar la producción de ventas generadas mediante el aplicativo web	Los procesos de ventas generadas físicamente no son fiables.	Producción y bodega	Control de Generación de venta	Falta de capacitación de los procesos

Capítulo III: Problemas y Objetivos

3.01 Árbol de Problemas

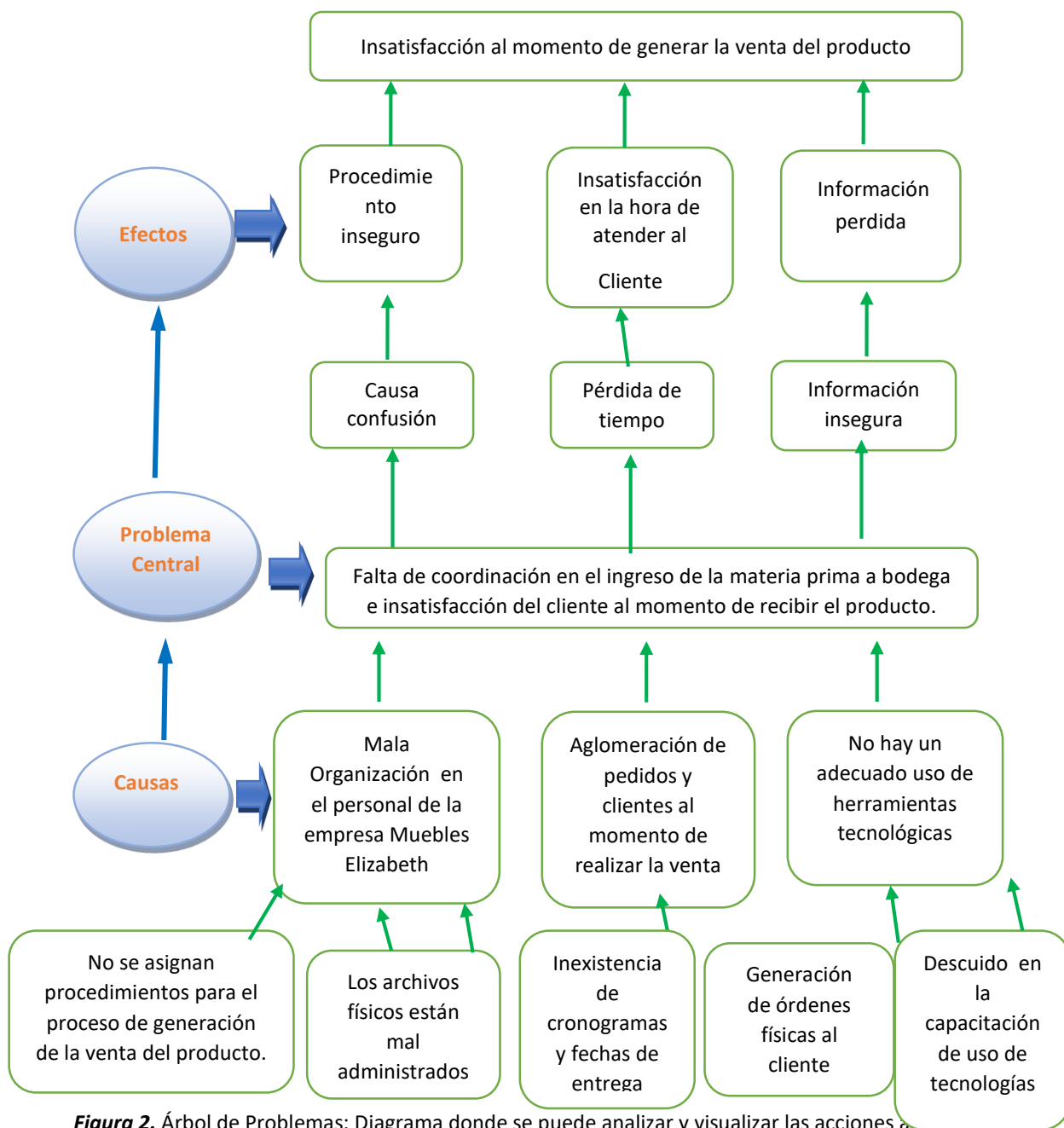


Figura 2. Árbol de Problemas: Diagrama donde se puede analizar y visualizar las acciones a

problemática central al proceso de generación de ingreso y venta de la materia prima mediante un aplicativo web.

3.02 Árbol de Objetivos

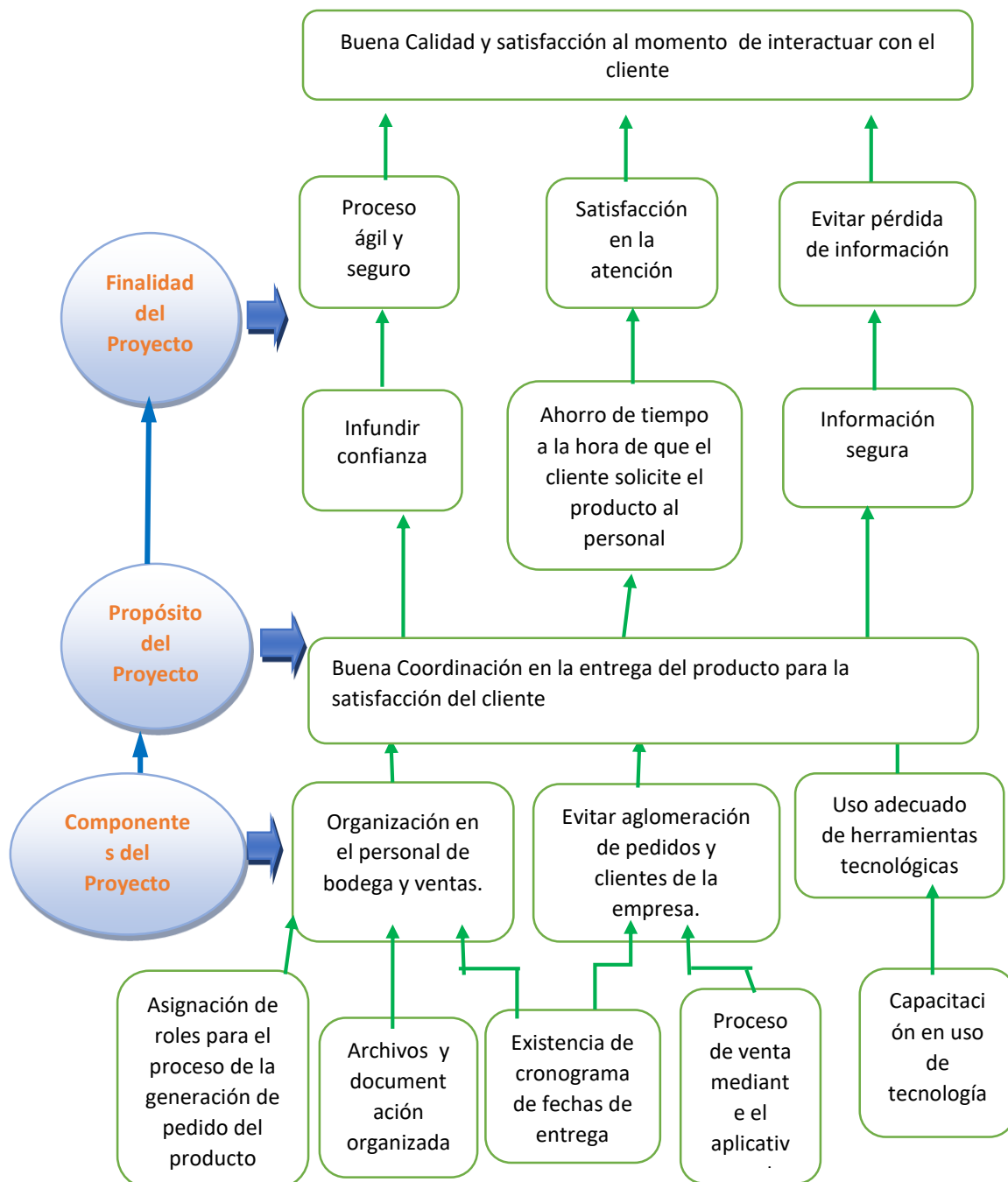


Figura 3. Árbol de Objetivos: Proceso que permitirá analizar el propósito y finalidad verificando así el árbol de problemas realizando una verificación y solución al problema central.

Diagrama de casos de uso general

UC001

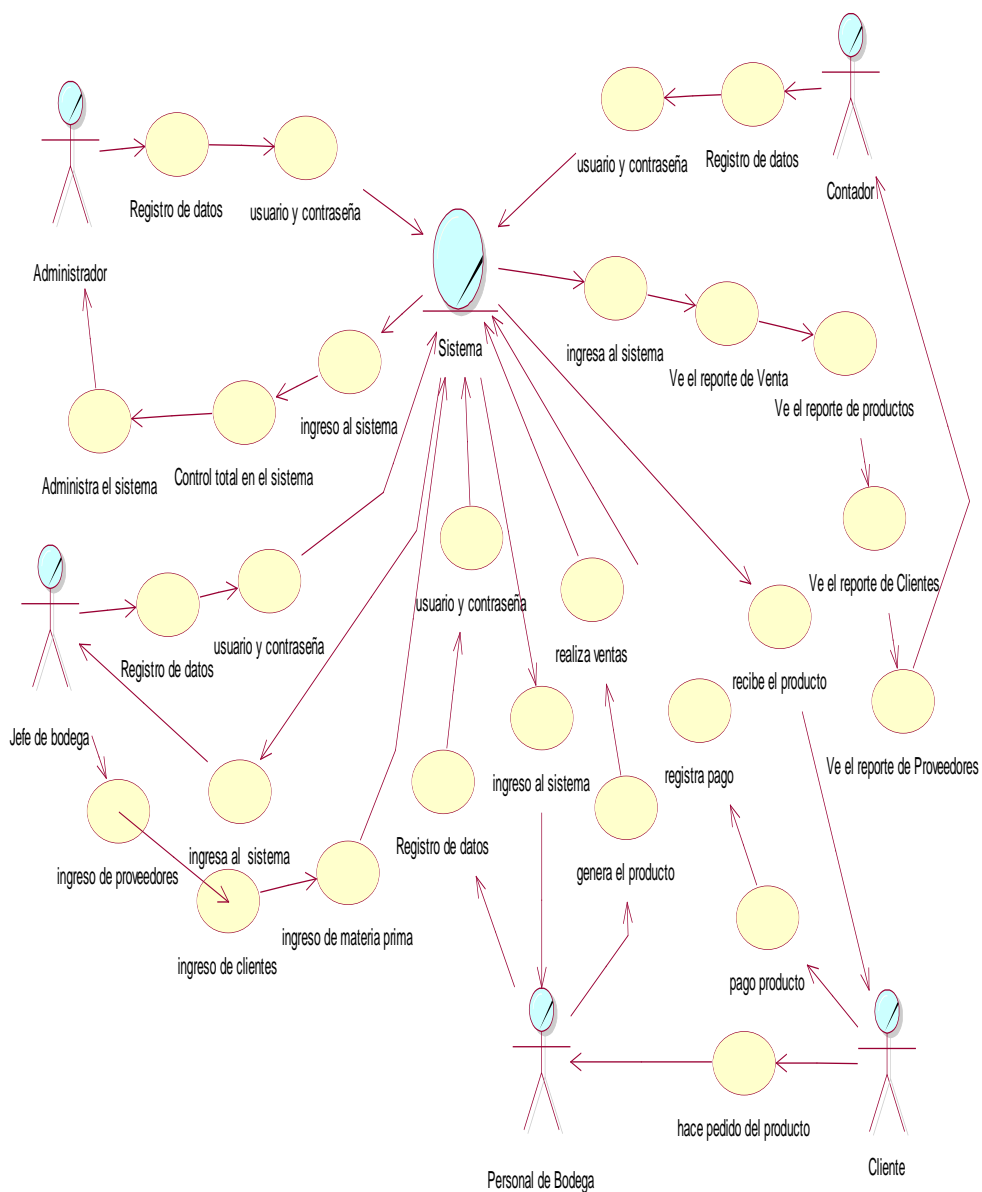


Figura 4. Diagrama de caso de uso General de la reglas del negocio UC001.

3.03.01 Casos de uso individual

UC002

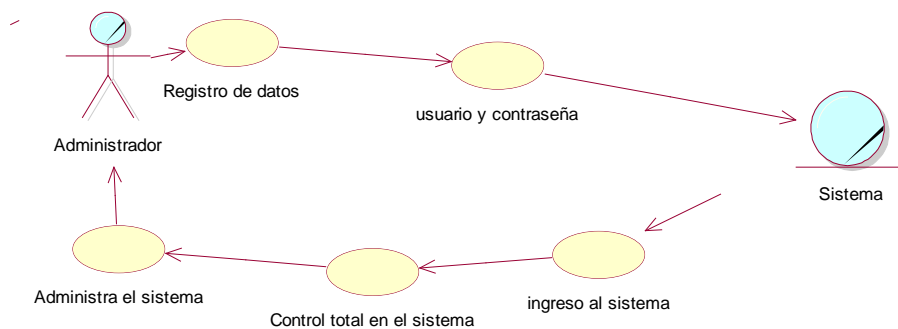


Figura 5. Diagrama de caso de uso Administrador UC002.

UC003

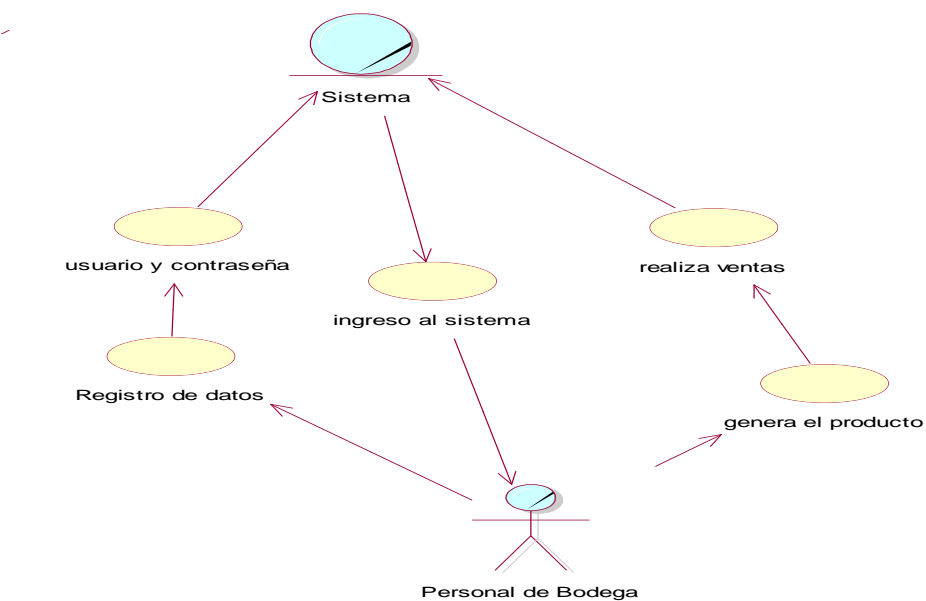


Figura 6. Diagrama de caso de uso Personal de Bodega UC003.

UC004

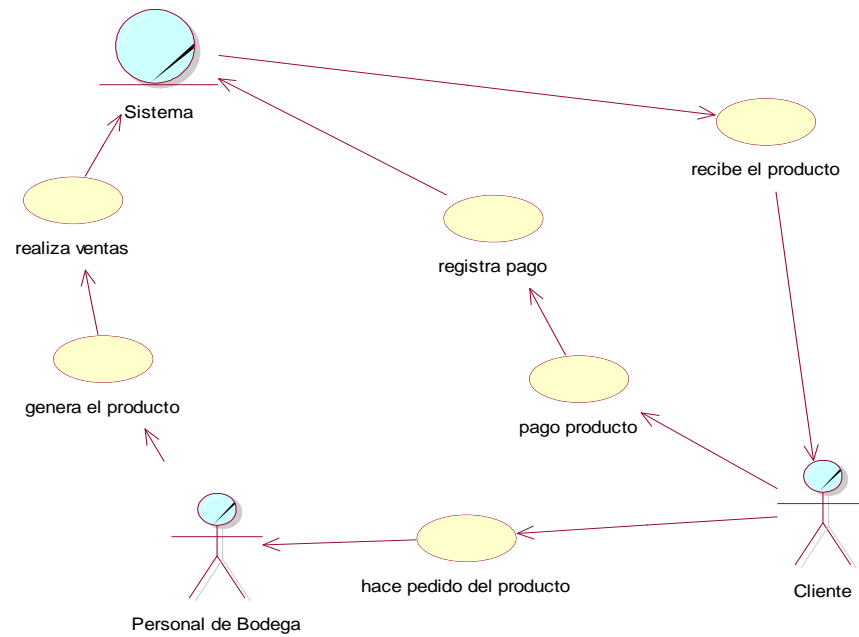


Figura 7. Diagrama de caso de uso Cliente UC004

UC005

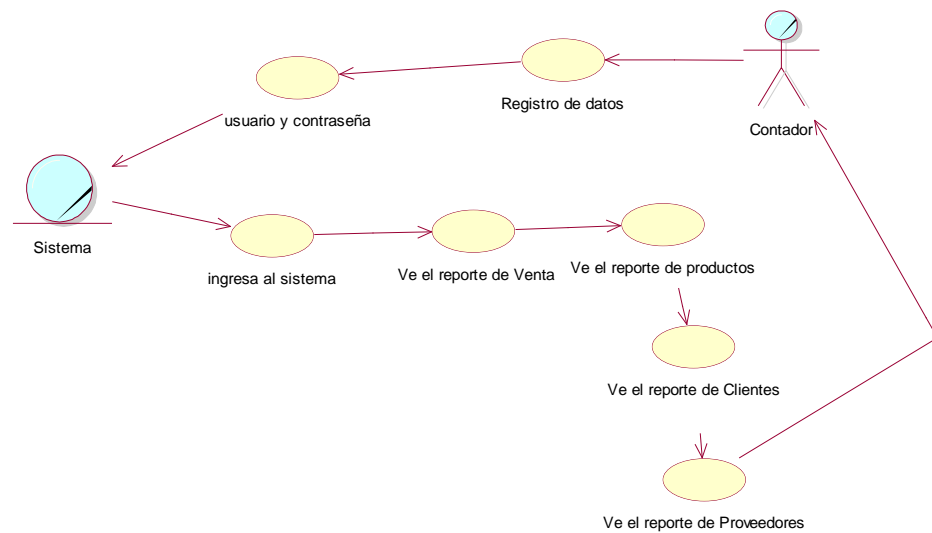


Figura 8. Diagrama de caso de uso Contador UC005.

UC006

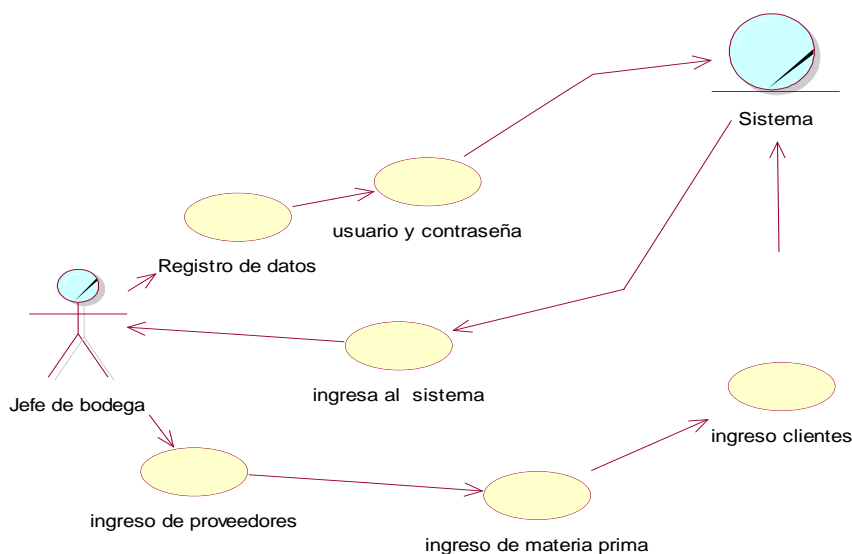


Figura 9. Diagrama de caso de uso Jefe de Bodega UC006.

3.04 Especificación de casos de uso

Tabla 17

Especificación de caso de uso Administrador UC002

Casos de Uso	Administrador
Identificador	UC002
CURSO TÍPICO DE EVENTOS	
Usuario	Sistema
1. Caso de uso del Administrador realiza funciones primordiales dentro del sistema	Administrador realiza asignación de roles, privilegios, accesos al personal.
2. El Administrador tendrá accesibilidad a todo el sistema para realizar mantenimientos y buena administración de datos y recursos.	El sistema permitirá la visualización a los módulos por perfiles de asignación del Administrador de la plataforma al personal.
CURSOS ALTERNATIVOS	
El Administrador realizara procedimientos importantes dentro de cada uno de los módulos.	

Tabla 18

Especificación de caso de uso Personal de Bodega UC003

Casos de Uso	Personal
Identificador	UC003
CURSO TÍPICO DE EVENTOS	
Usuario	Sistema
3. El movimiento del personal se realizara mediante caso de uso presentado.	La información registrada en el sistema debe ser autenticada por el personal el cual realizara una búsqueda para que no haya datos similares.
4. 636+3Registro de la información proporcionada por el personal.	El sistema almacena los datos de manera simultánea.
CURSOS ALTERNATIVOS	
Verificación si el personal si existe para su acceso.	

Tabla 19

Especificación de caso de uso Cliente UC004.

Casos de Uso	Módulo Cliente
Identificador	UC004
CURSO TÍPICO DE EVENTOS	
Usuario	Sistema
5. El cliente hará el pedido de su producto.	El cliente realiza el pedido al personal de bodega el cual verifica en el sistema que exista el producto para realizar la venta.
6. Se puede visualizar el caso de uso del cliente el cual detalla el proceso que va hacer.	Al hacer el pedido debe cancelar el producto para poder recibirlo.
CURSOS ALTERNATIVOS	
El personal realizara la generación de documentos y estos serán guardados en un historial del sistema.	

Tabla 20

Especificación de caso de uso Contador UC005.

Casos de Uso	Contador
Identificador	UC005
CURSO TÍPICO DE EVENTOS	
Usuario	Sistema
7. El Contador será el encargado de verificar la contabilidad de la empresa y ver cada uno de los reportes que el sistema genere.	El sistema almacena los datos de cada movimiento en las bodegas al registrar los productos dentro de la plataforma.
8. El personal deberá ingresar bien al información caso contrario al hacerlo mal el contador procederá a sancionarlos.	La plataforma recibe la información almacena con seguridad.
CURSOS ALTERNATIVOS	
El Contador dar un seguimiento a toda la información del sistema para que no halla fraude de información que perjudique a la empresa.	

Tabla 21

Especificación de caso de uso Jefe de Bodega UC006

Casos de Uso	Jefe de Bodega
Identificador	UC006
CURSO TÍPICO DE EVENTOS	
Usuario	Sistema
9. El Jefe de bodega será el encargado de ingresar toda la materia prima a bodega, proveedores y clientes.	El sistema grabara toda la información que se requiera.
10. El sistema dará reportes de proveedores, clientes productos y ventas con fecha, día y hora.	La plataforma busca y permitirá visualizar todos los reportes requeridos.
CURSOS ALTERNATIVOS	
Dando así una correcta utilización de la información en el sistema.	

3.05 Casos de uso de realización

UCR002:

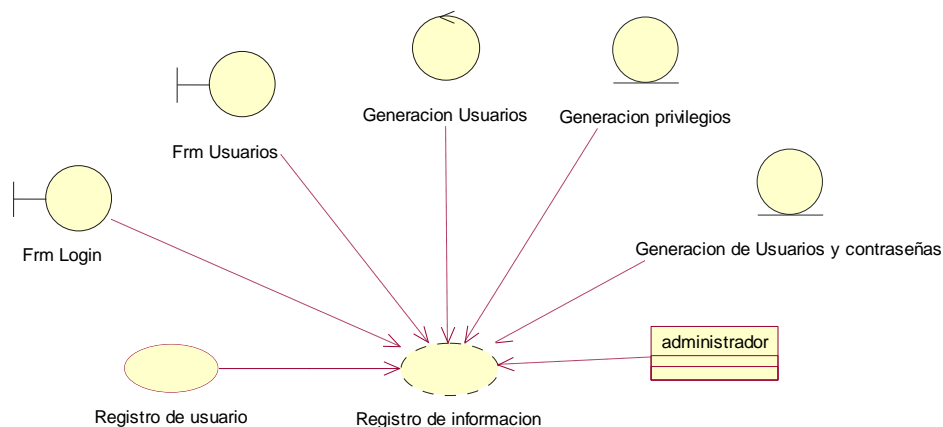


Figura 10. Diagrama de Realización de Administrador UCR002: Se diagrama la representación de la funcionalidad y registros.

Tabla 22

Caso de uso de realización Administrador UCR002

NOMBRE: Registro Administrador

IDENTIFICADOR UCR002

TIPO Sistema

REFERENCIAS DE CASO DE USO: Administrador

PRECONDICIONES

De Instancia

-Ingresos exigidos de parte del sistema con autenticación en la información.

-Verificación de la información de autenticación para validar el acceso.

De Relación

Administrador-Empresa

POSCONDICIONES

De instancia

Validar los campo

De Relación

Administrador – Empresa

Salida de Pantalla

Mensaje de registro completo

UCR003

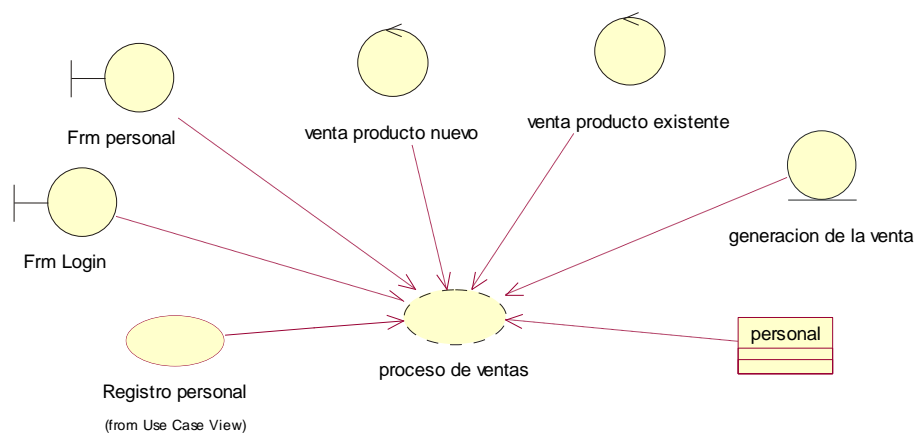


Figura 11. Diagrama Realización Personal UCR003: En el diagrama se visualiza la formulación de registro de personal y su función.

Tabla 23

Caso de uso de realización Personal UCR003

NOMBRE: Proceso Personal

IDENTIFICADOR UCR003

TIPO Sistema

Referencias de caso de uso: Personal

PRECONDICIONES

De Instancia

-Ingresos exigidos de parte del sistema con autenticación en los datos.

-Verificación de la información de autenticación para el acceso.

-Encargado de realizar las ventas por medio del sistema al cliente.

De Relación

Personal-Administrador-Jefe de bodega

POSCONDICIONES

De instancia

Validar los campo

De Relación

Personal-Administrador-Jefe de bodega

Salida de Pantalla

Venta realizada

UCR004

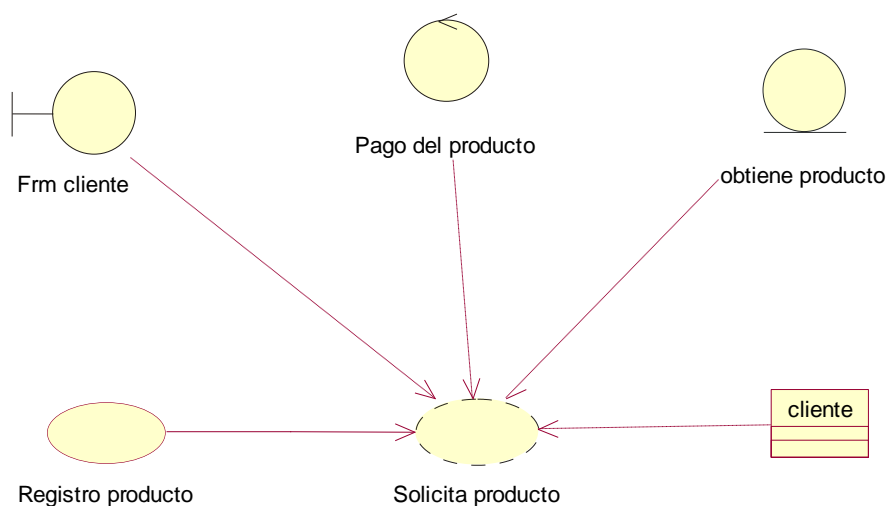


Figura 12. Diagrama de Realización Cliente UCR004: Diagrama donde se visualiza el proceso que el cliente hará para obtener el producto.

Tabla 24

Caso de uso de realización Cliente UCR004

NOMBRE	Proceso Cliente
Identificador	UCR004
TIPO	Sistema
Referencias de caso de uso: Cliente	
PRECONDICIONES	
De Instancia	
--Ingresos exigidos de parte del sistema para registro de datos.	
-Se verifica los datos del cliente para poder realizar la venta del producto.	
De Relación	
Cliente-Personal	
POSCONDICIONES	
De instancia	
Validar los campo	
De Relación	
Cliente-Personal	
Salida de Pantalla	
Total y cambio	

UCR005

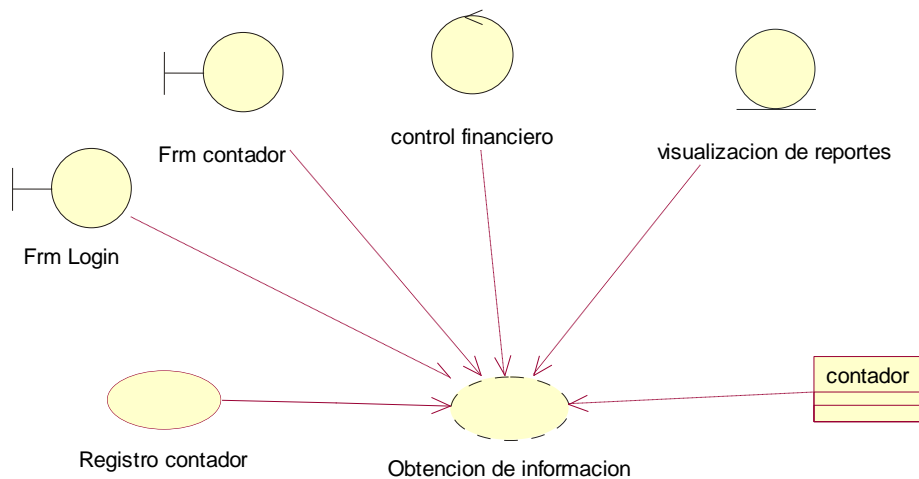


Figura 13. Diagrama Realización Contador UCR005. En este diagrama se visualiza el proceso que realizara el contador dentro del sistema.

Tabla 25

Caso de uso de realización Contador UCR005

NOMBRE	Proceso Contador
Identificador	UCR005
TIPO	Sistema
Referencias de caso de uso: Contador	
PRECONDICIONES	
De Instancia	
<ul style="list-style-type: none"> -Ingresos exigidos de parte del sistema con autenticación de la información. -Verificación de la información para poder acceder al sistema. -Procesos y funciones que realizara dentro del sistema. 	
De Relación	
Contador-Administrador	

POSCONDICIONES

De instancia

Validar los campo

De Relación

Contador-Administrador

Salida de Pantalla

Notificación registro satisfactorio.

UCR006

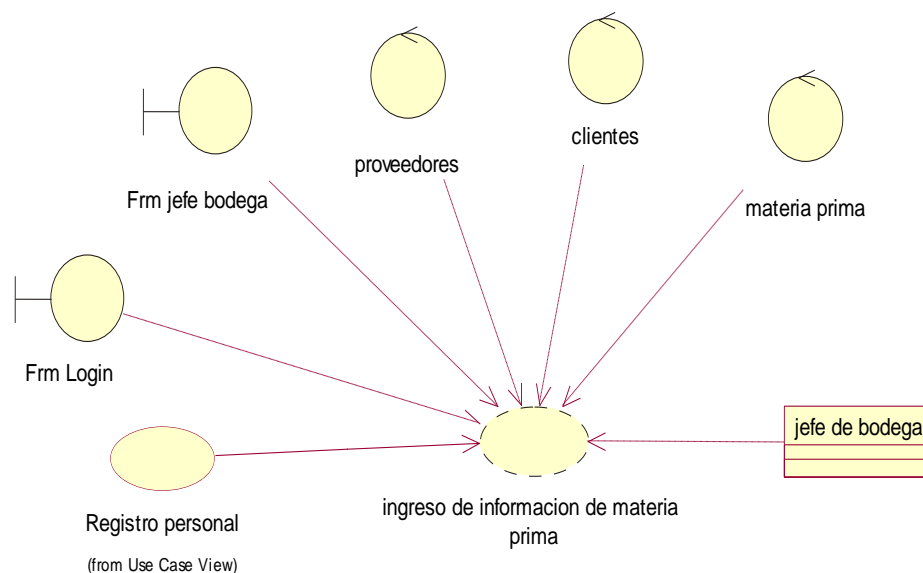


Figura 14. Diagrama Realización Jefe de Bodega UCR006. En este diagrama se visualiza el proceso que realizara el jefe de bodega dentro del sistema.

Tabla 26

Caso de uso de realización Jefe de Bodega UCR006

NOMBRE	Proceso Jefe de Bodega
Identificador	UCR006
TIPO	Sistema
Referencias de caso de uso: Jefe de Bodega	
PRECONDICIONES	
De Instancia	
-Se necesita ingresar datos para la autenticación de la información.	
-Ingresara si los datos son verídicos el cual permitirán el acceso al sistema.	
-Ingreso de proveedores, clientes, materia nueva y existente.	
De Relación	
Jefe de Bodega-Administrador, Personal	
POSCONDICIONES	
De instancia	
Validar los campo	
De Relación	
Jefe de Bodega,Administrador,Personal	

3.05.01 Diagrama de secuencia del sistema

El diagrama de secuencia es la representación de los procesos a seguir a lo largo de los diferentes módulos los cuales se representan gráficamente paso a paso.

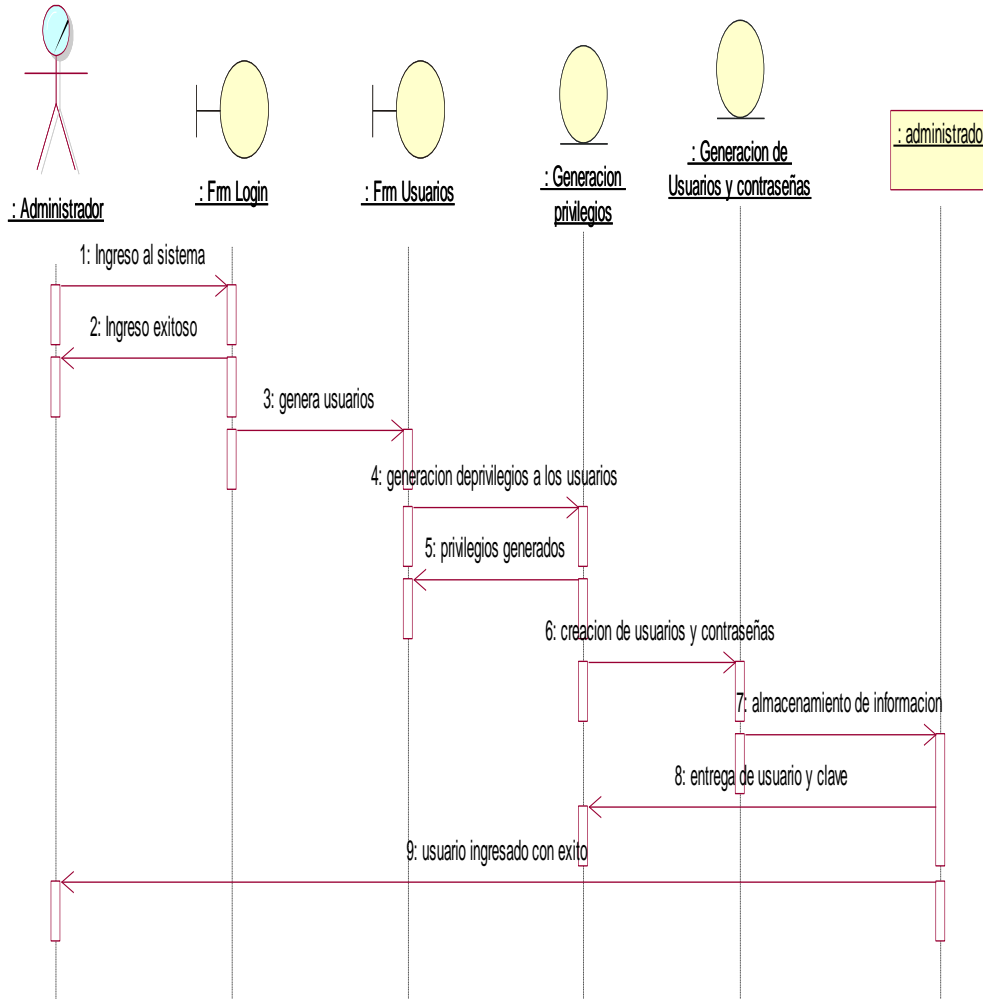


Figura 15. Diagrama de Secuencia Administrador:

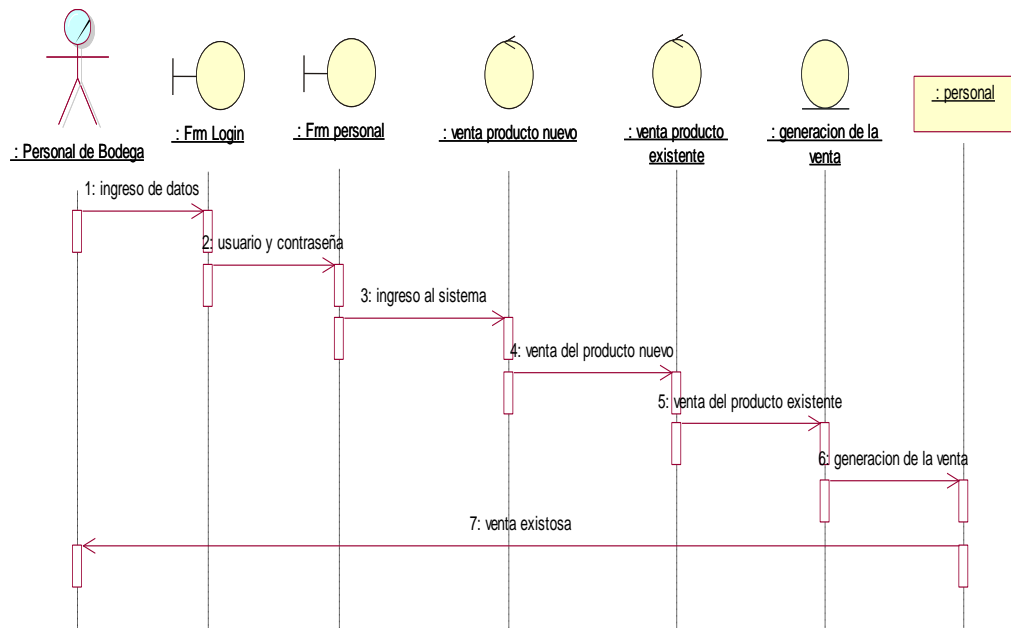


Figura 16. Diagrama de Secuencia Personal.

