



# **UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO**



**FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS**

**ESCUELA PROFESIONAL DE INGENIERÍA EN  
COMPUTACIÓN E INFORMÁTICA**

**“IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO  
DIRIGIDO AL PROCESO ACADÉMICO CON SOPORTE  
BASADO EN NOTIFICACIONES PARA EL COLEGIO  
MIGUEL ANGEL BUONARROTI”**

## **TESIS**

**PARA OPTAR EL TÍTULO PROFESIONAL DE  
INGENIERO EN COMPUTACIÓN E INFORMÁTICA**

**AUTORA:**

**BACH. ANILU NOEMI CHINCHAY PACHECO**

**ASESOR:**

**ING. FRANKLIN EDINSON TERÁN SANTA CRUZ**

**LAMBAYEQUE-PERÚ**

**2019**



# **UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO**




**FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS**

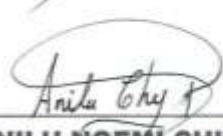
**ESCUELA PROFESIONAL DE INGENIERÍA EN  
COMPUTACIÓN E INFORMÁTICA**

**“IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO  
DIRIGIDO AL PROCESO ACADÉMICO CON SOPORTE  
BASADO EN NOTIFICACIONES PARA EL COLEGIO  
MIGUEL ANGEL BUONARROTI”**

## **TESIS**

**PARA OPTAR EL TÍTULO PROFESIONAL DE  
INGENIERO EN COMPUTACIÓN E INFORMÁTICA**

  
Ing. FRANKLIN EDINSON TERÁN SANTA CRUZ  
ASESOR

  
Bach. ANILU NOEMI CHINCHAY PACHECO  
AUTORA



**UNIVERSIDAD NACIONAL**

**PEDRO RUIZ GALLO**



**FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS**

**ESCUELA PROFESIONAL DE INGENIERÍA EN  
COMPUTACIÓN E INFORMÁTICA**

**"IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO  
DIRIGIDO AL PROCESO ACADÉMICO CON SOPORTE  
BASADO EN NOTIFICACIONES PARA EL COLEGIO  
MIGUEL ANGEL BUONARROTI"**

**TESIS**

**PARA OPTAR EL TÍTULO PROFESIONAL DE  
INGENIERO EN COMPUTACIÓN E INFORMÁTICA**

APROBADA POR:

**Dr. IVÁN EDUARDO SALVADOR BRICEÑO  
PRESIDENTE DE JURADO**

**Mg. Ing. CARLOS ALBERTO VALDIVIA SALAZAR  
SECRETARIO DE JURADO**

**Mg. Ing. PERCY JAVIER CELIS BRAVO  
VOCAL DE JURADO**

## **AGRADECIMIENTO**

Agradecer a Dios por darme sabiduría y fortaleza para que fuera posible alcanzar este triunfo.

A mis padres, quienes me han apoyado incondicionalmente en cada etapa de mi vida.

Al Ing Franklin Terán Santa Cruz por mostrar su apoyo en este proyecto de Tesis.

## **DEDICATORIA**

Dedicada a Dios, por brindarme la sabiduría, inteligencia y las fuerzas cada día para seguir adelante, Ya que gracias a él he logrado concluir esta etapa de formación profesional de mi vida.

A mis padres por su lucha constante y su amor latente todo el tiempo, por cada palabra y gesto de cariño que han guiado los pasos a lo largo de mi vida, y por ser los mejores padres del mundo.

A mis hermanos, quienes me brindaron su apoyo y confianza.

## **PRESENTACIÓN**

Conforme a lo dispuesto por el reglamento de Grados y Títulos Vigente de la Universidad Nacional Pedro Ruiz Gallo con el fin de optar el Título profesional de Ingeniero en Computación e Informática ponemos a consideración la siguiente tesis intitulada:

### **IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO DIRIGIDO AL PROCESO ACADÉMICO CON SOPORTE BASADO EN NOTIFICACIONES PARA EL COLEGIO MIGUEL ANGEL BUONARROTI**

El presente trabajo es desarrollado para la gestión académica del colegio Miguel Angel Buonarroti, con el objetivo de elaborar un Sistema Informático que permita la matriculación y la correcta gestión académica de los alumnos de los diversos grados, así mismo el sistema genera los reportes para un mejor manejo de información, también podrán ser notificados los padres de familia sobre el avance de sus hijos y lograr un manejo óptimo de la información, elevando así su nivel competitivo.

Aprovechando la oportunidad para expresar mi agradecimiento a todas las autoridades y cuerpo docente de la Escuela Profesional de Ingeniería en Computación e Informática por sus valiosas enseñanzas que posibilitaron mi formación académica y profesional.

## RESUMEN

La presente tesis tiene como principal objetivo, implementar un Sistema Informático para gestionar la información académica del Colegio Miguel Angel Buonarroti. Para lograr tal fin se utilizó la metodología XP (Xtreme Programming), ya que agiliza el desarrollo del proyecto en aspectos de planificación y diseño.

Se identifica y se detalla cada una de las etapas, comenzando con la recopilación de datos, que nos permitió especificar los requerimientos precisos para dar solución a la problemática. Lo siguiente fue definir la arquitectura que se utilizó, ya que se debió integrar diferentes tecnologías como Hibernate y Bootstrap, Luego se implementó el sistema según la arquitectura definida y tomando en cuenta los requerimientos capturados, para que este cumpla con dichos requerimientos.

Finalmente se realizaron las pruebas necesarias para comprobar el correcto funcionamiento del sistema informático.

Con el desarrollo de la investigación se comprobó que si se implementa el sistema informático se da solución a las necesidades de organizar de manera eficiente la información proveniente del área académica del colegio Miguel Angel Buonarroti.

**Palabras claves:** Sistema informático, proceso, soporte, notificaciones.

## **ABSTRACT**

The main objective of this thesis is to implement a Computer System to manage the academic information of the Miguel Angel Buonarroti School. To achieve this end, the XP methodology (Xtreme Programming) was used, since it speeds up the development of the project in aspects of planning and design.

Each of the stages is identified and detailed, starting with the data collection, which allowed us to specify the precise requirements to solve the problem. The following was to define the architecture that was used, since it had to integrate different technologies such as Hibernate and Bootstrap, then the system was implemented according to the defined architecture and taking into account the captured requirements, so that it meets those requirements.

Finally, the necessary tests were carried out to verify the correct functioning of the computer system.

With the development of the research it was proved that if the computer system is implemented, a solution is found to the needs of efficiently organizing the information coming from the academic area of the Miguel Angel Buonarroti school.

**Keywords:** Computer system, process, support, notifications.



## ÍNDICE

<b>AGRADECIMIENTO .....</b>	<b>4</b>
<b>DEDICATORIA .....</b>	<b>5</b>
<b>PRESENTACIÓN .....</b>	<b>6</b>
<b>RESUMEN .....</b>	<b>7</b>
<b>ABSTRACT.....</b>	<b>8</b>
<b>ÍNDICE GENERAL.....</b>	<b>9</b>
<b>INTRODUCCIÓN .....</b>	<b>20</b>
<b>CAPÍTULO I: DATOS GENERALES DE LA ORGANIZACIÓN .....</b>	<b>22</b>
<b>1.1. MISIÓN, VISIÓN Y OBJETIVOS DE LA ORGANIZACIÓN .....</b>	<b>22</b>
1.1.1. Misión.....	22
1.1.2. Visión .....	22
1.1.3. Objetivos de la organización .....	22
<b>CAPÍTULO II: PROBLEMÁTICA DE LA INVESTIGACIÓN .....</b>	<b>24</b>
<b>2.1. REALIDAD PROBLEMÁTICA.....</b>	<b>24</b>
2.1.1. Planteamiento del problema.....	24
2.1.2. Formulación del problema.....	25
<b>2.2. JUSTIFICACIÓN E IMPORTANCIA DE LA INVESTIGACIÓN .....</b>	<b>25</b>
2.2.1. Justificación.....	25
2.2.2. Importancia .....	25
<b>2.3. OBJETIVOS DE LA INVESTIGACIÓN .....</b>	<b>26</b>
2.3.1. Objetivo general .....	26
2.3.2. Objetivos específicos .....	26
<b>CAPÍTULO III: MARCO METODOLÓGICO.....</b>	<b>28</b>
<b>3.1. TIPO DE INVESTIGACIÓN .....</b>	<b>28</b>
<b>3.2. HIPÓTESIS.....</b>	<b>28</b>
<b>3.3. VARIABLES .....</b>	<b>28</b>

3.3.1. Variable independiente.....	28
3.3.2. Variable dependiente .....	29
<b>CAPÍTULO IV: DISEÑO TEÓRICO .....</b>	<b>31</b>
<b>4.1. ANTECEDENTES DE LA INVESTIGACIÓN.....</b>	<b>31</b>
4.1.1. Antecedentes internacionales .....	31
4.1.2. Antecedentes nacionales .....	32
4.1.3. Antecedentes locales .....	34
<b>4.2. BASE TEÓRICA.....</b>	<b>35</b>
4.2.1. Conceptos generales .....	35
4.2.2. Modelado de datos.....	39
4.2.3. Desarrollo .....	43
4.2.3.1. Modelo vista controlador (MVC) .....	43
4.2.3.2. Java .....	44
4.2.3.3. Spring .....	47
4.2.3.4. Hibernate .....	50
4.2.3.5. Bootstrap .....	51
4.2.4. Base de datos .....	52
4.2.4.1. ....	
MySQL .....	53
4.2.5. Herramientas informáticas .....	57
4.2.5.1. Erwin .....	57
4.2.5.2. NetBeans IDE .....	57
4.2.5.3. Apache .....	58
<b>4.3. DEFINICIÓN DE TÉRMINOS Y CONCEPTOS .....</b>	<b>59</b>
<b>CAPÍTULO V: DESARROLLO DE LA PROPUESTA .....</b>	<b>62</b>
<b>5.1. TÉCNICAS DE RECOLECCIÓN DE DATOS.....</b>	<b>62</b>
5.1.1. Observación .....	62

5.1.2. Entrevista .....	62
<b>5.2 CONFIGURACIONES PARA EL DESARROLLO DEL SISTEMA.....</b>	<b>66</b>
5.2.1. Base de datos .....	66
5.2.1.1. Diseño .....	66
5.2.1.2. Gestor de base de datos .....	67
5.2.2. Software .....	67
<b>5.3. PLANIFICACIÓN Y DISEÑO .....</b>	<b>77</b>
5.3.1. Iteración 1 .....	77
5.3.1.1. Historias de usuario .....	77
5.3.1.2. Diseño de la iteración 1 .....	82
5.3.1.3. Diagrama de base de datos iteración 1 .....	83
5.3.2. Iteración 2 .....	84
5.3.2.1. Historias de usuario .....	84
5.3.2.2. Diseño de la iteración 2 .....	90
5.3.2.3. Diagrama de base de datos iteración 2 .....	92
5.3.3. Iteración 3 .....	93
5.3.3.1. Historias de usuario .....	93
5.3.3.2. Diseño de iteración 3 .....	98
5.3.3.3. Diagrama de base de datos iteración 3 .....	100
5.3.4. Iteración 4 .....	100
5.3.4.1. Historias de usuario .....	101
5.3.4.2. Diseño de la iteración 4 .....	105
5.3.4.3. Diagrama de base de datos iteración 4 .....	107
5.3.5. Iteración 5 .....	107
5.3.5.1. Historias de usuario .....	108
5.3.5.2. Diseño de la iteración 5 .....	114
5.3.5.3. Diagrama de base de datos iteración 5 .....	116

5.3.6.	Iteración 6 .....	116
5.3.6.1.	Historias de usuario .....	117
5.3.6.2.	Diseño de la iteración 6 .....	121
5.3.6.3.	Diagrama de base de datos iteración 6 .....	123
5.3.7.	Iteración 7 .....	123
5.3.7.1.	Historias de usuario .....	124
5.3.7.2.	Diseño de la iteración 7 .....	127
5.3.7.3.	Diagrama de base de datos iteración 7 .....	128
<b>5.4.</b>	<b>DIAGRAMA DE COMPONENTES .....</b>	<b>129</b>
<b>5.5.</b>	<b>DIAGRAMA DE DESPLIEGUE .....</b>	<b>130</b>
<b>5.6.</b>	<b>BASE DE DATOS .....</b>	<b>131</b>
<b>5.7.</b>	<b>IMPLEMENTACIÓN DE UNA FUNCIONALIDAD .....</b>	<b>132</b>
5.7.1.	Guardar matrícula .....	132
<b>CAPÍTULO VI: ASPECTO ADMINISTRATIVO .....</b>		<b>154</b>
<b>6.1.</b>	<b>CRONOGRAMA DE ACTIVIDADES .....</b>	<b>154</b>
<b>6.2.</b>	<b>ANÁLISIS DE COSTOS .....</b>	<b>155</b>
6.2.1.	Costo de materiales .....	155
6.2.2.	Costo de servicios .....	155
6.2.3.	Costo personal .....	156
6.2.4.	Resumen de costos .....	157
6.2.5.	Financiamiento .....	157
<b>RESULTADO .....</b>		<b>158</b>
<b>DISCUSIÓN .....</b>		<b>159</b>
<b>CONCLUSIONES .....</b>		<b>160</b>
<b>RECOMENDACIONES .....</b>		<b>161</b>
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>		<b>162</b>
<b>ANEXOS .....</b>		<b>166</b>

## ÍNDICE DE FIGURAS

Figura 1. Visualización del proceso de Aplicación Web .....	39
Figura 2. Arquitectura de MVC .....	44
Figura 3. Herramientas Usadas con Hibernate .....	51
Figura 4. Ficha de Matricula .....	65
Figura 5. Base de Datos Modelada en ERWIN .....	66
Figura 6. Tablas de la Base de Datos en el Gestor de Base HeidiSQL .....	67
Figura 7. Archivo de Configuración para Spring ( spring – servlet.xml).....	68
Figura 8. Archivo de Configuración de la Aplicación Maven (pom.xml).....	69
Figura 9. Estructura de Archivos de la Carpeta WEB - INF.....	69
Figura 10. Estructura de Archivos de la Carpeta Resources.....	70
Figura 11. Estructura de Archivos de la Carpeta Source Packages.....	71
Figura 12. Archivo Bean para la Entidad Alumno.....	72
Figura 13. Archivo Controller para el Controlador de la Entidad Alumno .....	73
Figura 14. Archivo para el Acceso a Datos (DAO) de la Entidad Alumno .....	74
Figura 15. Archivo Service (capa de negocio) donde se definen las transacciones .....	75
Figura 16. Definición de Beans en el Archivo spring – servlet.xml.....	76
Figura 17. Configuración de Conexión a la Base de Datos (spring – servlet.xml) .....	76
Figura 18. Gestión de Usuarios.....	82
Figura 19. Gestión de Perfiles .....	82
Figura 20. Gestión de Opciones de Menú .....	83
Figura 21. Diagrama de Base de Datos Iteración 1.....	83
Figura 22. Gestión de Padres.....	90

Figura 23. Gestión de Alumnos .....	91
Figura 24. Gestión de Personal .....	91
Figura 25. Gestión de Docentes .....	92
Figura 26. Diagrama de Base de Datos Iteración 2.....	92
Figura 27. Gestión de Años.....	98
Figura 28. Gestión de Periodos Académicos .....	98
Figura 29. Gestión de Instituciones Educativas.....	99
Figura 30. Diagrama de Base de Datos Iteración 3.....	100
Figura 31. Gestión de Niveles .....	105
Figura 32. Gestión de Áreas.....	106
Figura 33. Gestión de Grados .....	106
Figura 34. Diagrama de Base de Datos Iteración 4.....	107
Figura 35. Gestión de Cursos.....	114
Figura 36. Gestión de Aulas .....	114
Figura 37. Gestión de Secciones .....	115
Figura 38. Asignaciones de Cursos.....	115
Figura 39. Diagrama de Base de Datos Iteración 5.....	116
Figura 40. Gestión de Matrícula .....	121
Figura 41. Marcar Asistencia.....	122
Figura 42. Notificaciones de Falta (Justificación) .....	122
Figura 43. Diagrama de Base de Datos Iteración 6.....	123
Figura 44. Registro de Promedios por Año y Periodo (Curso y Área) .....	127
Figura 45. Diagrama de Base de Datos Iteración 7.....	128
Figura 46. Diagrama de Componentes .....	129
Figura 47. Diagrama de Despliegue.....	130

Figura 48. Base de Datos.....	131
Figura 49. Campos de Tabla Matrícula .....	132
Figura 50. Interfaz Registro Matrícula .....	132
Figura 51. Código Javascript Obtener Item .....	133
Figura 52. Código Javascript Guardar Matrícula .....	133
Figura 53. Controller Matrícula .....	134
Figura 54. Controller Matrícula (Método Guardar).....	134
Figura 55. Método Guardar Capa de Servicio .....	135
Figura 56. Comando Autowired.....	135
Figura 57. Método Guardar Capa Servicio .....	135
Figura 58. Método Guardar Capa DAO (Acceso a Datos) .....	136
Figura 59. Login de Usuarios .....	140
Figura 60. Registro de Usuarios.....	141
Figura 61. Registro de Perfiles .....	141
Figura 62. Registro de Menú .....	142
Figura 63. Registro de Padre o Apoderado .....	142
Figura 64. Registro de Alumnos.....	143
Figura 65. Registro de Personal.....	143
Figura 66. Registro de Docentes.....	144
Figura 67. Registro de Año Académico.....	144
Figura 68. Registro de Periodos.....	145
Figura 69. Registro de Instituciones .....	145
Figura 70. Registro de Niveles .....	146
Figura 71. Registro de Áreas.....	146
Figura 72. Registro de Grados .....	147

Figura 73. Registro de Cursos.....	147
Figura 74. Registro de Aulas .....	148
Figura 75. Registro de Secciones .....	148
Figura 76. Registro de Asignaciones.....	149
Figura 77. Registro Matrícula .....	150
Figura 78. Marcar Asistencia.....	151
Figura 79. Enviar Notificaciones.....	151
Figura 80. Registro de Promedios de Cursos.....	152
Figura 81. Registrar Citación.....	152



## ÍNDICE DE TABLAS

Tabla 1	Variable Independiente .....	28
Tabla 2	Variable Dependiente.....	29
Tabla 3	Historia de Usuario N° 1.....	77
Tabla 4	Tarea Diseño de Interfaz Gestión de Usuarios .....	78
Tabla 5	Tarea Mantenimiento de Usuarios .....	78
Tabla 6	Historia de Usuario N° 2.....	79
Tabla 7	Tarea Diseño de Interfaz Gestión de Perfiles.....	79
Tabla 8	Tarea Mantenimiento de Perfiles .....	80
Tabla 9	Historia de Usuario N° 3.....	80
Tabla 10	Tarea Diseño de Interfaz Gestión de Opciones de Menú.....	81
Tabla 11	Tarea Mantenimiento de Opciones de Menú .....	81
Tabla 12	Historia de Usuario N° 4.....	84
Tabla 13	Tarea Diseño de Interfaz Gestión de Padres .....	85
Tabla 14	Tarea Mantenimiento de Padres .....	85
Tabla 15	Historia de Usuario N° 5.....	86
Tabla 16	Tarea Diseño de Interfaz Gestión de Alumnos.....	86
Tabla 17	Tarea Mantenimiento de Alumnos .....	87
Tabla 18	Historia de Usuario N° 6.....	87
Tabla 19	Tarea Diseño de Interfaz Gestión de Personal .....	88
Tabla 20	Tarea Mantenimiento de Personal .....	88
Tabla 21	Historia de Usuario N° 7 .....	89
Tabla 22	Tarea Diseño de Interfaz Gestión de Docentes .....	89
Tabla 23	Tarea Mantenimiento de Docentes .....	90
Tabla 24	Historia de Usuario N° 8.....	93

Tabla 25 Tarea Diseño de Interfaz Gestión de Años .....	94
Tabla 26 Tarea Mantenimiento de Años .....	94
Tabla 27 Historia de Usuario N° 9.....	95
Tabla 28 Tarea Diseño de Interfaz Gestión de Periodos Académicos .....	95
Tabla 29 Tarea Mantenimiento de Periodos Académicos .....	96
Tabla 30 Historia de Usuario N° 10.....	96
Tabla 31 Tarea Diseño de Interfaz de Instituciones Educativas.....	97
Tabla 32 Tarea Mantenimiento de Instituciones Educativas .....	97
Tabla 33 Historia de Usuario N° 11 .....	101
Tabla 34 Tarea Diseño de Interfaz Gestión de Niveles.....	101
Tabla 35 Tarea Mantenimiento de Niveles.....	102
Tabla 36 Historia de Usuario N° 12.....	102
Tabla 37 Tarea Diseño de Interfaz Gestión de Áreas .....	103
Tabla 38 Tarea Mantenimiento de Áreas .....	103
Tabla 39 Historia de Usuario N° 13.....	104
Tabla 40 Tarea Diseño de Interfaz Gestión de Grados.....	104
Tabla 41 Tarea Mantenimiento de Grados.....	105
Tabla 42 Historia de Usuario N° 14.....	108
Tabla 43 Tarea Diseño de Interfaz Gestión de Cursos .....	108
Tabla 44 Tarea Mantenimiento de Cursos .....	109
Tabla 45 Historia de Usuario N° 15.....	109
Tabla 46 Tarea Diseño de Interfaz Gestión de Aulas.....	110
Tabla 47 Tarea Mantenimiento de Aulas .....	110
Tabla 48 Historia de Usuario N° 16.....	111
Tabla 49 Tarea Diseño de Interfaz Gestión de Secciones .....	111

Tabla 50 Tarea Mantenimiento de Secciones .....	112
Tabla 51 Historia de Usuario N° 17 .....	112
Tabla 52 Tarea Diseño de Interfaz Asignación de Cursos .....	113
Tabla 53 Tarea Mantenimiento de Asignación de Cursos.....	113
Tabla 54 Historia de Usuario N° 18.....	117
Tabla 55 Tarea Diseño de Interfaz Gestión de Matrícula .....	117
Tabla 56 Tarea Mantenimiento de Matrícula.....	118
Tabla 57 Historia de Usuario N° 19.....	118
Tabla 58 Tarea Diseño de Interfaz Marcar Asistencia .....	119
Tabla 59 Tarea Mantenimiento de Marcar Asistencia .....	119
Tabla 60 Historia de Usuario N° 20.....	120
Tabla 61 Tarea Desarrollo de Funcionalidad Enviar Notificación.....	120
Tabla 62 Historia de Usuario N° 21 .....	124
Tabla 63 Tarea Diseño de Interfaz Registro de Notas por Año.....	124
Tabla 64 Tarea Mantenimiento de Registro de Promedios por Año.....	125
Tabla 65 Historia de Usuario N° 22.....	125
Tabla 66 Tarea Diseño de Interfaz Registro de Promedios por Periodo ...	126
Tabla 67 Tarea Mantenimiento de Registro de Promedios por Periodo....	126
Tabla 68 Costo de Materiales .....	155
Tabla 69 Costo de Servicios .....	155
Tabla 70 Costo de Personal.....	156
Tabla 71 Bienes Materiales.....	156
Tabla 72 Resumen de Costos.....	157

## INTRODUCCIÓN

El proyecto lleva por título “Implementación De Un Sistema Informático Dirigido Al Proceso Académico Con Soporte Basado En Notificaciones Para El Colegio Miguel Angel Buonarroti”, y consta de 6 capítulos, los cuales van detallando el desarrollo del problema y dando los alcances necesarios. Los capítulos son Datos Generales de la Organización, Problemática de la Investigación, Marco Metodológico, Marco Teórico, Desarrollo de la Propuesta y Aspecto Administrativo.

Capítulo I, en el cual se hace referencia a la información de la Institución donde se llevó a cabo el proyecto, es decir, su contexto organizacional: misión y visión.

Capítulo II, describe el planteamiento del problema, justificación e importancia y los objetivos a cumplir por el desarrollo del sistema.

Capítulo III, describe tipo de investigación, hipótesis, variables y diseño y contrastación de la hipótesis.

Capítulo IV, describe antecedentes, base teórica, definición de términos.

Capítulo V: describe cada uno de los pasos que se realizaron durante el desarrollo de la investigación, teniendo en cuenta cada una de las fases de la metodología XP, dentro de esta descripción se han incluido puntos sobre el desarrollo de la arquitectura utilizada en el proyecto, dicha arquitectura es producto de una extensa investigación sobre cómo integrar diferentes herramientas que optimizan el desarrollo de software como frameworks de desarrollo.

Capítulo VI, describe cronograma de actividades, presupuesto.

# **CAPÍTULO I**

## **DATOS GENERALES DE LA ORGANIZACIÓN**

## **CAPÍTULO I: DATOS GENERALES DE LA ORGANIZACIÓN**

### **1.1. MISIÓN, VISIÓN Y OBJETIVOS DE LA ORGANIZACIÓN**

#### **1.1.1. Misión**

“Miguel Angel Buonarroti” es una institución que promueve una educación integral: en valores y conocimientos, preparándolos, no sólo para la vida universitaria; sino también para la vida en sociedad con alta calidad de vida espiritual, económica u moral; capaz de resolver con creatividad los desafíos que le plantea este mundo globalizado.

#### **1.1.2. Visión**

Mantener el liderazgo educativo en nuestra comunidad con los más exigentes estándares internacionales de calidad, contribuyendo con la mejora de la educación en nuestro país.

#### **1.1.3. Objetivos de la organización**

El objetivo principal del colegio “Miguel Angel Buonarroti” consiste que el alumno se desarrolle integralmente (intelectual, social, física, emocional y espiritualmente). Forjar en nuestros alumnos el sentido de compromiso, así como la calidad humana y académica en su enseñanza diaria.

## **CAPÍTULO II**

# **PROBLEMÁTICA DE LA INVESTIGACIÓN**

## **CAPÍTULO II: PROBLEMÁTICA DE LA INVESTIGACIÓN**

### **2.1. REALIDAD PROBLEMÁTICA**

#### **2.1.1. Planteamiento del problema**

El colegio “Miguel Angel Buonarroti”, cuenta con el área administrativa donde a diario se realizan procesos para gestionar matrícula, notas, eventos, y demás material académico, lo que supone que esta información está sujeta a constantes actualizaciones.

Actualmente la gestión del material académico está en registros físicos o tablas de Excel, así como los expedientes y demás procesos, de tal manera que es un poco complicado consultar la información del alumno, y es también complicado generar reportes académicos ya que se tienen que consultar muchos archivos.

La información es manejada por personal autorizado que no lleva un control de los datos de una manera automatizada y computarizada que les permita que la información sea ágil, eficiente y eficaz.

No se ofrecen facilidades informáticas tanto para los alumnos, padres de familia y personal académico. Los encargados del área administrativa necesitan que la información del material académico esté adecuadamente organizada y de fácil acceso ya que actualmente no se cuenta con un sistema que realice dicha actividad.

Para dar solución a los inconvenientes descritos en los párrafos anteriores se propone implementar un Sistema Informático que facilite la labor administrativa, así como mejorar la interrelación con los padres de familia.



### **2.1.2. Formulación del problema**

¿Cómo la implementación del Sistema Informático, podrá mejorar el proceso académico y la interrelación con los padres de familia?

## **2.2. JUSTIFICACIÓN E IMPORTANCIA DE LA INVESTIGACIÓN**

### **2.2.1. Justificación**

La implementación del sistema informático pondrá la información del material académico del colegio “Miguel Angel Buonarroti” al alcance de los alumnos, docentes, padres de familia y personal administrativo, proporcionando una mejor efectividad en el manejo del flujo y procesamiento de los datos de los expedientes de sus estudiantes, y al mismo tiempo facilitando el manejo de información que representa un recurso viable para que la información sea ágil, eficiente y eficaz cuando sea requerida.

### **2.2.2. Importancia**

La importancia de la implementación de este sistema radica en organizar correctamente la información académica del colegio, identificando los procedimientos que desarrollan en la administración; asimismo dando un gran paso a la introducción en la tecnología y globalización.

## **2.3. Objetivos de la investigación**

### **2.3.1. Objetivo general**

Implementar un Sistema Informático para apoyar el proceso académico con soporte basado en notificaciones para el colegio Miguel Angel Buonarroti.

### **2.3.2. Objetivos específicos**

1. Identificar los requerimientos necesarios para el proceso académico.
2. Modelar la base de datos que de soporte al sistema y a las consultas requeridas.
3. Utilizar la metodología ágil XP.
4. Diseñar las interfaces del sistema conforme a los requerimientos encontrados.
5. Implementar el sistema académico con soporte a notificaciones para los resultados necesarios.
6. Probar el funcionamiento del sistema.

# **CAPÍTULO III**

## **MARCO METODOLÓGICO**

## CAPÍTULO III: MARCO METODOLÓGICO

### 3.1. TIPO DE INVESTIGACIÓN

Investigación Tecnológica Formal

### 3.2. HIPÓTESIS

La implementación del Sistema Informático ayudará a mejorar el proceso académico y la interrelación con los padres de familia.

### 3.3. VARIABLES

#### 3.3.1. Variable independiente

*Tabla 1*  
*Variable Independiente*

Variable Independiente	Definición	Dimensiones	Indicadores
	Es un conjunto de		
Sistema Informático	Funciones interrelacionadas, hardware, software y de recurso humano	Manipulación del sistema informático	El usuario realiza la gestión en el sistema

*Fuente:* Elaboración Propia

### 3.3.2. Variable dependiente

Tabla 2  
Variable Dependiente

Variable Dependiente	Definición	Dimensiones	Indicadores
Proceso académico con soporte en notificaciones	Gestión académico con notificaciones automáticas	Implementación de las notificaciones	El sistema notifica a los padres en tiempo real.

Fuente: Elaboración Propia

## **CAPÍTULO IV**

### **DISEÑO TEÓRICO**

## **CAPÍTULO IV: DISEÑO TEÓRICO**

### **4.1. ANTECEDENTES DE LA INVESTIGACIÓN**

#### **4.1.1. Antecedentes internacionales**

Mora y Vega, (2013) en el trabajo realizado en “SISTEMA INFORMÁTICO DE CONTROL DE NOTAS Y PROCESO DE MATRICULACIÓN DE LA UNIDAD EDUCATIVA WENCESLAO RIJAVEC DE LA CIUDAD DE CALCETA – CANTÓN BOLÍVAR” en la Escuela Superior Politécnica Agropecuaria de Manabí Manuel Félix López, Calceta - Ecuador.

Se encontró los siguientes resultados:

- Con la utilización de herramientas de desarrollo se logró crear el software de acuerdo al diseño establecido, con una interfaz agradable y con una funcionalidad de acuerdo a los requerimientos establecidos por el usuario.
- Mediante la elaboración de prototipos basados en el modelo iterativo e incremental se tomó en cuenta las observaciones en cada una las pruebas que complementaron el funcionamiento de un software seguro, estable y confiable.
- Las pruebas realizadas a la aplicación permitieron determinar el funcionamiento del software llegando a concluir que todas las funciones y procedimientos fueron correctamente implementados.

Gonzalez y Quintero, (2010) en el trabajo realizado en “ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA INFORMÁTICO PARA EL APOYO AL PROCESO DE ASIGNACIÓN DE LA CARGA ACADÉMICA USANDO PROGRAMACIÓN CON RESTRICCIONES” en la Universidad Tecnológica de Pereira, Pereira - Colombia.

Se encontró los siguientes resultados:

- La programación con restricciones es una técnica que se ajusta muy bien a la resolución de este tipo de problemas, puesto que las características del modelo matemático formulado guardan una similitud con las del modelo propio de la técnica, lo que se significa una reducción en la complejidad del proceso de traducción de un modelo al otro.
- El hecho de contar con un manual de usuario interactivo y entendible para el usuario, permite una mejor comprensión de la forma como se gestionan los procesos que brinda el software de distribución y una fácil adaptación a él.
- Siguiendo las metodologías especificadas en el capítulo siete (XP y Cascada), se realizaron todas las etapas para el desarrollo del software.

#### **4.1.2. Antecedentes nacionales**

Osorio N, (2016) en el trabajo realizado en “ DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE MATRÍCULA WEB USANDO SOFTWARE LIBRE EN EL CENTRO EDUCATIVO ESPAÑA, DISTRITO - BREÑA 2013” en la Universidad de Ciencias y Humanidades, Lima – Perú.

Se encontró los siguientes resultados:

- Logró disminuir la pérdida de datos, y de igual manera reducir el costo de la licencia del software a cero, ya que es un software libre.



- Al mismo tiempo, se reduce el tiempo de atención al usuario, ya que con este nuevo sistema se puede tener una interfaz amigable y una manera sencilla de utilizarla para mejorar el proceso de atención.
- También, se concluye que con la implementación del sistema de matrícula web se optimiza el tiempo de registro de acuerdo a la disponibilidad del usuario. Por último, se logró implementar el módulo de reportes con el cual el uso del sistema de matrícula web permite una mayor viabilidad, usabilidad y seguridad en la información, así como el uso de los recursos (hardware, software, trabajadores) en el centro educativo.
- De igual manera, se optimiza la toma de decisiones ya que cuenta con reportes personalizados.

Córdova J, ( 2014) en el trabajo realizado en “IMPLEMENTACIÓN DE UN SISTEMA DE MATRÍCULAS Y PAGOS PARA EL CENTRO DE INFORMÁTICA DE LA UNIVERSIDAD CESAR VALLEJO” en la Universidad de San Martín de Porres, Lima – Perú.

Se encontró los siguientes resultados:

- Se logró comprender en forma correcta y clara el proceso de matrículas y pagos del Centro de Informática.
- Lograron identificar todos los requerimientos funcionales y no funcionales asociados a los procesos de matrículas y pagos.
- Logró diseñar y construir un software escalable con todos los requerimientos analizados.
- Observando las proyecciones se lograría matricular a más de cuatro mil alumnos durante el transcurso del año.

- Teniendo en cuenta las proyecciones realizadas sobre las matriculas se lograría que la universidad obtenga ganancias de más de un millón de soles.

#### **4.1.3. Antecedentes locales**

Cubas y Nuñez, (2015) en el trabajo realizado en "PLAN ESTRATÉGICO DE SISTEMAS Y TECNOLOGÍAS DE INFORMACIÓN PARA OPTIMIZAR LA GESTIÓN EDUCATIVA DE LA INSTITUCIÓN PRIVADA AMANCIO VARONA - TUMÁN" en la Universidad Nacional Pedro Ruiz Gallo, Lambayeque – Perú.

Se encontró los siguientes resultados:

- Se hizo un estudio de las diferentes tecnologías de información existentes, logrando determinar la que mejor se adecue a los requerimientos de la institución, permitiendo así brindar un mejor servicio y desarrollo de sus clientes.
- Se logró definir los proyectos de Sistemas de Información en la Institución, basados en los requerimientos elaborados en el presente estudio. Obteniendo de esa forma un mayor conocimiento de los procesos que están implicados, su función y su desenvolvimiento en la institución, permitiendo mayor rapidez y seguridad en los procesos de la gestión educativa.

