



**UNIVERSIDAD TÉCNICA DE AMBATO**

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E  
INDUSTRIAL**

**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E  
INFORMÁTICOS**

**Tema:**

---

**SISTEMA WEB USANDO EL FRAMEWORK LARAVEL PARA LA  
GESTIÓN DE CALIFICACIONES Y ASISTENCIA DE LA UNIDAD  
EDUCATIVA HUACHI GRANDE.**

---

Trabajo de Titulación Modalidad: Proyecto de Investigación, presentado previo a la obtención del título de Ingeniero en Sistemas Computacionales e Informáticos.

**ÁREA:** Software

**LÍNEA DE INVESTIGACIÓN:** Desarrollo de software

**AUTOR:** Carlos Luis Morales Gutama

**TUTOR:** Ing. Félix Oscar Fernández Peña, PhD.

**Ambato - Ecuador**

**julio – 2023**

## **APROBACIÓN DEL TUTOR**

En calidad de tutor del Trabajo de Titulación con el tema: SISTEMA WEB USANDO EL FRAMEWORK LARAVEL PARA LA GESTIÓN DE CALIFICACIONES Y ASISTENCIA DE LA UNIDAD EDUCATIVA HUACHI GRANDE, desarrollado bajo la modalidad Proyecto de Investigación por el señor Carlos Luis Morales Gutama, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 15 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y el numeral 7.4 del respectivo instructivo.

Ambato, julio 2023.

-----  
Ing. Félix Oscar Fernández Peña, PhD.

TUTOR

## AUTORÍA

El presente Proyecto de Investigación titulado: SISTEMA WEB USANDO EL FRAMEWORK LARAVEL PARA LA GESTIÓN DE CALIFICACIONES Y ASISTENCIA DE LA UNIDAD EDUCATIVA HUACHI GRANDE es absolutamente original, auténtico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, julio 2023.



---

Carlos Luis Morales Gutama

C.C 1804569778

AUTOR

## **DERECHOS DE AUTOR**

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, julio 2023.



---

Carlos Luis Morales Gutama

C.C 1804569778

AUTOR

## **APROBACIÓN DEL TRIBUNAL DE GRADO**

En calidad de par calificador del Informe Final del Trabajo de Titulación presentado por el señor Carlos Luis Morales Gutama, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado SISTEMA WEB USANDO EL FRAMEWORK LARAVEL PARA LA GESTIÓN DE CALIFICACIONES Y ASISTENCIA DE LA UNIDAD EDUCATIVA HUACHI GRANDE, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 17 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, julio 2023.

-----

Ing. Pilar Urrutia, Mg.

PRESIDENTE DEL TRIBUNAL

-----

Ing. Hernán Naranjo

PROFESOR CALIFICADOR

-----

Ing. Marco Guachimboza

PROFESOR CALIFICADOR

## **DEDICATORIA**

Con profunda gratitud y amor, dedico este logro a Dios, quien ha sido mi guía constante a lo largo de este camino académico.

A mi padre, pilar inquebrantable de mi vida, le dedico mi más sincero agradecimiento. Su incondicional apoyo, amor y sacrificio durante el tiempo que compartimos han sido la fuerza que me ha impulsado a alcanzar este logro. Gracias por creer en mí, por animarme en los momentos difíciles, su presencia siempre estará presente en mí.

A mi madre, fuente de amor y sabiduría, le dedico mi eterno agradecimiento. Tu amor incondicional, tus palabras de aliento y tu apoyo constante han sido un faro en mi camino. Gracias por tu apoyo incansable y por enseñarme el valor de la perseverancia y la dedicación.

También, quiero extender mi agradecimiento a Manuel Moreta, quien ha sido como un segundo padre para mí. Tu apoyo constante, tu sabiduría y tus consejos han sido invaluable en mi camino hacia esta meta.

**Carlos Luis Morales.**

## **AGRADECIMIENTO**

A Dios, quien ha sido mi guía constante y fuente de fortaleza a lo largo de todo este proceso académico.

Al ingeniero Félix Fernández, mi tutor de tesis, por su invaluable orientación, conocimientos y dedicación. Su guía experta y apoyo constante han sido esenciales para el éxito de este trabajo.

Al ingeniero Franklin Mayorga, quien fue un gran guía durante los últimos semestres de la universidad. Su apoyo y enseñanzas han dejado una huella significativa en mi desarrollo académico.

A mis padres, Antonio Morales y Sandra Gutama, les dedico mi más sincero agradecimiento. Su amor incondicional, su apoyo constante y su ejemplo de perseverancia han sido el cimiento sobre el cual he construido mis logros. Gracias por creer en mí y por estar a mi lado en cada paso del camino.

A mis queridos hermanos, cuyo apoyo y aliento han sido invaluable, les agradezco de corazón. Gracias por ser mi fuente de inspiración y por compartir esta alegría conmigo.

También quiero expresar mi agradecimiento a mis amigos, Héctor G., William S., David S. y Freddy S.. Su amistad, compañerismo y apoyo han sido una luz en mi vida durante este período. Gracias por estar a mi lado, por escucharme, alentarme y celebrar cada pequeño avance conmigo.

**¡Gracias a todos!"**

## ÍNDICE DE CONTENIDOS

APROBACIÓN DEL TUTOR.....	ii
AUTORÍA .....	iii
DERECHOS DE AUTOR .....	iv
APROBACIÓN DEL TRIBUNAL DE GRADO .....	v
DEDICATORIA .....	vi
AGRADECIMIENTO .....	vii
RESUMEN EJECUTIVO .....	xix
ABSTRACT.....	xx
CAPITULO I MARCO TEORICO.....	1
1.1 Tema de investigación.....	1
1.2 Antecedentes Investigativos .....	1
1.2.1 Contextualización del Problema .....	3
1.3 Fundamentación Teórica .....	4
1.3.1 Tecnologías de la Información y la Comunicación.....	5
1.3.2 Tecnologías de Desarrollo Web .....	5
1.3.3 Educación.....	6
1.3.4 Gestión escolar .....	7
1.3.5 Gestión de calificaciones y la asistencia .....	7
1.3.6 Sistema Educativo por Quimestres .....	8
1.3.7 Framework .....	9
1.3.8 Tecnologías del Back-End. ....	10
1.3.9 Análisis de framework de desarrollo back-end.....	11
1.4 Sistemas Gestores de Bases de Datos Relacionales. ....	14
1.5 Sistemas Gestores de Bases de Datos No Relacionales .....	17
1.6 Comparación de bases de datos: Análisis comparativo de diferentes bases de datos. ....	18



1.7	Clasificación de metodologías de desarrollo.....	19
1.8	Análisis comparativo de metodologías de desarrollo.....	21
1.8.1	Análisis comparativo de metodologías de desarrollo ágiles.....	26
1.8.2	Livewire.....	27
1.8.3	Bootstrap.....	28
1.8.4	Laragon.....	28
1.8.5	Visual Studio Code.....	29
1.8.6	GitHub.....	29
1.9	Objetivos.....	30
1.9.1	Objetivo General.....	30
1.9.2	Objetivos Específicos.....	30
CAPÍTULO II METODOLOGÍA.....		30
2.1	Materiales.....	30
2.1.1	Institucionales.....	31
2.1.2	Humano.....	31
2.1.3	Otros.....	31
2.1.4	Económicos.....	32
2.2	Métodos.....	34
2.2.1	Modalidad de Investigación.....	34
2.2.2	Población Y Muestra.....	34
2.2.3	Recolección De Información.....	35
2.2.4	Procesamiento y análisis de datos:.....	36
CAPITULO III. RESULTADOS Y DISCUSION.....		36
3.1	Análisis y discusión de resultados de la entrevista.....	36
3.2	Desarrollo de la propuesta.....	42
3.3	Aplicación de la metodología.....	43
3.3.1	Arquitectura del sistema web.....	43

3.3.2	Descripción de los procesos.....	43
3.3.3	Fase 1: Exploración.....	45
3.3.4	Fase 2: Valoración de las historias de usuario .....	53
3.3.5	Fase 3: Iteraciones.....	77
3.3.6	Fase 4: Producción .....	106
3.3.7	Fase 5: Mantenimiento.....	121
CAPITULO IV. CONCLUSIONES Y RECOMENDACIONES.....		121
4.1	Conclusiones .....	121
4.2	Recomendaciones.....	122
Bibliografía .....		123
Anexos .....		125

## Índice tablas

Tabla 1. Sistema Educativo por Quimestres. ....	9
Tabla 2. Cuadro comparativo de framework de desarrollo back-end. ....	12
Tabla 3. Cuadro comparativo entre bases de datos. ....	19
Tabla 4. Análisis comparativo de metodologías de desarrollo tradicionales y ágiles. .....	22
Tabla 5. Cuadro comparativo de metodologías ágiles. ....	27
Tabla 6. Población de estudio. ....	35
Tabla 7. Resultado de la entrevista. ....	42
Tabla 8. Historia de usuario – Formulario de ingreso al sistema. ....	45
Tabla 9. Historia de usuario – Gestionar perfiles de usuarios. ....	46
Tabla 10. Historia de usuario – Gestionar áreas. ....	46
Tabla 11. Historia de usuario – Gestionar asignatura. ....	47
Tabla 12. Historia de usuario – Gestionar años lectivos. ....	47
Tabla 13. Historia de usuario – Diseño de modulo interfaz cursos. ....	47
Tabla 14. Historia de usuario – Gestionar matrículas de estudiantes. ....	48
Tabla 15. Historia de usuario – Gestionar asignación de asignaturas a profesores. ..	48
Tabla 16. Historia de usuario – Registrar calificaciones. ....	49
Tabla 17. Historia de usuario – Registrar asistencia. ....	49
Tabla 18. Historia de usuario - Ver calificaciones actuales de estudiantes. ....	50
Tabla 19. Historia de usuario - Descargar reporte de calificaciones. ....	50
Tabla 20. Historia de usuario - Descargas reporte de asistencia. ....	50
Tabla 21. Historia de usuario - Consultar calificaciones de estudiante. ....	51
Tabla 22. Historia de usuario - Auditoria por cabio o borrado de los registros de calificaciones. ....	51
Tabla 23. Historia de usuario - Notificaciones de estudiantes con asistencia insuficiente. ....	52
Tabla 24. Historia de usuario - General certificado de promoción. ....	52
Tabla 25. Historia de usuario - Generar boleta de calificaciones. ....	53
Tabla 26. Historia de usuario - Generar reporte de calificaciones concentrado. ....	53
Tabla 27. Estimación de historias de usuario. ....	54
Tabla 28. Definición de roles. ....	56
Tabla 29. Tarea – Diseño del formulario para registro e ingreso al sistema. ....	56

Tabla 30. Tarea – Funcionalidad para el formulario para registro e ingreso al sistema. .....	57
Tabla 31. Tarea – Diseño de interfaz del sistema. ....	57
Tabla 32. Tarea – Diseño del formulario roles y permisos.....	58
Tabla 33. Tarea – Funcionalidad para el formulario roles y permisos.....	58
Tabla 34. Tarea – Diseño del formulario asistencia.....	59
Tabla 35. Tarea – Listar áreas. ....	59
Tabla 36. Tarea – Funcionalidad para el formulario áreas.....	59
Tabla 37. Tarea – Diseño del formulario asistencia.....	60
Tabla 38. Tarea – Listar asignaturas. ....	60
Tabla 39. Tarea – Funcionalidad para el formulario asignaturas.....	61
Tabla 40. Tarea – Diseño del formulario años lectivos. ....	61
Tabla 41. Tarea – Listar años lectivos. ....	62
Tabla 42. Tarea – Funcionalidad para el formulario años lectivos. ....	62
Tabla 43. Tarea – Diseño del formulario cursos.....	62
Tabla 44. Tarea – Listar cursos.....	63
Tabla 45. Tarea – Funcionalidad para el formulario cursos.....	63
Tabla 46. Tarea – Diseño del formulario matrículas de estudiantes.....	64
Tabla 47. Tarea – Listar matrículas de estudiantes.....	64
Tabla 48. Tarea – Funcionalidad para el formulario matrículas de estudiantes.....	64
Tabla 49. Tarea – Funcionalidad para el formulario importar Excel.....	65
Tabla 50. Tarea – Diseño de formulario asignación de asignaturas a profesores.....	65
Tabla 51. Tarea – Listar asignaturas a profesores.....	66
Tabla 52. Tarea – Funcionalidad para el formulario asignación de asignaturas a profesores.....	66
Tabla 53. Tarea – Diseño del formulario talleres.....	67
Tabla 54. Tarea – Listar talleres.....	67
Tabla 55. Tarea – Funcionalidad para el formulario talleres. ....	67
Tabla 56. Tarea – Diseño del formulario asistencia.....	68
Tabla 57. Tarea – Listar asistencia.....	68
Tabla 58. Tarea – Funcionalidad para el formulario asistencia. ....	69
Tabla 59. Tarea – Diseño de formulario ver calificaciones. ....	69
Tabla 60. Tarea – Funcionalidad para el formulario ver calificaciones.....	70

