



UNIVERSIDAD HISPANOAMERICANA
ESCUELA DE INFORMATICA

BACHILLERATO EN INGENIERÍA INFORMÁTICA

DESARROLLO DE SISTEMA DE GESTIÓN DE INFORMACION EN
FUNDACIÓN HEDWIG Y ROBERT SAMUEL

Sustentante:

Hairom Josué Ramirez Solano

Tutor:

Ing. Cristian Paz Campos Agüero

2022

Índice de contenidos

Índice de contenidos	2
Índice de tablas	10
Índice de imágenes.....	12
Declaración Jurada.....	19
Cartas de aprobación del Tutor y Lector.....	20
Dedicatoria.....	23
Agradecimiento.....	24
Abreviaturas	25
Resumen.....	26
Capítulo I: Problema del proyecto	27
1.1 Antecedentes y justificación	28
1.1.1 Marco de Referencia Empresarial y Contextual	28
1.1.2 Justificación del proyecto	33
1.2 Definición del problema	34
1.2.1 Problemática	34
1.2.2 Diagrama causa – efecto	35
1.2.3 Problema General.....	36
1.2.4 Problemas Específicos	36
1.3 Objetivos.....	36
1.3.1 Objetivo general.....	36
1.3.2 Objetivos específicos	37
1.4 Alcances y limitaciones	37

1.4.1	Alcance del proyecto.....	37
1.4.2	Limitaciones del proyecto.....	38
1.5	Cronograma de actividades.....	39
Capitulo II Marco teórico.....		40
2.1	Ingeniería del <i>Software</i>	41
2.2	<i>Software</i>	41
2.3	Áreas de aplicación del <i>software</i>	43
2.4	Proceso de desarrollo del <i>software</i>	44
2.4.1	El modelo en cascada.....	46
2.4.2	Desarrollo incremental.....	47
2.4.3	Patrones de diseño.....	48
2.4.4	Ingeniería de Software orientada a la reutilización.....	53
2.4.5	Arquitectura de Capas y niveles	55
2.4.6	Desarrollo rápido de software.....	58
2.4.7	Scrum	61
2.4.7.1.2	Artefactos y herramientas	62
2.4.8	Programación eXtrema	64
2.5	Etapas del modelo en cascada.....	67
2.5.1	Definición de los requerimientos	67
2.5.2	Requerimientos funcionales.....	68
2.5.3	Requerimientos no funcionales.....	68
2.5.4	Diseño del sistema	69
2.5.5	Implementación.....	70

2.5.6	Pruebas	70
2.5.7	Tipos de pruebas	70
2.5.8	Pruebas de sistema	70
2.5.9	Pruebas de integración	71
2.5.10	Pruebas de entrega	71
2.5.11	Pruebas de validación o aceptación	71
2.5.12	Pruebas de aceptación del usuario	72
2.5.13	Pruebas operativas	72
2.5.14	Pruebas alfa y beta	72
2.5.15	Pruebas de caja blanca	73
2.5.16	Pruebas de caja negra.....	73
2.5.17	Herramientas de Pruebas.....	73
2.4.1.5	Mantenimiento	75
2.6	UML.....	76
2.6.1	Casos de uso.....	76
2.6.2	Diagrama de clases	77
2.6.3	Clases	77
2.6.4	Asociaciones	78
2.6.5	Multiplicidad.....	79
2.6.6	Composición y Agregación.....	79
2.6.7	Diagramas de Objetos	80
2.6.8	Diagramas de estados.....	81
2.6.9	Diagramas de actividades	82

2.6.10	Diagrama de componente	85
Capitulo III: Marco Metodológico.....		86
3.1	Tipo y enfoque de la investigación	87
3.1.1	Tipo de investigación.....	87
3.1.2	Enfoque de la investigación.....	87
3.2	Fuentes de información.....	88
3.2.1	Fuentes de información primaria	88
3.2.2	Entrevista	88
3.2.3	Fuentes secundarias	89
3.2.4	Libro electrónico.....	89
3.2.5	Páginas Web.....	90
3.2.6	Sujetos de información	90
3.3	Técnicas y herramientas de recolección de datos	97
3.3.1	Entrevista no estructurada y entrevista dirigida.....	97
3.3.2	La entrevista en profundidad	98
3.3.3	Observación	98
3.3.4	Observación simple.....	98
3.3.5	Documentos existentes.....	99
3.4	Variables	99
3.5	Diseño de la investigación	100
3.6	Matriz de coherencia.....	100
Capitulo IV: Diagnostico de la situación actual		103
4.1	Diagnostico Administrativo U Operativo	104

4.2	Diagnostico técnico.....	105
4.2.1	Equipo de computación y servidor local.....	105
4.2.2	Lenguaje de programación.....	106
4.2.3	IDE de desarrollo	106
4.2.4	Motor de base de datos	107
4.3	Diagnóstico de percepción	107
4.3.1	Herramientas aplicadas sobre el recurso humano	107
4.3.2	Resultado del instrumento.....	107
	Brechas o conclusiones	111
4.3.3	Situación deseada.....	111
4.3.4	Brecha	112
	Capítulo V: Propuesta del proyecto	113
5.1	Requerimientos	114
5.1.1	Identificación de actores	114
5.1.2	Requerimientos funcionales.....	114
5.1.3	Requerimientos no funcionales.....	118
5.1.4	Diagramas de Casos de uso.....	120
5.1.5	Inicio de sesión	121
5.1.6	Cerrar sesión	122
5.1.7	Agregar usuario.....	123
5.1.8	Actualizar usuario	124
5.1.9	Registro de estudiante	125
5.1.10	Actualizar estudiante.....	127

5.1.11	Eliminar, inhabilitar registro de estudiante	128
5.1.12	Registrar notas	129
5.1.13	Modificar notas	130
5.1.14	Diagrama de actividad	131
5.2	Diagrama relacional	138
5.3	Desarrollo.....	139
5.3.1	Arquitectura de la solución	139
5.3.2	Formulario de ingreso al sistema	143
5.3.3	Formulario Principal	144
5.3.4	Formulario Informe detallado	144
5.3.5	Formulario Carreras	149
5.3.6	Formulario Materias.....	150
5.3.7	Formulario profesores	150
5.3.8	Formulario Control de Asistencia.....	152
5.3.9	Formulario Registro de notas.....	152
5.4	Plan Piloto y Pruebas	153
5.4.1	Usuario	154
5.4.2	Inicio de sesión	157
5.4.3	Asignaturas	158
5.4.4	Carreras	162
5.4.5	Profesores.....	165
5.4.6	Generación	169
5.4.7	Estudiantes	173

5.4.8	Asistencia.....	177
5.4.9	Calificaciones.....	179
5.4.10	Reportes	180
Capítulo VI Conclusiones y recomendaciones del proyecto		184
6.1	Conclusiones.....	185
6.2	Recomendaciones	189
Bibliografía.....		191
Anexos		195
Anexo 1. Datos ampliados		196
Anexo 2. Documento de registro de estudiantes.....		197
Anexo 3. Documento de registro de notas		198
Anexo 4. Minutas de reuniones		199
Anexo 5. Capa Datos – Clase Conexión y Asistencia		201
Anexo 6. Capa Entidades – Clase Asistencia		203
Anexo 7. Capa Negocio – Clase Asistencia.....		204
Anexo 8. Prueba de gestión de usuario.....		207
Anexo 9. Prueba de gestión de inicio de sesión.....		208
Anexo 10. Prueba de gestión de asignaturas.....		209
Anexo 11. Prueba de gestión de carreras		210
Anexo 12. Prueba de gestión de profesores		211
Anexo 13. Prueba de gestión de generación		212
Anexo 14. Prueba de gestión de estudiantes.....		213
Anexo 15. Prueba de gestión de asistencia		214

Anexo 16. Prueba de gestión de Calificaciones	215
Anexo 17. Pruebas de gestión de reportes	216

Índice de tablas

<i>Tabla 1. Cronograma de actividades.....</i>	<i>39</i>
<i>Tabla 2. Comparación de los modelos de proceso de software</i>	<i>54</i>
<i>Tabla 3. Sujetos de información</i>	<i>91</i>
<i>Tabla 4. Definición de variables.....</i>	<i>99</i>
<i>Tabla 5. Matriz de coherencia</i>	<i>101</i>
<i>Tabla 6. Características de equipo de computo</i>	<i>105</i>
<i>Tabla 7. Puestos de trabajo de las personas entrevistadas y los años de trabajar en la Fundación.</i>	<i>108</i>
<i>Tabla 8. Respuesta de la pregunta número 1 de la herramienta de diagnóstico.....</i>	<i>108</i>
<i>Tabla 9. RF01 Inicio de sesión</i>	<i>114</i>
<i>Tabla 10. RF02 Cerrar sesión</i>	<i>115</i>
<i>Tabla 11. RF03 Agregar usuario.....</i>	<i>115</i>
<i>Tabla 12. RF04 Actualizar usuario</i>	<i>115</i>
<i>Tabla 13. RF05 Pantalla principal.....</i>	<i>116</i>
<i>Tabla 14. RF06 Registro de estudiantes</i>	<i>116</i>
<i>Tabla 15. RF07 Actualizar estudiante</i>	<i>117</i>
<i>Tabla 16. RF08 Eliminar, inhabilitar registro estudiante</i>	<i>117</i>
<i>Tabla 17. RF09 Registro de notas</i>	<i>117</i>
<i>Tabla 18. FR10 Modificar notas.....</i>	<i>118</i>
<i>Tabla 19. RF11 Control de asistencia</i>	<i>118</i>

<i>Tabla 20. RNF01 Estilo visual.....</i>	<i>119</i>
<i>Tabla 21. RNF02 Seguridad.....</i>	<i>119</i>
<i>Tabla 22. RNF03 Restricción de contenido.....</i>	<i>119</i>
<i>Tabla 23. RNF04 Tiempo de respuesta.....</i>	<i>119</i>
<i>Tabla 24. RNF05 Confidencialidad.....</i>	<i>120</i>
<i>Tabla 25. Caso de uso inicio de Sesión.....</i>	<i>121</i>
<i>Tabla 26. Caso de uso Cerrar Sesión.....</i>	<i>122</i>
<i>Tabla 27. Caso de uso Agrega usuario.....</i>	<i>123</i>
<i>Tabla 28. Caso de uso Actualizar usuario.....</i>	<i>124</i>
<i>Tabla 29. Caso de uso registro de estudiante.....</i>	<i>126</i>
<i>Tabla 30. Caso de uso actualizar registro estudiante.....</i>	<i>127</i>
<i>Tabla 31. Caso de uso eliminar registro de estudiante.....</i>	<i>128</i>
<i>Tabla 32. Caso de uso registrar notas.....</i>	<i>129</i>
<i>Tabla 33. Caso de uso Modificar notas.....</i>	<i>130</i>
<i>Tabla 34. Diagrama de actividad inicio de sesión.....</i>	<i>133</i>
<i>Tabla 35. Diagrama de actividad registro de estudiantes.....</i>	<i>134</i>
<i>Tabla 36. Diagrama de actividad Actualizar estudiante.....</i>	<i>135</i>

Índice de imágenes

<i>Figura 1. Organigrama de la Fundación Samuel en Costa Rica.....</i>	<i>29</i>
<i>Figura 2. Sedes de la Fundación Samuel.....</i>	<i>30</i>
<i>Figura 3. Árbol de causa y efecto</i>	<i>35</i>
<i>Figura 4. Áreas de aplicación del software</i>	<i>43</i>
<i>Figura 5. Modelo en Cascada.....</i>	<i>46</i>
<i>Figura 6. Desarrollo incremental</i>	<i>47</i>
<i>Figura 7. Proceso de la ingeniería de software orientada a la reutilización.....</i>	<i>53</i>
<i>Figura 8. Nivel de aplicación y nivel de datos.....</i>	<i>55</i>
<i>Figura 9. Arquitectura en dos niveles y dos capas</i>	<i>56</i>
<i>Figura 10. Arquitectura en tres capas</i>	<i>57</i>
<i>Figura 11. Arquitectura de tres capas con la capa Entidades.....</i>	<i>58</i>
<i>Figura 12. Diagrama sencillo del proceso Scrum</i>	<i>63</i>
<i>Figura 13. organización de las practicas del modelo XP</i>	<i>67</i>
<i>Figura 14. Tipos de requerimientos no funcionales.....</i>	<i>69</i>
<i>Figura 15. Diagrama de caso de uso</i>	<i>77</i>
<i>Figura 16. Clase automóvil.....</i>	<i>78</i>
<i>Figura 17. Relación, asociación y dirección entre clases.....</i>	<i>78</i>
<i>Figura 18. Ejemplo de multiplicidad.....</i>	<i>79</i>
<i>Figura 19. Composición y Agregación</i>	<i>80</i>
<i>Figura 20. Diagrama de Objetos</i>	<i>81</i>
<i>Figura 21. Diagrama de estado</i>	<i>81</i>

<i>Figura 22. Simbología del diagrama de actividades</i>	83
<i>Figura 23. Simbología del diagrama de componentes</i>	85
<i>Figura 24. Etapas del desarrollo del proyecto</i>	100
<i>Figura 25. Diagrama de caso de uso de inicio de Sesión</i>	121
<i>Figura 26. Diagrama de caso de uso Cerrar Sesión</i>	122
<i>Figura 27. Diagrama de caso de uso Agregar usuario</i>	123
<i>Figura 28. Diagrama de caso de uso actualizar usuario</i>	124
<i>Figura 29. Diagrama de caso de uso Registro de estudiante</i>	125
<i>Figura 30. Diagrama caso de uso actualizar estudiante</i>	127
<i>Figura 31. Diagrama de caso de uso eliminar registro de estudiante</i>	128
<i>Figura 32. Diagrama caso de uso registrar notas</i>	129
<i>Figura 33. Diagrama de caso de uso Modificar notas</i>	130
<i>Figura 34. Diagrama de actividad inicio de sesión</i>	132
<i>Figura 35. Diagrama de actividad Registro de estudiantes</i>	133
<i>Figura 36. Diagrama de actividad Actualizar estudiante</i>	134
<i>Figura 37. Diagrama de actividad Eliminar registro de estudiante</i>	135
<i>Figura 38. Diagrama de actividad Registrar notas</i>	136
<i>Figura 39. Diagrama de actividad modificar nota</i>	137
<i>Figura 40. Diagrama relacional</i>	138
<i>Figura 41. Arquitectura en capas</i>	139
<i>Figura 42. Formulario Ingreso de usuario</i>	143
<i>Figura 43. Formulario Principal</i>	144
<i>Figura 44. Pestaña Listado del formulario Informe detallado</i>	145

Figura 45. Pestaña Listado Activar, desactivar y borrar registro.....	146
Figura 46. Pestaña Mantenimiento del formulario Informe detallado.....	147
Figura 47. Pestaña Mantenimiento del Formulario Informe detallado.....	148
Figura 48. Formulario Carreras.....	149
Figura 49. Formulario Materias.....	150
Figura 50. Pestaña Listado del formulario Profesores.....	151
Figura 51. Pestaña Mantenimiento del formulario Profesores	151
Figura 52. Formulario Control de asistencias.....	152
Figura 53. Formulario Registro de notas.	153
Figura 54. Prueba de registro de usuario exitoso	155
Figura 55. Prueba de registro de usuarios con datos incompletos o erróneos	155
Figura 56. Actualización de los datos de un usuario.....	156
Figura 57. Prueba de actualización de usuarios con datos incompletos o erróneos	156
Figura 58. Inicio de sesión con éxito	157
Figura 59. Inicio de sesión con error de autenticación.....	158
Figura 60. Prueba de registro de asignaturas exitosa.....	159
Figura 61. Prueba de registro de asignaturas con datos incompletos o erróneos	159
Figura 62. Actualización de las asignaturas exitoso	160
Figura 63. Eliminación de asignaturas	160
Figura 64. Deshabilitar asignaturas.....	161
Figura 65. Habilitar asignaturas.....	161
Figura 66. Pruebas de registro de carreras.....	162
Figura 67. Prueba de registro de carreras con datos incompletos o erróneos	163

Figura 68. Actualización de las asignaturas.....	163
Figura 69. Eliminación de asignaturas	164
Figura 70. Deshabilitar asignaturas.....	164
Figura 71. Habilitar asignaturas	165
Figura 72. Prueba de registro de profesores	166
Figura 73. Prueba de registro de profesores con datos incompletos o erróneos	167
Figura 74. Actualización de los datos de un profesor.....	167
Figura 75. Eliminación de profesores.....	168
Figura 76. Habilitar profesor.....	168
Figura 77. Deshabilitar profesor	169
Figura 78. Prueba de registro de generación.....	170
Figura 79. Prueba de registro con datos incompletos o erróneos.....	171
Figura 80. Eliminación de registros	171
Figura 81. Habilitar registros de generación	172
Figura 82. Deshabilitar registro de generación.....	172
Figura 83. Prueba de registro de estudiantes	174
Figura 84. Prueba de registro de estudiantes con datos incompletos o erróneos.....	174
Figura 85. Actualización de los registros de estudiantes.....	175
Figura 86. Eliminación de los registros de estudiantes.....	175
Figura 87. Habilitar registros de estudiantes	176
Figura 88. Deshabilitar registro de estudiantes	176
Figura 89. Prueba de registro de asistencia.....	177
Figura 90. Prueba de registro de asistencia con datos incompletos o erróneos.....	178

<i>Figura 91. Actualización de los registros de asistencia de estudiantes.....</i>	<i>178</i>
<i>Figura 92. Prueba de registro de calificaciones.....</i>	<i>179</i>
<i>Figura 93. Prueba de registro de calificaciones con datos incompletos o erróneos.....</i>	<i>180</i>
<i>Figura 94. Reporte de información básica de los estudiantes</i>	<i>181</i>
<i>Figura 95. Reporte según genero.....</i>	<i>181</i>
<i>Figura 96. Reporte según la escolaridad.....</i>	<i>182</i>
<i>Figura 97. Reporte de Calificaciones por especialidad, materia y generación</i>	<i>182</i>
<i>Figura 98. Reporte de notas entre fechas y por estudiante.....</i>	<i>183</i>

Documento de aceptación del sistema

Documento de Aceptación del Sistema

Proyecto: Sistema de gestión Académica

Cliente: La Hedwig y Robert Samuel

Fecha: 23/06/2023

A través de este documento, se confirma que el sistema desarrollado para La Hedwig y Robert Samuel ha sido evaluado y se considera aceptado para su implementación y uso en producción.

Detalles del Sistema:

- Nombre del Sistema: Sistema de gestión Académica

- Versión del Sistema: Versión 1.0

Requisitos y Funcionalidades:

El sistema cumple con los requisitos y funcionalidades acordadas previamente con el Cliente.

Aceptación por parte de la Empresa donde se Aplicó:

La Fundación Hedwig y Robert Samuel acepta formalmente el proyecto y el sistema desarrollado. Se ha revisado y validado el sistema de acuerdo con las especificaciones acordadas.

Responsabilidades Futuras:

El Cliente asume la responsabilidad del mantenimiento, actualizaciones y soporte continuo del sistema.

Firma de Aceptación:

Nombre: Karla Arguedas Jiménez

Cargo: Directora Académica

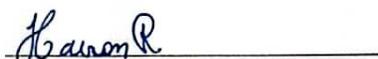
Fecha: 23/06/23



Declaración Jurada

DECLARACIÓN JURADA

Yo Hairom Josué Ramirez Solano, mayor de edad, portador de la cédula de identidad número 111950088 egresado de la carrera de Ingeniería informática de la Universidad Hispanoamericana, hago constar por medio de éste acto y debidamente apercibido y entendido de las penas y consecuencias con las que se castiga en el Código Penal el delito de perjurio, ante quienes se constituyen en el Tribunal Examinador de mi trabajo de tesis para optar por el título de Bachillerato en Ingeniería en sistemas de información juro solemnemente que mi trabajo de investigación titulado: Desarrollo de sistema de gestión de información en Fundación Hedwig y Robert Samuel es una obra original que ha respetado todo lo preceptuado por las Leyes Penales, así como la Ley de Derecho de Autor y Derecho Conexos número 6683 del 14 de octubre de 1982 y sus reformas, publicada en la Gaceta número 226 del 25 de noviembre de 1982; incluyendo el numeral 70 de dicha ley que advierte; artículo 70. Es permitido citar a un autor, transcribiendo los pasajes pertinentes siempre que éstos no sean tantos y seguidos, que puedan considerarse como una producción simulada y sustancial, que redunde en perjuicio del autor de la obra original. Asimismo, quedo advertido que la Universidad se reserva el derecho de protocolizar este documento ante Notario Público. en fe de lo anterior, firmo en la ciudad de San José, a los 22 días del mes de julio del año dos mil veintitres.



Firma del estudiante

Cédula 111950088

Cartas de aprobación del Tutor y Lector

CARTA DEL TUTOR

San José, 5 de mayo del 2023

Señora:
María Isabel Losilla Barrientos
Directora de Carrera
Ingeniería Informática
Universidad Hispanoamericana

Estimado señor:

El estudiante Hairom Josué Ramírez Solano, cédula de identidad número 1-1195-0088, me ha presentado, para efectos de revisión y aprobación, el trabajo de investigación denominado **Desarrollo de Sistema de Gestión de Información en Fundación Hewing y Robert Samuel**, el cual ha elaborado para optar por el grado académico de Bachiller

En mi calidad de tutor, he verificado que se han hecho las correcciones indicadas durante el proceso de tutoría y he evaluado los aspectos relativos a la elaboración del problema, objetivos, justificación; antecedentes, marco teórico, marco metodológico, tabulación, análisis de datos; conclusiones y recomendaciones.

De los resultados obtenidos por el postulante, se obtiene la siguiente calificación:

a)	ORIGINAL DEL TEMA	10%	10
b)	CUMPLIMIENTO DE ENTREGA DE AVANCES	20%	16
c)	COHERENCIA ENTRE LOS OBJETIVOS, LOS INSTRUMENTOS APLICADOS Y LOS RESULTADOS DE LA INVESTIGACION	30%	30
d)	RELEVANCIA DE LAS CONCLUSIONES Y RECOMENDACIONES	20%	17
e)	CALIDAD, DETALLE DEL MARCO TEORICO	20%	20
	TOTAL		95

En virtud de la calificación obtenida, se avala el traslado al proceso de lectura.

Atentamente,

Ing. Cristian Campos Agüero
Cédula residencia 160400100307
Carné CPIC 3568

CRISTIAN PAZ
CAMPOS
AGUERO
(FIRMA)

Firmado digitalmente
 por CRISTIAN PAZ
 CAMPOS AGUERO
 (FIRMA)
 Fecha: 2023.05.05
 20:29:39 -06'00'

CARTA DE LECTOR

San José,

Universidad Hispanoamericana
Sede Llorente
Carrera de Informática

Estimado señor

El estudiante Hairom Josué Ramirez Solano, cédula de identidad 111950088, me ha presentado para efectos de revisión y aprobación, el trabajo de investigación denominado "Desarrollo De Sistema De Gestión De Informacion En Fundación Hedwig Y Robert Samuel".

He revisado y he hecho las observaciones relativas al contenido analizado, particularmente lo relativo a la coherencia entre el marco teórico y análisis de datos, la consistencia de los datos recopilados y la coherencia entre éstos y las conclusiones; asimismo, la aplicabilidad y originalidad de las recomendaciones, en términos de aporte de la investigación.

Por consiguiente, este trabajo cuenta con mi aval para ser presentado en la defensa pública.

Atte.

**Randall
Vargas
Villalobos** Firmado
digitalmente por
Randall Vargas
Villalobos
Fecha: 2023.07.14
21:18:27 -06'00'

Firma
Randall Vargas Villalobos
Cédula: 1-1140-0113

Carta de autorización de los lectores para la consulta

UNIVERSIDAD HISPANOAMERICANA
CENTRO DE INFORMACION TECNOLOGICO (CENIT)
CARTA DE AUTORIZACIÓN DE LOS AUTORES PARA LA CONSULTA, LA
REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA
DE LOS TRABAJOS FINALES DE GRADUACION

San José, Tibás, Llorente

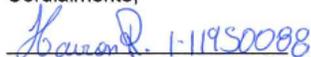
Señores:
Universidad Hispanoamericana
Centro de Información Tecnológico (CENIT)

Estimados Señores:

El suscrito (a) Hairom Josué Ramírez Solano con número de identificación 1 1195 0088 autor (a) del trabajo de graduación titulado Desarrollo de Sistema de Gestión de Información en Fundación Hedwig y Robert Samuel presentado y aprobado en el año 2023 como requisito para optar por el título de Bachiller en Ingeniería Informática; SI autorizo al Centro de Información Tecnológico (CENIT) para que con fines académicos, muestre a la comunidad universitaria la producción intelectual contenida en este documento.

De conformidad con lo establecido en la Ley sobre Derechos de Autor y Derechos Conexos N° 6683, Asamblea Legislativa de la República de Costa Rica.

Cordialmente,


Firma y Documento de Identidad

Dedicatoria

Quiero dedicar este proyecto a mis padres, Marielos Solano Acuña y Miguel Ramirez Solano, quienes me han formado como persona y a quienes debo muchos de mis logros incluyendo este. Agradezco cada uno de sus sacrificios los cuales hacían sin esperar ninguna retribución. Ellos me dieron el mejor regalo que se le puede dar a alguien, enseñarme entre muchas cosas la importancia del estudio, me enseñaron a ser responsable y a luchar por mis metas. Por esto y por más, esta dedicatoria en símbolo de agradecimiento a mis padres.

A mi hija que es mi gran motivación, quien me ayuda a liberar mi mente de obstáculos y me impulsa a superar las adversidades que se presentan a diario. Ella no lo entiende muy bien, pero me da la fuerza para continuar y superarme con la esperanza de darle un mejor futuro.

A mi familia en general, por su apoyo incondicional en los buenos y malos momentos. Incluyo a mi hermano que nos dejó cuando yo era un niño, el nunca morirá en nuestros corazones por sus acciones en vida y por las cuales le recordamos con amor. Espero que podamos reencontrarnos algún día y contarte de mis logros.

A mi esposa Karina Barboza Navarro, quien ha estado conmigo en los momentos más complicados. Este trabajo fue complejo muchas veces, pero estuvo motivándome hasta donde su alcance se lo permitía.

Agradecimiento

Agradezco, inicialmente a Dios por dejarme llegar a este punto de mi carrera y permitirme desarrollar un proyecto que pone a prueba los conocimientos adquiridos en la universidad, la capacidad de investigar y la humildad para preguntar a quienes saben más. En Segundo lugar, agradezco a mi tutor Cristian Paz Campos Agüero quien me ha guiado durante todo este proceso supervisando, asesorando y contribuyendo con su experiencia siempre de forma amable y oportuna.

También agradezco en gran manera a la Fundación Samuel por permitirme desarrollar este proyecto en la sede de Costa Rica, a su director ejecutivo Wilber Pérez Solano, a su directora Académica Karla Arguedas Jiménez, y a la asistente de dirección académica Norma Hamlett Nicaragua, por apoyarme facilitando información necesaria para recolección de los requerimientos y demás documentación.

Finalmente quiero agradecer a mis compañeras de trabajo, Marjorie Dávila Gómez, Pamela Mora Vizcaíno y Krisna Jiménez Echeverría, por sus consejos y su apoyo incondicional que fueron muchas veces de gran ayuda en este proceso.

Abreviaturas

CRUD : Acronimo de “Create, Read, Update and Delete”.

CU: Caso de uso.

DA: Diagrama de actividad.

IBM: International Business Machines.

IDE: Integrated Development Environment.

MVC: Modelo Vistas Controlador.

PDM: Program Development Monitor.

RF: Requerimiento funcional.

RNF: Requerimiento no funcional.

SQL: Structured Query Language.

UML: Unified Modeling Language.

XP: Extreme Programming.

Resumen

Este proyecto presenta el desarrollo y la implementación de un *software* para la gestión de la información de los estudiantes de la Fundación Hedwig y Robert Samuel. Dicha institución carece de un sistema que les permita registrar la información de sus estudiantes, almacenarla de forma segura y obtener informes de forma rápida y simple.

La finalidad de este proyecto es proporcionar un sistema informático que agilice los procesos de registro y seguimiento de los estudiantes logrando centralizar la información referente a estos. Durante el proceso del desarrollo de la solución, se tendrá contacto directo con la institución con el fin de obtener un producto lo más cercano a cubrir sus necesidades actuales.

Capítulo I: Problema del proyecto

1.1 Antecedentes y justificación

1.1.1 Marco de Referencia Empresarial y Contextual

La Fundación Hedwig Y Robert Samuel tiene más de 25 años de experiencia en la formación técnica a nivel internacional. En Costa Rica se han graduado 22 generaciones de técnicos. Los estudiantes reciben los medios necesarios para obtener una formación técnica profesional calificada, dándoles las posibilidades de optar por una mejor calidad de vida, mejores oportunidades laborales, una autoestima sana, con nuevas perspectivas y esperanza en su futuro.

A través de la formación Samuel, los estudiantes aprenden disciplina, orden, respeto y una actitud proactiva en el desarrollo de sus labores.

Give youth a chance (Una oportunidad para la juventud). Este es el lema de la fundación Samuel. (Samuel F. H., 2020)

Misión:

“Proporcionar educación y formación profesional calificadas en nuestros centros y proyectos, brindando así a un número de personas cada vez mayor las herramientas necesarias para abandonar el ciclo de la pobreza y, de esta manera también mejorar las perspectivas futuras de las siguientes generaciones”

Visión:

Un mundo donde las oportunidades de educación y perspectivas futuras sean justas para todos”

Objetivos:

Del gran grupo de necesidades sociales, promovemos aquellos jóvenes, que, por su inteligencia y personalidad, estén en condiciones de comenzar y poder llevar a cabo una vida digna y con autodeterminación, gracias al apoyo puesto a su disposición por la Fundación Samuel como base. (Rex, 2010)

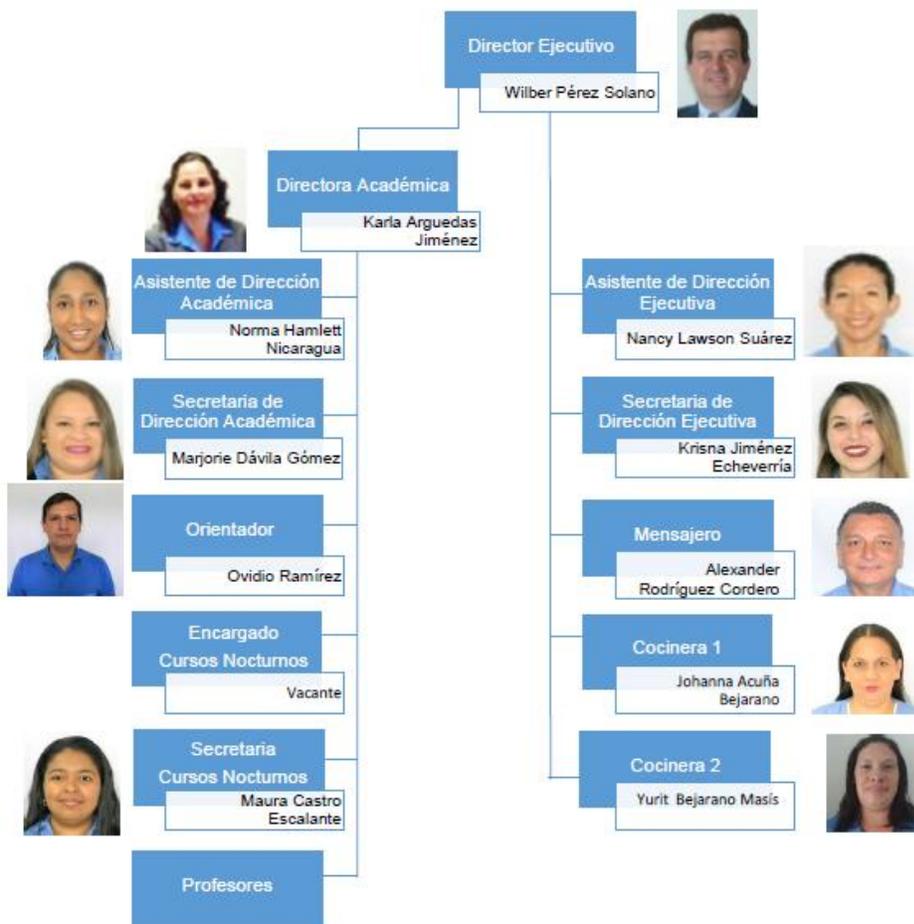
Organización:

La junta directiva

De acuerdo con los estatutos, la junta directiva es el único órgano oficial de la fundación Samuel y, por lo tanto, el principal órgano de decisión. Para los directores ejecutivos de cada sede de país es la Junta Directiva la máxima autoridad jerárquica.

Presidente: Martin Barth, La junta directiva: Karen Reuter, Michael Barth.

Figura 1. Organigrama de la Fundación Samuel en Costa Rica



Dirección ejecutiva de las sedes

Cada sede de país de la fundación Samuel es dirigida por un/a Director/a, quienes dependen jerárquica y directamente de la Junta Directiva en Alemania.

La sede central de la Fundación se encuentra en Düsseldorf Alemania. Actualmente cuenta con sedes y centros de formación técnica en San José (Costa Rica), Managua (Nicaragua), y Nueva Delhi (India). Además, son socios fundadores de una institución especializada en niños víctimas de violencia: KiD en Düsseldorf.

La **¡Error! No se encuentra el origen de la referencia.** muestra los países en los que está presente la Fundación Samuel y la cantidad de empleados en cada sede.

Figura 2. Sedes de la Fundación Samuel



Nota. Sedes de la Fundación Hedwig y cantidad de empleados. Samuel. (2022).

[Imagen]. (<https://www.samuel.de/es>)

Negocio al que se dedica

En sus centros, la fundación ofrece becas de formación técnica a adolescentes y jóvenes económicamente desfavorecidos. Estas becas tienen un carácter profesional y holístico, se desarrollan a tiempo completo y están basadas en el modelo del sistema dual alemán.

Todos los programas de formación técnica de la fundación están diseñados para una duración de 15 o 19 meses a tiempo completo, incluyendo un periodo de prácticas laborales de tres meses. Durante este tiempo, los estudiantes adquieren todas las competencias necesarias para desenvolverse en el mercado laboral actual, moldear con éxito su vida profesional futura y reaccionar con flexibilidad ante los próximos desarrollos. Al impartir las denominadas "competencias clave para el aprendizaje permanente", la fundación cumple con las recomendaciones del marco de referencia de la Unión Europea y posibilita que los estudiantes participen plenamente en la vida social y gestionen con éxito su transición hacia el mercado laboral.

De este modo, la formación que se imparte en estos centros no sólo conduce a una calificación técnica. Según Samuel: “Este enfoque tiene un efecto más sostenible. Mediante la colocación de los graduados en empleos calificados y las oportunidades materiales que esto conlleva, la fundación abre perspectivas completamente nuevas a estos jóvenes y les permite empezar una nueva vida” (Samuel , 2022)

Historia de la organización

El 13 de junio de 1932, fue creada oficialmente la Fundación Hedwig y Rober Samuel. Ese día, el ministerio prusiano estatal autorizó los documentos y estatutos de la Fundación, cumpliéndose así el último paso formal necesario.

La primera junta directiva todavía estaba formada por los testamentarios, tres comerciantes judíos, amigos de Robert Samuel. A finales de los años 30, ellos renunciaron al cargo, cediendo a la presión nacionalista, esto protegió solo en parte a la Fundación de la codicia de las organizaciones nazis.

Al crearse la Fundación, se adjudicó comercial y de oficinas de la Königsalle 14 en Duesseldorf como capital de la sociedad. La cofundadora Hedwig Samuel recibió los ingresos originales del mismo. En 1976 la Fundación tuvo los recursos necesarios para poner en práctica su misión. Sin embargo, el tiempo y sobre todo los años de guerra, dejaron sus marcas en la casa de Königsalle 14. Por este motivo primero se dio prioridad a la reparación completa del edificio para asegurar su manutención. Pero también en esa fase, la Fundación fomentó con ayuda financiera proyectos sociales, principalmente en la ciudad de Duesseldorf. En la actualidad, la fundación continúa trabajando activamente en Duesseldorf, por ejemplo, en la institución para niños kiD gGmbH, de la cual la Fundación Samuel es socia.

A partir de 1984, con la creación de una infraestructura propia, la Fundación comenzó con su trabajo social en el extranjero. En los inicios solo se fomentaron proyectos de otras organizaciones, con el paso del tiempo, la realización de proyectos propios en países en vía de desarrollo fue tomando protagonismo.

En un lapso de tres a cuatro años, se fundaron las diferentes representaciones de la Fundación en el extranjero: 1987 en Costa Rica, 1989 en Nicaragua, 1991 en Nigeria y 1993 en la india. En 1999 la Fundación dio por terminada su labor en Nigeria, debido a las dificultades cada vez más grandes. Al principio de 2002 se ha instalado una oficina de la Fundación en Tegucigalpa, Honduras, para que sirva de arranque para crear un nuevo centro de capacitación. (Rex, 2010)

1.1.2 Justificación del proyecto

Obtener información de una forma rápida y ordenada, ayuda a las empresas a tomar decisiones en un tiempo menor y de forma más acertada. De igual forma se debe buscar la seguridad e integridad de la información, esto para evitar que la información se pierda, se duplique o simplemente no sea de utilidad.

Los sistemas de información están cambiando en la actualidad la forma en que operan las organizaciones. Mediante su uso se obtiene grandes mejoras, ya que automatizan los procesos operativos que se pueden llevar a cabo en toda la empresa, proporcionan información de apoyo al proceso de tomas de decisiones y facilitan el logro de ventajas competitivas a través de su implantación dentro de la organización. (Hamidian Fernández, 2015)

La implementación de un sistema académico permitirá a la dirección académica llevar un control de las ausencias de los estudiantes, anotaciones según su comportamiento, comentarios de los profesores, y datos personales como lugar de residencia, teléfono, edad, entre otros, se agilizarían las búsquedas de la información de los estudiantes.

Según ha indicado la directora académica de la Fundación. “El uso de un sistema de informático podría simplificar el registro y la búsqueda de la información de los estudiantes” (K. Arguedas, comunicación personal, 14 de octubre de 2022)

Uno módulo del sistema permitirá minimizar los errores de los trámites de cobros, entre los que se puede mencionar: duplicidad en los cobros, errores en los montos o las pérdidas de facturas. Las asistentes de la dirección académica que son las encargadas de estos cobros podrán llevar un mejor control de la cancelación de los montos establecidos y generar un recibo a los estudiantes que les servirá como un respaldo y garantía de los pagos realizados.

Con respecto a la viabilidad del proyecto, se contempla el uso de los recursos tecnológicos con los que cuenta la Fundación Samuel, las cuales son computadoras, internet y el servidor local. También se cuenta con la ayuda de la persona encargada del mantenimiento de las redes internas y del servidor, por lo cual se asume que el proyecto es factible en el tiempo establecido.

1.2 Definición del problema

1.2.1 Problemática

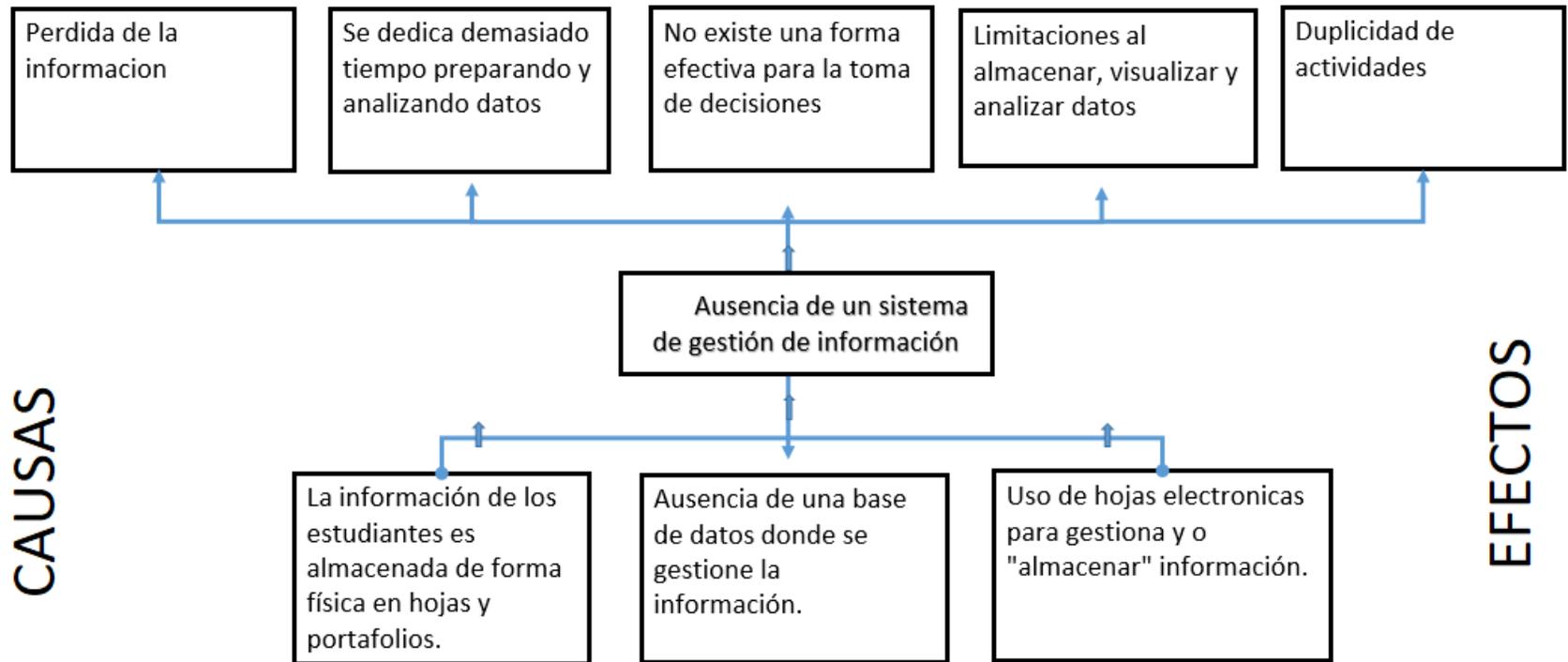
Para la Fundación Samuel es de gran importancia el contar con un sistema académico que permita registrar y consultar la información de los estudiantes de forma eficaz, ya que son la razón de ser de la organización. La fundación ha buscado la forma de adquirir un sistema adecuado que permita cubrir sus necesidades académicas y que, además se adapte a su modelo de trabajo, sin embargo, se les dificulta cubrir el costo de este. (K. Arguedas, comunicación personal, 14 de octubre de 2022)

La manera en que se registra la información de los estudiantes es de forma digital en hojas de cálculo, lo que es un método anticuado y dificulta la obtención de información al área de la dirección académica, quienes la utilizan para sus distintas actividades diarias. El desarrollo de un sistema académico permitirá gestionar información desde una base de datos asegurando la integridad, confiabilidad e integridad de los registros además de agilizar la toma de decisiones gracias a que se podrá reunir la información y generar los reportes.