Estela E, (2016) en el trabajo realizado en “APLICACIÓN MÓVIL DE ALERTAS PARA APOYAR LA COMUNICACIÓN ENTRE LOS AGENTES EDUCATIVOS DEL COLEGIO SAN AGUSTÍN DE CHICLAYO” en la Universidad Católica Santo Toribio de Mogrovejo, Chiclayo – Perú.

Se encontró los siguientes resultados:

- Con los resultados obtenidos del análisis estadístico se pudo determinar que la aplicación móvil mejoró el tiempo de entrega de la información al padre de familia o apoderado del colegio San Agustín de Chiclayo.
- El 95% de padres de familia recibió los anuncios emitidos por las instancias del colegio dentro de las primeras 12 horas de haber sido enviados.
- El 95% de padres de familia recibió los comunicados emitidos por los profesores de sus menores hijos el mismo día en que fueron enviados.
- El 95% de padres de familia recibieron oportunamente las alertas de fechas de presentación de tareas académicas de su menor hijo.
- El 80% de padres de familia calificó como alto el nivel de la transmisión de la información haciendo uso de la aplicación móvil.

## **4.2. BASE TEÓRICA**

### **4.2.1. Conceptos generales**

A continuación, se detallarán los términos que utilizaremos en el proyecto.

#### **Proceso académico**

Los antecedentes teóricos de la organización escolar abordan diferentes estudios sobre organización escolar desde la perspectiva de la sociología moderna en el contexto educativo explicando sus dinámicas y configuraciones. La organización escolar se propone como contexto de la planeación y administración educativa; la forma como se constituye el sistema educativo y las instituciones de educación corresponden a un conjunto de elementos institucionales que estructuran el sistema educativo en el contexto social.

La administración adquiere un doble sentido, por un lado, se considera que la disciplina administrativa tiene en cuenta un conjunto de técnicas y procedimientos que racionalizan los recursos materiales para el funcionamiento del sistema educativo e institucional; y por el otro, se considera el acto administrativo como el conjunto de acciones educativas como recurso de la acción administrativa materializados en proyectos, prácticas y objetivos institucionales en relación con el acto educativo.

Como ya se ha dicho, la administración se refiere a los procedimientos de una actividad y la forma como se realiza esta. La administración es el control y manejo de los recursos materiales y humanos para ponerlos al servicio de los grandes fines y aspiraciones determinados por la política educacional. Esta concepción educacional de la administración parte de la actividad de manejo de los recursos materiales y humanos en el contexto educativo, sin embargo, la disciplina administrativa como conjunto de procesos y actividades de la empresa, hacen parte de dicho manejo de recursos materiales en el contexto educativo.

Entonces, el concepto administrativo en el campo educativo, desborda el ámbito empresarial y amplía sus acciones a actividades propias de la dinámica educativa, tales como, los procesos, proyectos, prácticas y estrategias pedagógicas que hacen posible el acto educativo en función de la educación en el contexto político. (2011).

## **Sistema informático**

Un sistema informático es un conjunto de partes o recursos formados por el Hardware, software y las personas que lo emplean, que se relacionan entre sí para almacenar y procesar información con un objetivo en común.

**Recurso físico (o de hardware):** computadoras, impresoras, escáneres, memorias, lectores de código de barras, estructura física de una red de computadoras, etc.

**Recurso lógico (o de software):** manuales de uso, sistema operativo, archivos, documentos, aplicaciones, firmware, base de datos, información de una red de computadora, etc.

**Recurso humano:** son todas las personas que forman parte del sistema, como ser de los operadores del sistema, los técnicos que lo mantienen y los usuarios finales.

La finalidad u objetivo más general de los sistemas informáticos es hacer las tareas más rápidas, flexibles y cómodas para los usuarios, empleando la tecnología informática eficientemente para tal fin. (Alegsa, 2018).

## **Aplicación web**

Una aplicación web es una aplicación o herramienta informática accesible desde cualquier navegador, bien sea a través de internet (lo habitual) o bien a través de una red local. A través del navegador se puede acceder a toda la funcionalidad.

### **Ventajas**

- ✓ No necesita ningún tipo de instalación, ya que se accede a través de un navegador.

- ✓ Multiplataforma y multidispositivo.
- ✓ La potencia no está en el dispositivo que accede por lo que, aunque no tengamos un superordenador la aplicación puede ser muy potente, hay que el peso no lo soporta el equipo desde el que se accede sino el servidor donde está alojada.
- ✓ La aplicación puede estar en la nube, con lo que sería accesible para cualquier ordenador con acceso a internet (aunque también podría ser una aplicación local en una intranet)
- ✓ Es muy adaptable, visualmente intuitiva y muy fácil de actualizar si fuera necesario.

### **La aplicación web tiene 3 partes:**

- La base de datos, que es donde se almacenan los datos de la solución.
- El código de la aplicación, es decir, la aplicación en sí, que se almacena en un servidor de aplicaciones (normalmente en la nube, pero puede estar alojado también en un servidor local)
- El usuario, que accede con cualquier dispositivo a través de un navegador. Aquí se incluye, dependiendo de la aplicación, tanto el administrador o gestor como el usuario final (en caso de que se ofrezca un servicio), que accederán a diferentes secciones según sus permisos (o incluso aplicaciones). (2018).

Cada vez es más frecuente que en cualquier tipo de entidad se necesite una solución informática ajustada a sus necesidades. Y cada vez es más común que se nombre una “Aplicación Web” como respuesta a esta necesidad.

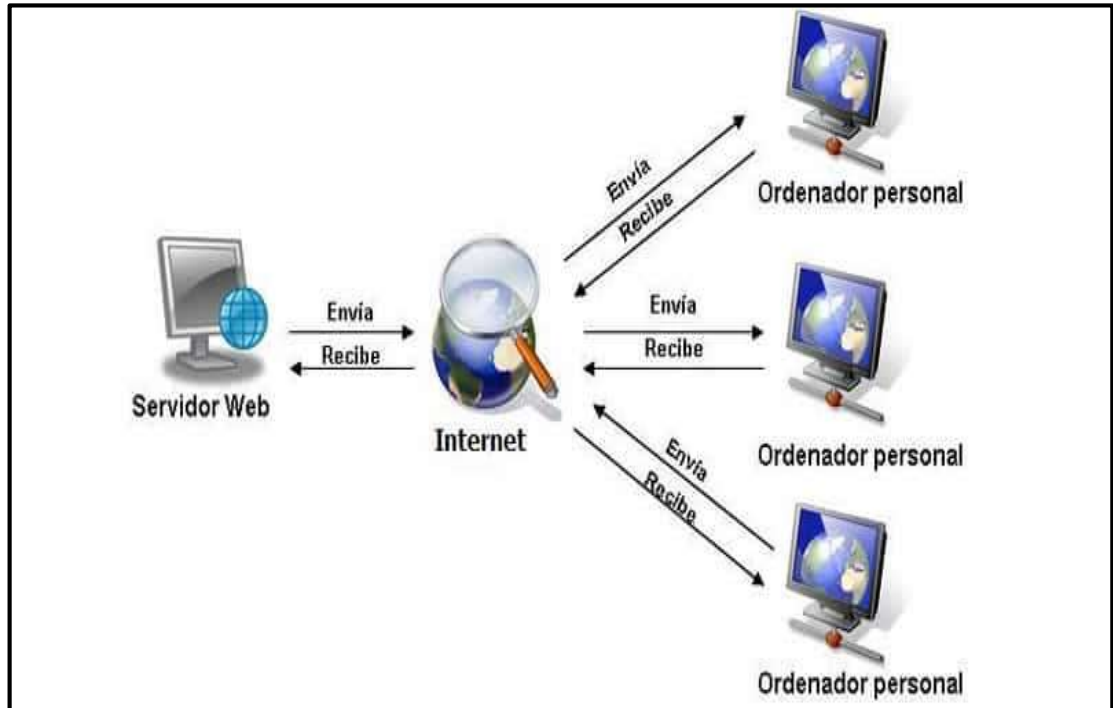


Figura 1. Visualización del proceso de Aplicación Web  
Fuente: Elaboración Propia

#### 4.2.2. Modelado de datos

A continuación, se detalla la metodología XP, fue elegida por ser una metodología ágil capaz de soportar cambios en el desarrollo de software.

##### ¿Qué es la programación extrema?

La programación extrema o Extreme Programming (PX) es un enfoque de la Ingeniería del Software formulado por Kent Beck, autor del primer libro sobre la materia, *Extreme Programming Explained: Embrace Change* 1999.

Es el más destacado de los procesos ágiles de desarrollo de software.

La programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del Desarrollo de Proyectos.

Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos.

- **Valores de la programación extrema**

**Simplicidad:** Es la base de la programación extrema. Se simplifica el diseño para agilizar el desarrollo y facilitar el mantenimiento.

Para mantener la simplicidad es necesaria la Refactorización del código, ésta es la manera de mantener el código simple a medida que crece.

**Comunicación:** Se realiza de diferentes formas, para los programadores el código comunica mejor cuanto más simple sea. Si el código es complejo hay que esforzarse para hacerlo inteligible.

Las pruebas unitarias son otra forma de comunicación ya que describen el diseño de las clases y los métodos al mostrar ejemplos concretos de cómo utilizar su funcionalidad. Los programadores se comunican constantemente gracias a la programación por parejas. La comunicación con el cliente es fluida ya que el cliente forma parte del equipo de desarrollo. El cliente decide qué características tienen prioridad y siempre debe estar disponible para solucionar dudas.

**Retroalimentación feedback:** Al estar el cliente integrado en el Proyecto, su opinión sobre el estado del proyecto se conoce en tiempo real.

Las pruebas que se realizan al software nos mantiene informados del grado de fiabilidad del sistema.

**Coraje o valentía:** Asumir retos, ser valientes ante los problemas y afrontarlos. El intentar mejorar algo que ya funciona.



Hay que ser valientes en que la programación por parejas beneficia la calidad del código sin repercutir negativamente en la productividad. La simplicidad es uno de los principios más difíciles de adoptar. Se requiere coraje para implementar las características que el cliente quiere ahora sin caer en la tentación de optar por un enfoque más flexible que permita futuras modificaciones.

**Codificar:** Es la única actividad de la que no podremos prescindir. Sin código fuente no hay programa. Por tanto, necesitamos codificar y plasmar nuestras ideas a través del código.

**Hacer pruebas:** Las características del software que no pueden ser demostradas mediante pruebas simplemente no existen. Las pruebas me dan la oportunidad de saber si lo implementé es lo que en realidad yo pensaba que había implementado. Las pruebas nos indican que nuestro trabajo funciona.

Programar y probar es más rápido que sólo programar. Puedes ganar media hora de productividad sin hacer pruebas, pero perderás mucho tiempo en la Depuración.

**Escuchar:** Tenemos que escuchar a nuestros clientes cuales son los problemas de su negocio, debemos de tener una escucha activa explicando lo que es fácil y difícil de obtener, y la Realimentación entre ambos nos ayudan a todos a entender los problemas.

**Diseñar:** El Diseño crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar. Tenemos que codificar porque sin código no hay programas, tenemos que hacer pruebas porque sin pruebas no sabemos si hemos acabado de

codificar, tenemos que escuchar, porque si no escuchamos no sabemos que codificar ni probar, y tenemos que diseñar para poder codificar.

**El juego de la Planificación:** Es un permanente diálogo entre las partes empresarial (deseable) y técnica (posible).

**Pequeñas entregas:** Cada versión debe de ser tan pequeña como fuera posible, conteniendo los requisitos de negocios más importantes.

**Metáfora:** Una metáfora es una historia que todo el mundo puede contar acerca de cómo funciona el sistema. Las metáforas ayudan a cualquier persona a entender el objeto del programa.

**Refactorización:** Cuando implementamos nuevas características en nuestros programas nos planteamos la manera de hacerlo lo más simple posible, después de implementar esta característica, nos preguntamos cómo hacer el programa más simple sin perder funcionalidad, este proceso se le denomina recodificar o refactorizar.

**Cliente deberá formar parte del equipo de desarrollo:** Se le dará poder para determinar los requisitos de la aplicación, definir la funcionalidad y dar prioridad a determinadas cosas.

Gracias a esto, habrá una fuerte interacción con los programadores, disminuyendo así el tiempo de comunicación y la cantidad de documentación a redactar. El cliente estará en el equipo durante todo el proceso de desarrollo del proyecto.

**Programación por parejas:** Según los experimentos ya realizados sobre este método. Se producen mejores y más consistentes aplicaciones a igual o menor coste.

**Integración continua:** El código se debe integrar como mínimo una vez al día, y realizar las pruebas sobre la totalidad del sistema.

**40 horas semanales:** Si los programadores tienen que estar haciendo horas extra, quiere decir que el problema está mal planteado. Es posible que haya que trabajar más horas, pero no como se debe tomar como norma. Programadores cansados – peor código.

**Estándares de codificación:** Todo el código debe estar escrito de acuerdo a un estándar de codificación. ( 2016).

La metodología XP fue elegida por ser una metodología ágil capaz de soportar cambios de requerimientos furtivos gracias al alto nivel de participación por parte del usuario, aparte de la gran velocidad de transición de la documentación al desarrollo que esta posee.

#### **4.2.3. Desarrollo**

##### **4.2.3.1. Modelo vista controlador (MVC)**

Modelo Vista Controlador (MVC) es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

Se trata de un modelo que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones, y sobre multitud de lenguajes y plataformas de desarrollo.

El Modelo que contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia.

La Vista, o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con este.

El Controlador, que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno. (2016).

MVC es un patrón de diseño enfocado a separar las responsabilidades dentro de nuestra aplicación y es muy utilizado en la web por su enfoque y las ventajas que ofrece con respecto a algunas otras formas o patrones de desarrollo de aplicaciones web.



*Figura 2. Arquitectura de MVC*  
*Fuente: Elaboración Propia*

#### **4.2.3.2. Java**

La tecnología Java se usa para desarrollar aplicaciones para un amplio alcance de entornos, desde dispositivos del consumidor hasta sistemas empresariales heterogéneos.

Como cualquier lenguaje de programación, el lenguaje Java tiene su propia estructura, reglas de sintaxis y paradigma de programación. El paradigma de programación del lenguaje Java se basa en el concepto de programación orientada a objetos (OOP), que las funciones del lenguaje soportan.

El lenguaje Java es un derivado del lenguaje C, por lo que sus reglas de sintaxis se parecen mucho a C: por ejemplo, los bloques de códigos se modularizan en métodos y se delimitan con llaves ({y}) y las variables se declaran antes de que se usen.

Estructuralmente, el lenguaje Java comienza con paquetes. Un paquete es el mecanismo de espacio de nombres del lenguaje Java.

Dentro de los paquetes se encuentran las clases y dentro de las clases se encuentran métodos, variables, constantes, entre otros. En este tutorial, aprenderá acerca de las partes del lenguaje Java.

### **El compilador Java**

Cuando usted programa para la plataforma Java, escribe el código de origen en archivos .java y luego los compila. El compilador verifica su código con las reglas de sintaxis del lenguaje, luego escribe los códigos byte en archivos .class. Los códigos byte son instrucciones estándar destinadas a ejecutarse en una Java Virtual Machine (JVM).

Al agregar este nivel de abstracción, el compilador Java difiere de los otros compiladores de lenguaje, que escriben instrucciones apropiadas para el chipset de la CPU en el que el programa se ejecutará.

### **La JVM**

Al momento de la ejecución, la JVM lee e interpreta archivos .class y ejecuta las instrucciones del programa en la plataforma de hardware nativo para la que se escribió la JVM.

La JVM interpreta los códigos byte del mismo modo en que una CPU interpretaría las instrucciones del lenguaje del conjunto. La diferencia es

que la JVM es un software escrito específicamente para una plataforma particular.

La JVM es el corazón del principio “escrito una vez, ejecutado en cualquier lugar” del lenguaje Java. Su código se puede ejecutar en cualquier chipset para el cual una implementación apropiada de la JVM está disponible. Las JVM están disponibles para plataformas principales como Linux y Windows y se han implementado subconjuntos del lenguaje Java en las JVM para teléfonos móviles y aficionados de chips.

### **El recolector de basura**

En lugar de forzarlo a mantenerse a la par con la asignación de memoria (o usar una biblioteca de terceros para hacer esto), la plataforma Java proporciona una gestión de memoria lista para usar. Cuando su aplicación Java crea una instancia de objeto al momento de ejecución, la JVM asigna automáticamente espacio de memoria para ese objeto desde el almacenamiento dinámico, que es una agrupación de memoria reservada para que use su programa. El recolector de basura Java se ejecuta en segundo plano y realiza un seguimiento de cuáles son los objetos que la aplicación ya no necesita y recupera la memoria que ellos ocupan. Este abordaje al manejo de la memoria se llama gestión de la memoria implícita porque no le exige que escriba cualquier código de manejo de la memoria. La recogida de basura es una de las funciones esenciales del rendimiento de la plataforma Java.

## **El kit de desarrollo de Java**

Cuando se descarga un kit de desarrollo de Java (JDK), obtiene, además del compilador y otras herramientas, una librería de clase completa de programas de utilidad pre construidos que lo ayudan a cumplir cualquier tarea común al desarrollo de aplicaciones. El mejor modo para tener una idea del ámbito de los paquetes y bibliotecas JDK es verificar la documentación API JDK.

## **El Java RuntimeEnvironment**

El Java RuntimeEnvironment (JRE, también conocido como el Java Runtime) incluye las bibliotecas de códigos de la JVM y los componentes que son necesarios para programas en ejecución escritos en el lenguaje Java. Está disponible para múltiples plataformas. Puede redistribuir libremente el JRE con sus aplicaciones, de acuerdo a los términos de la licencia del JRE, para darles a los usuarios de la aplicación una plataforma en la cual ejecutar su software. El JRE se incluye en el JDK. (Steven, 2012).

### **4.2.3.3. Spring**

Millones de desarrolladores de todo el mundo usan Spring Framework para crear código de alto rendimiento, fácilmente comprobable y reutilizable.

Spring framework es una plataforma Java de código abierto. Fue escrito inicialmente por Rod Johnson y se lanzó por primera vez bajo la licencia Apache 2.0 en junio de 2003.

Las características principales de Spring Framework se pueden usar para desarrollar cualquier aplicación Java, pero existen extensiones para crear aplicaciones web sobre la plataforma Java EE. Los objetivos de Spring

Framework hacen que el desarrollo de J2EE sea más fácil de usar y promueve buenas prácticas de programación al habilitar un modelo de programación basado en POJO.

- **Beneficios de usar Spring Framework**

Spring permite a los desarrolladores desarrollar aplicaciones de clase empresarial utilizando POJO. El beneficio de usar solo POJOs es que no necesita un producto de contenedor EJB, como un servidor de aplicaciones, pero tiene la opción de usar solo un contenedor robusto de servlets como Tomcat o algún producto comercial.

Spring está organizado de forma modular. A pesar de que la cantidad de paquetes y clases es sustancial, solo debe preocuparse por los que necesita e ignorar el resto.

Spring no reinventa la rueda, sino que realmente utiliza algunas de las tecnologías existentes, como varios marcos ORM, marcos de registro, JEE, temporizadores de cuarzo y JDK y otras tecnologías de visualización.

Probar una aplicación escrita con Spring es simple porque el código dependiente del entorno se traslada a este marco. Además, mediante el uso de JavaBeanstyle POJO, se vuelve más fácil utilizar la inyección de dependencia para inyectar datos de prueba.

El marco web de Spring es un framework MVC web bien diseñado, que proporciona una gran alternativa a los frameworks web como Struts u otros frameworks web sobre – diseñados o menos populares.

Spring proporciona una API conveniente para traducir excepciones específicas de la tecnología (generadas por JDBC, Hibernate o JDO, por ejemplo) en excepciones consistentes y no verificadas.



Los contenedores livianos de IoC tienden a ser livianos, especialmente cuando se comparan con los contenedores EJB, por ejemplo. Esto es beneficioso para desarrollar y desplegar aplicaciones en computadoras con memoria limitada y recursos de CPU.

Spring proporciona una interfaz de gestión de transacciones coherente que puede reducirse a una transacción local (utilizando una única base de datos, por ejemplo) y ampliarse a transacciones globales (utilizando JTA, por ejemplo).

### **Inyección de dependencia (DI)**

La tecnología con la que más se identifica Spring es el sabor de inyección de control de Inyección de Dependencia (DI). La inversión de control (IoC) es un concepto general y se puede expresar de muchas maneras diferentes. La Inyección de Dependencia es simplemente un ejemplo concreto de Inversión de Control.

### **Programación orientada a aspectos (AOP)**

Uno de los componentes claves de Spring es el marco de Programación Orientada a Aspectos (AOP). Las funciones que abarcan múltiples puntos de una aplicación se llaman preocupaciones transversales y estas preocupaciones transversales están conceptualmente separadas de la lógica de negocios de la aplicación.

La unidad clave de modularidad en OOP es la clase, mientras que en AOP la unidad de modularidad es el aspecto. DI le ayuda a desacoplar sus objetos de aplicación entre sí, mientras que AOP le ayuda a desacoplar las preocupaciones transversales de los objetos que afectan.

El módulo AOP de Spring Framework proporciona una implementación de programación orientada a aspectos que le permite definir interceptores de métodos y puntos de corte para desvincular el código que implementa la funcionalidad que debe separarse. (2018).

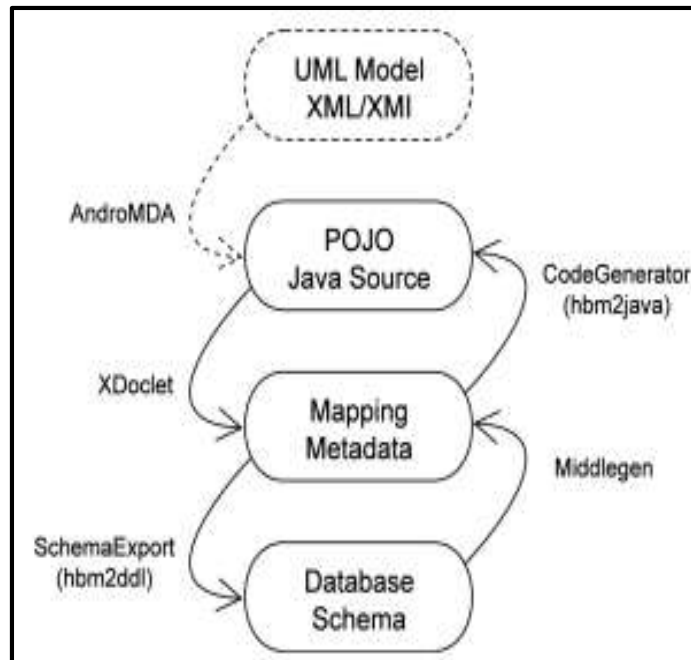
#### **4.2.3.4. Hibernate**

Hibernate es una herramienta de Mapeo objeto-relacional (ORM) para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans de las entidades que permiten establecer estas relaciones.

Hibernate es un Framework que agiliza la relación entre la aplicación y la base de datos. (2013).

#### **Hibernate query language (HQL)**

Es un lenguaje de consulta orientado a objetos, similar a SQL, pero en lugar de operar en tablas y columnas, HQL trabaja con objetos persistentes y sus propiedades. Las consultas HQL son traducidas por Hibernate en consultas SQL convencionales, que a su vez realizan acciones en la base de datos. (2018).



*Figura 3. Herramientas Usadas con Hibernate*  
*Fuente: Elaboración Propia*

#### **4.2.3.5. Bootstrap**

Bootstrap, es un framework originalmente creado por Twitter, que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice.

Esta técnica de diseño y desarrollo se conoce como Responsive Desing o diseño Adaptativo. (2011).

#### **Características**

- ✓ Bootstrap ofrece una serie de plantillas CSS y ficheros Javascript que nos permiten integrar el framework de forma sencilla y potente en nuestros proyectos webs.
- ✓ Permite crear interfaces que se adapten a los diferentes navegadores, tanto de escritorio como tablets y móviles a distintas escalas y resoluciones.

- ✓ Se integra perfectamente con las principales librerías Javascript.
- ✓ Ofrece un diseño sólido usando LESS y estándares como CSS3/HTML5
- ✓ Es un framework ligero que se integra de forma limpia en nuestro proyecto actual.
- ✓ Funciona con todos los navegadores, incluido Internet Explorer usando HTML Shim para que reconozca los tags HTML5.
- ✓ Dispone de distinto layout predefinidos con estructuras fijas a 940 píxeles de distintas columnas o diseños fluidos. (2012).

#### **4.2.4. Base de datos**

Un sistema de base de datos es básicamente un sistema computarizado para guardar registros; es decir es un sistema computarizado cuya finalidad general es almacenar información y permitir a los usuarios recuperar y actualizar esa información con base en peticiones. La información en cuestión puede ser cualquier cosa que sea de importancia para el individuo u organización; en otras palabras, todo lo que sea necesario para auxiliarse en el proceso general de su administración.

Un sistema de base de datos comprende cuatro componentes principales:

- ✓ Datos
- ✓ Hardware
- ✓ Software
- ✓ Usuarios

#### **Beneficios del enfoque de la base de datos:**

- Los datos pueden compartirse.
- Es posible reducir la redundancia.

- Es posible (hasta cierto grado) evitar la inconsistencia.
- Es posible brindar un manejo de transacciones.
- Es posible mantener la integridad.
- Es posible hacer cumplir la seguridad.
- Es posible equilibrar los requerimientos en conflicto.
- Es posible hacer cumplir los estándares. (Date, 2001).

#### **4.2.4.1. MySQL**

Inicialmente, MySQL se hizo muy popular debido a su velocidad y a su simplicidad. Pero también hubo críticas, ya que carecía de funciones como transacciones y compatibilidad con claves secundarias. MySQL continuó su desarrollo y no sólo incluyó estos elementos sino también otras funciones como duplicación, subconsultas, procedimientos almacenados, vistas y desencadenadores. Estas características permitieron que MySQL se abriera paso en el mundo de las aplicaciones empresariales. Como resultado, los usuarios empiezan a tener cuenta MySQL.

MySQL es portable y se ejecuta en sistemas operativos comerciales (como Mac OS X, HP – UX y Windows) y en hardware como servidores empresariales. Es más, su rendimiento supera al de cualquier otro sistema de base de datos con el que lo compare, y puede procesar bases de datos de gran tamaño con miles de millones de filas. En el mundo empresarial, la presencia de MySQL sigue aumentando ya que las empresas descubren su capacidad para solucionar sus necesidades de base de datos por mucho menos de lo que solían pagar por asistencia y licencias comerciales.

## **Herramientas proporcionadas con MySQL**

Las distribuciones de MySQL incluyen las siguientes herramientas:

**Un servidor SQL:** El motor de MySQL que permite acceder a las bases de datos.

Programas cliente y utilidades: Incluye un programa cliente interactivo que le permite introducir consultas directamente y ver los resultados. También existen otros programas administrativos y utilidades para ejecutar su sitio. Uno le permite monitorizar y controlar el servidor; el resto le permite importar datos, realizar copias de seguridad, comprobar problemas en tablas, etc.

**Una biblioteca cliente:** Puede crear programas cliente C porque la biblioteca está en C, pero también se puede vincular a otros procesadores de lenguaje como Perl, PHP o Ruby como base para interfaces MySQL en dichos lenguajes.

Además del software proporcionado con MySQL, los usuarios diseñan software para mejorar su productividad, programas que suelen compartir. Como resultado, dispone de acceso a diversas herramientas de terceros que facilitan el uso de MySQL o amplían sus prestaciones en campos como el desarrollo de sitios Web.

## **Por qué seleccionar MySQL**

Si buscas un sistema de administración de base de datos gratuito o de bajo coste, puede escoger entre varios, como por ejemplo MySQL, PostgreSQL o SQLite. Al comparar MySQL con otros sistemas de base de datos, piense en sus necesidades. Debe tener en cuenta el rendimiento, la

compatibilidad, sus funciones (compatibilidad con SQL o extensiones ), condiciones de licencia y restricciones , y el precio. Dados estos factores, las características de MySQL resultan muy atractivas:

**Velocidad:** MySQL es rápido. Sus programadores afirman que es el sistema de base de datos más rápido que existe.

**Facilidad de uso:** MySQL es un sistema de alto rendimiento, pero relativamente sencillo, y mucho menos complejo de configurar y administrar que otros sistemas de mayor tamaño.

**Compatibilidad con el lenguaje de consultas:** MySQL comprende SQL (Lenguaje estructurado de consultas), el lenguaje estándar de todos los sistemas modernos de base de datos.

**Prestaciones:** El servidor MySQL es de subprocesamiento múltiple, de modo que varios clientes pueden conectarse al mismo tiempo. Cada cliente puede utilizar varias bases de datos simultáneamente. Puede acceder a MySQL de forma interactiva con varias interfaces que le permiten introducir consultas y ver los resultados: clientes de línea de comandos, navegadores Web o clientes GUI. Además, existen interfaces de programación para diversos lenguajes como C, Perl, Java, PHP, Python y Ruby.

También puede acceder a MySQL por medio de aplicaciones compatibles con ODBC y .NET (protocolos desarrollados por Microsoft). De este modo puede utilizar software cliente pre configurado o crear sus propias aplicaciones personalizadas.

**Conectividad y seguridad:** MySQL está preparado para redes y se puede acceder a las bases de datos a través de Internet, de modo que puede compartir sus datos con terceros desde cualquier ubicación. Pero MySQL

también cuenta con control de acceso para proteger los datos de los usuarios. Para reforzar la seguridad, admite conexiones codificadas a través del protocolo SSL.

**Portabilidad:** MySQL se ejecuta en distintas variedades de Unix y Linux, así como en otros sistemas como Windows y NetWare. MySQL se puede ejecutar en servidores comerciales o en equipos personales (incluso en dispositivos manuales).

**Tamaño reducido:** El tamaño de la distribución MySQL es reducido si lo comparamos con la cantidad de espacio en disco que ocupan otros sistemas de base de datos.

**Disponibilidad y coste:** MySQL es un proyecto de código abierto disponible bajo distintas licencias. Por un lado, está disponible bajo la licencia general pública GNU, de modo que no supone coste alguno para uso doméstico. Por otra parte, para organizaciones que prefieran o necesiten condiciones formales o no limitarse a las condiciones de GPL, existen licencias comerciales. (DuBois, 2009).

## **SQL**

El acrónimo SQL procede de Structured Query Language, que podríamos traducir como lenguaje estructurado de consultas. Se trata, por tanto, de un lenguaje de ordenador, un grupo de palabras, a las que se llama normalmente instrucciones u órdenes, que se combinan conforme a unas reglas sintácticas y semánticas para dar forma a frases conocidas como sentencias. No hay mucha diferencia, consecuentemente, con una lengua hablada, salvo por la gran simplicidad de SQL ya que el número de palabras y reglas es muy pequeño si lo comparamos con una lengua como podría



ser el castellano. Otra diferencia, lógica y obvia, es que SQL es una lengua artificial, creada a medida para un determinado propósito: facilitar la comunicación entre las personas y los programas que gestionan las bases de datos.

Actualmente SQL es un estándar ISO, algo que representa una cierta garantía en cuanto a la uniformidad y consistencia de la implementación del lenguaje en los productos de distintos fabricantes. (Charte, 2010).

#### **4.2.5. Herramientas informáticas**

##### **4.2.5.1. Erwin**

La mejor manera de diseñar e implementar bases de datos relacionales y apoyar el desarrollo de aplicaciones, Erwin Data Modeler (DM) sigue siendo el estándar de oro. Las organizaciones más grandes del mundo y las principales marcas lo usan para descubrir, diseñar, visualizar, estandarizar e implementar activos de datos de alta calidad a través de una interfaz gráfica intuitiva. Con Erwin DM, visualice “cualquier dato” desde “cualquier lugar” para obtener consistencia, claridad y reutilización de artefactos a través de integración de datos a gran escala, administración de datos maestros, Big Data e iniciativas de inteligencia / análisis empresarial. La visibilidad de dónde y cómo se deben usar los datos como parte de un repositorio central convierte a Erwin DM en la clave para un gobierno de datos efectivo. (2018).

##### **4.2.5.2. NetBeans IDE**

El NetBeans IDE es un IDE de código abierto escrito completamente en Java usando la plataforma NetBeans. El NetBeans IDE soporta el

desarrollo de todos los tipos de aplicación Java (J2SE, EJB y aplicaciones móviles).

NetBeans es una herramienta que se utiliza para desarrollar aplicaciones Web, Móvil y de escritorio para diferentes lenguajes de programación como son Java, C++, Ruby y PHP entre otros. Es de código abierto, es multiplataforma, multilenguaje y contiene servidores web.

El objetivo del IDE es ayudar a la integración de los lenguajes de programación con las plataformas de los sistemas operativo o entorno de programación, facilita el diseño y desarrollo de una aplicación de sistemas informáticos escritorio, web o móvil y a su vez ayuda la productividad de la persona ya sea programador o desarrollador en el momento de la creación, actualización, compilación, depuración prueba e implementación de aplicaciones informáticas. (Mendoza, 2015).

#### **4.2.5.3. Apache**

Apache es un servidor web HTTP de código abierto para la creación de páginas y servicios web. Es un servidor multiplataforma, gratuito, muy robusto y que destaca por su seguridad y rendimiento.

El servidor Apache se desarrolla dentro del proyecto HTTP Server (http) de la Apache Software Foundation.

Apache es altamente configurable, admite bases de datos de autenticación y negociado de contenido, aunque carece de una interfaz gráfica que ayude en su configuración.

Apache es una aplicación que permite montar un servidor web en cualquier equipo y casi cualquier sistema operativo.

Apache soporta PHP como lenguaje de programación. Con los módulos adecuados, Apache puede soportar también ASP. (2017).

En cuanto a la administración los archivos de configuración de Apache están en ASCII, por lo que tiene un formato simple, y pueden ser editados tan solo con un editor de texto. Estos son transferibles, lo que permite la clonación efectiva de un servidor. El servidor puede ser administrativo vía línea de comandos, lo que hace la administración remota muy conveniente.

#### **4.3. Definición de términos y conceptos**

**Proceso académico:** Proceso orientado a mejorar los proyectos educativos institucionales y los procesos pedagógicos, con el fin de responder a las necesidades educativas locales y regionales.

**Sistema:** Un sistema es un conjunto de elementos relacionados entre sí, de forma tal que un cambio en un elemento afecta al conjunto de todos ellos.

**Sistema informático:** Un sistema informático es un sistema que permite almacenar y procesar información, conjunto de partes interrelacionadas.

**Aplicación web:** Es un programa informático que es accedido vía web por una red como internet o una intranet.

**Modelo:** Es una representación de un objeto, sistema o idea, de forma diferente al de la entidad misma. Un modelo de un objeto puede ser una réplica exacta de éste o una abstracción de las propiedades dominantes del objeto.