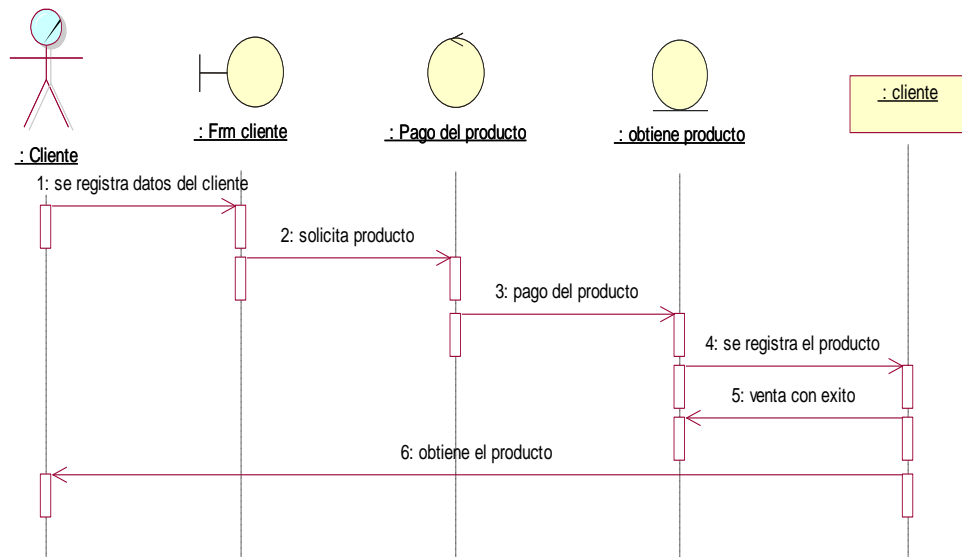


Figura 17. Diagrama de Secuencia Cliente.

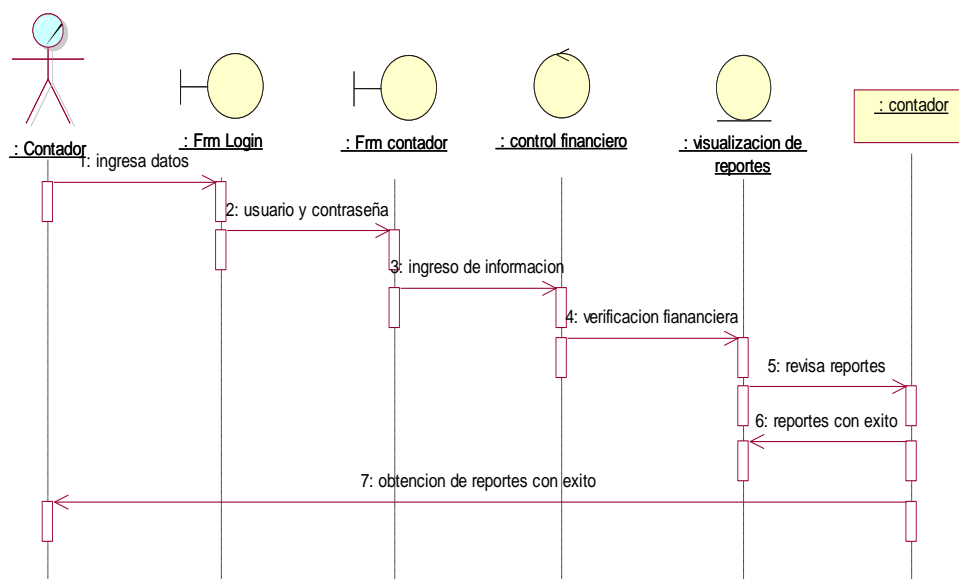


Figura 18. Diagrama de Secuencia Contador.

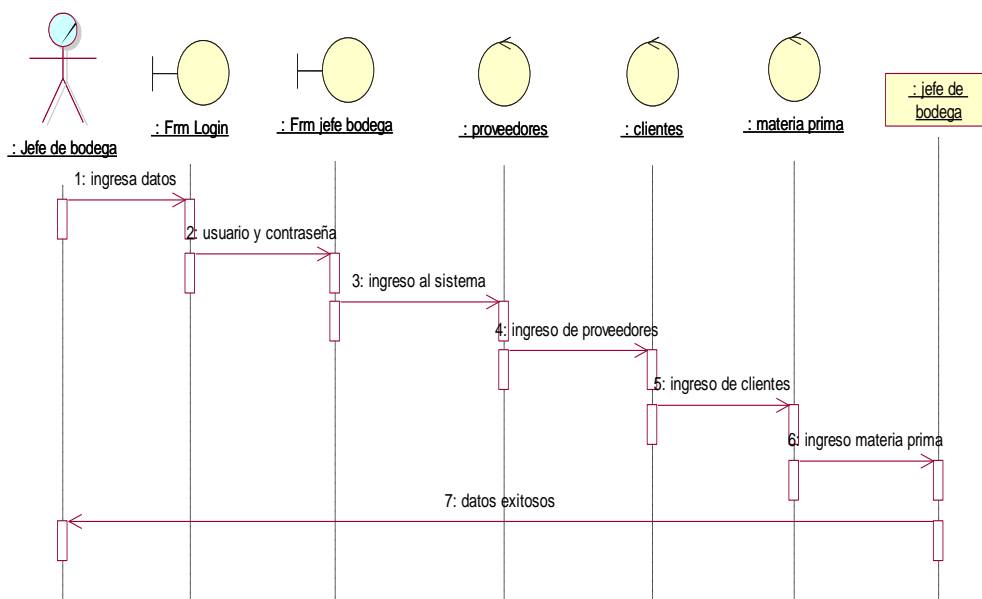


Figura 19. Diagrama de Secuencia de Jefe de Bodega.

Capítulo IV: Análisis de Alternativas

4.01 Matriz de Análisis de Alternativas

Haciendo un análisis de varias alternativas se deriva a una investigación a fondo donde vamos a determinar alternativas y estrategias para dar un buen enfoque al árbol de objetivos, realizaremos ejecuciones donde analizaremos el cambio de la situación real a la situación que debe llegar el objetivo primordial, a cada una de ellas se califica de acuerdo al impacto y factibilidad de tipo técnico, financiero, social y político con una escala del 1 al 5.

Tabla 27

Matriz de análisis de alternativas

Matriz De Análisis De Alternativas							
Objetivos	Impacto sobre el propósito	Factibilidad Técnica	Factibilidad Financiera	Factibilidad Social	Factibilidad Política	Total	Categoría
Automatizar el proceso de inventario y ventas.	5	4	4	4	2	19	Alta
Información precisa al momento de generar la información.	5	4	3	4	2	18	Alta
Reducir el tiempo de ejecución en la generación de la información de la materia prima.	5	4	2	4	3	18	Alta
Mantener en orden la información del personal	5	3	3	4	2	17	Media Alta

Tener siempre el registro de todos los históricos del stock de la materia prima	5	4	2	4	2	17	Media Alta
Mantener la información del personal en confidencialidad.	5	4	2	3	2	16	Media Baja
TOTAL	30	23	16	23	13	105	

Nota: Escala: 1= Baja 2=Media Baja 3=Medio 4=Media Alta 5=Alta

4.02 Matriz de Impacto de los Objetivos

Análisis de cada uno de los objetivos propuestos permite confirmar la realización y verificación de la permanencia que llevara el proyecto a lo largo del desarrollo en la ejecución de tal manera se realizara una visualización al impacto que tendrá a nivel profesional y empresarial con los existentes métodos de los servicios.

Tabla 28

Matriz de análisis de impacto de los objetivos

Objetivos	Factibilidad de lograrse	Impacto de Genero	Impacto Ambiental	Relevancia	Sostenibilidad
Finales	Minimizar procedimientos en los tiempos de ejecución del sistema.	Seguridad de los datos que brinda el sistema.	El entorno cultural es maximizado.	Garantía al personal de la empresa.	El tiempo de búsqueda de la información es rápida.
Incremento del rendimiento o laboral de todo el personal	El rendimiento de todo el personal es favorable.	Mayor fuente de ingreso a la Empresa Muebles Elizabeth.	Comodidad al ambiente laboral del personal.	Genera grandes ingresos a la Empresa.	La distribución y administración del personal es buena.

Confiabilidad en los cálculos de los movimientos al momento de generar ventas	Existe tecnología adecuada para la realización del sistema.	Satisfacción por parte del personal que utiliza el sistema.	Mejoramiento en el entorno social y económico de la empresa.	Reducir fallos de información al generar la venta.	Disponer de información actualizada.
Mantener la información actualizada del personal que está registrado en el sistema	Aumentar la confianza de la empresa con la información verídica del personal.	Aumento de relaciones interpersonales e institucionales con proveedores y clientes.	Se disminuye los suministros de papel al momento de generar los reportes.	Acceder a la información permanentemente.	Capacitación al personal para generar soluciones innovadoras.

4.03 Estándares para el Diseño de Clases

Lenguaje unificado de modelado (UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group).

1

El Lenguaje Unificado de Modelado (UML) fue creado para forjar un lenguaje de modelado visual común y semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento. UML tiene aplicaciones más allá del desarrollo de software, p. ej., en el flujo de procesos en la fabricación.²

¹https://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado, Párrafo 1

²<https://www.lucidchart.com/pages/es/qu%C3%A9-es-el-lenguaje-unificado-de-modelado-uml>, Párrafo 2

Es comparable a los planos usados en otros campos y consiste en diferentes tipos de diagramas. En general, los diagramas UML describen los límites, la estructura y el comportamiento del sistema y los objetos que contiene.³

UML no es un “lenguaje de programación”, pero existen herramientas que se pueden usar para generar código en diversos lenguajes usando los diagramas UML. UML guarda una relación directa con el análisis y el diseño orientados a objetos.⁴

UML no puede compararse con la programación estructurada, pues UML significa Lenguaje Unificado de Modelado, no es programación, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, la programación orientada a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos⁵.

UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas.⁶

³<https://www.lucidchart.com/pages/es/qu%C3%A9-es-el-lenguaje-unificado-de-modelado-uml>, Párrafo 3

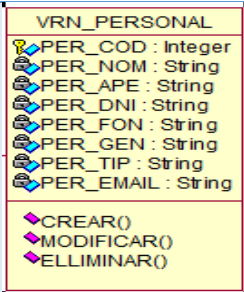
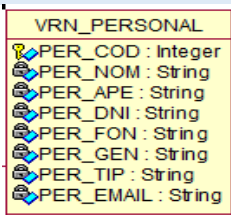
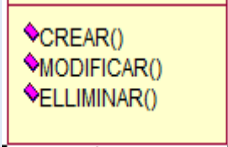
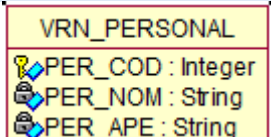


⁴<https://www.lucidchart.com/pages/es/qu%C3%A9-es-el-lenguaje-unificado-de-modelado-uml>, Párrafo 4

⁵ https://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado, Párrafo 5

⁶ https://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado, Párrafo 6

Tabla 29

Parámetros para el diagrama de clases

Nombre	Gráfico	Estándar
Clase	 <pre> classDiagram class VRN_PERSONAL { +PER_COD : Integer +PER_NOM : String +PER_APE : String +PER_DNI : String +PER_FON : String +PER_GEN : String +PER_TIP : String +PER_EMAIL : String +CREAR() +MODIFICAR() +ELIMINAR() } </pre>	<p>El nombre de la clase es nombrado por la relación de sus objetos.</p> <p>Ej.: "VRN_PERSONAL".</p>
Atributos	 <pre> classDiagram class VRN_PERSONAL { +PER_COD : Integer +PER_NOM : String +PER_APE : String +PER_DNI : String +PER_FON : String +PER_GEN : String +PER_TIP : String +PER_EMAIL : String } </pre>	<p>Recomendación utilizar iniciales de la clase para contener los atributos en orden y excelente distribución.</p> <p>Ej.: PER_NOM</p>
Métodos	 <pre> classDiagram class VRN_PERSONAL { +CREAR() +MODIFICAR() +ELIMINAR() } </pre>	<p>Las funciones y operaciones que va a realizar al atributo.</p> <p>Ej.: crear()</p>
Primary Key	 <pre> classDiagram class VRN_PERSONAL { +PER_COD : Integer +PER_NOM : String +PER_APE : String } </pre>	<p>La clave de identificación principal de cada tabla va la primera con el símbolo de la llave amarilla.</p>
Herencia		<p>Subclase hereda los métodos y atributos específicos de la clase superior.</p>
Asociación		<p>Permite asociar objetos que ayudan entre sí en una relación</p>

4.04 Diagrama de Clases

Es aquel donde podemos observar las distintas funciones y como está estructurado el sistema con operaciones y atributos de cada tabla.

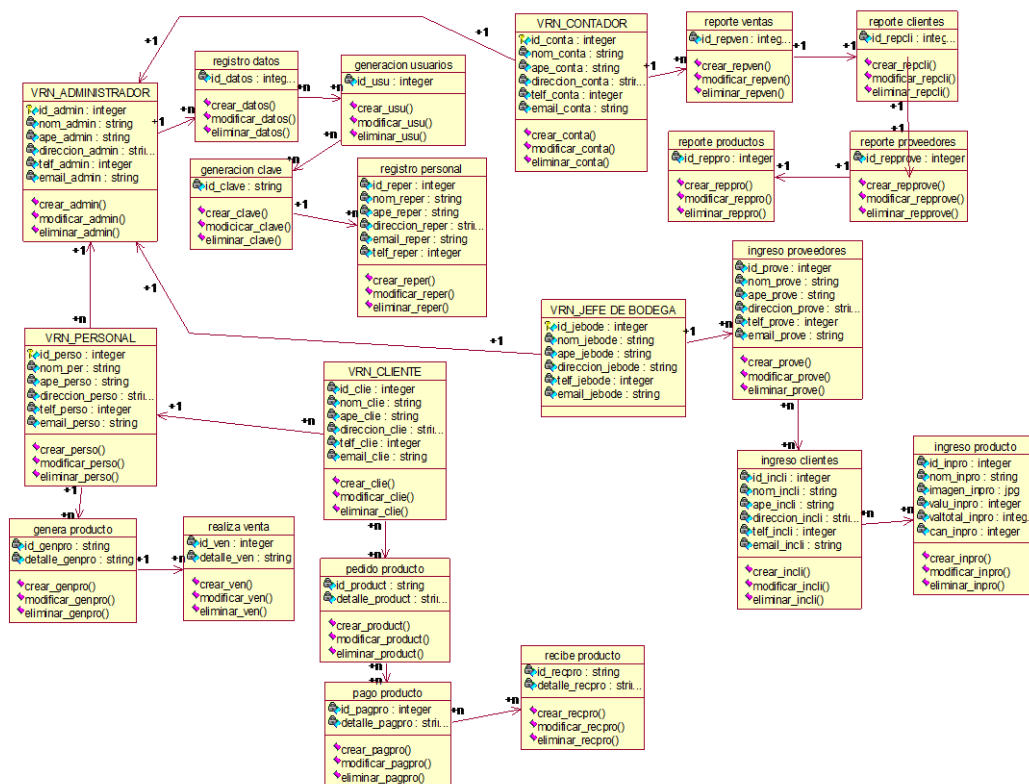


Figura 20. Diagrama de Clases donde visualizamos la estructura del sistema

4.05 Modelo Lógico

En el siguiente diagrama se presenta la convergencia del esquema conceptual (entidad – relación) en un esquema lógico, se presenta la clave primaria, atributos y métodos con los que trabaja nuestra base de datos.

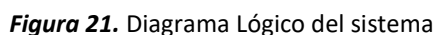


Diagrama Físico

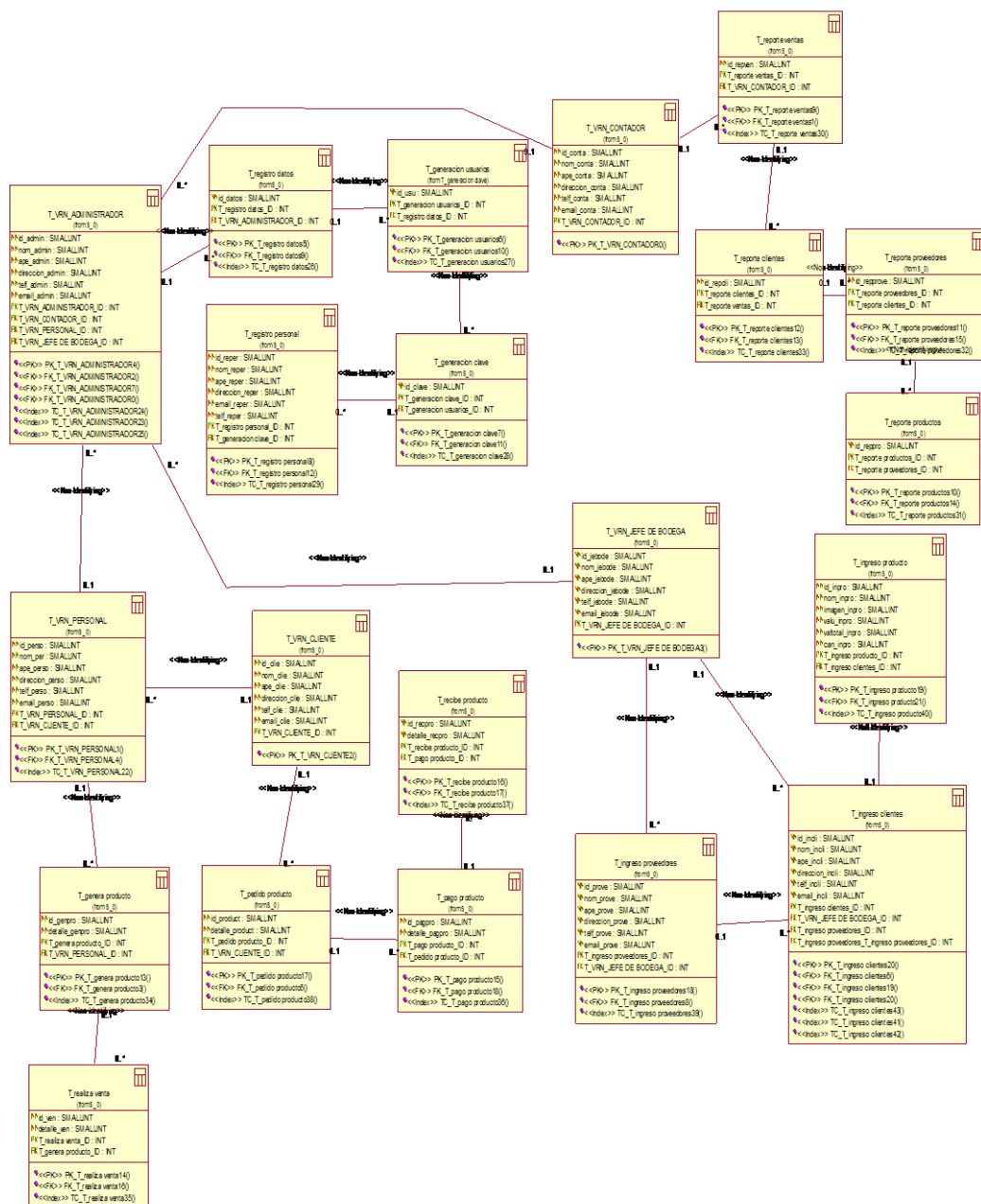


Figura 22. Diagrama Físico del sistema

4.06 Diagrama de Componentes

Es en donde se realiza la visualización de la estructura del sistema en la organización y sus dependencias el cual se realizara el mapeo de los componentes que se va a utilizar en el desarrollo de la programación que se implementara.

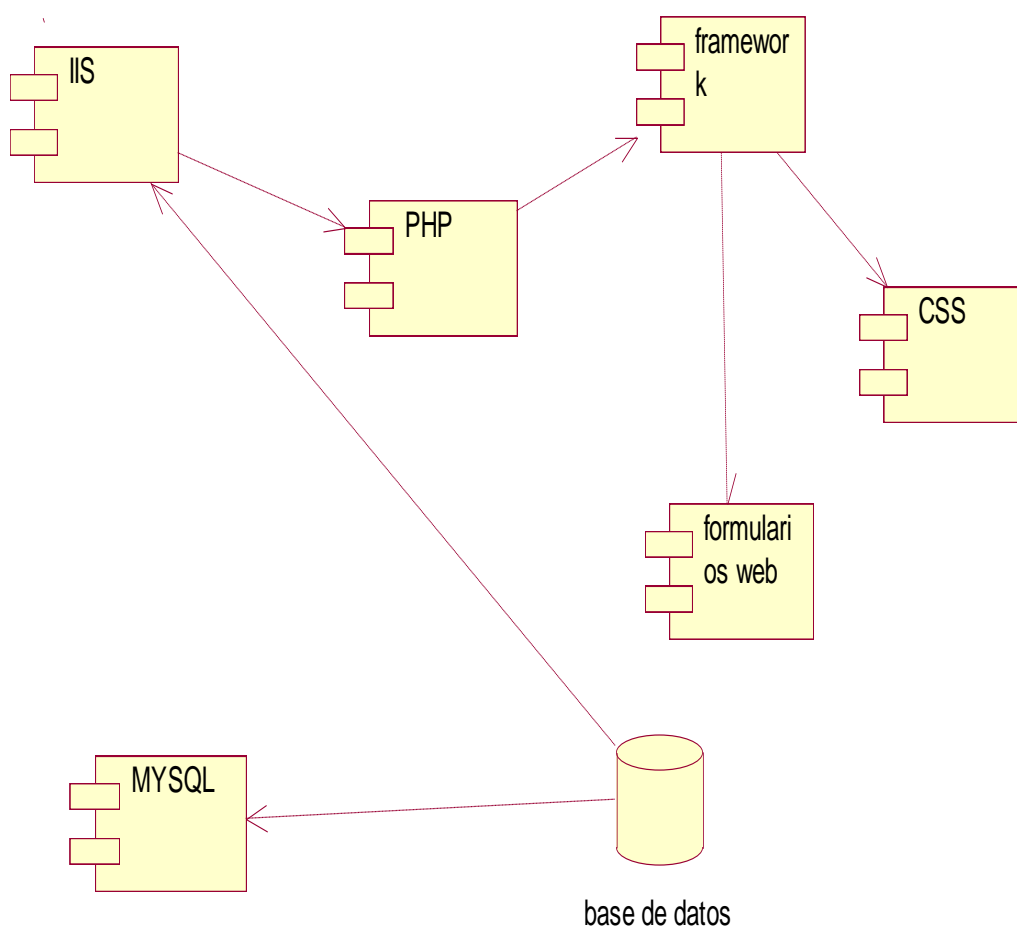


Figura 23. Diagrama de Componentes

4.07 Diagrama de Estrategias

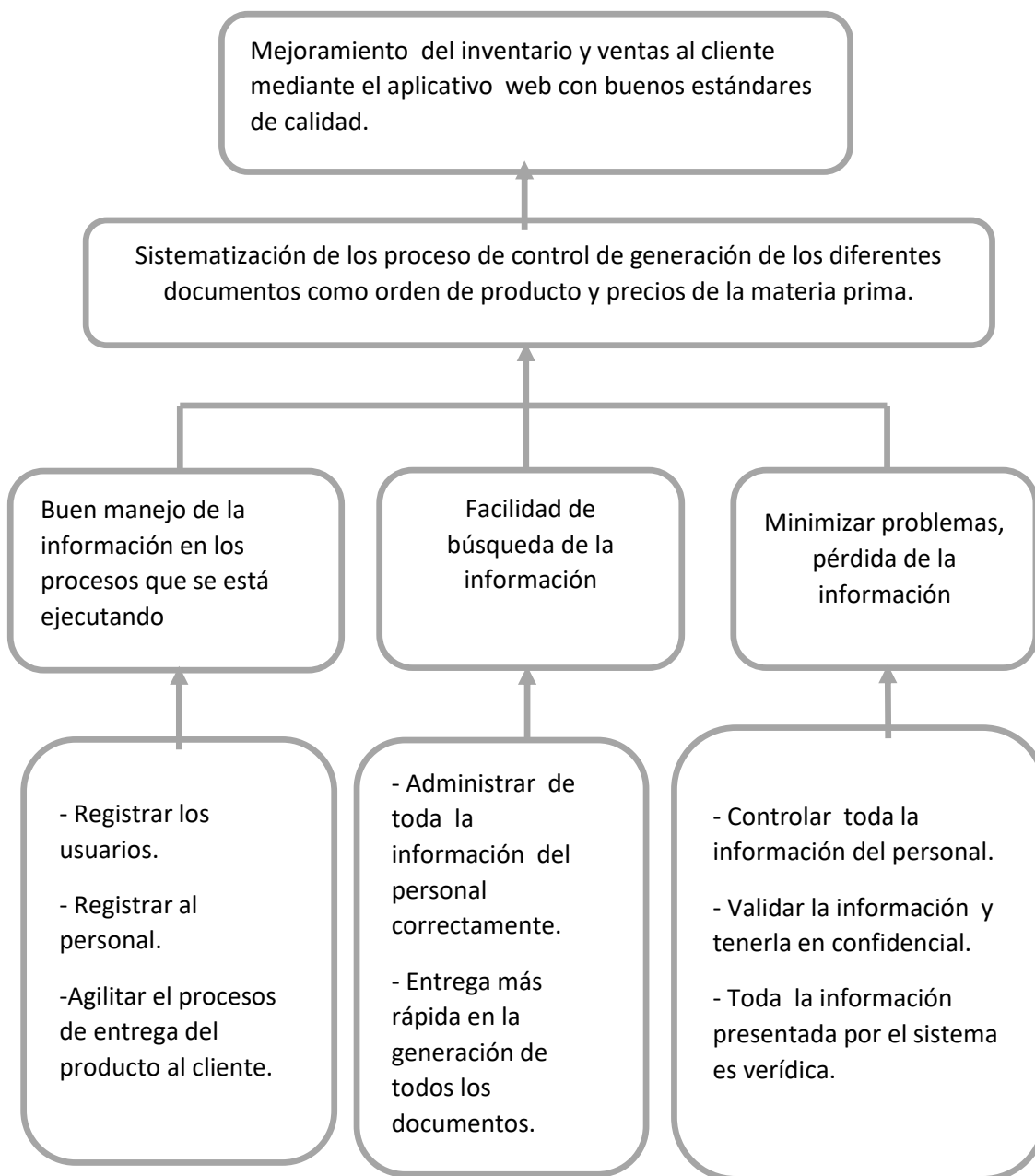


Figura 24. Diagrama de Estrategias

4.08 Matriz de Marco Lógico

Tabla 30

Matriz de Marco Lógico.

RESUMEN NARRATIVO	INDICADORES	MEDIOS DE VERIFICACIÓN	SUPUESTOS
Finalidad			
Mejorar el inventario y ventas mediante el aplicativo web.	Confianza en la información en la entrega de los productos.	Registro de clientes y usuarios Registro de personal	Áreas que pertenecen a la empresa respecto a la elaboración del producto.
Propósito			
Sistematización del proceso de orden y venta del producto mediante el aplicativo web.	Satisfacción de la empresa al momento de realizar la sistematización de las ventas de los productos de bodega.	Mantenimiento de la información deben ser mensuales para tener una buena administración.	Por una mala capacitación del uso de la herramienta tecnológica se puede perder de información del sistema.
Componentes			
- Automatizar el proceso de inventario de ingresos y egresos del producto a bodega.	Minimizar el tiempo de generación de reportes.	Describirle al personal sobre la herramienta que fue desarrollada.	El personal y los usuarios se encuentran molestos por el servicio prestado.
Actividades			
- Mejorar la actividad laboral de la empresa.	La generación de los reportes es de manera rápida al momento de realizar el proceso.	- Registro del proceso de órdenes de pedido y venta.	- El Personal no es eficiente en el computador.
- Hacer el levantamiento de requerimientos.			- La empresa debe mantener el aplicativo web en línea.
- Desarrollar y Diseñar el sistema.			- Uso erróneo de la información de usuarios.

4.09 Vistas Arquitectónicas

La visualización de estructura y arquitectura de un sistema se clasifica en componentes, por lo cual la interconexión realizada entre los componentes realiza una buena comunicación.

4.09.01 Vista Lógica

Diagrama de arquitectura llamado también vista lógica realizada en base a la organización e infraestructura del sistema, se manifiestan los cuatro principales requisitos funcionales como: abstracción, encapsulación y herencia realizando un requisito funcional por lo cual el sistema debe llevar los servicios al personal y al cliente.

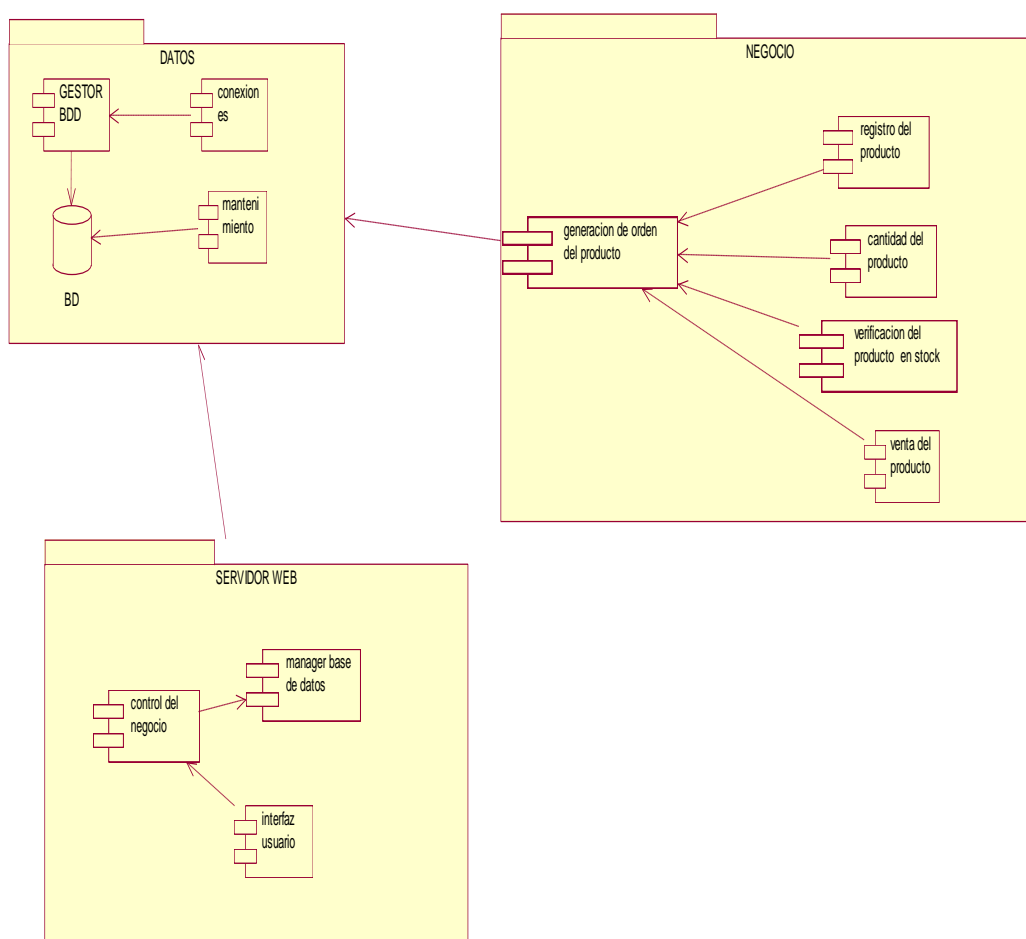


Figura 25. Vista Lógica

4.09.02 Vista Física

En este diagrama se toma en cuenta los componentes, el cual presenta los requisitos funcionales y no funcionales del sistema.

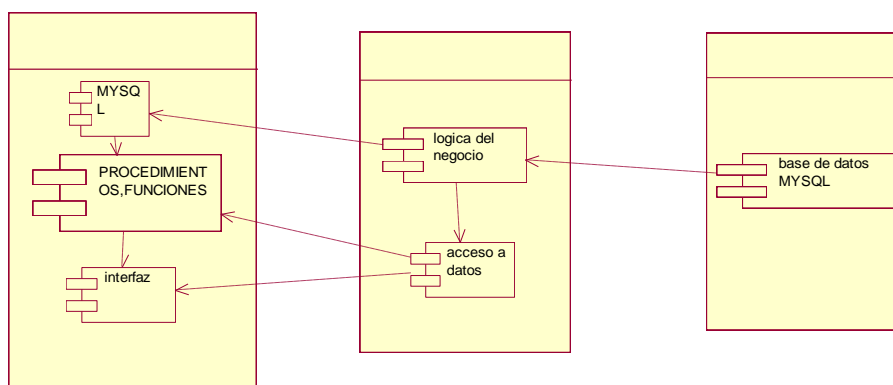


Figura 26. Vista Física

4.09.03 Vista de Desarrollo

Se presenta la organización de los módulos del sistema que realizara un empaquetamiento de partes de subsistemas en el cual se presentan las jerarquías.

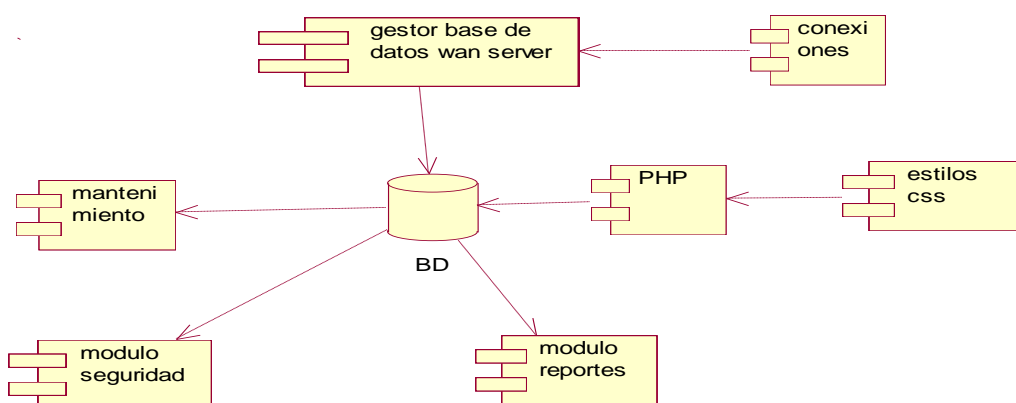


Figura 27. Vista de Desarrollo

4.09.04 Vista de Procesos

En el siguiente diagrama de procesos se puede identificar cada uno de los procedimientos en base a los objetivos de concurrencia, distribución, integridad del sistema y tolerancia a errores.

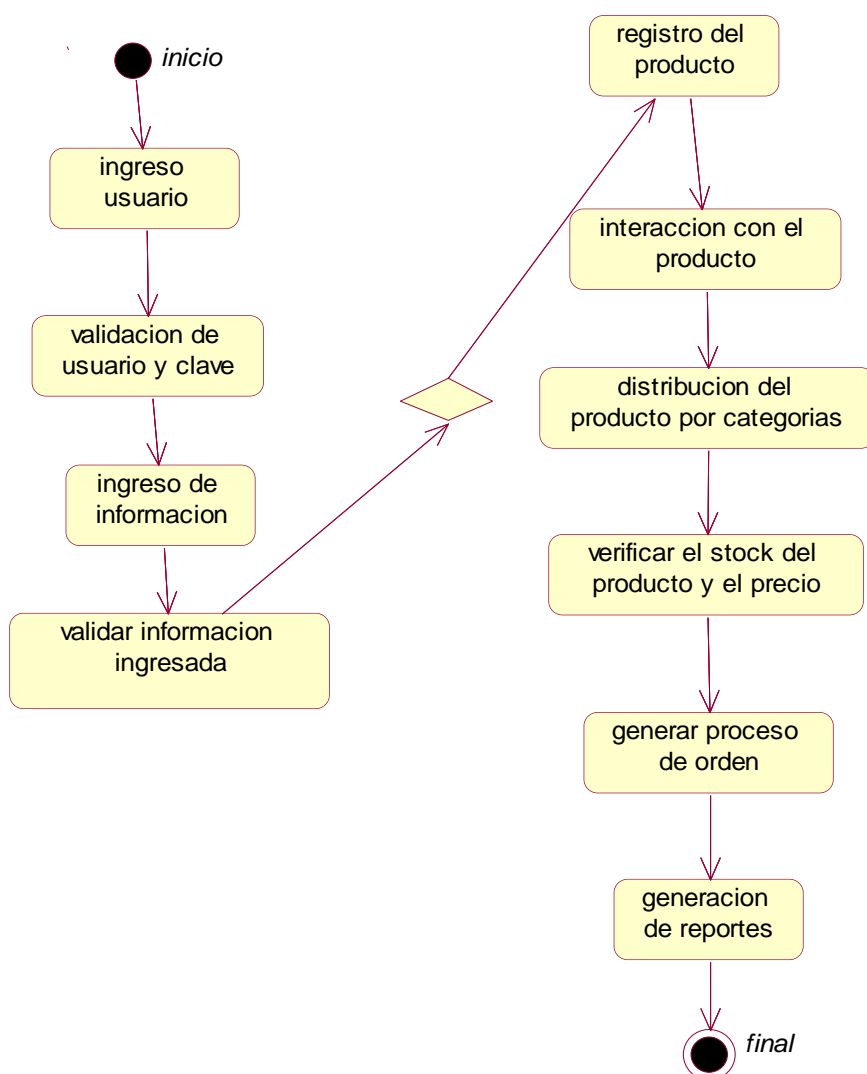


Figura 28. Vista de Procesos

Capítulo V: Propuesta

5.01 Especificación de estándares de programación

Estándar de programación es la forma de normalizar la programación por lo cual al laborar un proyecto cualquiera de las personas involucradas en el mismo podrán tener acceso y comprenderán el código porque serán considerados como guías en las etapas del diseño del sistemas que se está realizando.