Ante esto la fundación Samuel busca desarrollar un *software* a la medida que permita automatizar ciertos procesos y facilite el trabajo de la dirección académica y los docentes, con tecnología actual y que sea capaz de adaptarse e integrar otros módulos en el futuro

1.2.2 Diagrama causa – efecto

Figura 3. Árbol de causa y efecto



Fuente: Creación propia

1.2.3 Problema General

¿Desarrollar un sistema informático a la medida de las necesidades de la Fundación Hedwig y Robert Samuel permitirá la gestión de la información de los estudiantes y facilitar las labores del área académica?

1.2.4 Problemas Específicos

1. ¿Cómo eliminar el uso de papel para registrar el comportamiento y desempeño de los estudiantes, evitando almacenar documentos físicos?

2. ¿Se puede asegurar usando un control de acceso de usuarios, que los registros y las calificaciones de los estudiantes son confiables y no han sido modificados por terceros?

3. ¿La implementación de una base de datos permitirá una mejor visualización de la información existente?

1.3 Objetivos

1.3.1 Objetivo general

Desarrollar e implementar un sistema informático de gestión académica para la Fundación Samuel que permita registrar la información de los estudiantes, notas y cuentas pendientes, mejorando la eficiencia y precisión en la gestión académica y facilitando la toma de decisiones en la organización.

1.3.2 *Objetivos específicos*

- Identificar los requerimientos específicos para el diseño y desarrollo de un sistema de gestión de datos que satisfaga las necesidades de la Fundación Hedwig y Robert Samuel, asegurando la eficiencia, calidad y seguridad del Sistema
- Diseñar un sistema académico de gestión de datos que permita a la Fundación Hedwig y Robert Samuel gestionar y mantener la información de los estudiantes, docentes, y programas académicos de manera eficiente y precisa.
- Desarrollar cada uno de los módulos de la aplicación de gestión académica, con base en los requerimientos definidos por la Fundación Hedwig y Robert Samuel, para garantizar su funcionalidad y eficiencia en el proceso de gestión académica.
- Ejecutar un plan piloto con una muestra representativa de los procesos y procedimientos de la Fundación Hedwig y Robert Samuel, para evaluar la viabilidad del *software* de gestión académica y asegurar que se ajusta a las necesidades de la organización.

1.4 Alcances y limitaciones

1.4.1 *Alcance del proyecto*

Desarrollo de un sistema académico para la Fundación Samuel tomando en cuenta las etapas de planificación, diseño y la implementación del proyecto.

El sistema académico estará listo para ser entregado cuando cuente con los módulos necesarios para el ingreso de usuarios, gestión de datos, mantenimientos, y la base de datos. Los módulos del sistema deben ser eficiente y cumplir con los requerimientos de la Fundación Samuel. El diseño del sistema debe basarse en las necesidades de los usuarios que lo utilizaran.

Las restricciones al proyecto están sometidas a los formatos que utiliza la Fundación en cada una de sus sedes ya que los reportes que entregue el sistema deben de cumplir con las características de los documentos que ellos utilizan.

1.4.2 Limitaciones del proyecto

- La disponibilidad de algunas personas necesarias para el análisis de requerimientos y accesos a equipos como computadoras o el servidor, ya que no siempre están en las instalaciones de la organización.
- Se debe de seguir el formato que utiliza la organización. Todos los reportes deben de acoplarse de la mejor forma al formato establecido por la sede principal.
- Al nivel de diseño, debo limitarme a utilizar las licencias con las que cuenta la organización, como por ejemplo SQL SERVER.

1.5 Cronograma de actividades

Tabla 1. Cronograma de actividades

Fases del proyecto	Descripción de la actividad	Octubre				Noviembre				Diciembre				Enero				Febrero				Marzo				Abril				Fecha de inicio	Fecha de finalización
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4		
Corrección de formato del trabajo escrito				■	■																							07/10/2022	18/10/2022		
Mejoras del Capítulo I. Inicio del capítulo II				■	■																							22/10/2022	06/11/2022		
Mejoras del Capítulo II. Inicio del capítulo III					■	■	■																					09/11/2022	20/11/2022		
Mejoras del Capítulo III							■	■	■																			24/11/2022	06/12/2022		
Mejoras del trabajo en general. Inicio del capítulo IV									■	■																		10/12/2022	18/12/2022		
Mejoras del capítulo IV										■	■	■	■															21/12/2022	16/01/2023		
Avanzar en el capítulo V														■	■	■	■											19/01/2023	06/02/2023		
Mejoras del capítulo V. Inicio del desarrollo del sistema															■	■												09/02/2023	19/02/2023		
Desarrollo del sistema																■	■	■	■	■	■							19/02/2023	23/03/2023		
Desarrollo del sistema. Mejoras en el trabajo escrito																						■	■	■				25/03/2023	09/04/2023		

Fuente: Creación propia.

Capítulo II Marco teórico

En esta unidad se abordarán términos y conceptos relacionados al proyecto, para dar a entender con más claridad términos informáticos usados en este documento.

2.1 Ingeniería del *Software*

La ingeniería de *software* es una disciplina de la ingeniería cuya meta es el desarrollo costeable de sistema de *software*. Éste es abstracto e intangible. No está restringido por materiales, o gobernado por leyes físicas o por procesos de manufactura. De alguna forma esto simplifica la ingeniería del *software* ya que no existen limitaciones físicas del potencial del *software*. (Sommerville, 2005, p. 6)

2.2 *Software*

Muchas personas asocian el término *software* con los programas de computadora. (Sommerville, 2005), en su texto Ingeniería del *Software* declara que “el *software* no son solo los programas, sino todos los documentos asociados y la configuración de datos que se necesitan para hacer que estos programas operen de manera correcta”. (p.5). Actualmente, hay siete grandes categorías de *software* de computadoras: de sistemas, de aplicación, de ingeniería y ciencias, incrustado, de línea de productos, aplicaciones web y de inteligencia artificial. (Pressman, 2010, p. 6)

En su libro (Pressman, 2010, p. 6-7.) describe 7 grandes categorías de *software* que se describen a continuación:

Software de sistemas: conjunto de programas escritos para dar servicio a otros programas. Determinado *software* de sistemas (por ejemplo, compiladores, editores y herramientas para administrar archivos) procesa estructuras de información complejas pero deterministas. Otras aplicaciones de sistemas (por ejemplo, componentes de sistemas operativos, manejadores, *software* de redes, procesadores de

telecomunicaciones) procesan sobre todo datos indeterminados. En cualquier caso, el área de *software* de sistemas se caracteriza por: gran interacción con el *hardware* de la computadora, uso intensivo por parte de usuarios múltiples, operación concurrente que requiere la secuenciación, recursos compartidos y administración de un proceso sofisticado, estructuras complejas de datos e interfaces externas múltiples.

Software de aplicación: Programas aislados que resuelven una necesidad específica de negocios. Las aplicaciones en esta área procesan datos comerciales o técnicos en una forma que facilita las operaciones de negocios o la toma de decisiones administrativas o técnicas. Además de las aplicaciones convencionales de procesamiento de datos, el *software* de aplicación se usa para controlar funciones de negocios en tiempo real (por ejemplo, procesamiento de transacciones en punto de venta, control de procesos de manufactura en tiempo real).

Software de ingeniería y ciencias: se ha caracterizado por algoritmos “devoradores de números”. Las aplicaciones van de la astronomía a la vulcanología, del análisis de tensiones en automóviles a la dinámica orbital del transbordador espacial, y de la biología molecular a la manufactura automatizada. Sin embargo, las aplicaciones modernas dentro del área de la ingeniería y las ciencias están abandonando los algoritmos numéricos convencionales. El diseño asistido por computadora, la simulación de sistemas y otras aplicaciones interactivas, han comenzado a hacerse en tiempo real e incluso han tomado características del *software* de sistemas.

Software incrustado: reside dentro de un producto o sistema y se usa para implementar y controlar características y funciones para el usuario final y para el sistema en sí. El *software* incrustado ejecuta funciones limitadas y particulares (por ejemplo, control del tablero de un horno de microondas) o provee una capacidad significativa de funcionamiento y control.

Software de línea de productos: es diseñado para proporcionar una capacidad específica para uso de muchos consumidores diferentes. El *software* de línea de productos se centra en algún mercado limitado y particular (por ejemplo, control del inventario de productos) o se dirige a mercados masivos de consumidores (procesamiento de textos, hojas de cálculo, gráficas por computadora, multimedios, entretenimiento, administración de base de datos y aplicaciones para finanzas personales o de negocios).

Aplicaciones web: llamadas “*webapps*”, esta categoría de *software* centrado en redes agrupa una amplia gama de aplicaciones. En su forma más sencilla, las *webapps* son poco más que un conjunto de archivos de hipertexto vinculados que presentan información con uso de texto y gráficas limitadas. Sin embargo, desde que surgió Web 2.0, las *webapps* están evolucionando hacia ambientes de cómputo sofisticados que no sólo proveen características aisladas, funciones de cómputo y contenido para el usuario final, sino que también están integradas con bases de datos corporativas y aplicaciones de negocios.

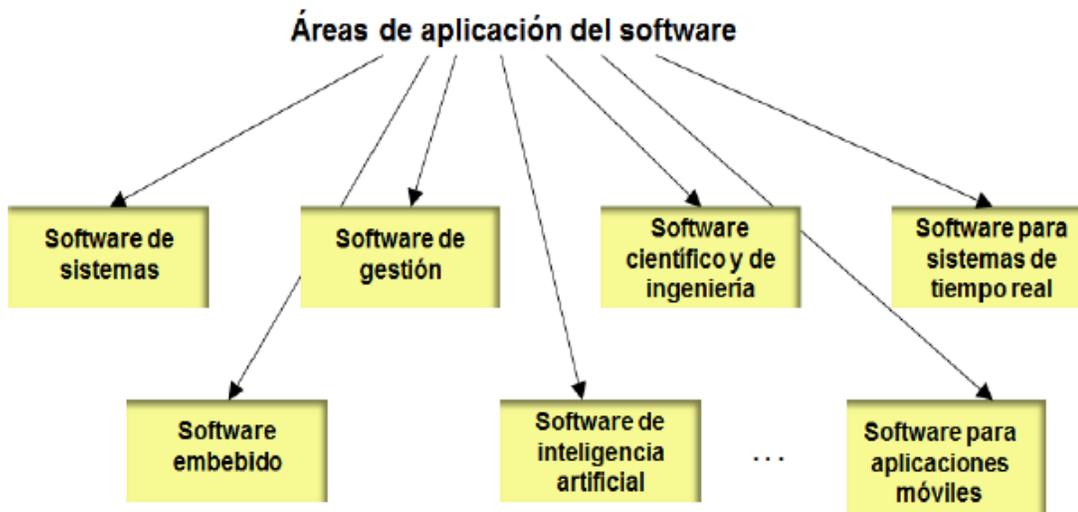
Software de inteligencia artificial: hace uso de algoritmos no numéricos para resolver problemas complejos que no son fáciles de tratar computacionalmente o con el análisis directo. Las aplicaciones en esta área incluyen robótica, sistemas expertos, reconocimiento de patrones (imagen y voz), redes neurales artificiales, demostración de teoremas y juegos.

2.3 Áreas de aplicación del *software*

El *software* es el elemento de un sistema basado en computadora, a diferencia del *hardware* que es el elemento físico. El *software* se crea a través de una metodología de desarrollo con fases y actividades claramente definidas, y no se fabrica en un sentido clásico como la mayoría de los productos. Hoy en día, el desarrollo de *software* implica el ensamblaje, reusabilidad y extensibilidad de componentes existentes. (Gómez, Cervantes, & González, 2019, p. 19)

Los componentes del *software* son los programas ejecutables y los datos existentes en una computadora. En la figura 4 se ilustran algunos ejemplos de las áreas de aplicación del *software*.

Figura 4. Áreas de aplicación del *software*



Fuente: Fundamentos de la ingeniería del *software*

2.4 Proceso de desarrollo del *software*

Para lograr un *software* de calidad, se buscan las acciones y tareas que permitan que el trabajo se realice de forma ordenada siguiendo los procesos que lleven a una entrega exitosa. Estos pasos, tareas y acciones suman lo que se conoce como proceso.

Un proceso de desarrollo de *software* es un proceso de actividades relacionadas que conducen a la elaboración de un producto de *software*. Estas actividades pueden incluir el desarrollo de *software* desde cero en un lenguaje de programación estándar como Java o C. Sin embargo, las aplicaciones de negocios no se desarrollan precisamente de esta forma.

(Sommerville, 2005, p. 28)

(Pressman, 2010, p. 13) explica de la siguiente forma las 5 actividades esenciales del proceso del *software*:

Comunicación. Antes de que comience cualquier trabajo técnico, tiene importancia crítica comunicarse y colaborar con el cliente (y con otros participantes). Se busca entender los objetivos de los participantes respecto del proyecto, y reunir los requerimientos que ayuden a definir las características y funciones del *software*.

Planeación. Cualquier viaje complicado se simplifica si existe un mapa. Un proyecto de *software* es un viaje difícil, y la actividad de planeación crea un “mapa” que guía al equipo mientras viaja. El mapa —llamado *plan del proyecto de software*— define el trabajo de ingeniería de *software* al describir las tareas técnicas por realizar, los riesgos probables, los recursos que se requieren, los productos del trabajo que se obtendrán y una programación de las actividades.

Modelado. Ya sea usted diseñador de paisaje, constructor de puentes, ingeniero aeronáutico, carpintero o arquitecto, a diario trabaja con modelos. Crea un bosquejo del objeto por hacer a fin de entender el panorama general —cómo se verá arquitectónicamente, cómo ajustan entre sí las partes constituyentes y muchas características más—. Si se requiere, refina el bosquejo con más y más detalles en un esfuerzo por comprender mejor el problema y cómo resolverlo. Un ingeniero de *software* hace lo mismo al crear modelos a fin de entender mejor los requerimientos del *software* y el diseño que los satisfará.

Construcción. Esta actividad combina la generación de código (ya sea manual o automatizada) y las pruebas que se requieren para descubrir errores en éste.

Despliegue. El *software* (como entidad completa o como un incremento parcialmente terminado) se entrega al consumidor que lo evalúa y que le da retroalimentación, misma que se basa en dicha evaluación.

Todos los modelos del proceso del *software* pueden incluir las actividades generales anteriormente mencionadas, pero podemos encontrar algunos modelos que definen alguna etapa de forma diferente o le dan un énfasis distinto. A continuación, se resumen algunos de los modelos del desarrollo del *software*.

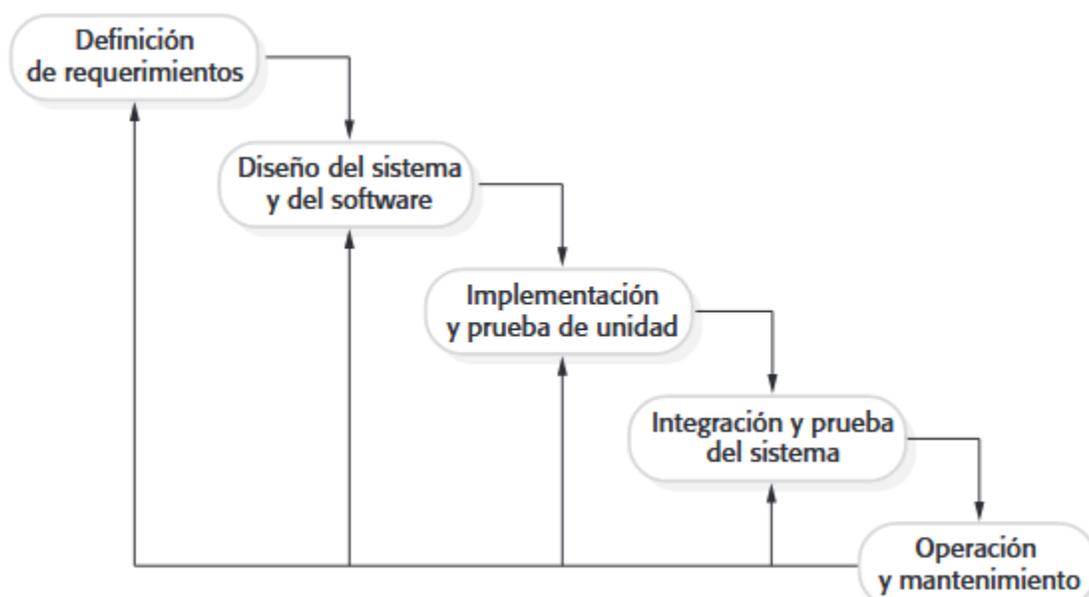
2.4.1 El modelo en cascada

Este modelo también se conoce como “ciclo de vida del *software*. El modelo en cascada es un ejemplo de un proceso dirigido por un plan; en principio se plantean y programan todas las actividades para luego trabajar en ellas.

Para iniciar una fase, debe aprobarse la anterior a esta. Ninguna fase inicia sino hasta que termine la fase previa, además, cada fase debe que terminarse con una secuencia determinada. La forma en que se implementa este modelo permite medir el desarrollo con cada avance del proyecto.

Un punto fundamental del modelo en Cascada es que no hay posibilidad de cambios, por lo que una buena planificación es fundamental ya que esto marcará el resultado del proyecto. Si se detecta algún error será en la etapa de pruebas y esto llevaría a un rediseño y nueva programación. En la figura 5 se muestra el modelo en cascada con cada una de sus etapas.

Figura 5. Modelo en Cascada



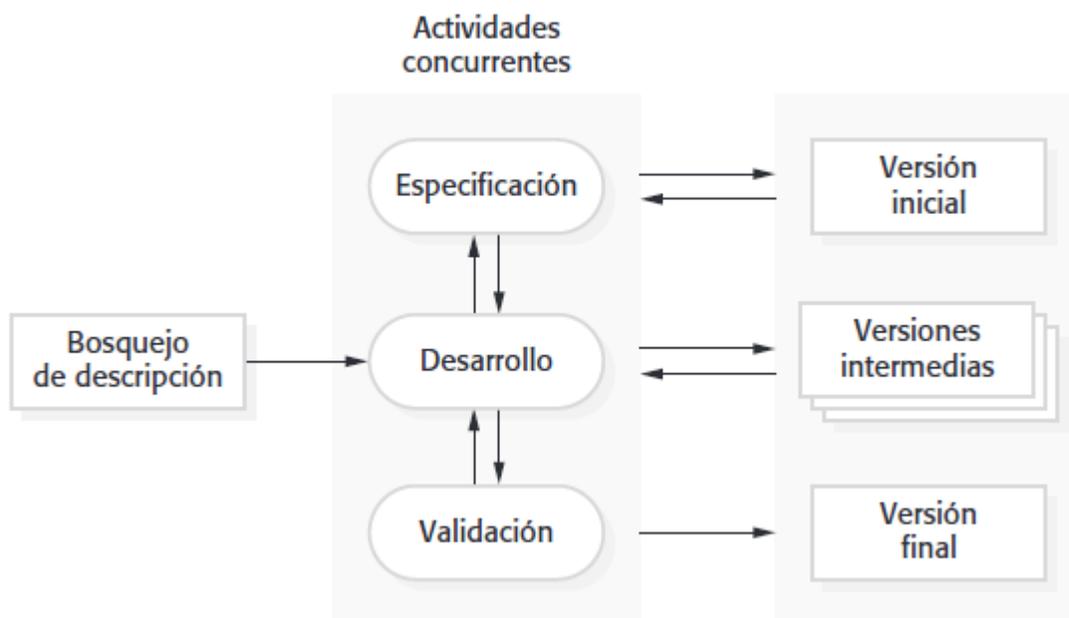
Fuente: Ingeniería del *Software* un enfoque práctico

2.4.2 Desarrollo incremental

Según (Sommerville, 2005, p. 30) “Este enfoque vincula las actividades de especificación, desarrollo y validación. El sistema se desarrolla como una serie de versiones, y cada versión añade funcionalidad a la versión anterior”. En la figura 6 se muestran las actividades del desarrollo incremental.

Cuando se utiliza un modelo incremental, es frecuente que el primer incremento sea el producto fundamental. Es decir, se abordan los requerimientos básicos, pero no se proporcionan muchas características suplementarias (algunas conocidas y otras no). El cliente usa el producto fundamental (o lo somete a una evaluación detallada). Como resultado del uso y/o evaluación, se desarrolla un plan de incremento que sigue. El plan incluye la modificación del producto fundamental para cumplir mejor la necesidad del cliente, así como la entrega de las características adicionales y más fundamentales. Este proceso se repite después de entregar cada incremento, hasta terminar el producto final. (Pressman, 2010, p. 35)

Figura 6. Desarrollo incremental



Fuente: Ingeniería del *Software* un enfoque práctico

El desarrollo incremental permite presentar avances del proyecto y conforme se compruebe su funcionalidad agregar otras nuevas funciones o retroceder si se encuentran errores. Este tipo de desarrollo resulta más barato al permitir cambios en el *software* mientras se desarrolla.

2.4.3 Patrones de diseño

Es posible tener aplicaciones distintas entre ellas con partes que se asemejen como el acceso a la base de datos, la interconexión entre sistemas, comportamiento, entre otras cosas. Cuando se habla de patrones de desarrollo en el software, se habla de soluciones probadas para problemas comunes. Al usar estos patrones, podemos lograr que nuestro código sea más reutilizable y que las aplicaciones tengan una estructura y diseño más coherentes.

Se consideran tres niveles dentro de los patrones de software:

- **Patrones arquitectónicos:** Los cuales describen soluciones al más alto nivel de software y hardware. Normalmente soportan requerimientos no funcionales (criterios de evaluación).
- **Patrones de diseño:** Describen soluciones en un nivel medio de estructuras de software. Normalmente soportan requerimientos funcionales (especificaciones técnicas del sistema).
- **Patrones de programación:** Describen soluciones en el nivel de software más bajo, nivel de clases y métodos. Normalmente soportan características específicas de un lenguaje de programación. (C & Gálvez, 2015, p.4)

Cada patrón consta de cuatro elementos esenciales: el nombre del patrón, que resume el problema y la solución en pocas palabras; el problema, que describe el contexto y las condiciones para aplicar el patrón; la solución, que ofrece una descripción abstracta de cómo los elementos del diseño resuelven el problema; y las consecuencias, que son los resultados y beneficios obtenidos al aplicar el patrón, incluyendo impactos en la flexibilidad, extensibilidad y portabilidad del sistema. Estos elementos son cruciales para comprender y evaluar los patrones de diseño en el desarrollo de software orientado a

objetos, fomentando la reutilización y mejorando la calidad del diseño. (C & Gálvez, 1994, p3)

2.4.3.1 Patrón de diseño Modelo-Vistas-Controlador (MVC)

El patrón de diseño Modelo-Vista-Controlador (MVC) asigna objetos a tres roles y define la comunicación entre ellos. Los objetos en una implementación MVC se agrupan en capas, como la capa del modelo. La implementación de MVC ofrece beneficios como la reutilización de objetos, interfaces mejor definidas y facilidad para extender las aplicaciones.

El Modelo encapsula la información y lógica de manipulación de datos, pudiendo tener relaciones con otros objetos del Modelo. Los objetos del Modelo representan conocimiento de un dominio específico y pueden ser reutilizados en problemas similares. La Vista muestra la información al usuario y permite la edición, comunicándose con el Modelo a través del Controlador. Los objetos de Vista se enteran de los cambios a través del Controlador y envían información a éste. El Controlador actúa como intermediario entre la Vista y el Modelo, interpretando acciones del usuario y comunicando cambios entre ellos. (C & Gálvez, 2015, p. 4-5)

2.4.3.2 Patol de diseño creacional

En el diseño de sistemas, los patrones creacionales abstraen el proceso de creación y composición de objetos, permitiendo la independencia del sistema respecto a cómo se crean y representan los objetos. Hay patrones creacionales de clases que utilizan la herencia para variar la clase que crea una instancia, y patrones creacionales de objetos que delegan la instanciación a otro objeto. A medida que los sistemas dependen más de la composición de objetos que de la herencia de clases, los patrones creacionales se vuelven importantes, ya que permiten definir comportamientos fundamentales que pueden formar

parte de conjuntos más complejos. Estos patrones proporcionan flexibilidad en la creación, configuración y unión de objetos, permitiendo variaciones en su estructura y funcionalidad.

2.4.3.2.1 Patrón de diseño creacional Singleton

“El patrón Singleton garantiza una única instancia de una clase con acceso global, restringiendo la creación de instancias y utilizando un constructor privado y un método estático.” (Blancarte, 2021)

Los componentes del patrón Singleton son el Client, que solicita una instancia de la clase Singleton, y el Singleton, que implementa el patrón y tiene una única instancia en toda la vida de la aplicación. La implementación del patrón permite gestionar la configuración del sistema desde un único punto centralizado, cargando la configuración inicial al iniciar la aplicación y haciéndola disponible en toda la aplicación. (Blancarte, 2021)

2.4.3.3 Patrón de diseño estructural

Los patrones de diseño estructural se centran en la composición de clases y objetos para crear estructuras más complejas. El patrón estructural de clases utiliza la herencia para combinar la implementación de múltiples clases base, lo que permite que clases desarrolladas de manera independiente trabajen juntas. Por otro lado, el patrón estructural de objetos describe cómo los objetos se componen para adaptarse a nuevas funcionalidades, permitiendo una composición flexible y cambios en tiempo de ejecución.

Los patrones estructurales más conocidos son:

- **El patrón Composite** crea una jerarquía de clases que contiene elementos primitivos y compuestos, permitiendo crear estructuras complejas.

- **El patrón Proxy** actúa como sustituto de otro objeto y puede representar un objeto remoto, protegiendo el acceso a información sensible.
- **El patrón Flyweight** se centra en compartir objetos para ahorrar espacio y mejorar la eficiencia.
- **El patrón Facade** crea un objeto que representa un subsistema completo y envía mensajes a las instancias para comunicar responsabilidades.
- **El patrón Bridge** separa la abstracción de la implementación, permitiendo modificar el comportamiento de manera independiente.
- **El patrón Decorator** agrega funcionalidades dinámicamente a los objetos, permitiendo agregar un número ilimitado de funciones adicionales.

(C & Gálvez, 2015, p. 10-11)

2.4.3.4 Patrones de diseño de comportamiento

Los patrones de diseño de comportamiento se centran en los algoritmos y la asignación de responsabilidades entre objetos. Además de describir patrones de objetos o clases, también abordan los patrones de comunicación entre ellos. Estos patrones se caracterizan por un control de flujo complejo, que puede ser difícil de manejar en tiempo de ejecución. Por tanto, su enfoque se centra en la comunicación entre objetos en lugar de controlar el flujo.

Los patrones de comportamiento más conocidos son:

- **El patrón de comportamiento de clases** utiliza la herencia para distribuir el comportamiento entre clases. Dentro de los patrones de este tipo se encuentran: el métodoTemplate (modelo) y el Interpreter (intérprete).
- **El método Template** es un patrón muy utilizado y fácil de implementar, consiste en una definición abstracta de un algoritmo, paso a paso, donde en cada paso se invoca ya sea a una operación abstracta o a una operación primitiva.

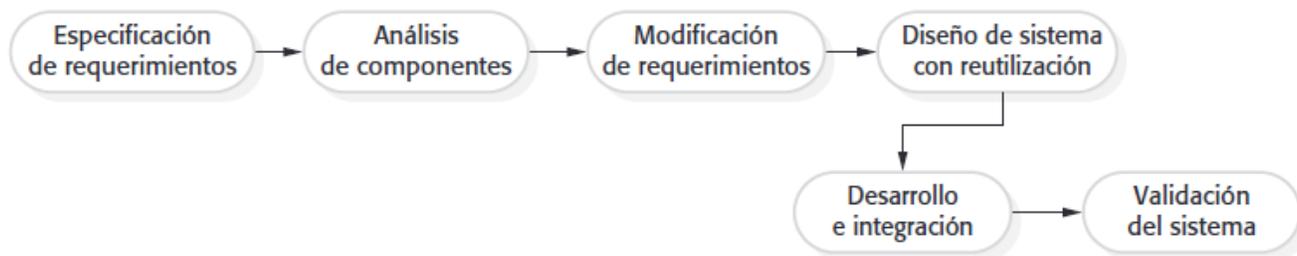
- **El Interpreter** define un conjunto de normas y reglas como una jerarquía de clases, e implementa un intérprete como una operación en las instancias de estas clases.
- **El patrón de comportamiento de objetos** se basa la composición de objetos más que en la herencia. Algunos describen cómo un grupo de dos objetos cooperan para desarrollar una tarea que no podría realizar un objeto solo. Un detalle que sobresale es cómo el par de objetos se conocen para cooperar entre sí, ya que entonces los objetos deberían tener una referencia uno hacia el otro, lo que incrementaría el acoplamiento. Además, en un caso extremo, cada objeto debería poseer una referencia hacia todos los otros objetos.
- **El patrón Mediator (mediador)** permite optimizar las referencias de los pares de objetos utilizando un objeto mediador entre ellos, y como los objetos no necesitan este objeto mediador se elimina el acoplamiento.
- **El patrón de Chain of responsibility** (cadena de responsabilidades) también permite bajar el acoplamiento entre objetos enviando solicitudes a un objeto utilizando una cadena de objetos candidatos, donde cualquier candidato podría cumplir la solicitud dependiendo de las condiciones de ejecución. El número de candidatos es ilimitado y se puede seleccionar qué objetos van a ser parte de la cadena de candidatos en tiempo de ejecución.
- **El patrón Observer (observador)** define, mantiene y administra una dependencia entre objetos, informando en tiempo real sobre los cambios que se presentan en uno u otro objeto.
- **El patrón Strategy (estratégico)** encapsula un algoritmo dentro de un objeto, haciendo con esto fácil de especificar y cambiar el algoritmo que utiliza un objeto.
- **El patrón Command (comando)** encapsula una solicitud dentro de un objeto, de tal manera que éste se puede pasar como parámetro, guardar en un histórico o manipular de alguna otra manera. (C & Gálvez, 2015, p.12-13)

2.4.4 Ingeniería de Software orientada a la reutilización

Para las personas que suelen trabajar en proyectos de programación es muy común usar códigos o técnicas de programación en sus diferentes proyectos, podemos decir que es una reutilización “informal”.

Este tipo de modelo es ampliamente usado y se puede desarrollar siempre que se sigan una serie de etapas. La figura 7 se muestra el proceso de la ingeniería de *software* orientada a la reutilización.

Figura 7. Proceso de la ingeniería de software orientada a la reutilización.



Fuente: Ingeniería del *Software* un enfoque práctico

Teniendo una serie de requerimientos, se buscan componentes que puedan acoplarse, generalmente hay pocas coincidencias por lo que se usan las que se asemejen más modificándolos o analizando otras alternativas. Se estudian los conceptos del programa existente para saber si es factible reutilizarlos e integrarlos al nuevo sistema, esta integración puede tomarse como parte del desarrollo.

Tabla 2. Comparación de los modelos de proceso de software

Modelo	Características	Ventajas	Desventajas
Modelo en Cascada	Proceso dirigido por un plan previo. Sus actividades fundamentales son: análisis de requerimientos, diseño, implementación y prueba, integración y mantenimiento.	Cada etapa se fortalece de la anterior. Al modelo permite la retroalimentación de una fase a otra identificando problemas.	Después de un cierto número de repeticiones en alguna etapa, el proceso puede volverse tedioso detener el resto del desarrollo.
Modelo Incremental	Se desarrolla un sistema inicial y se agregan funcionalidades al sistemas en actualizaciones posteriores .	Este tipo de desarrollo, resulta más barato y fácil de hacer cambios en el <i>software</i> , mientras se diseña. 2. Los clientes pueden ser su opinión sobre el desarrollo del <i>software</i> . La entrega del producto tiende hacer más rápida.	Pueden darse problemas de coordinación en aquellas empresas acostumbradas a procesos más estructurados. No es posible medir los avances, esto evita la estimar el costo del <i>software</i> .
Ingeniería del software Orientada a la reutilización	Se basa en la reutilización o adaptación de las partes de un <i>software</i> que cumpla con los requerimientos del cliente.	Reduce la cantidad de trabajo de los programadores, esto reduce el costo. Las entregas del producto es más rápida.	Se puede llegar a perder el control del avance del sistema. Puede que no se cumplan con exactitud los requerimientos de los usuarios.

Fuente: Creación propia

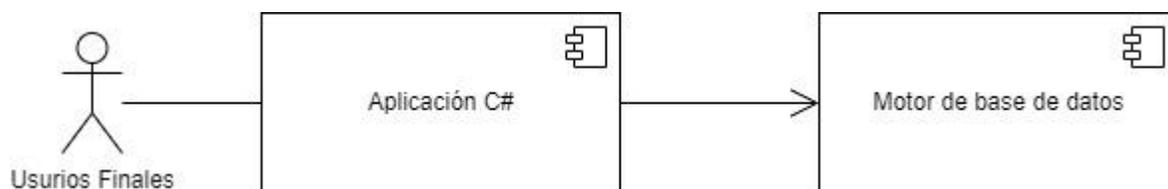
2.4.5 Arquitectura de Capas y niveles

Lo que se conoce como arquitectura en capas es en realidad un estilo de programación donde el objetivo principal es separar y organizar los diferentes aspectos del desarrollo de software en capas lógicas, cada una con una responsabilidad en específico.

El termino capa se utiliza para diferenciar a las diferentes partes en las que se divide una aplicación desde el punto de vista lógico; mientras que nivel se refiere a la forma física en que está organizada una aplicación.

Con los primeros lenguajes de programación la aparición de los motores de base de datos se diferencian dos niveles, el primero consiste en el codificador y el segundo en la base de datos. La figura 8 se muestra el nivel de la aplicación y el nivel de datos donde está la base de datos.

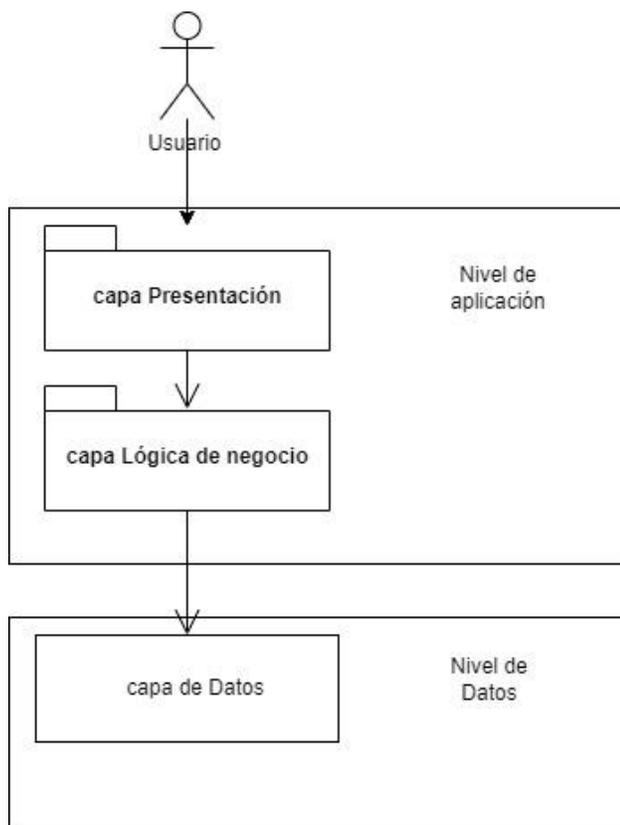
Figura 8. Nivel de aplicación y nivel de datos



Fuente: Creación propia

Cuando se desarrolla un software, es importante que algunas partes del mismo puedan ser reemplazadas o modificadas sin tener que hacer cambios en todo el programa. Esto se logra mediante la arquitectura en capas. La figura 9 muestra cómo está organizado un software en dos niveles y dos capas. En este caso, la capa de presentación y la capa lógica de negocio interactúan entre sí.

Figura 9. Arquitectura en dos niveles y dos capas



Fuente: Creación propia

La capa de presentación es la que se encarga de la parte visual y de interactuar con el usuario. Muestra la interfaz gráfica con la que el usuario puede interactuar y proporciona servicios para que el usuario realice acciones en el software.

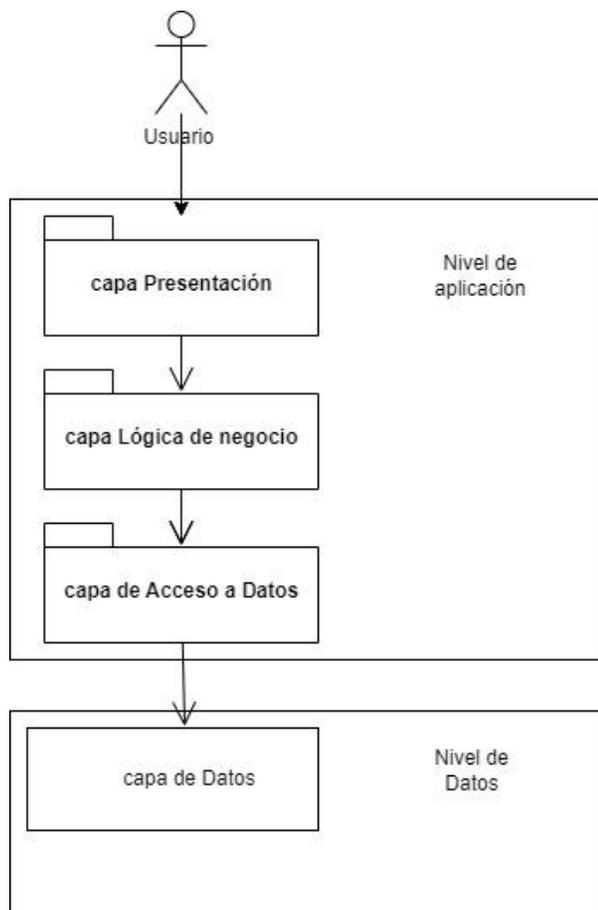
La capa lógica de negocio es la que contiene la lógica principal del software. Aquí se definen las reglas y procesos necesarios para que el software funcione correctamente. La capa de presentación utiliza los servicios que la capa lógica de negocio ofrece para realizar acciones y presentar la información adecuada al usuario.

La arquitectura en 3 capas es una forma de organizar una aplicación de software en tres partes diferentes. Cada parte tiene un trabajo específico y juntas hacen que la aplicación funcione correctamente.

La capa de acceso a datos es una parte del código que se encarga de obtener la información que se necesita de la base de datos. Permite conectar y traer los datos que se utilizan en la aplicación.

La capa de datos es donde se almacenan los datos en sí. Aquí se guardan y organizan los diferentes tipos de información, como si fuera una lista o una tabla. La figura 10 muestra una arquitectura de 3 capas.

Figura 10. *Arquitectura en tres capas*



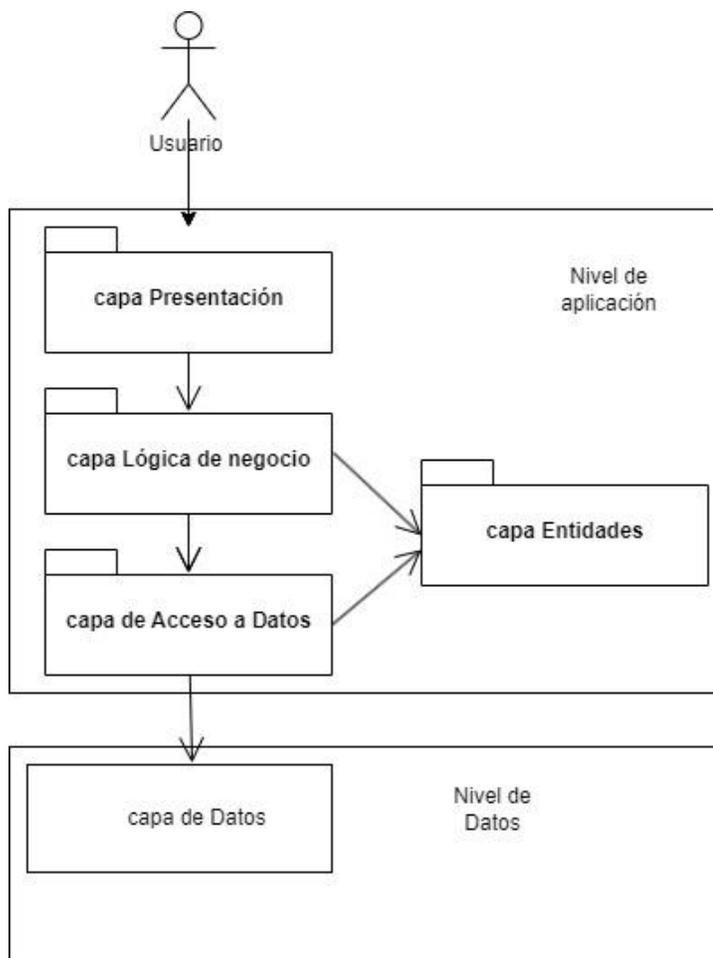
Fuente: Creación propia.

Los componentes de la capa de lógica de negocios necesitan referenciar a instancias de las clases del dominio y los componentes de la capa de acceso a datos también tienen que referenciarlas para poner rellenas tales instancias con los datos que obtienen de las capas inferiores.

La capa Entidades corresponde al dominio de la aplicación. En esta capa se encuentra la declaración de las entidades, de manera que se puedan referenciar desde otras capas sin entrar en

ciclos recursivos de compilación. Este esquema propone una total independencia entre la lógica de negocios y las entidades.

