



UNIVERSIDAD JOSÉ ANTONIO PÁEZ

**DESARROLLO DE SISTEMA WEB
PARA LA OPTIMIZACIÓN DEL
PROCESO DE ELABORACIÓN DE
CARNETS DE LA EMPRESA
CARNETIZA, C.A. UBICADA EN
VALENCIA ESTADO CARABOBO**

Autor: Emilio Clemente
C.I. V-21.280.375

Urb. Yuma II, Calle N° 3, Municipio San Diego
Teléfono: (0241) 8714240 (Master) - Fax: (0241) 871239



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

**DESARROLLO DE SISTEMA WEB PARA LA OPTIMIZACIÓN DEL
PROCESO DE ELABORACIÓN DE CARNETS DE LA EMPRESA
CARNETIZA, C.A. UBICADA EN VALENCIA ESTADO CARABOBO**

**Trabajo de grado presentado como requisito para optar al título de
INGENIERO DE COMPUTACIÓN**

Autor: Emilio Clemente
C.I. V-21.280.375
Tutor: Lcda. Soraya Aguilar

San Diego, Noviembre de 2015



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

ACEPTACIÓN DEL TUTOR

Quien suscribe, Soraya Aguilar portadora de la cédula de identidad N° 13.509.090, en mi carácter de tutor del trabajo de grado presentado por el ciudadano Emilio Clemente portador de la cédula de identidad N° 21.280.375, titulado **DESARROLLO DE SISTEMA WEB PARA LA OPTIMIZACIÓN DEL PROCESO DE ELABORACIÓN DE CARNETS DE LA EMPRESA CARNETIZA, C.A. UBICADA EN VALENCIA ESTADO CARABOBO** presentado como requisito parcial para optar al título de Ingeniero de Computación, considero que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del jurado examinador que se designe.

En San Diego, a los dos (2) días del mes de Noviembre del año dos mil quince.

Lcda. Soraya Aguilar
C.I: 13.509.090



**REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN**

San Diego, Junio de 2015

ACTA DE REVISIÓN METODOLÓGICA DEL TRABAJO DE GRADO

Quienes suscriben esta Acta, dejan constancia que el Proyecto de Trabajo de Grado: **DESARROLLO DE SISTEMA WEB PARA LA OPTIMIZACIÓN DEL PROCESO DE ELABORACIÓN DE CARNETS DE LA EMPRESA CARNETIZA, C.A.** ha sido revisado y, cumpliendo con los requisitos exigidos para su aprobación, recomiendan su tramitación ante el organismo académico correspondiente.

Lcda. Soraya Aguilar

Tutor Académico

Firma

Fecha

Ing. Alicia Pizzella

Tutor Metodológico

Firma

Fecha

ÍNDICE GENERAL

CONTENIDO	Pág.
ÍNDICE DE CUADROS.....	xi
ÍNDICE DE FIGURAS	xiv
RESUMEN.....	xv
INTRODUCCIÓN	1
CAPÍTULO	
I EL PROBLEMA	
1.1 Planteamiento del Problema.....	3
1.1.1 Formulación del Problema	5
1.2 Objetivos de la Investigación	5
1.2.1 Objetivo General	5
1.2.2 Objetivos Específicos	5
1.3 Justificación.....	5
II MARCO TEÓRICO	
2.1 Antecedentes	8
2.2 Bases Teóricas.....	11
2.2.1 Framework de Desarrollo Web	11
2.2.2 Ingeniería de Software.....	12
2.2.3 Manejador de Base de Datos	13
2.2.4 Metodología de Desarrollo XP	14
2.2.5 Sistema de Base de Datos.....	17
2.2.6 Sistema de Carnetización	20
2.2.7 Sistema de Información	21
2.2.8 Symfony	23
2.3 Bases Legales	24
2.4 Definición de Términos Básicos	25

III MARCO METODOLÓGICO

3.1 Tipo de Investigación	28
3.2 Nivel de la Investigación.....	29
3.3 Diseño de la Investigación	29
3.4 Población y Muestra.....	30
3.5 Fases Metodológicas	30
3.5.1 Analizar y Planificar.....	31
3.5.1.1 Historias de Usuario.....	31
3.5.1.2 Plan de Reuniones.....	31
3.5.1.3 Plan de Liberación	32
3.5.2 Diseñar.....	32
3.5.2.1 Diagrama de Casos de Uso	33
3.5.2.2 Diagrama de Actividades.....	33
3.5.2.3 Diagrama de Secuencia.....	33
3.5.2.4 Diseño de la Interfaz de Usuario.....	33
3.5.2.5 Mapa de Navegación.....	34
3.5.2.6 Modelo Entidad Relación.....	34
3.5.2.7 Modelo Lógico.....	34
3.5.3 Codificar	34
3.5.4 Probar	35

IV RESULTADOS

4.1 Fase I: Analizar	37
4.1.1 Historias de Usuario	37
4.1.2 Plan de Reuniones	41
4.1.3 Plan de Liberaciones.....	44
4.1.4 Definición de Requerimientos Funcionales y No Funcionales del Sistema	46
4.1.4.1 Requerimientos Funcionales	46

4.1.4.2	Requerimientos No Funcionales	47
4.2	Fase II: Diseñar	47
4.2.1	Diagrama de Casos de Uso.....	48
4.2.2	Diagrama de Actividades	59
4.2.2.1	Diagrama de Actividades del Cliente.....	59
4.2.2.2	Diagrama de Actividades del Empleado.....	61
4.2.2.3	Diagrama de Actividades del Administrador.....	61
4.2.3	Diagrama de Secuencia	62
4.2.3.1	Diagrama de Secuencia del Cliente	62
4.2.3.2	Diagrama de Secuencia del Empleado.....	63
4.2.3.3	Diagrama de Secuencia del Administrador	64
4.2.4	Diseño de la Interfaz de Usuario	65
4.2.4.1	Principios	65
4.2.4.2	Características	65
4.2.4.3	Elementos.....	67
4.2.4.4	Esquema de la Interfaz Web	68
4.2.4.5	Esquema de la Interfaz Móvil	69
4.2.5	Mapa de Navegación	70
4.2.6	Modelo Entidad Relación.....	71
4.2.7	Modelo Lógico	72
4.3	Fase III: Codificar	73
4.3.1	Framework.....	73
4.3.2	Bundles.....	73
4.3.3	Librerías Externas.....	74
4.3.4	Plugins	74
4.3.5	Pantallas del Sistema	75
4.4	Fase IV: Pruebas.....	80

V CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones 85

5.2 Recomendaciones 86

REFERENCIAS

Referencias Bibliográficas 88

Referencias en Línea 89

ÍNDICE DE CUADROS

CUADRO	Pág.
1 Historia de Usuario: Administración de clientes	38
2 Historia de Usuario: Administración de empresa	38
3 Historia de Usuario: Administración de diseño	38
4 Historia de Usuario: Edición de diseño.....	38
5 Historia de Usuario: Administración de empleados.....	39
6 Historia de Usuario: Administración de carnets	39
7 Historia de Usuario: Solicitud de pedido	39
8 Historia de Usuario: Visualización de catálogo	39
9 Historia de Usuario: Visualización de cotización	40
10 Historia de Usuario: Visualización de pedidos	40
11 Historia de Usuario: Impresión de carnets	40
12 Historia de Usuario: Exportación de carnets.....	40
13 Historia de Usuario: Generación de reportes	41
14 Historia de Usuario: Importación de datos.....	41
15 Plan de reuniones: Administración de clientes	42
16 Plan de reuniones: Administración de empresa	42
17 Plan de reuniones: Administración de diseño	42
18 Plan de reuniones: Edición de diseño.....	42
19 Plan de reuniones: Administración de empleados	42
20 Plan de reuniones: Administración de carnets	43
21 Plan de reuniones: Solicitud de pedido	43
22 Plan de reuniones: Visualización de catálogo	43
23 Plan de reuniones: Visualización de cotización	43
24 Plan de reuniones: Visualización de pedidos	43
25 Plan de reuniones: Impresión de carnets	44
26 Plan de reuniones: Exportación de carnets.....	44

27 Plan de reuniones: Generación de reportes	44
28 Plan de reuniones: Importación de datos	44
29 Plan de liberaciones: Primera liberación.....	45
30 Plan de liberaciones: Segunda liberación.....	45
31 Plan de liberaciones: Tercera liberación	45
32 Plan de liberaciones: Cuarta liberación.....	45
33 Plan de liberaciones: Quinta liberación.....	45
34 Plan de liberaciones: Sexta liberación.....	46
35 Plan de liberaciones: Séptima liberación	46
36 Plan de liberaciones: Octava liberación	46
37 Plan de liberaciones: Novena liberación	46
38 Descripción de Caso de Uso: Administrar usuario	49
39 Descripción de Caso de Uso: Administrar empresa.....	50
40 Descripción de Caso de Uso: Administrar carnet	51
41 Descripción de Caso de Uso: Administrar diseño	52
42 Descripción de Caso de Uso: Solicitar pedido.....	53
43 Descripción de Caso de Uso: Visualizar catálogo	53
44 Descripción de Caso de Uso: Importar datos carnet	54
45 Descripción de Caso de Uso: Visualizar cotización	55
46 Descripción de Caso de Uso: Visualizar pedido	55
47 Descripción de Caso de Uso: Imprimir carnets	56
48 Descripción de Caso de Uso: Exportar carnets	57
49 Descripción de Caso de Uso: Generar reportes.....	58
50 Paleta de Colores.....	67
51 Prueba 1: Solicitud de Pedido	81
52 Prueba 2: Edición de Diseño.....	81
53 Prueba 3: Administrar Empresa	82
54 Prueba 4: Visualización de Catalogo	82
55 Prueba 5: Cargar archivos	83

56 Prueba 6: Generar gráficas de pedidos..... 83
57 Prueba 7: Exportar carnets a PDF 84

ÍNDICE DE FIGURAS

FIGURA	Pág.
1 Diagrama de flujo de datos del sistema actual	37
2 Casos de Uso	48
3 Diagrama de Actividades: Cliente (1)	59
4 Diagrama de Actividades: Cliente (2)	59
5 Diagrama de Actividades: Cliente (3)	60
6 Diagrama de Actividades: Cliente (4)	60
7 Diagrama de Actividades: Empleado.....	61
8 Diagrama de Actividades: Administrador.....	61
9 Diagrama de Secuencia: Cliente	62
10 Diagrama de Secuencia: Empleado.....	63
11 Diagrama de Secuencia: Administrador.....	64
12 Logo de La Empresa	68
13 Esquema de la Interfaz Web	68
14 Esquema de la Interfaz Móvil	69
15 Mapa de Navegación.....	70
16 Modelo Entidad-Relación	71
17 Modelo Lógico	72
18 Pantalla del Sistema: Ingreso	75
19 Pantalla del Sistema: Base de Datos	76
20 Pantalla del Sistema: Edición de Diseño.....	76
21 Pantalla del Sistema: Administración de Campos	77
22 Pantalla del Sistema: Solicitud de Pedido.....	78
23 Pantalla del Sistema: Menú Móvil.....	79
24 Pantalla del Sistema: Empresas Móvil.....	79
25 Pantalla del Sistema: Diseño Móvil.....	80



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE COMPUTACIÓN

**DESARROLLO DE UNA APLICACIÓN WEB PARA LA OPTIMIZACIÓN
DEL PROCESO DE ELABORACIÓN DE CARNETS DE LA EMPRESA
CARNETIZA, C.A. UBICADA EN VALENCIA ESTADO CARABOBO**

Autor: Emilio Clemente

Tutor: Lcda. Soraya Aguilar

Fecha: Noviembre, 2015

RESUMEN

Los carnets de identificación son tarjetas que reflejan información relevante acerca del portador de esta. La empresa Carnetiza, C.A. ofrece el servicio de elaboración de estos carnets a distintas organizaciones, el presente proyecto trata del desarrollo de un sistema con una aplicación web y otra móvil el cual automatizó la transcripción de los datos del personal y redujo los errores cometidos al momento de ingresar dichos datos manualmente. Por otra parte, el sistema es capaz de aportar funcionalidades como la generación de reportes, estadísticas y visualizar el estado de los carnets solicitados. El proyecto fue basado en la metodología de desarrollo de software eXtreme Programming que se dividió en cuatro fases: análisis, diseño, codificación y pruebas, esto debido a su capacidad de adaptación a cambios y la liberación temprana del software. El sistema satisface tanto las necesidades diagnosticadas por la investigación que se llevó a cabo como las indicadas por los empleados de la empresa Carnetiza, C.A.

Descriptores: Carnets, Symfony, XP.

INTRODUCCIÓN

Actualmente las empresas a nivel mundial manejan gran cantidad de información, hasta el punto tal que esta no puede ser procesada en su totalidad de manera manual, sino que requieren de un sistema que automatice el flujo de dicha información y facilite los procesos que la involucran. En vista de estas necesidades, las empresas han cambiado la manera en que manejan la información, apoyándose en la creación de departamentos de informática con personal capacitado para analizar los aspectos que se buscan automatizar y en base a esto diseñar e implementar sistemas que satisfagan las necesidades de la empresa.

La implementación de sistemas de información en las empresas que manejen un volumen de información considerable ofrece beneficios tales como la disponibilidad de la información al momento que se necesite para una mejor gestión de procesos, adicionalmente simplifica el trabajo de almacenamiento, evitando la pérdida de documentos y reduciendo el uso innecesario de espacio para guardar papel caduco al hacer uso de bases de datos, por otra parte genera estadísticas y reportes de utilidad que apoyan la toma de decisiones, además de aumentar la calidad del servicio y producto de la empresa.

La elaboración del presente proyecto tuvo como finalidad implementar un sistema de información en la empresa Carnetiza, C.A. aportando todos los beneficios que estos ofrecen, de manera tal que se evitarán el uso de planillas impresas en papel, automatizando el proceso de transcripción de datos y se minimizarán los errores cometidos por el hombre. Por otra parte el sistema permite la creación de carnets, realizar solicitudes de impresión de los mismos y generar reportes del estado de dichas solicitudes, así como la previsualización de los carnets y la capacidad de realizar modificaciones sobre ellos en el momento que se requiera.

La estructura del presente trabajo consta de cinco capítulos explicados a continuación:

Capítulo I: El Problema. Se definieron la situación actual y se identificó y analizó el problema existente, adicionalmente se describieron los objetivos generales y específicos pautados para el trabajo, así como la justificación por la cual se realizó.

Capítulo II: Marco Teórico. Se presentaron los antecedentes, es decir, estudios realizados anteriormente por otros autores que posean características en común con el presente trabajo y que contribuyan a la elaboración del mismo, igualmente se establecieron los conceptos que se deben tener en cuenta a lo largo del proyecto y por último definición de términos básicos.

Capítulo III: Marco Metodológico. Se definieron el tipo, nivel y diseño de la investigación, así como la población y muestra y del mismo modo se especificaron las fases de la metodología *eXtreme Programming* la cual consta de cuatro fases: planificación, diseño, codificación y pruebas.

Capítulo IV: Resultados. En este capítulo se plasmaron los resultados de las diferentes fases que se aplicaron durante la elaboración del proyecto. Estos involucran el análisis, diseño codificación y pruebas del sistema.

Capítulo V: Conclusiones y Recomendaciones: En este quinto capítulo se mencionan los efectos que la investigación tuvo sobre la empresa para la cual fue realizado el trabajo, así como los aspectos en los que se propone hacer más énfasis posteriormente para un correcto mantenimiento del sistema y una continua mejora del mismo.

Por último, se reseñan las referencias bibliográficas que sirvieron de apoyo durante la realización del proyecto.

CAPÍTULO I

EL PROBLEMA

1.1. Planteamiento del Problema

En la actualidad las empresas se encuentran en constante crecimiento, por lo tanto, se han visto en la necesidad de optimizar y agilizar las labores que desempeñan para ofrecer la mejor calidad de productos y servicios. Mantener el orden en registros e información para utilizarla posteriormente (generar reportes, estadísticas, actualizar información, entre otros) es de vital importancia para administrar la empresa de manera conveniente y apoyar la toma de decisiones apropiadas para conseguir un desempeño adecuado.

En vista de estas necesidades, se han implementado métodos ampliamente utilizados en la actualidad que facilitan estas tareas, como por ejemplo los sistemas de bases de datos y los sistemas de información, que permiten almacenar y manipular toda la información a conveniencia con la finalidad de encaminar de la mejor manera a determinada empresa o compañía. De otro modo, manejar dicha información manualmente y sin la implementación de algún software que apoye esa labor, sería una tarea tediosa y poco eficiente.

Hoy en día, la empresa Carnetiza, C.A. se rige por el mismo sistema que ha venido utilizando durante dieciséis años, el cual consta de un software de escritorio que permite la creación de carnets. Al momento de realizar una carnetización, la empresa recibe el listado con los datos (nombres, cédulas, cargos, áreas, gerencias, códigos, fechas de ingreso y vencimiento, entre otros) de los carnets a realizar, posteriormente un empleado debe transcribir dichos datos manualmente, lo cual se convierte en un trabajo engorroso al momento de manejar un listado amplio.

Debido al notorio crecimiento de los clientes y pedidos que esta empresa maneja, el sistema actual no se encuentra en la capacidad de garantizar un óptimo servicio, lo que podría acarrear problemas en los carnets como nombres mal escritos, cédulas incorrectas, fechas erradas, entre otros. Esto a su vez conlleva a la reimpresión de cada carnet que presente algún error, consumiendo mayor cantidad de material y tiempo, lo cual se ve reflejado tanto económicamente como en la reducción de la calidad del servicio que se ofrece.

Por otra parte, el sistema vigente de la empresa no permite a los clientes tener control de su propia información como lo pueden ser los carnets que ha realizado, los que se encuentran por realizar, estado de solicitudes, fecha de creación y vencimiento, entre otros reportes de interés. Además el sistema no cuenta con la opción de que los clientes elaboren sus propios carnets en formato digital a través de internet, lo que implica que cualquier operación que necesiten realizar se vean en la obligación de dirigirse a las instalaciones de la empresa o realizar una solicitud por correo.

La principal consecuencia de continuar utilizando el sistema actual es la reducción en la calidad del servicio, puesto que el tiempo estimado de entrega es alto ya que es difícil de determinar para los empleados cuáles son los pedidos que se encuentran por realizar, y cuales fueron elaborados ya, al igual que la probabilidad de cometer errores en transcripción de datos, lo que supondría un problema en el desempeño. Por lo tanto, la empresa se ve afectada económicamente debido al mayor consumo de material y tiempo, que los empleados podrían invertir en otros procesos que aporten mayores beneficios.

Este proyecto buscó automatizar y optimizar los procesos realizados además de establecer el orden mencionado en dicha empresa de carnetización, la cual ofrece el servicio principalmente al personal de otras empresas, de ahí el hecho de enfocarse en manejar la información necesaria de las empresas con su respectivo personal para brindar la individualidad que amerita cada cliente. Teniendo este aspecto en cuenta, se planteó establecer una aplicación web y una aplicación móvil, que permitió ofrecer un servicio de carnetización óptimo.

1.1.1. Formulación del Problema

¿De qué manera se puede optimizar el proceso de elaboración de carnets de identificación en la empresa Carnetiza, C.A.?

1.2. Objetivos de la Investigación

1.2.1. Objetivo General

Desarrollar un sistema web para optimizar el proceso de elaboración de carnets en la empresa Carnetiza, C.A.

1.2.2. Objetivos Específicos

- Analizar los requisitos de cada empresa o compañía cliente en función de la información que desee plasmar en cada carnet.
- Diseñar un sistema web que permita a cada cliente gestionar su propio proceso de carnetización.
- Codificar el sistema en base a los requerimientos fijados por la empresa y el diseño del mismo.
- Efectuar las pruebas pertinentes para garantizar el correcto desempeño y la estabilidad del sistema en base a todos los requerimientos.

1.3. Justificación

Actualmente los sistemas de información constituyen una parte importante para alcanzar el éxito a nivel empresarial, han venido obteniendo mayor relevancia los últimos años debido a la capacidad que poseen de optimizar las actividades diarias, aumentar el desempeño y la productividad, mejorar la eficiencia en el servicio y la calidad del producto y minimizar los errores cometidos por el hombre. Además de ofrecer una gran cantidad de funciones que apoyan la toma de decisiones y el rendimiento del personal en las labores cotidianas, mejorando así el ambiente de trabajo en la empresa.

Partiendo de lo mencionado anteriormente, se puede articular que la implementación de un sistema de información que cuente con las utilidades que estos brindan y que permita a cada cliente gestionar y tener control de su propio proceso de carnetización, trajo un gran beneficio tanto para la empresa como para los clientes, ya

que la empresa se abstraigo de la engorrosa labor de transcribir gran cantidad de datos manualmente, y el cliente tiene la posibilidad de dirigir el proceso de carnetización a su conveniencia, traduciéndose en una ventaja competitiva en el campo.