Se define la estructura y la organización del código fuente de un programa. Al momento de seguir un estándar de programación se facilita como programador la modificación del código fuente aunque no se esté trabajando en el mismo equipo que se realiza el sistema.

La legibilidad del código fuente repercute directamente en lo bien que un programador comprende un sistema de software. El mantenimiento del código es la facilidad con que el sistema de software puede modificarse para añadirle nuevas características, modificar las ya existentes, depurar errores, o mejorar el rendimiento.

(Creación de Software, 2004, Capítulo 5: Concepto, párr. 1).

5.01.01 Prefijos aplicados para controladores

Tabla 31

Descripción de las Variables

Tipo de variable	Estructura
Integer	[per_cod] [int] NOT NULL
Varchar	[per_nom] [varchar] NOT NULL
Char	[per_tip] [char] NOT NULL
Decimal	[pro_pre] [decimal(10,2)] NOT NULL
Date	[pro_fech] [date] NOT NULL

Tabla 32

Descripción de Control de Prefijos

Tipo de Control	Prefijo	Ejemplo
Label	Lbl	lbl_ced
Button	Btn	btn_guar
TextBox	Txt	txt_dni
Calendar	Cal	cal_fech
Imagen	Img	txt_img
ComboBox	Cbo	Cbo_gen

5.02 Diseño de interfaces de usuario

Inicio del Sistema

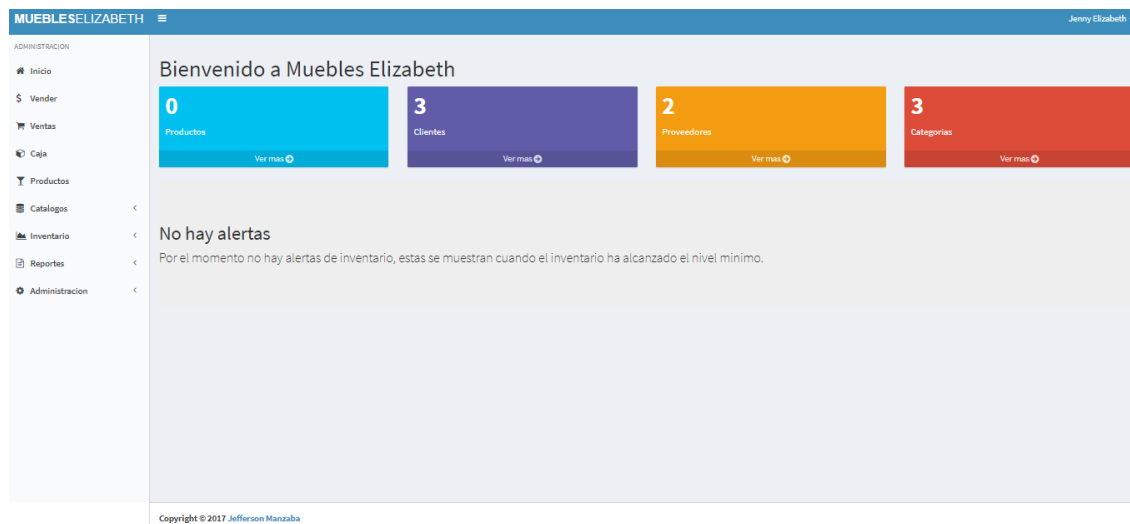


Figura 29. Visualización de inicio del Inicio del Sistema

Tabla 33

Componentes que se utilizaron para el Ingreso de Login

Ítems	Representaciones	Descripciones
A	Textbox	Usuario
B	Textbox	Contraseña
C	Button	Acceder



Figura 30. Visualización de Login: en la siguiente figura se logra visualizar el ingreso del usuario.

Tabla 34

Componentes que se utilizaron para el Ingreso Usuario

Ítem	Representación	Descripción
A	Button	Nuevo Usuario
B	Button	Editar
C	Label	Nombres
D	Label	Apellidos
E	Label	Nombre de Usuario
F	Label	Email
G	Label	Contraseña
H	Button	Agregar Usuario

Ingreso Usuario

MUEBLESELIZABETH Jenny Elizabeth

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Agregar Usuario

E Nombres* C
 Apellidos* D
 Nombre de usuario*
 Email*
 Contraseña G

* Campos obligatorios

H Agregar Usuario

Copyright © 2017 Jefferson Manzaña

MUEBLESELIZABETH Jenny Elizabeth

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Lista de Usuarios

A Nuevo Usuario

Nombre completo	Nombre de usuario	Email	Activo	Admin	
Jenny Elizabeth Castro Pazmiño	Jenny	jenny45@hotmail.es	✓	✓	B Editar
Dario Alberto Casa Hernandez	Dario	dario29@gmail.com	✓		Editar

Copyright © 2017 Jefferson Manzaña

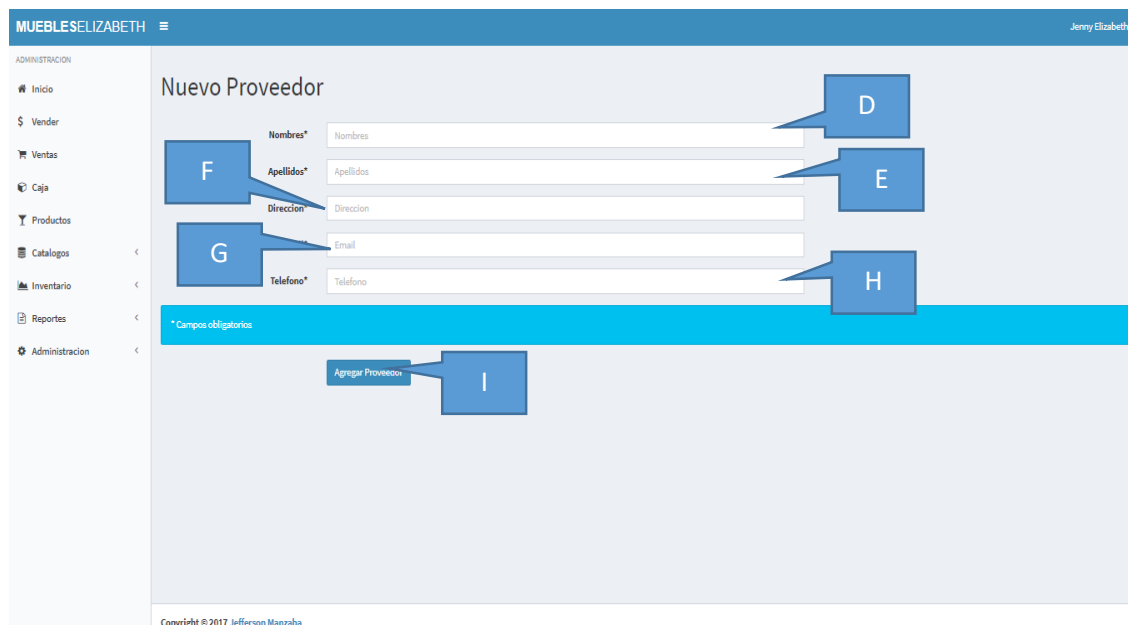
Figura 31. Interfaz Usuario

Tabla 35

Componentes que se utilizaron para el ingreso Proveedor

Ítem	Representación	Descripción
A	Button	Nuevo Proveedor
B	Button	Editar
C	Button	Eliminar
D	Label	Nombres
E	Label	Apellidos
F	Label	Dirección
G	Label	Email
H	Label	Teléfono
I	Button	Agregar proveedor

Ingreso de Proveedores



MUEBLES ELIZABETH

Jerry Elizabeth

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Nuevo Proveedor

Nombres* Nombres

Apellidos* Apellidos

Dirección Dirección

Email Email

Teléfono* Teléfono

* Campos obligatorios

Agregar Proveedor

Copyright © 2017 Jefferson Manzano

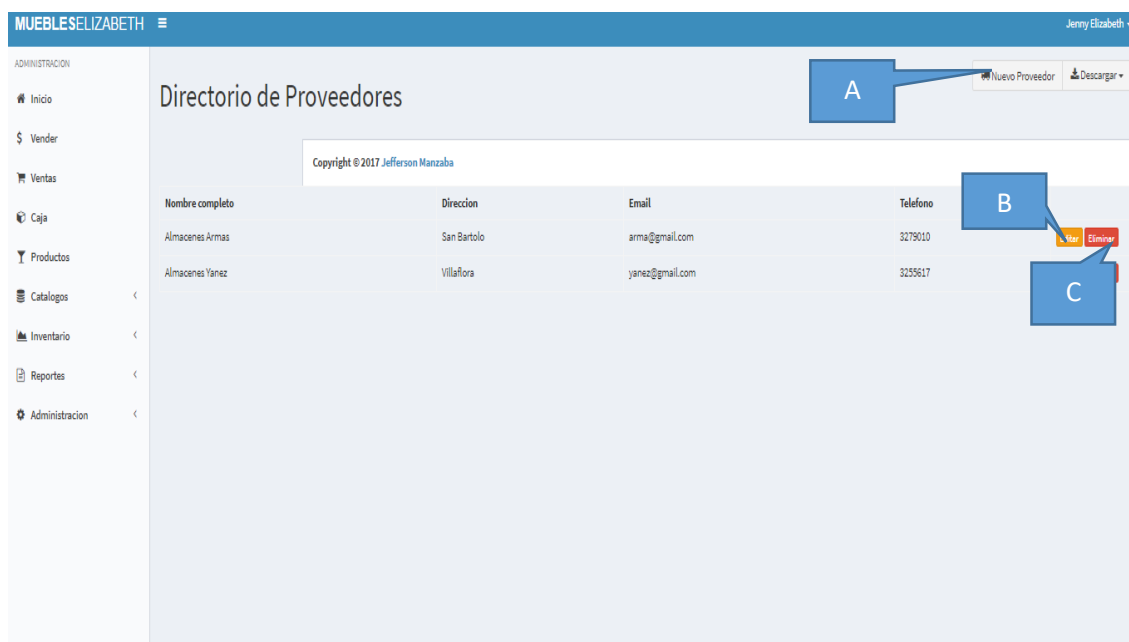


Figura 32. Interfaz Ingreso de Proveedores

Tabla 36

Componentes que se utilizaron para el Ingreso de Clientes

Ítem	Representación	Descripción
A	Button	Nuevo Cliente
B	Button	Editar
C	Button	Eliminar
D	Label	Nombres
E	Label	Apellidos
F	Label	Dirección
G	Label	Email
H	Label	Teléfono
I	Button	Agregar Cliente

Ingreso de Clientes

MUEBLES ELIZABETH Jenny Elizabeth

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Nuevo Cliente

Nombres* **D**
Apellidos* **E**
Direccion* **F**
Email* **G**
Telefono* **H**

*** Campos obligatorios** **C**

Agregar Cliente **B**

Copyright © 2017 Jefferson Manza

MUEBLES ELIZABETH Jenny Elizabeth

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Directorio de Clientes

A Nuevo Cliente Descargar

Copyright © 2017 Jefferson Manza

Nombre completo	Direccion	Email	Telefono	B
EDISON MOLINA	SAN BLAS	EDISON@POPUSI.COM	521452145	Editar Eliminar
Johnny Coronel	La Prensa	jony@gmail.com	3256894	Editar Eliminar
Laura Lorena Alban Torres	San Martin	lore@gmail.com	3259687	Editar Eliminar C

Figura 33. Interfaz Ingreso de Clientes

Tabla 37

Componentes que se utilizaron para el Ingreso de Categorías de Productos

Ítem	Representación	Descripción
A	Button	Nueva Categoría
B	Button	Editar
C	Button	Eliminar
D	Label	Nombre
E	Button	Agregar Categoría

Ingreso de Categorías Productos

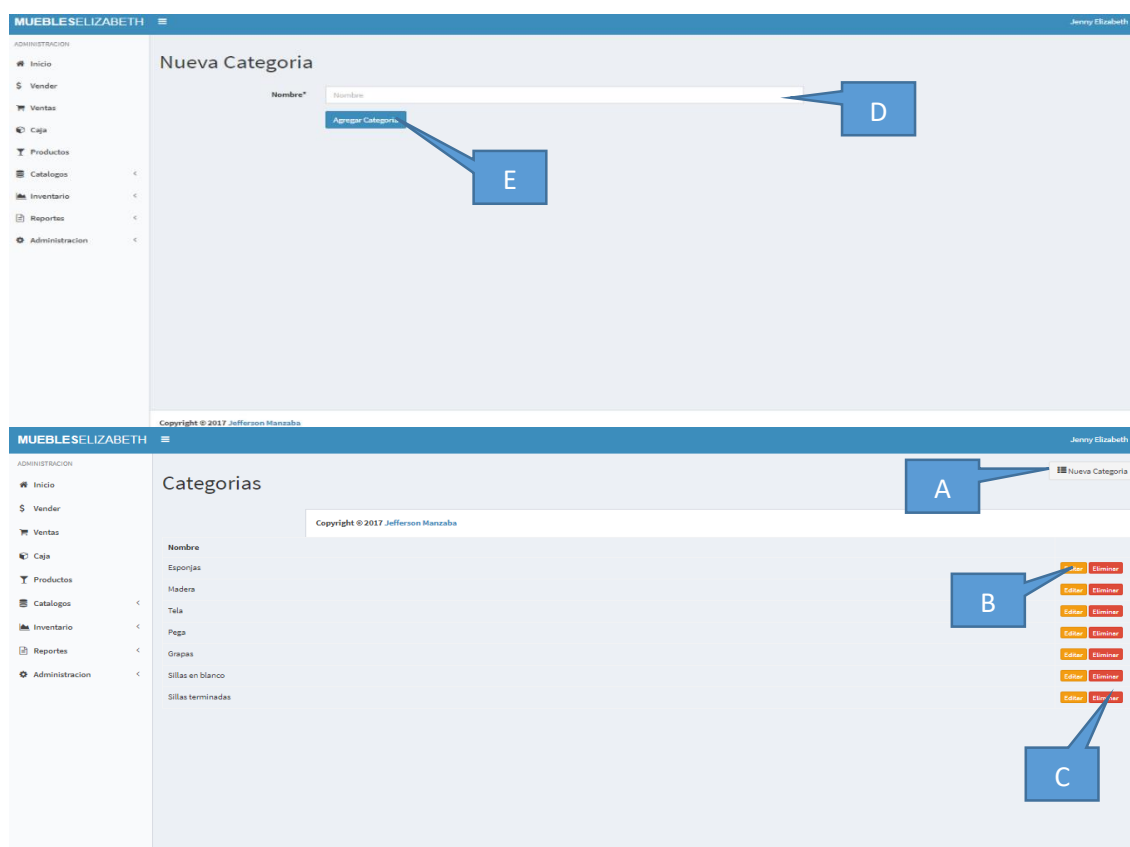


Figura 34. Interfaz Categorías de Productos

Tabla 38

Componentes que se utilizó en el sistema Interfaz Ingreso de Productos

Ítem	Representación	Descripción
A	Button	Agregar Producto
B	Button	Editar
C	Button	Eliminar
D	Button	Subir Imagen
E	Label	Cuadros de textos
F	Label	Código
G	Label	Nombre
H	Label	Categoría
I	Label	Descripción
J	Label	Precio de entrada
K	Label	Precio de salida
L	Label	Unidad
M	Label	Presentación
N	Label	Mínima en inventario
Ñ	Button	Inventario inicial

Ingreso de Productos

MUEBLESELIZABETH Jenny Elizabeth

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Nuevo Producto

Imagen: No se eligió archivo

Código*:

Nombre*:

Categoría:

Descripción:

Precio de Entrada*:

Precio de Salida*:

Unidad*:

Presentación:

Mínima en Inventario:

Inventario Inicial:

MUEBLESELIZABETH Jenny Elizabeth

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Lista de Productos

Página 2 de 2

Código	Imagen	Nombre	Precio Entrada	Precio Salida	Categoría	Mínima	Activo
11		Gallón cola blanca	\$ 5.00	\$ 10.00	Pega	10	<input checked="" type="checkbox"/>
12		Grapas delgadas	\$ 3.00	\$ 5.00	Grapas	20	<input checked="" type="checkbox"/>
13		Grapas de clavos	\$ 10.00	\$ 14.00	Grapas	20	<input checked="" type="checkbox"/>
14		Silla en seike lineal	\$ 30.00	\$ 40.00	Sillas en blanco	20	<input checked="" type="checkbox"/>
15		Silla Regina	\$ 120.00	\$ 200.00	Sillas terminadas	20	<input checked="" type="checkbox"/>

Figura 35. Interfaz Ingreso de Productos

Tabla 39

Componentes que se utilizaron para realizar el Ingreso de Venta

Ítem	Representación	Descripción
A	Label	Buscar Producto nombre
B	Button	Buscar
C	Label	Cliente
D	Label	Descuento
E	Label	Efectivo
F	Button	Cancelar
G	Button	Finalizar venta

Ingreso de Venta

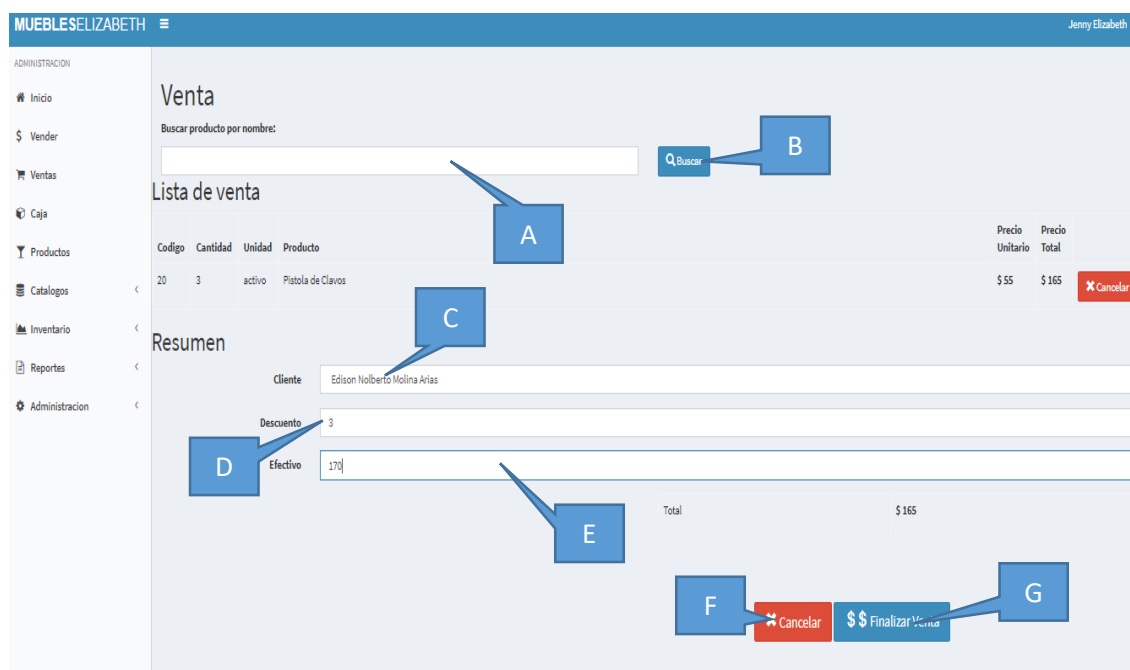


Figura 36. Interfaz Ingreso de Venta

5.03 Especificación de pruebas de unidad

Las pruebas consisten en realizar un enfoque determinado de los procesos que va a realizar el sistema en sus procedimientos en cada uno de los módulos para que exista una buena satisfacción del funcionamiento del nivel físico y lógico en la empresa y al personal que manejara el sistema web.

Tabla 40

Especificación de prueba de unidad EPDU001

Identificador de la prueba:	EPDU001
Método a probar:	Interface
Objetivo de la prueba:	El Sistema va a ser el más factible para el usuario.
Datos de Entrada	
Registro de información personal de la empresa y clientes.	
Resultado Esperado	
<ol style="list-style-type: none">1. Analizar los errores que existan en la ejecución de registro de datos.2. Corregir los errores al momento de manipularlos.3. Administrar y Distribuir de buena manera todos los procedimientos en el sistema.	
Comentarios	
El sistema debe tener una facilidad de manejo al momento que un usuario lo pone en función.	

Tabla 41

Especificación de prueba de unidad EPDU002 Registro nuevo

Identificador de la prueba:	EPDU002
Método a probar:	Verificar los campos al momento de ingresar la información.
Objetivo de la prueba:	El sistema permitirá el registro de un nuevo usuario si existen datos iguales el sistema no permitirá hacer duplicados de la información.
Datos de Entrada	
Ingreso los datos del personal requerido.	
Resultado Esperado	
<ol style="list-style-type: none">1. Verificación de los campos que estén llenos.2. Verificación de los campos de texto que solo ingresen textos.3. Verificación de los campos numéricos que solo ingresen números.4. Verificación del campo calendario todo registro de las fechas están parametrizadas.5. Verificación de los registros que no exista duplicado de información.	

Comentarios

Realizar el ingreso de toda la información que especifica el sistema.

Tabla 42

Especificación de prueba de aceptación EPDU003 Login

Identificador de la prueba:	EPDU003
Método a probar:	Verificación de los datos de acceso al sistema.
Objetivo de la prueba:	Autenticación del uso de la información emitida por el administrador y personal que está en función.
Datos de Entrada	
Usuarios, Administrador y Password.	
Resultado Esperado	
<ol style="list-style-type: none">1. El usuario debe estar registrado y si no lo está se procederá a realizar su registro.2. El ingreso de la contraseña debe ser correcta.3. El usuario deberá ingresar únicamente al módulo que el administrador le diga.	
Comentarios	
Si la información no es correcta no se podrá ingresar al sistema.	

Tabla 43

Especificación de prueba de unidad EPDU004 Mantenimiento a los distintos Módulos

Identificador de la prueba:	EPDU004
Método a probar:	Mantenimientos a los distintos módulos del sistema.
Objetivo de la prueba:	Se da un enfoque a la información mediante una actualización, eliminación y registro de nueva información para brindar una mejor herramienta a la empresa.
Datos de Entrada	
Registro de información en todo el sistema.	
Resultado Esperado	
<ol style="list-style-type: none">1. Verificación de información ingresada.2. Verificación de información eliminada.3. Verificación de información modificada.4. Verificación de nueva información.	
Comentarios	
Realizando las siguientes modificaciones en los módulos se dará un mejor aspecto de trabajo al personal que está en función.	

5.04 Especificación de pruebas de aceptación

Realizando las pruebas satisfactorias de aceptación del desarrollo final del sistema web el cual se encuentra encaminado con el objetivo de verificar las pruebas del sistema.

Tabla 44

Especificación de prueba de aceptación EPDA001

Identificador de la prueba:	EPDA001
Caso de uso	UCR001
Tipo de usuario:	Administrador
Objetivo de la prueba:	Ingresa información del personal, clientes y materia prima.
Secuencia de eventos	
1. Asigna perfiles de accesos.	
2. Mantenimientos de los módulos del sistema.	
3. Registro y Modificaciones de la información en el sistema.	
Resultados esperados	
Realización de mantenimientos y procesos del sistema.	
Comentarios	
Asignación de perfiles de ingreso y registro nuevo de usuarios al sistema.	
Estado:	Aceptado

Tabla 45

Especificación de prueba de aceptación EPDA002 Ingreso Proveedores

Identificador de la prueba:	EPDA002
Caso de uso	UCR002
Tipo de usuario:	Jefe de Bodega
Objetivo de la prueba:	Registro de los proveedores que están trabajando con la Empresa Muebles Elizabeth.
Secuencia de eventos	
1. Registro Nombre.	
2. Registro Dirección.	
3. Registro Email.	
4. Registro Teléfono.	
Resultados esperados	
Registro de la información sin problema.	
Comentarios	
Se realiza la validación de datos ingresados.	
Estado:	Aceptado

5.05 Especificación de pruebas de carga

Realizando las respectivas pruebas de carga en función de los procesos del sistema desarrollando así su funcionabilidad para exigir toda su capacidad para realizar las respectivas ejecuciones al momento que el personal está en función.

Se relaciona con las pruebas que se ha hecho viendo el tiempo de respuesta, tiempo de entrega y envíos de información al usuario.

Se realiza esto con el objetivo de tener un excelente procesamiento de respuesta al momento de enviar la información para satisfacer las necesidades del usuario y la empresa que trabaja con sus clientes para que no existan inconvenientes de demoras en cargar la información.

Tabla 46

Especificación de prueba de carga EPDC001 Actualización del Sistema

Identificador de la prueba:	EPDC001
Tipo de prueba:	Actualización
Objetivo de la prueba:	Realización del tiempo de ejecución en el procesamiento de información.
Descripción	
Verificación del tiempo de respuesta del sistema.	
Resultados esperados	
Actualización verificada el tiempo de respuesta es exitoso.	
Comentarios	
No se encontraron errores al momento de actualizar el sistema.	

Tabla 47

Especificación de prueba de carga EPDC002 Registros de Datos

Identificador de la prueba:	EPDC002
Tipo de prueba:	Verificación del funcionamiento al momento de enviar información en grandes cantidades.
Objetivo de la prueba:	Realizar muchos registros con el fin de que el sistema se mantenga firme.
Descripción	Registro de usuarios, personal, producto, clientes, proveedores, categorías del producto.
Resultados esperados	Verificación exitosa el ingreso de datos es satisfactorio no se perdió ningún tipo de información.
Comentarios	Verificación exitosa.

Tabla 48

Especificación de prueba de carga EPDC003 Validación de Registros

Identificador de la prueba:	EPDC003
Tipo de prueba:	Verificación de Validaciones
Objetivo de la prueba:	Autenticación de registro con las respectivas verificaciones.
Descripción	<ul style="list-style-type: none">- Registros de usuarios.- Registro de clientes.- Registro de proveedores.- Registro de categoría del producto.- Registro de producto.- Registro de ventas.
Resultados esperados	Verificación de información si existe duplicados en los datos o no se completan los campos no se registraran.
Comentarios	Sin problemas.

5.06 Configuración del Ambiente mínima / ideal

Requerimientos Mínimos

Tabla 49

Especificación de requerimientos del Hardware

Gestor de Base de Datos (MYSQL)	<ul style="list-style-type: none">- Intel Core i5 3 GHz (Procesador)- RAM 4 GB (8 GB)- Hard Disk (1TB)
Equipo Administrador	<ul style="list-style-type: none">- Intel Core i3 2 GHz (Procesador)- RAM 4 GB- Hard Disk (1TB)
Equipo Personal	<ul style="list-style-type: none">- Intel Core i3 2 GHz (Procesador)- RAM 4 GB- Hard Disk (1TB)

Tabla 50

*Especificación de Requerimientos del Software***Equipo Administrador**

- Sistema operativo (S.O) Windows 7, 8, 10

? Gestor de Base de Datos

- XAMPP
- MYSQL

? Plataforma de Programación

- PHP
- NOTE PAD++

Requisitos recomendados

Es necesario estas capacidades ya que son de suma importancia al momento de procesar la información en grandes cantidades para cumplir las necesidades del Administrador y personal de la Empresa lo mínimo son estos requisitos fundamentales para el desarrollo del sistema.

Capítulo VI: Aspectos Administrativos

6.01 Recursos

Tabla 51

Recursos humanos utilizados en la realización del Sistema

P8/ersona	Nombre	Cargo
Tutor	Ing. Jhonny Coronel	Tutor Responsable del seguimiento del proyecto.
Lector	Ing. Jaime Basantes	Lector asesor del seguimiento del proyecto.
Autor	Est. Jefferson Alexander Manzaba Castro	Realización del documentos y programación del aplicativo web.

Tabla 52

Recursos económicos utilizados en la realización del Sistema.

RECURSOS ECONÓMICOS			
Detalle	Cantidad	Precio unitario	Valor total
EQUIPO DE COMPUTO	1	900	900,00
IMPRESORA (CANON)	1	400	400,00
IMPRESIONES (COLOR)	80	0,50	40,00
Impresiones (B/N)	110	0,10	11,00
SUMINISTROS OFICINA	9	9	81,00
INTERNET	1	40	40
SERVICIOS BASICOS(INTERNOS)	3	30	60,00
CAPACITACION (CURSOS Y TUTORIAS)	1	200	200,00
MOVILIZACION	80	3,00	240,00
BEATICOS(ALIMENTACIO N)	110	2,50	275,00

DISEÑO,DESARROLLO Y 1 900 900,00

IMPLEMENTACION DEL
SISTEMA

VALOR TOTAL 3147.00

Tabla 53

Recursos de hardware utilizado en el proceso del Sistema

RECURSOS TECNOLÓGICOS		
Cantidad	Equipo	Detalle
1	Equipo de Computo	- Monitor, Teclado, 8 GB RAM - DVD - 1TB Disco Duro - GATEWAY(AMD A4)
1	Impresora(CANON)	- MP 250

Tabla 54

Recursos de software utilizado en el proceso del Sistema

HERRAMIENTAS DE DESARROLLO	
Tipo	Detalle
SISTEMA OPERATIVO	Windows 10
MYSQL XAMPP PHPMYADMIN	Gestor de Base de Datos donde se guardará toda la información.
IBM RATIONAL ROSE	Este Software es primordial para la elaboración de diseños y diagramas UML.
NOTT PAD++	Este Software es empleado para sistematizar el sistema en todas las fases.
POWER DESIGNER	Software indispensable para el diseño y sistematización de la base de datos a Mysql.

SOFTWARE EMPLEADOS	<ul style="list-style-type: none"> - MICROSOFT POWER POINT - MICROSOFT WORD - MICROSOFT EXCEL
--------------------	--

6.02 Presupuesto

Tabla 55

Presupuesto Estimado

Cant	Descripción	P/U	Valor total
1	DESARROLLO DEL APLICATIVO WEB	500,00	500,00
1	EQUIPO DE COMPUTO	1000,00	1000,00
1	IMPRESORA	400,00	400,00
150	IMPRESIONES (COLOR)	0,50	100,00
400	IMPRESIONES (B/N)	0,10	80,00
7	SUMINISTROS OFICINA	10,00	100,00
5	IINTERNET (FIBRA OPTICA)	40	200
3	SERVICIOS BASICOS(IINTERNOS)	20,00	60,00
1	Tutorías	200,00	200
3	CD	0,60	1,80
65	Movilización	3	300
85	Beaticos(Alimentación)	2,50	250
VALOR TOTAL			3191.80

6.03 Cronograma

							viembre	01 enero	01 marzo		01 mayo		01 julio		01 septi		
							21/11	19/12	16/01	13/02	13/03	10/04	08/05	05/06	03/07	31/07	28/08
DIAGRAMA DE GANTT.	1		Poyecto reunión con las autoridades de MUEBLES ELIZABETH	2 días	sáb 08/04/17	dom 09/04/17											
	2		Formulación del Problema	6 días	lun 10/04/17	sáb 15/04/17											
	3		Recopilación de Datos	3 días	lun 17/04/17	mié 19/04/17											
	4		Definición del Tema	4 días	mié 19/04/17	sáb 22/04/17											
	5		Aprobación del Tema	3 días	lun 24/04/17	mié 26/04/17											
	6		Identificación de Objetivos	3 días	jue 27/04/17	sáb 29/04/17											
	7		Análisis de Datos	3 días	lun 01/05/17	mié 03/05/17											
	8		Regla del Negocio	6 días	jue 04/05/17	jue 11/05/17											
	9		Modelamiento de la BD	12 días	mié 10/05/17	jue 25/05/17											
	10		Programación Modulo de Seguridad	11 días	vie 26/05/17	vie 09/06/17											
	11		Programación mantenimientos	14 días	sáb 10/06/17	mié 28/06/17											
	12		Programación regla del negocio	6 días	jue 29/06/17	jue 06/07/17											
	13		Correcciones del	10 días	vie 07/07/17	jue 27/07/17											
	14		Documentación Borrador Final	5 días	vie 21/07/17	jue 27/07/17											
	15		Revisión Urkund del Documento	3 días	vie 28/07/17	mar 01/08/17											
	16		Proceso de Lectura	3 días	lun 18/09/17	mié 20/09/17											
	17		Entrega de Acta 005	2 días	jue 28/09/17	vie 29/09/17											
	18		Entrega de trabajos y cd a consejo de Carrera	3 días	lun 02/10/17	mié 04/10/17											
	19		Sustentación de Grado	2 días	jue 05/10/17	vie 06/10/17											

Figura 37. Cronograma: Con la ayuda del cronograma se realizara cada proceso desde cero.

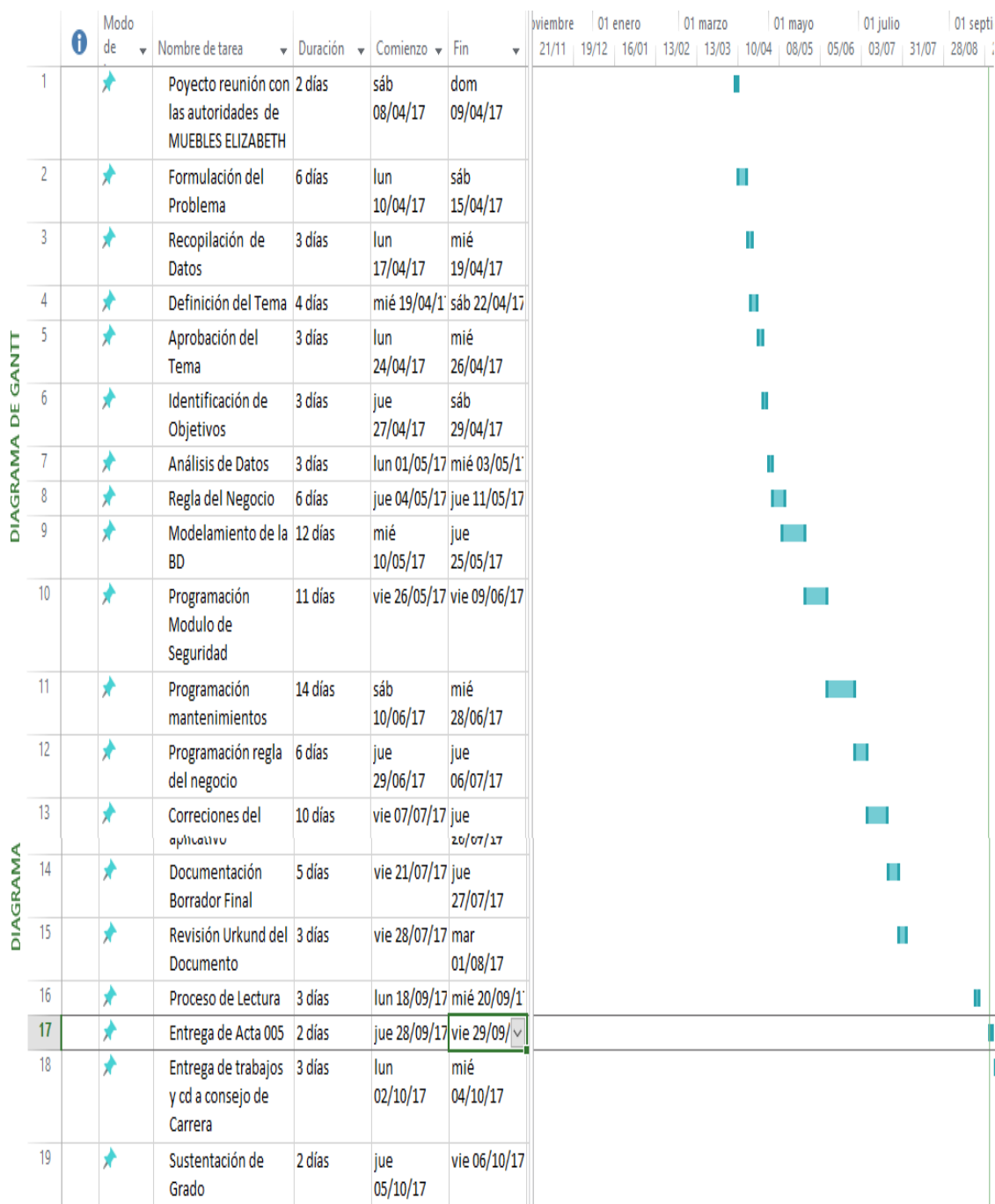


Figura 38. Cronograma: Con la ayuda del cronograma se presenta un esquema donde se visualizara la secuencia de fechas para la realización, desarrollo y ejecución del sistema.

Capítulo VII: Conclusiones y Recomendaciones

7.01 Conclusiones

Se ha realizado un sistema web de inventario ingresos, egresos y ventas de productos de bodega de materiales de Mueblería, para brindar un mejor servicio al cliente al momento de obtener su producto. El aplicativo web ha obtenido excelentes resultados ya que se ha adaptado a las necesidades de la empresa Muebles Elizabeth Quito, Ecuador.