Tabla 61. Tarea – Diseño informe calificaciones por asignatura.....	70
Tabla 62. Tarea – Funcionalidad para el formulario ver calificaciones.....	71
Tabla 63. Tarea – Diseño informe reportes de asistencia. ....	71
Tabla 64. Tarea – Funcionalidad para el formulario ver calificaciones.....	71
Tabla 65. Tarea – Diseño de formulario consultar calificaciones de estudiantes. ....	72
Tabla 66. Tarea – Funcionalidad para el formulario ver calificaciones.....	72
Tabla 67. Tarea – Implementar auditoría de cambios en registros de calificaciones en los insumos.....	73
Tabla 68. Tarea – Diseño de interfaz de auditoría de cambios. ....	73
Tabla 69. Tarea – Implementar notificaciones de asistencia insuficiente.....	74
Tabla 70. Tarea – Diseño de informe de notificaciones de asistencia insuficiente....	74
Tabla 71. Tarea – Diseño de informe de notificaciones de asistencia insuficiente....	75
Tabla 72. Tarea – Funcionalidad para el formulario generación de certificados de promoción. ....	75
Tabla 73. Tarea – Diseño del certificado de promoción. ....	75
Tabla 74. Tarea – Diseño de formulario generación boleta de calificaciones. ....	76
Tabla 75. Tarea – Funcionalidad para el formulario generación boleta de calificaciones. ....	76
Tabla 76. Tarea – Diseño del certificado de promoción. ....	77
Tabla 77. Historias de usuario iteración 1.....	78
Tabla 78. Tarjeta CRC ingreso a la aplicación. ....	78
Tabla 79. Tarjeta CRC gestionar roles.....	78
Tabla 80. Tarjeta CRC gestionar áreas. ....	79
Tabla 81. Tarjeta CRC gestionar asignaturas.....	79
Tabla 82. Tarjeta CRC gestionar años lectivos.....	79
Tabla 83. Tarjeta CRC gestionar cursos. ....	80
Tabla 84 Tarjeta CRC matriculas estudiantes.....	80
Tabla 85 Tarjetas CRC asignaturas de profesores. ....	80
Tabla 86 Tarjeta CRC registro de calificaciones. ....	81
Tabla 87. Tarjeta CRC registro de asistencia.....	81
Tabla 88 Tarjeta CRC consultar calificaciones.....	81
Tabla 89 Tarjeta CRC generar reportes. ....	82
Tabla 90 Tarjeta CRC notificaciones.....	82

Tabla 91. Historias de usuario iteración 2.....	90
Tabla 92. Historias de usuario iteración 3.....	95
Tabla 93. Historias de usuario iteración 4.....	102
Tabla 94. Prueba de aceptación – Registro usuario. ....	107
Tabla 95. Prueba de aceptación – Inicio de sesión.....	107
Tabla 96. Prueba de aceptación – Lista de usuarios.....	108
Tabla 97. Prueba de aceptación – Crear rol. ....	108
Tabla 98. Prueba de aceptación – Gestión de áreas. ....	109
Tabla 99. Prueba de aceptación – Gestión de asignaturas. ....	110
Tabla 100. Prueba de aceptación – Gestión de años lectivos.....	111
Tabla 101. Prueba de aceptación – Gestión de cursos. ....	111
Tabla 102. Prueba de aceptación – Gestión matrículas de estudiantes. ....	112
Tabla 103. Prueba de aceptación – Importar matrículas de estudiantes. ....	113
Tabla 104. Prueba de aceptación – Gestión de asignación de asignaturas a profesores. .....	114
Tabla 105. Prueba de aceptación – Gestión de calificaciones. ....	115
Tabla 106. Prueba de aceptación – Gestión de asistencia.....	115
Tabla 107. Prueba de aceptación – Mostrar calificaciones actuales de estudiantes. ....	116
Tabla 108. Prueba de aceptación – Descargar reporte de calificaciones por asignatura. .....	116
Tabla 109. Prueba de aceptación – Descargar reporte de asistencia por asignatura. ....	117
Tabla 110. Prueba de aceptación – Consultar calificaciones de estudiante. ....	118
Tabla 111. Prueba de aceptación – Auditoría por cambio o borrado de los registros de calificaciones.....	118
Tabla 112. Prueba de aceptación – Notificaciones de estudiantes con asistencia insuficiente. ....	119
Tabla 113. Prueba de aceptación – Descargar certificado de promoción. ....	119
Tabla 114. Prueba de aceptación – Generar boleta de calificaciones. ....	120
Tabla 115. Prueba de aceptación – Generar boleta de calificaciones. ....	121

## Índice Figuras

Figura 1. Esquema de la metodología Scrum.....	23
Figura 2. Modelo de metodología Programación Extrema. ....	24
Figura 3. Arquitectura de sistema web con framework Laravel .....	42
Figura 4. Proceso de ingreso de calificaciones. ....	44
Figura 5. Proceso de control de asistencia de estudiantes.....	45
Figura 6. Diseño de base de datos.....	84
Figura 7. Creación del proyecto en terminal Laragon.....	85
Figura 8. Página de inicio del sistema para la Unidad Educativa. ....	85
Figura 9. Formulario de ingresos al sistema. ....	86
Figura 10. Formulario de registro al sistema. ....	86
Figura 11. Listar roles del sistema. ....	87
Figura 12. Formulario administrar rol y otorgar permisos.....	87
Figura 13. Lista de usuarios. ....	87
Figura 14. Formulario para asignar roles y permisos.....	88
Figura 15. Lista de áreas. ....	88
Figura 16. Formulario administrar áreas.....	89
Figura 17. Lista de asignaturas. ....	89
Figura 18. Formulario asignatura.....	90
Figura 19. Lista de años lectivos.....	91
Figura 20. Formulario para administrar los años lectivos.....	91
Figura 21. Lista de cursos. ....	92
Figura 22. Formulario para administrar cursos.....	92
Figura 23. Lista estudiantes matriculados .....	93
Figura 24. Formulario para administrar a los estudiantes matriculados. ....	93
Figura 25. Lista de asignación de asignaturas a docentes.....	94
Figura 26. Formulario para administrar la asignación de asignaturas a docentes.....	94
Figura 27. Tarjetas de asignaturas de docente .....	95
Figura 28. Tarjetas de insumos y asistencia.....	96
Figura 29. Formulario administrar insumo. ....	96
Figura 30. Formulario para administrar las calificaciones por insumo.....	97
Figura 31. Lista de insumo creado. ....	97
Figura 32. Formulario para crear horarios de sesiones. ....	98

Figura 33. Lista de sesiones para tomar asistencia. ....	98
Figura 34. Formulario para registrar asistencia.....	99
Figura 35. Vista lista de asignaturas de estudiante. ....	99
Figura 36. Vista ver calificaciones estudiantes.....	100
Figura 37. Reporte calificaciones versión extendida. ....	100
Figura 38. Reporte calificaciones versión resumida. ....	101
Figura 39. Reporte de asistencia estudiantes en documentos en formato PDF.....	101
Figura 40. Vista ver calificaciones estudiantes.....	102
Figura 41. Vista tabla auditoria de calificaciones.....	103
Figura 42. Vista notificaciones leídas y no leídas.....	103
Figura 43. Vista de notificación de estudiantes con inasistencia.....	104
Figura 44. Certificado de promoción.....	104
Figura 45. Boleta de calificaciones.....	105
Figura 46. Reporte concentrado de calificaciones por paralelo.....	106
Figura 47. Diseño de formulario de ingreso al sistema.....	126
Figura 48. Diseño de formulario de registro del sistema.....	126
Figura 49. Diseño página de inicio.....	127
Figura 50. Diseño de vista lista de usuarios.....	127
Figura 51. Componente Livewire listar y buscar usuarios.....	128
Figura 52. Función para listar y buscar usuarios.....	128
Figura 53. Función para editar roles de usuarios.....	128
Figura 54. Función para administrar roles y permisos de usuarios.....	129
Figura 55. Diseño de vista lista de áreas.....	129
Figura 56. Componente Livewire listar y buscar áreas.....	130
Figura 57. Función para listar, buscar y eliminar áreas.....	130
Figura 58. Formulario para crear y actualizar área.....	131
Figura 59. Función crear área.....	131
Figura 60. Función actualizar área.....	131
Figura 61 Diseño de vista lista de asignaturas.....	132
Figura 62. Componente Livewire listar y buscar asignaturas.....	132
Figura 63. Función para listar, buscar y eliminar asignatura.....	133
Figura 64. Formulario para crear y actualizar asignaturas.....	133
Figura 65. Función para crear asignatura.....	133



Figura 66. Función actualizar asignatura. ....	134
Figura 67. Diseño de vista lista de años lectivos. ....	134
Figura 68. Componente Livewire listar y buscar años lectivos. ....	135
Figura 69. Función para listar, buscar y eliminar años lectivos. ....	135
Figura 70. Formulario para crear y actualizar años lectivos. ....	136
Figura 71. Función crear año lectivo. ....	136
Figura 72. Función actualizar año lectivo. ....	136
Figura 73. Diseño de vista lista de cursos. ....	137
Figura 74. Componente Livewire listar y buscar cursos. ....	137
Figura 75. Función para listar, buscar y eliminar cursos. ....	138
Figura 76. Formulario para crear y actualizar cursos. ....	138
Figura 77. Función crear curso. ....	138
Figura 78. Función actualizar curso. ....	139
Figura 79. Diseño de vista lista de matrículas. ....	139
Figura 80. Componente Livewire listar y buscar matriculas. ....	140
Figura 81. Función para listar, buscar y eliminar matriculas. ....	140
Figura 82. Formulario para crear y actualizar matricula. ....	141
Figura 83. Función crear matricula. ....	141
Figura 84. Función importar matriculas desde Excel. ....	141
Figura 85. Función actualizar matricula. ....	142
Figura 86. Importar matriculas Excel. ....	142
Figura 87. Diseño de vista lista de asignación de asignaturas a profesor. ....	143
Figura 88. Componente Livewire listar y buscar asignación de signaturas a profesores. .....	143
Figura 89. Función para listar, buscar y eliminar asignación de asignaturas. ....	144
Figura 90. Formulario para crear y actualizar asignación de asignaturas. ....	144
Figura 91. Función crear asignación de asignatura. ....	144
Figura 92. Función actualizar asignación de asignatura. ....	145
Figura 93. Diseño de vista lista de insumos. ....	145
Figura 94. Componente Livewire listar y buscar insumo. ....	146
Figura 95. Formulario para crear y actualizar insumo. ....	146
Figura 96. Formulario registrar calificaciones. ....	147
Figura 97. Formulario para actualizar calificación. ....	147

Figura 98. Función crear insumo. ....	147
Figura 99. Función actualizar insumo. ....	148
Figura 100. Función crear calificación de estudiantes. ....	148
Figura 101. Función actualizar calificación estudiante. ....	148
Figura 102. Diseño de vista listas horario de asistencias. ....	149
Figura 103. Componente Livewire listar y buscar horarios de asistencia. ....	149
Figura 104. Función para listar, buscar y eliminar horarios de asistencia. ....	150
Figura 105. Formulario para crear y actualizar horario de asistencia. ....	150
Figura 106. Formulario actualizar asistencia. ....	151
Figura 107. Función para eliminar múltiples registros de asistencia. ....	151
Figura 108. Función crear horarios de asistencia. ....	152
Figura 109. Función crear asistencia. ....	152
Figura 110. Función actualizar asistencia. ....	152
Figura 111. Diseño de vista de calificaciones de insumo. ....	153
Figura 112. Función ver asignaturas. ....	153
Figura 113. Función ver calificaciones de insumos. ....	154
Figura 114. Función calificaciones por asignatura Excel. ....	154
Figura 115. Función ver calificaciones de insumos. ....	155
Figura 116. Función dibujar imagen en Excel. ....	155
Figura 117. Función estilos reporte asistencia. ....	156
Figura 118. Función generar reporte asistencia. ....	156
Figura 119. Función ver calificaciones de insumos. ....	157
Figura 120. Configuración de los eventos auditar. ....	157
Figura 121. Función notificación con estudiantes con inasistencia superior 10%... ..	158
Figura 122. Función generar boleta de calificaciones. ....	159

## RESUMEN EJECUTIVO

La presente tesis tiene como objetivo principal el diseño e implementación de un sistema web para automatizar los procesos de gestión de calificaciones y asistencia de los estudiantes en la unidad educativa Huachi Grande, utilizando las tecnologías de desarrollo web Laravel, Livewire y Bootstrap 5. El sistema web desarrollado permite a los docentes registrar y gestionar las calificaciones de los estudiantes, así como también llevar un control de su asistencia a clases. Además, los padres de familia pueden acceder al sistema para consultar las calificaciones y asistencia de sus hijos en tiempo real. El sistema web cuenta con una interfaz amigable y de fácil uso, lo que permite a los usuarios acceder a la información de manera rápida y eficiente. Además, está diseñado para ser escalable y adaptable a futuras necesidades y requerimientos. Los resultados obtenidos demuestran que el sistema web desarrollado cumple con los objetivos propuestos, facilitando la gestión de calificaciones y asistencia de los estudiantes en la unidad educativa Huachi Grande. Se espera que este trabajo pueda ser de utilidad para otras instituciones educativas que deseen implementar un sistema similar.

**Palabras clave:** Sistema web, Laravel, Livewire, Gestión Calificaciones, Gestión Asistencia.

## **ABSTRACT**

The main objective of this thesis is the design and implementation of a web system to automate the process of managing grades and attendance of students in the educational unit Huachi Grande, using web development technologies Laravel, Livewire and Bootstrap 5. The developed web system allows teachers to register and manage students' grades, as well as to keep track of their attendance to classes. In addition, parents can access the system to check their children's grades and attendance in real time. The web-based system has a user-friendly and easy-to-use interface, allowing users to access information quickly and efficiently. In addition, it is designed to be scalable and adaptable to future needs and requirements. The results obtained show that the developed web system meets the proposed objectives, facilitating the management of grades and attendance of students in the educational unit Huachi Grande. It is hoped that this work can be useful for other educational institutions that wish to implement a similar system.

**Keywords:** Web system, Laravel, Livewire, Grade Management, Attendance Management.

## **CAPITULO I MARCO TEORICO**

### **1.1 Tema de investigación**

SISTEMA WEB USANDO EL FRAMEWORK LARAVEL PARA LA GESTIÓN DE CALIFICACIONES Y ASISTENCIA DE LA UNIDAD EDUCATIVA HUACHI GRANDE.

### **1.2 Antecedentes Investigativos**

Después de revisar la información disponible en repositorios de diferentes universidades, entre ellos la biblioteca de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato, se han identificado varios trabajos que fueron de gran utilidad como apoyo para la investigación realizada.

