

**ESCUELA MILITAR DE INGENIERÍA  
MCAL. ANTONIO JOSÉ DE SUCRE  
“BOLIVIA”**

**PROYECTO DE GRADO**



**SISTEMA DE GESTIÓN DE PENSIONES CON FACTURACIÓN  
COMPUTARIZADA. CASO DE ESTUDIO: UNIDAD EDUCATIVA  
MARÍA AUXILIADORA CRISTO REY**

**RICHARD LORENZO SEVERICH PAREDES**

**COCHABAMBA 2015**

**ESCUELA MILITAR DE INGENIERÍA  
MCAL. ANTONIO JOSÉ DE SUCRE  
“BOLIVIA”**

**PROYECTO DE GRADO**

**SISTEMA DE GESTIÓN DE PENSIONES CON FACTURACIÓN  
COMPUTARIZADA. CASO DE ESTUDIO: UNIDAD EDUCATIVA  
MARÍA AUXILIADORA CRISTO REY**

**RICHARD LORENZO SEVERICH PAREDES**

**Modalidad: Proyecto de grado  
presentado como requisito  
para optar al título de  
Licenciado en Ingeniería de  
Sistemas**

**TUTOR: ING. MSC. YANINA GALABURDA**

**COCHABAMBA 2015**

# ÍNDICE DE CONTENIDO

TITULO	PÁGINA
<b>1. GENERALIDADES.....</b>	<b>17</b>
<b>1.1. INTRODUCCIÓN.....</b>	<b>17</b>
<b>1.2. ANTECEDENTES.....</b>	<b>18</b>
<b>1.3. PLANTEAMIENTO DEL PROBLEMA. ....</b>	<b>21</b>
1.3.1. Identificación del problema. ....	21
1.3.1.1. Identificación de la situación problemática. ....	21
1.3.1.2. Identificación de las causas. ....	22
1.3.2. Formulación del problema. ....	22
1.3.3. Análisis causa efecto. ....	23
<b>1.4. OBJETIVOS Y ACCIONES. ....</b>	<b>23</b>
1.4.1. Objetivo general.....	23
1.4.2. Objetivos específicos y acciones. ....	23
<b>1.5. JUSTIFICACIÓN. ....</b>	<b>28</b>
1.5.1. Justificación técnica. ....	28
1.5.2. Justificación económica. ....	28
1.5.3. Justificación social. ....	29
1.5.4. Justificación operativa.....	29
<b>1.6. ALCANCE .....</b>	<b>29</b>
1.6.1. Alcance temático. ....	30
1.6.2. Alcance institucional. ....	30

1.6.3. Alcance temporal .....	30
<b>1.7. HIPÓTESIS.....</b>	<b>30</b>
1.7.1. Análisis de variables.....	30
Variable Independiente.....	30
Variables Dependientes .....	31
1.7.2. Definición conceptual.....	31
1.7.3. Operativización de variables.....	32
<b>1.8. MATRIZ DE CONSISTENCIA.....</b>	<b>33</b>
<b>2. MARCO TEÓRICO.....</b>	<b>34</b>
<b>2.1. TÉCNICAS DE RECOLECCIÓN DE INFORMACIÓN.....</b>	<b>34</b>
2.1.1. Observación.....	34
2.1.2. Entrevistas .....	35
2.1.3. Cuestionarios.....	36
<b>2.2. APLICACIONES WEB MULTIPLATAFORMA.....</b>	<b>37</b>
2.2.1. Características de las aplicaciones web multiplataforma.....	37
2.2.2. Arquitecturas de aplicaciones web multiplataforma. ....	39
2.2.2.1. Modelo Cliente – Servidor.....	39
2.2.2.2. Patrón modelo-vista-controlador .....	42
<b>2.3. FACTURACIÓN COMPUTARIZADA.....</b>	<b>44</b>
<b>2.3.1. La factura. ....</b>	<b>44</b>
<b>2.3.2. Tipos de factura.....</b>	<b>45</b>
<b>2.3.3. Modalidad facturación computarizada. ....</b>	<b>45</b>
<b>2.3.4. Fecha límite de emisión.....</b>	<b>47</b>

<b>2.3.5. Aspectos técnicos de las facturas.....</b>	<b>47</b>
<b>2.4. INGENIERÍA DE SOFTWARE .....</b>	<b>51</b>
2.4.1. Modelo incremental.....	51
2.4.2. Modelo Espiral.....	52
2.4.3. Modelo Basado en Componentes.....	54
2.4.4. Pruebas De Software.....	56
2.4.4.1. Pruebas de Integración.....	56
2.4.4.2. Pruebas de despliegue .....	57
2.4.4.3. Pruebas de Aceptación.....	57
2.4.4.4. Pruebas Funcionales .....	57
<b>2.5. HERRAMIENTAS DE DESARROLLO.....</b>	<b>58</b>
2.5.1. Frameworks para el desarrollo de aplicaciones web.....	58
2.5.1.1. Codelgniter.....	58
2.5.1.2. PhoneGap.....	60
2.5.1.3. JQuery Mobile.....	61
2.5.2. Lenguajes de Programación para desarrollo de aplicaciones web.....	62
2.5.2.1. Php.....	62
2.5.2.2. Java .....	63
2.5.2.3. ASP.NET .....	64
2.5.3. Herramientas para el desarrollo de aplicaciones web.....	65
2.5.3.1. Html5.....	65
2.5.3.2. Java Script .....	67
2.5.3.3. CSS3.....	68

2.5.3.4. JQuery .....	69
<b>2.6. GESTORES DE BASES DE DATOS.....</b>	<b>71</b>
2.6.1. MySQL.....	71
2.6.2. SQL Server 2012.....	72
2.6.3. PostgresSQL.....	73
<b>2.7. Librerías.....</b>	<b>76</b>
2.7.1. Librería PHP Código QR.....	76
2.7.2. Librería FPDF pdf generador.....	77
<b>3. MARCO PRÁCTICO.....</b>	<b>78</b>
<b>3.1. DISEÑO DEL MODELADO DE NEGOCIO ALTERNATIVO DEL PROYECTO DE GESTIÓN DE PENSIONES.....</b>	<b>78</b>
3.1.1. Modelado de negocio actual.....	79
3.1.2. Evaluación de las deficiencias existentes en el proceso actual.....	80
3.1.3. Modelado de Negocio Alternativo.....	81
<b>Figura 13: Modelado de Negocio Alternativo.....</b>	<b>81</b>
3.1.3.1. Proceso de Inscripciones y Pago de Pensiones.....	82
3.1.3.2. Proceso de Averiguar Pensiones e Inicio de Gestión.....	83
<b>3.2. DESARROLLO DEL SUBSISTEMA MULTIPLATAFORMA DE GESTIÓN DE USUARIOS, NIVELES EDUCATIVOS CON MONTOS DE MENSUALIDADES POR GESTIÓN Y PLANIFICACIÓN DE CURSOS.....</b>	<b>84</b>
3.2.1. Selección del modelo de desarrollo de software.....	84
3.2.2. Planificación de Incrementos del proyecto según a la metodología seleccionada.....	87

3.2.3.	Selección de lenguajes de programación .....	88
3.2.4.	Selección de framework.....	90
3.2.5.	Selección del Gestor de Bases de datos .....	94
3.2.6.	Análisis .....	97
3.2.6.1.	Requerimientos del sistema .....	97
3.2.6.2.	Identificación de Actores .....	98
3.2.6.3.	Diagramas de casos de uso por actor.....	98
3.2.6.4.	Diagrama de Casos de Uso del sistema .....	99
3.2.6.5.	Descripción de cada caso de uso .....	99
3.2.7.	Diseño.....	102
3.2.7.1.	Diagramas de Colaboración.....	103
3.2.7.2.	Diagrama de clases .....	104
3.2.7.3.	Diagrama de base de datos .....	105
3.2.7.4.	Diccionario de datos.....	105
3.2.8.	Implementación .....	108
3.2.8.1.	Arquitectura del Software .....	108
3.2.8.2.	Organización del proyecto .....	112
3.2.8.3.	Interfaces del sistema .....	115
3.2.9.	Pruebas .....	122
3.2.9.1.	Pruebas Funcionales .....	122
<b>3.3.</b>	<b>DISEÑO E IMPLEMENTACIÓN DEL MODULO DE INSCRIPCIONES DE ESTUDIANTES INTEGRADO AL SUBSISTEMA MULTIPLATAFORMA .....</b>	<b>124</b>

3.3.1.	Análisis .....	124
3.3.1.1.	Requerimientos del sistema .....	124
3.3.1.2.	Identificación de actores .....	125
3.3.1.3.	Diagrama de casos de uso por actor .....	125
3.3.1.4.	Diagrama de casos de uso del sistema .....	126
3.3.1.5.	Descripción de cada caso de uso .....	127
3.3.2.	Diseño.....	129
3.3.2.1.	Diagramas de colaboración .....	129
3.3.2.2.	Diagrama de clases .....	130
3.3.2.3.	Diagrama de base de datos .....	131
3.3.2.4.	Diccionario de datos.....	132
3.3.3.	Implementación .....	136
3.3.3.1.	Arquitectura del Software.....	136
3.3.3.2.	Organización del proyecto .....	138
3.3.3.3.	Interfaces del sistema .....	139
3.3.4.	Pruebas .....	143
3.3.4.1.	Pruebas funcionales.....	143
<b>3.4.</b>	<b>DESARROLLO DE LOS MÓDULOS DE COBRO DE PENSIONES CON FACTURACIÓN Y REPORTES INTEGRADO AL SUB SISTEMA MULTIPLATAFORMA.....</b>	<b>144</b>
3.4.1.	Análisis .....	144
3.4.1.1.	Análisis del código de control.....	145
3.4.1.2.	Requerimiento del sistema.....	148

3.4.1.3. Identificación de Actores .....	148
3.4.1.4. Casos de Uso por Actor .....	149
3.4.1.5. Casos de Uso del sistema .....	150
3.4.1.6. Descripción de casos de uso .....	152
3.4.2. Diseño.....	155
3.4.2.1. Diagramas de colaboración .....	155
3.4.2.2. Diagrama de clases .....	157
3.4.2.3. Diagrama de base de datos .....	158
3.4.2.4. Diccionario de datos.....	159
3.4.3. Implementación .....	161
3.4.3.1. Arquitectura del software .....	161
3.4.3.2. Organización del Proyecto .....	163
3.4.3.3. Interfaces del sistema .....	165
3.4.4. Pruebas .....	170
3.4.4.1. Pruebas Funcionales .....	170
3.4.4.2. Pruebas al código de control.....	171
<b>3.5. PRUEBAS FINALES AL SISTEMA MULTIPLATAFORMA .....</b>	<b>173</b>
<b>3.6. DEMOSTRACIÓN DE LA HIPÓTESIS.....</b>	<b>178</b>
3.6.1. Demostración de la primera variable dependiente.....	178
3.6.2. Demostración de la segunda variable dependiente .....	180
3.6.3. Demostración de la tercera variable dependiente.....	182
3.6.4. Demostración de la variable independiente .....	183
3.6.5. Definición de la hipótesis .....	185

3.6.6. Calculo del estadístico t.....	186
<b>4. ANÁLISIS DE VIABILIDAD.....</b>	<b>190</b>
<b>4.1. VIABILIDAD TÉCNICA .....</b>	<b>190</b>
<b>4.2. VIABILIDAD ECONÓMICA.....</b>	<b>193</b>
<b>4.3. VIABILIDAD OPERATIVA .....</b>	<b>207</b>
<b>5. CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>209</b>
<b>5.1. CONCLUSIONES.....</b>	<b>209</b>
<b>5.2. RECOMENDACIONES .....</b>	<b>212</b>
<b>BIBLIOGRAFÍA.....</b>	<b>214</b>

## **ANEXOS**

## ÍNDICE DE TABLAS

TITULO	PÁGINA
<b>Tabla 1:</b> Objetivos específicos y acciones .....	24
<b>Tabla 2:</b> Operativización de variables.....	32
<b>Tabla 3:</b> Matriz de consistencia.....	33
<b>Tabla 4:</b> Contenido del código QR para facturación computarizada.....	46
<b>Tabla 5:</b> Fecha límite de emisión de facturas según la modalidad .....	47
<b>Tabla 6:</b> Deficiencias existentes en el proceso actual.....	80
<b>Tabla 7:</b> Comparación entre modelos de desarrollo de software.....	84
<b>Tabla 8:</b> Comparación entre Lenguajes .....	88
<b>Tabla 9:</b> Comparación entre Frameworks .....	90
<b>Tabla 10:</b> Comparación entre gestores de base de datos. ....	94
<b>Tabla 11:</b> Descripción de caso de uso ingresar al sistema .....	99
<b>Tabla 12:</b> Descripción de caso de uso gestionar usuarios.....	100
<b>Tabla 13:</b> Descripción de caso de uso año educativo.....	101
<b>Tabla 14:</b> Descripción de caso de uso gestionar cursos. ....	102
<b>Tabla 15:</b> Diccionario de datos tabla usuarios .....	105
<b>Tabla 16:</b> Diccionario de datos tabla gestión .....	107
<b>Tabla 17:</b> Diccionario de datos tabla cursos .....	108
<b>Tabla 18:</b> Pruebas de funcionalidad de primer incremento.....	122
<b>Tabla 19:</b> Descripción del caso de uso gestionar estudiantes. ....	127
<b>Tabla 20:</b> Descripción del caso de uso gestionar inscripciones.....	128

<b>Tabla 21:</b> Diccionario de datos tabla estudiantes.	132
<b>Tabla 22:</b> Diccionario de datos tabla inscripciones.	133
<b>Tabla 23:</b> Diccionario de datos tabla pensiones.	134
<b>Tabla 24:</b> Pruebas de funcionalidad del incremento.	143
<b>Tabla 25:</b> Descripción del caso de uso gestionar dosificación.	152
<b>Tabla 26:</b> Descripción del caso de uso gestionar pago de pensiones.	153
<b>Tabla 27:</b> Descripción del caso de uso ver estado de pensiones.	154
<b>Tabla 28:</b> Diccionario de datos tabla dosificación.	159
<b>Tabla 29:</b> Diccionario de datos tabla pensiones.	160
<b>Tabla 30:</b> Pruebas funcionales del incremento.	170
<b>Tabla 31:</b> Tabla comparativa del sistema.	173
<b>Tabla 32:</b> Demostración de la primera variable dependiente	179
<b>Tabla 33:</b> Demostración de la segunda variable dependiente	181
<b>Tabla 34:</b> Demostración de la tercera variable dependiente	182
<b>Tabla 35:</b> Demostración e variable independiente	183
<b>Tabla 36:</b> Requerimientos del servidor.	190
<b>Tabla 37:</b> Requerimientos para el equipo-usuarios.	191
<b>Tabla 38:</b> Requerimientos para la tablet o smartphone	192
<b>Tabla 39:</b> Costo del servidor	193
<b>Tabla 40:</b> Costos del equipo-usuarios.	194
<b>Tabla 41:</b> Costo de las tablets o smartphones	195
<b>Tabla 42:</b> Valores constantes	197
<b>Tabla 43:</b> Factores de costo en COCOMO	200

**Tabla 44:** Costos mínimos y recomendados del proyecto..... 204

**Tabla 45:** Análisis costo beneficio..... 205

# ÍNDICE DE FIGURAS

TITULO	PÁGINA
<b>Figura 1:</b> Organigrama .....	19
<b>Figura 2:</b> Adaptabilidad a diferentes resoluciones .....	38
<b>Figura 3:</b> Estructura de un sistema multiplataforma.....	39
<b>Figura 4:</b> Ciclo cliente - servidor .....	40
<b>Figura 5:</b> Ciclo de vida MVC .....	43
<b>Figura 6:</b> Modelo incremental .....	51
<b>Figura 7:</b> Modelo espiral .....	53
<b>Figura 8:</b> Modelo basado en componentes.....	55
<b>Figura 9:</b> Flujo de control de pruebas de integración.....	56
<b>Figura 10:</b> Diagrama de codelgniter. ....	59
<b>Figura 11:</b> Modelado de negocio actual.....	79
<b>Figura 12:</b> Modelado de negocio alternativo.....	81
<b>Figura 13:</b> Proceso de inscripciones y pago de pensiones.....	82
<b>Figura 14:</b> Proceso de averiguar pensiones e inicio de gestión.....	83
<b>Figura 15:</b> Casos de uso por actor. ....	98
<b>Figura 16:</b> Casos de uso general del sistema .....	99
<b>Figura 17:</b> Diagrama de colaboración usuarios .....	103
<b>Figura 18:</b> Diagrama de colaboración año educativo. ....	103
<b>Figura 19:</b> Diagrama de colaboración cursos .....	104
<b>Figura 20:</b> Diagrama de clases.....	104

<b>Figura 21:</b> Diagrama de base de datos .....	105
<b>Figura 22:</b> Arquitectura para el caso de uso gestionar usuarios.....	109
<b>Figura 23:</b> Arquitectura para el caso de uso gestionar año educativo.....	110
<b>Figura 24:</b> Arquitectura para el caso de uso gestionar cursos.....	111
<b>Figura 25:</b> Primer nivel del proyecto.....	112
<b>Figura 26:</b> Contenido de la carpeta application.....	113
<b>Figura 27:</b> Contenido de la carpeta views.....	114
<b>Figura 28:</b> Contenido de la carpeta controllers.....	114
<b>Figura 29:</b> Contenido de la carpeta models.....	114
<b>Figura 30:</b> Interface de la página principal-galería de fotos.....	115
<b>Figura 31:</b> Interface de la página principal-ingresar al sistema.....	115
<b>Figura 32:</b> Interface de la página principal- información.....	116
<b>Figura 33:</b> Interface de la página principal- contáctanos.....	116
<b>Figura 34:</b> Interface ingresar al sistema usuarios.....	117
<b>Figura 35:</b> Interface menú principal .....	117
<b>Figura 36:</b> Interface menú gestionar usuarios.....	118
<b>Figura 37:</b> Interface menú gestionar año educativo.....	118
<b>Figura 38:</b> Interface menu gestionar cursos.....	118
<b>Figura 39:</b> Interface registro de usuarios.....	119
<b>Figura 40:</b> Interface registro año educativo.....	119
<b>Figura 41:</b> Interface de registro cursos.....	120
<b>Figura 42:</b> Interface mostrar usuarios.....	120
<b>Figura 43:</b> Interface mostrar gestiones.....	121

<b>Figura 44:</b> Interface mostrar cursos.....	121
<b>Figura 45:</b> Casos de uso por actor. ....	125
<b>Figura 46:</b> Casos de uso del sistema. ....	126
<b>Figura 47:</b> Diagrama de colaboración estudiantes. ....	129
<b>Figura 48:</b> Diagrama de colaboración inscripciones.....	129
<b>Figura 49:</b> Diagrama de clases.....	130
<b>Figura 50:</b> Diagrama de base de datos. ....	131
<b>Figura 51:</b> Arquitectura para el caso de uso gestionar estudiantes. ....	136
<b>Figura 52:</b> Arquitectura para el caso de uso gestionar inscripciones.....	137
<b>Figura 53:</b> Contenido de la carpeta views. ....	138
<b>Figura 54:</b> Contenido de la carpeta controllers. ....	138
<b>Figura 55:</b> Contenido de la carpeta models.....	139
<b>Figura 56:</b> Interface menú principal. ....	139
<b>Figura 57:</b> Interface gestionar estudiantes. ....	140
<b>Figura 58:</b> Interface gestionar inscripciones. ....	140
<b>Figura 59:</b> Interface registro estudiantes. ....	141
<b>Figura 60:</b> Interface registro inscripciones elegir curso. ....	141
<b>Figura 61:</b> Interface registro inscripciones elegir estudiantes. ....	142
<b>Figura 62:</b> Interface mostrar estudiantes. ....	142
<b>Figura 63:</b> Interface mostrar inscripciones.....	143
<b>Figura 64:</b> Algoritmos utilizados para la generación del código de control....	145
<b>Figura 65:</b> Datos iniciales para el código de control. ....	146
<b>Figura 66:</b> Datos obtenidos por el algoritmo verhoeff.....	146

<b>Figura 67:</b> Sumatoria de los datos obtenidos por el algoritmo verhoeff.....	146
<b>Figura 68:</b> Resultado obtenido del módulo.....	147
<b>Figura 69:</b> Resultado del algoritmo base 64.....	147
<b>Figura 70:</b> Resultado del algoritmo allegedRC4.....	147
<b>Figura 71:</b> Código de control.....	147
<b>Figura 72:</b> Diagrama de casos de uso por actor cajero.....	149
<b>Figura 73:</b> Diagrama de casos de uso por actor estudiante.....	150
<b>Figura 74:</b> Diagrama de casos de uso del sistema.....	151
<b>Figura 75:</b> Diagrama de colaboración dosificación.....	155
<b>Figura 76:</b> Diagrama de colaboración pago de pensiones.....	156
<b>Figura 77:</b> Diagrama de colaboración facturación.....	156
<b>Figura 78:</b> Diagrama de clases.....	157
<b>Figura 79:</b> Diagrama de base de datos.....	158
<b>Figura 80:</b> Arquitectura para el caso de uso gestionar dosificación.....	161
<b>Figura 81:</b> Arquitectura para el caso de uso pago de pensiones.....	162
<b>Figura 82:</b> Contenido de la carpeta views.....	163
<b>Figura 83:</b> Contenido de la carpeta controllers.....	163
<b>Figura 84:</b> Contenido de la carpeta models.....	164
<b>Figura 85:</b> Contenido de la carpeta librerías.....	164
<b>Figura 86:</b> Contenido de la carpeta librería PDF.....	164
<b>Figura 87:</b> Contenido de la carpeta librería código control .....	165
<b>Figura 88:</b> Contenido de la carpeta librería código QR.....	165
<b>Figura 89:</b> Interface gestionar dosificación.....	166

<b>Figura 90:</b> Interface registrar dosificación.....	166
<b>Figura 91:</b> Interface buscar dosificación. ....	167
<b>Figura 92:</b> Interface de caso de uso pago pensiones buscar estudiante.....	167
<b>Figura 93:</b> Interface de el caso de uso pagar pensiones. ....	168
<b>Figura 94:</b> Interface caso de uso pagar pensiones.....	168
<b>Figura 95:</b> Interface factura pdf.....	169
<b>Figura 96:</b> Registro de datos código de control .....	171
<b>Figura 97:</b> Prueba del código de control.....	171
<b>Figura 98:</b> Resultados de los 5 mil casos de prueba .....	172
<b>Figura 99:</b> Campana de Gauss para la aceptación de la hipótesis.....	188

# **1. GENERALIDADES.**

## **1.1. INTRODUCCIÓN.**

Las Tecnologías web han transformado a la sociedad y la manera como se maneja la información. Actualmente el uso extendido de aplicaciones web multiplataforma se puede encontrar en todas las áreas de interacción humana.

Las aplicaciones web multiplataforma son las que se ejecutan dentro de un navegador y por ende tienen menos requerimiento de software ya que se ejecutan en un servidor, también ofrecen compatibilidad con distintos dispositivos como Tablet, Smartphone, laptops, etc de manera que se adaptan al diseño de una pantalla y resolución más pequeñas ya que pueden ser implementados con CSS, HTML5 para su mejor adaptabilidad a la pantalla de los dispositivo móviles, también pueden desarrollarse con diferentes herramientas, tecnologías, lenguajes de programación (Java, Grails, Php, Ruby, Python, etc.) y varias arquitecturas de software y no así solo con el típico cliente – servidor.

Se puede afirmar que el uso de aplicaciones web multiplataforma en el ámbito educativo en los últimos años ha crecido debido a que estos permiten una interacción oportuna entre distintos actores participantes del proceso educativo. En ese sentido se puede mencionar la utilización de aplicaciones web multiplataforma tanto en los colegios y universidades.

Por otro lado la facturación computarizada se ha vuelto imprescindible para todas las instituciones que facturan ya que es más rápida, económica y segura a comparación de la facturación manual. Se dice que es más segura ya que usa código de control y código qr los cuales usan algoritmos de datos complejos, más económica porque desechan la necesidad de comprar talonarios de factura y más rápida ya que elimina el tener que transcribir a mano los datos del cliente y el detalle de la factura. Otras de

las ventajas más notorias son la reducción de los papeleos que hacen las instituciones respecto a las facturas, como también de los errores en las transcripciones ya que los datos del cliente se guardan en una base de datos.

El presente proyecto propone el desarrollo de una aplicación web multiplataforma con facturación computarizada para la gestión de pensiones de la Unidad educativa María Auxiliadora Cristo Rey donde la gran mayoría de los estudiantes y padres de familia cuentan con teléfonos inteligentes (Smartphone), laptops, computadoras con acceso a internet, y a través de ellos podrán realizar el control de pago de pensiones.

## **1.2. ANTECEDENTES.**

La Unidad Educativa María Auxiliadora Cristo Rey fundada el año 1988 ubicado en la Av. Hernando de Soto esq. Diego de Almagro, Cochabamba, Bolivia se dedica a la enseñanza y formación de estudiantes de pre escolar inicial, primaria y secundaria, cuenta con 59 estudiantes en el nivel inicial, 238 en primaria y 294 en secundaria lo que hace un total de 591 estudiantes.

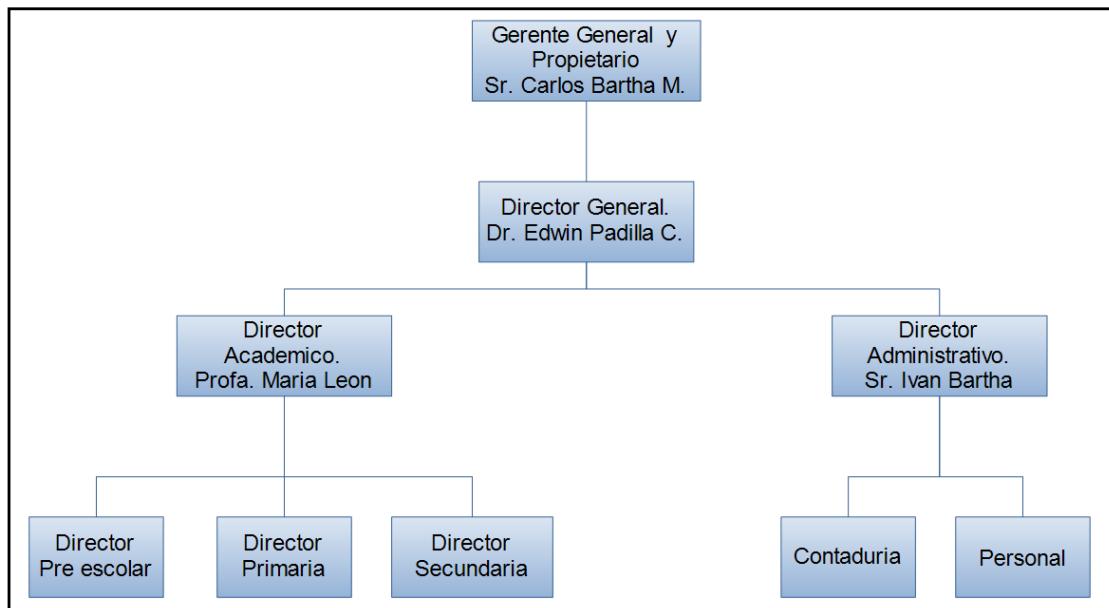
Tiene como Misión: formar estudiantes autónomos y respetuosos de las diferencias culturales, con sólidos conocimientos humanos y científicos que posibiliten continuar con éxitos estudios universitarios o incorporarse al mundo del trabajo, sustentados en una actitud profundamente democrática y sólida.

La Unidad Educativa María Auxiliadora Cristo Rey se divide en dos áreas principales administrativa y académica:

- El área administrativa tiene la responsabilidad de velar por los asuntos económicos, financieros, relaciones laborales y servicios generales, para realizar estas tareas esta área tiene dos sub áreas: contabilidad y personal. (Figura N° 1).

- El área académica tiene la responsabilidad de planificar, organizar, coordinar, orientar, supervisar el proceso enseñanza – aprendizaje en los tres niveles pre escolar, primaria y secundaria, para esto tiene tres sub directores. (Figura N° 1).

**Figura 1:** Organigrama



**Fuente:** Elaboración Propia.

La gestión de pensiones de los estudiantes pertenece al área administrativa, a la parte contable específicamente. El cobro y la facturación está a cargo de una secretaría contable y se hace de la siguiente manera: se emplea un sistema de escritorio desarrollado en Visual Basic 2005 con bases de datos sql server 2000 donde se registra cada uno de los estudiantes (**Ver Anexo A**), y cada uno de estos tiene las respectivas mensualidades a pagar (**Ver Anexo B**), Cuando entra un padre de familia a la institución para hacer el pago respectivo de las pensiones, debe pedir información a la secretaría o encargada de pensiones de cuantas pensiones debe su hijo/a.

El padre tiene que brindar la información del curso y nombre de su hijo/a, a la secretaría para buscarlo en el sistema de escritorio y responder sobre el estado de las pensiones de ese alumno, si el alumno está en pre escolar cancelara un monto de Bs. 260, si

está en primaria deberá cancelar un monto igual a Bs. 335 y si esta secundaria un monto de Bs. 367, también existe descuentos por tercer hermano del 30 % y por cuarto hermano del 100%. Una vez ya cancelada la pensión se registra el mes o meses de los que se está pagando en el sistema (**Ver Anexos C**) y luego se factura a mano dicha pensión (**Ver Anexos D**).

El servicio de información y facturación brindado a los estudiantes y padres de familia para acceder y pagar las pensiones no cuenta con facilidad y comodidad ya que los padres de familia o interesados requieren forzosamente apersonarse a la secretaría y hacer largas colas, ya sea para hacer el pago de la pensión o simplemente informar de estado de las pensiones.

Los ingresos de la unidad educativa se registran en el sistema de escritorio al momento de facturar la pensión (**Ver Anexos C**), también el sistema de escritorio genera reportes diarios en un pdf llamado hoja de colecta diaria, también permite reportes por fecha. (**Ver Anexos E**).

Para realizar el llenado de las facturas de manera manual se demora un tiempo de 4 a 7 minutos aproximadamente por factura, existe también la necesidad de solicitar talonarios de facturas a una imprenta lo que genera gastos a la Unidad Educativa ya que el encargado de solicitar talonarios tiene que solicitarlos y luego ir a recogerlos, así mismo estos requieren una suministración continua para que no exista escases de talonarios, ya que en ocasiones las papeletas de facturas se terminan lo que hace imposible seguir facturando lo cual requiere que el padre de familia retorne a la institución en otra ocasión. También se observó que la encargada produce errores en la transcripción ya hubo casos en los cuales la facturas quedaron invalidadas ya sea por errores en el NIT, nombre o monto.

Se ha observado que existen largas colas al momento de pagar las pensiones y esto se debe principalmente a que algunos de los que hacen cola solo están interesados en averiguar el estado de sus pensiones y no cancelar la pensión.

Por otro lado cabe recalcar que el sistema de escritorio genera la necesidad de transcripción y/o copiado de datos de los estudiantes al sistema cada vez que inicia una nueva gestión tomando mucho tiempo el copiado de los estudiantes de anterior gestión a la nueva gestión.

También se ha percibido que la encargada de entrega de boletas de calificaciones no cuenta con la accesibilidad inmediata de dar a conocer la situaciones de pensiones de los estudiantes, para dar cumplimiento a ciertas normas y reglas internas, ya que los estudiantes con deudas económicas no tienen derecho a conocer sus calificaciones.

### **1.3. PLANTEAMIENTO DEL PROBLEMA.**

A continuación se mencionarán los siguientes puntos para poder determinar el problema.

#### **1.3.1. Identificación del problema.**

El proceso actual existente en la Unidad Educativa María auxiliadora Cristo Rey para la gestión de pensiones tiene deficiencias en cuanto al registro, pagos realizados por los interesados ya que la información se guarda en un sistema de escritorio y la facturación se hace de forma manual.

##### **1.3.1.1. Identificación de la situación problemática.**

- El proceso de llenado manual de las facturas provoca largas filas y descontento en los padres de familia o tutores.
- La compra de talonarios de facturas que involucra ir a la imprenta para realizar el pedido y luego recogerlos provoca gastos de dinero.
- La terminación de talonarios de facturas en el pago de pensiones provoca retrasos en el pago de pensiones y molestias para los padres de familia ya que deben retornar otro día para realizar el pago respectivo.

- Los errores en el llenado de facturas provocan pérdida de tiempo y generación de varias facturas anuladas.
- La poca flexibilidad en la consulta del pago de pensiones provoca molestias y descontento entre los padres de familia o tutores.
- La encargada de entrega de calificaciones no cuenta con acceso inmediato a las pensiones lo cual provoca retraso en la entrega de los boletines a los estudiantes o padres de familia.
- La necesidad de presencia física de los interesados para conocer la situación de pago de pensiones de los estudiantes provoca molestias y largas filas.

#### **1.3.1.2. Identificación de las causas.**

- El proceso de llenado manual de las facturas
- La compra de talonarios de facturas que involucra ir a la imprenta para realizar el pedido y luego recogerlos.
- La terminación de talonarios de facturas en el pago de pensiones.
- Los errores en el llenado de facturas.
- La poca flexibilidad en la consulta del pago de pensiones.
- La encargada de entrega de calificaciones no cuenta con acceso inmediato a las pensiones.
- La necesidad de presencia física de los interesados para conocer la situación de pago de pensiones de los estudiantes.

#### **1.3.2. Formulación del problema.**

Las limitaciones técnicas del sistema de escritorio y el procedimiento manual del proceso de gestión de pensiones provocan tiempo excesivo en la atención de los

interesados, gastos económicos para la institución, errores en el llenado de facturas, largas filas y descontentos.

### **1.3.3. Análisis causa efecto.**

**CAUSA:** Las limitaciones técnicas del sistema de escritorio y el procedimiento manual de facturación en la “Gestión de pensiones”

**EFFECTO:** Tiempo excesivo en la atención de los interesados, gastos económicos para la institución, errores en el llenado de facturas, largas filas y descontentos.

## **1.4. OBJETIVOS Y ACCIONES.**

A continuación se realizará el objetivo general y los objetivos específicos del proyecto.

### **1.4.1. Objetivo general.**

Desarrollar sistema de gestión de pensiones con facturación computarizada, para lograr un proceso más eficiente y una forma más satisfactoria de conocer el estado de pensiones de estudiantes.

### **1.4.2. Objetivos específicos y acciones.**

- Diseñar modelado de negocio alternativo del proyecto de gestión de pensiones.
- Desarrollar subsistema multiplataforma de gestión de usuarios, niveles educativos con montos de mensualidades por gestión y planificación de cursos.
- Diseñar e implementar módulo de inscripciones de estudiantes integrado al subsistema multiplataforma
- Desarrollar módulos de cobro de pensiones y reportes integrado al subsistema multiplataforma.
- Realizar las pruebas con usuarios finales al sistema concluido.

A continuación se muestra la tabla objetivos específicos y acciones en la cual se muestra a detalle las acciones a tomar para completar cada uno de los objetivos específicos ya planteados.

**Tabla 1:** Objetivos específicos y acciones.

OBJETIVOS ESPECÍFICOS	ACCIONES
Diseñar modelado de negocio alternativo del proyecto de gestión de pensiones.	<ul style="list-style-type: none"> <li>• Establecer el modelado de negocio actual del proceso de gestión de pensiones.</li> <li>• Evaluar las deficiencias existentes en el proceso actual.</li> <li>• Elaborar una propuesta de modelado de negocio alternativo basado en el sistema propuesto y solucionando los problemas del proceso actual.</li> <li>• Analizar propuestas de modelado de negocio alternativo con los involucrados.</li> <li>• Elaborar el modelado de negocio alternativo final.</li> </ul>
Desarrollar subsistema multiplataforma de gestión de usuarios, niveles educativos con montos de mensualidades por gestión y planificación de cursos.	<ul style="list-style-type: none"> <li>• Seleccionar metodología de desarrollo que se aadecue al sistema propuesto.</li> </ul>

OBJETIVOS ESPECÍFICOS	ACCIONES
<p>.Desarrollar subsistema multiplataforma de gestión de usuarios, niveles educativos con montos de mensualidades por gestión y planificación de cursos. (continuación)</p>	<ul style="list-style-type: none"> <li>• Planificar Incrementos o iteraciones del proyecto según a la metodología seleccionada.</li> <li>• Seleccionar lenguajes de programación para lograr la implementación de la aplicación web multiplataforma.</li> <li>• Seleccionar el framework adecuado para el desarrollo de la aplicación web multiplataforma.</li> <li>• Seleccionar gestor de base de datos que se aadecue a la aplicación web multiplataforma.</li> <li>• Realizar actividades de análisis para el sub sistema web multiplataforma.</li> <li>• Elaborar diagramas correspondientes para el subsistema multiplataforma.</li> <li>• Diseñar la base de datos del sub sistema multiplataforma.</li> <li>• Implementar los módulos de gestión de usuarios, niveles educativos con montos de mensualidades por gestión y planificación y cursos.</li> <li>• Realizar pruebas funcionales al módulo desarrollado.</li> </ul>

OBJETIVOS ESPECÍFICOS	ACCIONES
Diseñar e implementar módulo de inscripciones de estudiantes integrado al subsistema multiplataforma.	<ul style="list-style-type: none"> <li>• Realizar actividades de análisis para el modulo.</li> <li>• Elaborar diagramas correspondientes para el módulo de inscripciones.</li> <li>• Diseñar las tablas de la base de datos del módulo inscripciones e integrarla a la base de datos del sub sistema web multiplataforma.</li> <li>• Implementar el módulo de inscripciones integrado al subsistema multiplataforma.</li> <li>• Realizar pruebas funcionales al sistema.</li> </ul>
Desarrollar módulos de cobro de pensiones con facturación y reportes integrado al subsistema multiplataforma.	<ul style="list-style-type: none"> <li>• Analizar normativas y procedimiento a seguir para facturación computarizada del servicio de impuestos nacionales Bolivia.</li> <li>• Realizar actividad de análisis para el módulo de cobro de pensiones y reportes.</li> <li>• Analizar normativas y procedimiento a seguir para facturación computarizada de impuesto nacionales Bolivia.</li> </ul>

OBJETIVOS ESPECÍFICOS	ACCIONES
<p>Desarrollar módulos de cobro de pensiones con facturación y reportes integrado al subsistema multiplataforma.(Continuación)</p>	<ul style="list-style-type: none"> <li>• Realizar actividad de análisis para el módulo de cobro de pensiones y reportes.</li> <li>• Analizar el pseudocódigo de los algoritmos establecidos por el SIN. Para la generación de código de control y código QR.</li> <li>• Elaborar diagramas correspondientes para el módulo de cobro de pensiones y reportes.</li> <li>• Diseñar las tablas de base de datos del módulo de cobro de pensiones y reportes e integrarlo a la base de datos del sistema.</li> <li>• Implementar el módulo de cobro de pensiones y reportes e integrarlo al sub sistema multiplataforma y reportes.</li> <li>• Realizar pruebas funcionales con los datos brindados por el SIN en formato Excel (casos de prueba para la generación de códigos Ver. 7.0)</li> </ul>

OBJETIVOS ESPECÍFICOS	ACCIONES
Efectuar las pruebas finales al sistema multiplataforma concluido.	<ul style="list-style-type: none"> <li>• Realizar pruebas con distintos dispositivos (laptops, tablets, smartphones).</li> <li>• Realizar pruebas con distintos sistemas operativos.</li> <li>• Realizar pruebas con distintos navegadores.</li> </ul>

**Fuente:** Elaboración propia.

## 1.5. JUSTIFICACIÓN.

### 1.5.1. Justificación técnica.

El presente proyecto a través de la aplicación web multiplataforma con facturación computarizada permitirá tener menos requerimientos de software ya que se ejecutara dentro de un navegador en un servidor web en la nube ya sea de paga o gratuito, también ofrece adaptabilidad a diferentes resoluciones y pantallas más pequeñas, por lo que los padres de familia, estudiantes o interesados podrán acceder a consultar y visualizar el estado de pensiones de los cursos realizados desde sus celulares, tablets, laptops o desde cualquier dispositivo que soporte un navegador web.

### 1.5.2. Justificación económica.

Con el sistema web multiplataforma con facturación computarizada se logrará la reducción de costos ya que la facturación computarizada a diferencia de la facturación manual no exige gastos extras como en material de escritorio por ejemplo lapiceros, correctores y por otro lado también cabe decir que no se generan

gastos en talonarios de facturas, así como también el costo invertido en el transporte al realizar la compra de estos talonarios. También las licencias que se usaran en el desarrollo del presente proyecto serán gratuitas, por otro lado cabe indicar que los dispositivos móviles como Smartphone no representaran un costo adicional ya que todas las personas tienen por lo menos un celular o un dispositivo móvil que soporte un navegador web.

#### **1.5.3. Justificación social.**

Con la utilización del sistema web multiplataforma con facturación computarizada, el personal de trabajo usará menos tiempo en la realización de las actividades de gestión de pensiones, además proporcionará a los estudiantes y padres de familia una cómoda forma de pedir su información, ya que ellos podrán conocer el estado de pago de pensiones al instante desde cualquier lugar en el que se encuentren evitando las largas filas y brindando comodidad.

#### **1.5.4. Justificación operativa.**

El uso de la aplicación web multiplataforma con facturación computarizada no presentará ningún reto para el personal, los estudiantes y padres de familia, debido a que se utilizará a través de un navegador web y no será necesario instalarlo, su interfaz será comprensible y amigable ya que se acomodara a resoluciones de smartphones, tablets y laptops, de manera que sea fácil de usar para cualquier usuario que tiene una mínima experiencia de trabajo con interfaces que se despliegan a través de internet.

### **1.6. ALCANCE**

A continuación se describen los alcances del proyecto:

### **1.6.1. Alcance temático.**

El sistema abarcará las áreas de: Ingeniería de Software, Paradigma de Análisis y Diseño de Sistemas, Sistemas Operativos, Gestor de Bases de Datos, Tecnologías web multiplataforma.

### **1.6.2. Alcance institucional.**

El sistema de gestión de pensiones con facturación computarizada tiene como objetivo la gestión de pensiones en el área administrativa de la Unidad Educativa María Auxiliadora Cristo Rey.

### **1.6.3. Alcance temporal.**

El sistema para la gestión de pensiones con facturación computarizada de la Unidad Educativa María Auxiliadora Cristo Rey se terminara en la gestión 2015, y tendrá una durabilidad de cinco años ya que las crecientes tecnologías informáticas lo irán dejando obsoleto.

## **1.7. HIPÓTESIS.**

El sistema de gestión de pensiones con facturación computarizada en el área administrativa de la Unidad Educativa María Auxiliadora Cristo Rey permitirá reducir el tiempo en la atención de los interesados, gastos económicos para la institución y errores en el llenado de facturas.

### **1.7.1. Análisis de variables.**

#### **Variable Independiente**

- El sistema de gestión de pensiones con facturación computarizada en el área administrativa.

## **Variables Dependientes**

- Tiempo en la atención de los interesados.
- Gastos económicos para la institución.
- Errores en el llenado de facturas.

### **1.7.2. Definición conceptual.**

#### **Variable independiente**

- El sistema de gestión de pensiones y facturación computarizada en el área administrativa permitirá brindar información sobre el pago de pensiones a los estudiantes y padres de familia desde un dispositivo móvil o laptop y facturación computarizada.

#### **Variable independiente**

- Tiempo en la atención de los interesados, involucra la optimización de tiempo que se requiere al momento de atender y emitir las facturas a los padres de familia que realizan el pago de pensiones, así como reducir las colas que se forman durante esas fechas críticas, también en la obtención de información del estado de pensiones de los estudiantes, permite reducir el tiempo empleado en consultar el estado de pago de pensiones de cada estudiante y brindar la posibilidad de que padres de familia o estudiantes puedan obtener dicha información de manera flexible y rápida.
- Gastos económicos para la institución, permite la reducción de gastos económicos en compra de talonarios y material de escritorio.
- Errores en el llenado de facturas, permite la reducción en la cantidad de facturas anuladas.

### 1.7.3. Operativización de variables.

**Tabla 2:** Operativización de variables.

Variable	Dimensión	Indicador
<b><u>Variable Independiente</u></b>  El sistema web multiplataforma de gestión de pensiones y facturación computarizada en el área administrativa.	El desarrollo de la aplicación web multiplataforma.	Comparación de Beneficios con y sin sistema.
<b><u>Variables Dependientes</u></b>  Tiempo en la atención de los interesados.	Tiempo que se requiere para emitir factura y realizar consulta del estado de cuentas.	Tiempo en minutos
Gastos económicos para la institución.	Gastos económicos en la compra de talonarios y material de escritorio.	Cantidad de dinero
Errores en el llenado de facturas.	La cantidad de facturas emitidas con error y anuladas.	Cantidad de facturas con errores.

**Fuente:** Elaboración Propia.

## 1.8. MATRIZ DE CONSISTENCIA.

**Tabla 3:** Matriz de consistencia.

PROBLEMA	OBJETIVO	HIPÓTESIS
<b>SISTEMA DE GESTIÓN DE PENSIONES CON FACTURACIÓN COMPUTARIZADA. CASO DE ESTUDIO: UNIDAD EDUCATIVA MARÍA AUXILIADORA CRISTO REY</b>		
Las limitaciones técnicas del sistema de escritorio y el procedimiento manual del proceso de gestión de pensiones	Desarrollar un sistema de gestión de pensiones con facturación computarizada.	El sistema de gestión de pensiones con facturación computarizada en el área administrativa de la Unidad Educativa María Auxiliadora Cristo Rey.
<p>↓ PROVOCA</p> <p>Tiempo excesivo en la atención de los interesados, gastos económicos para la institución, errores en el llenado de facturas, largas filas y descontento de los padres de familia.</p> <p>-</p>	<p>↓ PARA</p> <p>Lograr un proceso más eficiente y una forma más satisfactoria de conocer el estado de pensiones de estudiantes</p>	<p>↓ PERMITIRÁ</p> <p>Reducir el tiempo en la atención de los interesados, gastos económicos para la institución y errores en el llenado de facturas.</p>

**Fuente:** Elaboración propia.

## **2. MARCO TEÓRICO.**

### **2.1. TÉCNICAS DE RECOLECCIÓN DE INFORMACIÓN.**

Las técnicas de recolección de información son los procesos de obtención de datos empíricos a través de diversas herramientas en diferentes áreas del conocimiento que permiten la medición de variables para el estudio de un determinado problema.

#### **2.1.1. Observación.**

La observación es la técnica de recolección de datos que permite acumular y sistematizar información sobre un hecho o fenómeno que tiene relación con un determinado problema. En la aplicación de esta técnica el investigador: Identifica y demita el problema, toma contacto directo a través de la observación con el hecho o fenómeno, registra o toma nota de lo observado conjuntamente al desarrollo de los acontecimientos sin embargo el registro debe hacerse en circunstancias que eviten poner en peligro el desarrollo normal del fenómeno de manera que los individuos observados no cambien su comportamiento habitual. (Universidad de Fribourg, 2014).