**Controlador:** Es un programa informático que permite al sistema operativo interaccionar con un periférico.

**Servidor:** Computadora central en un sistema de red que provee servicios a otras computadoras.

**Base de datos:** Una colección de datos, ordenados e interrelacionados.

**Framework:** Es una estructura conceptual y tecnológica de soporte definida, en la cual otro proyecto de software puede ser organizado y desarrollado.

**Herramienta ORM:** Es un modelo de programación para convertir datos entre el lenguaje de programación orientado a objetos y el sistema de base de datos relacional.

**POJO:** Objeto Java Plano Antiguo, sirve para enfatizar el uso de clases simples y que no dependen de un framework en especial.

**DI:** Inyección de Dependencia, patrón de diseño orientado a objetos.

**Lenguaje HQL:** Hibernate utiliza un lenguaje de consulta llamado HQL similar a SQL, HQL es definitivamente orientado a objetos.

**GUI:** Interfaz gráfica de usuarios, de manera que posibilitan la interacción de un sistema con los usuarios utilizando formas gráficas e imágenes.

## **CAPÍTULO V**

### **DESARROLLO DE LA PROPUESTA**

## **CAPÍTULO V: DESARROLLO DE LA PROPUESTA**

### **5.1. TÉCNICAS DE RECOLECCIÓN DE DATOS**

#### **5.1.1. Observación**

Se realizó al momento de ver con los recursos que cuenta y el flujo de procesos y actividades que son realizadas en la Institución Educativa. Realizando la investigación de campo se procedió al análisis y estudio detallado del proceso que se realiza en la gestión académica. Esta observación se realizó en el área administrativa donde se efectúan dicho proceso y actividades.

#### **Conductas a observar**

- En qué forma se lleva a cabo el manejo de datos.
- Las definiciones con la que cuenta el control de datos actualmente.
- En qué forma podemos mejorar y sistematizar el manejo de control de datos.

#### **5.1.2. Entrevista**

Se realizaron una serie de visitas a la Institución Educativa Miguel Angel Buonarroti, que permitieron obtener conocimientos de los procesos internos realizados en la institución. En entrevista con el Director y la secretaria se adquirió información importante de la misma, con el propósito de identificar los requerimientos necesarios para el desarrollo del software.

**Entrevistado:**

---

**Entrevistador:**

---

**Objetivo:** Recopilar información relevante de la Institución Educativa Miguel Angel Buonarroti y de los procesos que se realizan en el mismo para tomarlos como puntos de partida en el desarrollo del trabajo de investigación.

- 1) ¿Quién está al frente de la Institución Educativa y el área administrativa?
- 2) ¿Cómo está organizado la Institución Educativa y el área administrativa?
- 3) ¿Cuál es el volumen de información que genera habitualmente esta Institución?
- 4) ¿Cómo se maneja en la actualidad el procesamiento de la información?
- 5) ¿Cuáles son los procesos generales que se realizan en esta Institución?
- 6) ¿Cuál es el medio de información para el padre de familia sobre el rendimiento de su menor hijo?
- 7) A su parecer ¿Sería de gran ayuda contar con una herramienta tecnológica que facilite el procesamiento de información? ¿Porque?
- 8) En termino generales ¿Qué expectativas debería cumplir dicha herramienta tecnológica?

Además de la entrevista antes mencionada, se planificaron varias reuniones con el Ing. Franklin Terán cuyo resultado fue la recopilación de la información más relevante sobre las actividades, datos, procesos y necesidades a los cuales el software debería dar soluciones óptimas, Durante conversaciones se fue analizando y clasificando conjuntamente la información obtenida, para ir estableciendo las características y funcionalidad del sistema de acuerdo a los diferentes procesos y actividades que en la institución se realizan; de igual forma se eliminaron los posibles factores de riesgo con respecto a la implementación y funcionamiento del sistema.

**Se logró determinar factores como:**

- La información de las actividades y procesos realizados en la institución educativa no se encuentra organizada y clasificada adecuadamente.
- Los reportes académicos lo realizan con pérdida de tiempo en la búsqueda de información.
- El medio de información al padre de familia sobre el rendimiento de su menor hijo es mediante la agenda educativa.
- El proceso de matrícula lo realizan manualmente en fichas y archivadas en folder utilizando la información que se muestra en la Figura 4.



# Ficha de Matricula



**COLEGIO PRIVADO**  
**"MIGUEL ANGEL BUONARROTTI"**  
 RD N° 0707-86 - PRIMARIA - SECUNDARIA  
 JOSÉ LEONARDO ORTIZ



## FICHA DE PROCESO DE MATRICULA

### 201\_\_

ALUMNO(A):.....  
 GRADO:..... SECCION:..... NIVEL:.....  
 FECHA DE NAC.:...../...../..... EDAD:..... SEXO:.....  
 LUGAR DE NAC. DPTO.:..... PROV.:..... DISTRITO:.....  
 DNI:.....  
 DIRECCION:.....  
SITUACION DEL NIÑO:  
 INGRESANTE:..... PROMOVIDO:..... REPITENTE:..... REENTRANTE:.....  
ALUMNO NUEVO:  
 INSTITUCION EDUCATIVA DE PROCEDENCIA:.....  
 CODIGO MODULAR DE LA I.E. DE PROCEDENCIA:.....  
 OBSERVACION:.....

### DATOS DE LOS FAMILIARES

DATOS	PADRE	MADRE	APODERADO	GARANTE
APELLIDO PATERNO				
APELLIDO MATERNO				
NOMBRES				
FECHA DE NAC.				
DNI				
GRADO DE INSTRUCCIÓN				
Ocupacion				
TELEFONO				
DOMICILIO				
LUGAR DE NAC.				

Av. Villa Hermosa #700 - José Leonardo Ortiz  
 Tel: (074) 772196 RPM: #953568667

Figura 4. Ficha de Matricula  
 Fuente: Elaboración Propia

## 5.2 CONFIGURACIONES PARA EL DESARROLLO DEL SISTEMA

Para desarrollar el presente proyecto se hizo uso de múltiples herramientas de programación y base de datos, dichas herramientas son en su mayoría complementos estandarizados que integrados formarán una robusta arquitectura de software.

### 5.2.1. Base de datos

Para la implementación de la base de datos se tendrán en cuenta herramientas tanto en el diseño (modelado) como en la gestión de la misma.

#### 5.2.1.1. Diseño

Para el diseño de la base de datos utilizamos la clásica herramientas ERWIN en su versión 7.3 por ser una herramienta de fácil manejo.

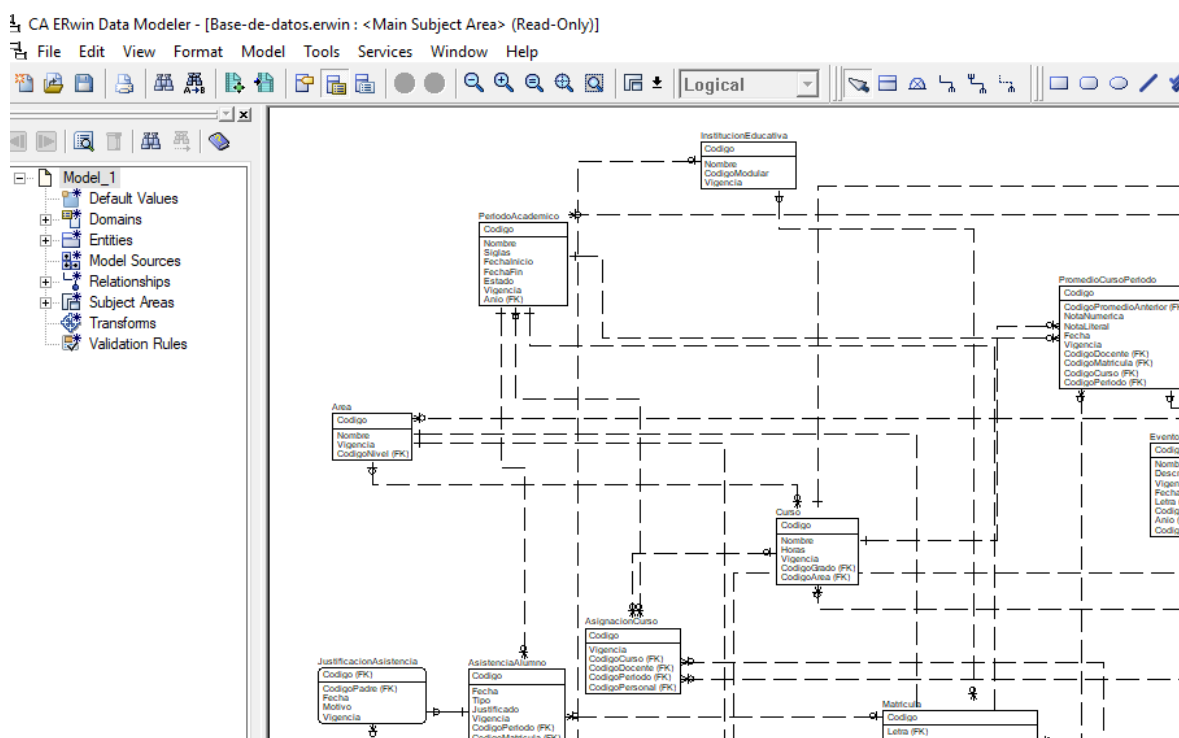
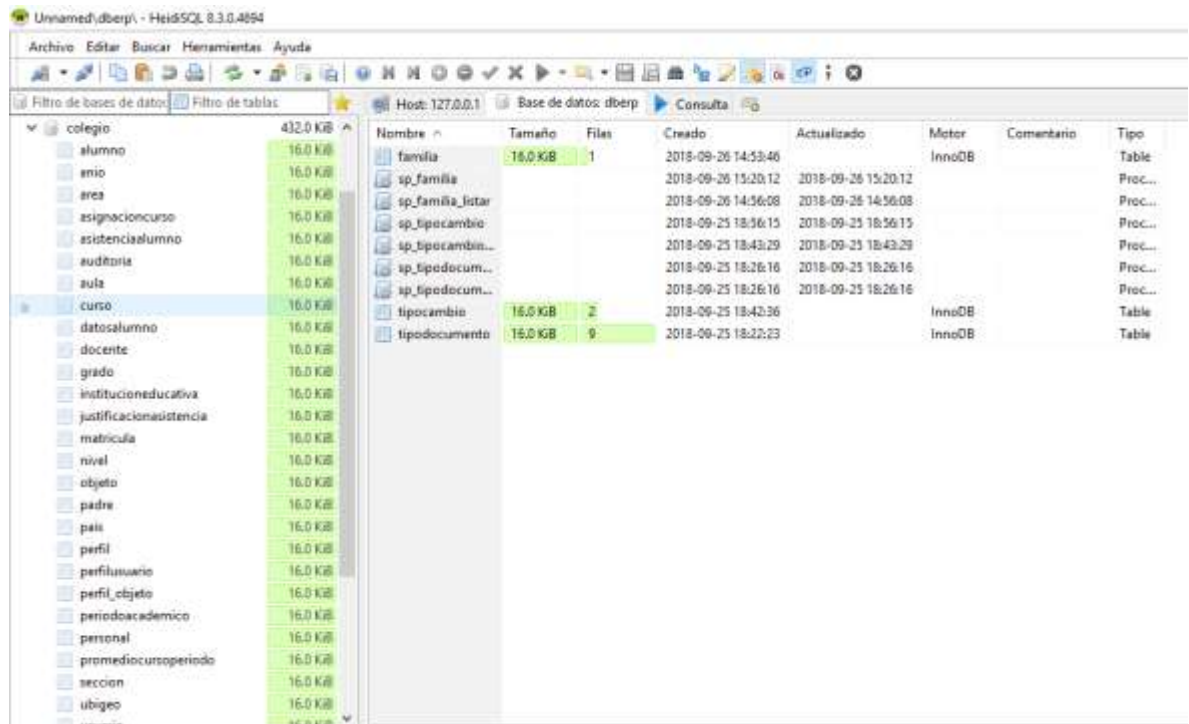


Figura 5. Base de Datos Modelada en ERWIN  
Fuente: Elaboración Propia

### 5.2.1.2. Gestor de base de datos

El gestor de base de datos es una herramienta que además de ligera es fácil de utilizar, HeidiSQL, nos permite gestionar múltiples Bases de Datos



Unamed\dberp\ - HeidiSQL 8.3.0.4854

Archivo Editar Buscar Herramientas Ayuda

Filtro de bases de datos Filtro de tablas

Host: 127.0.0.1 Base de datos: dberp Consulta

Nombre	Tamaño	Files	Creado	Actualizado	Motor	Comentario	Tipo
familia	16.0 KB	1	2018-09-26 14:53:46		InnoDB		Table
sp_familia			2018-09-26 15:20:12	2018-09-26 15:20:12			Proc...
sp_familia_listar			2018-09-26 14:56:08	2018-09-26 14:56:08			Proc...
sp_tipocambio			2018-09-25 18:56:15	2018-09-25 18:56:15			Proc...
sp_tipocambio...			2018-09-25 18:43:29	2018-09-25 18:43:29			Proc...
sp_tipodocum...			2018-09-25 18:26:16	2018-09-25 18:26:16			Proc...
sp_tipodocum...			2018-09-25 18:26:16	2018-09-25 18:26:16			Proc...
tipocambio	16.0 KB	2	2018-09-25 18:42:36		InnoDB		Table
tipodocumento	16.0 KB	9	2018-09-25 18:22:23		InnoDB		Table

Mysql.

Figura 6. Tablas de la Base de Datos en el Gestor de Base HeidiSQL

Fuente: Elaboración Propia

### 5.2.2. Software

Para el desarrollo del actual proyecto se utilizaron herramientas de programación (frameworks) tanto para la persistencia de datos como para el controlador y las interfaces, estos frameworks integrados hacen posible una arquitectura de programación sólida.

## Entorno de desarrollo (Netbeans 8.2)

Es un IDE de programación que en este caso nos permitiría desarrollar el código fuente del proyecto escrito en el lenguaje de programación JAVA.

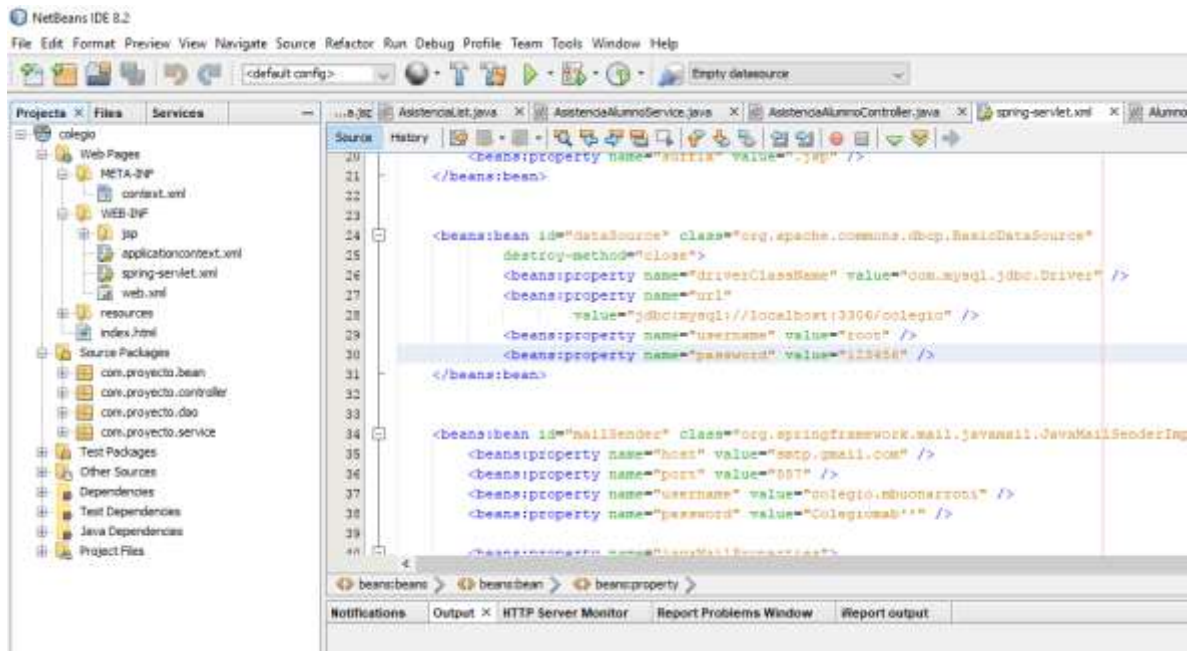


Figura 7. Archivo de Configuración para Spring ( spring – servlet.xml)  
Fuente: Elaboración Propia

## Configuración del framework de desarrollo (Spring)

Para desarrollar nuestro sistema creamos un proyecto de tipo MAVEN que nos ayudará a gestionar la compilación del código fuente a través de las dependencias establecidas en el archivo de tipo XML llamado pom.xml.

En el archivo pom.xml se definen las configuraciones de compilación del proyecto así como cuáles son las dependencias y versiones de las librerías que utilizaremos (Spring, Hibernate, Mysql, Apache, etc.).

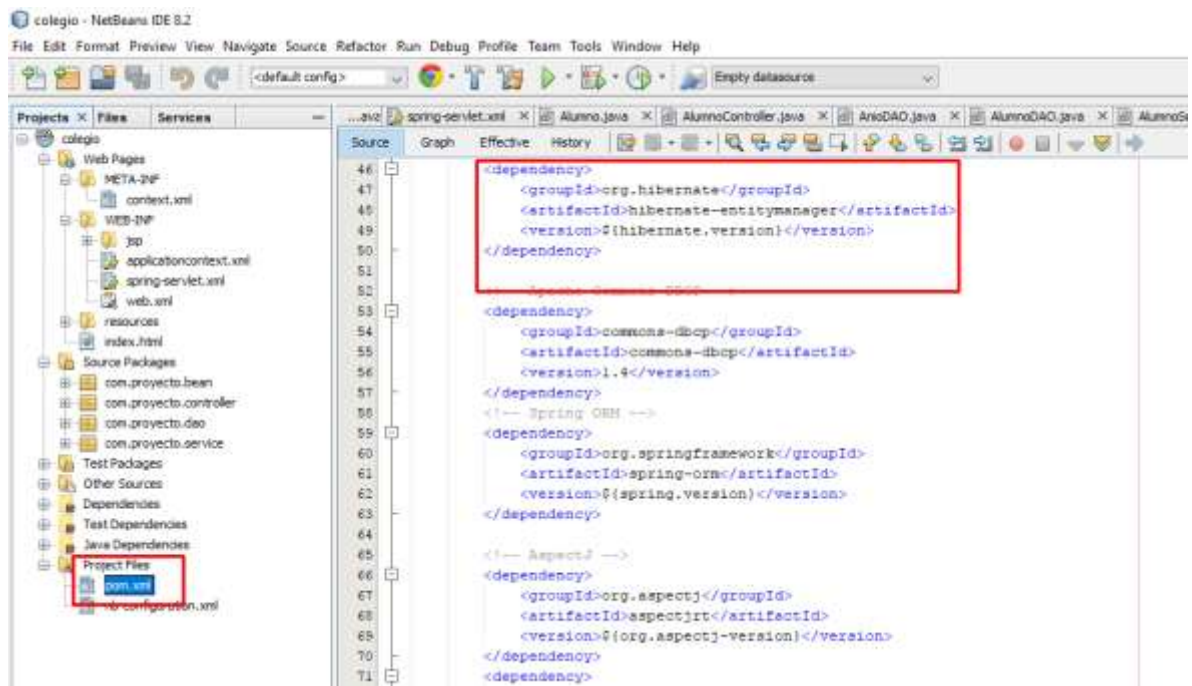


Figura 8. Archivo de Configuración de la Aplicación Maven (pom.xml)  
Fuente: Elaboración Propia

Después de configurar el archivo POM configuraremos nuestros archivos necesarios para que spring funcione en nuestra aplicación.

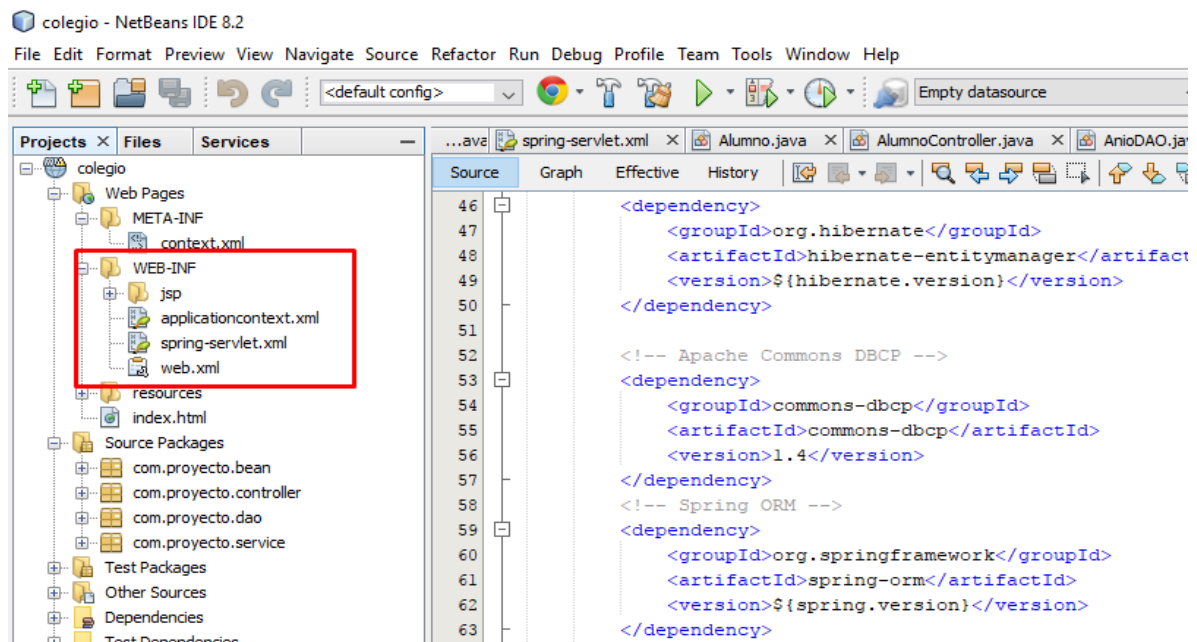


Figura 9. Estructura de Archivos de la Carpeta WEB - INF  
Fuente: Elaboración Propia

1. Agregamos la carpeta resources que es la que va a contener las imágenes, funciones y además los archivos necesarios para que nuestra plantilla bootstrap funcione correctamente, previo a esto se descargó una plantilla bootstrap libre de la web.

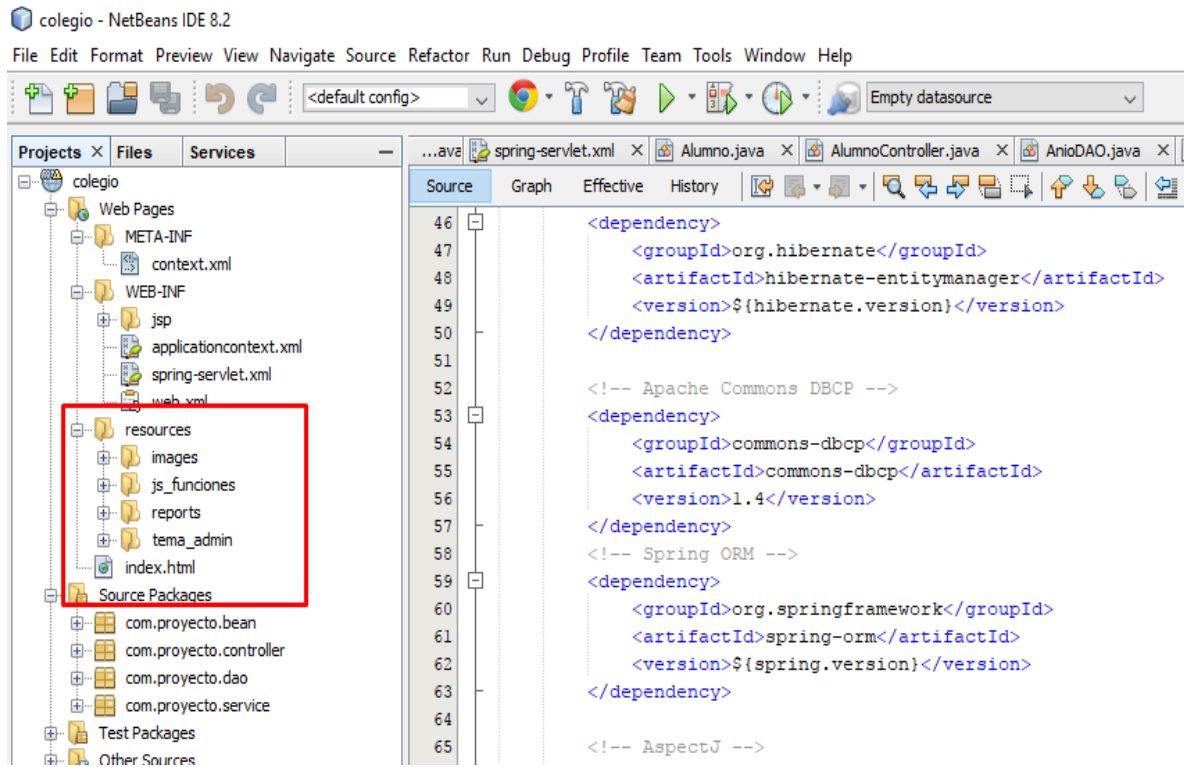


Figura 10. Estructura de Archivos de la Carpeta Resources  
Fuente: Elaboración Propia



- Después agregamos las cuatro capas que utilizaremos en el proyecto dentro de la carpeta Source Packages (bean, controller, dao, service).

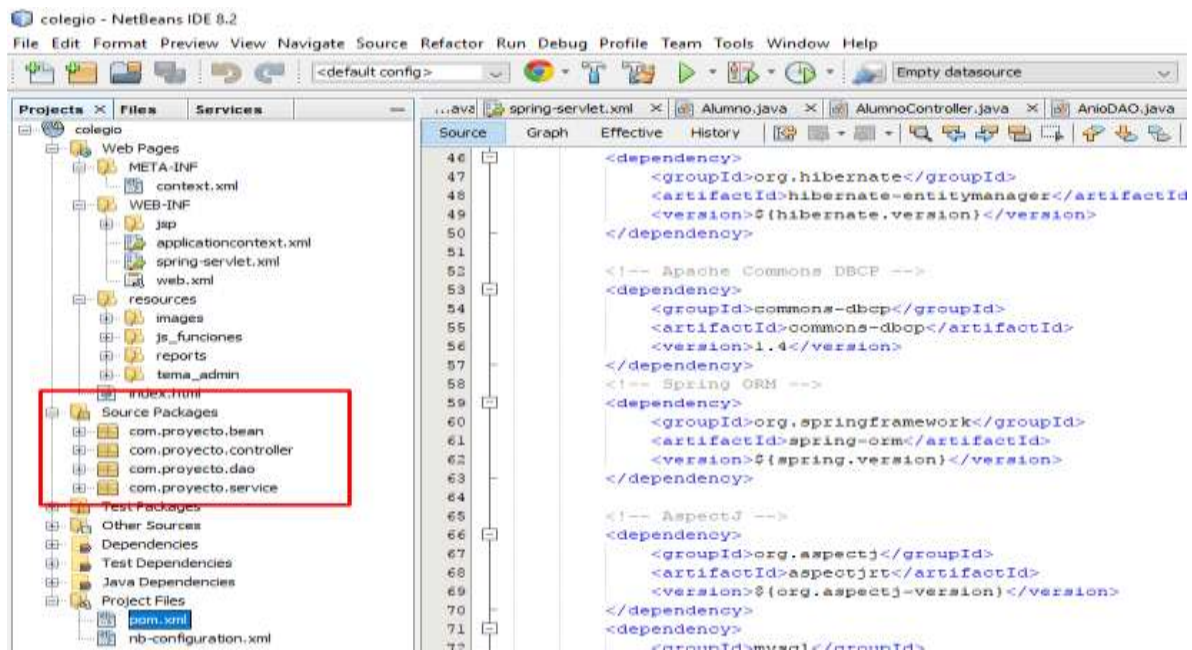
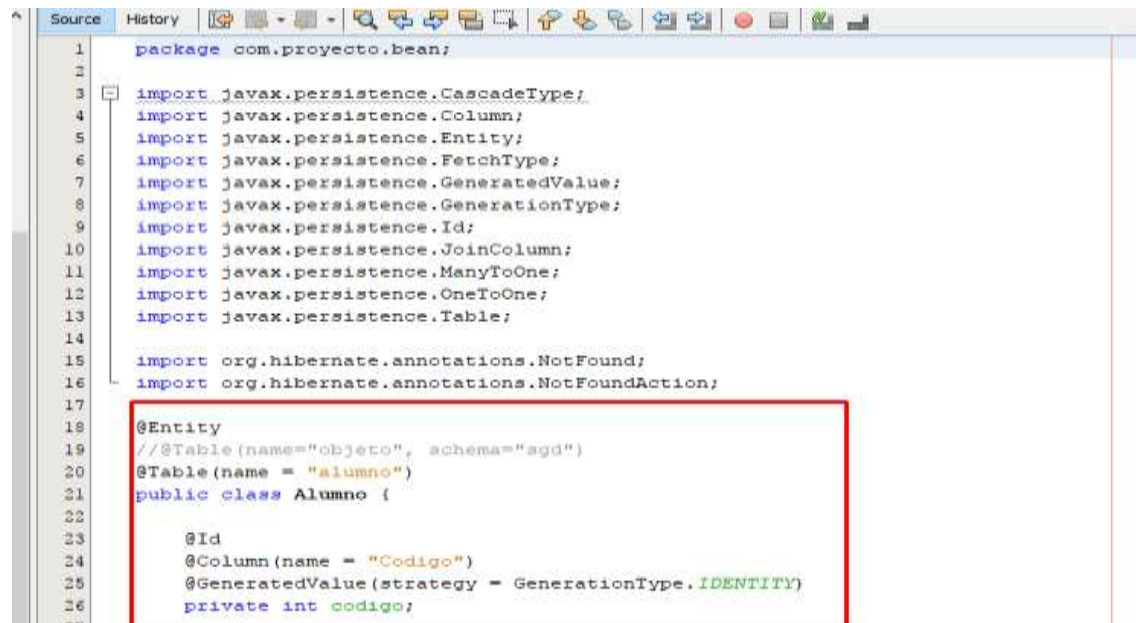


Figura 11. Estructura de Archivos de la Carpeta Source Packages

Fuente: Elaboración Propia

- a. **Bean:** Esta capa contiene los JavaBeans los cuales nos permiten agregar las propiedades que representan a las tablas de la base de datos. Estos objetos son gestionados directamente por Hibernate.



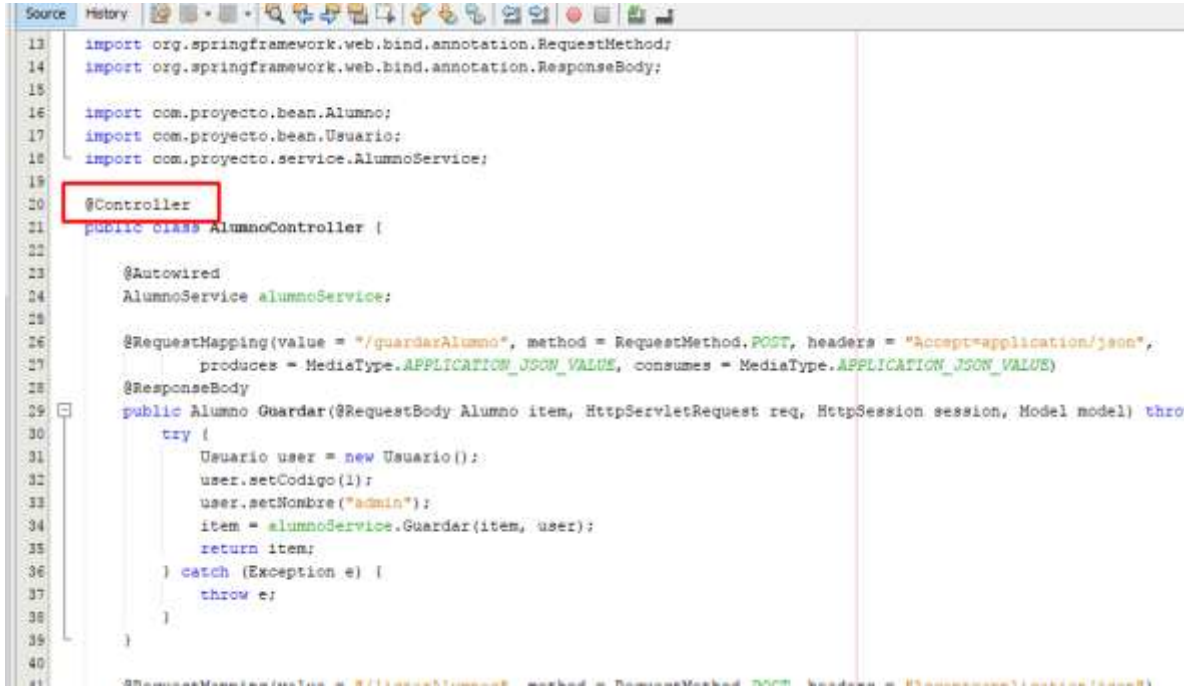
```
1 package com.proyecto.bean;
2
3 import javax.persistence.CascadeType;
4 import javax.persistence.Column;
5 import javax.persistence.Entity;
6 import javax.persistence.FetchType;
7 import javax.persistence.GeneratedValue;
8 import javax.persistence.GenerationType;
9 import javax.persistence.Id;
10 import javax.persistence.JoinColumn;
11 import javax.persistence.ManyToOne;
12 import javax.persistence.OneToOne;
13 import javax.persistence.Table;
14
15 import org.hibernate.annotations.NotFound;
16 import org.hibernate.annotations.NotFoundAction;
17
18 @Entity
19 // @Table(name="objeto", schema="agd")
20 @Table(name = "alumno")
21 public class Alumno {
22
23     @Id
24     @Column(name = "Codigo")
25     @GeneratedValue(strategy = GenerationType.IDENTITY)
26     private int codigo;
```

Figura 12. Archivo Bean para la Entidad Alumno

Fuente: Elaboración Propia



- b. **Controller:** Esta capa contiene a los controladores del sistema, los que nos permiten la navegación por los diferentes métodos haciendo uso de la anotación `@Controller` propia del framework Spring.



```
13 import org.springframework.web.bind.annotation.RequestMethod;
14 import org.springframework.web.bind.annotation.ResponseBody;
15
16 import com.proyecto.bean.Alumno;
17 import com.proyecto.bean.Usuario;
18 import com.proyecto.service.AlumnoService;
19
20 @Controller
21 public class AlumnoController {
22
23     @Autowired
24     AlumnoService alumnoService;
25
26     @RequestMapping(value = "/guardarAlumno", method = RequestMethod.POST, headers = "Accept=application/json",
27         produces = MediaType.APPLICATION_JSON_VALUE, consumes = MediaType.APPLICATION_JSON_VALUE)
28     @ResponseBody
29     public Alumno Guardar(@RequestBody Alumno item, HttpServletRequest req, HttpSession session, Model model) thro
30     {
31         try {
32             Usuario user = new Usuario();
33             user.setCodigo(1);
34             user.setNombre("admin");
35             item = alumnoService.Guardar(item, user);
36             return item;
37         } catch (Exception e) {
38             throw e;
39         }
40     }
41 }
```

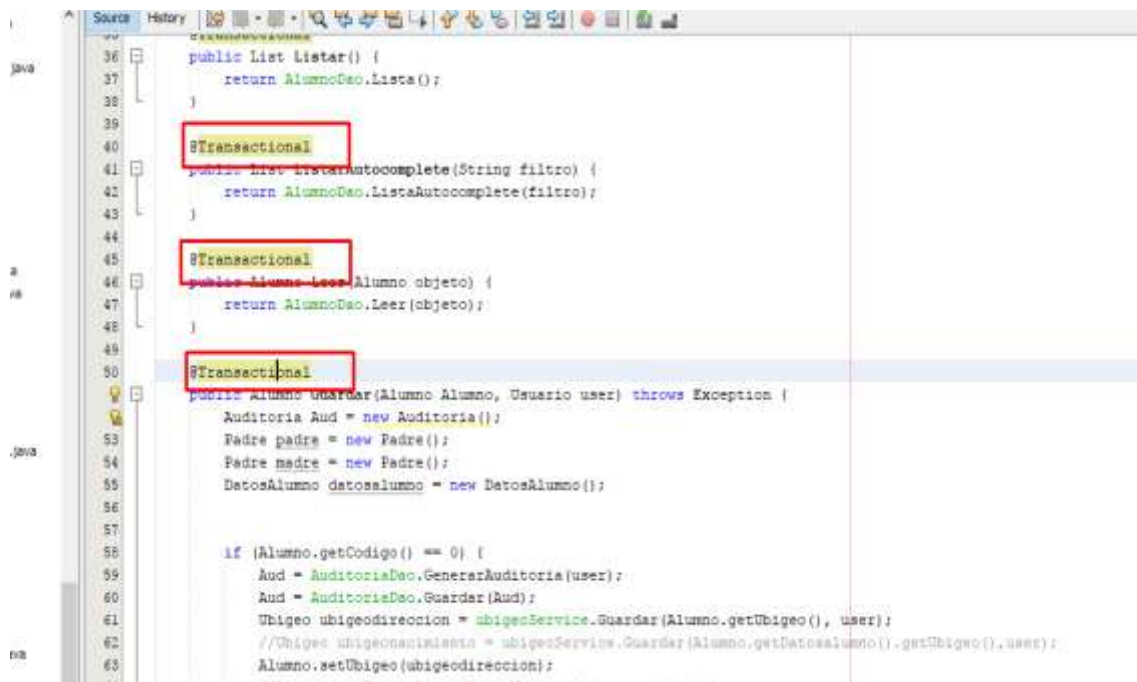
Figura 13. Archivo Controller para el Controlador de la Entidad Alumno  
Fuente: Elaboración Propia

- c. **Dao:** En esta capa se controla todo el acceso a la base de datos, es gestionada por el framework Hibernate utilizando sus anotaciones y su propio lenguaje de consultas (HQL).

```
3  }
5
7
9  public Alumno Guardar(Alumno Alumno) throws Exception {
10      Session session = this.sessionFactory.getCurrentSession();
11      try {
12          session.persist(Alumno);
13      } catch (Exception e) {
14          throw e;
15      } finally {
16          //session.close();
17      }
18      return Alumno;
19  }
```

Figura 14. Archivo para el Acceso a Datos (DAO) de la Entidad Alumno  
Fuente: Elaboración Propia

**d. Service:** Esta capa es una conexión entre la capa de datos y la capa de controlador, en esta capa de servicio se encuentra la Lógica de la programación, además se utilizan las Transacciones de Spring con la anotación `@Transactional`, lo que nos permite manejar todas las consultas e inserción de registros de manera transaccional haciendo un rollback cuando alguna de las operaciones falle.