Se ha establecido que el sistema web INVEN MUEBLES ELIZABETH es de vital importancia, ya que en la empresa Muebles Elizabeth todos los procesos de inventario y ventas lo hacían de la manera tradicional en hojas físicas y con la ayuda del sistema ahora tienen un buen control de registro de clientes, proveedores, productos y ventas lo cual es satisfactorio para la empresa.

El sistema ofrece soluciones eficientes a la empresa ya que ahorra tiempo y los costos que se utilizaban, siendo favorable para el cliente y existe un mejor desempeño en la empresa mediante este sistema web.

7.02 Recomendaciones

Dentro de las recomendaciones del sistema al momento de usarlo se va a considerar nombrar las siguientes:

Toda la información debe ser clara y concisa en el sistema para ganar tiempo y agilizar el proceso evitando el uso erróneo del aplicativo.

Realizar mantenimientos continuos al sistema para evitar que existan problemas en su funcionamiento.

Capacitación constante a los usuarios para que puedan utilizar de mejor manera el sistema y no le ocasionen daños o errores al momento que lo utilizan.

Anexos

A.01 Manual de Instalación



INSTITUTO TECNOLÓGICO
"CORDILLERA"

CARRERA DE ANÁLISIS DE SISTEMAS

SISTEMATIZACIÓN DEL PROCESO DE INVENTARIO MEDIANTE UN APLICATIVO WEB PARA LA EMPRESA MUEBLES ELIZABETH UBICADA EN LA CIUDAD DE QUITO

**Proyecto de investigación previo a la obtención del título de Tecnólogo en
Análisis de Sistemas**

Autor: Jefferson Alexander Manzaba Castro

Tutor: Ing. Johnny Patricio Coronel Ordoñez

Quito, Octubre 2017

ÍNDICE GENERAL

Contenido	Página
1 Introducción.....	85
2 Instalación de XAMPP.....	85
3 Instalación de NotePad++.....	89

ÍNDICE DE FIGURAS

Título	Página
<i>Figura 39. Descarga de xampp.....</i>	<i>85</i>
<i>Figura 40. Descomprimir el setup de la instalación de xampp</i>	<i>86</i>
<i>Figura 41. Proceso inicio de instalación xampp</i>	<i>86</i>
<i>Figura 42. Componentes del aplicativo xampp</i>	<i>87</i>
<i>Figura 43. Ubicación instalación de archivos de xampp.....</i>	<i>87</i>
<i>Figura 44. Finalización de instalación del aplicativo xampp.....</i>	<i>88</i>
<i>Figura 45. Conexión a los servidores.....</i>	<i>88</i>
<i>Figura 46. Ingreso al Local host phpmyadmin en el navegador.....</i>	<i>89</i>
<i>Figura 47. Descargamos el aplicativo notepad++ desde el navegador.....</i>	<i>89</i>
<i>Figura 48. Descomprimir archivo que contiene el archivo de instalación notepad++.....</i>	<i>90</i>
<i>Figura 49. Inicio de instalación notepad+.....</i>	<i>90</i>
<i>Figura 50. Componentes para la instalación de notepad++.....</i>	<i>91</i>
<i>Figura 51. Ubicación de instalación archivos y componentes notepad++.....</i>	<i>91</i>
<i>Figura 52. Finalización de instalación notepad++.....</i>	<i>92</i>
<i>Figura 53. Inicio de notepad++.....</i>	<i>92</i>
<i>Figura 54. Inicio de la programación para el desarrollo del software.....</i>	<i>93</i>

1 Introducción

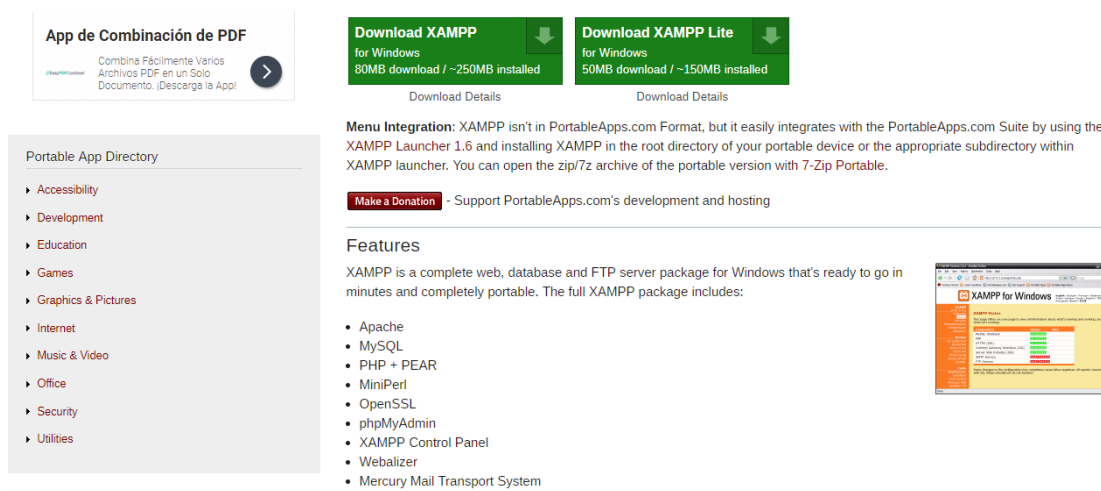
El siguiente manual se ha realizado con el fin de ayudar al usuario mediante una guía de todas las herramientas necesarias para que el software funcione correctamente.

El manual presentara varios procesos que van a favorecer al usuario al momento de implementar las herramientas en cualquier computador y en el cual deberá seguir todo el proceso de instalación para el buen cumplimiento de todo el desarrollo del aplicativo web.

2 Instalación del XAMPP

Primero se procederá a descargarlo desde el navegador con todos los paquetes de instalación.

<https://portableapps.com/apps/development/xampp>



The screenshot shows the XAMPP download page on PortableApps.com. At the top, there are two green buttons: "Download XAMPP for Windows" (80MB download / ~250MB installed) and "Download XAMPP Lite for Windows" (50MB download / ~150MB installed). Below these are "Download Details" links. A "Menu Integration" section explains how to integrate XAMPP with the PortableApps.com Suite using XAMPP Launcher 1.6. A "Make a Donation" button is also present. The "Features" section lists the components included in the XAMPP package: Apache, MySQL, PHP + PEAR, MiniPerl, OpenSSL, phpMyAdmin, XAMPP Control Panel, Webalizer, and Mercury Mail Transport System. On the left, there is a "Portable App Directory" with various categories like Accessibility, Development, Education, Games, Graphics & Pictures, Internet, Music & Video, Office, Security, and Utilities. On the right, there is a small image of the XAMPP for Windows installer window.

Figura 39. Descarga de XAMPP

Después de descargar buscamos el archivo para descomprimir la carpeta que contiene el ejecutable para la instalación.

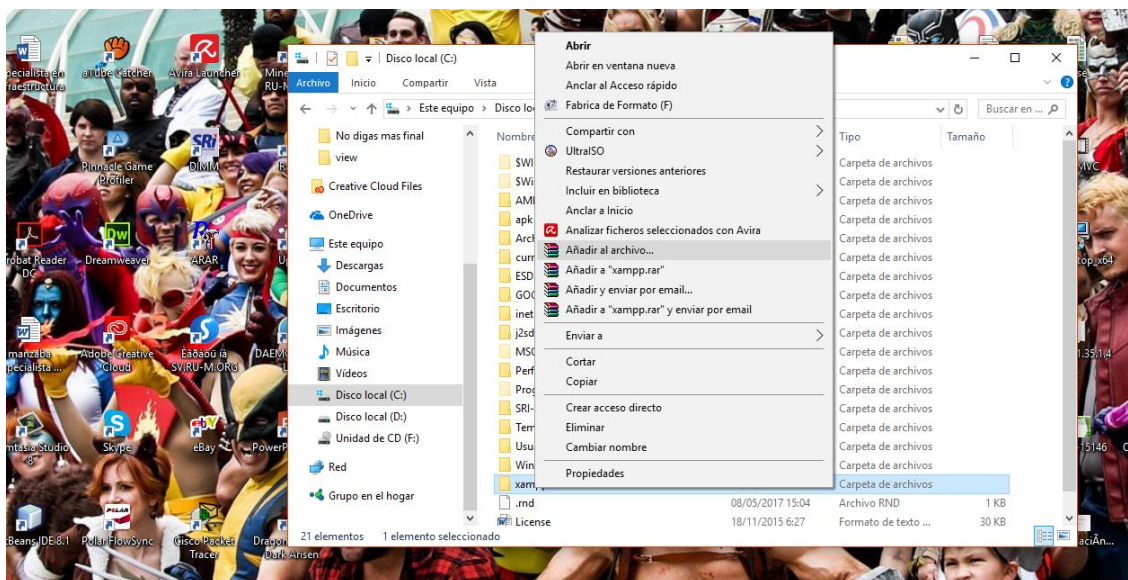


Figura 40. Descomprimir el setup de la instalación de xampp.

Dar clic en continuar para comenzar con la instalación del aplicativo.

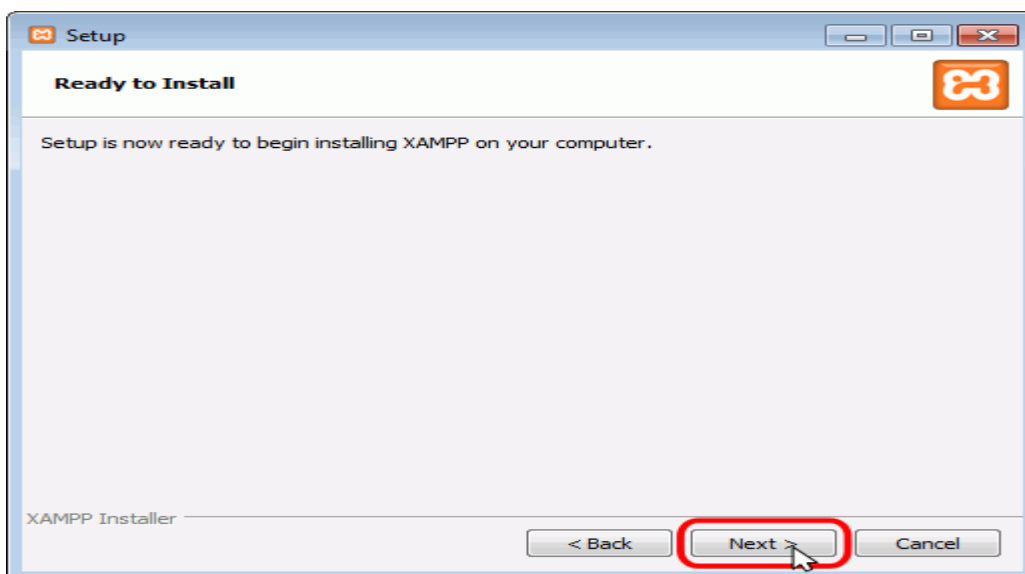


Figura 41. Proceso inicio de instalación xampp.

Seleccionamos los componentes requeridos para la instalación.

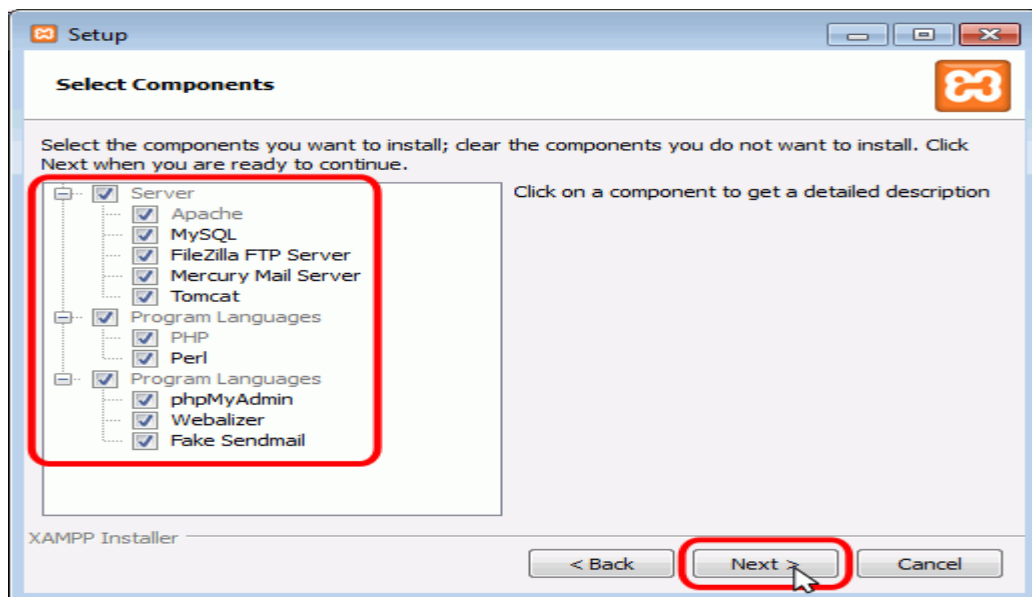


Figura 42. Componentes del aplicativo xampp.

Escogemos la ubicación de instalación de los archivos.

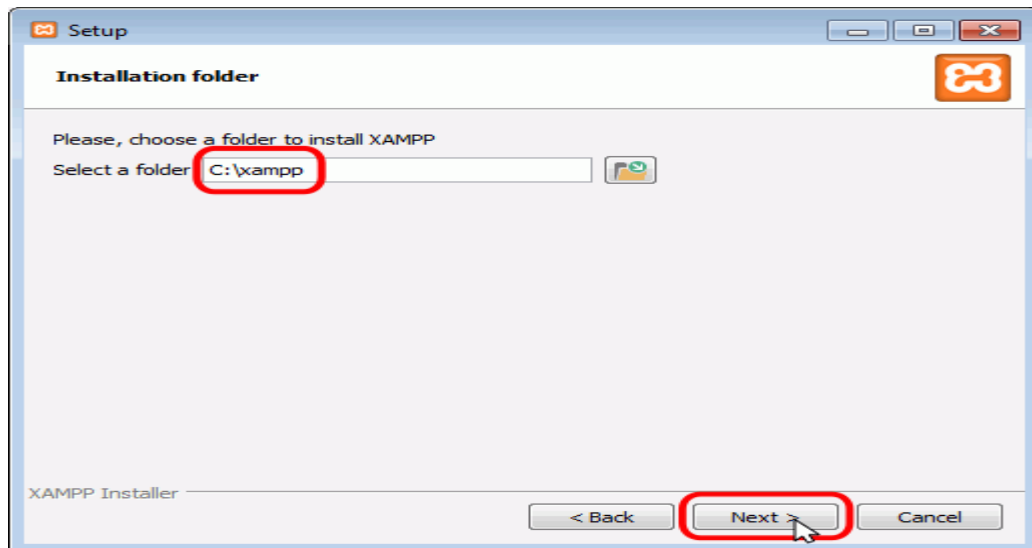


Figura 43. Ubicación instalación de archivos de xampp.

Finalización de instalación del aplicativo seleccionamos los accesos directos en el equipo.



Figura 44. Finalización de instalación del aplicativo xampp.

Pantalla principal donde activaremos la conexión al servidor Mysql y Apache clic en start para activarlos.

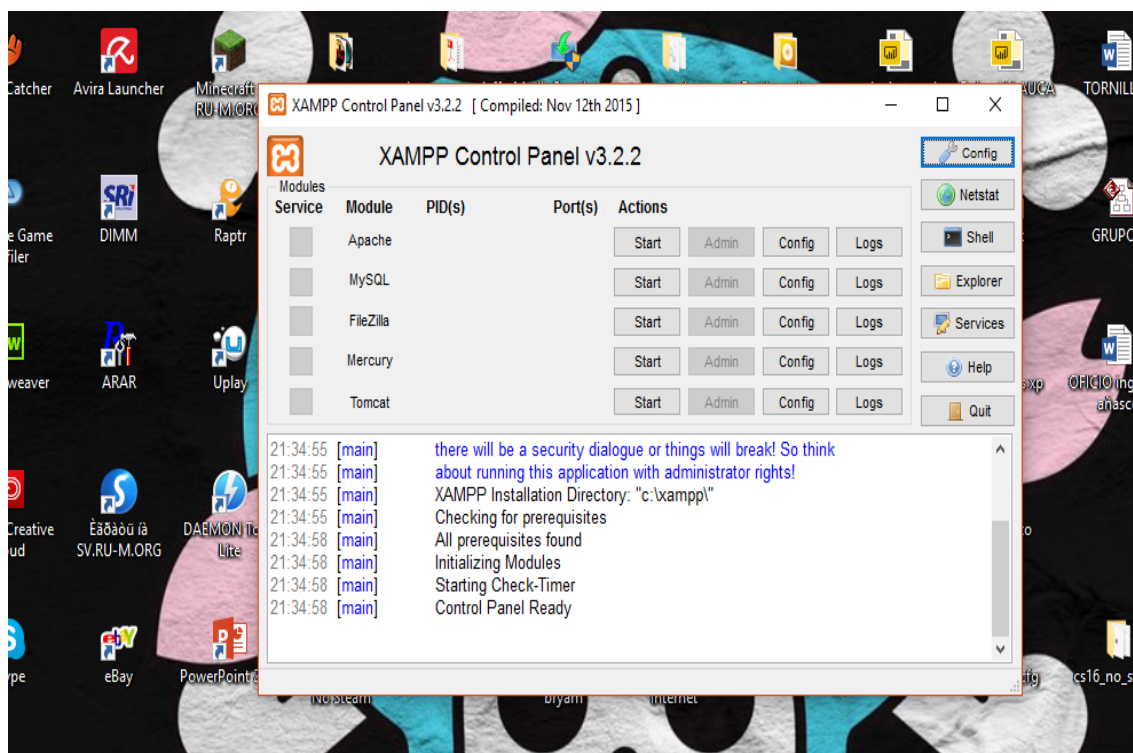


Figura 45. Conexión a los servidores.

En el navegador colocamos el <http://localhost:8080/phpmyadmin> ingresamos escogemos Base de Datos y comenzaremos a crearla si ya tenemos creada una base de datos la mandaremos a llamar en importar y se cargaran todas las tablas.

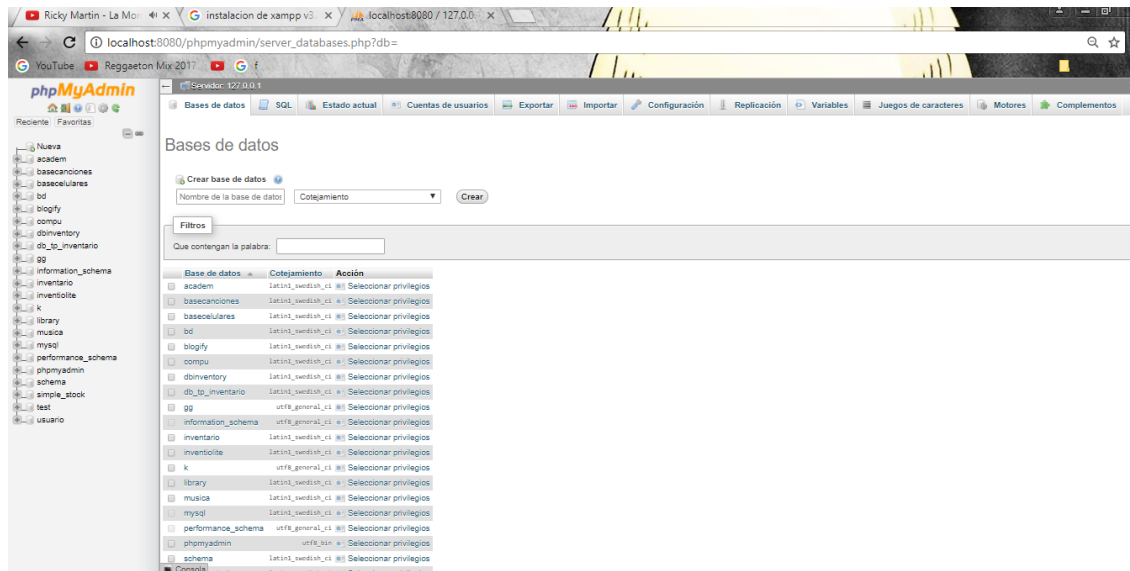


Figura 46. Ingreso al Local host phpmyadmin en el navegador para creación o importación de la base de datos.

3 Instalación de NotePad++

Descarga del aplicativo NotePad++ en el navegador <https://notepad-plus-plus.org/download/v5.9.8.html>.

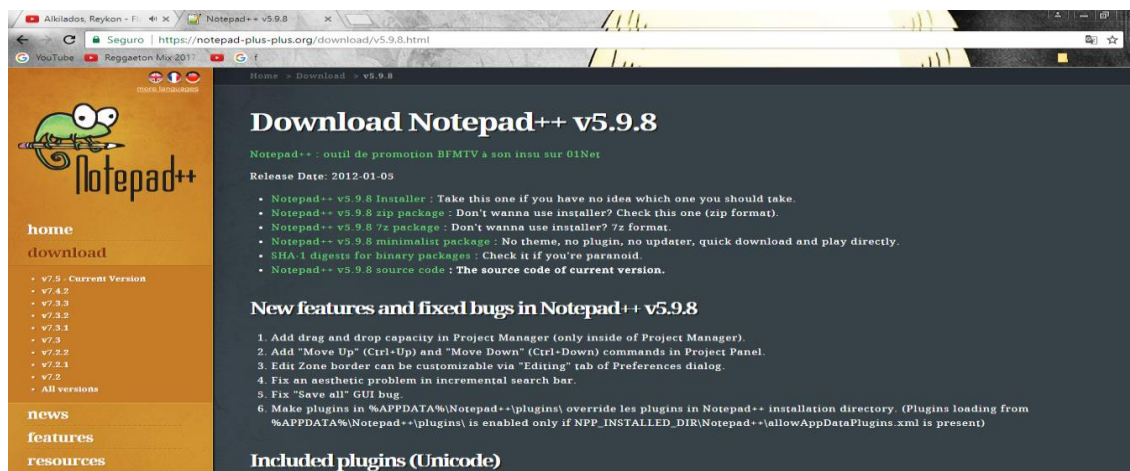


Figura 47. Descargamos el aplicativo notepad++ desde el navegador.

Descomprimos el archivo para obtener la carpeta de instalación.



Figura 48. Descomprimir archivo que contiene el archivo de instalación notepad++.

Seguir todos los pasos para el proceso de instalación.

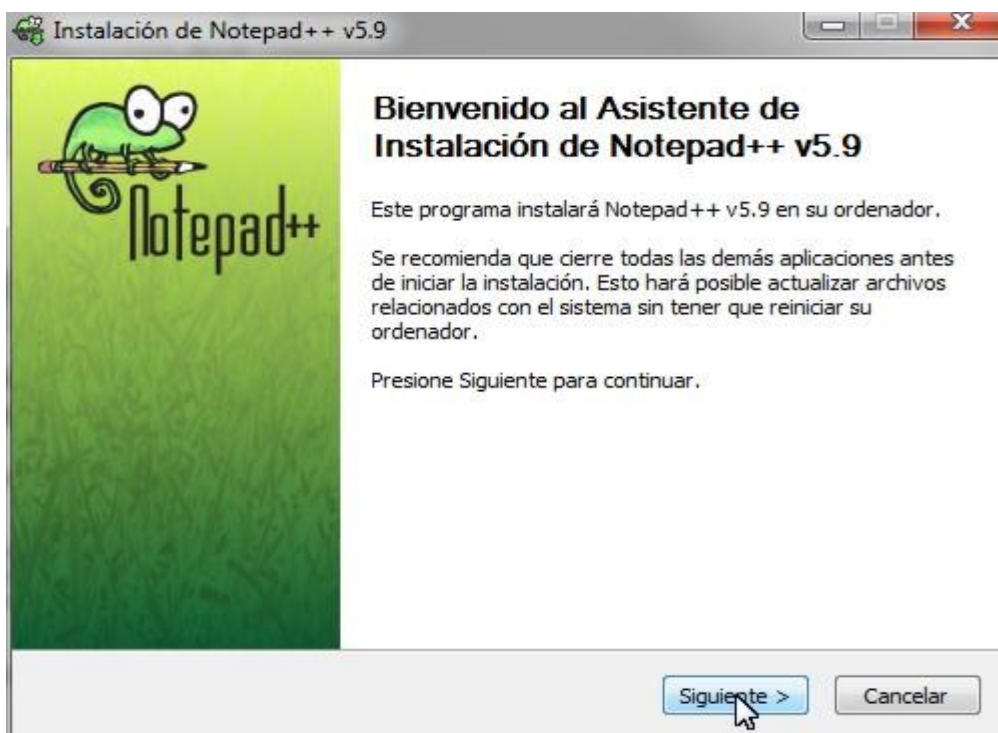


Figura 49. Inicio de instalación notepad++.

Seleccionar los componentes para seguir con la instalación.

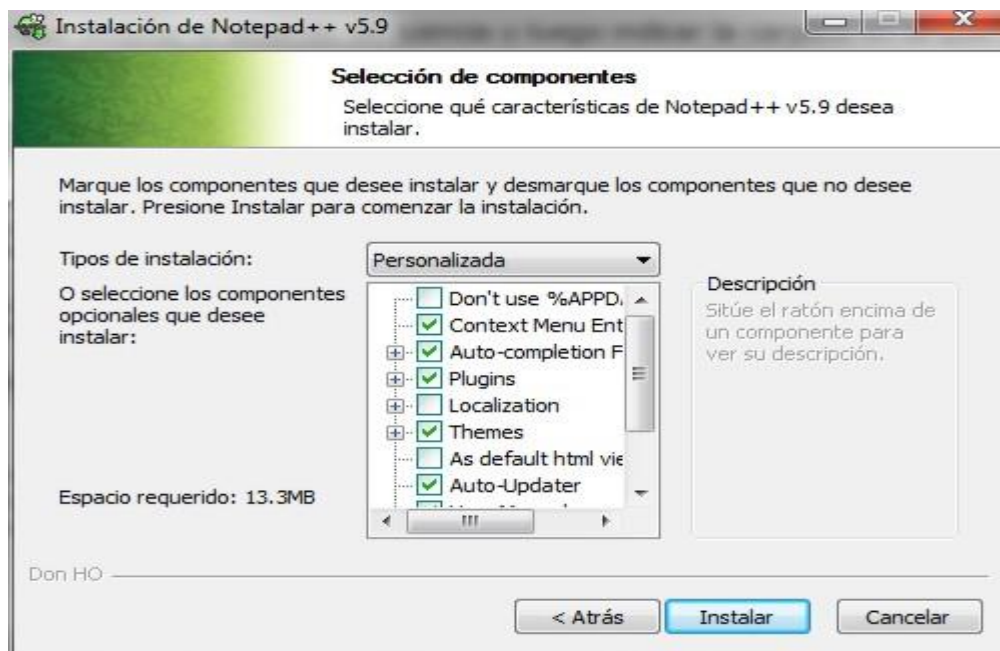


Figura 50. Componentes para la instalación de notepad++.

Escoger la ubicación de instalación de archivos y componentes.

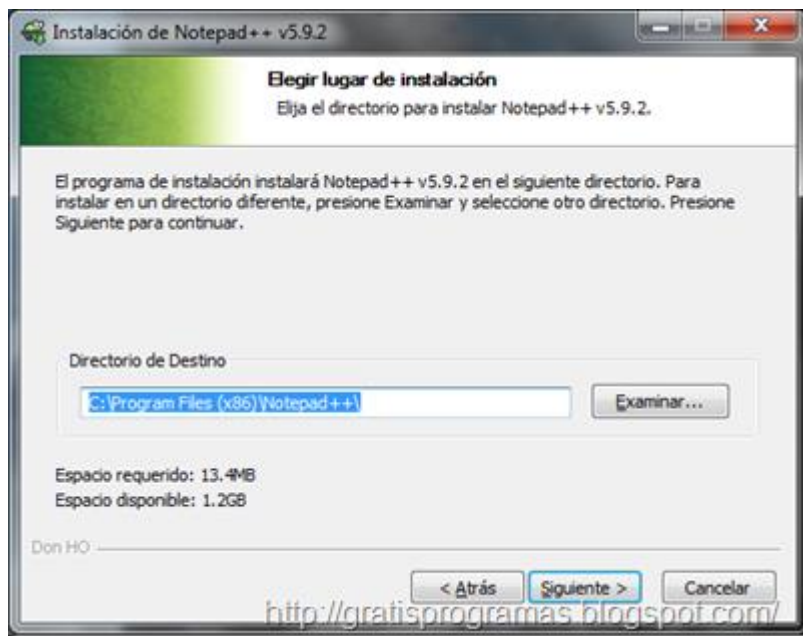


Figura 51. Ubicación de instalación archivos y componentes de notepad++.

Finalización de instalación de notepad++ escogemos los accesos directos en el equipo.

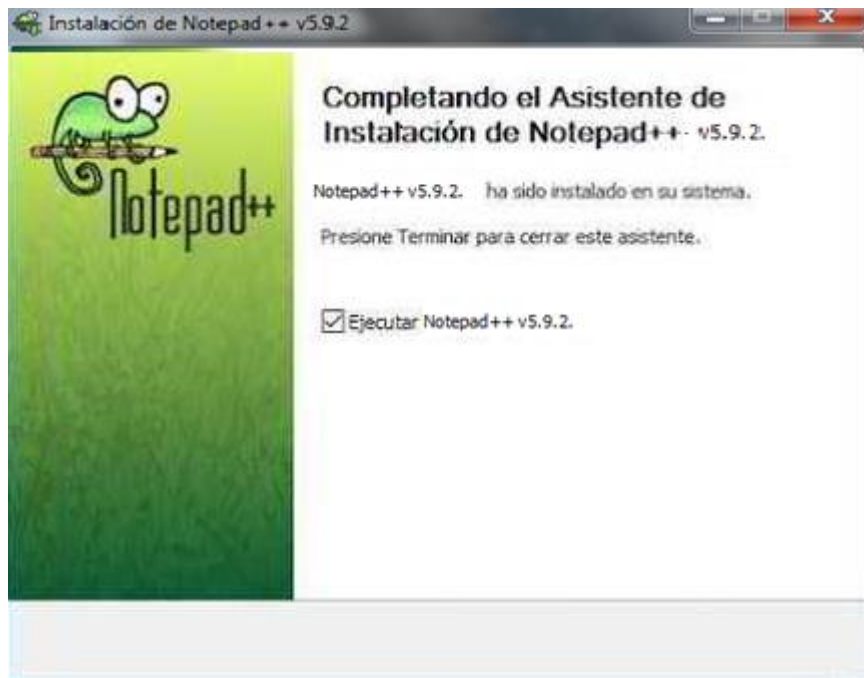


Figura 52. Finalización de instalación notepad++.

Pantalla de inicio de notepad++.

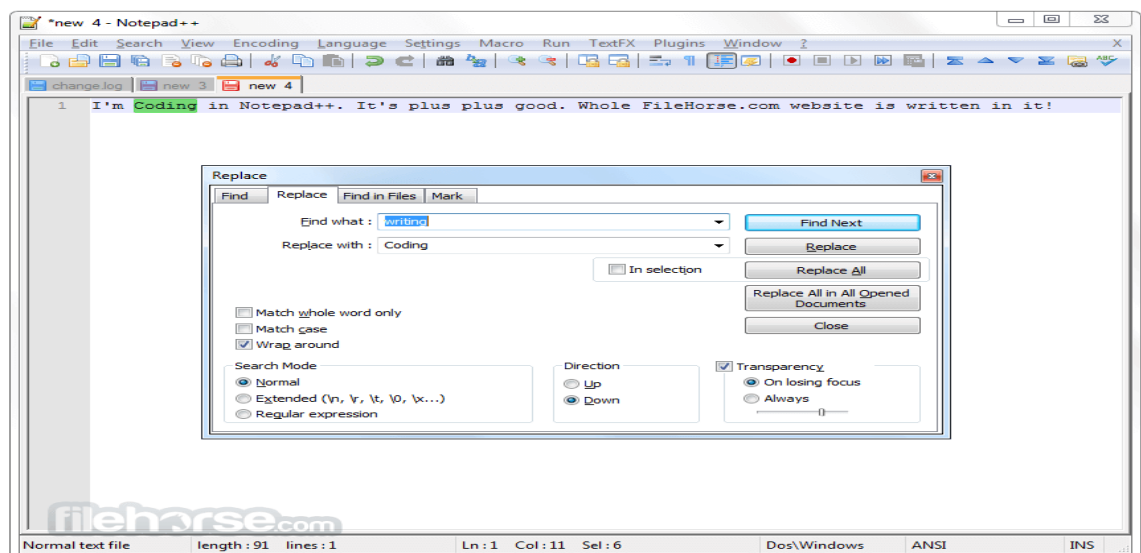
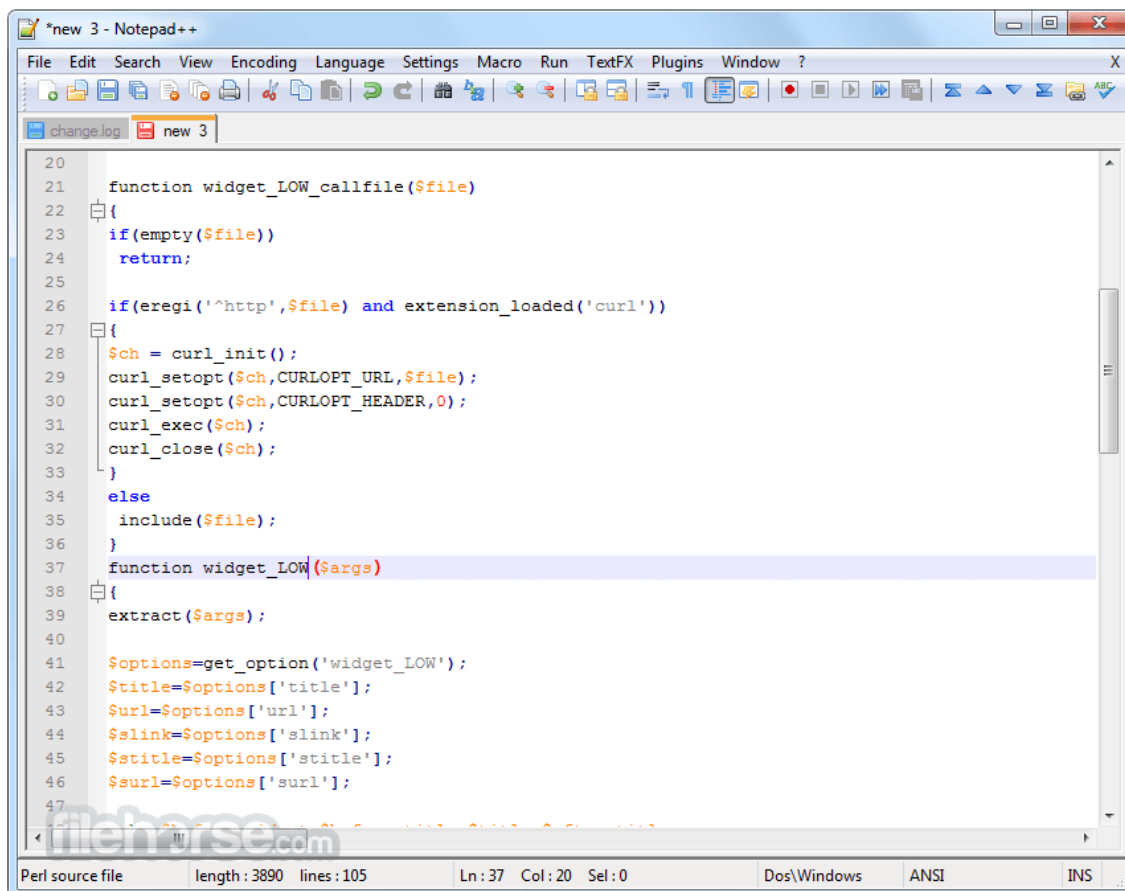


Figura 53. Inicio de notepad++.

Comenzamos a programar para desarrollar el aplicativo web.



```
20
21 function widget_LOW_callfile($file)
22 {
23     if(empty($file))
24         return;
25
26     if(eregi('^http',$file) and extension_loaded('curl'))
27     {
28         $ch = curl_init();
29         curl_setopt($ch,CURLOPT_URL,$file);
30         curl_setopt($ch,CURLOPT_HEADER,0);
31         curl_exec($ch);
32         curl_close($ch);
33     }
34     else
35         include($file);
36 }
37 function widget_LOW($args)
38 {
39     extract($args);
40
41     $options=get_option('widget_LOW');
42     $title=$options['title'];
43     $url=$options['url'];
44     $slink=$options['slink'];
45     $stitle=$options['stitle'];
46     $surl=$options['surl'];
47
48     filehorse.com
```

Perl source file length : 3890 lines : 105 Ln : 37 Col : 20 Sel : 0 Dos\Windows ANSI INS

Figura 54. Inicio de la programación para el desarrollo del software.