Figura 11. *Arquitectura de tres capas con la capa Entidades*



Fuente: Creación propia

2.4.6 *Desarrollo rápido de software*

Existen varios métodos enfocados en la entrega rápida de los proyectos de desarrollo de *software*. Esta sección está enfocada en describir las diferencias entre un desarrollo incremental y el que está basado en especificaciones y diseño.

Actualmente, los negocios operan en un entorno global que cambia rápidamente. Tienen que responder a nuevas oportunidades y mercados, condiciones económicas cambiantes y la aparición de servicios y productos competidores. El *software* es parte de casi todas las operaciones de negocio, por lo que es fundamental que el *software* nuevo se desarrolle rápidamente para aprovechar nuevas oportunidades y responder a la presión competitiva. Por lo tanto, actualmente el desarrollo y entregas rápidas son a menudo los requerimientos más críticos de los sistemas *software*.

Los procesos de desarrollo rápido de *software* están diseñados para producir *software* útil de forma rápida. Generalmente, son procesos iterativos en los que se entrelazan la especificación, el diseño, el desarrollo y las pruebas. El *software* no se desarrolla y se utiliza en su totalidad, sino en una serie de incrementos, donde en cada incremento se incluyen nuevas funcionalidades al sistema. Aunque existen muchos enfoques para el desarrollo rápido de *software*, comparten las mismas características fundamentales

1. Los procesos de especificación, diseño e implementación son concurrentes. No existe una especificación del Sistema detallada, y la documentación del diseño se minimiza o es generada automáticamente por el entorno de programación utilizado para implementar el sistema. El documento de requerimientos del usuario define solamente las características más importantes del sistema.
2. El sistema se desarrolla en una serie de incrementos. Los usuarios finales y otros stakeholders del sistema participan en la especificación y evaluación de cada incremento. Pueden proponer cambios en el *software* y nuevos requerimientos que se deben implementar en un incremento posterior al sistema.
3. A menudo se desarrollan las interfaces del usuario del sistema utilizando un sistema de desarrollo interactivo que permite que el diseño de la interfaz se cree rápidamente dibujando y colocando iconos en la interfaz. El sistema puede generar una interfaz basada en web para un navegador o una interfaz para una plataforma como Microsoft Windows. (Sommerville, 2005, págs. 358-359).

Según (Bahit, 2012, p. 16) “Los enfoques ágiles se diferencian (y también guardan ciertos aspectos en común) de los predictivos, básicamente por la forma de abordaje de un proyecto.”

El enfoque predictivo, es aquel que plantea el abordaje estricto de un proyecto, sobre la base del cumplimiento de tres aspectos predefinidos al comienzo del proyecto: alcance, costo y

tiempo; mientras tanto el enfoque ágil, plantea los proyectos desde el cumplimiento de un objetivo más amplio: entregar *software* con el mayor valor posible. (Bahit, 2012, p. 16)

Las metodologías tradicionales (o predictivas), encaran las fases que componen el ciclo de vida del desarrollo de *software*, de manera sucesiva. Es decir, que una fase sucede a otra, y cada fase ocupa un espacio lineal en él.

En cambio, las metodologías ágiles, solapan estas etapas, permitiendo ahorrar tiempo, evitando la dependencia (cada etapa es independiente de la otra) y haciendo y haciendo del ciclo de vida un proceso iterativo (se inicia con el relevamiento, se finaliza con la implementación y se vuelve a comenzar para abordar nuevas funcionalidades. (Bahit, 2012, p. 21)

Las metodologías ágiles poseen su propio manifiesto: el Agile Manifiesto (o Manifiesto Ágil en español).

El Manifiesto Ágil es un documento redactado en 2001 por 17 expertos en programación que supuso un cambio radical en la forma de desarrollar '*software*'. Frente a los modelos tradicionales excesivamente rígidos y alejados de las necesidades de los clientes, estos gurús propusieron cuatro valores que inspiran las diferentes metodologías ágiles que han surgido desde entonces. (BBVA, 2022)

En el manifiesto Ágil, podemos leer los doce principios en los cuales se argumentan los valores del Manifiesto. Estos son:

Principio #1: Satisfacer al cliente mediante la entrega temprana y continua de *software* con valor.

Principio #2: Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.

Principio #3: Entregamos *software* funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.

Principio #4: Los responsables de negocio y desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.

Principio #5: los proyectos se desarrollan en torno a individuos motivados. Hay que darles el apoyo y el entorno que necesitan, y confiarles la ejecución del proyecto.

Principio #6: Conversación cara a cara.

Principio #7: El *software* funcionando es la medida principal de progreso.

Principio #8: Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.

Principio #9: La atención continua a la excelencia técnica y al buen diseño mejora la agilidad.

Principio #10: La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.

Principio #11: Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.

Principio #12: A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.

2.4.7 *Scrum*

"De todas las metodologías ágiles, Scrum es única porque introduce la idea del control empírico de los procesos. Esto significa que Scrum utiliza el progreso real de un proyecto para planificar y concertar los lanzamientos. En Scrum, los proyectos se dividen en ritmos de trabajo breves, conocidos como sprints. Normalmente, tienen una, dos o tres semanas de duración. Al final de cada sprint, el cliente y los miembros del equipo se reúnen para evaluar el progreso del proyecto y planear los siguientes pasos a seguir. Esto permite que la dirección del proyecto se ajuste o se reoriente una vez finalizado el trabajo, sin especulaciones ni predicciones”
(Pecharromán, 2015, p.28)

Es una metodología ágil para la gestión de proyectos relacionados con el desarrollo de *software* caracterizado por:

1. Equipos autónomos y auto gestionados que comparten su conocimiento de forma abierta y aprenden juntos.
2. Una estrategia de desarrollo incremental, en lugar de planificación complete del producto.

3. Basar la calidad del resultado en el conocimiento tácito de las personas y su creatividad; no en la calidad de los procesos empleados.
4. Solapar las diferentes fases del desarrollo, en lugar de realizarlas una tras otra en un ciclo secuencial. (Palacio, 2022, p. 21)

2.4.7.1.1 Roles

Los equipos de Scrum definen tres roles bien diferenciados:

- **El dueño del producto:** Es la única persona autorizada para decidir sobre cuales funcionalidades y características funcionales tendrá el producto. Es quien representa al cliente, usuarios del *software* y todas aquellas partes interesadas en el producto.
- **El Scrum Master:** Un error frecuente es llamarlo “líder”, puesto que el Scrum no es líder típico, sino que es un auténtico servidor neutral, que será el encargado de fomentar e instruir sobre los principios ágiles de Scrum.
- **El equipo de desarrollo (Scrum Team o Equipo Scrum):** Es el equipo de desarrolladores multidisciplinario, integrado por programadores, diseñadores, arquitectos, testers y demás, que, en forma auto-organizada, será los encargados de desarrollar el producto.
(Bahit, 2012, págs. 33-37)

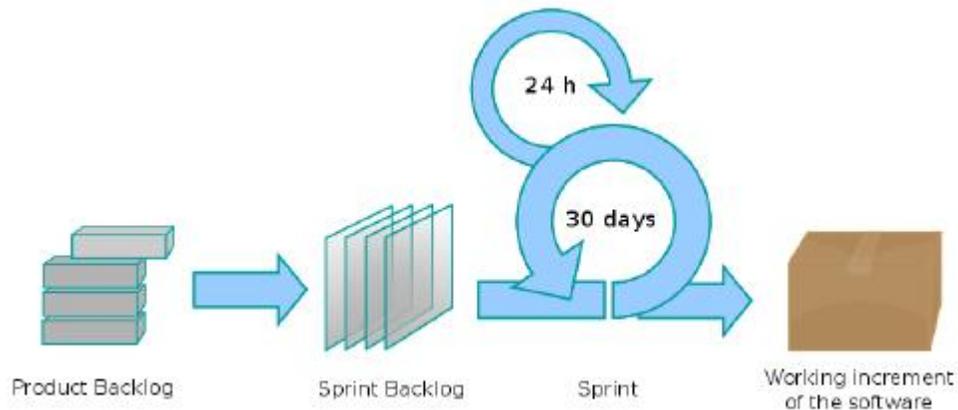
2.4.7.1.2 Artefactos y herramientas

Scrum, propone tres herramientas o "artefactos" para mantener organizados nuestros proyectos. Estos artefactos, ayudan a planificar y revisar cada uno de los *Sprints*, aportando medios ineludibles para efectuar cada una de las ceremonias que veremos más adelante. Ahora, nos concentraremos principalmente, en el *backlog* de producto, el *backlog* de *Sprint* y el *Scrum Taskboard*.

- **Backlog de producto:** Es listado dinámico y públicamente visible para todos los involucrados en el proyecto.

- **Backlog de Sprint:** Lista reducida de ítems del *Backlog* de producto, que han sido negociados el dueño de Producto y el *Scrum Team* durante la planificación del *Sprint*.
 - **Scrum taskboard:** Un tablero físico que permite la transparencia de los ítems de trabajos pendientes, y terminados de un mismo *Sprint*.
- (Bahit, 2012, págs. 39-45)

Figura 12. Diagrama sencillo del proceso Scrum



Fuente: Aplicación de la metodología de desarrollo ágil Scrum para el desarrollo de un Sistema de gestión de empresas

"La metodología Scrum para el desarrollo ágil de software representa un punto de partida de la gestión en cascada. De hecho, Scrum y otro tipo de procesos ágiles se inspiraron en sus limitaciones. La metodología Scrum enfatiza en la comunicación y colaboración, el funcionamiento del software, y la flexibilidad de la que dispone para adaptarse a las emergentes realidades de las empresas - todos los atributos de los que carecía el modelo de cascada" (Pecharromán, 2015, p.28)

2.4.8 Programación eXtrema

Según (Bahit, 2012, p. 70) “*eXtreme Programming* (programación extrema) también llamado XP, propone solo un conjunto de prácticas técnicas, que, aplicadas de manera simultánea, pretender enfatizar los aspectos positivos de un proyecto de desarrollo de *software*”

Se apoya en cinco valores, los cuales enfatizan la esencia colaborativa del equipo. Estos valores son:

- **Comunicación:** Todo es trabajado en equipo, desde el relevamiento y el análisis hasta el código fuente desarrollado. Todo se conversa cara a cara, procurando hallar soluciones en conjunto a los problemas que puedan surgir.
- **Simplicidad:** Se pretende desarrollar solo lo necesario y no perder tiempo en detalles que no sean requeridos en el momento. En este aspecto, se asemeja a otra metodología ágil. Denominada Kanban, en el cual, un proceso “anterior” solo produce lo que el proceso posterior demanda.
- **Retroalimentación:** Se demanda al cliente una retroalimentación continua a fin de conocer sus requerimientos e implementar los cambios tan pronto como sea posible.
- **Respeto:** El equipo respeta la idoneidad del cliente como tal (sólo éste, es quien conoce el valor para el negocio) y el cliente, a la vez, respeta la idoneidad del equipo (confiando en ellos profesionalmente para definir y decidir el “cómo” se llevará a cabo el desarrollo de lo requerido).
- **Coraje:** Se dice que en XP un equipo debe tener el valor para decir la verdad sobre el avance del proyecto y las estimaciones del mismo, planificando el éxito en vez de buscar excusas sobre los errores.

(Bahit, 2012, págs. 70-71)

2.4.8.1.1 Roles

XP define 7 roles en su proceso:

- **Programador:** El programador escribe las pruebas unitarias y produce el código del sistema. Debe existir una comunicación y coordinación adecuada entre los programadores y otros miembros del equipo.
- **Cliente:** El cliente escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio. El cliente es sólo uno dentro del proyecto, pero puede corresponder a un interlocutor que está representando a varias personas que se verán afectadas por el sistema.
- **Encargado de pruebas (*Tester*):** El encargado de pruebas ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.
- **Encargado de seguimiento (*tracker*):** El encargado de seguimiento proporciona realimentación al equipo en el proceso XP. Su responsabilidad es verificar el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, comunicando los resultados para mejorar futuras estimaciones. También realiza el seguimiento del progreso de cada iteración y evalúa si los objetivos son alcanzables con las restricciones de tiempo y recursos presentes. Determina cuándo es necesario realizar algún cambio para lograr los objetivos de cada iteración.
- **Entrenador (*Coach*):** Es responsable del proceso global. Es necesario que conozca a fondo el proceso XP para proveer guías a los miembros del equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.
- **Consultor:** Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto. Guía al equipo para resolver un problema específico.

- **Gestor:** Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación.
(Letelier & Penadés, 2006, págs. 12,13)

2.4.8.1.2 Proceso XP

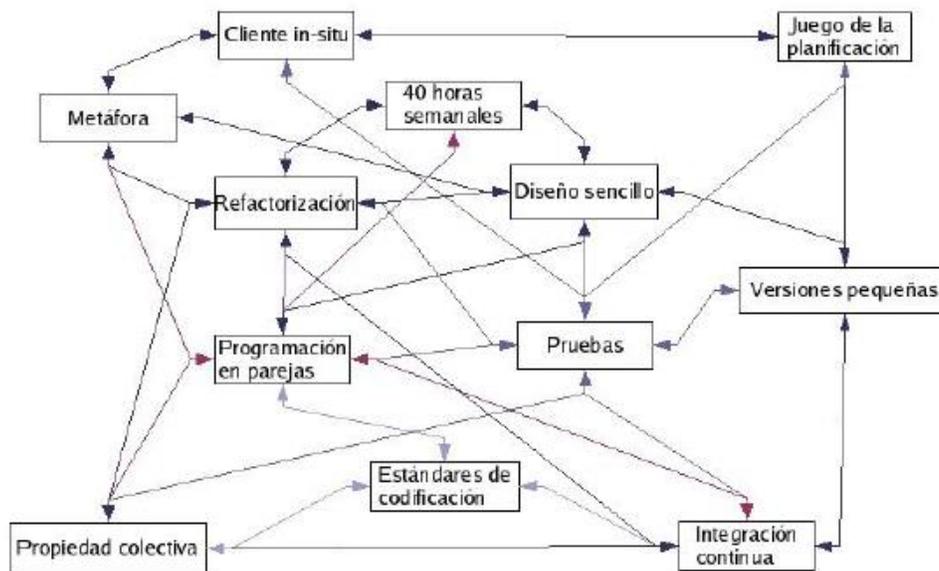
El ciclo de desarrollo consiste en los siguientes pasos:

- **Exploración:** Los clientes plantean las historias de los usuarios, que son de interés para la primera entrega del producto. El equipo de desarrollo se familiariza con las herramientas de desarrollo. Esta etapa puede durar pocas semanas o meses.
- **Planificación de la entrega:** En esta etapa se establece la prioridad de las historias de los usuarios y los programadores estiman el costo de cada uno.
- **Iteraciones:** consiste en los ciclos repetibles proyecto antes de ser entregado. Cada iteración debe durar como máximo tres semanas.
- **Producción:** consiste en la fase de llevar el proyecto terminado en las iteraciones a al ambiente de trabajo diario.
- **Mantenimiento:** una vez que la iteración termina, se lleva a producción en tanto la siguiente iteración se encuentra en proceso; XP asegura el funcionamiento del proyecto ofreciéndole soporte al cliente.
- **Muerte del proyecto:** ocurre cuando el proyecto termina con las historias de usuario. (Letelier & Penadés, 2006, págs. 14-15)

La principal suposición que se realiza en XP es la posibilidad de disminuir la mítica curva exponencial del costo del cambio a lo largo del proyecto, lo suficiente para que el diseño evolutivo funcione. XP apuesta por un crecimiento lento del costo del cambio y con un comportamiento asintótico. Esto se consigue gracias a las tecnologías disponibles para ayudar en el desarrollo de *software* y a la aplicación disciplinada de las prácticas: el juego de la planificación, entregas pequeñas, metáfora, diseño simple, refactorización,

programación en parejas, propiedad colectiva del código, integración continua, 40 horas por semana, cliente *in situ* y estándares de programación. (Letelier & Penadés, 2006, p. 17).

Figura 13. organización de las practicas del modelo XP



Fuente: Metodologías ágiles para el desarrollo de *software*: eXtreme Programming (XP)

2.5 Etapas del modelo en cascada

El modelo de cascada se divide en fases, dependiendo de la literatura o la fuente, estas fases pueden ser más, o llamarse de diferente forma. En esta sección hablare de 5 fases:

2.5.1 Definición de los requerimientos

Esta etapa es la esencial, pues aquí se establecen las bases del programa que se va a desarrollar.

Es esta etapa se crean documentos donde el cliente plantea los requisitos que debe tener el producto final. “La comunicación entre la empresa y el cliente es esencial para cerrar las pautas que se deben cumplir y no dejar ningún cabo suelto, igualmente, la creación de este documento puede ayudar a la empresa a establecer plazos de entrega o costes” (Solbyte, 2021)

2.5.2 *Requerimientos funcionales*

Según (Sommerville, 2005, p. 107) “Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que este deber reaccionar a entradas particulares y de cómo se debe de comportar en situaciones particulares.” En algunos casos, los requerimientos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer.

Los requerimientos funcionales para un sistema, reflejan lo que el sistema debe hacer. Los requerimientos dependen del sistema que se esté desarrollando, de los usuarios finales y del enfoque que la organización les dé a estos requerimientos.

2.5.3 *Requerimientos no funcionales*

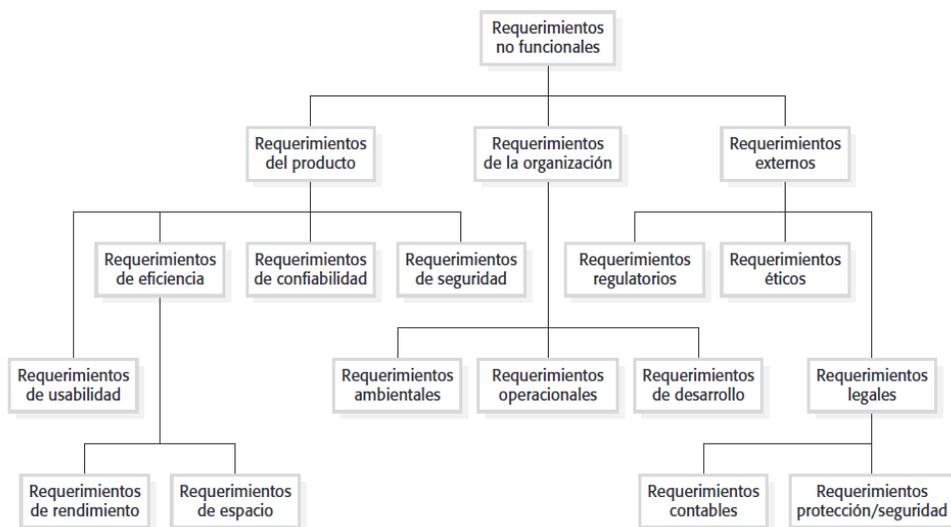
Según (Sommerville, 2005, p. 88) “Son aquellos requerimientos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de este como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento.”

Estos tipos de requerimientos no se relaciona de forma directa con las funcionalidades del sistema, más bien están relacionados a otros aspectos como, tiempo de respuesta, facilidad de uso, y almacenamiento, que son algunos factores que podría restringir de alguna forma las características del sistema.

“Los requerimientos no funcionales provienen de características requeridas del *software* (requerimientos del producto), La organización que desarrolla el *software* (requerimientos de la

organización) o fuentes externas”. (Sommerville, 2005, p. 89). La figura 14 muestra los tipos de requerimientos no funcionales.

Figura 14. Tipos de requerimientos no funcionales



Fuente: Ingeniería del *Software* un enfoque práctico

2.5.4 Diseño del sistema

En esta siguiente fase se define una estructura para el programa. “Esto da lugar a la creación de un documento en el que se describe la estructura global del sistema, así como una especificación de cuál es la función de cada una de sus partes o cómo se combinan entre ellas” (Solbyte, 2021, p. 63).

En la sección 2.6 se abarcará con más detenimiento del tema del Lenguaje Unificado de Modelado (UML), que desempeña un papel muy importante en el desarrollo de *software*.

2.5.5 Implementación

En esta etapa se toma el diseño y los requisitos recogidos para realizar los algoritmos necesarios. “...es la primera etapa en la que comenzamos a obtener resultados” (Solbyte, 2021, p.63)

2.5.6 Pruebas

Consiste en la comprobar que cada uno de los componentes del programa realicen las tareas para los que fueron diseñados. Permite encontrar mejoras en el caso de ser necesarias mejorar el código y aumentar la calidad del sistema.

2.5.7 Tipos de pruebas

Conforme se vaya avanzando en el desarrollo del *software*, se irán realizando diferentes tipos de pruebas. Este mismo proceso nos indicara cual es la prueba más indicada.

2.5.8 Pruebas de sistema

Las pruebas de sistema representan un paso fundamental del proceso, según (Ramirez, Zapata, & Acosta, 2021, p. 15) Se comprueba si el producto con los requisitos especificados. Las pruebas de sistema pueden incluir pruebas basadas en riesgos y/o especificaciones de requisitos, procesos de negocios, casos de uso descripciones de texto de alto nivel o modelos de comportamiento de sistema, interrelaciones con el sistema operativo y recursos del sistema.

Las pruebas de sistema pueden clasificarse en dos diferentes tipos: pruebas de integración y pruebas de entregas.

2.5.9 Pruebas de integración

Las pruebas de integración se basan en unir las partes de un programa y probar producto resultante para encontrar los programas que puedan surgir de la unión de los componentes. Estos componentes pueden ser reutilizados y adaptados al sistema o componentes nuevos desarrollados. Las pruebas de integración comprueban que los componentes funcionen entre sí, con un tiempo de respuesta aceptable. (Sommerville, 2005, p. 495)

2.5.10 Pruebas de entrega

Los clientes son un elemento de primer orden de importancia puesto que las acciones tomadas se desarrollan en pro de su satisfacción. Según (Sommerville, 2005, p 497) proceso de probar una entrega del sistema que será distribuida a los clientes. El principal objetivo de este proceso objetivo de este proceso es incrementar la confianza del suministrador en que el sistema satisface sus requerimientos. Si es así, este puede entregarse como un producto o ser entregado al cliente.

2.5.11 Pruebas de validación o aceptación

Este tipo de pruebas son responsabilidad cliente y participan de forma directa. Esta prueba se aplica antes de que el sistema se ponga en total uso y debe cumplir las expectativas del cliente. Este tipo de prueba puede clasificarse en: pruebas de aceptación del usuario, pruebas operativas, y pruebas alfa y beta.

2.5.12 Pruebas de aceptación del usuario

En el mercadeo se profesa que el cliente siempre tiene la razón y es lo más importante, según (Ramírez, Zapata, & Acosta, 2021, p. 16) “Hay casos en que los clientes y el usuario final son diferentes y lo que parece valido a un usuario final, puede que no le parezca valido a otro. Es fundamental pruebas con los usuarios finales.”

2.5.13 Pruebas operativas

Según (Ramírez, Zapata, & Acosta, 2021, p. 16) “Estas pruebas son llevadas a cabo por los administradores del sistema que se va a poner en producción”

2.5.14 Pruebas alfa y beta

Cuando un programa es utilizado por varios usuarios, no es conveniente hacer prueba de aceptación en cada uno de los clientes. Para esto se utilizan la pruebas alfa y beta para encontrar errores que podría detectar el usuario final.

2.5.14.1 Pruebas alfa

Estas pruebas las realizan representantes de los usuarios finales junto con los desarrolladores encontrando los errores y problemas de uso.

2.5.14.2 Pruebas beta

Estas pruebas se realizan en el sitio del cliente en situaciones específicas. Un sistema usado por varios clientes puede presentar distintos problemas en distintas situaciones.

2.5.15 Pruebas de caja blanca

Es un tipo de prueba que usa sentencias de control para obtener los casos de pruebas. Dentro de estas sentencias podemos encontrar las de decisiones o las llamadas banderas.

2.5.16 Pruebas de caja negra

Son las que utilizan el análisis de especificación, tanto funcional como no funcional, sin tener en cuenta la estructura interna del programa para diseñar los casos de pruebas. A diferencia de las pruebas de caja blanca, estas suelen realizarse durante las últimas etapas de las pruebas. (Ramirez, Zapata, & Acosta, 2021, p. 15)

2.5.17 Herramientas de Pruebas

Existen herramientas que permiten realizar pruebas automáticas al *software* permitiendo que los proyectos se manejen de forma más eficiente y con un alto nivel de calidad, en esta sección se presentan algunas:

2.5.17.1.1 Herramientas para pruebas estáticas

Este tipo de herramientas ayudan a encontrar defectos en las etapas tempranas del proyecto.

- **PDM** (libre): es una analizadora de código fuente, encuentra defectos comunes de programación como las variables utilizadas, bloques *catch* vacíos, creación de objetivos innecesarios, etc.
- **Ckckstyle** (libre): es una herramienta de desarrollo para ayudar a los programadores a chequear que el código Java cumple un estándar de codificación.
- **Simian** (libre): herramienta que detecta *software* duplicado. Acepta: Java, C#, C++, C Cobol, Ruby, asp, jsp, HTML, xml, y Visual Studio.
(Ramirez, Zapata, & Acosta, 2021, p. 18)

2.5.17.1.2 Herramientas para planificación y gestión

- **Testlink** (libre): permite crear y gestionar casos de prueba y organizarlo dentro de planes de pruebas, permite generación de informes, así como priorizar y asignar tareas.
- **Mantis** (libre): herramienta para gestionar incidencias.
- **IBM Rational Quality Manager (licencia)**: permite gestionar requisitos del proyecto, la gestión, diseño y la creación de pruebas y gestión de incidencias.
- **Testopia**: es una administradora de casos de pruebas. Diseñado para realizar seguimiento a los casos de pruebas. (Ramirez, Zapata, & Acosta, 2021, págs. 17 y 18)

2.5.17.1.3 Herramientas para pruebas de automatización

-
- **Selenium:** es un conjunto de herramientas para automatizar los navegadores web a través de muchas plataformas que nos permitirá crear conjuntos de pruebas sobre aplicaciones web.
- **SoapUI:** permite probar, simular y generar código de servicios web partiendo del contrato de los mismos en formato WSDL y con vinculo SOAP sobre http (Ramirez, Zapata, & Acosta, 2021, p. 19)

2.5.17.1.4 Herramientas para pruebas de carga y rendimiento

- **Jmeter:** herramienta de prueba de carga para análisis y medir el desempeño de una variedad de servicios con énfasis en aplicaciones web.
- **NMAP:** Permite la exploración de las redes. Utiliza paquetes IP para determinar que equipos están disponibles en la red, que servicios ofrecen, que sistemas operativos están usando, o que tipo de filtros/cortafuegos están usando, etc. (Ramirez, Zapata, & Acosta, 2021, p. 19)

2.4.1.5 Mantenimiento

Según (Solbyte, 2021) “ una vez estén listas cada una de las funcionalidades del producto y hemos comprobado que cada una funciona correctamente, instalamos la aplicación y verificamos que su funcionamiento sea correcto en el entorno que va a ser utilizado”

Una forma efectiva para la implementación de un sistema y la recolección de requerimientos de los participantes que interactúan en un proyecto, es representar en una forma simple la realidad, describiendo lo que se supone que el sistema debería de hacer. El lenguaje unificado de modelación (UML), está compuesto por elementos que permiten elevar diagramas

que permiten presentar diversas perspectivas de un sistema. A continuación, se amplía el concepto de UML y se describen los diagramas más comunes.

2.6 UML

Según (Rumbaugh, Jacobson, & Booch, 2007, p. 15) “... UML es un lenguaje de modelado visual de propósito general que se utiliza para especificar, visualizar, construir y documentar los artefactos de un sistema de *software*”.

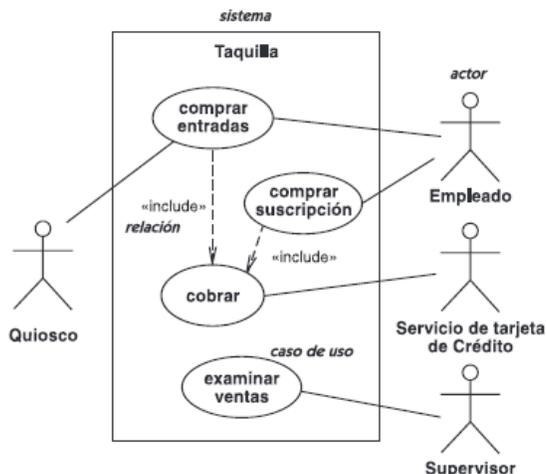
La función principal del lenguaje UML es demostrar por medio de diagramas, como se comportaría un sistema, usando una serie de símbolos y reglas universales que, sin importar el lenguaje de programación o experiencia de un programador, le permita entender el trabajo de otro.

Según (Pressman, 2010) “Si usted entiende el vocabulario de UML (los elementos pictóricos de los diagramas y su significado) puede comprender y especificar con mucha más facilidad un sistema, y explicar su diseño a otros”.

2.6.1 Casos de uso

Un caso de uso es la descripción de las acciones de un sistema desde el punto de vista del usuario. Esta herramienta permite obtener los requerimientos desde el punto de vista de los usuarios. La figura 15 muestra los componentes de un diagrama de uso clásico

Figura 15. Diagrama de caso de uso



Fuente: El lenguaje unificado de modelado, manual de referencia.

El rectángulo representa los límites del sistema que contiene los casos de uso. Los actores se ubican fuera de los límites del sistema, los casos de uso se representan con óvalos que indican la función del sistema, mientras que los actores son los usuarios del sistema.

2.6.2 Diagrama de clases

Los diagramas de clases describen la estructura estática de un sistema.

2.6.3 Clases

Un rectángulo representa a la clase dividida en tres partes: la superior contiene el nombre de la clase, la central los atributos y la inferior las acciones. En la figura 16 se muestra un ejemplo de la representación una clase y sus partes.

Figura 16. Clase automóvil

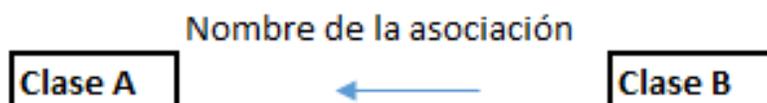


Fuente: creación propia

2.6.4 Asociaciones

Las asociaciones representan las relaciones entre las clases. El nombre de la asociación se coloca sobre la línea que la representa. Se debe de indicar la dirección de la asociación, para eso se utiliza una flecha rellena. La forma en que las clases se ven entre ellas se conoce como roles, los roles se representan al final de cada asociación. La figura 17 muestra un ejemplo de clases con su dirección, asociación y roles.

Figura 17. Relación, asociación y dirección entre clases

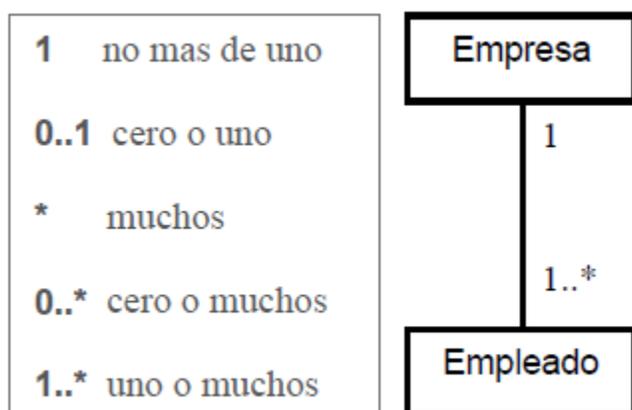


Fuente: Creación propia

2.6.5 Multiplicidad

La multiplicidad indica el número de instancias de una clase vinculadas a una de las instancias de la otra clase. Por ejemplo, un grupo puede tener un solo profesor, pero un profesor puede tener varios grupos. La multiplicidad se señala cerca del final de una asociación. En la figura 18 se muestra un ejemplo de cómo se señala la multiplicidad.

Figura 18. Ejemplo de multiplicidad.



Fuente: Autor desconocido

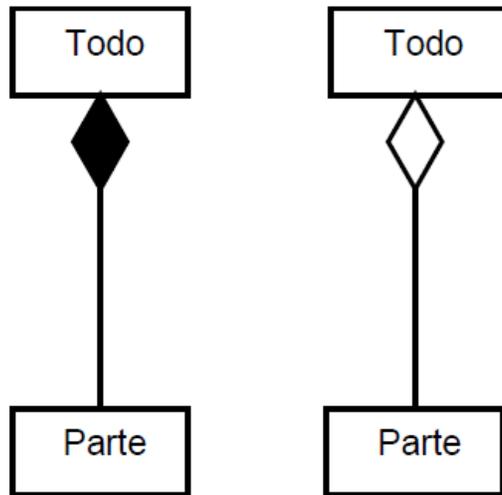
2.6.6 Composición y Agregación

“La composición es un tipo de agregación que denota una fuerte posesión de la clase “Todo”, a una clase “Parte”. Se grafica con un rombo relleno contra la clase que representa el todo”. (Diagramas del UML , sf)

La agregación es una relación en la que la clase “Todo” juega un rol más importante que la clase “Parte”, pero las dos clases no son dependientes una de la otra. Se grafica con un rombo vacío contra la clase “Todo”. (Diagramas del UML , sf)

La figura 19 muestra un ejemplo de Composición y Agregación.

Figura 19. Composición y Agregación



Fuente: Autor desconocido

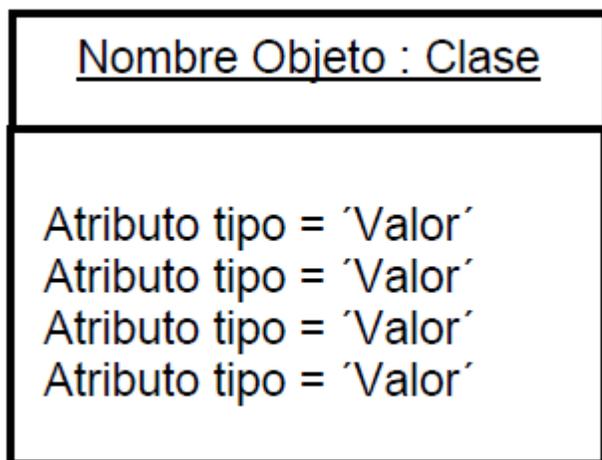
2.6.7 Diagramas de Objetos

Los diagramas de Objetos están vinculados con los diagramas de Clases.

Un objeto es una estancia de una clase, por lo que un diagrama de objetos puede ser visto como una instancia de un diagrama de clases. Los diagramas de objetos describen la estructura estática de un sistema en un momento particular y son usados para probar la precisión de los diagramas de clases. (Diagramas del UML , sf)

Como se muestra en la figura 20, los objetos son representados con un rectángulo que contiene el nombre del objeto y su clase subrayadas separadas por puntos mientras que los atributos se listan en la parte inferior con un valor asignado.

Figura 20. Diagrama de Objetos

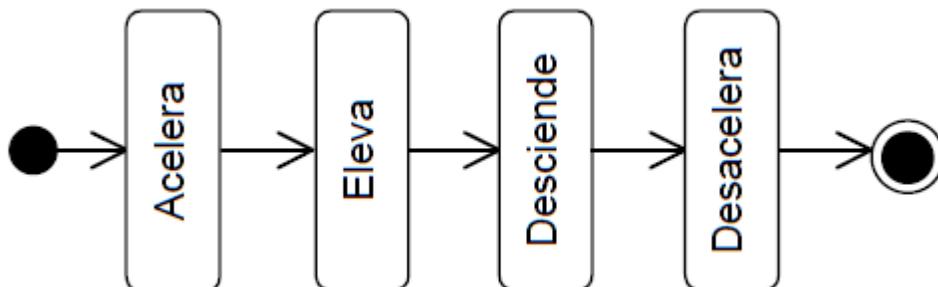


Fuente: Autor desconocido.

2.6.8 Diagramas de estados

Toma como referencia el estado en el que están los objetos en un momento determinado. Por ejemplo, un auto puede estar detenido o en movimiento. El este diagrama tipo de diagrama UML captura esta pequeña realidad. La Figura 21 muestra un ejemplo de un diagrama de estado.

Figura 21. Diagrama de estado

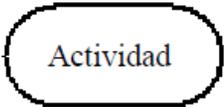
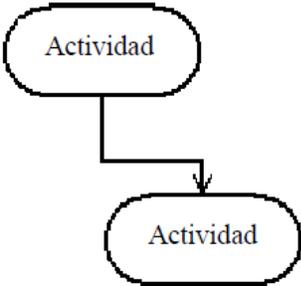
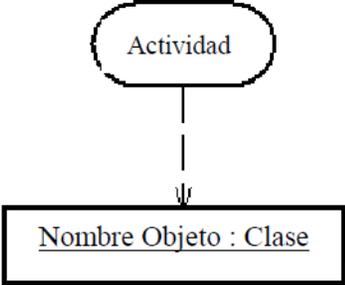
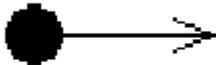
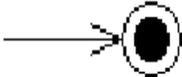


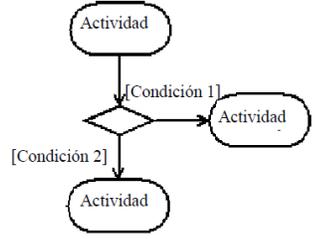
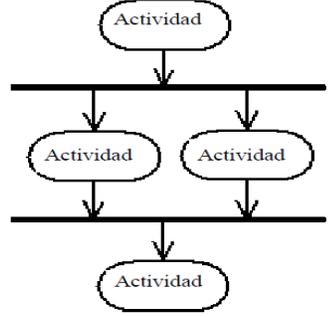
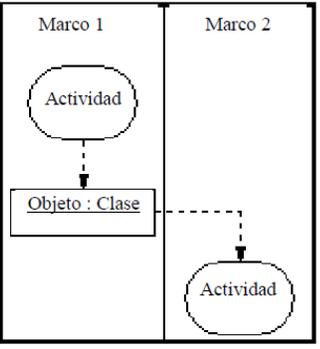
Fuente: Autor desconocido.

2.6.9 Diagramas de actividades

Los diagramas de clases y los objetos representan información estática. No obstante, en un sistema funcional, los objetos interactúan entre sí, y tales interacciones suceden con el tiempo. El diagrama de secuencias UML muestra la mecánica de la interacción con base en tiempos. (Diagramas del UML , sf). En la figura 22 se explican los símbolos del diagrama de actividades con su función.

Figura 22. Simbología del diagrama de actividades

Símbolo	Nombre	Función
	Estados de acción	Representan las acciones no interrumpidas de los objetos.
	Flujo de acción	Ilustran la relaciones entre los estados de acción.
	Flujo de objetos	Una flecha de flujo de objeto, desde una acción está creando o influyendo sobre dicho objeto. Una flecha de flujo de objeto, desde un objeto a una acción, indica que el estado de acción utiliza dicho objeto.
	Estado inicial	Estado inicial de un estado de acción
	Estado final	Estado final de un estado de acción

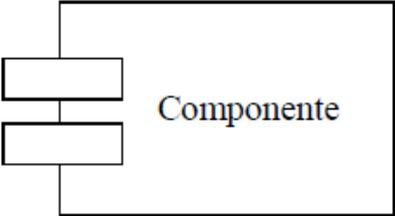
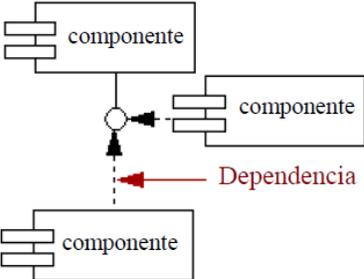
Símbolo	Nombre	Función
	Ramificación	Un rombo representa una decisión con caminos alternos. Las salidas alternativas deben de estar etiquetadas con una condición
	Sincronización	Una barra de sincronización ayuda a ilustrar la ocurrencia de transiciones paralelas, así quedan representadas las acciones concurrentes.
	Marcos de responsabilidad	Agrupar las actividades relacionadas en una misma columna.

Fuente: Creación propia

2.6.10 Diagrama de componente

Un diagrama de componentes describe la organización de los componentes físicos de un sistema. La figura 23 muestra la simbología del diagrama de actividades.

Figura 23. Simbología del diagrama de componentes

Símbolo	Nombre	Función
	Componente	Un componente es un bloque de construcción física del sistema.
	Interface	Describe a un grupo de operaciones usada o creada por componentes.
	Dependencias	Las dependencias entre componentes se grafican usando flechas de puntos.

Fuente: Creación propia.

Capítulo III: Marco Metodológico

3.1 Tipo y enfoque de la investigación

En esta sección se explicará la metodología con que se llevará a cabo este proyecto, además de la manera en que se ha obtener los datos de la investigación.

3.1.1 Tipo de investigación

Esta investigación se basa en la práctica de la investigación aplicada, la cual según (Cordero & Rosa, 2009, p. 159) "... se caracteriza por que busca la aplicación o utilización de los conocimientos adquiridos, a la vez que se adquieren otros, después de implementar y sistémica la practica basada en la investigación. La investigación aplicada es el método que se utilizara con el propósito de poner en práctica conceptos adquiridos referentes a la gestión de proyectos y el desarrollo de un sistema de gestión de información que sustente la necesidad de un centro de formación técnica.

3.1.2 Enfoque de la investigación

El enfoque de la investigación será cualitativo, que según (Sampieri, Collado, & Lucio, 2014, p. 358) ... "Se selecciona cuando el propósito es examinar la forma en que los individuos perciben y experimentan los fenómenos que los rodean, profundizando en sus puntos de vista, interpretaciones y significados.

Los planteamientos cualitativos son una especie de plan de exploración (entendimiento emergente) y resultan apropiados cuando el investigador se interesa en el significado de las experiencias y los valores humanos, el punto de vista interno e individual de las personas y el ambiente natural en que ocurre el fenómeno estudiado, así como cuando se busca una perspectiva cercana de los participantes. (Sampieri, Collado, & Lucio, 2014, p. 364)

3.2 Fuentes de información

Las fuentes de información primarias y secundarias representan el origen de los principales datos de una investigación.

3.2.1 Fuentes de información primaria

Las fuentes primarias ofrecen un relato de primera mano sobre un acontecimiento o un periodo de tiempo y se consideran fidedignas.