Además de los beneficios principales mencionados, la implementación de dicho sistema brindó otras utilidades adicionales para la empresa, como la capacidad de generar reportes y estadísticas hechas a la medida que permitan visualizar el desenvolvimiento de la misma día a día, indicando información útil como lo es la frecuencia con que cada cliente hace uso del sistema, cantidad de solicitudes por cliente y por período de tiempo, siendo esto material de provecho al momento de tomar alguna decisión o llevar a cabo alguna acción dentro de la empresa.

Adicionalmente, los clientes se ven beneficiados en ciertos aspectos, como la capacidad de crear su propio diseño de carnet, teniendo la posibilidad de elegir el fondo de este y los campos que irán ahí reflejados y ubicarlos a conveniencia, de manera que se obtenga un diseño organizado y elaborado a la medida que satisfaga las necesidades del cliente. Así mismo pueden crear carnets ingresando los datos en el momento en que lo necesiten sin necesidad de esperar a la respuesta de una solicitud a la empresa, para posteriormente enviarlos a imprimir y recibir sus carnets listos.

Por otra parte, la correcta utilización del sistema redujo significativamente los problemas causados por los errores de transcripción que pueda cometer cada empleado, evitando así la necesidad de reimprimir cada carnet errado y la posible pérdida de material y dinero en un proceso de carnetización, puesto que cada cliente se haría cargo de suministrar los datos correspondientes a cada carnet, con la posibilidad de la revisión de estos mediante una previsualización que permita corregir los errores existentes antes de solicitar la impresión.

Finalmente, se pudo notar un significativo beneficio para la empresa a mediano plazo como lo es el aumento de los productos elaborados en menor cantidad de tiempo, debido a que los empleados no atienden solicitudes muy extensas ni los clientes esperan a ser atendidos durante un largo período de tiempo, traduciéndose en un aumento en el rendimiento y los ingresos de dicha empresa, además de que los

empleados podrán invertir el tiempo ahorrado en actividades que requieran de mayor atención por parte de estos.

CAPÍTULO II

MARCO TEÓRICO

Según Tamayo (2003) la teoría se define como “un conjunto de proposiciones lógicamente articuladas que tiene como fin la explicación y predicción de las conductas de un área determinada de fenómenos” (p. 145). De igual manera indica que “el marco teórico nos amplía la descripción del problema. Integra la teoría con la investigación y sus relaciones mutuas y tiene como fin ayudar a precisar y organizar los elementos contenidos en la descripción del problema” (p. 144).

2.1. Antecedentes

Como lo establece Tamayo (2003) “Todo hecho anterior a la formulación del problema que sirve para aclarar, juzgar e interpretar el problema planteado, constituye los antecedentes del problema” (p. 146). Una vez aclarado el concepto, y con la finalidad de aumentar la eficiencia y la calidad del presente trabajo, se tomaron en cuenta algunos estudios realizados anteriormente que poseen aspectos en común, para tomar como referencia información que sea de utilidad y que permitió orientar de la mejor manera el desarrollo del proyecto.

Mieres Angel, Buttó R. Yulianny, Castillo Jesús, Serrano Leoneidys y Guevara José (2012-2013) elaboraron una tesis en la Universidad Politécnica Territorial del Oeste de Sucre titulada: **“Implantación de un Sistema de Información Automatizado bajo ambiente web para el proceso de inscripción en la Escuela Bolivariana José Antonio Ramos Sucre Ubicada en la calle Zea, sector Río Manzanares, Parroquia Altigracia, Municipio Sucre, Cumaná, Edo Sucre 2012-2013”** en el cual se comprende la automatización del proceso de inscripciones de la Escuela Bolivariana mencionada y tiene como finalidad agilizar dicho proceso,

además de minimizar el tiempo de registro de los estudiantes, y de igual manera ayudar a que el personal administrativo no se vea involucrado en la tediosa tarea de inscribir a los alumnos de manera manual, sino que realice este proceso haciendo uso de un sistema automatizado.

El sistema planteado es de gran beneficio tanto para la comunidad como para la escuela, ya que les permite organizar y acceder de manera sencilla a los datos personales de cada alumno cuando se requiera dicha información. Debido al origen del problema que se busca solventar en la tesis mencionada, se pudo establecer una relación con el proyecto actual e identificar aspectos en común como la necesidad de registrar cierta cantidad de personas en el sistema de manera automatizada y haciendo uso de la tecnología web, para así agilizar el proceso y minimizar el tiempo de ingreso de los registros al sistema, al igual que el objetivo de abstraer al personal administrativo de la tarea de ingresar los datos manualmente, evitando cometer errores humanos. En este orden de ideas, el presente trabajo se apoyó en los parámetros establecidos y la metodología utilizada en el proceso referido para solucionar el problema en cuestión.

Por otra parte Noguera, Victoria (2013) en su trabajo de grado titulado **“Desarrollo de un sistema de información de soporte a partir de patrones de diseño utilizando Symfony en la empresa ICAD Diseño Inteligente, C.A.”** para optar por el título de Ingeniero en Computación en la Universidad José Antonio Páez, indica que el desarrollo del sistema en cuestión será apoyado por el framework de desarrollo web Symfony, con la finalidad de hacer uso de las herramientas y funciones que este ofrece y así automatizar las tareas comunes al momento de codificar y poder aportar mayor atención y dedicación a los aspectos específicos del sistema a desarrollar. De igual manera hace mención que dicho framework separa las reglas de negocio, las lógicas del servidor y las vistas de presentación, para así trabajar de manera más organizada y clara, además posee una gran variedad de clases enfocadas a reducir el tiempo de desarrollo de una aplicación web compleja debido a que no es necesario reescribir el código que ya se ha escrito.

La investigación mencionada realizó un aporte significativo para este trabajo debido a que hace mención de las distintas ventajas del framework que se utilizó para apoyar el desarrollo del sistema a tratar y cómo aprovechar las distintas herramientas y clases que se encuentran a disposición del programador con la finalidad de optimizar el proceso de desarrollo y obtener un sistema más acertado al que se desea, por otra parte explica el patrón de diseño a seguir modelo-vista-controlador y los distintos manejadores de bases de datos disponibles para el framework en cuestión.

Finalmente, Da Silva Almeida, Sergio Daniel (2013) realizó un trabajo de grado para optar por el título de Ingeniero en Computación en la Universidad José Antonio Páez titulado **“Desarrollo de un sistema de información para la gestión de los archivos inactivos en la empresa Distribuidora Universal Kia, C.A.”** en el cual establece que con la finalidad de cumplir con los objetivos del proyecto, basándose en las necesidades del cliente, la complejidad del sistema y el tiempo disponible se necesita de una metodología que permita el desarrollo organizado y eficiente del nuevo sistema, en base a esto indica que se hace uso de la metodología de desarrollo ágil conocida como XP (eXtreme Programming), de igual manera señala que esta metodología aporta ciertas ventajas como por ejemplo, prestar mayor atención en entregarle al cliente el software que requiere en el menor período de tiempo posible, involucrar al cliente y al programador juntos como grupo de trabajo en el desarrollo del software y modificar o integrar requisitos a medida que se requieran.

Dicho trabajo de grado aportó información valiosa para la realización del proyecto actual, por el hecho de que especificó los parámetros a seguir de la metodología, así como los valores que se deben cumplir al hacer uso de ella y las fases, las cuales son un conjunto de cuatro etapas que incluyen planificación, diseño, codificación y la elaboración de pruebas, de igual manera se pueden establecer ventajas y desventajas reflejadas en el trabajo mencionado, aprovechando esta información para identificar aspectos mejorables e implementarlos de manera óptima en el desarrollo del sistema.

2.2. Bases Teóricas

Según Bavaresco (2006) las bases teóricas tienen que ver con las teorías que brindan al investigador el apoyo inicial dentro del conocimiento del objeto de estudio. Con esto presente, a continuación se enunciaron los conceptos que se deben tener presentes para una correcta comprensión del trabajo.

2.2.1. Framework de Desarrollo Web

"Framework" es una palabra de origen inglés, que se traduce como "marco de trabajo". Potencier, F., Zaninotto F. (2007) estiman que un framework "simplifica el desarrollo de aplicaciones mediante la automatización de muchos de los patrones de diseño empleados para un propósito dado". Por otra parte, involucra la estructura de código, lo que sugiere al desarrollador realizar código más eficiente, más legible y más fácil de mantener y que a su vez pueda ser refactorizado y optimizado sin que esto implique realizar cambios en la funcionalidad del software.

Cuando se trata del desarrollo de software, un framework es un esquema conceptual y tecnológico debidamente soportado, casi en su totalidad integrados con artefactos o módulos de software definidos, que sirven de apoyo al momento de organizar y desarrollar el software. En la mayoría de los casos, contiene soporte de un lenguaje interpretado, de bibliotecas y de programas, además de otras herramientas útiles, como la creación de pruebas unitarias o las interfaces para depurar y de este modo facilitar la tarea de desarrollar y unir los diferentes segmentos que componen un proyecto.

Por otra parte, los frameworks tienen a disposición una gran variedad de herramientas que apoyan el desarrollo de sitios web dinámicos, las cuales cuentan con bibliotecas de clases que permiten establecer conexiones y accesos a bases de datos mediante un servidor, vinculando un manejador de bases de datos, y facilitando estructuras para plantillas, gestión de sesiones y usuarios y comúnmente facilitan la reutilización de código, lo que a su vez reduce el tiempo de codificación puesto a que dispone del código funcional en cualquier momento de la fase de desarrollo, además

de que permite la refacción del software, la cual consiste en reescribir el código sin alterar la funcionalidad del sistema con la finalidad de optimizarlo.

Teniendo en cuenta las facilidades que la utilización de un framework aporta al desarrollo de software, este fue de gran utilidad para el presente proyecto y permitió su realización en el tiempo establecido, de una manera correcta y organizada.

2.2.2. Ingeniería de Software

Con la finalidad de comprender de la mejor manera el concepto de ingeniería de software, se establecerá la definición fundamental de software, que según el autor Ian Sommerville en su publicación titulada “Ingeniería del Software” en su séptima edición, “el software son todos los documentos asociados y la configuración de datos que se necesitan para hacer que estos programas operen de manera correcta”. Una vez aclarada la definición de software que se estará manejando, se puede proceder al análisis del concepto de ingeniería de software, el cual el autor determina como “una disciplina de la ingeniería que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de este después de que se utiliza”.

Generalmente, los ingenieros de software hacen uso de un esquema sistemático y organizado al momento de trabajar, esto debido a que es la manera más efectiva de producir software competente y de alta calidad. Sin embargo, aunque lo estipulado es seleccionar el método más apropiado en base al entorno y las circunstancias en las que se trabaja, en algunas oportunidades un punto de vista más informal y creativo de desarrollo podría ser efectivo, específicamente cuando se trata de desarrollo de sistemas basados en web, que requieren una mezcla de técnicas de software y diseño gráfico.

Lo anteriormente mencionado implicó un aporte significativo para el presente proyecto, debido a que la utilización del mencionado esquema permitió trabajar de una mejor manera y en consecuencia obtener un software óptimo y eficiente el cual cumplió con las expectativas de la empresa.

Para apoyar el desarrollo de software bajo la ingeniería de software, existen herramientas denominadas herramientas CASE, las cuales son distintos programas utilizados para facilitar las actividades del proceso de software a lo largo de su ciclo de vida, como los son por ejemplo el análisis de requerimientos, el modelado de sistemas, la depuración y las pruebas. Actualmente la mayoría de los métodos para desarrollo de software vienen acompañados de tecnología CASE, como los son los editores para las notaciones utilizadas o los generadores de informes que apoyan la creación de la documentación del sistema. En algunos casos, las herramientas CASE ofrecen un generador de código fuente automático tomando como referencia el modelo del sistema.

2.2.3. Manejador de Base de Datos

Ramos, María; Ramos, Alicia y Montero Fernando en su libro titulado Sistemas Gestores de Bases de Datos publicado por la editorial Mcgraw Hill, define a estos sistemas como “una colección de datos relacionados entre sí, estructurados y organizados, y un conjunto de programas que acceden y gestionan esos datos” de igual manera indica que “el objetivo primordial de un gestor es proporcionar eficiencia y seguridad a la hora de extraer o almacenar información de las bases de datos”.

Estos sistemas están diseñados para gestionar bloques de información de una gran magnitud, que implican la definición de estructuras para el almacenamiento y de igual manera la de mecanismos para la gestión de la información, lo cual permite definir, crear y mantener la base de datos, y a su vez ofrece un acceso controlado a esta. Según los autores antes mencionados, un manejador de bases de datos debe prestar los siguientes servicios:

- **Creación y definición de la Base de Datos:** involucra la especificación de la estructura, de igual manera el tipo de los datos, las restricciones y relaciones entre ellos haciendo uso de lenguajes de definición de datos, cuya información se almacena en el diccionario de datos, el sistema ofrece mecanismos para la gestión del mencionado diccionario de datos.

- **Manipulación de los datos:** a través de la realización de consultas, ingresando nuevos datos o modificando los mismos haciendo uso de lenguajes que permitan la manipulación de datos.

- **Acceso controlado a los datos de la Base de Datos:** mediante mecanismos de seguridad de acceso a los usuarios.

- **Mantener integridad y consistencia:** haciendo uso de mecanismos que eviten que los datos se vean comprometidos por cambios no autorizados.

- **Acceso compartido a la Base de Datos:** controlando la interacción y entre los usuarios concurrentes.

- **Mecanismos de respaldo y recuperación:** que sea capaz de restablecer la información a un estado anterior, en caso de fallos en el sistema.

El lenguaje mayormente utilizado por los manejadores de bases de datos es SQL, un lenguaje que involucra tanto una parte de definición de datos y otra de manipulación de datos. Algunos ejemplos que se pueden conseguir en prácticamente cualquier ambiente de trabajo son los gestores: MySQL, Oracle, PostgreSQL, SQLite, SQL Server, entre otros.

La interfaz que un manejador de base de datos ofrece es el principal aporte para el actual proyecto, ya que facilitó las actividades que se realizaron con respecto a los datos del sistema, reduciendo el tiempo invertido en su manipulación y manteniendo la integridad y consistencia.

2.2.4. Metodología de Desarrollo XP

XP es una de varias metodologías ágiles que existen. Ya ha sido probado, siendo bastante satisfactorio en muchas compañías de distintos tamaños e industrias en todo el mundo, indican la organización encargada del proyecto en su página web oficial. Adicionalmente, precisan que XP es exitoso debido a que hace énfasis en la satisfacción del cliente, en vez de entregar todo lo que podría posiblemente querer en algún momento en el futuro, este proceso realiza el software que se necesita y como se necesita. XP permite a los programadores a responder a requerimientos del cliente cambiantes, incluso más adelante en el ciclo de vida.

XP hace énfasis en el trabajo en equipo. Los gerentes, los clientes, y los programadores son compañeros por igual. XP implementa un simple, pero efectivo ambiente permitiendo a los equipos ser altamente productivos. El equipo se organiza de manera automática ante el problema para solventarlo tan eficiente como sea posible. En la mencionada página web, se establecen ciertas reglas que hay que seguir para llevar a cabo esta metodología, estas son:

- **Planificación**

- Escribir historias de usuario.
- Al crear la planificación, se crea el calendario.
- Hacer pequeñas publicaciones frecuentemente.
- El proyecto está dividido en iteraciones.
- La planificación de las iteraciones comienzan en cada iteración
- Si una publicación no es aprobada, se hará en la siguiente iteración

- **Gerencia**

- Darle al equipo un espacio abierto de trabajo dedicado.
- Establecer un ritmo sustentable.
- Una reunión comienza cada día.
- La velocidad del proyecto es medida.
- Mantener a las personas en movimiento.
- Acomodar XP si falla.

- **Diseño**

- Simplicidad.
- Escoger una metáfora para el sistema.
- Usar tarjetas de clases, responsabilidades y colaboraciones (CRC) para sesiones de diseño.
- Crear soluciones potenciales exploradas por un programa que reduzcan el riesgo.
- Ninguna funcionalidad extra es agregada al principio.
- Pulir todo el tiempo y donde sea posible.

- **Codificación**

- El cliente siempre está disponible.
- El código debe ser escrito para satisfacer estándares.
- Codifica la prueba de unidad primero.
- Todo el código de producción es programado por una pareja.
- Solo una pareja a la vez integra código.
- Integrar código con frecuencia.
- Configurar una computadora dedicada a la integración.
- Usar propiedades colectivas.

- **Prueba**

- Todo el código debe tener pruebas de unidad.
- Todo el código debe pasar las pruebas de unidad antes de que puedan ser lanzadas.
- Cuando se consigue un error, se crean pruebas.
- Las pruebas aceptadas son ejecutadas a menudo y la calificación es publicada.

Adicionalmente a las reglas establecidas de las cuales hay que hacer seguimiento, se indican 5 valores los cuales hay que tomar en cuenta al momento de hacer uso de la metodología en cuestión.

- **Sencillez:** se hará lo que es necesitado y pedido, nada más.
- **Comunicación:** todos son parte del equipo, y se comunican en persona diariamente. Se trabajará juntos en todo, desde los requerimientos hasta el código.
- **Retroalimentación:** se tomará cada compromiso de cada iteración seriamente entregando software funcionando, se escuchará atentamente y se hará cualquier cambio necesitado.
- **Respeto:** todos aportan y sienten el respeto que merecen como un valioso miembro del equipo.

- **Esfuerzo:** se dirá la verdad acerca del progreso y las estimaciones. No se documentarán excusas por fallar porque se planifica para tener éxito. Se deberán adaptar a los cambios en cualquier momento que estos ocurran.

La utilización de la metodología XP, aportó al proyecto un desarrollo de software más ágil, en donde tanto el programador como el cliente trabajaron en conjunto, obteniendo así un sistema más acertado a las necesidades del cliente, y la minimización de trabajo extra puesto que el trabajo estuvo bien definido.

2.2.5. Sistema de Base de Datos

Según lo establece el autor C.J. Date en su publicación Introducción a los Sistemas de Bases de Datos “un sistema de base de datos es básicamente un sistema computarizado para guardar registros, cuya finalidad general es almacenar información y permitir a los usuarios recuperar y actualizar esa información con base en peticiones”, a continuación agrega “La información en cuestión puede ser cualquier cosa que sea de importancia para el individuo u organización”, con lo que se refiere a que un sistema de base de datos involucrará toda la información que sea de relevancia y que apoye en el proceso general de administración.

En base a la publicación antes mencionada, se puede establecer que un sistema de base de datos cuenta con entidades, “el término entidad es empleado comúnmente en los círculos de bases de datos para referirse a cualquier objeto distinguible que va a ser representado en esta” y del mismo modo cuenta con vínculos entre dichas entidades, cuya finalidad es definir en qué modo se relaciona cada una de ellas con las demás entidades existentes, aportándole así un sentido coherente y una representación del mundo real a la base de datos en cuestión.

Adicionalmente, cada una de las entidades cuenta con propiedades que competen a la información que se desea trabajar y registrar sobre ellas, y dichas propiedades deben estar representadas en la base de datos, como por ejemplo, una entidad llamada persona puede tener propiedades asociadas como: cédula, fecha de nacimiento, dirección, teléfono, género, y cualquier otra información requerida por el individuo u organización.

En el capítulo II de la publicación, se establece que existen dos arquitecturas en las que se pueden presentar los sistemas de base de datos, una de ellas es la arquitectura ANSI/SPARC y la otra es la arquitectura Cliente/Servidor. A continuación se definirá el funcionamiento de un sistema de base de datos que se rige bajo la primera arquitectura mencionada.

Se divide en tres niveles:

- **Nivel interno o nivel físico:** es el que está más cerca del almacenamiento físico; es decir, es el que tiene que ver con la forma en que los datos están almacenados físicamente.
- **Nivel externo o nivel lógico de usuario:** es el más próximo a los usuarios; es decir, el que tiene que ver con la forma en que los usuarios individuales ven los datos.
- **Nivel conceptual o nivel lógico de la comunidad:** es un nivel de indirección entre los otros dos.

La manera básica y conceptual en la que funciona un sistema de administración de base de datos (DBMS), el cual se encarga de manejar todo acceso a la base de datos bajo la arquitectura ANSI/SPARC, es la siguiente:

1. Un usuario emite una petición de acceso, utilizando algún sublenguaje de datos específico (por lo regular SQL).
2. El DBMS intercepta esa petición y la analiza.
3. El DBMS inspecciona, en su momento, el esquema externo para ese usuario, la transformación externa/conceptual correspondiente, el esquema conceptual, la transformación conceptual/interna y la definición de la estructura de almacenamiento.
4. El DBMS ejecuta las operaciones necesarias sobre la base de datos almacenada.