*Figura 15. Archivo Service (capa de negocio) donde se definen las transacciones*  
*Fuente: Elaboración Propia*

3. Los Beans son declarados en el archivo de spring “spring-servlet.xml”.

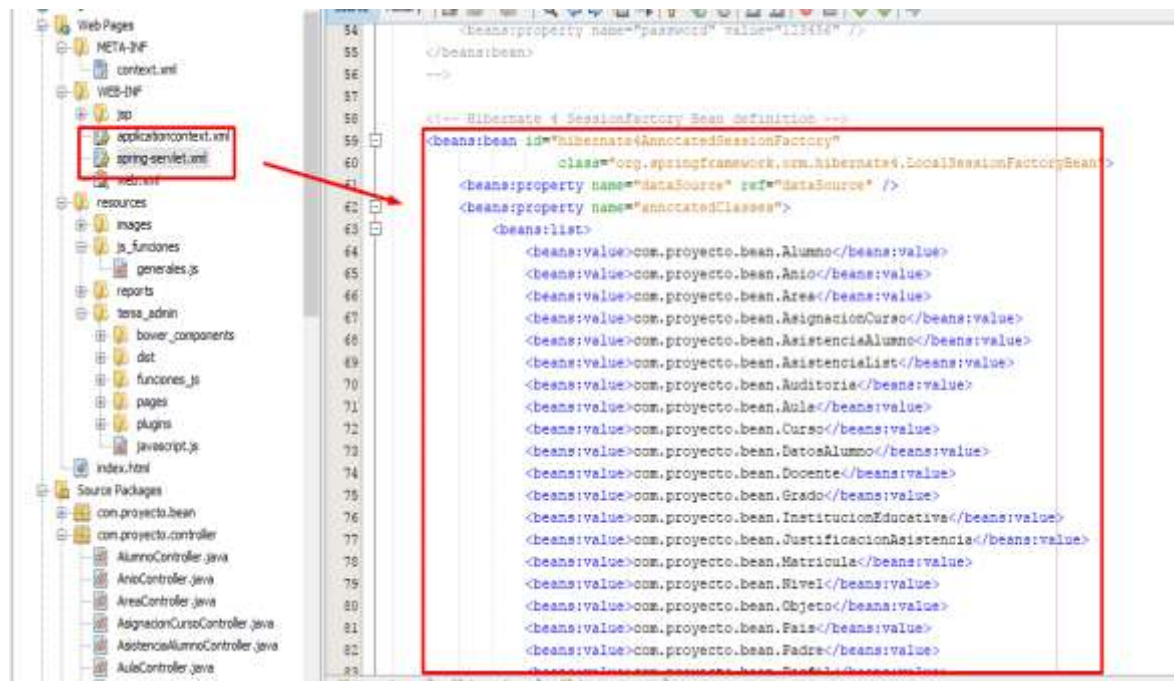


Figura 16. Definición de Beans en el Archivo spring – servlet.xml  
Fuente: Elaboración Propia

4. La conexión a la base de datos debe realizarse en el archivo también en el archivo spring – servlet.xml.



Figura 17. Configuración de Conexión a la Base de Datos (spring – servlet.xml)  
Fuente: Elaboración Propia

5. Luego de realizar las configuraciones podemos ejecutar nuestra aplicación.

### 5.3. PLANIFICACIÓN Y DISEÑO

#### 5.3.1. Iteración 1

Esta iteración presenta 3 historias de usuario y describe las configuraciones y manejo de usuarios y seguridad del sistema.

1. Gestión de Usuarios
2. Gestión de Perfiles
3. Gestión de Opciones de Menú

##### 5.3.1.1. Historias de usuario

Tabla 3  
Historia de Usuario N° 1

Numero 01		Nombre: Gestión de Usuarios	
<b>Usuario:</b> Administrador web			
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)	
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)	
<b>Descripción:</b> El usuario podrá realizar la gestión de los usuarios que intervienen en el sistema, Indicando su nombre y clave, datos que servirán para que puedan acceder al sistema. Se podrá registrar, modificar, listar y eliminar los registros de usuarios del sistema.			
<b>Observaciones:</b> Ninguna.			

Fuente: Elaboración Propia

Tabla 4  
Tarea Diseño de Interfaz Gestión de Usuarios

Tarea Nº 1	
<b>Número de historia:</b> 1	
<b>Nombre:</b> Diseño de interfaz Gestión de Usuarios	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de usuarios en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros	
<i>Fuente:</i> Elaboración Propia	

Tabla 5  
Tarea Mantenimiento de Usuarios

Tarea Nº 2	
<b>Número de historia:</b> 1	
<b>Nombre:</b> Mantenimiento de Usuarios	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de usuarios. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 6  
Historia de Usuario N° 2

Numero 02		Nombre: Gestión de Perfiles
<b>Usuario:</b> Administrador web		
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar la gestión de los perfiles que se manejarán en el sistema, estos datos servirán para definir los roles con los que los usuarios accederán al sistema, con el fin de identificarlos y asignarles los permisos correspondientes. Se podrá registrar, modificar, listar y eliminar los registros de perfiles del sistema.		
<b>Observaciones:</b> Ninguna.		
<i>Fuente:</i> Elaboración Propia		

Tabla 7  
Tarea Diseño de Interfaz Gestión de Perfiles

Tarea N° 1	
<b>Número de historia:</b> 2	
<b>Nombre:</b> Diseño de interfaz Gestión de Perfiles	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de perfiles en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros	
<i>Fuente:</i> Elaboración Propia	

Tabla 8  
Tarea Mantenimiento de Perfiles

Tarea N° 2	
<b>Número de historia:</b> 2	
<b>Nombre:</b> Mantenimiento de Perfiles	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de perfiles. En este mismo formulario se realiza la modificación y eliminación de registros.	
Fuente: Elaboración Propia	

Tabla 9  
Historia de Usuario N° 3

Numero 03	Nombre: Gestión de Opciones de Menú
<b>Usuario:</b> Administrador web	
<b>Modificación de historia número:</b> 0	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)	<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar la gestión de las opciones de menú, esto nos permitirá definir de manera dinámica las opciones del sistema para luego asignarlas al perfil correspondiente, otorgando seguridad de acceso a las opciones. Se podrá registrar, modificar, listar y eliminar los registros de opciones del sistema.	
Observaciones: Ninguna.	
Fuente: Elaboración Propia	



Tabla 10  
Tarea Diseño de Interfaz Gestión de Opciones de Menú

Tarea Nº 1	
<b>Número de historia:</b> 3	
<b>Nombre:</b> Diseño de interfaz Gestión de Opciones de Menú	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de opciones en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros	
<i>Fuente:</i> Elaboración Propia	

Tabla 11  
Tarea Mantenimiento de Opciones de Menú

Tarea Nº 2	
<b>Número de historia:</b> 3	
<b>Nombre:</b> Mantenimiento de Opciones de Menú	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de opciones. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

### 5.3.1.2. Diseño de la iteración 1

#### Gestión de Usuarios

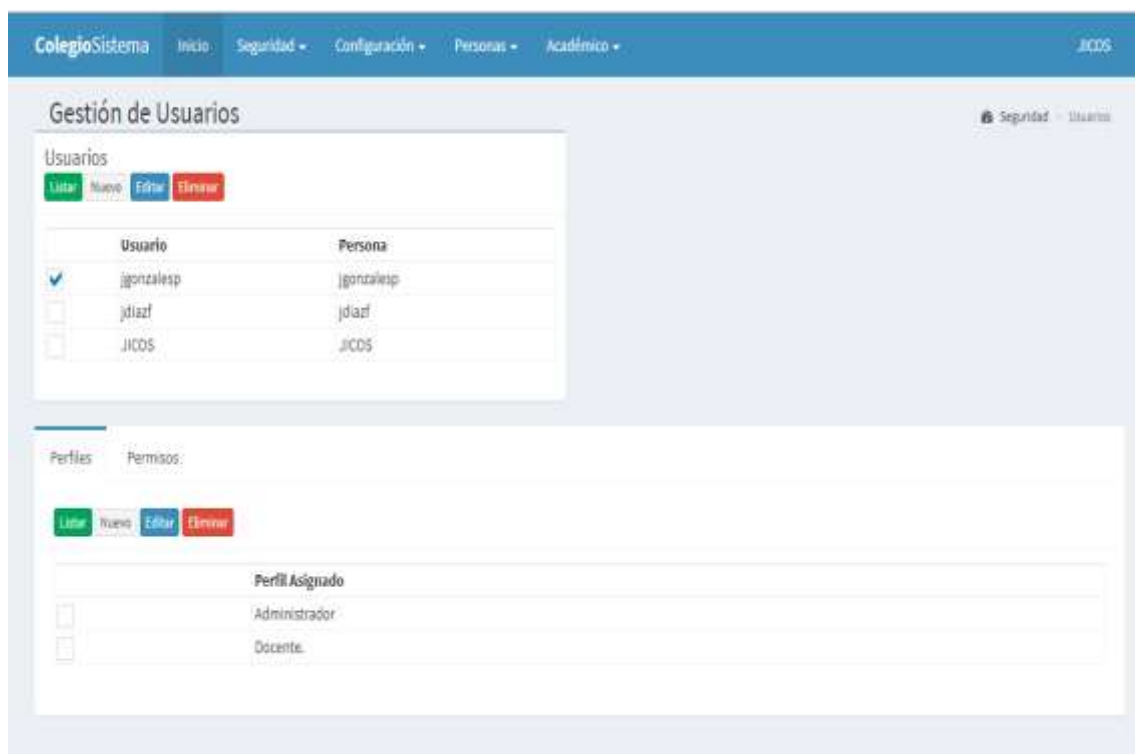


Figura 18. Gestión de Usuarios

Fuente: Elaboración Propia

#### Gestión de Perfiles

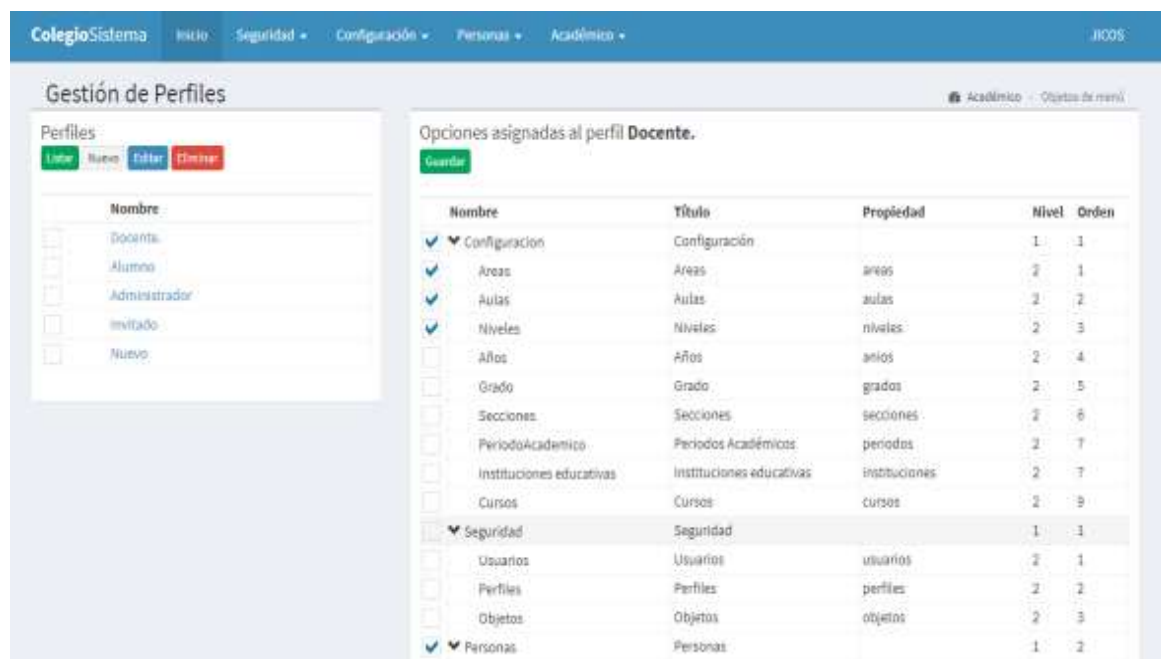


Figura 19. Gestión de Perfiles

Fuente: Elaboración Propia

## Gestión de Opciones de Menú

ColegioSistema

Inicio

Seguridad +

Configuración +

Personas +

Académico +

JICOS

Gestión de Objetos de Menú

Seguridad

Objetos de menú

Lista

Guardar

Eliminar

	Nombre	Título	Propiedad	Nivel	Orden
<input type="checkbox"/>	▼ Configuración	Configuración		1	1
<input type="checkbox"/>	Áreas	Áreas	areas	2	1
<input type="checkbox"/>	Aulas	Aulas	aulas	2	2
<input type="checkbox"/>	Niveles	Niveles	niveles	2	3
<input type="checkbox"/>	Años	Años	años	2	4
<input type="checkbox"/>	Grado	Grado	grados	2	5
<input type="checkbox"/>	Secciones	Secciones	secciones	2	6
<input type="checkbox"/>	PeriodoAcademico	Periodos Académicos	periodos	2	7
<input type="checkbox"/>	Instituciones educativas	Instituciones educativas	instituciones	2	7
<input type="checkbox"/>	Cursos	Cursos	curso	2	9
<input type="checkbox"/>	▼ Seguridad	Seguridad		1	1
<input type="checkbox"/>	Usuarios	Usuarios	usuarios	2	1
<input type="checkbox"/>	Perfiles	Perfiles	perfiles	2	2
<input type="checkbox"/>	Objetos	Objetos	objetos	2	3
<input type="checkbox"/>	► Personas	Personas		1	2
<input type="checkbox"/>	► Académico	Académico		1	4

Figura 20. Gestión de Opciones de Menú

Fuente: Elaboración Propia

### 5.3.1.3. Diagrama de base de datos iteración 1

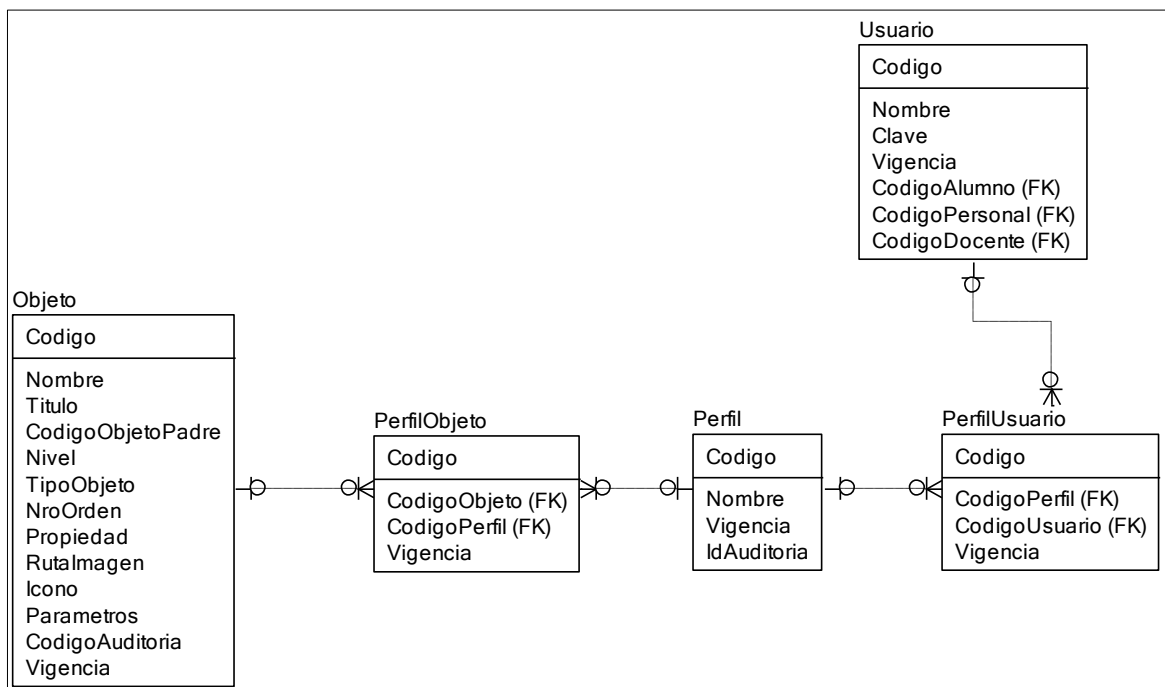


Figura 21. Diagrama de Base de Datos Iteración 1

Fuente: Elaboración Propia

### 5.3.2. Iteración 2

Esta iteración presenta 4 historias de usuario, y equivale a las configuraciones iniciales del sistema con respecto a las personas que harán uso del sistema y las que se registrarán en el mismo.

1. Gestión de Padres
2. Gestión de Alumnos
3. Gestión de Personal
4. Gestión de Docentes

#### 5.3.2.1. Historias de usuario

Tabla 12  
Historia de Usuario N° 4

Numero 04		Nombre: Gestión de Padres	
<b>Usuario:</b> Administrador web			
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)	
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)	
<b>Descripción:</b> El usuario podrá realizar la gestión integral de los padres que intervienen en el sistema, llámese padres no solo a padres biológicos sino también personas que serán apoderados de los alumnos. Se podrá registrar, modificar, listar y eliminar los registros de padres del sistema.			
<b>Observaciones:</b> Ninguna.			

Fuente: Elaboración Propia

Tabla 13  
Tarea Diseño de Interfaz Gestión de Padres

Tarea Nº 1	
<b>Número de historia:</b> 4	
<b>Nombre:</b> Diseño de interfaz Gestión de Padres	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de padres en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros	
<i>Fuente:</i> Elaboración Propia	

Tabla 14  
Tarea Mantenimiento de Padres

Tarea Nº 2	
<b>Número de historia:</b> 4	
<b>Nombre:</b> Mantenimiento de Padres	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de padres. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 15  
Historia de Usuario N° 5

Numero 05		Nombre: Gestión de Alumnos
<b>Usuario:</b> Administrador web		
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar la gestión integral de los alumnos que son la base del sistema ya que a partir de estos registros se desarrolla todo el sistema, sobre todo en la generación de matrículas. Se podrá registrar, modificar, listar y eliminar los registros de alumnos del sistema.		
<b>Observaciones:</b> Ninguna.		
<i>Fuente:</i> Elaboración Propia		

Tabla 16  
Tarea Diseño de Interfaz Gestión de Alumnos

Tarea N° 1	
<b>Número de historia:</b> 5	
<b>Nombre:</b> Diseño de interfaz Gestión de Alumnos	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de alumnos en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros	
<i>Fuente:</i> Elaboración Propia	

Tabla 17  
Tarea Mantenimiento de Alumnos

Tarea N° 2	
<b>Número de historia:</b> 5	
<b>Nombre:</b> Mantenimiento de Alumnos	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de alumnos. En este mismo formulario se realiza la modificación y eliminación de registros.	
Fuente: Elaboración Propia	

Tabla 18  
Historia de Usuario N° 6

Numero 06	Nombre: Gestión de Personal
<b>Usuario:</b> Administrador web	
<b>Modificación de historia número:</b> 0	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)	<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar la gestión integral del personal que labora o que va a hacer uso del sistema, esto relacionado con el usuario que tendrá para controlar su acceso. Se podrá registrar, modificar, listar y eliminar los registros de personal del colegio.	
<b>Observaciones:</b> Ninguna.	
Fuente: Elaboración Propia	

Tabla 19  
Tarea Diseño de Interfaz Gestión de Personal

Tarea N° 1	
<b>Número de historia:</b> 6	
<b>Nombre:</b> Diseño de interfaz Gestión de Personal	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de personal en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros	
<i>Fuente:</i> Elaboración Propia	

Tabla 20  
Tarea Mantenimiento de Personal

Tarea N° 2	
<b>Número de historia:</b> 6	
<b>Nombre:</b> Mantenimiento de Personal	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de personal. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	



Tabla 21  
Historia de Usuario N° 7

Numero 07		Nombre: Gestión de Docentes
<b>Usuario:</b> Administrador web		
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar la gestión integral de los docentes en el sistema, estos registros nos serán útiles para registrar las matrículas, asignar aulas, etc. Se podrá registrar, modificar, listar y eliminar los registros de Docentes en el sistema.		
<b>Observaciones:</b> Ninguna.		
<i>Fuente:</i> Elaboración Propia		

Tabla 22  
Tarea Diseño de Interfaz Gestión de Docentes

Tarea N° 1	
<b>Número de historia:</b> 7	
<b>Nombre:</b> Diseño de interfaz Gestión de Docentes	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de docentes en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros	
<i>Fuente:</i> Elaboración Propia	

Tabla 23  
Tarea Mantenimiento de Docentes

## Tarea Nº 2

**Número de historia:** 7

**Nombre:** Mantenimiento de Docentes

**Tipo de Tarea:** Desarrollo

**Puntos estimados:** 2

**Programador responsable:** Anilu

**Descripción:** Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de docentes. En este mismo formulario se realiza la modificación y eliminación de registros.

Fuente: Elaboración Propia

### 5.3.2.2. Diseño de la iteración 2

#### Gestión de Padres

Figura 22. Gestión de Padres

Fuente: Elaboración Propia

## Gestión de Alumnos

**ColegioSistema** Inicio Configuración Seguridad Personal Académico

### Gestión de Alumnos

Lista: Actualizar Nuevo Editar Eliminar

	Nº Doc.	
<input type="checkbox"/>	44132423	
<input type="checkbox"/>	44253286	
<input type="checkbox"/>	12141512	
<input type="checkbox"/>	43121563	
<input type="checkbox"/>	42521219	

#### REGISTRO DE DATOS DE ALUMNO

Nro Doc.

Nombres

Apellido Paterno

Apellido Materno

Sexo MASCULINO

Dirección

Departamento

Provincia

Distrito

Teléfono

Correo

Inclusivo

Guardar Cancelar

Copyright © 2018 SISTEMA INTEGRADOS DE COLEGIO. Todos los derechos reservados.

Figura 23. Gestión de Alumnos  
Fuente: Elaboración Propia

## Gestión de Personal

**ColegioSistema** Inicio Configuración Seguridad Personal Académico

### Gestión de Personal

Lista: Actualizar Nuevo Editar Eliminar

	Nº Doc.	
<input type="checkbox"/>	41151212	
<input type="checkbox"/>	49464525	
<input type="checkbox"/>	47284452	

#### REGISTRO DE DATOS

Nro Doc.

Nombres

Apellido Paterno

Apellido Materno

Celular

Correo

Fecha Nac.

Guardar Cancelar

Figura 24. Gestión de Personal  
Fuente: Elaboración Propia

## Gestión de Docentes

The screenshot shows a web application interface for 'Gestión de Docentes'. On the left, there's a sidebar with a 'Lista' button and a form for 'N° Doc.' with values '12345678' and '45678901'. The main area displays a 'REGISTRO DE DATOS' form with the following fields: Nro Doc., Nombres, Apellido Paterno, Apellido Materno, Celular, Correo, Especialidad, Grado de Instrucción, Título, and Fecha Nac. At the bottom of the form are 'Guardar' and 'Cancelar' buttons.

Figura 25. Gestión de Docentes

Fuente: Elaboración Propia

### 5.3.2.3. Diagrama de base de datos iteración 2

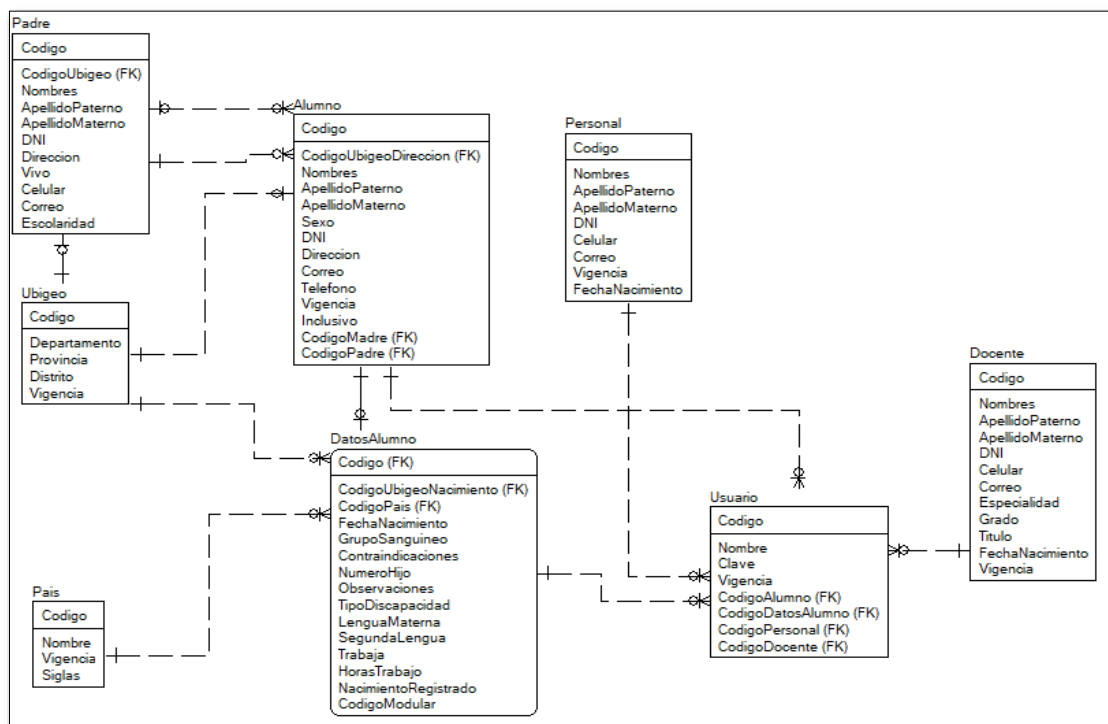


Figura 26. Diagrama de Base de Datos Iteración 2

Fuente: Elaboración Propia

### 5.3.3. Iteración 3

Esta iteración consta de 3 historias de usuarios.

1. Gestión de Años
2. Gestión de Periodos Académicos
3. Gestión de Instituciones Educativas

#### 5.3.3.1. Historias de usuario

Tabla 24  
Historia de Usuario N° 8

Numero 08		Nombre: Gestión de Años	
<b>Usuario:</b> Administrador web			
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)	
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)	
<b>Descripción:</b> El usuario podrá realizar el registro de años académicos que corresponden a los mismos años calendario para establecer luego los periodos (bimestre, trimestre, etc.). Se podrá registrar, modificar, listar y eliminar los registros de años en el sistema.			
<b>Observaciones:</b> Ninguna.			

Fuente: Elaboración Propia

Tabla 25  
Tarea Diseño de Interfaz Gestión de Años

Tarea Nº 1	
<b>Número de historia:</b> 8	
<b>Nombre:</b> Diseño de interfaz Gestión de Años	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de años en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 26  
Tarea Mantenimiento de Años

Tarea Nº 2	
<b>Número de historia:</b> 8	
<b>Nombre:</b> Mantenimiento de Años	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de años académicos. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 27  
Historia de Usuario N° 9

Numero 09		Nombre: Gestión de Periodos académicos
<b>Usuario:</b> Administrador web		
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar el registro de periodos académicos que corresponden a la agrupación de meses tal y como indicará la organización del colegio. Se podrá registrar, modificar, listar y eliminar los registros de periodos en el sistema.		
<b>Observaciones:</b> Ninguna.		
<i>Fuente:</i> Elaboración Propia		

Tabla 28  
Tarea Diseño de Interfaz Gestión de Periodos Académicos

Tarea N° 1	
<b>Número de historia:</b> 9	
<b>Nombre:</b> Diseño de interfaz Gestión de Periodos Académicos	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de periodos académicos en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 29  
Tarea Mantenimiento de Periodos Académicos

Tarea N° 2	
<b>Número de historia:</b> 9	
<b>Nombre:</b> Mantenimiento de Periodos Académicos	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de periodos. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 30  
Historia de Usuario N° 10

Numero 10	Nombre: Gestión de Instituciones educativas
<b>Usuario:</b> Administrador web	
<b>Modificación de historia número:</b> 0	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)	<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar el registro de instituciones educativas externas para luego ser utilizada como referencia e indicar de qué colegio de procedencia llega un alumno. Se podrá registrar, modificar, listar y eliminar los registros de instituciones en el sistema.	
<b>Observaciones:</b> Ninguna.	
<i>Fuente:</i> Elaboración Propia	



Tabla 31  
Tarea Diseño de Interfaz de Instituciones Educativas

Tarea N° 1	
<b>Número de historia:</b> 10	
<b>Nombre:</b> Diseño de interfaz Gestión de Instituciones Educativas	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de instituciones en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 32  
Tarea Mantenimiento de Instituciones Educativas

Tarea N° 2	
<b>Número de historia:</b> 10	
<b>Nombre:</b> Mantenimiento de Instituciones Educativas	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de instituciones. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

### 5.3.3.2. Diseño de iteración 3

#### Gestión de Años

The screenshot shows the 'Gestión de Años Académicos' interface. A modal window titled 'REGISTRO DE DATOS' is open, allowing data entry. The form includes the following fields:

- Año:** A text input field containing the value '0'.
- Fecha Inicio:** An empty date input field.
- Fecha Fin:** An empty date input field.
- Comentario:** A larger text input field for additional notes.

At the bottom of the modal, there are two buttons: 'Guardar' (blue) and 'Cancelar' (red).

Figura 27. Gestión de Años  
Fuente: Elaboración Propia

#### Gestión de Periodos Académicos

The screenshot shows the 'Gestión de Periodos Académicos' interface. A modal window titled 'REGISTRO DE DATOS' is open, allowing data entry. The form includes the following fields:

- Nombre:** A text input field.
- Siglas:** A text input field.
- Fecha Inicio:** A date input field containing '2018-09-29'.
- Fecha Fin:** A date input field containing '2018-09-29'.
- Año:** A dropdown menu with the text 'Seleccione' and a downward arrow.
- Estado:** A text input field.

At the bottom of the modal, there are two buttons: 'Guardar' (blue) and 'Cancelar' (red).

Figura 28. Gestión de Periodos Académicos  
Fuente: Elaboración Propia

## Gestión de Instituciones Educativas

The image shows a web application interface for managing educational institutions. The main header is 'ColegioSistema' with navigation links: Inicio, Configuración, Seguridad, Personas, and Académico. The main section is titled 'Gestión de Instituciones'. Below this, there is a 'Lista' section with buttons: Listar (green), Nuevo (grey), Editar (blue), and Eliminar (red). A table lists institutions with checkboxes and names: Colegio Santo Toribio de Mogro, Colegio Santo Tomás de Aquino, and Colegio Santa María Reina. A modal window titled 'REGISTRO DE INSTITUCIONES' is open, containing two input fields: 'Nombres' and 'Código Modular'. At the bottom of the modal are two buttons: 'Guardar' (blue) and 'Cancelar' (red). The modal also has a close button (X) in the top right corner.