Representan el pensamiento original, informan sobre descubrimientos o acontecimientos, o pueden compartir información nueva. A menudo, estas fuentes se crean en el momento en que ocurren los hechos, pero también pueden incluir fuentes creadas posteriormente. Suelen ser la primera aparición formal de una investigación original. En otras palabras, las fuentes primarias te dan acceso directo al tema que estás investigando o aprendiendo; contienen información en bruto. (testsiteforme, 2022)

Según (D'Aquino & Barrón, 2007, p. 51) " datos primarios: cuando los datos o hechos son recogidos por aquellos que investigan directamente la realidad."

Existen las fuentes primarias que se refiere a entrevistas con los involucrados. En la siguiente sección se amplía un poco más el concepto de la entrevista.

3.2.2 Entrevista

La entrevista es un método de sensibilidad y poder únicos para captar las experiencias y los significados vividos del mundo cotidiano de los sujetos. Las entrevistas permiten a los sujetos expresar a otros su situación desde su propia perspectiva y en sus propias palabras. (Kvale, 2011, p. 28)

La entrevista es una conversación que tiene una estructura y un propósito determinados, por una parte: el entrevistador. Es una interacción profesional que va más allá del intercambio espontáneo de ideas como en la conversación cotidiana y se convierte en un

acercamiento basado en el interrogatorio cuidadoso y la escucha con el propósito de obtener conocimiento meticulosamente comprobado. (Kvale, 2011, p .29)

La entrevista pretende conseguir relatos matizados de diferentes aspectos del mundo de la vida del entrevistado; trabaja con palabras y no con números. En las entrevistas cualitativas, la precisión en la descripción y la rigurosidad en la interpretación del significado se corresponden con la exactitud en las mediciones cuantitativas. (Kvale, 2011, p. 33)

3.2.3 Fuentes secundarias

Según (D'Aquino & Barrón, 2007, p. 51) “Se usan resultados de otras investigaciones, realizados por otros investigadores, y se los analiza o interpreta.”

Una fuente secundaria puede ser un libro, un artículo de revista o un documento digital o impreso creado por alguien que no participó directamente en los eventos. En esta sección se mencionarán las fuentes secundarias usadas en la investigación.

3.2.4 Libro electrónico

Los libros digitales son en realidad archivos de texto que incluyen algunas características avanzadas que facilitan la lectura: índice interactivo para saltar a capítulos específicos, hipervínculos a notas (fundamentales para estudio e investigación), resaltado y la posibilidad de agregar comentarios personales en cualquier parte. (ICBC, 2021)

3.2.5 *Páginas Web*

Una página de Internet o página Web es un documento electrónico adaptado particularmente para el Web, que contiene información específica de un tema en particular y que es almacenado en algún sistema de cómputo que se encuentre conectado a la red mundial de información denominada Internet, de tal forma que este documento pueda ser consultado por cualesquier persona que se conecte a esta red mundial de comunicaciones y que cuente con los permisos apropiados para hacerlo. (Informatica milenium, 2022)

3.2.6 *Sujetos de información*

Personal de la Fundación Hedwig y Robert Samuel encargados de la asistencia y la dirección académica. Ver tabla 3.

Tabla 3. Sujetos de información

Puesto	Laboral o Descripción general	Profesión u Oficio	Experiencia	Relación con el Tema
Directora académica	Master en administración educativa	<p>Dirigir, ejecutar y asesorar para el desarrollo de programas y proyectos que dirigen y apoyan la labor del personal.</p> <p>Implementar y supervisar la normativa, reglamentos y manuales vigentes en la Fundación.</p> <p>Brindar técnicas de orientación personal, social, educativa y vocacional del centro educativo.</p> <p>Dirigir, coordinar, evaluar y controlar las actividades académicas que se realizan en la Fundación.</p> <p>Seleccionar y contratar el personal</p>	<p>Envía los documentos con los formatos oficiales de la fundación, además recibe la información de los profesores y las asistentes. hace anotaciones en los folders de los estudiantes.</p> <p>Hace los reportes de ausencias, tardías y notas a la sede de Alemania</p>	

Puesto	Laboral o	Profesión u	Experiencia	Relación con el Tema
Descripción	Oficio			
general				
			del área académica.	
			Realizar proceso de inducción al personal nuevo.	
			Velar por el desarrollo y cumplimiento de los programas de estudio, planeamientos, pruebas mensuales, pruebas de medio periodo, finales y lo relacionado al área académica. Retroalimentar al personal a cargo para que se logren las metas y objetivos, tanto a nivel de	
			competencias, como a nivel de conocimientos técnicos. Entre otras funciones. 13 años en el puesto.	

Puesto	Laboral o Descripción general	Profesión u Oficio	Experiencia	Relación con el Tema
Asistente de dirección académica	Técnica de asistente en administración	<p>Elaboración actas de inicio, finalizaciones de programas acreditados por el INA y títulos INA, además de la coordinación del envío de estos para firma ante la entidad.</p> <p>Llevar el control de asistencia de los estudiantes diariamente e informar a la orientadora de los ausentes del día.</p> <p>Dar seguimiento a situaciones de ausencia, conducta, tardías de estudiantes cuando sea necesario.</p> <p>Apoyar en la recepción en atención telefónica, atención a visitas y todo lo que se requiera en el momento.</p> <p>Impresión y custodia de los</p>	<p>Recibe los documentos de la directora académica para imprimirlos o registrar la información obtenida en otros documentos.</p>	

Puesto	Laboral o	Profesión u	Experiencia	Relación con el Tema
Descripción	Oficio			
general				
			<p>exámenes mensuales, pruebas de medio periodo y finales</p> <p>Realizar el trámite para el depósito de los subsidios de transporte de los estudiantes que tengan el beneficio.</p> <p>Elaboración y actualización de la base de datos de la generación actual.</p> <p>Apoyar en la digitación de la información de la base de datos de los aspirantes desde información de la solicitud hasta las calificaciones obtenidas en las pruebas por parte de los aspirantes.</p>	

Puesto	Laboral o Descripción general	Profesión u Oficio	Experiencia	Relación con el Tema
Secretaria de dirección académica	Técnica de asistente en administración	<p>Atender a compañeros, estudiantes y público en general, con el fin de evacuar consultas generales y ponerlo en contacto con quien corresponda.</p> <p>Velar por el buen funcionamiento y uso de equipos e instrumentos que utiliza en el desarrollo de sus actividades, reportando cualquier anomalía o daño importante que se presente al superior inmediato, tales con video beam, Laptops.</p> <p>Apoyo con facturación electrónica y seguimiento de las cuentas por pagar, relacionado con la practica supervisada de los estudiantes, esto por un lapso de tres meses.</p> <p>Elaboración de base de datos.</p>	<p>Registra la información de los estudiantes en hojas electrónicas que coloca en folders para luego entregarlos a la dirección académica. Se encarga de registrar los pagos de los estudiantes.</p>	

Puesto	Laboral o Descripción general	Profesión u Oficio	Experiencia	Relación con el Tema
			<p>Archivar información de documentos de estudiantes.</p> <p>Entrega y recepción de formularios para beca. 6 años en el puesto</p>	

Fuente: Creación Propia

3.3 Técnicas y herramientas de recolección de datos

Según, (Sampieri, Collado, & Lucio, 2014, p. 198) “Recolectar datos implica elaborar un plan detallado de procedimientos que nos conduzcan a reunir datos con un propósito específico”. En esta investigación se utilizarán métodos de recolección de datos cualitativos.

Según, (Álvarez, 2011, p. 149) “Los estudios que utilizan datos de tipo cualitativo emplean métodos de recolección que difieren de aquellos que se utilizan en estudios cuantitativos.”

Para el proyecto de investigación, estos métodos se aplicaron en la etapa de recolección de requerimientos, incluyendo a un grupo de personas que forman parte del personal y son detallados en la sección de sujetos de información.

3.3.1 *Entrevista no estructurada y entrevista dirigida*

La entrevista no estructurada es flexible y abierta, en ella se procede sin un concepto preconcebido del contenido o flujo de información que se desea obtener, aunque los objetivos de la investigación rigen las preguntas. El contenido, orden, profundidad y formulación se hallan sujetos al criterio del entrevistador. Las entrevistas no estructuradas se efectúan mediante conversaciones y en medios naturales. Su objetivo es captar la percepción del entrevistado, sin imponer la opinión del investigador. Si bien el investigador, sobre la base del problema, los objetivos y las variables, elabora las preguntas antes de realizar la entrevista, modifica el orden, la forma de encauzar las preguntas o su formulación para adaptarlas a las diversas situaciones y características particulares de los sujetos de estudio.

Las entrevistas dirigidas son semiestructuradas y en ellas se usa una lista de áreas hacia las que hay que enfocar las preguntas, es decir, se utiliza una guía de temas. El entrevistador permite que los participantes se expresen con libertad con respecto a todos los temas de la lista y registra sus respuestas (con frecuencia mediante grabadora). En lugar de hacer preguntas tomadas directamente de un cuestionario, el investigador procede a un interrogatorio partiendo

de un guion de tópicos o un conjunto de preguntas generales que le sirven de guía para obtener la información requerida. (Álvarez, 2011, p. 149)

3.3.2 *La entrevista en profundidad*

Todas las entrevistas comparten una estructura básica en la que el investigador tiene las preguntas y el sujeto da las respuestas. Sin embargo, las entrevistas cualitativas, a diferencia de las estructuradas, siguen el modelo de una conversación entre iguales, donde el propio investigador es el instrumento y no lo que está escrito en el papel: éste avanza lentamente al principio, intenta establecer un punto inicial, plantea preguntas no directivas, etc. aunque esto se lleve a cabo en situaciones preparadas. (Álvarez, 2011, p. 153)

3.3.3 *Observación*

Su objetivo es comprender el comportamiento y las experiencias de las personas como ocurren en su medio natural. Por lo tanto, se intenta observar y registrar información de las personas en sus medios con un mínimo de estructuras y sin interferencia del investigador. (Álvarez, 2011, p. 153)

3.3.4 *Observación simple*

“En la observación simple, no regulada o no controlada sólo se tienen unos lineamientos generales para la observación sobre aspectos del fenómeno que el investigador tiene interés en conocer.” (Álvarez, 2011, p 153)

3.3.5 Documentos existentes

Se utilizará la información existente en el centro de formación técnica. La Fundación cuenta con documentación definida, como reglamentos y estructura de documentos.

3.4 Variables

En la tabla 4, se detallan las variables del proyecto.

Tabla 4. Definición de variables.

Objetivos específicos	Variables asociadas	Descripción
Establecer los requerimientos para la elaboración de un sistema de gestión de datos.	Requerimientos funcionales y no funcionales.	Se documenta los requerimientos del sistema, los que serán la base del desarrollo del sistema.
Diseñar el sistema académico de gestión de datos.	Diseño de casos de uso.	Se diseña las vistas de usuario y los casos de uso que permitan visualizar la interacción de los usuarios con el sistema.
Desarrollo de la aplicación y cada uno de los módulos según los requerimientos.	Desarrollo del sistema.	Se desarrolla el sistema de gestión de información.
Realizar pruebas del sistema de gestión de información.	Protocolo de pruebas.	Se documentan las pruebas así como el resultado de ellas.
Implementar el sistema de gestión de información	Instalación del sistema en las computadoras.	Utilizar algunos puestos de trabajo para corroborar que cumpla con lo necesario.

Fuente: Creación propia.

3.5 Diseño de la investigación

En la figura 24, se representan las etapas del desarrollo del proyecto.

Figura 24. Etapas del desarrollo del proyecto



Fuente: Creación propia.

A Continuación, la descripción de las etapas:

Etapa de análisis: Observar las actividades que realizan durante las pruebas del *software*. Se elabora el documento de requerimientos del sistema de gestión de información. Se utilizan la técnica de obtención de datos.

Etapa de diseño: Se diseñan los casos de uso, se prepara el ambiente de desarrollo. La herramienta en esta etapa es el diagrama de casos de uso.

Etapa de desarrollo: Esta es la etapa de construcción, se construye la base de datos y el sistema como tal. En esta etapa las herramientas fueron los lenguajes de programación y los IDE de desarrollo.

Etapa de pruebas: Se elabora el protocolo de pruebas y se planifica una sesión con los usuarios finales para la ejecución del protocolo. Es el protocolo de pruebas la herramienta usada en esta etapa.

Etapa de implementación: Se crea el documento técnico del sistema y se instala el sistema en el ambiente de producción. Como herramienta en esta etapa se usa el documento técnico del sistema.

3.6 Matriz de coherencia.

La tabla 5 muestra la matriz de coherencia, que tiene como finalidad demostrar la relación del objetivo, entregable, instrumentos y temas del marco teórico del proyecto.

Tabla 5. Matriz de coherencia

Objetivo	Entregable	Fase, o etapa del proyecto	Técnicas o Métodos de recolección de la información	Instrumento	Temas relacionados para el marco teórico
Establecer los requerimientos para la elaboración de un sistema de gestión de información para la Fundación Samuel	Documentación de los requerimientos para la elaboración del sistema de gestión de información	Etapa de análisis	Entrevista, observación y uso de documentos existentes.	Minutas	Análisis Requerimientos funcionales Requerimientos no funcionales Documentación
Desarrollar la programación del sistema de gestión de información	Base de datos, sistema con los requerimientos especificados	Etapa de desarrollo	Desarrollo de la base de datos y del sistema de gestión de información	SQL Server Visual studio	Construcción de la base de datos IDE Visual Studio Lenguaje de programación C# Microsoft SQL Server Desarrollo en cascada
Realizar pruebas del sistema de	Protocolo de pruebas	Etapa de pruebas	Protocolo de pruebas	Protocolo de pruebas	Pruebas de error de <i>software</i>

Objetivo	Entregable	Fase, o etapa del proyecto	Técnicas o Métodos de recolección de la información	Instrumento	Temas relacionados para el marco teórico
gestión de información					
Implementar el sistema de gestión de información	Documento técnico del sistema. Sistema funcionando en el ambiente de producción	Etapa de implementación	Documento técnico del sistema	Documento técnico del sistema	Puesta en producción

Capitulo IV: Diagnostico de la situación actual

El objetivo de este capítulo es representar el estado actual de los procesos que intervienen entre sí para la obtención, registro, edición y consulta de la información de los estudiantes de la Fundación Hedwig y Robert Samuel.

4.1 Diagnostico Administrativo U Operativo

Actualmente el área de la dirección académica no posee un sistema informático que les permita manipular la información de los estudiantes y obtener resultados de forma rápida y eficiente. El proceso para registrar la información se hace de forma manual en hojas de Excel, los cuales dependiendo del tipo documento (notas, tardías, ausencias, otros) pasa por varias personas para ser revisado, editado o archivado. Los documentos digitales se almacenan en un servidor, mientras que la información de los estudiantes se guarda en hojas dentro de carpetas para poder consultarlos y si es necesario agregar notas.

Los siguientes procesos son los que utilizan para obtener y registrar los datos de los estudiantes:

- **Registro de la información de los estudiantes y Rendimiento Académico:** La información de los estudiantes es obtenida de la boleta de solicitud de matrícula. La secretaria de la dirección académica es la encargada de solicitar a la asistente de la dirección académica las boletas de solicitud de matrícula de los aspirantes a los que se les aprobó la beca y crear un archivo de Excel que incluye: El nombre del estudiante, la escolaridad, dirección, y si es solicitante de subsidio de transporte, entre otros varios datos. (Anexo 1).

A diferencia de otros documentos, este no se guarda en el servidor, es impreso y se le ajunta una fotografía del estudiante. Los documentos se guardan en folders separados por especialidad y se le entrega a la directora académica que se encarga de hacer anotaciones y registra los meses que aprobaron o no los estudiantes. (Anexo 2)

- **Registro de las calificaciones de los estudiantes:** Los profesores de cada una de las especialidades y asignaturas registran las notas en un archivo de Excel que al completarse es enviado a la dirección académica por correo electrónico. La directora revisa y hace ajustes de ser necesario. La directora académica se encarga de aprobar a los profesores

para guardar este archivo en una carpeta de sus computadoras ligadas al servidor local, además envía el archivo por correo electrónico a la asistente de la dirección académica para que las imprima y se las entregue a los profesores en las reuniones de rendimiento académico. (Anexo 3)

4.2 Diagnostico técnico

La siguiente sección se enfocará en describir las especificaciones técnicas que se estarán usando e implementando en el desarrollo del *software* para la gestión de información de los estudiantes de la de fundación Hedwig y Robert Samuel.

4.2.1 Equipo de computación y servidor local

La Fundación Samuel cuenta con varios equipos de cómputo, distribuidos entre profesores, personal administrativo y la dirección académica. Además, la información se guarda en un servidor local. La tabla 6 muestra las características de los equipos de cómputo y el servidor.

Tabla 6. Características de equipo de computo

Ubicación	Cantidad	Especificaciones técnicas	Especificaciones del sistema operativo
Profesores	5	Procesador: Intel (R) Core (TM) i5-2500 CPU 3.30 GHz RAM instalada: 4.00 GB (3.88 GB usable) Tipo de sistema: Sistema Operativo de 64 bits	Windows 10 Pro Versión 21H2

Ubicación	Cantidad	Especificaciones técnicas	Especificaciones del sistema operativo
Personal administrativo	3	Procesador: Intel (R) Core (TM) i5-2500 CPU 3.30 GHz RAM instalada: 4.00 GB (3.88 GB usable) Tipo de sistema: Sistema Operativo de 64 bits	Windows 10 Pro Versión 21H2
Dirección académica	2	Procesador: Intel (R) Core (TM) i5-2500 CPU 3.30 GHz RAM instalada: 4.00 GB (3.88 GB usable) Tipo de sistema: Sistema Operativo de 64 bits	Windows 10 Pro Versión 21H2
Servidor Local	1	Procesador: Intel (R) CPU ES-2420 1.90 GHz Memoria Ram Instalada: 16.0 GB Disco duro: 2 Teras Tipo de sistema: Sistema Operativo de 64 bits	Windows Server 2012 R2 Standar

Fuente: Creación Propia

4.2.2 Lenguaje de programación

Para la construcción del sistema informático se utilizará el lenguaje de programación C#. C# es un lenguaje de programación *orientado a componentes*, orientado a objetos.

4.2.3 IDE de desarrollo

Los entornos de desarrollo (IDE) son las herramientas con las cuales los programadores crean aplicaciones. El IDE de desarrollo utilizado será Visual Studio Community 2019.

4.2.4 *Motor de base de datos*

Para el diseño de la base de datos se utilizará el gestor de base de datos Microsoft SQL server.

4.3 Diagnóstico de percepción

4.3.1 *Herramientas aplicadas sobre el recurso humano*

Las herramientas de recolección de datos utilizadas en este proyecto sobre el recurso humano fueron la observación y la encuesta. La observación se realizó en los puestos de trabajo de las partes involucradas dejando como resultado la forma en que se realizan los procesos. Las encuestas fueron enfocadas de igual forma a los 4 puestos que intervienen entre sí para la toma y registro de la información, esta herramienta tiene como objetivo demostrar cuales funciones son realizadas por varias personas de forma directa o indirecta, que herramientas utilizan y tener el panorama de la necesidad de un sistema informático que ayude a gestión la información.

4.3.2 *Resultado del instrumento*

A continuación, se representan los resultados del instrumento que se utilizó inicialmente para obtener un diagnóstico inicial.

Se consultó por el puesto de trabajo y el tiempo de laborar en la Fundación Samuel. La tabla 7 muestra los resultados.

Tabla 7. Puestos de trabajo de las personas entrevistadas y los años de trabajar en la Fundación.

Puesto que desempeña en la fundación	Años de trabajar en la fundación
Directora académica	13
Asistente de dirección académica	7
Asistente contable	6
Secretaria de dirección académica	6

Fuente: Creación propia

A continuación, se muestran las preguntas realizadas y sus respuestas.

- 1. Marque en cuál o cuáles de las siguientes actividades está relacionado (a) y si es de forma directa o indirecta. Las respuestas se representan en la tabla 8.**

Tabla 8. Respuesta de la pregunta número 1 de la herramienta de diagnóstico

Actividades	Directora académica	Asistente de dirección académica	Asistente de contabilidad	Secretaria de dirección académica
	Participación	Participación	Participación	Participación
a. Registro, archivo, modificación y búsqueda de información relacionada con los estudiantes como nombre, edad, domicilio, entre otras.	directa	directa	indirecta	directa

b. Registro, archivo, modificación y búsqueda de información académica de los estudiantes como notas, tardías, ausencias entre otras.	Participación directa	Participación directa	Participación directa	Participación directa
c. Diseño de informes a partir de información obtenida de otros.	Participación directa	Participación directa	Participación directa	Participación directa
d. Cobros y facturación	Participación indirecta	Participación directa	Participación indirecta	Participación directa

Fuente: Creación propia

2. De las actividades anteriores, describa las que usted realiza, las herramientas que utiliza (Excel, Power Bi, SQL...), y si existe algún procedimiento, reglamento, formato o parámetro que deba cumplir

- **Respuesta de la directora académica:** Las herramientas utilizadas son en Excel cumpliendo con los formatos establecidos.
- **Respuesta de la asistente de la dirección académica:** Excel, Word, a la hora de pasar la información, los nombres están mal escritos o los números de cédulas son incorrectos.
- **Respuesta de la asistente contable: del sistema contable actual (NOVA),** permite a los usuarios crear un perfil a los y las estudiantes (fotografía, nombre completo, cédula, número de teléfono, domicilio y correo electrónico). Este perfil permite que el usuario contable, pueda emitir y enviar un estado de cuenta a cada cliente al correo electrónico con el saldo actual y el desglose de los abonos efectuados. Por otro lado, para un control más efectivo, se utiliza un auxiliar de cuentas por cobrar en Excel, con la finalidad de que, al realizar el cierre mensual, se compare el auxiliar con la cuenta contable. Con referencia al diseño de

informes, esté se realiza mensualmente. Estos informes principalmente son de los ingresos y gastos emitidos durante el mes, y son confeccionados en Excel y Word. De acuerdo con facturación y cobros únicamente se realiza el registro de este y la revisión y comparación con auxiliares al finalizar el cierre.

- **Respuesta de la secretaria de la dirección académica:** Con Excel control de uniformes.

3. ¿Ha encontrado o generado algún tipo de error en las labores que realiza? En el caso de ser afirmativa, describa los problemas

- **Respuesta de la directora académica:** Si, Hay muchos documentos, que para completar uno se debe recurrir a otros documentos; lo cual hace que la labor sea más lenta y engorrosa.
- **Respuesta de la asistente de la dirección académica:** Si, en las edades o en los nombres de los estudiantes.
- **Respuesta de la asistente contable:** Si, El sistema contable actual, es un sistema que tiene aproximadamente tres meses de utilizarse en la Fundación, por lo tanto, se han encontrado errores a nivel de sistema (Mal cálculo de porcentajes, distribución por centros de costos, errores en planilla, erros en facturación), de las cuales se han tenido que realizar modificaciones por la estructura contable que desarrolla la Fundación. Además, la falta de conocimiento y experiencia han permitido que a nivel contable existan deficiencias y mal registro de información.
- **Respuesta de la secretaria de dirección académica:** Si, Por ejemplo al momento de realizar un proceso, se requiere de 2 o más personas.

4. ¿Considera que un sistema informático podría mejorar sus funciones?

- **Respuesta de la directora académica:** Si
- **Respuesta de la asistente de la dirección académica:** Si
- **Respuesta de la asistente contable:** Si
- **Respuesta de la secretaria de la dirección académica:** Si

Brechas o conclusiones

Con base en la información anterior se presenta en esta sección el estado actual de la Fundación Samuel, la situación deseada y la brecha existente.

Los funcionarios de la Fundación Samuel nunca han utilizado un sistema académico que les permita manipular información de forma rápida, ágil y segura, única forma en la que han manipulado información es en hojas digitales. Este método ha generado errores no solo en el ingreso de la información, sino también en la búsqueda o la actualización de los datos. Los encargados de las áreas administrativas comparten información o alimenta otros documentos con la información obtenida por alguien más, esto ha generado errores de inconsistencia.

La fundación obtuvo en un sistema contable que les permite entre otras cosas llevar un control de los cobros que se realizan a los estudiantes y los saldos pendientes, sin embargo, este sistema presenta errores en ciertos cálculos y por este motivo se sigue registrando información en hojas electrónicas como un tipo de respaldo.

4.3.3 Situación deseada

Contar con un sistema de gestión de información adecuado, que cumpla con la necesidad de los usuarios, obteniendo información importante basada en sus necesidades y experiencia en el puesto que desempeñan. Un proceso correcto de recolección de requerimientos permitirá cumplir con la mayor parte de las necesidades de los usuarios.

4.3.4 Brecha

La fundación Samuel debe tener un sistema informático para asegurar que los procesos de manipulación de información se realizan de forma confiable y segura cumpliendo con las necesidades diarias de los usuarios y que permita reducir los errores humanos y el coste de un trabajo manual.

Se necesita que este sistema informático permita gestionar toda la información relativa a los estudiantes con el objetivo de realizar actividades administrativas que permitan medir y mejorar la calidad del servicio ofrecido en la Fundación Samuel.

Capítulo V: Propuesta del proyecto

El objetivo de este capítulo es el de representar el diseño propuesto del sistema gestor de información para la Fundación Hedwig Y Robert Samuel, usando la información obtenida del análisis y de las herramientas de investigación y obtención de datos.

5.1 Requerimientos

5.1.1 Identificación de actores

A continuación, se indican los actores que utilizarán el sistema de gestión de información que se pretende desarrollar:

- Administración
- Dirección académica
- Asistente de dirección académica
- Secretaria de dirección académica
- Profesores

5.1.2 Requerimientos funcionales

Tabla 9. RF01 Inicio de sesión

Código de requerimiento	Descripción	Objetivo	Prioridad
RF01	<p>Desarrollar un formulario de inicio de sesión con los campos: Email. Este campo es el que se use como usuario</p> <p>Contraseña: Este campo debe de ser de tipo contraseña para que este oculta. Se debe validar lo siguiente: Que el usuario y la contraseña coincidan con la registrada en el sistema, de no ser así se debe de mostrar un mensaje que lo indique.</p>	Controla el ingreso al sistema por medio de un formulario de autenticación	Alta

Código de requerimiento	Descripción	Objetivo	Prioridad
	Si el usuario y la contraseña coinciden el usuario podrá ingresar.		

Fuente: Creación propia

Tabla 10. RF02 Cerrar sesión

Código de requerimiento	Descripción	Objetivo	Prioridad
RF02	Desarrollar una opción en el menú que permita cerrar la Sesión del usuario. Al presionar <i>cerrar sesión</i> se debe de salir del sistema y regresar al módulo de inicio de Sesión.	Permitir al usuario cerrar la Sesión y salir del sistema	Alta

Fuente: Creación propia

Tabla 11. RF03 Agregar usuario

Código de requerimiento	Descripción	Objetivo	Prioridad
RF03	Se creará un módulo de <i>mantenimiento de usuarios</i> para que el usuario registrado como administrador pueda registrar un nuevo usuario. El sistema debe validar que los datos son corrector y están completos, además de que el usuario no exista ya en el sistema, de ser así se mostrara un mensaje de registro exitoso, de lo contrario se mostrara un mensaje de error de registro.	Permitir al Administrador el registro de un nuevo usuario.	Alta

Fuente: Creación propia

Tabla 12. RF04 Actualizar usuario

Código de requerimiento	Descripción	Objetivo	Prioridad
RF04	Se creará un módulo de <i>mantenimiento de usuarios</i> para que el usuario registrado como administrador pueda actualizar un usuario. El sistema debe validar que los datos son corrector y están	Permitir al Administrador actualizar el registro de un usuario existen.	Alta

Código de requerimiento	Descripción	Objetivo	Prioridad
	completos, de ser así se mostrara un mensaje de registro actualizado, de lo contrario se mostrara un mensaje de error de actualización de registro.		

Fuente: Creación propia

Tabla 13. RF05 Pantalla principal

Código de requerimiento	Descripción	Objetivo	Prioridad
RF05	Se creará un formulario que funcionará como pantalla principal con un menú de acuerdo a los roles del usuario. Esta pantalla debe tener el formato que usa la institución. Dicha información está en el manual de la institución.	Crear una pantalla principal que muestre a los usuarios las opciones a las que puede acceder	Alta

Fuente: Creación propia

Tabla 14. RF06 Registro de estudiantes

Código de requerimiento	Descripción	Objetivo	Prioridad
RF06	Este formulario al que se accede desde el menú, está diseñado para registrar la siguiente información de los estudiantes: Nombre y apellidos. Número de identificación. Una fotografía del estudiante. Las imágenes se cargarán desde una carpeta especificada por el sistema. Dirección. Nivel académico. Además, se asignará un espacio para anotaciones de la dirección académica. Se debe de contar con formularios que permita editar cualquiera de los registros, y un mantenimiento preciso para los estudiantes del sistema. Se debe de evitar la repetición de estudiantes y que la información suministrada cumpla con los requisitos básicos como formatos de entrada.	Crear un formulario para el registro de la información de los estudiantes.	Alta

Código de requerimiento	Descripción	Objetivo	Prioridad
-------------------------	-------------	----------	-----------

Tabla 15. RF07 Actualizar estudiante

Código de requerimiento	Descripción	Objetivo	Prioridad
RF07	Desde el formulario de registro de estudiantes, se debe seleccionar el registro que se quiere actualizar. Se activarán los campos que se requieran actualizar, y al presionar el botón <i>actualizar</i> se guardarán los cambios. Si la información es correcta y está completa, se desplegará un mensaje indicándolo, de no ser así, el sistema lo indicará mediante un mensaje de error.	Crear la funcionalidad para la actualización de la información del estudiante.	Alta

Fuente: Creación propia

Tabla 16. RF08 Eliminar, inhabilitar registro estudiante

Código de requerimiento	Descripción	Objetivo	Prioridad
RF08	Desde el formulario de registro de estudiantes, se debe seleccionar el registro que se quiere eliminar o inhabilitar, en este momento se activará el botón eliminar y el botón inhabilitar. Al eliminar se o inhabilitarse el registro se mostrará un mensaje en la pantalla. Desde se debe permitir seleccionar varios registros a la vez o uno solo, además de mostrarlos todos o buscar aplicando filtros.	Crear la funcionalidad para eliminar o inhabilitar el registro de un estudiante.	Alta

Fuente: Creación propia

Tabla 17. RF09 Registro de notas

Código de requerimiento	Descripción	Objetivo	Prioridad
RF09	Se creará un módulo donde se pueda ingresar la nota de los estudiantes de una carrera en específico. Cada profesor de especialidad deberá poder generar un formulario de calificación por mes con el desglose de la nota que se le apruebe para ese mes. El sistema deberá validar que todos los campos están completos y dar un mensaje cuando se termine el registro de las notas.	Desarrollo de un módulo que permita registrar las notas de los estudiantes por especialidad y de forma mensual.	Alta

Fuente: Creación propia

Tabla 18. FR10 Modificar notas

Código de requerimiento	Descripción	Objetivo	Prioridad
RF10	El módulo de ingreso de notas tendrá la opción de seleccionar los estudiantes para la modificación de las notas. Al presionar el botón <i>actualizar</i> se mostrará un mensaje de confirmación si los campos se llenaron de forma correcta y completa, de lo contrario se mostrará un mensaje de error.	Desarrollo de una funcionalidad que le permita al usuario actualizar las notas del estudiante.	Alta

Fuente: Creación propia

Tabla 19. RF11 Control de asistencia

Código de requerimiento	Descripción	Objetivo	Prioridad
RF11	Crear un módulo de asistencia, por normas internas deber ser usado solo por la asistente de la dirección académica. Las opciones de asistencia serán: ausente, tarde o presente.	Desarrollo de un módulo que permita registrar el estado de la asistencia de los estudiantes	Alta

Fuente: Creación propia

5.1.3 Requerimientos no funcionales

Tabla 20. RNF01 Estilo visual

Código de requerimiento	Descripción	Prioridad
RNF01	El aspecto de sistema debe ser consistente en todos sus módulos, además de intuitivo hacia el usuario.	Alta

Fuente: Creación propia

Tabla 21. RNF02 Seguridad

Código de requerimiento	Descripción	Prioridad
RNF02	La seguridad en el módulo de usuario debe de ser lo suficientemente confiable. La contraseña debe tener algún nivel de encriptación. La información de la base de datos será se debe de proteger para que son sea leída por terceros en caso de algún robo o filtración, esto por medio de la encriptación de la base de datos	Alta

Fuente: Creación propia

Tabla 22. RNF03 Restricción de contenido

Código de requerimiento	Descripción	Prioridad
RNF03	El acceso a cada módulo del sistema está determinada por el rol del usuario.	Alta

Fuente: Creación propia

Tabla 23. RNF04 Tiempo de respuesta

Código de requerimiento	Descripción	Prioridad
RNF04	El <i>software</i> debe manejar la información con fluidez con un tiempo óptimo de respuesta que no supere los 3 segundos.	Media

Fuente: Creación propia

Tabla 24. *RNF05 Confidencialidad*

Código de requerimiento	Descripción	Prioridad
RNF04	Toda la información otorgada por los usuarios se manipulará únicamente con fines de la corporativos y de manera responsable.	Alta

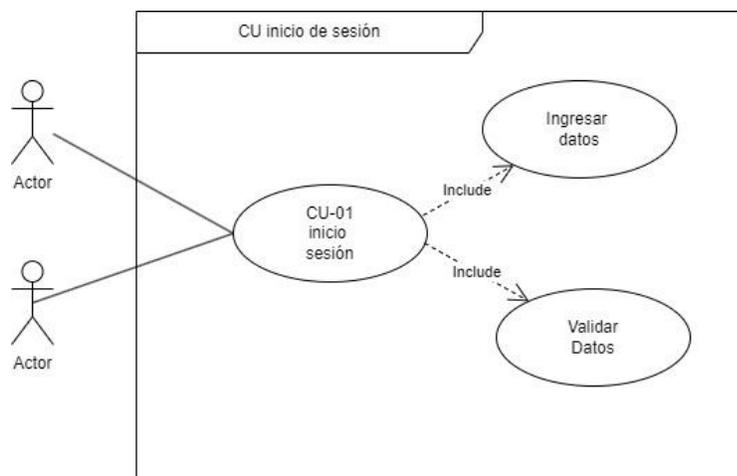
Fuente: Creación propia

5.1.4 *Diagramas de Casos de uso*

En esta sección se usarán casos de uso para expresar los requisitos funcionales de forma gráfica y secuencial.

5.1.5 Inicio de sesión

Figura 25. Diagrama de caso de uso de inicio de Sesión



Fuente: Creación propia

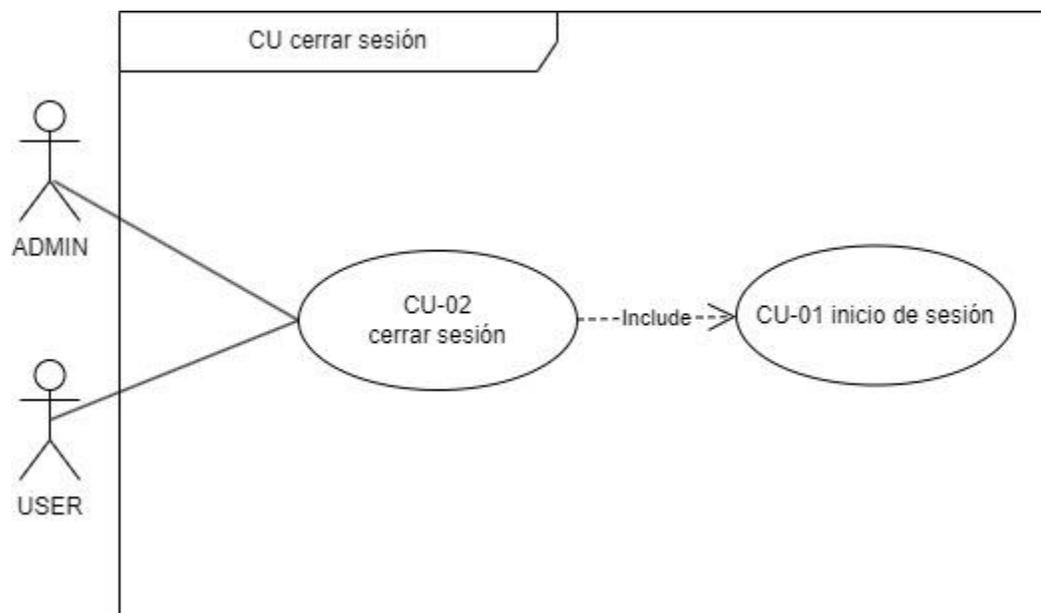
Tabla 25. Caso de uso inicio de Sesión

Caso de uso	Inicio de Sesión
Código de Caso de Uso	CU-01
Código de Requerimiento	RF01
Actor (es)	Administrador, Usuarios
Precondición	Los Usuarios deben de estar registrados en la base de datos y tener un perfil establecido
Postcondición	
Flujo básico	<ol style="list-style-type: none"> 1. El usuario Digita el correo y la contraseña y presiona el botón <i>ingresar</i>. 2. El sistema valida los datos. 3. Si los datos son correctos, se indica con un mensaje. 4. El usuario ingresa a la pantalla de inicio.
Flujo alternativo	<ol style="list-style-type: none"> 3. Si el usuario ingresado en el paso 1 ya está registrado el sistema mostrará un mensaje. 4. Si en el paso 1 el usuario ingresa información incorrecta o incompleta, se mostrará un mensaje indicando que el dato ingresado no es correcto o está incompleto.

Fuente: Creación propia

5.1.6 Cerrar sesión

Figura 26. Diagrama de caso de uso Cerrar Sesión



Fuente: Creación propia

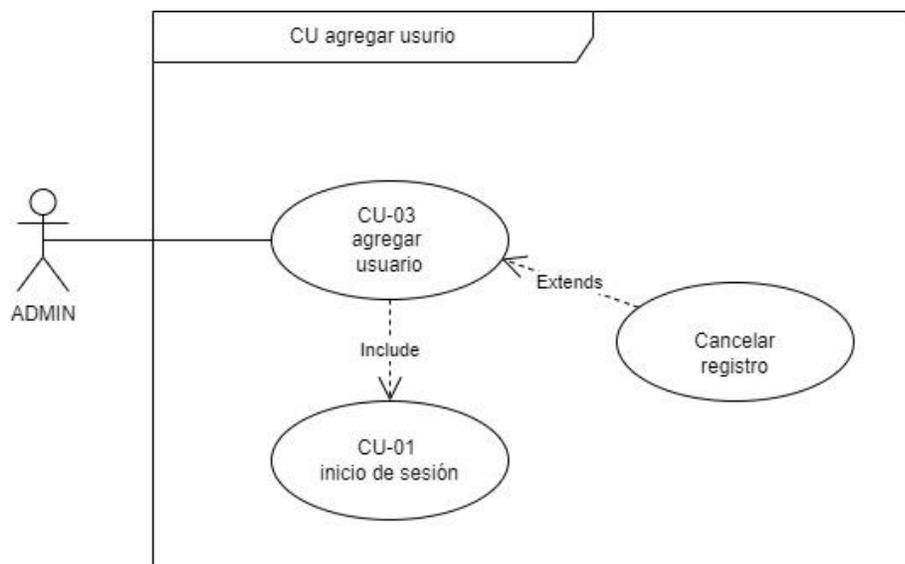
Tabla 26. Caso de uso Cerrar Sesión

Caso de uso	Cerrar Sesión
Código de Caso de Uso	CU-02
Código de Requerimiento	RF-02
Actor (es)	Administrador, Usuarios
Precondición	El usuario ha iniciado Sesión en el sistema.
Postcondición	
Flujo básico	<ol style="list-style-type: none"> 1. El Usuario presiona la opción del menú Cerrar Sesión. 2. El sistema cierra la sesión del usuario y vuelve a la módulo de inicio de sesión.
Flujo alternativo	<ol style="list-style-type: none"> 2. Si no se desea cerrar sesión , esta permanece abierta y en estado activo.

Fuente: Creación propia

5.1.7 Agregar usuario

Figura 27. Diagrama de caso de uso Agregar usuario



Fuente: Creación propia

Tabla 27. Caso de uso Agrega usuario

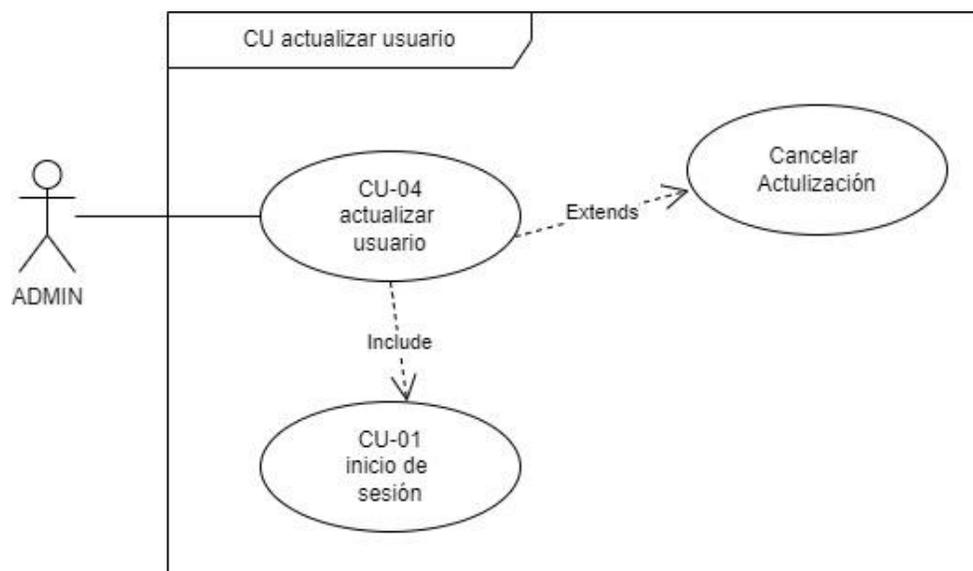
Caso de uso	Agregar usuario
Código de Caso de Uso	CU-03
Código de Requerimiento	RF-03
Actor (es)	Administrador
Precondición	El usuario registrado en la base de datos con permisos administrador.
Postcondición	Realizar el ingreso respectivo.
Flujo básico	<ol style="list-style-type: none"> 1. El usuario ingresa en el menú a la opción <i>mantenimiento de usuarios</i>. 2. Presiona el botón <i>agregar usuario</i>. 3. Se muestran en los campos para el ingreso del usuario. 4. Se ingresa la información solicitada en los campos y se presiona el botón <i>guardar</i>. 5. El sistema valida la información ingresada, si es correcta presenta un mensaje de registro de usuario correcto.
Flujo alternativo	

4. En el caso de que los datos estén en un formato distinto al necesitado se mostrara un mensaje de error y se volverá al formulario.
5. En el caso de que uno de los campos solicitados estén vacíos se mostrara un mensaje de error y se volverá al formulario.