Mientras que en la arquitectura Cliente/Servidor un sistema de base de datos puede ser visto como un sistema que tiene una estructura muy sencilla de dos partes,

las cuales consisten en un servidor y un conjunto de clientes. A continuación se detallan cada una de estas partes.

- **El servidor** es básicamente el DBMS en sí. Soporta todas las funciones básicas del DBMS (definición de datos, manipulación de datos, seguridad e integridad de los datos, entre otros). En particular, proporciona todo el soporte de los niveles externo, conceptual e interno explicados anteriormente. Por lo tanto, se puede decir que el servidor es sólo el nombre del DBMS.
- **Los clientes** son las diversas aplicaciones que se ejecutan sobre el DBMS, tanto aplicaciones escritas por el usuario como aplicaciones integradas, las cuales usan la misma interfaz con el servidor.

Según lo indica Leonardo Víquez Acuña en su publicación “Estructura de Bases de Datos” un sistema de bases de datos facilita un lenguaje de definición de datos para especificar la estructura y el esquema de la base de datos y un lenguaje de manipulación de datos para establecer las consultas y las modificaciones a la base de datos. Aunque en la práctica ambos lenguajes se trabajan en conjunto y forman parte de un único lenguaje, como por ejemplo lo es SQL (Structured Query Language).

De lo anteriormente mencionado, se puede inferir que una base de datos es un conjunto de datos lógicamente coherentes y relacionados entre sí que establecen un significado y representan algún aspecto del mundo real, conocida como universo (del sistema), este universo puede sufrir modificaciones que se reflejan en dicha base de datos. Toda base de datos está diseñada y construida con un propósito específico, por lo tanto, posee ciertas aplicaciones que interesan al grupo de usuarios al cual está dirigida tal base de datos.

Esto significa un aporte tanto para la investigación como para el sistema de carnetización ya que toda la información en una base de datos está organizada y controlada de manera tal que los usuarios tengan la capacidad de agregar, buscar, modificar y eliminar datos cuando sea necesario, permitiéndole cumplir con las funciones principales de un sistema de información. Con frecuencia las bases de datos son tratadas a través de un manejador de base de datos.

2.2.6. Sistema de Carnetización

En base a la definición de sistema que Vicen Fernández Alarcón establece en su libro titulado “Desarrollo de Sistemas de Información. Una Metodología Basada en el Modelado” un sistema es “un conjunto de componentes que interaccionan entre sí para lograr un objetivo común” por lo que se puede aplicar dicho concepto al área de la carnetización y así deducir lo expresado a continuación.

Un sistema de carnetización se puede definir como un conjunto de herramientas y procesos que se relacionan entre sí para lograr un objetivo, mantener la organización correcta de los empleados de una empresa o compañía, estableciendo identificación personal, así como privilegios, jerarquías y grupos de manera tal que facilite el desempeño laboral en la institución. Adicional a esto, el carnet de identificación es requisito indispensable frente a las autoridades tanto del establecimiento como de la región en donde se desenvuelve dicho establecimiento.

Un sistema de carnetización puede presentarse de diversas formas, una de ellas es que la organización, empresa o compañía que desea carnetizar cuente con sus propias máquinas para la impresión de los carnets, además del material como las cintas y paneles de impresión y los consumibles como por ejemplo el plástico pvc donde se imprime, adicional a esto, recopilar los datos y fotografías organizadas y debidamente identificadas de todo el personal a carnetizar, bien sea tomadas en el establecimiento con una cámara digital o escaneadas o bien extraída de alguna base de datos existente. Y finalmente proceder con la realización del diseño del carnet, plasmar la información correspondiente, realizar el montaje de las fotos e imprimir.

Por otra parte, la empresa puede recurrir a un agente externo que ofrezca el servicio de carnetización, en donde se hacen cargo de estas tareas, ofrecen la creación del diseño, un fotógrafo que capture todas las fotos necesarias, y transcribir los datos del personal para finalmente proceder a la impresión y entrega de los carnets.

El conocimiento de este término representó un aporte a la presente investigación ya que se pudo comprender de mejor manera el funcionamiento y el

modo en que trabaja la empresa, para así desarrollar un software que se adaptara completamente a sus necesidades y los servicios que ofrecen.

2.2.7. Sistema de Información

Tal como lo define Vicen Fernández Alarcón en su publicación “Desarrollo de Sistemas de Información. Una Metodología Basada en el Modelado” un sistema es “un conjunto de componentes que interaccionan entre sí para lograr un objetivo común”. A pesar de que existen diversos sistemas, la mayoría de estos se pueden representar mediante un modelo o esquema formado por cinco secciones básicas, las cuales son: elementos de entrada, elementos de salida, sección de transformación, mecanismos de control y objetivos. Esencialmente el proceso que se lleva a cabo en un sistema es iniciado por el acceso de los recursos al sistema a través de los elementos de entrada para posteriormente ser modificados en la sección de transformación. A su vez este proceso es controlado por el mecanismo de control con la finalidad de lograr el objetivo establecido. Posteriormente, cuando se ha ejecutado la transformación, el resultado sale del sistema a través de los elementos de salida.

Adicionalmente expresa que a pesar de que hay un gran consenso en la definición de sistema, no existe tal para la definición de sistema de información. Actualmente, la expresión sistema de información es utilizada comúnmente y de manera habitual en las organizaciones; sin embargo, cada escuela y autor que trata dicho tema le otorga una connotación distinta. No obstante, partiendo de la definición dada de sistema, se puede realizar una definición bastante acertada “un sistema de información es un conjunto de componentes que interaccionan entre sí para lograr un objetivo común: satisfacer las necesidades de información de una organización.”, expresa Fernández en su libro.

Según el mencionado autor y en base a la definición de sistemas de información anteriormente mencionada propuesta por Whitten, Bentley y Dittman (2004). Dicha visión de sistemas de información propone diversos componentes que deben interactuar entre ellos para un correcto desarrollo del sistema de información.

A continuación, se exponen de manera resumida cada uno de estos componentes y sus relaciones como lo analiza el autor.

Los individuos participantes, es el componente más importante, está formado por las personas. Según Whitten, Bentley y Dittman (2004) todos los individuos que pueden y deben participar en el desarrollo de un sistema de información se pueden clasificar en función de la visión que tienen de un sistema de información. En este caso, la clasificación está formada por cinco grandes grupos:

- **Propietarios:** aquellas personas que patrocinan y promueven los sistemas de información.
- **Usuarios:** son aquellas personas que utilizan los sistemas de información de una forma regular para capturar, introducir, validar, transformar y almacenar datos e información.
- **Diseñadores:** son expertos en tecnología que resuelven las necesidades y las restricciones manifestadas por los usuarios de la empresa mediante recursos tecnológicos.
- **Constructores:** son los encargados de fabricar sistemas de información basados en las especificaciones de diseño obtenidas de los diseñadores de sistemas
- **Analistas:** son las personas que estudian los problemas y las necesidades de una empresa para determinar cómo podrían combinarse los recursos humanos, los procesos, los datos y la tecnología de la información para obtener mejoras en la empresa.
- **Project Manager:** profesional experimentado que acepta la responsabilidad de planificar, supervisar, y controlar proyectos en lo que concierne al calendario, el presupuesto, la satisfacción del cliente, las normas técnicas y la calidad del sistema

El segundo componente son los datos e información. Es muy común encontrar en el día a día en la calle, organizaciones y universidades, que se utiliza de igual manera los conceptos datos, información y conocimiento; sin embargo, no debe ser

de esta manera, puesto que no tienen el mismo significado y existe una gran diferencia entre ellos.

“Los datos consisten en hechos y cifras que tiene de algún modo una existencia propia e independiente y que tiene poco significado para el usuario” indica Vicen Fernández Alarcón, los datos tienen una característica propia, y es que por sí solos no indican si son relevantes o irrelevantes, puesto que es necesario establecerlos en un contexto definido.

Luego de procesar los datos existentes estos son transformados en información. “Se puede considerar la información como un conjunto de datos procesados con significado, y dotados de relevancia y propósito”, teniendo en cuenta esta definición, la percepción de los hechos ubicados en un contexto es la que indica finalmente si un dato (o conjunto de ellos) es relevante o no. Debido a la relevancia de los dos conceptos, tanto de dato como información, lo que es información para una persona puede ser simplemente datos para otra.

Poder visualizar las aplicaciones a desarrollar desde el punto de vista de un sistema de información facilitó el desarrollo de este, ya que se pudo identificar los elementos de entrada y de salida, así como la sección de transformación, los mecanismos de control y los objetivos, estructurando de mejor manera el sistema y trabajando cada elemento por separado para su posterior integración y así lograr un correcto funcionamiento en conjunto, representando esto un aporte para la investigación.

2.2.8. Symfony

En la actualidad existe una gran cantidad de frameworks, entre ellos se encuentra symfony que según Potencier, et al (2007) es “un framework completo diseñado para optimizar el desarrollo de aplicaciones web por medio de varias características clave”. Este framework se encuentra en la clasificación de los MVC, es decir los que trabajan bajo el esquema de Modelo (base de datos) - Vista (presentación al usuario) - Controlador (lógica de negocio). Está dotado de gran cantidad de herramientas cuya finalidad es básicamente facilitar el desarrollo de una

aplicación web compleja, automatizando las tareas comunes y reduciendo el tiempo de trabajo significativamente, lo que a su vez permite al desarrollador enfocarse en los requerimientos específicos de la aplicación.

Symfony está desarrollado completamente en PHP 5 y ha sido probado en diversos proyectos del mundo real reflejando un óptimo desempeño, en consecuencia, es usado ampliamente para la alta demanda de comercio electrónico y sitios web. Además posee una característica de gran importancia, es compatible con la mayoría de motores de bases de datos que se encuentran actualmente en uso, como por ejemplo MySQL, PostgreSQL, Oracle, SQL Server.

Por otra parte, Symfony trabaja bajo una estructura de directorios que conduce al desarrollador a escribir código más organizado, reusable y más fácil de mantener, de igual manera permite realizar cambios y mejoras en este sin la necesidad de realizar gran cantidad de modificaciones adicionales, que al final terminan perjudicando el desempeño de la aplicación. En el mismo orden de ideas, Symfony hace uso de lenguajes, motores y formatos integrados en sus aplicaciones que facilitan aún más la escritura de código, otorgándole un aspecto más agradable, elegante y legible. Algunos ejemplos de estos elementos son TWIG, YAML y Annotations, lo cual resulta siendo un aporte para el desarrollo del sistema.

2.3. Bases Legales

Las bases legales del presente proyecto se encuentran definidas entre las normas y reglamentos públicos, de las empresas a las que se ofrece servicio, y las políticas internas del establecimiento en cuestión (La empresa Carnetiza, C.A.), donde queda explícitamente indicado que cada individuo debe poseer una planilla de autorización firmada y sellada por la empresa en la que labora para la emisión y renovación del carnet, la misma debe estar vigente al momento de la solicitud del carnet.

Las carnetizaciones a institutos educativos, universidades y cuerpos policiales deben ser carnetizaciones completas, es decir, no se permite la elaboración de carnets a particulares. De igual manera, para emitir cualquier carnet, la empresa debe estar

registrada, para lo que debe contar con una fotocopia del registro de comercio, una fotocopia del RIF, y una carta de autorización firmada y sellada, autorizándole a Carnetiza, C.A. la emisión de carnets de la empresa.

En cuanto a políticas internas de cada empresa, se tratan por separado y con la individualidad que ameritan.

2.4. Definición de términos básicos

Ajax: acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web aunque es confundido por muchos como un lenguaje de programación, y tiene la finalidad de crear aplicaciones interactivas en donde interviene un servidor. Las aplicaciones que hacen uso de esta técnica son ejecutadas en el navegador del usuario, y a su vez mantiene una comunicación asíncrona con el servidor en segundo plano. La principal ventaja de esta técnica es que permite interactuar y realizar cambios sobre la aplicación sin recargarla, lo que se convierte en mayor velocidad de respuesta y usabilidad de la aplicación.

Annotation: Son metadatos (datos de datos) asociados al texto, imágenes u otros datos, frecuentemente las anotaciones hacen referencia a una parte específica de la información original.

Aplicación Web: Es aquella que los usuarios pueden utilizar accediendo a un servidor web a través de internet o de una intranet mediante un navegador y contiene información que puede ir desde texto, imágenes, sonido, etc., hasta un sistema completo. En otras palabras es una aplicación software que se codifica en un lenguaje soportado por los navegadores web (HTML, JavaScript, CSS, Java, PHP, etc) en la que se confía la ejecución del navegador.

Automatizar: Aplicar máquinas o procedimientos automáticos a un sistema, en la realización de un proceso o en una industria.

Dato: es una representación simbólica (numérica, alfabética, algorítmica, espacial, etc.) de un atributo o variable cuantitativa o cualitativa. Los datos describen hechos empíricos, sucesos y entidades. Es un valor o referente que recibe el computador por

diferentes medios, los datos representan la información que el programador manipula en la construcción de una solución o en el desarrollo de un algoritmo.

Entorno: Conjunto de circunstancias o factores sociales, culturales, morales, económicos, profesionales, etc., que rodean una cosa o a una persona, colectividad o época e influyen en su estado o desarrollo.

HTML: Del acrónimo *HyperText Markup Language*, que traducido al español significa lenguaje de marcas de hipertexto. Son documentos que contienen texto plano organizado mediante etiquetas y usan la extensión .html.

Información: Es un conjunto organizado de datos procesados, que constituyen un mensaje que cambia el estado de conocimiento del sujeto o sistema que recibe dicho mensaje.

JavaScript: Es un lenguaje de programación interpretado que se ejecuta del lado del cliente, es decir, en el navegador del usuario, aportando interactividad a la página, mejorando la interfaz de usuario y las páginas web dinámicas.

Metodología: Conjunto de métodos que se siguen en una investigación científica, un estudio o una exposición doctrinal.

Optimizar: Conseguir que algo llegue a la situación óptima o de los mejores resultados posibles.

PHP: Acrónimo de *Hypertext Preprocessor*. Es un lenguaje de programación interpretado utilizado en el desarrollo de aplicaciones y es ejecutado del lado del servidor.

Proceso: secuencia de pasos dispuesta con algún tipo de lógica que se enfoca en lograr algún resultado específico

Proyecto: Idea de una cosa que se piensa hacer y para la cual se establece un modo determinado y un conjunto de medios necesarios.

TWIG: Es un motor de plantillas para PHP el cual compila las plantillas escritas bajo su propio lenguaje, optimizándolas y evitando código no confiable. Ofrece la posibilidad de crear etiquetas personalizadas para facilitar su uso al desarrollador.

Web: es un vocablo inglés que significa “red”, “telaraña” o “malla”. El concepto se utiliza en el ámbito tecnológico para nombrar a una red informática y, en general, a Internet.

XML: Acrónimo de *eXtensible Markup Language*, lo cual traducido al español significa lenguaje de marcas generalizado, es un lenguaje usado para estructurar información en cualquier documento o fichero que contenga texto, incorporando etiquetas que contienen información adicional acerca de su presentación, como por ejemplo ficheros de configuración de un programa o una tabla de datos.

YAML: Acrónimo de *Ain't Markup Language*, es un formato de codificación de datos amigable, el cual es legible por humanos y su propósito está centrado en los datos en vez de el marcado de documentos.

CAPÍTULO III

MARCO METODOLÓGICO

Arias (2012) “La metodología del proyecto incluye el tipo o tipos de investigación, las técnicas y los instrumentos que serán utilizados para llevar a cabo la indagación” (p. 110), adicionalmente indica que “es el cómo se realizará el estudio para responder al problema planteado”.

En base a lo anteriormente mencionado, a continuación se describió el diseño de estudio, haciendo énfasis en el nivel de la investigación así como el propósito de la misma. Del mismo modo, se presentaron la población, las técnicas e instrumentos de recolección de datos y la validez, con la finalidad de dar respuestas de manera organizada y sistemática a las interrogantes planteadas. En última instancia, se enumeraron y describieron las fases a seguir establecidas por la metodología de desarrollo de software eXtreme Programming para el desarrollo del sistema.

3.1. Tipo de Investigación

Héctor Mijares (2007) en su publicación de las normas de trabajo de grado de la Universidad José Antonio Páez, indica que “Los proyectos especiales consistirán en las creaciones tangibles, susceptibles de ser utilizadas como soluciones a problemas demostrados, o que respondan a necesidades e intereses de tipo cultural”, además añade que en esta categoría se incluye el desarrollo de software y hardware y prototipos de productos tecnológicos en general, por lo que se puede articular que el presente es un proyecto de tipo especial, debido a su origen y propósito, el cual es el desarrollo de un software.

Bajo los esquemas del concepto anterior, esta investigación está enfocada a la necesidad que tenía la empresa Carnetiza, C.A. de contar con un sistema de información, en este caso aplicación web que apoye el sistema actual de carnetización

en dicha empresa, partiendo de los conocimientos básicos sobre cómo desarrollar un sistema de información.

3.2. Nivel de la Investigación

Arias (2012) menciona que, en el nivel de investigación “se indica el tipo de investigación según el nivel o grado de profundidad con el que se realizará el estudio” (p. 110). Agregándole a este concepto, la definición según el mismo autor Fideas G. Arias (2012) “La investigación descriptiva consiste en la caracterización de un hecho, fenómeno, individuo o grupo, con el fin de establecer su estructura o comportamiento. Los resultados de este tipo de investigación se ubican en un nivel intermedio en cuanto a la profundidad de los conocimientos se refiere.”

Tomando en cuenta el concepto anteriormente expuesto, se pudo determinar que el presente estudio se encuentra ubicado dentro de los parámetros de la investigación descriptiva puesto que plantea resolver problemas o intervenir en el desenvolvimiento natural del proceso de producción de carnets de la empresa Carnetiza, C.A. involucrando la innovación tecnológica y científica.

3.3. Diseño de la Investigación

Arias (1999) “El diseño de la investigación es la estrategia que adopta el investigador para responder al problema planteado”. Una vez aclarado el concepto general, y partiendo del análisis de los distintos diseños de investigación existentes, se puede determinar que el presente proyecto está basado en un estudio de campo que según el autor “la investigación de campo consiste en la recolección de datos directamente de la realidad donde ocurren los hechos, sin manipular o controlar variable alguna”.

En un principio, se evaluó y diagnosticó la situación que existía en la oficina de Carnetiza, C.A. la cual es objeto de estudio, con la finalidad de determinar las necesidades que ahí se generaban, para que posteriormente, a partir de los resultados del diagnóstico se formuló y diseñó el sistema de información que optimizaría el proceso de elaboración de carnets tomando en cuenta todas las especificaciones estipuladas.

3.4. Población y Muestra

Para que las estadísticas cobren sentido se consideran y relacionan dentro del contexto en el que se trabaja, por lo tanto, se debe establecer dicho contexto determinando la población y la muestra de la investigación. Morles (1994) “La población o universo se refiere al conjunto para el cual serán válidas las conclusiones que se obtengan: a los elementos o unidades (personas instituciones o cosas) involucradas en la investigación” (p. 17), por lo tanto, se pudo inferir que la población es la totalidad de individuos y objetos que conservan algunas características en común, mientras que la muestra según Morles (1994) “es un subconjunto representativo de un universo o población” (p. 17).

Según lo define Arias (2006), “El muestreo intencional es aquel donde los elementos muestrales son escogidos en base a criterios o juicios preestablecidos por el investigador”. En base a este concepto, se pudo establecer que para efectos de la investigación se realizó un muestreo intencional, ya que se posee un conocimiento previo de la población que se investigó, es decir, las empresas que buscan realizar sus respectivas carnetizaciones y se puede determinar los elementos más representativos.

En el caso de la empresa de carnetización, se utilizó una cantidad de población total de 20 personas, en donde se encuentran involucrados empleados encargados de la transcripción de datos, diseñadores gráficos, en cargados de impresión, personal de administración, entre otros.

3.5. Fases metodológicas

En función de cumplir con los objetivos planteados, se hizo uso de una metodología que permitió desarrollar el sistema en cuestión, en base a las necesidades del cliente, la complejidad que estas aporten al proyecto y el tiempo del que se dispone. Una vez mencionado esto, se puede considerar la metodología de desarrollo eXtreme Programming (XP) creada por Don Wells como una alternativa. A continuación se realizó un pequeño análisis en base a lo estipulado por la organización encargada de dicha metodología en su sitio web oficial.