El trabajo investigativo desarrollado por Karina Del Rocío Cedeño Vargas con el tema: “Diseño e implementación de un sistema web de control de matrícula y calificaciones para el colegio Rashid Torbay Sismarashid en el cantón playas, provincia del Guayas “Concluyó con el desarrollo de un sistema web que automatiza los procesos de matrícula y calificaciones en una institución educativa. Esto ha agilizado el ingreso de datos y reducido el tiempo necesario para completar estas tareas. Al optimizar el uso de recursos y minimizar la impresión de informes en papel, se reducen los gastos administrativos. El tiempo de proceso de matrícula, ingreso de calificaciones y generación de informes se ha reducido significativamente, mejorando la obtención de información precisa. Además, la base de datos flexible MySQL permite un amplio almacenamiento de información y adaptabilidad a futuros requisitos y automatizaciones en otras áreas [1].

El trabajo investigativo desarrollado por Anrrango Benavides Darío Xavierdel con el tema: “Sistema web para la gestión de matrículas y calificaciones de la unidad educativa fiscomisional Fray Bartolomé de las casas Salasaca” Se implementó un sistema web en la Unidad Educativa para mejorar los procesos de registro de información personal de estudiantes, personal administrativo y docentes, con mayor claridad y eficiencia, brindando un mejor servicio a la comunidad educativa. Se mejoraron tareas como la matriculación, ingreso de notas y generación de informes, adaptándose a las necesidades de cada usuario. Se realizó un estudio comparativo de diferentes lenguajes de programación, determinando que PHP, junto con JavaScript y JQuery, fueron los más adecuados para el desarrollo del proyecto [2].

El trabajo investigativo desarrollado por Álvarez Cisneros, Ana Verónica con el tema: “Sistema web para la gestión de la pensión diferenciada en la unidad educativa González Suárez” el objetivo principal del proyecto fue desarrollar una solución informática para gestionar de manera más eficiente las solicitudes de becas de apoyo económico en la Unidad Educativa "González Suárez", mejorando así el bienestar estudiantil. Para seleccionar las tecnologías adecuadas en el desarrollo del sistema web, se llevó a cabo un análisis comparativo de metodologías de desarrollo de software como Scrum, XP y el Modelo en V. Las tecnologías seleccionadas garantizarán un desarrollo eficaz del sistema [3].

El trabajo investigativo desarrollado por Rodolfo Antonio Villafuerte López Andrés, Felipe Rodríguez Enríquez con el tema: “Implementación de aplicación web para la gestión de matrículas y calificaciones, en la escuela de educación básica Enrique Fierro” Se implementó un sistema web en la Escuela de Educación Básica "Enrique Fierro" para administrar las matrículas y calificaciones. El sistema automatiza el proceso de matriculación y registro de calificaciones, optimizando la atención a los representantes y facilitando la búsqueda de información de los estudiantes. La automatización de las calificaciones mejora significativamente el tiempo empleado por los docentes para registrar las calificaciones de las evaluaciones. La generación de

informes brinda un apoyo importante al personal administrativo y docente, automatizando procesos y cálculos que antes se realizaban manualmente [4].

El artículo científico desarrollado por: Esperanza Manrique Rojas, Margarita Ramírez Ramírez, Hilda Beatriz Ramírez Moreno, María del Consuelo Salgado Soto, Nora del Carmen Osuna Millán, Luis Manuel Cerda Suarez con el tema: “Sistema de Gestión Académica a través del desarrollo de Modelo-Vista-Controlador” tuvo como finalidad desarrollar un API utilizando Laravel, JavaScript, HTML5 y CSS3, que retorna información en formato JSON en respuesta a peticiones HTTP. Esto permitió crear diversas aplicaciones para el sistema SIGAF en el futuro, utilizando cualquier tecnología Frontend para consumir los datos. Se empleó el patrón Modelo Vista Controlador (MVC) para desarrollar de manera independiente y coordinada los componentes de datos, interfaz de usuario y lógica de control, logrando un funcionamiento perfecto y una armonía entre todas las secciones del sistema. Este logro fue posible gracias al equipo de desarrollo y las habilidades de cada miembro involucrado [5].

### **1.2.1 Contextualización del Problema**

En la actualidad, la gestión tecnológica se centra en facilitar el acceso a la información de muchos campos, dando lugar a la World Wide Web como fuente integral de investigación y permitiendo que la mayoría de las publicaciones se centren en la red informática.

El sistema que se implementa en la Unidad Educativa es ampliamente utilizado por la mayoría de las instituciones del sector público en Ecuador desde hace décadas en el ámbito educativo, este sistema se centra en el registro de la información personal de los estudiantes para su posterior matriculación, así como en el registro y almacenamiento de las calificaciones por parte de los docentes en hojas de cálculo [4].

La Unidad Educativa Huachi Grande dispone de la tecnología necesaria para que los docentes puedan acceder a Internet desde sus instalaciones. La evolución del sistema manual tradicional en los procesos principales de gestión se hace indispensable para mejorar la rapidez y efectividad en la gestión de las calificaciones y la asistencia de los estudiantes dentro de sus instalaciones. Gracias a la tecnología, los docentes podrán llevar a cabo estas tareas de manera más ágil, lo que se traduce en una mejor experiencia para todo el cuerpo docente.

En esta Unidad Educativa se manejan varios procesos de gestión uno de ellos son el manejo de matrículas, calificaciones, asistencia y reportes de los estudiantes estos procesos toman mucho tiempo y presentan inconvenientes durante el uso de hojas de cálculo por parte del cuerpo docente de la unidad educativa, dichos inconvenientes producen pérdida de información, duplicidad de archivos y prolongados tiempos para realizar reportes, para lo que se requiere establecer un sistema automatizado que disminuya el tiempo empleado para dichos procesos.

Se espera que la implementación de una metodología de desarrollo ágil adecuada para el desarrollo del sistema web de gestión de calificaciones y asistencia de la Unidad Educativa resulte en un proceso de desarrollo más flexible y colaborativo, lo que permitirá una mayor adaptación a las necesidades cambiantes del entorno educativo y una mayor satisfacción de los usuarios finales.

### **1.3 Fundamentación Teórica**

El marco teórico de este trabajo contempla diversos temas que tienen como objetivo el diseño y desarrollo de una plataforma web que sea óptima y logre resultados satisfactorios, con el propósito de facilitar la comunicación e intercambio de información entre docentes, estudiantes y padres de familia. Entre los temas que se incluyen destacan: Tecnologías de la Información y la Comunicación, Tecnologías de Desarrollo Web, Educación, Gestión Escolar, Gestión de Calificaciones y Asistencia,



y Sistema Educativo por Quimestres, así como las herramientas específicas que se emplearán en el proyecto. Resulta fundamental tomar en cuenta los objetivos educativos y administrativos de la institución para garantizar un sistema personalizado y adaptado a las necesidades de la comunidad educativa.

### **1.3.1 Tecnologías de la Información y la Comunicación**

Las Tecnologías de la Información y la Comunicación (TIC) abarcan una amplia gama de herramientas y recursos tecnológicos utilizados para procesar, almacenar, transmitir y gestionar información, incluyendo hardware, software, redes y medios de comunicación. Estas tecnologías han transformado la forma en que vivimos y trabajamos, permitiéndonos acceder, procesar y compartir información de manera más rápida y eficiente en diversos sectores como la educación, la sanidad, las finanzas, la industria y el ocio. Ejemplos comunes de TIC incluyen ordenadores, teléfonos inteligentes, Internet, redes sociales y cámaras digitales.

Las TIC tienen un gran impacto en la economía mundial, el desarrollo social y el bienestar individual, al crear nuevas oportunidades para la innovación, el emprendimiento y el crecimiento económico, al tiempo que mejoran el acceso a servicios esenciales como la sanidad y la educación. Sin embargo, su creciente uso también plantea preocupaciones sobre la privacidad, la seguridad y el acceso, y es necesario utilizarlas de manera responsable y ética para garantizar que todos se beneficien de sus ventajas y minimizar los riesgos potenciales [6].

### **1.3.2 Tecnologías de Desarrollo Web**

Las tecnologías de desarrollo web son un conjunto de herramientas, lenguajes y framework utilizados para crear aplicaciones web interactivos. Estas tecnologías incluyen tanto tecnologías del lado del cliente como del lado del servidor y abarcan

una amplia gama de funcionalidades como el desarrollo front-end, el desarrollo back-end, la gestión de bases de datos y la administración de servidores.

Las tecnologías de desarrollo web front-end se utilizan para crear la interfaz de usuario de un sitio o aplicación web. Incluyen lenguajes como HTML, CSS y JavaScript, así como framework y bibliotecas como React, Angular y Vue.js. Mientras tanto, las tecnologías de desarrollo web back-end se utilizan para construir los componentes del lado del servidor de una aplicación web o sitio web. Incluyen lenguajes de programación como PHP, Python, Ruby y Java, así como marcos de desarrollo web como Laravel, Django, Ruby on Rails y Spring.

### **1.3.3 Educación**

La educación es un proceso vital que permite a las personas adquirir conocimientos y habilidades a lo largo de toda la vida. Su principal objetivo es proporcionar a los individuos las herramientas necesarias para convertirse en miembros activos y productivos de la sociedad, capaces de contribuir a sus comunidades y mejorar su calidad de vida. La educación puede ser formal o informal, y puede abarcar una amplia gama de materias y disciplinas.

La educación es esencial para el desarrollo personal y profesional, y es un derecho humano fundamental que debe estar disponible para todos, sin importar su origen o circunstancias. Además, tiene el poder de transformar vidas, elevar comunidades y promover la justicia social, desempeñando un papel vital en el progreso social y económico [7].

#### **1.3.4 Gestión escolar**

La gestión escolar se refiere a la administración y supervisión de todos los aspectos del funcionamiento de una escuela, incluidos los programas educativos, la gestión del personal, la asignación de recursos, la elaboración de presupuestos y los asuntos estudiantiles. El principal objetivo de la gestión escolar es crear un entorno en el que los alumnos puedan aprender y alcanzar su potencial académico, al tiempo que se garantiza el funcionamiento fluido y eficaz del centro [8].

La gestión escolar eficaz incluye la planificación, organización y coordinación de actividades como el desarrollo del plan de estudios, la formación del profesorado, la evaluación, la disciplina escolar y la interacción con la comunidad. También se encarga de administrar los recursos físicos y educativos del centro. La gestión escolar implica gestionar las relaciones con las partes interesadas, como padres, profesores, alumnos y la comunidad en general, lo que requiere una comunicación y colaboración efectivas para satisfacer las necesidades de todos.

En general, la gestión escolar desempeña un papel fundamental a la hora de proporcionar a los alumnos un entorno de aprendizaje seguro, solidario y eficaz. Requiere un fuerte enfoque en la organización, la comunicación, la colaboración y el liderazgo para garantizar que todos los aspectos de las operaciones de la escuela funcionen de manera óptima.

#### **1.3.5 Gestión de calificaciones y la asistencia**

La gestión de calificaciones y asistencia en educación se refiere a la recopilación, organización y uso de datos para evaluar y reportar el progreso académico de los estudiantes. La gestión de calificaciones incluye la asignación y registro de notas, la creación de escalas de calificación y la generación de informes de progreso. Por otro lado, la gestión de asistencia se enfoca en el seguimiento y supervisión de la asistencia

de los estudiantes, identificando patrones de ausencias y retrasos y proporcionando apoyo adicional a los estudiantes con problemas de asistencia.

Ambos procesos son importantes para garantizar el éxito académico de los estudiantes. Las herramientas de gestión de calificaciones pueden ayudar a proporcionar información útil a los estudiantes sobre su rendimiento y a generar informes para padres y administradores. La gestión de asistencia puede ayudar a identificar a los estudiantes con problemas de asistencia temprano para que los educadores puedan brindar apoyo adicional y prevenir la deserción escolar [9].

### 1.3.6 Sistema Educativo por Quimestres

El nuevo sistema de educación se divide en dos quimestres de cinco meses cada uno. Cada quimestre cuenta con dos parciales en los cuales se evaluarán diferentes insumos, tales como tareas, talleres grupales, investigaciones, pruebas, proyectos y refuerzos académicos. Al final de cada quimestre, se realizará un examen acumulativo. Las notas se otorgarán en una escala de 10/10 y la calificación mínima para aprobar el año es de 7/10.

Además, se establece que los estudiantes que falten más del 10% de las horas lectivas reprobarán automáticamente. Los profesores también tienen la responsabilidad de presentar informes escritos al Ministerio de Educación.

Quimestre 1			Quimestre 2		
Parcial 1	Parcial 2	Examen quimestral 1	Parcial 1	Parcial 2	Examen quimestral 2
Insumo 1	Insumo 1		Insumo 1	Insumo 1	Insumo 1
Insumo 2	Insumo 2		Insumo 2	Insumo 2	Insumo 2
					Insumo ...

Insumo	Insumo		Insumo	Insumo	
...	...		...	...	
80%		20%	80%		20%

Tabla 1. Sistema Educativo por Quimestres.

Elaborado: Morales Carlos, 2023.

### 1.3.7 Framework

Un framework es un conjunto de herramientas, librerías y patrones de diseño que proporcionan una estructura predefinida para el desarrollo de software en el ámbito de la informática y la programación. Al utilizar un framework, los desarrolladores pueden acceder a funcionalidades preconstruidas y características que les permiten concentrarse en la lógica específica de su aplicación, en lugar de preocuparse por la infraestructura y funcionalidad básica. Los frameworks se utilizan en diferentes lenguajes de programación y tienen ventajas como una mayor productividad, facilidad para mantener y actualizar el software, estandarización del código, entre otras, pero también tienen desventajas como limitaciones en la personalización del software y complejidad en el aprendizaje de un nuevo framework.

Los frameworks son una herramienta muy útil en el desarrollo de software, ya que permiten simplificar y acelerar el proceso de desarrollo al proporcionar una estructura predefinida que los desarrolladores pueden utilizar. Aunque tienen ventajas, también tienen desventajas que los desarrolladores deben tener en cuenta al utilizarlos. Los frameworks se utilizan en diferentes aplicaciones y lenguajes de programación, como aplicaciones web, móviles, de escritorio, entre otras, y pueden mejorar la productividad, facilitar el trabajo en equipo y proporcionar una arquitectura de referencia para construir aplicaciones de manera más coherente y eficiente.