Fuente: Creación propia

5.1.8 Actualizar usuario

Figura 28. Diagrama de caso de uso actualizar usuario



Fuente: Creación propia

Tabla 28. Caso de uso Actualizar usuario

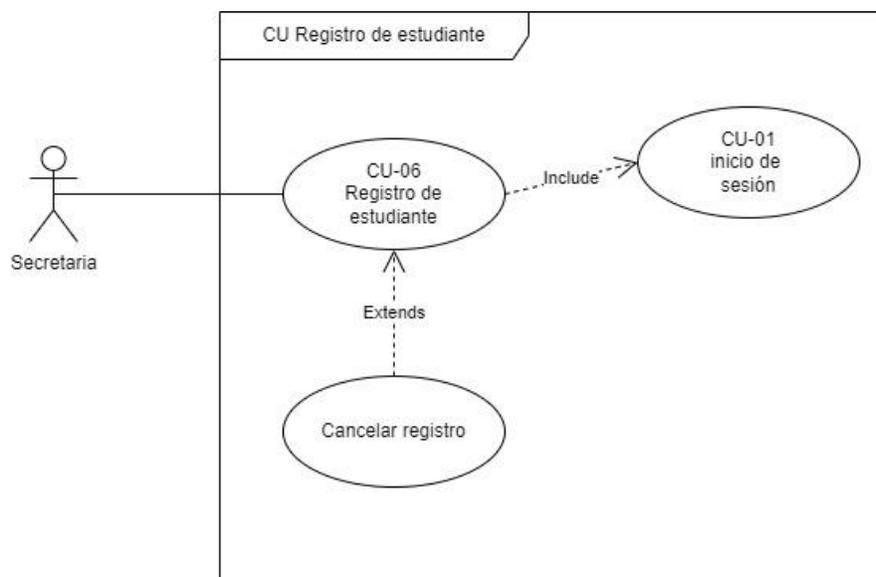
Caso de uso	Actualizar usuario
Código de Caso de Uso	CU-04
Código de Requerimiento	RF-04
Actor (es)	Administrador

Precondición	Usuarios deben de estar registrados en la base como administrador
Postcondición	Realizar la(s) modificación(es) correcta(s) en el/los campo(s) específico(s).
Flujo básico	
<ol style="list-style-type: none"> 1. El administrador ingresa a la opción del menú <i>Mantenimiento de Usuarios</i>. 2. Se muestran en los campos del usuario que se pueden actualizar. 3. Se ingresa la información solicitada en los campos y se presiona el botón <i>actualizar</i> 4. El sistema valida la información ingresada, si es correcta presenta un mensaje de actualización de usuario correcta. 	
Flujo alternativo	
<ol style="list-style-type: none"> 4. En el caso que no convaliden los datos especificados en el formulario se mostrara un mensaje de error y se volverá al formulario. 	

Fuente: Creación propia

5.1.9 Registro de estudiante

Figura 29. Diagrama de caso de uso Registro de estudiante



Fuente: Creación propia

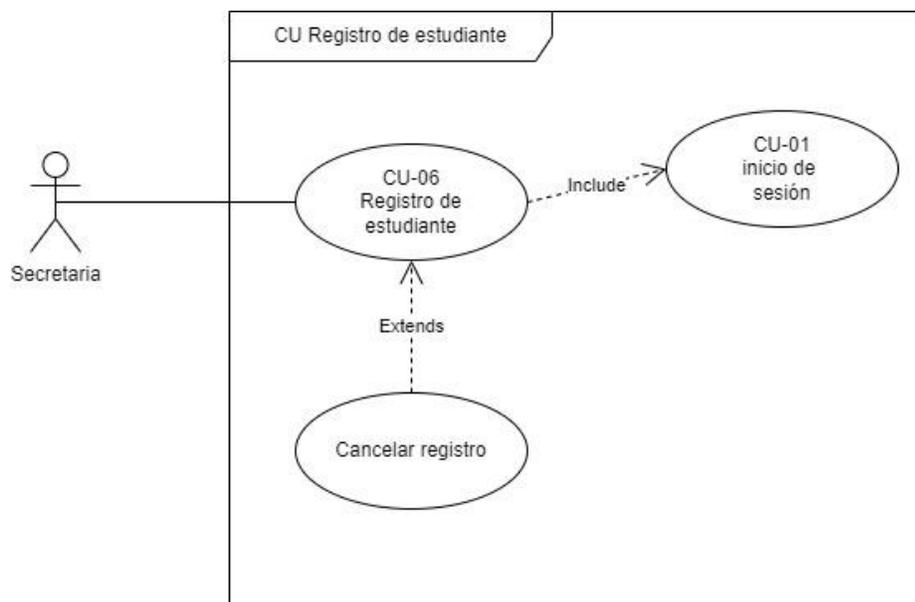
Tabla 29. Caso de uso registro de estudiante

Caso de uso	Registro de estudiante
Código de Caso de Uso	CU-06
Código de Requerimiento	RF-06
Actor (es)	Administrador
Precondición	El usuario registrado en la base de datos con permisos administrador.
Postcondición	Realizar el ingreso respectivo.
Flujo básico	<ol style="list-style-type: none"> 1. Se ingresa al formulario de mantenimiento de estudiantes. 2. Se muestran en los campos para el ingreso del registro. 3. Se ingresa la información solicitada en los campos y se presiona el botón <i>guardar</i>. 4. El sistema valida la información ingresada, si es correcta presenta un mensaje de registro correcto.
Flujo alternativo	<ol style="list-style-type: none"> 1. Si en el paso 5 el usuario ya no desea agregar un usuario puede terminar en el botón <i>cancelar</i>.

Fuente: Creación propia

5.1.10 Actualizar estudiante

Figura 30. Diagrama caso de uso actualizar estudiante



Fuente: Creación propia

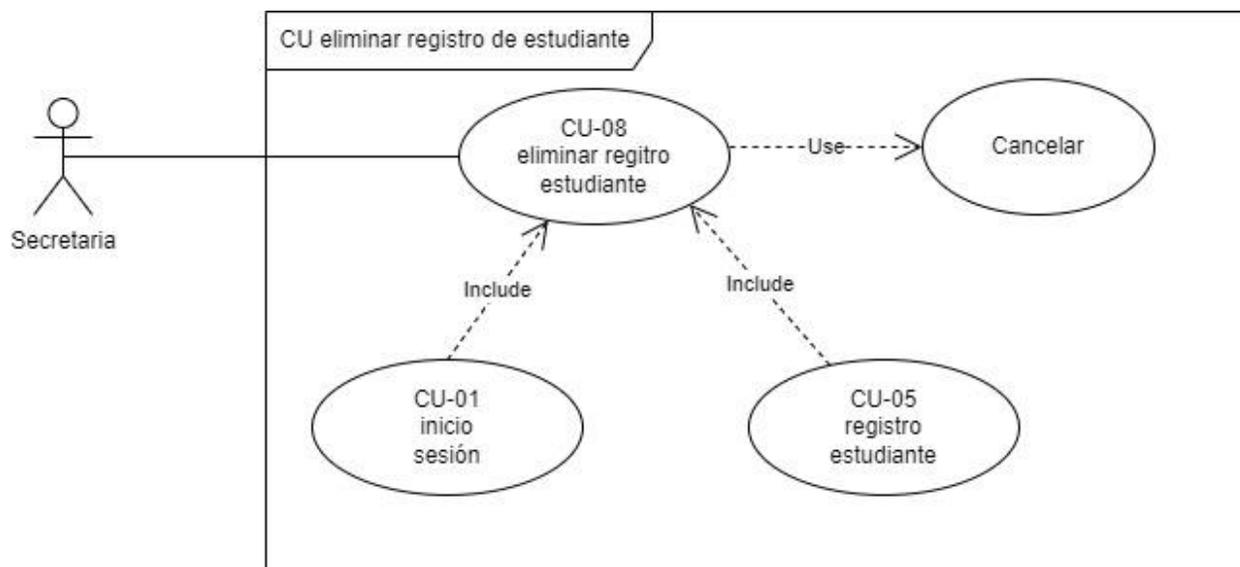
Tabla 30. Caso de uso actualizar registro estudiante

Caso de uso	Actualizar estudiante
Código de Caso de Uso	CU-07
Código de Requerimiento	RF-07
Actor (es)	Usuarios
Precondición	Usuarios deben de estar registrados en la base como administrador
Postcondición	Realizar la(s) modificación(es) correcta(s) en el/los campo(s) específico(s).
Flujo básico	
<ol style="list-style-type: none"> 1. Se ingresa en el mantenimiento de estudiantes 2. Se muestran en los campos del usuario que se pueden actualizar. 3. Se ingresa la información solicitada en los campos y se presiona el botón <i>actualizar</i> 4. El sistema valida la información ingresada, si es correcta presenta un mensaje de actualización de usuario correcta. 	
Flujo alternativo	
<ol style="list-style-type: none"> 4. En el caso que no convaliden los datos especificados en el formulario se mostrara un mensaje de error y se volverá al formulario. 	

Fuente: Creación propia

5.1.11 Eliminar, inhabilitar registro de estudiante

Figura 31. Diagrama de caso de uso eliminar registro de estudiante



Fuente: Creación propia

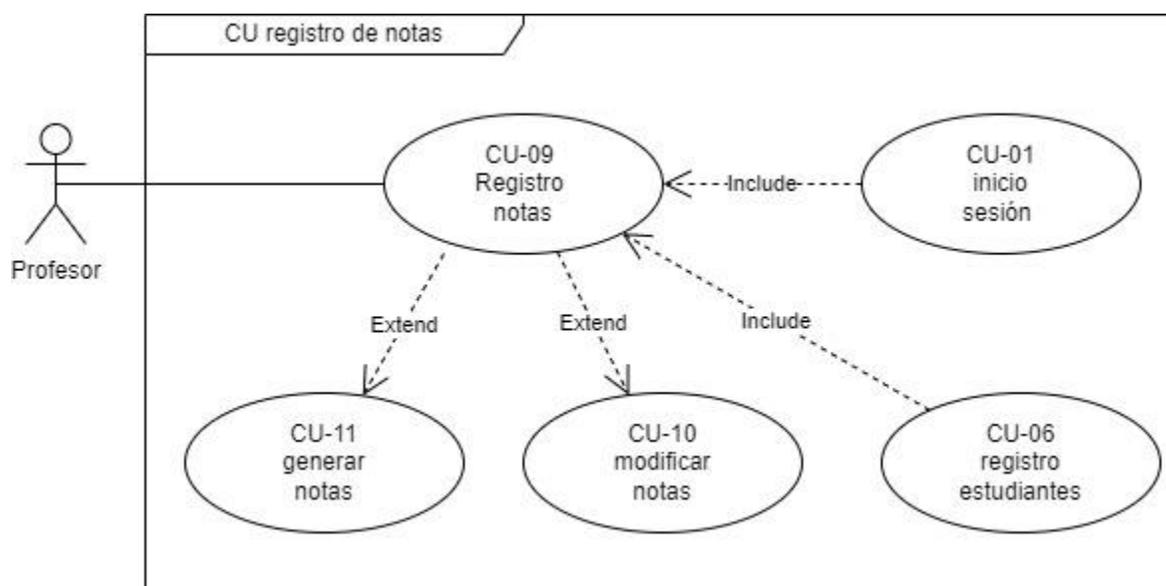
Tabla 31. Caso de uso eliminar registro de estudiante

Caso de uso	Eliminar registro de estudiante
Código de Caso de Uso	CU-08
Código de Requerimiento	RF-08
Actor (es)	Usuarios
Precondición	
Postcondición	Verificar con exactitud el registro que se deseara eliminar.
Flujo básico	<ol style="list-style-type: none"> 1. Selecciona el (los) registro (os) que desea eliminar o inhabilitar. 2. Presionar el botón <i>eliminar</i> o <i>inhabilitar</i>. 3. El sistema eliminará completamente el registro, o lo inhabilitara. 4. Se mostrará un mensaje de eliminación de registro o inhabilitación de registro
Flujo alternativo	<ol style="list-style-type: none"> 3. El usuario puede presionar el botón <i>cancelar</i> en cualquier momento y volver al inicio del módulo.

Fuente: Creación propia

5.1.12 Registrar notas

Figura 32. Diagrama caso de uso registrar notas



Fuente: Creación propia

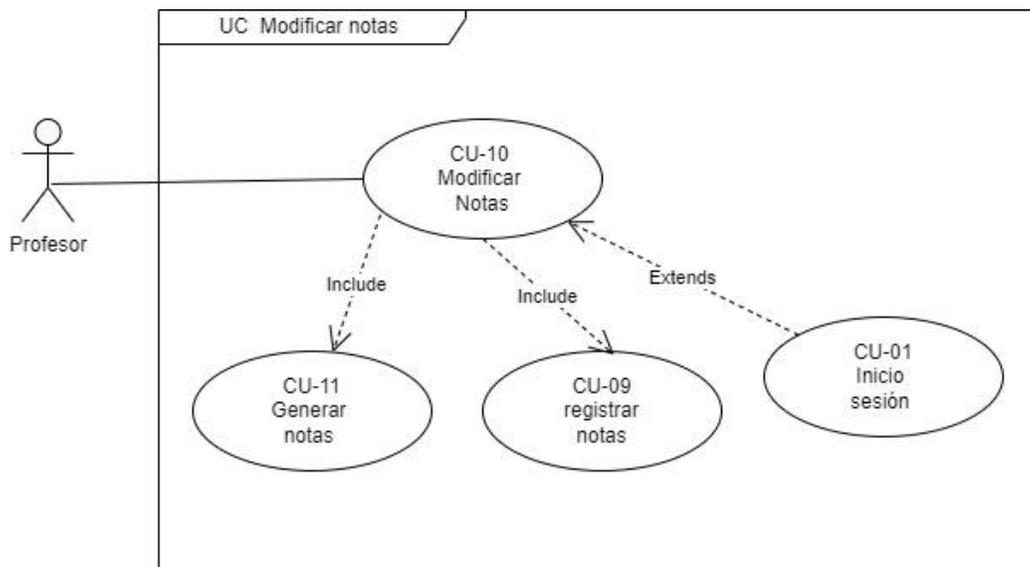
Tabla 32. Caso de uso registrar notas

Caso de uso	Registro de notas
Código de Caso de Uso	CU-09
Código de Requerimiento	RF-09
Actor (es)	Profesor
Precondición	El estudiante debe estar registrado en el sistema.
Postcondición	Verificar con exactitud el registro que se desea agregar
Flujo básico	<ol style="list-style-type: none"> 1. Llenar el formulario de la notas de los estudiantes específicos. 2. El sistema validará y almacenara las notas en la base de datos. 3. Se mostrará un mensaje de confirmación.
Flujo alternativo	<ol style="list-style-type: none"> 2. En el caso de que alguno de los campos solicitados este vacío se mostrara un mensaje de error.

Fuente: Creación propia

5.1.13 Modificar notas

Figura 33. Diagrama de caso de uso Modificar notas



Fuente: Creación propia

Tabla 33. Caso de uso Modificar notas

Caso de uso	Registro de notas
Código de Caso de Uso	CU-10
Código de Requerimiento	RF-10
Actor (es)	Profesor
Precondición	El estudiante debe estar registrado en el sistema.
Postcondición	Realizar la(s) modificación(es) correcta(s) en el/los campo(s) específico(s).
Flujo básico	<ol style="list-style-type: none"> 1. Modificar la nota errónea en el formulario de modificación. 2. El sistema validará y almacenará las notas en la base de datos. 3. Se mostrará un mensaje de confirmación de modificación de nota.
Flujo alternativo	

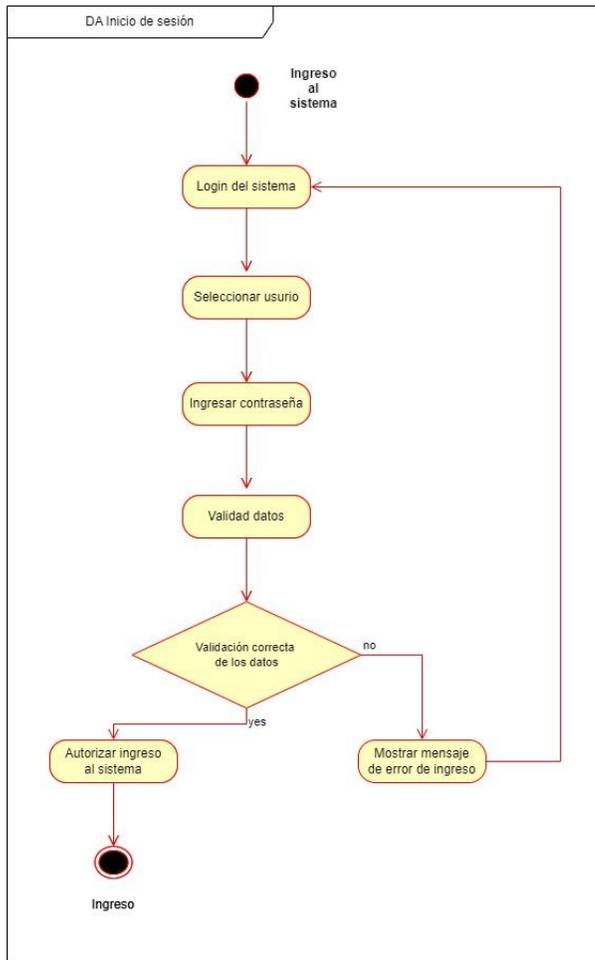
-
2. En el caso de un error en la validación de los datos el sistema mostrara un error y volverá al formulario.
-

Fuente: Creación propia

5.1.14 Diagrama de actividad

En la siguiente sección, se presentarán algunas de las actividades realizadas por el sistema mediante diagramas de actividad. Estos diagramas ilustran el flujo de trabajo entre los usuarios y el sistema, mostrando de manera visual cómo se llevan a cabo las diferentes acciones y procesos en el sistema.

Figura 34. Diagrama de actividad inicio de sesión



Fuente: Creación propia

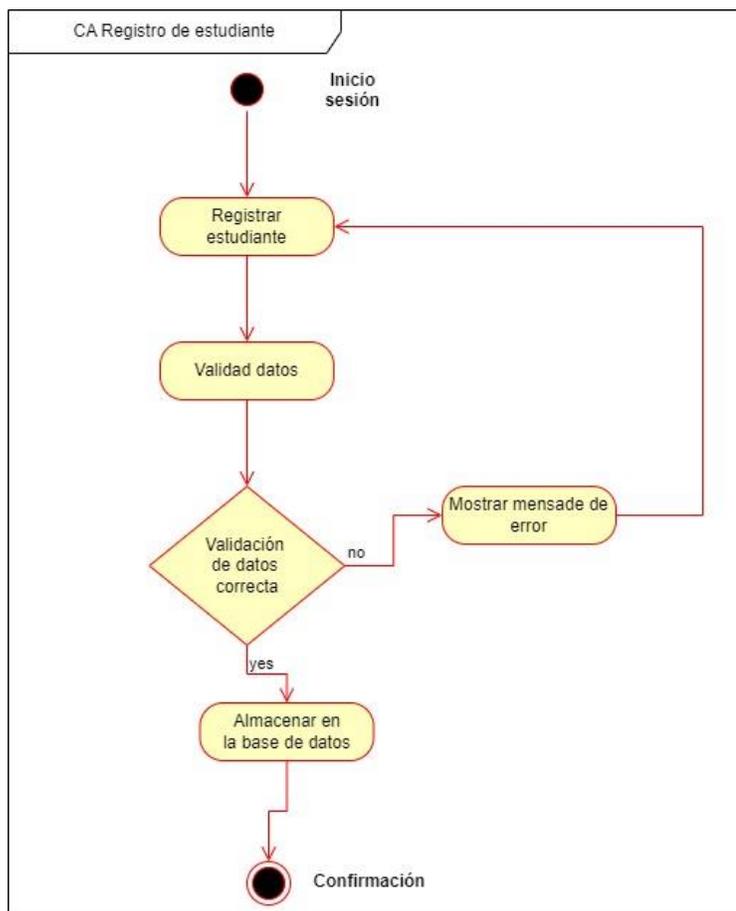
Tabla 34. Diagrama de actividad inicio de sesión

Diagrama de actividad	Inicio de sesión
Código de Caso de Uso	DA-01
Actor (es)	Usuario
Descripción	

El usuario ingresa los datos, el sistema valida que el usuario este registrado y que los espacios se lleven de forma completa y correcta, si la validación es correcta el usuario tiene acceso al sistema, de lo contrario el sistema envía un mensaje de error.

Fuente: Creación propia.

Figura 35. Diagrama de actividad Registro de estudiantes



Fuente: Creación propia

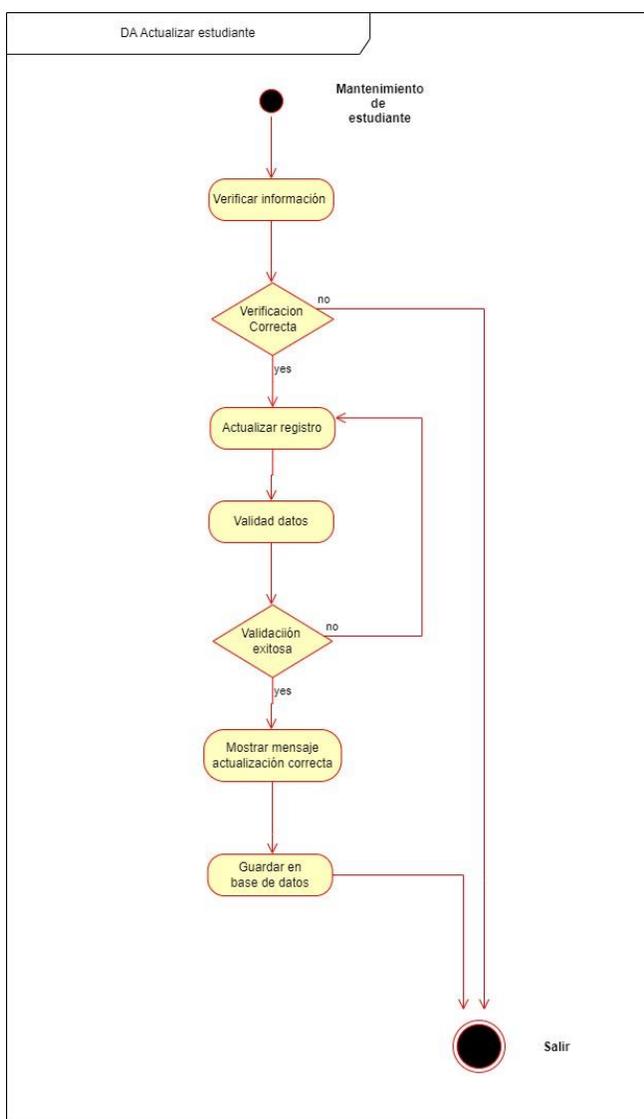
Tabla 35. Diagrama de actividad registro de estudiantes

Diagrama de actividad	Registro de estudiantes
Código de Caso de Uso	DA-02
Actor (es)	Usuario
Descripción	

El usuario autorizada inicia el registro del estudiante, el sistema valida que la información ingresada este completa, y el formato correcto , si las validaciones son correcta el sistema almacena el nuevo registro, de los contrario envía un mensaje al usuario.

Fuente: Creación Propia

Figura 36. Diagrama de actividad Actualizar estudiante

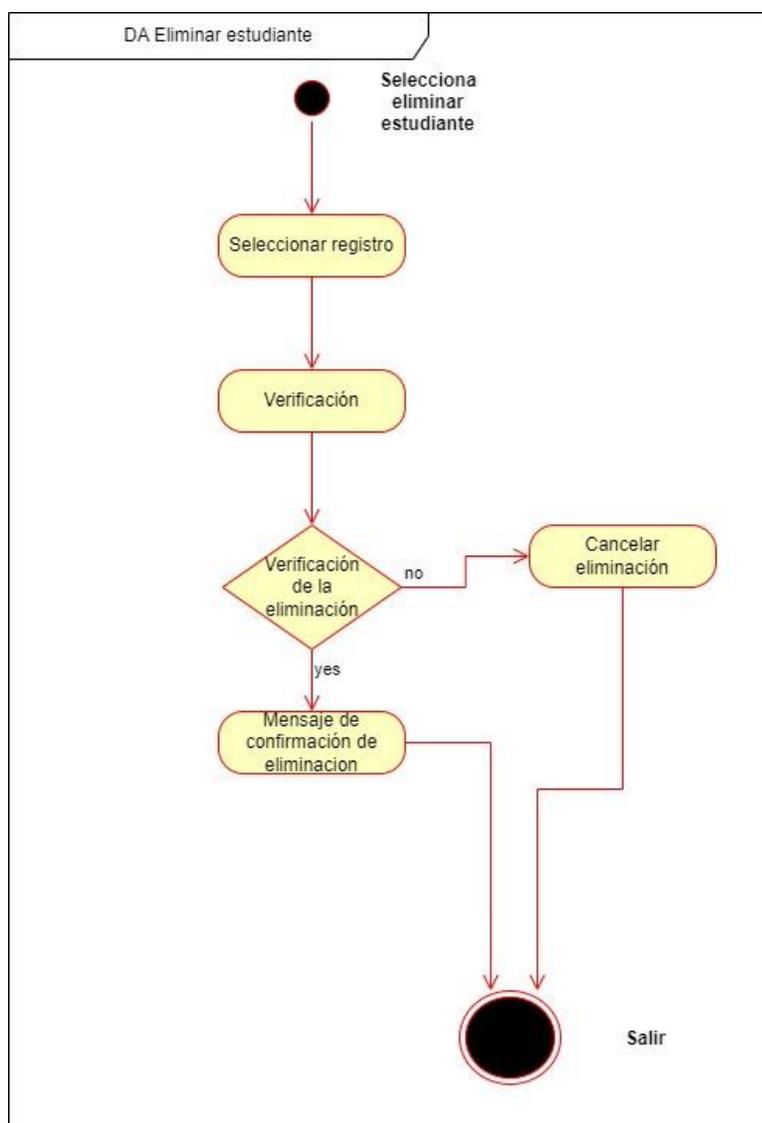


Fuente: Creación propia

Tabla 36. Diagrama de actividad Actualizar estudiante

Diagrama de actividad	Actualizar estudiante
Código de Caso de Uso	DA-03
Actor (es)	Usuario
Descripción	El usuario autorizada selecciona el registro del estudiante desea actualizar, se modifican los espacios que se desean actualizar, el sistema valida que la información este completa y sea correcta, si la validación es exitosa, se guarda el registro en el sistema de lo contrario se mostrara un mensaje de error.

Fuente: Creación propia.

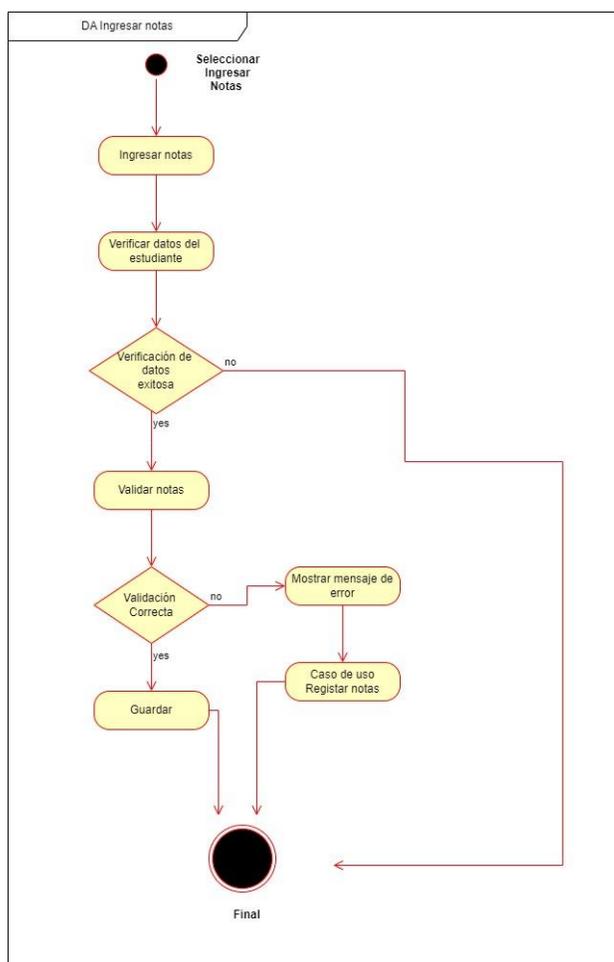
Figura 37. Diagrama de actividad Eliminar registro de estudiante

Fuente: Creación propia

Tabla 37. Diagrama de actividad Eliminar estudiante

Diagrama de actividad	Eliminar estudiante
Código de Caso de Uso	DA-04
Actor (es)	Usuario
Descripción	El usuario autorizada selecciona el registro del estudiante desea eliminar, se debe de verificar que reamente se desea eliminar el registro del estudiante si o puede cancelar dicha acción. Fuente: Creación propia.

Figura 38. Diagrama de actividad Registrar notas

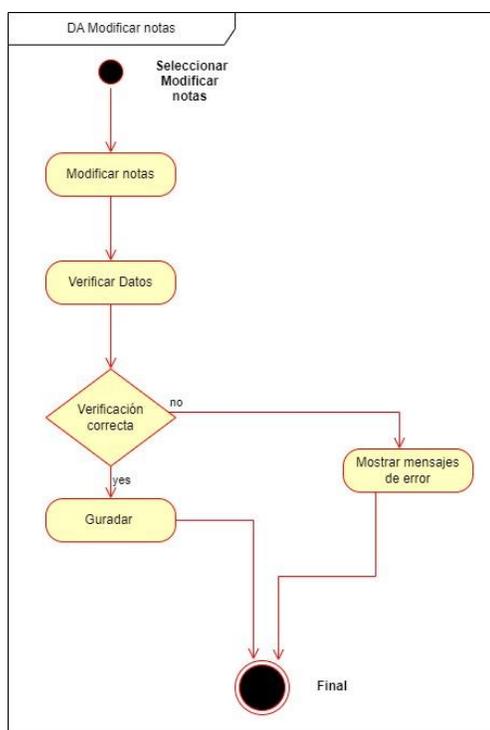


Fuente: Creación propia

Tabla 38. Diagrama de actividad Registrar notas

Diagrama de actividad	Registrar notas
Código de Caso de Uso	DA-05
Actor (es)	Profesor
Descripción	
El profesor encargado ingresa la calificación del estudiante, el sistema verifica que la información correspondiente al estudiante sea la correcta, si la verificación es correcta, se valida que la notas estén en el formato correspondiente si la validación es correcta se guardan las notas, de lo contrario se muestra un mensaje de error.	

Fuente: Creación propia.

Figura 39. Diagrama de actividad modificar nota

Fuente: Creación propia

Diagrama de actividad	Modificar notas
Código de Caso de Uso	DA-05
Actor (es)	Profesor
Descripción	
El profesor autorizada selecciona el registro de la nota que desea actualizar, se modifican los espacios que se desean actualizar, el sistema valida que la información este completa y sea	

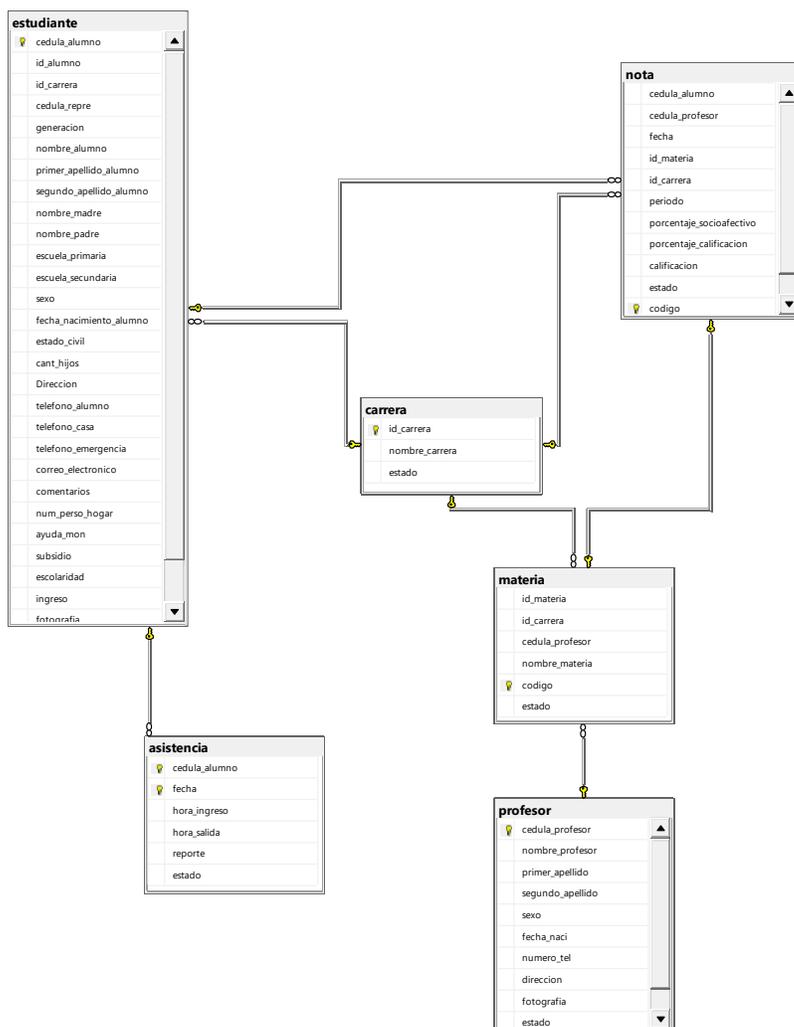
correcta, si la validación es exitosa, se guarda el registro en el sistema de lo contrario se mostrara un mensaje de error.

Fuente: Creación propia

5.2 Diagrama relacional

Este diagrama relacional representa una estructura de base de datos que permite almacenar información detallada sobre estudiantes, notas, carreras, materias, profesores y asistencia, y cómo estas entidades están relacionadas entre sí.

Figura 40. Diagrama relacional



Fuente: Creación propia.

5.3 Desarrollo

En esta sección se describirá el desarrollo del sistema de gestión de información, elaborado según los requisitos recolectados en la etapa de análisis y que se encuentran detallados en este documento en la etapa de requerimientos funcionales y no funcionales.

Durante la etapa del análisis demuestra que es necesario manipular la información de los estudiantes de forma centralizada en una sola herramienta, que permita registrar muchos más registros en una base de datos que permita a su vez mantener estos datos con integridad.

El desarrollar el sistema que permita manejar la información en un solo lugar evitará el tener que obtener datos de diferentes hojas de cálculo, generando en muchos casos duplicidad de datos, además obligará al usuario ingresar los datos en el formato que se necesita y de esta forma generar los reportes previstos en el sistema o por medio de consultas a la base de datos.

El sistema se desarrolló en C# utilizando como IDE el visual studio 2019, usando una arquitectura de diseño en capas.

5.3.1 *Arquitectura de la solución*

La arquitectura de diseño en capas permite separar el trabajo a realizar en cada capa, de modo que se pueda modificar una capa sin afectar al resto de las capas. La programación en capas también permite migrar el motor de base de datos modificando una sola capa. La figura 37 muestra una arquitectura en capas.

Figura 41. *Arquitectura en capas*



Fuente: Creación propia

A continuación, se describen las capas de este modelo:

- **Capa de presentación:** Es la capa encargada de interactuar con el usuario, compuesta por aquellos elementos encargados de comunicar al usuario final con la aplicación (ventanas, cuadros de diálogo, mensajes, entre otros). Por medio de esta capa el usuario solicita que se ejecuten las tareas por medio de parámetros de entrada y recibiendo datos. Esta capa se comunica con la capa de negocio enviando y recibiendo información y con la capa entidades interactuando con sus objetos.
- **Capa de negocio:** Se encarga de implementar todo lo que el *software* debe considerar antes o después de realizar una acción. La capa lógica de negocio recibe de la capa presentación las solicitudes, valida que las condiciones que establece la capa negocio se cumplan antes de realizar la acción o hacer una solicitud a la capa de acceso a datos.
- **Capa de acceso a datos:** Capa encargada de la comunicación con la capa de datos, en esta capa se alojan todas las acciones CRUD. Esta capa conoce el motor de base de datos que se está utilizando, pero desconoce el Front. La capa de datos

recibe las peticiones de la capa lógica de negocio ejecuta dichas acciones y devuelve el resultado.

- **Entidades:** No es una capa como tal, aquí se encuentran todos aquellos objetos o clases que representan al Negocio y es la única que puede comunicarse con el resto de las capas, es un puente de transporte de datos. Esta capa puede ser complemento de la Capa Negocio.

5.3.1.1 Capa Datos

En la capa datos se crea la clase “Conexión”, en esta clase se encuentra el nombre y la dirección del servidor a la que el programa se va a conectar. Es clase utiliza el patrón de diseño Singleton para asegurar que solo haya una instancia de la clase “Conexión”, esto mediante el mediante un campo estático llamado “Con” que almacena la instancia existente de la clase o crea una nueva si aún no existe, y un método estático llamado “getInstancia” encargado de devolver la instancia única de la clase “Conexión”.

Para establecer la conexión a la base de datos se utiliza el método “Crear Conexión” que construye la cadena de conexión con la información proporcionada por los campos de la clase. Para acceder a la base de datos puede utilizar la autenticación de Windows o la autenticación de SQL.

También se definen las otras clases cada una con una serie de métodos para interactuar con la base de datos. Cada método se encarga de establecer una conexión con la base de datos, ejecutar una consulta o procedimiento almacenado y recuperar los resultados en un objeto DataTable. Los métodos también manejan la apertura y cierre de la conexión a la base de datos y capturan posibles excepciones que puedan ocurrir durante la ejecución. (Anexo 5.)

5.3.1.2 Capa Entidades

En esta capa se definen las clases y sus propiedades. Estas clases representan un objeto y sus propiedades. Estas clases permiten almacenar información de los objetos. Las propiedades son públicas y tienen los modificadores "get" y "set", lo que significa que se puede acceder y modificar el valor de cada propiedad desde otras partes del código. (Anexo 6.)

5.3.1.3 Capa Negocio

Las clases que están dentro de esta capa contienen métodos estáticos que interactúan con la capa y realizan operaciones específicas relacionadas con las entidades.

La capa negocio proporciona métodos que permiten realizar operaciones y manipular los datos de las clases de la capa Entidades. Algunas de las funciones específicas que realiza la capa Negocio son: Listar, Buscar, Insertar, Actualizar, entre otras. (Anexo 7.)

5.3.1.4 Capa presentación

Esta capa es la encargada de la interfaz de usuario y su interacción del usuario con el sistema. Aquí se muestran elementos gráficos para el usuario interactúe. Algunos ejemplos de estas herramientas graficas son los siguientes:

- **Botones:** Permiten al usuario interactuar el programa.
- **Campos de entrada:** Estos elementos permiten al usuario ingresar texto o datos. Por ejemplo, un campo de texto para ingresar un nombre o un campo numérico para ingresar una cantidad.
- **Menús desplegables:** Estos elementos proporcionan una lista de opciones para que el usuario seleccione una. Por ejemplo, un menú desplegable para seleccionar el país de residencia.

- **Casillas de verificación:** Permiten al usuario seleccionar opciones múltiples. Por ejemplo, una lista de casillas de verificación para seleccionar los elementos de una lista.
- **Barras de desplazamiento:** Estos elementos permiten al usuario desplazarse por contenido adicional que no cabe en la pantalla. Por ejemplo, una barra de desplazamiento vertical en una ventana de chat.
- **Ventanas emergentes:** Puedes utilizar ventanas emergentes para mostrar información adicional o solicitar confirmación al usuario. Por ejemplo, una ventana emergente que muestra un mensaje de confirmación antes de eliminar un archivo.

A continuación, se presentan las pantallas del sistema:

5.3.2 *Formulario de ingreso al sistema*

Este formulario consta de dos campos que debe de llenar el usuario para poder ingresar al sistema. El campo “usuario” corresponde al correo institucional y el campo “clave” a la contraseña proporcionada por el administrador a cada uno de los usuarios.

Figura 42. *Formulario Ingreso de usuario*

The image shows a login window titled "Login" with a close button in the top right corner. On the left side, there is a logo for "SAMUEL" (a globe icon) and the text "FUNDACIÓN HEDWIG Y ROBERT SAMUEL". The main area contains two input fields: the first is empty, and the second is labeled "CLAVE" and has a small eye icon to its right. Below the fields is a button labeled "ACCEDER".

Fuente: Creación propia.

5.3.3 *Formulario Principal*

Desde este formulario el usuario puede acceder al resto de los formularios del sistema dependiendo de los privilegios asignados.

Figura 43. *Formulario Principal*



Fuente: Creación propia.