XP se enfoca mayormente en prestar atención a las necesidades del cliente y desarrollarlo en el menor tiempo posible, además de involucrar al cliente y dar una mejor respuesta a los requisitos cambiantes durante la elaboración del sistema. Entre las ventajas que aporta esta metodología se encuentran las siguientes:

- Se logra el software que realmente necesita el usuario y que cumpla con las expectativas de este, puesto a que posee un mayor enfoque en cada uno de los requisitos del cliente.
- Se evitan en grandes cantidades posibles retrasos en la planificación.
- Realizar el desarrollo en el menor tiempo posible, lo que reduce el costo y mantenimiento del sistema.

Una vez referido esto, se puede determinar que dicha metodología es adecuada para la elaboración del sistema. Por otra parte, XP sugiere cuatro fases por las cuales se regirá el proyecto y se describen a continuación.

3.5.1. Analizar y Planificar

3.5.1.1. Historias de Usuario

El primer paso al momento de crear un nuevo software, es la planificación, esto se logra creando las historias de usuarios. Se usaron para crear estimaciones de tiempo para reuniones de planificación de liberación y para evitar un extenso documento de requerimientos.

Las historias de usuario fueron escritas por los clientes como cosas que el sistema necesita hacer para ellos. Cuando llegó el momento de implementar la historia se acudió al cliente para recibir una descripción detallada de los requerimientos en persona. Se mantuvo las historias enfocadas en las necesidades del usuario y sus beneficios contrariamente a especificar la disposición de la interfaz gráfica de usuario.

3.5.1.2. Plan de Reuniones

Posteriormente, luego de haber creado las historias de usuario, se acordó el plan de reuniones, que define el proyecto en general. La esencia del plan de reuniones es estimar cada historia de usuario en términos de semanas de programación ideal. El

cliente decidió qué historia es la más importante y tiene la mayor prioridad para ser completada.

3.5.1.3. Plan de Liberación

A partir de las reuniones, se creó un plan de liberación, el cual especificó las historias de usuario que fueron implementadas para cada liberación del sistema y las fechas de esas liberaciones.

Se hizo pequeñas liberaciones del sistema al cliente frecuentemente, al final de cada iteración se tuvo el software probado, funcionando y listo para mostrarlo a los clientes. Esto fue crítico para obtener una valiosa retroalimentación a tiempo y surtió efecto sobre el desarrollo del sistema.

3.5.2. Diseñar

En esta fase se realizó el diseño del sistema, de manera sencilla y clara para así reducir el tiempo en el momento de la codificación. Es posible realizar diagramas, siguiendo los principios de la metodología ágil, el cual indica que deben ser de utilidad para el proyecto y no deben expresar complejidad, sino que deben ser prácticos y que se realicen en poco tiempo. Una vez aclarado esto, para efectos de este proyecto, se utilizó el Lenguaje Unificado de Modelado (UML) en su segunda versión, el cual según se indica en su documentación oficial ofrece tres tipos de diagramas:

- **Diagramas de comportamiento:** representa características de comportamiento de un sistema o su lógica de negocio. Incluye actividad, máquina de estado, y utiliza diagramas de casos tanto como los cuatro diagramas de interacción.
- **Diagramas de interacción:** es un subconjunto de los diagramas de comportamiento, que enfatiza la interacción entre objetos. Incluye comunicación, resumen de interacción, secuencia y diagramas de tiempo.
- **Diagramas de estructura:** un tipo de diagrama que representa los elementos de una especificación que son independientes del tiempo.

Incluye clase, estructura compuesta, componente, despliegue, objeto y diagrama de paquetes.

3.5.2.1. Diagrama de Casos de Uso

Se implementó el diagrama de casos de uso para definir los requerimientos de uso del sistema, así como para indicar las personas involucradas (actores), las tareas que podrán realizar (casos) y las interacciones de dichos actores (asociaciones). Con la implementación de este diagrama se buscó detectar con mayor facilidad lo que realmente debe hacer el sistema en base a cada requerimiento, y así adquirir información valiosa al momento de desarrollar el mismo.

3.5.2.2. Diagrama de Actividades

Se realizó un diagrama de actividades, el cual se encuentra entre los diagramas de comportamiento de UML con la finalidad de mostrar el flujo de trabajo y la secuencia de actividades que tiene el sistema desde el punto de inicio hasta el punto final detallando algunas de las rutas de decisiones que existen durante el progreso de los eventos.

3.5.2.3. Diagrama de Secuencia

Se hizo uso del diagrama de secuencia con la finalidad de modelar el flujo lógico dentro del sistema de una manera visual, este está representado por la secuencia de mensajes entre los objetos a lo largo de una línea de tiempo vertical. Por otra parte permitió documentar y validar la lógica del sistema y sirvió tanto para el análisis como para el diseño ya que se centra en identificar el comportamiento dentro del sistema.

3.5.2.4. Diseño de la Interfaz de Usuario

Con la finalidad de crear una aplicación con una grata experiencia de usuario y completamente interactiva e intuitiva se realizó el diseño de la interfaz de usuario, la cual se usó para establecer los parámetros por los cuales se estará trabajando, como lo son la gama de colores, las metáforas, la simbología y la disposición de los distintos elementos que se encontrarán tanto en la aplicación web como la móvil.

3.5.2.5. Mapa de Navegación

Con el objetivo de facilitar el uso de las aplicaciones (web y móvil) al usuario, se realizó un mapa de navegación en el cual se especificaron las distintas vistas y secciones a las cuales se podrá acceder mediante un menú.

3.5.2.6. Modelo Entidad-Relación

Adicionalmente para el diseño de la base de datos, se realizó el modelo de entidad-relación (ER) que según Addison Welsey (1990) “es una técnica para definir las necesidades de información de su organización y proporciona una buena base para sistemas de alta calidad dirigidos a satisfacer las necesidades de su empresa”. Una manera fácil de explicar este modelo es que busca identificar los aspectos relevantes en la organización (entidades), sus propiedades (atributos) y cómo se relacionan entre sí (relaciones), y para que obtenga valor se implementó dentro de un contexto que involucra la manera en que se realizaron los procesos dentro de la organización.

3.5.2.7. Modelo Lógico

Con la finalidad de definir más detalladamente la estructura de la base de datos utilizada en el proyecto, se realizó un modelo lógico, en el cual se muestran las tablas que finalmente contendrá la base de datos, con cada una de sus columnas (atributos). Luego de haber realizado la conversión del modelo Entidad-Relación al modelo lógico se pudo observar como las relaciones de muchos a muchos y de igual manera relaciones con atributos propios, se desplegaron en tablas adicionales lo cual indica el correcto esquema final de las base de datos.

3.5.3. Codificar

Al momento de codificar, uno de los pocos requerimientos de la metodología XP es que el cliente siempre esté disponible, no sólo para ayudar al equipo de desarrollo, sino para ser parte de él. Por otra parte, el código debe seguir los estándares de programación para mantener la consistencia en el código y sea fácil de leer y refactorizar, lo cual significa reescribir el código sin alterar su funcionalidad, con la finalidad de optimizarlo.

Las pruebas de unidad se crearon primero, es decir, antes del código de la aplicación. Según la metodología XP esto es un estándar que permite crear el código de manera más fácil y rápida, además de que ayudará a tener en cuenta lo que realmente se necesita hacer y obtener una retroalimentación inmediata mientras se trabaja, así como poder saber cuándo se alcanza el objetivo de la iteración en la que se esté trabajando.

Se estuvo integrando segmentos de código frecuentemente, es decir, cada módulo o funcionalidad o historia de usuario que se fue codificando se unió al resto, esto permitía la detección temprana de problemas de compatibilidad, de esta manera se estuvo trabajando en todo momento en el contexto de la versión del sistema más reciente.

Por otra parte, se hizo uso de la reutilización de código funcional, aprovechando las ventajas del framework de desarrollo web Symfony para reducir el tiempo de codificación, además de dar mayor enfoque a las necesidades particulares del cliente, en vez de enfocarse en aspectos genéricos que ya se encuentran codificados. Además se aprovechó esta característica para refactorizar constantemente el código, luego de ser escrito.

Como información adicional, debido a que el sistema a desarrollar tendrá una aplicación bajo entorno web y otra bajo entorno Android, se hizo uso de lenguajes de programación acordes a estos, como lo son PHP, HTML, CSS y JavaScript principalmente para la aplicación web, mientras que para la aplicación Android se hizo uso del lenguaje Java y XML, del mismo modo, para la base de datos se usó el lenguaje SQL.

3.5.4. Probar

La metodología XP, expresa claramente que todo el código debe tener pruebas unitarias, que consisten en pruebas de caja blanca para observar todo el proceso que ocurre dentro del módulo. Para cumplir con esta especificación se hizo uso de la herramienta del framework Symfony para crear un conjunto de pruebas unitarias automatizadas, permitiendo así la refactorización, y verificar que luego un pequeño

cambio en la estructura del código no implique un cambio en la funcionalidad del sistema. Crear las pruebas unitarias antes del código ayuda a concretar los requerimientos y mejora el enfoque del desarrollo.

Todo el código debe aprobar las pruebas unitarias antes de que pueda ser liberado, si se consigue algún error se crearán más pruebas antes de proceder a codificar la solución. Finalmente se aplicarán pruebas de aceptación frecuentemente, las cuales consisten en pruebas de caja negra, es decir, se obvia todo el proceso que pueda contener el módulo internamente y se observan las entradas y las salidas, el cliente define si el módulo pasa la prueba o no, y se publicará la calificación obtenida, en caso de no aprobarla en un 100%, esta historia de usuario se atenderá nuevamente en la próxima iteración.

CAPÍTULO IV

RESULTADOS

4.1. Fase I: Analizar

En la presente fase, se hizo uso de las herramientas que ofrece la metodología de desarrollo *eXtreme Programming*, las cuales esquematizan y facilitan el análisis de la situación actual de la empresa, para así determinar los requerimientos de la misma mediante las historias de usuarios y establecer una planificación, tanto de reuniones como de liberación, en base a estos aspectos y adicionalmente realizar las pruebas de aceptación pertinente de cada iteración. A continuación, se muestra un diagrama de flujo de datos del proceso actual y se enumeran y explican las actividades realizadas para el cumplimiento de la primera fase.

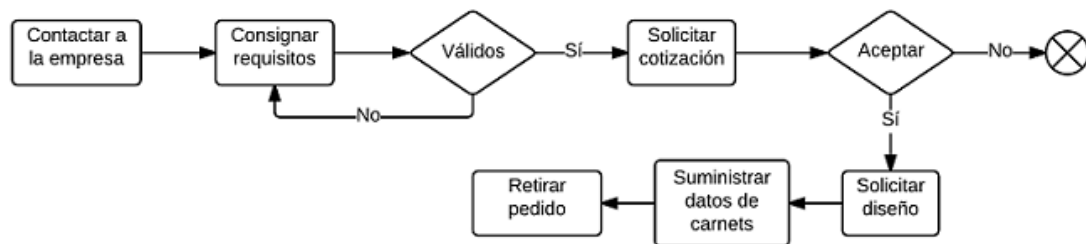


Figura 1. Diagrama de flujo de datos del sistema actual

Fuente: Emilio Clemente (2015)

4.1.1. Historias de usuario

Las historias de usuario se crearon con la finalidad de conocer las necesidades del cliente, donde se especifican las funciones con las que debe contar el sistema principalmente, con las que posteriormente se estableció un plan de reuniones, en las cuales se acudió al cliente para obtener más especificaciones acerca de los requerimientos de la historia de usuario que se estuviera desarrollando en el momento, lo cual permitió evitar un extenso documento de requerimientos.

A continuación se presentan las historias de usuarios mencionadas.

Cuadro 1. Historia de Usuario: Administración de clientes

Fuente: Emilio Clemente (2015)

Historia de Usuario	
Identificador: 1	Usuario: Emilio José Clemente
Nombre: Administración de clientes	
Prioridad: Alta	Programador: Emilio David Clemente
Descripción: El cliente puede ingresar al módulo de registro en el sistema y registrar un usuario con privilegios de cliente, de igual manera modificar sus datos posteriormente	

Cuadro 2. Historia de Usuario: Administración de empresa

Historia de Usuario	
Identificador: 2	Usuario: Emilio José Clemente
Nombre: Administración de empresa	
Prioridad: Alta	Programador: Emilio David Clemente
Descripción: El cliente ya registrado puede ingresar al módulo “Empresas” y registrar una nueva empresa, de igual manera poder eliminarla en caso de ser necesario	

Fuente: Emilio Clemente (2015)

Cuadro 3. Historia de Usuario: Administración de diseño

Historia de Usuario	
Identificador: 3	Usuario: Emilio José Clemente
Nombre: Administración de diseño	
Prioridad: Alta	Programador: Emilio David Clemente
Descripción: Registrar un diseño, indicando el nombre, orientación y los campos. De igual manera tener la posibilidad de modificarlo en caso de ser necesario	

Fuente: Emilio Clemente (2015)

Cuadro 4. Historia de Usuario: Edición de diseño

Historia de Usuario	
Identificador: 4	Usuario: Emilio José Clemente
Nombre: Edición de diseño	
Prioridad: Media	Programador: Emilio David Clemente
Descripción: Se puede editar un diseño ya creado, seleccionando el fondo, ubicando los campos correspondientes y estableciendo el reverso	

Fuente: Emilio Clemente (2015)

Cuadro 5. Historia de Usuario: Administración de empleados

Historia de Usuario	
Identificador: 5	Usuario: Emilio José Clemente
Nombre: Administración de empleados	
Prioridad: Alta	Programador: Emilio David Clemente
Descripción: Se puede crear nuevos usuarios con el rol de empleado, así como modificar sus datos o eliminarlo posteriormente de ser necesario	

Fuente: Emilio Clemente (2015)

Cuadro 6. Historia de Usuario: Administración de carnets

Historia de Usuario	
Identificador: 6	Usuario: Emilio José Clemente
Nombre: Administración de carnets	
Prioridad: Alta	Programador: Emilio David Clemente
Descripción: El cliente puede realizar carnets suministrando la información correspondiente a los campos seleccionados para el diseño, así como editarlos posteriormente o eliminar el carnet en caso de ser necesario. Se puede crear carnets para cualquier diseño disponible	

Fuente: Emilio Clemente (2015)

Cuadro 7. Historia de Usuario: Solicitud de pedido

Historia de Usuario	
Identificador: 7	Usuario: Emilio José Clemente
Nombre: Solicitud de pedido	
Prioridad: Alta	Programador: Emilio David Clemente
Descripción: El cliente puede seleccionar los carnets que requiera imprimir, así como los accesorios que desee y solicitar un pedido a la empresa de carnetización	

Fuente: Emilio Clemente (2015)

Cuadro 8. Historia de Usuario: Visualización de catálogo

Historia de Usuario	
Identificador: 8	Usuario: Lisbeth Sánchez
Nombre: Visualización de catálogo	
Prioridad: Baja	Programador: Emilio David Clemente
Descripción: Se puede observar los productos que ofrece la empresa, con su respectiva descripción, precio y disponibilidad	

Fuente: Emilio Clemente (2015)

Cuadro 9. Historia de Usuario: Visualización de cotización

Historia de Usuario	
Identificador: 9	Usuario: Lisbeth Sánchez
Nombre: Visualización de cotización	
Prioridad: Media	Programador: Emilio David Clemente
Descripción: El cliente puede visualizar la cotización de un pedido al momento de realizarlo, al igual que al momento de revisar el catálogo	

Fuente: Emilio Clemente (2015)

Cuadro 10. Historia de Usuario: Visualización de pedidos

Historia de Usuario	
Identificador: 10	Usuario: Soraya Silmi
Nombre: Visualización de pedidos	
Prioridad: Alta	Programador: Emilio David Clemente
Descripción: Tanto el empleado como el administrador pueden observar los pedidos pendientes con su respectiva información	

Fuente: Emilio Clemente (2015)

Cuadro 11. Historia de Usuario: Impresión de carnets

Historia de Usuario	
Identificador: 11	Usuario: Emilio José Clemente
Nombre: Impresión de carnets	
Prioridad: Alta	Programador: Emilio David Clemente
Descripción: El empleado puede imprimir un carnet creado o un pedido completo desde la interfaz del sistema	

Fuente: Emilio Clemente (2015)

Cuadro 12. Historia de Usuario: Exportación de carnets

Historia de Usuario	
Identificador: 12	Usuario: Soraya Silmi
Nombre: Exportación de carnets	
Prioridad: Baja	Programador: Emilio David Clemente
Descripción: El empleado puede exportar un carnet creado o un pedido completo a un archivo PDF o JPG desde la interfaz del sistema	

Fuente: Emilio Clemente (2015)

Cuadro 13. Historia de Usuario: Generación de reportes

Historia de Usuario	
Identificador: 13	Usuario: Soraya Silmi
Nombre: Generación de reportes	
Prioridad: Alta	Programador: Emilio David Clemente
Descripción: El administrador puede generar reportes sobre los carnets impresos por un cliente entre dos fechas especificadas, el cliente que más carnets ha impreso el último mes, cantidad de carnets por cliente, cliente con más pedidos generados, y exportarlo en un archivo Excel o PDF con la información pertinente para así facilitar la toma de decisión y seguimiento de la empresa	

Fuente: Emilio Clemente (2015)

Cuadro 14. Historia de Usuario: Importación de datos

Historia de Usuario	
Identificador: 14	Usuario: Fabiola Soto
Nombre: Importación de datos	
Prioridad: Alta	Programador: Emilio David Clemente
Descripción: El cliente tiene la posibilidad de importar los datos correspondientes a los campos de los carnets desde un archivo Excel externo y crear tantos carnets automáticamente como hayan en el listado o nómina importado	

Fuente: Emilio Clemente (2015)

4.1.2. Plan de Reuniones

En esta actividad de la fase de análisis se estableció el tiempo para la culminación del sistema, para ello se fijaron fechas para la realización de cada historia de usuario, de las cuales el cliente fijó la prioridad de cada una de ellas y el orden en que se realizaron, para posteriormente establecer el plan de liberación en base a esta información.

A continuación, se muestran los cuadros que indican la historia de usuario a tratar y su tiempo de realización estimado.

Cuadro 15. Plan de reuniones: Administración de clientes

Plan de Reuniones
Identificador de la historia de usuario: 1
Nombre de la historia de usuario: Administración de clientes
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 16. Plan de reuniones: Administración de empresa

Plan de Reuniones
Identificador de la historia de usuario: 2
Nombre de la historia de usuario: Administración de empresa
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 17. Plan de reuniones: Administración de diseño

Plan de Reuniones
Identificador de la historia de usuario: 3
Nombre de la historia de usuario: Administración de diseño
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 18. Plan de reuniones: Edición de diseño

Plan de Reuniones
Identificador de la historia de usuario: 4
Nombre de la historia de usuario: Edición de diseño
Tiempo estimado en semanas: 2

Fuente: Emilio Clemente (2015)

Cuadro 19. Plan de reuniones: Administración de empleados

Plan de Reuniones
Identificador de la historia de usuario: 5
Nombre de la historia de usuario: Administración de empleados
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 20. Plan de reuniones: Administración de carnets

Plan de Reuniones
Identificador de la historia de usuario: 6
Nombre de la historia de usuario: Administración de carnets
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 21. Plan de reuniones: Solicitud de pedido

Plan de Reuniones
Identificador de la historia de usuario: 7
Nombre de la historia de usuario: Solicitud de pedido
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 22. Plan de reuniones: Visualización de catálogo

Plan de Reuniones
Identificador de la historia de usuario: 8
Nombre de la historia de usuario: Visualización de catálogo
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 23. Plan de reuniones: Visualización de cotización

Plan de Reuniones
Identificador de la historia de usuario: 9
Nombre de la historia de usuario: Visualización de cotización
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 24. Plan de reuniones: Visualización de pedidos

Plan de Reuniones
Identificador de la historia de usuario: 10
Nombre de la historia de usuario: Visualización de pedidos
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 25. Plan de reuniones: Impresión de carnets

Plan de Reuniones
Identificador de la historia de usuario: 11
Nombre de la historia de usuario: Impresión de carnets
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 26. Plan de reuniones: Exportación de carnets

Plan de Reuniones
Identificador de la historia de usuario: 12
Nombre de la historia de usuario: Exportación de carnets
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

Cuadro 27. Plan de reuniones: Generación de reportes

Plan de Reuniones
Identificador de la historia de usuario: 13
Nombre de la historia de usuario: Generación de reportes
Tiempo estimado en semanas: 2

Fuente: Emilio Clemente (2015)

Cuadro 28. Plan de reuniones: Importación de datos

Plan de Reuniones
Identificador de la historia de usuario: 14
Nombre de la historia de usuario: Importación de datos
Tiempo estimado en semanas: 1

Fuente: Emilio Clemente (2015)

4.1.3. Plan de Liberaciones

A partir de la prioridad que el cliente fijó para cada historia de usuario, y la fecha estimada de realización de cada una de ellas estipuladas en el plan de reuniones, se definió el presente plan de liberaciones, el cual cumplió con la finalidad de establecer el orden en el que se realizarán las historias de usuarios y las fechas para la entrega de estas como software funcional, probado y listo para ser usado por el cliente.