A.02 Manual Usuario



INSTITUTO TECNOLÓGICO
"CORDILLERA"

CARRERA DE ANÁLISIS DE SISTEMAS

SISTEMATIZACIÓN DEL PROCESO DE INVENTARIO MEDIANTE UN APLICATIVO WEB PARA LA EMPRESA MUEBLES ELIZABETH UBICADA EN LA CIUDAD DE QUITO

**Proyecto de investigación previo a la obtención del título de Tecnólogo en
Análisis de Sistemas**

Autor: Jefferson Alexander Manzaba Castro

Tutor: Ing. Johnny Patricio Coronel Ordoñez

Quito, Octubre 2017

ÍNDICE GENERAL

Contenido	Página
1 Introducción	99
2 Contenido.....	99
2.01 Presentación del Aplicativo.....	99
2.02 Menú Login	100
2.03 Menú Registro de Personal	100
2.04 Menú Registro de Cliente.....	101
2.05 Menú Registro de Proveedores	102
2.06 Menú Registro de Categoría de Producto.....	103
2.07 Menú Registro de Producto	104
2.08 Menú Registro de Venta.....	105
2.09 Menú Lista de Venta.....	105
2.10 Menú de Caja	106
2.11 Menú Inventario de Productos	106
2.12 Menú Registro de Reabastecimiento de Inventario	107
2.13 Menú Vista de Reabastecimiento	107
2.14 Menú Reporte de Inventario.....	108
2.15 Menú Reporte de Ventas	108

ÍNDICE DE FIGURAS

Título	Página
<i>Figura 55. Inicio del sistema</i>	<i>99</i>
<i>Figura 56. Login</i>	<i>100</i>
<i>Figura 57. Registro Personal.....</i>	<i>100</i>
<i>Figura 58. Registro Nuevo Cliente.....</i>	<i>101</i>
<i>Figura 59. Registro Cliente editar y eliminar información</i>	<i>101</i>
<i>Figura 60. Registro Nuevo Proveedor</i>	<i>102</i>
<i>Figura 61. Registro Proveedores editar y eliminar información</i>	<i>102</i>
<i>Figura 62. Registro Nueva Categoría</i>	<i>103</i>
<i>Figura 63. Registro Categorías de productos editar y eliminar información</i>	<i>103</i>
<i>Figura 64. Registro Nuevo Producto</i>	<i>104</i>
<i>Figura 65. Registro Producto editar y eliminar información</i>	<i>104</i>
<i>Figura 66. Registro Venta</i>	<i>105</i>
<i>Figura 67. Lista de Ventas.....</i>	<i>105</i>
<i>Figura 68. Caja.....</i>	<i>106</i>
<i>Figura 69. Inventario de Productos.....</i>	<i>106</i>
<i>Figura 70. Reabastecimientos</i>	<i>107</i>

Página

<i>Figura 71. Vista de Reabastecimientos</i>	<i>107</i>
<i>Figura 72. Vista Reporte Inventario</i>	<i>108</i>
<i>Figura 73. Vista Reporte Ventas.....</i>	<i>108</i>

1 Introducción

El siguiente manual servirá a los diferentes usuarios que utilicen el aplicativo contara con una guía de cómo utilizar el sistema sin ningún problema para que no halla errores en su funcionalidad.

2 Contenido

2.01 Presentación del aplicativo

Al momento que ingresemos en nuestro navegador web con esta dirección <http://localhost:8080/inventario/index.php?view=home> nos mostrara el aplicativo web.

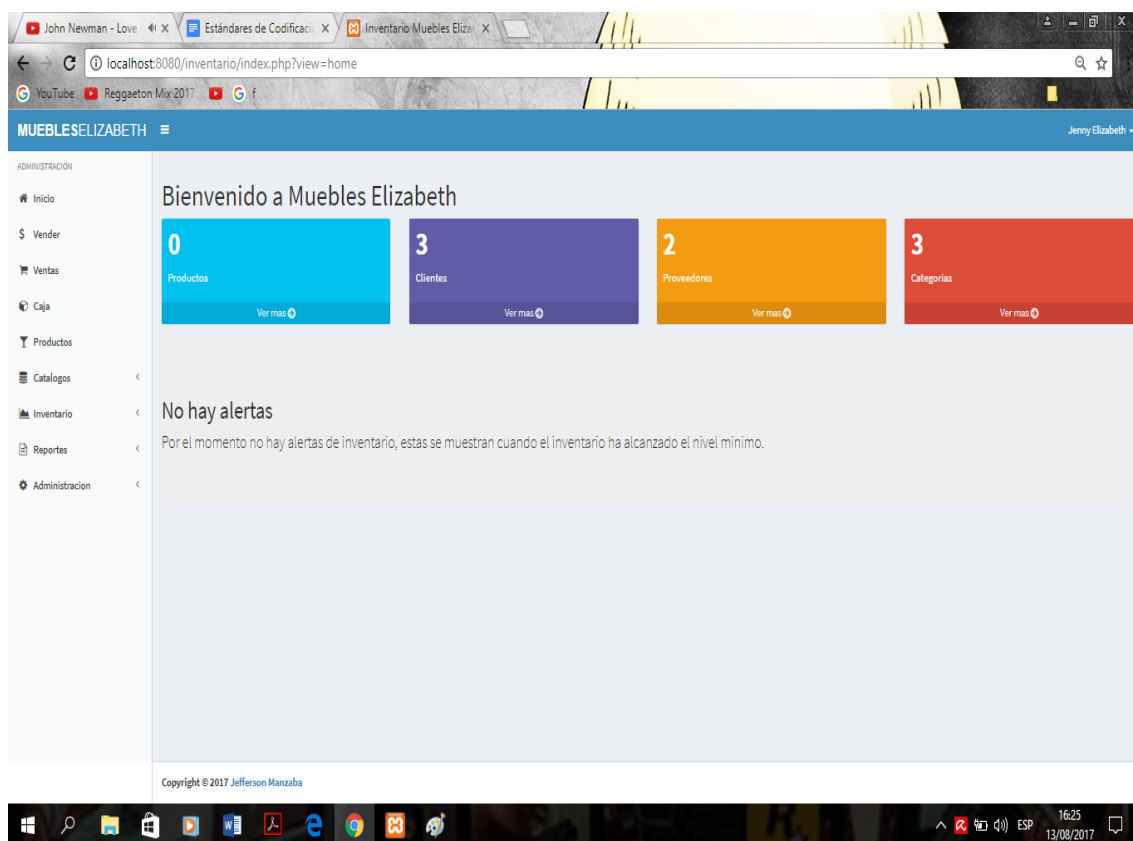


Figura 55. Inicio del sistema

2.02 Menú Login

Aquí se presenta el ingreso al sistema con el correspondiente Usuario y Clave si no se ingresa correctamente la información no podrá ingresar al sistema.

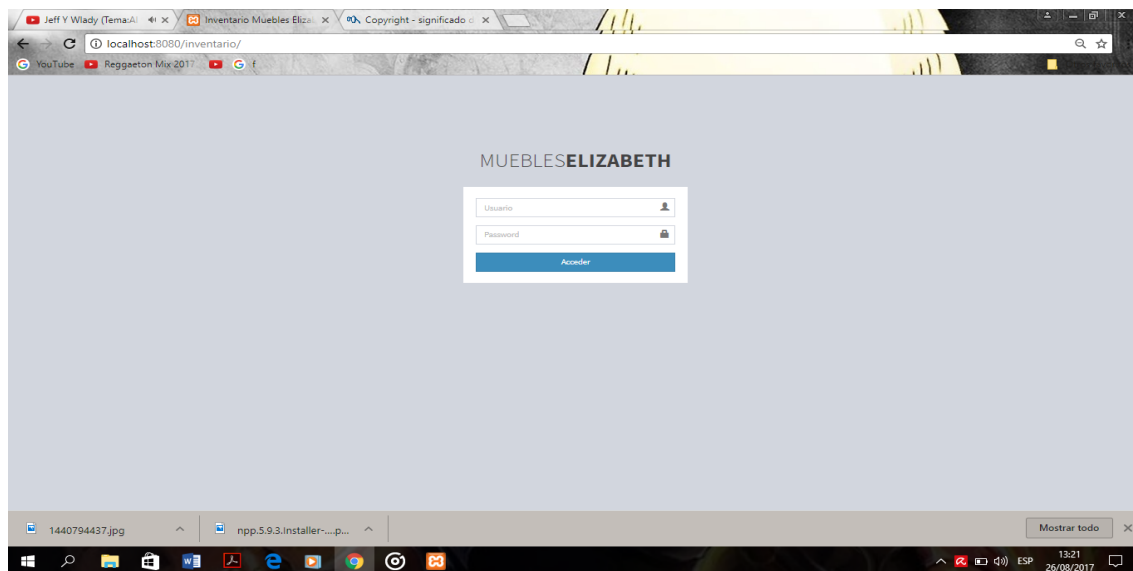


Figura 56. Login

2.03 Menú Registro de Personal

Se presenta el ingreso de usuarios o personal que utilizara el sistema el Administrador será quien ingrese la información del personal y les de su usuario y clave.

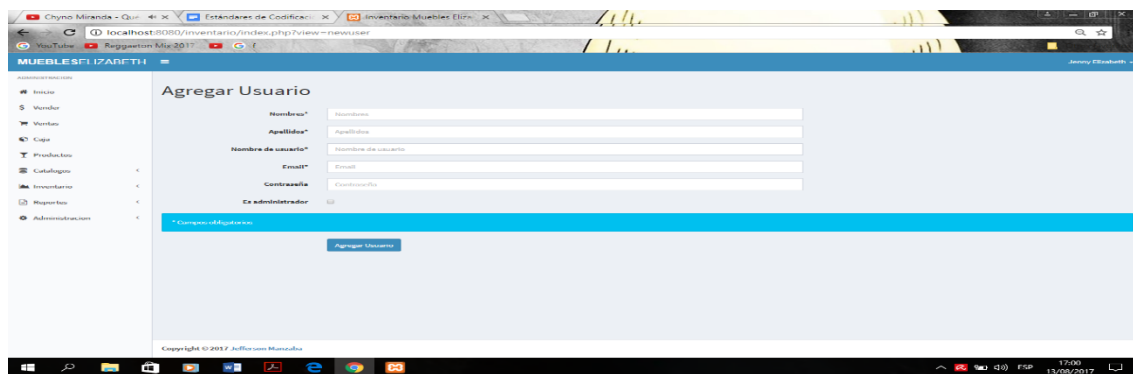


Figura 57. Registro Personal

2.04 Menú Registro de Cliente

Se muestra el ingreso de información de clientes.

MUEBLESELIZABETH

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Nuevo Cliente

Nombres*

Apellidos*

Direccion*

Email*

Telefono*

* Campos obligatorios

[Agregar Cliente](#)

Copyright © 2017 Jefferson Manzaba

Figura 58. Registro Nuevo Cliente.

MUEBLESELIZABETH

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Directorio de Clientes

[Nuevo Cliente](#) [Descargar](#)

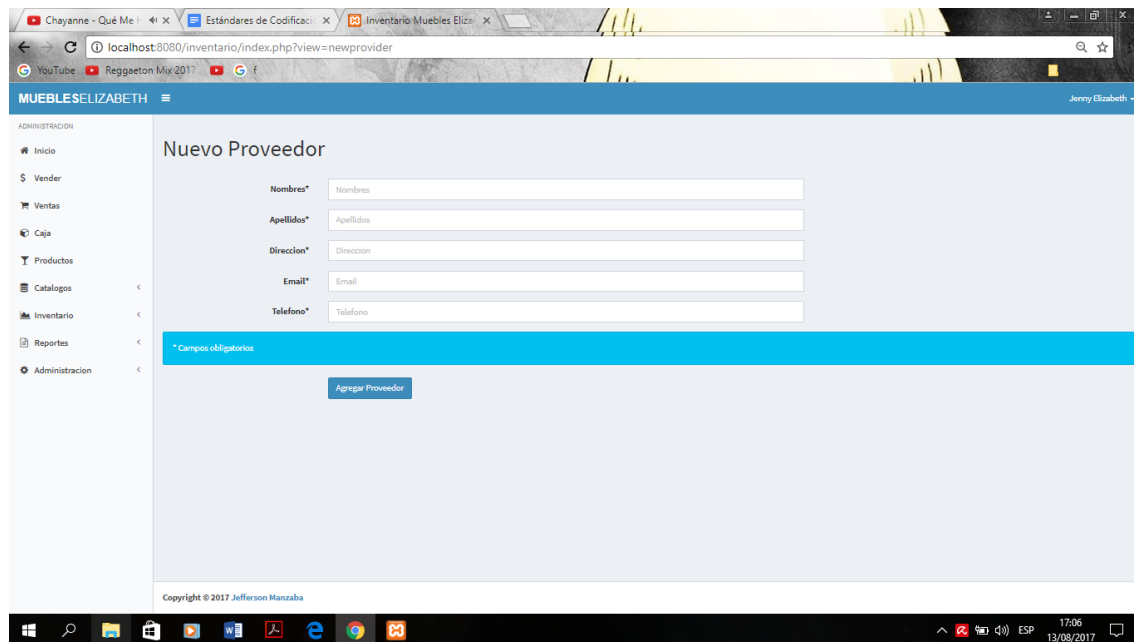
Copyright © 2017 Jefferson Manzaba

Nombre completo	Direccion	Email	Telefono	
EDISON MOLINA	SAN BLAS	EDISON@POPUSI.COM	521432145	Editar Eliminar
Johnny Coronel	La Prensa	jony@gmail.com	3258894	Editar Eliminar
Laura Lorena Alban Torres	San Martin	lore@gmail.com	3259687	Editar Eliminar

Figura 59. Registro Cliente editar y eliminar información.

2.05 Menú Registro de Proveedores

Aquí se mostrara el ingreso de la información de los proveedores.



MUEBLESELIZABETH

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Nuevo Proveedor

Nombres*

Apellidos*

Direccion*

Email*

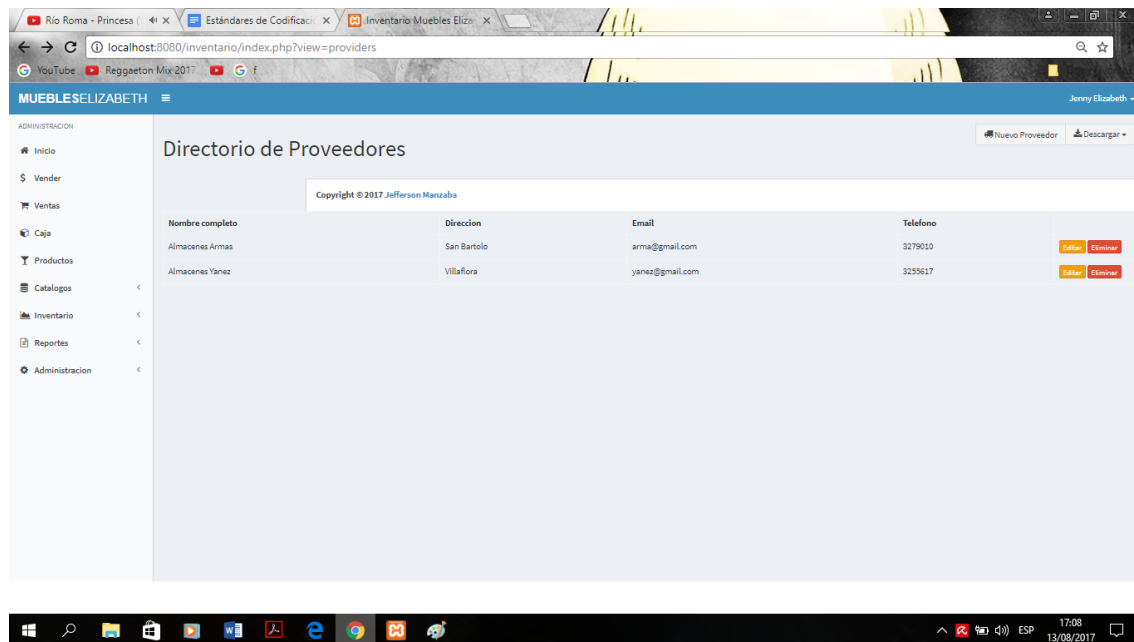
Telefono*

* Campos obligatorios

[Agregar Proveedor](#)

Copyright © 2017 Jefferson Manzaba

Figura 60. Registro Nuevo Proveedor.



MUEBLESELIZABETH

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Directorio de Proveedores

[Nuevo Proveedor](#) [Descargar](#)

Copyright © 2017 Jefferson Manzaba

Nombre completo	Direccion	Email	Telefono	
Almacenes Armas	San Bartolo	arma@gmail.com	3279010	Editar Eliminar
Almacenes Yanez	Villafiora	yanez@gmail.com	3255617	Editar Eliminar

Figura 61. Registro Proveedores editar y eliminar información.

2.06 Menú Registro de Categoría de Productos

Mostrará el ingreso de todas las categorías de productos que existirán dentro del sistema.

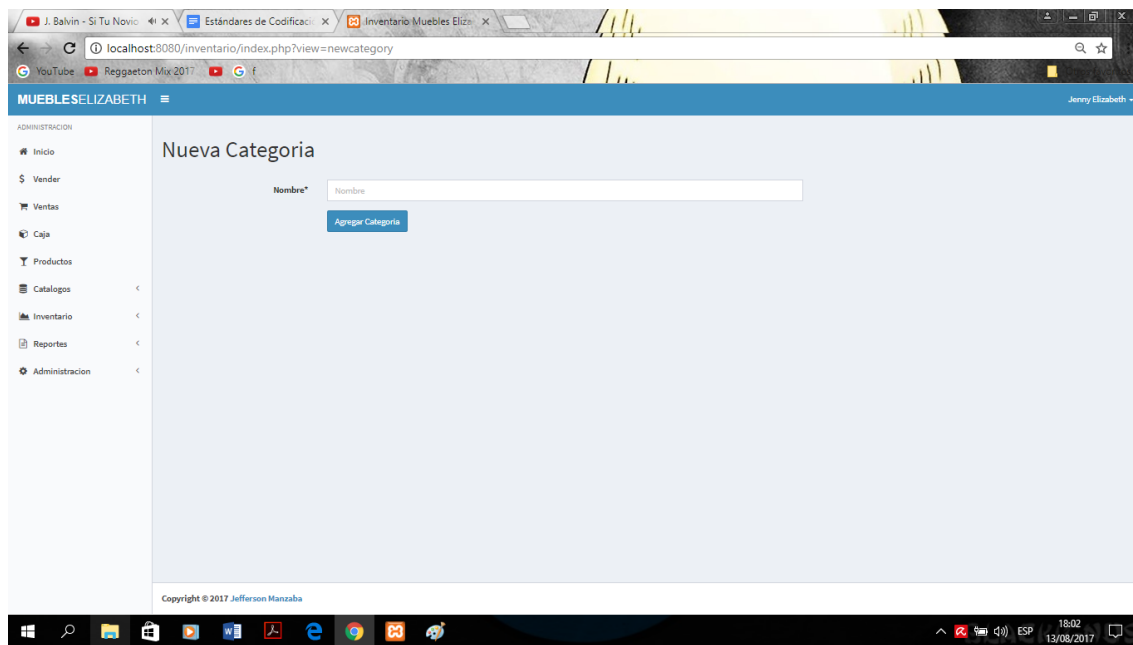


Figura 62. Registro Nueva Categoría.

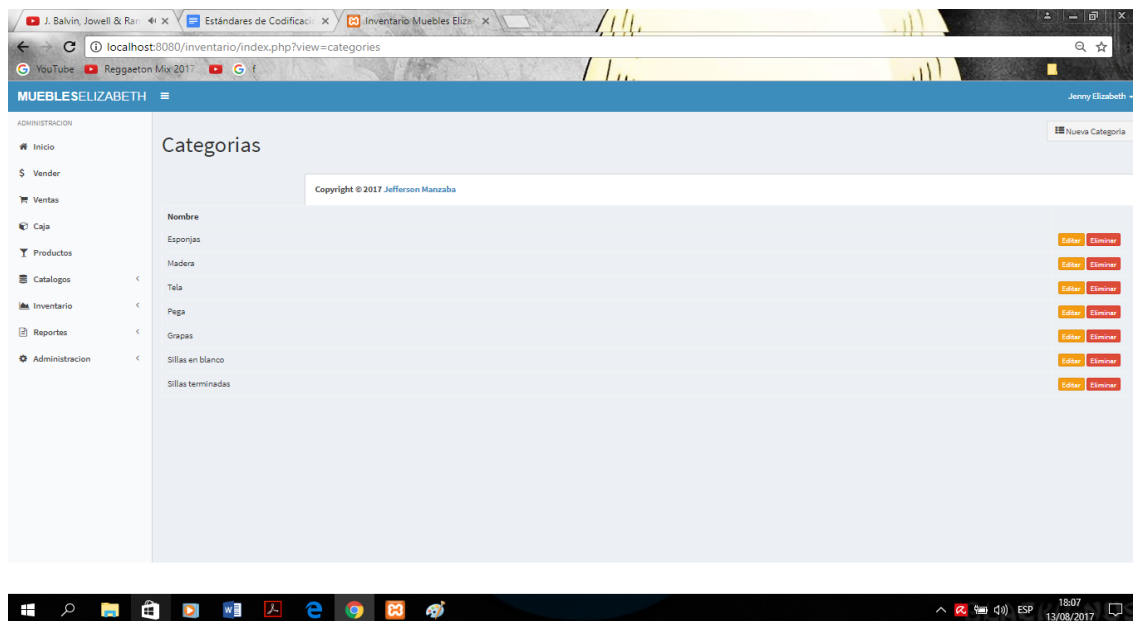
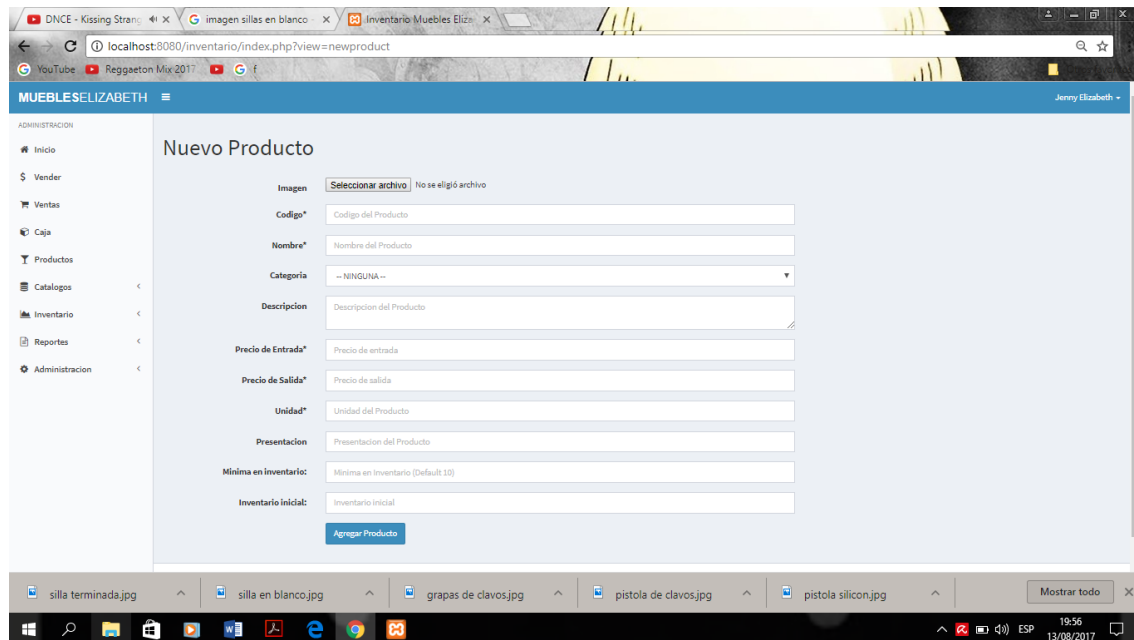


Figura 63. Registro Categorías de Productos editar y eliminar información.

2.07 Menú Registro de Productos

Se muestra el ingreso de nuevos productos toda la información requerida.



MUEBLESELIZABETH

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Nuevo Producto

Imagen: [Seleccionar archivo](#) No se eligió archivo

Código*:

Nombre*:

Categoría:

Descripción:

Precio de Entrada*:

Precio de Salida*:

Unidad*:

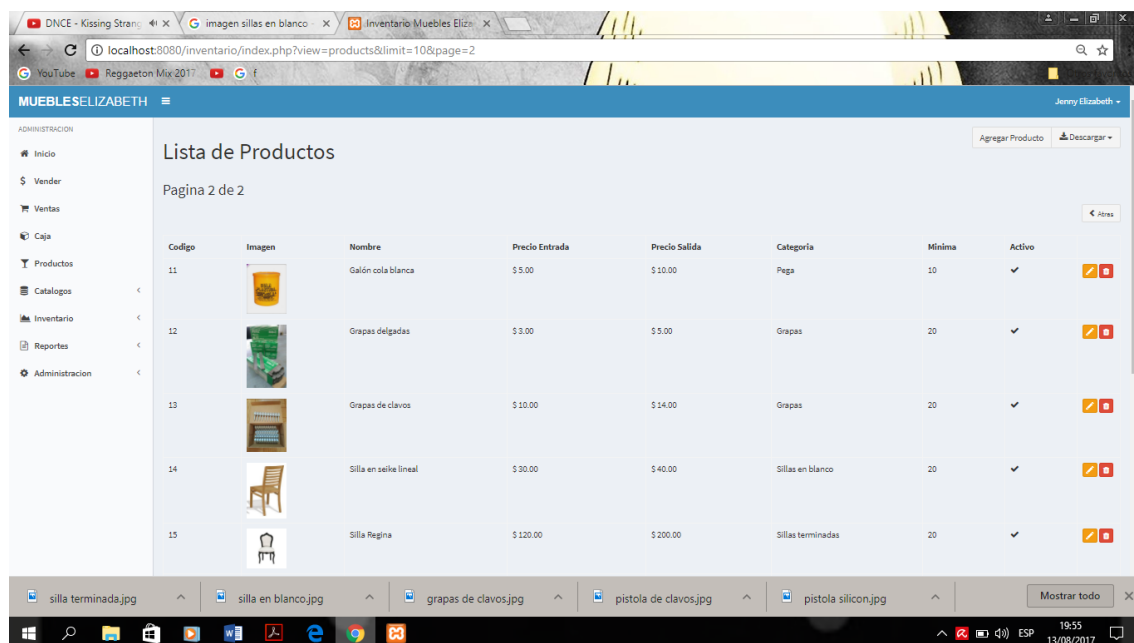
Presentación:

Mínima en Inventario:

Inventario Inicial:

[Agregar Producto](#)

Figura 64. Registro Nuevo Producto.



MUEBLESELIZABETH

ADMINISTRACION

- Inicio
- Vender
- Ventas
- Caja
- Productos
- Catalogos
- Inventario
- Reportes
- Administracion

Lista de Productos

Página 2 de 2

[Agregar Producto](#) [Descargar](#)






Código	Imagen	Nombre	Precio Entrada	Precio Salida	Categoría	Mínima	Activo
11		Gallón cola blanca	\$ 5.00	\$ 10.00	Pega	10	✓
12		Grapas delgadas	\$ 3.00	\$ 5.00	Grapas	20	✓
13		Grapas de clavos	\$ 10.00	\$ 14.00	Grapas	20	✓
14		Silla en sille lineal	\$ 30.00	\$ 40.00	Sillas en blanco	20	✓
15		Silla Regina	\$ 120.00	\$ 200.00	Sillas terminadas	20	✓

Figura 65. Registro Productos editar y eliminar información.

2.08 Menú de Venta

Después de haber ingresado toda la información requerida por el sistema como personal, clientes, proveedores y productos el aplicativo cuenta con toda la información para poder realizar la venta sin problemas.

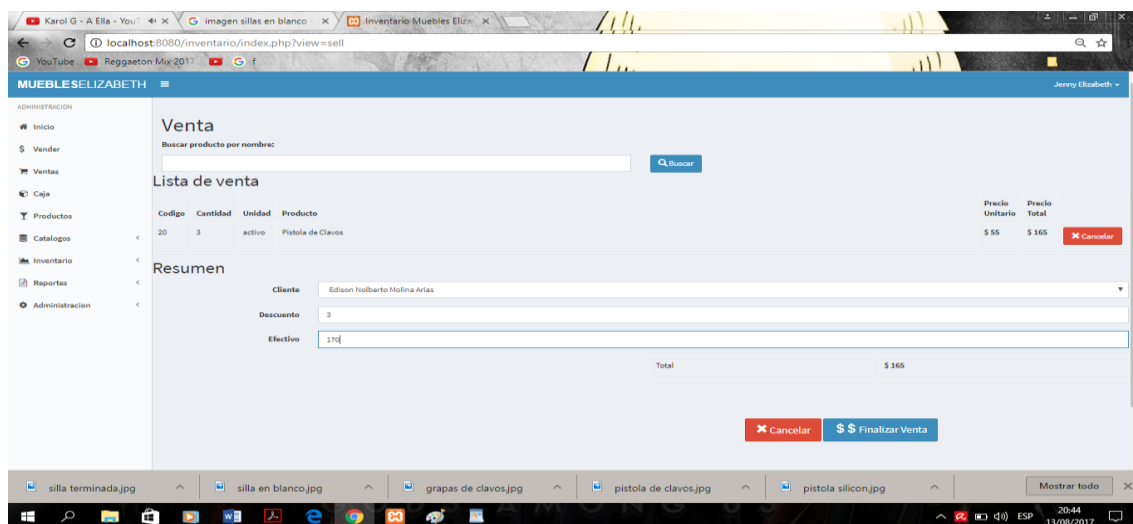


Figura 66. Registro Venta.

2.09 Menú Lista de Ventas

Aquí se mostrara la lista de ventas realizadas en el día.

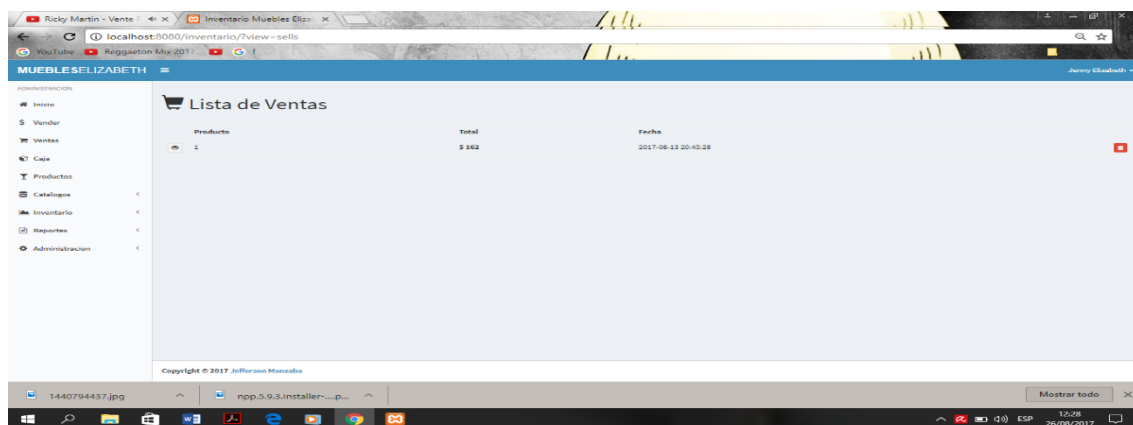


Figura 67. Lista de Ventas.

2.10 Menú de Caja

Se puede visualizar cuánto dinero hay en caja por ventas realizadas.

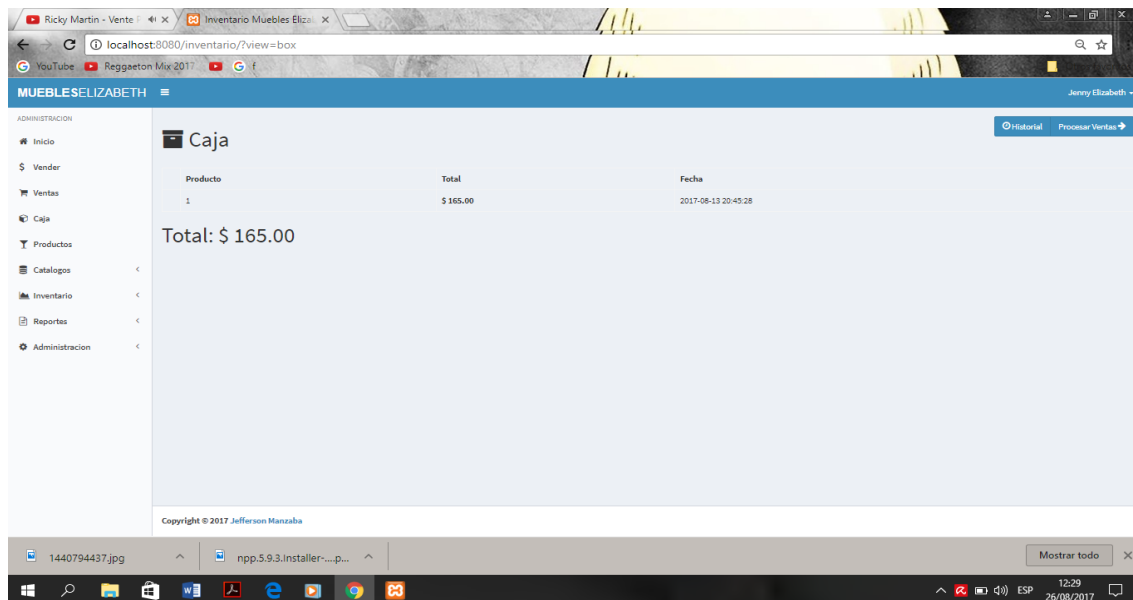


Figura 68. Caja.

2.11 Menú Inventario de Productos

Se puede visualizar el inventario de los productos.

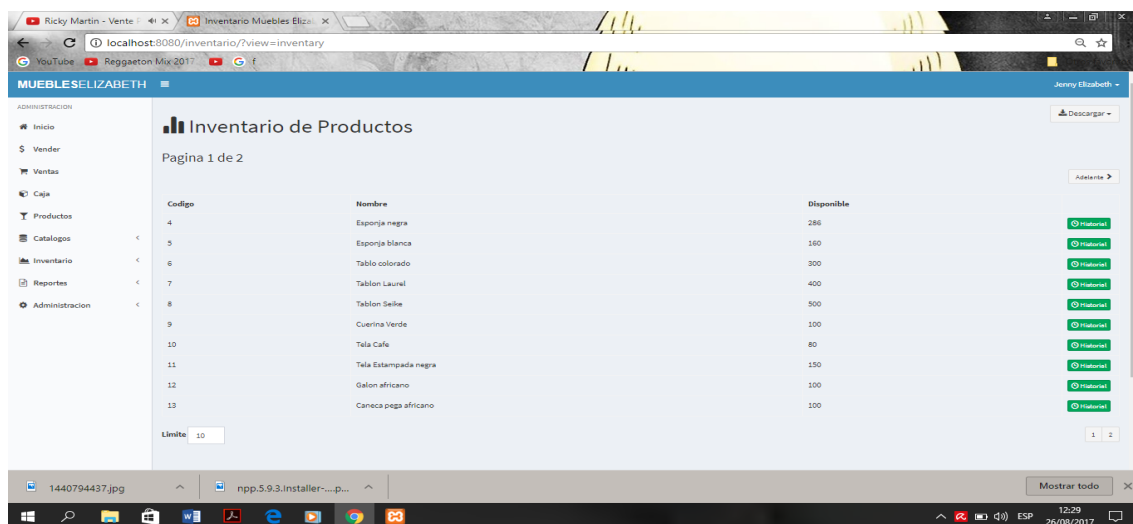


Figura 69. Inventario de productos.

2.12 Menú Reabastecimiento de Inventario

Muestra el ingreso de reabastecimiento de productos al inventario cuando se encuentra con pocos productos.

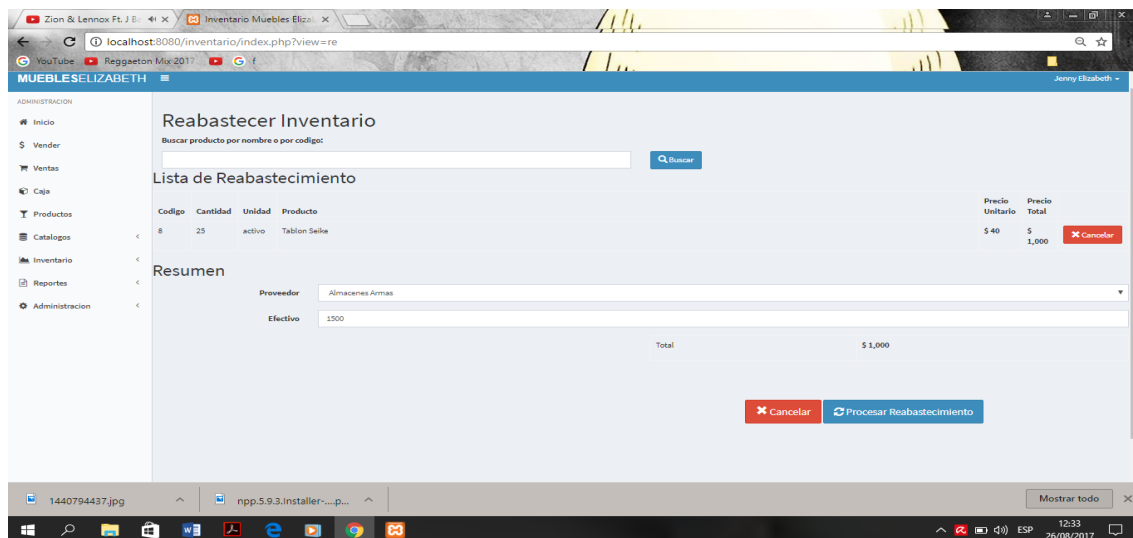


Figura 70. Reabastecimientos.

2.13 Menú de Vista de Reabastecimiento

Después de reabastecer el inventario se puede visualizar el listado de la información ingresada.