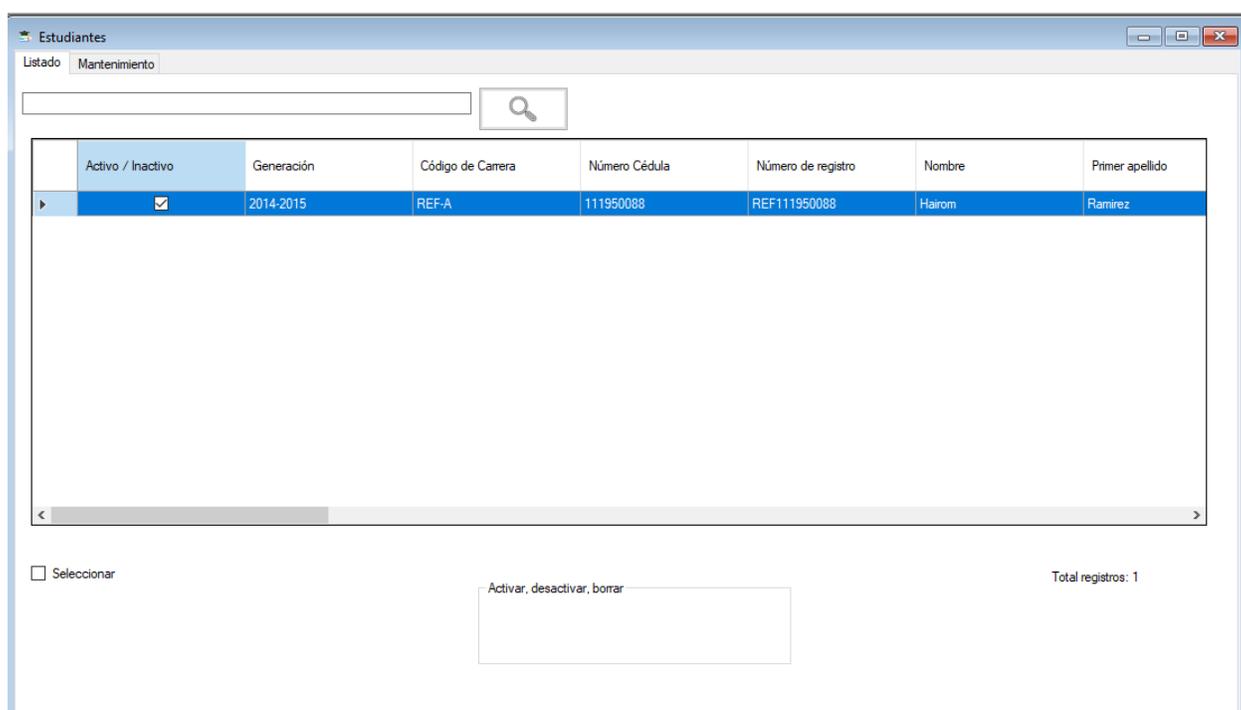
5.3.4 *Formulario Informe detallado*

El formulario de informe detallado es una herramienta importante para recopilar y visualizar gran cantidad de información del sistema. Este formulario consta de dos pestañas: Listado y Mantenimiento.

En la pestaña Listado, se muestra la información detallada de cada uno de los estudiantes que ingresan a la Fundación Samuel. Esta información se registra en el sistema y se presenta en el formulario de forma clara y organizada.

La pestaña Mantenimiento, por otro lado, permite realizar ajustes y actualizaciones en la información registrada en el sistema. Desde allí, se pueden realizar cambios y modificar la información de los estudiantes según sea necesario.

Figura 44. Pestaña Listado del formulario Informe detallado



Activo / Inactivo	Generación	Código de Carrera	Número Cédula	Número de registro	Nombre	Primer apellido
<input checked="" type="checkbox"/>	2014-2015	REF-A	111950088	REF111950088	Hairom	Ramirez

Seleccionar

Activar, desactivar, borrar

Total registros: 1

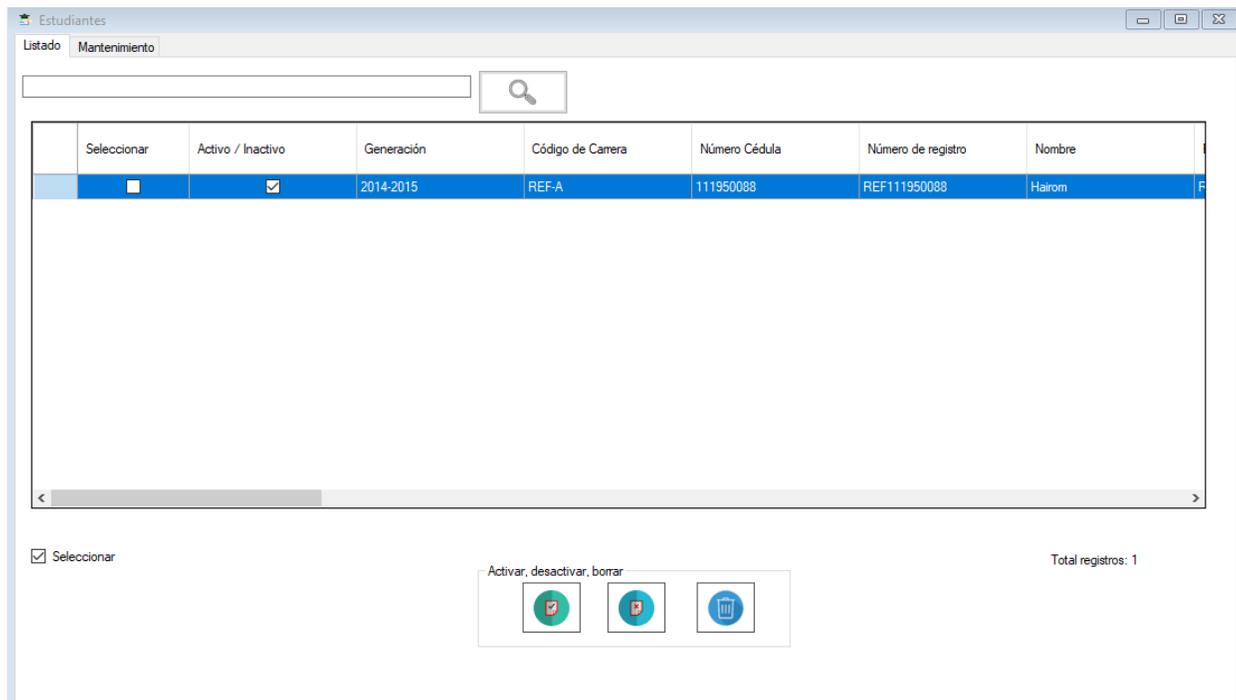
Fuente: Creación propia.

La pestaña Mantenimiento del formulario de informe detallado es una herramienta de gestión de datos que permite a los usuarios autorizados realizar cambios y actualizaciones en los registros de estudiantes.

En esta pestaña, los usuarios con permisos específicos tendrán la capacidad de activar, desactivar o borrar registros seleccionados de manera segura y controlada. Es importante destacar que los botones de "Activar", "Desactivar" y "Borrar" sólo serán visibles para los

usuarios con estos privilegios, asegurando así la integridad de los datos y la seguridad del sistema.

Figura 45. Pestaña Listado Activar, desactivar y borrar registro



Fuente: Creación propia.

El formulario de informe detallado cuenta con una funcionalidad de búsqueda que permite a los usuarios obtener información específica de un estudiante o grupo de estudiantes utilizando filtros. Esta herramienta es útil para encontrar información precisa de manera rápida y eficiente.

Además, si se desea acceder a la información detallada de un estudiante, se puede hacer doble clic en el registro correspondiente. Esto llevará al usuario al formulario de mantenimiento, que se describe con más detalle en la pestaña de Mantenimiento del formulario de informe detallado.

Figura 46. Pestaña Mantenimiento del formulario Informe detallado

Listado | **Mantenimiento**

5

Nombre(*)
 Primer Apellido(*)
 Segundo Apellido(*)
 Número de cédula
 Edad(*)
 Estado Civil(*) Sexo(*)
 Correo electrónico
 Teléfono
 Teléfono de la casa
 Teléfono de emergencia
 Fecha de nacimiento(*)

Número de registro
 Carrera
 Generación(*)
 Colegio(*)
 Escuela(*)

Nombre de la madre(*)
 Nombre del padre(*)
 Cédula del padre(*)
 Comentarios
 Dirección(*)

Persona en el hogar(*) Ingreso(*)
 Subsidio(*)
 Cantidad de Hijos(*) Escolaridad(*)

(*) Indica que el dato es obligatorio

Fuente: Creación propia.

Al hacer doble clic en un registro específico en la pestaña "Listado" del formulario de informe detallado, se habilitan los botones de actualización para modificar un registro existente en el sistema. Si se desea realizar un nuevo registro, el usuario deberá volver manualmente a la pestaña de "Mantenimiento".

Es importante tener en cuenta que los botones de actualización sólo se activan después de que se ha seleccionado un registro específico en la pestaña "Listado". Si se desea realizar un nuevo registro, el usuario deberá cambiar a la pestaña "Mantenimiento" y llenar los campos requeridos para ingresar la información.

Figura 47. Pestaña Mantenimiento del Formulario Informe detallado.

The form contains the following fields and controls:

- Profile:** Placeholder for 'Fotografía'.
- Personal Information:**
 - Nombre(*)
 - Primer Apellido(*)
 - Segundo Apellido(*)
 - Número de cédula
 - Edad(*)
 - Estado Civil(*)
 - Sexo(*)
 - Comeo electrónico
 - Teléfono
 - Teléfono de la casa
 - Teléfono de emergencia
 - Fecha de nacimiento(*)
- Family and Education:**
 - Número de registro
 - Carrera (Asis. Administracion)
 - Generación(*)
 - Colegio(*)
 - Escuela(*)
 - Persona en el hogar(*)
 - Ingreso(*)
 - Subsidio(*)
 - Monto Subsidio
 - Cantidad de Hijos(*)
 - Escolaridad(*)
- Parental Information:**
 - Nombre de la madre(*)
 - Nombre del padre(*)
 - Cédula del padre(*)
 - Comentarios
 - Dirección(*)
- Legend:** (*) Indica que el dato es obligatorio.

Fuente: Creación propia.

El formulario de informe detallado es un medio directo para que el usuario responsable ingrese y seleccione información de manera eficiente. Dependiendo del caso, el usuario debe digitar o seleccionar los datos necesarios para completar los campos requeridos.

Es importante destacar que la precisión y la integridad de los datos ingresados es fundamental para garantizar la confiabilidad del sistema. Por lo tanto, es esencial que el usuario responsable verifique cuidadosamente la información ingresada antes de guardarla en el sistema.

5.3.5 Formulario Carreras

El formulario de Carreras es una herramienta esencial en el sistema de la Fundación Samuel, ya que es aquí donde se almacena, actualiza y muestra información sobre las diferentes especializaciones que se imparten en la institución.

Además, este formulario permite a los usuarios con los permisos correspondientes activar, desactivar o eliminar un registro según sea necesario. Es importante destacar que la información ingresada en este formulario será utilizada en otros formularios del sistema, por lo que su exactitud y actualización son vitales para el correcto funcionamiento del mismo.

Figura 48. Formulario Carreras

	Activo/Inactivo	Codigo de la carrera	Nombre de la carrera
▶	<input checked="" type="checkbox"/>	ADM	Asis, Administraci...
	<input checked="" type="checkbox"/>	CON	Contabilidad
	<input checked="" type="checkbox"/>	MEC	Mecánica Autom...
	<input checked="" type="checkbox"/>	REF-A	Refrigeración A

Carreras

Código de Carrera

Carrera

Guardar, Actualizar, Cancelar

Activar, desactivar, borrar

Seleccionar

Fuente: Creación propia.

5.3.6 *Formulario Materias*

El formulario de Materias tiene como función principal la gestión de información de las asignaturas que se imparten en la Fundación Samuel, organizadas por especialidad. A través de este formulario, se pueden registrar, actualizar y visualizar los datos correspondientes a cada materia.

Asimismo, el formulario de Materias cuenta con la opción de activar, desactivar o borrar registros, siempre y cuando el usuario cuente con los permisos necesarios para llevar a cabo estas acciones. Es importante tener en cuenta que la información registrada en este formulario será utilizada en otros formularios del sistema.

Figura 49. *Formulario Materias*

Activo / Inactivo	Código de Asignatura	Código de Carrera	Nombre de Asignatura
<input checked="" type="checkbox"/>	ELEC	REF-A	electricidad

Seleccione

Seleccionar

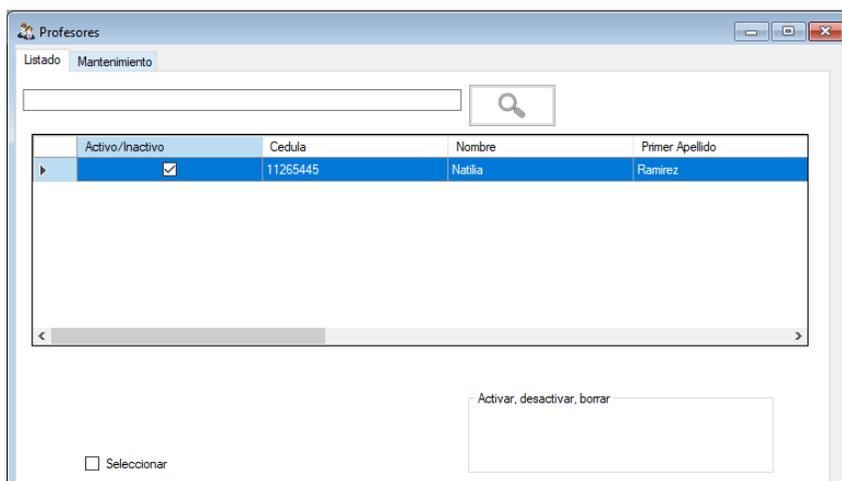
Fuente: Creación propia.

5.3.7 *Formulario profesores*

Este formulario está compuesto por dos pestañas: Listado y Mantenimiento. En la pestaña listado se muestran los datos de los profesores. Desde aquí se puede activar, desactivar o borrar

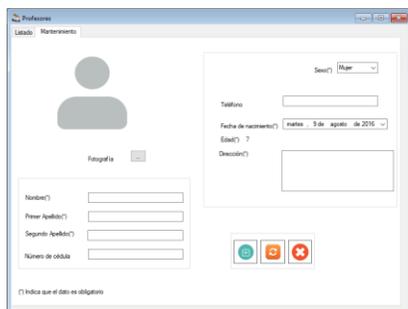
un registro si se tiene los permisos necesarios. En la pestaña Mantenimiento se puede actualizar un registro, a esta pestaña se accede haciendo doble clic sobre el registro que se quiere actualizar desde la pestaña “Listado”. Si se accede a la pestaña “Mantenimiento” de forma manual, se podrá agregar un nuevo registro.

Figura 50. Pestaña Listado del formulario Profesores



Fuente: Creación propia

Figura 51. Pestaña Mantenimiento del formulario Profesores



Fuente: Creación propia.

5.3.8 Formulario Control de Asistencia

Este formulario de control de asistencias diarias permite seleccionar a los estudiantes, registrar la fecha y el estado de su asistencia, y opcionalmente, la hora de llegada. Este formulario será útil para mantener un registro preciso de la asistencia de los estudiantes y para tomar medidas cuando los estudiantes falten a clases.

Figura 52. Formulario Control de asistencias

The screenshot shows a web interface titled "Control de Asistencia" for "FUNDACIÓN SAMUEL HEDWIG Y ROBERT SAMUEL". It features a search bar with a magnifying glass icon and a "Buscar" button. Below the search bar are two dropdown menus for "Generación" and "Carrera". A table with the following columns is visible: "Estudiante", "Fecha", "Entrada", "Salida", "Estado", and "generacion". To the right of the table is a dropdown menu for selecting a student, with "Merilyn Dayana Campos Gonzalez" selected. Below the table are input fields for "Estudiante" (with the selected name), "Numero de indentificación" (with the value "118580201"), "Fecha" (with the value "09/03/2023"), "Hora Salida" (with the value "0 : 00"), "Reporte" (with a dropdown menu), and "Hora de Ingreso" (with the value "0 : 00"). There is also a "Seleccionar" checkbox. On the right side, there are three buttons: "Actualizar", "Ocultar lista de estudiantes" (with an eye icon), "Guardar" (with a save icon), and "Cancelar" (with a close icon).

Fuente: Creación propia.

5.3.9 Formulario Registro de notas

Este formulario tiene como función principal el registro de las notas de los estudiantes en la Fundación Samuel. Se despliega una lista con los nombres de los estudiantes y se puede seleccionar un grupo de estudiantes por carrera y/o asignatura. Al seleccionar un estudiante, la

información del profesor y la asignatura que imparte se completan de forma automática. Se debe seleccionar la fecha e ingresar la calificación para posteriormente guardar el registro. El usuario tiene la posibilidad de ver los registros del sistema y aplicar filtros si desea obtener información específica.

Figura 53. *Formulario Registro de notas.*

FUNDACIÓN HEDWIG Y ROBERT SAMUEL Fecha 23/03/2023

Profesor: ...
Estudiante: ...
Materia: ...

Generación Carrera Materia Mes
Seleccione Seleccione Seleccione Seleccione

ESTUDIANTE	MES	ASIGNATURA	PORCENTAJE SOCIOAFECTIVO	PORCENTAJE PRUEBA	NOTA

Generación Carrera Materia
Seleccione Seleccione Seleccione

ESTUDIANTE

Porcentaje prueba Teórica
Detalle 1 Detalle 2 Detalle 3
0.00 0.00 0.00

Calificación Socioafectiva
0.00

Porcentaje Práctica
0.00

Nota obtenida

Guardar Cancelar

Generar Nota
 Seleccionar

Fuente: Creación propia.

5.4 Plan Piloto y Pruebas

Durante el desarrollo del programa de gestión académica en la Fundación Samuel, se realizaron pruebas en algunos de los puestos de trabajo para evaluar su funcionamiento y detectar

errores o problemas. En esta sección, se describirán estas pruebas y las mejoras realizadas al sistema.

Las pruebas fueron realizadas en las computadoras del asistente de la dirección académica y la secretaria de la dirección académica usando una instancia local de SQL Server.

5.4.1 Usuario

A continuación, se describen las pruebas realizadas:

- **Prueba de registro de usuario:** Se realizaron pruebas agregando usuarios con todos los campos requeridos por el sistema. Se verificó que los datos fueran registrados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Prueba de registro de usuarios con datos incompletos o erróneos:** Esta prueba consistió en tratar de agregar a registros sin completar datos o completar la información con datos en formatos no admitidos por el sistema. Se verificó que el sistema solicitara completar los campos por medio de un mensaje y que no permitiera guardar los registros sin completar los campos o si los datos no tuvieran el formato correcto. Esta prueba fue exitosa.
- **Actualización de los datos de un usuario:** Se realizaron pruebas para la actualización de usuarios existentes en el sistema. Se verificó que los datos fueran actualizados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Prueba de actualización de usuarios con datos incompletos o erróneos:** Se llevaron a cabo pruebas para actualizar usuarios sin completar datos o completar la información con datos en formatos no admitidos por el sistema. Se verificó que el sistema solicitara completar los campos por medio de un mensaje y que no permitiera guardar los registros sin completar los campos o si los datos no tuvieran el formato correcto. Esta prueba fue exitosa. (Anexo 8.)

Figura 54. Prueba de registro de usuario exitoso

The screenshot displays a user management interface. At the top, a table lists existing users with columns for Estado, ID Usuario, ID Rol, Usuario, Email, Clave, and Rol. Below the table, a form allows for adding or updating a user. The form includes a dropdown for 'Rol de usuario' (set to 'Profesor'), a text input for 'Usuario' (set to 'PEDRO VARGAS'), and a text input for 'Email' (set to 'pedrov@gmail.com'). A confirmation dialog box titled 'Sistema Academico' is overlaid on the form, displaying an information icon and the message 'Se insertó de forma correcta el registro', with an 'Aceptar' button.

Estado	ID Usuario	ID Rol	Usuario	Email	Clave	Rol
<input checked="" type="checkbox"/>	2	2	HAIROM RAMI...	hyron@gmail.c...	-zínÉÓÁs'b'É...	Profesor
<input checked="" type="checkbox"/>	3	3	PABLO ESCO...	pablo@gmail.c...	-zínÉÓÁs'b'É...	Asistente
<input checked="" type="checkbox"/>	4	4	POPO	popo@gmail.com	-zínÉÓÁs'b'É...	Director
<input checked="" type="checkbox"/>	5	1	HAIROM	hyron@gmail.c...	123456	Administrador

Fuente: Creación propia.

Figura 55. Prueba de registro de usuarios con datos incompletos o erróneos

The screenshot shows the same user management interface as Figure 54, but with validation errors. The 'Rol de usuario' dropdown is set to 'Seleccione' and the 'Email' field is empty. Both fields have red error icons. A tooltip over the 'Email' field says 'Seleccione una opción.'. Below the form, a message box provides instructions: '* Para insertar un usuario, la clave es obligatoria.' and '* Para actualizar deje en blanco el campo Clave y la clave no se actualizará.'.

Estado	ID Usuario	ID Rol	Usuario	Email	Clave	Rol
<input checked="" type="checkbox"/>	2	2	HAIROM RAM...	hyron@gmail.c...	-zínÉÓÁs'b'É...	Profesor
<input checked="" type="checkbox"/>	3	3	PABLO ESCO...	pablo@gmail.c...	-zínÉÓÁs'b'É...	Asistente
<input checked="" type="checkbox"/>	4	4	POPO	popo@gmail.c...	-zínÉÓÁs'b'É...	Director
<input checked="" type="checkbox"/>	5	1	HAIROM	hyron@gmail.c...	123456	Administrador
<input checked="" type="checkbox"/>	6	2	PEDRO VARG...	pedrov@gmail...	-zínÉÓÁs'b'É...	Profesor

Fuente: Creación propia.

Figura 56. Actualización de los datos de un usuario

The screenshot shows a user management interface. At the top, there is a table with columns: Estado, ID Usuario, ID Rol, Usuario, Email, Clave, and Rol. The table contains six rows of user data. Below the table, there is a form for updating a user. The 'Rol de usuario' field is set to 'Director' and the 'Usuario' field is set to 'JOSE VALVERDE A'. A modal dialog titled 'Sistema Academico' is open, displaying a success message: 'Se actualizo de forma correcta el registro'. Below the message is an 'Aceptar' button. In the background, the form has a note: 'la clave no se actualizara.' and a 'Seleccionar' checkbox.

Estado	ID Usuario	ID Rol	Usuario	Email	Clave	Rol
<input checked="" type="checkbox"/>	2	2	HAIROM RAM...	hyron@gmail.c...	-ñinÉÓÀš.b'€...	Profesor
<input checked="" type="checkbox"/>	3	3	PABLO VARG...	pablo@gmail.c...	-ñinÉÓÀš.b'€...	Asistente
<input checked="" type="checkbox"/>	4	4	JOSE VALVE...	popo@gmail.c...	-ñinÉÓÀš.b'€...	Director
<input checked="" type="checkbox"/>	5	1	HAIROM	hyron@gmail.c...	123456	Administrador
<input checked="" type="checkbox"/>	6	2	PEDRO VARG...	pedrov@gmail...	-ñinÉÓÀš.b'€...	Profesor

Fuente: Creación propia

Figura 57. Prueba de actualización de usuarios con datos incompletos o erróneos

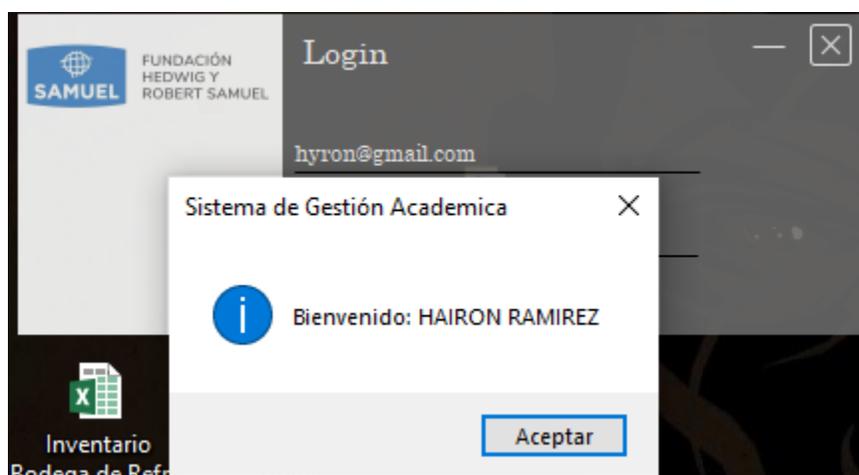
The screenshot shows the same user management interface as in Figure 56. The 'Rol de usuario' field is set to 'Director' and the 'Usuario' field is set to 'JOSE VALVERDE A'. A modal dialog titled 'Sistema Academico' is open, displaying an error message: 'Dirección de correo no válida'. Below the message is an 'Aceptar' button. In the background, the form has a note: 'es obligatoria. Para actualizar deje en blanco el campo Clave y la clave no se actualizara.' and a 'Seleccionar' checkbox.

Fuente: Creación propia.

5.4.2 Inicio de sesión

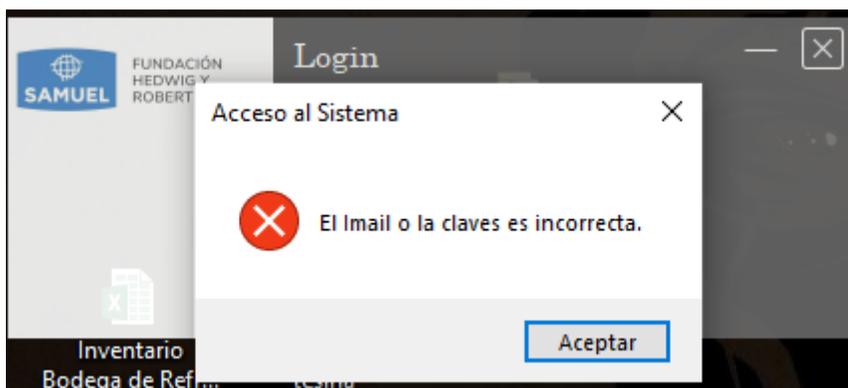
- **Inicio de sesión con éxito:** Se inicia sesión con los datos de usuario ya registrados. El Sistema verifica y muestra un mensaje de inicio de sesión. Esta prueba fue exitosa.
- **Inicio de sesión con error de autenticación:** El sistema no realiza inicio de sesión al no completarse un campo o tratar de iniciar con un usuario no registrado. Esta prueba fue exitosa. (Anexo 9)

Figura 58. Inicio de sesión con éxito



Fuente: Creación propia

Figura 59. Inicio de sesión con error de autenticación



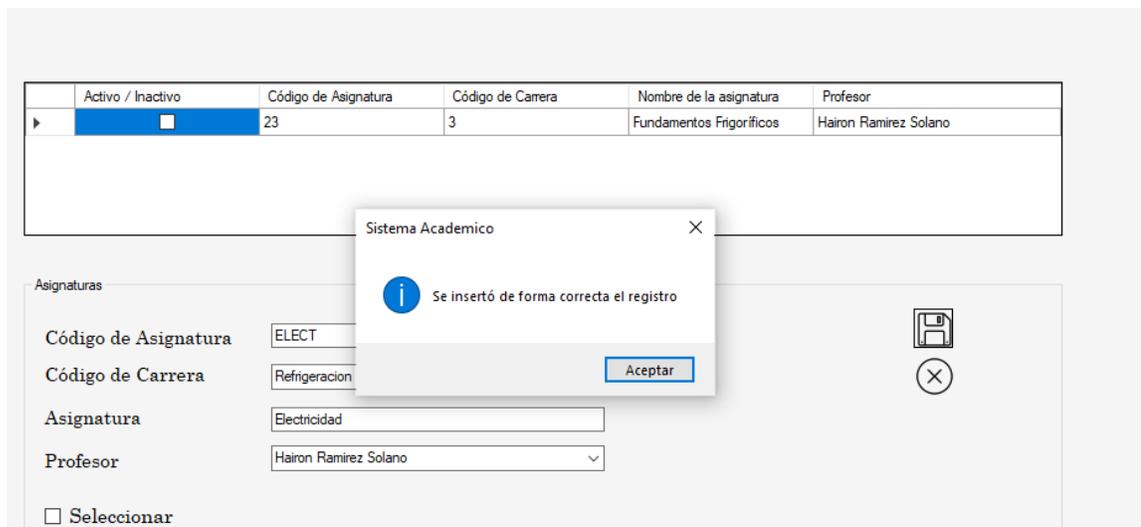
Fuente: Creación propia

5.4.3 Asignaturas

- **Prueba de registro de asignaturas:** Se realizaron pruebas agregando asignaturas con todos los campos requeridos por el sistema. Se verifico que los datos fueran registrados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Prueba de registro de asignaturas con datos incompletos o erróneos:** Esta prueba consistió en tratar de agregar asignaturas al sistema sin completar datos o completar la información. Se verifico que el sistema solicitara completar los campos por medio de un mensaje y que no permitiera guardar los registros sin completar los campos. Esta prueba fue exitosa.
- **Actualización de las asignaturas:** Se realizaron pruebas para la actualización de asignaturas registradas en el sistema. Se verifico que los datos fueran actualizados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Eliminación de asignaturas:** Se prueba seleccionando asignaturas de la lista y eliminándolas del sistema. Se muestra un mensaje verificando la eliminación de las asignaturas. Esta prueba fue exitosa.

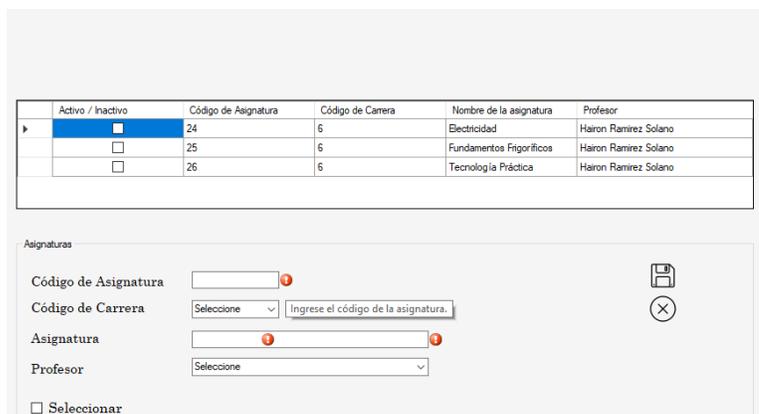
- **Habilitar y deshabilitar asignaturas:** Se prueba seleccionando asignaturas de la lista habilitando y deshabilitando algunas. Se muestra un mensaje verificando las acciones. Esta prueba fue exitosa. (Anexo 10)

Figura 60. Prueba de registro de asignaturas exitosa



Fuente: Creación propia

Figura 61. Prueba de registro de asignaturas con datos incompletos o erróneos



Fuente: Creación propia

Figura 62. Actualización de las asignaturas exitoso

	Activo / Inactivo	Código de Asignatura	Código de Carrera	Nombre de la asignatura	Profesor
	<input type="checkbox"/>	24	6	Electricidad	Hairon Ramirez Solano
▶	<input type="checkbox"/>	25	3	Fundamentos Frigorifico	Hairon Ramirez Solano
	<input type="checkbox"/>	26	6	Tecnología Práctica	Hairon Ramirez Solano

Asignaturas

Código de Asignatura: FUNDFRIGO-A6

Código de Carrera: 3

Asignatura: Fundamentos Frigorifico

Profesor: Hairon Ramirez Solano

Seleccionar

Sistema Academico

 Se actualizó de forma correcta el registro

Fuente: Creación propia

Figura 63. Eliminación de asignaturas

	Activo / Inactivo	Código de Asignatura	Código de Carrera	Nombre de la asignatura	Profesor
	<input type="checkbox"/>	24	6	Electricidad	Hairon Ramirez Solano
▶	<input type="checkbox"/>	25	3	Fundamentos Frigorifico	Hairon Ramirez Solano
	<input type="checkbox"/>	26	6	Tecnología Práctica	Hairon Ramirez Solano

Asignaturas

Código de Asignatura: FUNDFRIGO

Código de Carrera: 3

Asignatura: Fundamentos Frigorificos

Profesor: Hairon Ramirez Solano

Seleccionar

Sistema Academico

 Se actualizó de forma correcta el registro

Fuente: Creación propia

Figura 64. *Deshabilitar asignaturas*

	Seleccionar	Activo / Inactivo	Código de Asignatura	Código de Carrera	Nombre de la asignatura	Profesor
▶	<input checked="" type="checkbox"/>	<input type="checkbox"/>	24	6	Electricidad	Hairon Ramirez Solano
	<input type="checkbox"/>	<input type="checkbox"/>	25	3	Fundamentos Frigorific...	Hairon Ramirez Solano
	<input type="checkbox"/>	<input type="checkbox"/>	26	6	Tecnología Práctica	Hairon Ramirez Solano

Sistema Academico ✕

 Se desactivo el registro: Electricidad

Asignaturas

Código de Asignatura

Código de Carrera

Asignatura

Profesor

Seleccionar







Fuente: Creación propia

Figura 65. *Habilitar asignaturas*

	Seleccionar	Activo / Inactivo	Código de Asignatura	Código de Carrera	Nombre de la asignatura	Profesor
▶	<input checked="" type="checkbox"/>	<input type="checkbox"/>	24	6	Electricidad	Hairon Ramirez Solano
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	25	3	Fundamentos Frigorific...	Hairon Ramirez Solano
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	26	6	Tecnología Práctica	Hairon Ramirez Solano

Sistema Academico ✕

 Se activo el registro: Electricidad

Asignaturas

Código de Asignatura

Código de Carrera

Asignatura

Profesor

Seleccionar





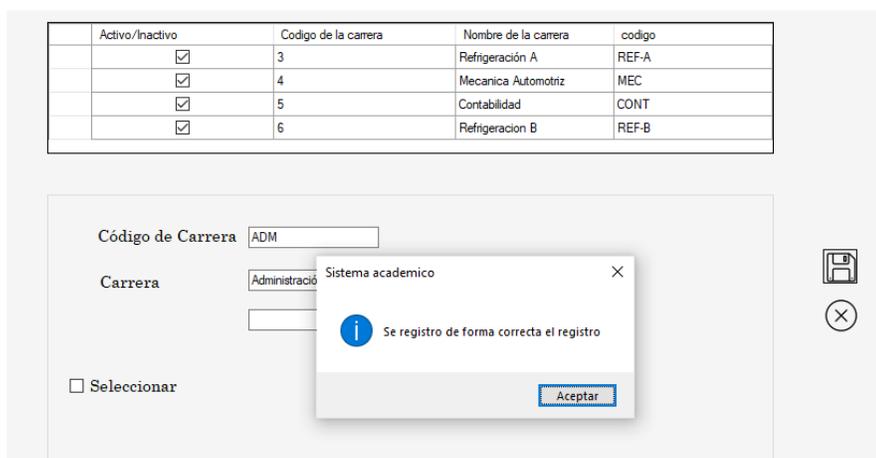


Fuente: Creación propia

5.4.4 Carreras

- **Prueba de registro de carreras:** Se realizaron pruebas agregando carreras con todos los campos requeridos por el sistema. Se verificó que los datos fueran registrados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Prueba de registro de carreras con datos incompletos o erróneos:** Esta prueba consistió en tratar de agregar registros al sistema sin completar datos o completar la información. Se verificó que el sistema solicitara completar los campos por medio de un mensaje y que no permitiera guardar los registros sin completar los campos. Esta prueba fue exitosa.
- **Actualización de las asignaturas:** Se realizaron pruebas para la actualización de las carreras registradas en el sistema. Se verificó que los datos fueran actualizados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Eliminación de asignaturas:** Se prueba seleccionando carreras de la lista y eliminándolas del sistema. Se muestra un mensaje verificando la eliminación de las asignaturas. Esta prueba fue exitosa.
- **Habilitar y deshabilitar asignaturas:** Se prueba seleccionando carreras de la lista habilitando y deshabilitando algunas. Se muestra un mensaje verificando las acciones. Esta prueba fue exitosa. (Anexo 11)

Figura 66. Pruebas de registro de carreras



Fuente: Creación propia

Figura 67. Prueba de registro de carreras con datos incompletos o erróneos

	Activo/Inactivo	Codigo de la carrera	Nombre de la carrera	codigo
	<input checked="" type="checkbox"/>	4	Mecanica Automotriz	MEC
	<input checked="" type="checkbox"/>	5	Contabilidad	CONT
	<input checked="" type="checkbox"/>	6	Refrigeracion B	REF-B
	<input checked="" type="checkbox"/>	8	Administración	ADM

Código de Carrera

Carrera

Seleccionar




Fuente: Creación propia

Figura 68. Actualización de las asignaturas

	Activo/Inactivo	Codigo de la carrera	Nombre de la carrera	codigo
	<input checked="" type="checkbox"/>	4	Mecanica Automotriz	MEC
	<input checked="" type="checkbox"/>	5	Contabilidad	CONT
	<input checked="" type="checkbox"/>	6	Refrigeracion B	REF-B
▶	<input checked="" type="checkbox"/>	8	Administración	ADM

Código de Carrera

Carrera

Seleccionar




Sistema academico

 Se actualizo de forma correcta el registro

Aceptar

Fuente: Creación propia

Figura 69. Eliminación de asignaturas

	Seleccionar	Activo/Inactivo	Codigo de la carrera	Nombre de la carrera	codigo
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	5	Contabilidad	CONT
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	6	Refrigeracion B	REF-B
▶	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	8	Administració	ADM

Código de Carrera

Carrera

Seleccionar

Sistema academico ×

i Se elimino el registro: Administració

Aceptar

↺ ↻ ↻

⊗

□

☑

🗑

Fuente: Creación propia

Figura 70. Deshabilitar asignaturas

	Seleccionar	Activo/Inactivo	Codigo de la carrera	Nombre de la carrera	codigo
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	3	Refrigeración A	REF-A
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	4	Mecanica Automotriz	MEC
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	5	Contabilidad	CONT
▶	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	6	Refrigeracion B	REF-B

Código de Carrera

Carrera

Seleccionar

Sistema academico ×

i Se desactivo el registro: Refrigeracion B

Aceptar

↺ ↻ ↻

⊗

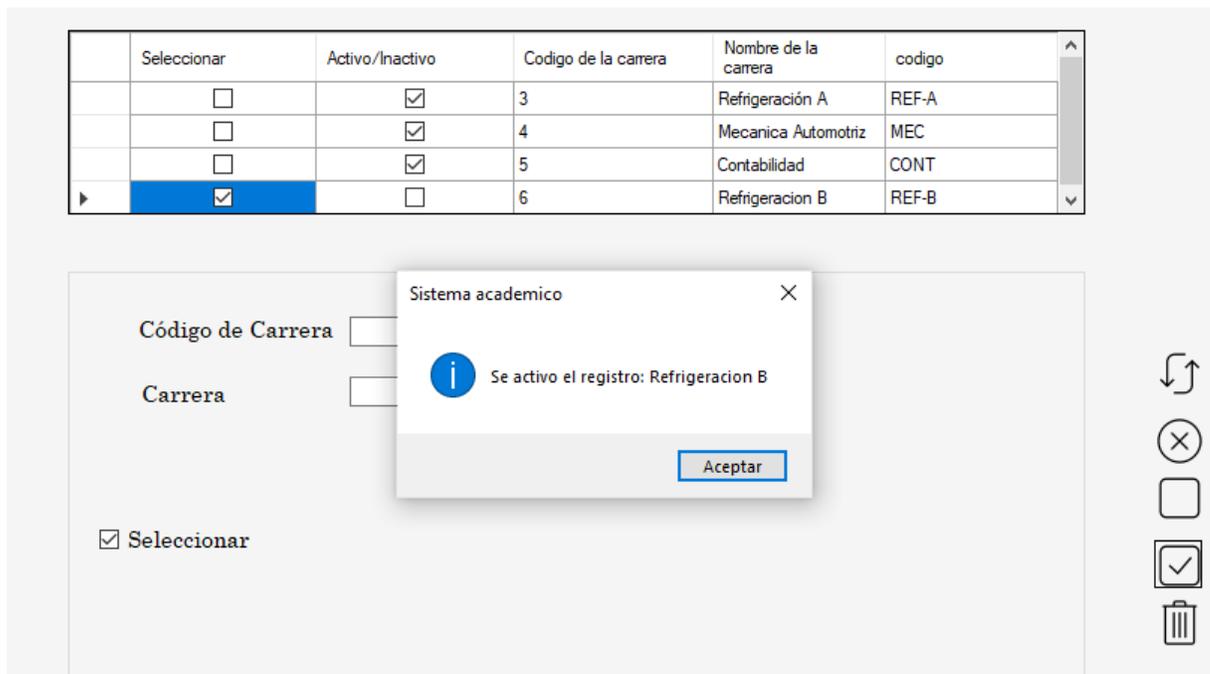
□

☑

🗑

Fuente: Creación propia

Figura 71. Habilitar asignaturas



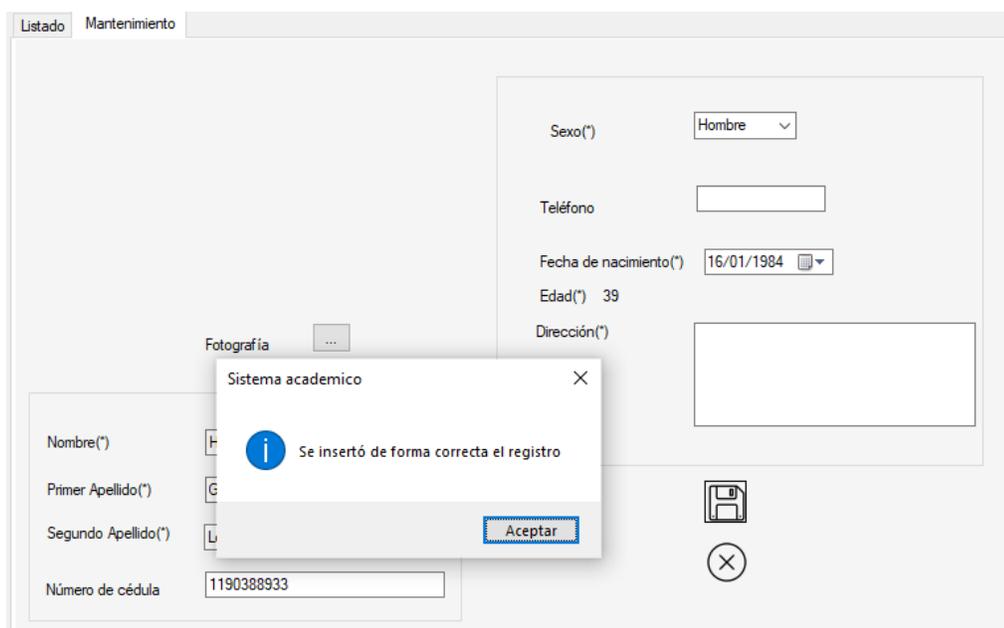
Fuente: Creación propia

5.4.5 Profesores

- **Prueba de registro de profesores:** Se realizaron pruebas agregando profesores con todos los campos requeridos por el sistema. Se verificó que los datos fueran registrados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Prueba de registro de profesores con datos incompletos o erróneos:** Esta prueba consistió en tratar de agregar registros sin completar datos o completar la información con datos en formatos no admitidos por el sistema. Se verificó que el sistema solicitara completar los campos por medio de un mensaje y que no permitiera guardar los registros sin completar los campos o si los datos no tuvieran el formato correcto. Esta prueba fue exitosa.