Figura 29. Gestión de Instituciones Educativas

Fuente: Elaboración Propia

### 5.3.3.3. Diagrama de base de datos iteración 3

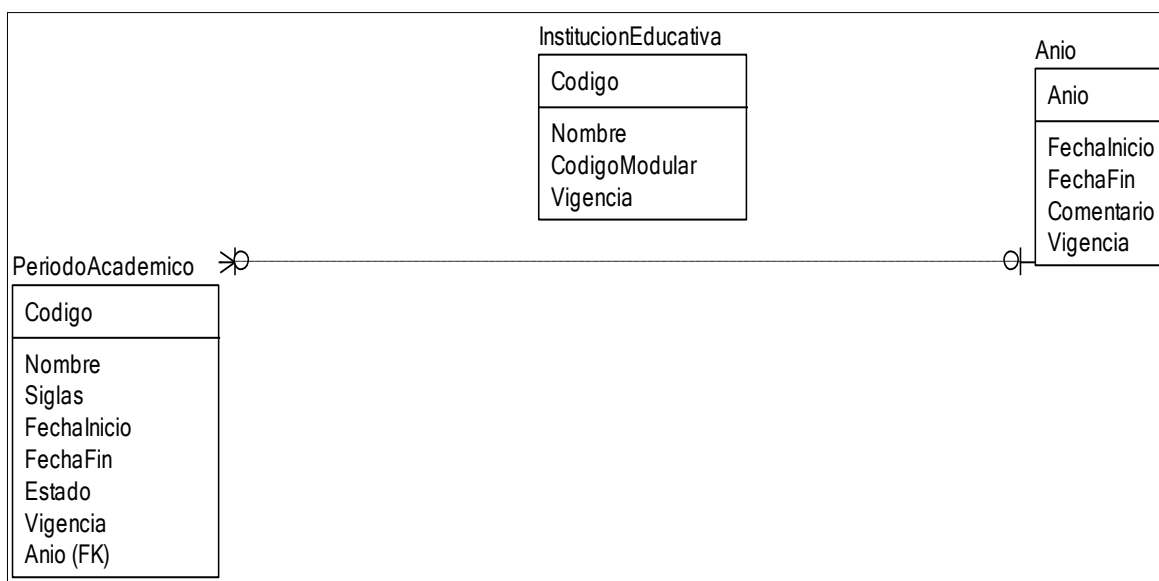


Figura 30. Diagrama de Base de Datos Iteración 3

Fuente: Elaboración Propia

### 5.3.4. Iteración 4

Esta iteración consta de 3 historias de usuarios.

1. Gestión de Niveles
2. Gestión de Áreas
3. Gestión de Grados

### 5.3.4.1. Historias de usuario

Tabla 33  
Historia de Usuario N° 11

Numero 11		Nombre: Gestión de Niveles
<b>Usuario:</b> Administrador web		
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar el registro de niveles del colegio (Inicial, Primaria, Secundaria), estos niveles también están en casi todos los procesos del sistema. Se podrá registrar, modificar, listar y eliminar los registros de niveles en el sistema.		
<b>Observaciones:</b> Ninguna.		

Fuente: Elaboración Propia

Tabla 34  
Tarea Diseño de Interfaz Gestión de Niveles

Tarea N° 1	
<b>Número de historia:</b> 11	
<b>Nombre:</b> Diseño de interfaz Gestión de Niveles	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de niveles en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.	

Fuente: Elaboración Propia

Tabla 35  
Tarea Mantenimiento de Niveles

Tarea N° 2	
<b>Número de historia:</b> 11	
<b>Nombre:</b> Mantenimiento de Niveles	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de niveles. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 36  
Historia de Usuario N° 12

Numero 12	Nombre: Gestión de Áreas
<b>Usuario:</b> Administrador web	
<b>Modificación de historia número:</b> 0	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)	<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar el registro de áreas del colegio (Comunicación, Personal social, etc.), estas áreas son equivalentes a los cursos pero más generales. Se podrá registrar, modificar, listar y eliminar los registros de áreas en el sistema.	
<b>Observaciones:</b> Ninguna.	
<i>Fuente:</i> Elaboración Propia	

Tabla 37  
Tarea Diseño de Interfaz Gestión de Áreas

### Tarea N° 1

**Número de historia:** 12

**Nombre:** Diseño de interfaz Gestión de Áreas

**Tipo de Tarea:** Desarrollo

**Puntos estimados:** 2

**Programador responsable:** Anilu

**Descripción:** Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de Áreas en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.

*Fuente:* Elaboración Propia

Tabla 38  
Tarea Mantenimiento de Áreas

### Tarea N° 2

**Número de historia:** 12

**Nombre:** Mantenimiento de Áreas

**Tipo de Tarea:** Desarrollo

**Puntos estimados:** 2

**Programador responsable:** Anilu

**Descripción:** Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de Áreas. En este mismo formulario se realiza la modificación y eliminación de registros.

*Fuente:* Elaboración Propia

Tabla 39  
Historia de Usuario N° 13

Numero 13		Nombre: Gestión de Grados
<b>Usuario:</b> Administrador web		
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar el registro de grados del colegio (PRIMERO, SEGUNDO, TERCERO, CUARTO, etc.), los grados son datos de configuración básica del sistema. Se podrá registrar, modificar, listar y eliminar los registros de grados en el sistema.		
<b>Observaciones:</b> Ninguna.		
<i>Fuente:</i> Elaboración Propia		

Tabla 40  
Tarea Diseño de Interfaz Gestión de Grados

Tarea N° 1	
<b>Número de historia:</b> 13	
<b>Nombre:</b> Diseño de interfaz Gestión de Grados	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de Grados en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.	
<i>Fuente:</i> Elaboración Propia	



Tabla 41  
Tarea Mantenimiento de Grados

## Tarea Nº 2

**Número de historia:** 13

**Nombre:** Mantenimiento de Grados

**Tipo de Tarea:** Desarrollo

**Puntos estimados:** 2

**Programador responsable:** Anilu

**Descripción:** Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de Grados. En este mismo formulario se realiza la modificación y eliminación de registros.

Fuente: Elaboración Propia

### 5.3.4.2. Diseño de la iteración 4

#### Gestión de Niveles

Figura 31. Gestión de Niveles  
Fuente: Elaboración Propia

## Gestión de Áreas

**ColegioSistema** Inicio Seguridad Configuración Personas Académico

### Gestión de Áreas

Lista

Listar Nuevo Editar Eliminar

	Nombre
<input type="checkbox"/>	Habilidades aritméticas
<input type="checkbox"/>	Habilidades cognitivas

**REGISTRO DE DATOS**

Nombre

Nivel Seleccione

Guardar Cancelar

Figura 32. Gestión de Áreas  
Fuente: Elaboración Propia

## Gestión de Grados

**ColegioSistema** Inicio Seguridad Configuración Personas Académico

### Gestión de Grados

Filtros

Seleccione Nivel INICIAL

Listar Nuevo Editar Eliminar

	Nombre
<input type="checkbox"/>	3 años
<input type="checkbox"/>	5 años
<input type="checkbox"/>	4 años

**REGISTRO DE DATOS**

Nombre SEGUNDO

Número

Tipo Calificación Cualitativa

Nivel INICIAL

Guardar Cancelar

Figura 33. Gestión de Grados  
Fuente: Elaboración Propia

#### 5.3.4.3. Diagrama de base de datos iteración 4

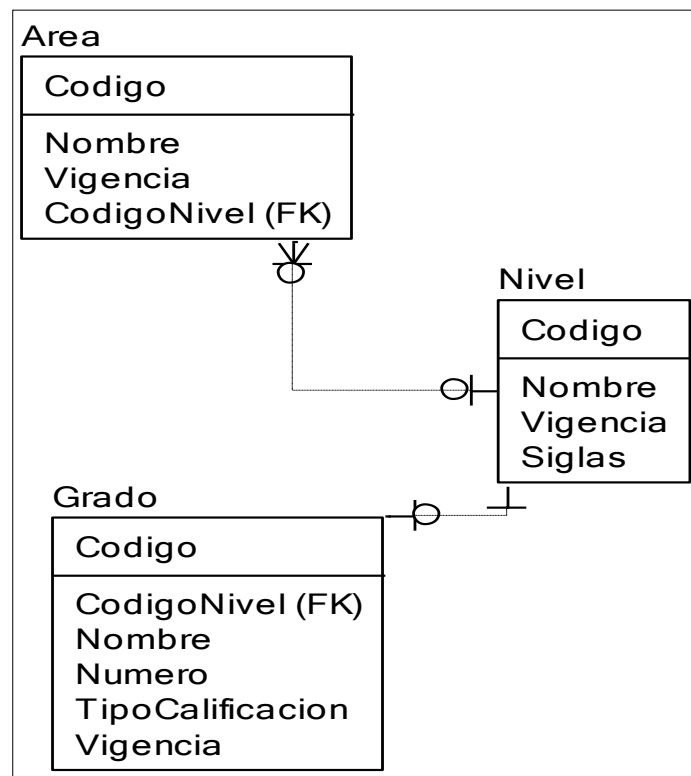


Figura 34. Diagrama de Base de Datos Iteración 4  
Fuente: Elaboración Propia

#### 5.3.5. Iteración 5

Esta iteración consta de 4 historias de usuarios.

1. Gestión de Cursos
2. Gestión de Aulas
3. Gestión de Secciones
4. Asignaciones de Cursos

### 5.3.5.1. Historias de usuario

Tabla 42

Historia de Usuario N° 14

Numero 14		Nombre: Gestión de Cursos
<b>Usuario:</b> Administrador web		
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar el registro de cursos en el sistema, estos datos servirán para la asignación de cursos a docentes. Se podrá registrar, modificar, listar y eliminar los registros de cursos en el sistema.		
<b>Observaciones:</b> Ninguna.		

Fuente: Elaboración Propia

Tabla 43

Tarea Diseño de Interfaz Gestión de Cursos

Tarea N° 1	
<b>Número de historia:</b> 14	
<b>Nombre:</b> Diseño de interfaz Gestión de Cursos	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de Cursos en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.	

Fuente: Elaboración Propia

Tabla 44  
Tarea Mantenimiento de Cursos

Tarea N° 2	
<b>Número de historia:</b> 14	
<b>Nombre:</b> Mantenimiento de Cursos	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de Cursos. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 45  
Historia de Usuario N° 15

Numero 15	Nombre: Gestión de Aulas
<b>Usuario:</b> Administrador web	
<b>Modificación de historia número:</b> 0	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)	<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar el registro de aulas con las que cuenta el colegio, según información de los propios encargados. Se podrá registrar, modificar, listar y eliminar los registros de aulas en el sistema.	
<b>Observaciones:</b> Ninguna.	
<i>Fuente:</i> Elaboración Propia	

Tabla 46  
Tarea Diseño de Interfaz Gestión de Aulas

Tarea Nº 1	
<b>Número de historia:</b> 15	
<b>Nombre:</b> Diseño de interfaz Gestión de Aulas	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de Aulas en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 47  
Tarea Mantenimiento de Aulas

Tarea Nº 2	
<b>Número de historia:</b> 15	
<b>Nombre:</b> Mantenimiento de Aulas	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de Aulas. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 48

Numero 16		Nombre: Gestión de Secciones
<b>Usuario:</b> Administrador web		
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar el registro de secciones, conformadas por el nivel, grado y letra con las que cuenta el colegio, según información de los propios encargados. Se podrá registrar, modificar, listar y eliminar los registros de secciones en el sistema.		
<b>Observaciones:</b> Ninguna.		

Historia de Usuario N° 16

Fuente: Elaboración Propia

Tabla 49

Tarea Diseño de Interfaz Gestión de Secciones

Tarea N° 1	
<b>Número de historia:</b> 16	
<b>Nombre:</b> Diseño de interfaz Gestión de Secciones	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de Secciones en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.	

Fuente: Elaboración Propia

Tabla 50  
Tarea Mantenimiento de Secciones

Tarea N° 2	
<b>Número de historia:</b> 16	
<b>Nombre:</b> Mantenimiento de Secciones	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de Secciones. En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 51  
Historia de Usuario N° 17

Numero 17	Nombre: Asignación de Cursos
<b>Usuario:</b> Administrador web	
<b>Modificación de historia número:</b> 0	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)	<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá realizar la asignación de cursos a los determinados docentes que van a dictar en el colegio, los cursos corresponden a cada periodo y año académico y puede cambiar una asignación con respecto a otro periodo. Se podrá registrar, modificar, listar y eliminar las asignaciones de cursos en el sistema.	
<b>Observaciones:</b> Ninguna.	
<i>Fuente:</i> Elaboración Propia	



Tabla 52  
Tarea Diseño de Interfaz Asignación de Cursos

### Tarea Nº 1

**Número de historia:** 17

**Nombre:** Diseño de interfaz Asignación de Cursos

**Tipo de Tarea:** Desarrollo

**Puntos estimados:** 2

**Programador responsable:** Anilu

**Descripción:** Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de Asignación de Cursos en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.

*Fuente:* Elaboración Propia

Tabla 53  
Tarea Mantenimiento de Asignación de Cursos

### Tarea Nº 2

**Número de historia:** 17

**Nombre:** Mantenimiento de Asignación Cursos

**Tipo de Tarea:** Desarrollo

**Puntos estimados:** 2

**Programador responsable:** Anilu

**Descripción:** Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de Asignación de Cursos. En este mismo formulario se realiza la modificación y eliminación de registros.

*Fuente:* Elaboración Propia

### 5.3.5.2. Diseño de la iteración 5

#### Gestión de Cursos

The screenshot displays the 'Gestión de Cursos' (Course Management) interface. A modal window titled 'REGISTRO DE CURSO' is open, allowing for the creation or editing of a course. The modal contains the following fields:

- Nombre:** A text input field.
- Horas:** A numeric input field with a value of 0 and a blue '+' button.
- Grado:** A dropdown menu with 'INICIAL - 3 años' selected.
- Area:** A dropdown menu with 'INICIAL - Habilidades aritméticas' selected.

At the bottom of the modal are two buttons: 'Guardar' (Save) in blue and 'Cancelar' (Cancel) in red. In the background, the 'Gestión de Cursos' page is visible, showing a list of courses with checkboxes and a table with columns for 'Nombre' and 'Número'.

Figura 35. Gestión de Cursos

Fuente: Elaboración Propia

#### Gestión de Aulas

The screenshot displays the 'Gestión de Aulas' (Classroom Management) interface. A modal window titled 'REGISTRO DE DATOS' is open, allowing for the creation or editing of a classroom. The modal contains the following fields:

- Número:** A numeric input field with a value of 0.
- Capacidad:** A numeric input field with a value of 0.
- Tipo:** A text input field.

At the bottom of the modal are two buttons: 'Guardar' (Save) in blue and 'Cancelar' (Cancel) in red. In the background, the 'Gestión de Aulas' page is visible, showing a list of classrooms with checkboxes and a table with columns for 'Número' and 'Tipo'.

Figura 36. Gestión de Aulas

Fuente: Elaboración Propia

## Gestión de Secciones

**REGISTRO DE DATOS**

Letra	A
Año	2017
Nivel	PRIMARIA
Grado	SEGUNDO
Tutor	PRADO VILCHEZ ALEX
Aula	3
Tipo Calificación	Númerica

**Guardar** **Cancelar**

Figura 37. Gestión de Secciones  
Fuente: Elaboración Propia

## Asignaciones de Cursos

**REGISTRO DE CURSO**

Docente	PRADO VILCHEZ ALEX
Curso	Literatura
Periodo Académico	TERCER TRIMESTRE
Personal	GONZALES PEREZ JOSE

**Guardar** **Cancelar**

Figura 38. Asignaciones de Cursos  
Fuente: Elaboración Propia

### 5.3.5.3. Diagrama de base de datos iteración 5

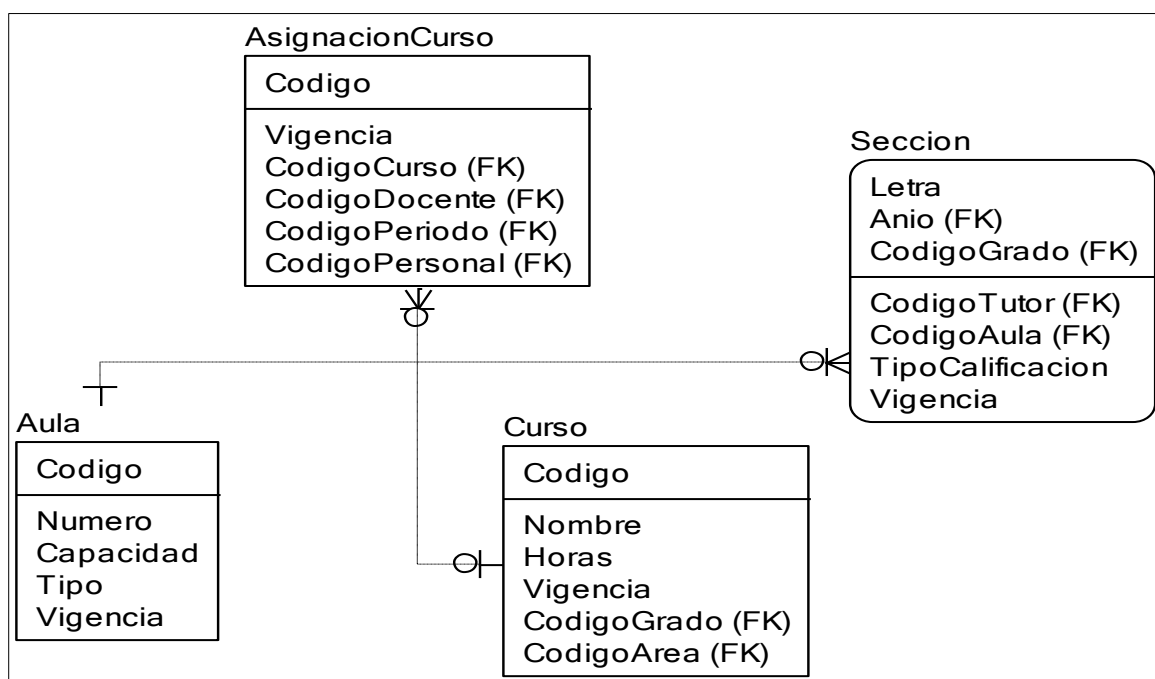


Figura 39. Diagrama de Base de Datos Iteración 5

Fuente: Elaboración Propia

### 5.3.6. Iteración 6

Esta iteración consta de 3 historias de usuario.

1. Gestión de Matrícula
2. Marcar Asistencia
3. Notificaciones de falta (Justificación)

### 5.3.6.1. Historias de usuario

Tabla 54

Historia de Usuario N° 18

Numero 18		Nombre: Gestión de Matrícula
<b>Usuario:</b> Administrador web		
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario hará uso de esta pantalla para registrar y gestionar las matrículas que son la base del sistema, gracias a este proceso se matricula al alumnado por lo tanto se asigna a sus diferentes secciones con su docente asignado. Se podrá registrar, modificar, listar y eliminar las matrículas registradas por el personal del colegio.		
<b>Observaciones:</b> Ninguna.		

Fuente: Elaboración Propia

Tabla 55

Tarea Diseño de Interfaz Gestión de Matrícula

Tarea N° 1	
<b>Número de historia:</b> 18	
<b>Nombre:</b> Diseño de interfaz Matricula	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una interfaz de usuario para gestionar el registro y mantenimiento de Matricula en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.	

Fuente: Elaboración Propia

Tabla 56  
Tarea Mantenimiento de Matrícula

Tarea N° 2	
<b>Número de historia:</b> 18	
<b>Nombre:</b> Mantenimiento de Matrícula	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de matrícula (con los datos del alumno a matricular). En este mismo formulario se realiza la modificación y eliminación de registros.	
<i>Fuente:</i> Elaboración Propia	

Tabla 57  
Historia de Usuario N° 19

Numero 19	Nombre: Marcar asistencia
<b>Usuario:</b> Administrador web	
<b>Modificación de historia número:</b> 0	<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)	<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)	<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá generar y marcar las asistencias de todos los alumnos. Se podrá registrar, modificar, listar y eliminar las asistencias en el sistema.	
<b>Observaciones:</b> Ninguna.	
<i>Fuente:</i> Elaboración Propia	

Tabla 58  
Tarea Diseño de Interfaz Marcar Asistencia

### Tarea N° 1

**Número de historia:** 19

**Nombre:** Diseño de interfaz Marcar Asistencia

**Tipo de Tarea:** Desarrollo

**Puntos estimados:** 2

**Programador responsable:** Anilu

**Descripción:** Se diseñará una interfaz de usuario para gestionar el marcar asistencia en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.

*Fuente:* Elaboración Propia

Tabla 59  
Tarea Mantenimiento de Marcar Asistencia

### Tarea N° 2

**Número de historia:** 19

**Nombre:** Mantenimiento de Marcar Asistencia

**Tipo de Tarea:** Desarrollo

**Puntos estimados:** 2

**Programador responsable:** Anilu

**Descripción:** Al pulsar el botón “Nuevo” se ejecuta una ventana modal en la que se deben ingresar los datos básicos del registro de marcaciones de asistencia. En este mismo formulario se realiza la modificación y eliminación de registros.

*Fuente:* Elaboración Propia

Tabla 60  
Historia de Usuario N° 20

Numero 20		Nombre: Notificaciones de Falta	
<b>Usuario:</b> Administrador web			
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1	
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)	
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)	
<b>Descripción:</b> El usuario podrá generar notificaciones de falta de los alumnos para poder avisar vía correo electrónico que su menor hijo no asistió al colegio. Se podrá registrar, modificar, listar y eliminar las asistencias en el sistema.			
<b>Observaciones:</b> Ninguna.			
<i>Fuente:</i> Elaboración Propia			

Tabla 61  
Tarea Desarrollo de Funcionalidad Enviar Notificación

Tarea N° 1	
<b>Número de historia:</b> 20	
<b>Nombre:</b> Desarrollo de funcionalidad Enviar Notificación	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Se diseñará una funcionalidad para gestionar el registro de notificaciones del sistema a los correos de los apoderados de los alumnos indicados anteriormente en la matricula.	
<i>Fuente:</i> Elaboración Propia	



### 5.3.6.2. Diseño de la iteración 6

#### Gestión de Matrícula

The image shows a web application interface for 'ColegioSistema'. A modal window titled 'REGISTRO MATRÍCULA' is open, displaying a form for student registration. The form includes the following fields and options:

- Alumno:** A text input field with a blue 'Nuevo +' button to its right.
- Sección:** A dropdown menu with the selected value 'PRIMARIA - PRIMERO - "A"'. A small downward arrow is visible on the right.
- Colegio de Procedencia:** A dropdown menu with the selected value 'Colegio Santo Toribio de Mogrovejo'. A small downward arrow is visible on the right.
- Apoderado:** A text input field with a blue 'Nuevo +' button to its right.
- Parentesco:** A dropdown menu with the selected value 'Seleccione'. A small downward arrow is visible on the right.
- Fecha:** A date input field showing '2018-09-29'.
- Estado:** A dropdown menu with the selected value 'REGISTRADA'. A small downward arrow is visible on the right.
- Condición Final:** A dropdown menu with the selected value 'APROBADO'. A small downward arrow is visible on the right.
- Docente:** A dropdown menu with the selected value 'Seleccione'. A small downward arrow is visible on the right.
- Personal:** A dropdown menu with the selected value 'Seleccione'. A small downward arrow is visible on the right.

At the bottom of the modal, there are two buttons: a blue 'Guardar' button on the left and a red 'Cancelar' button on the right.

Figura 40. Gestión de Matrícula  
Fuente: Elaboración Propia

## Marcar Asistencia

**ColegioSistema** Inicio Configuración Seguridad Personas Académico

### Marcar Asistencia

Año: 2018 Grado: PRIMERO Fecha: 2018-09-29  
Nivel: PRIMARIA Letra: A

Listar Guardar

N°	Nombre	Acción
1	VARGAS COLLIQUE VICTORIA	<input checked="" type="checkbox"/> Enviar Notificación
2	VILLEGAS SANCHEZ MARCELO	<input type="checkbox"/> Enviar Notificación
3	PEREZ CRUZ MARIA	<input checked="" type="checkbox"/> Enviar Notificación

Figura 41. Marcar Asistencia  
Fuente: Elaboración Propia

## Notificaciones de Falta (Justificación)

**ColegioSistema** Inicio Configuración Seguridad Personas Académico

### Marcar Asistencia

Año: 2018 Grado: PRIMERO Fecha: 2018  
Nivel: PRIMARIA Letra: A

Listar Guardar

N°	Nombre	Acción
1	VARGAS COLLIQUE VICTORIA	<input checked="" type="checkbox"/> Enviar Notificación
2	VILLEGAS SANCHEZ MARCELO	<input checked="" type="checkbox"/> Enviar Notificación
3	PEREZ CRUZ MARIA	<input checked="" type="checkbox"/> Enviar Notificación

Figura 42. Notificaciones de Falta (Justificación)  
Fuente: Elaboración Propia

### 5.3.6.3. Diagrama de base de datos iteración 6

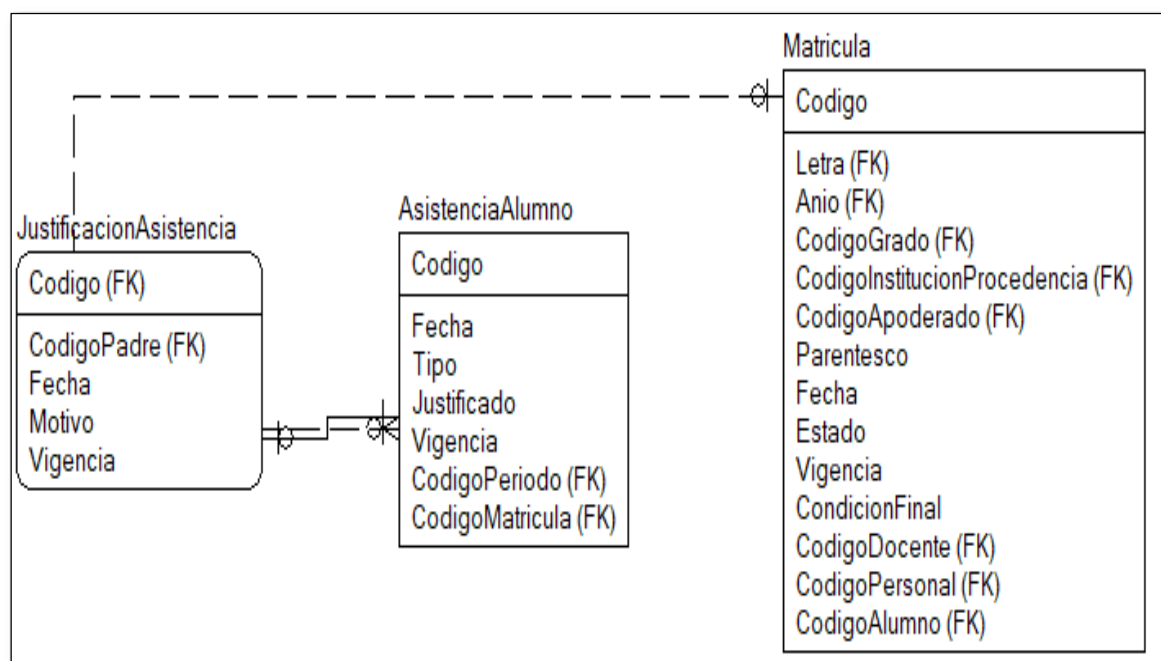


Figura 43. Diagrama de Base de Datos Iteración 6  
Fuente: Elaboración Propia

### 5.3.7. Iteración 7

Esta iteración consta de 2 Historias de Usuario:

1. Registro de Promedios por Año (Curso y Área)
2. Registro de Promedios por Periodo (Curso y Área)

### 5.3.7.2. Historias de usuario

Tabla 62

Historia de Usuario N° 21

Numero 21		Nombre: Registro de Promedios Por Año (Curso y Área)
<b>Usuario:</b> Administrador web		
<b>Modificación de historia número:</b> 0		<b>Iteración asignada:</b> 1
<b>Prioridad en negocio:</b> Alta (Alta / Media / Baja)		<b>Puntos estimados:</b> 8 (1 – 10)
<b>Riesgo en desarrollo:</b> Media (Alta / Media / Baja)		<b>Puntos reales:</b> 5 (1 – 10)
<b>Descripción:</b> El usuario podrá registrar los promedios de notas de los alumnos por año tanto en cursos como en áreas a través de un formulario de ingreso de notas. Se podrá registrar, modificar, listar y eliminar las notas de alumnos		
<b>Observaciones:</b> Ninguna.		

Fuente: Elaboración Propia

Tabla 63

Tarea Diseño de Interfaz Registro de Notas por Año

Tarea Nº 1	
Número de historia: 21	
Nombre: Diseño de interfaz Registro de Notas por Año	
Tipo de Tarea: Desarrollo	Puntos estimados: 2
Programador responsable: Anilu	
Descripción: Se diseñará una interfaz de usuario para gestionar el registro de promedios por año en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.	

Fuente: Elaboración Propia

Tabla 64  
Tarea Mantenimiento de Registro de Promedios por Año

Tarea N° 2	
<b>Número de historia:</b> 21	
<b>Nombre:</b> Mantenimiento de Registro de Promedios por Año	
<b>Tipo de Tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Programador responsable:</b> Anilu	
<b>Descripción:</b> Al pulsar el botón “Listar” aparece la lista de alumnos que están en el curso seleccionado. En este mismo formulario se realiza la modificación y eliminación de registros.	
Fuente: Elaboración Propia	

Tabla 65  
Historia de Usuario N° 22

Numero 22		Nombre: Registro de Promedios Por Periodo (Curso y Área)	
Usuario: Administrador web			
Modificación de historia número: 0		Iteración asignada: 1	
Prioridad en negocio: Alta (Alta / Media / Baja)		Puntos estimados: 8 (1 – 10)	
Riesgo en desarrollo: Media (Alta / Media / Baja)		Puntos reales: 5 (1 – 10)	
Descripción: El usuario podrá registrar los promedios de notas de los alumnos por periodo tanto en cursos como en áreas a través de un formulario de ingreso de notas. Se podrá registrar, modificar, listar y eliminar las notas de alumnos			
Observaciones: Ninguna.			
Fuente: Elaboración Propia			

Tabla 66

Tarea Diseño de Interfaz Registro de Promedios por Periodo

---

**Tarea Nº 1**

---

**Número de historia:** 22

**Nombre:** Diseño de interfaz Registro de Promedios por Periodo

**Tipo de Tarea:** Desarrollo

**Puntos estimados:** 2

**Programador responsable:** Anilu

**Descripción:** Se diseñará una interfaz de usuario para gestionar el registro de promedios por periodo en el sistema, contando con botones básicos para registro, modificación y eliminación de registros así como ventanas modales para realizar dichos registros.

---

*Fuente:* Elaboración Propia

Tabla 67

Tarea Mantenimiento de Registro de Promedios por Periodo

---

**Tarea Nº 22**

---

**Número de historia:** 18

**Nombre:** Mantenimiento de Registro de Promedios por Periodo

**Tipo de Tarea:** Desarrollo

**Puntos estimados:** 2

**Programador responsable:** Anilu

**Descripción:** Al pulsar el botón “Listar” aparece la lista de alumnos que están en el curso seleccionado. En este mismo formulario se realiza la modificación y eliminación de registros.