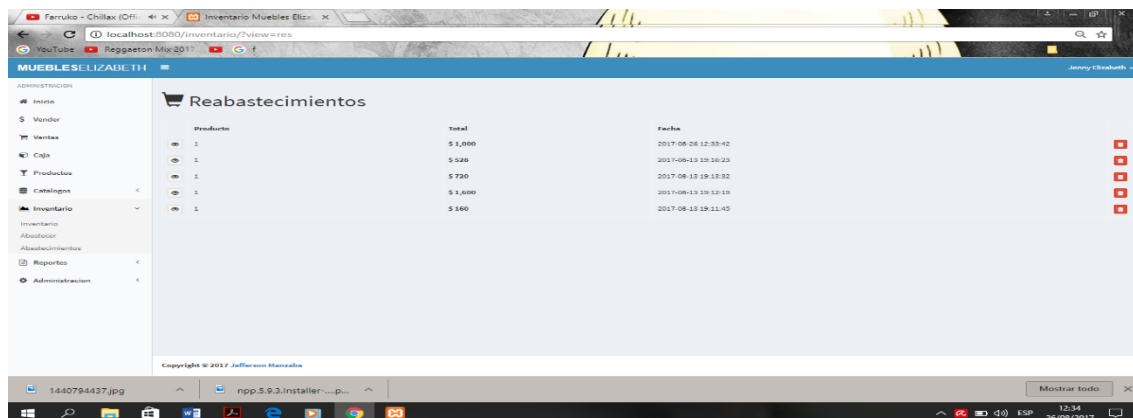
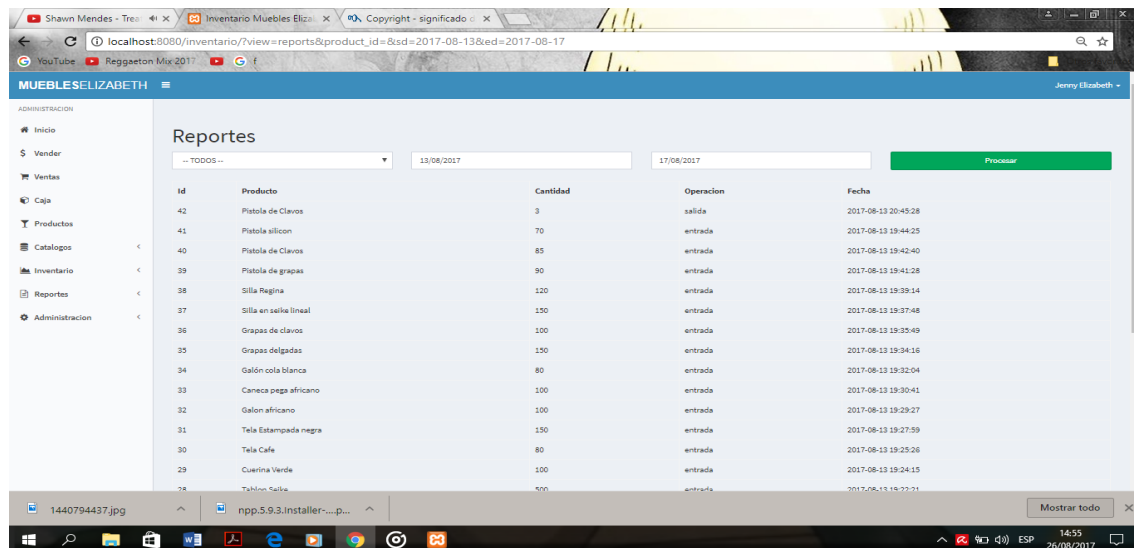


Figura 71. Vista Reabastecimientos.

2.14 Menú Reporte Inventario

Permitirá visualizar todo el reporte de productos ingresados al inventario.

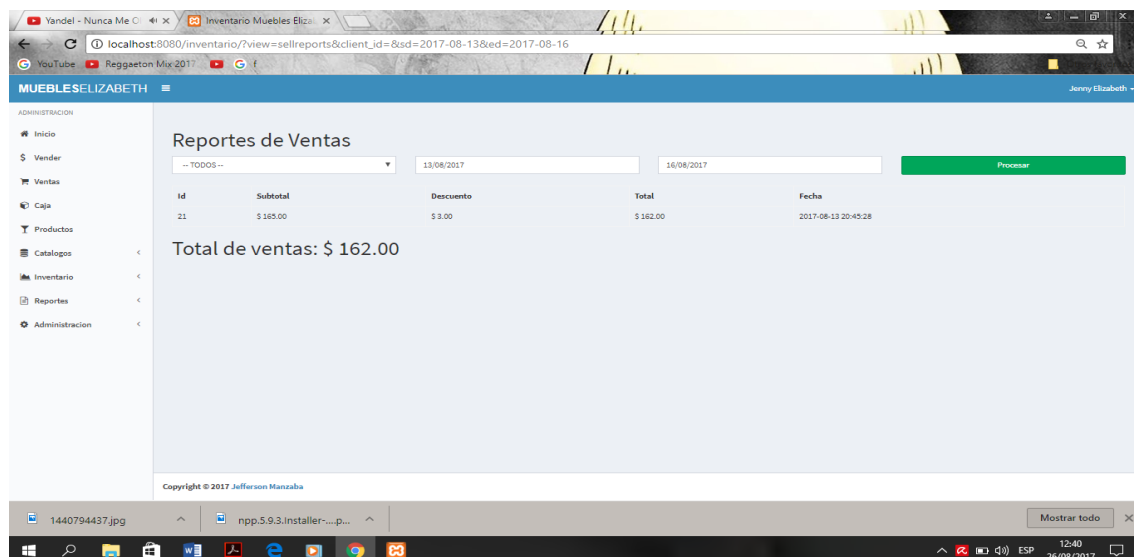


Id	Producto	Cantidad	Operacion	Fecha
42	Pistola de Clavos	3	salida	2017-08-13 20:45:28
41	Pistola silicon	70	entrada	2017-08-13 19:44:25
40	Pistola de Clavos	85	entrada	2017-08-13 19:42:40
39	Pistola de grapas	90	entrada	2017-08-13 19:41:28
38	Silla Regina	120	entrada	2017-08-13 19:39:14
37	Silla en selke lineal	150	entrada	2017-08-13 19:37:48
36	Grapas de clavos	100	entrada	2017-08-13 19:35:49
35	Grapas delgadas	150	entrada	2017-08-13 19:34:16
34	Galón cola blanca	80	entrada	2017-08-13 19:32:04
33	Cameca pega africano	100	entrada	2017-08-13 19:30:41
32	Galón africano	100	entrada	2017-08-13 19:29:27
31	Tela Estampada negra	150	entrada	2017-08-13 19:27:59
30	Tela Cafe	80	entrada	2017-08-13 19:25:26
29	Cuerina Verde	100	entrada	2017-08-13 19:24:15
28	Tablón Salta	500	entrada	2017-08-13 19:22:31

Figura 72. Vista Reportes inventario.

2.15 Menú Reporte Ventas

Permitirá visualizar todo el reporte de ventas realizadas como día, fecha y año.



Id	Subtotal	Descuento	Total	Fecha
21	\$ 165.00	\$ 3.00	\$ 162.00	2017-08-13 20:45:28

Total de ventas: \$ 162.00

Figura 73. Vista Reportes Ventas.

A.03 Manual Técnico



INSTITUTO TECNOLÓGICO
"CORDILLERA"

CARRERA DE ANÁLISIS DE SISTEMAS

SISTEMATIZACIÓN DEL PROCESO DE INVENTARIO MEDIANTE UN APLICATIVO WEB PARA LA EMPRESA MUEBLES ELIZABETH UBICADA EN LA CIUDAD DE QUITO

**Proyecto de investigación previo a la obtención del título de Tecnólogo en
Análisis de Sistemas**

Autor: Jefferson Alexander Manzaba Castro

Tutor: Ing. Johnny Patricio Coronel Ordoñez

Quito, Octubre 2017

ÍNDICE GENERAL

Contenido	Página
1 Introducción	113
2 Contenido.....	113
2.01 Script Base de Datos	113
2.02 Interfaz Inicio Aplicativo	123
2.03 Interfaz Login.....	126
2.04 Interfaz Registro de Personal	128
2.05 Interfaz Registro de Cliente	132
2.06 Interfaz Registro de Proveedores	136
2.07 Interfaz Registro de Categoría de Producto.....	140
2.08 Interfaz Venta	143
2.09 Interfaz Lista de Ventas	147
2.10 Interfaz Caja	149
2.11 Interfaz Inventario de Productos	151
2.12 Interfaz Reabastecer Inventario.....	154
2.13 Interfaz Vista Reabastecimiento.....	159
2.14 Interfaz Reporte Inventario	161
2.15 Interfaz Reporte Ventas	164

ÍNDICE DE FIGURAS

Título	Página
<i>Figura 74. Inicio del Sistema</i>	123
<i>Figura 75. Login</i>	126
<i>Figura 76. Ingreso Personal.....</i>	128
<i>Figura 77. Editar y eliminar Personal</i>	130
<i>Figura 78. Nuevo Cliente... ..</i>	132
<i>Figura 79. Editar y eliminar Cliente.....</i>	134
<i>Figura 80. Nuevo Proveedor.....</i>	136
<i>Figura 81. Editar y eliminar Proveedor</i>	138
<i>Figura 82. Nueva Categoría.....</i>	140
<i>Figura 83. Editar y eliminar Categoría de Producto</i>	141
<i>Figura 84. Venta</i>	143
<i>Figura 85. Lista de Ventas</i>	147
<i>Figura 86. Caja</i>	149
<i>Figura 87. Inventario de Productos</i>	151
<i>Figura 88. Reabastecimiento Inventario</i>	154
<i>Figura 89. Vista Reabastecimiento</i>	159
<i>Figura 90. Reportes</i>	161
<i>Figura 91. Reporte de Ventas.....</i>	164

1 Introducción

Este manual permite visualizar la estructura del código fuente de todo el desarrollo del aplicativo web para dar facilidad al usuario de cómo se creó el proyecto.

2 Contenido

2.01 Script Base de Datos

```
-- phpMyAdmin SQL Dump
-- version 4.7.0
-- https://www.phpmyadmin.net/
--
-- Servidor: 127.0.0.1
-- Tiempo de generación: 26-08-2017 a las 23:54:35
-- Versión del servidor: 10.1.22-MariaDB
-- Versión de PHP: 7.1.4

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET AUTOCOMMIT = 0;
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT
*/;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION
*/;
/*!40101 SET NAMES utf8mb4 */;

--
-- Base de datos: `inventiolite`
--

--
-- Estructura de tabla para la tabla `box`
--

CREATE TABLE `box` (
  `id` int(11) NOT NULL,
  `created_at` datetime DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Volcado de datos para la tabla `box`
```

```
--

INSERT INTO `box` (`id`, `created_at`) VALUES
(1, '2017-08-05 13:32:25'),
(2, '2017-08-05 13:34:14');

-- -----

--
-- Estructura de tabla para la tabla `category`
--

CREATE TABLE `category` (
  `id` int(11) NOT NULL,
  `image` varchar(255) DEFAULT NULL,
  `name` varchar(50) DEFAULT NULL,
  `description` text,
  `created_at` datetime DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Volcado de datos para la tabla `category`
--

INSERT INTO `category` (`id`, `image`, `name`, `description`,
`created_at`) VALUES
(1, NULL, 'Esponjas', NULL, '2017-06-10 11:51:45'),
(2, NULL, 'Madera', NULL, '2017-06-12 18:17:33'),
(3, NULL, 'Tela', NULL, '2017-08-06 10:03:37'),
(4, NULL, 'Pega', NULL, '2017-08-13 18:05:57'),
(5, NULL, 'Grapas', NULL, '2017-08-13 18:06:10'),
(6, NULL, 'Sillas en blanco', NULL, '2017-08-13 18:06:48'),
(7, NULL, 'Sillas terminadas', NULL, '2017-08-13 18:07:01'),
(8, NULL, 'Pistolas', NULL, '2017-08-13 18:58:32');

-- -----

--
-- Estructura de tabla para la tabla `configuration`
--

CREATE TABLE `configuration` (
  `id` int(11) NOT NULL,
  `short` varchar(255) DEFAULT NULL,
  `name` varchar(255) DEFAULT NULL,
  `kind` int(11) DEFAULT NULL,
  `val` varchar(255) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Volcado de datos para la tabla `configuration`
--
```

```
--
```

```
INSERT INTO `configuration` (`id`, `short`, `name`, `kind`,  
`val`) VALUES  
(1, 'title', 'Titulo del Sistema', 2, 'Inventio Lite'),  
(2, 'use_image_product', 'Utilizar Imagenes en los productos',  
1, '0'),  
(3, 'active_clients', 'Activar clientes', 1, '0'),  
(4, 'active_providers', 'Activar proveedores', 1, '0'),  
(5, 'active_categories', 'Activar categorias', 1, '0'),  
(6, 'active_reports_word', 'Activar reportes en Word', 1,  
'0'),  
(7, 'active_reports_excel', 'Activar reportes en Excel', 1,  
'0'),  
(8, 'active_reports_pdf', 'Activar reportes en PDF', 1, '0');
```

```
-- -----
```

```
--
```

```
-- Estructura de tabla para la tabla `operation`
```

```
--
```

```
CREATE TABLE `operation` (  
  `id` int(11) NOT NULL,  
  `product_id` int(11) DEFAULT NULL,  
  `q` float DEFAULT NULL,  
  `operation_type_id` int(11) DEFAULT NULL,  
  `sell_id` int(11) DEFAULT NULL,  
  `created_at` datetime DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--
```

```
-- Volcado de datos para la tabla `operation`
```

```
--
```

```
INSERT INTO `operation` (`id`, `product_id`, `q`,  
`operation_type_id`, `sell_id`, `created_at`) VALUES  
(20, 5, 40, 1, NULL, '2017-08-13 19:01:31'),  
(22, 4, 20, 1, 17, '2017-08-13 19:11:45'),  
(23, 4, 200, 1, 18, '2017-08-13 19:12:19'),  
(24, 5, 120, 1, 19, '2017-08-13 19:13:32'),  
(25, 4, 66, 1, 20, '2017-08-13 19:16:23'),  
(26, 6, 300, 1, NULL, '2017-08-13 19:19:11'),  
(27, 7, 400, 1, NULL, '2017-08-13 19:20:38'),  
(28, 8, 500, 1, NULL, '2017-08-13 19:22:21'),  
(29, 9, 100, 1, NULL, '2017-08-13 19:24:15'),  
(30, 10, 80, 1, NULL, '2017-08-13 19:25:26'),  
(31, 11, 150, 1, NULL, '2017-08-13 19:27:59'),  
(32, 12, 100, 1, NULL, '2017-08-13 19:29:27'),  
(33, 13, 100, 1, NULL, '2017-08-13 19:30:41'),  
(34, 14, 80, 1, NULL, '2017-08-13 19:32:04'),  
(35, 15, 150, 1, NULL, '2017-08-13 19:34:16'),
```



```
(36, 16, 100, 1, NULL, '2017-08-13 19:35:49'),
(37, 17, 150, 1, NULL, '2017-08-13 19:37:48'),
(38, 18, 120, 1, NULL, '2017-08-13 19:39:14'),

(39, 19, 90, 1, NULL, '2017-08-13 19:41:28'),
(40, 20, 85, 1, NULL, '2017-08-13 19:42:40'),
(41, 21, 70, 1, NULL, '2017-08-13 19:44:25'),
(42, 20, 3, 2, 21, '2017-08-13 20:45:28'),
(43, 8, 25, 1, 22, '2017-08-26 12:33:42'),
(44, 7, 2, 2, 23, '2017-08-26 12:41:51'),
(45, 13, 2, 2, 24, '2017-08-26 14:35:00');

-- -----

--
-- Estructura de tabla para la tabla `operation_type`
--

CREATE TABLE `operation_type` (
  `id` int(11) NOT NULL,
  `name` varchar(50) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Volcado de datos para la tabla `operation_type`
--

INSERT INTO `operation_type` (`id`, `name`) VALUES
(1, 'entrada'),
(2, 'salida');

-- -----

--
-- Estructura de tabla para la tabla `person`
--

CREATE TABLE `person` (
  `id` int(11) NOT NULL,
  `image` varchar(255) DEFAULT NULL,
  `name` varchar(255) DEFAULT NULL,
  `lastname` varchar(50) DEFAULT NULL,
  `company` varchar(50) DEFAULT NULL,
  `address1` varchar(50) DEFAULT NULL,
  `address2` varchar(50) DEFAULT NULL,
  `phone1` varchar(50) DEFAULT NULL,
  `phone2` varchar(50) DEFAULT NULL,
  `email1` varchar(50) DEFAULT NULL,
  `email2` varchar(50) DEFAULT NULL,
  `kind` int(11) DEFAULT NULL,
  `created_at` datetime DEFAULT NULL
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Volcado de datos para la tabla `person`
--

INSERT INTO `person` (`id`, `image`, `name`, `lastname`,
`company`, `address1`, `address2`, `phone1`, `phone2`,
`email1`, `email2`, `kind`, `created_at`) VALUES
(1, NULL, 'Almacenes ', 'Armas', NULL, 'San Bartolo', NULL,
'3279010', NULL, 'arma@gmail.com', NULL, 2, '2017-06-10
16:54:10'),
(2, NULL, 'Edison Nolberto', 'Molina Arias', NULL, 'SAN BLAS',
NULL, '3118081', NULL, 'edison @gnail.com', NULL, 1, '2017-06-
10 19:38:28'),
(3, NULL, 'Albert Esteban', 'Casas Torres', NULL, 'La Prensa',
NULL, '3256894', NULL, 'albert12@gmail.com', NULL, 1, '2017-
06-12 18:14:04'),
(4, NULL, 'Almacenes ', 'Yanez', NULL, 'Villaflora', NULL,
'3255617', NULL, 'yanez@gmail.com', NULL, 2, '2017-06-12
18:18:30'),
(5, NULL, 'Laura Lorena', 'Alban Torres', NULL, 'San Martin',
NULL, '3259687', NULL, 'lore@gmail.com', NULL, 1, '2017-08-05
12:23:09');

-- -----
--
-- Estructura de tabla para la tabla `product`
--

CREATE TABLE `product` (
  `id` int(11) NOT NULL,
  `image` varchar(255) DEFAULT NULL,
  `barcode` varchar(50) DEFAULT NULL,
  `name` varchar(50) DEFAULT NULL,
  `description` text,
  `inventory_min` int(11) DEFAULT '10',
  `price_in` float DEFAULT NULL,
  `price_out` float DEFAULT NULL,
  `unit` varchar(255) DEFAULT NULL,
  `presentation` varchar(255) DEFAULT NULL,
  `user_id` int(11) DEFAULT NULL,
  `category_id` int(11) DEFAULT NULL,
  `created_at` datetime DEFAULT NULL,
  `is_active` tinyint(1) DEFAULT '1'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Volcado de datos para la tabla `product`
--
```

```
INSERT INTO `product` (`id`, `image`, `barcode`, `name`,
`description`, `inventory_min`, `price_in`, `price_out`,
`unit`, `presentation`, `user_id`, `category_id`,
`created_at`, `is_active`) VALUES
(4, 'esponja_negra.jpg', '1', 'Esponja negra', 'Buen estado',
150, 8, 11, 'activo', 'Buena', 2, 1, '2017-08-13 18:39:54',
1),

(5, 'esponja_blanca.jpg', '2', 'Esponja blanca', 'Buen
estado', 100, 6, 8, 'activo', 'Buena', 2, 1, NULL, 1),
(6, 'tablon_colorado.jpg', '3', 'Tablo colorado', 'Estado
Bueno', 100, 25, 30, 'activo', 'Buena', 2, 2, NULL, 1),
(7, 'tablon_laurel.jpg', '4', 'Tablon Laurel', 'Estado Bueno',
100, 20, 25, 'activo', 'Buena', 2, 2, NULL, 1),
(8, 'tablon_seike.jpg', '5', 'Tablon Seike', 'Buen estado',
100, 40, 50, 'activo', 'Buena', 2, 2, NULL, 1),
(9, 'cuerina_verde.jpg', '6', 'Cuerina Verde', 'Estado Buena',
50, 150, 180, 'activo', 'Buena', 2, 3, NULL, 1),
(10, 'ctela_cafe.jpg', '7', 'Tela Cafe', 'Estado Buena', 30,
120, 150, 'activo', 'Buena', 2, 3, NULL, 1),
(11, 'tela_estampada.jpg', '8', 'Tela Estampada negra',
'Estado Bueno', 60, 180, 220, 'activo', 'Buena', 2, 3, NULL,
1),
(12, 'galon_peg_africano.jpg', '9', 'Galon africano', 'Estado
Bueno', 20, 7, 9, 'activo', 'Buena', 2, 4, NULL, 1),
(13, 'caneka_peg_africano.jpg', '10', 'Caneca pega africano',
'Estado Bueno', 10, 40, 50, 'activo', 'Buena', 2, 4, NULL, 1),
(14, 'galon_cola_blanca.jpg', '11', 'GalÃ³n cola blanca',
'Estado Bueno', 10, 5, 10, 'activo', 'Buena', 2, 4, NULL, 1),
(15, 'caja_grapas.jpg', '12', 'Grapas delgadas', 'Estado
Bueno', 20, 3, 5, 'activo', 'Buena', 2, 5, NULL, 1),
(16, 'grapas_de_clavos.jpg', '13', 'Grapas de clavos',
'Estado Bueno', 20, 10, 14, 'activo', 'Buena', 2, 5, NULL, 1),
(17, 'silla_en_blanco.jpg', '14', 'Silla en seike lineal',
'Estado Bueno', 20, 30, 40, 'activo', 'Buena', 2, 6, NULL, 1),
(18, 'silla_terminada.jpg', '15', 'Silla Regina', 'Estado
Bueno', 20, 120, 200, 'activo', 'Buena', 2, 7, NULL, 1),
(19, 'pistola_delagada.jpg', '16', 'Pistola de grapas',
'Estado Bueno', 20, 20, 26, 'activo', 'Buena', 2, 8, NULL, 1),
(20, 'pistola_de_clavos.jpg', '17', 'Pistola de Clavos',
'Estado Bueno', 10, 45, 55, 'activo', 'Buena', 2, 8, NULL, 1),
(21, 'pistola_silicon.jpg', '18', 'Pistola silicon', 'Estado
Bueno', 10, 9, 15, 'activo', 'Buena', 2, 8, NULL, 1);

-- -----

--
-- Estructura de tabla para la tabla `sell`
--

CREATE TABLE `sell` (
  `id` int(11) NOT NULL,
  `person_id` int(11) DEFAULT NULL,
```

```
`user_id` int(11) DEFAULT NULL,
`operation_type_id` int(11) DEFAULT '2',
`box_id` int(11) DEFAULT NULL,
`total` double DEFAULT NULL,
`cash` double DEFAULT NULL,
`discount` double DEFAULT NULL,
`created_at` datetime DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Volcado de datos para la tabla `sell`
--

INSERT INTO `sell` (`id`, `person_id`, `user_id`,
`operation_type_id`, `box_id`, `total`, `cash`, `discount`,
`created_at`) VALUES
(17, 1, 2, 1, NULL, NULL, NULL, NULL, '2017-08-13 19:11:45'),
(18, 1, 2, 1, NULL, NULL, NULL, NULL, '2017-08-13 19:12:19'),
(19, 4, 2, 1, NULL, NULL, NULL, NULL, '2017-08-13 19:13:32'),
(20, 1, 2, 1, NULL, NULL, NULL, NULL, '2017-08-13 19:16:23'),
(21, 2, 2, 2, NULL, 165, NULL, 3, '2017-08-13 20:45:28'),
(22, 1, 2, 1, NULL, NULL, NULL, NULL, '2017-08-26 12:33:42'),
(23, 5, 2, 2, NULL, 50, NULL, 5, '2017-08-26 12:41:51'),
(24, 3, 2, 2, NULL, 100, NULL, 0, '2017-08-26 14:35:00');

-- -----
--
-- Estructura de tabla para la tabla `user`
--

CREATE TABLE `user` (
`id` int(11) NOT NULL,
`name` varchar(50) DEFAULT NULL,
`lastname` varchar(50) DEFAULT NULL,
`username` varchar(50) DEFAULT NULL,
`email` varchar(255) DEFAULT NULL,
`password` varchar(60) DEFAULT NULL,
`image` varchar(255) DEFAULT NULL,
`is_active` tinyint(1) NOT NULL DEFAULT '1',
`is_admin` tinyint(1) NOT NULL DEFAULT '0',
`created_at` datetime DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--
-- Volcado de datos para la tabla `user`
--

INSERT INTO `user` (`id`, `name`, `lastname`, `username`,
`email`, `password`, `image`, `is_active`, `is_admin`,
`created_at`) VALUES
```

```
(2, 'Jenny Elizabeth', 'Castro PazmiÃfÃto', 'Jenny',
'jenny45@hotmail.es',
'90d6ca87acf3ecb33372be0304f4464e5247b4c8', NULL, 1, 1, '2017-
06-10 11:49:04'),
(4, 'Dario Alberto', 'Casa Hernandez', 'Dario',
'dario29@gmail.com',
'b2ba3603bde2d84c6fd8a2e884d8a53837974dfc', NULL, 1, 0, '2017-
08-13 16:29:01');
```

```
--
```

```
-- Ãndices para tablas volcadas
```

```
--
```

```
--
```

```
-- Indices de la tabla `box`
```

```
--
```

```
ALTER TABLE `box`
  ADD PRIMARY KEY (`id`);
```

```
--
```

```
-- Indices de la tabla `category`
```

```
--
```

```
ALTER TABLE `category`
  ADD PRIMARY KEY (`id`);
```

```
--
```

```
-- Indices de la tabla `configuration`
```

```
--
```

```
ALTER TABLE `configuration`
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY `short` (`short`),
  ADD UNIQUE KEY `name` (`name`);
```

```
--
```

```
-- Indices de la tabla `operation`
```

```
--
```

```
ALTER TABLE `operation`
  ADD PRIMARY KEY (`id`),
  ADD KEY `product_id` (`product_id`),
  ADD KEY `operation_type_id` (`operation_type_id`),
  ADD KEY `sell_id` (`sell_id`);
```

```
--
```

```
-- Indices de la tabla `operation_type`
```

```
--
```

```
ALTER TABLE `operation_type`
  ADD PRIMARY KEY (`id`);
```

```
--
```

```
-- Indices de la tabla `person`
```

```
--
```

```
ALTER TABLE `person`
  ADD PRIMARY KEY (`id`);

--
-- Indices de la tabla `product`
--
ALTER TABLE `product`
  ADD PRIMARY KEY (`id`),
  ADD KEY `category_id` (`category_id`),
  ADD KEY `user_id` (`user_id`);

--
-- Indices de la tabla `sell`
--
ALTER TABLE `sell`
  ADD PRIMARY KEY (`id`),
  ADD KEY `box_id` (`box_id`),
  ADD KEY `operation_type_id` (`operation_type_id`),
  ADD KEY `user_id` (`user_id`),
  ADD KEY `person_id` (`person_id`);

--
-- Indices de la tabla `user`
--
ALTER TABLE `user`
  ADD PRIMARY KEY (`id`);

--
-- AUTO_INCREMENT de las tablas volcadas
--

--
-- AUTO_INCREMENT de la tabla `box`
--
ALTER TABLE `box`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=3;

--
-- AUTO_INCREMENT de la tabla `category`
--
ALTER TABLE `category`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=9;

--
-- AUTO_INCREMENT de la tabla `configuration`
--
ALTER TABLE `configuration`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,
  AUTO_INCREMENT=9;

--
-- AUTO_INCREMENT de la tabla `operation`
```

```
--  
ALTER TABLE `operation`  
  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,  
    AUTO_INCREMENT=46;  
--  
-- AUTO_INCREMENT de la tabla `operation_type`  
--  
ALTER TABLE `operation_type`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,  
    AUTO_INCREMENT=3;  
--  
-- AUTO_INCREMENT de la tabla `person`  
--  
ALTER TABLE `person`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,  
    AUTO_INCREMENT=6;  
--  
-- AUTO_INCREMENT de la tabla `product`  
--  
ALTER TABLE `product`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,  
    AUTO_INCREMENT=22;  
--  
-- AUTO_INCREMENT de la tabla `sell`  
--  
ALTER TABLE `sell`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,  
    AUTO_INCREMENT=25;  
--  
-- AUTO_INCREMENT de la tabla `user`  
--  
ALTER TABLE `user`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,  
    AUTO_INCREMENT=5;  
--  
-- Restricciones para tablas volcadas  
--  
  
--  
-- Filtros para la tabla `operation`  
--  
ALTER TABLE `operation`  
    ADD CONSTRAINT `operation_ibfk_1` FOREIGN KEY (`product_id`)  
    REFERENCES `product` (`id`),  
    ADD CONSTRAINT `operation_ibfk_2` FOREIGN KEY  
    (`operation_type_id`) REFERENCES `operation_type` (`id`),  
    ADD CONSTRAINT `operation_ibfk_3` FOREIGN KEY (`sell_id`)  
    REFERENCES `sell` (`id`);
```

```
--

-- Filtros para la tabla `product`
--
ALTER TABLE `product`
  ADD CONSTRAINT `product_ibfk_1` FOREIGN KEY (`category_id`)
REFERENCES `category` (`id`),

  ADD CONSTRAINT `product_ibfk_2` FOREIGN KEY (`user_id`)
REFERENCES `user` (`id`);

--

-- Filtros para la tabla `sell`
--
ALTER TABLE `sell`
  ADD CONSTRAINT `sell_ibfk_1` FOREIGN KEY (`box_id`)
REFERENCES `box` (`id`),
  ADD CONSTRAINT `sell_ibfk_2` FOREIGN KEY
(`operation_type_id`) REFERENCES `operation_type` (`id`),
  ADD CONSTRAINT `sell_ibfk_3` FOREIGN KEY (`user_id`)
REFERENCES `user` (`id`),
  ADD CONSTRAINT `sell_ibfk_4` FOREIGN KEY (`person_id`)
REFERENCES `person` (`id`);
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT
*/;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS
*/;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION
*/;
```

2.02 Interfaz Inicio Aplicativo

Inicio del sistema.

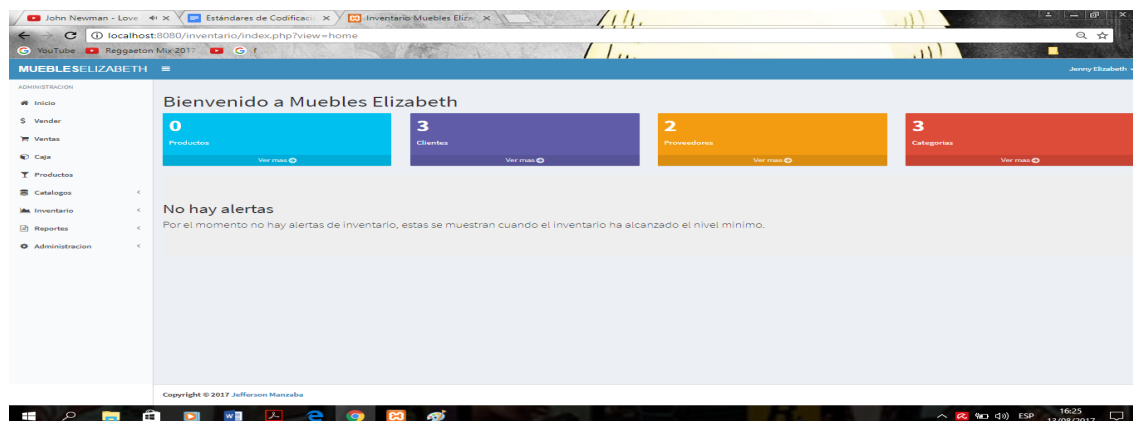


Figura 74. Inicio del sistema.

Código Fuente

```
<?php
    $found=true;
    $products = ProductData::getAll();
    foreach($products as $product){
        $q=OperationData::getQYesF($product->id);
        if($q<=$product->inventory_min){
            $found=true;

            break;

        }
    }
    ?>
<div class="row">
    <div class="col-md-12">
        <h1>Bienvenido a Muebles Elizabeth</h1>
    </div>
</div>
<div class="row">
    <div class="col-lg-3 col-xs-6">
        <!-- small box -->
        <div class="small-box bg-aqua">
            <div class="inner">
                <h3><?php echo
count(ProductData::getAll());?></h3>

                <p>Productos</p>
            </div>
            <div class="icon">
                <i class="ion ion-bag"></i>
            </div>
            <a href="./?view=products" class="small-box-
footer">Ver mas <i class="fa fa-arrow-circle-right"></i></a>
            </div>
        </div>
        <!-- ./col -->
        <div class="col-lg-3 col-xs-6">
            <!-- small box -->
            <div class="small-box bg-purple">
                <div class="inner">
                    <h3><?php echo
count(PersonData::getClients());?></h3>

                    <p>Clientes</p>
                </div>
                <div class="icon">
                    <i class="ion ion-stats-bars"></i>
```

```

        </div>

        <a href="./?view=clients" class="small-box-
footer">Ver mas <i class="fa fa-arrow-circle-right"></i></a>
    </div>
</div>
<!-- ./col -->
<div class="col-lg-3 col-xs-6">
    <!-- small box -->
    <div class="small-box bg-yellow">
        <div class="inner">

            <h3><?php echo
count(PersonData::getProviders());?></h3>

            <p>Proveedores</p>
        </div>
        <div class="icon">
            <i class="ion ion-person-add"></i>
        </div>
        <a href="./?view=providers" class="small-box-
footer">Ver mas <i class="fa fa-arrow-circle-right"></i></a>
    </div>
</div>
<!-- ./col -->
<div class="col-lg-3 col-xs-6">
    <!-- small box -->
    <div class="small-box bg-red">
        <div class="inner">
            <h3><?php echo
count(CategoryData::getAll());?></h3>

            <p>Categorías</p>

        </div>
        <div class="icon">
            <i class="ion ion-pie-graph"></i>
        </div>
        <a href="./?view=categories" class="small-box-
footer">Ver mas <i class="fa fa-arrow-circle-right"></i></a>
    </div>
</div>
<!-- ./col -->
</div>
<!-- /.row -->

<div class="row">
    <div class="col-md-12">
        <?php if($found):?>
            <ul class="dropdown-menu" role="menu">

```

```
<li><a href="report/alerts-word.php">Word 2007
(.docx)</a></li>
</ul>
</div>
<?php endif;?>

</div>
<div class="clearfix"></div>
<?php if(count($products)>0){?>
<br><table class="table table-bordered table-hover">
    <thead>
        <th >Codigo</th>
        <th>Nombre del producto</th>

        <th>En Stock</th>
        <th></th>
    </thead>
    <?php
foreach($products as $product):
    $q=OperationData::getQYesF($product->id);
    ?>
    <?php if($q<=$product->inventory_min):?>
    <tr class="<?php if($q==0){ echo "danger"; }else
if($q<=$product->inventory_min/2){ echo "danger"; } else
if($q<=$product->inventory_min){ echo "warning"; } ?>">
        <td><?php echo $product->id; ?></td>
        <td><?php echo $product->name; ?></td>
        <td><?php echo $q; ?></td>
        <td>
            <?php if($q==0){ echo "<span class='label label-
danger'>No hay existencias.</span>"; }else if($q<=$product-
>inventory_min/2){ echo "<span class='label label-
danger'>Quedan muy pocas existencias.</span>"; } else
if($q<=$product->inventory_min){ echo "<span class='label
label-warning'>Quedan pocas existencias.</span>"; } ?>
        </td>
    </tr>

    <?php endif;?>
<?php
endforeach;
?>
</table>

<div class="clearfix"></div>

    <?php
}else{
    ?>
    <div class="jumbotron">
        <h2>No hay alertas</h2>
```

```

        <p>Por el momento no hay alertas de inventario,
estas se muestran cuando el inventario ha alcanzado el nivel
minimo.</p>
    </div>
    <?php
}

?>
<br><br><br><br><br><br><br><br><br>
    </div>
</div>

```

2.03 Interfaz Login

Ingreso Usuario y Contraseña.

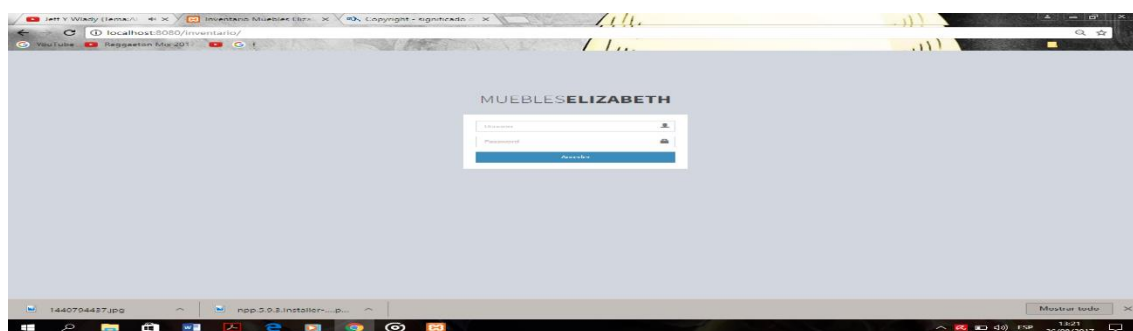


Figura 75. Login.

Código Fuente

```

<?php

if(isset($_SESSION["user_id"]) && $_SESSION["user_id"]!=""){
    print
    "<script>>window.location='index.php?view=home';</script>";
}

?>
<br><br><br><br><br>
<div class="content">
<div class="container">
<div class="row">
    <div class="col-md-4 col-md-offset-4">
        <?php if(isset($_COOKIE['password_updated'])):??>
            <div class="alert alert-success">
                <p><i class='glyphicon glyphicon-off'></i> Se ha
cambiado la contraseña exitosamente !!</p>
                <p>Pruebe iniciar sesion con su nueva
contraseña.</p>
            </div>
            <?php setcookie("password_updated","",time()-18600);
            endif; ??>

```

```
<div class="card">
    <div class="card-header" data-
background-color="green">
        <h4 class="title">Acceder a
Inventio Lite</h4>
    </div>
    <div class="card-content">
        <form accept-charset="UTF-8" role="form" method="post"
action="index.php?view=processlogin">
            <fieldset>
                <div class="form-group">
                    <input class="form-control"
placeholder="Usuario" name="mail" type="text">
                </div>
                <div class="form-group">
                    <input class="form-control"
placeholder="Contraseña" name="password" type="password"
value="">
                </div>
                <input class="btn btn-lg btn-
primary btn-block" type="submit" value="Iniciar Sesion">
            </fieldset>
        </form>
    </div>
</div>
</div>
</div>
</div>
<br><br><br><br><br><br><br><br><br><br><br>
```

2.04 Interfaz Registro de Personal

Registro Personal.

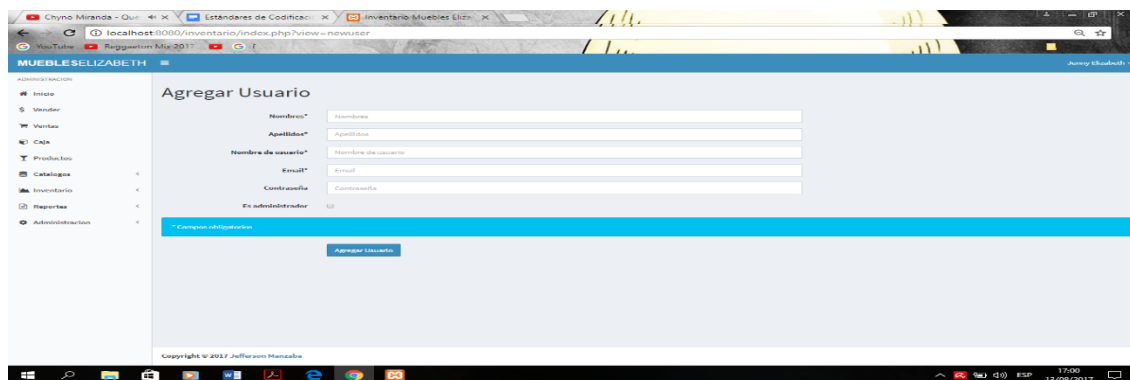


Figura 76. Ingreso Personal.