#### **Ventajas de la observación:**

- Se pueden describir procesos naturales y sociales con ella.
- Se acerca a la realidad de lo que realmente acontece.
- El investigador debe de ser un punto objetivo de referencia.

#### **Desventajas de la observación**

- Se torna sólo desde la perspectiva del investigador.
- Al observarse desde fuera se puede perder un poco de lo que los actores consideran importante de la práctica social.
- Al observarse desde fuera se puede correr el riesgo de emitir juicios de lo que observa.

### **2.1.2. Entrevistas.**

Consiste en una conversación preparada como una dinámica de preguntas y respuestas abiertas en las cuales se socializan sobre una temática determinada relacionada con la problemática a estudiar, esta técnica permite conocer el punto de vista de diferentes partes involucradas en la discusión y está apoyada por una secuencia de preguntas que se desea conocer. (Universidad de Fribourg, 2014).

#### **Ventajas de la entrevista**

- Es más detallado con la adaptación de las preguntas según las características del entrevistado.
- La posibilidad de perder información en la entrevista por lo general es menor, con relación a la observación por su misma naturaleza.
- De igual manera en la entrevista generalmente es menor la posibilidad de perder información en comparación al cuestionario.
- Permite obtener mucha mayor información que el cuestionario.
- Se adapta con mucha más facilidad que el cuestionario a cualquier nivel cultural del informante.

#### **Desventajas de la entrevista**

- Es más costosa que el cuestionario y observación, sobre todo para muestras grandes, y con mayor razón si los individuos están dispersos geográficamente, por cuanto exige la presencia de entrevistadores.
- Se necesita de entrevistadores altamente especializados en el tema de investigación, es decir, personas muy bien entrenadas en el tema de la entrevista que le permita profundizar en la búsqueda del dato a partir de las respuestas dadas por el informante.
- La entrevista generalmente requiere de mayor tiempo que el cuestionario.

- La abundante información recolectada dificulta su registro y puede ser fuente de error en el análisis.
- El entrevistador, por la flexibilidad de la técnica, puede influenciar en las respuestas del informante.
- La abundante información que se obtiene mediante la entrevista hace más costosa su sistematización y procesamiento estadístico.
- En la entrevista hay el riesgo de interpretar mal las respuestas. (Universidad de Fribourg, 2014).

### **2.1.3. Cuestionarios.**

Consiste en una serie de preguntas con opciones múltiples, referentes a una temática determinada que permite conocer el punto de vista de las personas hacia el problema que trata y a su vez permite recopilar información sobre el grado de conocimiento de los temas tratados en el presente proyecto. (Universidad de Fribourg, 2014).

#### **Ventajas del cuestionario**

- Es menos costoso que la entrevista, por cuanto en muchos casos no es imprescindible la presencia de unas personas en la aplicación del cuestionario (puede ser por correo electrónico).
- La aplicación del cuestionario no necesita de un personal especializado en el tema de la investigación, como si es imprescindible en el caso de la entrevista.
- Es más uniforme en los datos que se recolecta, pues las preguntas son las mismas para todos los informantes.
- El cuestionario, a diferencia de la entrevista, es más funcional en su aplicación a muestras grandes, incluso por más dispersos que los informantes estén geográficamente; pues, como se ha señalado anteriormente, el cuestionario puede ser enviado por correo.

- Si el cuestionario es enviado por correo, el informante puede sentirse más seguro del anonimato de sus respuestas y dar una mayor información confiable.
- Es menos costosa la sistematización y procesamiento estadístico de la información, que en el caso de la entrevista.

### **Desventajas del cuestionario**

- Es demasiado rígido y en consecuencia permite la recolección únicamente del dato al que se refiere la pregunta. Esto puede dar lugar a una pérdida de información importante para el análisis del problema motivo de investigación.
- Es demasiado formal y puede ocasionar resistencia en el informante a contestar determinadas preguntas.
- En la medida que las preguntas deben ser hechas a todos los informantes, tal como están escritas,
- hay mayor posibilidad, que en la entrevista, de obtener demasiadas “no respuestas” o respuestas erróneas; sobre todo cuando el informante no comprende el correcto sentido de la pregunta. (Universidad de Fribourg, 2014).

## **2.2. APPLICACIONES WEB MULTIPLATAFORMA.**

Una aplicación web multiplataforma es aquella que se ejecuta y se adapta a las distintas resoluciones y tamaños para su mejor visualización en Smartphones, Tablets, laptops y pc de escritorio, también se ejecuta en diferentes sistemas operativos, esta es accedida por medio de un navegador web vía internet o una intranet. (developer , 2014).

### **2.2.1. Características de las aplicaciones web multiplataforma.**

- El usuario puede acceder fácilmente a estas aplicaciones empleando un navegador web dese un Smartphone, Tablet o laptop.

- Si es por internet, el usuario puede entrar desde cualquier lugar del mundo donde tengas internet, si es por un área local se accederá por la red.
- Pueden existir miles de usuarios en una sola aplicación instalada en el servidor por lo tanto puede actualizar y mantener y todos los usuarios verán resultado inmediatamente.
- Proveen gran compatibilidad entre distintas plataformas ya que opera en un navegador web.
- Puede ser desarrollado por una sola pila de tecnologías (HTML5, CSS, Java Script) y entregar el desarrollo a todas las plataformas.

Para efectuar un resumen del concepto podemos definirlo como la cualidad de la aplicación web para adaptarse a las diferentes resoluciones de pantalla, para que sea correctamente visible en dispositivos móviles, laptops, pc de escritorio, etc también cabe recalcar que puede ser desarrollado con una sola pila de tecnologías de desarrollo como ser HTML5, CSS y Java Script, combinado con algún lenguaje de programación. (developer , 2014)

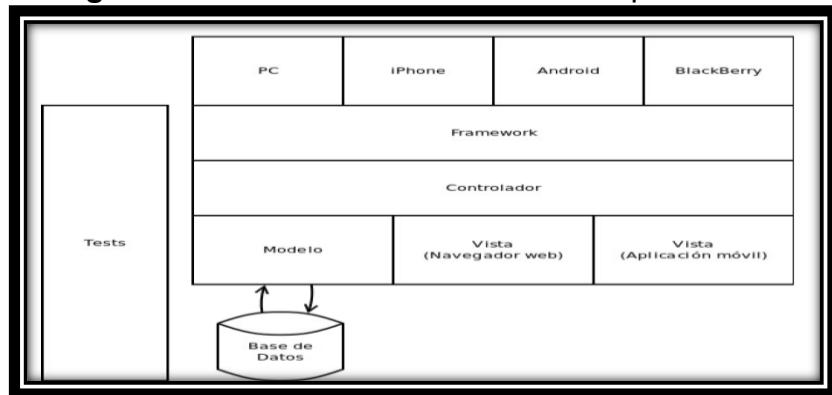
**Figura 2:** Adaptabilidad a diferentes resoluciones



**Fuente:** (Universidad Carlos III de Madrid, 2014).

Los sistemas multiplataforma en general están basados en el patrón Modelo – Vista – Controlador pero también se puede aplicar con otras arquitecturas, teniendo una estructura como la siguiente figura representa:

**Figura 3:** Estructura de un sistema multiplataforma.



**Fuente:** (Universidad Carlos III de Madrid, 2014)

Los dispositivos controlados por los usuarios son los que realizan una petición, dependiendo de qué plataforma realice la petición, tendrá una respuesta adaptada a su dispositivo de forma transparente, de tal forma que el usuario no tenga que informar cómo accede al sistema. (Universidad Carlos III de Madrid, 2014).

### **2.2.2. Arquitecturas de aplicaciones web multiplataforma.**

#### **2.2.2.1. Modelo Cliente – Servidor.**

La arquitectura cliente-servidor es un modelo de aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. Esta idea también se puede aplicar a programas que se ejecutan sobre una sola computadora, aunque es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras. (Universidad de cauca, 2014).

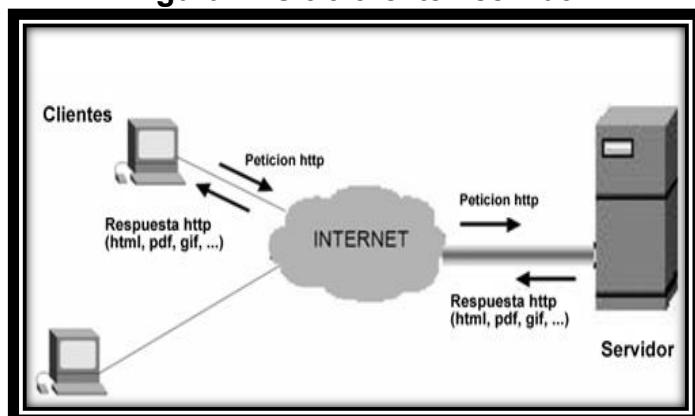
En esta arquitectura la capacidad de proceso está repartida entre cliente y servidor es una separación de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un sólo programa. Los tipos específicos de servidores incluyen los servidores web, los servidores de archivo, los servidores del

correo, etc. Mientras que sus propósitos varían de unos servicios a otros, la arquitectura básica seguirá siendo la misma. (Universidad de cauca, 2014).

Una disposición muy común son los sistemas multicapa en los que el servidor se descompone en diferentes programas que pueden ser ejecutados por diferentes computadoras aumentando así el grado de distribución del sistema.

La red cliente-servidor es una red de comunicaciones en la cual los clientes están conectados a un servidor, en el que se centralizan los diversos recursos y aplicaciones con que se cuenta; y que los pone a disposición de los clientes cada vez que estos son solicitados. Esto significa que todas las gestiones que se realizan se concentran en el servidor, de manera que en él se disponen los requerimientos provenientes de los clientes que tienen prioridad, los archivos que son de uso público y los que son de uso restringido, los archivos que son de sólo lectura y los que, por el contrario, pueden ser modificados, etc. Este tipo de red puede utilizarse conjuntamente en caso de que se esté utilizando en una red mixta. (Universidad de cauca, 2014).

**Figura 4:** Ciclo cliente - servidor



Fuente : (Universidad de cauca, 2014).

### Ventajas Cliente-Servidor

- Centralización del control: los accesos, recursos y la integridad de los datos son controlados por el servidor de forma que un programa cliente defectuoso o no

autorizado no pueda dañar el sistema. Esta centralización también facilita la tarea de poner al día datos u otros recursos.

- Escalabilidad: se puede aumentar la capacidad de clientes y servidores por separado. Cualquier elemento puede ser aumentado (o mejorado) en cualquier momento, o se pueden añadir nuevos nodos a la red (clientes y/o servidores).
- Fácil mantenimiento: al estar distribuidas las funciones y responsabilidades entre varios ordenadores independientes, es posible reemplazar, reparar, actualizar, o incluso trasladar un servidor, mientras que sus clientes no se verán afectados por ese cambio (o se afectarán mínimamente). Esta independencia de los cambios también se conoce como encapsulación.

Existen tecnologías, suficientemente desarrolladas, diseñadas para el paradigma de C/S que aseguran la seguridad en las transacciones, la amigabilidad de la interfaz, y la facilidad de empleo. (Universidad de cauca, 2014).

### **Desventajas Cliente-Servidor**

- La congestión del tráfico ha sido siempre un problema en el paradigma de C/S. Cuando una gran cantidad de clientes envían peticiones simultáneas al mismo servidor, puede ser que cause muchos problemas para éste (a mayor número de clientes, más problemas para el servidor). Al contrario, en las redes P2P como cada nodo en la red hace también de servidor, cuantos más nodos hay, mejor es el ancho de banda que se tiene.
- El paradigma de C/S clásico no tiene la robustez de una red P2P. Cuando un servidor está caído, las peticiones de los clientes no pueden ser satisfechas. En la mayor parte de redes P2P, los recursos están generalmente distribuidos en varios nodos de la red. Aunque algunos salgan o abandonen la descarga; otros pueden todavía acabar de descargar consiguiendo datos del resto de los nodos en la red.

- El software y el hardware de un servidor son generalmente muy determinantes. Un hardware regular de un ordenador personal puede no poder servir a cierta cantidad de clientes. Normalmente se necesita software y hardware específico, sobre todo en el lado del servidor, para satisfacer el trabajo. Por supuesto, esto aumentará el coste.
- El cliente no dispone de los recursos que puedan existir en el servidor. Por ejemplo, si la aplicación es una Web, no podemos escribir en el disco duro del cliente o imprimir directamente sobre las impresoras sin sacar antes la ventana previa de impresión de los navegadores. (Universidad de cauca, 2014).

#### **2.2.2.2. Patrón Modelo-Vista-Controlador.**

El modelo–vista–controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario. Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento. (Universidad Complutense, 2014).

#### **Flujo de control**

- El usuario realiza una acción en la interfaz.
- El controlador trata el evento de entrada.
- El controlador notifica al modelo la acción del usuario, lo que puede implicar un cambio del estado del modelo (si no es una mera consulta).
- Se genera una nueva vista.

- La interfaz de usuario espera otra interacción del usuario, que comenzará otro nuevo ciclo. (Universidad Complutense, 2014).

## MVC en aplicaciones web

Aunque originalmente MVC fue desarrollado para aplicaciones de escritorio, ha sido ampliamente adaptado como arquitectura para diseñar e implementar aplicaciones web en los principales lenguajes de programación. Se han desarrollado multitud de frameworks, comerciales y no comerciales, que implementan este patrón.

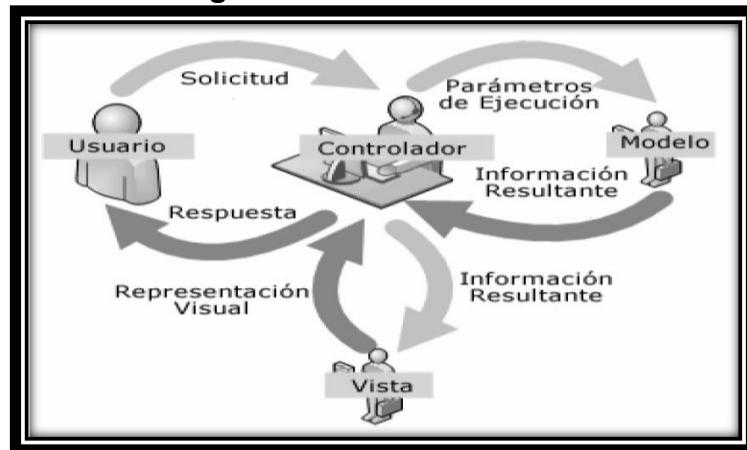
Los frameworks MVC para desarrollo web plantean un enfoque de cliente ligero en el que casi todas las funciones, tanto de la vista, el modelo y el controlador recaen en el servidor. Este enfoque es de la siguiente manera:

**Vista:** la página HTML

**Controlador:** código que obtiene datos dinámicamente y genera el contenido HTML

**Modelo:** la información almacenada en una base de datos o en XML junto con las reglas de negocio que transforman esa información (teniendo en cuenta las acciones de los usuarios). (Universidad Complutense, 2014).

**Figura 5:** Ciclo de vida MVC



**Fuente:** (Universidad Complutense, 2014).

## **Ventajas de MVC**

- La separación del modelo de la vista, es decir separar la lógica de las vistas mostradas al usuario.
- Es mucho más sencillo Agregar múltiples representaciones de los datos e información. (Universidad de las Américas Puebla, 2014).
- Agregar nuevos tipos de datos según sea requerido por la aplicación ya que son independientes del funcionamiento de otras capas.
- Crea independencia de funcionamiento.
- Ofrece una manera más sencilla para probar el correcto funcionamiento de sistema.

## **Desventajas de MVC**

- La separación de Conceptos en capas agrega complejidad al sistema.
- La cantidad de archivos a mantener y desarrollar se incrementa considerablemente.
- La curva de aprendizaje del diseño es más alta que usando otros modelos más sencillos. (Universidad de las Américas Puebla, 2014).

## **2.3. FACTURACIÓN COMPUTARIZADA.**

### **2.3.1. La factura.**

La factura es un documento de suma importancia y debe ser emitida con la mayor precisión, claridad y exactitud. Cualquier error en la confección de la factura supone un problema de cobro o de descargo de esta. La factura debe incluir todos los datos que obligan la legislación, servicio de impuestos nacionales y asimismo los que solicita el cliente. La factura tiene tributarios al tener un IVA (Impuesto al valor agregado).

### **2.3.2. Tipos de factura.**

Existen varios tipos de facturación como ser la facturación manual. Facturación pre valorado, facturación computarizada, facturación oficina virtual, facturación electrónica web, facturación electrónica por ciclos y cada una de estas tiene distintas características técnicas como también en el formato de estas y la realización de estas cabe decir que el tipo de factura que este proyecto realizará es la “facturación computarizada”, a continuación se realizará una explicación detallada de todos los aspectos de esta factura computarizada.

### **2.3.3. Modalidad facturación computarizada.**

Es la modalidad en la que la emisión de Factura o Nota Fiscal se realiza a partir de un sistema computarizado desarrollado o adquirido por el propio Sujeto Pasivo o Tercero Responsable, mismo que debe cumplir con los aspectos técnicos para la generación del Código de Control y la impresión del Código QR. (**Ver Anexo F**). En los siguientes puntos se detallarán con exactitud estos aspectos técnicos de la factura para su mejor entendimiento.

Para la generación del código de control se utiliza algoritmos de alta complejidad descritas a detalle en la parte de anexos (**Ver Anexo G**), también existen 5000 mil casos de pruebas para probar si los algoritmos ya implementados están funcionando correctamente, estos 5 mil casos de prueba son proporcionados por el Servicio de Impuestos Nacionales en la página de esta misma (**Ver Anexo H**), por otro lado para la generación del código QR se utilizan librerías que generen este, y este se puede probar con software de lectura de código QR para dispositivos móviles u otros, también es necesario recalcar que para la generación del código QR se utilizan los datos de la siguiente tabla, marcados como “si” si son obligatorios y como “cuando corresponda”, cuando el dato entra a la generación en casos especiales : (Servicio de Impuestos Nacionales, 2014).

**Tabla 4:** Contenido del código QR para facturación computarizada.

NOMBRE DEL CAMPO	TIPO DE DATO	DESCRIPCIÓN	OBLIGATORIEDAD
NIT Emisor (Número de Identificación Tributaria)	Numérico	NIT del emisor.	SI
Nombre o Razón Social	Alfanumérico	Nombre o Razón Social del emisor.	SI
Número de Factura	Numérico	Número correlativo de Factura o Nota Fiscal.	SI
Número de Autorización	Numérico	Número otorgado por la Administración Tributaria para identificar la dosificación.	SI
Fecha de emisión	Fecha	Con formato: DD/MM/AAAA.	SI
Importe de la compra	Numérico	Monto total consignado en la Factura o Nota Fiscal (con dos decimales para centavos).	SI
Código de Control	Alfanumérico	Código que identifica la transacción comercial realizada con la Factura o Nota Fiscal).	SI
Fecha Límite de Emisión	Fecha	Con formato: DD/MM/AAAA.	SI
Importe ICE (Importe de Impuestos al Consumo Específico)	Numérico	Monto ICE, en el caso de no corresponder consignar el carácter cero (0).	CUANDO CORRESPONDA
Importe por ventas no Gravadas o Gravadas a Tasa Cero	Numérico	Cuando corresponda, caso contrario se consignará el carácter cero (0). (Con dos decimales para centavos).	CUANDO CORRESPONDA
NIT / NDI Comprador (Número de Identificación Tributaria o Documento de Identidad)	Alfanumérico	NIT del comprador, en caso de no contar se consignará el número de Cédula de Identidad o Carnet de Extranjería o el carácter cero (0).	SI
Nombre o Razón Social del comprador	Alfanumérico	Nombre o Razón Social del emisor.	SI

**Fuente:** (Servicio de Impuestos Nacionales, 2014)

#### **2.3.4. Fecha límite de emisión.**

Las facturas deben ser emitidas hasta la fecha límite de emisión por el sujeto según la modalidad de facturación conforme a la siguiente tabla:

**Tabla 5:** Fecha límite de emisión de facturas según la modalidad

Modalidad de Facturación	Vigencia	Excepciones
Manual	6 meses	Alquileres: 24 meses (2 años)
Prevalorada	12 meses	Telecomunicaciones: 24 meses (2 años) Espectáculos públicos: Mismo día del evento
Computarizada	6 meses	---
Oficina Virtual	Mismo día de emisión	---
Electrónica Web	Mismo día de emisión	---
Electrónica Por Ciclos	Durante la vigencia del Ciclo	

**Fuente:** (Servicio de Impuestos Nacionales, 2014).

#### **2.3.5. Aspectos técnicos de las facturas.**

El formato general para la facturación indistintamente de cualquier modalidad de facturación que adopte el sujeto pasivo o tercero responsable, sus facturas, notas fiscales o documentos equivalentes deben contener obligatoriamente los siguientes datos: (Servicio de Impuestos Nacionales, 2014).

- Título
- Datos Básicos;
- Datos de Dosificación;
- Datos de la Transacción Comercial;
- Datos Finales.

A continuación se describen con detalle cada uno de estos datos:

**Datos básicos.** Deberán ser consignados en la parte superior izquierda.

- Razón Social en el caso de Personas Jurídicas, Nombre(s) y Apellido(s) en el caso de las Personas Naturales y Empresas Unipersonales. Opcionalmente en el caso de Empresas Unipersonales o de Personas Naturales podrán consignar el Nombre Comercial;
- Domicilio tributario, (casa matriz), número(s) telefónico(s) y la Alcaldía a la que pertenece (o domicilio y Alcaldía declarada para el caso de espectáculos públicos eventuales);
- Número de sucursal, dirección, número(s) telefónico(s) y la Alcaldía a la que pertenece (excepto en espectáculos públicos eventuales); sólo en el caso que la Factura o Nota sea emitida a partir de una dosificación asignada a una sucursal. (Servicio de Impuestos Nacionales, 2014).

**Datos de Dosificación.** Deberán ser consignados en la parte superior derecha:

- Número de Identificación Tributaria (NIT).
- Número de Autorización asignado.
- Número correlativo de Factura, Nota Fiscal o Documento Equivalente,
- El Término “ORIGINAL” o “COPIA” según corresponda,
- Descripción de la Actividad Económica. (Servicio de Impuestos Nacionales, 2014).

**Título y Subtítulo.** Debe ser consignado en la parte superior central:

- Título: Consignar el tipo de Factura, Nota Fiscal o Documento Equivalente, es decir: “FACTURA”, “FACTURA POR TERCEROS”, “FACTURA CONJUNTA”, “COMERCIAL DE EXPORTACIÓN”, “FACTURA TURÍSTICA”, “FACTURA DE VENTA EN ZONAS FRANCAS”, “NOTA CRÉDITO - DÉBITO”, “RECIBO DE ALQUILER”, según corresponda.
- Subtítulo: Consignar las características especiales, es decir: “SIN DERECHO A CRÉDITO FISCAL”, “TASA CERO – SIN DERECHO A CRÉDITO FISCAL” o

“SIN DERECHO A CRÉDITO FISCAL - VENTA MONEDA EXTRANJERA”.  
(Servicio de Impuestos Nacionales, 2014).

**Datos de la Transacción Comercial.** Deberán ser consignados en la parte del cuerpo:

- Lugar y Fecha de emisión, en el siguiente orden: LUGAR, DÍA, MES y AÑO;
- Primer Apellido o Razón Social del comprador.
- Número de Identificación Tributaria (NIT) o número de Cédula de Identidad o Carnet de Extranjería del comprador.
- Domicilio del comprador, sólo en caso de servicios básicos o de tránsito continuo (Electricidad, Agua, Gas domiciliario, Teléfono, Internet, Alquileres).
- Para el caso específico de las Estaciones de Servicio de Venta de Combustible, además de estos datos deben ser registrados el número de placa del vehículo automotor, identificando si es placa nacional o extranjera. Para ventas menores se consignará el tipo de envase: bidones (B), botellas (T) u otros (O).
- Detalle: Cantidad, Concepto, Precio Unitario y Total Ítem (en caso de corresponder: Descuentos e ICE). Excepcionalmente y para el caso de Facturas emitidas en la Modalidad de Facturación Manual, por la venta de alimentos y/o bebidas en restaurantes o similares por un monto menor o igual a Bs300.- (Trescientos 00/100 Bolivianos), se podrá consignar en el campo Concepto o Descripción la leyenda “Consumo”, en los demás casos se debe consignar el detalle de la compra y/o la prestación del servicio.
- Para el caso de las Estaciones de Servicio de Venta de Combustibles es obligatorio el desglose de la cantidad del producto vendido debiendo emitirse una Factura por tipo de producto, además de señalar el importe válido para el Crédito Fiscal del IVA.
- Discriminación del Descuento y del Impuesto al Consumo Específico cuando corresponda.

- Tipo de cambio oficial de venta en moneda nacional correspondiente a la fecha de la transacción, cuando la operación sea en moneda extranjera; con excepción de las Facturas Pre valoradas.
- Total General en bolivianos (numeral y literal), para los casos de Facturas Comerciales de Exportación, aquellas emitidas en Zonas Francas y Facturas Pre valoradas por tarifa para embarque de vuelos internacionales, indistintamente se podrá consignar este dato en moneda nacional (bolivianos) u otra moneda extranjera. Cuando el Total General contenga importes que no están sujetos a crédito fiscal, se debe discriminar el importe que corresponde para crédito fiscal señalando el valor seguido de la leyenda “IMPORTE VÁLIDO PARA CRÉDITO FISCAL”.
- Los Usuarios de Zonas Francas deben consignar el Número de Parte de Recepción, otorgado por el concesionario de Zona Franca, en la(s) venta(s) que comprenden el lote de la mercadería internada con dicho Número. (Servicio de Impuestos Nacionales, 2014).

**Datos Finales para la facturación computarizada en la parte inferior derecha:**

- Consignar en negrillas y mayúsculas la leyenda “LA ALTERACIÓN, FALSIFICACIÓN O COMERCIALIZACIÓN ILEGAL DE ESTE DOCUMENTO TIENE CÁRCEL”.
- Las Facturas emitidas por las Estaciones de Servicio para la venta de Gasolina Especial, Gasolina Premium o Diésel Oil, deben insertar en forma legible la leyenda: “De acuerdo a la Ley N° 317, del importe total de la Factura sólo es válido para el crédito fiscal el 70%.”. (Servicio de Impuestos Nacionales, 2014).

**En la parte inferior izquierda:**

- Código de Control
- Fecha Límite de emisión
- Código de Respuesta Rápida (Código QR)

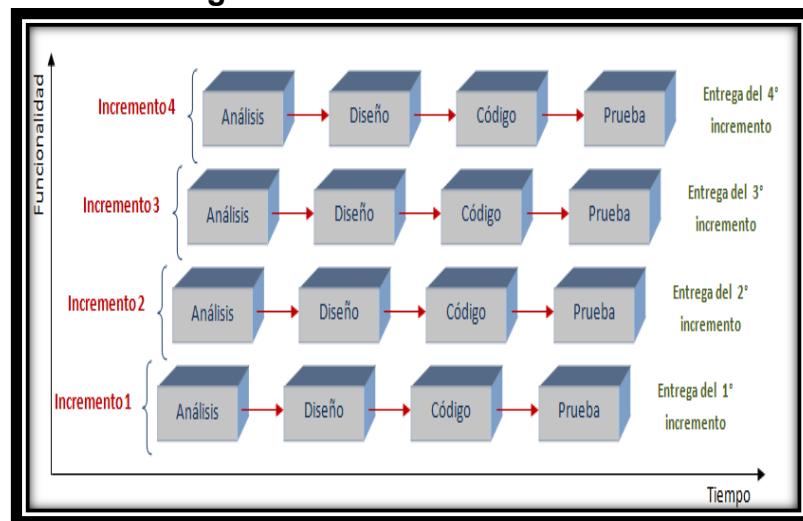
## 2.4. INGENIERÍA DE SOFTWARE.

El proceso de desarrollo de software, es una estructura aplicada para el ciclo de vida de desarrollo de un producto de software, existen varios modelos a seguir, cada uno de los cuales describe un enfoque diferente para cada actividad que tiene lugar durante el proceso. Algunos autores consideran un modelo de ciclo de vida, como un término más general que un determinado proceso para el desarrollo de software.

### 2.4.1. Modelo incremental.

El modelo incremental es aquel que aplica secuencias lineales de forma escalonada mientras avanza el tiempo. Corrige la necesidad de una secuencia no lineal de pasos de desarrollo. Cada secuencia lineal produce un incremento del software. El modelo incremental entrega el software en partes pequeñas, pero utilizables, llamadas “incrementos”. En general, cada incremento se construye sobre aquel que ya ha sido entregado. (Sommerville, 2005).

**Figura 6:** Modelo incremental



Fuente: (Sommerville, 2005).

## **Ventajas modelo incremental**

- Puede monitorearse el cambio relativo de varios aspectos de un producto o pueden proveer los límites de las medidas para apuntar a problemas potenciales y anomalías.
- Se centra en la entrega de un producto operacional con cada incremento.
- Es más fácil probar y depurar en una iteración más pequeña.
- Cada iteración es un hito gestionado fácilmente.
- Proporcionan al usuario la funcionalidad que precisa..
- Los primeros incrementos se pueden implementar con menos personas.
- Permite su desarrollo aunque no se disponga de todos los requerimientos del usuario. (Sommerville, 2005).

## **Desventajas modelo incremental**

- Infunde responsabilidad en el equipo de desarrollo al trabajar directamente con el cliente, requiriendo de profesionales sobre el promedio.
- Cada fase de una iteración es rígida y no se superponen con otras.
- Los primeros incrementos son versiones incompletas del producto final.
- Se maneja tiempos de entrega de los distintos módulos. (Sommerville, 2005).

### **2.4.2. Modelo espiral.**

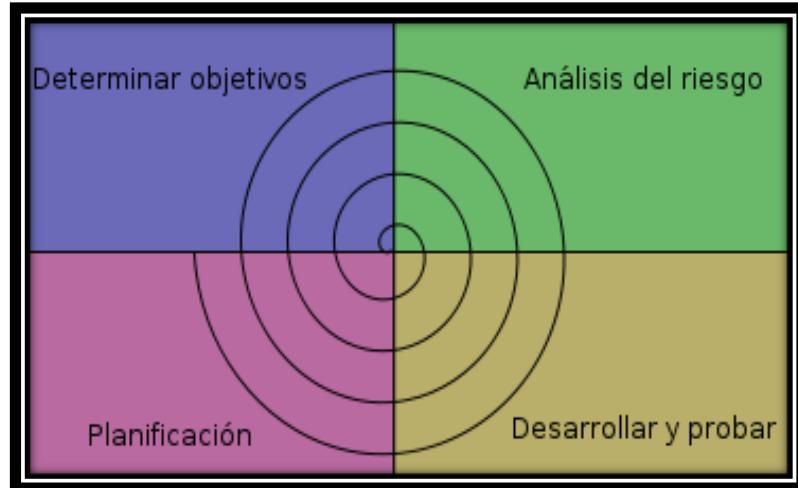
Modelo Espiral, propuesto originalmente por Boehm, es un modelo de proceso evolutivo que conjuga la naturaleza iterativa de construcción de prototipos con los aspectos controlados y sistemáticos del modelo lineal secuencial. Proporciona el potencial para el desarrollo rápido de versiones incrementales del software.

En el modelo espiral, el software se desarrolla en una serie de versiones incrementales. Durante las primeras iteraciones, la versión incremental podría ser un

modelo en papel o un prototipo. Durante las últimas iteraciones, se producen versiones cada vez más completas del sistema diseñado. (Pressman Roger S, 2014).

A continuación en la figura 7 se muestra el ciclo del modelo espiral para su mejor entendimiento:

**Figura 7:** Modelo espiral



**Fuente:** (Pressman Roger S., 2010).

#### **Ventajas modelo en espiral:**

- Puede adaptarse y aplicarse a lo largo de la vida del software de computadora.
- Es un enfoque realista del desarrollo de sistemas y de software a gran escala.
- Como el software evoluciona, a medida que progresa el proceso el desarrollador y el cliente comprende y reaccionan mejor ante riesgos en cada uno de los niveles evolutivos.
- Utiliza la construcción de prototipos como mecanismo de reducción de riesgos.
- Permite a quien lo desarrolla aplicar el enfoque de construcción de prototipos en cualquier etapa de evolución del producto.

- Mantiene el enfoque sistemático de los pasos sugeridos por el ciclo de vida clásico, pero lo incorpora al marco de trabajo iterativo que refleja de forma más realista el mundo real.
- Demanda una consideración directa de los riesgos técnicos en todas las etapas del proyecto, y si se aplica adecuadamente, debe reducir los riesgos antes de que se conviertan en problemáticos. (Pressman Roger S., 2010).

### **Desventajas modelo en espiral**

- Puede resultar difícil convencer a grandes clientes (particularmente en situaciones bajo contrato) de que el enfoque evolutivo es controlable.
- Requiere una considerable habilidad para la evaluación del riesgo.
- No se ha utilizado tanto como los paradigmas lineales secuenciales o de construcción de prototipos. (Pressman Roger S., 2010).

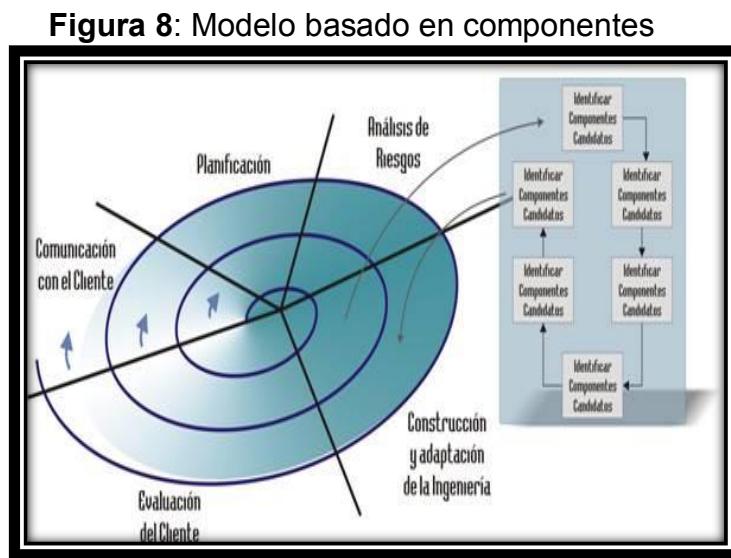
#### **2.4.3. Modelo basado en componentes.**

En esencia, un componente es una pieza de código pre elaborado que encapsula alguna funcionalidad expuesta a través de interfaces estándar . Los componentes son los "ingredientes de las aplicaciones", que se juntan y combinan para llevar a cabo una tarea. Es algo muy similar a lo que podemos observar en el equipo de música que tenemos en nuestra sala.

Cada componente de aquel aparato ha sido diseñado para acoplarse perfectamente con sus pares, las conexiones son estándar y el protocolo de comunicación está ya preestablecido. Al unirse las partes, obtenemos música para nuestros oídos.

El paradigma de ensamblar componentes y escribir código para hacer que estos componentes funcionen se conoce como Desarrollo de Software Basado en Componentes. (Microsoft, 2014).

A continuación en la siguiente imagen se muestran los pasos del modelo basado en componentes para su mejor entendimiento:



Fuente: (Microsoft, 2014)

#### Ventajas modelo basado en componentes:

- Reutilización del software. Lleva a alcanzar un mayor nivel de reutilización de software.
- Simplifica las pruebas. Permite que las pruebas sean ejecutadas probando cada uno de los componentes antes de probar el conjunto completo de componentes ensamblados.
- Simplifica el mantenimiento del sistema. Cuando existe un débil acoplamiento entre componentes, el desarrollador es libre de actualizar y/o agregar componentes según sea necesario, sin afectar otras partes del sistema.
- Mayor calidad. Dado que un componente puede ser construido y luego mejorado continuamente por un experto u organización, la calidad de una aplicación basada en componentes mejorará con el paso del tiempo. (Microsoft, 2014).

### **Desventajas modelo basado en componentes:**

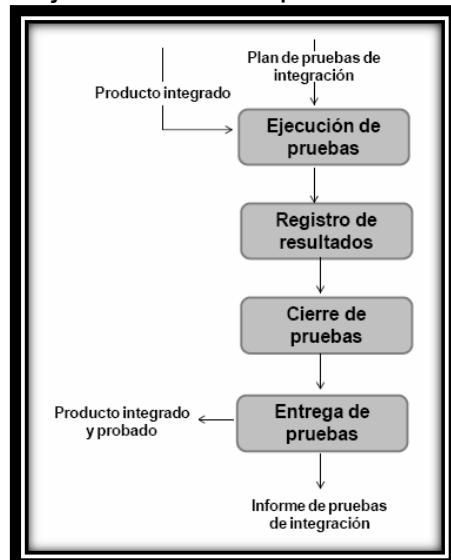
- Es necesario tener un manejo excelente de los conceptos, lo cual puede resultar difícil para quien no tenga experiencia en el uso de procesos de ingeniería de software.
- Los “compromisos” en los requisitos son inevitables, por lo cual puede que el software no cumpla las expectativas del cliente.
- Las actualizaciones de los componentes adquiridos no están en manos de los desarrolladores del sistema. (Microsoft, 2014).

#### **2.4.4. Pruebas de software.**

##### **2.4.4.1. Pruebas de integración.**

Consiste en construir el sistema a partir de los distintos componentes y probarlo con todos integrados. Estas pruebas deben realizarse progresivamente. El foco de atención es el diseño y la construcción de la arquitectura de software. (Universidad Cibertec, 2014).

**Figura 9:** Flujo de control de pruebas de integración



**Fuente:** (Universidad Cibertec, 2014).

#### **2.4.4.2. Pruebas de despliegue.**

En muchos casos, el software debe ejecutarse en varias plataformas y bajo más de un entorno de sistema operativo. La *prueba de despliegue*, en ocasiones llamada *prueba de configuración*, ejercita el software en cada entorno en el que debe operar. Una prueba de despliegue más profunda puede abarcar combinaciones de navegadores web con varios sistemas operativos (por ejemplo, Linux, Mac OS, Windows). Puesto que la seguridad es un tema principal, un juego completo de pruebas de seguridad se integraría con la prueba de despliegue. (Pressman, 2010).

#### **2.4.4.3. Pruebas de aceptación.**

Son las únicas pruebas que son realizadas por los usuarios expertos, todas las anteriores las lleva a cabo el equipo de desarrollo. Consiste en comprobar si el producto está listo para ser implantado para el uso operativo en el entorno del usuario final. (Universidad Cibertec, 2014).

Podemos distinguir entre dos tipos de pruebas principalmente las cuales son las pruebas alfas y las pruebas beta, cabe decir que en ambas existe retroalimentación por parte del usuario experto:

**Pruebas alfa:** las realiza el usuario en presencia de personal de desarrollo del proyecto haciendo uso de una máquina preparada para las pruebas.

**Pruebas beta:** las realiza el usuario después de que el equipo de desarrollo les entregue una versión casi definitiva del producto. (Universidad Cibertec, 2014).

#### **2.4.4.4. Pruebas funcionales.**

Este tipo de prueba se realiza sobre el sistema funcionando, comprobando que cumpla con la especificación (normalmente a través de los casos de uso). Para estas pruebas, se utilizan las especificaciones de casos de prueba. (Universidad Cibertec, 2014).

## **2.5. HERRAMIENTAS DE DESARROLLO.**

### **2.5.1. Frameworks para el desarrollo de aplicaciones web.**

#### **2.5.1.1. Codelgniter.**

Codelgniter es un framework para desarrollo de aplicaciones tiene un conjunto de herramientas para gente que construye sitios web usando PHP como lenguaje base. Su objetivo es permitir el desarrollo de proyectos mucho más rápido que hacerlo escribiendo el código desde cero, proveyendo un conjunto de bibliotecas para tareas comunes, así como una interfaz sencilla y una estructura lógica para acceder a esas bibliotecas. Codelgniter le permite enfocarse creativamente al proyecto al minimizar la cantidad de código necesaria para una tarea realizada. (Codelgniter, 2014)

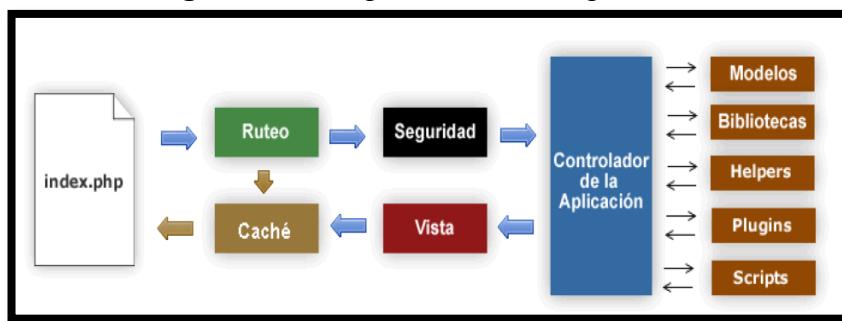
#### **Ventajas Codelgniter**

- Codelgniter es libre bajo licencias open source.
- Es liviano y rápido ya que el núcleo del sistema sólo requiere algunas bibliotecas muy pequeñas y las bibliotecas adicionales se cargan dinámicamente bajo pedido del usuario basado en sus necesidades.
- Codelgniter usa el enfoque Modelo-Vista-Controlador, que permite una gran separación entre la lógica y la presentación. Es particularmente bueno para proyectos en los que los diseñadores trabajan en sus archivos de plantillas, ya que el código en estos archivos será mínimo.
- Codelgniter tiene una gama completa de bibliotecas que facilitan las tareas de desarrollo web más comúnmente usadas, como acceso a base de datos, envío de correo electrónico, validación de datos de formularios, manejo de sesiones, manipulación de imágenes, etc.
- El sistema se puede extender fácilmente a través de sus propias bibliotecas, helpers, clases. (Codelgniter, 2014).

## Desventajas de codeigniter

- Es un framework que casi no necesite configuración.
- Es un framework que no usa la línea de comandos.
- Es un framework que no utiliza reglas de codificación restrictivas.
- tiene un lenguaje de plantillas y un motor de plantillas disponible.
- Tiene complejidad, pero también favorece a soluciones.
- No tiene una documentación clara y completa.(CodeIgniter, 2014)

**Figura 10:** Diagrama de codeigniter.



**Fuente:** (CodeIgniter, 2014)

- El index.php sirve como el controlador frontal, inicializando los recursos básicos que necesita CodeIgniter para ejecutar.
- El Ruteado examina la solicitud HTTP para determinar qué debería hacer con ella.
- Si existe el archivo de caché, se lo envía directamente al navegador, sin pasar por la ejecución normal del sistema.
- Seguridad. Antes de que se cargue el controlador de la aplicación, por razones de seguridad se filtran la solicitud HTTP y cualquier otro dato enviado por el usuarios.
- El controlador carga el modelo, las bibliotecas del núcleo, helpers, y cualquier otro recurso requerido para procesar una solicitud específica.

- La Vista terminada se procesa y se envía al navegador para que se pueda ver. Si el caché está habilitado, la vista se cachea primero para que las siguientes solicitudes que la necesiten puedan ser servidas. (CodeIgniter, 2014).

### **2.5.1.2. PhoneGap.**

PhoneGap (anteriormente llamado Apache Callback, pero actualmente Apache Cordova) es un framework de desarrollo de código abierto para móvil producido por Nitobi, y actualmente comprado por Adobe Systems.

Permite a los programadores de software desarrollar aplicaciones web multiplataforma usando JavaScript, HTML5 y CSS3, en vez de lenguajes de bajo nivel como Objective-C. Las aplicaciones resultantes son híbridos, lo que significa que no son ni verdaderamente nativa (todos el rendimiento de diseño se realiza a través del navegador web de la plataforma nativa de la interfaz de usuario) ni puramente basado en la web (que no son aplicaciones sólo web pero empaquetados para la distribución de la AppStore, Android o cualquier otra y tener acceso a una parte de las funciones del dispositivo).

PhoneGap actualmente soporta para desarrollar en sistemas operativos como Apple iOS, Google Android, HP webOS, Microsoft Windows Phone, Nokia Symbian OS RIM BlackBerry. Soporta versiones recientes como BlackBerry 5 y 6 y Windows Phone 7. (Universidad Carlos III de Madrid, 2014).

#### **Ventajas de PhoneGap.**

- Es gratuito.
- Es multiplataforma ya que corre dentro de un navegador web. Compatible con varias plataformas iOS, Android, Blackberry, Windows Phone, webOS, Symbian y Bada.

- Es fácil de desarrollar y ofrece muchas posibilidades a aquellos que conocen bien los lenguajes basados en tecnologías web (HTML, CSS y JavaScript).
- Existe mucha documentación acerca de PhoneGap e incluso la propia web proporciona muchos ejemplos.
- La alternativa que ofrece mediante el compilador en la nube es una forma muy sencilla que permite mejorar muchas de las desventajas de PhoneGap.

### **Desventajas de PhoneGap.**

- En algunos casos es necesario usar el sistema operativo de la plataforma. Por ejemplo empaquetar aplicaciones Windows Phone solo es posible con el sistema operativo Windows. Lo mismo ocurre con iOS, es necesario usar un Mac.
- Dependiendo de la plataforma se necesita un sistema diferente, para Android se requiere el uso de Eclipse y para iOS el uso de Xcode.
- En caso de compilar para iOS es necesario introducir un código de desarrollador dado por la empresa Apple. (Glera, 2014).

#### **2.5.1.3. JQuery Mobile.**

jQuery Mobile es una librería de Javascript optimizada para web en móviles que actualmente está siendo desarrollada por el equipo del proyecto jQuery.

El desarrollo se centra en crear un framework compatible con el mayor número de smartphones o tablets debido a la gran heterogeneidad que hay en el mercado de estos dispositivos móviles. Es compatible con otros frameworks de aplicaciones para móvil como PhoneGap, Worklight y otros. (Chaffer y Swedberg, 2007).

#### **Las ventajas de jQuery Mobile:**

- Compatible con la mayoría de las plataformas móviles como la mayoría de los navegadores: iOS, Android, Blackberry, WebOS, Symbian, Windows Phone 7.

- Construido sobre el núcleo jQuery, para que la curva de aprendizaje sea mínima para desarrolladores familiarizados con la sintaxis de jQuery.
- Un framework de temas que permite personalizar uno rápidamente.
- Dependencias limitadas y ligero que optimizan su velocidad.
- El mismo código de base escalará el tamaño automáticamente a cualquier pantalla.
- Posibilidad de navegación entre páginas por AJAX con animaciones en las transiciones de página.
- UI widgets optimizados para táctil y para quien desconozca la plataforma.(Chaffer y Swedberg, 2007)

### **Las desventajas de jQuery Mobile:**

- Las funciones que ofrece son muchas, pero resultan difíciles de personalizar. Su aspecto visual es estandarizado y no se integra con el de la plataforma. En algunos casos, no queda otra opción que usar JavaScript simple para adaptar la aplicación a nuestras necesidades.
- Como es necesario invocar a un archivo para utilizar sus funciones, ralentiza ligeramente la carga de la página.
- Su manejo de CSS suele resultar innecesariamente complejo. A veces cuesta saber qué clases utilizar.
- No existen muchas plantillas prediseñadas sobre las cuales empezar a construir nuestra aplicación. (Chaffer y Swedberg, 2007).