---

*Fuente:* Elaboración Propia

### 5.3.7.2. Diseño de la iteración 7

Registro de Promedios por Año y Periodo (Curso y Área)

**ColegioSistema** Inicio Configuración Seguridad Personas Académico JICOS

#### Promedios de Cursos por Peiodo Académico

Académico > Promedios

Año: 2018 Período: PRIMER TRIMESTRE Fecha: 2018-09-29  
Nivel: PRIMARIA Grado: PRIMERO Letra: A  
Curso: Literatura Docente: PRADO VILCHEZ ALEX

Listar Guardar

Nº	Nombre	Promedio
1	PEREZ CRUZ MARIA	0
2	VARGAS COLLIQUE VICTORIA	0
3	VILLEGAS SANCHEZ MARCELO	0

Figura 44. Registro de Promedios por Año y Periodo (Curso y Área)  
Fuente: Elaboración Propia

### 5.3.7.3. Diagrama de base de datos iteración 7

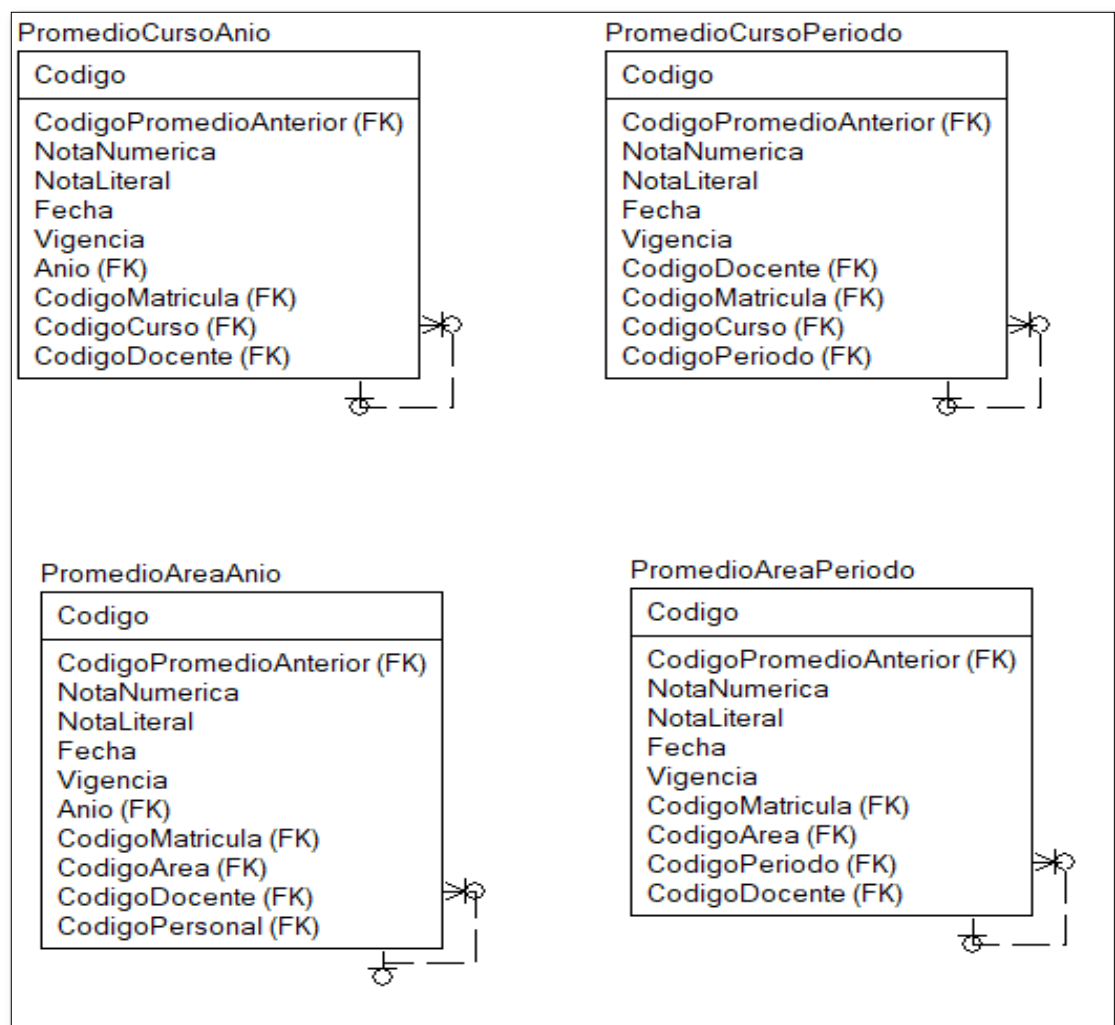


Figura 45. Diagrama de Base de Datos Iteración 7

Fuente: Elaboración Propia



5.4. DIAGRAMA DE COMPONENTES

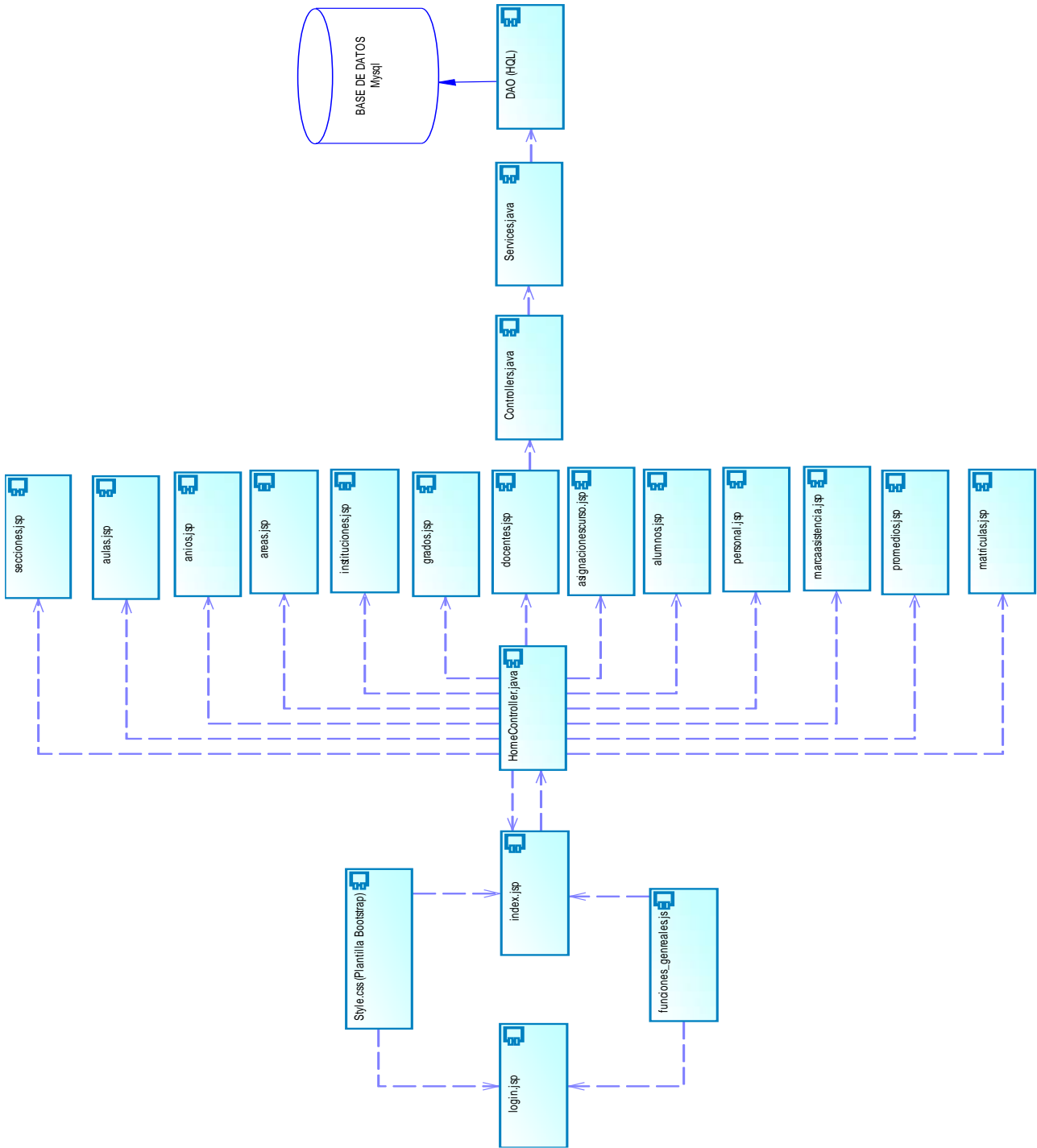


Figura 46. Diagrama de Componentes  
Fuente: Elaboración Propia

## 5.5. DIAGRAMA DE DESPLIEGUE

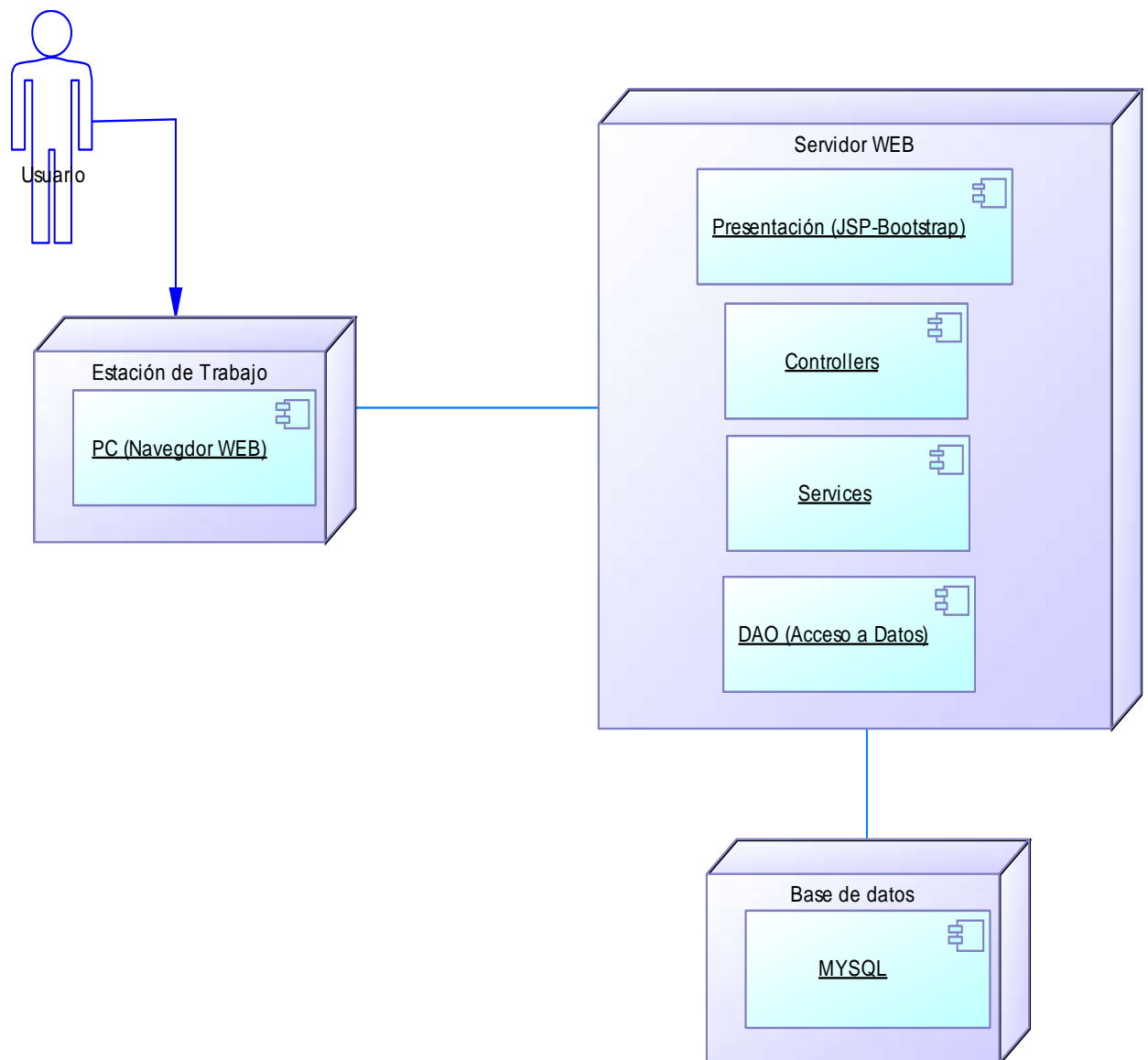


Figura 47. Diagrama de Despliegue  
Fuente: Elaboración Propia



## 5.7. IMPLEMENTACIÓN DE UNA FUNCIONALIDAD

### 5.7.1. Guardar matrícula

Tabla de Base de Datos

#	Nombre	Tipo de datos
1	<b>Codigo</b>	<b>INT</b>
2	CodigoAlumno	INT
3	CodigoSeccion	INT
4	CodigoInstitucionProcedencia	INT
5	CodigoApoderado	INT
6	Parentesco	VARCHAR
7	Fecha	DATE
8	Estado	VARCHAR
9	CondicionFinal	VARCHAR
10	CodigoDocente	INT
11	CodigoPersonal	INT
12	IdAuditoria	INT
13	Vigencia	BIT

Figura 49. Campos de Tabla Matrícula

Fuente: Elaboración Propia

Figura 50. Interfaz Registro Matrícula

Fuente: Elaboración Propia

## Captura de objetos desde el formulario en formato JSON

```
function ObtenerItem(tipo) {
    var activo = true;
    if($("#evento").val()=="D"){
        activo = false;
    }
    var json = {
        "codigo": $("#txtCodigo").val(),
        "alumno": {
            "codigo": $("#txtCodigoAlumno").val()
        },
        "seccion": {
            "codigo": $("#cmbSeccion").val()
        },
        "institucionprocedencia": {
            "codigo": $("#cmbInstitucion").val()
        },
        "apoderado": {
            "codigo": $("#txtCodigoApoderado").val()
        },
        "parentesco": $("#cmbParentesco").val(),
        "fecha": $("#txtFecha").val(),
        "estado": $("#cmbEstado").val(),
        "condicionfinal": $("#cmbCondicionFinal").val(),
        "docente": {
            "codigo": $("#cmbDocente").val()
        }
    },
}
```

Figura 51. Código Javascript Obtener Item

Fuente: Elaboración Propia

Esta función retorna un objeto el cual utilizaremos para enviarlo al controlador a través de la ruta

```
$("#frmRegistro").submit(function (e) {
    e.preventDefault();
    var url = "guardarMatricula";
    $.ajax({
        type: "POST",
        async: false,
        data: JSON.stringify(ObtenerItem($("#evento").val())),
        contentType: "application/json",
        dataType: "json",
        url: url,
        success: function (a)
        {
            if (a.codigo != 0) {
                $('#modal-registro-basico').modal('hide');
                $('#modal-ok').modal('show');
                Limpiar();
                Listar();
            } else {
                $('#modal-error').modal('show');
            }
        },
        error: function (jqXHR, textStatus, message) {
            var response = $.parseHTML(jqXHR.responseText);
            console.log(response);
            $('#msgRegistro').html(response.filter('h1').text());
            $('#modal-error').modal('show');
        }
    });
});
```

Figura 52. Código Javascript Guardar Matrícula

Fuente: Elaboración Propia

Gracias a esa ruta en el código fuente el Controller decide a que método acceder.



Figura 53. Controller Matrícula

Fuente: Elaboración Propia

En la imagen anterior se visualiza que se debe acceder a la capa de Servicio

“sMatricula”. Se inicializa esta variable para acceder a los métodos de la clase.

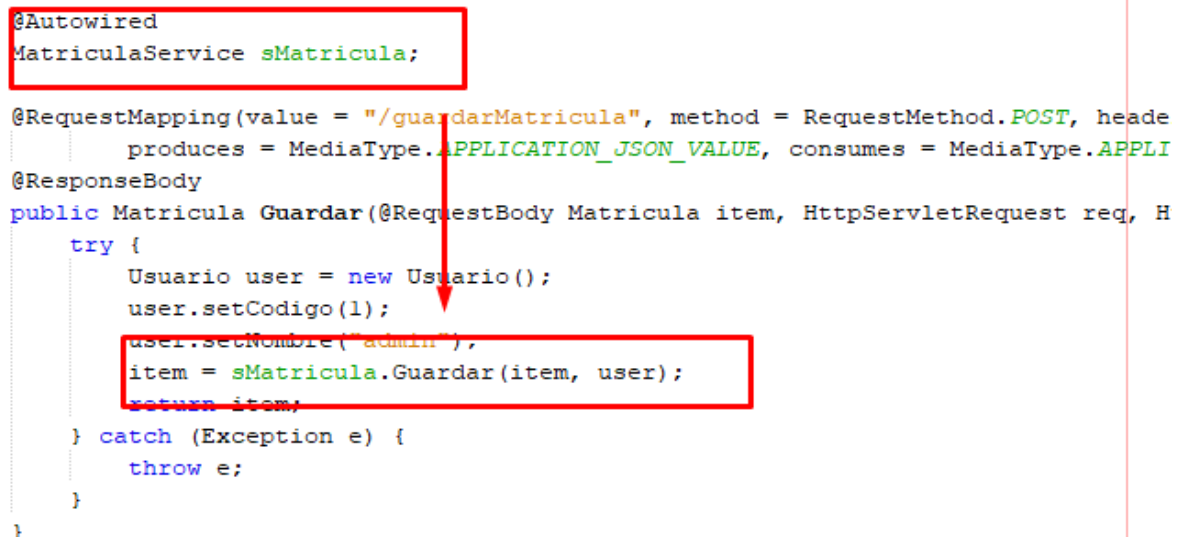


Figura 54. Controller Matrícula (Método Guardar)

Fuente: Elaboración Propia

Cuando el objeto va a la capa de Servicio es donde se realiza la transacción.

```
@Transactional
public Matricula Guardar(Matricula Matricula, Usuario user) throws Exception {
    Auditoria Aud = new Auditoria();
    Padre padre = new Padre();
    Padre madre = new Padre();

    if (Matricula.getCodigo() == 0) {
        Aud = dAuditoria.GenerarAuditoria(user);
        Aud = dAuditoria.Guardar(Aud);
        Matricula.setAuditoria(Aud);
        Matricula = dMatricula.Guardar(Matricula);
    } else {
        Aud = dAuditoria.Leer(Matricula.getAuditoria().getId());
        Aud.setActivo(Matricula.getVigencia());
        Aud = dAuditoria.ActualizarAuditoria(user, Aud);
        Matricula = dMatricula.Actualizar(Matricula);
    }

    return Matricula;
}
```

Figura 55. Método Guardar Capa de Servicio

Fuente: Elaboración Propia

Aquí se define a través del comando @Transactional una transacción gracias a SPRING FRAMEWORK. En esta capa se utilizan los métodos de la capa DAO (Datos), por lo que tenemos que inicializar las variables por medio del comando @Autowired.

```
@Autowired
MatriculaDAO dMatricula;
```

Figura 56. Comando Autowired

Fuente: Elaboración Propia

```
Aud = dAuditoria.GenerarAuditoria(user);
Aud = dAuditoria.Guardar(Aud);
Matricula.setAuditoria(Aud);
Matricula = dMatricula.Guardar(Matricula);
} else {
    Aud = dAuditoria.Leer(Matricula.getAuditoria().getId());
    Aud.setActivo(Matricula.getVigencia());
```

Figura 57. Método Guardar Capa Servicio

Fuente: Elaboración Propia

El método guardar se define en la capa DAO, que es el guardado ya en la base de datos gracias a Hibernate.

```
83     }
84     return lista;
85 }
86
87 public Matricula Guardar(Matricula Matricula) throws Exception {
88     Session session = this.sessionFactory.getCurrentSession();
89     try {
90         session.persist(Matricula);
91     } catch (Exception e) {
92         throw e;
93     } finally {
94         //session.close();
95     }
96     return Matricula;
97 }
98 }
```

*Figura 58. Método Guardar Capa DAO (Acceso a Datos)*  
*Fuente: Elaboración Propia*



# MANUAL DE USUARIO



**“IMPLEMENTACIÓN DE UN SISTEMA  
INFORMÁTICO DIRIGIDO AL PROCESO  
ACADÉMICO CON SOPORTE BASADO EN  
NOTIFICACIONES PARA EL COLEGIO MIGUEL  
ANGEL BUONARROTTI”**

## **1. Guía de usuario**

Con el propósito de tener un mayor entendimiento y facilidad de manejo del sistema, a continuación, se describe la guía de usuario, que nos enseña la manera de gestionar y manipular la información.

El sistema informático consta de formularios, los cuales sirven tanto para ingreso, configuración y reportes de la información almacenada en la base de datos y que a su vez está siendo consultada a través de los controles que posee cada uno de ellos.

**PASOS:**

1. Lo primero que debe hacerse es ingresar con la cuenta administrador que por defecto es jgonzalesp tanto para usuario y 123456 para contraseña, con el objetivo de configurar las cuentas de usuarios.
2. Abrir el formulario área e ingresar los datos requeridos en ese formulario.
3. En el formulario aula ingresar datos y guardar.
4. Abrir el formulario niveles, para introducir los respectivos niveles.

Existen 3 niveles de educación:

- Inicial: De 3 a 5 años
  - Primaria: De PRIMERO A SEXTO
  - Secundaria: De Primero a Quinto
5. En el formulario año ingresar el año lectivo.
  6. Nos dirigimos al formulario grado e ingresamos la información solicitada.
  7. Posteriormente nos dirigimos al formulario secciones e ingresamos datos requeridos.

8. Abrir el formulario instituciones educativas e ingresar datos y guardar.
9. Configurar el periodo lectivo e ingresar los datos requeridos en ese formulario.
10. Lo siguiente es ingresar los cursos que se darán en el periodo académico.
11. A continuación, abrimos el formulario alumnos e ingresamos los datos.
12. Abrir el formulario docentes e ingresamos datos requeridos.
13. En el formulario personal agregamos información requerida.
14. Configurar el formulario parientes e ingresar datos.
15. Es sustancial relacionar los cursos con su respectivo docente, para realizar este proceso debemos abrir el formulario asignación de cursos.
16. En el formulario registro matrícula podremos matricular a los alumnos en el periodo correspondiente, asimismo en el formulario matrícula se visualizará el reporte general e individual.
17. Para configurar los promedios nos dirigimos al formulario promedio de curso por periodo, ingresar datos requeridos así mismo seleccionar la opción enviar nota. La información se enviará por correo electrónico al padre de familia.
18. En el formulario marcar asistencia ingresar datos requeridos así mismo seleccionar la opción enviar notificación. La información se enviará por correo electrónico al padre de familia.
19. En el formulario citación ingresar información requerida. La citación se enviará por correo electrónico al padre de familia.

## 2. Introducción

Para un completo y detallado manejo de la aplicación, se ha diseñado un manual que le guiará paso a paso cada uno de los procesos y detalles de los formularios del sistema informático.

Se incorpora cada uno de los menús con sus respectivas opciones que la aplicación posee, ya sea botones, mensajes, pantalla u otras características que necesitan una completa explicación funcional dependiendo de los procesos a ejecutar.

## 3. Descripción del sistema



The image shows a login interface for 'Colegio Miguel Angel Buonarroti'. The title 'Colegio' is in a large, bold, dark font, and 'Miguel Angel Buonarroti' is in a smaller, regular, dark font below it. The background is a light gray. In the center, there is a white rectangular box containing the login form. Inside this box, the text 'Ingresa tus datos para iniciar sesión' is centered at the top. Below this, there are two input fields: 'Usuario' with an email icon on the right, and 'Contraseña' with a lock icon on the right. To the right of the 'Contraseña' field is a blue button with the text 'Ingresar'. Below the input fields, there are two links: 'Olvidé mi contraseña' and 'Solicitar registro', both in blue text.

*Figura 59. Login de Usuarios*

*Fuente: Elaboración Propia*

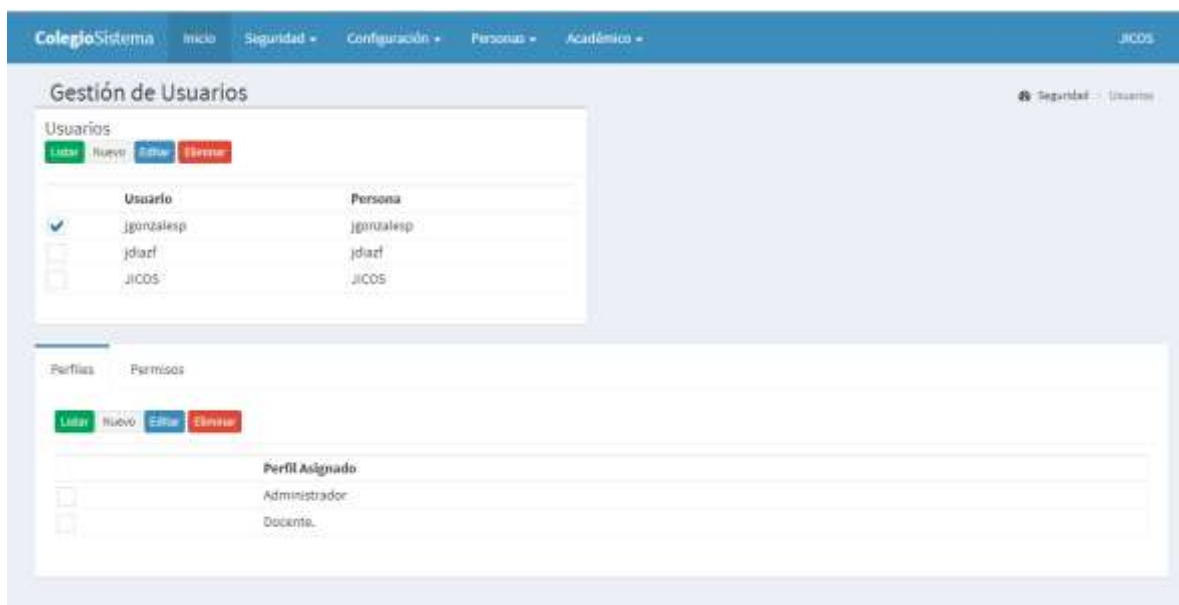


Figura 60. Registro de Usuarios  
Fuente: Elaboración Propia

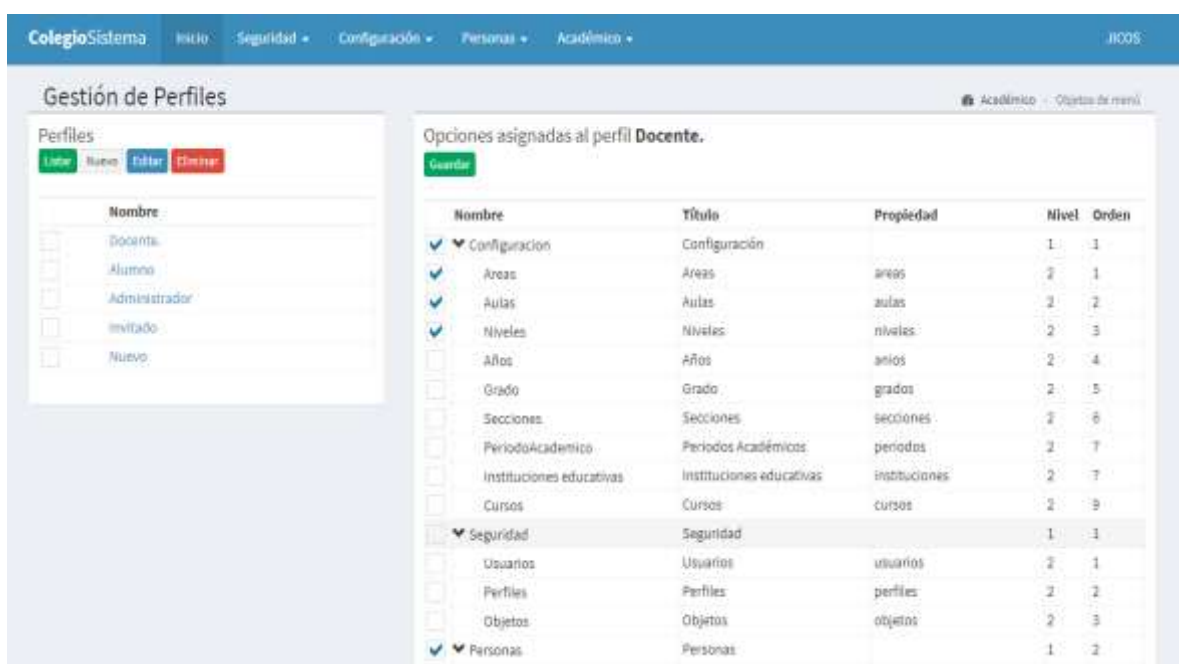


Figura 61. Registro de Perfiles  
Fuente: Elaboración Propia

ColegioSistema					
Inicio Seguridad Configuración Personas Académico					
Gestión de Objetos de Menú					
Lista					
<div> <input type="button" value="Nuevo"/> <input type="button" value="Editar"/> <input type="button" value="Eliminar"/> </div>					
Nombre	Título	Propiedad	Nivel	Orden	
▼ Configuración	Configuración		1	1	
Area	Area	areas	2	1	
Aulas	Aulas	aulas	2	2	
Niveles	Niveles	niveles	2	3	
Años	Años	anos	2	4	
Grado	Grado	grados	2	5	
Secciones	Secciones	secciones	2	6	
PeriodoAcademico	Periodos Académicos	periodos	2	7	
Instituciones educativas	Instituciones educativas	instituciones	2	7	
Cursos	Cursos	curso	2	9	
▼ Seguridad	Seguridad		1	1	
Usuarios	Usuarios	usuarios	2	1	
Perfiles	Perfiles	perfiles	2	2	
Objetos	Objetos	objetos	2	3	
► Personas	Personas		1	2	
► Académico	Académico		1	4	

Figura 62. Registro de Menú

Fuente: Elaboración Propia

ColegioSistema
Inicio
Configuración
Seguridad
Personas
Académico

Gestión de Parientes

Lista

N° Doc.

47264482

47284482

62325252

78727874

REGISTRO DE DATOS

Nro Doc.

Nombres

Apellido Paterno

Apellido Materno

Dirección

Dpto./Provincia/Distrito

Departamento

Provincia

Distrito

Vive

☐

Celular

Correo

Escolaridad

Guardar

Cancelar

Figura 63. Registro de Padre o Apoderado

Fuente: Elaboración Propia

**Registro de Datos de Alumno**

Nro. Doc.

Nombres

Apellido Paterno

Apellido Materno

Sexo

Dirección

Departamento

Provincia

Distrito

Teléfono

Correo

Inclusive

Figura 64. Registro de Alumnos  
Fuente: Elaboración Propia

**Registro de Datos**

Nro. Doc.

Nombres

Apellido Paterno

Apellido Materno

Celular

Correo

Fecha Nac.

Figura 65. Registro de Personal  
Fuente: Elaboración Propia

**Registro de Docentes**

REGISTRO DE DATOS

Nro Doc.

Nombres

Apellido Paterno

Apellido Materno

Celular

Correo

Especialidad

Grado de Instrucción

Título

Fecha Nac.

Figura 66. Registro de Docentes  
Fuente: Elaboración Propia

**Registro de Año Académico**

REGISTRO DE DATOS

Año

Fecha Inicio

Fecha Fin

Comentario

Figura 67. Registro de Año Académico  
Fuente: Elaboración Propia



**Registro de Periodos**

**Filtros:**  
 Seleccione Año: 2018  
 Listar Nuevo Editar Eliminar

**Nombre:**  
☐ PRIMER TRIMESTRE  
☐ SEGUNDO TRIMESTRE  
☐ TERCER TRIMESTRE

**REGISTRO DE DATOS**

Nombre:   
 Siglas:   
 Fecha Inicio: 2018-09-29  
 Fecha Fin: 2018-09-29  
 Año: Seleccione  
 Estado:

Guardar Cancelar

Figura 68. Registro de Periodos  
 Fuente: Elaboración Propia

**Registro de Instituciones**

**Lista:**  
 Listar Nuevo Editar Eliminar

**Nombre:**  
☐ Colegio Santo Toribio de Mogro  
☐ Colegio Santo Tomás de Aquino  
☐ Colegio Santa María Reina

**REGISTRO DE INSTITUCIONES**

Nombres:   
 Código Modular:

Guardar Cancelar

Figura 69. Registro de Instituciones  
 Fuente: Elaboración Propia

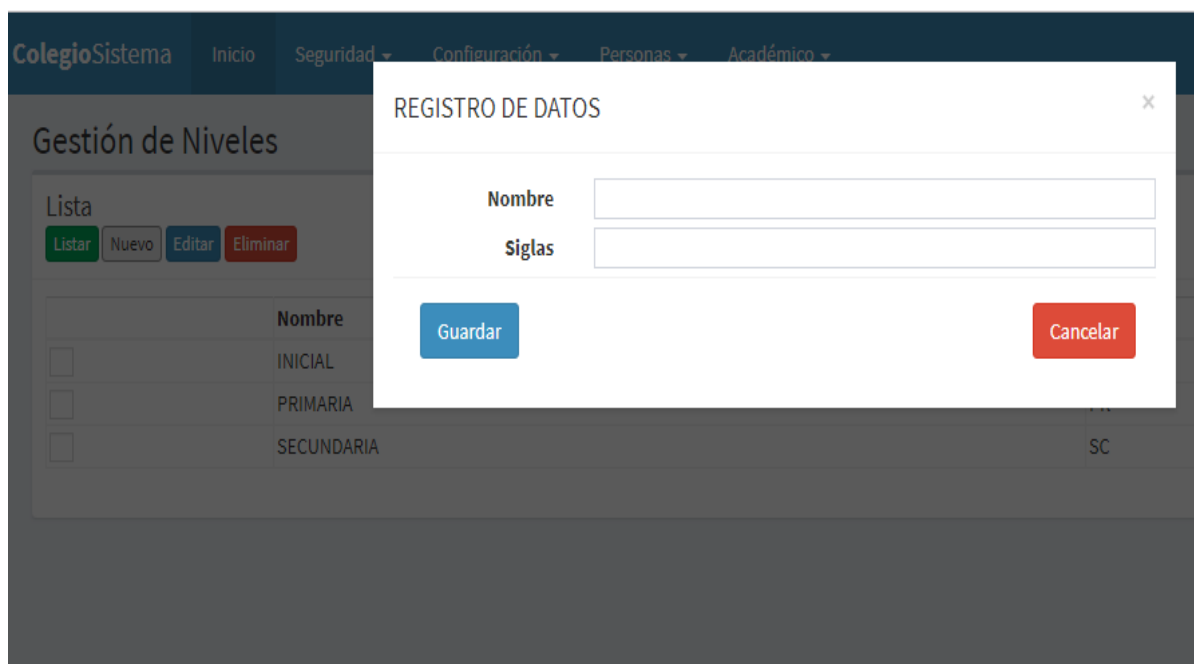


Figura 70. Registro de Niveles  
Fuente: Elaboración Propia

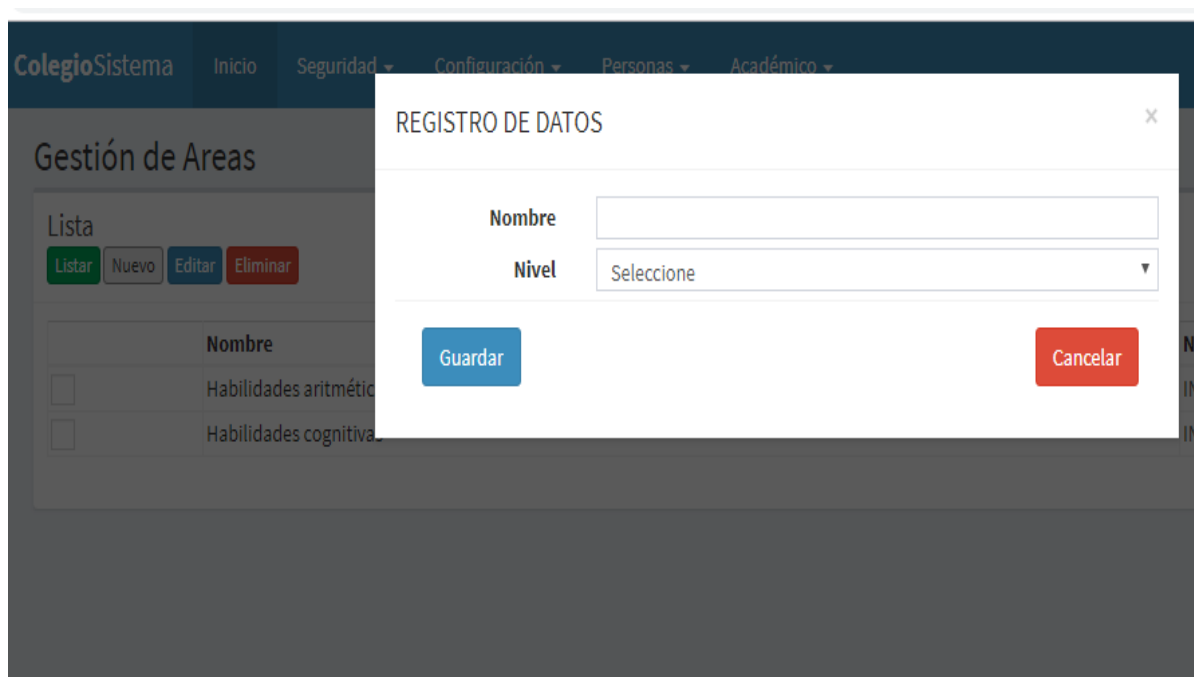


Figura 71. Registro de Áreas  
Fuente: Elaboración Propia

**Registro de Grados**

**REGISTRO DE DATOS**

Nombre: SEGUNDO

Número:

Tipo Calificación: Cualitativa

Nivel: INICIAL

Guardar Cancelar

Figura 72. Registro de Grados  
Fuente: Elaboración Propia

**Registro de Cursos**

**REGISTRO DE CURSO**

Nombre:

Horas: 0

Grado: INICIAL - 3 años

Area: INICIAL - Habilidades aritméticas

Guardar Cancelar

Figura 73. Registro de Cursos  
Fuente: Elaboración Propia

**ColegioSistema** Inicio Configuración Seguridad Personas Académico

### Gestión de Aulas

Lista Listar Nuevo Editar Eliminar

	Número
<input type="checkbox"/>	1
<input type="checkbox"/>	3

**REGISTRO DE DATOS**

Número:

Capacidad:

Tipo:

Guardar Cancelar

Figura 74. Registro de Aulas  
Fuente: Elaboración Propia

**ColegioSistema** Inicio Configuración Seguridad Personas Académico

### Gestión de Secciones

Lista Listar Nuevo Editar Eliminar

	Letra	Año	Nivel
<input type="checkbox"/>	A	2018	INICIAL
<input type="checkbox"/>	A	2018	INICIAL
<input type="checkbox"/>	B	2018	PRIMARIA
<input type="checkbox"/>	A	2018	PRIMARIA

**REGISTRO DE DATOS**

Letra:

Año:

Nivel:

Grado:

Tutor:

Aula:

Tipo Calificación:

Guardar Cancelar

Figura 75. Registro de Secciones  
Fuente: Elaboración Propia

**ColegioSistema** Inicio Seguridad Configuración Personas Académico

## Gestión de Asignaciones

Filtros

Seleccione Año 2018

Listar Nuevo Editar Eliminar

	Curso
<input type="checkbox"/>	Matemática
<input type="checkbox"/>	Literatura

### REGISTRO DE CURSO

**Docente** PRADO VILCHEZ ALEX

**Curso** Literatura

**Periodo Académico** TERCER TRIMESTRE

**Personal** GONZALES PEREZ JOSE

Guardar Cancelar

Figura 76. Registro de Asignaciones  
Fuente: Elaboración Propia

ColegioSistema Inicio Configuración Seguridad Personas Académico

## Gestión de Matrícula

Registrar Nueva Matrícula

Selecione

Listar Nuevo Editar

Fecha

- ☐ 2018-07-25
- ☐ 2018-07-27
- ☐ 2018-08-04
- ☐ 2018-08-07

### REGISTRO MATRÍCULA

Alumno

Nuevo +

Sección

PRIMARIA - PRIMERO - "A"

▼

Colegio de Procedencia

Colegio Santo Toribio de Mogrovejo

▼

Apoderado

Nuevo +

Parentesco

Selecione

▼

Fecha

2018-09-29

Estado

REGISTRADA

▼

Condición Final

APROBADO

▼

Docente

Selecione

▼

Personal

Selecione

▼

Guardar

Cancelar

Figura 77. Registro Matrícula  
Fuente: Elaboración Propia

**ColegioSistema** Inicio Configuración Seguridad Personas Académico

### Marcar Asistencia

Año: 2018 Grado: PRIMERO Fecha: 2018-09-29  
 Nivel: PRIMARIA Letra: A

[Listar](#) [Guardar](#)

N°	Nombre		Acción
1	VARGAS COLLIQUE VICTORIA	<input checked="" type="checkbox"/>	<a href="#">Enviar Notificación</a>
2	VILLEGAS SANCHEZ MARCELO	<input type="checkbox"/>	<a href="#">Enviar Notificación</a>
3	PEREZ CRUZ MARIA	<input checked="" type="checkbox"/>	<a href="#">Enviar Notificación</a>

Figura 78. Marcar Asistencia  
Fuente: Elaboración Propia

**ColegioSistema** Inicio Configuración Seguridad Personas Académico

### Marcar Asistencia

Año: 2018 Grado: PRIMERO Fecha: 2018  
 Nivel: PRIMARIA Letra: A

[Listar](#) [Guardar](#)

N°	Nombre		Acción
1	VARGAS COLLIQUE VICTORIA	<input checked="" type="checkbox"/>	<a href="#">Enviar Notificación</a>
2	VILLEGAS SANCHEZ MARCELO	<input type="checkbox"/>	<a href="#">Enviar Notificación</a>
3	PEREZ CRUZ MARIA	<input checked="" type="checkbox"/>	<a href="#">Enviar Notificación</a>

Figura 79. Enviar Notificaciones  
Fuente: Elaboración Propia

**ColegioSistema** Inicio Configuración Seguridad Personas Académico

### Promedios de Cursos por Peiodo Académico

Académico > Promedios

Año: 2018 Período: PRIMER TRIMESTRE Fecha: 2019-01-22  
 Nivel: PRIMARIA Grado: PRIMERO Letra: A  
 Curso: Matemática Docente: TERAN SANTA CRUZ FRANKLIN

Listar Guardar

N°	Nombre	Promedio	Enviar Nota
1	PEREZ CRUZ MARIA	15	Enviar Nota
2	RAMIREZ LI JUAN	0	Enviar Nota
3	SANCHEZ PEREZ JOSE	0	Enviar Nota
4	VARGAS COLLIQUE VICTORIA	0	Enviar Nota
5	VILLEGAS SANCHEZ MARCELO	0	Enviar Nota

Figura 80. Registro de Promedios de Cursos  
Fuente: Elaboración Propia

**ColegioSistema** Inicio JICOS

### Gestión de Citaciones

Académico > Matrícula

Fecha Desde:   
 Fecha Hasta:

Seleccione Apoderado:

Listar Nuevo Editar Eliminar

Fecha	Apoderado	Asunto	Acción
<input type="checkbox"/> 2019-01-20	SANCHEZ UBILLUS ERMELINDA	Reunión de rutina"	<input type="button" value="X Eliminar"/>
<input type="checkbox"/> 2019-01-20	GONZALES HERNANDEZ JOSE MANUEL	Reunión Urgente"	<input type="button" value="X Eliminar"/>

Figura 81. Registrar Citación  
Fuente: Elaboración Propia



## **CAPÍTULO VI**

### **ASPECTO ADMINISTRATIVO**

## CAPÍTULO VI: ASPECTO ADMINISTRATIVO

### 6.1. CRONOGRAMA DE ACTIVIDADES

Nº	DESCRIPCION	MAYO				JUNIO				JULIO				AGOSTO				SETIEMBRE				OCTUBRE			
1	<b>ITERACIÓN 1:</b>																								
2	Planificación																								
3	Diseño																								
4	Codificación																								
5	Pruebas																								
6	<b>ITERACIÓN 2:</b>																								
7	Planificación																								
8	Diseño																								
9	Codificación																								
10	Pruebas																								
11	<b>ITERACIÓN 3:</b>																								
12	Planificación																								
13	Diseño																								
14	Codificación																								
15	Pruebas																								
16	<b>ITERACIÓN 4:</b>																								
17	Planificación																								
18	Diseño																								
19	Codificación																								
20	Pruebas																								
21	<b>ITERACIÓN 5:</b>																								
22	Planificación																								
23	Diseño																								
24	Codificación																								
25	Pruebas																								
26	<b>ITERACIÓN 6:</b>																								
27	Planificación																								
28	Diseño																								
29	Codificación																								
30	Pruebas																								
31	<b>ITERACIÓN 7:</b>																								
32	Planificación																								
33	Diseño																								
34	Codificación																								
35	Pruebas																								

## 6.2. ANÁLISIS DE COSTOS

### 3.3.2. Costo de materiales

Tabla 68  
Costo de Materiales

MATERIAL	COSTO
Papel Bond A4 (2 millar)	S/. 36.00
Lapiceros, Lápices, Borrador	S/. 12.00
Copias, Impresiones, Anillados	S/. 100.00
Fólder, USB	S/. 25.00
CD	S/. 30.00
<b>TOTAL</b>	<b>S/. 203.00</b>

Fuente: Elaboración Propia

### 3.3.3. Costo de servicios

Tabla 69  
Costo de Servicios

SERVICIO	COSTO
Internet y Teléfono	S/. 1200.00
Servicio de Luz	S/. 1200.00
Pasajes	S/. 1300.00
<b>TOTAL</b>	<b>S/. 3700.00</b>

Fuente: Elaboración Propia

### 3.3.4. Costo personal

*Tabla 70*  
*Costo de Personal*

PERSONA	REMUNERACIÓN
Analista, Diseñador	S/. 7000.00
Programador	S/. 7000.00
<b>TOTAL</b>	<b>S/. 14000.00</b>

*Fuente:* Elaboración Propia

### Bienes materiales

*Tabla 71*  
*Bienes Materiales*

BIEN	COSTO
3 PCs de Escritorio	S/. 4800.00
Antivirus (Licencia original)	S/. 120.00
Windows 10 (Licencia original)	S/. 130.00
<b>TOTAL</b>	<b>S/. 5050.00</b>

*Fuente:* Elaboración Propia

### 3.3.5. Resumen de costos

*Tabla 72*  
*Resumen de Costos*

BIEN	COSTO
MATERIALES	S/. 178.00
SERVICIOS	S/. 3700.00
PERSONALES	S/. 14000.00
BIENES MATERIALES	S/. 5050.00
TOTAL	S/. 22928.00

---

*Fuente:* Elaboración Propia

### 3.3.6. Financiamiento

El proyecto estará financiado por el Colegio “Miguel Angel Buonarroti”.

## **RESULTADOS**

Teniendo en cuenta el primer objetivo se ejecutó exitosamente la etapa de análisis, los datos se obtuvieron por medio de las diferentes visitas a la institución educativa, la cual se mantuvieron diferentes conversaciones con las personas involucradas entre ellos el Director y la secretaria, con el fin de conocer las características del proceso académico.

Los requerimientos funcionales expresan la naturaleza del funcionamiento del sistema tomando como referencia lo antes citado sirvió para plasmar los requerimientos funcionales de un sistema informático que ayude a realizar las actividades de matrícula y notificaciones de manera eficiente y eficaz.

Se describió el enfoque del desarrollo del proyecto a través de la metodología utilizada.

Se logró definir las actividades que se realizan, dentro del proceso de desarrollo, en términos de las fases e iteraciones para el desarrollo del Sistema Informático para el colegio Miguel Angel Buonarroti.

Se procedió a codificar el sistema, se utilizó NetBeans IDE para la generación de código del sistema.

Se realizaron las pruebas que consistió en validar la información que se almacena en la base de datos, verificar los resultados, detección de errores y así garantice un sistema exitoso.

## DISCUSIÓN

Las herramientas informáticas son realmente necesarias para el progreso de toda organización.

Los sistemas informáticos son realmente necesarios, en el caso de las instituciones educativas resulta muy provechoso el uso de sistemas informáticos que le permitan llevar un control oportuno de sus labores emprendidas, así como también de los datos de profesores, colaboradores, padres de familia y estudiantes en general. Mora y Vega, (2013).

Teniendo en cuenta el uso de los sistemas informáticos en las instituciones educativas, se tomó como referencia análisis, diseño e implementación de un sistema informático para el apoyo al proceso de asignación de la carga académica usando programación con restricciones. Gonzales y Quintero, (2010), en el cual se pudo evidenciar que el programa manejaba todos los procesos y reportes necesarios para garantizar un eficaz y eficiente sistema. Sin embargo, no facilita el manejo del sistema, por cual no cumple con la recomendación que según, Osorio N, (2016), señala que estos sistemas que se puede realizar por módulos, los cuales se pueden integrar fácilmente al sistema actual.

## CONCLUSIONES

1. La información brindada por la institución educativa “Miguel Angel Buonarroti” permitió definir todos los requerimientos para el desarrollo del presente proyecto.
2. El uso de Hibernate para la persistencia de datos también fue una gran ventaja ya que permitió ahorrar muchas líneas de código (SQL), por ejemplo, trabajar más fácilmente con transacciones.
3. La metodología XP (Extreme Programming) permitió un eficiente proceso de planificación y pruebas, además facilitó los cambios que se produjeron tanto en el desarrollo como en producción.
4. La utilización de Spring como framework principal en la arquitectura de nuestro proyecto permitió codificar más rápidamente las capas del sistema, haciendo uso de su sistema de anotaciones, las cuales hacen que se ahorre una gran cantidad de código, además de tener un buen sistema de dependencias y una arquitectura ordenada.
5. Las notificaciones emitidas mejoró el tiempo de entrega de la información al padre de familia o apoderado.
6. Las pruebas realizadas a la aplicación permitió determinar el funcionamiento del software.



## **RECOMENDACIONES**

1. Se deberá tomar en cuenta todas las herramientas necesarias al momento de la recopilación de información, para que esta sea clara, precisa y útil en la realización del sistema.
2. Para desarrollar el software se debe trabajar en base a los requerimientos encontrados y la estructura de la base de datos diseñada.
3. El diseño de la base de datos debe estar muy bien estructurado, definiendo las tablas adecuadas para almacenar información, con el fin de que no exista inconsistencia de datos.
4. Se recomienda la participación activa del usuario en todo el proceso de desarrollo del proyecto.
5. Se recomienda utilizar gestor de base de datos sin costo.
6. Se recomienda la realización de Backups para prever problemas en la seguridad de la información.

## REFERENCIAS BIBLIOGRÁFICAS

Alegsa, L. (2018, Junio 10). *Sistema Informático*. Recuperado de :

[http://www.alegsa.com.ar/Dic/sistema\\_informatico.php](http://www.alegsa.com.ar/Dic/sistema_informatico.php)

*Apache Servidor HTTP*.(2017, Junio 11) . Recuperado de

<http://www.ajpdsoft.com/modules.php?name=Encyclopedia&op=content&tid=820>

*Aplicación Web*.(20018, Junio 13). Recuperado de

<https://www.neosoft.es/blog/que-es-una-aplicacion-web/>

Bootstrap. (2011, Febrero 16).Recuperado de

<http://programacion.jias.es/2015/05/web-%C2%BFque-es-el-framework-bootstrap-ventajas-desventajas/>

Charte, F. (2010). *Sql*. Madrid: Anaya Multimedia ( Grupo Anaya, S.A).

Córdova, J. (2014, Febrero 20). *Implementación de un sistema de matrículas y*

*pagos para el centro de informática de la universidad César Vallejo*.Tesis

para optar el Título profesional de Ingeniero de Estadística e Informática.

Recuperado de:

[http://www.repositorioacademico.usmp.edu.pe/bitstream/usmp/1030/1/cordova\\_ja.pdf](http://www.repositorioacademico.usmp.edu.pe/bitstream/usmp/1030/1/cordova_ja.pdf)

Cubas, R.,y Nuñez, J. (2015, Marzo 16). *Plan estratégico de sistemas y*

*tecnologías de información para optimizar la gestión educativa de la*

*institución privada Amancio Varona - Tumbán*. Tesis para obtener el Título

profesional de Ingeniero en Computación e Informática. Recuperado de:

<http://repositorio.unprg.edu.pe/bitstream/handle/UNPRG/472/BC-TES-4233.pdf?sequence=1&isAllowed=y>

Date, C. (2001). *Introducción a los sistemas de base de datos*. México: Pearson Educación.

DuBois, P. (2009). *MySQL*. Madrid: Anaya Multimedia ( Grupo Anaya, S.A).

*Erwin Data Modeler*. (2018, Mayo 18). Recuperado de  
<https://erwin.com/products/data-modeler/>

Estela, E. (2016, Febrero 21). *Aplicación móvil de alertas para apoyar la comunicación entre los agentes educativos del colegio san agustín de chiclayo*. Tesis para optar el Título de Ingeniero de Sistemas y Computación. Recuperado de:  
[http://tesis.usat.edu.pe/bitstream/usat/698/1/TL\\_Estela\\_Vasquez\\_EdgarGonzalo.pdf](http://tesis.usat.edu.pe/bitstream/usat/698/1/TL_Estela_Vasquez_EdgarGonzalo.pdf)

González, B., y Quintero, J. (2010, Marzo 15). *Análisis diseño e implementación de un sistema informático para el apoyo al proceso de asignación de la carga académica usando programación con restricciones*. Tesis para optar el Título de Ingeniero de Sistemas y Computación. Recuperado de:  
<http://repositorio.utp.edu.co/dspace/bitstream/handle/11059/1333/6584038011G643.pdf?sequence=1>

*Hibernate*. (2013, Febrero 16). Recuperado de  
<http://blog.educacionit.com/2013/02/07/que-es-java-hibernate/>

*Hibernate Query Language*. (2018, Marzo 16). Recuperado de  
[https://www.tutorialspoint.com/hibernate/hibernate\\_query\\_language.htm](https://www.tutorialspoint.com/hibernate/hibernate_query_language.htm)

*Introducción a la programación extrema*. (2016, Febrero 14). Recuperado de  
<http://www.davidvalverde.com/blog/introduccion-a-la-programacion-extrema-xp/>

Mendoza, G. (2015, Marzo 18). *Herramienta de Desarrollo Netbeans*.

Recuperado de: [https://www.consultorjava.com/wp/wp-content/uploads/2015/09/herramienta\\_desarrollo\\_netbeans.pdf](https://www.consultorjava.com/wp/wp-content/uploads/2015/09/herramienta_desarrollo_netbeans.pdf)

*Modelo Vista Controlador*.(2016, Marzo 11). Recuperado de:

<https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>

Mora, N., y Vega, C. (2013, Febrero 15). *Sistema informático de control de notas y proceso de matriculación de la unidad educativa wenceslao rijavec de la ciudad de calceta - cantón bolívar*. Tesis previa a la obtención del Título de Ingeniero en Informática. Recuperado de:

<http://repositorio.espam.edu.ec/bitstream/42000/71/2/N%C3%89STOR%20ADRI%C3%81N%20MORA%20MAC%C3%8DAS%20%20CRISTHIAN%20XAVIER%20VEGA%20INTRIAGO.pdf>

Osorio, N. (2016, Marzo 12). *Diseño e implementación de un sistema de matrícula web usando software libre en el centro educativo "españa", distrito - breña 2013*. Tesis para optar el Título profesional de Ingeniero de Sistemas e Informática. Recuperado de:

[http://repositorio.uch.edu.pe/bitstream/handle/uch/82/Osorio\\_Alvarez\\_Neil\\_Angelo.pdf?sequence=1&isAllowed=y](http://repositorio.uch.edu.pe/bitstream/handle/uch/82/Osorio_Alvarez_Neil_Angelo.pdf?sequence=1&isAllowed=y)

*Proceso Académico*.(2011, Marzo 11). Recuperado de

<https://dialnet.unirioja.es/descarga/articulo/4165990.pdf>

*Programación Extrema*.(2017, Febrero 19). Recuperado de

<https://geekytheory.com/programacion-extrema-que-es-y-principios-basicos>

*Programacion Extrema xp*.(2010, Marzo 15). Recuperado de

[https://www.ecured.cu/Programaci%C3%B3n\\_Extrema\\_\(XP\)](https://www.ecured.cu/Programaci%C3%B3n_Extrema_(XP))

Rodriguez, T. (2012, Marzo 17). *Bootstrap*. Recuperado de:

<https://www.genbeta.com/desarrollo/bootstrap>

*Spring*.(2018, Febrero 26). Recuperado de

[https://www.tutorialspoint.com/spring/spring\\_overview.htm](https://www.tutorialspoint.com/spring/spring_overview.htm)

Steven, J. (2012, Febrero 24). *Conceptos básicos del lenguaje java*. Recuperado

de: <https://www.ibm.com/developerworks/ssa/java/tutorials/j->

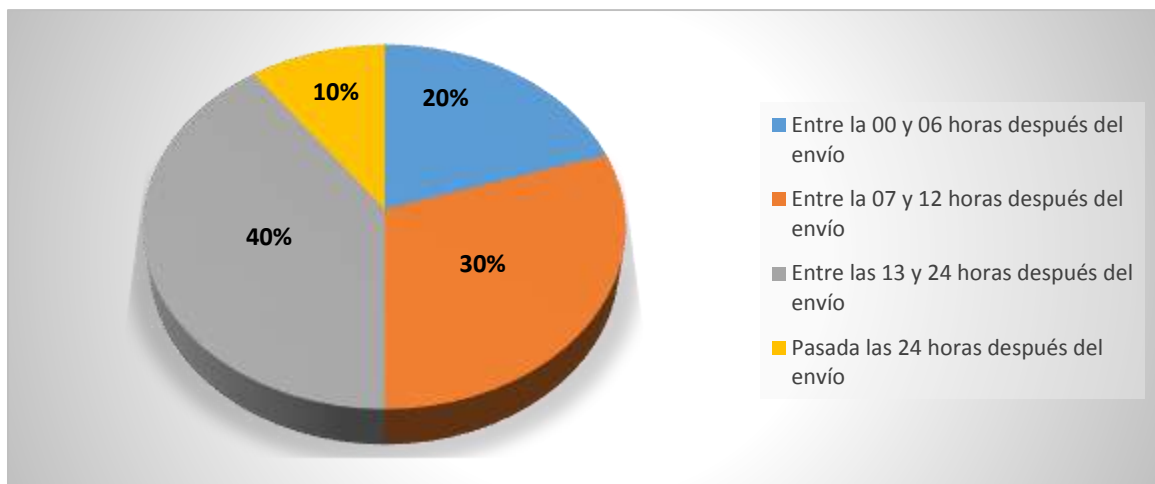
[introtojava1/index.html](https://www.ibm.com/developerworks/ssa/java/tutorials/j-introtojava1/index.html)

## **ANEXOS**

**ENCUESTA REALIZADA AL PADRE DE FAMILIA O APODERADO SOBRE  
EL RANGO DE TIEMPO EN QUE RECEPCIONA Y LEE LA INFORMACIÓN  
EMITIDA POR EL COLEGIO HACIENDO USO DE LOS MEDIOS  
TRADICIONALES**

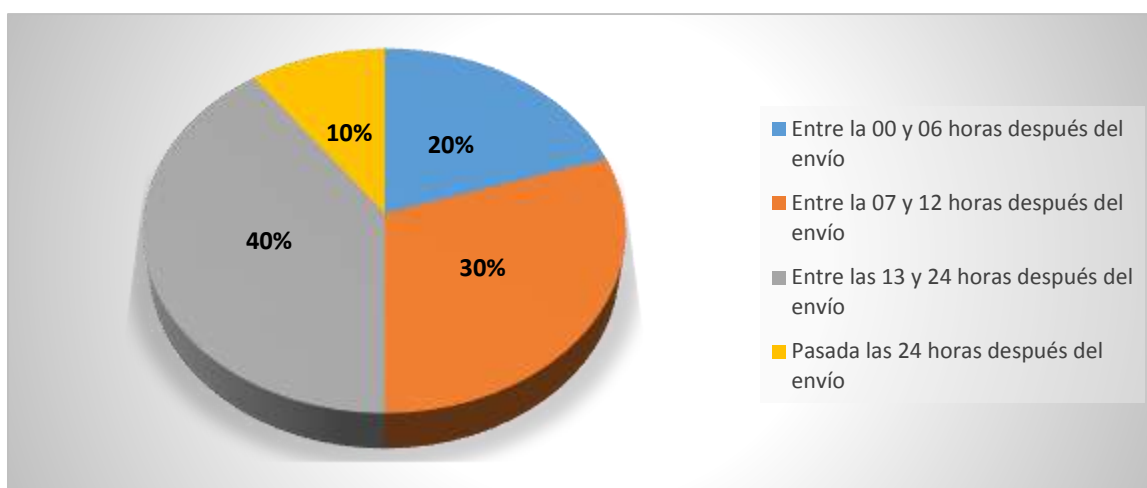
1. ¿En qué rango de tiempo recibe y lee Usted los anuncios enviados por las distintas instancias del colegio, a través de los medios tradicionales luego de haber sido emitidos?

a)	Entre la 00 y 06 horas después del envío	20
b)	Entre las 07 y 12 horas después del envío	30
c)	Entre las 13 y 24 horas después del envío	40
d)	Pasada las 24 horas después del envío	10



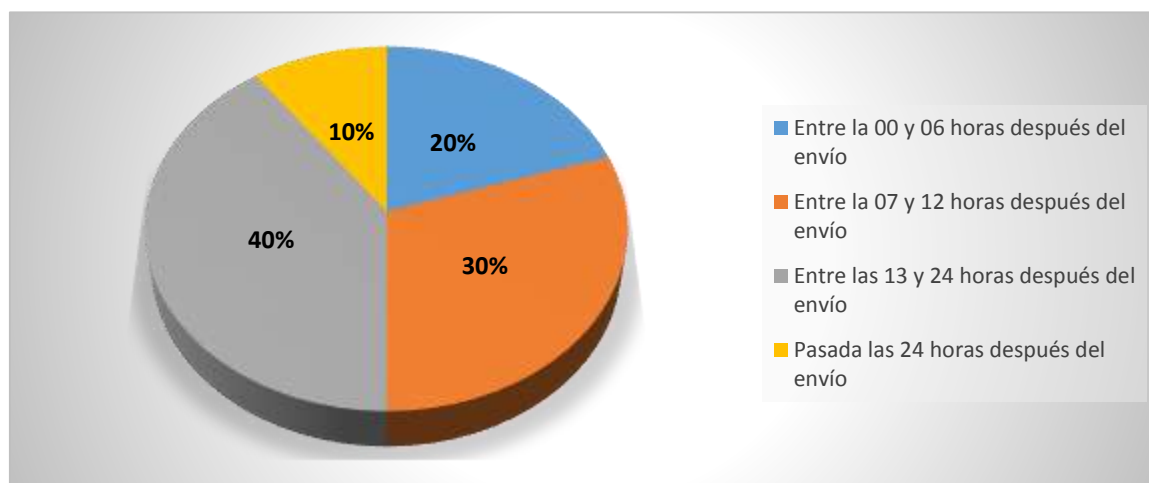
2. ¿En qué rango de tiempo recibe y lee Usted las incidencias enviadas por los profesores de su menor hijo, a través de los medios tradicionales luego de haber sido emitidas?

a)	Entre la 00 y 06 horas después del envío	20
b)	Entre las 07 y 12 horas después del envío	30
c)	Entre las 13 y 24 horas después del envío	40
d)	Pasada las 24 horas después del envío	10



3. ¿En qué rango de tiempo recibe y lee Usted los comunicados enviados por los profesores de su menor hijo, a través de los medios tradicionales luego de haber sido emitidos?

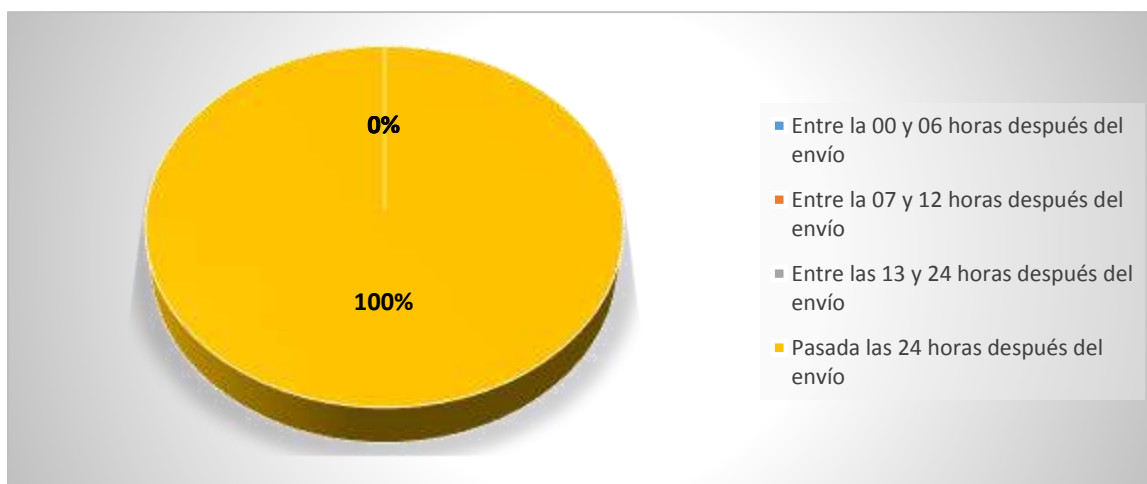
a)	Entre la 00 y 06 horas después del envío	20
b)	Entre las 07 y 12 horas después del envío	30
c)	Entre las 13 y 24 horas después del envío	40
d)	Pasada las 24 horas después del envío	10





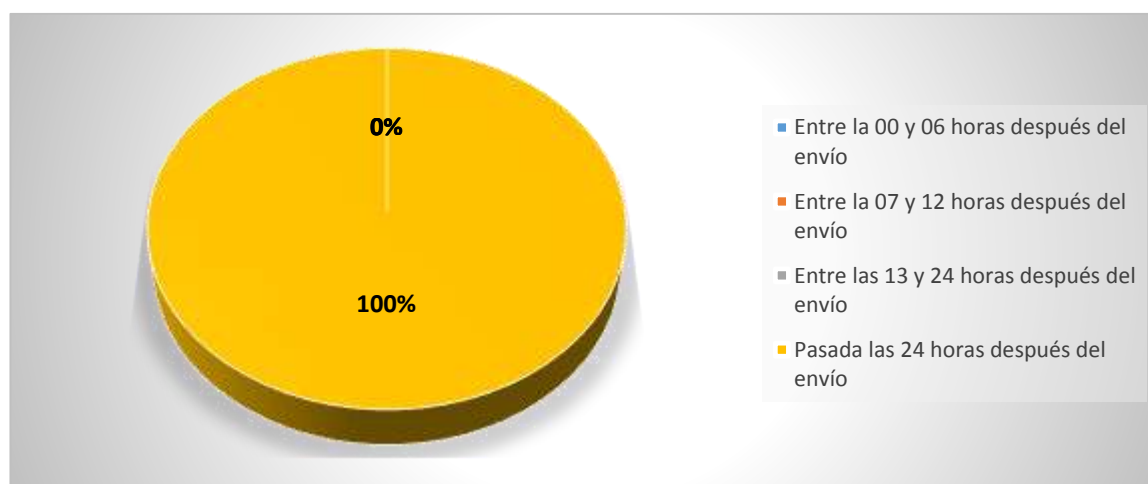
4. ¿En qué rango de tiempo recibe y lee Usted las solicitudes de citación por parte de los profesores de su menor hijo, a través de los medios tradicionales luego de haber sido emitida?

a)	Entre la 00 y 06 horas después del envío	00
b)	Entre las 07 y 12 horas después del envío	00
c)	Entre las 13 y 24 horas después del envío	00
d)	Pasada las 24 horas después del envío	100



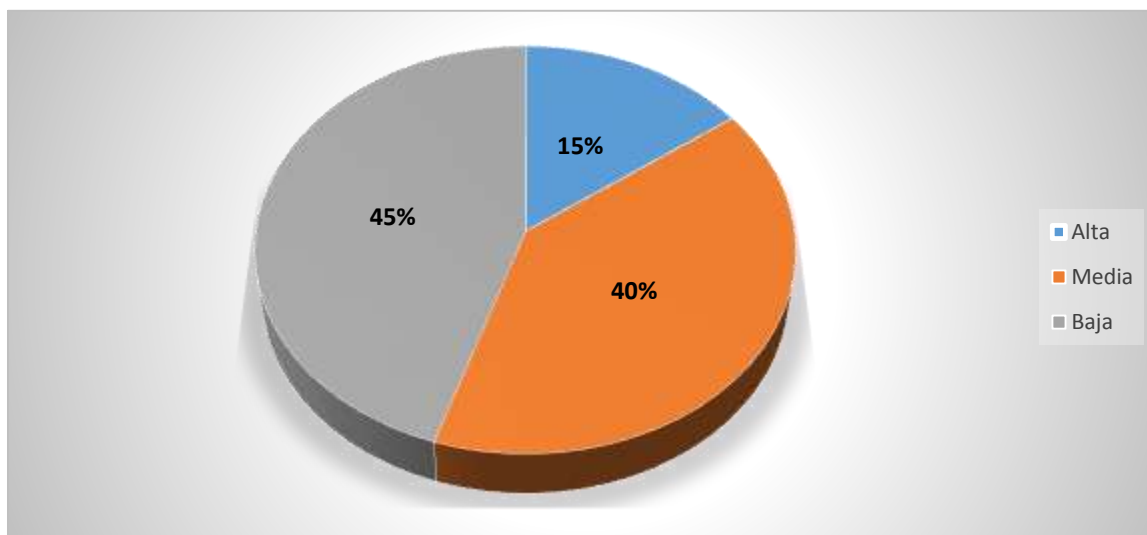
5. ¿En qué rango de tiempo recibe y lee Usted la alerta de tardanza y/o inasistencia de su menor hijo, a través de los medios tradicionales?

a)	Entre la 00 y 06 horas después del envío	00
b)	Entre las 07 y 12 horas después del envío	00
c)	Entre las 13 y 24 horas después del envío	00
d)	Pasada las 24 horas después del envío	100



6.¿Cómo calificaría Usted la efectividad de la transmisión de la información a los padres de familia o apoderados del colegio?