### **1.3.8 Tecnologías del Back-End.**

#### **Asp.net**

ASP.NET es un framework web de código abierto desarrollado por Microsoft que es ampliamente utilizado en el desarrollo de aplicaciones web dinámicas e interactivas. La tecnología permite a los desarrolladores crear sitios y aplicaciones web utilizando tecnologías .NET como C#, VB.NET y F#, lo que facilita enormemente el proceso de desarrollo. Además, cuenta con una amplia variedad de controles, como cuadros de texto, botones y etiquetas, que se pueden arrastrar y soltar fácilmente en una página web para acelerar el proceso de desarrollo [10].

Entre las características adicionales de ASP.NET se incluyen la autenticación de usuario integrada, el almacenamiento en caché y la vinculación de datos, lo que lo convierte en una opción popular para aplicaciones web a gran escala. Además, la última versión del marco, ASP.NET Core, se ejecuta en varias plataformas, lo que lo convierte en una opción atractiva para el desarrollo de aplicaciones web basadas en la nube.

#### **Laravel**

Laravel es un framework gratuito y de código abierto diseñado para crear aplicaciones web modernas, robustas y de fácil mantenimiento en PHP. Desde entonces se ha convertido en uno de los frameworks PHP más populares debido a su sintaxis limpia y elegante, que permite a los desarrolladores escribir código más rápido y con menos errores [11].

Laravel incluye numerosas características como un potente sistema ORM (Object-Relational Mapping), un sistema de enrutamiento fácil de usar y una funcionalidad

integrada de autenticación y autorización. Además, cuenta con una variedad de herramientas y bibliotecas integradas que facilitan la realización de tareas comunes de desarrollo web.

El framework sigue el patrón arquitectónico Modelo-Vista-Controlador (MVC), el cual separa la aplicación en tres componentes principales: el modelo (datos e interacción con la base de datos), la vista (interfaz de usuario) y el controlador (maneja las entradas del usuario y actualiza el modelo y la vista). Esta separación permite una mayor organización y facilita el desarrollo y mantenimiento de aplicaciones web grandes y complejas [12].

Además, Laravel cuenta con una gran comunidad de desarrolladores que crean y comparten paquetes (código reutilizable) para ampliar la funcionalidad del framework, lo que hace posible agregar nuevas características a la aplicación sin tener que escribir código desde cero [13].

### **1.3.9 Análisis de framework de desarrollo back-end.**

Se procede a analizar dos frameworks de desarrollo para el back-end: ASP.NET y Laravel, examinando sus ventajas, desventajas y características principales. También se considerará la experiencia previa en el desarrollo de proyectos en niveles anteriores de la universidad para ayudarnos a tomar una decisión informada sobre la selección del framework adecuado. Al evaluar estos aspectos, podremos determinar qué framework será más adecuado para el proyecto en cuestión y asegurarnos de que el desarrollo sea exitoso.

Categoría	ASP.NET	Laravel
Lenguajes.	C#, VB, .NET, F#	PHP

Tipo de framework.	Framework de aplicaciones completas.	Framework de aplicaciones web completas
Curva de aprendizaje.	Curva de aprendizaje pronunciada para principiante.	Fácil de usar para principiantes y con excelente documentación.
Escalabilidad.	Escalable en proyectos grandes y complejos.	Escalable para proyectos pequeños y medianos.
Apoyo comunitario.	Posee una gran y activa comunidad.	Una comunidad en crecimiento con gran cantidad de excelentes recursos.
Velocidad de desarrollo.	Menor velocidad de desarrollo debido a la complejidad.	Mayor velocidad de desarrollo con simplicidad.
Arquitectura.	Arquitectura monolítica.	Arquitectura de microservicios.
Motor de plantilla.	Razor.	Blade.
Mapeo objeto-relacional (ORM).	Entity Framework.	Eloquent.
Ventajas.	Alto rendimiento, gran apoyo de la comunidad, escalable para proyectos complejos.	Fácil de usar para principiantes, mayor velocidad de desarrollo, arquitectura de microservicios, sintaxis sencilla.
Desventajas	Curva de aprendizaje pronunciada, sintaxis compleja, limitado al alojamiento Windows.	Rendimiento menos optimizado, no tan escalable para grandes proyectos, menor apoyo de la comunidad.

Tabla 2. Cuadro comparativo de framework de desarrollo back-end.

Elaborado: Morales Carlos, 2023.

Después de una evaluación minuciosa de las tecnologías de back-end ASP.NET y Laravel, se compararon sus características, ventajas y desventajas en la Tabla 2.



Después de una revisión exhaustiva de ambas opciones, se llegó a la conclusión de que Laravel era la mejor elección para el proyecto debido a las ventajas que se describen en la tabla.

Laravel destaca por su facilidad de uso, su arquitectura limpia y su amplia comunidad de desarrolladores, lo que garantiza una documentación y soporte de calidad. Además, Laravel ofrece una gran cantidad de herramientas integradas y bibliotecas que facilitan la implementación de diversas funcionalidades.

Si bien ASP.NET también cuenta con ventajas como su alto rendimiento y su capacidad para manejar grandes volúmenes de datos, se determinó que Laravel ofrecía una mejor relación costo-beneficio para este proyecto en particular.

En resumen, después de considerar cuidadosamente las características, ventajas y desventajas de ambas tecnologías de back-end, se decidió que Laravel era la mejor opción para este proyecto debido a sus ventajas descritas en la tabla, lo que garantiza el éxito del desarrollo y la implementación del sistema.

Entre las ventajas de Laravel se encuentran:

- Framework robusto y completo para la creación de aplicaciones web.
- Laravel incluye medidas de seguridad integradas para proteger la aplicación web contra vulnerabilidades comunes, como inyección de SQL y XSS.
- Laravel incluye herramientas y características que permiten a los desarrolladores crear aplicaciones web de manera más rápida y eficiente, lo que significa que se pueden desarrollar aplicaciones de manera más rápida y con menos errores.
- Laravel incluye un ORM (Object-Relational Mapping) incorporado llamado Eloquent, que permite a los desarrolladores trabajar con la base de datos de una manera más intuitiva y eficiente.
- Laravel cuenta con una comunidad de desarrolladores activa y comprometida que trabaja en la mejora constante del framework, lo que significa que siempre hay nuevas actualizaciones y recursos disponibles.

- Laravel posee un sistema de autenticación integrado que facilita el registro y autenticación para los usuarios en la aplicación web.
- Laravel cuenta con un sistema de plantillas llamado Blade.

#### **1.4 Sistemas Gestores de Bases de Datos Relacionales.**

##### **SQL Server**

SQL Server es un sistema de gestión de bases de datos relacionales desarrollado por Microsoft. Es utilizado para almacenar y gestionar grandes cantidades de datos de forma estructurada, y ofrece diversas funciones como seguridad, copia de seguridad y recuperación, alta disponibilidad y ajuste del rendimiento.

SQL Server puede integrarse con otros productos de Microsoft como Excel, SharePoint y Power BI, y utiliza el lenguaje SQL para gestionar y recuperar datos. Está disponible en varias ediciones con diferentes características y capacidades, desde la versión Express gratuita hasta la edición Enterprise de gama alta con funciones avanzadas como OLTP en memoria y almacenamiento de datos.

Algunas de las características de Microsoft SQL Server:

- Sistema de gestión de bases de datos relacionales: SQL Server es un potente RDBMS diseñado para almacenar, gestionar y recuperar grandes cantidades de datos estructurados.
- Seguridad de los datos: SQL Server proporciona un completo conjunto de funciones de seguridad para proteger sus datos, como el cifrado, la seguridad basada en funciones y el enmascaramiento de datos.
- Alta disponibilidad: SQL Server ofrece varias funciones para garantizar una alta disponibilidad, como clustering, mirroring y grupos de disponibilidad siempre activos.

- Escalabilidad: SQL Server puede escalarse para satisfacer las necesidades de las empresas más grandes, con soporte para miles de usuarios simultáneos y petabytes de datos.
- Inteligencia empresarial: SQL Server incluye una serie de funciones de inteligencia empresarial, como herramientas de generación de informes, análisis y visualización de datos.
- Integración en la nube: SQL Server se puede utilizar en la nube a través de Azure SQL Database, Azure SQL Managed Instance o SQL Server on Azure Virtual Machines.
- Integración con otros productos de Microsoft: SQL Server se integra bien con otros productos de Microsoft como Excel, SharePoint y Power BI.
- Optimización del rendimiento: SQL Server incluye herramientas para supervisar y optimizar el rendimiento de las bases de datos, como SQL Server Profiler y Database Engine Tuning Advisor.
- Herramientas para desarrolladores: SQL Server proporciona una serie de herramientas para desarrolladores, como SQL Server Management Studio, Visual Studio y PowerShell.
- Funciones avanzadas: SQL Server incluye muchas funciones avanzadas, como OLTP en memoria, índices columnstore y PolyBase, que permite consultar datos en Hadoop o Azure Blob Storage.

## **MySQL**

MySQL es un sistema de gestión de bases de datos relacionales de alto rendimiento y fiabilidad. Una base de datos es una colección organizada de datos que permite almacenar, buscar, ordenar y recuperar información de manera eficiente. MySQL es un servidor de bases de datos que administra el acceso a los datos para garantizar el uso simultáneo de múltiples usuarios, brindando acceso a los datos solo a usuarios autorizados.

MySQL es un servidor multiusuario y de subprocesamiento múltiple, lo que significa que puede manejar múltiples conexiones de usuario y múltiples solicitudes de datos al mismo tiempo. El uso de SQL (Structured Query Language) como lenguaje de consulta estándar permite a los usuarios interactuar con la base de datos de manera fácil e intuitiva.

MySQL fue lanzado por primera vez en 1996, pero su origen se remonta a 1979. Desde entonces, ha sido ampliamente utilizado en una variedad de aplicaciones y sitios web debido a su velocidad, confiabilidad y escalabilidad. Como profesor universitario, es importante destacar que MySQL es un componente esencial en muchos programas de computación y ciencias de la información y que su conocimiento es valioso para el desarrollo de habilidades en la administración de bases de datos [14].

Posee características que son:

- Alto rendimiento: MySQL es conocido por su alto rendimiento y escalabilidad. Puede manejar grandes cantidades de datos y transacciones de manera eficiente.
- Fiabilidad: MySQL es un sistema de base de datos estable y fiable. Ha sido ampliamente probado y utilizado por millones de usuarios en todo el mundo.
- Seguridad: MySQL proporciona características avanzadas de seguridad como autenticación de usuarios, encriptación y controles de acceso para garantizar la privacidad e integridad de los datos.
- Facilidad de uso: MySQL es fácil de instalar, configurar y utilizar. Tiene una interfaz de usuario sencilla e intuitiva, y sus comandos y sintaxis son fáciles de entender.
- Portabilidad: MySQL es independiente de la plataforma, lo que significa que puede ejecutarse en varios sistemas operativos como Windows, Linux y macOS.
- Flexibilidad: MySQL soporta múltiples lenguajes de programación y frameworks, incluyendo PHP, Java, Python y más.
- Replicación: MySQL proporciona una función de replicación que le permite crear copias de su base de datos con fines de copia de seguridad o equilibrio de carga.

- Código abierto: MySQL es un sistema de base de datos de código abierto, lo que significa que es libre de usar y distribuir. También puede modificar y personalizar su código fuente según sus necesidades.

## 1.5 Sistemas Gestores de Bases de Datos No Relacionales

### MongoDB

MongoDB es un sistema de bases de datos NoSQL que almacena datos en documentos flexibles y escalables de tipo JSON. Utiliza un modelo de documentos que permite un diseño de esquemas dinámico y flexible y admite el escalado horizontal mediante la distribución de datos en varios servidores. MongoDB también utiliza un lenguaje de consulta denominado MongoDB Query Language (MQL) y proporciona un amplio conjunto de API y controladores. Es utilizado por muchas grandes organizaciones y startups para diversos casos de uso, como el comercio electrónico, las redes sociales y las aplicaciones IoT.

- MongoDB es una base de datos NoSQL que utiliza un modelo de documentos orientado a objetos en lugar de tablas relacionales.
- Permite el almacenamiento de datos en documentos JSON flexibles, lo que permite un diseño de esquemas dinámico y adecuado para almacenar datos semiestructurados y no estructurados.
- Ofrece escalabilidad horizontal a través de la distribución de datos en varios servidores y funciones como la conmutación por error automática, los conjuntos de réplicas y el equilibrio de carga.
- Utiliza un lenguaje de consulta denominado MongoDB Query Language (MQL), similar a SQL pero diseñado para consultar documentos JSON, y proporciona un amplio conjunto de API y controladores para interactuar con la base de datos.
- Ofrece funciones de alta disponibilidad y escalabilidad como la conmutación por error automática y los conjuntos de réplicas.

- Cuenta con una comunidad de desarrolladores activa y comprometida que trabaja en la mejora constante de la base de datos, lo que significa que siempre hay nuevas actualizaciones y recursos disponibles.