### **2.5.2. Lenguajes de programación para desarrollo de aplicaciones web.**

#### **2.5.2.1. Php**

(PHP Hypertext Pre-processor) es un lenguaje de programación interpretado, diseñado en 1994 para la creación de páginas web dinámicas. Siendo multiplataforma y teniendo un gran parecido con los lenguajes más comunes de programación estructurada

permiten que el aprendizaje sea muy corto. Todo esto junto con la facilidad de instalación hizo que PHP cogiera fama muy rápidamente. Cuando el cliente realiza una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP procesando el script solicitado y generando el contenido de manera dinámica. (The PHP Development Team, 2014).

#### **Las ventajas de PHP son:**

- Lenguaje multiplataforma.
- Orientado al desarrollo de web dinámicas con acceso a información de base de datos.
- Soporte con diferentes tipos de servidores de bases de datos, como por ejemplo MySQL.
- Integración con librerías externas hace que la funcionalidad aumente bastante.
- Código invisible al usuario que lo solicita, programación segura y confiable.
- Permite aplicar técnicas de programación orientada a objetos.
- No requiere definición de tipos, sino que decide en periodo de ejecución.
- Manejo de excepciones.
- Uso libre. (The PHP Development Team, 2014)

#### **Las desventajas de PHP son:**

- la desventaja más grande es que al ser un lenguaje interpretado se necesita tener el intérprete de PHP, y esto trae al alto consumo de las aplicaciones, ya que son imposibles de compilar. (Ecured, 2014).

#### **2.5.2.2. Java**

Es un lenguaje orientado a objetos, es decir un POO, lo que significa que toda la programación JAVA gira en torno a clases. Podemos decir que una clase es un molde del que se pueden hacer copias y cada copia es un objeto.

El un lenguaje multiplataforma, esto significa que no está enlazado a un sistema operativo en concreto y los programas desarrollados con el funcionaran correctamente tanto en Windows como en Linux. (García, 2014).

### **Ventajas de Java**

- Manejo es un lenguaje multiplataforma.
- Es orientado a objetos.
- Su sintaxis es similar a c++.
- Fácil de aprender.
- Es libre.

### **Desventajas de Java**

- Requiere un intérprete.
- Una mala implementación de un programa java resulta lento.

#### **2.5.2.3. ASP.NET**

ASP.NET proporciona un modelo de desarrollo Web unificado que incluye los servicios necesarios para crear aplicaciones Web. Se trata de un entorno compilado basado en .NET, como C# o Visual Basic. Además .NET Framework está disponible en su totalidad para cualquier aplicación ASP.NET, permitiéndola así interactuar con el sistema operativo.

Las aplicaciones ASP.NET se implementan a través de mecanismos que permiten a un navegador, o cliente Web, acceder a una página Web a través de su dirección (URL). Todas las páginas y demás ficheros que componen una aplicación ASP.NET son puestas a disposición de un usuario a través de IIS.

Cualquier persona que está familiarizada con el desarrollo de aplicaciones web sabrá que el desarrollo web no es una tarea simple. Ya que mientras que un modelo de programación para aplicaciones de uso común está muy bien establecido y soportado

por un gran número de lenguajes, herramientas de desarrollo, la programación web es una mezcla de varios lenguajes de etiquetas, un gran uso de lenguajes de script y plataformas de servidor. Para el programador de nivel intermedio, el conocimiento y las habilidades que se necesitan para desarrollar aplicaciones web requieren de un mayor conocimiento tanto de lenguajes de programación, etiquetado y formato, como de diversas tecnologías de software relativas al desarrollo distribuido y concurrente, de las que son necesarias en el desarrollo tradicional de aplicaciones.(Microsoft, 2014).

### **Ventajas ASP.NET**

- Permite actualizaciones automáticas del Software.
- Disponibilidad garantizada.
- Permite probar software antes de comprarlo.
- Se encarga de detectar el tipo de navegador utilizado por el cliente a la hora de realizar una petición al servidor y en consecuencia, determina la versión HTML que este soporta.

### **Desventajas ASP.NET**

- No es multiplataforma, tiene que correr en Pc normales que tengan Windows y un servidor Web. (Microsoft, 2014).

### **2.5.3. Herramientas para el desarrollo de aplicaciones web.**

#### **2.5.3.1. Html5.**

Al HTML5 (Lenguaje de marcado de hipertexto, versión 5) es el lenguaje en que hoy en día se programan las páginas web y también es la quinta versión importante del lenguaje básico HTML. HTML5 todavía se encuentra en modo experimental, lo cual indica la misma W3C, aunque ya es usado por múltiples desarrolladores web por sus avances, mejoras y ventajas. Al no ser reconocido en viejas versiones de navegadores por sus nuevas etiquetas, se le recomienda al usuario común actualizar a la versión

más nueva, para poder disfrutar de todo el potencial que provee HTML5 el desarrollo de este lenguaje de marcado es regulado por el Consorcio W3C.

HTML5 establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos. Algunos de ellos son técnicamente similares a las

Etiquetas `<div>` y `<span>`, pero tienen un significado semántico, como por ejemplo `<nav>` (Bloque de navegación del sitio web) y `<footer>`. Otros elementos proporcionan nuevas Funcionalidades a través de una interfaz estandarizada, como los elementos `<audio>` y `<Video>`. Mejoras en el elemento `<canvas>`, capaz de renderizar en los navegadores más importantes (Mozilla, Crome, Opera, Safari e IE) elementos 3D. Algunos elementos de HTML 4.01 han quedado obsoletos, incluyendo elementos puramente de presentación, como `<font>` y `<center>`, cuyos efectos son manejados por el CSS. (World Wide Web Consortium, 2014)

### **Las ventajas de HTML 5 son:**

- Incorpora etiquetas (canvas 2D y 3D, audio, video) con codecs para mostrar los Contenidos multimedia. Actualmente hay una lucha entre imponer codecs libres (WebM + VP8) o privados (H.264/MPEG-4 AVC).
- Dispone de etiquetas para manejar grandes conjuntos de datos: Datagrid, Details, Menu y Command. Permiten generar tablas dinámicas que pueden filtrar, ordenar y Ocultar contenido en cliente.
- Mejoras en los formularios. Nuevos tipos de datos (eMail, number, url, datetime) y facilidades para validar el contenido sin Javascript. Visores: MathML (fórmulas matemáticas) y SVG (gráficos vectoriales). En general se deja abierto a poder interpretar otros lenguajes XML.
- Drag & Drop. Nueva funcionalidad para arrastrar objetos como imágenes.(World Wide Web Consortium, 2014).

### **Las Desventajas de Html 5 son:**

- No tiene semántica. Uso de etiquetas con nombres diferentes.
- El contenido no puede ser reconocido ni procesado por programas
- Tiene un costoso mantenimiento de las páginas
- No tiene estándares comunes.
- Solo tiene hiperenlaces simples (XML puede tener de 1 a n enlaces). (euskal, 2014).

#### **2.5.3.2. Java Script.**

**JavaScript** (abreviado comúnmente "JS") es un lenguaje de programación interpretado, se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipiado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web para interactuar.

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. JavaScript se interpreta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML. (Sánchez, 2014).

## **Ventajas de Java Script**

- Ligero de carga.
- Fácil de integrar.
- Cientos de aplicaciones disponibles para uso
- Puede agregar interactividad a elementos web.
- Compatible con la gran mayoría de los navegadores modernos incluyendo iPhone, mobiles & PS3. (Sánchez, 2014).

## **Desventajas de Java Script**

- No es tan extenso en recursos como Flash.
- Sistemas complejos pueden no funcionar tan bien como Flash.
- Opciones de 3D limitadas.
- Usuarios pueden deshabilitar JavaScript en su navegador. (Sánchez, 2014).

### **2.5.3.3. CSS3**

El nombre hojas de estilo en cascada viene del inglés Cascading Style Sheets, del que toma sus siglas. CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación.

La información de estilo puede ser adjuntada como un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "<style>". (Ojeda, 2014).

## **Ventajas CSS3**

- Se obtiene un mayor control de la presentación del sitio al poder tener todo el código CSS reunido en uno, lo que facilita su modificación.
- Al poder elegir el archivo CSS que deseamos mostrar, puede aumentar la accesibilidad ya que podemos asignarle un código CSS concreto a personas con deficiencias visuales, por ejemplo. Esto lo detecta el navegador web. Conseguimos hacer mucho más legible el código HTML al tener el código CSS aparte (Siempre que no usemos estilos en línea, claro está).
- Pueden mostrarse distintas hojas de estilo según el dispositivo que estemos utilizando (versión impresa, versión móvil, leída por un sintetizador de voz...) o dejar que el usuario elija.
- Gracias a la técnica CSS podemos aligerar la carga de nuestro sitio al juntar todas las imágenes en una.
- Las novedades de CSS3 nos permiten ahorrarnos tiempo y trabajo al poder seguir varias técnicas (bordes redondeados, sombra en el texto, sombra en las cajas, etc.) sin necesidad de usar un editor gráfico. (Ojeda, 2014)

## **Desventajas CSS3**

- Dependiendo del navegador la página que ha sido maquetada con CSS puede verse distinta (Aunque, si hemos seguido los estándares web de forma correcta, el problema es del navegador). (Ojeda, 2014)

### **2.5.3.4. JQuery**

JQuery es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en

JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

Las empresas Microsoft y Nokia anunciaron que incluirán la biblioteca en sus plataformas. Microsoft la añadirá en su IDE Visual Studio y la usará junto con los frameworks ASP.NET AJAX y ASP.NET MVC, mientras que Nokia los integrará con su plataforma Web Run-Time. (Chaffer y Swedberg, 2007).

### **Las ventajas de JQuery son:**

- La ventaja principal de jQuery es que es mucho más fácil que usar css, ya que se puede agregar plugins fácilmente, traduciéndose esto en un ahorro substancial de tiempo y esfuerzo. De hecho, una de las principales razones por la cual crearon jQuery fue para ganar tiempo (en el mundo de desarrollo web).
- La licencia open source de jQuery permite que la librería siempre cuente con soporte constante y rápido, publicándose actualizaciones de manera constante. La comunidad jQuery es activa.
- Otra ventaja de jQuery sobre sus competidores como Flash y puro CSS es su excelente integración con AJAX.

### **Las desventajas de JQuery son:**

- Una de las principales desventajas de jQuery es la gran cantidad de versiones publicadas en el corto tiempo. No importa si usted está corriendo la última versión de jQuery, usted tendrá que hostear la librería usted mismo (y actualizarla constantemente), o descargar la librería desde Google (atractivo, pero puede traer problemas de incompatibilidad con el código).
- Su aprendizaje es más complicado y requiere más tiempo que css.

- Si jQuery es implementado inapropiadamente como un Framework, el entorno de desarrollo se puede salir de control.

## 2.6. GESTORES DE BASES DE DATOS.

### 2.6.1. MySQL.

MySQL es un sistema gestor de bases de datos (SGBD, DBMS por sus siglas en inglés) muy conocido y ampliamente usado por su simplicidad y notable rendimiento. Aunque carece de algunas características avanzadas disponibles en otros SGBD del mercado, es una opción atractiva tanto para aplicaciones comerciales, como de entretenimiento precisamente por su facilidad de uso y tiempo reducido de puesta en marcha. Esto y su libre distribución en Internet bajo licencia GPL le otorgan como beneficios adicionales (no menos importantes) contar con un alto grado de estabilidad y un rápido desarrollo.

MySQL está disponible para múltiples plataformas, la seleccionada para los ejemplos de este libro es GNU/Linux. Sin embargo, las diferencias con cualquier otra plataforma son prácticamente nulas, ya que la herramienta utilizada en este caso es el cliente *mysql-client*, que permite interactuar con un servidor MySQL (local o remoto) en modo texto. De este modo es posible realizar todos los ejercicios sobre un servidor instalado localmente o, a través de Internet, sobre un servidor remoto.

Para la realización de todas las actividades, es imprescindible que dispongamos de los datos de acceso del usuario administrador de la base de datos. Aunque en algunos de ellos los privilegios necesarios serán menores. (Casillas, 2014).

#### Ventajas de MySQL:

- Mysql es Open Source: significa que es posible usar y modificar el software gratuitamente.
- Trabaja en entornos cliente servidor o incrustados.

- Posee control de acceso de usuarios y seguridad en los datos.
- Integra perfectamente el lenguaje PHP.
- Soporte complejo para cláusulas, funciones, tipos de datos y comandos estándar.
- Soporte para control de transacciones en tablas transaccionales.
- Gran portabilidad entre los diferentes sistemas y plataformas.

### **Desventajas de MySQL**

- Un gran porcentaje de las utilidades de MySQL no están documentadas en español. (Casillas, 2014).

#### **2.6.2. SQL Server 2012.**

SQL Server 2012 es la última plataforma de base de datos de Microsoft y está preparada para la nube de Microsoft. Las organizaciones pueden utilizar SQL Server 2012 para proteger de manera eficiente, desbloquear, y escalar sus datos, dispositivo móvil, centro de datos, y ya sea una nube privada o pública. Basándose en el éxito del SQL Server 2008 versión R2, SQL Server 2012 ofrece una variedad de soluciones que puede elegir. (Microsoft, 2014).

#### **Ventajas de SQL Server 2012:**

- Utiliza una extensión al SQL estándar, que se denomina Transact SQL. El Transact SQL, soporta la definición, modificación y eliminación de bases de datos, tablas, atributos, índices, etc., es decir, el lenguaje de definición de datos , así como la consulta, actualización y borrado de tuplas de tablas, es decir, el lenguaje de manipulación de datos.
- Para empezar SQL Server nos permite pasar desapercibido los ficheros que forman la base de datos ya que al simplemente abrir la consola de SQL Server tenemos todas las bases de datos de todos nuestros programas.

- Si trabajamos en una red local permite agregar otros servidores de SQL Server, una base de datos en servidor local SQL y otra base de datos, se pueden conectar los SQL entre si y acceder a las bases de datos.
- Seguridad: SQL permite administrar permisos, permisos a nivel de servidor, seguridad en tablas, permitir o no lectura, escritura, ejecución; seguridad en los procedimientos almacenados, también tiene la opción de permitir que una tercera persona se conecte a la SQL, pero sin embargo se le puede dar permisos solo a ciertas bases de datos. (Microsoft, 2014).

#### **Desventajas de SQL Server 2012:**

- la enorme cantidad de memoria RAM que utiliza para la instalación y utilización del software.
- Una de las desventajas de SQL es que si lo quieres para prácticas no te va hacer útil porque en él se prohíben muchas cosas tiene restricciones a diferencia sobre MySQL.
- La relación calidad-precio está muy debajo comparado con MySQL.

(Microsoft, 2014)

#### **2.6.3. PostgreSQL.**

PostgreSQL es un gestor de bases de datos orientadas a objetos (SGBDOO o ORDBMS en sus siglas en inglés) muy conocido y usado en entornos de software libre porque cumple los estándares SQL92 y SQL99, y también por el conjunto de funcionalidades avanzadas que soporta, lo que lo sitúa al mismo o a un mejor nivel que muchos SGBD comerciales.

El origen de PostgreSQL se sitúa en el gestor de bases de datos POSTGRES desarrollado en la Universidad de Berkeley y que se abandonó en favor de PostgreSQL a partir de 1994. Ya entonces, contaba con prestaciones que lo hacían único en el

mercado y que otros gestores de bases de datos comerciales han ido añadiendo durante este tiempo.

PostgreSQL se distribuye bajo licencia BSD, lo que permite su uso, redistribución, modificación con la única restricción de mantener el *copyright* del software a sus autores, en concreto el PostgreSQL Global Development Group y la Universidad de California.

PostgreSQL puede funcionar en múltiples plataformas (en general, en todas las modernas basadas en Unix) y, a partir de la próxima versión 8.0 (actualmente en su segunda beta), también en Windows de forma nativa. Para las versiones anteriores existen versiones binarias para este sistema operativo, pero no tienen respaldo oficial. (Marc, 2014).

### **Ventajas de PostgreSQL:**

- A pesar de que la velocidad de respuesta pueda parecer deficiente en bases de datos pequeñas, esa velocidad se mantiene al aumentar el tamaño de la base de datos, cosa que no sucede con otros programas, que se enlentecen brutalmente.
- Instalación ilimitada: No se puede demandar a una empresa por instalarlo en más ordenadores de los que la licencia permite, ya que no hay costo asociado a la licencia de *software*. Esto permite un negocio más rentable con instalaciones a gran escala, no debemos preocuparnos por ser auditados para ver si cumplimos con la licencia, y hay flexibilidad y desarrollo sin costos adicionales de licenciamiento.
- Ahorros considerables de costos de operación: PostgreSQL ha sido diseñado para tener un mantenimiento y ajuste menor que los productos de proveedores comerciales, conservando todas las características, estabilidad y rendimiento.

- Estabilidad y confiabilidad: No se han presentado caídas de la base de datos.
- Extensible: El código fuente está disponible de forma gratuita, para que quien necesite extender o personalizar el programa pueda hacerlo sin costes.
- Multiplataforma: Está disponible en casi cualquier Unix, con 34 plataformas en la última versión estable, además de una versión nativa de Windows en estado de prueba.
- Diseñado para ambientes de alto volumen: Utilizando una estrategia de almacenamiento de filas llamada MVCC, consigue mejor respuesta en grandes volúmenes. Además, MVCC permite a los accesos de solo lectura continuar leyendo datos consistentes durante la actualización de registros, permitiendo copias de seguridad en caliente
- Herramientas gráficas de diseño y administración de bases de datos.
- Soporta los tipos de datos, cláusulas, funciones y comandos de tipo estándar SQL92/SQL99 y extendido propios de PostgreSQL.
- Puede operar sobre distintas plataformas, incluyendo Linux, Windows, Unix, Solaris y MacOS X.
- Buen sistema de seguridad mediante la gestión de usuarios, grupos de usuarios y contraseñas.
- Gran capacidad de almacenamiento.
- Buena escalabilidad ya que es capaz de ajustarse al número de CPU y a la cantidad de memoria disponible de forma óptima, soportando una mayor cantidad de peticiones simultáneas a la base de datos de forma correcta. (Marc, 2014).

### **Desventajas de PostgreSQL:**

- En comparación con MySQL es más lento en inserciones y actualizaciones, ya que cuenta con cabeceras de intersección que no tiene MySQL.
- Soporte en línea: Hay foros oficiales, pero no hay una ayuda obligatoria.

- Consume más recursos que MySQL.
- La sintaxis de algunos de sus comandos o sentencias no es nada intuitiva. (Marc, 2014).

## 2.7. Librerías.

A continuación se realizará una descripción de las librerías a usar en el presente proyecto.

### 2.7.1. Librería PHP Código QR.

PHP Código QR es de código abierto, esta es una biblioteca para la generación de código QR, código de barras de 2 dimensiones. Basado en libqrencode biblioteca C, proporciona API para crear Código QR imágenes de código de barras (PNG, JPEG gracias a GD2). Implementado puramente en PHP, sin dependencias externas. (The PHP Development Team, 2014).

Las características de la biblioteca son:

- Soporta las versiones de código QR (tamaño) 1-40
- Numérico, alfanumérico, 8 bits y la codificación del kanji. (Codificación kanji no fue totalmente probado, si usted es japan-codificación activado, puede contribuir verifing él :))
- Implementadas puramente en PHP, sin dependencias externas excepto GD2
- Las exportaciones a PNG, imágenes JPEG, también exporta como bit-tabla
- TCPDF 2-D integración API de código de barras
- Fácil de configurar
- Caché de datos para el cálculo de aceleración
- Herramienta de combinación proporcionado ayuda a desplegar la biblioteca como un archivo de una gran dependencia de las personas menos, sencillo de "inclusión y no worry"

- Volcado de datos de depuración, el registro de errores, la evaluación comparativa tiempo
- documentación de la API
- Ejemplos detallados
- 100% de código abierto, con licencia GPL

### **2.7.2. Librería FPDF pdf generador.**

FPDF es una clase escrita en PHP que permite generar documentos PDF directamente desde PHP, es decir, sin usar la biblioteca PDFlib. La F de FPDF significa *Free* (gratis y libre): puede usted usarla para cualquier propósito y modificarla a su gusto para satisfacer sus necesidades. (FPDF, 2014).

FPDF tiene otras ventajas: funciones de alto nivel. Esta es una lista de sus principales características:

- Elección de la unidad de medida, formato de página y márgenes
- Gestión de cabeceras y pies de página
- Salto de página automático
- Salto de línea y justificación del texto automáticos
- Admisión de imágenes (JPEG, PNG y GIF)
- Colores
- Enlaces
- Admisión de fuentes TrueType, Type1 y codificación
- Compresión de página

### **3. MARCO PRÁCTICO.**

#### **3.1. DISEÑO DEL MODELADO DE NEGOCIO ALTERNATIVO DEL PROYECTO DE GESTIÓN DE PENSIONES.**

Para el desarrollo del sistema multiplataforma, se realizará el modelado de negocio actual, mediante el cual se va a describir detalladamente el estado de los procedimientos y actividades actuales para el proceso de gestión de pensiones posteriormente se propondrá un modelado de negocio alternativo, en el cual se pretende optimizar el proceso que se lleva a cabo, mediante la automatización de varios de los procesos.

El modelado alternativo cuenta con 3 actores principales: la encargada de cobro de pensiones, el interesado en pagar pensiones y el encargado de entrega de boletín de notas.

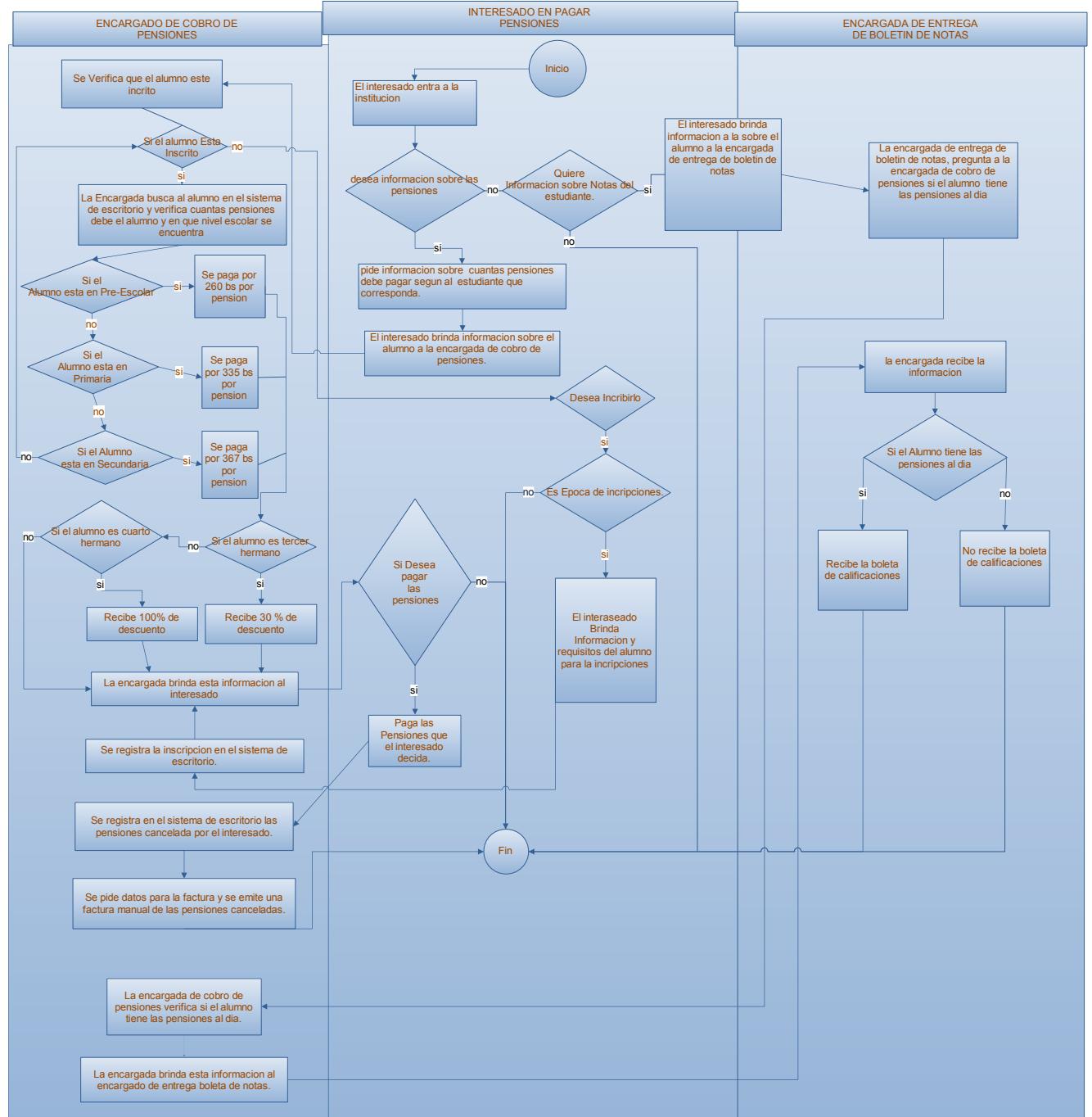
**Encargada de cobro de pensiones:** Cumple la función de secretaria y a la vez es la que cobra las mensualidades de los estudiantes, también tiene la función de brindar información sobre el estado de pensiones de los estudiantes y brindar información a la encargada de boletín de notas para que entregue las respectivas notas cumpliendo las normas del colegio.

**El interesado en pagar pensiones:** puede ser el mismo estudiante, el padre de familia, el tutor del estudiante o cualquier tercera persona interesada en pagar pensiones.

**Encargada de Entrega de boletín de notas:** esta es la persona encargada de realizar la entrega de los boletines de notas, según las reglas del colegio solo puede entregar las notas a los estudiantes que estén al día con sus respectivas pensiones.

### 3.1.1. Modelado de negocio actual.

**Figura 11: Modelado de negocio actual.**



Fuente: (Elaboración propia).

### **3.1.2. Evaluación de las deficiencias existentes en el proceso actual.**

En la siguiente tabla se muestran las deficiencias existentes en el modelado de negocio actual.

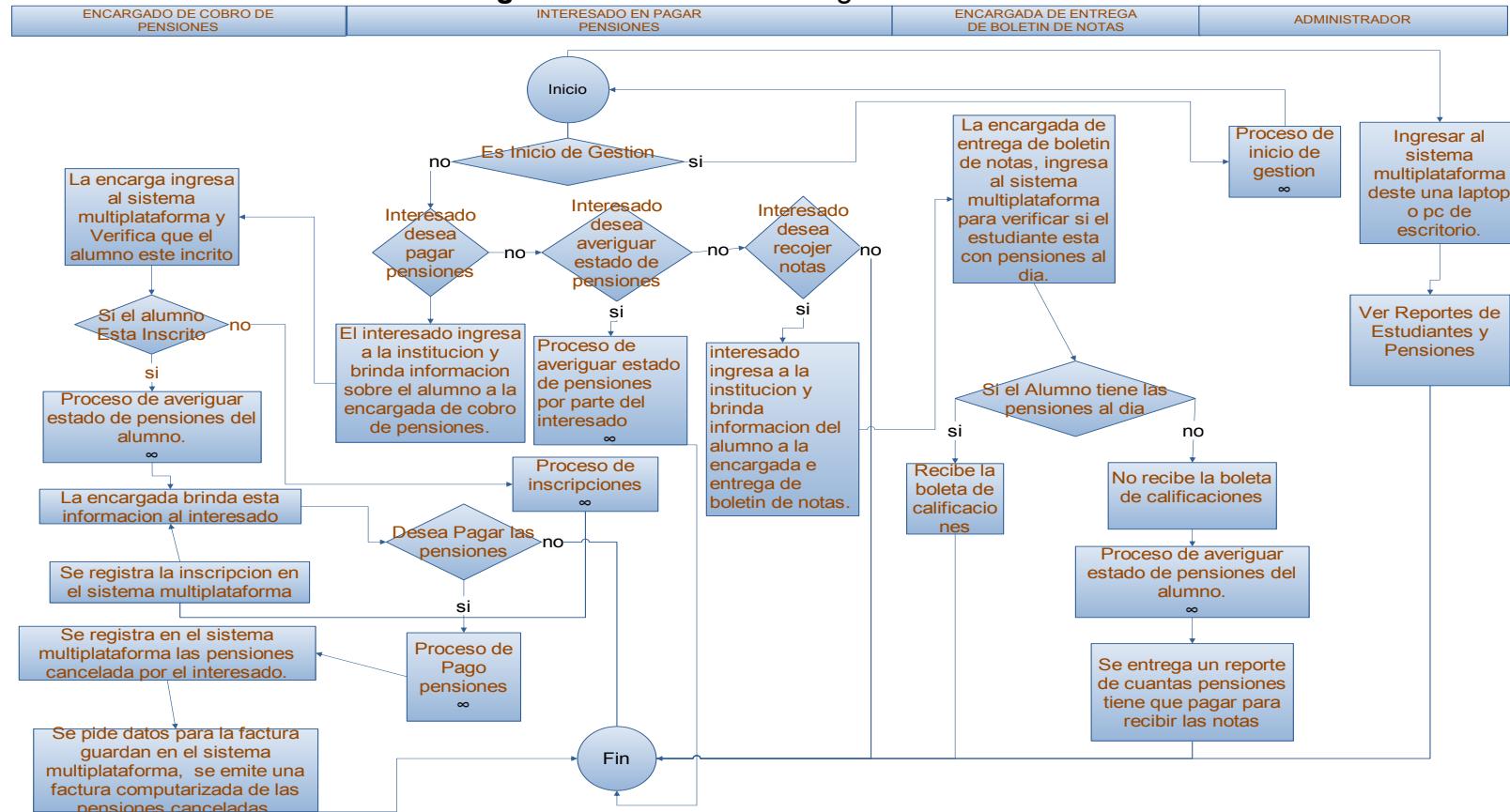
**Tabla 6:** Deficiencias existentes en el proceso actual.

TAREAS	DEFICIENCIAS/ PROBLEMAS
Proceso manual de llenado de facturas.	<ul style="list-style-type: none"><li>• Pérdida de tiempo en el registro manual.</li><li>• Datos repetidos cada vez que se factura.</li><li>• Largas filas.</li><li>• Errores en el llenado de facturas.</li></ul>
Entrega de notas por parte de la encargada entrega de las notas.	<ul style="list-style-type: none"><li>• Pérdida de tiempo en acceder a la información del interesado para verificar si tiene las pensiones al día.</li><li>• No contar con el acceso inmediato a la verificación del estado de pensiones de los alumnos.</li></ul>
Reportes.	<ul style="list-style-type: none"><li>• Pérdida de tiempo en la realización de reportes ya que los reportes que saca el sistema son muy escasos.</li></ul>
El averiguar el estado de pensiones por parte de Interesado.	<ul style="list-style-type: none"><li>• Largas filas.</li><li>• Presencia física del interesado en la institución.</li></ul>

**Fuente:** (Elaboración propia).

### 3.1.3. Modelado de Negocio Alternativo.

Figura 12: Modelado de negocio alternativo.

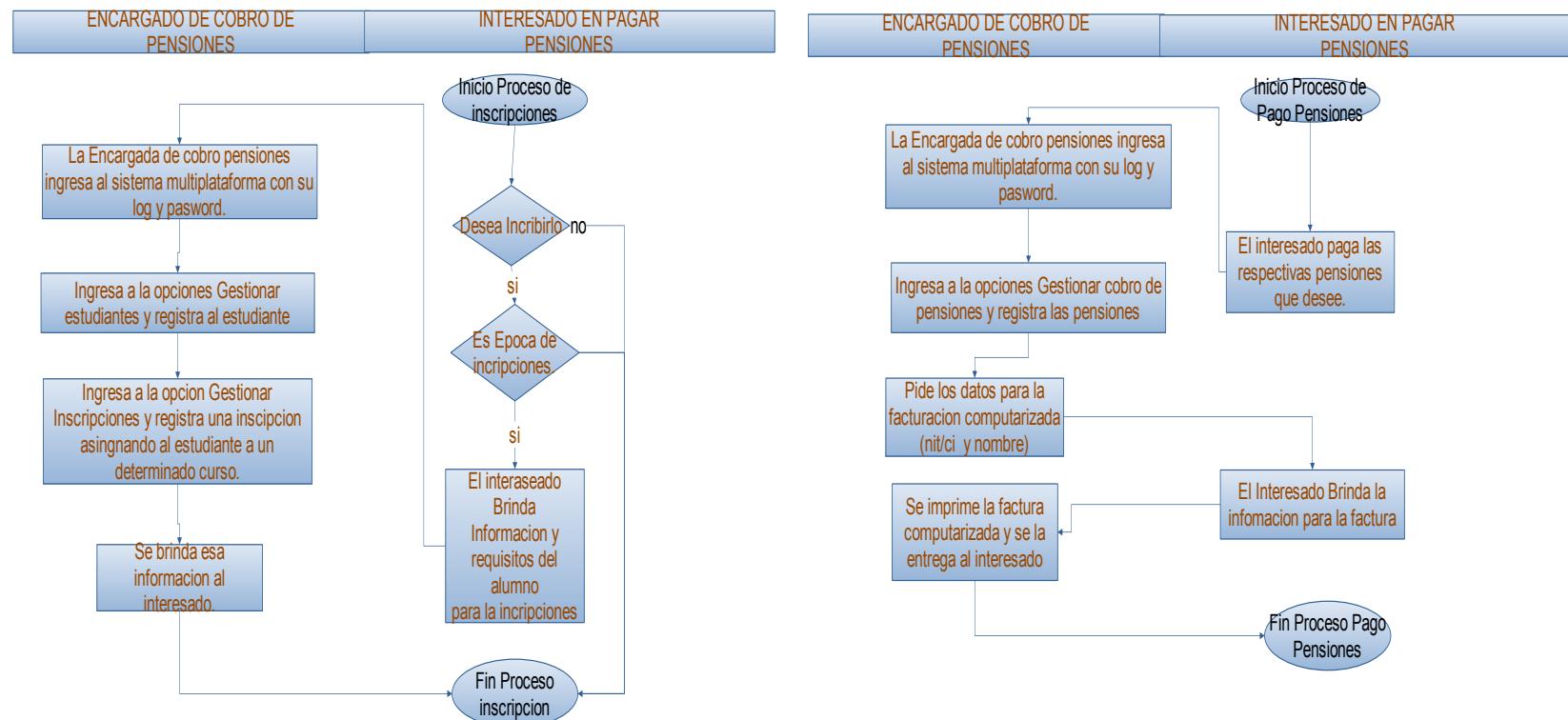


Fuente: (Elaboración propia).

### 3.1.3.1. Proceso de Inscripciones y Pago de Pensiones.

A continuación se muestra el modelado del proceso de inscripciones y pago de pensiones:

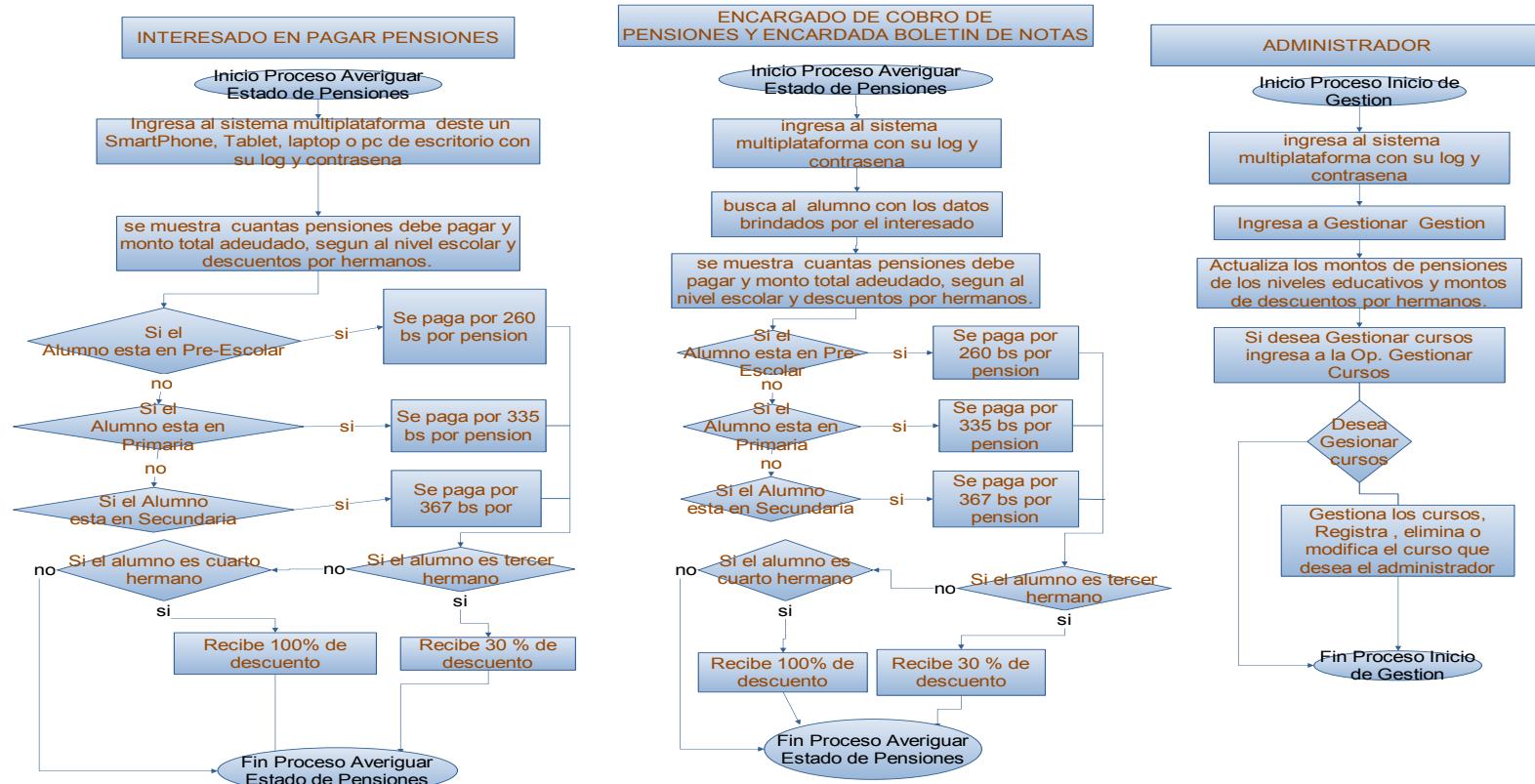
**Figura 13:** Proceso de inscripciones y pago de pensiones.



**Fuente:** (Elaboración propia).

### 3.1.3.2. Proceso de Averiguar Pensiones e Inicio de Gestión.

**Figura 14:** Proceso de averiguar pensiones e inicio de gestión.



Fuente: (Elaboración propia).

## **3.2. DESARROLLO DEL SUBSISTEMA MULTIPLATAFORMA DE GESTIÓN DE USUARIOS, NIVELES EDUCATIVOS CON MONTOS DE MENSUALIDADES POR GESTIÓN Y PLANIFICACIÓN DE CURSOS.**

### **3.2.1. Selección del modelo de desarrollo de software.**

Para realizar un análisis de las metodologías de desarrollo se realizará un cuadro comparativo de las ventajas y desventajas.

**Tabla 7:** Comparación entre modelos de desarrollo de software.

Modelo	Ventajas	Desventajas
Incremental	<ul style="list-style-type: none"><li>• Puede monitorearse el cambio de varios aspectos de un producto y apuntar a problemas potenciales y anomalías.</li><li>• Se centra en la entrega de un producto operacional con cada incremento.</li><li>• Es más fácil probar y depurar en una iteración más pequeña.</li><li>• Cada iteración es un hito gestionado fácilmente.</li><li>• Permite su desarrollo aunque no se disponga de todos los requerimientos.</li></ul>	<ul style="list-style-type: none"><li>• Infunde responsabilidad en el equipo de desarrollo al trabajar directamente con el cliente, requiriendo de profesionales sobre el promedio.</li><li>• Cada fase de una iteración es rígida y no se superponen con otras.</li><li>• Los primeros incrementos son versiones incompletas del producto final.</li><li>• Se maneja tiempos de entrega de los distintos módulos.</li></ul>

Modelo	Ventajas	Desventajas
Modelo Espiral	<ul style="list-style-type: none"> <li>• Como el software evoluciona y se realiza análisis de riesgos, a medida que progresá el desarrollador, el cliente comprende y reacciona mejor ante riesgos en cada uno de los niveles evolutivos.</li> <li>• Utiliza la construcción de prototipos como mecanismo de reducción de riesgos.</li> <li>• Mantiene el enfoque sistemático de los pasos sugeridos por el ciclo de vida clásico, pero lo incorpora al marco de trabajo iterativo que refleja de forma más realista el mundo real.</li> <li>• Demanda una consideración directa de análisis de riesgos técnicos en todas las etapas.</li> </ul>	<ul style="list-style-type: none"> <li>• Puede resultar difícil convencer a grandes clientes (particularmente en situaciones bajo contrato) de que el enfoque evolutivo es controlable.</li> <li>• Requiere una considerable habilidad para la evaluación del riesgo.</li> <li>• No se ha utilizado tanto como los paradigmas lineales secuenciales o de construcción de prototipos.</li> </ul>
Basado en componentes	<ul style="list-style-type: none"> <li>• <b>Reutilización del software.</b> lleva a alcanzar un mayor nivel de reutilización de software.</li> </ul>	<ul style="list-style-type: none"> <li>• Es necesario tener un manejo excelente de los conceptos, lo cual puede resultar difícil para quien no</li> </ul>

Modelo	Ventajas	Desventajas
Basado en componentes	<ul style="list-style-type: none"> <li>• <b>Simplifica las pruebas.</b> Permite que las pruebas sean ejecutadas probando cada uno de los componentes antes de probar el conjunto completo de componentes ensamblados.</li> <li>• <b>Simplifica el mantenimiento del sistema.</b> Cuando existe un débil acoplamiento entre componentes, el desarrollador es libre de actualizar y/o agregar componentes según sea necesario, sin afectar otras partes del sistema.</li> <li>• <b>Mayor calidad.</b> Dado que un componente puede ser construido y luego mejorado continuamente por un experto u organización, la calidad de una aplicación basada en componentes mejorará con el paso del tiempo.</li> </ul>	<p>tenga experiencia en el uso de procesos de ingeniería de software.</p> <ul style="list-style-type: none"> <li>• Los “compromisos” en los requisitos son inevitables, por lo cual puede que el software no cumpla las expectativas del cliente.</li> <li>• Las actualizaciones de los componentes adquiridos no están en manos de los desarrolladores del sistema.</li> </ul>

Fuente: (Elaboración propia)

El modelo en espiral contiene una etapa de análisis de riesgos que requiere de una habilidad y amplia experiencia por parte del desarrollador, sin embargo el autor de este proyecto no posee experiencia suficiente en esta área, también una desventaja importante es que el análisis de riesgos puede aumentar el costo del proyecto. Además es una desventaja es el hecho de que el modelo en espiral genera mucho tiempo en el desarrollo del sistema, pero el tiempo establecido por la EMI es reducido lo cual indica que este modelo no es aplicable en el presente proyecto.

El proceso basado en componentes no es un buen candidato para el presente proyecto debido a que el autor no ha desarrollado anteriormente sistemas similares y además exige tener experiencia para que sea bien aplicado.

Finalmente es necesario resaltar que el modelo incremental exige entrega de módulos, funcionando en cada incremento lo que permitirá obtener una retroalimentación oportuna del cliente (en caso de ser necesario) implementar mejoras conforme avanza el desarrollo de software. Es un hecho común que los usuarios al inicio del proyecto no puedan especificar de manera explícita todos los requerimientos funcionales, por lo cual el modelo incremental facilita al usuario la definición de requerimientos conforme avanza el proyecto. Por estas razones ya mencionadas, para el desarrollo se tomó la decisión de aplicar el modelo incremental.

### **3.2.2. Planificación de Incrementos del proyecto según a la metodología seleccionada.**

**Primer incremento.-** Desarrollar Subsistema multiplataforma de gestión de usuarios, niveles educativos con montos de mensualidades por gestión y planificación de cursos

**Segundo incremento.-** Diseño e implementación del módulo de inscripciones de estudiantes integrado al subsistema multiplataforma.

**Tercer incremento.-** Desarrollar los módulos de cobro de pensiones y reportes integrado al subsistema multiplataforma.

### 3.2.3. Selección de lenguajes de programación.

Para realizar un análisis de los lenguajes de programación se realizará un cuadro comparativo de las ventajas y desventajas.

**Tabla 8:** Comparación entre Lenguajes

Lenguaje de Programación	Ventajas	Desventajas
Java	<ul style="list-style-type: none"> <li>• Manejo es un lenguaje multiplataforma.</li> <li>• Es orientado a objetos.</li> <li>• Su sintaxis es similar a c++.</li> <li>• Fácil de aprender.</li> <li>• Es libre.</li> </ul>	<ul style="list-style-type: none"> <li>• Requiere un intérprete.</li> <li>• Una mala implementación de un programa java resulta lento.</li> </ul>
Php	<ul style="list-style-type: none"> <li>• Lenguaje multiplataforma.</li> <li>• Orientado al desarrollo de web dinámicas con acceso a información de base de datos.</li> <li>• Soporte con diferentes tipos de servidores de bases de datos, como por ejemplo MySQL.</li> <li>• Integración con librerías externas hace que la funcionalidad aumente bastante.</li> </ul>	<ul style="list-style-type: none"> <li>• La desventaja más grande es que al ser un lenguaje interpretado se necesita tener el intérprete de PHP, y esto trae al alto consumo de las</li> </ul>

Lenguaje de Programación	Ventajas	Desventajas
<b>Php</b>	<ul style="list-style-type: none"> <li>• Código invisible al usuario que lo solicita, programación segura y confiable.</li> <li>• Permite aplicar técnicas de programación orientada a objetos.</li> <li>• No requiere definición de tipos, sino que decide en periodo de ejecución.</li> <li>• Manejo de excepciones.</li> <li>• Uso libre.</li> </ul>	aplicaciones, ya que son imposibles de compilar.
<b>Asp.net</b>	<ul style="list-style-type: none"> <li>• Permite actualizaciones automáticas del Software.</li> <li>• Disponibilidad garantizada.</li> <li>• Permite probar software antes de comprarlo.</li> <li>• Se encarga de detectar el tipo de navegador utilizado por el cliente a la hora de realizar una petición.</li> </ul>	<ul style="list-style-type: none"> <li>• No es multiplataforma, tiene que correr en Pc normales que tengan Windows y un servidor Web.</li> </ul>

**Fuente:** (Elaboración Propia).

Asp descartado como lenguaje para el desarrollo del proyecto debido a que principalmente corre con la plataforma de Windows. También ha sido descartado java considerando que una inadecuada implementación del software se transformara en un programa lento, en cambio el lenguaje Php es un lenguaje multiplataforma y esta

específicamente orientado al desarrollo web, adicionalmente para implementar sistema web multiplataforma se aplicaran Html5, Java Script , Jquery y CSS3 cuyas definiciones ventajas y desventajas están descritas en el marco teórico.