A continuación, se presentan los cuadros en donde se refleja dicho plan de liberación.

Cuadro 29. Plan de liberaciones: Primera liberación

Liberación	
Número de liberación: 1	
Historias de usuario liberadas: 1, 2, 3	
Fecha de Inicio: 20/07/2015	Fecha de Finalización: 03/08/2015

Fuente: Emilio Clemente (2015)

Cuadro 30. Plan de liberaciones: Segunda liberación

Liberación	
Número de liberación: 2	
Historias de usuario liberadas: 4	
Fecha de Inicio: 03/08/2015	Fecha de Finalización: 17/08/2015

Fuente: Emilio Clemente (2015)

Cuadro 31. Plan de liberaciones: Tercera liberación

Liberación	
Número de liberación: 3	
Historias de usuario liberadas: 5	
Fecha de Inicio: 17/08/2015	Fecha de Finalización: 24/08/2015

Fuente: Emilio Clemente (2015)

Cuadro 32. Plan de liberaciones: Cuarta liberación

Liberación	
Número de liberación: 4	
Historias de usuario liberadas: 6, 10	
Fecha de Inicio: 24/08/2015	Fecha de Finalización: 07/09/2015

Fuente: Emilio Clemente (2015)

Cuadro 33. Plan de liberaciones: Quinta liberación

Liberación	
Número de liberación: 5	
Historias de usuario liberadas: 14	
Fecha de Inicio: 07/09/2015	Fecha de Finalización: 14/09/2015

Fuente: Emilio Clemente (2015)

Cuadro 34. Plan de liberaciones: Sexta liberación

Liberación	
Número de liberación: 6	
Historias de usuario liberadas: 7	
Fecha de Inicio: 14/09/2015	Fecha de Finalización: 21/09/2015

Fuente: Emilio Clemente (2015)

Cuadro 35. Plan de liberaciones: Séptima liberación

Liberación	
Número de liberación: 7	
Historias de usuario liberadas: 8, 9	
Fecha de Inicio: 21/09/2015	Fecha de Finalización: 05/10/2015

Fuente: Emilio Clemente (2015)

Cuadro 36. Plan de liberaciones: Octava liberación

Liberación	
Número de liberación: 8	
Historias de usuario liberadas: 11, 12	
Fecha de Inicio: 05/10/2015	Fecha de Finalización: 19/10/2015

Fuente: Emilio Clemente (2015)

Cuadro 37. Plan de liberaciones: Novena liberación

Liberación	
Número de liberación: 9	
Historias de usuario liberadas: 13	
Fecha de Inicio: 19/10/2015	Fecha de Finalización: 26/10/2015

Fuente: Emilio Clemente (2015)

4.1.4. Definición de Requerimientos Funcionales y No Funcionales del Sistema

La definición de requerimientos funcionales y no funcionales es la manera idónea de complementar la funcionalidad correcta de un sistema para un uso e implementación adecuada. A partir de las historias de usuario antes descritas se establecieron los siguientes requerimientos:

4.1.4.1. Requerimientos Funcionales

- Agregar pedido

- Agregar usuarios
- Agregar accesorios
- Agregar empresa
- Agregar Diseño de carnet nuevo
- Generar PDF de carnet para su impresión
- Crear carnet
- Diseñar carnet
- Importar datos de carnet a partir de una hoja de calculo
- Previsualización del carnet
- Mostrar los detalles de pedido
- Generar gráficas con estadísticas
- Generar reportes
- Mostrar pedidos por fecha y estados
- Buscar y filtrar los datos en tablas del sistema

4.1.4.2. Requerimientos No Funcionales

- Mantener la integridad de los datos que maneja el sistema
- Fácil uso de las diferentes funcionalidades para los usuarios del sistema
- Guardar una experiencia de usuario similar en todo el sistema
- Componer módulos robustos que garanticen la estabilidad del sistema
- Preservar la arquitectura del sistema para aumentar la escalabilidad del mismo
- Reducir la cantidad de recursos que necesite el sistema

4.2. Fase II: Diseñar

En esta fase, se realizaron las actividades relacionadas con el diseño del sistema, haciendo uso de los diagramas que ofrece el Lenguaje Unificado de Modelado (UML) por sus siglas en inglés. Dichos diagramas se dividieron en tres tipos: diagramas de interacción, diagramas de comportamiento y diagrama de estructura, los cuales serán presentados más adelante en la explicación de la presente fase.

Por otra parte, se realizó el diseño de la base de datos mediante el modelo de “Entidad - Relación” (MER) para identificar los tipos de entidades presentes en el sistema, así como sus atributos y las relaciones entre dichas entidades, posteriormente se realizó el modelo lógico en el cual se define la totalidad de tablas en la base de datos, con sus atributos y relaciones entre claves primarias y claves foráneas.

A continuación se muestran los diagramas anteriormente mencionados.

4.2.1. Diagrama de Casos de Uso

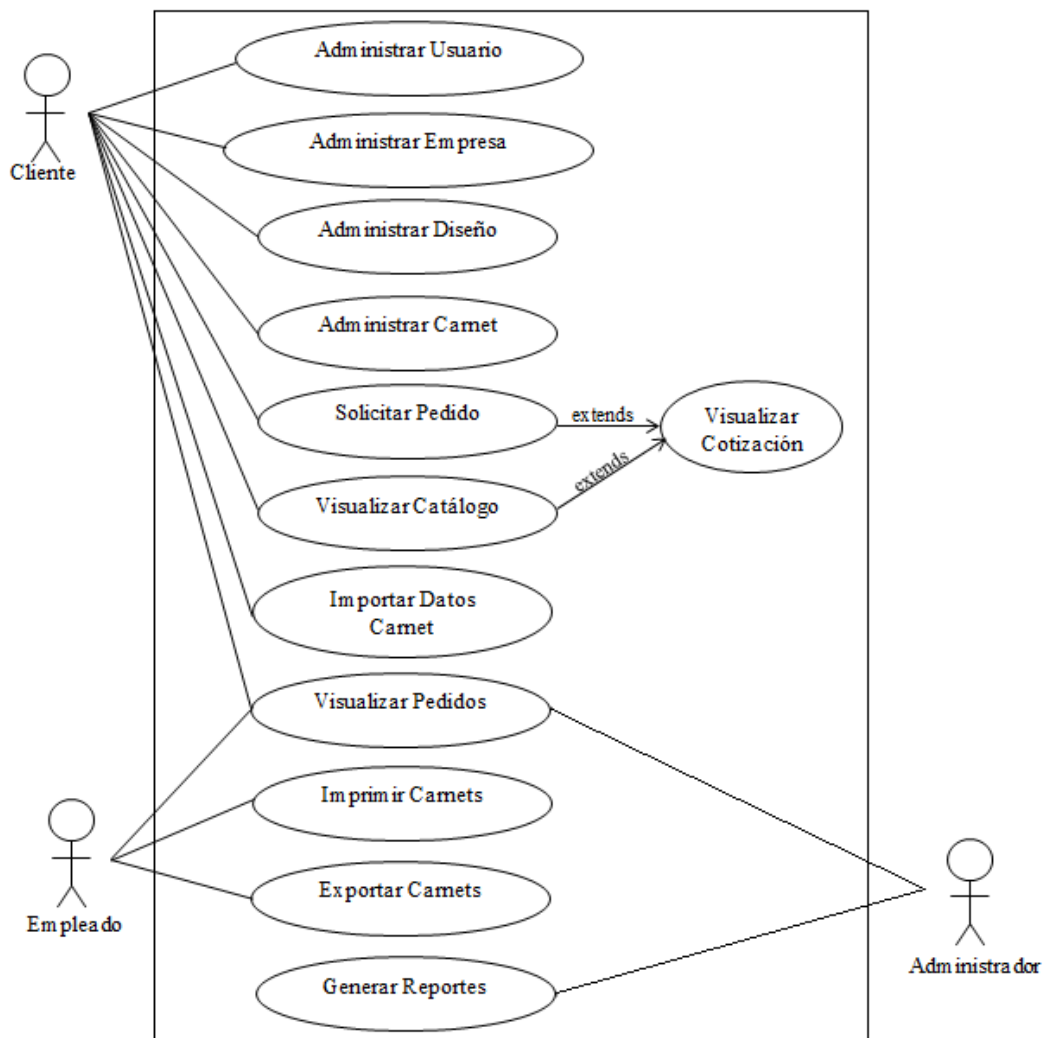


Figura 2. Casos de Uso
Fuente: Emilio Clemente (2015)

Una vez establecidos los casos de uso, se realizaron una serie de cuadros donde se describen detalladamente la función de cada uno, así como su flujo de actividad normal y alternativo y la precondición que el usuario debe cumplir para llevarlo a cabo, y por último, la condición resultante posterior a haber ejecutado la acción correctamente.

Cuadro 38. Descripción de Caso de Uso: Administrar usuario

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 11/08/2015
Nombre: Administrar usuario	Actor: Cliente
Descripción: Permite crear y actualizar un usuario	
Precondición: El usuario debe haber ingresado al módulo de registro del sistema para crearlo y a su respectivo perfil para actualizar la información de dicho usuario	
Flujo normal: <ol style="list-style-type: none"> 1. El actor ingresa a la pestaña “Registro” 2. El sistema muestra un formulario que deberá ser llenado debidamente 3. El actor llena los datos del formulario 4. El actor pulsa el botón de “Registrarse” 5. El sistema valida los datos y procede a guardar los datos del usuario en la base de datos 6. Se muestra un mensaje en pantalla indicando que el usuario ha sido registrado exitosamente 7. El actor ingresa al sistema con el nuevo usuario 8. En la vista del perfil, el actor modifica los datos que se le permita modificar 	
Flujo alternativo: <ol style="list-style-type: none"> 5. El sistema valida los datos y si existe algún error, se notifica al usuario del mismo, y no se procede a registrar al usuario hasta que este corrija el error 	
Post-condición: Un nuevo registro en la base de datos con los datos del usuario, habilitando a este a ingresar al sistema	

Fuente: Emilio Clemente (2015)

Cuadro 39. Descripción de Caso de Uso: Administrar empresa

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 11/08/2015
Nombre: Administrar empresa	Actor: Cliente
Descripción: Permite registrar y actualizar una empresa	
Precondición: El cliente debe haber ingresado al sistema como usuario	
Flujo normal: <ol style="list-style-type: none">1. El actor ingresa a la pestaña “Empresas”2. El sistema muestra un formulario que deberá ser llenado debidamente3. El actor llena los datos del formulario4. El actor pulsa el botón de “Registrar Empresa”5. El sistema valida los datos y procede a guardar los datos de la empresa en la base de datos6. Se muestra un mensaje en pantalla indicando que la empresa ha sido registrada7. Luego de la empresa ser creada, en la misma pestaña puede seleccionarla y hacer clic en el botón “Editar”8. El usuario modifica los campos necesarios y que le sean permitidos y presiona el botón “Aceptar”	
Flujo alternativo: <ol style="list-style-type: none">5. El sistema valida los datos y si existe algún error, se notifica al usuario del mismo, y no se procede a registrar la empresa hasta que este corrija el error	
Post-condición: Un nuevo registro en la base de datos con los datos de la empresa	

Fuente: Emilio Clemente (2015)

Cuadro 40. Descripción de Caso de Uso: Administrar carnet

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 11/08/2015
Nombre: Administrar carnet	Actor: Cliente
Descripción: Permite crear y modificar un carnet	
Precondición: El cliente debe estar ingresado en el sistema e identificado como usuario y haber seleccionado una empresa sobre la cual estará trabajando	
Flujo normal: <ol style="list-style-type: none">1. El actor ingresa a la pestaña “Base de Datos”2. El actor selecciona el diseño del carnet que creará3. El sistema muestra un formulario para ingresar los datos pertinentes4. El usuario suministra los datos correspondientes5. El actor hace clic en el botón de “Crear Carnet”6. El sistema valida los datos y procede a registrar el carnet en la base de datos7. Se muestra un mensaje en pantalla indicando que el carnet ha sido creado con éxito8. El usuario realiza una búsqueda del carnet en la tabla mostrada en la misma pestaña9. Selecciona la opción “Editar”10. El sistema muestra un formulario con los datos editables del carnet11. El usuario ingresa los datos y presiona el botón “Aceptar”	
Flujo alternativo: <ol style="list-style-type: none">6. El sistema valida los datos y si hay algún error en ellos, se indica mediante un mensaje en pantalla que no se ha podido crear el carnet debido al mismo motivo	
Post-condición: Un nuevo carnet creado e ingresado en el sistema	

Fuente: Emilio Clemente (2015)

Cuadro 41. Descripción de Caso de Uso: Administrar diseño

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 11/08/2015
Nombre: Administrar diseño	Actor: Cliente
Descripción: Permite crear y modificar un diseño	
Precondición: El cliente debe estar ingresado en el sistema e identificado como usuario y haber seleccionado una empresa sobre la cual estará trabajando	
Flujo normal: <ol style="list-style-type: none">1. El actor ingresa a la pestaña “Diseños”2. El actor llena los datos del formulario “Crear nuevo diseño”3. El actor hace clic en el botón de “Crear Diseño”4. El sistema valida los datos y procede a registrar el diseño en la base de datos5. Se muestra un mensaje en pantalla indicando que el diseño ha sido creado con éxito6. El usuario selecciona el diseño creado y hace clic en el botón “Editar Diseño”7. El sistema muestra los campos del carnet y los fondos disponibles para este8. Se ubican los campos dentro del carnet, se selecciona el fondo deseado y se edita el reverso del carnet	
Flujo alternativo: <ol style="list-style-type: none">4. El sistema valida los datos y si hay algún error en ellos, se indica mediante un mensaje en pantalla que no se ha podido crear el diseño especificando el motivo	
Post-condición: Un nuevo diseño creado e ingresado en el sistema	

Fuente: Emilio Clemente (2015)

Cuadro 42. Descripción de Caso de Uso: Solicitar pedido

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 20/08/2015
Nombre: Solicitar pedido	Actor: Cliente
Descripción: Permite solicitar un pedido de carnets con sus accesorios	
Precondición: El cliente debe estar ingresado en el sistema, haber seleccionado una empresa sobre la cual trabajar y tener carnet creados con sus respectivos diseños	
Flujo normal: <ol style="list-style-type: none"> 1. El actor ingresa a la pestaña “Solicitud de Carnets” 2. El actor selecciona los carnets que desea enviar a imprimir 3. El actor hace clic en el botón “Aceptar” 4. Se muestra la lista de accesorios que puede agregar 5. El actor selecciona los accesorios que desee e indica la cantidad de los mismos 6. El sistema valida la solicitud y muestra la cotización en pantalla 7. El actor presiona el botón “Confirmar Solicitud” 8. Se muestra un mensaje en pantalla indicando que la solicitud ha sido enviada con éxito 	
Flujo alternativo: <ol style="list-style-type: none"> 6. El sistema valida la solicitud y si hay algún error, se indica mediante un mensaje en pantalla que no se ha podido realizar la operación 	
Post-condición: Una solicitud de carnets en espera	

Fuente: Emilio Clemente (2015)

Cuadro 43. Descripción de Caso de Uso: Visualizar catálogo

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 20/08/2015
Nombre: Visualizar catálogo	Actor: Cliente
Descripción: Muestra los productos disponibles y su precio	
Precondición: El cliente debe haber ingresado en el sistema	
Flujo normal: <ol style="list-style-type: none"> 1. El actor ingresa a la pestaña “Catálogo” 2. El actor realiza la búsqueda de los productos de interés 3. El sistema muestra los precios y la disponibilidad del producto 	
Flujo alternativo: <ol style="list-style-type: none"> 2. El actor realiza la búsqueda del producto, si este no existe, se muestra un mensaje en pantalla indicando que el producto no existe 	
Post-condición: Una lista de productos con sus respectivos precios y disponibilidad	

Fuente: Emilio Clemente (2015)

Cuadro 44. Descripción de Caso de Uso: Importar datos carnet

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 18/09/2015
Nombre: Importar datos carnet	Actor: Cliente
Descripción: Permite suministrar la información que irá reflejada en los carnets importando un archivo Excel con los datos organizados coherentemente	
Precondición: El cliente debe haber ingresado en el sistema, tener una empresa registrada y un diseño creado, del mismo modo la plantilla en Excel suministrada por el sistema para organizar la información	
Flujo normal: <ol style="list-style-type: none">1. El actor ingresa a la pestaña “Base de Datos”2. El actor pulsa el botón “Obtener Plantilla”3. El actor organiza la información de manera coherente con la plantilla Excel suministrada4. El actor pulsa sobre el botón “Importar Datos”5. Se muestra una ventana donde el actor selecciona el archivo a importar6. El actor hace clic en “Aceptar”7. El sistema valida el archivo suministrado y procede a cargar los datos8. El sistema muestra un mensaje en pantalla indicando que los datos fueron cargados correctamente y se han creado los carnets satisfactoriamente	
Flujo alternativo: <ol style="list-style-type: none">7. El sistema valida el archivo suministrado, si este contiene algún error, se mostrará un mensaje en pantalla indicando sobre el mismo	
Post-condición: Un listado de carnet creados a partir de un archivo Excel importado	

Fuente: Emilio Clemente (2015)

Cuadro 45. Descripción de Caso de Uso: Visualizar cotización

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 20/08/2015
Nombre: Visualizar cotización	Actor: Cliente
Descripción: Muestra el total en costo de los productos solicitados	
Precondición: El cliente debe haber ingresado en el sistema y haber realizado una solicitud de carnets o visualizado el catálogo de productos	
Flujo normal:	
<ol style="list-style-type: none"> 1. El sistema valida los productos solicitados 2. Se muestra una lista con la descripción de los productos, sus precios y el costo total 3. El cliente presiona el botón “Aprobar Cotización” 	
Flujo alternativo:	
<ol style="list-style-type: none"> 1. El sistema valida los productos solicitados, si existe algún error, se informa en pantalla del mismo 	
Post-condición: Una cotización aprobada	

Fuente: Emilio Clemente (2015)

Cuadro 46. Descripción de Caso de Uso: Visualizar pedido

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 20/08/2015
Nombre: Visualizar pedido	Actores: Cliente, empleado, administrador
Descripción: Permite revisar un pedido solicitado	
Precondición: El usuario debe haber ingresado en el sistema y debe haber un pedido pendiente	
Flujo normal:	
<ol style="list-style-type: none"> 1. El actor ingresa a la pestaña “Solicitud de Carnets” 2. El actor ingresa los datos del pedido pendiente o realiza la búsqueda de todos los pedidos pendientes 3. Hace clic en el botón “Visualizar Pedido” en el pedido correspondiente 4. El sistema muestra los detalles del pedido en cuestión 	
Flujo alternativo:	
<ol style="list-style-type: none"> 2. El actor realiza la búsqueda del pedido, si no existe, el sistema muestra un mensaje en pantalla indicando que no hay pedidos pendientes 	
Post-condición: Un pedido visualizado en pantalla	

Fuente: Emilio Clemente (2015)

Cuadro 47. Descripción de Caso de Uso: Imprimir carnets

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 20/08/2015
Nombre: Imprimir carnets	Actor: Empleado
Descripción: Permite imprimir los carnets de pedidos pendientes	
Precondición: El usuario debe haber ingresado en el sistema identificado como empleado y debe haber al menos un pedido pendiente	
Flujo normal: <ol style="list-style-type: none"> 1. El actor ingresa a la pestaña “Solicitud de Carnets” 2. El actor ingresa los datos del pedido pendiente, o realiza la búsqueda de todos los pedidos pendientes 3. Hace clic en el botón “Visualizar Pedido” en el pedido que desea imprimir 4. El sistema muestra los detalles del pedido en cuestión 5. El actor hace clic en el botón “Imprimir Pedido”, configura la impresora en la ventana emergente y envía a imprimir 6. Se muestra un mensaje en pantalla indicando que la operación se realizó exitosamente 	
Flujo alternativo: <ol style="list-style-type: none"> 2. El actor realiza la búsqueda del pedido, si no existe, el sistema muestra un mensaje en pantalla indicando que no hay pedidos pendientes 	
Post-condición: Un pedido de carnets impreso	

Fuente: Emilio Clemente (2015)