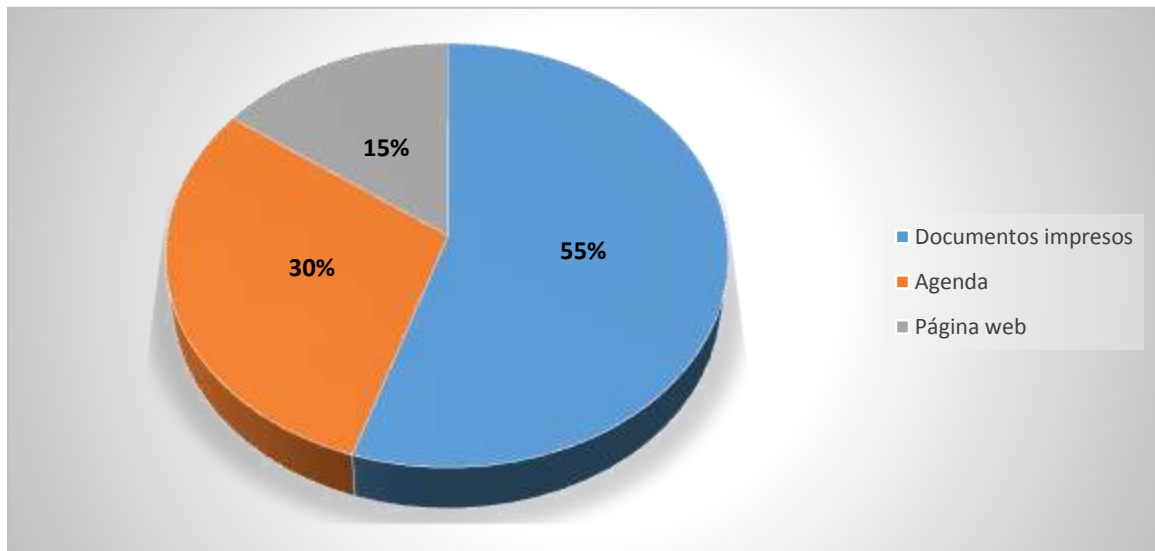
a)	Alta	15
b)	Media	40
c)	Baja	45



**ENCUESTA REALIZADA AL PADRE DE FAMILIA O APODERADO SOBRE  
EL MEDIO UTILIZADO POR EL COLEGIO PARA COMUNICARLE ASUNTOS  
RELACIONADOS CON SU MENOR HIJO**

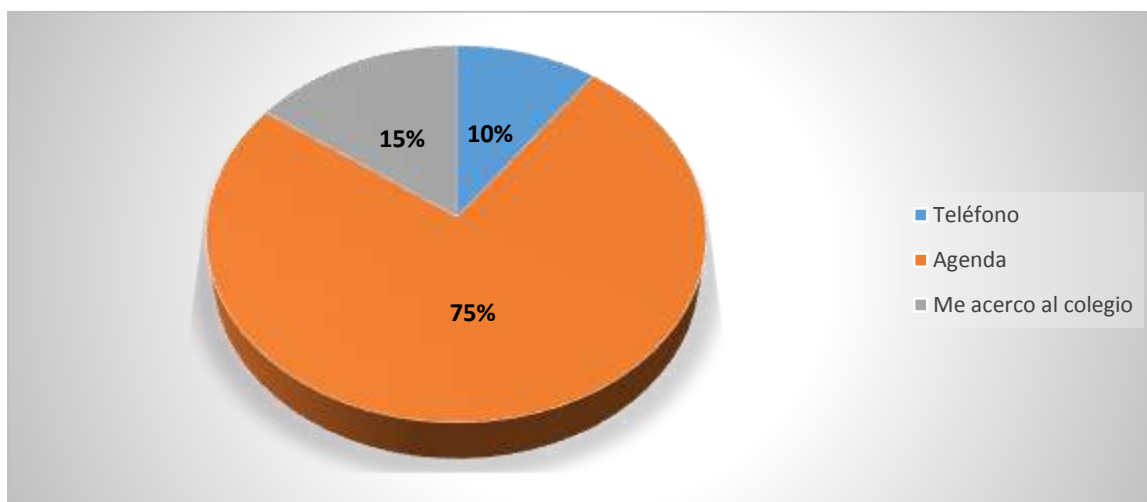
1. ¿Cuál es el medio de comunicación que utiliza el colegio para dar a conocer al padre de familia o apoderado los anuncios emitidos por las diferentes instancias educativas?

a)	Documentos impresos	55
b)	Agenda	30
d)	Página web	15



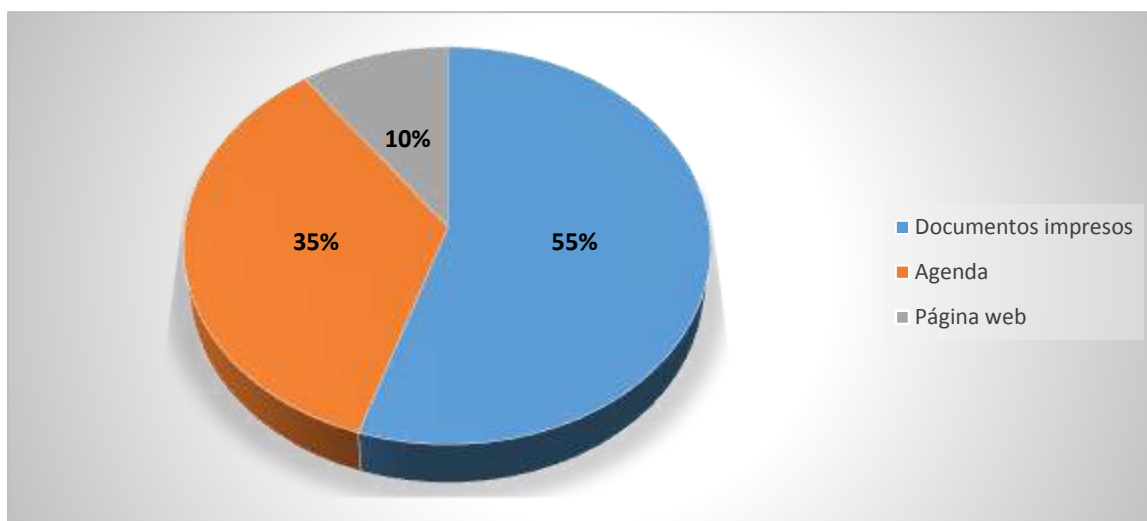
2. ¿Cuál es el medio de comunicación que Usted utiliza para solicitar una citación con los profesores de su menor hijo?

a)	Teléfono	10
b)	Agenda	75
c)	Me acerco al colegio	15



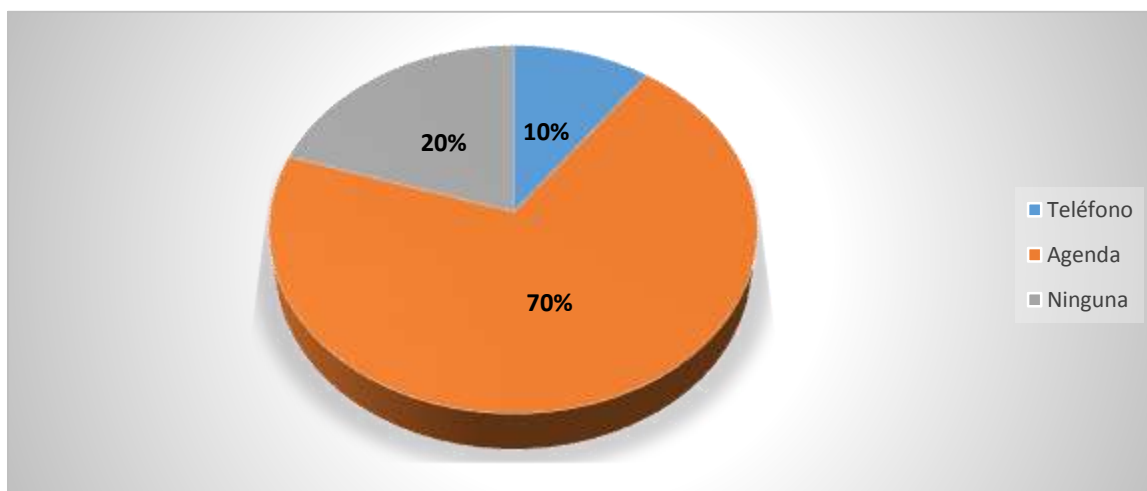
3. ¿Cuál es el medio de comunicación que utilizan los profesores para enviar comunicados al padre de familia o apoderado?

a)	Documentos impresos	55
b)	Agenda	35
c)	Página web	10



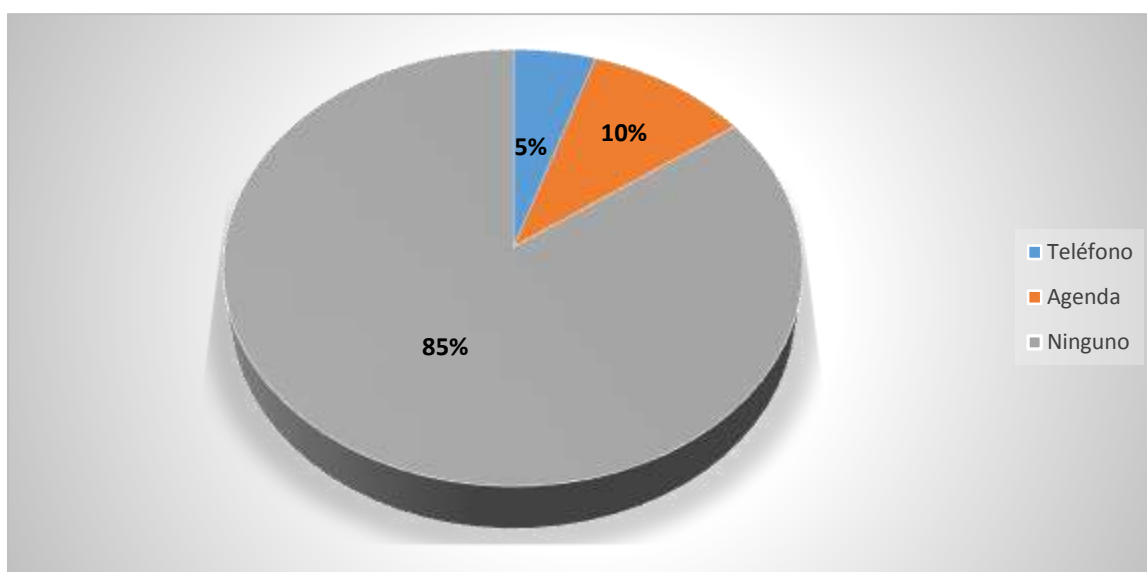
4. ¿Cuál es el medio de comunicación que utilizan los profesores para darle a conocer las incidencias ocurridas en aula, con su menor hijo?

a)	Teléfono	10
b)	Agenda	70
c)	Ninguno	20



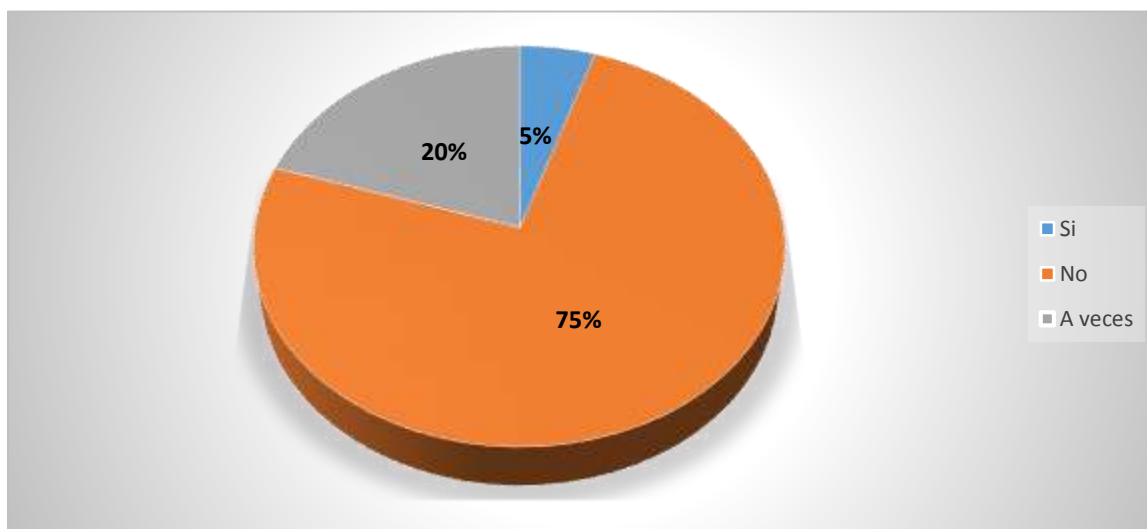
5. ¿Cuál es el medio de comunicación que usa el colegio para dar a conocer las inasistencias y / o tardanzas de su menor hijo al centro de estudios?

a)	Teléfono	05
b)	Agenda	10
c)	Ninguno	85



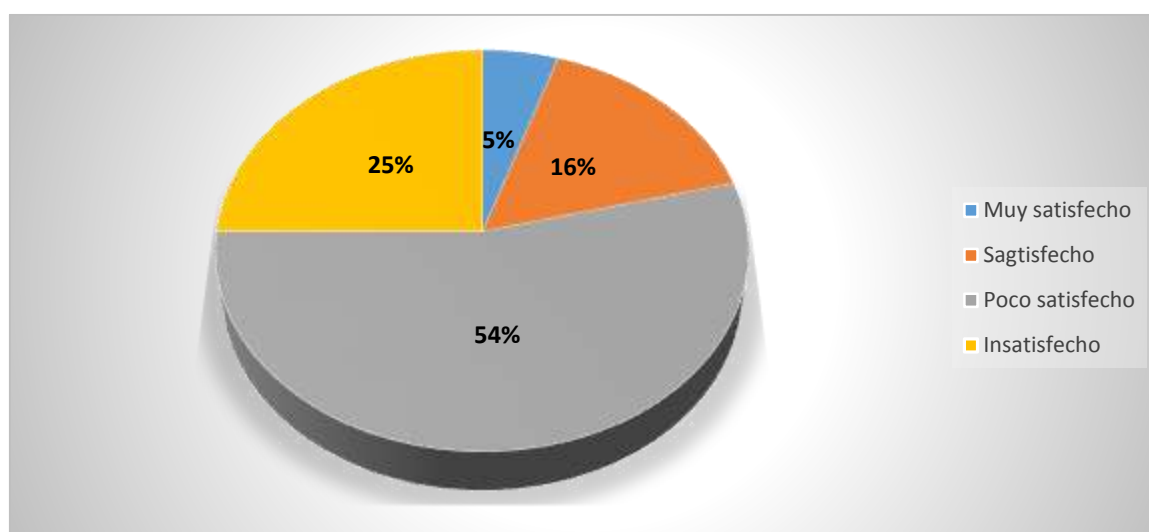
6.¿La información del proceso formativo de su menor hijo, es comunicada de varias formas hasta asegurar que ésta le llegue de manera oportuna hacia Usted ?

a)	Si	5
b)	No	75
c)	A veces	20



7. En cuanto al nivel de comunicación que el colegio tiene con Usted, en general se siente:

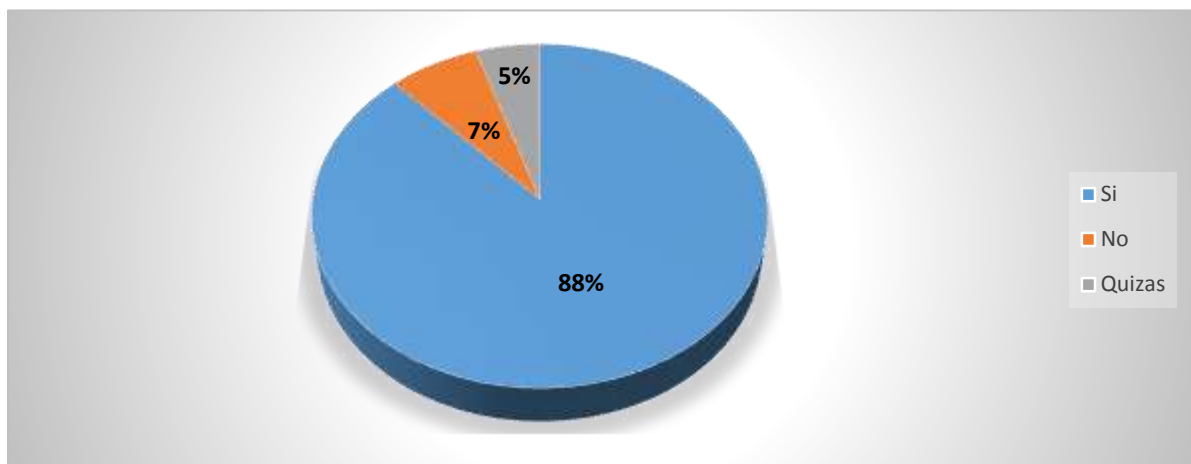
a)	Muy satisfecho	5
b)	Satisfecho	16
c)	Poco satisfecho	54
d)	Insatisfecho	25



**ENCUESTA REALIZADA AL PADRE DE FAMILIA O APODERADO PARA  
CONOCER LA VIABILIDAD DE LA PROPUESTA TECNOLÓGICA EN  
MEJORA DE LA COMUNICACIÓN**

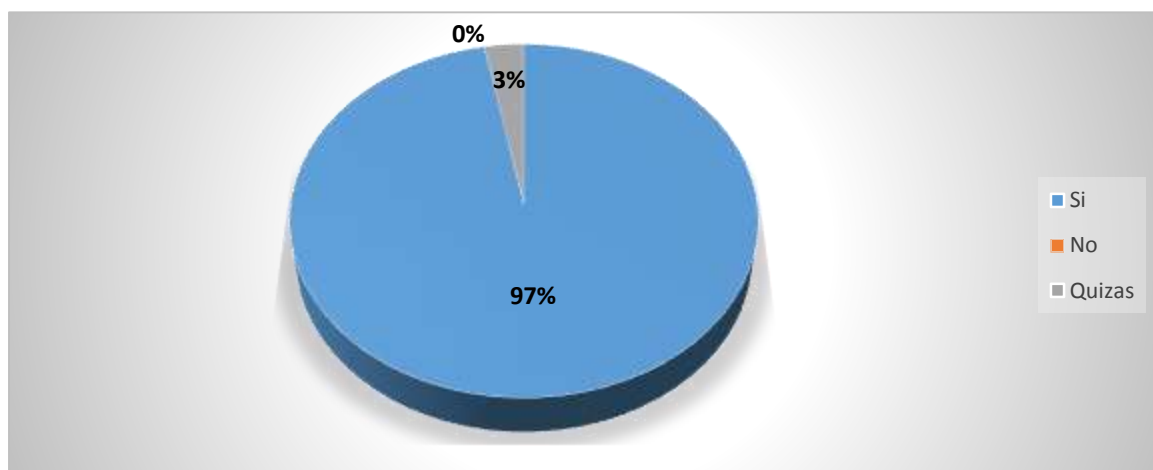
1. ¿Le gustaría recibir, en su correo electrónico, los diversos anuncios emitidos por el colegio de su hijo?

a)	Si	88
b)	No	7
d)	Quizás	5



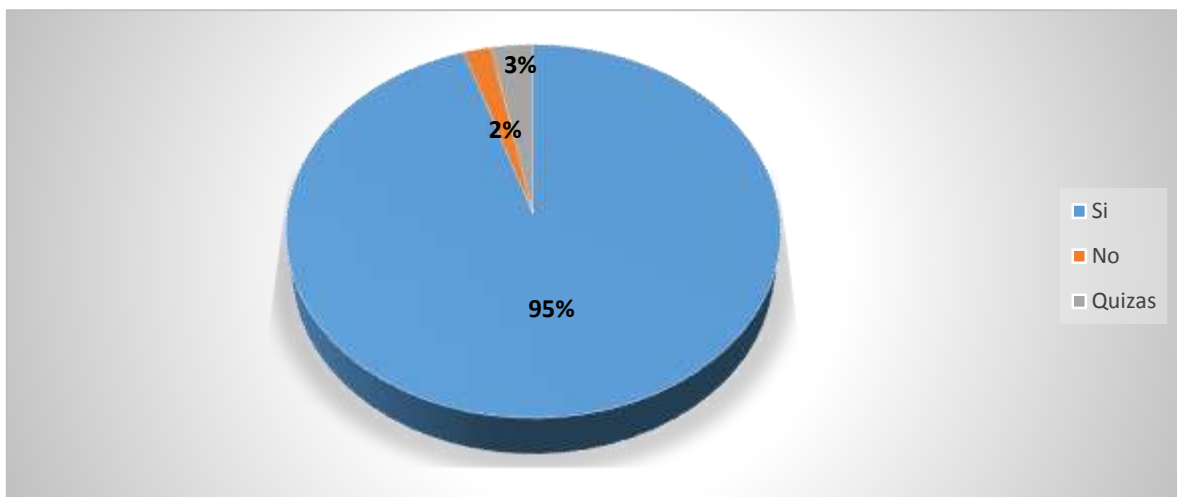
2. ¿Le gustaría recibir, en su correo electrónico, información acerca de las incidencias ocurridas en clase con su menor hijo?

a)	Si	97
b)	No	00
c)	Quizás	3



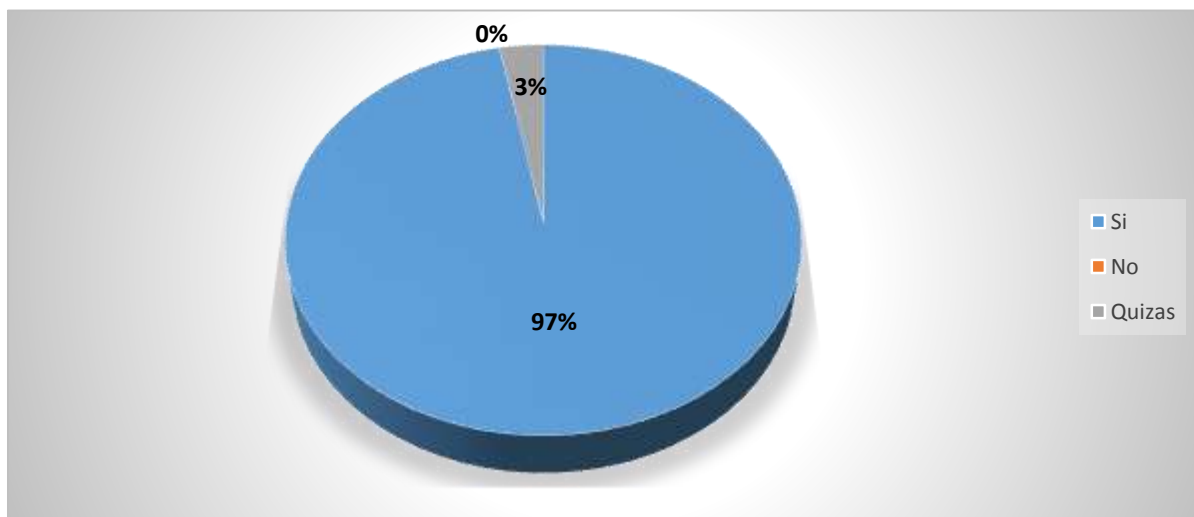
3. ¿Le gustaría recibir, en su correo electrónico, los diversos comunicados emitidos por los profesores de su hijo?

Si	95
No	2
Quizás	3



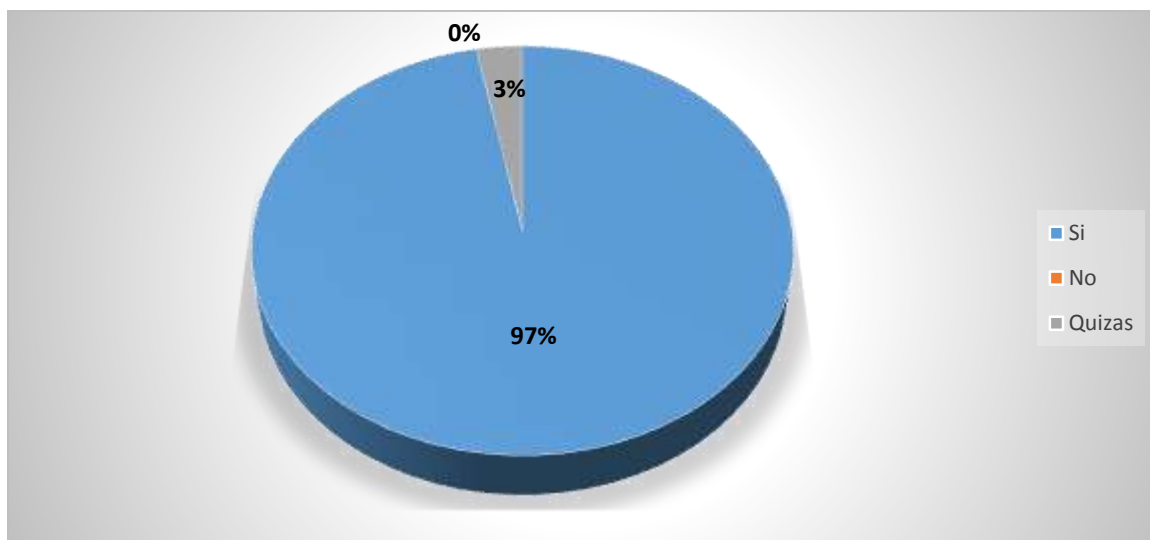
4. ¿Le gustaría recibir, en su correo electrónico información, acerca de las inasistencias y/o tardanzas de su menor hijo al colegio ?

a) Si	97
b) No	00
c) Quizás	3





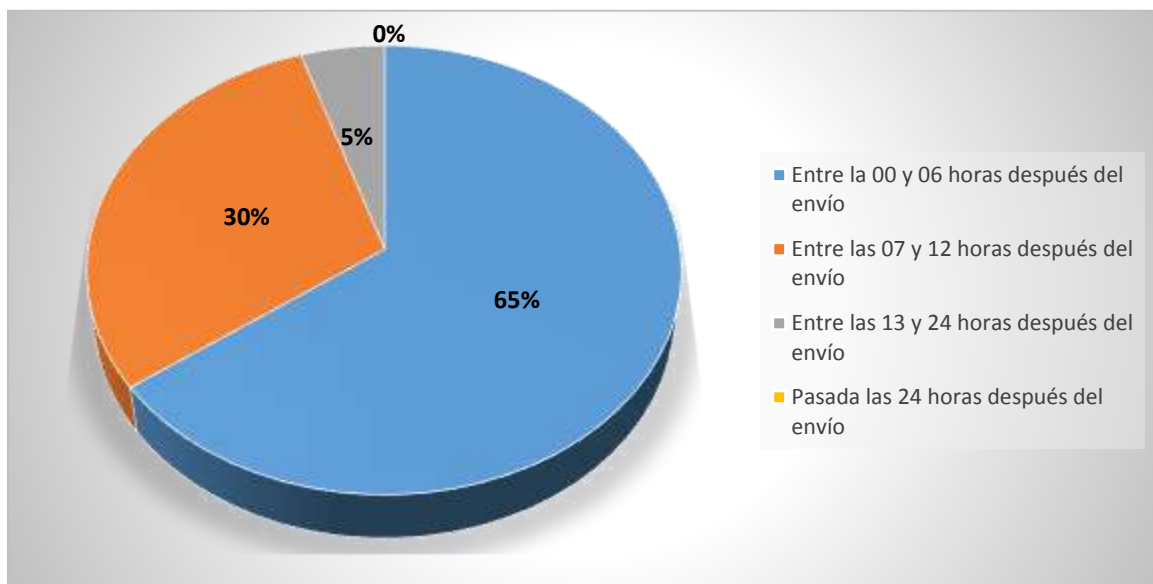
5.¿ Le gustaría solicitar, desde su correo electrónico, una citación con algún profesor de su menor hijo?		
a)	Si	97
b)	No	00
c)	Quizás	03



**ENCUESTA REALIZADA AL PADRE DE FAMILIA O APODERADO SOBRE  
EL RANGO DE TIEMPO EN QUE RECEPCIONA Y LEE LA INFORMACIÓN  
EMITIDA POR EL COLEGIO HACIENDO USO DEL SISTEMA INFORMÁTICO**

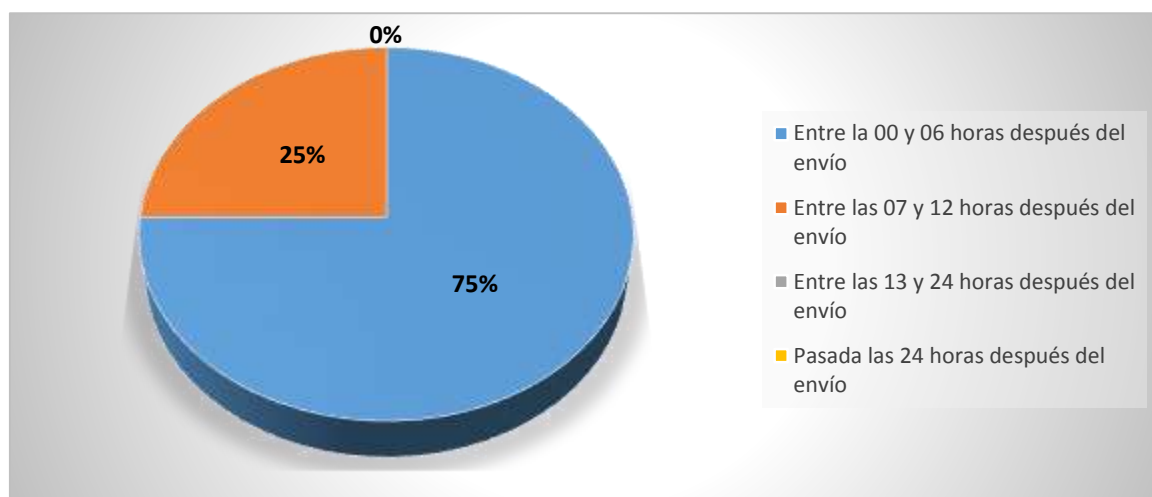
1. ¿En qué rango de tiempo recibe y lee Usted los anuncios enviados por las distintas instancias del colegio, a través de los medios tradicionales luego de haber sido emitidos?

a)	Entre la 00 y 06 horas después del envío	65
b)	Entre las 07 y 12 horas después del envío	30
c)	Entre las 13 y 24 horas después del envío	05
d)	Pasada las 24 horas después del envío	00



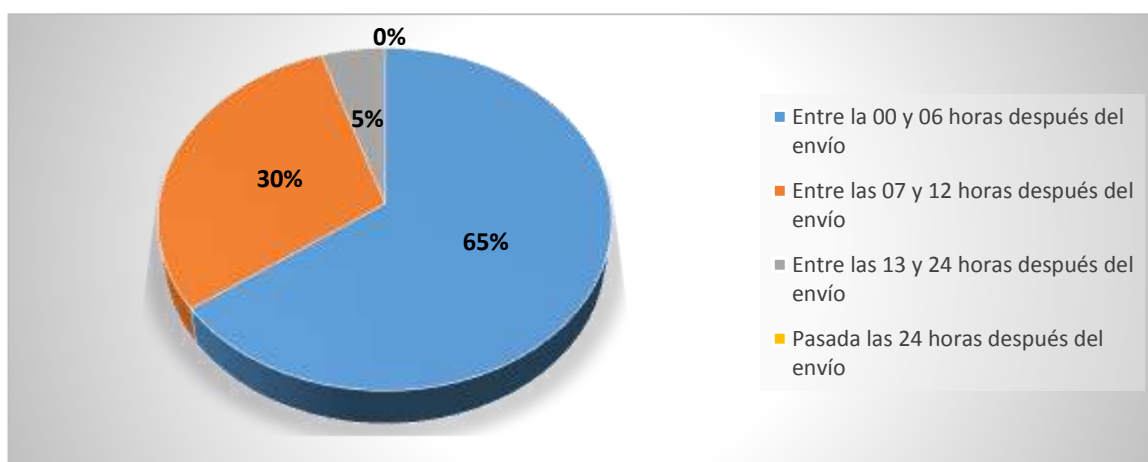
2. ¿En qué rango de tiempo recibe y lee Usted las incidencias enviadas por los profesores de su menor hijo, a través de los medios tradicionales luego de haber sido emitidas?

a)	Entre la 00 y 06 horas después del envío	75
b)	Entre las 07 y 12 horas después del envío	25
c)	Entre las 13 y 24 horas después del envío	00
d)	Pasada las 24 horas después del envío	00



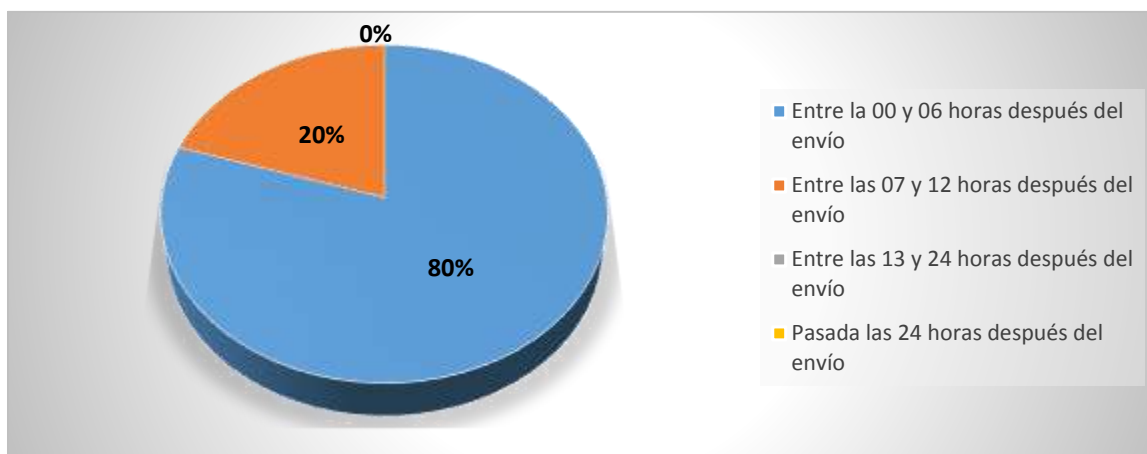
3. ¿En qué rango de tiempo recibe y lee Usted los comunicados enviados por los profesores de su menor hijo, a través de los medios tradicionales luego de haber sido emitidos?

a)	Entre la 00 y 06 horas después del envío	65
b)	Entre las 07 y 12 horas después del envío	30
c)	Entre las 13 y 24 horas después del envío	05
d)	Pasada las 24 horas después del envío	00



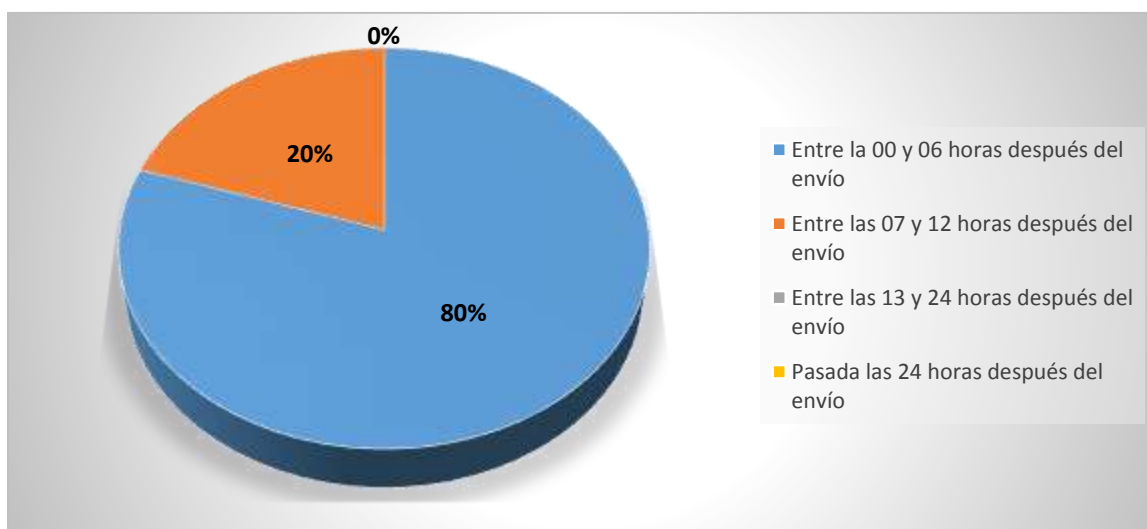
4. ¿En qué rango de tiempo recibe y lee Usted las solicitudes de citación por parte de los profesores de su menor hijo, a través de los medios tradicionales luego de haber sido emitida?

a)	Entre la 00 y 06 horas después del envío	80
b)	Entre las 07 y 12 horas después del envío	20
c)	Entre las 13 y 24 horas después del envío	00
d)	Pasada las 24 horas después del envío	00



5. ¿En qué rango de tiempo recibe y lee Usted la alerta de tardanza y/o inasistencia de su menor hijo, a través de los medios tradicionales?

a)	Entre la 00 y 06 horas después del envío	80
b)	Entre las 07 y 12 horas después del envío	20
c)	Entre las 13 y 24 horas después del envío	00
d)	Pasada las 24 horas después del envío	00



6.¿Cómo calificaría Usted la efectividad de la transmisión de la información a los padres de familia o apoderados del colegio?

a)	Alta	80
b)	Media	20
c)	Baja	00