### **3.2.4. Selección de framework.**

Para realizar un análisis de los Frameworks de desarrollo se realizará un cuadro comparativo de las ventajas y desventajas.

**Tabla 9:** Comparación entre Frameworks.

Frameworks	Ventajas	Desventajas
<b>CodeIgniter</b>	<ul style="list-style-type: none"> <li>• CodeIgniter es libre bajo licencias open source.</li> <li>• Es liviano y rápido ya que el núcleo del sistema sólo requiere algunas bibliotecas muy pequeñas y las bibliotecas adicionales se cargan dinámicamente bajo pedido del usuario basado en sus necesidades.</li> <li>• CodeIgniter usa el enfoque Modelo-Vista-Controlador, lo que permite una gran separación entre la lógica y la presentación. Es particularmente bueno para proyectos en los que los diseñadores trabajan en sus</li> </ul>	<ul style="list-style-type: none"> <li>• Es un framework que casi no necesita configuración.</li> <li>• Es un framework que no utiliza reglas de codificación restrictivas.</li> <li>• Tiene complejidad, pero también favorece a soluciones.</li> <li>• No tiene una documentación clara y completa.</li> </ul>

Frameworks	Ventajas	Desventajas
<b>CodeIgniter</b>	<p>archivos de plantillas, ya que el código en estos archivos será mínimo.</p> <ul style="list-style-type: none"> <li>• CodeIgniter tiene una gama completa de bibliotecas que facilitan las tareas de desarrollo web más comúnmente usadas, como acceso a base de datos, envío de correo electrónico, validación de datos de formularios, manejo de sesiones, manipulación de imágenes, etc.</li> </ul>	
<b>PhoneGap</b>	<ul style="list-style-type: none"> <li>• Es gratuito.</li> <li>• Es multiplataforma ya que corre dentro de un navegador web. Compatible con varias plataformas iOS, Android, Blackberry, Windows Phone, webOS, Symbian y Bada.</li> <li>• Es fácil de desarrollar y ofrece muchas posibilidades a aquellos que conocen bien los lenguajes basados en</li> </ul>	<ul style="list-style-type: none"> <li>• En algunos casos es necesario usar el sistema operativo de la plataforma. Por ejemplo empaquetar aplicaciones Windows Phone solo es posible con el sistema operativo Windows. Lo mismo ocurre con iOS, es necesario usar un Mac.</li> </ul>

Frameworks	Ventajas	Desventajas
<b>PhoneGap</b>	<p>tecnologías web (HTML, CSS y JavaScript).</p> <ul style="list-style-type: none"> <li>• Existe mucha documentación acerca de PhoneGap e incluso la propia web proporciona muchos ejemplos.</li> <li>• La alternativa que ofrece mediante el compilador en la nube es una forma muy sencilla que permite mejorar muchas de las desventajas de PhoneGap.</li> </ul>	<ul style="list-style-type: none"> <li>• Dependiendo de la plataforma se necesita un sistema diferente, para Android se requiere el uso de Eclipse y para iOS el uso de Xcode.</li> <li>• En caso de compilar para iOS es necesario introducir un código de desarrollador dado por la empresa Apple.</li> </ul>
<b>JQuery Mobile</b>	<ul style="list-style-type: none"> <li>• Compatible con la mayoría de las plataformas móviles como la mayoría de los navegadores: iOS, Android, Blackberry, WebOS, Symbian, Windows Phone 7.</li> <li>• Construido sobre el núcleo jQuery, para que la curva de aprendizaje sea mínima para desarrolladores familiarizados con la sintaxis de jQuery.</li> </ul>	<ul style="list-style-type: none"> <li>• Las funciones que ofrece son muchas, pero resultan difíciles de personalizar. Su aspecto visual es estandarizado y no se integra con el de la plataforma. En algunos casos, no queda otra opción que usar JavaScript simple para adaptar la aplicación a nuestras necesidades.</li> </ul>

Frameworks	Ventajas	Desventajas
<b>JQuery Mobile</b>	<ul style="list-style-type: none"> <li>• Un framework de temas que permite personalizar uno rápidamente.</li> <li>• Dependencias limitadas y ligero que optimizan su velocidad.</li> <li>• El mismo código de base escalará el tamaño automáticamente a cualquier pantalla.</li> <li>• Posibilidad de navegación entre páginas por AJAX con animaciones en las transiciones de página</li> <li>• UI widgets optimizados para táctil y para quien desconozca la plataforma.</li> </ul>	<ul style="list-style-type: none"> <li>• Como es necesario invocar a un archivo para utilizar sus funciones, ralentiza levemente la carga de la página.</li> <li>• Su manejo de CSS suele resultar innecesariamente complejo. A veces cuesta saber qué clases utilizar.</li> <li>• No existen muchas plantillas prediseñadas sobre las cuales empezar a construir nuestra aplicación.</li> </ul>

**Fuente:** (Elaboración propia)

Se seleccionó Codeigniter por ser un framework flexible ya que se puede adaptar fácilmente a aplicaciones web multiplataforma por su considerada adaptabilidad al uso de CSS3 y Php en el proyecto, el framework jQuery Movile no es un buen candidato por su manejo complejo de hojas de estilo, adicionalmente la principal desventaja de Phone Gap es la necesidad de compilar para cada sistema operativo lo cual también se rechazó como opción de desarrollo del presente proyecto.

### 3.2.5. Selección del Gestor de Bases de datos.

Para realizar un análisis de los gestores de bases de datos se realizará un cuadro comparativo de las ventajas y desventajas.

**Tabla 10:** Comparación entre gestores de base de datos.

Gestor de base de datos	Ventajas	Desventajas
MySql	<ul style="list-style-type: none"> <li>• Mysql es Open Source: significa que es posible usar y modificar el software gratuitamente.</li> <li>• Trabaja en entornos cliente servidor o incrustados.</li> <li>• Posee control de acceso de usuarios y seguridad en los datos.</li> <li>• Integra perfectamente el lenguaje PHP.</li> <li>• Soporte complejo para cláusulas, funciones, tipos de datos y comandos estándar.</li> <li>• Soporte para control de transacciones en tablas transaccionales.</li> <li>• Gran portabilidad entre los diferentes sistemas y plataformas.</li> </ul>	<ul style="list-style-type: none"> <li>• Un gran porcentaje de las utilidades de MySQL no están documentadas en español.</li> </ul>

<b>Gestor de base de datos</b>	<b>Ventajas</b>	<b>Desventajas</b>
<b>SQL Server 2012</b>	<ul style="list-style-type: none"> <li>• Utiliza una extensión al SQL estándar, que se denomina Transact SQL.</li> <li>• SQL Server nos permite pasar desapercibido los ficheros que forman la base de datos ya que al simplemente al abrir la consola de SQL Server tenemos todas las bases de datos de todos nuestros programas.</li> <li>• Si trabajamos en una red local permite agregar otros servidores de SQL Server, una base de datos en servidor local SQL y otra base de datos, se pueden conectar los SQL entre si y acceder a las bases de datos.</li> <li>• SQL permite administrar permisos a nivel de servidor, seguridad en tablas, permitir o no lectura, escritura, ejecución seguridad en los procedimientos almacenados, también tiene la</li> </ul>	<ul style="list-style-type: none"> <li>• la enorme cantidad de memoria RAM que utiliza para la instalación y utilización del software.</li> <li>• SQL versión gratuita prohíbe muchas opciones y tiene restricciones a diferencia sobre MySQL.</li> <li>• La relación calidad-precio está muy debajo comparado con MySQL.</li> </ul>

<b>Gestor de base de datos</b>	<b>Ventajas</b>	<b>Desventajas</b>
	opción de permitir que una tercera persona se conecte a la SQL, se le puede dar permisos solo a ciertas bases de datos.	
<b>PostgresSQL</b>	<ul style="list-style-type: none"> <li>• Estabilidad y confiabilidad: No se han presentado caídas de la base de datos.</li> <li>• Soporta los tipos de datos, cláusulas, funciones y comandos de tipo estándar SQL y propios de PostgreSQL.</li> <li>• Puede operar sobre distintas plataformas, incluyendo Linux, Windows, Unix, Solaris y MacOS X.</li> <li>• Buen sistema de seguridad mediante la gestión de usuarios, grupos de usuarios y contraseñas.</li> <li>• Gran capacidad de almacenamiento.</li> </ul>	<ul style="list-style-type: none"> <li>• En comparación con <i>MySQL</i> es más lento en inserciones y actualizaciones, ya que cuenta con cabeceras de intersección que no tiene <i>MySQL</i>.</li> <li>• Soporte en línea: Hay foros oficiales, pero no hay una ayuda obligatoria.</li> <li>• Consume más recursos que <i>MySQL</i></li> </ul>

**Fuente:** (Elaboración propia).

Se seleccionó MySQL por su gran compatibilidad con el lenguaje PHP y el framework CodeIgniter, también cabe decir que se descartó SQL Server 2012 por su bloqueo de funciones al usarlo, es decir que la versión gratuita de SQL 2012 restringe varias funciones, por otro lado Postgres fue descartado por su escasa documentación y por saturación, lentitud al hacer peticiones a las bases de datos.

### **3.2.6. Análisis**

#### **3.2.6.1. Requerimientos del sistema**

##### **Requerimientos Funcionales**

- El sistema debe permitir registrar información nueva de usuarios y guardarlas en la base de datos.
- El sistema debe permitir buscar usuarios para modificarlos o eliminarlos de la base de datos.
- El sistema debe permitir registrar la información de nuevas gestiones educativas y guardarlas en la base de datos.
- El sistema debe permitir buscar gestiones educativas para poder modificarlas o eliminarlas de la base de datos.
- El sistema debe permitir registrar la información de nuevos cursos asignados a una gestión educativa y guardarlos en la base de datos.
- El sistema debe permitir buscar cursos para poder modificarlos o eliminarlos de la base de datos.

##### **Requerimientos no funcionales**

- **Compatibilidad:** El sistema debe visualizarse correctamente en cualquier plataforma y dispositivo, que sea capaz de ejecutar un navegador web.
- **Disponibilidad:** El sistema debe estar disponible para su uso las 24 horas para que cualquier usuario pueda ingresar.

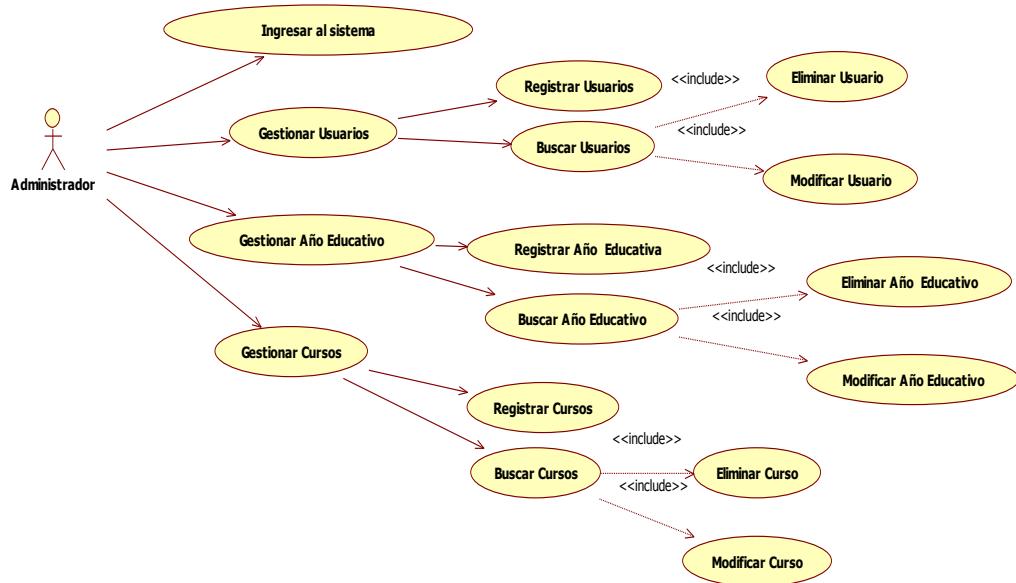
- **Extensibilidad:** el sistema debe ser capaz de integrarse con facilidad a diferentes sistemas.
- **Mantenibilidad:** El sistema debe estar listo para agregar nuevos módulos debido a que se usa el modelo incremental y arquitectura mvc.
- **Seguridad:** El sistema ser accedido con un id y contraseña, por lo que para poder acceder al sistema los usuarios deben estar previamente registrados.
- **Usabilidad:** El sistema debe estar diseñado de manera amigable, para que cualquier persona con poca experiencia con las computadoras pueda usarlas.

### 3.2.6.2. Identificación de Actores

El administrador: es el encargado de realizar cada inicio de año educativo la gestión de usuarios, gestión educativa con montos de mensualidades, descuentos por segundo, tercero y cuarto hermano, además la planificación de cursos disponibles para la respectiva gestión.

### 3.2.6.3. Diagramas de casos de uso por actor

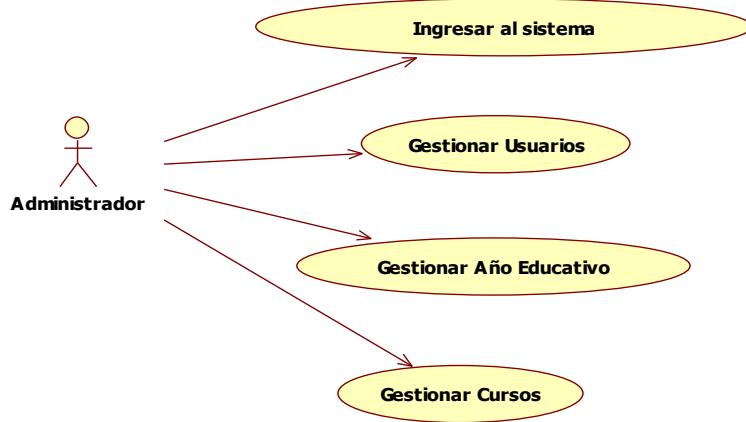
**Figura 15:** Casos de uso por actor.



**Fuente:** Elaboración Propia

### 3.2.6.4. Diagrama de Casos de Uso del sistema

**Figura 16:** Casos de uso general del sistema



**Fuente:** Elaboración Propia

### 3.2.6.5. Descripción de cada caso de uso

**Tabla 11:** Descripción de caso de uso ingresar al sistema

CASO DE USO	INGRESAR AL SISTEMA	
ACTOR	Administrador, Secretaria, Cajero.	
SECUENCIA NORMAL	PASO	ACCIÓN
	1	El usuario ingresa a la página web e ingresa a la opción ingresar al sistema.
	2	El sistema direcciona a la interfaz de autenticación de usuarios,
	3	El usuario ingresa su ci y contraseña e ingresar la opción ingresar.
	4	El sistema verifica que los datos sean correctos y direcciona a al menú principal.
Post condición	El usuario debe estar previamente registrado en el sistema para poder acceder a las funcionalidades del sistema de acuerdo al privilegio asignado.	
Excepciones	PASO	ACCIÓN

	<b>1</b>	Si los campos de ingresar al sistema usuarios están vacíos, no deja presionar el botón ingresar hasta llenar estos campos.
	<b>2</b>	Si los campos de ingresar al sistema usuarios son incorrectos, no existen, o no coinciden manda el mensaje de: "ci o contraseña incorrectos".
<b>Comentarios</b>	El ingresar al sistema lo realizan todos los tipos de usuarios.	

**Fuente:** Elaboración propia.

**Tabla 12:** Descripción de caso de uso gestionar usuarios.

CASO DE USO	GESTIONAR USUARIOS	
ACTOR	Administrador.	
SECUENCIA NORMAL	PASO	ACCIÓN
	<b>1</b>	EL Sistema direcciona al menú “Gestionar Usuarios” en el cual se encuentran las opciones de registrar usuarios y buscar usuarios.
	<b>2</b>	El usuario ingresa a la opción registrar usuarios y registra un usuario nuevo.
	<b>3</b>	El usuario ingresa a la opción buscar usuarios y se muestra todos los usuarios que están registrado, además de las opciones de modificar y eliminar.
	<b>4</b>	El usuario puede ingresar un ci de algún usuario y buscar a un usuario en particular.
<b>Post condición</b>	El usuario no puede registrar dos usuarios con la misma cedula de identidad.	
Excepciones	PASO	ACCIÓN
	<b>1</b>	Si los campos de registrar o modificar usuarios están vacíos no deja presionar el botón registrar hasta llenar estos campos.
	<b>2</b>	Si los campos de registrar o modificar usuarios son incorrectos, el sistema devuelve un mensaje de: “Error”.

<b>Comentarios</b>	En el registro de usuarios se asignan los privilegios que tiene un respectivo usuario ya sea administrador, cajero o secretaria.
--------------------	--

**Fuente:** Elaboración propia.

**Tabla 13:** Descripción de caso de uso año educativo.

<b>CASO DE USO</b>	<b>GESTIONAR AÑO EDUCATIVO</b>	
<b>ACTOR</b>	Administrador.	
<b>SECUENCIA NORMAL</b>	<b>PASO</b>	<b>ACCIÓN</b>
	<b>1</b>	EL Sistema direcciona al menú “Gestionar Año Educativo” en el cual se encuentran las opciones de registrar gestión y buscar gestión.
	<b>2</b>	El usuario ingresa a la opción registrar gestión y registra una nueva gestión.
	<b>3</b>	El usuario ingresa a la opción buscar gestiones y se muestra todos las gestiones que se registraron, además de las opciones de modificar y eliminar.
	<b>4</b>	El usuario puede ingresar a la opción de buscar por año educativo y buscar a una gestión en particular.
<b>Post condición</b>	El usuario no puede registrar dos años educativos con el mismo año.	
<b>Excepciones</b>	<b>PASO</b>	<b>ACCIÓN</b>
	<b>1</b>	Si los campos de registrar o modificar gestiones están vacíos no deja presionar el botón registrar hasta llenar estos campos.
	<b>2</b>	Si los campos de registrar o modificar gestiones son incorrectos, el sistema devuelve un mensaje de: “Error”.
<b>Comentarios</b>	El registro de año educativo es en el cual se registran los montos de mensualidad según el nivel educativo.	

**Fuente:** Elaboración propia.

**Tabla 14:** Descripción de caso de uso gestionar cursos.

CASO DE USO	GESTIONAR CURSOS	
ACTOR	Administrador.	
SECUENCIA NORMAL	PASO	ACCIÓN
	1	EL Sistema direcciona al menú “Gestionar Cursos” en el cual se encuentran las opciones de registrar cursos y buscar cursos.
	2	El usuario ingresa a la opción registrar cursos y registra un curso nuevo.
	3	El usuario ingresa a la opción buscar cursos y se muestra todos los cursos que están registrado, además de las opciones modificar y eliminar.
	4	El usuario puede ingresar en buscar cursos por año educativo y se muestran los cursos pertenecientes al respectivo año educativo.
Post condición	El usuario deberá haber registrado por lo menos un año educativo antes de registrar un curso nuevo.	
Excepciones	PASO	ACCIÓN
	1	Si los campos de registrar o modificar cursos están vacíos no deja presionar al botón registrar hasta llevar estos campos.
	2	Si los campos de registrar o modificar cursos son incorrectos, el sistema manda un mensaje de: “Error”.
Comentarios	El usuario no puede registrar dos cursos con los mismos atributos en el mismo año educativo.	

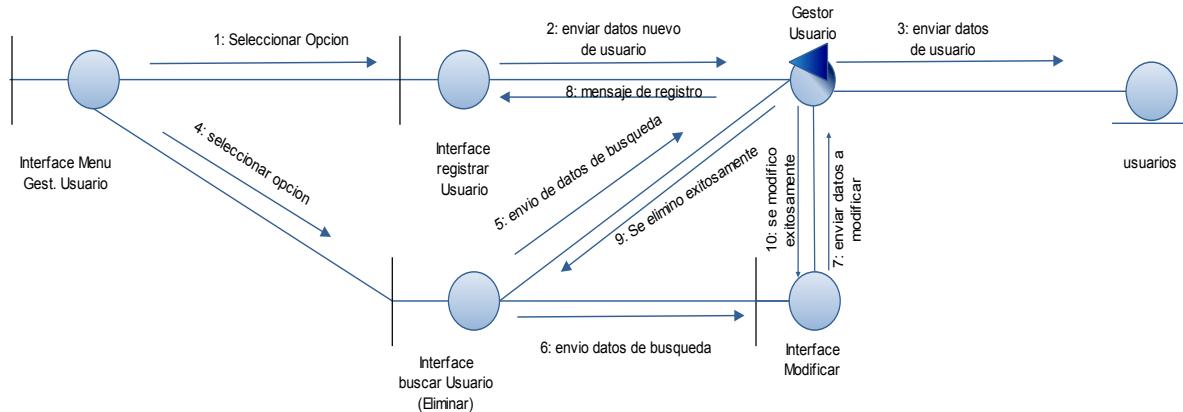
**Fuente:** Elaboración propia.

### 3.2.7. Diseño

A continuación se presentan los diagramas de colaboración para cada caso de uso identificado.

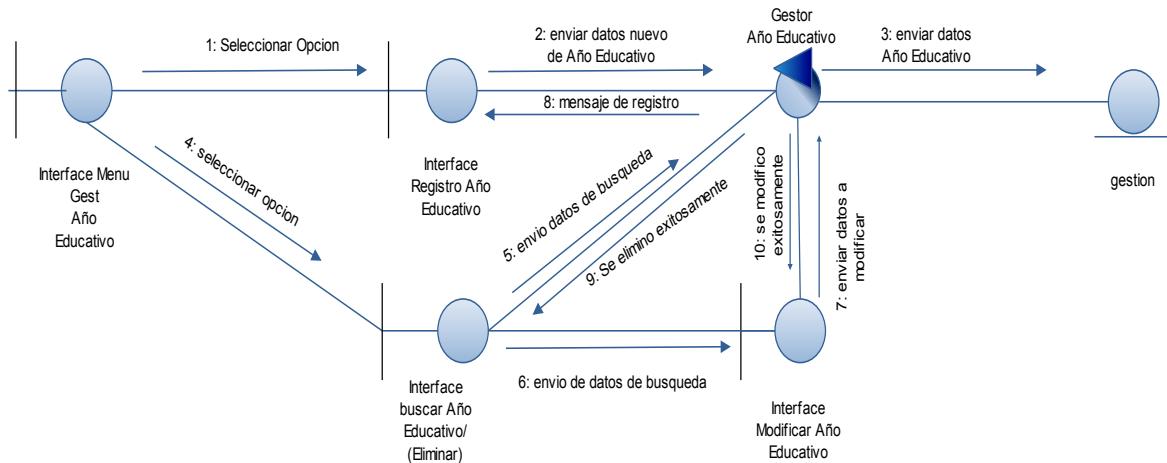
### 3.2.7.1. Diagramas de Colaboración

**Figura 17:** Diagrama de colaboración usuarios



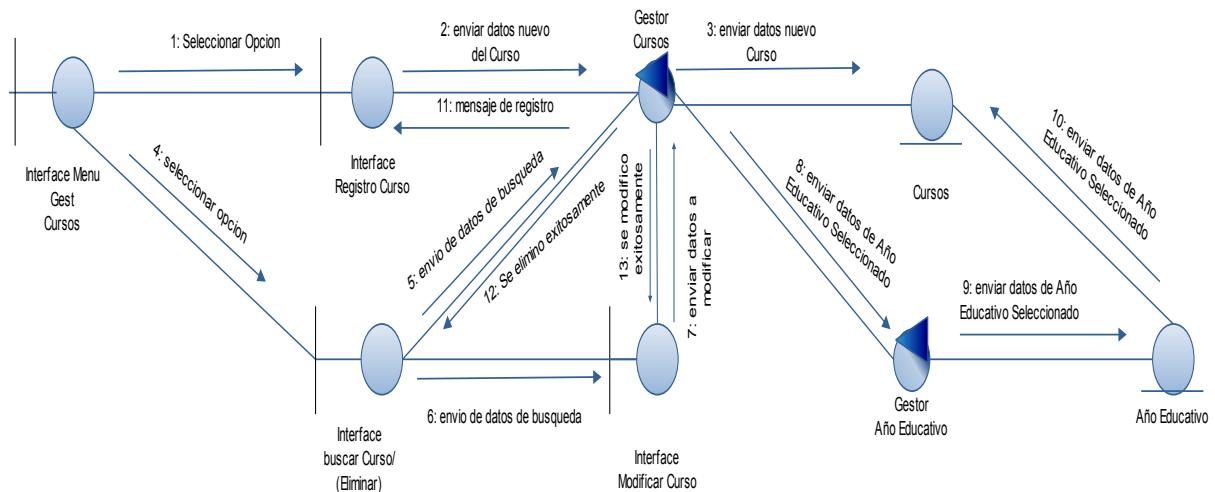
**Fuente:** Elaboración Propia.

**Figura 18:** Diagrama de colaboración año educativo.



**Fuente:** Elaboración Propia

**Figura 19:** Diagrama de colaboración cursos

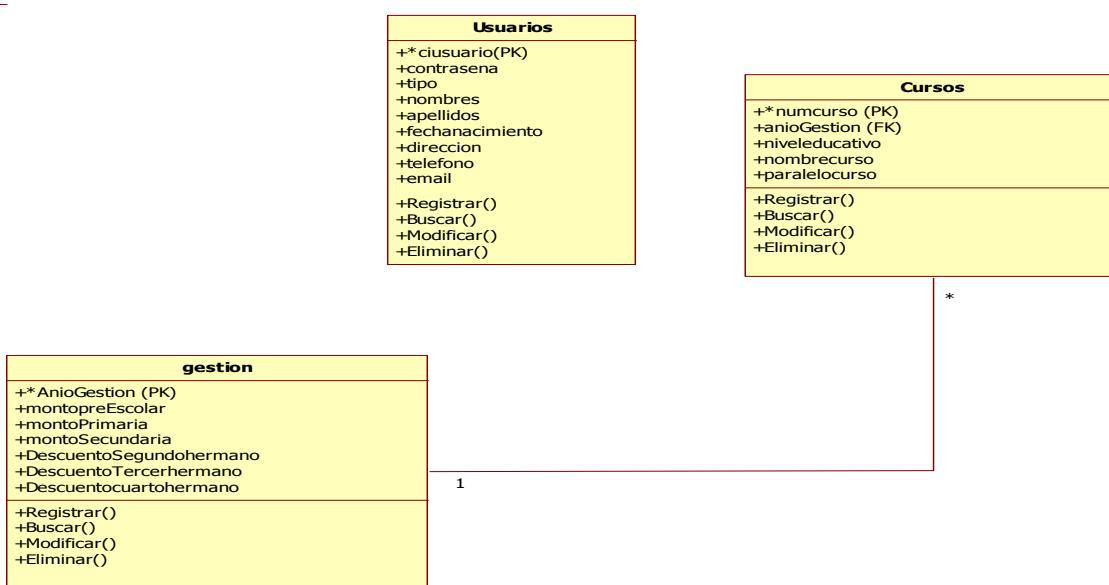


**Fuente:** Elaboración Propia

En base a los diagramas de colaboración ya mostrados anteriormente se ha podido elaborar los diagramas de clases y bases de datos los que se muestran a continuación:

### 3.2.7.2. Diagrama de clases

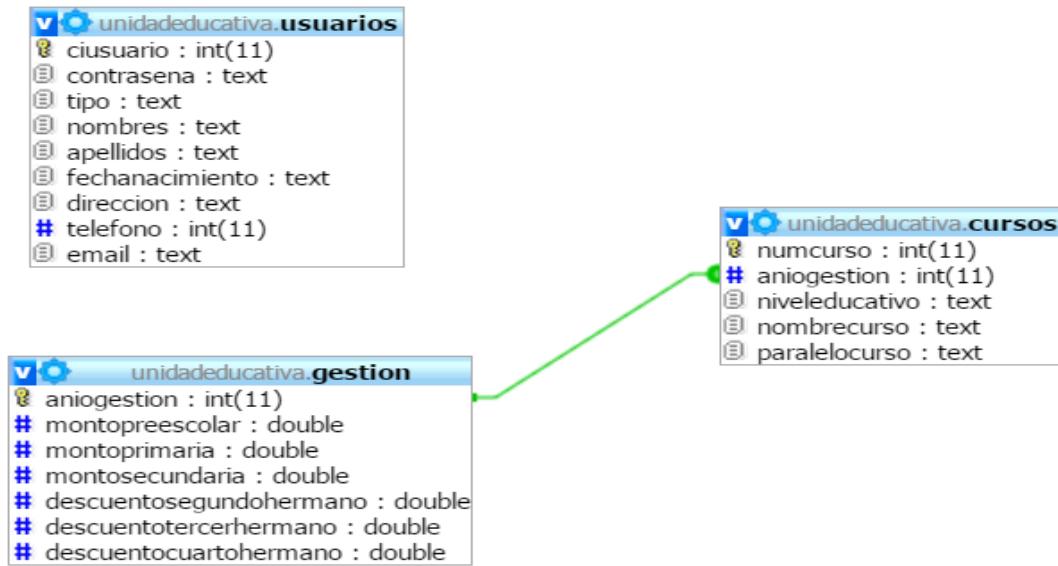
**Figura 20:** Diagrama de clases



**Fuente:** Elaboración Propia

### 3.2.7.3. Diagrama de base de datos

Figura 21: Diagrama de base de datos



Fuente: Elaboración Propia

### 3.2.7.4. Diccionario de datos

A continuación se muestra el diccionario de datos de la tabla usuarios, gestión y cursos:

Tabla 15: Diccionario de datos tabla usuarios

Columna	Tipo de dato	Null	Llave primaria	Llave foránea	Descripción
ciusuario	Int	No	Si	No	Llave primaria de la tabla usuarios.
contrasena	Text	No	No	No	Registra la contraseña del usuario.

tipo	Text	No	No	No	Registra el tipo de usuario (administrador, secretaria o cajero).
nombres	Text	No	No	No	Registra los nombres del usuario.
apellidos	Text	No	No	No	Registra los apellidos del usuario.
fechanacimiento	Text	No	No	No	Registra la fecha de nacimiento del usuario.
dirección	Text	No	No	No	Registra la dirección donde reside el usuario.
teléfono	Int	No	No	No	Registra el teléfono o celular del usuario.
email	Text	No	No	No	Registra el correo electrónico del usuario.

**Fuente:** Elaboración propia.

A continuación en la siguiente tabla se muestra el diccionario de datos de la tabla gestión:

**Tabla 16:** Diccionario de datos tabla gestión

Columna	Tipo de dato	Null	Llave primaria	Llave foránea	Descripción
anioeducativo	Int	No	Si	No	Llave primaria de la tabla gestion.
montopreescolar	Double	No	No	No	Registra el monto en bolivianos de la pensión de preescolar.
montoprimaria	Double	No	No	No	Registra el monto en bolivianos de la pensión de primaria.
montosecundaria	Double	No	No	No	Registra el monto en bolivianos de la pensión de secundaria.
descuentosegundohermano	Double	No	No	No	Registra el porcentaje del descuento de segundo hermano.
descuentotercerhermano	Double	No	No	No	Registra el porcentaje del descuento del tercer hermano.
descuentocuartohermano	Double	No	No	No	Registra el porcentaje de descuento del cuarto hermano.

**Fuente:** Elaboración propia.

**Tabla 17:** Diccionario de datos tabla cursos

Columna	Tipo de dato	Null	Llave primaria	Llave foránea	Descripción
numcurso	Int	No	Si	No	Llave primaria de la tabla cursos.
aniogestion	Int	No	No	Si	Registra el año de gestión en la cual está siendo asignado el curso.
niveleducativo	Text	No	No	No	Registra el nivel educativo en la que se encuentra el curso.
nombrecurso	Text	No	No	No	Registra el nombre del curso.
paralelocurso	Text	No	No	No	Registra el paralelo del curso.

**Fuente:** Elaboración propia.

### 3.2.8. Implementación

#### 3.2.8.1. Arquitectura del Software

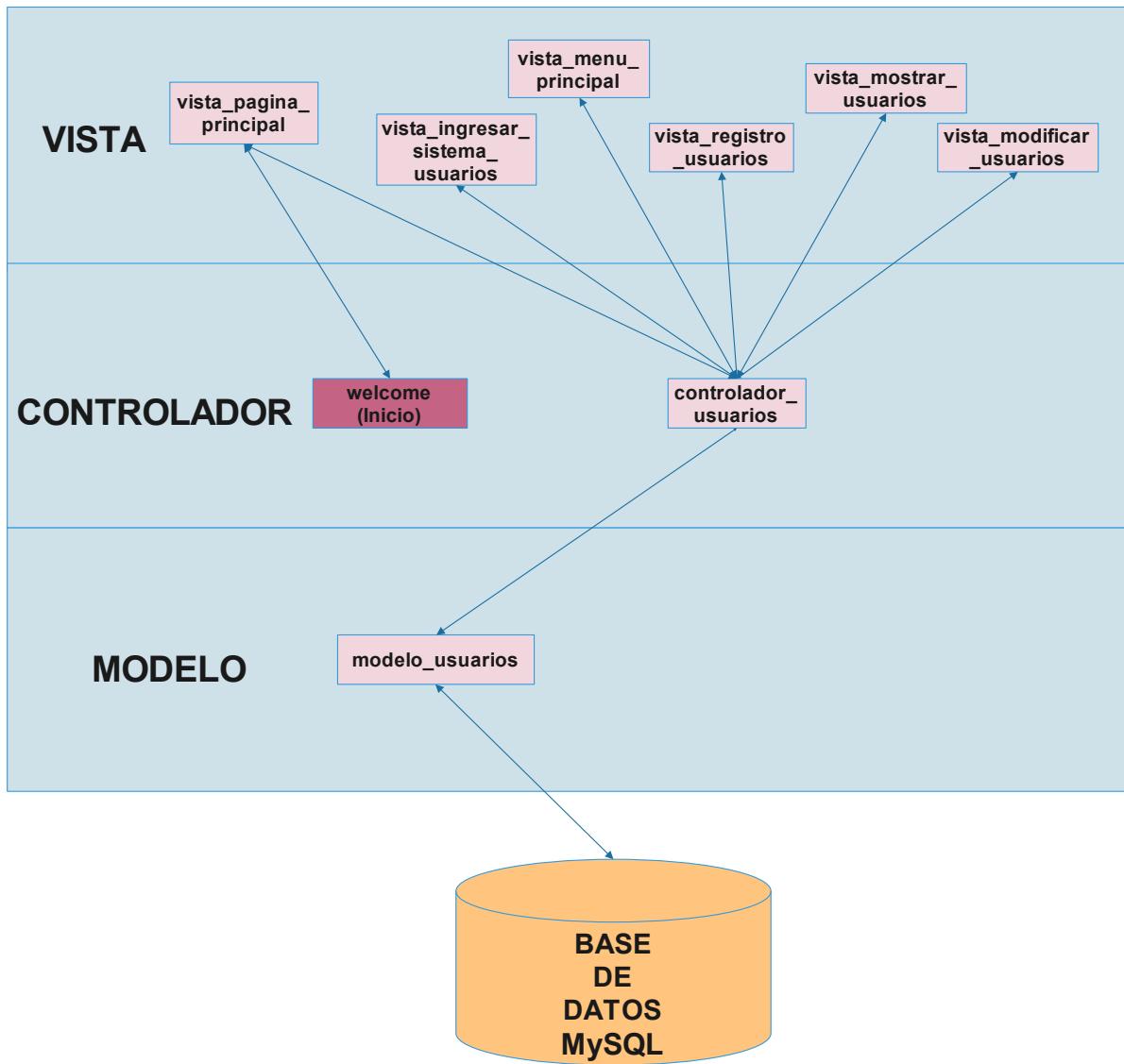
Para la realización la arquitectura de software del sistema cabe decir que se utilizó una figura de las conexiones y relaciones entre los diferentes archivos del sistema ya sea controlador, vista o modelo, conectada con una flecha para indicar que el archivo tiene relación directa con otra.

Como el framework de codeigniter utiliza la arquitectura modelo-vista-controlador se optó por describir las conexiones mostrando una clara diferencia entre los archivos destinados a las vistas, así como también los archivos destinados a controladores y a modelos.

Seguidamente se realizará las arquitecturas de los componentes del sistema para llevar a cabo la gestión de usuarios, año educativo y gestión de cursos.

A continuación se muestra la arquitectura de los componentes del sistema para llevar a cabo la gestión de usuarios.

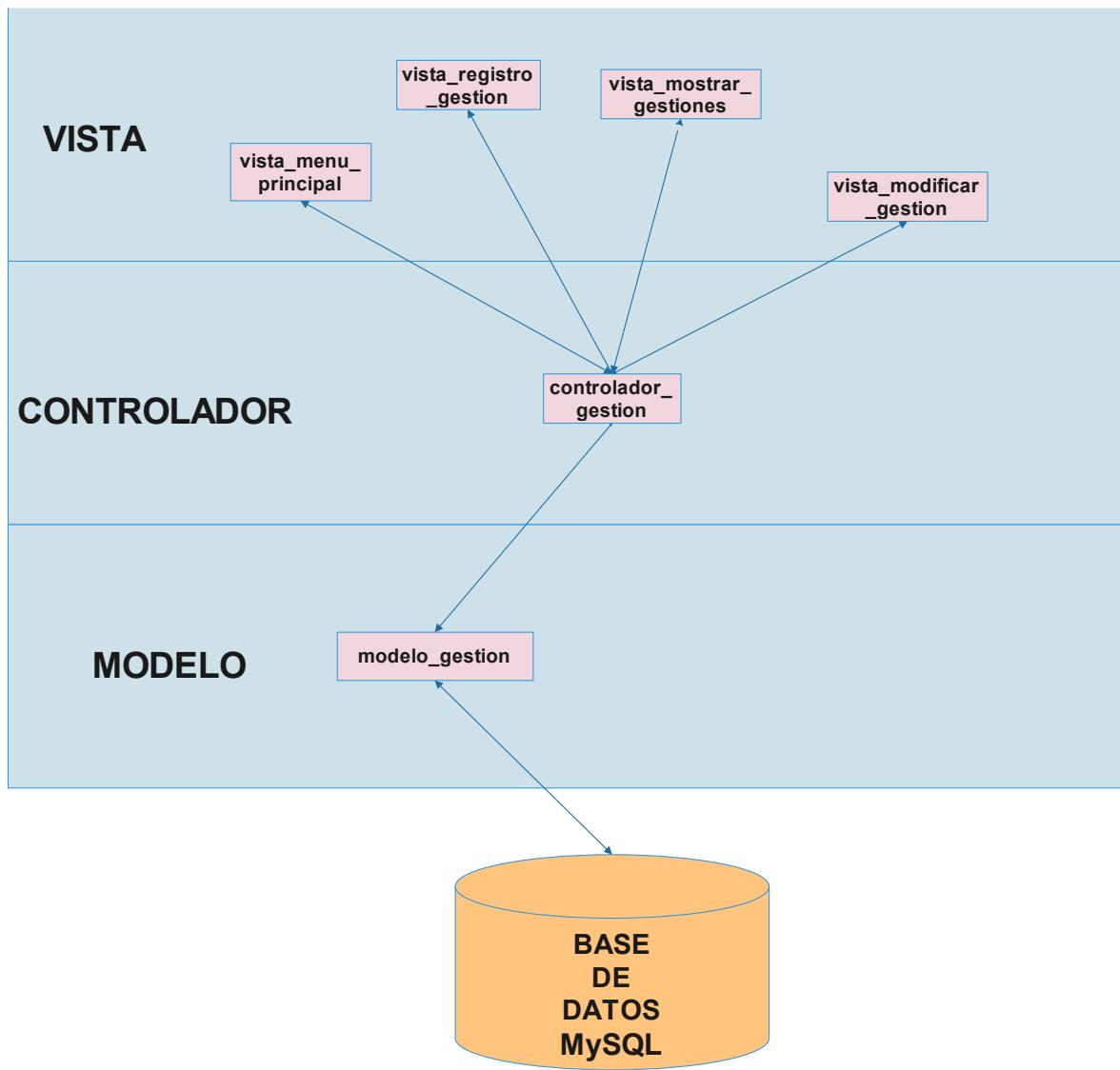
**Figura 22:** Arquitectura para el caso de uso gestionar usuarios.



**Fuente:** Elaboración propia.

Seguidamente se muestra la arquitectura de los componentes del sistema para llevar a cabo la gestión de año educativo.

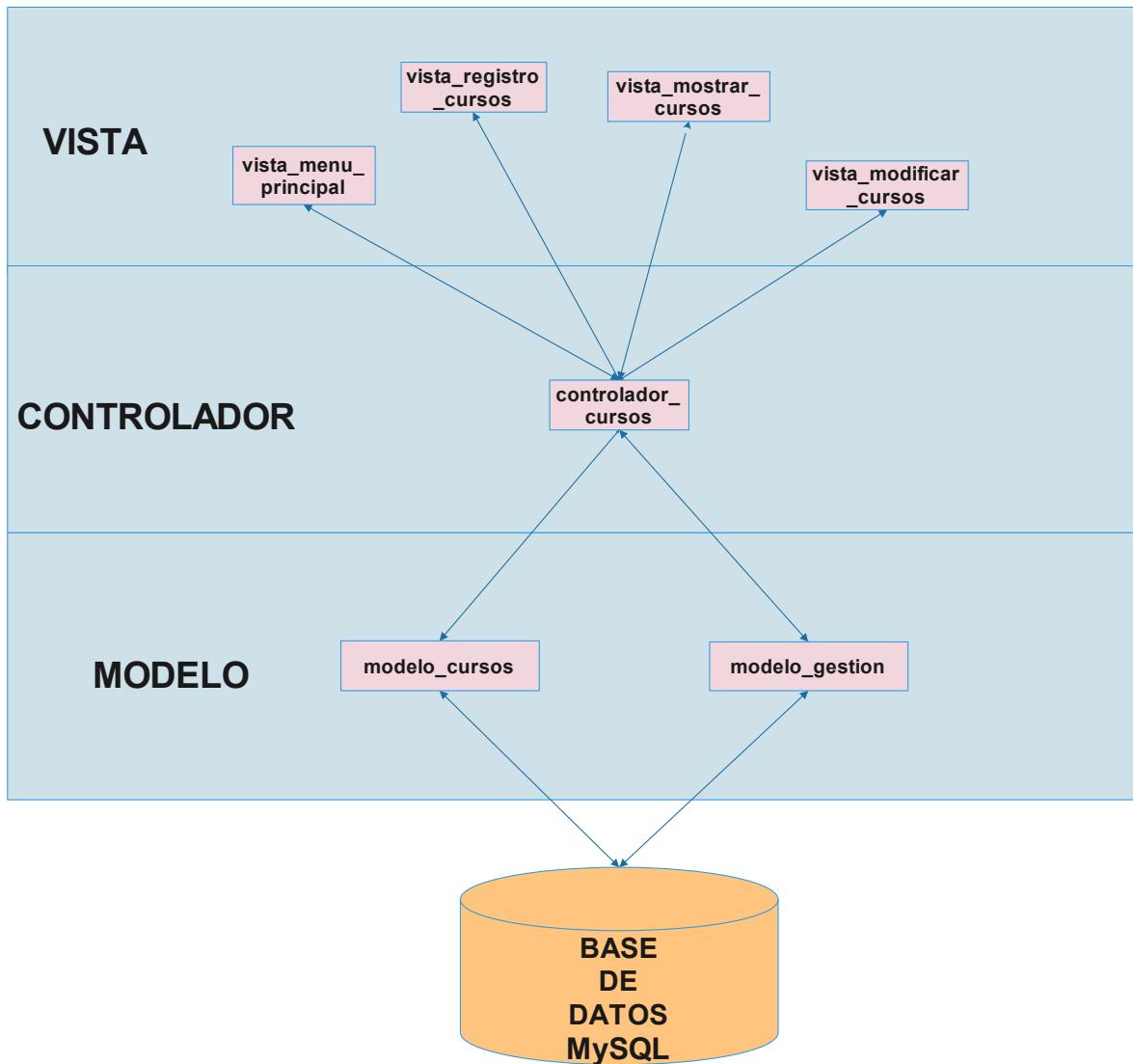
**Figura 23:** Arquitectura para el caso de uso gestionar año educativo.



**Fuente:** Elaboración propia.

A continuación se muestra la arquitectura de los componentes del sistema para llevar a cabo la gestión de cursos.

**Figura 24:** Arquitectura para el caso de uso gestionar cursos.

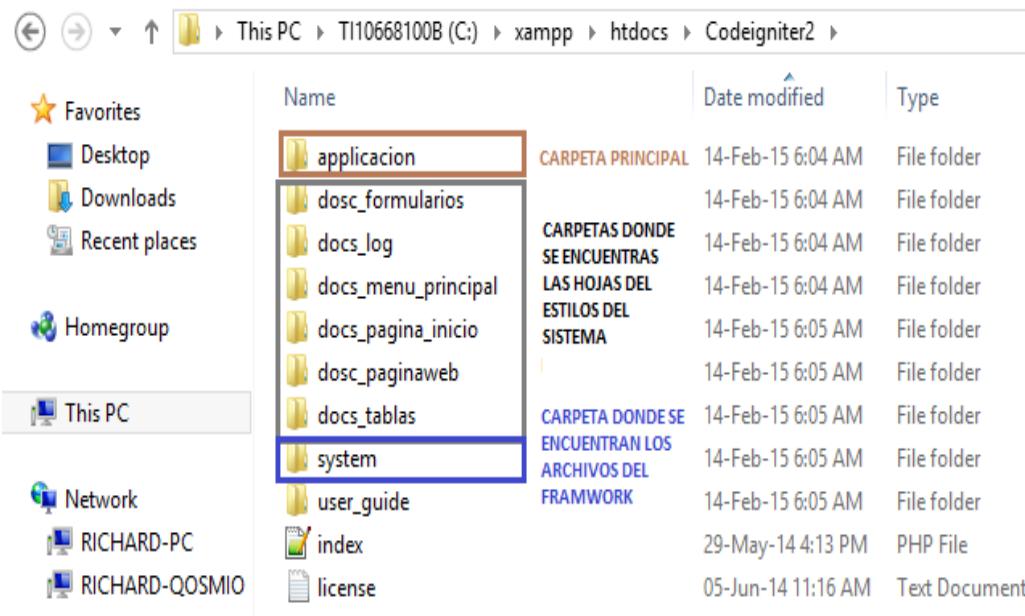


**Fuente:** Elaboración propia.

### 3.2.8.2. Organización del proyecto

A continuación se muestran las carpetas del proyecto por niveles:

**Figura 25:** Primer nivel del proyecto.



The screenshot shows a Windows File Explorer window displaying the contents of the 'Codeigniter2' folder located at 'C:\xampp\htdocs\Codeigniter2'. The left sidebar shows 'Favorites' with icons for Desktop, Downloads, Recent places, Homegroup, This PC, Network, RICHARD-PC, and RICHARD-QOSMIO. The main pane lists the following files and folders:

Name	Date modified	Type
aplicacion	CARPETA PRINCIPAL	File folder
dosc_formularios	14-Feb-15 6:04 AM	File folder
docs_log	14-Feb-15 6:04 AM	File folder
docs_menu_principal	14-Feb-15 6:04 AM	File folder
docs_pagina_inicio	14-Feb-15 6:05 AM	File folder
dosc_paginaweb	14-Feb-15 6:05 AM	File folder
docs_tablas	CARPETA DONDE SE ENCUENTRAN LOS ARCHIVOS DEL FRAMEWORK	File folder
system	14-Feb-15 6:05 AM	File folder
user_guide	14-Feb-15 6:05 AM	File folder
index	29-May-14 4:13 PM	PHP File
license	05-Jun-14 11:16 AM	Text Document

**Fuente:** Elaboración propia.

Cabe mencionar que las carpetas docs\_formularios, docs\_log, docs\_menu\_principal, docs\_pagina\_inicio, docs\_pagina\_web y docs\_tablas contienen las hojas de estilos del proyecto.