Cuadro 48. Descripción de Caso de Uso: Exportar carnets

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 20/08/2015
Nombre: Exportar carnets	Actor: Empleado
Descripción: Permite imprimir los carnets de pedidos pendientes	
Precondición: El usuario debe haber ingresado en el sistema identificado como empleado y debe haber al menos un pedido pendiente	
Flujo normal: <ol style="list-style-type: none">1. El actor ingresa a la pestaña “Pedidos”2. El realiza la búsqueda de todos los pedidos pendientes3. Hace clic en el botón “Visualizar Pedido” en el pedido que desea exportar4. El sistema muestra los detalles del pedido en cuestión5. El actor hace clic en el botón “Exportar Pedido”6. Se muestra un mensaje en pantalla indicando que la operación se realizó exitosamente	
Flujo alternativo: <ol style="list-style-type: none">2. El actor realiza la búsqueda del pedido, si no existe, el sistema muestra un mensaje en pantalla indicando que no hay pedidos pendientes	
Post-condición: Un archivo PDF con los carnets del pedido correspondiente	

Fuente: Emilio Clemente (2015)

Cuadro 49. Descripción de Caso de Uso: Generar reportes

Descripción de Caso de Uso	
Autor: Emilio Clemente	Fecha: 20/08/2015
Nombre: Generar reportes	Actor: Administrador
Descripción: Permite generar reportes con información sobre los pedidos, como fecha de solicitud, fecha de impresión al igual que fecha de creación, modificación y eliminación de diseños y de carnets	
Precondición: El usuario debe haber ingresado en el sistema identificado como administrador	
Flujo normal: <ol style="list-style-type: none">1. El actor ingresa a la pestaña “Reportes”2. El actor selecciona el tipo de reporte que desee realizar con las opciones que el sistema ofrece3. El actor hace clic en el botón “Generar Reporte”4. El sistema valida los datos5. Se muestra un mensaje en pantalla indicando que la operación se realizó exitosamente	
Flujo alternativo: <ol style="list-style-type: none">5. El sistema valida los datos, si existe algún error, se muestra un mensaje en pantalla indicando que no se ha podido realizar el reporte y la razón	
Post-condición: Un archivo Excel o PDF con el reporte solicitado	

Fuente: Emilio Clemente (2015)

4.2.2. Diagrama de Actividades

4.2.2.1. Diagrama de Actividades del Cliente

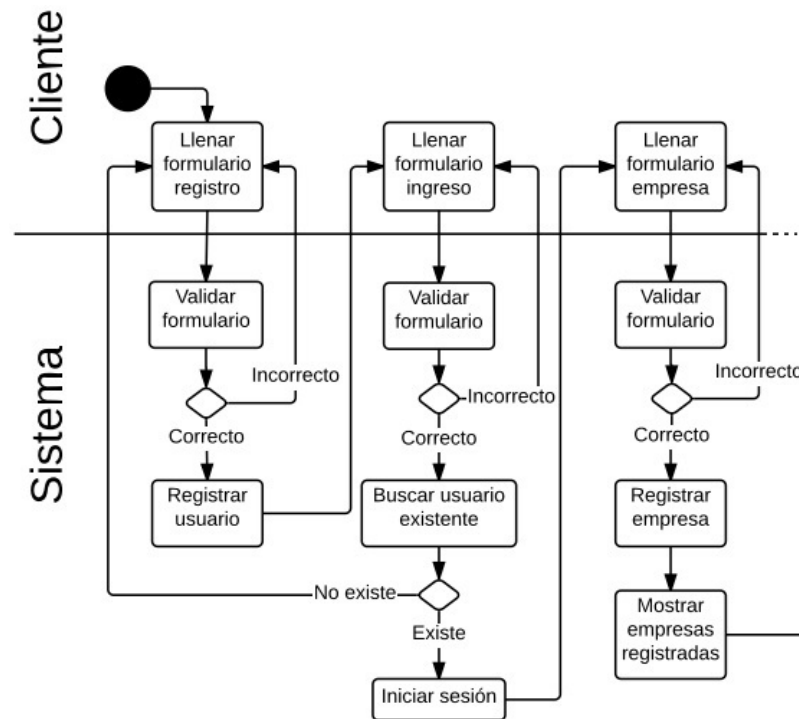


Figura 3. Diagrama de Actividades: Cliente (1)

Fuente: Emilio Clemente (2015)

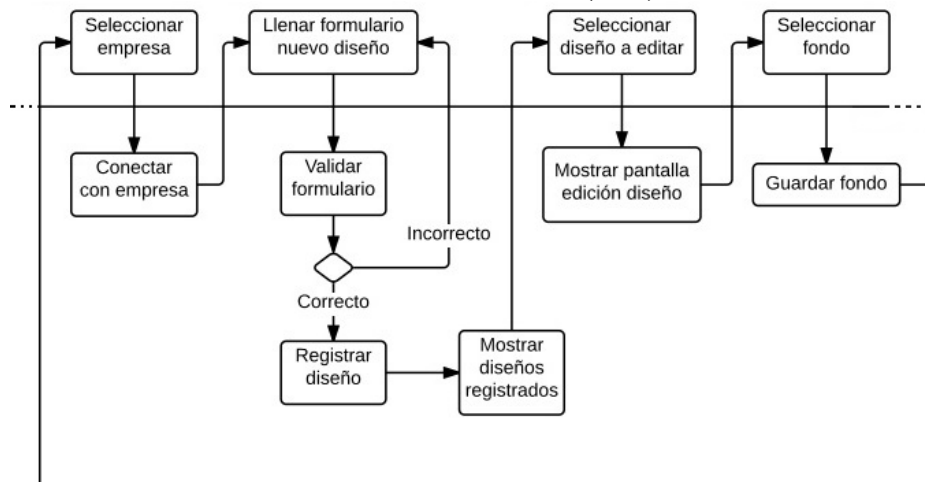


Figura 4. Diagrama de Actividades: Cliente (2)

Fuente: Emilio Clemente (2015)

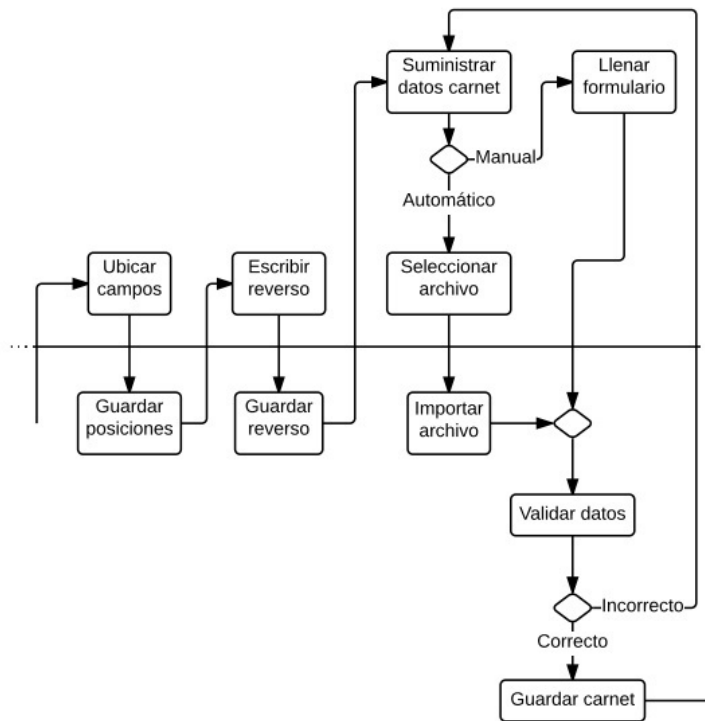


Figura 5. Diagrama de Actividades: Cliente (3)

Fuente: Emilio Clemente (2015)

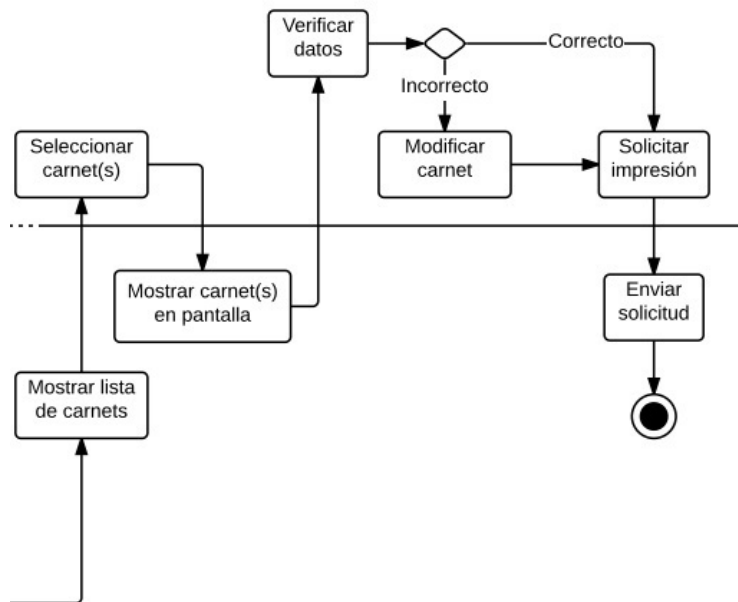


Figura 6. Diagrama de Actividades: Cliente (4)

Fuente: Emilio Clemente (2015)

4.2.2.2. Diagrama de Actividades del Empleado

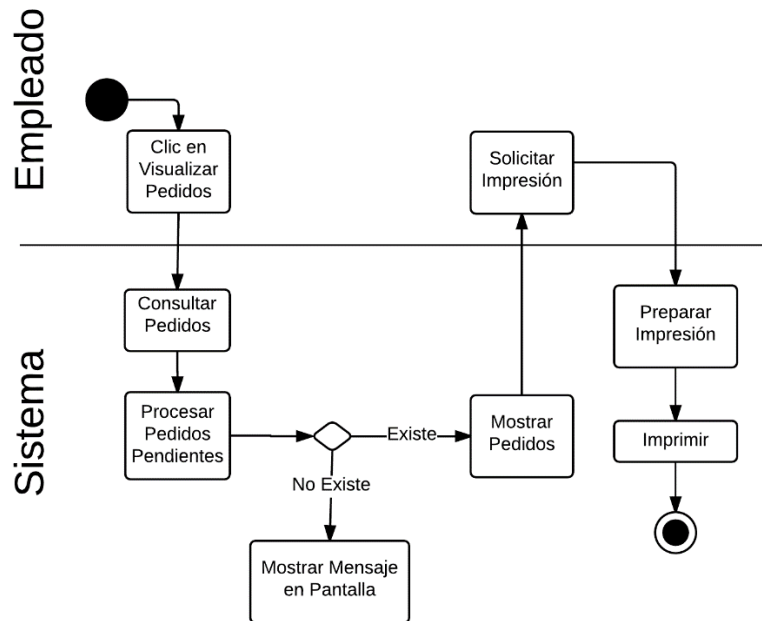


Figura 7. Diagrama de Actividades: Empleado

Fuente: Emilio Clemente (2015)

4.2.2.3. Diagrama de Actividades del Administrador

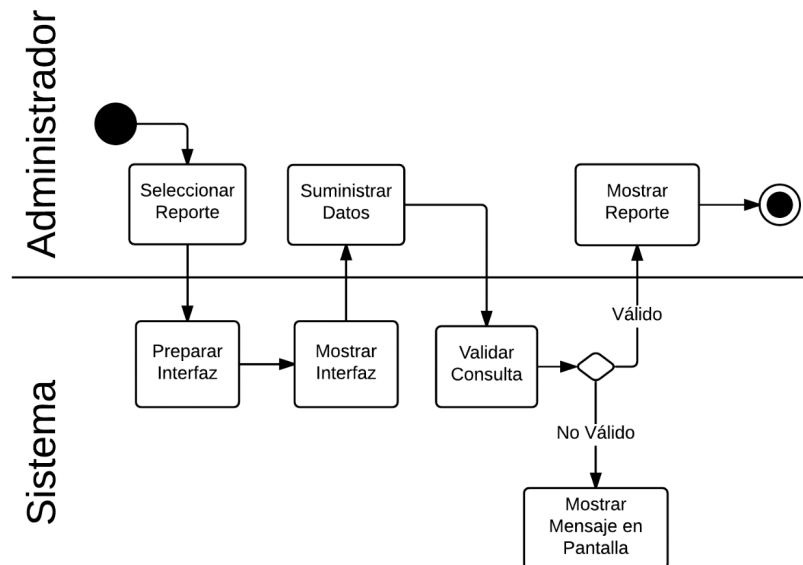


Figura 8. Diagrama de Actividades: Administrador

Fuente: Emilio Clemente (2015)

4.2.3. Diagrama de Secuencia

4.2.3.1. Diagrama de Secuencia del Cliente

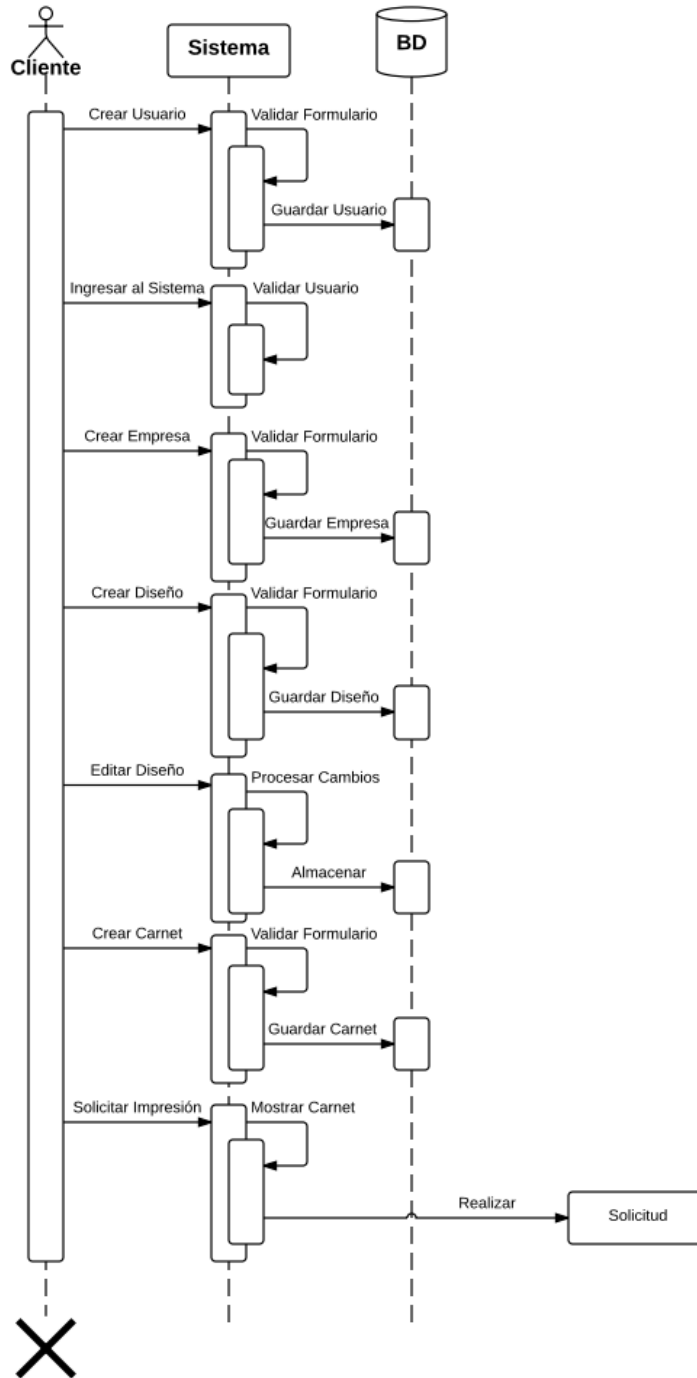


Figura 9. Diagrama de Secuencia: Cliente

Fuente: Emilio Clemente (2015)

4.2.3.2. Diagrama de Secuencia del Empleado

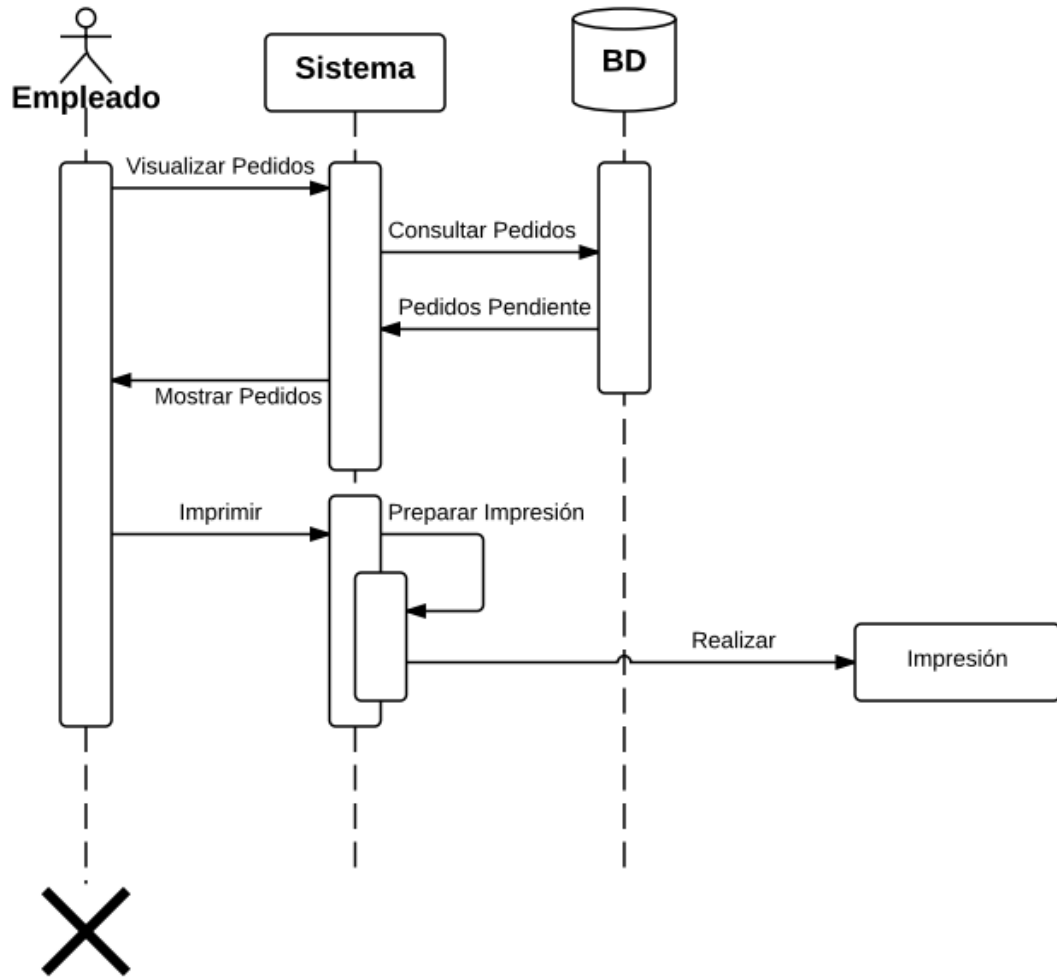


Figura 10. Diagrama de Secuencia: Empleado

Fuente: Emilio Clemente (2015)

4.2.3.3. Diagrama de Secuencia del Administrador

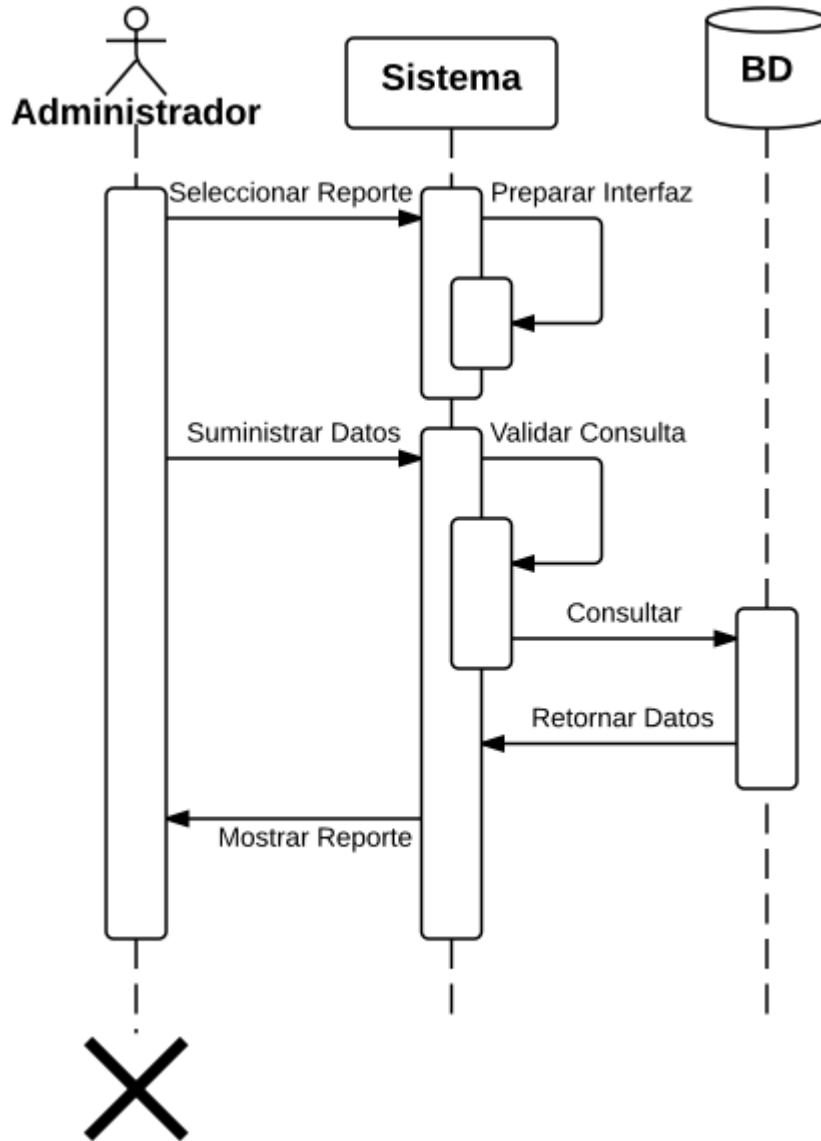


Figura 11. Diagrama de Secuencia: Administrador

Fuente: Emilio Clemente (2015)

4.2.4. Diseño de la Interfaz de Usuario

En la presente actividad de la fase de diseño del sistema, se establecieron los parámetros base en cuanto a la interfaz de usuario, para elaborar a partir de estos, una aplicación altamente interactiva y agradable para el usuario. En este orden de ideas, se describen los principios por los que se rige la mencionada interfaz, así como las características principales que la definen y en última instancia los elementos que la componen. Adicionalmente se planteó un esquema básico de cómo se encuentra estructurada la interfaz.