- **Actualización de los datos de un profesor:** Se realizaron pruebas para la actualización de profesores existentes en el sistema. Se verificó que los datos fueran actualizados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Eliminación de profesores:** Se prueba seleccionando profesores de la lista y eliminándolos del sistema. Se muestra un mensaje verificando la eliminación de los registros. Esta prueba fue exitosa.
- **Habilitar y deshabilitar profesores:** Se prueba seleccionando profesores de la lista habilitando y deshabilitando los registros. Se muestra un mensaje verificando las acciones. Esta prueba fue exitosa. (Anexo 12)

Figura 72. Prueba de registro de profesores



Fuente: Creación propia

Figura 73. Prueba de registro de profesores con datos incompletos o erróneos

The screenshot shows a web form for registering a teacher. The form is divided into two main sections. The left section contains fields for 'Nombre(*)', 'Primer Apellido(*)', 'Segundo Apellido(*)', and 'Número de cédula', each with a red exclamation mark icon indicating an error. The right section contains fields for 'Sexo(*)' (set to 'Hombre'), 'Teléfono', 'Fecha de nacimiento(*)' (set to '16/01/1984'), 'Edad(*)' (set to '39'), and 'Dirección(*)'. A tooltip with the text 'Ingrese el nombre.' is visible over the 'Nombre(*)' field. At the bottom right, there are icons for saving (a floppy disk) and canceling (a circle with an 'X').

Fuente: Creación propia

Figura 74. Actualización de los datos de un profesor

The screenshot shows the same teacher registration form, but now it is used for updating data. The 'Nombre(*)' field is empty, while 'Primer Apellido(*)' contains 'Gonzales', 'Segundo Apellido(*)' contains 'Loria', and 'Número de cédula' contains '1190388933'. The right section shows 'Sexo(*)' as 'Hombre', 'Teléfono' is empty, 'Fecha de nacimiento(*)' as '16/01/1984', 'Edad(*)' as '39', and 'Dirección(*)' is empty. A modal dialog box titled 'Sistema academico' is overlaid on the form, displaying a blue information icon and the message 'Se Actualizó de forma correcta el registro'. The dialog has an 'Aceptar' button. At the bottom right, there are icons for refreshing (a circular arrow) and canceling (a circle with an 'X').

Fuente: Creación propia

Figura 75. Eliminación de profesores

	Seleccionar	Estado	Número de cédula	Nombre	Primer Apellido	Segundo Apellido	direccion	Edad	id_profesor
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	1119500888	Allan	Gau	Jimenez		39	4
▶	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	1190388933	Harry	Gonzales	Loria		39	6
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	111950088	Natanael	Madrigal	Gonzalez		39	3
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	11195008	Hairon	Ramirez	Solano		39	2

Sistema academico ×

i Se elimino el registro: Harry

Aceptar

Seleccionar

Fuente: Creación propia

Figura 76. Habilitar profesor

Listado Mantenimiento

	Seleccionar	Estado	Número de cédula	Nombre	Primer Apellido	Segundo Apellido	direccion	Edad	id_profesor
▶	<input checked="" type="checkbox"/>	<input type="checkbox"/>	1119500888	Allan	Gau	Jimenez		39	4
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	111950088	Natanael	Madrigal	Gonzalez		39	3
	<input type="checkbox"/>	<input checked="" type="checkbox"/>	11195008	Hairon	Ramirez	Solano		39	2

Sistema academico ×

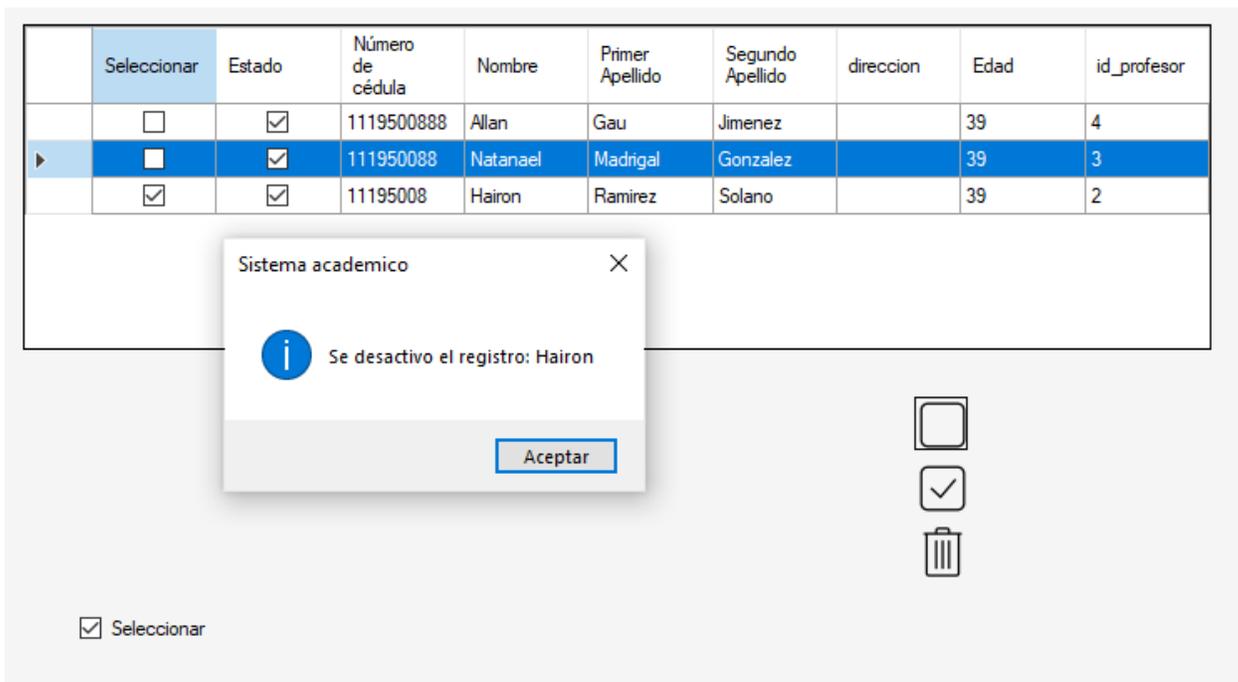
i Se activo el registro: Allan

Aceptar

Seleccionar

Fuente: Creación propia

Figura 77. Deshabilitar profesor



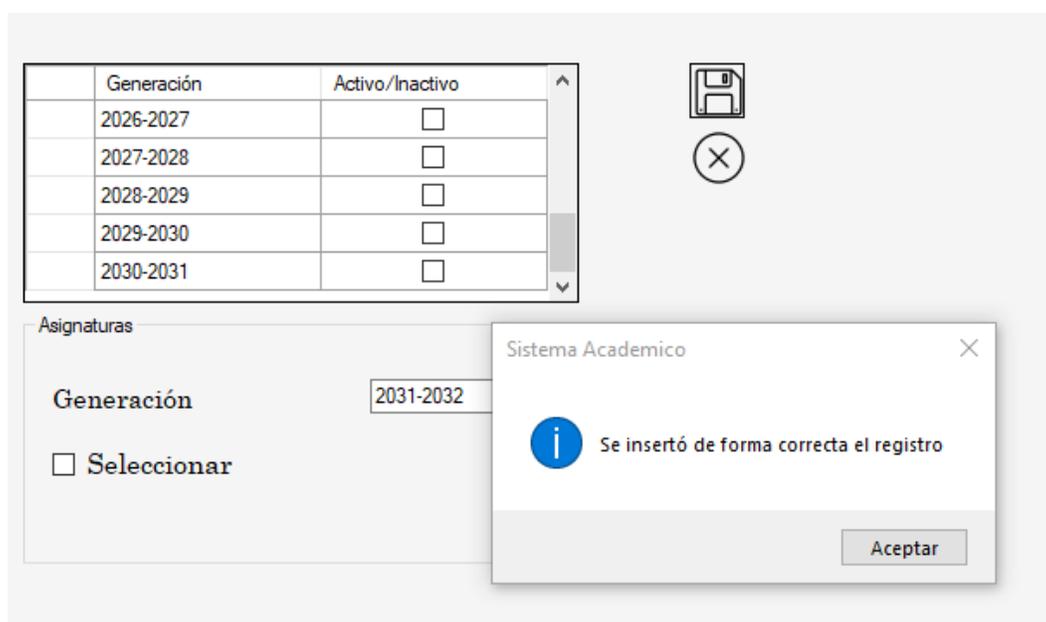
Fuente: Creación propia

5.4.6 Generación

- **Prueba de registro de generación:** Se realizaron pruebas agregando registros con todos los campos requeridos por el sistema. Se verificó que los datos fueran registrados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Prueba de registro con datos incompletos o erróneos:** Esta prueba consistió en tratar de agregar registros sin completar datos o completar la información con datos en formatos no admitidos por el sistema. Se verificó que el sistema solicitara completar los campos por medio de un mensaje y que no permitiera guardar los registros sin completar los campos o si los datos no tuvieran el formato correcto. Esta prueba fue exitosa.

- **Eliminación de registros:** Se prueba seleccionando registros de la lista y eliminándolas del sistema. Se muestra un mensaje verificando la eliminación de los registros. Esta prueba fue exitosa.
- **Habilitar y deshabilitar registros:** Se prueba seleccionando registros de la lista habilitando y deshabilitando algunos. Se muestra un mensaje verificando las acciones. Esta prueba fue exitosa. (Anexo 13)

Figura 78. Prueba de registro de generación



Fuente: Creación propia

Figura 79. Prueba de registro con datos incompletos o erróneos

The screenshot shows a web application interface. At the top, there is a table with two columns: 'Generación' and 'Activo/Inactivo'. The table contains five rows of data:

Generación	Activo/Inactivo
2015-2016	<input type="checkbox"/>
2016-2017	<input type="checkbox"/>
2017-2018	<input type="checkbox"/>
2018-2019	<input type="checkbox"/>
2019-2020	<input type="checkbox"/>

Below the table, there is a section titled 'Asignaturas'. It contains a label 'Generación' followed by an empty text input field. To the right of the input field is a red error icon. Below the input field is a checkbox labeled 'Seleccionar'. A tooltip message 'Ingrese la generación.' is displayed next to the error icon. To the right of the table and form, there are two icons: a save icon (floppy disk) and a close icon (circle with an X).

Fuente: Creación propia

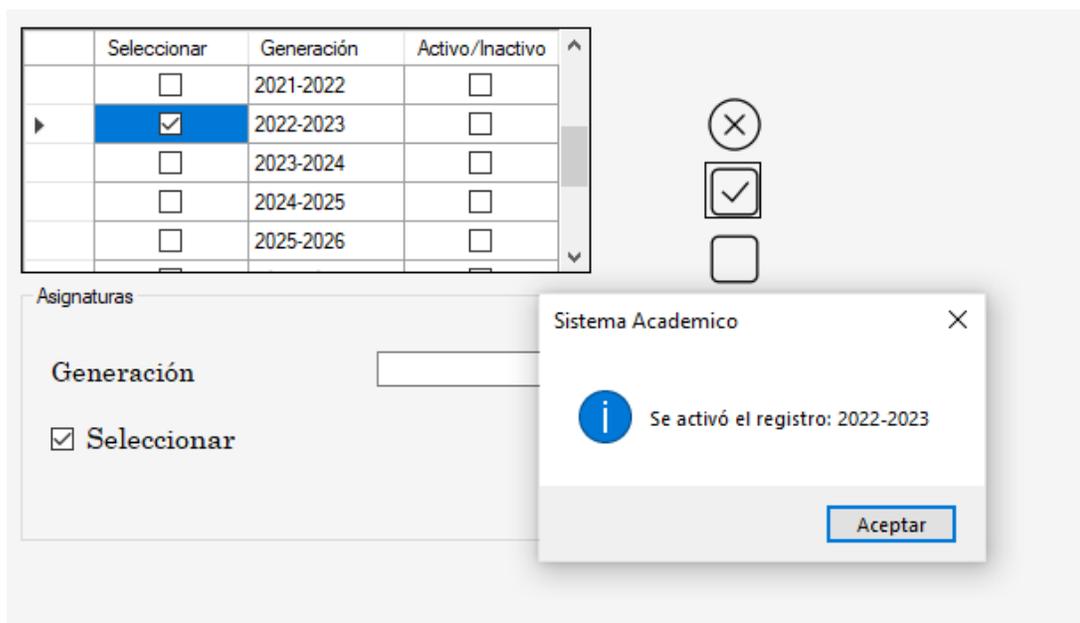
Figura 80. Eliminación de registros

The screenshot shows a web application interface. At the top, there is a table with four columns: 'Seleccionar', 'Generación', and 'Activo/Inactivo'. The table contains five rows of data:

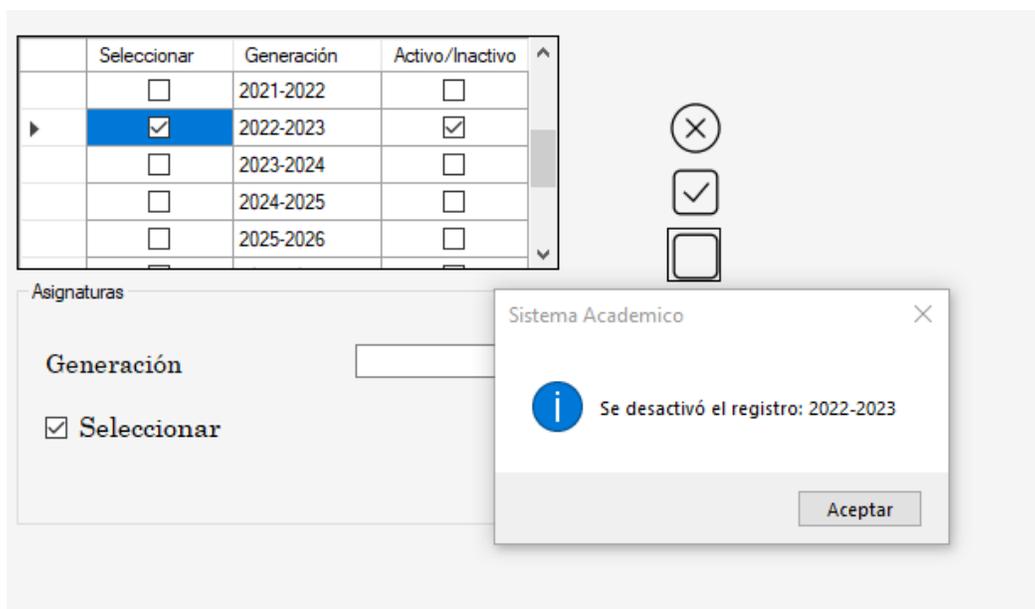
Seleccionar	Generación	Activo/Inactivo
<input type="checkbox"/>	2027-2028	<input type="checkbox"/>
<input type="checkbox"/>	2028-2029	<input type="checkbox"/>
<input type="checkbox"/>	2029-2030	<input type="checkbox"/>
<input type="checkbox"/>	2030-2031	<input type="checkbox"/>
<input checked="" type="checkbox"/>	2031-2032	<input checked="" type="checkbox"/>

Below the table, there is a section titled 'Asignaturas'. It contains a label 'Generación' followed by an empty text input field. Below the input field is a checkbox labeled 'Seleccionar', which is checked. To the right of the table and form, there are three icons: a close icon (circle with an X), a checkmark icon, and an empty square icon. A dialog box titled 'Sistema Academico' is open in the foreground. The dialog box contains an information icon (i) and the text 'Se eliminó el registro: 2031-2032'. At the bottom of the dialog box is an 'Aceptar' button.

Fuente: Creación propia

Figura 81. Habilitar registros de generación

Fuente: Creación propia

Figura 82. Deshabilitar registro de generación

Fuente: Creación propia

5.4.7 *Estudiantes*

- **Prueba de registro de estudiantes:** Se realizaron pruebas agregando estudiantes con todos los campos requeridos por el sistema. Se verificó que los datos fueran registrados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Prueba de registro de estudiantes con datos incompletos o erróneos:** Esta prueba consistió en tratar de agregar asignaturas al sistema sin completar datos o completar la información. Se verifico que el sistema solicitara completar los campos por medio de un mensaje y que no permitiera guardar los registros sin completar los campos. Esta prueba fue exitosa.
- **Actualización de los registros de estudiantes:** Se realizaron pruebas para la actualización de los registros. Se verificó que los datos fueran actualizados correctamente en el sistema por lo que las pruebas fueron exitosas.
- **Eliminación de los registros de estudiantes:** Se prueba seleccionando registros de la lista y eliminándolos del sistema. Se muestra un mensaje verificando la eliminación. Esta prueba fue exitosa.
- **Habilitar y deshabilitar registros de estudiantes:** Se prueba seleccionando estudiantes de la lista habilitando y deshabilitando algunos registros. Se muestra un mensaje verificando las acciones. Esta prueba fue exitosa. (Anexo 14)

Figura 83. Prueba de registro de estudiantes

The screenshot shows a web application interface for student registration. A modal window titled "Sistema Academico" is displayed in the center, containing the message "Se insertó de forma correcta el registro" (The record was inserted correctly) and an "Aceptar" (Accept) button. The background form is partially visible, showing fields for student information (Name: Hairom, Surname: Ramirez, Solano, ID: 111950088) and family information (Mother: Maria Solano Acuña, Father: Miguel Ramirez Henriquez, ID: 884939955). The form also includes fields for career (Refrigeracion B), generation (2022-2023), school (Mauro Fernandez Acuña), and school (Jose Rafael Araya). A note at the bottom indicates that an asterisk (*) denotes mandatory data.

Fuente: Creación propia

Figura 84. Prueba de registro de estudiantes con datos incompletos o erróneos

The screenshot shows the same student registration form as in Figure 83, but with several validation errors. Red exclamation mark icons are placed next to the following fields: "Número de registro" (REF-B-), "Carrera" (Seleccione), "Generación" (Seleccione), "Colegio" (empty), "Escuela" (empty), "Nombre de la madre" (empty), "Nombre del padre" (empty), "Cédula del padre" (empty), "Estado Civil" (Seleccione), "Sexo" (Seleccione), "Correo electrónico" (empty), "Teléfono" (empty), "Teléfono de la casa" (empty), "Teléfono de emergencia" (empty), "Fecha de nacimiento" (16/01/1984), "Persona en el hogar" (Seleccione), "Subido" (Seleccione), "Cantidad de Hijos" (Seleccione), and "Escolaridad" (Seleccione una opción). A note at the bottom indicates that an asterisk (*) denotes mandatory data.

Fuente: Creación propia

Figura 85. Actualización de los registros de estudiantes

The screenshot shows a web interface for updating student records. On the left, there is a profile section with a placeholder for a photo and fields for Name, Surnames, ID number, Civil Status, Sex, Email, Phone, and Birth Date. On the right, there are fields for Registration Number, Career, Generation, School, and Mother's Name. A central dialog box titled 'Sistema Academico' displays the message 'Se actualizo de forma correcta el registro' (Record updated correctly) with an 'Aceptar' (Accept) button. At the bottom, there are icons for refresh and close, and a note: '(*) Indica que el dato es obligatorio'.

Fuente: Creación propia

Figura 86. Eliminación de los registros de estudiantes

The screenshot shows a list of student records in a table. The table has columns for selection, active status, code of career, ID number, name, first surname, second surname, and non-active status. A dialog box titled 'Sistema Academico' is overlaid on the table, displaying the message 'Se elimino el registro: Hairon' (Record deleted: Hairon) with an 'Aceptar' (Accept) button. Below the table, there are checkboxes for 'Seleccionar' (Select) and icons for refresh and delete. The total number of records is shown as 18.

	Activo /	Código de Carrera	Número Cédula	Nombre	Primer apellido	Segundo Apellido	Non
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118690536	Nathan Moises	Cortes	Rivera	Shirley
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118650077	Joiney	Flores	Salas	Yesse
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118250540	Makel	Jimenez	Monge	Raquel
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118090492	Moises	Lizano	Hernandez	Karen
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118410669	Irvin	Mora	Jara	Gustel
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	117830379	Alejandro	Penant	Barahona	Jennif
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	111950088	Hairon	Ramirez	Solano	Maria
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B			Ramos	Tjeerino	Leslie
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B			Rovira	Vargas	Ana Lu
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B			neth	Baldano	Alba E
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B			dall	Sanamusa	Maria
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B			Joafv	Thomas	Laura
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B			in	Vindas	Laura

Fuente: Creación propia

Figura 87. *Habilitar registros de estudiantes*

The screenshot shows a web application interface for student record management. At the top, there are tabs for 'Listado' and 'Mantenimiento'. Below the tabs, there are filters for 'Generación' (set to 2022-2023) and 'Carrera'. A search bar is also present. The main area contains a table with columns: 'Seleccionar', 'Activo /', 'Código de Carrera', 'Número Cédula', 'Nombre', 'Primer apellido', 'Segundo Apellido', and 'Non'. The first row is highlighted in blue, and a dialog box titled 'Sistema Academico' is overlaid on the table, displaying the message 'Se activo el registro: Oscar' and an 'Aceptar' button. Below the table, there is a 'Seleccionar' checkbox, a 'Total registros: 17' label, and three icons: a checkmark, a square, and a trash can.

Seleccionar	Activo /	Código de Carrera	Número Cédula	Nombre	Primer apellido	Segundo Apellido	Non
<input checked="" type="checkbox"/>	<input type="checkbox"/>	REF-B	504450780	Oscar	Agüero	Cardenas	Kathia
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118390234	Gustavo	Betancourt	Sanchez	Maria
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118580201	Merilyn Dayana	Campos	Gonzalez	Ana Lu
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	119190879	Mikenver	Castillo	Sanchez	Adina
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	119070047	Kellor	Chavarria	Chavarria	Priscill
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118690536	Nathan Moises	Cortes	Rivera	Shirley
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118650077	Josiney	Flores	Salas	Yesse
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		ikel	Jimenez	Monge	Raque
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		ises	Lizano	Hernandez	Karen
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		in	Mora	Jara	Guisel
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		jandro	Penant	Barahona	Jennif
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		ul	Ramos	Tijeirino	Leslie
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		leria	Rovira	Vargass	Ana Lu
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		nneth	Saenz	Baltodano	Alba E

Fuente: Creación propia

Figura 88. *Deshabilitar registro de estudiantes*

The screenshot shows the same web application interface as Figure 87. The table of students is visible, and a dialog box titled 'Sistema Academico' is overlaid, displaying the message 'Se desactivo el registro: Oscar' and an 'Aceptar' button. The 'Activo /' column for the first row now shows a checked checkbox, indicating the record is active. Below the table, there is a 'Seleccionar' checkbox, a 'Total registros: 17' label, and three icons: a checkmark, a square, and a trash can.

Seleccionar	Activo /	Código de Carrera	Número Cédula	Nombre	Primer apellido	Segundo Apellido	Non
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	504450780	Oscar	Agüero	Cardenas	Kathia
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118390234	Gustavo	Betancourt	Sanchez	Maria
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118580201	Merilyn Dayana	Campos	Gonzalez	Ana Lu
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	119190879	Mikenver	Castillo	Sanchez	Adina
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	119070047	Kellor	Chavarria	Chavarria	Priscill
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118690536	Nathan Moises	Cortes	Rivera	Shirley
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B	118650077	Josiney	Flores	Salas	Yesse
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		ikel	Jimenez	Monge	Raque
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		ises	Lizano	Hernandez	Karen
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		in	Mora	Jara	Guisel
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		jandro	Penant	Barahona	Jennif
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		ul	Ramos	Tijeirino	Leslie
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		leria	Rovira	Vargass	Ana Lu
<input type="checkbox"/>	<input checked="" type="checkbox"/>	REF-B		nneth	Saenz	Baltodano	Alba E

Fuente: Creación propia

5.4.8 Asistencia

- **Prueba de registro de asistencia:** Se realizaron pruebas agregando registros de asistencia. Se verificó que los datos fueran registrados correctamente en el sistema seleccionando los estudiantes de una lista filtrándolos por generación y carrera. Las pruebas fueron exitosas.
- **Prueba de registro de asistencia con datos incompletos o erróneos:** Esta prueba consistió en tratar de agregar el registro de asistencia de algunos estudiantes al sistema sin completar datos o completar la información. Se verificó que el sistema solicitara completar los campos por medio de un mensaje y que no permitiera guardar los registros sin completar los campos. Esta prueba fue exitosa.
- **Actualización de los registros de asistencia de estudiantes:** Se realizaron pruebas para la actualización de los registros. Se verificó que los datos fueran actualizados correctamente en el sistema por lo que las pruebas fueron exitosas. (Anexo 15)

Figura 89. Prueba de registro de asistencia

The screenshot shows a web application interface for student attendance. At the top, there are dropdown menus for 'Carrera:' and 'Generación', and a search bar with the name 'Nathan Moises Cortes Rivera'. Below this is a table with columns: Estudiante, Fecha, Entrada, Salida, and Estado. The table lists several students with their respective dates and attendance status. A modal dialog box is open in the center, titled 'Sistema Academico', with an information icon and the message 'Se insertó de forma correcta el registro'. Below the message is an 'Aceptar' button. At the bottom of the interface, there are input fields for 'Fecha' (29/04/2023), 'Reporte' (Presente), 'Hora Salida' (0:00), and 'Hora de Ingreso' (0:00). There are also icons for a printer and a close button.

Estudiante	Fecha	Entrada	Salida	Estado
Oscar Aguero Cardenas	29/04/2023			Presente
Gustavo Betancourt Sa...	29/04/2023			Presente
Merilyn Dayana Campos...	29/04/2023			Presente
Mikenver Castillo Sanchez	29/04/2023			Presente
Keilor Chavarria Chavarria	29/04/2023			Ausente

Fuente: Creación propia

Figura 90. Prueba de registro de asistencia con datos incompletos o erróneos

Carrera: Seleccione  

Generación: Seleccione

Estudiante	Fecha	Entrada	Salida	Estado
Oscar Agüero Cardenas	29/04/2023			Presente
Gustavo Betancourt Sa...	29/04/2023			Presente
Merilyn Dayana Campos...	29/04/2023			Presente
Mikenver Castillo Sanchez	29/04/2023			Presente
Keilor Chavarria Chavar...	29/04/2023			Ausente
Nathan Moises Cortes R...	29/04/2023			Presente
Josiney Flores Salas	29/04/2023			Presente

Estudiante

- Oscar Agüero Cardenas
- Gustavo Betancourt Sanchez
- Merilyn Dayana Campos Gonzalez
- Mikenver Castillo Sanchez
- Keilor Chavarria Chavarria
- Nathan Moises Cortes Rivera
- Josiney Flores Salas**
- Maikel Jimenez Monge
- Moises Lizano Hernandez
- Irvin Mora Jara

Fecha: 29/04/2023 Hora Salida: 0 : 00

Reporte: Seleccione Hora de Ingreso: 0 : 00

Fuente: Creación propia

Figura 91. Actualización de los registros de asistencia de estudiantes

Carrera: Refrigeracion B  

Generación: Seleccione

Estudiante	Fecha	Entrada	Salida	Estado
Oscar Agüero Cardenas	29/04/2023			Ausente
Gustavo Betancourt Sa...	29/04/2023			Presente
Merilyn Dayana Campos...	29/04/2023			Presente
Mikenver Castillo Sanchez	29/04/2023			Presente
Keilor Chavarria Chavar...	29/04/2023			Ausente
Nathan Moises Cortes R...	29/04/2023			Presente
Josiney Flores Salas	29/04/2023			Presente
Josiney Flores Salas	24/04/2023			Ausente

Sistema Academico X

 Se actualizó de forma correcta el registro

Fecha: 29/04/2023 Hora Salida: 0 : 00

Reporte: Presente Hora de Ingreso: 0 : 00

Fuente: Creación propia

5.4.9 Calificaciones

- **Prueba de registro de calificaciones:** Se realizaron pruebas agregando registros de calificaciones. Se verificó que los datos fueran registrados correctamente en el sistema seleccionando los estudiantes de una lista filtrándolos por generación y carrera. Las pruebas fueron exitosas.
- **Prueba de registro de calificaciones con datos incompletos o erróneos:** Esta prueba consistió en tratar de agregar el registro de calificaciones de algunos estudiantes al sistema sin completar datos o completar la información. Se verificó que el sistema solicitara completar los campos por medio de un mensaje y que no permitiera guardar los registros sin completar los campos. Esta prueba fue exitosa.
- **Actualización de los registros de asistencia de estudiantes:** Se realizaron pruebas para la actualización de los registros. Se verificó que los datos fueran actualizados correctamente en el sistema por lo que las pruebas fueron exitosas. (Anexo 16)

Figura 92. Prueba de registro de calificaciones

NOTA MENSUAL REFRIGERACIÓN B 119070047

Profesor: Fecha:

Materia:

Generación: Mes:

ESTUDIANTE	SOCIOAFECTIVO	PRUEBA MENSUAL	PORCENTAJE PRACTICA	PROMEDIO
Oscar Agüero Cárdenas	8,00	64,00	0,00	82
Gustavo Betancourt Sanchez	7,00			
Merlyn Dayana Campos Gonzalez	7,00			
Mkenver Castillo Sanchez	10,00			

Sistema academico

Se insertó de forma correcta el registro

ESTUDIANTE

Oscar Agüero Cárdenas

Gustavo Betancourt Sanchez

Merlyn Dayana Campos Gon...

Mkenver Castillo Sanchez

Kalor Chavarria Chavarria

Nathan Moises Cortes Rivera

Josiney Flores Salas

Maikel Jimenez Monge

Moises Lizano Hernandez

Porcentaje prueba Teórica	Calificación Socioafectiva	Porcentaje Práctica	Nota obtenida	0.00
Detalle 1: <input type="text" value="54"/> Detalle 2: <input type="text" value="0.00"/> Detalle 3: <input type="text" value="0.00"/>	<input type="text" value="10"/>	<input type="text" value="6"/>	<input type="text" value="70.00"/>	

Fuente: Creación propia

Figura 93. Prueba de registro de calificaciones con datos incompletos o erróneos

NOTA MENSUAL REFRIGERACIÓN B

Profesor: Seleccione Fecha: 30/05/2023

Materia: Seleccione

Generación: Seleccione Mes: Seleccione

ESTUDIANTE	SOCIOAFECTIV	PRUEBA MENSUAL	PORCENTAJE PRACTICA	PROMEDIO
Irvin Mora Jara	6,00	54,00	0,00	60
Alejandro Penant Barah...	7,00	66,00	0,00	83
Saul Ramos Tijeerino	6,00	48,00	0,00	64
Valeria Rovira Vargass	8,00	54,00	0,00	72
Kenneth Saenz Baltoda...	6,00	0,00	0,00	16
Kendall Sanamusia Sán...	6,00	49,00	0,00	65
Omar Joafv Thomas Ob...	7,00	60,00	0,00	77
Kervin Vindas Madrigal	7,00	64,00	0,00	81

ESTUDIANTE

Moises Lizano Hernandez

Irvin Mora Jara

Alejandro Penant Barahona

Saul Ramos Tijeerino

Valeria Rovira Vargass

Kenneth Saenz Baltodano

Kendall Sanamusia Sánchez

Omar Joafv Thomas Obando

Kervin Vindas Madrigal

Seleccione un estudiante de la lista.

Porcentaje prueba Teórica

Detalle 1 Detalle 2 Detalle 3

0,00 0,00 0,00

Calificación Socioafectiva

0,00

Porcentaje Práctica

0,00

Nota obtenida

0,00

0,00

Fuente: Creación propia

Hay que tomar en cuenta que se diseñó un módulo de calificaciones para cada especialidad y se filtran los estudiantes desde una lista por generación y Asignatura.

5.4.10 Reportes

- **Reporte de estudiantes:** Se crea de forma correcta un reporte que muestra la información de los estudiantes filtrando por generación, género y por escolaridad. Se comprobó que permitiera imprimir o guardar los reportes en formato Word, PDF o Excel.
- **Reporte de asistencia:** Se crea de forma correcta un reporte que muestra la asistencia de los estudiantes filtrando por especialidad, generación, además de filtrar entre fechas. Se comprobó que permitiera imprimir o guardar los reportes en formato Word, PDF o Excel. (Anexo 17)

Figura 94. Reporte de información básica de los estudiantes

FrmReporteListaEstudiantes

de 1 de 2 Buscar | Siguiente

FUNDACIÓN HEDWIG Y ROBERT SAMUEL
CENTRO DE FORMACIÓN TÉCNICA, COSTA RICA
INFORMACIÓN DE ESTUDIANTES 21/06/2023 11:25:30

Generación	Especialidad	Nombre	Primer Apellido	Segundo Apellido	Teléfono	Edad	Escolaridad
2022-2023	REF-B	Oscar	Aguero	Cardenas	84400276	21	11
2022-2023	MEC	asdfgsasd	ASDFDGA	asdfghg	3456789765	27	9
2022-2023	REF-B	Gustavo	Betancourt	Sanchez	85204742	21	10
2022-2023	REF-B	Merilyn Dayana	Campos	Gonzalez	60809048	21	8
2022-2023	REF-B	Mikenver	Castillo	Sanchez	85005821	19	11
2022-2023	REF-B	Keilor	Chavarría	Chavarría	85470748	19	10
2022-2023	REF-B	Nathan Moises	Cortes	Rivera	72675289	20	11
2022-2023	REF-B	Josiney	Flores	Salas	71221186	20	10
2022-2023	REF-B	Maikel	Jimenez	Monge	62285901	22	11
2022-2023	REF-B	Moises	Lizano	Hernandez	86640500	22	9
2022-2023	REF-B	Irvin	Mora	Jara	85599467	21	12
2022-2023	REF-B	Alejandro	Penant	Barahona	62979318	23	11
2022-2023	REF-B	Saul	Ramos	Tijeerino	71032468	22	12
2022-2023	REF-B	Valeria	Rovira	Vargass	70248421	21	11
2022-2023	REF-B	Kenneth	Saenz	Baltodano	61343454	25	6
2022-2023	REF-B	Kendall	Sanarrusia	Sánchez	89410346	21	10
2022-2023	REF-B	Omar Joafr	Thomas	Obando	72993132	19	11
2022-2023	REF-B	Kervin	Vindas	Madrigal	60295172	20	11

Fuente: Creación propia

Figura 95. Reporte según género

de 1 de 2 Buscar | Siguiente

FUNDACIÓN HEDWIG Y ROBERT SAMUEL
CENTRO DE FORMACIÓN TÉCNICA, COSTA RICA
INFORMACIÓN DE ESTUDIANTES SEGUN GÉNERO 21/06/2023 11:29:24

2022-2023
REF-B

CANTIDAD DE MUJERES:	<input type="text" value="2"/>	CANTIDAD DE HOMBRES:	<input type="text" value="15"/>
PORCENTAJE DE MUJERES:	<input type="text" value="12"/>	PORCENTAJE DE HOMBRES:	<input type="text" value="88"/>

Fuente: Creación propia

Figura 96. Reporte según la escolaridad

1 de 1 100 % Buscar | Siguiente

FUNDACIÓN HEDWIG Y ROBERT SAMUEL
CENTRO DE FORMACIÓN TÉCNICA .COSTA RICA
INFORMACIÓN DE ESTUDIANTES POR ESCOLARIDAD

2022-2023
REF-B

Estudiantes con sexto	1	%	5,9
Estudiantes con séptimo	0	%	0,0
Estudiantes con octavo	1	%	5,9
Estudiantes con noveno:	1	%	5,9
Estudiantes con décimo	4	%	23,5
Estudiantes con undécimo:	8	%	47,1
Estudiantes con duodécimo:	2	%	11,8

Fuente: Creación propia

Figura 97. Reporte de Calificaciones por especialidad, materia y generación

FUNDACIÓN HEDWIG Y ROBERT SAMUEL
CENTRO DE FORMACIÓN TÉCNICA .COSTA RICA
REFRIGERACIÓN Y AIRE ACONDICIONADO B

21/06/2023 21:41:11

Generación:
Materia: Electricidad
Profesor: Hairon Ramirez Solano
Mes:

Alumno	Promedio
Oscar Aguero Cardenas	82
Gustavo Betancourt Sanchez	81
Merylyn Dayana Campos Gonzalez	71
Mikenver Castillo Sanchez	65
Keilor Chavarria Chavarria	70
Nathan Moises Cortes Rivera	81
Josiney Flores Salas	81
Maikel Jimenez Monge	86
Moises Lizano Hernandez	71
Irvin Mora Jara	60
Alejandro Penant Barahona	83
Saul Ramos Tijerino	64
Valeria Rovira Vargass	72
Kenneth Saenz Baltodano	16
Kendall Sanarrusia Sánchez	65
Omar Joafv Thomas Obando	77
Kervin Vindas Madrigal	81

Fuente: Creación propia

Figura 98. *Reporte de notas entre fechas y por estudiante*

FUNDACIÓN HEDWIG Y ROBERT SAMUEL
CENTRO DE FORMACIÓN TÉCNICA, COSTA RICA
REFRIGERACIÓN Y AIRE ACONDICIONADO B

21/06/2023 21:47:57

Alumno	Materia	Promedio	Periodo	Mes
Oscar Agüero Cardenas	Electricidad	50	Mesual	Junio
Oscar Agüero Cardenas	Electricidad	82	Mesual	Mayo

Fuente: Creación propia

El sistema se está utilizando junto con los métodos habituales de recopilación de información de la Fundación Samuel. Actualmente a la espera de la aprobación de la sede para instalar la base de datos en el servidor. Esto permitiría una mayor accesibilidad a la información y un manejo más eficiente de los datos.

Capítulo VI Conclusiones y recomendaciones del proyecto

6.1 Conclusiones

Objetivo 1 - Identificar los requerimientos específicos para el diseño y desarrollo de un sistema de gestión de datos que satisfaga las necesidades de la Fundación Hedwig y Robert Samuel, asegurando la eficiencia, calidad y seguridad del Sistema.

Mediante la realización de sesiones en conjunto con el personal de las diferentes áreas de trabajo de la Fundación Hedwig Y Robert Samuel, se pudo observar cómo los empleados llevan a cabo sus labores diarias, las experiencias y sugerías relacionadas con la gestión de datos académicos y los flujos de trabajo permitió definir los requerimientos necesarios para el desarrollo de un sistema de gestión de información a medida.

Además de la observación directa, se realizaron entrevistas individuales con las personas intervienen en los procesos relacionados con el registro o manipulación de la información. Estas entrevistas permitieron obtener mejores detalles sobre las funciones con las que debe contar el sistema de gestión de información.

Esta metodología permitió obtener una mejor percepción de los procesos relacionados con la obtención, almacenamiento y manipulación de la información, lo que permitió comprender con mayor claridad las necesidades a cubrir y los requerimientos básicos del sistema a desarrollar. Toda la información recopilada fue clave para elaborar un documento de especificación de requerimientos en el que se detallaron las características específicas de algunos de los procesos de gestión de información.

En conclusión, se puede afirmar que aplicar buenas técnicas para la obtención de requerimientos es esencial para obtener un producto que se ajuste lo más posible a las necesidades del cliente. Además, esto asegura un tiempo de entrega óptimo y una calidad de *software* satisfactoria. Por lo tanto, la metodología utilizada para levantar los requerimientos permitió comprender las necesidades del cliente de manera efectiva y así desarrollar un sistema de gestión de información a medida que cumple con los requisitos específicos de la Fundación Hedwig Y Robert Samuel.

Objetivo 2 - Diseñar un sistema académico de gestión de datos que permita a la Fundación Hedwig y Robert Samuel gestionar y mantener la información de los estudiantes, docentes, y programas académicos de manera eficiente y precisa.

Durante el proceso de desarrollo del sistema de gestión académica, uno de los primeros pasos fue llevar a cabo el diseño y desarrollo del ambiente de la base de datos y del ambiente de desarrollo del sistema. Para poder expresar los requerimientos específicos del sistema, se utilizó la técnica de casos de uso, la cual permite describir de forma detallada las interacciones del usuario con el sistema en términos de acciones y respuestas. Asimismo, se elaboraron diagramas de caso de uso para ilustrar el funcionamiento del sistema.

El diseño de la base de datos requirió una constante revisión y adaptación, ya que es la pieza clave en la que se almacena la información que se utilizará en el momento necesario. En un inicio, el diseño se basó en los requerimientos que se obtuvieron a través del levantamiento de información, pero durante el proceso de desarrollo, surgieron nuevas necesidades y requerimientos que se debieron integrar al diseño de la base de datos. Por lo tanto, se realizó una revisión constante del diseño para asegurarse de que fuera lo más preciso y correcto posible.

Por otro lado, el diseño del sistema se centró en cumplir con los requerimientos específicos que se obtuvieron a través de las sesiones de trabajo con las diferentes áreas de la Fundación Hedwig y Robert Samuel. Se buscó desarrollar un sistema que cubriera las necesidades de los usuarios y que facilitara el trabajo diario de los mismos. Para lograr esto, se utilizaron varios formularios, siendo el más importante el registro de la información de los estudiantes. Para este formulario, se planificó detalladamente su funcionalidad y se utilizó como base para el resto del proyecto. Además, se llevaron a cabo reuniones con las jefaturas para comprobar que los diferentes formularios cumplieran con los requerimientos y realizar los cambios necesarios.

Es importante mencionar que todo el proceso de diseño y desarrollo del sistema se basó en el análisis minucioso de los requerimientos específicos de la Fundación Hedwig y Robert Samuel. Se implementaron documentos y plantillas para describir los requerimientos iniciales, revisarlos y registrar los cambios. Este proceso permitió desarrollar un sistema que cumple con las necesidades específicas de la organización y que facilita la gestión académica en diferentes áreas, como la gestión de notas, asistencias, registros, entre otras.

En resumen, el diseño de la base de datos y del sistema de gestión académica fue un proceso crucial para el éxito del proyecto. Se utilizó la técnica de casos de uso para expresar los requerimientos específicos del sistema, se realizaron constantes revisiones del diseño de la base de datos y se planificó detalladamente la funcionalidad de los formularios. Todo esto permitió

desarrollar un sistema que cumple con las necesidades específicas de la Fundación Hedwig y Robert Samuel y que optimiza la gestión académica en diferentes áreas.