Código Fuente

```
<div class="row">
    <div class="col-md-12">
        <h1>Agregar Usuario</h1>
        <br>
        <form class="form-horizontal" method="post"
id="addproduct" action="index.php?view=adduser" role="form">

            <div class="form-group">
                <label for="inputEmail1" class="col-lg-2 control-
label">Nombres*</label>
                <div class="col-md-6">
                    <input type="text" name="name" class="form-control"
id="name" placeholder="Nombres">
                </div>
            </div>
            <div class="form-group">
                <label for="inputEmail1" class="col-lg-2 control-
label">Apellidos*</label>
                <div class="col-md-6">
                    <input type="text" name="lastname" required class="form-
control" id="lastname" placeholder="Apellidos">
                </div>
            </div>
            <div class="form-group">

                <label for="inputEmail1" class="col-lg-2 control-
label">Nombre de usuario*</label>
                <div class="col-md-6">
                    <input type="text" name="username" class="form-control"
required id="username" placeholder="Nombre de usuario">
                </div>
            </div>
            <div class="form-group">
                <label for="inputEmail1" class="col-lg-2 control-
label">Email*</label>
                <div class="col-md-6">
                    <input type="text" name="email" class="form-control"
id="email" placeholder="Email">
                </div>
            </div>
            <div class="form-group">
                <label for="inputEmail1" class="col-lg-2 control-
label">Contrase&ntilde;a</label>
                <div class="col-md-6">
                    <input type="password" name="password" class="form-
control" id="inputEmail1" placeholder="Contrase&ntilde;a">
                </div>
            </div>
```

```

</div>

<div class="form-group">
  <label for="inputEmail1" class="col-lg-2 control-label">Es
administrador</label>
  <div class="col-md-6">
<div class="checkbox">
  <label>
    <input type="checkbox" name="is_admin">
  </label>
</div>
</div>
</div>

<p class="alert alert-info">* Campos obligatorios</p>

<div class="form-group">
  <div class="col-lg-offset-2 col-lg-10">
    <button type="submit" class="btn btn-primary">Agregar
Usuario</button>
  </div>
</div>
</form>
</div>
</div>

```

Editar y eliminar Personal.

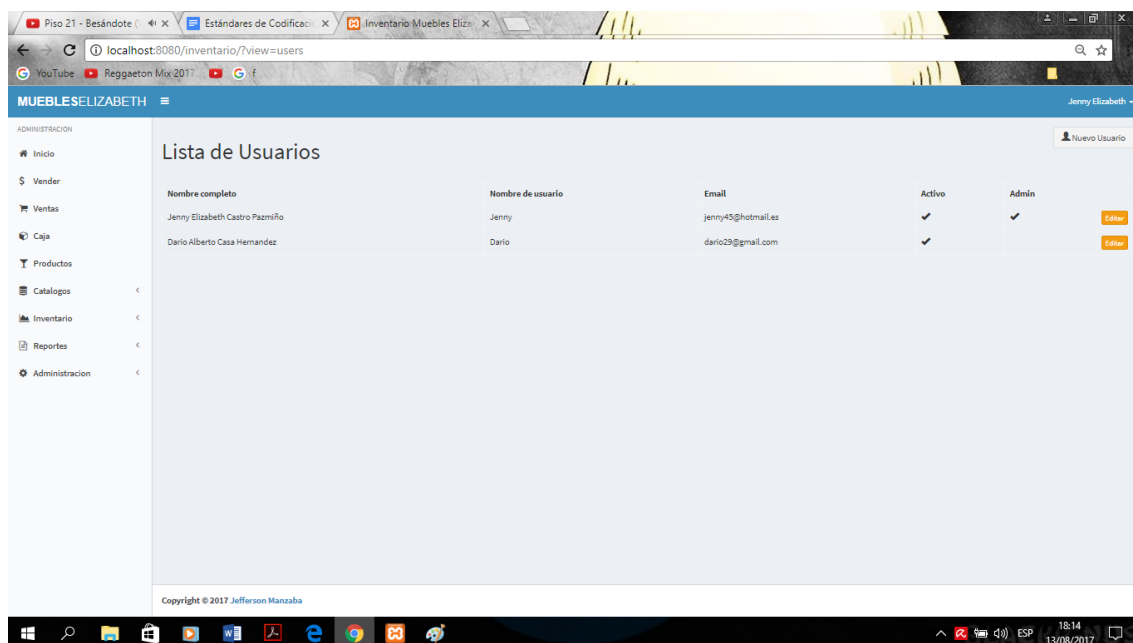


Figura 77. Editar y eliminar Personal.

Código Fuente

```
<div class="row">
    <div class="col-md-12">
        <a href="index.php?view=newuser" class="btn btn-default
pull-right"><i class='glyphicon glyphicon-user'></i> Nuevo
Usuario</a>
        <h1>Lista de Usuarios</h1>
    <br>
    <?php
    /*
    $u = new UserData();
    print_r($u);
    $u->name = "Agustin";
    $u->lastname = "Ramos";
    $u->email = "evilnapsis@gmail.com";
    $u->password = sha1(md5("l00lapal00za"));
    $u->add();

    $f = $u->createForm();
    print_r($f);
    echo $f->label("name")." ".$f->render("name");
    */
    ?>
    <?php

    $users = UserData::getAll();
    if(count($users)>0){
        // si hay usuarios
        ?>
        <table class="table table-bordered table-
hover">
            <thead>
            <th>Nombre completo</th>
            <th>Nombre de usuario</th>
            <th>Email</th>
            <th>Activo</th>
            <th>Admin</th>
            <th></th>
            </thead>
            <?php
            foreach($users as $user){
                ?>
                <tr>
                <td><?php echo $user->name." ".$user-
>lastname; ?></td>
                <td><?php echo $user->username; ?></td>
                <td><?php echo $user->email; ?></td>
                <td>
```



```

        <?php if($user->is_active):?>

            <i class="glyphicon glyphicon-ok"></i>

        <?php endif; ?>
    </td>
    <td>
        <?php if($user->is_admin):?>
            <i class="glyphicon glyphicon-ok"></i>

        <?php endif; ?>
    </td>
    <td style="width:30px;"><a
href="index.php?view=edituser&id=<?php echo $user->id;?>"
class="btn btn-warning btn-xs">Editar</a></td>
</tr>
<?php

    }
    echo "</table>";

    }else{
        // no hay usuarios
    }

    ?>

</div>

</div>

```

2.05 Interfaz Registro de Cliente

Ingreso Cliente.

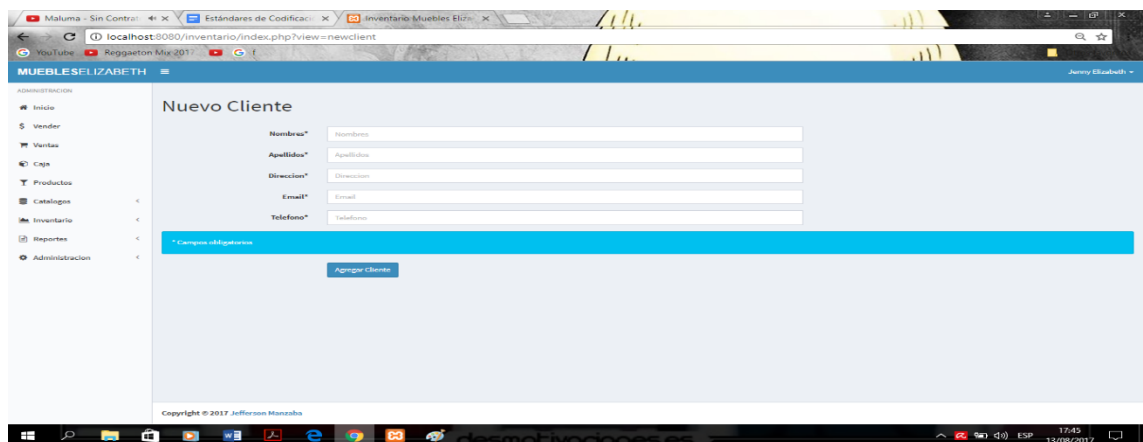


Figura 78. Nuevo Cliente.

Código Fuente

```
<div class="row">
  <div class="col-md-12">
    <h1>Nuevo Cliente</h1>
    <br>
    <form class="form-horizontal" method="post"
id="addproduct" action="index.php?view=addclient" role="form">

      <div class="form-group">
        <label for="inputEmail1" class="col-lg-2 control-
label">Nombres*</label>
        <div class="col-md-6">
          <input type="text" name="name" class="form-control"
id="name" placeholder="Nombres">
        </div>
      </div>
      <div class="form-group">
        <label for="inputEmail1" class="col-lg-2 control-
label">Apellidos*</label>
        <div class="col-md-6">

          <input type="text" name="lastname" required class="form-
control" id="lastname" placeholder="Apellidos">
        </div>
      </div>
      <div class="form-group">
        <label for="inputEmail1" class="col-lg-2 control-
label">Direccion*</label>
        <div class="col-md-6">
          <input type="text" name="address1" class="form-control"
required id="address1" placeholder="Direccion">
        </div>
      </div>
      <div class="form-group">
        <label for="inputEmail1" class="col-lg-2 control-
label">Email*</label>
        <div class="col-md-6">
          <input type="text" name="email1" class="form-control"
id="email1" placeholder="Email">
        </div>
      </div>
      <div class="form-group">
        <label for="inputEmail1" class="col-lg-2 control-
label">Telefono*</label>
        <div class="col-md-6">
          <input type="text" name="phone1" class="form-control"
id="phone1" placeholder="Telefono">
        </div>
      </div>
    </form>
  </div>
</div>
```

```

    </div>
</div>

<p class="alert alert-info">* Campos obligatorios</p>

<div class="form-group">
    <div class="col-lg-offset-2 col-lg-10">
        <button type="submit" class="btn btn-primary">Agregar
Cliente</button>
    </div>
</div>
</form>
    </div>
</div>

```

Editar y eliminar Cliente.

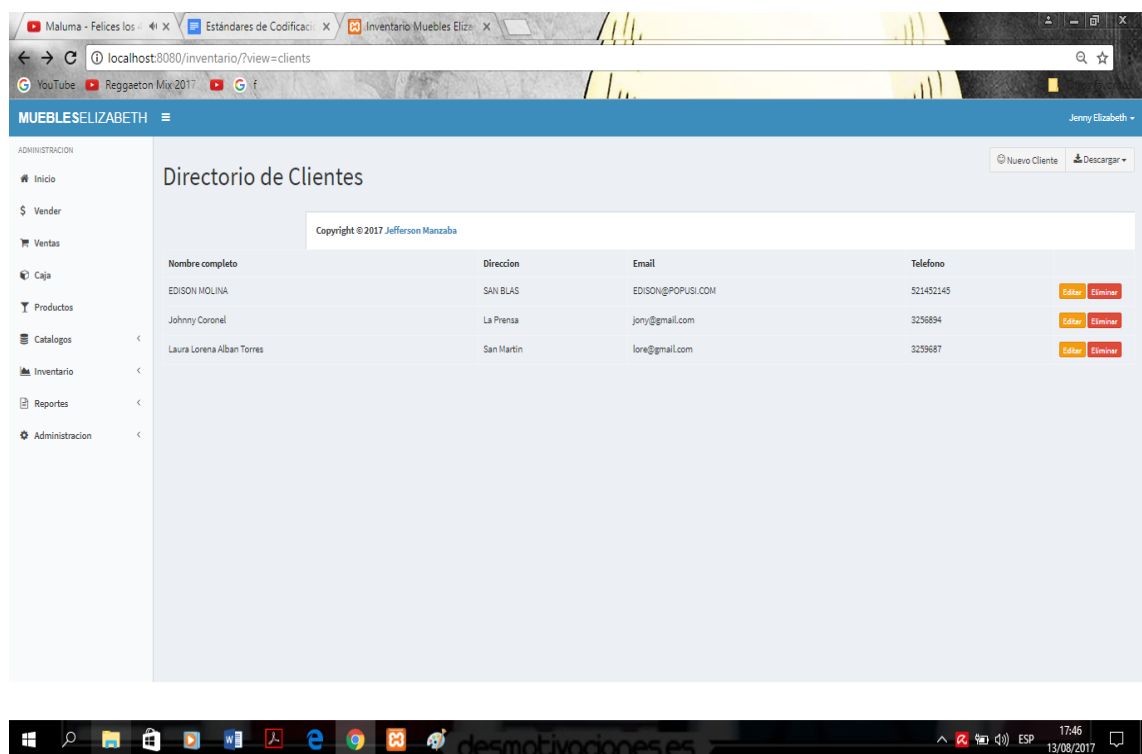


Figura 79. Editar y eliminar Cliente.

Código Fuente

```
<div class="row">
    <div class="col-md-12">
<div class="btn-group pull-right">
    <a href="index.php?view=newclient" class="btn btn-
default"><i class='fa fa-smile-o'></i> Nuevo Cliente</a>
<div class="btn-group pull-right">
    <button type="button" class="btn btn-default dropdown-
toggle" data-toggle="dropdown">
        <i class="fa fa-download"></i> Descargar <span
class="caret"></span>
    </button>
    <ul class="dropdown-menu" role="menu">
        <li><a href="report/clients-word.php">Word 2007
(.docx)</a></li>
    </ul>
</div>
</div>

    <h1>Directorio de Clientes</h1>
<br>
<?php

$users = PersonData::getClients();
if(count($users)>0){
    // si hay usuarios

    ?>

    <table class="table table-bordered table-
hover">

        <thead>
        <th>Nombre completo</th>
        <th>Direccion</th>
        <th>Email</th>
        <th>Telefono</th>
        <th></th>
        </thead>
        <?php
        foreach($users as $user){
            ?>
            <tr>
            <td><?php echo $user->name." ".$user-
>lastname; ?></td>
                <td><?php echo $user->address1; ?></td>
                <td><?php echo $user->email1; ?></td>
                <td><?php echo $user->phone1; ?></td>
                <td style="width:130px;">
                <a
href="index.php?view=editclient&id=<?php echo $user->id;?>"
class="btn btn-warning btn-xs">Editar</a>
```

```

        <a
href="index.php?view=delclient&id=php echo $user-&gt;id;?&gt;"
class="btn btn-danger btn-xs"&gt;Eliminar&lt;/a&gt;
        &lt;/td&gt;
    &lt;/tr&gt;
    &lt;?php

    }

    }else{
        echo "&lt;p class='alert alert-danger'&gt;No hay
clientes&lt;/p&gt;";
    }

    ?&gt;

&lt;/div&gt;
&lt;/div&gt;
</pre

```

2.06 Interfaz Registro de Proveedores

Ingreso Proveedor.

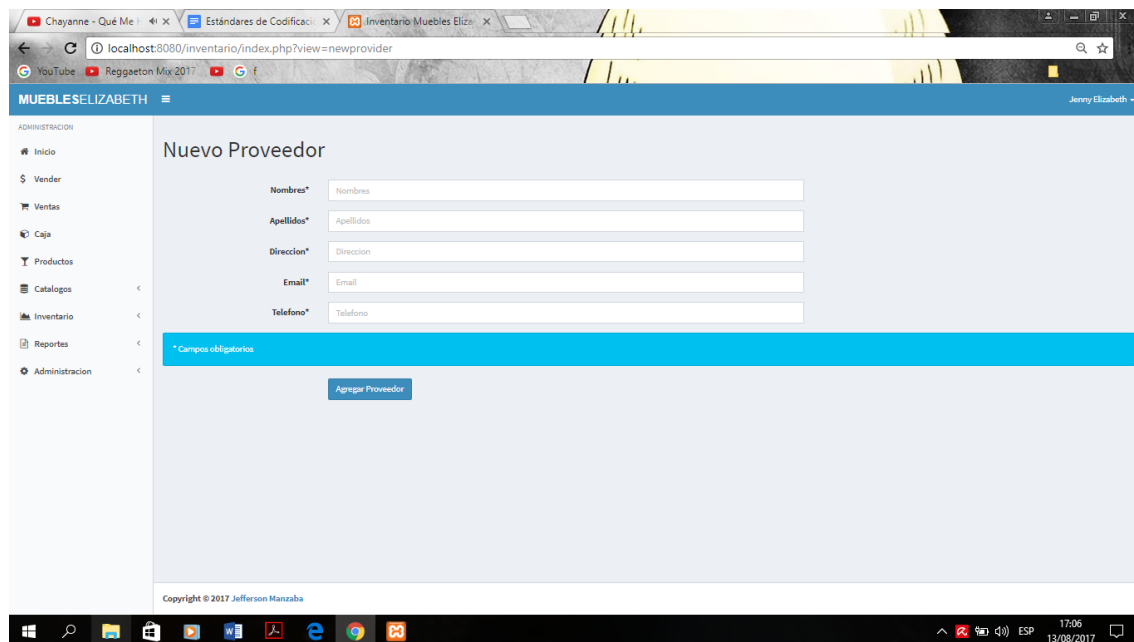


Figura 80. Nuevo Proveedor.

Código Fuente

```
<div class="row">
  <div class="col-md-12">

    <h1>Nuevo Proveedor</h1>
    <br>
    <form class="form-horizontal" method="post"
id="addproduct" action="index.php?view=addprovider"
role="form">

      <div class="form-group">
        <label for="inputEmail1" class="col-lg-2 control-
label">Nombres*</label>
        <div class="col-md-6">
          <input type="text" name="name" class="form-control"
id="name" placeholder="Nombres">
        </div>
      </div>
      <div class="form-group">
        <label for="inputEmail1" class="col-lg-2 control-
label">Apellidos*</label>
        <div class="col-md-6">
          <input type="text" name="lastname" required class="form-
control" id="lastname" placeholder="Apellidos">
        </div>
      </div>

      <div class="form-group">
        <label for="inputEmail1" class="col-lg-2 control-
label">Direccion*</label>
        <div class="col-md-6">
          <input type="text" name="address1" class="form-control"
required id="address1" placeholder="Direccion">
        </div>
      </div>
      <div class="form-group">
        <label for="inputEmail1" class="col-lg-2 control-
label">Email*</label>
        <div class="col-md-6">
          <input type="text" name="email1" class="form-control"
id="email1" placeholder="Email">
        </div>
      </div>

      <div class="form-group">
        <label for="inputEmail1" class="col-lg-2 control-
label">Telefono*</label>
        <div class="col-md-6">
          <input type="text" name="phone1" class="form-control"
id="phone1" placeholder="Telefono">
        </div>
      </div>
```

```
<p class="alert alert-info">* Campos obligatorios</p>

<div class="form-group">
  <div class="col-lg-offset-2 col-lg-10">
    <button type="submit" class="btn btn-primary">Agregar
Proveedor</button>
  </div>
</div>
</form>
</div>
</div>
```

Editar y eliminar Proveedor.

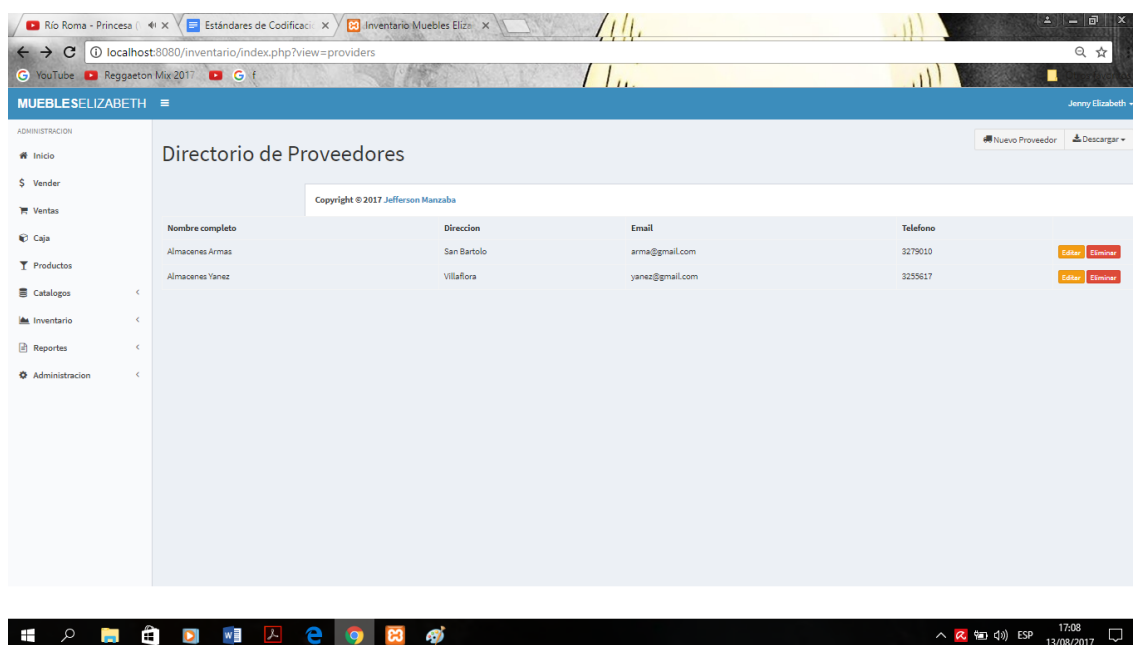


Figura 81. Editar y eliminar Proveedor.

Código Fuente

```
<div class="row">
  <div class="col-md-12">
<div class="btn-group pull-right">
  <a href="index.php?view=newprovider" class="btn btn-
default"><i class='fa fa-truck'></i> Nuevo Proveedor</a>
<div class="btn-group pull-right">
  <button type="button" class="btn btn-default dropdown-
toggle" data-toggle="dropdown">
    <i class="fa fa-download"></i> Descargar <span
class="caret"></span>
```

```

</button>
<ul class="dropdown-menu" role="menu">
  <li><a href="report/providers-word.php">Word 2007
(.docx)</a></li>

</ul>
</div>
</div>

<br>

<h1>Directorio de Proveedores</h1>

<?php

$users = PersonData::getProviders();
if(count($users)>0){
    // si hay usuarios

    ?>

    <table class="table table-bordered table-
hover">

    <thead>
    <th>Nombre completo</th>
    <th>Direccion</th>
    <th>Email</th>
    <th>Telefono</th>
    <th></th>
    </thead>
    <?php
    foreach($users as $user){
        ?>
        <tr>
        <td><?php echo $user->name." ".$user-
>lastname; ?></td>
        <td><?php echo $user->address1; ?></td>
        <td><?php echo $user->email1; ?></td>
        <td><?php echo $user->phone1; ?></td>
        <td style="width:130px;">
        <a
href="index.php?view=editprovider&id=<?php echo $user->id;?>"
class="btn btn-warning btn-xs">Editar</a>
        <a
href="index.php?view=delprovider&id=<?php echo $user->id;?>"
class="btn btn-danger btn-xs">Eliminar</a>

        </td>
        </tr>
        <?php

    }

}
else{

```



```
        echo "<p class='alert alert-danger'>No hay  
proveedores</p>";  
    }
```

```
?>
```

```
</div>  
</div>
```

2.07 Interfaz Registro de Categoría de Producto

Ingreso Categoría de Producto.

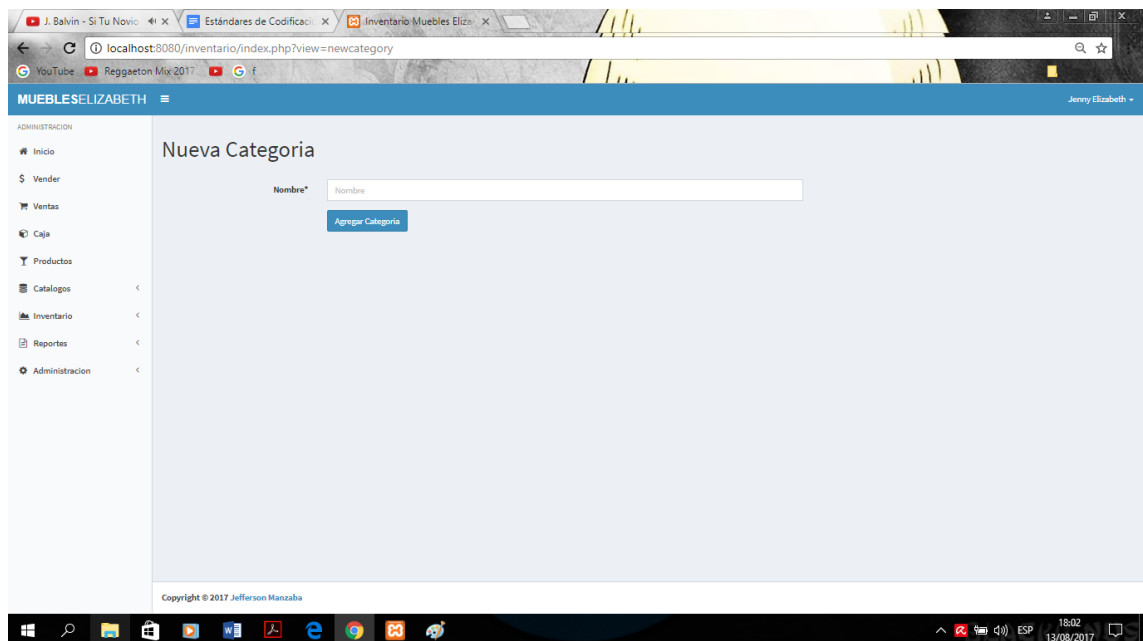


Figura 82. Nueva Categoría.

Código Fuente

```
<div class="row">  
    <div class="col-md-12">  
        <h1>Nueva Categoría</h1>  
        <br>  
        <form class="form-horizontal" method="post"  
id="addcategory" action="index.php?view=addcategory"  
role="form">  
  
        <div class="form-group">
```

```
<label for="inputEmail1" class="col-lg-2 control-label">Nombre*</label>
<div class="col-md-6">
    <input type="text" name="name" required class="form-control" id="name" placeholder="Nombre">
</div>

</div>

<div class="form-group">
    <div class="col-lg-offset-2 col-lg-10">
        <button type="submit" class="btn btn-primary">Agregar
        Categoría</button>
    </div>
</div>
</form>
</div>
</div>
```

Editar y eliminar Categoría de Producto.

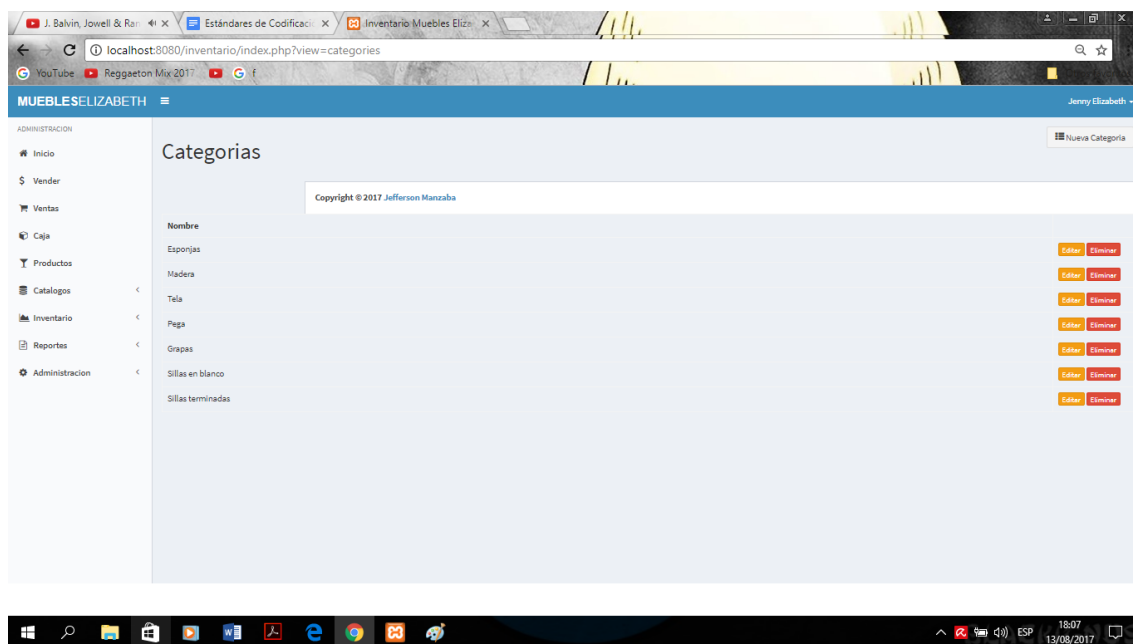


Figura 83. Editar y eliminar Categoría de Producto.

Código Fuente

```
<div class="row">
    <div class="col-md-12">
<div class="btn-group pull-right">
    <a href="index.php?view=newcategory" class="btn btn-
    default"><i class='fa fa-th-list'></i> Nueva Categoría</a>
</div>
```

```
<h1>Categorias</h1>
<br>
<?php

$users = CategoryData::getAll();
if(count($users)>0){

    // si hay usuarios
    ?>

    <table class="table table-bordered table-
hover">

    <thead>
    <th>Nombre</th>
    <th></th>
    </thead>

    <?php
    foreach($users as $user){
        ?>
        <tr>
        <td><?php echo $user->name." ".$user-
>lastname; ?></td>
        <td style="width:130px;"><a
href="index.php?view=editcategory&id=<?php echo $user->id;?>"
class="btn btn-warning btn-xs">Editar</a> <a
href="index.php?view=delcategory&id=<?php echo $user->id;?>"
class="btn btn-danger btn-xs">Eliminar</a></td>
        </tr>
        <?php

    }

    }else{
        echo "<p class='alert alert-danger'>No hay
Categorias</p>";
    }

    ?>

</div>
</div>
```

2.08 Interfaz Venta

Proceso de Venta del Producto.

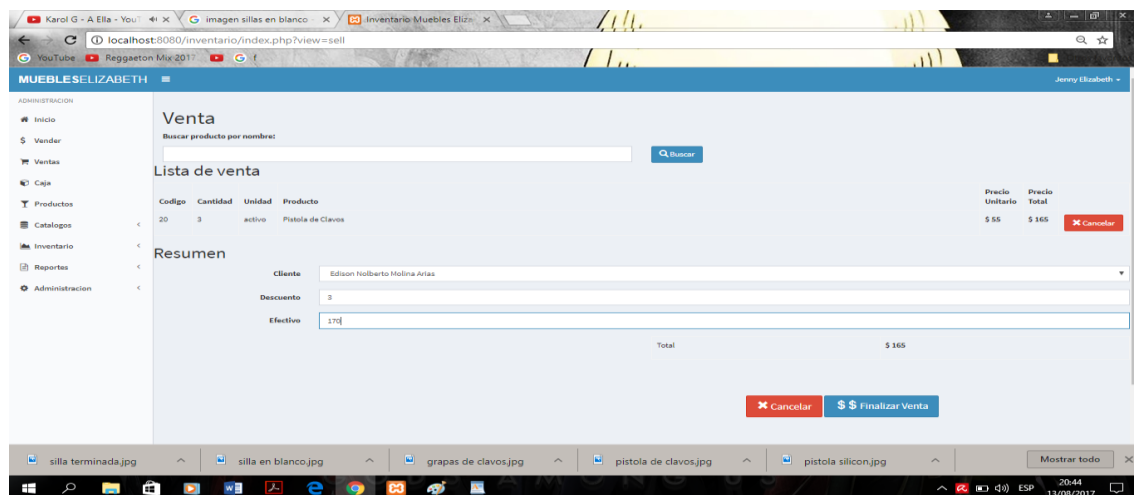


Figura 84. Venta.

Código Fuente

```
<div class="row">
  <div class="col-md-12">
    <h1>Venta</h1>
    <p><b>Buscar producto por nombre:</b></p>
    <form id="searchp">
      <div class="row">
        <div class="col-md-6">
          <input type="hidden" name="view"
value="sell">
          <input type="text" id="product_code"
name="product" class="form-control">
        </div>
        <div class="col-md-3">
          <button type="submit" class="btn btn-
primary"><i class="glyphicon glyphicon-search"></i>
Buscar</button>
        </div>
      </div>
    </form>
  </div>
</div>
<div id="show_search_results"></div>
<script>
//jQuery.noConflict();

$(document).ready(function() {
  $("#searchp").on("submit", function(e) {
    e.preventDefault();
```

```
$.get("./?action=searchproduct",$("#searchp").serialize(
),function(data){
    $("#show_search_results").html(data);
});
$("#product_code").val("");

});
});

$(document).ready(function(){
    $("#product_code").keydown(function(e){
        if(e.which==17 || e.which==74){
            e.preventDefault();
        }else{
            console.log(e.which);
        }
    })
});
</script>

<?php if(isset($_SESSION["errors"])):?>
<h2>Errores</h2>
<p></p>
<table class="table table-bordered table-hover">
<tr class="danger">
    <th>Codigo</th>
    <th>Producto</th>
    <th>Mensaje</th>
</tr>
<?php foreach ($_SESSION["errors"] as $error):
$product = ProductData::getId($error["product_id"]);
?>
<tr class="danger">
    <td><?php echo $product->id; ?></td>
    <td><?php echo $product->name; ?></td>
    <td><b><?php echo $error["message"]; ?></b></td>
</tr>

<?php endforeach; ?>
</table>
<?php
unset($_SESSION["errors"]);
endif; ?>

<!-- Carrito de compras :> -->
<?php if(isset($_SESSION["cart"])):
$total = 0;
?>
<h2>Lista de venta</h2>
<table class="table table-bordered table-hover">
```

```
<thead>

    <th style="width:30px;">Codigo</th>
    <th style="width:30px;">Cantidad</th>
    <th style="width:30px;">Unidad</th>
    <th>Producto</th>
    <th style="width:30px;">Precio Unitario</th>
    <th style="width:30px;">Precio Total</th>
    <th ></th>
</thead>
<?php foreach($_SESSION["cart"] as $p):
$product = ProductData::getById($p["product_id"]);
?>
<tr >
    <td><?php echo $product->id; ?></td>
    <td ><?php echo $p["q"]; ?></td>
    <td><?php echo $product->unit; ?></td>
    <td><?php echo $product->name; ?></td>
    <td><b>$ <?php echo number_format($product->price_out);
?></b></td>
    <td><b>$ <?php $pt = $product->price_out*$p["q"];
$total += $pt; echo number_format($pt); ?></b></td>
    <td style="width:30px;"><a
href="index.php?view=clearcart&product_id=<?php echo $product-
>id; ?>" class="btn btn-danger"><i class="glyphicon glyphicon-
remove"></i> Cancelar</a></td>
</tr>

<?php endforeach; ?>
</table>
<form method="post" class="form-horizontal" id="processsell"
action="index.php?view=processsell">
<h2>Resumen</h2>
<div class="form-group">
    <label for="inputEmail1" class="col-lg-2 control-
label">Cliente</label>
    <div class="col-lg-10">
        <?php
$clients = PersonData::getClients();
?>
        <select name="client_id" class="form-control">
            <option value="">-- NINGUNO --</option>
            <?php foreach($clients as $client):?>
                <option value="<?php echo $client->id;?>"><?php echo
$client->name." ". $client->lastname;?></option>
            <?php endforeach;?>
        </select>
    </div>
</div>
<div class="form-group">
    <label for="inputEmail1" class="col-lg-2 control-
label">Descuento</label>
    <div class="col-lg-10">
```

```
<input type="text" name="discount" class="form-control"
required value="0" id="discount" placeholder="Descuento">
</div>
</div>
<div class="form-group">
<label for="inputEmail1" class="col-lg-2 control-
label">Efectivo</label>
<div class="col-lg-10">
<input type="text" name="money" required class="form-
control" id="money" placeholder="Efectivo">
</div>
</div>
<input type="hidden" name="total" value="<?php echo
$total; ?>" class="form-control" placeholder="Total">

<div class="row">
<div class="col-md-6 col-md-offset-6">
<table class="table table-bordered">
<tr>
</tr>
<tr>
</tr>
<tr>
<td><p>Total</p></td>
<td><p><b>$ <?php echo number_format($total);
?></b></p></td>
</tr>
</table>
<div class="form-group">
<div class="col-lg-offset-2 col-lg-10">
<div class="checkbox">
<label>
<input name="is_oficial" type="hidden" value="1">
</label>
</div>
</div>
</div>
<div class="form-group">
<div class="col-lg-offset-2 col-lg-10">
<div class="checkbox">
<label>
<a href="index.php?view=clearcart" class="btn btn-
lg btn-danger"><i class="glyphicon glyphicon-remove"></i>
Cancelar</a>
<button class="btn btn-lg btn-primary"><i
class="glyphicon glyphicon-usd"></i><i class="glyphicon
glyphicon-usd"></i> Finalizar Venta</button>
</label>
</div>
</div>
</div>
```

```

</form>
<script>
    $("#processsell").submit(function(e) {
        discount = $("#discount").val();
        money = $("#money").val();
        if(money<(<?php echo $total;?>-discount)){
            alert("No se puede efectuar la operacion");
            e.preventDefault();
        }else{
            if(discount==""){ discount=0;}
            go = confirm("Cambio: $" + (money-(<?php echo
            $total;?>-discount ) ) );
            if(go){
                else{e.preventDefault();}
            }
        }
    });
</script>
</div>
</div>

<br><br><br><br><br>
<?php endif; ?>

</div>

```

2.09 Interfaz Lista de Ventas

Visualización de la Lista de Ventas.