### 1.6 Comparación de bases de datos: Análisis comparativo de diferentes bases de datos.

Característica	Servidor SQL	MySQL	MongoDB
Modelo de datos	Relacional	Relacional	NoSQL basado en documentos
Escalabilidad	Escalado vertical horizontal	Escala horizontal	Escala horizontal
Replicación	Soporte	Soporte	Soporte
Cumplimiento de la normativa ACID	Totalmente conforme	Totalmente conforme	No totalmente conforme
Almacenamiento de datos	Almacenamiento en disco tradicional	Almacenamiento en disco tradicional	Caché en memoria y disco
Rendimiento	Buen rendimiento con grandes volúmenes de datos	Tratamiento rápido y eficaz	Rápido rendimiento de lectura/Escritura
Seguridad	Funciones de seguridad avanzadas	Elementos de seguridad estándar	Elementos básicos de seguridad
Coste	Costoso	Ediciones gratuitas y de pago	Ediciones gratuitas y de pago
Apoyo al desarrollo	Amplio conjunto de herramientas para desarrolladores	Soporte limitado de herramientas para desarrolladores	Soporte limitado de herramientas para desarrolladores
Comunidad de usuarios	Gran comunidad de usuarios	Gran comunidad de usuarios	Gran comunidad de usuarios

Facilidad de uso	Puede ser complejo de configurar y utilizar	Fácil de instalar y utilizar	Fácil de instalar y utilizar
Compatibilidad SQL	Soporte completo de SQL.	Soporte completo de SQL	Soporte limitado de SQL.
Ventajas	Amplio conjunto de funciones.	Bajo coste y fácil de usar	Escalable flexible y rápido
Desventajas	Costoso y complejo.	Soporte limitado de herramientas para desarrolladores	Sin conformidad total con ACID

Tabla 3. Cuadro comparativo entre bases de datos.

Elaborado por: Morales Carlos, 2023.

La Tabla 3 presenta una comparación detallada de las características, ventajas y desventajas de varios sistemas de bases de datos, con el fin de evaluar cuál es la mejor opción para nuestra aplicación web. Después de una evaluación cuidadosa de cada opción, se determinó que MySQL era la mejor elección debido a sus ventajas y popularidad en el mercado. Además, la integración con Laravel es muy intuitiva ya que sus librerías están integradas en la instalación, lo que hará que el proceso de desarrollo de la aplicación sea más fácil y eficiente, mejorando su rendimiento. En resumen, la elección de MySQL como sistema de bases de datos para nuestra aplicación web se basa en una evaluación exhaustiva de las opciones y en su integración intuitiva con Laravel, lo que garantizará un rendimiento óptimo de la aplicación.

## 1.7 Clasificación de metodologías de desarrollo

### Metodología de desarrollo tradicional

Las metodologías de desarrollo de software tradicionales se caracterizan por tener una estructura de desarrollo claramente establecida, lineal y con poca flexibilidad ante

cambios en un entorno en constante evolución. Algunos ejemplos de metodologías tradicionales son Waterfall (cascada), V-Model y Modelos Incrementales. Estas metodologías se centran en planificar todo el proceso de desarrollo de software antes de comenzar a escribir código, lo que puede llevar a un mayor tiempo y costo en el proyecto. Además, su rigidez puede dificultar el manejo de cambios y requerimientos nuevos por parte del cliente en el transcurso del proyecto. Sin embargo, estas metodologías han sido utilizadas durante décadas y han demostrado tener éxito en proyectos de gran envergadura o donde se requiere una mayor atención a los detalles y control del proceso [15].

### **Metodologías ágiles de desarrollo**

Las metodologías de desarrollo ágil son un enfoque del desarrollo de software que prioriza la flexibilidad, la colaboración y la iteración rápida sobre la planificación y la documentación rígidas. En lugar de crear un plan detallado por adelantado y seguirlo estrictamente, las metodologías ágiles hacen hincapié en trabajar en ciclos cortos llamados sprints, durante los cuales el equipo de desarrollo produce una pequeña pieza de funcionalidad que puede probarse y refinarse antes de pasar al siguiente sprint.

Uno de los principios clave del desarrollo ágil es que los requisitos y las soluciones evolucionan a través del esfuerzo colaborativo de equipos autoorganizados y multifuncionales. Las metodologías ágiles también dan prioridad a la satisfacción del cliente mediante la entrega temprana y continua de software valioso, y responden al cambio con rapidez para seguir el ritmo de las necesidades y circunstancias cambiantes [16].

Algunas metodologías ágiles habituales son Scrum, Kanban y Extreme Programming (XP). Scrum es un marco ágil popular que implica un Scrum Master que facilita el progreso del equipo y elimina cualquier impedimento a su trabajo, un Propietario de Producto que prioriza el trabajo pendiente en función del valor empresarial y un Equipo de Desarrollo que colabora para entregar software que funcione [17].



XP es una metodología Ágil más técnica que hace hincapié en prácticas de codificación como las pruebas automatizadas y la integración continua para mejorar la calidad del software. En general, las metodologías de desarrollo ágil se han popularizado en los últimos años por su flexibilidad, adaptabilidad y capacidad para ofrecer software de alta calidad de forma rápida y eficaz. Como profesor universitario, te animo a que aprendas más sobre las metodologías ágiles y sus aplicaciones en el desarrollo de software y la gestión de proyectos [18].

### 1.8 Análisis comparativo de metodologías de desarrollo

<b>Metodologías de Desarrollo</b>	<b>Tradicionales</b>	<b>Ágiles</b>
Enfoque	Planificación detallada y rigurosa, con énfasis en la documentación.	Enfoque más flexible y adaptable a cambios, con énfasis en la entrega rápida de software funcional.
Ciclo de vida	Ciclo de vida secuencial, con énfasis en la fase de diseño.	Ciclo de vida iterativo e incremental, con énfasis en la entrega constante de software funcional.
Participantes	Participantes especializados y jerarquizados.	Participantes multidisciplinares y autónomos.
Roles	Roles definidos y jerarquizados.	Roles más flexibles y adaptables a las necesidades del proyecto.
Planificación	Planificación detallada y rigurosa, con énfasis en el control del tiempo y los recursos.	Planificación más flexible y adaptable a los cambios, con énfasis en la entrega constante de software funcional.

Documentación	Documentación extensa y detallada, con énfasis en el seguimiento y la trazabilidad.	Documentación más reducida y enfocada en la entrega constante de software funcional.
Comunicación	Comunicación formal y estructurada, con énfasis en los informes y reuniones programadas.	Comunicación más informal y flexible, con énfasis en la colaboración constante y la retroalimentación continua.
Flexibilidad	Poca flexibilidad para cambios y adaptaciones en el proceso de desarrollo.	Mayor flexibilidad para cambios y adaptaciones en el proceso de desarrollo.
Calidad	Énfasis en la calidad del proceso y la documentación.	Énfasis en la calidad del software funcional y la satisfacción del cliente.

Tabla 4. Análisis comparativo de metodologías de desarrollo tradicionales y ágiles.

Elaborado por: Morales Carlos, 2023.

## Scrum

Scrum es un marco ágil para el desarrollo de software que hace hincapié en el trabajo en equipo, la colaboración y la iteración rápida. Se basa en los principios de transparencia, inspección y adaptación, y está diseñado para ayudar a los equipos a entregar software de alta calidad de forma rápida y flexible.

En Scrum, un equipo multifuncional de desarrolladores, probadores y otras partes interesadas trabajan juntos en ciclos cortos llamados sprints, que suelen durar entre 1 y 4 semanas. Al principio de cada sprint, el equipo celebra una reunión de planificación del sprint para identificar el trabajo a realizar y establecer un objetivo para el sprint. Durante el sprint, el equipo celebra reuniones diarias para revisar el progreso y discutir cualquier problema u obstáculo.

Al final del sprint, el equipo celebra una reunión de revisión del sprint para demostrar el trabajo realizado y recabar opiniones de las partes interesadas. También celebran una reunión de retrospectiva del sprint para reflexionar sobre el proceso del sprint e identificar oportunidades de mejora.

La siguiente figura de SCRUM enfatiza la fase de Sprint:

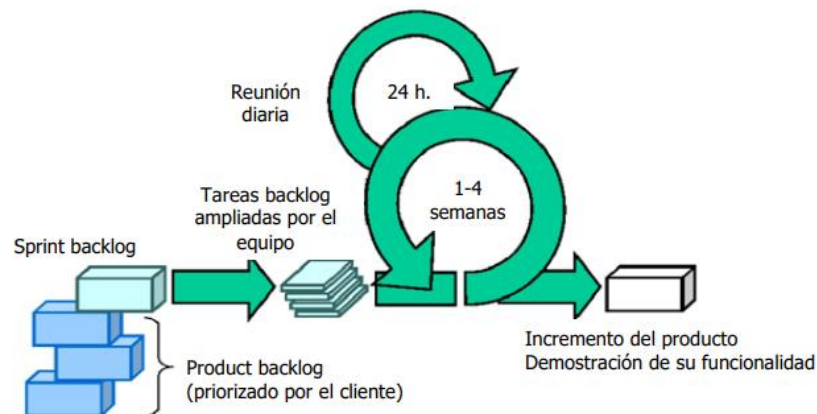


Figura 1. Esquema de la metodología Scrum.

Fuente: [19].

- Scrum también incluye varios roles y artefactos que ayudan a garantizar el éxito del marco. Estos incluyen:
- Scrum Master: Un facilitador que ayuda al equipo a implementar Scrum y eliminar cualquier impedimento para su progreso.
- Propietario del Producto: Un representante de las partes interesadas que es responsable de priorizar el backlog de trabajo y asegurar que el equipo ofrece valor al cliente.
- Sprint Backlog: Una lista del trabajo que debe completarse durante el sprint actual.
- Backlog del producto: Una lista priorizada de todo el trabajo que debe completarse para el proyecto.

En general, Scrum es un marco poderoso para el desarrollo de software que promueve el trabajo en equipo, la comunicación y la mejora continua. Como profesor universitario, le animo a aprender más sobre Scrum y sus aplicaciones en el desarrollo de software y gestión de proyectos.

### **Programación Extrema**

La Programación Extrema es una metodología ágil de desarrollo de software que hace hincapié en las prácticas de ingeniería de software y en la participación del cliente para producir software de alta calidad de forma rápida y eficaz. Se basa en los principios de simplicidad, retroalimentación, comunicación y valentía.

XP incluye varias prácticas que ayudan a alcanzar estos principios. Estas prácticas incluyen:

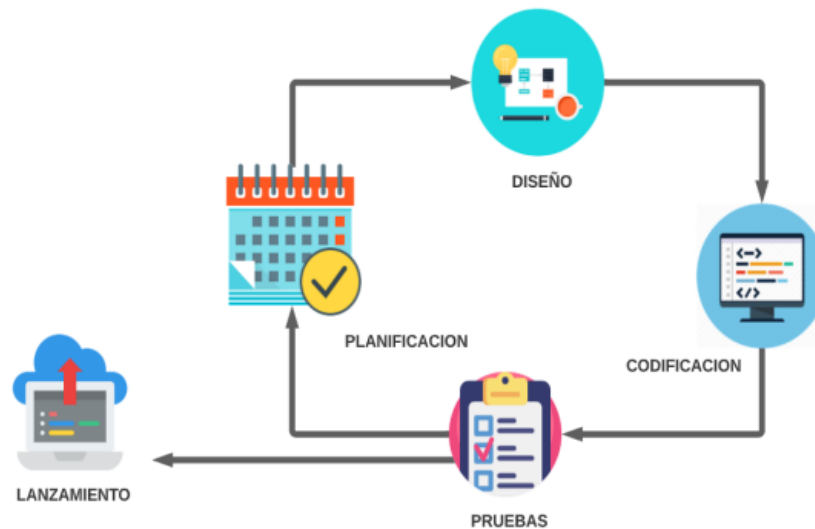


Figura 2. Modelo de metodología Programación Extrema.

Fuente: [20].

Desarrollo basado en pruebas (TDD): En TDD, los desarrolladores escriben pruebas automatizadas antes de escribir el código real. Esto ayuda a garantizar que el código

funciona como se espera y puede modificarse fácilmente sin romper la funcionalidad existente.

- Programación en parejas: En la programación por parejas, dos desarrolladores trabajan juntos en el mismo ordenador para escribir el código. Esto ayuda a garantizar que el código sea de alta calidad, reduce el riesgo de errores y ayuda a difundir el conocimiento y las habilidades entre los miembros del equipo.
- Integración continua: En la integración continua, los desarrolladores integran con frecuencia sus cambios en una base de código compartida y ejecutan pruebas automatizadas para detectar cualquier error en una fase temprana del ciclo de desarrollo.
- Refactorización: En la refactorización, los desarrolladores mejoran el diseño y la calidad del código sin cambiar su comportamiento externo. Esto ayuda a reducir la deuda técnica y facilita el mantenimiento y la ampliación del código.
- Diseño simple: En el diseño simple, los desarrolladores se centran en crear la solución más sencilla que cumpla los requisitos. Esto ayuda a reducir la complejidad y hace que el código sea más fácil de entender y modificar.

XP también hace hincapié en la participación del cliente a través de prácticas como el cliente in situ, las historias de usuario y las versiones frecuentes. Esto ayuda a garantizar que el software satisface las necesidades del cliente y proporciona valor al negocio.

En general, Programación Extrema es una potente metodología ágil que fomenta las prácticas de ingeniería de software y la implicación del cliente para producir software de alta calidad de forma rápida y eficaz. Como profesor universitario, le animo a aprender más sobre XP y sus aplicaciones en el desarrollo de software y la gestión de proyectos [21].

### 1.8.1 Análisis comparativo de metodologías de desarrollo ágiles.

En este análisis, hemos examinado dos metodologías ágiles: Scrum y XP. Estas metodologías son herramientas útiles para gestionar diferentes aspectos y procesos dentro de nuestros equipos, con el objetivo de mejorar el proceso de desarrollo de software. Programación Extrema se basa en una lista de principios básicos que promueven valores como la simplicidad, la comunicación, la retroalimentación y la valentía. Por otro lado, con el método Scrum es posible entregar versiones fiables, incluso ante cambios en los requisitos del sistema.

En resumen, ambas metodologías ágiles son complementarias y ofrecen enfoques diferentes para mejorar la gestión de proyectos de desarrollo de software en equipo. Es importante tener en cuenta que la selección de la metodología adecuada dependerá de las necesidades específicas de cada proyecto y equipo.