Por otro lado en la carpeta system se encuentran las librerías que vienen por defecto con el Framework Codeigniter. En la carpeta user\_guide se encuentra la guía de usuario de codeigniter que tambien viene por defecto con el framework. Tambien es necesario mencionar que en la carpeta application se encuentran las vistas, los controladores y los modelos del proyecto.

Seguidamente se muestra el contenido de la carpeta "aplicación":

**Figura 26:** Contenido de la carpeta application.

Name	Date modified	Type
cache	14-Feb-15 4:58 AM	File folder
config	07-Feb-15 1:55 PM	File folder
controllers	14-Feb-15 5:19 AM	File folder
core	07-Feb-15 1:55 PM	File folder
errors	07-Feb-15 1:55 PM	File folder
helpers	14-Feb-15 4:34 AM	File folder
hooks	07-Feb-15 1:55 PM	File folder
language	07-Feb-15 1:55 PM	File folder
libraries	09-Feb-15 7:13 PM	File folder
logs	09-Feb-15 6:25 PM	File folder
models	09-Feb-15 7:18 PM	File folder
third_party	07-Feb-15 1:55 PM	File folder
views	14-Feb-15 5:14 AM	File folder
.htaccess	29-May-14 4:13 PM	HTACCESS File
index	22-Nov-11 7:52 AM	Chrome HTML [

**Fuente:** Elaboración propia.

En la carpeta config se encuentran las configuraciones del framework como añadir librerías, y configurar a que bases de datos se va a dirigir el framework. Además de los drivers para diferentes gestores de bases de datos, también se especifica que las demás carpetas son carpetas que vienen por defecto con el framework codeigniter donde se encuentran los documentos necesarios para la ejecución del framework.

Por otro lado, se recalca que en la carpeta views se encuentran las vistas del proyecto, en controllers los controladores del proyecto y en models los modelos del proyecto.

A continuación se muestra el contenido de las carpetas modelos, vistas y controladores:

**Figura 27:** Contenido de la carpeta views.

Name	Date modified	Type
vista_ingresar_sistema_usuarios	29-May-14 4:13 PM	PHP File
vista_menu_principal	29-May-14 4:13 PM	PHP File
vista_modificar_cursos	29-May-14 4:13 PM	PHP File
<b>vista_modificar_gestiones</b>	29-May-14 4:13 PM	PHP File
vista_modificar_usuarios	29-May-14 4:13 PM	PHP File
vista_mostrar_cursos	29-May-14 4:13 PM	PHP File
vista_mostrar_gestiones	29-May-14 4:13 PM	PHP File
vista_mostrar_usuarios	29-May-14 4:13 PM	PHP File
vista_pagina_principal	29-May-14 4:13 PM	PHP File
vista_registro_cursos	29-May-14 4:13 PM	PHP File
vista_registro_gestion	29-May-14 4:13 PM	PHP File
vista_registro_usuarios	29-May-14 4:13 PM	PHP File

Fuente: Elaboración propia.

**Figura 28:** Contenido de la carpeta controllers.

Name	Date modified	Type
controlador_cursos	29-May-14 4:13 PM	PHP File
controlador_gestion	29-May-14 4:13 PM	PHP File
controlador_usuarios	29-May-14 4:13 PM	PHP File
<b>welcome</b>	29-May-14 4:13 PM	PHP File

Fuente: Elaboración propia.

**Figura 29:** Contenido de la carpeta models.

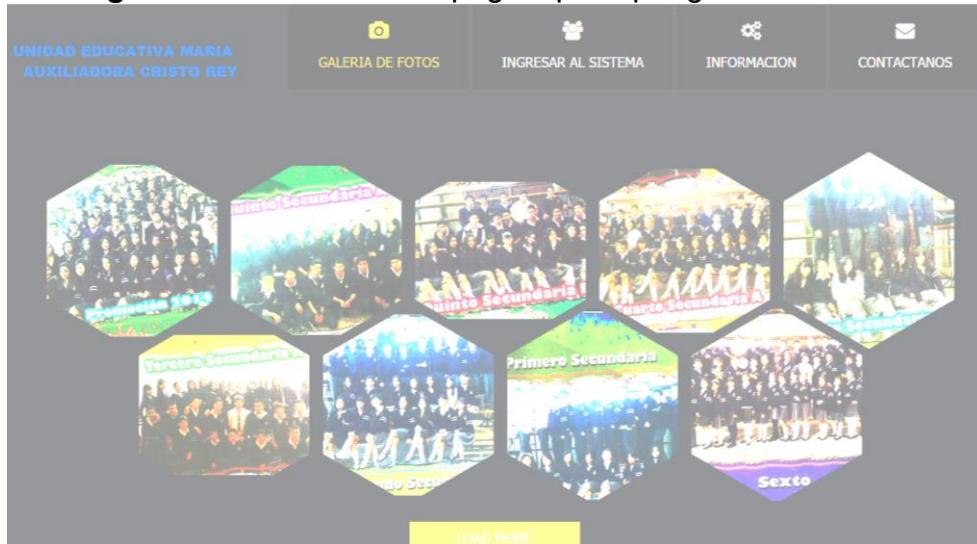
Name	Date modified	Type
modelo_cursos	29-May-14 4:13 PM	PHP File
modelo_gestion	29-May-14 4:13 PM	PHP File
modelo_usuarios	29-May-14 4:13 PM	PHP File

Fuente: Elaboración propia.

### 3.2.8.3. Interfaces del sistema

En la siguiente figura se muestra la interface de la página web galería de fotos en la cual se muestra fotografías de la Unidad educativa María Auxiliadora.

**Figura 30:** Interface de la página principal-galería de fotos.



**Fuente:** Elaboración propia.

En la siguiente imagen se muestra la interface de la página web ingresar al sistema en la cual tenemos ingresar como estudiante y como usuarios.

**Figura 31:** Interface de la página principal-ingresar al sistema.



**Fuente:** Elaboración propia.

En la siguiente figura se muestra la interface de la página web Información en la cual se tiene información sobre la Unidad Educativa y también una breve descripción de que dispositivos y plataformas soporta la aplicación web.

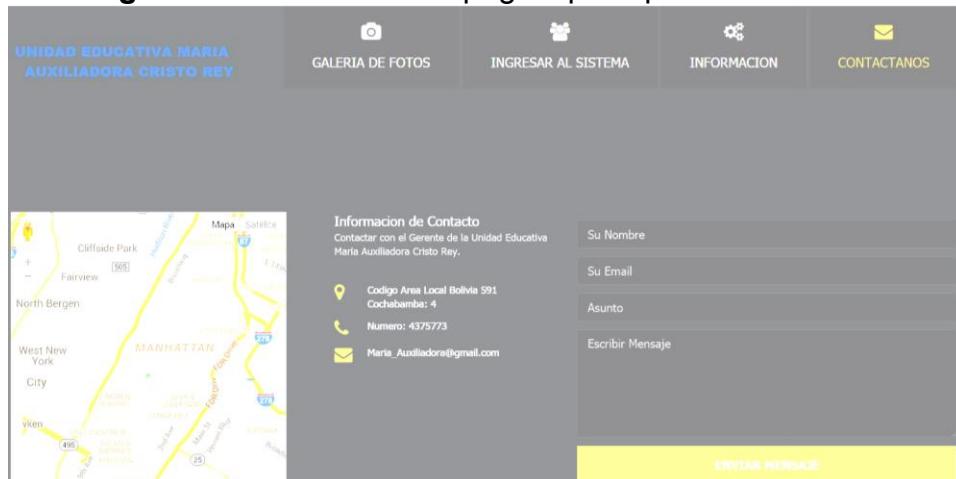
**Figura 32:** Interface de la página principal- información.



**Fuente:** Elaboración propia.

En la siguiente imagen se muestra la interface de la página web Contáctanos en la cual se tiene la opción de mandar un email a la Unidad Educativa para realizar consultas así como también brinda información de la ubicación y el teléfono.

**Figura 33:** Interface de la página principal- contáctanos.



**Fuente:** Elaboración propia.

En la siguiente figura se muestra la interface de la opción ingresar al sistema usuarios, en la cual se ingresa la cedula de identidad y la contraseña del usuario.

**Figura 34:** Interface ingresar al sistema usuarios.



**Fuente:** Elaboración propia.

A continuación se visualiza las interfaces del menú principal.

**Figura 35:** Interface menú principal



**Fuente:** Elaboración propia.

En las siguientes figuras se muestran las interfaces de gestionar usuarios, año educativo y cursos las cuales tiene las opciones registrar, buscar y salir.

**Figura 36:** Interface menú gestionar usuarios.



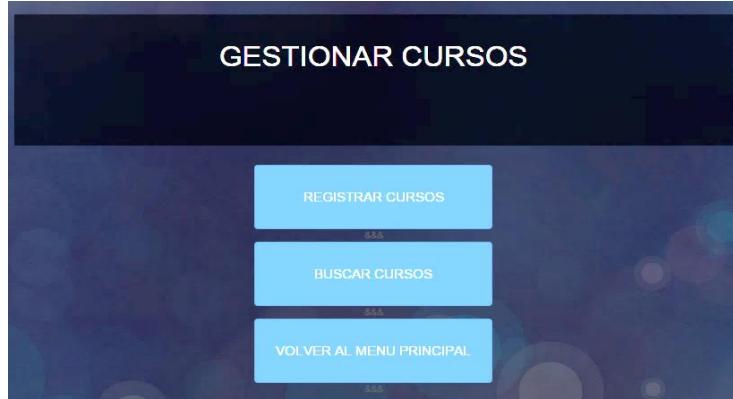
**Fuente:** Elaboración propia.

**Figura 37:** Interface menú gestionar año educativo.



**Fuente:** Elaboración propia.

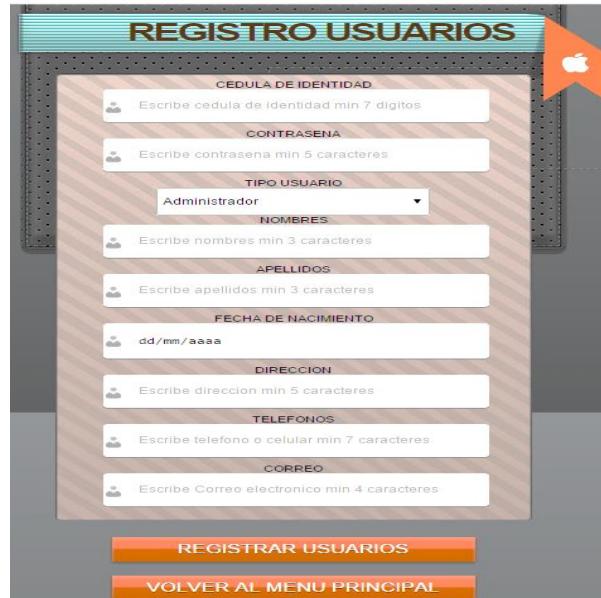
**Figura 38:** Interface menu gestionar cursos.



**Fuente:** Elaboración propia.

A continuación se muestra las interface de registro usuarios.

**Figura 39:** Interface registro de usuarios.



REGISTRO USUARIOS

CEDULA DE IDENTIDAD  
Escribe cedula de identidad min 7 digitos

CONTRASENA  
Escribe contraseña min 5 caracteres

TIPO USUARIO  
Administrador

NOMBRES  
Escribe nombres min 3 caracteres

APELLIDOS  
Escribe apellidos min 3 caracteres

FECHA DE NACIMIENTO  
dd/mm/aaaa

DIRECCION  
Escribe direccion min 5 caracteres

TELEFONOS  
Escribe telefono o celular min 7 caracteres

CORREO  
Escribe Correo electronico min 4 caracteres

REGISTRAR USUARIOS

VOLVER AL MENU PRINCIPAL

This form is titled "REGISTRO USUARIOS". It contains fields for personal information: CEDULA DE IDENTIDAD (with a note to enter at least 7 digits), CONTRASENA (password), TIPO USUARIO (set to "Administrador"), NOMBRES (first name), APELLIDOS (last name), FECHA DE NACIMIENTO (date of birth), DIRECCION (address), TELEFONOS (phone or mobile number), and CORREO (email). At the bottom are two buttons: "REGISTRAR USUARIOS" (Register User) and "VOLVER AL MENU PRINCIPAL" (Return to Main Menu).

**Fuente:** Elaboración propia.

A continuación se muestra las interface de registro año educativo.

**Figura 40:** Interface registro año educativo.



REGISTRO GESTION

AÑO GESTION  
Ingrese el año gestion 4 numeros

MONTO PENSION PRE ESCOLAR (BS)  
Ingrese monto de pension pre escolar

MONTO PENSION PRIMARIA (BS)  
Ingrese monto de pension primaria

MONTO PENSION SECUNDARIA (BS)  
Ingrese monto de pension secundaria

DESCUENTO SEGUNDO HERMANO (%)  
Ingrese % de descuento del segundo hermano

DESCUENTO TERCER HERMANO (%)  
Ingrese % de descuento del tercer hermano

DESCUENTO CUARTO HERMANO (%)  
Ingrese % de descuento del cuarto hermano

REGISTRAR GESTION

VOLVER AL MENU PRINCIPAL

This form is titled "REGISTRO GESTION". It contains fields for financial data: AÑO GESTION (year), MONTO PENSION PRE ESCOLAR (BS) (amount of pre-school pension), MONTO PENSION PRIMARIA (BS) (amount of primary school pension), MONTO PENSION SECUNDARIA (BS) (amount of secondary school pension), DESCUENTO SEGUNDO HERMANO (%) (discount for second sibling), DESCUENTO TERCER HERMANO (%) (discount for third sibling), and DESCUENTO CUARTO HERMANO (%) (discount for fourth sibling). At the bottom are two buttons: "REGISTRAR GESTION" (Register Management) and "VOLVER AL MENU PRINCIPAL" (Return to Main Menu).

**Fuente:** Elaboración propia.

A continuación se muestra las interface de registro cursos.

**Figura 41:** Interface de registro cursos.



REGISTRO DE CURSOS

AÑO GESTIÓN  
2000

NIVEL EDUCATIVO  
PRE-ESCOLAR

NOMBRE CURSO  
PRIMERO

PARALELO CURSO  
A

REGISTRAR CURSOS

VOLVER AL MENU PRINCIPAL

This screenshot shows a registration form titled "REGISTRO DE CURSOS". It includes dropdown menus for "AÑO GESTIÓN" (2000), "NIVEL EDUCATIVO" (PRE-ESCOLAR), "NOMBRE CURSO" (PRIMERO), and "PARALELO CURSO" (A). Below the form are two orange buttons: "REGISTRAR CURSOS" and "VOLVER AL MENU PRINCIPAL". In the top right corner, there is an orange arrow pointing upwards with an Apple logo inside it.

**Fuente:** Elaboración propia.

A continuación se muestran las interfaces de búsqueda de usuarios, gestiones y cursos las cuales incluyen eliminar y modificar.

**Figura 42:** Interface mostrar usuarios.



BUSCAR USUARIOS

Datos de los Usuarios:

BUSCAR

SALIR AL MENU PRINCIPAL

BUSCAR POR CI:

CI USUARIO	CONTRASENA	TIPO USUARIO	NOMBRES	APELLIDOS	FECHA DE NACIMIENTO	DIRECCION	TELEFONO	EMAIL	MODIFICAR	ELIMINAR
5928025	5928025	Administrador	Richard Lorenzo	Severich Paredes	1990-05-26	Av Blanco	76479666	richard_severich@hotmail.com	<a href="#">MODIFICAR</a>	<a href="#">ELIMINAR</a>
7294838	7294838	Administrador	Pablo	Ramirez	1990-12-26	circunvalacion	70414989	pablopzsj2602@gmail.com	<a href="#">MODIFICAR</a>	<a href="#">ELIMINAR</a>

**Fuente:** Elaboración propia.

**Figura 43:** Interface mostrar gestiones.

BUSCAR GESTION								
Datos de las Gestiones:								
AÑO DE LA GESTION	PENSION PRE-ESCOLAR BS	PENSION PRIMARIA BS	PENSION SECUNDARIA BS	DESCUENTO SEGUNDO HERMANO %	DESCUENTO TERCER HERMANO %	DESCUENTO CUARTO HERMANO %	MODIFICAR	ELIMINAR
2014	260 bs	335 bs	367 bs	0 %	30 %	100 %	MODIFICAR	ELIMINAR
2015	100 bs	100 bs	1000 bs	10 %	50 %	100 %	MODIFICAR	ELIMINAR

**Fuente:** Elaboración propia.

**Figura 44:** Interface mostrar cursos.

BUSCAR CURSOS					
CURSOS:					
AÑO GESTION	NIVEL EDUCATIVO	NOMBRE CURSO	PARALELO CURSO	MODIFICAR	ELIMINAR
2014	SECUNDARIA	PRIMERO	A	MODIFICAR	ELIMINAR
2014	SECUNDARIA	SEGUNDO	A	MODIFICAR	ELIMINAR

**Fuente:** Elaboración propia.

### 3.2.9. Pruebas

A continuación se elaboró una tabla de los resultados de las pruebas funcionales y de despliegue multiplataforma.

#### 3.2.9.1. Pruebas Funcionales

Seguidamente se muestran los resultados de las pruebas funcionales:

**Tabla 18:** Pruebas de funcionalidad de primer incremento.

Caso de prueba	Datos entrada	Resultado esperado	Resultado obtenido	Observaciones
Ingresar al sistema usuarios	Datos correcto de cedula de identidad y contraseña.	Ingresar al sistema.	Se obtuvo el resultado esperado	Ninguna
Ingresar al sistema usuarios	Datos incorrecto de cedula de identidad.	Mensaje de error contraseña o cedula incorrecta.	Se obtuvo el resultado esperado.	Ninguna
Registrar usuarios	Datos correctos de entrada.	Mensaje de registro exitosamente.	Se obtuvo el resultado esperado	Ninguna
Registrar usuarios	Datos incorrectos como cedula repetida, validación de campos.	Mensaje de error no se registró.	Se obtuvo el resultado esperado	Ninguna

Caso de prueba	Datos entrada	Resultado esperado	Resultado obtenido	Observaciones
Registrar Año educativo	Datos correctos de entrada	Mensaje de registro exitosamente.	Se obtuvo el resultado esperado	Ninguna
Registrar Año educativo	Datos incorrectos como año educativo repetido, y de validación campos.	Mensaje de error no se registró.	Se obtuvo el resultado esperado	Ninguna
Registrar cursos.	Datos correctos de entrada	Mensaje de registro exitosamente.	Se obtuvo el resultado esperado	Ninguna
Registrar cursos.	Datos incorrectos como datos repetidos, y de validación campos.	Mensaje de error no se registró.	Se obtuvo el resultado esperado	Ninguna
Mostrar Usuarios	Datos búsqueda. de	Paso	Se obtuvo el resultado esperado	Ninguna
Mostrar Anido Educativo	Datos búsqueda. de	Paso	Se obtuvo el resultado esperado	Ninguna
Mostrar Cursos	Datos búsqueda. de	Paso	Se obtuvo el resultado esperado	Ninguna

Fuente: Elaboración propia.

### **3.3. DISEÑO E IMPLEMENTACIÓN DEL MÓDULO DE INSCRIPCIONES DE ESTUDIANTES INTEGRADO AL SUBSISTEMA MULTIPLATAFORMA.**

#### **3.3.1. Análisis**

##### **3.3.1.1. Requerimientos del sistema**

###### **Requerimientos Funcionales**

- El sistema debe permitir registrar información nueva de estudiantes y guardarlas en la base de datos.
- El sistema debe permitir buscar estudiantes para modificarlos o eliminarlos de la base de datos.
- El sistema debe permitir registrar inscripciones asignando a varios estudiantes a un determinado curso y guardarlas en la base de datos.
- El sistema debe permitir buscar inscripciones para poder modificarlas o eliminarlas de la base de datos.

###### **Requerimientos no funcionales**

- **Compatibilidad:** El sistema debe visualizarse correctamente en cualquier plataforma y dispositivo, que sea capaz de ejecutar un navegador web.
- **Disponibilidad:** El sistema debe estar disponible para su uso las 24 horas para que cualquier usuario pueda ingresar.
- **Extensibilidad:** el sistema debe ser capaz de integrarse con facilidad a diferentes sistemas.
- **Seguridad:** El sistema ser accedido con un id y contraseña, por lo que para poder acceder al sistema los usuarios deben estar previamente registrados.
- **Usabilidad:** El sistema debe estar diseñado de manera amigable, para que cualquier persona con poca experiencia con las computadoras pueda usarlas.

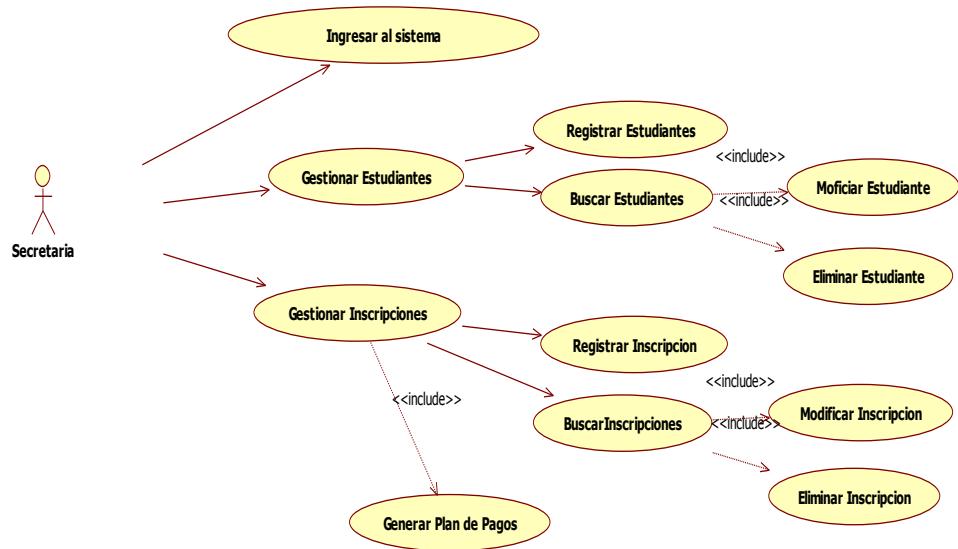
### 3.3.1.2. Identificación de actores

La secretaria es el actor encargado de registrar a los estudiantes nuevos, así como también de modificar errores y la eliminación de registros de alumnos del sistema multiplataforma, también se encarga de las inscripciones de estudiantes a un respectivo curso de una determinada gestión ya registrada por el usuario administrador, para realizar una inscripción primero se selecciona el curso al cual se desea inscribir al estudiante, seguidamente se seleccionará al estudiante previamente registrado, una vez terminada la operación tiene la opción de buscar estudiantes así como también la de buscar inscripciones en las cuales sale los detalles de los estudiantes y las inscripciones para verificar el correcto proceso.

### 3.3.1.3. Diagrama de casos de uso por actor

En la siguiente imagen se visualizará el diagrama de casos de uso del actor secretaria con los siguientes casos de uso: Ingresar al sistema, Gestionar Estudiantes y Gestionar Inscripciones, los cuales tienen las opciones de Registro y Búsqueda.

**Figura 45:** Casos de uso por actor.

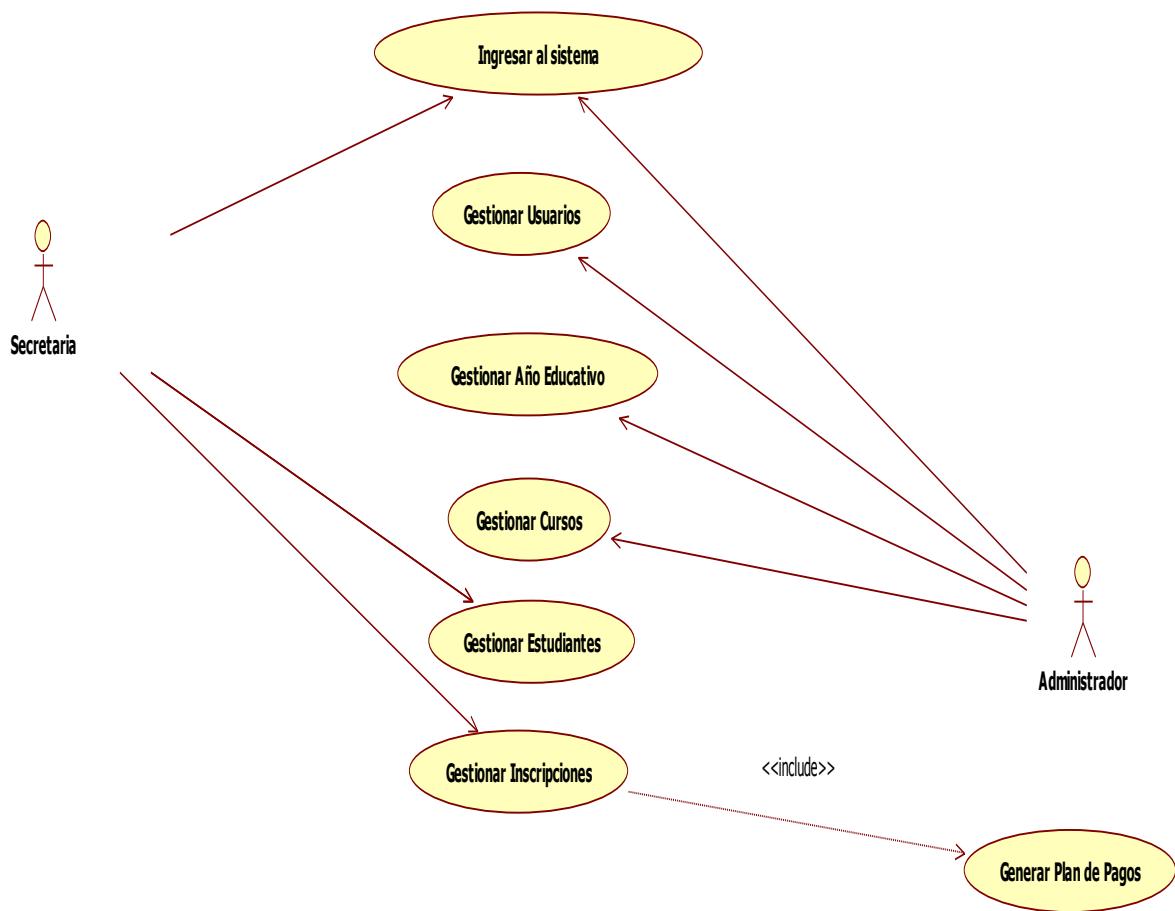


**Fuente:** Elaboración Propia.

### 3.3.1.4. Diagrama de casos de uso del sistema

A continuación se muestra el diagrama de casos de uso del sistema, en el cual existen dos actores: Secretaria y Administrador.

**Figura 46:** Casos de uso del sistema.



**Fuente:** Elaboración Propia.

### 3.3.1.5. Descripción de cada caso de uso

A continuación se realiza la descripción de los casos de uso de gestionar estudiantes y gestionar inscripciones:

**Tabla 19:** Descripción del caso de uso gestionar estudiantes.

CASO DE USO	GESTIONAR ESTUDIANTES	
ACTOR	Secretaria.	
SECUENCIA NORMAL	PASO	ACCIÓN
	1	EL Sistema direcciona al menú “Gestionar Estudiantes” en el cual se encuentran las opciones de registrar y buscar estudiantes.
	2	El usuario ingresa a la opción registrar estudiantes y registra un estudiante nuevo.
	3	El usuario ingresa a la opción buscar estudiantes y se muestra todos los estudiantes que están registrado, además de las opciones de modificar y eliminar.
	4	El usuario puede ingresar el ci de algún estudiante y buscar a un estudiante en particular.
Post condición	El usuario no puede registrar dos estudiantes con la misma cedula de identidad.	
Excepciones	PASO	ACCIÓN
	1	Si los campos de registrar o modificar estudiantes están vacíos el sistema no deja presionar al botón registrar hasta llenar estos campos.
	2	Si los campos de registrar o modificar estudiantes son incorrectos, el sistema devuelve un mensaje de: “Error”.
Comentarios	En el registro de estudiantes se realiza el registro del apoderado para verificar los descuentos.	

**Fuente:** Elaboración Propia.

**Tabla 20:** Descripción del caso de uso gestionar inscripciones.

CASO DE USO	GESTIONAR INSCRIPCIONES	
ACTOR	Secretaria.	
SECUENCIA NORMAL	PASO	ACCIÓN
	1	EL Sistema direcciona al menú “Gestionar Inscripciones” en el cual se encuentran las opciones de registrar y buscar inscripciones.
	2	El usuario ingresa a la opción registrar inscripciones y el sistema muestra una ventana en la cual se elegí un respectivo curso, posteriormente se elegí a los estudiantes que se desea inscribir.
	3	El usuario ingresa a la opción buscar inscripciones y se muestra todos las inscripciones que están registrado, además de las opciones de modificar y eliminar.
	4	El usuario puede ingresar el año educativo para ver inscripciones por gestión.
Post condición	Los cursos y los estudiantes deben estar registrados anteriormente para realizar el registro de una inscripción.	
Excepciones	PASO	ACCIÓN
	1	Si se inscribe a un estudiante dos veces en la misma gestión, el sistema manda un mensaje de error.
Comentarios	En el registro de inscripciones primero se selecciona un curso y luego se puede seleccionar a varios estudiantes para asignarlos al curso seleccionado.	

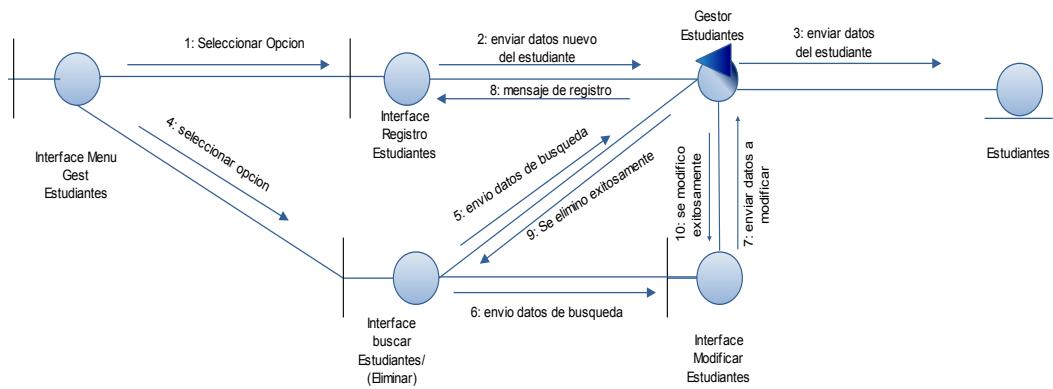
**Fuente:** Elaboración Propia.

### 3.3.2. Diseño

A continuación se presentan los diagramas de colaboración para cada caso de uso identificado.

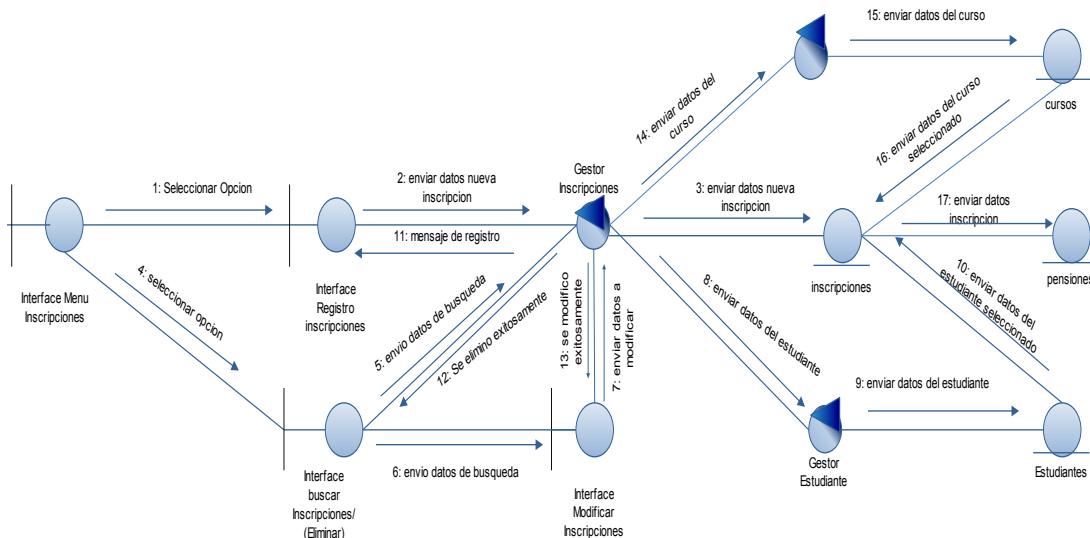
#### 3.3.2.1. Diagramas de colaboración

**Figura 47:** Diagrama de colaboración estudiantes.



Fuente: Elaboración Propia.

**Figura 48:** Diagrama de colaboración inscripciones.

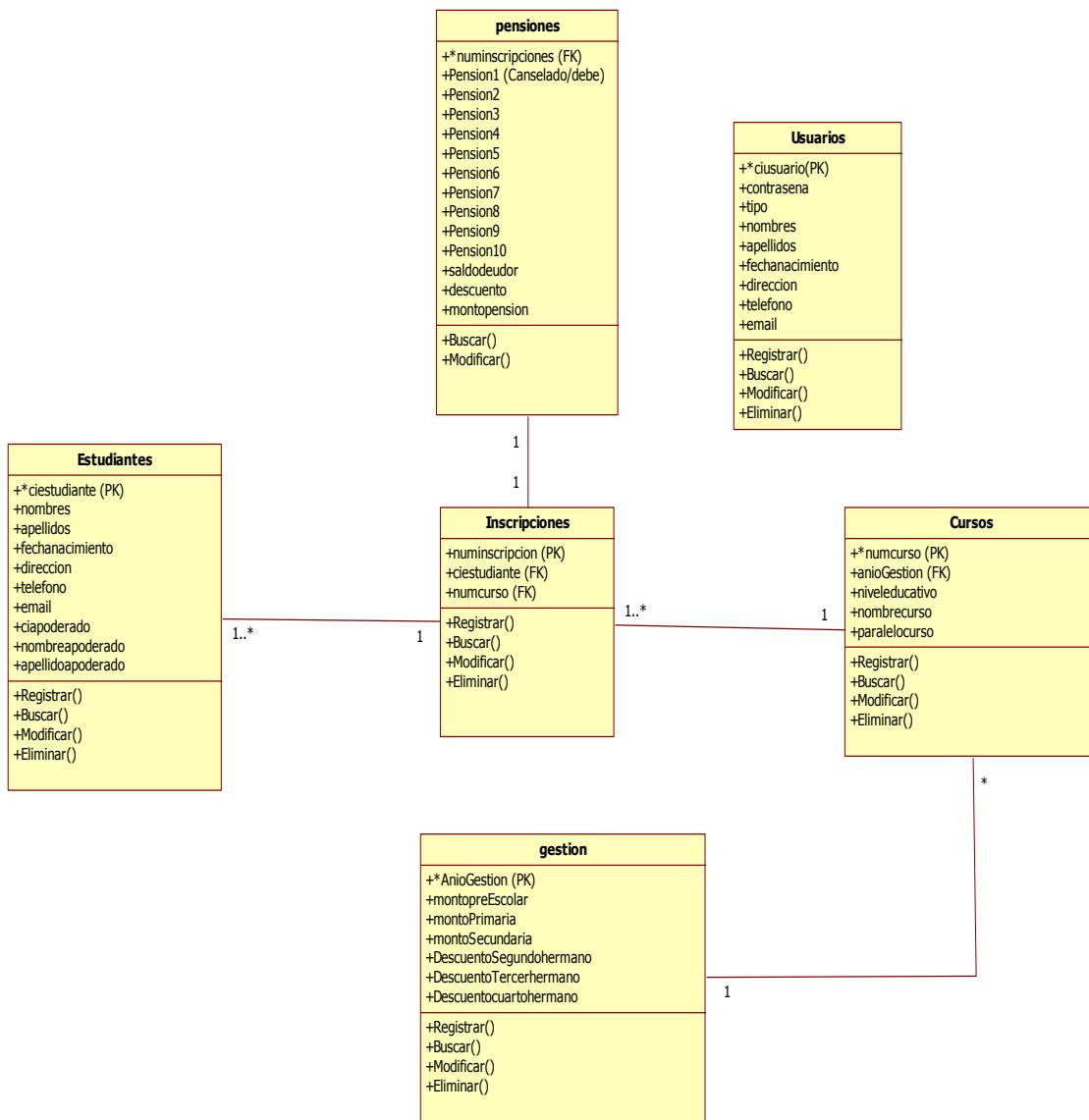


Fuente: Elaboración Propia.

En base a los diagramas de colaboración ya mostrados anteriormente se ha podido elaborar los diagramas de clases y bases de datos los que se muestran a continuación:

### 3.3.2.2. Diagrama de clases

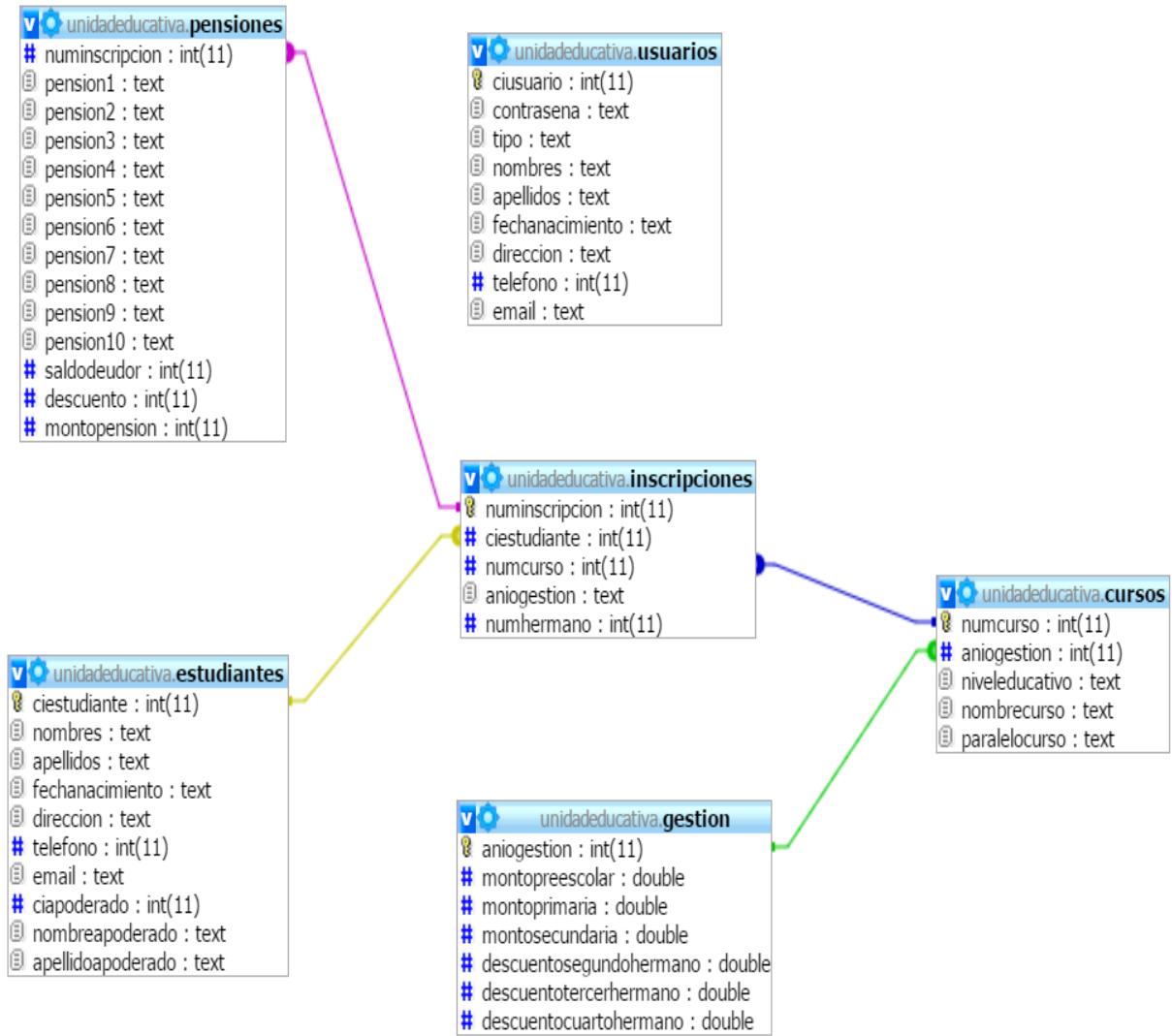
**Figura 49:** Diagrama de clases.



**Fuente:** Elaboración Propia.

### 3.3.2.3. Diagrama de base de datos

**Figura 50:** Diagrama de base de datos.



**Fuente:** Elaboración Propia.

### 3.3.2.4. Diccionario de datos

A continuación se muestra el diccionario de datos de la tabla usuarios, gestión y cursos:

**Tabla 21:** Diccionario de datos tabla estudiantes.

Columna	Tipo de dato	Null	Llave primaria	Llave foránea	Descripción
cuestudiante	Int	No	Si	No	Registra la cedula de identidad del estudiante y es la llave primaria de la tabla estudiantes.
nombres	Text	No	No	No	Registra los nombres del estudiante.
apellidos	Text	No	No	No	Registra los apellidos del estudiante.
fechanacimiento	Text	No	No	No	Registra la fecha de nacimiento del estudiante.
dirección	Text	No	No	No	Se registra la ubicación de la vivienda del estudiante.
teléfono	Int	No	No	No	Registra el teléfono o celular del estudiante.

email	Text	No	No	No	Registra el porcentaje de descuento por cuarto hermano o superior.
ciapoderado	Int	No	No	No	Registra la cedula de identidad del apoderado.
nombreapoderado	Text	No	No	No	Registra los nombres del apoderado.
apellidoapoderado	Text	No	No	No	Registra los apellidos del apoderado.

**Fuente:** Elaboración propia.

**Tabla 22:** Diccionario de datos tabla inscripciones.

Columna	Tipo de dato	Null	Llave primaria	Llave foránea	Descripción
numinscripcion	Int	No	Si	No	Registra el número de inscripción y es la clave primaria de la tabla inscripciones.
ciestudiante	Int	No	No	Si	Registra la cedula de identidad de un estudiante, del cual se desea inscribir al sistema.

numcurso	Int	No	No	Si	Registra la clave primaria del curso al que se desea inscribir.
aniogestion	Int	No	No	No	Registra el año de gestión de la inscripción.
numhermano	Int	No	No	No	Registra el número de hermano.

**Fuente:** Elaboración propia.

**Tabla 23:** Diccionario de datos tabla pensiones.

Columna	Tipo de dato	Null	Llave primaria	Llave foránea	Descripción
numinscripcion	Int	No	No	Si	Registra la cedula de identidad del estudiante y es la llave primaria de la tabla estudiantes.
Pension1	Text	No	No	No	Registra si debe o no la primera pensión.
Pension2	Text	No	No	No	Registra si debe o no la segunda pensión.
Pension3	Text	No	No	No	Registra si debe o no la tercera pensión.

Pension4	Text	No	No	No	Registra si debe o no la cuarta pensión.
Pension5	Int	No	No	No	Registra si debe o no la quinta pensión.
Pension6	Text	No	No	No	Registra si debe o no la sexta pensión.
Pension7	Int	No	No	No	Registra si debe o no la séptima pensión.
Pension8	Text	No	No	No	Registra si debe o no la octava pensión.
Pension9	Text	No	No	No	Registra si debe o no la novena pensión.
Pension10	Text	No	No	No	Registra si debe o no la décima pensión.
saldodeudor	Int	No	No	No	Registra el total del saldo deudor que falta.
descuento	Int	No	No	No	Registra el porcentaje de descuento.
montopension	Int	No	No	No	Registra el monto de cada pensión.