Objeto 3 - Desarrollar cada uno de los módulos de la aplicación de gestión académica, con base en los requerimientos definidos por la Fundación Hedwig y Robert Samuel, para garantizar su funcionalidad y eficiencia en el proceso de gestión académica.

Se desarrollaron los diferentes Formularios usando el IDE de Visual Studio y el lenguaje C#, siendo el formulario de ingreso de información el primero y tomado como base para el desarrollo del resto de los formularios. El sistema se desarrolló para que los formularios compartan información entre ellos y esta información sea almacenada en base de datos en sus campos correspondientes. Conforme se fueron desarrollando los formularios se presentaron a la dirección ejecutiva de la Fundación Samuel, que propusieron cambios en la parte visual y en la información presentada en los formularios. Los cambios buscaron garantizar que los formularios fueran lo más intuitivos, eficientes, claros y precisos.

Se trabajó en que la interfaz mostrada al usuario permitiera una navegación fluida entre los campos y secciones. Se tuvo en cuenta la estructura y el formato de los campos y etiquetas para asegurar que fueran fáciles de completar y comprender.

Se realizaron validaciones de datos para garantizar que la información ingresada fuera correcta y coherente. Además, se incluyeron ayudas visuales y mensajes de error claros para guiar a los usuarios en caso de que se cometieran errores al completar los campos.

Se logró que los formularios registren y muestren la información de los estudiantes y de los procesos contemplados en el proyecto, ya no en archivos individuales sino en un sistema que logra relacionar los datos.

La capacidad de relacionar los datos proporcionó una base sólida para la generación de reportes detallados y personalizados, que brindaran información valiosa para la toma de decisiones por parte de la fundación y el personal docente.

En conclusion, el sistema permite una gestión más eficiente, integrada y relacionada de la información académica de los estudiantes en la Fundación Hedwig y Robert Samuel.

Objeto 4 - Ejecutar un plan piloto con una muestra representativa de los procesos y procedimientos de la Fundación Hedwig y Robert Samuel, para evaluar la viabilidad del *software* de gestión académica y asegurar que se ajusta a las necesidades de la organización.

La implementación de un plan piloto es una etapa crucial en el desarrollo de cualquier *software*, ya que permite evaluar su desempeño y funcionamiento en condiciones reales de trabajo. En el caso específico de un *software* de gestión académica, el éxito del plan piloto dependerá en gran medida de la capacidad del *software* para cubrir las necesidades específicas de la Fundación Hedwig y Robert Samuel.

Algunos de los factores que pueden influir en la evaluación del *software* durante el plan piloto incluyen la facilidad de uso, la accesibilidad de la información, la capacidad para automatizar procesos y generar informes, así como su capacidad para integrarse con otros sistemas de información que se utilizan en la Fundación.

Si la implementación del plan piloto resulta exitosa, se espera que el *software* de gestión académica pueda proporcionar múltiples beneficios para la Fundación Hedwig y Robert Samuel. Por ejemplo, el *software* puede permitir una mayor eficiencia en la gestión de los recursos, al automatizar tareas como la matriculación de estudiantes, la gestión de horarios, y la elaboración de informes y estadísticas.

Además, el *software* puede permitir una mayor accesibilidad a la información, lo que permitiría a los administradores y personal de la Fundación acceder a los datos de manera más rápida y eficiente, y así tomar decisiones más informadas. También puede mejorar la calidad y precisión de la información, ya que el *software* puede verificar y validar automáticamente los datos ingresados.

En conclusión, la implementación exitosa de un plan piloto puede ser una señal prometedora de que el *software* de gestión académica es capaz de cumplir con los requisitos de la Fundación Hedwig y Robert Samuel, y que puede proporcionar múltiples beneficios para mejorar la gestión académica de la organización.

6.2 Recomendaciones

Luego del desarrollo del sistema de gestión de información, es posible ofrecer una serie de recomendaciones que se presentan a continuación

En cuanto a la recomendación de aplicar una metodología de desarrollo ágil, como SCRUM, es importante destacar que este enfoque se centra en entregar valor al cliente de manera constante y en incrementos pequeños, lo que permite ajustar y mejorar el producto de manera iterativa. Además, SCRUM promueve la colaboración entre el equipo de desarrollo y el cliente, lo que favorece la comprensión de las necesidades y expectativas del cliente y permite tomar decisiones informadas sobre el desarrollo del sistema.

En cuanto al uso de un lenguaje de programación sencillo y versátil como C#, es importante destacar que este lenguaje es ampliamente utilizado en el desarrollo de aplicaciones empresariales y se integra bien con tecnologías como .NET, lo que lo convierte en una opción sólida para el desarrollo del sistema de gestión de información de la Fundación Samuel.

En cuanto al uso de una aplicación web, es importante destacar que esta opción ofrece varias ventajas, como la accesibilidad desde cualquier dispositivo con conexión a internet y la menor necesidad de recursos de *hardware* en comparación con una aplicación de escritorio. Además, una aplicación web permite actualizaciones y mejoras continuas sin la necesidad de instalar *software* adicional en los dispositivos de los usuarios finales.

En cuanto a la base de datos, alojarla en la nube ofrece ventajas como una menor utilización de espacio, la flexibilidad de acceso desde cualquier lugar y la garantía de un respaldo de la información y menor riesgo de pérdida de datos en caso de fallas de *hardware*.

Es importante destacar la importancia de realizar respaldos periódicos de la base de datos y de los archivos almacenados en el sistema, para garantizar la integridad de la información y prevenir la pérdida de datos.

Finalmente, es importante tener en cuenta que la implementación de un sistema de gestión de información representa un cambio significativo para la organización y su personal. Por lo tanto, es recomendable implementar el sistema de manera gradual y en paralelo con los métodos convencionales de manipulación de datos, mientras se capacita al personal y se fomenta

su adaptación al cambio. De esta manera, se garantiza una transición suave y eficiente hacia el nuevo sistema de gestión de información.

Bibliografía

- Álvarez, C. A. (2011). *Metodología de la investigación cuantitativa y cualitativa Guía didáctica*. UNIVERSIDAD SURCOLOMBIANA.
- Bahit, E. (2012). *Scrum & extreme Programming*. Buenos Aires, Argentina: Safe Creative .
- BBVA. (24 de Noviembre de 2022). *Manifiesto 'agile', ¿qué es?* Obtenido de BBVA:
<https://www.bbva.com/es/agile-manifiesto-que-es/>
- C, A. N., & Gálvez, J. A. (1994). *Guía práctica de estudio 13: Patrones de diseño*.
- Cordero, V., & Rosa, Z. (2009). La investigación aplicada: una forma de conocer las realidades con evidencia científica. En V. Cordero, & Z. Rosa, *La investigación aplicada: una forma de conocer las realidades con evidencia científica* (págs. 155-165). San Pedro, Montes de Oca, Costa Rica: Educación.
- D'Aquino, M., & Barrón, V. (2007). *Proyecto y Metodología de la investigación*. Ituzaingó, Pcia. de Buenos Aires: Maipue.
- Diagramas del UML . (sf de sf de sf).
- Esquit, M. E. (2016). *Sistema para el control de solicitudes de compra de bienes, obras y servicios de las unidades y dependencias del instituto nacional de ciencias forenses de Guatemala (INACIF)*. Universidad de San Carlos de Guatemala Facultad de Ingeniería. Obtenido de <https://core.ac.uk/download/pdf/35294015.pdf>
- Gómez Fuentes, M. d., Cervantes Ojeda, J., & González Pérez , P. (2019). *Fundamentos de Ingeniería de Software* . Mexico : Universidad Autonoma Metropolitana .
- Hamidian Fernández, O. S. (2015). ¿ Por qué los sistemas de información son esenciales? *Anuario*, 4.

- ICBC. (11 de Noviembre de 2021). *Libros digitales: lectura sin límites* . Obtenido de [www.icbc.com.ar: https://www.icbc.com.ar/personas/como-ayudarte/estudio/Libros-digitales-lectura-sin-limites](https://www.icbc.com.ar/personas/como-ayudarte/estudio/Libros-digitales-lectura-sin-limites)
- Informatica milenium. (2022). *Páginas Web*. Obtenido de [www.informaticamilenium.com.mx: https://www.informaticamilenium.com.mx/es/temas/que-es-una-pagina-web.html](https://www.informaticamilenium.com.mx/es/temas/que-es-una-pagina-web.html)
- INVID. (2020). *¿Qué es el diseño orientado a objetos y cómo puede crear un gran software?* Obtenido de INVID: <https://invidgroup.com/es/que-es-el-diseno-orientado-a-objetos-y-como-puede-crear-un-gran-software/>
- Kvale, S. (2011). *Las entrevistas con investigación Cualitativa*. Madrid: Ediciones Morata.
- Letelier, P., & Penadés, M. C. (2006). *Métodologías ágiles para el desarrollo: eXtreme Programming (XP)*. Valencia.
- Marqués, M. (2009). *Base de datos* .
- Microsoft. (28 de 10 de 2022). *Le damos la bienvenida al IDE de Visual Studio*. Obtenido de [learn.microsoft.com: https://learn.microsoft.com/es-es/visualstudio/get-started/visual-studio-ide?view=vs-2022](https://learn.microsoft.com/es-es/visualstudio/get-started/visual-studio-ide?view=vs-2022)
- Microsoft. (2022). *Paseo por el lenguaje C#*. Obtenido de Microsoft: <https://learn.microsoft.com/es-es/dotnet/csharp/tour-of-csharp/>
- Palacio, M. (2022). Iubanis Info 4 media SL.
- Pecharromán, A. U. (2015). *Aplicación de la metodología de desarrollo ágil Scrum para desarrollo de un sistema de gestión de empresas*. Madrid.
- Pecharromán, A. U. (2015). *Aplicación de la metodología de desarrollo ágil Scrum para el desarrollo de un sistema de gestión de empresas*. Madrid.
- Pérez, J. C. (2018). *Entornos de desarrollo* . Madrid : Editorial Síntesis, S.A.

Pressman, R. S. (2010). *Ingeniería del software un enfoque práctico septima edicion*. Mexico: Mc Graw Hill.

Ramirez, I. M., Zapata, J. d., & Acosta, R. C. (2021). *Pruebas de software*. Chiclayo: Juliana del Pilar Alva Zapata.

Rex, M. B. (2010). *Apoyo a las necesidades sociales*. Managua: FUNDACION HEDWIG Y ROBERT SAMUEL.

Rumbaugh , J., Jacobson , I., & Booch, G. (2007). *El lenguaje Unificado de Modelado Manual de Referencia*. Madrid: Pearson Addison Wesley.

Sampieri, R. H., Collado, C. F., & Lucio, M. d. (2014). *Metodología de la investogación*. México D.F : Mc Graw Hill Education .

Samuel . (2022). *Nuestro enfoque formativo*. Obtenido de Fundación Hedwig y Robert Samuel: <https://www.samuel.de/es/nuestro-trabajo/formacion-tecnica>

Samuel. (2022). *samuel [Imagen]*. Obtenido de Samuel: <https://www.samuel.de/es>

Samuel, F. H. (2020). *panfleto*. San José 10803 | Costa Rica : Fundación Hedwig Y Robert Samuel.

Samuel, F. H. (29 de marzo de 2022). *Actividades-de-proyecto*. Obtenido de Fundación Samuel: <https://www.samuel.de/es/proyectos/actividades-de-proyecto/>

Samuel, F. H. (29 de marzo de 2022). *Sobre-Nosotros*. Obtenido de Fundación Samuel: <https://www.samuel.de/es/sobre-nosotros/organisation/>

Solbyte. (28 de marzo de 2021). *Solbyte Servicios informaticos* . Obtenido de 5 Fases del proceso de desarrollo de software: <https://www.solbyte.com/blog/5-etapas-del-proceso-de-desarrollo-de-software/>

Sommerville, I. (2005). *Ingeniería del software un enfoque práctico*. Madrid (España): Pearson Addison Wesley.

testsiteforme. (16 de Setiembre de 2022). *Fuentes primarias y secundarias: explicación con ejemplos*. Obtenido de [www.testsiteforme.com](https://www.testsiteforme.com/fuentes-primarias-y-secundarias/): <https://www.testsiteforme.com/fuentes-primarias-y-secundarias/>

Anexos

Anexo 1. Datos ampliados

BASE DE DATOS GENERACION 2019-2020 Datos Ampliados

Fundación Samuel Costa Rica

Número de Registro	Carrera	Número de Cédula	Apellidos	Nombre	Nombre del Padre	Nombre de la Madre	Nombre del / de la Representante Legal / Persona de Apoyo	Educación Primaria		Educación Secundaria		Número de personas que viven en el hogar	El/la estudiante/la familia recibe algún tipo de ayuda?		Total ingresos mensuales del hogar	Ingresos per Cápita
								Escuela	Año	Escuela	Año		▼	Monto de Ayuda		
DM 2019-020 15	ADM	117760117	Gutiérrez Ruíz	Wendy Daniela	Jose Paulino Gutiérrez Pavon	Yesenia Ruíz Rivera	Yesenia Ruíz Rivera			Colegio Superior de Señoritas	2018	5	No		353.000	70.600
DM 2019-020 16	ADM	117500824	Hernández Mejía	Kevin José	Fabián Vinicio Hernández Rojas	Kennyma Mejía Selva	Kennyma Mejía Selva	Escuela Santa Marta	2011	Liceo Monseñor Rubén Odio	2014	4	No		352.700	88.175
DM 2019-020 17	ADM	604460385	Largaespada Fernández	Loreana	Carol Mercedes	Largaespada Fernández	Carol Mercedes Largaespada Fernández	Unidad Pedagógica Caniel Obuber Quirós	2010	Liceo Luis Dobles Segreda	2015	5	No		462.811	92.562
DM 2019-020 18	ADM	118190909	Leiva Villalobos	Génesis	Carlos Manuel Leiva Corrales	Nydia Villalobos García	Nydia Villalobos García	Unidad Pedagógica José Rafael Araya Rojas	2013	Unidad Pedagógica José Rafael Araya Rojas	2018	5	Si	18.000	334.245	66.849
DM 2019-020 19	ADM	117680105	Lobo Serrano	Michelle	Mario Alberto Lobo Marín	Marcela Serrano Romero	Marcela Serrano Romero	Escuela Monte Verde	2013	Liceo Laboratorio Emma Gamboa	2015	4	No		320.000	80.000
DM 2019-020 20	ADM	116400144	Meneses González	Yendry Francinie	Francisco Leonel Meneses García	Elizabeth González Venegas	Elizabeth González Venegas			Liceo Napoleón Quesada Salazar	2016	2	No		90.000	45.000
DM 2019-020 21	ADM	117840022	Montero López	Brandon Javier	Wilber Montero Abarca	Sonia Andrea López Villalta	Sonia Andrea López Villalta			Liceo Napoleón Quesada Salazar	2017	4	No		92.500	23.125
DM 2019-020 22	ADM	402510613	Salas González	Elyzamar	Marcial Salas Brizuela	Marlene González Bolaños	Marlene González Bolaños	Escuela Alfredo Volio Jiménez	2013	Liceo Santa Barbara	2018	3	No		300.000	100.000
DM 2019-020 23	ADM	116210828	Solano Lizano	Ana Victoria	Álvaro Solano Angulo	Zoila Rosa Lizano Soto	Zoila Rosa Lizano Soto			Colegio Nacional Virtual Marco Tulio Salazar	2014	3	No		332.589	110.863
DM 2019-020 24	ADM	117130646	Tellez Gutiérrez	Dennis Janier	Denis René Tellez López	Angela Gutiérrez Pérez	Angela Gutiérrez Pérez	Escuela Alfredo Volio Jiménez	2010	Colegio Nocturno Guaycara	2016	4	No		347.167	86.792
DM 2019-020 25	ADM	118260627	Treminio Flores	Yorleny de los Angeles	René Alberto Treminio Jarquín	Gloria Flores Mendoza	Gloria Flores Mendoza			Unidad Pedagógica José Fidel Tristán	2018	3	No		120.000	40.000
DM 2019-020 26	ADM	118220309	Valverde Miranda	Sharon Pamela	Cristian Manrique Valverde Salas	Shirley de los Angeles Miranda	Shirley de los Angeles Miranda Gutiérrez	Escuela Ciudadela de Pavas	2013	Colegio Rincón Grande de Pavas	2018	2	Si	50.000	83.000	41.500

Anexo 3. Documento de registro de notas

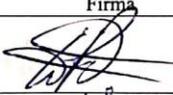
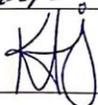
ALUMNO			90% Examen			10% Evaluación socio afectiva	Nota obtenida
			Detalle 1	Detalle 2	Detalle 3		
1	Agüero Cardenas	Oscar Eduardo	73,64			7	80,64
2	Amores Obando	Caleb Alonso	65,45			6	71,45
3	Betancourt Sánchez	Gustavo Adolfo	69,55			7	76,55
4	Campos González	Merilyn Dayana	71,59			6	77,59
5	Castillo Sánchez	Mikenver	69,55			5	74,55
6	Castro Salazar	José Pablo	69,55			5	74,55
7	Chavarria Chavarria	Keilor Osvaldo	67,50			5	72,50
8	Chaves Rojas	Carlos Josué	0				0,00
9	Cortes Rivera	Nathan Moises	68,32			5	73,32
10	Escalante Chaves	Derek Mauricio	0				0,00
11	Jiménez Monge	Maikel Jesús	72,00			7	79,00
12	Jiménez Umaña	Brandon David	65,45			6	71,45
13	Manzanares López	José Alberto					0,00
14	Mora Jara	Irvin Heli	68,32			5	73,32
15	Penant Barahona	Alejandro	65,45			5	70,45
16	Ramos Tijerino	Saúl Ezequiel	65,45			5	70,45
17	Rivera Ureña	Erick Gabriel					0,00
18	Rocha González	Kevin Ismael					0,00
19	Rodríguez Madrigal	Aaron Enrique					0,00
20	Rovira Vargas	Valeria	73,64			7	80,64
21	Saénez Baltodano	Kenneth Josue	72,41			6	78,41
22	Sanarrusia Sánchez	Kendall José	65,45			6	71,45
23	Sequeira Granja	Joshua Rolando					0,00
24	Thomas Obando	Omar Joafv	72,00			5	77,00
25	Vindas Madrigal	Kervin Elías	69,14			5	74,14
26	Flores Salas	Josiney de Jesús	68,32			5	73,32
13	Lizano Hernández	Moisés Edgardo	69,14			5	74,14

Anexo 4. Minutas de reuniones

MINUTA DE REUNIONES PROYECTO DE SISTEMA DE GESTIÓN DE INFORMACIÓN	
	
NUMERO DE MINUTA: 01	FECHA: 10/01/2023
ELABORADA POR: Hairom Ramírez S	FECHA DE ELABORACIÓN 11/01/2023
HORA DE INICIO: 3:00 pm	HORA DE FINALIZACIÓN: 3:35 pm
LUGAR: Fundación Samuel	
OBJETIVO DE LA REUNIÓN:	
Presentar el documento de especificación de requerimientos para la revisión y aprobación del director ejecutivo y la directora académica de la Fundación Samuel.	

TEMAS TRATADOS		
ASUNTO	COMENTARIOS	ACUERDOS
Los usuarios del sistema	No se tenía pensado al director académico como usuario.	Agregar al director académico como usuario y asignar un rol.
Información de los estudiantes	Se debe de agregar más información de los estudiantes a la base de datos.	Agregar más información a la base de datos referente a los estudiantes.
Requerimiento RF08	Solo se debe inhabilitar el registro del estudiante.	No se debe de diseñar este módulo con la opción de la eliminar un registro de estudiante.

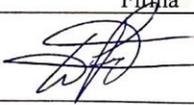
ASUNTOS PENDIENTES		
TAREA	RESPONSABLES	FECHA DE ENTREGA
Agregar a la base de datos la información solicitada y mostrar el diseño.	Hairom Ramírez S	13/01/2023

PARTICIPANTES		
Nombre	Área	Firma
Wilber Pérez	Director ejecutivo	
Karla Arguedas	Directora Académica	

MINUTA DE REUNIONES PROYECTO DE SISTEMA DE GESTION DE INFORMACIÓN	
	
NUMERO DE MINUTA: 02	FECHA: 20/02/2023
ELABORADA POR: Hairom Ramírez S	FECHA DE ELABORACIÓN 27/02/2023
HORA DE INICIO: 2:00 pm	HORA DE FINALIZACIÓN: 2:45 pm
LUGAR: Fundación Samuel	
OBJETIVO DE LA REUNIÓN:	
Presentar el avance del sistema al director ejecutivo para que dé su opinión.	

TEMAS TRATADOS		
ASUNTO	COMENTARIOS	ACUERDOS
Módulo de ingreso de los datos de los estudiantes.	Al tratar de ingresar un tipo específico de dato, el sistema no lo permite, en específico la cedula del estudiante	Arreglar el problema para que permita el ingreso de una célula o documento de un extranjero.
Dirección de los estudiantes	Se está pensando en posibilidad de recoger la información de la provincia y cantón.	Queda pendiente la confirmación para agregar estos datos.

ASUNTOS PENDIENTES		
TAREA	RESPONSABLES	FECHA DE ENTREGA
Solucionar problemas en el formulario de ingreso de estudiantes	Hairom Ramírez S	29/02/2023

PARTICIPANTES		
Nombre	Área	Firma
Wilber Pérez	Director ejecutivo	
Hairom Ramírez S	Desarrollador	

Anexo 5. Capa Datos – Clase Conexión y Asistencia

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6  using System.Data.SqlClient;
7
8  namespace SamuelAcademi.Datos
9  {
10     99+ referencias
11     public class Conexion
12     {
13         private string Base;
14         private string Servidor;
15         private string Usuario;
16         private string Clave;
17         private bool Seguridad;
18
19         private static Conexion Con = null; //instancia objeto Con que instancia a la clase conexion
20
21         1 referencia
22         private Conexion() //constructor
23         {
24             this.Base = "SamuelAcademi";
25             // this.Servidor = "DESKTOP-GE3VNIC\\SQLEXPRESS";
26             this.Servidor = "DESKTOP-N85OTU3\\SQLEXPRESS";
27             this.Usuario = "hyron";
28             this.Clave = "SQL";
29             this.Seguridad = true;
30         }
31
32         99+ referencias
33         public SqlConnection CrearConexion()
34         {
35             SqlConnection Cadena = new SqlConnection();
36             try
37             {
38                 Cadena.ConnectionString = "Server=" + this.Servidor + ";Database=" + this.Base + ";";
39                 if (this.Seguridad)
40                 {
41                     Cadena.ConnectionString = Cadena.ConnectionString + "Integrated Security = SSPI";
42                 }
43                 else
44                 {
45                     Cadena.ConnectionString = Cadena.ConnectionString + "User Id" + this.Usuario + ",Password =" + thi
46                 }
47             }
48             catch (Exception ex)
49             {
50                 Cadena = null;
51                 throw ex;
52             }
53             return Cadena;
54         }
55
56         99+ referencias
57         public static Conexion getInstancia() //metodo de tipo conexion
58         {
59             if (Con == null)
60             {
61                 Con = new Conexion();
62             }
63             return Con;
64         }
65     }
66 }

```

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6 using System.Data;
7 using System.Data.SqlClient;
8 using SamuelAcademi.Entidades;
9
10
11 namespace SamuelAcademi.Datos
12 {
13     22 referencias
14     public class Dasistencias
15     {
16         1 referencia
17         public DataTable Listar()
18         {
19             SqlDataReader Resultado;
20             DataTable Tabla = new DataTable();
21             SqlConnection SqlCon = new SqlConnection();
22             try
23             {
24                 SqlCon = Conexion.getInstancia().CrearConexion();
25                 SqlCommand Comando = new SqlCommand("asistencia_listar", SqlCon);
26                 Comando.CommandType = CommandType.StoredProcedure;
27                 SqlCon.Open();
28                 Resultado = Comando.ExecuteReader();
29                 Tabla.Load(Resultado);
30                 return Tabla;
31             }
32             catch (Exception ex)
33             {
34                 throw ex;
35             }
36             finally
37             {
38                 if (SqlCon.State == ConnectionState.Open) SqlCon.Close();
39             }
40         }
41         1 referencia
42         public DataTable asistencia_buscar_generacion_especialidad(string Valor1, string Valor2)
43         {
44             SqlDataReader Resultado;
45             DataTable Tabla = new DataTable();
46             SqlConnection SqlCon = new SqlConnection();
47             try
48             {
49                 SqlCon = Conexion.getInstancia().CrearConexion();
50                 SqlCommand Comando = new SqlCommand("asistencia_buscar_generacion_especialidad", SqlCon);
51                 Comando.CommandType = CommandType.StoredProcedure;
52                 Comando.Parameters.Add("@valor1", SqlDbType.VarChar).Value = Valor1;
53                 Comando.Parameters.Add("@valor2", SqlDbType.VarChar).Value = Valor2;
54                 SqlCon.Open();
55                 Resultado = Comando.ExecuteReader();
56                 Tabla.Load(Resultado);
57                 return Tabla;
58             }
59             catch (Exception ex)
60             {
61                 throw ex;
62             }
63             finally
64             {
65                 if (SqlCon.State == ConnectionState.Open) SqlCon.Close();
66             }
67         }
68         1 referencia
69         public DataTable ReporteAsistenciaEstudiante(string Valor1, string Valor2)
70         {
71             SqlDataReader Resultado;
72             DataTable Tabla = new DataTable();
73             SqlConnection SqlCon = new SqlConnection();
74             try
75             {
76                 SqlCon = Conexion.getInstancia().CrearConexion();
77                 SqlCommand Comando = new SqlCommand("reporte_asistencia_estudiantes", SqlCon);
78                 Comando.CommandType = CommandType.StoredProcedure;
79                 Comando.Parameters.Add("@valor1", SqlDbType.VarChar).Value = Valor1;
80                 Comando.Parameters.Add("@valor2", SqlDbType.VarChar).Value = Valor2;
81                 SqlCon.Open();
82                 Resultado = Comando.ExecuteReader();
83                 Tabla.Load(Resultado);
84                 return Tabla;
85             }
86         }
87     }
88 }

```

Anexo 6. Capa Entidades – Clase Asistencia

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace SamuelAcademi.Entidades
8 {
9     public class asistencia
10    {
11        public int id_asistencia { get; set; }
12        public int id_alumno { get; set; }
13        public DateTime fecha { get; set; }
14        public string hora_ingreso { get; set; }
15        public string hora_salida { get; set; }
16        public string reporte { get; set; }
17        public string codigo { get; set; }
18    }
19 }
20
21
```

Anexo 7. Capa Negocio – Clase Asistencia

```

1  using System;
2  using System.Data;
3  using SamuelAcademi.Datos;
4  using SamuelAcademi.Entidades;
5  namespace SamuelAcademi.Negocio
6  {
7      14 referencias
8      public class Nasistencia
9      {
10         3 referencias
11         public static DataTable Listar()
12         {
13             Dasistencias Datos = new Dasistencias();
14             return Datos.Listar();
15         }
16         2 referencias
17         public static DataTable asistencia_buscar_generacion_especialidad(string Valor1, string Valor2)
18         {
19             Dasistencias Datos = new Dasistencias();
20             return Datos.asistencia_buscar_generacion_especialidad(Valor1,Valor2);
21         }
22         2 referencias
23         public static DataTable ReporteAsistenciaEstudiante(string Valor1, string Valor2)
24         {
25             Dasistencias Datos = new Dasistencias();
26             return Datos.ReporteAsistenciaEstudiante(Valor1, Valor2);
27         }
28         public static DataTable listaEstudiantesGeneracionEspecialidad(string Valor1, string Valor2)
29         {
30             Dasistencias Datos = new Dasistencias();
31             return Datos.listaEstudiantesGeneracionEspecialidad(Valor1, Valor2);
32         }
33         0 referencias
34         public static DataTable Buscar(string Valor)
35         {
36             Dasistencias Datos = new Dasistencias();
37             return Datos.Buscar(Valor);
38         }
39         1 referencia
40         public static DataTable ConsultaFechas(DateTime FechaInicio, DateTime FechaFin)
41         {
42             Dasistencias Datos = new Dasistencias();
43             return Datos.ConsultaFechas(FechaInicio, FechaFin);
44         }
45         1 referencia
46         public static DataTable ConsultaEstudianteFechas(DateTime FechaInicio, DateTime FechaFin, int IdAlumno)
47         {
48             Dasistencias Datos = new Dasistencias();
49             return Datos.ConsultaEstudianteFechas(FechaInicio, FechaFin,IdAlumno);
50         }
51         public static DataTable Listar_estudiantes()
52         {
53             Dasistencias Datos = new Dasistencias();
54             return Datos.Listar_estudiantes();
55         }
56         1 referencia
57         public static string Eliminar(int Id)
58         {
59             Dasistencias Datos = new Dasistencias();
60             return Datos.Eliminar(Id);
61         }
62     }
63 }

```

```

56 public static string Actualizar
57 (
58     int id_asistencia,
59     int id_alumno,
60     DateTime fecha,
61     string hora_ingreso,
62     string hora_salida,
63     string reporte,
64     stringCodigoEs,
65     string codigo
66 )
67 {
68     Dasistencias Datos = new Dasistencias();
69     asistencia Obj = new asistencia();
70
71     if (CodigoEs.Equals(codigo))
72     {
73         Obj.id_asistencia = id_asistencia;
74         Obj.id_alumno = id_alumno;
75         Obj.fecha = fecha;
76         Obj.hora_ingreso = hora_ingreso;
77         Obj.hora_salida = hora_salida;
78         Obj.reporte = reporte;
79         Obj.codigo = codigo;
80
81         return Datos.Actualizar(Obj);
82         return Datos.Actualizar(Obj);
83     }
84     else
85     {
86         string existe = Datos.Existe(codigo);
87         if (existe.Equals("1"))
88         {
89             return "El registro ya existe";
90         }
91         else
92         {
93             //Obj.id_asistencia = id_asistencia;
94             Obj.id_alumno = id_alumno;
95             Obj.fecha = fecha;
96             Obj.hora_ingreso = hora_ingreso;
97             Obj.hora_salida = hora_salida;
98             Obj.reporte = reporte;
99             Obj.codigo = codigo;
100
101             return Datos.Actualizar(Obj);
102         }
103     }
104 }
105
106
107 public static string Insertar
108 (
109     //int id_asistencia,
110     int id_alumno,
111     DateTime fecha,
112     string hora_ingreso,
113     string hora_salida,
114     string reporte,
115     string codigo
116 )
117 {
118     Dasistencias Datos = new Dasistencias();
119     asistencia Obj = new asistencia();
120
121     string existe = Datos.Existe(codigo);
122
123     if (existe.Equals("1"))
124     {
125         return "Ya existe el registro";
126     }

```

```
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142

else
{
    //Obj.id_asistencia = id_asistencia;
    Obj.id_alumno = id_alumno;
    Obj.fecha = fecha;
    Obj.hora_ingreso = hora_ingreso;
    Obj.hora_salida = hora_salida;
    Obj.reporte = reporte;
    Obj.codigo = codigo;

    return Datos.Insertar(Obj);
}
}
```

Anexo 8. Prueba de gestión de usuario

	PRUEBAS PROYECTO DE SISTEMA DE GESTION DE INFORMACIÓN
	FECHA: 10/04/2023

PRUEBA DE GESTIÓN DE USUARIOS		
DESCRIPCIÓN DE LA PRUEBA	RESULTADO ESPERADO	RESULTADO OBTENIDO
Registrar un usuario con éxito	Registrar un usuario de manera exitosa.	Prueba exitosa.
Registrar un usuario con datos incompletos	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.
Actualizar los datos de un usuario con éxito	Que se actualice los datos de un usuario con éxito.	Prueba exitosa.
Actualizar un usuario con datos incompletos	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.

MEJORAS O ACTUALIZACIONES		
TAREA	RESPONSABLES	FECHA DE ENTREGA

PARTICIPANTES		
Nombre	Área	Firma
Norma Hamlett Nicaragua	Asistente de dirección Académica	
Maryory Dávila Gómez	Secretaria de Dirección Académica	

Anexo 9. Prueba de gestión de inicio de sesión

	PRUEBAS PROYECTO DE SISTEMA DE GESTION DE INFORMACIÓN
FECHA: 10/04/2023	

PRUEBA DE INICIO DE SESIÓN		
DESCRIPCIÓN DE LA PRUEBA	RESULTADO ESPERADO	RESULTADO OPTENIDO
Inicio de sesión con éxito.	Iniciar sesión de manera exitosa.	Prueba exitosa.
Inicio de sesión con error de autenticación.	El sistema no realice inicio de sesión y muestre un mensaje de error de autenticación.	Prueba exitosa.

MEJORAS O ACTUALIZACIONES		
TAREA	RESPONSABLES	FECHA DE ENTREGA

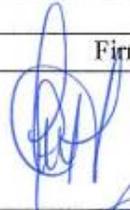
PARTICIPANTES		
Nombre	Área	Firma
Norma Hamlett Nicaragua	Asistente de Dirección Académica	
Maryory Dávila Gómez	Secretaria de Dirección Académica	

Anexo 10. Prueba de gestión de asignaturas

	PRUEBAS PROYECTO DE SISTEMA DE GESTION DE INFORMACIÓN
	FECHA: 10/04/2023

PRUEBA DE GESTIÓN DE ASIGNATURAS		
DESCRIPCIÓN DE LA PRUEBA	RESULTADO ESPERADO	RESULTADO OBTENIDO
Registrar una asignatura con éxito.	Registrar una asignatura de manera exitosa.	Prueba exitosa.
Registrar un asignatura con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.
Actualizar los datos de una asignatura con éxito.	Que se actualice los datos de un usuario con éxito.	Prueba exitosa.
Actualizar una asignatura con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa
Eliminar una asignatura	El sistema muestre mensajes que indiquen que se eliminó el registro.	Prueba exitosa.
Habilitar y deshabilitar asignatura	El sistema muestre mensajes que indiquen que se realizó la acción.	Prueba exitosa

MEJORAS O ACTULIZACIONES		
TAREA	RESPONSABLES	FECHA DE ENTREGA

PARTICIPANTES		
Nombre	Área	Firma
Norma Hamlett Nicaragua	Asistente de dirección Académica	
Maryory Dávila Gómez	Secretaria de Dirección Académica	

Anexo 11. Prueba de gestión de carreras

	PRUEBAS PROYECTO DE SISTEMA DE GESTIÓN DE INFORMACIÓN
	FECHA: 10/04/2023

PRUEBA DE GESTIÓN DE CARRERA		
DESCRIPCIÓN DE LA PRUEBA	RESULTADO ESPERADO	RESULTADO OBTENIDO
Registrar una carrera con éxito.	Registrar una carrera de manera exitosa.	Prueba exitosa.
Registrar un carrera con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.
Actualizar los datos de una carrera con éxito.	Que se actualice los datos de un usuario con éxito.	Prueba exitosa.
Actualizar una carrera con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa
Eliminar una carrera.	El sistema muestre mensajes que indiquen que se eliminó el registro.	Prueba exitosa.
Habilitar y deshabilitar una carrera.	El sistema muestre mensajes que indiquen que se realizó la acción.	Prueba exitosa

MEJORAS O ACTUALIZACIONES		
TAREA	RESPONSABLES	FECHA DE ENTREGA

PARTICIPANTES		
Nombre	Área	Firma
Norma Hamlett Nicaragua	Asistente de dirección Académica	
Maryory Dávila Gómez	Secretaria de Dirección Académica	

Anexo 12. Prueba de gestión de profesores

	PRUEBAS PROYECTO DE SISTEMA DE GESTION DE INFORMACIÓN
FECHA: 10/04/2023	

PRUEBA DE GESTIÓN DE PROFESORES		
DESCRIPCIÓN DE LA PRUEBA	RESULTADO ESPERADO	RESULTADO OPTENIDO
Registrar una profesor con éxito.	Realizar el registro de manera exitosa.	Prueba exitosa.
Registrar un profesor con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.
Actualizar los datos de un profesor con éxito.	Que se actualice los datos de un usuario con éxito.	Prueba exitosa.
Actualizar los datos de un profesor con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa
Eliminar el registro de un profesor	El sistema muestre mensajes que indiquen que se eliminó el registro.	Prueba exitosa.
Habilitar y deshabilitar un registro	El sistema muestre mensajes que indiquen que se realizó la acción.	Prueba exitosa

MEJORAS O ACTULIZACIONES		
TAREA	RESPONSABLES	FECHA DE ENTREGA

PARTICIPANTES		
Nombre	Área	Firma
Norma Hamlett Nicaragua	Asistente de dirección Académica	
Maryory Dávila Gómez	Secretaria de Dirección Académica	

Anexo 13. Prueba de gestión de generación

	PRUEBAS PROYECTO DE SISTEMA DE GESTION DE INFORMACIÓN
	FECHA: 10/04/2023

PRUEBA DE GESTIÓN DE GENERACIÓN		
DESCRIPCIÓN DE LA PRUEBA	RESULTADO ESPERADO	RESULTADO OPTENIDO
Registrar una generación con éxito.	Realizar el registro de manera exitosa.	Prueba exitosa.
Registrar una generación con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.
Actualizar los datos de una generación con éxito.	Que se actualice los datos de un usuario con éxito.	Prueba exitosa.
Actualizar una generación con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa
Eliminar una generación.	El sistema muestre mensajes que indiquen que se eliminó el registro.	Prueba exitosa.
Habilitar y deshabilitar una generación.	El sistema muestre mensajes que indiquen que se realizó la acción.	Prueba exitosa

MEJORAS O ACTULIZACIONES		
TAREA	RESPONSABLES	FECHA DE ENTREGA

PARTICIPANTES		
Nombre	Área	Firma
Norma Hamlett Nicaragua	Asistente de Dirección Académica	
Maryory Dávila Gómez	Secretaria de Dirección Académica	

Anexo 14. Prueba de gestión de estudiantes

	PRUEBAS PROYECTO DE SISTEMA DE GESTION DE INFORMACIÓN
FECHA: 10/04/2023	

PRUEBA DE GESTIÓN DE ESTUDIANTES		
DESCRIPCIÓN DE LA PRUEBA	RESULTADO ESPERADO	RESULTADO OBTENIDO
Registrar un estudiante con éxito.	Realizar el registro de manera exitosa.	Prueba exitosa.
Registrar un estudiante con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.
Actualizar los datos de un estudiante con éxito.	Que se actualice los datos de un usuario con éxito.	Prueba exitosa.
Actualizar un estudiante con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa
Eliminar un estudiante.	El sistema muestre mensajes que indiquen que se eliminó el registro.	Prueba exitosa.
Habilitar y deshabilitar un estudiante.	El sistema muestre mensajes que indiquen que se realizó la acción.	Prueba exitosa

MEJORAS O ACTULIZACIONES		
TAREA	RESPONSABLES	FECHA DE ENTREGA
Consulta como debe registrarse el código del estudiante	Hairom Ramírez	

PARTICIPANTES		
Nombre	Área	Firma
Norma Hamlett Nicaragua	Asistente de dirección Académica	
Maryory Dávila Gómez	Secretaria de Dirección Académica	

Anexo 15. Prueba de gestión de asistencia

	PRUEBAS PROYECTO DE SISTEMA DE GESTION DE INFORMACIÓN
FECHA: 10/04/2023	

PRUEBA DE GESTIÓN DE ASISTENCIA		
DESCRIPCIÓN DE LA PRUEBA	RESULTADO ESPERADO	RESULTADO OBTENIDO
Registrar la asistencia de un estudiante con éxito.	Realizar el registro de manera exitosa.	Prueba exitosa.
Registrar la asistencia de un estudiante con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.
Actualizar la asistencia de un estudiante con éxito.	Que se actualice los datos de un usuario con éxito.	Prueba exitosa.
Actualizar la asistencia de un estudiante con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.

MEJORAS O ACTUALIZACIONES		
TAREA	RESPONSABLES	FECHA DE ENTREGA

PARTICIPANTES		
Nombre	Área	Firma
Norma Hamlett Nicaragua	Asistente de dirección Académica	
Maryory Dávila Gómez	Secretaria de Dirección Académica	

Anexo 16. Prueba de gestión de Calificaciones

	PRUEBAS PROYECTO DE SISTEMA DE GESTION DE INFORMACIÓN
	FECHA: 10/04/2023

PRUEBA DE GESTIÓN DE CALIFICACIONES		
DESCRIPCIÓN DE LA PRUEBA	RESULTADO ESPERADO	RESULTADO OBTENIDO
Registrar las calificaciones de un estudiante con éxito.	Realizar el registro de manera exitosa.	Prueba exitosa.
Registrar las calificaciones de un estudiante con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.
Actualizar las calificaciones de un estudiante con éxito.	Que se actualice los datos de un usuario con éxito.	Prueba exitosa.
Actualizar las calificaciones de un estudiante con datos incompletos.	El sistema muestre mensajes que indiquen que ingrese información en los campos requeridos que están incompletos.	Prueba exitosa.

MEJORAS O ACTUALIZACIONES		
TAREA	RESPONSABLES	FECHA DE ENTREGA

PARTICIPANTES		
Nombre	Área	Firma
Norma Hamlett Nicaragua	Asistente de dirección Académica	
Maryory Dávila Gómez	Secretaria de Dirección Académica	

Anexo 17. Pruebas de gestión de reportes

	PRUEBAS PROYECTO DE SISTEMA DE GESTION DE INFORMACIÓN
FECHA: 10/04/2023	

PRUEBA DE GESTIÓN DE REPORTES DE REPORTES		
DESCRIPCIÓN DE LA PRUEBA	RESULTADO ESPERADO	RESULTADO OPTENIDO
Generar con éxito los reportes de estudiantes.	Se genera el reporte de manera exitosa.	Prueba exitosa.
Se genera un reporte según el género de los estudiantes.	Se genera el reporte de manera exitosa.	Prueba exitosa.
Se genera un reporte según el grado académico de los estudiantes.	Se genera el reporte de manera exitosa.	Prueba exitosa.

MEJORAS O ACTULIZACIONES		
TAREA	RESPONSABLES	FECHA DE ENTREGA

PARTICIPANTES		
Nombre	Área	Firma
Norma Hamlett Nicaragua	Asistente de Dirección Académica	
Maryory Dávila Gómez	Secretaria de Dirección Académica	