<b>Características</b>	<b>Scrum</b>	<b>Programación Extrema (XP)</b>
Principios	Transparencia, inspección y adaptación	Simplicidad, retroalimentación, comunicación y valentía
Enfoque	Gestión de proyectos	Ingeniería de software
Procesos clave	Sprints, reuniones diarias, reuniones de revisión y retrospectiva del sprint	Desarrollo basado en pruebas (TDD), programación en parejas, integración continua, refactorización y diseño simple
Rol clave	Scrum Master	Desarrollador
Artefactos clave	Sprint Backlog y Product Backlog	Código y pruebas automatizadas
Participación del cliente	A través del Propietario del Producto y las	A través del cliente in situ, historias de usuario y versiones frecuentes

	reuniones de revisión del sprint	
--	----------------------------------	--

Tabla 5. Cuadro comparativo de metodologías ágiles.

Elaborado por: Morales Carlos, 2023.

Después de un análisis riguroso, se determinó que la metodología ágil XP era la más apropiada para el desarrollo del sistema web. Una de las principales ventajas de XP es su habilidad para entregar pequeñas historias de usuario completamente funcionales y probadas incrementalmente, lo que garantiza la entrega del producto final en etapas más manejables. Además, XP se enfoca intensamente en la verificación y las pruebas, lo que garantiza que las entregas del producto sean de alta calidad. La elección de XP permitió un desarrollo más eficiente y enfocado en el usuario, mejorando la satisfacción del cliente y el éxito general del proyecto. En resumen, la selección de XP se basó en su capacidad para facilitar el desarrollo ágil, centrado en el usuario y enfocado en la calidad del producto.

### 1.8.2 Livewire

Livewire es una biblioteca de código abierto para Laravel que permite a los desarrolladores crear interfaces de usuario dinámicas utilizando solo PHP, sin la necesidad de escribir código en JavaScript del lado del cliente. Esto permite un desarrollo más rápido, un código más fácil de mantener y una mejor accesibilidad para los usuarios que puedan tener JavaScript desactivado. Con Livewire, los desarrolladores pueden crear componentes que encapsulan la interfaz de usuario y el comportamiento de una parte específica de una página web. Cuando un usuario interactúa con un componente Livewire, éste envía una petición AJAX al servidor, que actualiza el estado del componente y vuelve a renderizar el HTML.

Livewire también incluye características como la vinculación bidireccional de datos, la validación de formularios y los ganchos del ciclo de vida de los componentes. Además, es altamente extensible, lo que permite a los desarrolladores crear sus propios componentes y ganchos personalizados. En resumen, Livewire facilita la creación de interfaces de usuario interactivas utilizando PHP del lado del servidor en lugar de JavaScript, lo que ahorra tiempo y hace que el código sea más fácil de mantener.[11].

### **1.8.3 Bootstrap**

Bootstrap es un Framework de trabajo front-end popular para aplicaciones web responsivas y adaptables a cualquier tamaño de pantalla, desarrollado originalmente por Twitter y ahora mantenido por un equipo de desarrolladores y colaboradores. Incluye una gran variedad de componentes HTML, CSS y JavaScript que son fáciles de usar para crear interfaces de usuario rápidas y personalizadas. Además, los componentes se adaptan a una variedad de tamaños de pantalla y dispositivos para una experiencia de usuario óptima [22].

Las principales características de Bootstrap incluyen un potente sistema de rejilla para organizar el contenido de la página, gran variedad de componentes de interfaz de usuario predefinidos, como botones, formularios y menús de navegación, y un conjunto de estilos CSS personalizables y extensibles [23].

### **1.8.4 Laragon**

Laragon es un entorno de servidor local de alto rendimiento para sistemas operativos Windows, especialmente diseñado para desarrolladores web que trabajan con PHP, Node.js, MySQL y Apache. Se trata de un paquete de software gratuito, fácil de usar y ligero que incluye todas las herramientas y componentes necesarios para desarrollar, probar y ejecutar aplicaciones web en su ordenador Windows.



Con Laragon, los desarrolladores pueden crear y administrar fácilmente múltiples aplicaciones web en su máquina local, lo que les permite configurar un entorno de desarrollo en pocos clics. Además, Laragon cuenta con características adicionales como soporte SSL, hosts virtuales automáticos y recarga automática, que pueden mejorar significativamente el flujo de trabajo de desarrollo.

### **1.8.5 Visual Studio Code**

Visual Studio Code es un editor de código de código abierto gratuito desarrollado por Microsoft. Este editor está diseñado para ser ligero y rápido, lo que permite una experiencia de codificación ágil y eficiente. Además, Visual Studio Code cuenta con características poderosas de edición y depuración que pueden ayudar a los desarrolladores a mejorar la calidad de su código.

Este editor posee integración con Git, que es un sistema de control de versiones ampliamente utilizado por los desarrolladores para gestionar cambios en el código. Además, el editor incluye un terminal integrado, que permite a los desarrolladores ejecutar comandos directamente desde el editor.

### **1.8.6 GitHub**

GitHub es una plataforma de alojamiento de código fuente que permite a los desarrolladores compartir y colaborar en sus proyectos de programación. La plataforma cuenta con un sistema de control de versiones Git, facilitando el trabajo en equipo y la organización del código en diferentes versiones. GitHub es una herramienta importante para los desarrolladores y el control de versiones ayuda a evitar errores y confusiones en proyectos en equipo [24].

GitHub es una herramienta esencial para los desarrolladores que buscan colaborar en proyectos de programación. Su sistema de control de versiones Git y su plataforma de alojamiento de código fuente permiten a los desarrolladores trabajar en conjunto de manera efectiva y eficiente.

## **1.9 Objetivos**

### **1.9.1 Objetivo General**

Automatizar la gestión de calificaciones y asistencia de la Unidad Educativa Huachi Grande.

### **1.9.2 Objetivos Específicos**

- Caracterizar los procesos involucrados en la gestión de calificaciones y asistencia para levantar los requerimientos funcionales para el desarrollo de la aplicación web.
- Determinar una metodología de desarrollo adecuada para el desarrollo del sistema web para la gestión de calificaciones de la Unidad Educativa.
- Desarrollar la aplicación web para automatizar los procesos de gestión de calificaciones y asistencia en la Unidad Educativa.

## **CAPÍTULO II METODOLOGÍA**

### **2.1 Materiales**

Se emplearon distintas fuentes de información, incluyendo revistas científicas, sitios web, trabajos de titulación del área de informática y software relevante, durante el desarrollo de este proyecto de investigación. Además, se llevó a cabo una entrevista

con el cuerpo docente de la Unidad Educativa para identificar los requisitos funcionales necesarios para la implementación del proyecto.

La utilización de diversas fuentes de información y la realización de entrevistas permitieron obtener una visión completa y precisa de los requerimientos del proyecto. De esta manera, se logró recopilar información valiosa de diferentes fuentes, lo que posibilitó una evaluación crítica y un análisis profundo de los requisitos necesarios para el proyecto. Como resultado, se pudo crear una solución personalizada y adecuada para las necesidades de la Unidad Educativa.

### **2.1.1 Institucionales**

- Repositorio virtual de la Universidad Técnica de Ambato.
- Instalaciones de la Unidad Educativa Huachi Grande.

### **2.1.2 Humano**

- Cuerpo docente de la unidad Educativa.
- Investigador.

### **2.1.3 Otros**

- Laptop.
- Internet.
- Disco duro externo.
- Transporte.
- Visual Studio Code.
- Laragon
- Framework Laravel.

- Base de datos MySQL.
- Servidor de aplicaciones.

#### 2.1.4 Económicos

N°	Detalle	Unidad	Cantidad	Valor Unitario	Valor Total
1	Laptop	c/u	1	1200	1200
2	Internet	c/mes	6	28	162
3	Disco duro externo	c/u	1	50	50
4	Transporte	c/u	30	0,6	18
5	Framework Laravel	c/u	1	0	0
6	Base de datos MySQL	c/u	1	0	0
7	Servidor de aplicaciones	c/mes	6	5	30
8	Total				1460

Tabla 1. Presupuesto

Elaborado por: Morales Carlos, 2023.

#### ENTREVISTA.

Para la determinación de los requerimientos del proyecto, se realizó una entrevista con el personal docente de la Unidad Educativa Huachi Grande, con el objetivo de comprender sus procesos y necesidades en cuanto a la gestión de calificaciones y asistencia.

El objetivo principal de la entrevista fue identificar los requerimientos y características necesarias para la automatización de los procesos de gestión de calificaciones y asistencia en la Unidad Educativa Huachi Grande. De esta manera, se pudo obtener una comprensión detallada de los procesos existentes y los requisitos específicos de esta Unidad Educativa, lo que permitirá el desarrollo de un sistema acorde a sus necesidades.

N°	Pregunta
1	¿Cómo manejan actualmente las notas y la asistencia de los estudiantes?
2	¿Cuál es el proceso que siguen para registrar las notas y la asistencia de los estudiantes?
3	¿Cómo se realiza la comunicación de los resultados a los estudiantes y a sus padres?
4	¿Cuáles son los principales problemas o desafíos que enfrentan actualmente en la gestión de notas y asistencia?
5	¿Han considerado alguna vez la implementación de un sistema en línea para la gestión de notas y asistencia? ¿Por qué sí o por qué no?
6	¿Qué características consideran más importantes en un sistema de gestión de notas y asistencia?
7	¿Cómo se aseguran de que los datos de notas y asistencia se mantengan confidenciales y seguros?
8	¿Qué beneficios creen que traería la implementación de un sistema de gestión de notas y asistencia en línea?
9	¿Cuáles son los criterios para determinar qué usuarios tendrán acceso al sistema?

Tabla 2. Entrevista

Elaborado por: Morales Carlos, 2023.

## 2.2 Métodos

### 2.2.1 Modalidad de Investigación

La presente investigación se basa en tres modalidades: campo, bibliográfica-documental y aplicada. La implementación de metodologías ágiles permitirá al proyecto adaptarse a cambios no previstos.

**Modalidad de campo:** la investigación se realiza en el lugar donde se está desarrollando el proyecto y se relaciona directamente con las personas interesadas. Se llevan a cabo los estudios y análisis necesarios para comprender el problema y, de esta manera, formular la mejor sistematización de los servicios que ofrece la Unidad Educativa Huachi Grande.

**Modalidad bibliográfica o documental:** la investigación se basa en la revisión de material bibliográfico y documental. Esta modalidad incluye un conjunto de fases que abarcan la observación, la indagación, la conceptualización y la revisión de diferentes criterios de diversos autores sobre el problema, utilizando diferentes documentos que se encuentran en diversos repositorios

**Modalidad aplicada:** se utiliza esta modalidad porque se aplica el conocimiento adquirido en los diferentes niveles de la carrera universitaria.

### 2.2.2 Población Y Muestra

La población para la presente investigación está constituida así:

N°	Área	Descripción	Numero	Porcentaje (%)
1	Personal administrativo	Rectora	1	25%

2	Personal administrativo	Vicerrectora	1	25%
4	Personal docente	Docentes	2	50%
Total			4	100%

Tabla 6. Población de estudio.

Elaborado por: Morales Carlos, 2023.

Dado que la población de docentes de la unidad educativa es menor o igual a 100, no es necesario realizar una muestra y se puede trabajar con toda la población establecida por el personal docente de la Unidad Educativa Huachi Grande.

### 2.2.3 Recolección De Información

La recopilación de información se llevará a cabo mediante la observación y entrevista al cuerpo docente de la Unidad Educativa Huachi Grande, con el fin de sustentar el marco teórico.

<b>Preguntas Básicas</b>	<b>Explicación</b>
¿Para qué?	Para lograr los objetivos de la investigación.
¿De qué personas u objetos?	Cuerpo docente de la Unidad Educativa.
¿Sobre qué aspectos?	Proceso de gestión de calificaciones y asistencia de los estudiantes.
¿Quién?	Investigador.
¿A quiénes?	Rectora y cuerpo docentes de la Unidad Educativa.
¿Cuándo?	Octubre 2022 - Marzo 2023.
¿Dónde?	Unidad Educativa Huachi Grande.
¿Cuántas veces?	Una
¿Cómo?	Entrevista.
¿Qué técnicas de recolección?	Observación.

¿Con qué?	Libreta de anotaciones y pauta de entrevista.
-----------	---

Tabla 3. Recolección de información.

Elaborado por: Morales Carlos, 2023.

#### **2.2.4 Procesamiento y análisis de datos:**

Se proceso y analizo la información obtenida, siguiendo los siguientes pasos:

- Se asistió a las reuniones de la Unidad Educativa con el fin de observar el proceso y su fluidez.
- Se analizo los apuntes y documentación mantenida por la Unidad Educativa, como reportes de calificaciones, reportes de asistencia de los docentes, entre otros.
- Se solicito reuniones con el cuerpo docente para identificar sus necesidades y expectativas en cuanto a la educación en la Unidad.
- Se elaboro cuadros comparativos con la información obtenida de las reuniones y entrevistas con el cuerpo docente y la documentación analizada.

Con estos pasos se buscó obtener información precisa y detallada para sustentar la investigación y lograr los objetivos propuestos.

### **CAPITULO III. RESULTADOS Y DISCUSION**

#### **3.1 Análisis y discusión de resultados de la entrevista.**

Se ha realizado una síntesis de todas las entrevistas llevadas a cabo con los cuatro empleados del cuerpo docente.



<b>Pregunta</b>	<b>Resultado</b>	<b>Conclusión</b>
¿Cómo manejan actualmente las notas y la asistencia de los estudiantes?	Algunos profesores utilizan hojas de Excel para llevar un registro de las notas y la asistencia de los estudiantes, mientras que otros utilizan registros en papel.	En ambos casos, los profesores de la unidad educativa recopilan los datos manualmente esta práctica puede ser tediosa y propensa a errores.
¿Cuál es el proceso que siguen para registrar las notas y la asistencia de los estudiantes?	<ul style="list-style-type: none"> <li>• Crean una hoja de cálculo para cada curso que están enseñando.</li> <li>• En cada hoja de cálculo, crean columnas para el nombre del estudiante, número de identificación, asistencia, tareas, exámenes y la nota final.</li> <li>• Utilizamos fórmulas para calcular la nota final de cada estudiante, basándonos en el peso asignado a cada actividad (asistencia, tareas, exámenes, etc.).</li> <li>• Actualizan las hojas de cálculo regularmente, asegurándonos de que las notas y la asistencia de los estudiantes estén siempre actualizadas.</li> </ul>	
¿Cómo se realiza la comunicación de los	<ul style="list-style-type: none"> <li>• Preparan informes de calificaciones y</li> </ul>	Se preparan informes de calificaciones y asistencia.