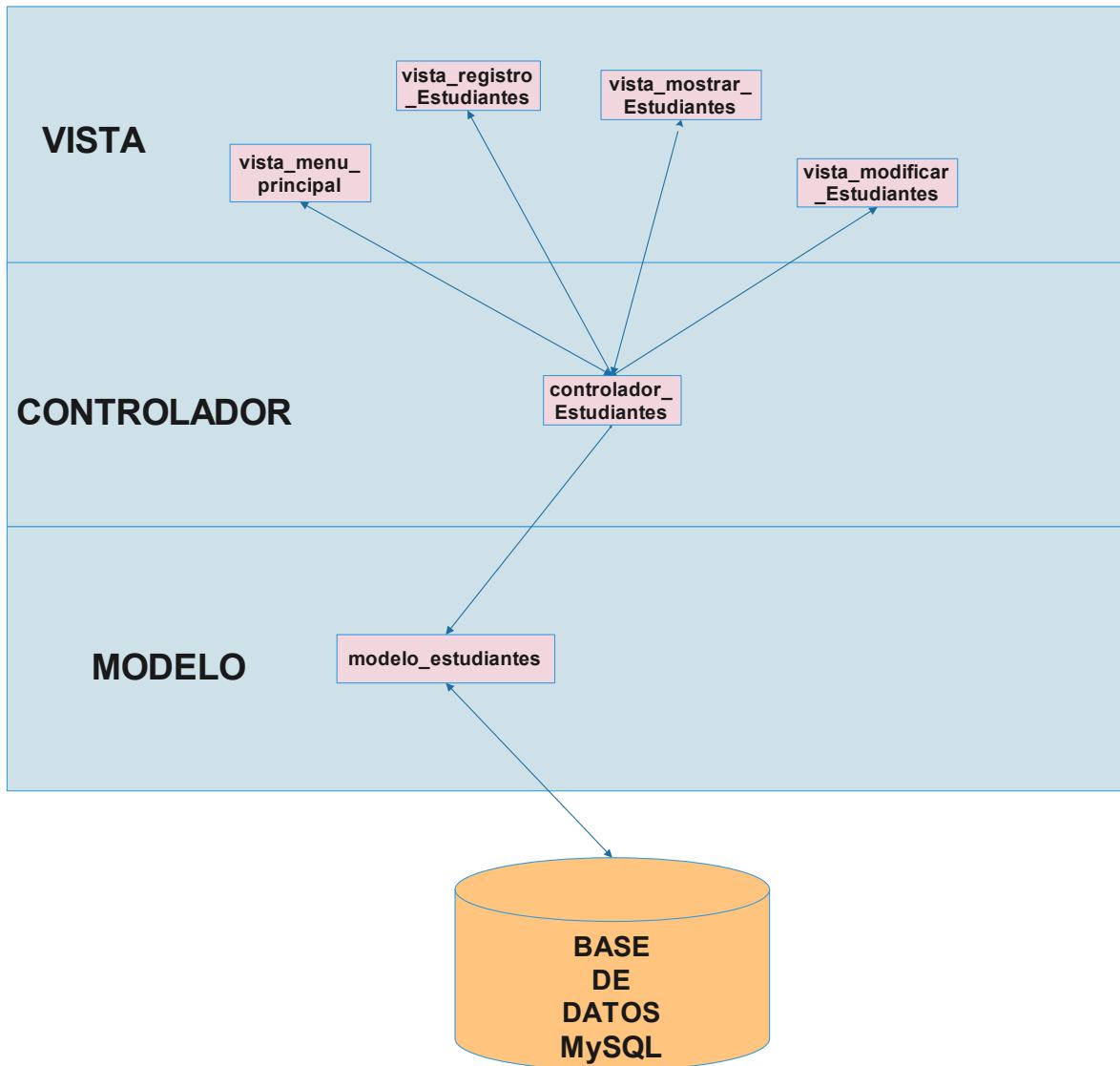
**Fuente:** Elaboración propia.

### 3.3.3. Implementación

#### 3.3.3.1. Arquitectura del Software

A continuación se muestra la arquitectura de los componentes del sistema para llevar a cabo la gestión de estudiantes.

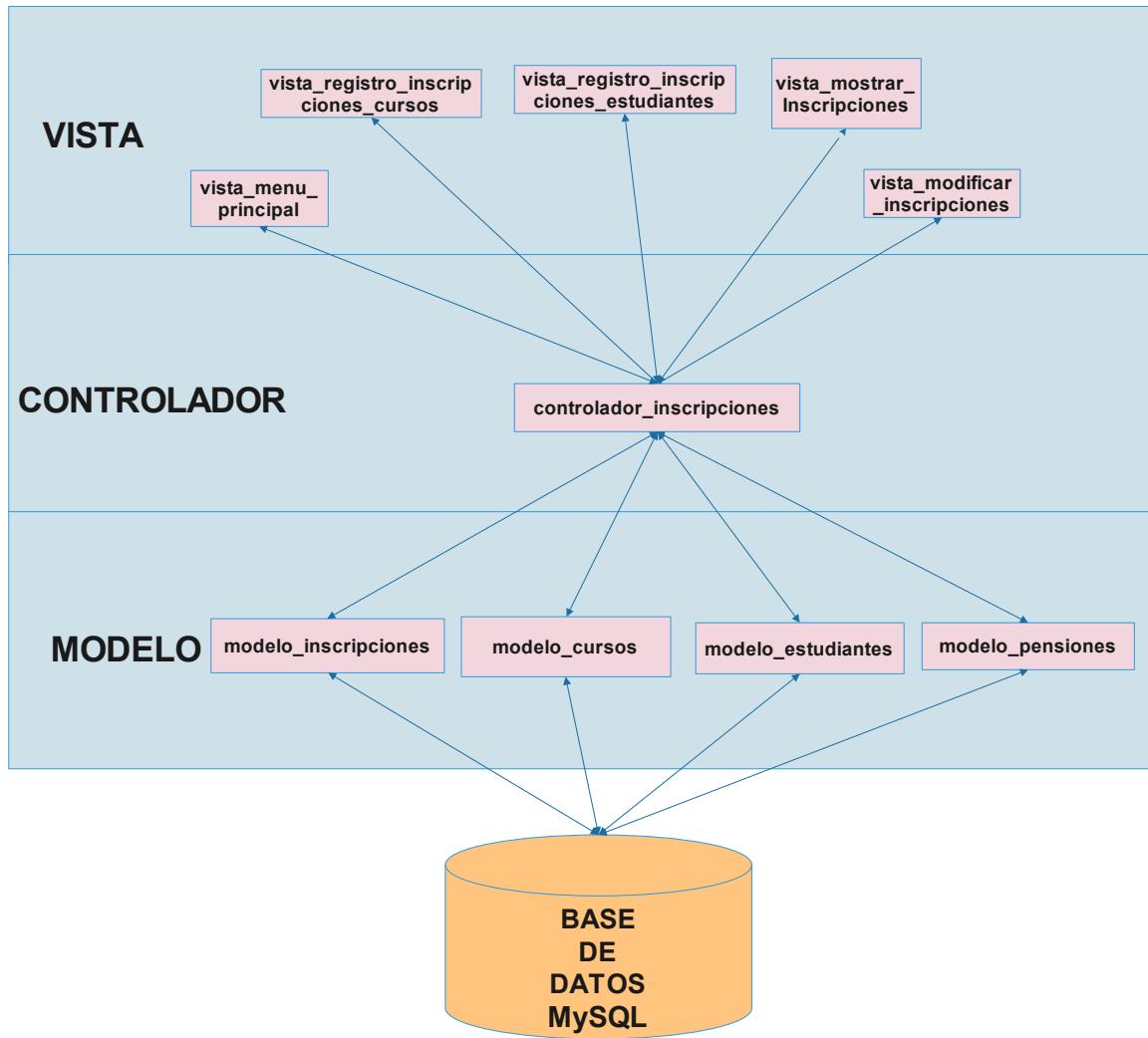
**Figura 51:** Arquitectura para el caso de uso gestionar estudiantes.



**Fuente:** Elaboración Propia.

A continuación se muestra la arquitectura de los componentes del sistema para llevar a cabo la gestión de inscripciones.

**Figura 52:** Arquitectura para el caso de uso gestionar inscripciones.



**Fuente:** Elaboración Propia.

### 3.3.3.2. Organización del proyecto

A continuación se muestra el contenido de la carpeta views con las respectivas vistas aumentadas al proyecto.

**Figura 53:** Contenido de la carpeta views.

Name	Date modified	Type
vista_ingresar_sistema_usuarios	29-May-14 4:13 PM	PHP File
vista_menu_principal	29-May-14 4:13 PM	PHP File
vista_modificar_cursos	29-May-14 4:13 PM	PHP File
vista_modificar_estudiantes	29-May-14 4:13 PM	PHP File
vista_modificar_gestiones	29-May-14 4:13 PM	PHP File
vista_modificar_inscripciones_elegir_cursos	29-May-14 4:13 PM	PHP File
vista_modificar_usuarios	29-May-14 4:13 PM	PHP File
vista_mostrar_cursos	29-May-14 4:13 PM	PHP File
vista_mostrar_estudiantes	29-May-14 4:13 PM	PHP File
vista_mostrar_gestiones	29-May-14 4:13 PM	PHP File
vista_mostrar_inscripciones	29-May-14 4:13 PM	PHP File
vista_mostrar_usuarios	29-May-14 4:13 PM	PHP File
vista_pagina_principal	29-May-14 4:13 PM	PHP File
vista_registro_cursos	29-May-14 4:13 PM	PHP File
vista_registro_estudiantes	29-May-14 4:13 PM	PHP File
vista_registro_gestion	29-May-14 4:13 PM	PHP File
vista_registro_inscripciones_cursos	29-May-14 4:13 PM	PHP File
vista_registro_inscripciones_estudiantes	29-May-14 4:13 PM	PHP File
vista_registro_usuarios	29-May-14 4:13 PM	PHP File

**Fuente:** Elaboración Propia.

A continuación se muestra el contenido de la carpeta controllers con los respectivos controladores aumentados al proyecto.

**Figura 54:** Contenido de la carpeta controllers.

Name	Date modified	Type
controlador_cursos	29-May-14 4:13 PM	PHP File
controlador_estudiantes	29-May-14 4:13 PM	PHP File
controlador_gestion	29-May-14 4:13 PM	PHP File
controlador_inscripciones	29-May-14 4:13 PM	PHP File
controlador_usuarios	29-May-14 4:13 PM	PHP File
welcome	29-May-14 4:13 PM	PHP File

**Fuente:** Elaboración Propia.

A continuación se muestra el contenido de la carpeta models con los respectivos modelos aumentados al proyecto.

**Figura 55:** Contenido de la carpeta models.

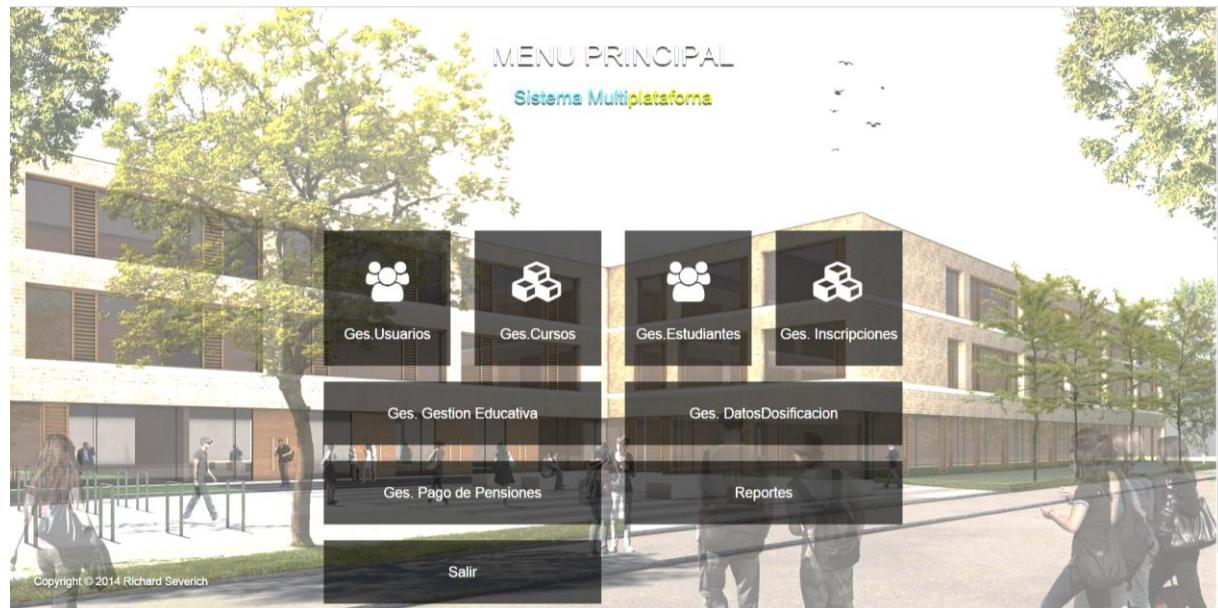
Name	Date modified	Type
modelo_cursos	29-May-14 4:13 PM	PHP File
modelo_estudiantes	29-May-14 4:13 PM	PHP File
modelo_gestion	29-May-14 4:13 PM	PHP File
modelo_inscripciones	29-May-14 4:13 PM	PHP File
modelo_pensiones	29-May-14 4:13 PM	PHP File
modelo_usuarios	29-May-14 4:13 PM	PHP File

**Fuente:** Elaboración Propia.

### 3.3.3.3. Interfaces del sistema

A continuación se visualiza la interface del menú principal.

**Figura 56:** Interface menú principal.



**Fuente:** Elaboración Propia.

A continuación se visualiza la interface de gestionar estudiantes en la cual se encuentras las opciones de registrar estudiante, buscar estudiante y salir al menú principal.

**Figura 57:** Interface gestionar estudiantes.



**Fuente:** Elaboración Propia.

A continuación se muestra la interface de gestionar inscripciones en la cual se encuentran las opciones de registrar inscripción, buscar inscripciones y salir al menú principal.

**Figura 58:** Interface gestionar inscripciones.



**Fuente:** Elaboración Propia.

A continuación se muestra la interfaz de registro de estudiantes.

**Figura 59:** Interface registro estudiantes.

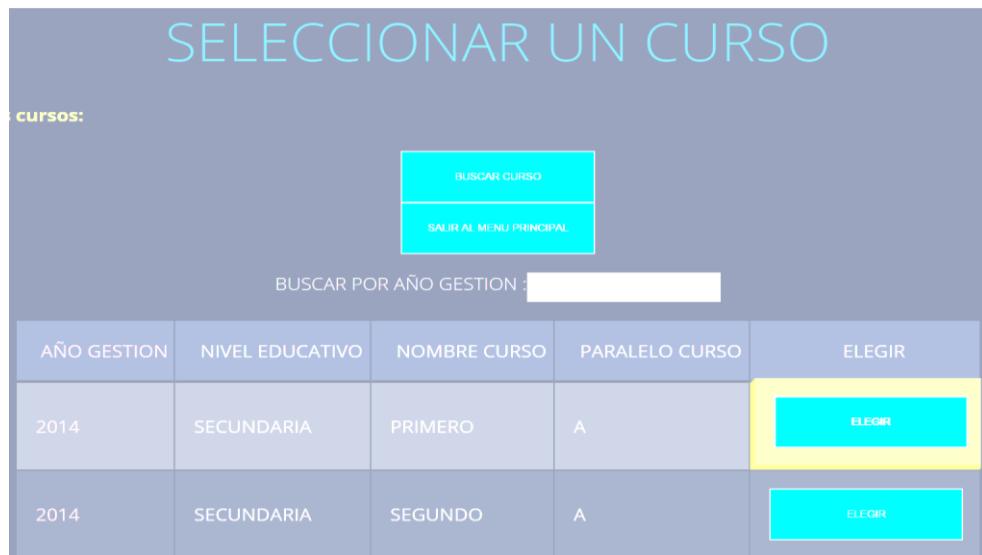


The interface titled "REGISTRO ESTUDIANTES" is designed for student registration. It features a central vertical column of input fields with placeholder text and validation messages. The fields are organized into sections: CEDULA DE IDENTIDAD, NOMBRES, APELLIDOS, FECHA DE NACIMIENTO, DIRECCION, TELEFONOS, CORREO, and another CEDULA DE IDENTIDAD section for the guardian. Each field includes a small icon and a validation message. At the bottom are two orange buttons: "REGISTRAR ESTUDIANTES" and "VOLVER AL MENU PRINCIPAL".

**Fuente:** Elaboración Propia.

A continuación se muestra la interfaz del registro de inscripciones con selección de curso, en la que se elige un curso para realizar una o varias inscripciones.

**Figura 60:** Interface registro inscripciones elegir curso.



The interface titled "SELECCIONAR UN CURSO" allows users to select a course. It includes a search bar for "CURSOS", a "BUSCAR CURSO" button, and a "SALIR AL MENU PRINCIPAL" button. Below is a search bar for "BUSCAR POR AÑO GESTION". A table lists courses by year, level, name, parallel, and an "ELEGIR" button. The "ELEGIR" button for the first row is highlighted with a yellow background.

AÑO GESTION	NIVEL EDUCATIVO	NOMBRE CURSO	PARALELO CURSO	ELEGIR
2014	SECUNDARIA	PRIMERO	A	<b>ELEGIR</b>
2014	SECUNDARIA	SEGUNDO	A	<b>ELEGIR</b>

**Fuente:** Elaboración Propia.

A continuación se muestra la interface de registro inscripciones con elegir estudiantes, en la cual se elige uno o varios estudiantes para inscribirlos, también tiene la opción buscar estudiantes por cedula de identidad.

**Figura 61:** Interface registro inscripciones elegir estudiantes.

CI ESTUDIANTES	NOMBRES	APPELLIDOS	FECHA DE NACIMIENTO	DIRECCION	TELEFONOS	EMAIL	CI APODERADO	NOMBRES APODERADO	APPELLIDOS APODERADO	INSCRIBIR
5928026	Richard Lorenzo	Severich Paredes	1999-11-18	Av Blango Galindo	4675773	richard_severich@hotmail.com	5928026	Richard Lorenzo	Severich Paredes	<b>INSCRIBIR</b>
5928027	Emanuelle	Rojas	1990-02-26	av llandival	4269855	richard_severich@gmail.com	5928027	eugeniarson	chachawama	<b>INSCRIBIR</b>

**Fuente:** Elaboración Propia.

A continuación se muestra la interface buscar estudiantes la cual incluye modificar y eliminar.

**Figura 62:** Interface mostrar estudiantes.

CI ESTUDIANTES	NOMBRES	APPELLIDOS	FECHA DE NACIMIENTO	DIRECCION	TELEFONOS	EMAIL	CI APODERADO	NOMBRES APODERADO	APPELLIDOS APODERADO	MODIFICAR	ELIMINAR
5928026	Richard Lorenzo	Severich Paredes	1999-11-18	Av Blango Galindo	4675773	richard_severich@hotmail.com	5928026	Richard Lorenzo	Severich Paredes	<b>MODIFICAR</b>	<b>ELIMINAR</b>
5928027	Emanuelle	Rojas	1990-02-26	av llandival	4269855	richard_severich@gmail.com	5928027	eugeniarson	chachawama	<b>MODIFICAR</b>	<b>ELIMINAR</b>
5928033	lolito	lazozo	1999-09-16	av muela	4375773	lolito_123@hotmail.com	5928054	lolo mayor	Caballo	<b>MODIFICAR</b>	<b>ELIMINAR</b>
5928090	Erick	cosmos	1998-09-11	calle zzz	4269855	richard_severich@hotmail.com	5928025	Richard	Severich	<b>MODIFICAR</b>	<b>ELIMINAR</b>

**Fuente:** Elaboración Propia.

A continuación se muestra la interfaz mostrar inscripciones la cual incluye modificar y eliminar, también tiene la opción buscar por año de gestión.

**Figura 63:** Interface mostrar inscripciones.

CI ESTUDIANTE	ANIOGESTION	CURSO	NIVEL	PARALELO	NUMERO HERMANO	MODIFICAR	ELIMINAR
8052938	2015	SEXTO	SECUNDARIA	B	1	<a href="#">MODIFICAR</a>	<a href="#">ELIMINAR</a>
7541254	2015	SEXTO	SECUNDARIA	B	2	<a href="#">MODIFICAR</a>	<a href="#">ELIMINAR</a>

**Fuente:** Elaboración Propia.

### 3.3.4. Pruebas

A continuación se realizarán las pruebas de funcionalidad:

#### 3.3.4.1. Pruebas funcionales

**Tabla 24:** Pruebas de funcionalidad del incremento.

Caso de prueba	Datos entrada	Resultado esperado	Resultado obtenido	OBSERVACIONES
Registrar Estudiantes	Datos correcto	Registro exitosamente.	Se obtuvo el resultado esperado	Ninguna
Registrar Estudiantes	Datos incorrectos como cedula de identidad repetida y	Mensaje de error no se registró.	Se obtuvo el resultado esperado.	Ninguna

Caso de prueba	Datos entrada	Resultado esperado	Resultado obtenido	OBSERVACIONES
	datos inválidos.			
Registrar inscripciones	Datos de selección curso y estudiantes a inscribir al curso.	Mensaje de registro exitosamente.	Se obtuvo el resultado esperado	Ninguna
Registrar inscripciones	Datos incorrectos de selección inscribir al mismo alguno a dos cursos en la misma gestión.	Mensaje de error.	Se obtuvo el resultado esperado	Ninguna

Fuente: Elaboración propia.

### **3.4. DESARROLLO DE LOS MÓDULOS DE COBRO DE PENSIONES CON FACTURACIÓN Y REPORTES INTEGRADO AL SUB SISTEMA MULTIPLATAFORMA.**

#### **3.4.1. Análisis**

El cajero es el encargado de cobro de pensiones de los estudiantes, cuando entra un interesado en pagar pensiones a la institución el cajero se encarga de buscar en el sistema al estudiante y brindar información de cuantas pensiones debe el respectivo estudiante, seguidamente el interesado paga la o las pensiones respectivas al cajero,

este registra en el sistema las respectivas pensiones y el sistema genera una factura computariza de las pensiones pagadas.

### 3.4.1.1. Análisis del código de control.

El código de control es un dato alfanumérico generado e impreso por el sistema de facturación computarizada, a tiempo de emitir la factura. Este código de control se genera en base a la información de dosificación de la factura, información de la transacción comercial, y un dato alfanumérico denominado llave, que el contribuyente recibirá por internet cada vez que solicite dosificaciones de facturas para su sistema de facturación computarizada.

La generación del código de control se realiza con tres algoritmos los cuales son mostrados y descritos en la siguiente figura:

**Figura 64:** Algoritmos utilizados para la generación del código de control.

<b>Alleged RC4</b>	<i>Un algoritmo de criptografía simétrica, basado en cifrado de flujo (stream cipher), muy utilizado por su rendimiento y simplicidad.</i>
<b>Verhoeff</b>	<i>Algoritmo de dígito verificador que trabaja con cadenas de dígitos decimales de cualquier tamaño. Además de detectar una amplia gama de errores en datos numéricos, este algoritmo también detecta casos de transposición de dígitos adyacentes.</i>
<b>Base 64</b>	<i>Algoritmo que convierte cifras en base 10 a base 64, utilizando divisiones sucesivas además de un diccionario de 64 caracteres. El diccionario a utilizarse para efectos del Código de Control es:</i> <i>0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z, a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, +, /</i>

**Fuente:** Elaboración Propia

El funcionamiento de los algoritmos mencionados están descritos en el anexo “G”, a continuación se detalla la generación de un código de control con un ejemplo:

- Para la generación del código de control se requiere de los siguiente datos:

**Figura 65:** Datos iniciales para el código de control.

Número de autorización:	<b>2901091557</b>
Número de factura:	<b>1503</b>
NIT / CI del cliente:	<b>3734314017</b>
Fecha de la transacción:	<b>20070610</b>
Monto total:	<b>4968</b>
Llave de la dosificación:	<b>SeSaMo</b>

**Fuente:** Elaboración Propia

- Se obtienen 2 dígitos verificador **Verhoeff** mediante concatenación al final de los datos: Número Factura, NIT / CI del Cliente, Fecha de la Transacción y Monto de la Transacción.

**Figura 66:** Datos obtenidos por el algoritmo verhoeff.

A cada uno de los datos obtenidos agregarle su dígito verificador Verhoeff.

Número de autorización:	<b>29010915579</b>
Número de factura:	<b>15031</b>
NIT / CI del cliente:	<b>37343140179</b>
Fecha de la transacción:	<b>200706101</b>
Monto total:	<b>49689</b>

**Fuente:** Elaboración Propia

- Luego se halla la sumatoria de los datos obtenidos en donde tenemos como resultado:

**Figura 67:** Sumatoria de los datos obtenidos por el algoritmo verhoeff.

Número de autorización:	<b>29010915579</b>
Número de factura:	<b>15031</b>
NIT / CI del cliente:	<b>37343140179</b>
Fecha de la transacción:	<b>200706101</b>
Monto total:	<b>49689</b>
Suma Aritmética:	<b>66554826579</b>

**Fuente:** Elaboración Propia

- Luego se halla el módulo del resultado obtenido:

**Figura 68:** Resultado obtenido del módulo.

Hallar el Módulo  $(64^5 - 1)$  del resultado obtenido.

$$66554826579 \text{ Mod } (64^5 - 1) = \boxed{\mathbf{1056575376}}$$

**Fuente:** Elaboración Propia

- Luego convertimos el resultado obtenido con el Algoritmo Base 64:

**Figura 69:** Resultado del algoritmo base 64.

Convertir el resultado obtenido a base 64.

$$\text{Base64}(1056575376) = \boxed{\mathbf{++W+G}}$$

**Fuente:** Elaboración Propia

- Usando el Algoritmo Alleged RC4 y la llave asignada por el SIN Ejemplo: SeSeMo, encriptar la información obtenida en el paso anterior:

**Figura 70:** Resultado del algoritmo allegedRC4.

$$\text{AllegedRC4}(++W+G, \text{SeSaMo}) = \boxed{\mathbf{C4 3B 93 A8 25}}$$

**Fuente:** Elaboración Propia

- La Información resultante del proceso de encriptación, expresada en formato hexadecimal se denominará código de control y deberá ser impresa en la factura emitida.

**Figura 71:** Código de control.

Código de Control: **C4-3B-93-A8-25**

**Fuente:** Elaboración Propia

### **3.4.1.2. Requerimiento del sistema**

#### **Requerimientos Funcionales**

- El sistema debe permitir registrar datos de dosificación y guardarlos en la base de datos.
- El sistema debe permitir buscar datos de dosificación para modificarlos o eliminarlos de la base de datos.
- El sistema debe permitir registrar pensiones pagadas de los estudiantes inscritos y guardarlos en la base de datos.
- El sistema debe permitir generar factura siguiendo las normativas del servicio de impuestos nacionales.

#### **Requerimientos no funcionales**

- **Compatibilidad:** El sistema debe visualizarse correctamente en cualquier plataforma y dispositivo, que sea capaz de ejecutar un navegador web.
- **Disponibilidad:** El sistema debe estar disponible para su uso las 24 horas para que cualquier usuario pueda ingresar.
- **Extensibilidad:** el sistema debe ser capaz de integrarse con facilidad a diferentes sistemas.
- **Mantenibilidad:** El sistema debe estar listo para agregar nuevos módulos y cumplir con requerimientos funcionales.
- **Seguridad:** El sistema debe ser accedido únicamente por usuarios registrados en el sistema.

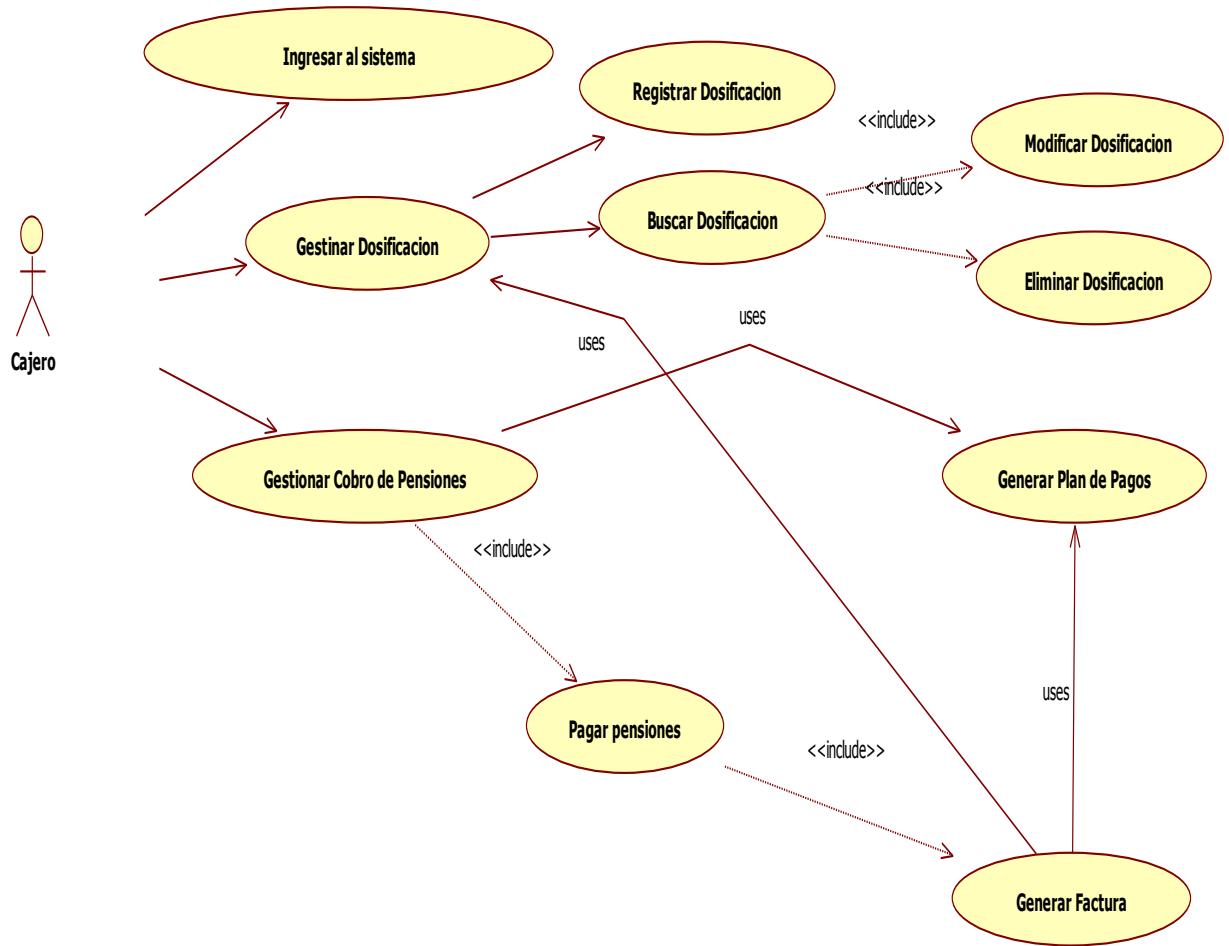
### **3.4.1.3. Identificación de Actores**

El cajero es el actor encargado de registrar los datos de dosificación como también de modificarlo y eliminarlos, por otro lado también se encarga del cobro de pensiones de los estudiantes inscritos así como la generación de factura computarizada de estos.

### 3.4.1.4. Casos de Uso por Actor

A Continuación se muestra el diagrama de casos de uso por actor cajero, el cual tiene tres casos de uso principales los cuales son ingresar al sistema, gestionar dosificación y gestionar cobro de pensiones:

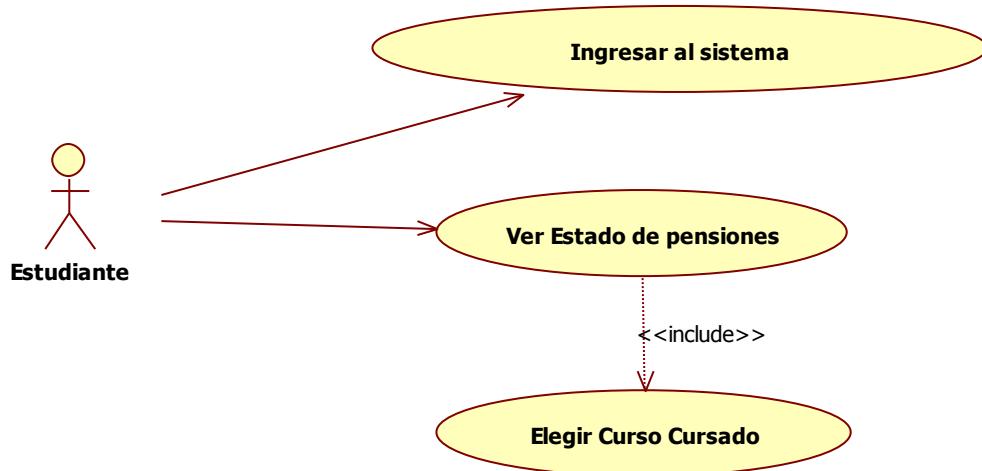
**Figura 72:** Diagrama de casos de uso por actor cajero.



**Fuente:** Elaboración propia.

A continuación se muestra el diagrama de casos de uso por actor, del actor estudiante el cual ingresa al sistema modo estudiantes con la cedula de identidad previamente registrado, una vez dentro del sistema se muestra la lista de cursos cursados por este estudiante por gestión, este debe elegir un curso para visualizar el estado de pensiones del curso seleccionado.

**Figura 73:** Diagrama de casos de uso por actor estudiante.



**Fuente:** Elaboración propia.

En la siguiente página se muestra el diagrama de casos de uso del sistema en el cual se encuentran todos los actores así como también todos los casos de uso del sistema final.

#### 3.4.1.5. Casos de Uso del sistema

A continuación en la siguiente imagen se muestra el diagrama de casos de uso del sistema:

**Figura 74:** Diagrama de casos de uso del sistema.



**Fuente:** Elaboración Propia.

### 3.4.1.6. Descripción de casos de uso

A continuación se muestra la descripción del caso de uso gestionar dosificación:

**Tabla 25:** Descripción del caso de uso gestionar dosificación.

CASO DE USO	GESTIONAR DOSIFICACIÓN	
ACTOR	Cajero.	
SECUENCIA NORMAL	PASO	ACCIÓN
	1	EL Sistema direcciona al menú “Gestionar Dosificación” en el cual se encuentran las opciones de registrar y buscar dosificación.
	2	El usuario ingresa a la opción de registrar dosificación y registra nuevos datos.
	3	El usuario ingresa a la opción buscar dosificación y se muestra todos los datos de dosificación, además de las opciones de modificar y eliminar.
	4	El usuario puede ingresar el nit del colegio de alguna dosificación y buscar a alguna en particular.
Post condición	El usuario no puede registrar dos registros de dosificación con el mismo nit.	
Excepciones	PASO	ACCIÓN
	1	Si los campos de registrar o modificar dosificación están vacíos el sistema no deja presionar al botón registrar hasta llenar estos campos.
	2	Si los campos de registrar o modificar estudiantes son incorrectos, el sistema devuelve un mensaje de: “Error”.
Comentarios	En el registro de dosificación se realiza la inserción de algunos datos que serán utilizados para la generación del código de control y código QR.	

**Fuente:** Elaboración Propia.

A continuación se muestra la descripción del caso de uso gestionar pago de Pensiones:

**Tabla 26:** Descripción del caso de uso gestionar pago de pensiones.

CASO DE USO	GESTIONAR PAGO DE PENSIONES	
ACTOR	Cajero.	
SECUENCIA NORMAL	PASO	ACCIÓN
	1	EL sistema direcciona a la interfaz buscar estudiantes inscritos en la cual se elige un estudiante, cabe decir que en esta opción se encuentra la opción de buscar estudiantes por ci.
	2	El usuario elige un estudiante y se le despliega una interfaz en la cual se muestra los datos del estudiante y las pensiones respectivas a pagar, cabe decir que en esta interfaz se encuentra la opción de generar factura.
	3	El usuario ingresa a la opción generar factura, se abre una ventana extra con la factura generada en formato PDF.
	4	El usuario puede pagar las pensiones que deseé antes de generar factura.
Post condición	El usuario genera factura sin haber pagado pensiones, el sistema no generara ninguna factura y devolverá un mensaje de error.	
Excepciones	PASO	ACCIÓN
	1	Si se quiere generar una factura sin haber registrado datos de dosificación se devuelve el mensaje “no puede facturar”.
	2	Si los campos de NIT cliente y apellido cliente se los deja en blanco se imprime la factura sin nombre.
Comentarios	AL generar la factura, se genera el código de control y código QR de esta.	

**Fuente:** Elaboración Propia.

A continuación se muestra la descripción del caso de uso ver estado de pensiones:

**Tabla 27:** Descripción del caso de uso ver estado de pensiones.

<b>CASO DE USO</b>	<b>ELEGIR CURSO CURSADO</b>	
<b>ACTOR</b>	Estudiante.	
<b>SECUENCIA NORMAL</b>	<b>PASO</b>	<b>ACCIÓN</b>
	<b>1</b>	EL Sistema direcciona al menú “Elegir curso cursado” en el cual se encuentran los cursos cursados por el estudiante.
	<b>2</b>	El estudiante elige un curso para ver el estado de pensiones del curso seleccionado.
	<b>3</b>	Al estudiante se le muestra el estado de pensiones.
	<b>4</b>	El usuario puede visualizar el estado de las diez pensiones así como también el saldo deudor, el monto de las pensiones a pagar y el descuento por hermanos.
<b>Post condición</b>	El estudiante debe estar previamente registrado e inscrito en un curso.	
<b>Excepciones</b>	<b>PASO</b>	<b>ACCIÓN</b>
	<b>1</b>	Si el estudiante no está inscrito en ningún curso, este no podrá ingresar al sistema, el sistema devolverá el siguiente mensaje: “cedula de identidad incorrecto”.
<b>Comentarios</b>	El estudiante solo puede visualizar el estado de pensiones, y no así modificar estos, el sistema también devuelve los datos del estudiante como ser el nombre, cedula de identidad y apellidos de este.	

**Fuente:** Elaboración Propia.

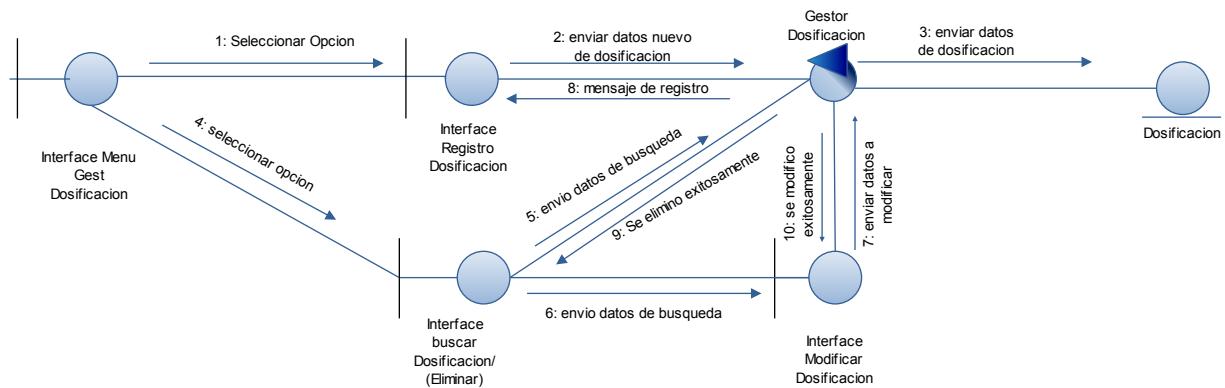
### 3.4.2. Diseño

A continuación se realizará los diagramas de colaboración, clases y diseño de la base de datos.

#### 3.4.2.1. Diagramas de colaboración

En la Siguiente Figura se muestra el diagrama de colaboración de la interfaz Gestión dosificación en la cual se encuentra las opciones de registro dosificación e buscar dosificación.

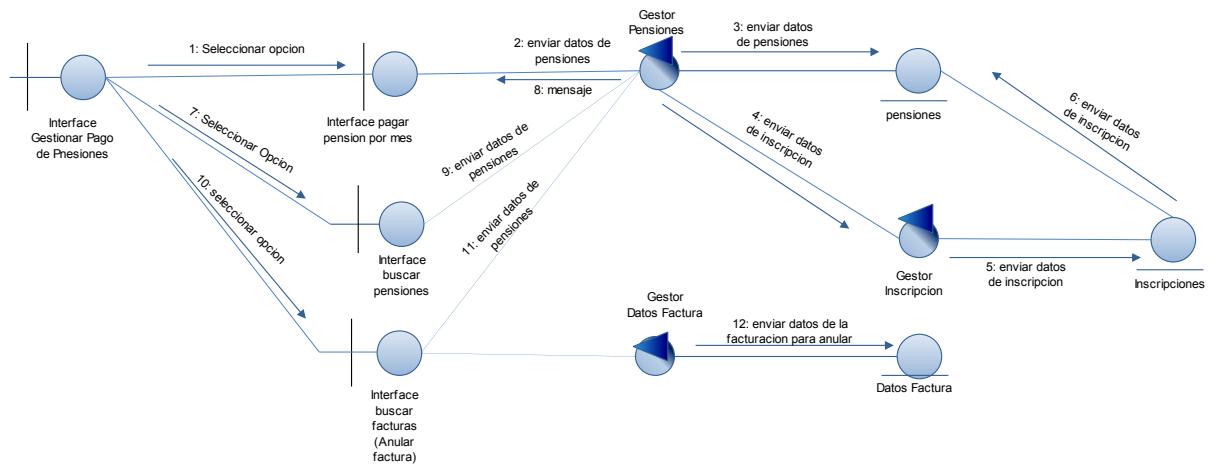
**Figura 75:** Diagrama de colaboración dosificación



**Fuente:** Elaboración Propia.

En la siguiente imagen se muestra la interfaz de pago de pensiones en la cual se encuentran las opciones de registrar pago de pensiones, buscar pensiones y buscar facturas, para la opción de registrar pago de pensiones es necesario enviar los datos de las pensiones, como también los datos de la inscripción de estudiante, para la búsqueda de pensiones es necesario el gestor de pensiones y por ultimo para la búsqueda de facturas es necesario la relación el gestor datos factura.

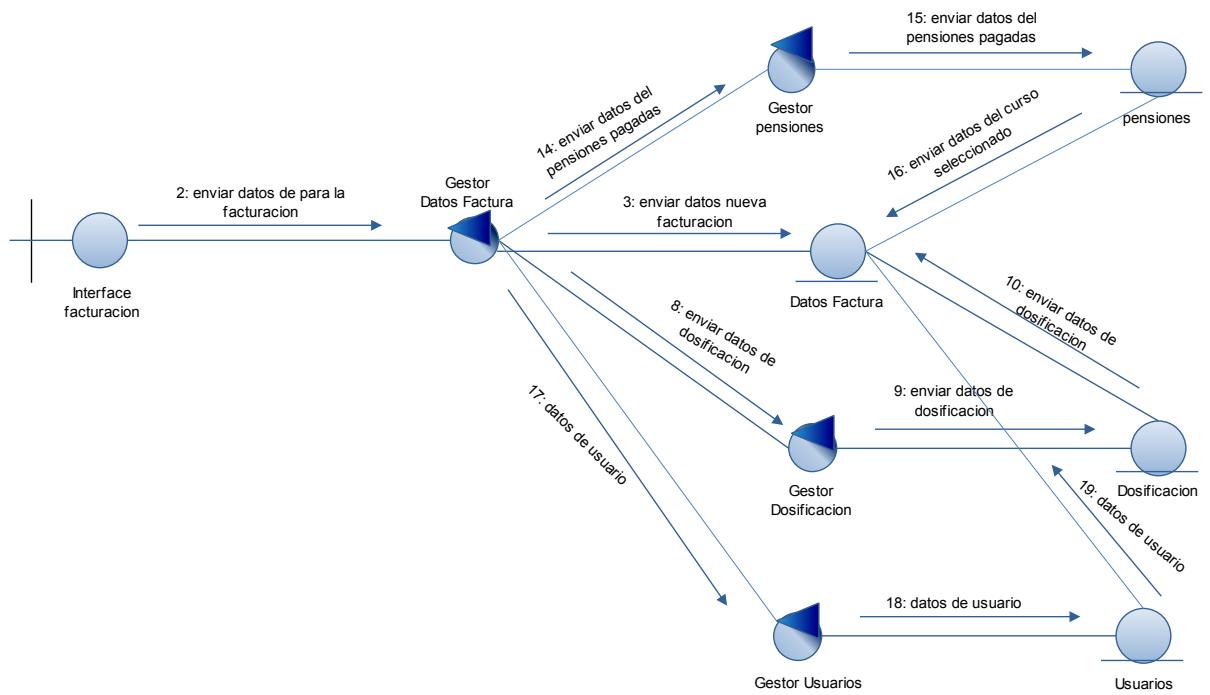
**Figura 76:** Diagrama de colaboración pago de pensiones.



**Fuente:** Elaboración Propia.

En la siguiente imagen se muestra la interfaz de facturación la cual tiene relación con tres gestores los cuales son gestor de pensiones, dosificación y usuarios.

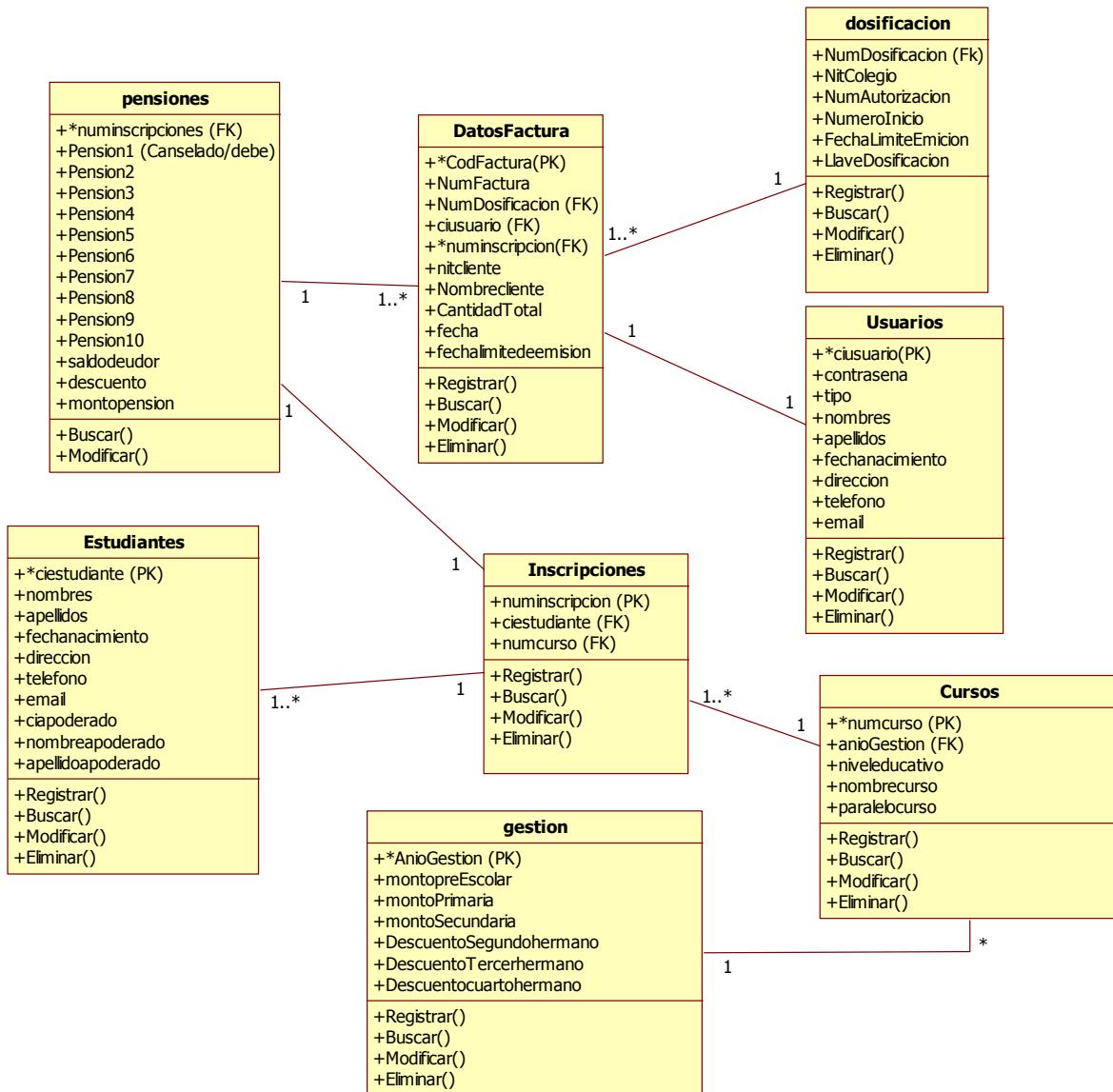
**Figura 77:** Diagrama de colaboración facturación.