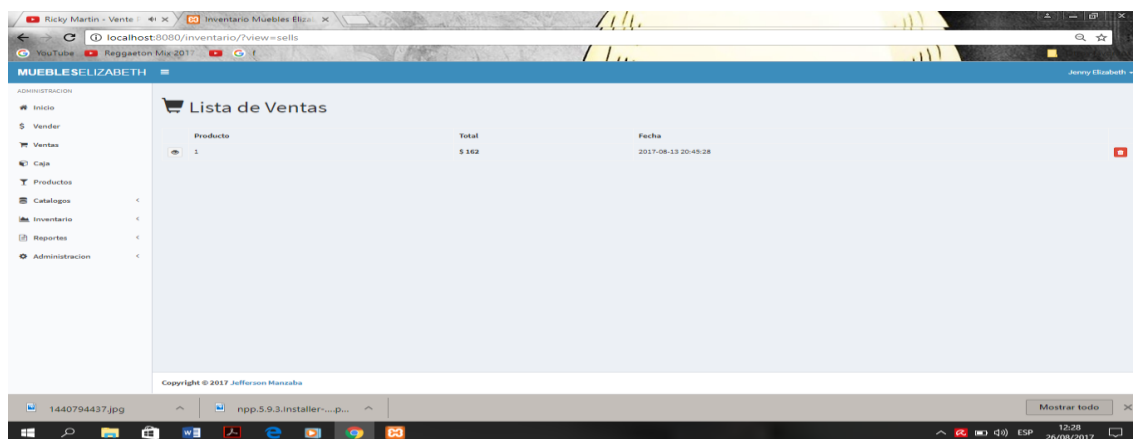


Figura 85. Lista de Ventas.

Código Fuente

```

<div class="row">
    <div class="col-md-12">
        <h1><i class='glyphicon glyphicon-shopping-
        cart'></i> Lista de Ventas</h1>
        <div class="clearfix"></div>

```



```
<?php

$products = SellData::getSells();

if(count($products)>0){

    ?>
    <br>
    <table class="table table-bordered table-hover  ">
        <thead>
            <th></th>
            <th>Producto</th>
            <th>Total</th>
            <th>Fecha</th>
            <th></th>
        </thead>
        <?php foreach($products as $sell):?>

            <tr>
                <td style="width:30px;">
                    <a href="index.php?view=onesell&id=<?php echo
$sell->id; ?>" class="btn btn-xs btn-default"><i
class="glyphicon glyphicon-eye-open"></i></a></td>

                <td>

<?php
$operations = OperationData::getAllProductsBySellId($sell-
>id);
echo count($operations);
?>

                <td>

<?php
$total= $sell->total-$sell->discount;
/*foreach($operations as $operation){
    $product = $operation->getProduct();
    $total += $operation->q*$product->price_out;
}*/
    echo "<b>$ ".number_format($total)."</b>";

?>

                </td>
                <td><?php echo $sell->created_at; ?></td>
                <td style="width:30px;"><a
href="index.php?view=delsell&id=<?php echo $sell->id; ?>"
class="btn btn-xs btn-danger"><i class="fa fa-
trash"></i></a></td>
            </tr>
        <?php endforeach; ?>
```

```
</table>

<div class="clearfix"></div>

    <?php
}else{
    ?>
    <div class="jumbotron">
        <h2>No hay ventas</h2>
        <p>No se ha realizado ninguna venta.</p>
    </div>
    <?php
}

?>
<br><br><br><br><br><br><br><br><br>
    </div>
</div>
```

2.10 Interfaz Caja

Visualización Caja.

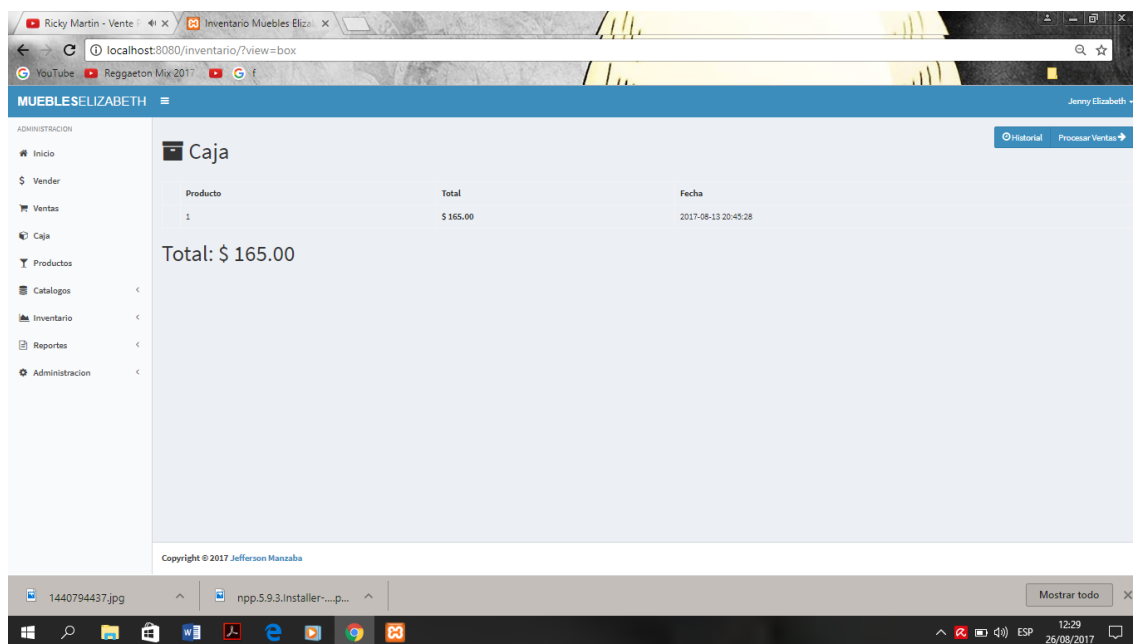


Figura 86. Caja.

Código Fuente

```
<div class="row">
    <div class="col-md-12">
<div class="btn-group pull-right">
<a href="./index.php?view=boxhistory" class="btn btn-primary
"><i class="fa fa-clock-o"></i> Historial</a>
```

```
<a href="./index.php?view=processbox" class="btn btn-primary
">Procesar Ventas <i class="fa fa-arrow-right"></i></a>

</div>

<h1><i class='fa fa-archive'></i> Caja</h1>
<div class="clearfix"></div>

<?php
$products = SellData::getSellsUnBoxed();
if(count($products)>0){
$total_total = 0;
?>
<br>
<table class="table table-bordered table-hover  ">
    <thead>
        <th></th>
        <th>Producto</th>
        <th>Total</th>
        <th>Fecha</th>
    </thead>
    <?php foreach($products as $sell):?>

        <tr>
            <td style="width:30px;">

                <td>

<?php
$operations = OperationData::getAllProductsBySellId($sell-
>id);
echo count($operations);
?>
                <td>

<?php
$total=0;
    foreach($operations as $operation){
        $product = $operation->getProduct();
        $total += $operation->q*$product->price_out;
    }
    $total_total += $total;
    echo "<b>$
.number_format($total,2,".",",")."</b>";
?>

                </td>
                <td><?php echo $sell->created_at; ?></td>
        </tr>

<?php endforeach; ?>
</table>
```

```
<h1>Total: <?php echo "$
".number_format($total_total,2,".",","); ?></h1>
<?php
}else {

?>

<div class="jumbotron">
    <h2>No hay ventas</h2>
    <p>No se ha realizado ninguna venta.</p>
</div>

<?php } ?>
<br><br><br><br><br><br><br>
</div>
</div>
```

2.11 Interfaz Inventario de Productos

Vista inventario de Productos.

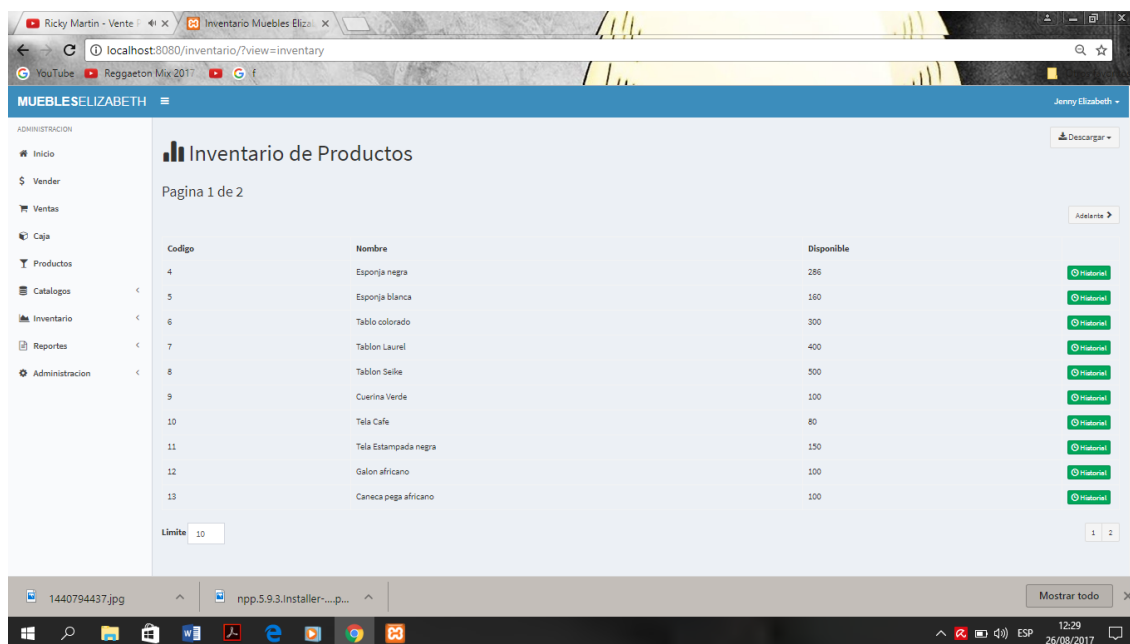


Figura 87. Inventario Productos.

Código Fuente

```
<div class="row">
    <div class="col-md-12">
<!-- Single button -->
<div class="btn-group pull-right">
    <button type="button" class="btn btn-default dropdown-
toggle" data-toggle="dropdown">
        <i class="fa fa-download"></i> Descargar <span
class="caret"></span>
```

```
</button>
<ul class="dropdown-menu" role="menu">
  <li><a href="report/inventory-word.php">Word 2007
(.docx)</a></li>

</ul>
</div>

<h1><i class="glyphicon glyphicon-stats"></i>
Inventario de Productos</h1>
<div class="clearfix"></div>

<?php
$page = 1;
if(isset($_GET["page"])){
    $page=$_GET["page"];
}
$limit=10;
if(isset($_GET["limit"]) && $_GET["limit"]!="") &&
$_GET["limit"]!=$limit){
    $limit=$_GET["limit"];
}
$products = ProductData::getAll();
if(count($products)>0){

if($page==1){
    $curr_products = ProductData::getAllByPage($products[0]-
>id,$limit);
}else{
    $curr_products = ProductData::getAllByPage($products[ ($page-
1)*$limit]->id,$limit);
}

$npaginas = floor(count($products)/$limit);
$spaginas = count($products)%$limit;

if($spaginas>0){ $npaginas++;}

    ?>

    <h3>Pagina <?php echo $page." de ".$npaginas; ?></h3>
<div class="btn-group pull-right">
<?php
$px=$page-1;
if($px>0):
    ?>
    <a class="btn btn-sm btn-default" href="<?php echo
"index.php?view=inventory&limit=$limit&page=".( $px); ?>"><i
class="glyphicon glyphicon-chevron-left"></i> Atras </a>
<?php endif; ?>

<?php
$px=$page+1;
if($px<=$npaginas):
    ?>
```

```
<a class="btn btn-sm btn-default" href="<?php echo
"index.php?view=inventory&limit=$limit&page=" . ($px);
?>">Adelante <i class="glyphicon glyphicon-chevron-
right"></i></a>

<?php endif; ?>
</div>
<div class="clearfix"></div>
<br><table class="table table-bordered table-hover">
    <thead>
        <th>Codigo</th>
        <th>Nombre</th>
        <th>Disponible</th>
        <th></th>
    </thead>
    <?php foreach($curr_products as $product):
        $q=OperationData::getQYesF($product->id);
        ?>
        <tr class="<?php if($q<=$product->inventory_min/2){ echo
"danger"; }else if($q<=$product->inventory_min){ echo
"warning"; }?>">
            <td><?php echo $product->id; ?></td>
            <td><?php echo $product->name; ?></td>
            <td>

                <?php echo $q; ?>

            </td>
            <td style="width:93px;">
                <!--
                <a href="index.php?view=input&product_id=<?php echo
$product->id; ?>" class="btn btn-xs btn-primary"><i
class="glyphicon glyphicon-circle-arrow-up"></i> Alta</a>-->
                <a href="index.php?view=history&product_id=<?php
echo $product->id; ?>" class="btn btn-xs btn-success"><i
class="glyphicon glyphicon-time"></i> Historial</a>
            </td>
        </tr>
    <?php endforeach; ?>
</table>
<div class="btn-group pull-right">
<?php

for($i=0;$i<$npaginas;$i++){
    echo "<a
href='index.php?view=inventory&limit=$limit&page=" . ($i+1) . "'
class='btn btn-default btn-sm'>". ($i+1) . "</a> ";
}
?>
</div>
<form class="form-inline">
    <label for="limit">Limite</label>
    <input type="hidden" name="view" value="inventory">
```

```
<input type="number" value=<?php echo $limit?>
name="limit" style="width:60px;" class="form-control">
</form>

<div class="clearfix"></div>

<?php
}else{
    ?>
    <div class="jumbotron">
        <h2>No hay productos</h2>
        <p>No se han agregado productos a la base de datos,
puedes agregar uno dando click en el boton <b>"Agregar
Producto"</b>.</p>
    </div>
    <?php
}

?>
<br><br><br><br><br><br><br><br><br>
    </div>
</div>
```

2.12 Interfaz Reabastecer Inventario

Reabastecimiento del Inventario.

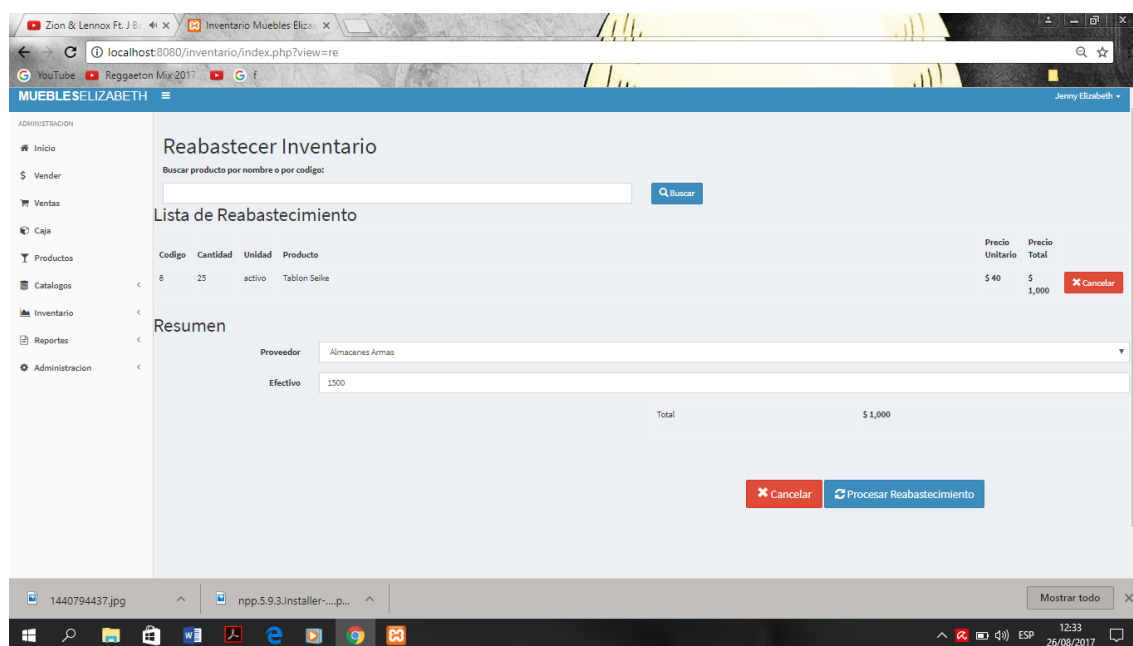


Figura 88. Reabastecimiento Inventario.

Código Fuente

```
<div class="row">
```

```
<div class="col-md-12">
<h1>Reabastecer Inventario</h1>
<p><b>Buscar producto por nombre o por codigo:</b></p>
    <form>
        <div class="row">

            <div class="col-md-6">
                <input type="hidden" name="view"
value="re">
                <input type="text" name="product"
class="form-control">
            </div>
            <div class="col-md-3">
                <button type="submit" class="btn btn-
primary"><i class="glyphicon glyphicon-search"></i>
Buscar</button>
            </div>
        </div>
    </form>
</div>

<?php if(isset($_GET["product"])):?>
    <?php
$products = ProductData::getLike($_GET["product"]);
if(count($products)>0){
    ?>
<h3>Resultados de la Búsqueda</h3>
<table class="table table-bordered table-hover">
    <thead>
        <th>Codigo</th>
        <th>Nombre</th>
        <th>Unidad</th>
        <th>Precio unitario</th>
        <th>En inventario</th>
        <th>Cantidad</th>
        <th style="width:100px;"></th>
    </thead>
    <?php
$products_in_cero=0;
    foreach($products as $product):
$q= OperationData::getQYesF($product->id);
    ?>
        <form method="post"
action="index.php?view=addtore">
            <tr class="<?php if($q<=$product->inventory_min){ echo
"danger"; }?>">
                <td style="width:80px;"><?php echo $product->id;
?></td>
                <td><?php echo $product->name; ?></td>
                <td><?php echo $product->unit; ?></td>
                <td><b>$<?php echo $product->price_in; ?></b></td>
                <td>
                    <?php echo $q; ?>

```



```
</td>
<td>
    <input type="hidden" name="product_id" value="<?php
echo $product->id; ?>">
    <input type="" class="form-control" required
name="q" placeholder="Cantidad de producto ..."></td>
    <td style="width:100px;">

        <button type="submit" class="btn btn-success"><i
class="glyphicon glyphicon-refresh"></i> Agregar</button>
    </td>
</tr>
</form>
<?php endforeach;?>
</table>
<?php
}
?>
<br><hr>
<hr><br>
<?php else:
?>

<?php endif; ?>

<?php if(isset($_SESSION["errors"])):?>
<h2>Errores</h2>
<p></p>
<table class="table table-bordered table-hover">
<tr class="danger">
    <th>Codigo</th>
    <th>Producto</th>
    <th>Mensaje</th>
</tr>
<?php foreach ($_SESSION["errors"] as $error):
$product = ProductData::getId($error["product_id"]);
?>
<tr class="danger">
    <td><?php echo $product->id; ?></td>
    <td><?php echo $product->name; ?></td>
    <td><b><?php echo $error["message"]; ?></b></td>
</tr>

<?php endforeach; ?>
</table>
<?php
unset($_SESSION["errors"]);
endif; ?>

<!-- Carrito de compras :) -->
<?php if(isset($_SESSION["reabastecer"])):
$total = 0;
```

```
?>
<h2>Lista de Reabastecimiento</h2>
<table class="table table-bordered table-hover">
<thead>
    <th style="width:30px;">Codigo</th>
    <th style="width:30px;">Cantidad</th>
    <th style="width:30px;">Unidad</th>

    <th>Producto</th>
    <th style="width:30px;">Precio Unitario</th>
    <th style="width:30px;">Precio Total</th>
    <th ></th>
</thead>
<?php foreach($_SESSION["reabastecer"] as $p):
$product = ProductData::getById($p["product_id"]);
?>
<tr >
    <td><?php echo $product->id; ?></td>
    <td ><?php echo $p["q"]; ?></td>
    <td><?php echo $product->unit; ?></td>
    <td><?php echo $product->name; ?></td>
    <td><b>$ <?php echo number_format($product->price_in);
?></b></td>
    <td><b>$ <?php $pt = $product->price_in*$p["q"]; $total
+=$pt; echo number_format($pt); ?></b></td>
    <td style="width:30px;"><a
href="index.php?view=clearre&product_id=<?php echo $product-
>id; ?>" class="btn btn-danger"><i class="glyphicon glyphicon-
remove"></i> Cancelar</a></td>
</tr>

<?php endforeach; ?>
</table>
<form method="post" class="form-horizontal" id="processsell"
action="index.php?view=processre">
<h2>Resumen</h2>
<div class="form-group">
    <label for="inputEmail1" class="col-lg-2 control-
label">Proveedor</label>
    <div class="col-lg-10">
        <?php
$clients = PersonData::getProviders();
?>
        <select name="client_id" class="form-control">
            <option value="">-- NINGUNO --</option>
            <?php foreach($clients as $client):?>
                <option value="<?php echo $client->id;?>"><?php echo
$client->name." ".$client->lastname;?></option>
            <?php endforeach;?>
        </select>
    </div>
</div>
<div class="form-group">
```

```
<label for="inputEmail1" class="col-lg-2 control-label">Efectivo</label>
<div class="col-lg-10">
    <input type="text" name="money" required class="form-control" id="money" placeholder="Efectivo">
</div>
</div>

<div class="row">
<div class="col-md-6 col-md-offset-6">
<table class="table table-bordered">
<tr>

    <td><p>Total</p></td>
    <td><p><b>$ <?php echo number_format($total);
?></b></p></td>
</tr>
</table>
<div class="form-group">
    <div class="col-lg-offset-2 col-lg-10">
        <div class="checkbox">
            <label>
                <input name="is_oficial" type="hidden" value="1">
            </label>
        </div>
    </div>
</div>
<div class="form-group">
    <div class="col-lg-offset-2 col-lg-10">
        <div class="checkbox">
            <label>
                <a href="index.php?view=clearre" class="btn btn-lg btn-danger"><i class="glyphicon glyphicon-remove"></i>
Cancelar</a>
                <button class="btn btn-lg btn-primary"><i class="fa fa-refresh"></i> Procesar Reabastecimiento</button>
            </label>
        </div>
    </div>
</div>
</form>
<script>
    $("#processsell").submit(function(e) {
        money = $("#money").val();
        if(money<<?php echo $total;?>){
            alert("No se puede efectuar la operacion");
            e.preventDefault();
        }else{
            go = confirm("Cambio: $" + (money-<?php echo $total;?>));
            if(go){}
            else{e.preventDefault();}
        }
    })
</script>
```

```

    });
</script>
</div>
</div>

<br><br><br><br><br>
<?php endif; ?>

</div>

```

2.13 Interfaz Vista Reabastecimiento

Vista Reabastecimiento.

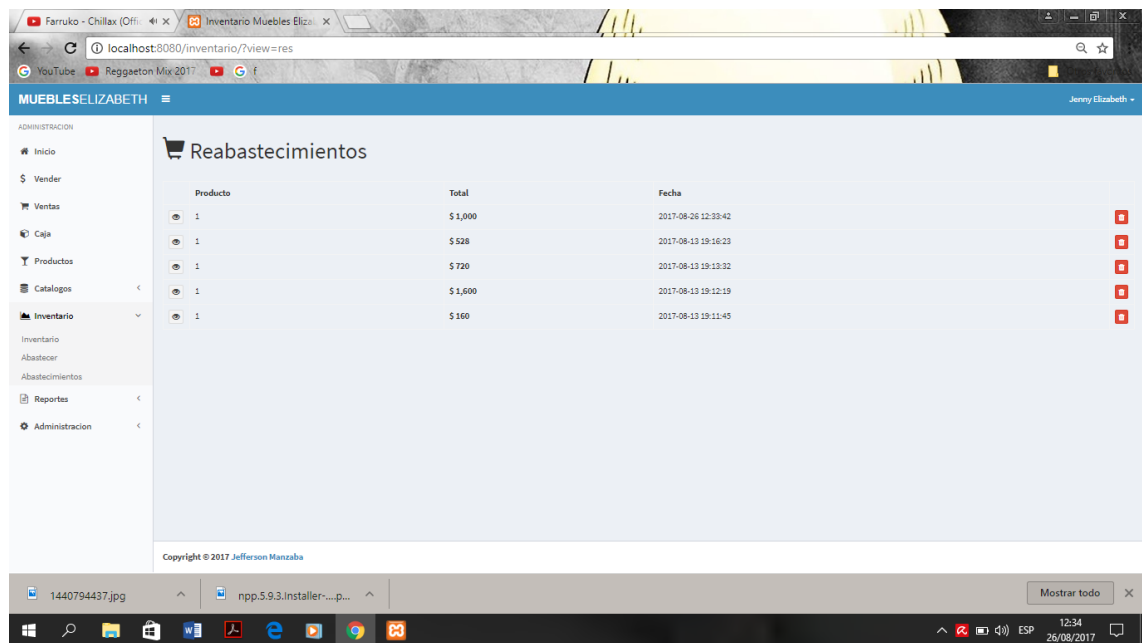


Figura 89. Vista Reabastecimiento.

Código Fuente

```

<div class="row">
    <div class="col-md-12">
        <h1><i class='glyphicon glyphicon-shopping-
        cart'></i> Reabastecimientos</h1>
        <div class="clearfix"></div>

        <?php
        $products = SellData::getRes();

        if(count($products)>0){
            ?>

            <br>
            <table class="table table-bordered table-hover">

```

```
<thead>
    <th></th>
    <th>Producto</th>
    <th>Total</th>
    <th>Fecha</th>
    <th></th>
</thead>

<?php foreach($products as $sell):?>

    <tr>
        <td style="width:30px;"><a
href="index.php?view=onere&id=<?php echo $sell->id; ?>"
class="btn btn-xs btn-default"><i class="glyphicon glyphicon-
eye-open"></i></a></td>

        <td>

<?php
$operations = OperationData::getAllProductsBySellId($sell-
>id);
echo count($operations);
?>

        <td>

<?php
$total=0;
foreach($operations as $operation){
    $product = $operation->getProduct();
    $total += $operation->q*$product->price_in;
}
    echo "<b>$ ".number_format($total)."</b>";

?>

        </td>
        <td><?php echo $sell->created_at; ?></td>
        <td style="width:30px;"><a
href="index.php?view=delre&id=<?php echo $sell->id; ?>"
class="btn btn-xs btn-danger"><i class="fa fa-
trash"></i></a></td>
    </tr>

<?php endforeach; ?>

</table>

<?php
}else{
    ?>
    <div class="jumbotron">
        <h2>No hay datos</h2>
```

```

        <p>No se ha realizado ninguna operacion.</p>
    </div>
    <?php
}

?>
<br><br><br><br><br><br><br><br><br>
</div>

</div>

```

2.14 Interfaz Reporte Inventario

Reporte Inventario.

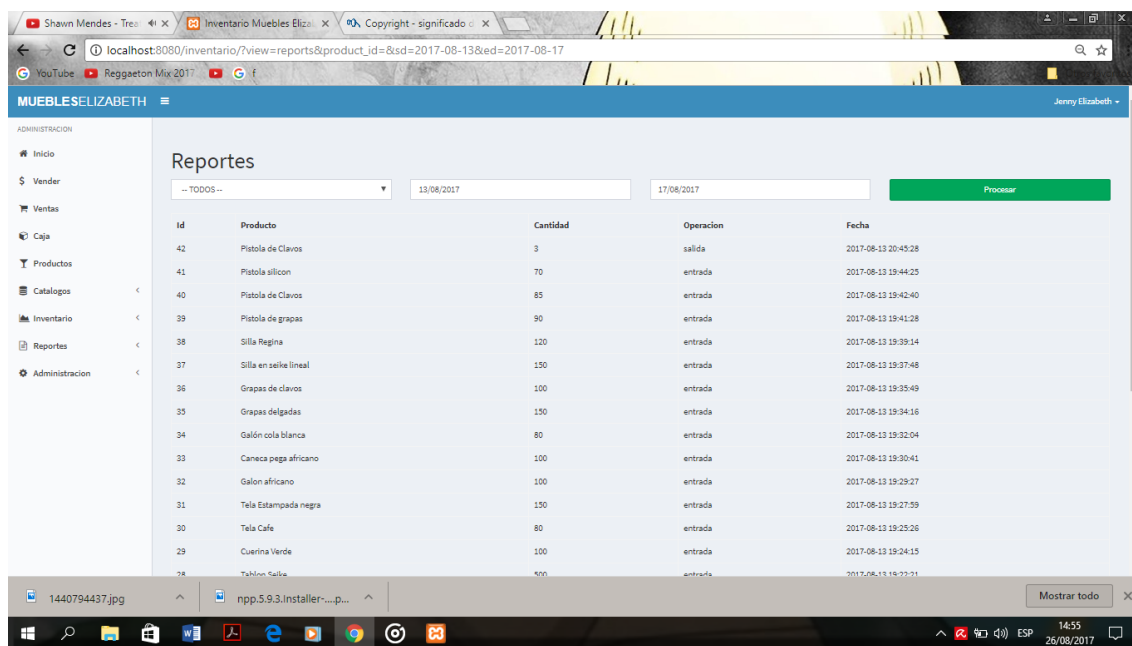


Figura 90. Reportes.

Código Fuente

```

<div class="row">
    <div class="col-md-12">
        <a href="index.php?view=addproduct" class="btn btn-
primary pull-right">Agregar Producto</a>
        <h1>Reporte por producto</h1>
        <p class="alert alert-success">Debe seleccionar un
producto para visualizar su reporte.</p>
        <div class="clearfix"></div>
    </div>
</div>

<?php
$page = 1;
if (isset($_GET["page"])) {

```

```
$page=$_GET["page"];
}
$limit=10;
if(isset($_GET["limit"]) && $_GET["limit"]!="") &&
$_GET["limit"]!=$limit){
    $limit=$_GET["limit"];
}

$products = ProductData::getAll();
if(count($products)>0){

if($page==1){
    $curr_products = ProductData::getAllByPage($products[0]-
    >id,$limit);
}else{
    $curr_products = ProductData::getAllByPage($products[($page-
    1)*$limit]->id,$limit);
}

$npaginas = floor(count($products)/$limit);
$spaginas = count($products)%$limit;

if($spaginas>0){ $npaginas++;}

    ?>

    <h3>Pagina <?php echo $page." de ".$npaginas; ?></h3>
<div class="btn-group pull-right">
<?php
$px=$page-1;
if($px>0):
?>
<a class="btn btn-sm btn-default" href="<?php echo
"index.php?view=products&limit=$limit&page=".( $px); ?>"><i
class="glyphicon glyphicon-chevron-left"></i> Atras </a>
<?php endif; ?>

<?php
$px=$page+1;
if($px<=$npaginas):
?>
<a class="btn btn-sm btn-default" href="<?php echo
"index.php?view=products&limit=$limit&page=".( $px);
?>">Adelante <i class="glyphicon glyphicon-chevron-
right"></i></a>
<?php endif; ?>
</div>
<div class="clearfix"></div>
<br><table class="table table-bordered table-hover">
    <thead>
        <th>Codigo</th>
        <th>Nombre</th>
```

```
<th></th>
</thead>
<?php foreach($curr_products as $product):?>
<tr>
    <td><?php echo $product->id; ?></td>
    <td><?php echo $product->name; ?></td>
    <td>
        <a href="index.php?view=productreport&id=<?php echo
$product->id; ?>" class="btn btn-primary"><i class="glyphicon
glyphicon-eye-open"></i> Ver Reporte</a>
    </td>
</tr>
<?php endforeach;?>
</table>
<div class="btn-group pull-right">
<?php

for($i=0;$i<$npaginas;$i++){
    echo "<a
href='index.php?view=products&limit=$limit&page=" . ($i+1) . "'
class='btn btn-default btn-sm'>". ($i+1) . "</a> ";
}
?>
</div>
<form class="form-inline">
    <label for="limit">Limite</label>
    <input type="hidden" name="view" value="products">
    <input type="number" value=<?php echo $limit?>
name="limit" style="width:60px;" class="form-control">
</form>

<div class="clearfix"></div>

<?php
}else{
    ?>
    <div class="jumbotron">
        <h2>No hay productos</h2>
        <p>No se han agregado productos a la base de datos,
puedes agregar uno dando click en el boton <b>"Agregar
Producto"</b>.</p>
    </div>
    <?php
}

?>
<br><br><br><br><br><br><br><br><br>
</div>
</div>
```


2.15 Interfaz Reporte Ventas

Reporte de Ventas.

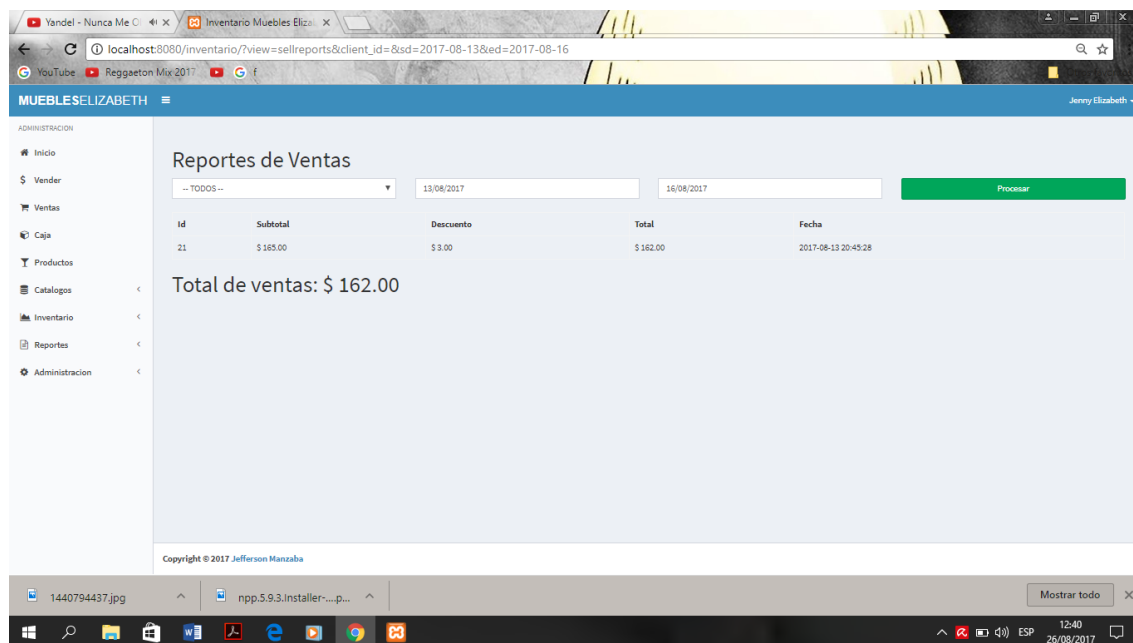


Figura 91. Reporte de Ventas.

Código Fuente

```
<?php
$clients = PersonData::getClients();
?>
<section class="content">
<div class="row">
    <div class="col-md-12">
        <h1>Reportes de Ventas</h1>

                                <form>
                                <input type="hidden"
name="view" value="sellreports">
<div class="row">
<div class="col-md-3">

<select name="client_id" class="form-control">
    <option value="">-- TODOS --</option>
    <?php foreach($clients as $p):?>
        <option value="<?php echo $p->id;?>"><?php echo $p-
>name;?></option>
        <?php endforeach; ?>
    </select>

</div>
```

```
<div class="col-md-3">
<input type="date" name="sd" value="<?php
if(isset($_GET["sd"])){ echo $_GET["sd"]; }?>" class="form-
control">
</div>
<div class="col-md-3">
<input type="date" name="ed" value="<?php
if(isset($_GET["ed"])){ echo $_GET["ed"]; }?>" class="form-
control">
</div>

<div class="col-md-3">
<input type="submit" class="btn btn-success btn-block"
value="Procesar">
</div>

</div>
<!--
<br>
<div class="row">
<div class="col-md-4">

<select name="mesero_id" class="form-control">
    <option value="">-- MESEROS --</option>
    <?php foreach($meseros as $p):?>
        <option value="<?php echo $p->id;?>"><?php echo $p-
>name;?></option>
    <?php endforeach; ?>
</select>

</div>

<div class="col-md-4">

<select name="operation_type_id" class="form-control">
    <option value="1">VENTA</option>
</select>

</div>

</div>
-->
</form>

    </div>
    </div>
<br><!-- -->
<div class="row">

    <div class="col-md-12">
        <?php if(isset($_GET["sd"]) && isset($_GET["ed"]))
):?>
```

```
<?php if($_GET["sd"]!=""&&$_GET["ed"]!="") :?>
    <?php
        $operations = array();

        if($_GET["client_id"]==""){
            $operations =
SellData::getAllByDateOp($_GET["sd"],$_GET["ed"],2);
        }
        else{
            $operations =
SellData::getAllByDateBCOp($_GET["client_id"],$_GET["sd"],$_GET["ed"],2);
        }

    ?>

    <?php if(count($operations)>0):?>
        <?php $supertotal = 0; ?>
<table class="table table-bordered">
    <thead>
        <th>Id</th>
        <th>Subtotal</th>
        <th>Descuento</th>
        <th>Total</th>
        <th>Fecha</th>
    </thead>
<?php foreach($operations as $operation):?>
    <tr>
        <td><?php echo $operation->id; ?></td>
        <td>$ <?php echo number_format($operation->total,2,'.',''); ?></td>
        <td>$ <?php echo number_format($operation->discount,2,'.',''); ?></td>
        <td>$ <?php echo number_format($operation->total-$operation->discount,2,'.',''); ?></td>
        <td><?php echo $operation->created_at; ?></td>
    </tr>
<?php
$supertotal+= ($operation->total-$operation->discount);
endforeach; ?>

</table>
<h1>Total de ventas: $ <?php echo
number_format($supertotal,2,'.',''); ?></h1>

    <?php else:
        // si no hay operaciones
    ?>

<script>
    $("#welcome").hide();
</script>
```

```
<div class="jumbotron">
    <h2>No hay operaciones</h2>
    <p>El rango de fechas seleccionado no proporciono ningun
resultado de operaciones.</p>
</div>

    <?php endif; ?>
<?php else:?>
<script>
    $("#wellcome").hide();
</script>
<div class="jumbotron">
    <h2>Fecha Incorrectas</h2>
    <p>Puede ser que no seleccione un rango de fechas, o el
rango seleccionado es incorrecto.</p>
</div>
<?php endif;?>

    <?php endif; ?>
</div>
</div>

<br><br><br><br>
</section>
```