<b>Pregunta</b>	<b>Resultado</b>	<b>Conclusión</b>
<p>resultados a los estudiantes y a sus padres?</p>	<p>asistencia para cada estudiante, utilizando la información registrada en Excel.</p> <ul style="list-style-type: none"> <li>• En caso de que algún estudiante tenga un bajo rendimiento o una asistencia insuficiente, nos comunicamos directamente con él o con los padres de familia para analizar las causas y buscar soluciones conjuntas.</li> </ul>	<p>Si un estudiante tiene un bajo rendimiento o asistencia insuficiente, se lo comunicación directa con él representante del estudiante.</p>
<p>¿Cuáles son los principales problemas o desafíos que enfrentan el cuerpo docente actualmente en la gestión de notas y asistencia de los estudiantes?</p>	<ul style="list-style-type: none"> <li>• Errores de ingreso de datos al momento de ingresan manualmente las notas y la asistencia en hojas de Excel o leccionarios, existe un alto riesgo de errores humanos, como errores de tipeo o errores de transcripción.</li> <li>• Dificultad para compartir información con otros profesores, padres de familia o estudiantes.</li> <li>• Falta de integración con las calificaciones del</li> </ul>	<p>Los problemas identificados han generado inconvenientes significativos en la gestión de calificaciones y asistencia de los estudiantes en la Unidad Educativa. Estos problemas se busca solucionarlos con el desarrollado un sistema web que permite la gestión eficiente de las calificaciones y la asistencia de los estudiantes.</p>

Pregunta	Resultado	Conclusión
	<p>mismo curso con todas las asignaturas.</p> <ul style="list-style-type: none"> <li>Actualización manual para reflejar cambios en las calificaciones y la asistencia.</li> </ul>	
<p>¿Le gustaría recibir notificaciones de manera automática en caso de que un estudiante esté perdiendo clases debido a la inasistencia?</p>	<p>El cuerpo de docentes le gustaría notificaciones automáticas de los estudiantes con un porcentaje de asistencia insuficiente para tomar las medidas adecuadas.</p>	<p>Los docentes requieren tener conocimiento al finalizar cada quimestre de los estudiantes para tomar medidas adecuadas.</p>
<p>¿Qué características consideran más importantes en un sistema de gestión de notas y asistencia?</p>	<p>El cuerpo de docentes considera que un sistema de gestión de notas y asistencia debe ser fácil de usar, accesible, seguro, funcional.</p>	<p>Es fundamental que el sistema diseñado sea completamente seguro, funcional y fácil de usar, ya que la Unidad Educativa aún no tiene mucha experiencia en el uso de sistemas informáticos. Por lo tanto, el sistema debe estar diseñado de manera intuitiva y con una interfaz amigable, para que los usuarios puedan adaptarse fácilmente a su uso y disfruten de una experiencia satisfactoria en su utilización.</p>
<p>¿Cómo se aseguran de que los datos de notas y asistencia se mantengan confidenciales y seguros?</p>	<ul style="list-style-type: none"> <li>Acceso restringido el cuerpo docente limita el acceso a los datos de notas y asistencia solo al personal escolar autorizado que tiene una necesidad legítima de acceder a ellos.</li> <li>Para proteger la información la mayoría del cuerpo docente establece contraseñas en</li> </ul>	<p>no tiene mucha experiencia en el uso de sistemas informáticos. Por lo tanto, el sistema debe estar diseñado de manera intuitiva y con una interfaz amigable, para que los usuarios puedan adaptarse fácilmente a su uso y disfruten de una experiencia satisfactoria en su utilización.</p>

Pregunta	Resultado	Conclusión
	<p>sus hojas de Excel como medida de seguridad.</p> <ul style="list-style-type: none"> <li>• Parte del cuerpo docente cuenta con copias de los archivos que contiene las calificaciones y asistencia de los estudiantes en sus computadoras personales o en dispositivos USB.</li> </ul>	
<p>¿Qué beneficios creen que traería la implementación de un sistema de gestión de calificaciones y asistencia en línea?</p>	<ul style="list-style-type: none"> <li>• Mayor eficiencia: un sistema en línea facilitaría la tarea de ingresar y gestionar las notas y la asistencia, ahorrando tiempo y reduciendo la carga de trabajo.</li> <li>• Accesibilidad: un sistema en línea permitiría el acceso a la información de notas y asistencia desde cualquier lugar y en cualquier momento, mejorando la comunicación y colaboración entre estudiantes, padres y profesores.</li> </ul>	<p>La implementación de un sistema de gestión de calificaciones y asistencia en línea proporcionaría una plataforma más accesible y colaborativa para mejorar la gestión educativa y el seguimiento del rendimiento estudiantil.</p>

Pregunta	Resultado	Conclusión
	<ul style="list-style-type: none"> <li>• Precisión: un sistema en línea permitiría una entrada de datos más precisa, reduciendo errores e inconsistencias en los registros.</li> <li>• Análisis de datos: un sistema en línea permitiría un mejor análisis de los datos de calificaciones y asistencia, identificando tendencias y áreas de mejora.</li> <li>• Comunicación: un sistema en línea permitiría una comunicación más eficiente y efectiva sobre el progreso académico y la asistencia.</li> <li>• Personalización: un sistema en línea permitiría la adaptación a las necesidades específicas de la institución, mejorando la efectividad y eficiencia del proceso de gestión de notas y asistencia.</li> </ul>	

Pregunta	Resultado	Conclusión
¿Cuáles son los criterios para determinar qué usuarios tendrán acceso al sistema?	Los usuarios principales que tendrán acceso serán los docentes activos durante el período en cuestión, incluyendo al rector o rectora, secretaria, inspectores y profesores.	Es importante considerar que, además de estos usuarios principales, también puede haber otros usuarios secundarios o externos.

Tabla 7. Resultado de la entrevista.

Elaborado por: Morales Carlos, 2023.

### 3.2 Desarrollo de la propuesta

Se implementó el patrón de arquitectura MVC (Modelo Vista Controlador) en el desarrollo de un sistema web para automatizar la gestión de calificaciones y asistencia en la unidad educativa Huachi Grande. La elección de este patrón se basó en su capacidad para separar claramente la lógica de negocio, la presentación de la interfaz y el control de flujo de datos en diferentes componentes, lo que facilita la escalabilidad y mantenimiento del sistema. Además, se utilizó el framework Laravel, reconocido por su compatibilidad con el patrón MVC y su facilidad de uso en el desarrollo web.

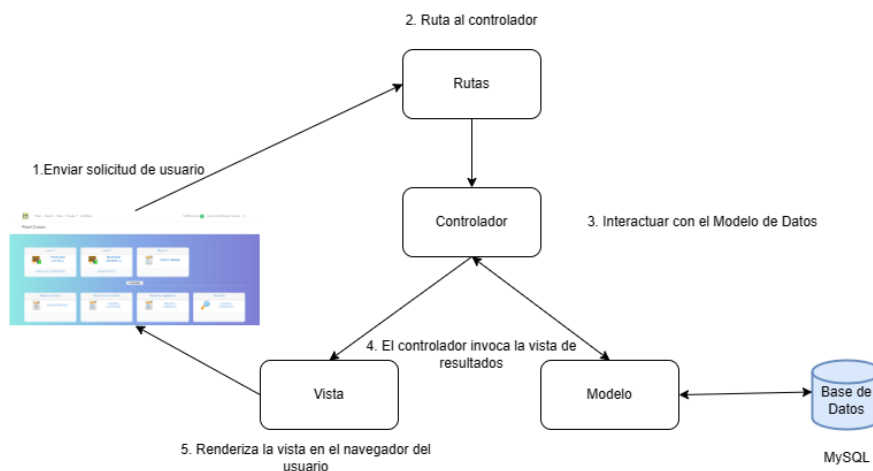


Figura 3. Arquitectura de sistema web con framework Laravel

Elaborado por: Morales Carlos, 2023.

### **3.3 Aplicación de la metodología**

Se utilizó la metodología XP en el desarrollo del sistema web para lograr pruebas unitarias y resultados rápidos. La metodología también permitió la posibilidad de adaptarse a cambios en los requerimientos del proyecto.

#### **3.3.1 Arquitectura del sistema web**

La solución propuesta ofrece una arquitectura modular y bien definida para la aplicación, lo que simplifica la tarea de mantener y ampliar el sistema con nuevas funcionalidades. Además, la separación clara entre la lógica de negocio, la presentación y el control permite una mejor organización del código y hace que sea más fácil de entender.

#### **3.3.2 Descripción de los procesos.**

##### **Proceso ingreso de calificaciones.**

En este proceso se lleva a cabo una serie de eventos, los cuales permiten el registro de calificaciones en los insumos impartidos en una clase por un docente, así como también de las diferentes asignaturas que se le fueron asignadas y de los estudiantes que se encuentran en los cursos ya mencionados.

Los procesos que se siguen son los siguientes:

- Iniciar sesión (login).
- Acceder al sistema.
- Seleccionar el curso y el paralelo correspondiente.
- Seleccionar el tipo de insumo y crearlo.
- Desplegar la lista de estudiantes.
- Ingresar la calificación correspondiente a cada estudiante.

- Registrar las calificaciones.
- Salir del sistema.

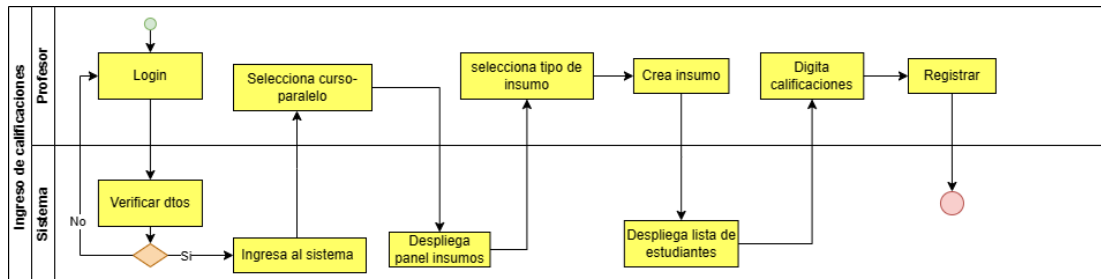


Figura 4. Proceso de ingreso de calificaciones.

Elaborado por: Morales Carlos, 2023.

En este proceso se lleva a cabo una serie de eventos, los cuales permiten el registro de la asistencia de los estudiantes en una clase impartida por un docente.

Los procesos que se siguen son los siguientes:

- Iniciar sesión (login).
- Acceder al sistema.
- Seleccionar el curso y el paralelo correspondiente.
- Seleccionar la opción de asistencia.
- Desplegar la lista de estudiantes.
- Registrar el horario correspondiente.
- Registrar la asistencia correspondiente a cada estudiante.
- Salir del sistema.



## Proceso control de asistencia.

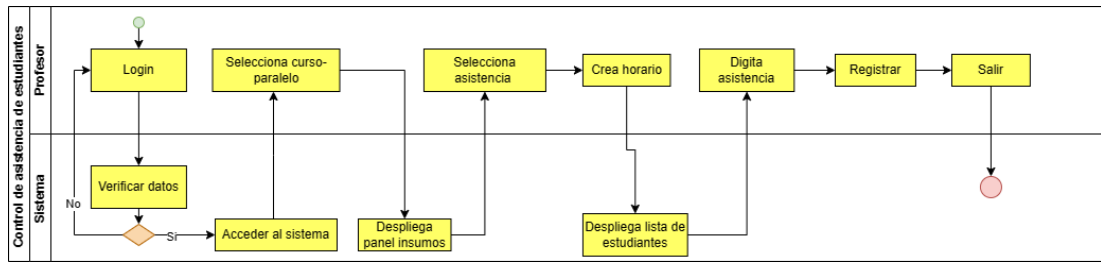


Figura 5. Proceso de control de asistencia de estudiantes.

Elaborado por: Morales Carlos, 2023.

### 3.3.3 Fase 1: Exploración

En la fase de exploración de la metodología XP, después de haber establecido la arquitectura y las funcionalidades generales del sistema, se procede a la creación de historias de usuario para satisfacer los requerimientos específicos del proyecto.

Historia de usuario	
<b>Clave:</b> HU001.	<b>Usuario:</b> Administrador.
<b>Nombre de la historia:</b> Formulario de ingreso al sistema.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Medio.
<b>Puntos estimados:</b> 4.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Como administrador del sistema, quiero poder crear usuarios y acceder al sistema utilizando un formulario de ingreso o registrarme en el sistema con un formulario de registro.	
<b>Observación:</b>	

Tabla 8. Historia de usuario – Formulario de ingreso al sistema.

Elaborado por: Morales Carlos, 2023.

Historia de usuario	
<b>Clave:</b> HU002.	<b>Usuario:</b> Administrador.

<b>Nombre de la historia:</b> Gestionar perfiles de usuarios.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Medio.
<b>Puntos estimados:</b> 4.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Como administrador del sistema, quiero poder gestionar los perfiles usuarios del sistema con el fin asignar roles y permisos a cada usuario, para que solo puedan acceder a la información y funciones que les corresponden.	
<b>Observación:</b> Los roles de usuario deben incluir administrador, profesor, inspector, secretaria y estudiante.	