**Fuente:** Elaboración Propia.

### 3.4.2.2. Diagrama de clases

Figura 78: Diagrama de clases.



Fuente: Elaboración Propia.

### 3.4.2.3. Diagrama de base de datos

**Figura 79:** Diagrama de base de datos.



**Fuente:** Elaboración Propia.

### 3.4.2.4. Diccionario de datos

A continuación se muestra el diccionario de datos de la tabla dosificación, la que tiene las siguientes columnas: nitcolegio, numautorizacion, numeroinicio, numerofinal y llavedosificacion:

**Tabla 28:** Diccionario de datos tabla dosificación.

Columna	Tipo de dato	Null	Llave primaria	Llave foránea	Descripción
nitcolegio	Int	No	Si	No	Registra el nit del colegio.
numautorizacion	Int	No	No	No	Registra el número de autorización del colegio.
numeroinicio	Int	No	No	No	Registra el número inicio de las facturas.
numerofinal	Int	No	No	No	Registra el número fin de las facturas.
llavedosificacion	Text	No	No	No	Registra la llave de dosificación que se usara para generar el código de control.

**Fuente:** Elaboración propia.

A continuación se muestra del diccionario de datos para la tabla pensiones, la que tiene las siguientes columnas: númerofactura, nitcolegio, numinscripcion, ciusuario, numautorizacion, nitcliente, nombrelcliente, cantidadtotal, fecha y fechahitedeemision:

**Tabla 29:** Diccionario de datos tabla pensiones.

Columna	Tipo de dato	Null	Llave primaria	Llave foránea	Descripción
numfactura	Int	No	si	no	Registra el número de facturación que se está realizando.
nitcolegio	Int	No	No	No	Registra el nit del colegio.
ciusuario	Int	No	No	No	Registra el ci del usuario que realizó la transacción.
numinscripcion	Int	No	No	No	Registra el número de inscripción.
numautorizacion	Int	No	No	No	Registra el número de autorización.
nitcliente	Int	No	No	No	Registra el NIT del cliente.
nombrecliente	Text	No	No	No	Registra el nombre o apellido del cliente.
cantidadtotal	Int	No	No	No	Registra la cantidad total de la transacción.
fecha	Text	No	No	No	Registra fecha del día que se realizó la transacción.
fechalimiteemision	Text	No	No	No	Registra la fecha límite de emisión de la factura.

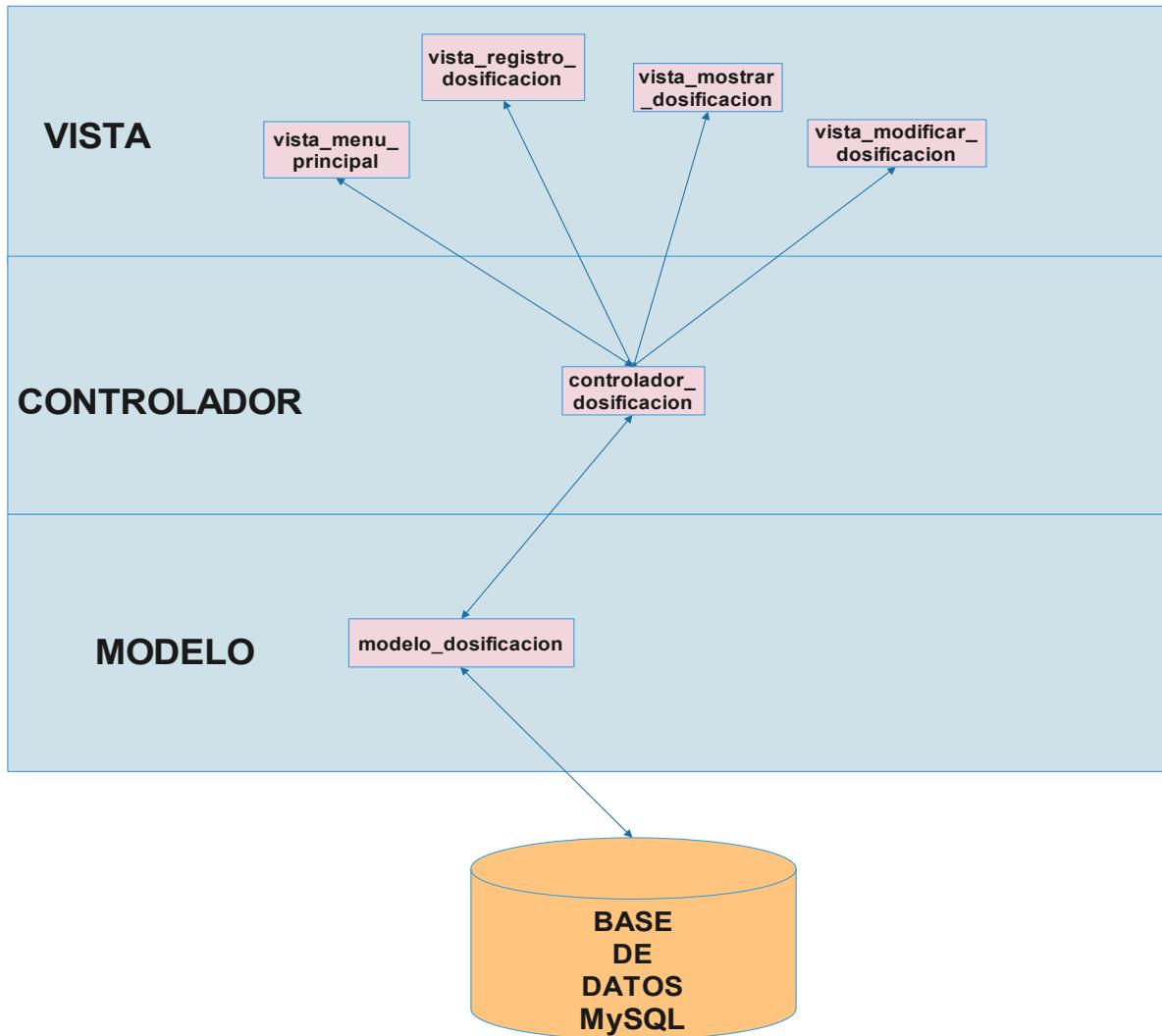
**Fuente:** Elaboración propia.

### 3.4.3. Implementación

#### 3.4.3.1. Arquitectura del software

A continuación se realizara las arquitecturas de los componentes necesarios para el funcionamiento de los casos de uso gestionar dosificación y gestionar pago de pensiones:

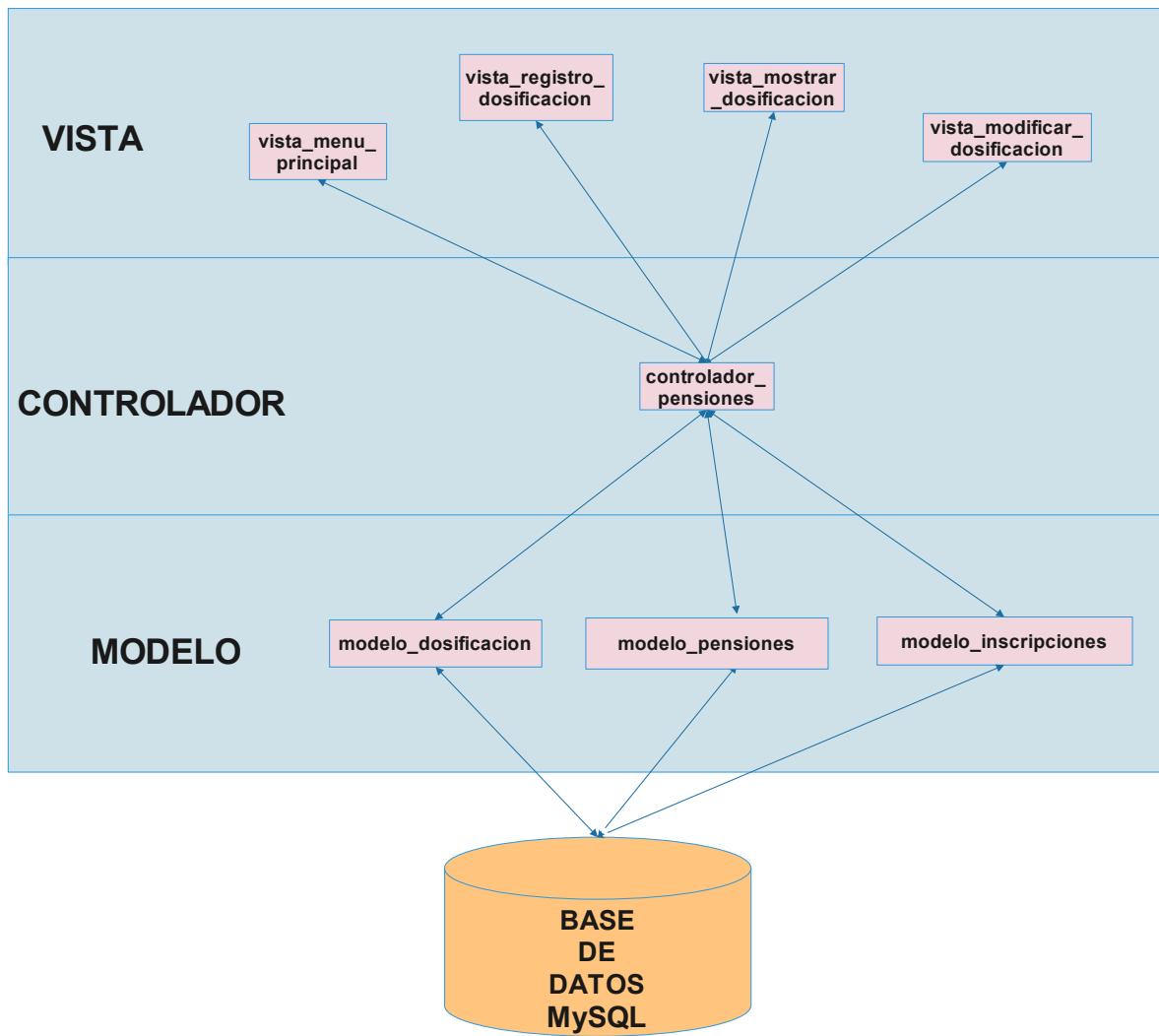
**Figura 80:** Arquitectura para el caso de uso gestionar dosificación.



**Fuente:** Elaboración propia.

A continuación se muestra la arquitectura de los componentes del sistema para llevar a cabo la gestión de pago de pensiones.

**Figura 81:** Arquitectura para el caso de uso pago de pensiones.



**Fuente:** Elaboración propia.

### 3.4.3.2. Organización del Proyecto

A continuación se muestra el contenido de la carpeta views con las respectivas vistas añadidas al proyecto.

**Figura 82:** Contenido de la carpeta views.

Name	Date modified	Type
vista_ingresar_sistema_usuarios	17-Nov-14 4:22 PM	PHP File
vista_menu_principal	14-Feb-15 5:24 AM	PHP File
vista_modificar_cursos	16-Nov-14 6:50 PM	PHP File
vista_modificar_dosificacion	11-Feb-15 6:51 PM	PHP File
vista_modificar_estudiantes	06-Dec-14 2:12 AM	PHP File
vista_modificar_gestion	14-Nov-14 9:45 PM	PHP File
vista_modificar_inscripciones_elegir_cursos	11-Feb-15 3:50 PM	PHP File
vista_modificar_usuarios	06-Dec-14 2:11 AM	PHP File
vista_mostrar_cursos	17-Nov-14 11:02 PM	PHP File
vista_mostrar_dosificacion	11-Feb-15 4:17 PM	PHP File
vista_mostrar_estudiantes	17-Nov-14 11:03 PM	PHP File
vista_mostrar_gestiones	17-Nov-14 11:01 PM	PHP File
vista_mostrar_inscripciones	11-Feb-15 2:37 PM	PHP File
vista_mostrar_usuarios	10-Feb-15 10:13 AM	PHP File
vista_pagina_principal	17-Nov-14 3:44 PM	PHP File
vista_registro_cursos	23-Oct-14 12:13 PM	PHP File
vista_registro_dosificacion	03-Feb-15 6:06 AM	PHP File
vista_registro_estudiantes	06-Dec-14 2:10 AM	PHP File
vista_registro_gestion	06-Dec-14 1:59 AM	PHP File
vista_registro_inscripciones_cursos	17-Nov-14 4:14 PM	PHP File
vista_registro_inscripciones_estudiantes	17-Nov-14 4:16 PM	PHP File
vista_registro_pensiones	11-Feb-15 9:41 PM	PHP File
vista_registro_pensiones_inscripciones	09-Feb-15 9:24 PM	PHP File
vista_registro_usuarios	06-Dec-14 2:01 AM	PHP File
vista_reportes_menu	14-Feb-15 9:38 AM	PHP File
welcome_message	29-May-14 4:13 PM	PHP File

**Fuente:** Elaboración propia.

A continuación se muestra el contenido de la carpeta controllers con los respectivos controladores añadidos al proyecto.

**Figura 83:** Contenido de la carpeta controllers.

Name	Date modified	Type
controlador_cursos	16-Nov-14 10:31 PM	PHP File
controlador_dosificacion	11-Feb-15 7:15 PM	PHP File
controlador_estudiantes	16-Nov-14 10:36 PM	PHP File
controlador_gestion	16-Nov-14 10:06 PM	PHP File
controlador_inscripciones	11-Feb-15 7:37 PM	PHP File
controlador_pensiones	14-Feb-15 5:08 AM	PHP File
controlador_reportes	14-Feb-15 5:32 AM	PHP File
controlador_usuarios	27-Sep-14 12:35 PM	PHP File
index	22-Nov-11 7:52 AM	Chrome HTML
welcome	12-Nov-14 4:54 PM	PHP File

**Fuente:** Elaboración propia.

A continuación se muestra el contenido de la carpeta models con los respectivos modelos añadidos al proyecto.

**Figura 84:** Contenido de la carpeta models.

Name	Date modified	Type	Size
index	22-Nov-11 7:52 AM	Chrome HTML Do...	1 KB
modelo_cursos	03-Feb-15 7:04 AM	PHP File	3 KB
modelo_dosificacion	11-Feb-15 7:12 PM	PHP File	2 KB
modelo_estudiantes	16-Nov-14 8:06 PM	PHP File	3 KB
modelo_gestion	14-Nov-14 9:36 PM	PHP File	3 KB
modelo_inscripciones	11-Feb-15 3:40 PM	PHP File	5 KB
modelo_pensiones	11-Feb-15 4:03 PM	PHP File	2 KB
modelo_usuarios	09-Feb-15 12:58 PM	PHP File	3 KB

**Fuente:** Elaboración propia.

También cabe decir que se aumentaron librerías para generar archivos pdf, código de control y código qr.

**Figura 85:** Contenido de la carpeta librerías.

Name	Date modified	Type
LibreriaCodigoControl	09-Feb-15 7:13 PM	File folder
LibreriaPDF	07-Feb-15 1:55 PM	File folder
LibreriaQR	14-Feb-15 4:34 AM	File folder

**Fuente:** Elaboración propia.

A continuación se muestra el contenido de la librería para pdf.

**Figura 86:** Contenido de la carpeta librería PDF.

Name	Date modified	Type	Size
doc	07-Feb-15 1:55 PM	File folder	
font	07-Feb-15 1:55 PM	File folder	
makefont	07-Feb-15 1:55 PM	File folder	
tutorial	07-Feb-15 1:55 PM	File folder	
changelog	18-Jun-11 3:26 PM	Firefox HTML Doc...	9 KB
FAQ	26-Oct-08 12:08 PM	Firefox HTML Doc...	16 KB
fpdf	19-Jul-08 3:04 PM	CSS File	2 KB
fpdf	18-Jun-11 1:07 PM	PHP File	46 KB
install	18-Jun-11 2:47 PM	Text Document	1 KB
license	03-Aug-08 10:52 A...	Text Document	1 KB

**Fuente:** Elaboración propia.

A continuación se muestra el contenido de la carpeta librería para la generación del código de control en la cual se encuentran los tres algoritmos para la generación de esta.

**Figura 87:** Contenido de la carpeta librería código control

Name	Date modified	Type
.gitignore	18-Jun-14 1:43 PM	GITIGNORE File
AllegedRC4	18-Jun-14 1:43 PM	PHP File
CodigoControlV7	18-Jun-14 1:43 PM	PHP File
LICENSE	18-Jun-14 1:43 PM	Text Document
README.md	18-Jun-14 1:43 PM	MD File
Verhoeff	18-Jun-14 1:43 PM	PHP File

**Fuente:** Elaboración propia.

A continuación se muestra la carpeta librería para la generación del código qr.

**Figura 88:** Contenido de la carpeta librería código QR.

Name	Date modified	Type	Size
qrcode	14-Feb-15 4:34 AM	File folder	
Ciqrcode	11-Dec-12 12:55 AM	PHP File	4 KB
README.md	11-Dec-12 12:55 AM	MD File	2 KB

**Fuente:** Elaboración propia.

### 3.4.3.3. Interfaces del sistema

A continuación se muestra la interface gestionar dosificación en la cual se encuentran las opciones de registrar dosificación, buscar dosificación y volver al menú principal.

**Figura 89:** Interface gestionar dosificación.



**Fuente:** Elaboración propia.

A continuación se muestra la interface de registro dosificación.

**Figura 90:** Interface registrar dosificación.



**Fuente:** Elaboración propia.

A continuación se muestra la interface de buscar dosificación en la cual se incluye las opciones de eliminar y modificar dosificación.

**Figura 91:** Interface buscar dosificación.

NIT COLEGIO	NUMERO DE AUTORIZACION	NUMERO INICIO	NUMERO FIN	LLAVE DOCIFICACION	MODIFICAR	ELIMINAR
1665979	7904006306693	876814	876850	zZ7Z]xssKqkEf_6K9uH(EcV+%x+u[Cca9T%+\$kIjT8(zr3T9b5Fx2xGD+_EBS	<a href="#">MODIFICAR</a>	<a href="#">ELIMINAR</a>

**Fuente:** Elaboración propia.

En la siguiente figura se muestra la interface del caso de uso pagar pensión en la cual se muestra todas las inscripciones de las cuales se elige al estudiante del cual se desea pagar la pensiones, también cabe decir que en esta interface incluye buscar estudiante por cedula de identidad y por año de gestión.

**Figura 92:** Interface de caso de uso pago pensiones buscar estudiante.

CI ESTUDIANTE	ANIOGESTION	CURSO	NIVEL	PARALELO	NUMERO HERMANO	PAGAR PENSION
5928026	2014	SEXTO	SECUNDARIA	A	1	<a href="#">PAGAR PENSIONES</a>
8052938	2015	SEXTO	SECUNDARIA	B	1	<a href="#">PAGAR PENSIONES</a>

**Fuente:** Elaboración propia.

Una vez ya elegido el estudiante del cual se desea pagar pensiones se muestra la siguiente interfaz (siguientes dos figuras), en la cual se muestran los datos del estudiantes como también de las pensiones a cancelar y las ya pagas. Cabe decir que en esta interface se incluye la opción de generar factura.

**Figura 93:** Interface de el caso de uso pagar pensiones.

COBRO DE PENSIONES

CI : 5928026

NOMBRES Richard

APELLIDOS Severich Gomes

NIT CLIENTE: 20202030 APELLIDO CLIENTE: Severich

GENERAL FACTURA

SALIR AL MENU PRINCIPAL

Detailed description: This is a screenshot of a user interface titled 'COBRO DE PENSIONES'. At the top, it displays the CI number '5928026' and the names 'Richard' and 'Severich Gomes'. Below this, there are input fields for 'NIT CLIENTE' (20202030) and 'APELLIDO CLIENTE' (Severich). Two buttons are present: 'GENERAL FACTURA' (in blue) and 'SALIR AL MENU PRINCIPAL' (in red).

**Fuente:** Elaboración propia.

**Figura 94:** Interface caso de uso pagar pensiones

SALDO DEUDOR	3670 Bs	
DESCUENTO	0 %	
MONTO PENSION	367 Bs	
PENSION 1	debe	PAGAR PENSION
PENSION 2	debe	PAGAR PENSION
PENSION 3	debe	PAGAR PENSION
PENSION 4	debe	PAGAR PENSION
PENSION 5	debe	PAGAR PENSION

Detailed description: This is a screenshot of a user interface showing a list of pension payments. It includes a summary row for 'SALDO DEUDOR' (3670 Bs) and 'DESCUENTO' (0 %). Below this, a table lists five pension entries, each with a status of 'debe' and a 'PAGAR PENSION' button. The table has three columns: the pension type, the status, and the payment button.

**Fuente:** Elaboración propia.

A continuación se muestra la factura generada con los tamaños y datos especificados y tipificados por la normativa de impuestos nacionales, como datos iniciales, datos de dosificación, datos de transacción y datos finales con el código de control y código QR, es necesario decir que cuando el usuario ingresa a la opción “Generar factura”, el sistema genera automáticamente un pdf de la factura la cual se podrá guardar o imprimir dependiendo de la necesidad del cliente.

**Figura 95:** Interface factura pdf.



**Fuente:** Elaboración propia.

### 3.4.4. Pruebas

A continuación se realizarán las pruebas funcionales y al código de control.

#### 3.4.4.1. Pruebas Funcionales

A continuación se muestra la tabla de resultados de las pruebas de funcionalidad.

**Tabla 30:** Pruebas funcionales del incremento.

Caso de prueba	Datos entrada	Resultado esperado	Resultado obtenido	Observaciones
Registrar Dosificaciones	Datos correcto	Registro exitosamente.	Se obtuvo el resultado esperado	Ninguna
Registrar Dosificación	Datos incorrectos como nit colegio repetida y datos inválidos.	Mensaje de error no se registró.	Se obtuvo el resultado esperado.	Ninguna
Cobro de pensiones selección estudiante.	Se seleccionó al estudiante de una lista de estudiantes.	Se muestra los datos del estudiante y las pensiones a pagar	Se obtuvo el resultado esperado	Ninguna
Cobro de pensiones	Se pagó pensiones.	Se muestra la misma ventana pero con el saldo deudor descontado.	Se obtuvo el resultado esperado.	Ninguna
Generar facturas	Se generó factura.	Se muestra la factura con las correspondientes pensiones pagadas y datos de la factura computarizada.	Se obtuvo el resultado esperado.	Ninguna

**Fuente:** Elaboración propia.

### 3.4.4.2. Pruebas al código de control

Para realizar pruebas al código de control se implementó un módulo extra en el cual se registran los siguientes datos para la generación del código de control.

**Figura 96:** Registro de datos código de control

AUTORIZACION:  
N FACTURA:  
NIT O CI:  
FECHA:  
MONTO:  
LLAVE:  
REGISTRAR

**Fuente:** Elaboración propia.

Posteriormente se probaron los cinco primeros casos de prueba otorgados por el servicio de impuestos nacionales en formato Excel, en la siguiente figura se muestran los resultados obtenidos.

**Figura 97:** Prueba del código de control.

NUM AUTORIZACION	NRO. FACTURA	NIT_CI	FECHA	MONTO	LLAVE	CODIGO DE CONTROL
7904006306693	876814	1665979	20080519	35959	zZ7Z]xssKqkEf_6K9uH(EcV+%x+u[Cca9T%+_\$kiLjT8(zr3T9b5Fx2xG-D+_EBS	7B-F3-48-A8
8004005263848	673173	1666188	20080810	51330	PNRU4cg27if]tr#]69j=yCS57i=uVZ\$rn@nv6wxarFP+AUp*T7Adiq3TT[Hw-@wt	2C-98-0E-96
7904006098968	165657	1666615	20080630	96459	m3dc5c)Dg#SN)prtK=9xn[rn+pgAxL%N67G)QfwNZM+)izCnvP\$T*qjEKhm]naDHm	AB-DD-0E-9B
7904004313753	826384	1666982	20080622	61103	Ebs[\$c2d2NCg5Fy@6nU5y##a5d)eDVz%jxW6bzcd)Kd)wl=c+)dZHneF#bcVL@	04-FF-52-67-CD
1904008691195	978256	0	20080201	26006	pPg(FS%v)@N4W3aQqqXCEHVS2[aDw_n%3)pFyU%bEB9)YXt%6xNBub4@PZ4S9)ct	62-12-AF-1B

**Fuente:** Elaboración propia

Seguidamente como se verificó que es un trabajo excesivo la inserción manual de los datos para las pruebas al código de control con el modulo creado se tuvo la necesidad de automatizar la prueba utilizando la librería PhpExcel para leer los datos directamente del archivo Excel e ingresarlos automáticamente al algoritmo.

El funcionamiento de esta automatización es el siguiente: primero se extraen los datos: número de autorización, número factura, NIT, fecha, monto y llave del archivo Excel y se los introduce a nuestro generador de código de control el cual devuelve el código de control generado, posteriormente se extrae el código de control del archivo en formato Excel y se realiza una comparación del código de control generado con el código de control obtenido del archivo Excel, seguidamente es señalando con la palabra éxito si son iguales y fallido si son diferentes.

Por último se concluye que los 5 mil casos de prueba salieron exitosamente, en la siguiente imagen se muestra la parte de los resultados de los 5 mil casos de prueba:

**Figura 98: Resultados de los 5 mil casos de prueba**

NRO PRUEBA	NRO AUTORIZACION	NRO FACTURA	NIT CI	FECHA	MONTO	LLAVE	CODIGO DE CONTROL GENERADO	CODIGO DE CONTROL EXISTOSO	FALLIDO
1	7904006306693	876814	1665979	2008/05/19	35958.60	zZ7Z]xssKqkEf_6K9uH(EcEV%+ox+u[Cea9T%+_SkIjT8(zr3T9b5Fx2xG-D+_EBS	7B-F3-48-A8	7B-F3-48-A8	EXITOSO
2	8004005263848	673173	1666188	2008/08/10	51330.00	PNRU4cgz7if]{tr#J69=j=yCS57i=uVZ\$u@nv6wxRFP+AUF*L7Adiq3TT[Hw-@wt	2C-98-0-E-96	2C-98-0-E-96	EXITOSO
3	7904006098968	165657	1666615	2008/06/30	96459.00	m3dcSe)Dg="#SN)prtK=9xn[m+p2AxL%N67G]QfwNZM+JzCnv:PST*qjEKhmJnaDHm	AB-DD-0E-9B	AB-DD-0E-9B	EXITOSO
4	7904004313753	826384	1666982	2008/06/22	61102.70	Ebs[\$c2d2NCg5FYj@6nU5y##a5d]eDVz%]xW6bzcd}Kd)w=e+d)ZHneF#bqVL@	04-FF-52-67-CD	04-FF-52-67-CD	EXITOSO
5	1904008691195	978256	0	2008/02/01	26006.00	pPgiFS%v}@N4W3aQqqXCEHVS2[aDw_n%3)pFyU%bEB9)YXt%oxNBu4@PZ4S9)ct	62-12-AF-1B	62-12-AF-1B	EXITOSO
6	1904002787435	996859	1671665	2008/05/28	16455.00	5J19cGiW\$r-w%+=7\$xnSnEjujWSpPUk=3SgHA6GHyBi^n(bZRwm)-_xQrzziFI@	7E-48-6E-CC	7E-48-6E-CC	EXITOSO
7	200400853354	742625	1672430	2007/04/23	4619.00	WT+uL)d Ie=IZpjQ!kr*py*B5gXD:A7DBy%-M q RnFrRGf[(xf%+NFWSBY{n%U	FB-1B-FE-71-6C	FB-1B-FE-71-6C	EXITOSO
8	8004009597732	897257	1675845	2008/11/10	70809.00	-C)B]zh]P%yUVnTaFL)9]2p]6cKh28RJu*KXUsu7%pqXSURQ)*#={Sd{nAs}--=J	F4-27-D8-1F	F4-27-D8-1F	EXITOSO
9	6004002578983	890986	1678842	2007/03/31	25089.49	hsKS KhKG-YmMGAsTUKN[CEYhEQFC8KS=4\$W19%uQGh[L]e78eF4V@{JxrFVqeh	EB-5E-52-57	EB-5E-52-76-57	EXITOSO
10	6004004541556	565398	1679177	2008/05/05	77256.60	#Vti+GE`sdsnsdHf4jjKtC*ImSWzyIsct{yqUGVmb)AEV]rxCSCua@#F*bR4-rti	F5-EB-EE-7D	F5-EB-EE-7D	EXITOSO
11	1004006752661	921113	1680744	2008/05/21	85419.00	dg(nz6#+pZlWaD5+A]Gz_gt73w=QciAb99A5W{8mKr2{5TQNwMa+JgneLs(Lm)Pw	13-6B-EC-3D-EA	13-6B-EC-3D-EA	EXITOSO
12	1004004682156	446338	1684254	2008/09/19	88940.00	vKrw9QxesAtIq6dz_Aq*(Q RyW]WUgzb6a8nXHJU)CAZM%*i9qa2vjYC3HDJ	8D-71-24-A5	8D-71-24-A5	EXITOSO
13	500400945523	867204	1686656	2007/10/13	14266.00	2k3(S)6SJ+_y5wcQUDJTrA6FE8@n5wWihVWHSZW@{c7Gm)u+7w=afdG4Es2j_*M]*	31-2B-A2-4E	31-2B-A2-4E	EXITOSO
14	7904001512631	857700	1689353	2007/05/31	84658.00	v+##uf)V+MnKe={Fbg=Unv4c]xN{MuXb)W%okVLI+H(bdiDrV@#i4GviBp*i+b	E1-A3-64-23-FB	E1-A3-64-23-FB	EXITOSO
15	800400944012	843713	1692823	2008/12/19	86733.49	{3Z*(YGI+MKmDV@p_+5LNwEKJlx6kd{jJ9pyI+rmyYjU[C434UCv546KG(f(zA5	46-74-EC-4D-7A	46-74-EC-4D-7A	EXITOSO
16	6004007615890	373578	1695741	2008/12/22	69811.00	[F2keCn@)MfnhS3f4=z74iBA)T.t4ei]PKoKf*DvPOBT.xNi;JN(øøZ3N43AI.Nøs3NA	8E-95-8B-8F-95-8B-38	8E-95-8B-8F-95-8B-38	EXITOSO

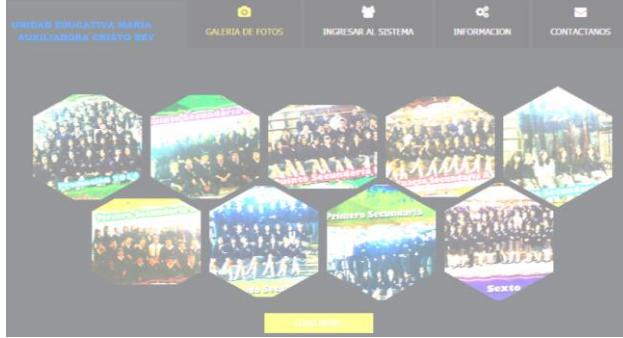
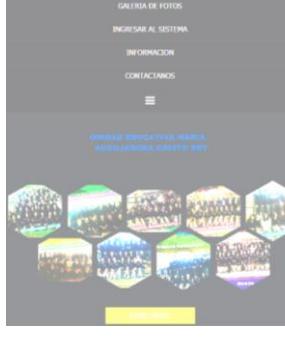
**Fuente:** Elaboración propia.

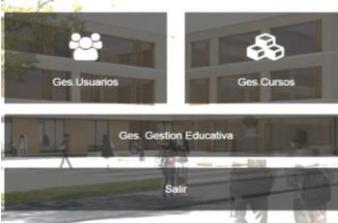
### 3.5. PRUEBAS FINALES AL SISTEMA MULTIPLATAFORMA

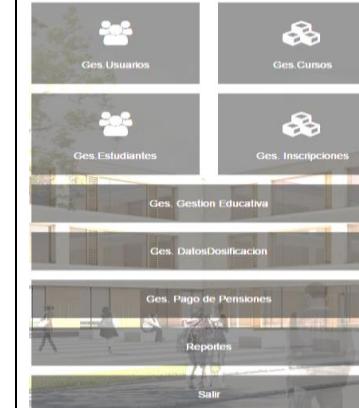
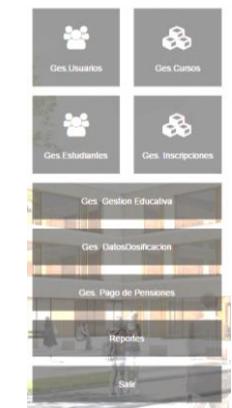
Como se había planteado y descrito en el capítulo uno respecto a objetivos específicos y acciones, al concluir el desarrollo del sistema se deben realizar las pruebas con distintos dispositivos (laptops, tablets, smartphones), con distintos sistemas operativos y con distintos navegadores. Por esta razón, a continuación se presentan los resultados de las pruebas ejecutadas.

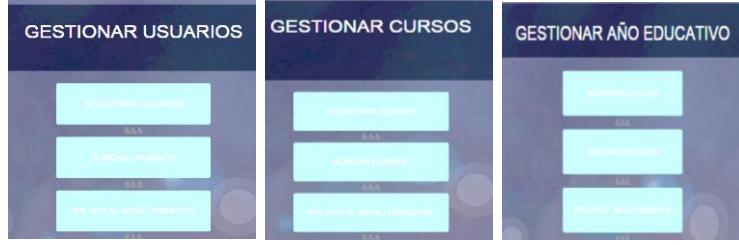
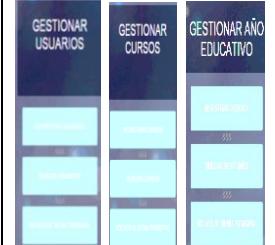
A continuación se muestra en una tabla los resultados del sistema visto en distintos dispositivos, navegadores y sistemas operativos:

**Tabla 31:** Tabla comparativa del sistema.

Caso de Prueba	Pc (Windows 8, explorer).	Tablet (iOS, Chrome).	Smartphone(Android, Firefox).
Pagina web principal (Galeria de Fotos)			

Caso de Prueba	Pc (Windows 8, explorer).	Tablet (iOS, Chrome).	Smartphone(Android, Firefox).
Pagina web principal (Ingresar al sistema Usuarios y estudiantes)	 <p>UNIDAD EDUCATIVA MARÍA AUXILIADORA CRISTO REY</p> <p>GALERIA DE FOTOS INGRESAR AL SISTEMA INFORMACIÓN CONTACTANOS</p> <p>USUARIOS ESTUDIANTES</p> <p>INGRESAR AL SISTEMA INGRESAR AL SISTEMA MODO USUARIO INGRESAR AL SISTEMA MODO ESTUDIANTE</p>	 <p>GALERIA DE FOTOS INGRESAR AL SISTEMA INFORMACIÓN CONTACTANOS</p> <p>UNIDAD EDUCATIVA MARÍA AUXILIADORA CRISTO REY</p> <p>USUARIOS ESTUDIANTES</p> <p>INGRESAR AL SISTEMA INGRESAR AL SISTEMA MODO USUARIO INGRESAR AL SISTEMA MODO ESTUDIANTE</p>	 <p>USUARIOS ESTUDIANTES</p> <p>INGRESAR AL SISTEMA INGRESAR AL SISTEMA MODO USUARIO INGRESAR AL SISTEMA INGRESAR AL SISTEMA MODO ESTUDIANTE</p>
Menu Principal del sistema (del Primer Incremento)	 <p>Ges. Gestión Educativa</p> <p>Ges. Usuarios Ges. Cursos Salir</p>	 <p>Ges. Gestión Educativa</p> <p>Ges. Usuarios Ges. Cursos Salir</p>	 <p>Ges. Gestión Educativa</p> <p>Ges. Usuarios Ges. Cursos Salir</p>

Caso de Prueba	Pc (Windows 8, explorer).	Tablet (iOS, Chrome).	Smartphone(Android, Firefox).
Menu Principal del sistema (del Segundo Incremento)			
Menu Principal del sistema (del Tercer Incremento)			

Caso de Prueba	Pc (Windows 8, explorer).	Tablet (iOS, Chrome).	Smartphone(Android, Firefox).
Gestionar Usuarios Cursos y año educativo.	 <p>The screenshot shows the Windows 8 Start screen with three large blue tiles. The first tile is labeled 'GESTIONAR USUARIOS' with sub-options 'REGISTRA ESTUDIANTES' (with 1000 icon), 'REGISTRA CURSOS' (with 1000 icon), and 'REGISTRA AÑO EDUCATIVO' (with 1000 icon). The second tile is labeled 'GESTIONAR CURSOS' with sub-options 'REGISTRA ESTUDIANTES' (with 1000 icon), 'REGISTRA CURSOS' (with 1000 icon), and 'REGISTRA AÑO EDUCATIVO' (with 1000 icon). The third tile is labeled 'GESTIONAR AÑO EDUCATIVO' with sub-options 'REGISTRA ESTUDIANTES' (with 1000 icon), 'REGISTRA CURSOS' (with 1000 icon), and 'REGISTRA AÑO EDUCATIVO' (with 1000 icon).</p>		 <p>The screenshot shows a mobile device screen with three large blue tiles. The tiles are identical to the ones in the PC screenshot, labeled 'GESTIONAR USUARIOS', 'GESTIONAR CURSOS', and 'GESTIONAR AÑO EDUCATIVO', each with three sub-options: 'REGISTRA ESTUDIANTES', 'REGISTRA CURSOS', and 'REGISTRA AÑO EDUCATIVO'. Below the tiles, the text 'Lo mismo que en pc.' is displayed.</p>
Gestionar Estudiantes, inscripciones y dosificacion	 <p>The screenshot shows the Windows 8 Start screen with three large blue tiles. The first tile is labeled 'GESTIONAR ESTUDIANTES' with sub-options 'REGISTRA ESTUDIANTES' (with 1000 icon) and 'REGISTRA CURSOS' (with 1000 icon). The second tile is labeled 'GESTIONAR INSCRIPCIONES' with sub-options 'REGISTRA INSCRIPCIONES' (with 1000 icon) and 'REGISTRA AÑO EDUCATIVO' (with 1000 icon). The third tile is labeled 'GESTIONAR DATOS DOSIFICACION' with sub-options 'REGISTRA DOSIFICACION' (with 1000 icon), 'REGISTRA DIFUSION' (with 1000 icon), and 'REGISTRA SEGUIMIENTO' (with 1000 icon).</p>		 <p>The screenshot shows a mobile device screen with three large blue tiles. The tiles are identical to the ones in the PC screenshot, labeled 'GESTIONAR ESTUDIANTES', 'GESTIONAR INSCRIPCIONES', and 'GESTIONAR DATOS DOSIFICACION', each with their respective sub-options. Below the tiles, the text 'Lo mismo que en pc' is displayed.</p>

Caso de Prueba	Pc (Windows 8, explorer).	Tablet (iOS, Chrome).	Smartphone(Android, Firefox).
Registro y búsqueda de Usuarios, cursos, Año Educativo, estudiantes, inscripciones, dosificación y pago de pensiones.	Lo mismo que en pc ya mostrado en la parte de codificación, interfaces del sistema en los respectivos incrementos.	Lo mismo que en pc	Lo mismo que en pc

**Fuente:** Elaboración propia.

Con el resultado presentado en la tabla anterior se ha demostrado que independiente del dispositivo, sistema operativo, navegador, características del hardware, tamaño de pantalla y resolución, las interfaces del sistema desarrollado se adoptan de forma automática sin ningún inconveniente.

### **3.6. DEMOSTRACIÓN DE LA HIPÓTESIS**

La hipótesis planteada en el presente proyecto menciona un “Sistema de gestión de pensiones con facturación computarizada”, el cual fue implementado para reducir el tiempo, costo y cantidad de errores, a continuación se realiza una breve descripción de cada uno de estos indicadores:

**Tiempo:** Se refiere al tiempo invertido en las diferentes actividades que implican la emisión de facturas y realizar consultas al estado de pensiones de los estudiantes.

**Gastos económicos:** Se refiere a los gastos económicos en la compra de talonarios y material de escritorio así como también el gasto que implica la adquisición de estos.

**Cantidad de errores:** Se refiere a la cantidad de errores cometidos en el llenado de las facturas emitidas, es decir la cantidad de errores que se pueden cometer por cada factura.

Para la demostración de la hipótesis es necesario analizar las variables dependientes e independientes del proyecto.

#### **3.6.1. Demostración de la primera variable dependiente**

Para la demostración de la variable dependiente “Tiempo de atención de los interesados” se hace uso de tiempos promedio invertidos por los interesados en cada actividad, obtenidas de una encuesta realizada a diferentes interesados desarrollada en el anexo “I”. Esto permitirá realizar el análisis del tiempo de pago de pensiones, proceso de facturación y consultas al estado de cuentas, para luego analizar la comparación de los tiempos que se invierten al realizar los procesos con sistema y sin sistema desarrollada en anexo “I”.

**Tabla 32:** Demostración de la primera variable dependiente

ACTIVIDAD	TIEMPO SIN SISTEMA	ACTIVIDAD	TIEMPO CON SISTEMA
<p>Averiguar estado de pensiones de los estudiantes.</p> <p>El interesado tiene que trasladarse a la unidad educativa, ingresar a la institución y brindar el ci del estudiante para que la secretaria lo busque y pueda brindar los datos.</p>	45 minutos	<p>El interesado ingresa al sistema desde cualquier pc, laptop, tablet o Smartphone con internet e ingresa a la página de la unidad educativa e inicia sesión con los datos del estudiante.</p>	2.5 minutos
<p>Pago de pensiones. El interesando se traspalada a la institución y brinda el ci del estudiante del cual requiere pagar pensiones, el cajero recibe el dinero y lo registra en el sistema.</p>	45 minutos	<p>Pago de pensiones. El interesando se traspalada a la institución y brinda el ci del estudiante del cual requiere pagar pensiones, el cajero recibe el dinero y lo registra en el sistema.</p>	45 minutos
<p>Facturación del pago realizado.</p> <p>El cajero pide NIT del cliente y apellido del cliente para emitir una factura escrita a mano utilizando un talonario pre</p>	10 minutos	<p>El cajero solo pide la primera vez el NIT y el apellido del cliente para emitir factura generada por el sistema.</p>	2 minutos

elaborado y sumando con calculadora el monto de varias pensiones si corresponde.			
TOTAL:	100 min		49.5 min

**Fuente:** Elaboración Propia.

Con los resultados obtenidos se obtuvo que:

100 minutos -> 100%

49.5 minutos -> 49.5%

Habiendo ya analizado el tiempo invertido en la realización proceso de gestión de cobro de pensiones con y sin el uso del sistema en la tabla anterior, se concluye que el tiempo invertido es menor con un 50.5%.

### **3.6.2. Demostración de la segunda variable dependiente**

Para la demostración de la variable independiente “Gastos económicos para la institución” se hace uso de costos invertidos donde se asignan valores en bolivianos invertidos en cada actividad, obtenidos de una entrevista con la secretaría de la institución desarrollado en el anexo “I”. Esto permitirá realizar el análisis del proceso de facturación, para luego analizar la comparación de los costos que se invierten al realizar los procesos con sistema y sin sistema.

Como existen 591 estudiantes actualmente se necesita 5910 facturas considerando que cada estudiante factura una vez por mes, y sabiendo que cada 100 facturas tiene un precio de 100 bs. Además, en ocasiones, al emitir la factura llenada a mano, se cometen errores y la respectiva factura es anulada y se emite otra.

También cabe decir que se tomaron en cuenta los gastos económicos en pasajes que usa el personal de la institución para dirigirse a la imprenta y realizar la impresión de

facturas correspondiente así como también los gastos en el material de escritorio al momento de realizar la facturación manual.

Por otro lado cabe decir que la institución cuenta con un ordenador de escritorio el cual será utilizado para el funcionamiento del sistema por lo tanto su costo es cero.

**Tabla 33:** Demostración de la segunda variable dependiente

ACTIVIDAD	SIN SISTEMA	ACTIVIDAD	CON SISTEMA
Gastos económicos en la compra de talonarios	5910 bs	Gastos económicos en la compra de impresora	3132 bs
Gastos económicos en los pasajes de la persona que va a comprar los talonarios	100 bs	Gastos económicos en tinta	200 Bs
Gastos económicos en material de escritorio	30 bs	Gatos económicos en papel membreteado para la impresión de la factura	250 bs
Gastos económicos en la pc de escritorio.	0 bs	Gastos económicos en la pc de escritorio.	0 bs
TOTAL	5990 bs		3582

**Fuente:** Elaboración Propia.

Habiendo analizada ya el costo invertido en la realización de la facturación con y sin sistema se concluye que en el primer año del funcionamiento del sistemas los gastos se reducen en un 60%. Cabe resaltar que en los años siguientes no se requerirá la adquisición de una impresora por lo cual la reducción de gastos será aún mayor.

### **3.6.3. Demostración de la tercera variable dependiente**

En la demostración de la variable dependiente “Errores en el llenado de facturas” se hace uso del número de errores en las facturas, donde se asignan un número de errores determinado, obtenido por entrevistas a la secretaría de la institución vistas en el anexo “I”, que puede suscitarse en cada actividad. Esto permitirá realizar el análisis de errores en la facturación del cobro de pensiones, para luego analizar la comparación con la facturación computarizada.

Sin sistema son 10 facturas que se pueden llenar por estudiante y en cada factura se puede cometer 3 tipos de errores (error en el NIT del cliente, apellido cliente y en el monto).

Con sistema también son 10 facturas que se pueden facturar por estudiante, pero el NIT y apellido del cliente solo se pide una vez por lo que la posibilidad de cometer errores es de 2 por cada factura.

**Tabla 34:** Demostración de la tercera variable dependiente

ACTIVIDAD	SIN SISTEMA	ACTIVIDAD	CON SISTEMA
Cantidad de errores en el NIT del cliente.	10	Cantidad de errores en el nit	1
Cantidad de errores en el apellido del cliente.	10	Cantidad de errores en el apellido del cliente	1

Cantidad de errores en el monto de mensualidades.	10	Cantidad de errores en el monto de mensualidades	0
Cantidad de facturas anuladas	25	Cantidad de facturas anuladas	1
TOTAL	55		3

**Fuente:** Elaboración Propia.

Habiendo analizado ya la cantidad de errores cometidos en la realización de la facturación con y sin sistema se concluye que los errores ocasionados sin sistema son mayores.