4.2.4.1. Principios

- **Sencillez:** El diseño hace uso mínimo de los elementos que componen la interfaz para transmitir el mensaje deseado de la mejor manera posible, siguiendo la filosofía “menos es más”, mientras más sencillo sea el diseño, más se entenderá.

- **Elegancia:** Mientras más sutiles y delicados sean los detalles del diseño, este ganará más elegancia, esto aporta a su vez una interfaz eficaz y completamente agradable al usuario. La predominación del color azul y blanco es importante ya que son considerados los colores de la elegancia.

- **Funcionalidad:** Para completar el diseño, se hace especial énfasis en que todos los elementos estén correctamente relacionados entre sí, agregando la mayor funcionalidad posible a la interfaz. La manera en que cada elemento interactúa con el resto tiene su propio significado y busca transmitir un mensaje útil para el usuario.

4.2.4.2. Características

- **Responsive:** El comportamiento *responsive* es una de las principales características con las que cuenta el diseño. Partiendo de la premisa de que cada vez más usuarios migran a los dispositivos móviles, es de vital importancia tener en cuenta las distintas resoluciones y tamaños de pantalla que se encuentran actualmente en el mercado, por lo tanto, realizar un diseño que se comporte de una manera coherente y funcional para todas las resoluciones existentes es un objetivo importante.

Se busca principalmente sacar el mayor provecho a las características de cada dispositivo (ventanas emergentes, interacción entre pantallas, etc) con el objetivo de que la interfaz se adapte a cada uno de ellos, sin importar en donde se esté visualizando el diseño.

- **Sistema de Grid:** El completo diseño cuenta además con un sistema de *grid* apropiado para que la interfaz tenga éxito en cada sector del mercado.

Bajo el principio de un comportamiento *responsive*, se hace uso de una división de contenido bastante particular. Se basa en un concepto de 12 columnas, distribuidas a lo largo de la pantalla, las cuales se adaptan completamente a cualquier contenido que se desee plasmar en la interfaz (siguiendo los mismos lineamientos del *framework Bootstrap*), pueden implementarse cualquier cantidad de filas siguiendo estas mismas reglas, cada una formada por 12 columnas, que se pueden utilizar conforme sea necesario para un correcto funcionamiento de la interfaz.

Además de la facilidad para plasmar el contenido, el sistema de *grid* ofrece una gran ventaja al momento de diseñar para las distintas resoluciones de pantalla, puesto que es sencillo e intuitivo configurar el comportamiento de las columnas para cada uno de los dispositivos, ahorrando así horas de trabajo y proporcionando un diseño más limpio y acertado.

- **Navegación:** El diseño cuenta con un menú sencillo y minimalista que facilita la navegación a través de la interfaz entera, se ubican los vínculos y botones de interés solamente, enfocando al usuario a lo que realmente desea acceder y dejando a un lado cualquier elemento que distraiga la atención y dificulte la navegación. Se agregan los detalles mínimos para que este conserve un aspecto agradable, y a su vez no se sobrecargue de información innecesaria.

Cada botón y cada vínculo cuentan con una metáfora asociada para mayor comprensión de la función que realiza. El principal objetivo del diseño es hacer una navegación intuitiva y sencilla, aportándole funcionalidad y elegancia a la interfaz.

4.2.4.3. Elementos

- **Íconos:** Hacen énfasis en la simplicidad, facilitando la orientación para el usuario. Cada ícono representa básicamente la función que realiza y enfoca al usuario en su objetivo principal.

- **Metáforas:** El diseño relaciona cada elemento de la interfaz con elementos cotidianos para el usuario, haciendo que la interacción con el sistema sea intuitiva y fácil de comprender.

- **Formularios:** Este diseño ofrece el espacio y la comodidad necesaria para una grata experiencia al momento de llenar un formulario, con amplios campos de texto e identificadores oportunos que resultan de gran ayuda para el usuario.

- **Colores:** El principal objetivo del diseño es ofrecer una variedad de colores que relacionados entre sí generen un ambiente agradable a la vista, que funcionen de manera coherente distinguiendo cada elemento del otro, ayudando al usuario a ubicarse en la interfaz. La paleta de colores cuenta con tonos corporativos, aportando un toque sofisticado al proyecto.

Cuadro 50. Paleta de Colores

Primario	Secundario	Éxito	Alerta	Advertencia
#008cba	#777777	#43ac6a	#f04124	#f08a24
Blanco	Gris Claro	Gris Medio	Gris Oscuro	Negro
#fefefe	#e6e6e6	#cacaca	#8a8a8a	#0a0a0a

- **Fuentes:** Partiendo del concepto de un diseño minimalista, se decidió utilizar una tipografía acorde con esto. Sans Serif es su principal familia de fuentes ya que posee un tipo de letra sencillo, evitando adornos excesivos y ofreciendo elegancia al texto de la interfaz.

- **Botones:** El diseño cuenta con un sistema de botones diseñados bajo el mismo esquema de sencillez, y los colores utilizados expresan explícitamente la función que ejerce cada uno de ellos (informativo, aprobación, peligro, entre otros).

-**Mensajes:** Siendo uno de los elementos más importantes con los que cuenta el diseño de la interfaz, los mensajes ofrecen una alta interacción entre el usuario con el sistema, puesto que estos informan de manera sencilla y comprensible, apoyado en colores que otorgan mayor significado aún al contenido y permiten al usuario estar al tanto de lo que ocurre mientras hace uso del sistema. Estos se encuentran ubicados en la parte superior de manera tal que sea claramente visible.

-**Logo:** El logo que se utilizó en la interfaz del sistema es el correspondiente a la empresa sobre la cual se basó la investigación, el cual se presenta a continuación.



Figura 12. Logo de La Empresa

Fuente: Carnetiza, C.A. (2015)

4.2.4.4. Esquema de la Interfaz Web



Figura 13. Esquema de la Interfaz Web

Fuente: Emilio Clemente (2015)

4.2.4.5. Esquema de la Interfaz Móvil

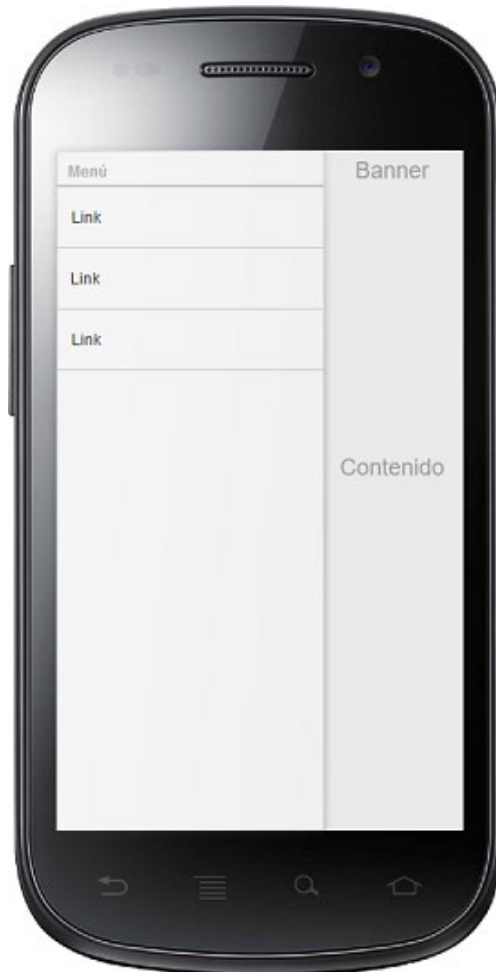


Figura 14. Esquema de la Interfaz Móvil

Fuente: Emilio Clemente (2015)

4.2.5. Mapa de Navegación

En el sistema realizado se pueden encontrar múltiples pantallas y vistas a las que el usuario puede acceder, con la finalidad de mostrar al usuario las posibles rutas que puede tomar, se realizó un mapa de navegación el cual se presenta a continuación.

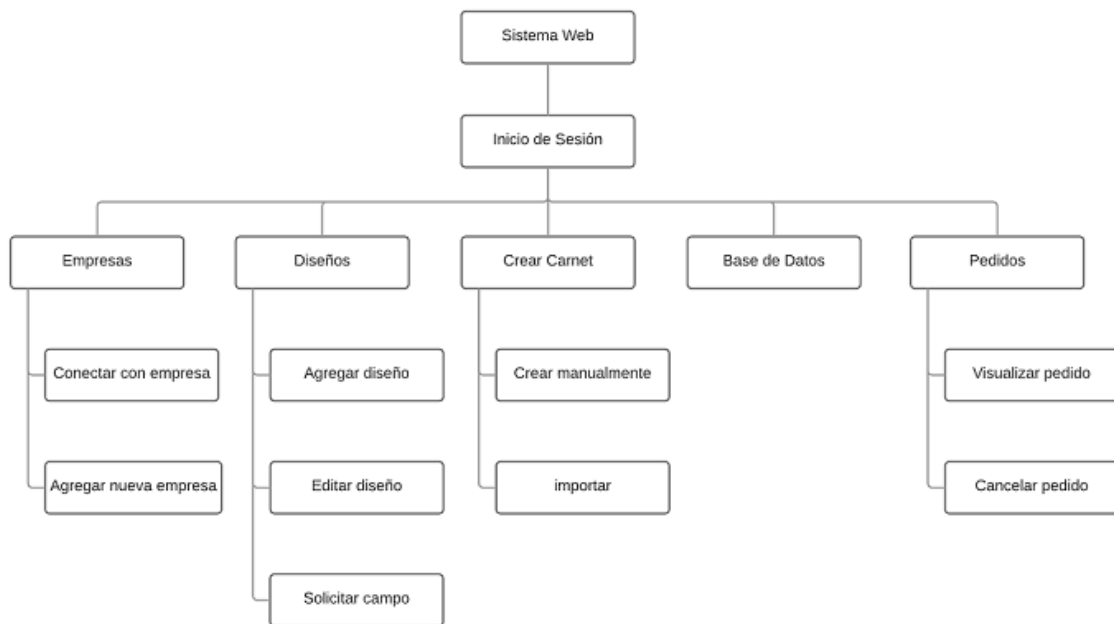


Figura 15. Mapa de Navegación

Fuente: Emilio Clemente (2015)

4.2.6. Modelo Entidad-Relación

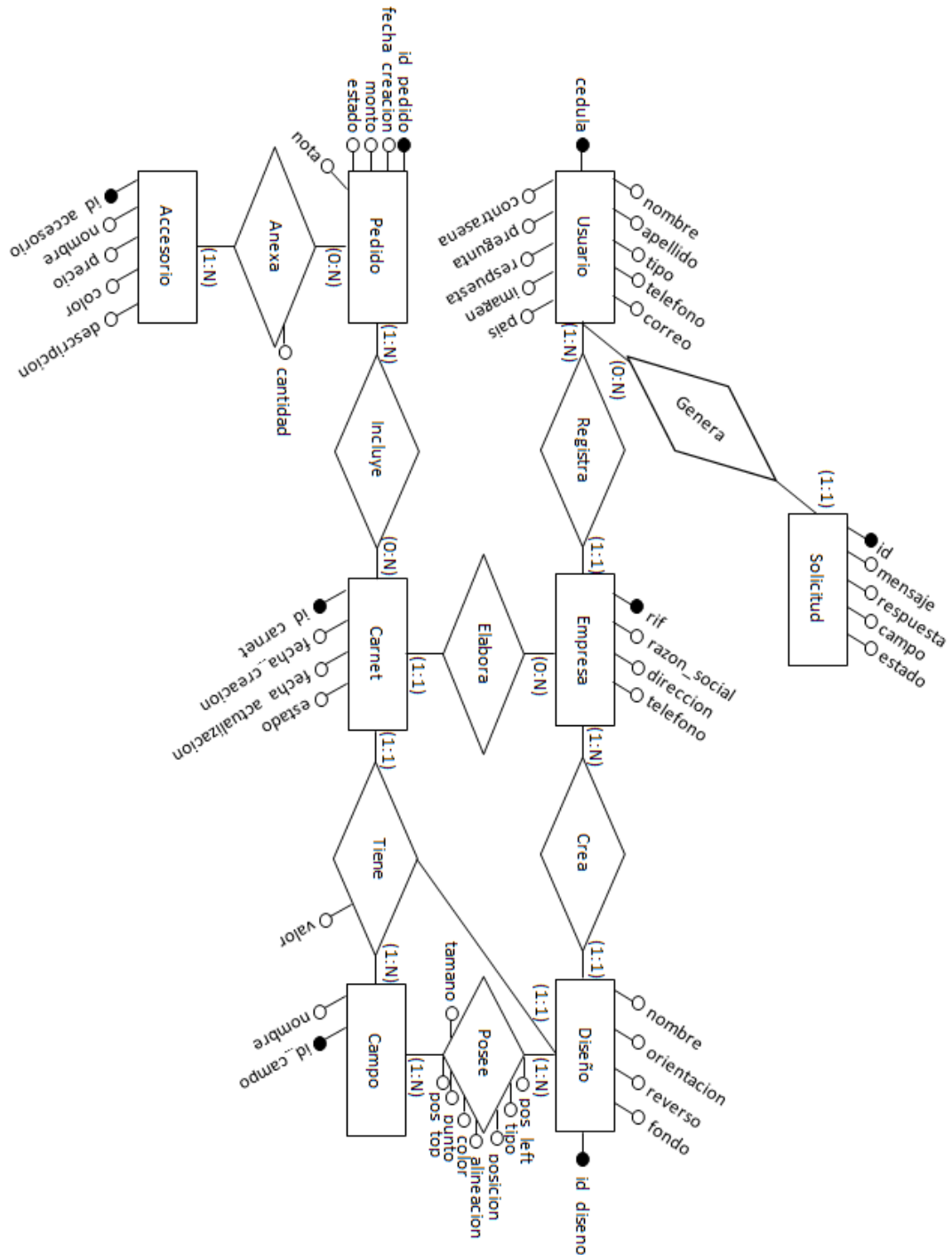


Figura 16. Modelo Entidad-Relación

Fuente: Emilio Clemente (2015)

4.2.7. Modelo Lógico

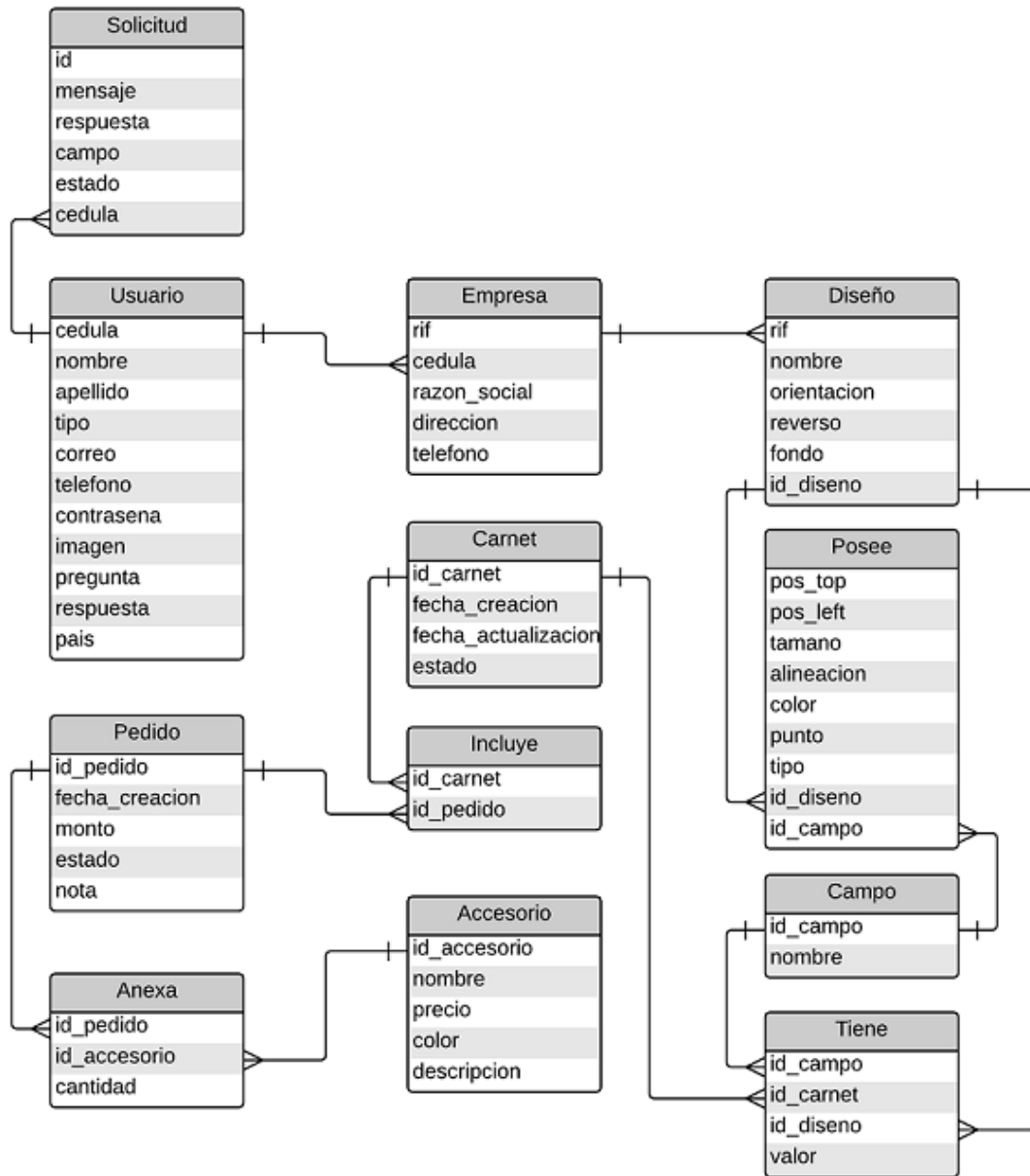


Figura 17. Modelo Lógico

Fuente: Emilio Clemente (2015)

4.3. Fase III: Codificar

En esta tercera fase del desarrollo, se realizó todo lo referente al código del sistema, el cual fue escrito en distintos lenguajes de programación, entre ellos se encuentran: PHP, TWIG, XML, YAML, Annotations, JavaScript, HTML5, CSS y SQL. Por otra parte, con el objetivo de aprovechar las ventajas de los avances a nivel de programación, se hizo uso de herramientas como *Frameworks*, *bundles*, librerías y *plugins* los cuales fueron útiles al momento de mantener el código organizado, óptimo y seguro, del mismo modo utilizar funciones ya implementadas y personalizar el sistema en el tiempo previsto para su realización.

A continuación se enumeran y explican brevemente las herramientas utilizadas para la codificación del sistema.

4.3.1. Framework

El *framework* utilizado para la construcción del código del sistema fue *Symfony* en su versión 2.6.4, el cual sirvió de apoyo para establecer lo que fue el marco de trabajo y cómo se estructuraron y ubicaron los archivos, clases, librerías, rutas, entre otros. Este *framework* está basado en el patrón de arquitectura “Modelo-Vista-Controlador” (MVC), el cual básicamente separa el sistema completo en tres partes: modelo, vista y controlador, lo que permitió mantener el código organizado y correctamente estructurado.

4.3.2. Bundles

Los bundles son los paquetes en los que se encuentra dividido un sistema desarrollado en *Symfony*, siendo el sistema en sí un *bundle*. Adicionalmente, existen *bundles* de terceros, que están a la disposición del programador para agregarlos a la aplicación principal y añadirle funcionalidades. Para el desarrollo del sistema se utilizaron tres *bundles* los cuales se mencionan a continuación.

- **ExcelBundle:** este paquete (o *bundle*) fue utilizado para la creación de la plantilla en Excel donde se ingresan los datos de los carnets que el cliente desee importar, del mismo modo para recibir los datos al momento de que el cliente importe la plantilla llena con información sobre los carnets.

- **SnappyBundle:** este *bundle* permitió la generación de los reportes en PDF que el administrador solicitara, permitiendo estructurar el documento y llenarlo con la información solicitada.

- **HighchartsBundle:** fue utilizado para generar los gráficos que se muestran al momento del administrador consultar alguna estadística. Permitted generar diversos gráficos que se adaptaran a las necesidades del cliente y permitieran visualizar la información de interés de una manera agradable, interactiva y precisa.

4.3.3. Librerías externas

- **jQuery:** esta librería está escrita en el lenguaje *javascript* y por lo tanto se ejecuta del lado del cliente, básicamente fue utilizada con el objetivo de escribir el código en dicho lenguaje de manera más ágil y óptima, además de aprovechar las funciones adicionales que ofrece lo que permitió realizar actividades en el sistema como la característica *Drag N' Drop* en la edición de los diseños de los carnets y otras interacciones con el usuario sin necesidad de recargar la página completamente.

- **imageCrop.js, jsPDF.js:** estas librerías, junto a otras utilizadas para su correcto funcionamiento están escritas en lenguaje *JavaScript*, y fueron implementadas al momento de que el cliente pudiera ajustar las fotos al tamaño acorde para el carnet y el empleado pudiera exportar los carnets a un documento PDF para imprimirlos posteriormente.

- **foundation.css, foundation.js:** con la finalidad de hacer el sistema más agradable al uso y a la vista del usuario, se implementaron estas dos librerías principalmente, creadas ambas por la misma fundación *ZURB Foundation*, las cuales facilitaron la codificación de la interfaz de usuario, así como algunas interacciones del sistema con el usuario.

4.3.4. Plugins

Fueron utilizados además *plugins* adicionales, con la finalidad de hacer la experiencia del usuario aún más agradable al momento de utilizar el sistema. Algunos de ellos se pueden encontrar funcionando al momento de visualizar el catálogo de accesorios dentro del sistema, el cual cuenta con un carrusel en donde se muestran

tarjetas con los productos; o al consultar alguna tabla del sistema, donde se permite al usuario filtrar los datos mostrados en la tabla a través de un cuadro de búsqueda, así como ordenar las filas de manera ascendente o descendente de acuerdo a una columna específica.

4.3.5. Pantallas del Sistema

A continuación se presentan las pantallas principales que forman parte del sistema, con el motivo de orientar al usuario de una mejor manera para el uso de las aplicaciones y así lograr una mejor experiencia de usuario.

La siguiente pantalla corresponde al módulo de ingreso al sistema web, donde el usuario se identifica con sus datos y accede a la aplicación.



Bienvenido al sistema de Carnetización Online de nuestra empresa

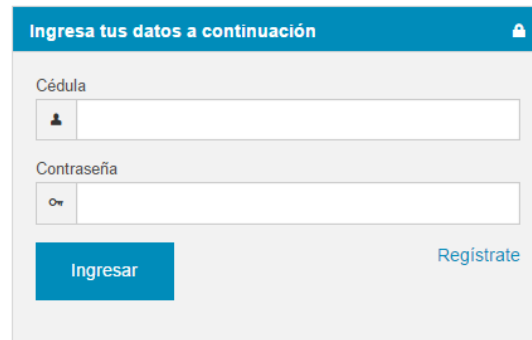
A login form with a blue header bar containing the text 'Ingresa tus datos a continuación' and a lock icon. Below the header, there are two input fields: 'Cédula' with a person icon and 'Contraseña' with an eye icon. At the bottom, there is a blue 'Ingresar' button and a blue 'Regístrate' link.

Figura 18. Pantalla del Sistema: Ingreso

Fuente: Emilio Clemente (2015)

La pantalla a continuación muestra el módulo en donde el usuario (cliente) puede administrar los carnets creados, distribuidos por diseños.



Figura 19. Pantalla del Sistema: Base de Datos

Fuente: Emilio Clemente (2015)

Una de las pantallas más importantes del sistema es la que se presenta a continuación, donde el usuario edita el diseño de acuerdo a sus necesidades.



Figura 20. Pantalla del Sistema: Edición de Diseño

Fuente: Emilio Clemente (2015)

La función del administrador del sistema para administrar los campos, se puede realizar en la pantalla siguiente, donde se muestran los campos solicitados por los clientes, así como los existentes y un pequeño formulario donde se puede agregar un campo en caso de requerirlo

Campos Solicitados













Mostrar registros Buscar:

Usuario	Campo	Mensaje	Estado
Emilio Clemente	Nombre	Este campo es esencial	Finalizada
Emilio Clemente	Foto	Deseo agregarle fotos a los carnets	Finalizada
No Existe	Algún Nombre	Aquí pondría el mensaje	Pendiente

Mostrando registros del 1 al 3 de un total de 3 registros Anterior **1** Siguiente

Campos Existentes

Mostrar registros Buscar:

Campo	Acción
Nombre	 
Foto	 
Apellido	 
Cédula	 
Fecha de Vencimiento	 
Código	 

Mostrando registros del 1 al 6 de un Anterior **1** Siguiente

Agregar Campo

Campo

Figura 21. Pantalla del Sistema: Administración de Campos
Fuente: Emilio Clemente (2015)

Al momento del usuario solicitar un pedido, se le muestra la pantalla que se presenta a continuación, permitiendo a este observar un resumen de la cotización, así como el monto por artículo y el monto total, además de poder agregar y remover dichos artículos del pedido. Por otra parte, posee un cuadro de texto donde puede indicar un comentario si así lo desea.

Accesorios agregados

Nombre	Color	Descripción	Precio	Cantidad	Total	Acción
Camet	Blanco	PVC	600	2	1200	
Gancho	Verde	Plástico	36.64	2	73.28	Eliminar

Nota o sugerencia

Si posee alguna sugerencia o comentario puede escribirlo en este recuadro

Total: 1273.28

Finalizar

Accesorios disponibles

Ganchos Cintas Portacarnets Otros



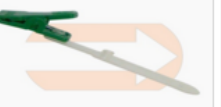

Gancho	Gancho	Gancho	Gancho
			
36.64	36.96	36.64	36.64
Accesorio	Accesorio	Accesorio	Accesorio
Rojo	Azul	Verde	Blanco
Plástico	Plástico	Plástico	Plástico
Agregar	Agregar	Agregado	Agregar

Figura 22. Pantalla del Sistema: Solicitud de Pedido

Fuente: Emilio Clemente (2015)

En cuanto a la aplicación móvil, esta cuenta con un menú desplegable en donde se encuentran las principales funcionalidades que puede ejercer el usuario.

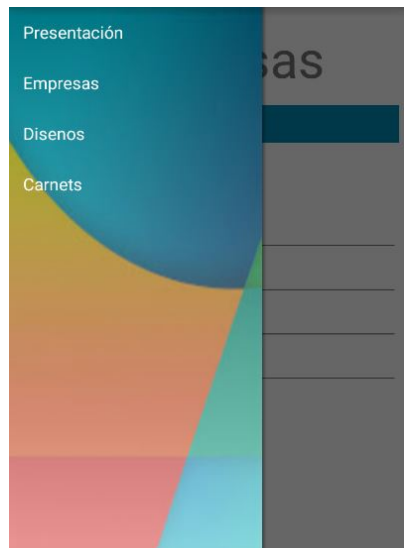


Figura 23. Pantalla del Sistema: Menú Móvil
Fuente: Emilio Clemente (2015)

La creación de una nueva empresa es otra de las funciones con las que cuenta la aplicación móvil y se presenta en la pantalla a continuación.

A screenshot of the 'Empresas' screen in the mobile application. The title 'Empresas' is at the top. Below it is a blue bar that says 'Conectado con: Mi Empresa, C.A.'. Underneath is the text 'Registrar nueva empresa'. There are four input fields: 'Razón Social', 'RIF', 'Dirección', and 'Teléfono'. At the bottom is a blue button labeled 'REGISTRAR'.

Figura 24. Pantalla del Sistema: Empresas Móvil
Fuente: Emilio Clemente (2015)

De igual manera, se encuentra el módulo para agregar un nuevo diseño, en donde se puede especificar el nombre de este, su orientación y los campos que contendrá.

Diseños

Conectado con: Mi Empresa, C.A.

Agregar nuevo diseño

Nombre del diseño

Orientación

Vertical Horizontal

Campos

Cédula

Nombre

Apellido

Foto

AGREGAR

Figura 25. Pantalla del Sistema: Diseño Móvil

Fuente: Emilio Clemente (2015)

4.4. Fase IV: Pruebas

Durante todo el desarrollo del sistema, se fueron realizando pruebas a los módulos y funcionalidades del mismo, esto para garantizar el funcionamiento adecuado del sistema en los diferentes escenarios que se puedan presentar. Se realizaron pruebas de caja blanca y caja negra, como apoyo a las pruebas unitarias y de aceptación, de las cuales se obtuvieron los resultados requeridos luego de realizar las correcciones necesarias. Por otro lado, se realizaron pruebas seguridad para asegurar la integridad de los datos en el sistema.

Cuadro 51. Prueba 1: Solicitud de Pedido

Solicitud de Pedido	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Se comprobó que el cliente pueda seleccionar los carnets que requiera imprimir, así como los accesorios que desee y solicitar un pedido a la empresa de carnetización.	
Entrada: N/A	Historia de usuario: Solicitud de Pedido
Resultado: Se muestra el detalle del pedido correctamente	
Observaciones: De requerirlo, el usuario puede agregar nota al pedido	
Correcciones: Ninguna	

Fuente: Emilio Clemente. (2015)

Cuadro 52. Prueba 2: Edición de Diseño

Edición de Diseño	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Se comprobó que se puede editar un diseño ya creado, seleccionando el fondo, ubicando los campos correspondientes y estableciendo el reverso	
Entrada: N/A	Historia de usuario: Edición de Diseño
Resultado: El diseño se puede editar correctamente	
Observaciones: el diseño una vez que este creado se puede editar por el cliente.	
Correcciones: Ninguna	

Fuente: Emilio Clemente. (2015)

Cuadro 53. Prueba 3: Administrar Empresa

Administrar Empresa	
Prueba: Aceptación	Tipo: Caja negra
Descripción: comprobar si se registró una nueva empresa, de igual manera poder eliminarla en caso de ser necesario	
Entrada: rif, razón social, dirección y teléfono	Historia de usuario: Administrar Empresa
Resultado: Se creó la empresa exitosamente	
Observaciones: No se muestran errores cuando alguno de los campos requeridos son incorrectos.	
Correcciones: Señalar los errores que existen en los campos requeridos.	

Fuente: Emilio Clemente. (2015)

Cuadro 54. Prueba 4: Visualización de Catalogo

Visualización de Catalogo	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Mostrar los productos que ofrece la empresa, con su respectiva descripción y precio	
Entrada: N/A	Historia de usuario: Visualización de Catálogo
Resultado: Se muestra correctamente todos los productos que posee actualmente la empresa	
Observaciones: Ninguna	
Correcciones: Ninguna	

Fuente: Emilio Clemente. (2015)

Cuadro 55. Prueba 5: Cargar archivos

Cargar archivos	
Prueba: Unitaria	Tipo: Caja blanca
Descripción: Se comprobó que la información en el archivo Excel cargado por el usuario sea el correspondiente a los campos del diseño	
Clase: Phpexcel	Función: getValue
Entrada: campo (String)	
Valor de entrada: \$campo = nombre	Valor esperado: 1
Resultado: Correcto	
Observaciones: La prueba fue realizada con los distintos campos del Diseño, en la totalidad de los casos, los resultados fueron correctos	

Fuente: Emilio Clemente. (2015)

Cuadro 56. Prueba 6: Generar gráficas de pedidos

Generar gráficas de pedidos	
Prueba: Unitaria	Tipo: Caja blanca
Descripción: Se comprobó que los datos enviados desde el controlador a la vista correspondieran a los solicitados por el usuario	
Clase: Highchart	Función: render
Entrada: fecha_desde (DateTime), fecha_hasta (DateTime), empresa (String)	
Valores de entrada: \$fecha_desde: 01-10-15 00:00:00 \$fehca_hasta: 25-10-15 23:59:59 \$empresa: J-00000000-0	Valor esperado: Objeto Highchart
Resultado: Correcto	
Observaciones: La prueba fue realizada sin valores de fecha, lo cual retorna pedidos desde el primero creado hasta la fecha actual	

Fuente: Emilio Clemente. (2015)

Cuadro 57. Prueba 7: Exportar carnets a PDF

Exportar carnets a PDF	
Prueba: Unitaria	Tipo: Caja blanca
Descripción: Se comprobó que los carnets mostrados en el archivo PDF coincidieran los con carnets incluidos en el pedido	
Clase: JsPDF	Función: addHTML
Entrada: \$carnet (Objeto DOM)	
Valores de entrada: \$carnet: object	Valor esperado: Archivo PDF
Resultado: Correcto	
Observaciones: La prueba fue realizada con distintos diseños y pedidos, retornando el archivo PDF con sus dimensiones y datos correspondientes.	

Fuente: Emilio Clemente. (2015)

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

Una vez finalizada la investigación del proyecto y completadas las cuatro fases que conformaron la metodología para la realización de un sistema para la optimización de la elaboración de carnets en la empresa Carnetiza, C.A. Se obtuvieron las conclusiones que a continuación se presentan.

- Se detectaron las principales fallas y problemas en el proceso de elaboración de carnets en la empresa.

- En base a las diferentes problemáticas encontradas se creó una lista de requerimientos para realizar un sistema que contara con una aplicación web y móvil, que permite el diseño y elaboración de carnets.

- Con la realización del sistema que ofrece una interfaz de elaboración de carnets para los clientes de la empresa, se solucionó la problemática planteada en el capítulo I de esta investigación.

- Se logró una fácil adaptación por parte de los usuarios ya que se puede acceder al sistema de manera nativa tanto desde un dispositivo móvil con sistema operativo Android como de un navegador web, lo cual evita una instalación compleja para utilizar el sistema.

- El sistema satisface tanto las necesidades diagnosticadas por la investigación que se llevó a cabo como las indicadas por los empleados de la empresa Carnetiza, C.A.

- El sistema se encuentra debidamente estructurado y documentado, dando lugar a posibles mejoras sin mayor dificultad. Del mismo modo, las herramientas utilizadas para el desarrollo del sistema son gratis y accesibles para cualquier usuario.

- La utilización de la metodología *eXtreme Programming* permitió una adecuada interacción entre el programador y el cliente, posibilitando una constante mejora de los requerimientos del sistema, proporcionando así un sistema más ajustado a las necesidades de la empresa.

- La empresa al contar con el sistema pudo reducir la carga de trabajo de los empleados, así como los errores cometidos al momento de ingresar los datos de los carnets manualmente.

- La posibilidad de realizar pedidos con detalles de accesorios y una nota adicional, mejoró el tiempo de respuesta de la empresa en la impresión de carnets requeridos por el cliente.

5.2 Recomendaciones

Con la finalidad de mantener el sistema en constante mejora y optimización, a continuación se plantean una serie de recomendaciones, de igual manera se indican mejoras que se pueden implementar para ampliar la funcionalidad del sistema, las cuales no se encuentran previstas en el alcance de esta investigación.

- Utilizar las versiones correspondientes de los lenguajes PHP y MySQL para el servidor web compatibles con la aplicación web desarrollada.

- Evitar realizar cambios mayores a la interfaz del sistema, ya que puede causar una confusión en los usuarios regulares del sistema.

- Mantener los precios de los carnets y accesorios actualizados para brindar al cliente la mejor experiencia al momento de realizar un pedido y este observe la cotización del mismo.

- Ofrecer al usuario plantillas de carnets ya elaboradas, con los campos ubicados de manera coherente y con fuentes, colores, y fondos ya preestablecidos.

- Desarrollar el sistema para otras plataformas como iOS o Windows Phone que permitan el acceso al sistema a los usuarios que posean dispositivos móviles con distintos sistemas operativos.

- Desarrollar un módulo que mantenga comunicación con el correo electrónico de la empresa, notificando a esta de la información de pedidos realizados en tiempo real a través de esa vía.

- Establecer una interfaz de comunicación entre el sistema administrativo de la empresa y el sistema web que mantenga los precios tanto de los carnets como de los accesorios actualizados.

- Establecer comunicación entre el sistema y la unidad de impresión, de manera que sea directa e inmediata la impresión de los carnets, sin necesidad de utilizar un generador de archivos PDF como intermediario.

REFERENCIAS

Referencias Bibliográficas

- Bavaresco, Aura M. (2006). **Proceso Metodológico en la investigación: Cómo hacer un Diseño de Investigación**. Maracaibo: EDILUZ. [12 de abril del 2015].
- C. J. Date (2001). **Introducción a los sistemas de bases de datos**. Pearson Educación [25 de abril del 2015].
- Ian Sommerville. (2005). **Ingeniería del Software**. Madrid: Pearson Education. [22 de abril del 2015].
- Jesús León Subero. (2004). **Material sobre verbos, palabras de enlaces, bibliografía, etc**. Universidad Centro Occidental Lisandro Alvarado. [02 de mayo del 2015].
- Kendall K., y Kendall, J. (2005). **Análisis y Diseño de Sistemas**. México: Pearson Educación. [20 de abril del 2015].
- Mijares, H. y García, L. (2007). **Normas para la elaboración y presentación de los anteproyectos, proyectos y trabajos de grado**. Universidad José Antonio Páez. [28 de marzo del 2015].
- Ramos, A., Ramos, M. y Montero, F. (2006). **Sistemas gestores de bases de datos: Guía didáctica**. España: McGraw-Hill. [27 de abril del 2015].
- Roberto Hernández Sampieri. (1991). **Metodología de la Investigación**. México: McGraw Hill. [11 de abril del 2015].
- Tamayo y Tamayo, M. (2003). **El Proceso de la Investigación Científica**. México: Limusa, S.A.
- UPEL. (2007). **Manual para la elaboración de Tesis Doctorales, Trabajos de Grado y Trabajos Especiales**. Universidad Central de Venezuela. [22 de marzo del 2015].

- Vicen Fernández Alarcón. (2010). **Desarrollo de sistemas de información: una metodología basada en el modelado**. Ediciones Upc Universidad Politécnica de Catalunya [20 de abril del 2015].
- Whitten Bently Dittman. (2004). **Análisis de sistemas y métodos de diseño**. McGraw Hill [28 de abril del 2015].

Referencias en Línea

- Ambler, Scott W. (2014). [Página Web en Línea]. **Introduction to the Diagrams of UML 2.X**. <http://www.agilemodeling.com/essays/umlDiagrams.htm>. [22 de mayo del 2015].
- Da Silva Almeida, Sergio Daniel. (2013). [Documento Digital]. **Desarrollo de un sistema de información para la gestión de los archivos inactivos en la empresa Distribuidora Universal Kia, C.A.** Universidad José Antonio Páez. [08 de abril del 2015].
- Don Wells. (2013). [Página Web en Línea]. **Extreme Programming**. <http://www.extremeprogramming.org/>. [02 de mayo del 2015].
- Mieres, A., Bultó, Y., Castillo, J., Serrano, L. y Guevara, J. (2013). [Documento Digital]. **Implantación de un Sistema de Información Automatizado bajo ambiente web para el proceso de inscripción en la Escuela Bolivariana “José Antonio Ramos Sucre” Ubicada en la calle Zea, sector Río Manzanares, Parroquia Altagracia, Municipio Sucre, Cumaná, Edo Sucre 2012-2013**. Universidad Politécnica Territorial del Oeste de Sucre. Cumaná. [02 de abril del 2015].
- Noguera, Victoria. (2013). [Documento Digital]. **Desarrollo de un sistema de información de soporte a partir de patrones de diseño utilizando Symfony en la empresa ICAD Diseño Inteligente**. Universidad José Antonio Páez. [06 de abril del 2015].
- Potencier F. y Zaninotto F. [Página Web en Línea]. **La guía definitiva para Symfony**. http://symfony.com/legacy/doc/book/1_0. [12 de mayo del 2015].