### 3.6.4. Demostración de la variable independiente

En la demostración de la variable independiente, se realiza el análisis a través de una muestra de todos los beneficios que tiene el sistema propuesto, y los procedimientos manuales empleados sin el uso del sistema, para el proceso de gestión de cobro de pensiones, calificando como un beneficio con la palabra “Sí”, y rechazando el beneficio con la palabra “No” en la siguiente tabla:

**Tabla 35:** Demostración e variable independiente

Nro. Factor	Factores Tomados en Cuenta Como Beneficio	Proceso Manual (Antes)	Proceso con el Sistema (Ahora)
1	Permite almacenar datos de la factura en el sistema.	No	Sí
2	Permite facturación computarizada.	No	Si

Nro. Factor	Factores Tomados en Cuenta Como Beneficio	Proceso Manual (Antes)	Proceso con el Sistema (Ahora)
3	Permite conceder permisos al personal	No	Sí
4	Permite reducir los gastos económicos que genera la facturación manual.	No	Si
5	Permite ejecución multiplataforma.	No	Si
6	Permite a los estudiantes ingresar al sistema para a verificar el estado de pensiones.	No	Sí
7	Permite que la encargada de entrega de calificaciones tenga un acceso inmediato al estado de pensiones del estudiante.	No	Si
8	Permite reportes de estudiantes con deudas	Si	Si
9	Permite procesar reportes de ingresos diarios.	Si	Si
<b>TOTAL BENEFICIOS</b>		<b>1 (Sí)</b>	<b>9 (Sí)</b>

**Fuente:** Elaboración Propia.

Teniendo como resultado del análisis de estas muestras, se concluye que el total de beneficios por el proceso manual tiene la cantidad de 2, y con el proceso del sistema es de 9. Estos resultados se tomarán en cuenta en el cálculo de estadístico T para realizar la aceptación o rechazo de la hipótesis que está detallado más adelante en el documento.

### 3.6.5. Definición de la hipótesis

Para la definir la hipótesis se considera el análisis de aceptación y rechazo de la misma, haciendo referencia a un  $P_n$  que representa una probabilidad, en nuestro caso tenemos que:

- $P_1$  = Los procedimientos manuales utilizados para realizar el proceso de gestión de pensiones.
- $P_2$  = El sistema de gestión de pensiones con facturación computarizada.

Y para los factores que influyen en la operativización son:

- Hipótesis nula: Los factores de éxito de antes son iguales a los factores de éxito posteriores:

$$H_0: P_1 = P_2$$

- Hipótesis alternativa: Los factores de éxito de antes son distintos a los factores de éxito posteriores:

$$H_1: P_1 <> P_2$$

Presenta dos casos, los cuales son:

- Los factores de éxito anteriores son mayores a los posteriores:  $P_1 > P_2$
- Los factores de éxito anteriores son menores a los posteriores:  $P_1 < P_2$

Y en nuestro proyecto tenemos que:

- $H_0: P_1 = P_2$  : (Los procedimientos manuales utilizados en la actualidad para el proceso de gestión de pensiones es igual al sistema de pensiones con facturación computarizada).

- $H_1: P_1 <> P_2$  : (Los procedimientos manuales utilizados en la actualidad para el proceso de gestión de pensiones es diferente al sistema de gestión de pensiones con facturación computarizada).
- $P_1 > P_2$  : (Los procedimientos manuales utilizados en la actualidad para el proceso de gestión de pensiones tiene mayores beneficios que el sistema de gestión de pensiones con facturación computarizada).
- $P_1 < P_2$  : (Los procedimientos manuales utilizados en la actualidad para el proceso de gestión de pensiones tiene menores beneficios que el sistema de gestión de pensiones con facturación computarizada).

### 3.6.6. Cálculo del estadístico t

Este cálculo del estadístico t permite conocer la aceptación de la hipótesis a partir de un dato obtenido de una serie de cálculos, teniendo en cuenta que existe un rango de aceptación en la campana de Gauss. Seguidamente con el dato encontrado y con las probabilidades se ubica si está dentro del rango de aceptación o rechazo. Para ello se realiza los cálculos con las siguientes fórmulas:

$$\triangleright P_n = \frac{x_n}{n}$$

$$\triangleright t = \frac{P_1 - P_2}{\sqrt{\frac{P_1 * Q_1}{n_1} + \frac{P_2 * Q_2}{n_2}}}$$

$$\triangleright Q = 1 - P_n$$

Dónde:

- $P_n$  = Probabilidad de ocurrencia de uno de los casos

- $X_n$  = Número de beneficios
- $n$  = Número de muestras (beneficios analizados de la Tabla anterior)
- $t$  = Valor estadístico para la aceptación o rechazo de la hipótesis
- $Q_n$  = Cantidad de valores que no representan beneficio.

$$n = 9 \quad x_1 = 2 \quad x_2 = 9$$

$P_1 = \frac{2}{9} = 0.2222 \rightarrow P_1$  Alcanzo un 11% de los beneficios de la muestra.

$P_2 = \frac{9}{9} = 1 \rightarrow P_2$  Alcanzo un 100% de los beneficios de la muestra.

$P_1 < P_2 \rightarrow$  Por lo tanto la  $P_2$  es mejor que la  $P_1$ .

$$Q_1 = 1 - p_1 \Rightarrow 1 - \frac{2}{9} \Rightarrow \frac{7}{9} \Rightarrow 0.8889$$

$$Q_2 = 1 - p_1 \Rightarrow 1 - \frac{9}{9} \Rightarrow 1 - 1 \Rightarrow 0$$

$$P_1 = \frac{2}{9} = 0.1111 \quad \text{Entonces} \quad Q_1 = 1 - \frac{2}{9} = 0.7778$$

$$P_2 = \frac{9}{9} = 1 \quad \text{Entonces} \quad Q_2 = 1 - 1 = 0$$

$$t = \frac{\frac{2}{9} - 1}{\sqrt{\frac{\frac{2}{9} * \frac{7}{9}}{9} + \frac{1 * 0}{9}}} = -5.6125$$

Ahora sacamos los grados de libertad

$$t_{\frac{\alpha}{2}, n-1}$$

Donde:

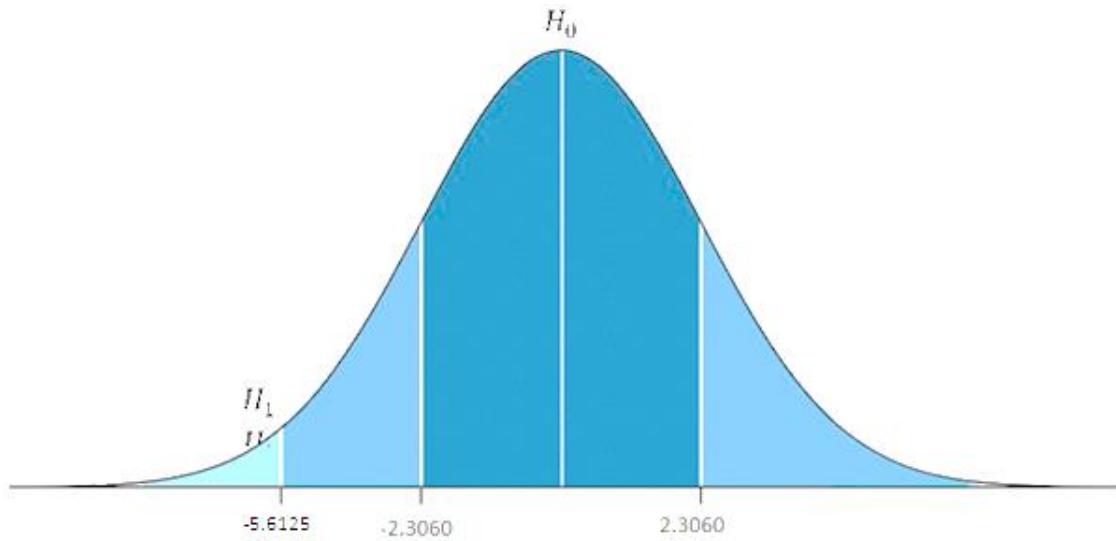
$\alpha = \text{Nivel de Significancia}$

$n = \text{Tamaño de la muestra}$

Grado de Error = 5% entonces

$$t_{\frac{\alpha}{2}, n-1} = t_{\frac{0.05}{2}, 9-1} = t_{0.025, 8}$$

**Figura 99:** Campana de Gauss para la aceptación de la hipótesis



**Fuente:** Elaboración Propia.

Después de todo el análisis del cálculo del estadístico t y por la campana de gauss elaborada, se puede indicar que se acepta la hipótesis, concluyendo que la hipótesis  $H_1$  queda demostrada, el cual dice: “El sistema de gestión de pensiones con

facturación computarizada en el área administrativa de la Unidad Educativa María Auxiliadora Cristo Rey permitirá reducir el tiempo en la atención de los interesados, gastos económicos para la institución y errores en el llenado de facturas”, Y que con el valor de éxito obtenido  $P_1 < P_2$ , se dice que “Los procedimientos manuales utilizados en la actualidad para el proceso de gestión de pensiones actual tiene menores beneficios que el sistema de gestión de pensiones con facturación computarizada”.

## **4. ANÁLISIS DE VIABILIDAD**

### **4.1. VIABILIDAD TÉCNICA.**

Para la puesta en marcha del sistema de gestión de pensiones con facturación computarizada es necesario buscar tecnología que pueda satisfacer los requerimientos de funcionamiento en componentes de hardware y software, para obtener una buena implementación sin inconvenientes se debe cumplir con los requerimiento mínimo o ideal.

Cabe decir que para el alojamiento del sistema se usara un servidor en la nube ya que es más económico que comprar un servidor físico, por otro lado este servidor en la nube tiene versión gratuita y de paga, En el siguiente cuadro se muestran los requerimientos mínimos e ideales del servidor en la nube:

**Tabla 36:** Requerimientos del servidor.

<b>Nombre</b>	<b>Requerimiento Mínimo</b>	<b>Requerimientos Ideal</b>	<b>Parte</b>
Servidor en la nube	Servidor gratuito www.000webhost.com.	Servidor gratuito www.000webhost.com.	Software
Lenguaje de Programación	Php versión 4.x.x	Php versión 4.x.x o superior	Software
Gestor de Base de Datos	My SQL	My SQL	Software

**Fuente:** Elaboración Propia.

En el siguiente cuadro se muestra los requerimientos mínimos y recomendados de las máquinas de los usuarios tanto como para hardware y software.

**Tabla 37:** Requerimientos para el equipo-usuarios

Nombre	Requerimiento Mínimo	Requerimientos Ideal	Parte
Sistema Operativo	Windows XP Professional	Windows 8.1	Software
Navegador	Crome, Firefox o Explorer	Crome, Firefox o Explorer	Software
Procesador	Intel pentium IV 1 Ghz o Equivalente	Intel i3 o equivalente en amd, o superior	Hardware
Memoria Ram	512 MB	4096 MB o superior	Hardware
Tarjeta de Video	32 MB	1024 MB	Hardware
Disco Duro	20 GB	80 GB o superior	Hardware
Impresora para factura computarizada	Epson TM-U220 o equivalente en otras marcas.	Epson TM-U220 o superior, equivalente en otras marcas.	Hardware
Teclado, ratón y case	Delux	Delux	Hardware

**Fuente:** Elaboración Propia.

Para el uso a través de dispositivos como tablets o smartphones, se lista a continuación los requerimientos mínimos y recomendados para el funcionamiento del sistema.

**Tabla 38:** Requerimientos para la tablet o smartphone

Nombre	Requerimiento Mínimo	Requerimientos Ideal	Parte
Sistema Operativo	Android 4.0 o equivalente en apple.	Android 4.0 o superior, equivalente en apple.	Software
Navegador	En Android Browser, Opera, Crome, Firefox, en el caso de iOS (safari).	En Android Browser, Opera, Crome, Firefox, en el caso de iOS (safari).	Software
Procesador	STE U8500 Sony Xperia o equivalente	SnapDragon 200 Quad-core o superior	Hardware
Memoria Ram	512 MB	1024 MB	Hardware
Tarjeta de Video	256 MB	1024 MB o superior	Hardware
Disco	4 GB	4 GB o superior	Hardware

**Fuente:** Elaboración Propia.

Analizando los requerimientos mínimos tanto de software como de hardware que se necesita para el “Servidor” y para los “Usuarios”, se concluyó que los requerimientos pueden ser obtenidos libremente y la tecnología requerida existe en nuestro medio, por lo tanto se determinó que el proyecto es técnicamente viable.

## 4.2. VIABILIDAD ECONÓMICA.

A continuación se realizará el análisis de costos que se necesita para los equipos con los requerimientos funcionales del hardware y software, además del costo que se tiene en la producción de programación y levantamiento del sistema.

Por otro lado cabe decir que para el levantamiento del sistema se utilizó el un hosting en la nube es decir un servidor en la nube el cual tiene versión de paga y gratuito, a continuación se muestran los detalles del servidor:

**Tabla 39:** Costo del servidor

Nombre	Requerimiento Mínimo	Costo (Dólares)	Requerimientos Ideal	Costo (Dólares)
Sistema Operativo	Servidor gratuito <a href="http://www.000webhost.com">www.000webhost.com</a>	0,00.-	Servidor de paga <a href="http://www.000webhost.com">www.000webhost.com</a>	62,0.-
Lenguaje de Programación	Php versión 4.x.x	0,00.-	Php versión 4.x.x o superior	0,00.-
Manejador de Base de Datos	My SQL.	0,00.-	My SQL	0,00.-
<b>TOTAL (\$)</b>		<b>0,00.-</b>		<b>62,00.-</b>

**Fuente:** Elaboración Propia.

A continuación se muestra los costos del equipo de los usuarios el cual incluye la impresora para la impresión de la facturación.

**Tabla 40:** Costos del equipo-usuarios

Nombre	Requerimiento Mínimo	Costo (Dólares)	Requerimientos ideal	Costo (Dólares)
Sistema Operativo	Windows XP Professional Service Pack 2	50,00.-	Windows 8.1	100,00.-
Navegador	Crome, Firefox o Explorer.	0,00.-	Crome, Firefox o Explorer.	0,00.-
Procesador	Intel pentium IV 1 Ghz o Equivalente en Amd.	20,00.-	Intel i3 o equivalente en amd, o superior	100,00.-
Memoria Ram	512 MB	30,00.-	4096 MB o superior	100,00.-
Tarjeta de Video	32 MB	30,00.-	1024 MB	80,00.-
Disco Duro	20 GB	20,00.-	80 GB o superior	40,00.-
Impresora para factura computarizada	Epson TM-U220 o equivalente en otras marcas.	450,00.-	Epson TM-U220 o superior,	450,00.-

Nombre	Requerimiento Mínimo	Costo (Dólares)	Requerimientos ideal	Costo (Dólares)
			equivalente en otras marcas.	
Teclado, ratón y case	Delux	50,00.-	Delux	50,00.-
<b>TOTAL (\$)</b>		<b>650,00.-</b>		<b>920,00.-</b>

**Fuente:** Elaboración Propia.

A continuación se muestra los costos del equipo de los estudiantes que ingresan al sistema con un dispositivo móvil.

**Tabla 41:** Costo de las tablets o smartphones

Nombre	Requerimiento Mínimo	Costo (Dólares)	Requerimientos Ideal	Costo (Dólares)
Sistema Operativo	Android 4.0 o equivalente.	0,00.-	Android 4.0 o equivalente.	0,00.-
Navegador	En Android Browser, Opera, Crome, Firefox, en el caso de iOS (safari).	0,00.-	En Android Browser, Opera, Crome, Firefox, en el caso de iOS (safari).	0,00.-

Nombre	Requerimiento Mínimo	Costo (Dólares)	Requerimientos Ideal	Costo (Dólares)
Procesador	STE U8500 Sony Xperia o equivalente	50,00.-	SnapDragon 200 Quad-core o superior	200,00.-
Memoria Ram	512 MB	50,00.-	1024 MB	100,00.-
Tarjeta de Video	256 MB	50,00.-	1024 MB o superior	100,00.-
Disco	4 GB	30,00.-	4 GB o superior	30,00.-
<b>TOTAL (\$)</b>		<b>180,00.-</b>		<b>430,00.-</b>

**Fuente:** Elaboración Propia.

Para encontrar el costo del trabajo realizado en la programación del proyecto se utiliza COCOMO Versión 1.0 (Constructive Cost Model, COCOMO BÁSICO) que se detalla a continuación:

Los Modelos COCOMO están definidos para tres tipos de proyectos de software. Utilizando la terminología de Boehm, son:

**Modelo Orgánico:** Proyectos de software relativamente pequeños y sencillos donde trabajan pequeños equipos, con buena experiencia en la aplicación, sobre un conjunto de requisitos poco rígidos (por ejemplo, un programa de análisis termal desarrollado para un grupo calorífico).

**Modelo Semiacoplado:** Proyectos de software intermedios en los que equipos, con variados niveles de experiencia, deben satisfacer requerimientos poco o medio rígidos.

**Modelo Empotrado:** Proyectos de software que deben ser desarrollados en un conjunto de hardware, software y restricciones operativas y muy restringidas.

Los datos del modelo orgánico (marcado con una flecha en la siguiente tabla) serán utilizados en el presente proyecto, ya que es el modelo que se usa para proyectos pequeños y sencillos, COCOMO hace referencia a tres modelos de desarrollo de proyectos que influyen directamente en la duración y costo del mismo:

**Tabla 42:** Valores constantes

Proyectos Software	A	B	C	D
Orgánico	3,2	1,05	2,5	0,38
Semi-acoplado	3,0	1,12	2,5	0,35
Empotrado	2,8	1,20	2,5	0,32



**Fuente:** Elaboración Propia.

Inicialmente se debe obtener el cálculo de la cantidad de líneas de código utilizadas en el proyecto en los modelos, vistas y controladores.

## MODELOS

- Modelo\_cursos = 300 líneas de código
- Modelo\_dosificacion = 260 líneas de código
- Modelo\_estudiantes = 240 líneas de código
- Modelo\_gestion = 420 líneas de código
- Modelo\_inscripciones = 480 líneas de código

- Modelo\_pruebas = 120 líneas de código
- Modelo\_usuarios = 260 líneas de código

## VISTAS

- Vista\_gestionar\_cobro\_pensiones= 80 líneas de código
- Vista\_ingresar\_sistema\_estudiantes= 130 líneas de código
- Vista\_ingresar\_sistema\_usuarios= 150 líneas de código
- Vista\_menu\_principal = 495 líneas de código
- Vista\_menu\_principal\_administrador= 455 líneas de código
- Vista\_menu\_principal\_cajero = 440 líneas de código
- Vista\_menu\_principal\_secretaria = 450 líneas de código
- Vista\_modificar\_cursos = 120 líneas de código
- Vista\_modificar\_dosificacion = 140 líneas de código
- Vista\_modificar\_estudiantes = 215 líneas de código
- Vista\_modificar\_gestion = 175 líneas de código
- Vista\_modificar\_inscripciones\_elegir\_cursos = 110 líneas de código
- Vista\_modificar\_usuarios = 295 líneas de código
- Vista\_mostrar\_cursos = 100 líneas de código
- Vista\_mostrar\_cursos\_estudiantes = 85 líneas de código
- Vista\_mostrar\_dosificaciones = 95 líneas de código
- Vista\_mostrar\_estudiantes= 115 líneas de código
- Vista\_mostrar\_facturas = 165 líneas de código
- Vista\_mostrar\_gestiones = 110 líneas de código
- Vista\_mostrar\_inscripciones = 110 líneas de código
- Vista\_mostrar\_pensiones = 130 líneas de código
- Vista\_mostrar\_pensiones\_estudiantes = 200 líneas de código
- Vista\_mostrar\_usuarios = 110 líneas de código
- Vista\_pagina\_principal = 740 líneas de código

- Vista\_pruebas\_codigo\_control = 250 líneas de código
- Vista\_pruebas\_menu = 70 líneas de código
- Vista\_registro\_cursos = 125 líneas de código
- Vista\_registro\_dosificacion = 135 líneas de código
- Vista\_registro\_estudiantes = 205 líneas de código
- Vista\_registro\_gestion = 165 líneas de código
- Vista\_registro\_inscripciones\_cursos = 105 líneas de código
- Vista\_registro\_inscripciones\_estudiantes = 130 líneas de código
- Vista\_registro\_pensiones = 505 líneas de código
- Vista\_registro\_pensiones\_inscripciones = 150 líneas de código
- Vista\_registro\_usuarios = 180 líneas de código
- Vista\_reportes\_diarios = 60 líneas de código
- Vista\_reportes\_elegir\_cursos = 99 líneas de código
- Vista\_reportes\_estudiantes\_deudores\_por\_curso = 110 líneas de código
- Vista\_reportes\_ingresos = 60 líneas de código
- Vista\_reportes\_menu\_administrador = 70 líneas de código
- Vista\_reportes\_menu\_secretaria = 65 líneas de código

## **CONTROLADORES**

- Controlador\_cursos = 360 líneas de código
- Controlador\_dosificacion = 340 líneas de código
- Controlador\_estudiantes = 360 líneas de código
- Controlador\_gestione = 340 líneas de código
- Controlador\_incripcion = 790 líneas de código
- Controlador\_pensiones = 1480 líneas de código
- Controlador\_pruebas = 190 líneas de código
- Controlador\_reportes = 290 líneas de código
- Controlador\_usuarios = 450 líneas de código

**Total Líneas de código de los Modelos = 2120 líneas de código.**

**Total Líneas de código de las vistas = 7699 líneas de código.**

**Total Líneas de código de las controladores = 4600 líneas de código.**

**TOTAL LÍNEAS DE CÓDIGO= 14 419 líneas de código.**

$$KLCD = \text{lineas de codigo}/1000$$

$$KLCD = 14\,419 / 1000$$

$$KLCD = 14,419$$

En la siguiente tabla se muestran los resultados de los factores de costos cocomo utilizadas en el cálculo del costo de desarrollo de software las cuales fueron elegidas según a las justificaciones desarrolladas en el Anexo J.

**Tabla 43: Factores de costo en COCOMO**

	FACTORES DE COSTO		MUY BAJO	BAJO	MEDIO	ALTO	MUY ALTO	EXTRA ALTO
ATRIBUTOS DEL PRODUCTO	Fiabilidad Requerida del Software	RELY	0,75	0,88	1	1,15	1,4	-
	Tamaño de la Base de Datos	DATA	-	0,94	1	1,08	1,16	-
	Complejidad del Producto	CPLX	0,7	0,85	1	1,15	1,3	1,65
ATRIBUTOS DEL COMPUTADOR	Restricciones del Tiempo de Ejecución	TIME	-	-	1	1,11	1,3	1,66
	Restricciones del Almacenamiento Principal	STOR	-	-	1	1,06	1,21	1,56
	Volatilidad de la Máquina Virtual	VIRT	-	0,87	1	1,15	1,3	-
	Tiempo de respuesta del ordenador	TURN	-	0,87	1	1,07	1,15	-

	FACTORES DE COSTO		MUY BAJO	BAJO	MEDIO	ALTO	MUY ALTO	EXTRA ALTO
ATRIBUTOS DEL PERSONAL	Capacidad del Analista	ACAP	1,46	1,19	1	0,86	0,71	-
	Experiencia en Aplicaciones	AEXP	1,29	1,13	1	0,91	0,82	-
	Capacidad de los programadores	PCAP	1,42	1,17	1	0,86	0,7	-
	Experiencia en Sistema Operativo utilizado	VEXP	1,21	1,1	1	0,9	-	-
	Experiencia con el Lenguaje de Programación	LEXP	1,14	1,07	1	0,95	-	-
ATRIBUTOS DEL PROYECTO	Prácticas de programación modernas	MODP	1,24	1,1	1	0,91	0,82	-
	Utilización de Herramientas de Software	TOOL	1,24	1,1	1	0,91	0,83	-
	Limitaciones de planificación del desarrollo del proyecto	SCED	1,23	1,08	1	1,04	1,1	-

**Nota.** Los valores marcados con fondo azul, se consideran seleccionados para la estimación.

**Fuente:** Elaboración Propia

$$FAE = 1 * 1 * 1 * 1 * 1 * 0,87 * 0,87 * 1 * 1,13 * 1,17 * 0,9 * 1,07 * 1,1 * 0,91 * 1$$

$$FAE = 0,964635355$$

#### Calculo del Esfuerzo del desarrollo:

$$E = FAE * A * KLDC^B$$

$$E = 0,964635355 * 3,2 * (14,419)^{1,05}$$

$$E = 50,8621896 \left[ \frac{\text{personas}}{\text{mes}} \right]$$

### **Calculo tiempo de desarrollo:**

$$T = C * \text{Esfuerzo}^D$$

$$T = 2,5 * (50,8621896)^{0,38}$$

$$T = 11,12679607 [\text{meses}]$$

### **Productividad:**

$$PR = \frac{LDC (\text{Lineas de codigo})}{\text{Esfuerzo}} = \frac{14\,419}{50,8621896}$$

$$PR = 283,4919247 \left[ \frac{\text{SLOC}}{\text{Persona}} * \text{mes} \right]$$

### **Personal promedio:**

$$\text{Personal} = \frac{\text{Esfuerzo}}{\text{Tiempo de Desarrollo}}$$

$$\text{Personal} = \frac{50,8621896}{11,12679607}$$

$$\text{Personal} = 4,571144 [\text{personas}]$$

Interpretando los valores obtenidos, se determina que en el proyecto existen 14 419 líneas de código funcionales, lo que se traduce en un esfuerzo de 50 personas por mes, el proyecto se ha desarrollado en 11 meses, lo ideal para desarrollar el proyecto es, contar con 5 programadores.

Para determinar el costo del proyecto se usa la siguiente ecuación: Personal \* Salario medio entre los programadores y analistas.

Costo proyecto=5\* 1000\$

Costo proyecto= 5000 \$

Pero la situación, es que en este caso, el proyecto ha sido desarrollado por una sola persona, así que los costos de desarrollo se calcularán considerando tal situación.

El desarrollador ha trabajado seis horas al día, quien ha realizado las tareas de análisis, diseño y codificación del proyecto. Se ha considerado el trabajo en días laborales de lunes a viernes, seis horas al día, lo que supone unas 120 horas al mes. Y según el tiempo estimado multiplicado por la cantidad de horas al mes, se obtiene un total de horas de trabajo de 1320 horas.

$$6 \frac{\text{horas}}{\text{día}} \Rightarrow 120 \frac{\text{horas}}{\text{mes}}$$

$$\textbf{Total Horas de Trabajo} = 120 * 11 = \mathbf{1320 [\text{horas}]}$$

Para determinar una aproximación del costo de desarrollo del proyecto se ha averiguado que el sueldo para el área de programación en Bolivia oscila el valor de \$us 4 dólares americanos la hora.

Por tanto, ajustando el horario de trabajo del autor del proyecto (3 horas al día), y considerando el pago por hora de \$us 4. Se procede a calcular el costo estimado del proyecto, multiplicando la cantidad de horas a trabajar en el lapso de tiempo estimado por el pago por hora determinado:

$$1320 [\text{horas}] * 4 \left[ \frac{\$us}{\text{hora}} \right] = \mathbf{5280[\$us]}$$

Observando el resultado de la ecuación se determina que el costo estimado del proyecto es de \$us 5280.- (dólares americanos). A continuación se muestra un resumen de los costos mínimos y recomendados necesarios para el proyecto usando 6,96 como tipo de cambio de dólares a bolivianos:

**Tabla 44:** Costos mínimos y recomendados del proyecto

	Requerimiento mínimo	Requerimiento ideal
Costo del servidor	0,00.-	62,00.-
Costo del equipo de los usuarios 2.	650,00.-	920,00.-
Costo del desarrollo	5280,00.-	5280,00.-
<b>TOTAL (\$)</b>	<b>5930,00.-</b>	<b>6262,00.-</b>
<b>TOTAL (Bs)</b>	<b>41272, 8.-</b>	<b>43583, 52.-</b>

**Fuente:** Elaboración Propia

El costo mínimo para el proyecto es 41272,8 bs y el costo recomendado es 43583,52 bs a continuación se realizará el análisis beneficio - costo para determinar si el proyecto es viable.

Para realizar el análisis de beneficio – costo del presente proyecto se emplearán los siguientes datos considerando que el desarrollo del proyecto será cero, ya que será obtenido gratuitamente, al monto total de requerimiento mínimo de obteniendo los siguientes datos:

- Cinco años, es el tiempo de vida útil del proyecto.
- 2,31% es la tasa de interés anual. (Fuente: BNB).
- Inversión inicial de 41272,8 Bs.

Teniendo en cuenta estos resultados se puede mencionar algunos beneficios otorgados por el sistema, justificado en el Anexo K, como ser:

A continuación se determina el análisis Beneficio/Costo detalladamente.

**Tabla 45:** Análisis costo beneficio.

DETALLE	MONTO SIN SISTEMA	MONTO POSIBLE CON SISTEMA
Costo en la compra de talonarios	5910,00 Bs.-	0,00 Bs.-
Costo económico en tinta	0,00 Bs.-	200,00 Bs.-
Costo económico en papel membretado	0,00 Bs.-	250,00 Bs.-
Costo en material de escritorio.	30,00 Bs.-	0,00 Bs.-
Costo de los pasajes invertidos en comprar talonarios.	100,00 Bs.-	0,00 Bs.-
Costo en la compra de impresora.	0 Bs.-	3132 Bs.-
Costo del tiempo invertido el proceso de pago de pensiones.	2070,98 Bs.-	2070,98 Bs.-
Costo invertido en el proceso de facturación por día.	690.3 Bs.-	358,97 Bs.-
<b>MONTO TOTAL ANUAL</b>	<b>8801,28 Bs.-</b>	<b>6011,95 Bs.-</b>

**Fuente:** Elaboración Propia.

Para el cálculo de la estimación Beneficio/Costo se tiene:

Si  $B/C > 1 \rightarrow$  el proyecto es viable económico ya que los beneficios superan a los costos.

Si  $B/C < 1 \rightarrow$  el proyecto no es viable económico ya que los beneficios son superados

$$B/C = \frac{\sum_{n=1}^n \frac{Y_n}{(1+i)^n}}{I_0 + \sum_{n=1}^n \frac{C_n}{(1+i)^n}}$$

$Y = \text{Beneficio neto}$

$C_n = \text{Costos en cada periodo}$

$n = \text{tiempo de vida de duración del proyecto}$  (5 años es el ciclo de vida de la aplicación ya que se mencionó en el capítulo de generalidades, donde se determinó el alcance temporal).

$Y_n = 8801,28 - 6011,95$

$Y_n = 2789 \text{ Bs.}$

$i = \text{Tasa de interés 2.3 del Banco Nacional de Bolivia}$

$$\begin{aligned} B/C &= \frac{\frac{2789,33}{(1+0,023)} + \frac{2789,33}{(1+0,023)^2} + \frac{2789,33}{(1+0,023)^3} + \frac{2789,33}{(1+0,023)^4} + \frac{2789,33}{(1+0,023)^5}}{41272,8 + (\frac{6011,95}{1+0,023} + \frac{6011,95}{(1+0,023)^2} + \frac{6011,95}{(1+0,023)^3} + \frac{6011,95}{(1+0,023)^4} + \frac{6011,95}{(1+0,023)^5})} \\ &= \frac{13033,69}{41272,8 + 28092,02} \end{aligned}$$

$$B/C = 0,1879$$

Al obtener el resultado menor a 1 se puede concluir que los beneficios que se obtienen con la implantación del proyecto son menores respecto al costo invertido para su desarrollo, pero por otro lado dará más comodidad y facilidad de acceso a los usuarios por lo tanto el proyecto es económicamente viable y beneficiará a la Unidad Educativa María Auxiliadora.

#### **4.3. VIABILIDAD OPERATIVA.**

El sistema permite al personal de la unidad educativa conceder permisos de usuario como administrador, secretaria, cajero y estudiante.

Al administrador se le permite registrar gestión educativa con montos de mensualidades y descuentos por segundo, tercer y cuarto hermano, también se le permite realizar la planificación de cursos, por otro lado permite registrar a los usuarios o personal de la unidad educativa concediéndole permisos de usuario, también le permite visualizar reportes diarios y generales de los ingresos realizados, por lo tanto su nivel de complejidad es media.

A la Secretaria se le permite gestionar estudiantes con su respectivo apoderado como también el registro de inscripción de estudiantes asignando a cursos previamente registrados por el administrador, a este usuario secretaria también se le permite visualizar estudiantes con deudas por cursos, por lo tanto su nivel de complejidad es media.

Al Cajero se le permite gestionar dosificación con los datos de dosificación brindado por Servicio de Impuestos Nacionales Bolivia como también gestionar el pago de pensiones con facturación computarizada, a este usuario también se le permite visualizar reportes de las facturas emitidas, así como también reportes de estado de pensiones de los estudiantes, por lo tanto su nivel de complejidad es media.

Al estudiante inscrito se le permite visualizar el estado de pensiones en el que se encuentra por cursos cursados en la unidad educativa, facilitando de esta manera el acceso al estado de pensiones ya que puede accederlo desde cualquier dispositivo ya sea móvil o pc el cual tenga internet.

El sistema es bastante sencillo de entender ya que presenta una interfaz Web muy intuitiva que solo requerirá de conceptos y conocimientos previos y estar familiarizado con el manejo de una PC y navegación en Internet, en la mayoría de las ventanas se ofrece básicamente dos botones para facilitar el trabajo de los usuarios, evitando de esta forma que el usuario pueda cometer errores y por otro lado facilitando la selección de opciones.

También cabe mencionar que se requerirá capacitación y charlas informativas a los diferentes actores para que estos se familiaricen con el sistema y así puedan trabajar con el mismo de una forma más eficiente y sin cometer errores.

Teniendo en cuenta los beneficios que brinda el sistema de gestión de pensiones con facturación computarizada, y tomando en cuenta además los resultados y conclusiones de las viabilidades Técnica y Económica se puede concluir que el sistema brinda beneficios que permiten llevar a cabo el proceso de gestión de pensiones de forma adecuada, por lo cual podemos decir el proyecto es operativamente Viable.

## **5. CONCLUSIONES Y RECOMENDACIONES**

Una vez alcanzados el objetivo general y los objetivos específicos a continuación se realizarán las conclusiones y recomendaciones.

### **5.1. CONCLUSIONES**

A continuación se mencionan las conclusiones para cada objetivo específico planteado:

- Para el primer objetivo se estableció el modelado de negocio actual efectuando el método de observación y cuestionarios al personal involucrado para determinar cómo se realiza este proceso el cual fue reflejado con un diagrama de flujo para su mejor entendimiento. Seguidamente de este se realizó la evaluación de las diferentes deficiencias existentes en el proceso actual para elaborar una propuesta de modelado de negocio alternativo solucionando estos problemas identificados el cual se plasmó en un diagrama de flujo para su mejor entendimiento.
- Para segundo objetivo, como primer paso se selección la metodología que mejor se adecue al sistema: Primero se realizó una tabla de comparativa con las ventajas y desventajas de los modelos más adecuados para el sistema en el cual se ha seleccionado el modelo incremental, con el que se hizo posible desarrollar una serie de incrementos obteniendo la entrega de productos operacionales en cada incremento. Como segundo paso se realizó una tabla comparativa para de los lenguajes de programación que más se adecuen al proyecto mencionando sus ventajas y desventajas en esta parte se seleccionó el lenguaje php por su gran compatibilidad con Html. Como tercer paso se realizó unatabla comparativa de los FrameWorks que se adecuen más al sistema mencionando sus ventajas y desventajas en el cual

se elegido a Codeigniter por su gran compatibilidad con sistemas multiplataforma y su facilidad de acceder a su documentación lo que ha permitido analizar, trabajar, evaluar y aplicar el uso de la arquitectura MVC (Modelo Vista Controlador), con modelo incremental para el desarrollo. Como Cuarto paso se realizó una tabla comparativa de las bases de datos, mencionando los gestores de bases de datos que más se adecuen al proyecto en el cual se elijo la base de datos Mysql con su gestor PhpMyAdmin versión 4.0 por la compatibilidad con el framework de Codeigniter y al lenguaje de php utilizadas, la cual facilito la implementación del sistema bajo arquitectura Modelo Vista Controlador. Seguidamente se realizó la actividad de análisis en la cual se realizó la identificación de los actores dando una descripción detallada de cada uno de estos, también se realizó el diseño por medio de los diagramas de colaboración, clases y de bases de datos. Seguidamente se procedió a la implementación en la cual se usaron hojas de estilos pre diseñadas como también una hoja de estilos llamada bootstrap en el cual se encuentra las diferentes adaptaciones que el sistema realizará cuando sea ingresado con diferentes resoluciones para asegurar su ejecución multiplataforma. Por último se realizaron las pruebas funcionales al módulo correspondiente.

- Para el tercer objetivo, primero se realizó la actividad de análisis dando una explicación detallada de los actores que se involucran en este módulo, seguidamente se elaboraron los diagramas de colaboración, clases y de bases de datos, para pasar al a implementación en la se utilizó Bootstrap en las vistas del sistema donde se configuraron la adaptación a las diferentes dispositivos y plataformas con diferentes resoluciones en las cuales se visualizó correctamente el sistema, una vez terminado se procedió a la realización de pruebas funcionales, concluyendo la implementación.

- Para el cuarto objetivo, como primer paso se procedió a analizar la normativa y procedimientos a seguir para la facturación computarizada regulada por Servicio de Impuestos Nacionales Bolivia, También se analizó el código de los algoritmos establecidos por el Sin para la generación del código de control y código QR, cabe decir que se aprendió y estudio los algoritmos utilizados para la generación del código de control y el funcionamiento de la librería para la generación del código QR. Seguidamente se procedió a realizar la actividad de análisis en la cual se describen a detalle los actores involucrados en este módulo, seguidamente se realizaron los diagramas de colaboración, clase y bases de datos para la correcta implementación de este, cabe decir que para los reportes se tuvo que aprender a utilizar la librería de Fpdf con la cual se puede pasar datos desde php a archivos pdf. En cuanto a la emisión de reportes, el sistema permite realizar la extracción de distintos tipos de reportes para cada tipo de usuario al administrador se le permite reportes de ingresos diarios y generales, a la secretaria se le permite reportes de estudiantes deudores por cursos ya que esta secretaria es la encargada de entrega de boletines. Finalmente se realizaron las pruebas funcionales como también los 5 mil casos de prueba exigidos por el Servicio de Impuestos Nacionales. Es necesario recalcar que para la realización de los 5 mil casos de prueba se tuvo que aprender a utilizar la librería de PhpExcel con la cual es posible leer datos de un documento Excel desde el lenguaje de php.
- Para el quinto objetivo, se realizaron pruebas funcionales con distintos dispositivos como tablets, smartphones y pc de escritorio como también laptops, en los cuales los resultados están reflejados en una tabla en la que se muestran las interfaces correctamente acoplada a los diferentes dispositivos verificando así su correcta ejecución.

Con la hipótesis planteada en el proyecto se ha demostrado que los beneficios principales del sistema desarrollado están relacionados con la reducción del tiempo invertido en el proceso de gestión de pensiones y disminución de errores en las facturas generadas y gastos económicos para la institución, por lo cual se concluye que la aplicación y puesta en marcha del sistema de gestión de pensiones, permitirá a la institución reemplazar procedimientos manuales y llegar a un proceso más eficiente.

Finalmente se concluye que se lograron alcanzar los objetivos establecidos al desarrollar un Sistema de Gestión de pensiones con Facturación computarizada, logrando obtener información confiable y rápida, reduciendo el tiempo invertido en el proceso de gestión de pensiones, disminuyendo los errores en la emisión de facturas y gastos económicos para la institución.

## **5.2. RECOMENDACIONES**

**Las recomendaciones de funcionamiento son:**

- Para el servidor realizar la configuración (descritas en el manual de instalación) necesarias para que el sistema pueda ser accedido desde cualquier dispositivo.
- En el servidor se recomienda verificar que ningún programa utilice el puerto 80 el cual es de gran importancia para el correcto funcionamiento del sistema.
- Para el caso de uso gestionar dosificación registrar cada seis meses según indica el SIN una nueva dosificación para que la facturación pueda efectuarse.
- Cambiar las contraseñas de los usuarios periódicamente de manera que se incremente la seguridad de acceso.
- Para un buen funcionamiento del sistema se recomienda cumplir con los requerimientos mínimos de hardware establecidos en la viabilidad técnica.

- Para proteger la información se recomienda realizar copias de respaldo, para mantener la disponibilidad de la información.
- Se recomienda agregar un servidor web alterno, en caso de que el servidor actual este en mantenimiento.
- Añadir criterios de búsquedas para los estudiantes de manera que permita obtener información requerida según las necesidades existentes.
- Anadir una función la cual permita bloquear los intentos fallidos de acceso al sistema mejorando la seguridad del mismo.
- Anadir reportes según las nuevas necesidades que aparezcan, de manera que permitan obtener la información requerida respecto a pago de pensiones

**Las recomendaciones a futuro son:**

- Se recomienda una función que permita añadir envío de correo electrónico a los padres de familia o apoderado para informar del estado de pensiones de los estudiantes.
- En el registro de estudiantes y usuarios añadir un módulo que permita añadir fotografía de los mismos de manera que permita la mejor identificación.
- Se recomienda agregar una función que permita recuperar a los usuarios y sus contraseñas correspondientes.
- Se recomienda actualizar el formato de las facturas computarizadas según a las normativas futuras del SIN.

## BIBLIOGRAFÍA

- Casillas, (2014), “Manual de Bases de datos MySQL”, Recuperado de: <http://www.fcca.umich.mx/descargas/apuntes/Academia%20de%20Informatica/Base%20de%20Datos%20%20I%20%20%20%20G.A.G.C/TUTORIAL%20B%C3%81SICO%20MySQL.pdf>.
- Castellar (2014), “Pruebas de Software”, Recuperado de: <http://elpuig.xeill.net/Members/vcarceler/asix-m09/uf1/nf1/a5>.
- Chaffer y Swedberg (2007), “Learning JQuery” cuarta edición editorial Packt Publishing Ltd. Reino Unido.
- Codeigniter (2014), “Guía de Usuario en español Versión 2.1.3”, Recuperado de: <http://www.etnassoft.com/biblioteca/codeigniter-guia-del-usuario-en-espanol-v-2-1-2/>.
- Developer, (2014). “Articulo para desarrolladores de aplicaciones web”, Recuperado de: [https://developer.mozilla.org/es/Apps/Para\\_desarrolladores\\_de\\_aplicaciones\\_para\\_dispositivos\\_m%C3%B3viles](https://developer.mozilla.org/es/Apps/Para_desarrolladores_de_aplicaciones_para_dispositivos_m%C3%B3viles).
- Euskal (2014), “Ventajas Y Desventajas de HTML”, Recuperado de: <http://www.euskalnet.net/jaoprogamador/webmaster/xml/xml05.htm>.
- Ecured,(2014),“Lenguaje PHP”, Recuperado de: [http://www.ecured.cu/index.php/PHPGTK\\_\(Lenguaje\\_de\\_programaci%C3%B3n\)](http://www.ecured.cu/index.php/PHPGTK_(Lenguaje_de_programaci%C3%B3n)).
- Glera (2014), “Guia de desarrollo para dispositivos Moviles”, Recuperado de: <http://academicae.unavarra.es/bitstream/handle/2454/7545/578082.pdf?sequence=1>.
- Garcia (2014), “Aprende Java Como si Estuviera en Primero”, Recuperado de: <http://www.matematica.ciens.ucv.ve/files/Manuales/Manuales/Programacion%20Web%20%20Aprenda%20Java%20como%20si%20estuviera%20en%20primero.pdf>.

- Marc (2014), “Bases de Datos PostgreSQL”, Recuperado de: [http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06\\_M2109\\_02152.pdf](http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02152.pdf).
- Microsoft, (2014), “Desarrollo de Software basado en componentes”, Recuperado de <http://msdn.microsoft.com/es-bo/library/bb972268.aspx>.
- Microfot (2014), “Introduccion a SQL Server”, Recuperado de: [http://download.microsoft.com/download/F/F/6/FF62CAE0-CE38-4228-9025FBF729312698/Microsoft\\_Press\\_eBook\\_Introducing\\_Microsoft\\_SQL\\_Server\\_2012\\_PDF.pdf](http://download.microsoft.com/download/F/F/6/FF62CAE0-CE38-4228-9025FBF729312698/Microsoft_Press_eBook_Introducing_Microsoft_SQL_Server_2012_PDF.pdf).
- Ojeda (2014), “Guía Completa CSS3”, Recuperado de: <http://www.etnassoft.com/biblioteca/guia-completa-de-css3/>
- Pressman Roger S. (2010). Ingeniería del software, Ed. Mc Graw Hill, 7<sup>a</sup> edición.
- Sánchez (2014),”Apuntes Básicos de Java Script”, Recuperado de: <http://www.jorgesanchez.net/web/lmsgi/LMSGI05.pdf>.
- Servicio de Impuestos Nacionales (2013). “Resolución Normativa de Directorio Numero 10-0049-113”, Recuperado de: <http://www.impuestos.gob.bo/images/GACCT/FACTURACION/Normativa.php>.
- Sommerville Ian (2005). Ingeniería de software, Ed. Addison Wesley 7<sup>a</sup> Edicion.
- The PHP Development Team (2014), “Documentación de Php ”, Recuperado de: <http://php.net/manual/es/index.php>
- Universidad Carlos III de Madrid (2014), “Framework de desarrollo de aplicaciones web multiplataforma”, Recuperado de : [http://earchivo.uc3m.es/bitstream/handle/10016/17139/tfg\\_javier\\_espino sa\\_alfonso\\_2012.pdf?sequence=1](http://earchivo.uc3m.es/bitstream/handle/10016/17139/tfg_javier_espino sa_alfonso_2012.pdf?sequence=1).

- Universidad Cibertec, (2014), “Pruebas de Software”, Recuperado de: <http://cibertec.googlecode.com/files/Pruebas%20de%20Software.pdf>.
- Universidad Complutense, (2014), “Modelo Vista Controlador”, Madrid, Recuperado de: <http://cic.puj.edu.co/wiki/lib/exe/fetch.php?media=materias:patronmvc.pdf>.
- Universidad de Cauca, (2014), “Arquitectura Cliente-Servidor”, Recuperado de: <http://artemisa.unicauc.edu.co/~ecaldon/docs/apiweb/Elec-Ses2-2k5-ClienteServidor.pdf>.
- Universidad de las Américas Puebla, (2014), “Articulo Sobre Arquitectura MVC”, México, Recuperado de: [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lis/rivera\\_l\\_a/capitulo2.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/rivera_l_a/capitulo2.pdf).
- Universidad Francisco Gavidia, (2014), “Instrumentos para la recolección de información recuperado de <http://wwwisis.ufg.edu.sv/wwwisis/documents/TE/371.33-V335i/371.33-V335i-Capitulo%20V.pdf>.
- Universidad de Fribourg de Suiza por Dennis Chávez (2014), “Técnicas de Recolección de información en la investigación”, Recuperado de: [https://www.unifr.ch/ddp1/derechopenal/articulos/a\\_20080521\\_56.pdf](https://www.unifr.ch/ddp1/derechopenal/articulos/a_20080521_56.pdf).
- World Wide Web Consortium (2014), “HTML5”, Recuperado de: <http://www.w3.org/TR/html5/>.