Tabla 9. Historia de usuario – Gestionar perfiles de usuarios.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU003.	<b>Usuario:</b> Administrador.
<b>Nombre de la historia:</b> Gestionar áreas.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 2.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Formularios intuitivos y fáciles de usar para gestionar las áreas a las que pertenecerán las asignaturas del sistema.	
<b>Observación:</b>	

Tabla 10. Historia de usuario – Gestionar áreas.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU004.	<b>Usuario:</b> Profesor.
<b>Nombre de la historia:</b> Gestionar asignatura.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 3.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	

<b>Descripción:</b> Formulario intuitivo y fácil de usar para gestionar las asignaturas del sistema, para que los usuarios puedan interactuar.
<b>Observación:</b>

Tabla 11. Historia de usuario – Gestionar asignatura.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU005.	<b>Usuario:</b> Profesor.
<b>Nombre de la historia:</b> Gestionar años lectivos.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 4.	<b>Iteración asignada:</b> 2.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Formulario intuitivo y fácil de usar para gestionar los años lectivos del sistema, para que los usuarios puedan interactuar.	
<b>Observación:</b>	

Tabla 12. Historia de usuario – Gestionar años lectivos.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU006.	<b>Usuario:</b> Profesor.
<b>Nombre de la historia:</b> Gestionar cursos.	
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 3.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Formulario intuitivo y fácil de usar para gestionar los cursos del sistema.	
<b>Observación:</b>	

Tabla 13. Historia de usuario – Diseño de modulo interfaz cursos.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU007.	<b>Usuario:</b> Profesor, secretaria
<b>Nombre de la historia:</b> Gestionar matrícula de estudiantes.	
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 4.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Formulario intuitivo y fácil de usar para gestionar la matrícula de los estudiantes en los diferentes cursos del sistema.	
<b>Observación:</b> Esta funcionalidad debe estar disponible para el uso exclusivo del usuario con el rol de profesor y secretaria para el registro de los estudiantes en los cursos.	

Tabla 14.Historia de usuario – Gestionar matrículas de estudiantes.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU008.	<b>Usuario:</b> Administrador.
<b>Nombre de la historia:</b> Gestionar asignación de asignaturas a profesores.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 3.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Como administrador del sistema, quiero poder gestionar la asignación de asignatura de profesores del sistema, asignar a los docentes sus respectivas asignaturas.	
<b>Observación:</b>	

Tabla 15. Historia de usuario – Gestionar asignación de asignaturas a profesores.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU009.	<b>Usuario:</b> Profesor.
<b>Nombre de la historia:</b> Registrar calificaciones.	

<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Medio.
<b>Puntos estimados:</b> 9.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Como profesor del sistema, quiero registrar las calificaciones de los estudiantes, con el fin de llevar un seguimiento de su desempeño académico por asignatura.	
<b>Observación:</b>	

Tabla 16. Historia de usuario – Registrar calificaciones.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU010.	<b>Usuario:</b> Profesor.
<b>Nombre de la historia:</b> Registrar asistencia.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 5.	<b>Iteración asignada:</b> 2.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Como profesor del sistema, quiero registrar la asistencia de los estudiantes con el fin de llevar un seguimiento de su asistencia a clases la información se organizará por asignatura.	
<b>Observación:</b>	

Tabla 17. Historia de usuario – Registrar asistencia.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU011.	<b>Usuario:</b> Estudiante.
<b>Nombre de la historia:</b> Ver calificaciones actuales de estudiantes.	
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 2.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	

<b>Descripción:</b> Como estudiante del sistema, quiero ver mis calificaciones en cada asignatura, con el fin de conocer mi rendimiento en cada evaluación.
<b>Observación:</b>

Tabla 18. Historia de usuario - Ver calificaciones actuales de estudiantes.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU012.	<b>Usuario:</b> Profesor.
<b>Nombre de la historia:</b> Descargar reporte de calificaciones	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 2.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Formulario para descargar un reporte de calificaciones de mis estudiantes con el fin de tener una copia impresa para reuniones y presentarlas el reporte debe incluir la calificación de cada estudiante en cada asignatura y promedio.	
<b>Observación:</b>	

Tabla 19.Historia de usuario - Descargar reporte de calificaciones.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU013.	<b>Usuario:</b> Profesor, Inspector.
<b>Nombre de la historia:</b> Descargar reporte de asistencia.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 2.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Formulario para descargar un reporte de asistencia de los estudiantes con el fin de tener una copia impresa para reuniones y presentarlas el reporte debe incluir el porcentaje de asistencia por cada estudiante en cada asignatura y promedio.	
<b>Observación:</b>	

Tabla 20. Historia de usuario - Descargas reporte de asistencia.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU014.	<b>Usuario:</b> Profesor, secretaria.
<b>Nombre de la historia:</b> Consultar calificaciones de estudiantes.	
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Formulario fácil de usar para consultar las calificaciones de un estudiante en específico con el respectivo nombre de actividad.	
<b>Observación:</b>	

Tabla 21. Historia de usuario - Consultar calificaciones de estudiante.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU015.	<b>Usuario:</b> Administrador.
<b>Nombre de la historia:</b> Auditoria por cabio o borrado de los registros de calificaciones.	
<b>Prioridad en Negocio:</b> Alto	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 5	<b>Iteración asignada:</b> 1
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Como administrador del sistema, quiero poder auditar los cambios y eliminaciones de los registros de calificaciones de los estudiantes, para asegurar la integridad y la confidencialidad de la información y detectar cualquier posible error o mal uso del sistema.	
<b>Observación:</b>	

Tabla 22. Historia de usuario - Auditoria por cabio o borrado de los registros de calificaciones.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU016.	<b>Usuario:</b> Profesor.
<b>Nombre de la historia:</b> Notificaciones estudiantes con asistencia insuficiente.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 2.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Como profesor del sistema, quiero recibir notificaciones de los estudiantes que tienen asistencia insuficiente en una asignatura, para que puedan tomar medidas para mejorar su asistencia y evitar posibles consecuencias académicas.	
<b>Observación:</b>	

Tabla 23. Historia de usuario - Notificaciones de estudiantes con asistencia insuficiente.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU017.	<b>Usuario:</b> Profesor.
<b>Nombre de la historia:</b> Generar certificado de promoción.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 2.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Como profesor del sistema, quiero poder generar un certificado con el promedio final de todas las asignaturas de un estudiante durante el año lectivo.	
<b>Observación:</b>	

Tabla 24. Historia de usuario - General certificado de promoción.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU018.	<b>Usuario:</b> Profesor.
<b>Nombre de la historia:</b> Generar boleta de calificaciones.	



<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 2.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Como profesor del sistema, quiero poder generar una boleta con las calificaciones de un estudiante determinado con los promedios de los quimestres en un periodo en un año lectivo.	
<b>Observación:</b>	

Tabla 25. Historia de usuario - Generar boleta de calificaciones.

Elaborado por: Morales Carlos, 2023.

<b>Historia de usuario</b>	
<b>Clave:</b> HU019.	<b>Usuario:</b> Profesor, Secretaria, Inspector.
<b>Nombre de la historia:</b> Generar reporte de calificaciones concentrado.	
<b>Prioridad en Negocio:</b> Alto.	<b>Riesgo en Desarrollo:</b> Alto.
<b>Puntos estimados:</b> 3.	<b>Iteración asignada:</b> 1.
<b>Programador Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Como profesor, necesito la funcionalidad de generar un informe que concentre las calificaciones de un curso y paralelo específico, incluyendo los promedios de los quimestres de todas las asignaturas, durante un año lectivo determinado.	
<b>Observación:</b>	

Tabla 26. Historia de usuario - Generar reporte de calificaciones concentrado.

Elaborado por: Morales Carlos, 2023.

### 3.3.4 Fase 2: Valoración de las historias de usuario

En la fase 2 del proyecto, se procede a valorar el tiempo que se requiere para cada historia de usuario. Esto se realiza mediante la estimación del tiempo empleado, trabajando durante 5 horas al día laborable. En esta etapa se realiza la estimación del esfuerzo requerido, el cual se muestra en detalle en la siguiente tabla.

N°	Historia de usuario	Tiempo estimado	
		Días	Horas
1	Formulario de ingreso al sistema	6	30
2	Gestionar perfiles de usuarios.	7	35
3	Gestionar áreas.	4	20
4	Gestionar asignatura.	3	15
5	Gestionar años lectivos.	3	15
6	Gestionar cursos.	5	25
7	Gestionar matrículas de estudiantes.	7	35
8	Gestionar asignación de asignaturas a profesores.	4	20
9	Registrar calificaciones.	11	55
10	Registrar asistencia.	9	45
11	Ver calificaciones actuales de estudiantes.	4	20
12	Descargar reporte de calificaciones.	6	30
13	Descargas reporte de asistencia.	5	25
14	Consultar calificaciones de estudiante.	4	20
15	Auditoria por cambio o borrado de los registros de calificaciones	4	20
16	Notificaciones de estudiantes con asistencia insuficiente.	3	15
17	Generar certificado de promoción.	5	25
18	Generar boleta de calificaciones.	5	25
19	Generar reporte de calificaciones concentrado.	4	20
	Tiempo estimado total	99	495

Tabla 27. Estimación de historias de usuario

Elaborado por: Morales Carlos, 2023.

Con la clara definición de los roles del equipo en un proyecto, se permite una mejor planificación y organización del mismo. Al definir los roles, cada miembro del equipo sabe qué se espera de él y cuáles son sus responsabilidades asignadas, lo que ayuda a evitar confusiones y asegurar que todas las tareas necesarias estén cubiertas. Además,

esto también facilita la comunicación y colaboración entre los miembros del equipo, ya que cada uno sabe qué rol desempeña y cómo su trabajo contribuye al éxito general del proyecto.

Metodología	Rol XP	Responsable	Descripción
Programación extrema (XP).	Usuario.	Cuerpo docente.	Es la apersona encargada de definir los requerimientos y especificaciones del sistema web
	Programador.	Carlos Luis Morales Gutama.	Esta persona es la encargada de asegurarse de que el sistema o la funcionalidad web cumpla con los requerimientos y necesidades del usuario. Asimismo, es responsable de revisar, corregir y realizar pruebas en el código fuente para garantizar su correcto funcionamiento.
	Coach.	Ing. Félix Fernández.	Esta persona es la encargada de supervisar el progreso del proyecto y verificar regularmente si se está cumpliendo con los plazos y objetivos establecidos en el plan de proyecto. Su tarea es asegurarse de que el proyecto avance según lo previsto y tomar medidas para corregir cualquier desviación o retraso que se presente.

	Tester.	Dr. Mirian Andrade.	Persona encargada de realizar las pruebas de aceptación en el sistema es responsable de verificar que el software cumpla con los requisitos y expectativas del usuario final.
--	---------	---------------------	---

Tabla 28. Definición de roles.

Elaborado por: Morales Carlos, 2023.

Siguiendo la metodología XP, las tareas se derivan de las historias de usuario y se detallan con una fecha de inicio y un responsable asignado para que las complete en un tiempo determinado. Esta forma de asignar tareas permite una mejor gestión del tiempo y una mayor eficiencia en el desarrollo del proyecto, ya que cada tarea está claramente definida y se sabe quién es responsable de su ejecución. Además, al tener un plazo definido para completar cada tarea, se promueve la productividad y se evita que el proyecto se extienda innecesariamente.

**Tareas: Formulario de ingreso al sistema.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T001.	<b>Clave de la historia:</b> HU001.
<b>Denominación de la tarea:</b> Diseño del formulario para registro e ingreso al sistema.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 1/11/22	<b>Fecha fin:</b> 8/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.	

Tabla 29. Tarea – Diseño del formulario para registro e ingreso al sistema.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T002.	<b>Clave de la historia:</b> HU001.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario para registro e ingreso al sistema.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 1/11/22	<b>Fecha fin:</b> 8/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario tanto para el registro como para el ingreso de los usuarios y validar los datos de estos formularios.	

Tabla 30. Tarea – Funcionalidad para el formulario para registro e ingreso al sistema.

Elaborado por: Morales Carlos, 2023.

**Tareas: Gestionar perfiles de usuarios.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T003.	<b>Clave de la historia:</b> HU002.
<b>Denominación de la tarea:</b> Diseño de vista de bienvenida para el sistema.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 9/11/22	<b>Fecha fin:</b> 17/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.	

Tabla 31. Tarea – Diseño de interfaz del sistema.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T004.	<b>Clave de la historia:</b> HU002
<b>Denominación de la tarea:</b> Diseño del formulario roles y permisos.	

<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 9/11/22	<b>Fecha fin:</b> 17/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.	

Tabla 32. Tarea – Diseño del formulario roles y permisos.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T005.	<b>Clave de la historia:</b> HU002
<b>Denominación de la tarea:</b> Funcionalidad para el formulario roles y permisos.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 9/11/22	<b>Fecha fin:</b> 17/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario para otorgar un rol y agregar o quitar los permisos a los usuarios que están en el sistema para proteger acciones a los diferentes usuarios.	

Tabla 33. Tarea – Funcionalidad para el formulario roles y permisos.

Elaborado por: Morales Carlos, 2023.

**Tareas: Gestionar áreas.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T006.	<b>Clave de la historia:</b> HU004.
<b>Denominación de la tarea:</b> Diseño del formulario áreas.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 18/11/22	<b>Fecha fin:</b> 23/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	

**Descripción:** Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.

Tabla 34. Tarea – Diseño del formulario asistencia.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T007.	<b>Clave de la historia:</b> HU004.
<b>Denominación de la tarea:</b> Listar áreas.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 18/11/22	<b>Fecha fin:</b> 23/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se listar todas las áreas a utilizarse en el sistema con sus nombres y descripción y un buscador con Livewire para facilitar la búsqueda.	

Tabla 35. Tarea – Listar áreas.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T038.	<b>Clave de la historia:</b> HU005.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario áreas.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 18/11/22	<b>Fecha fin:</b> 23/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario áreas para crear, editar y validar los datos correspondientes que se ingresan posterior ingresar las calificaciones de este taller.	

Tabla 36. Tarea – Funcionalidad para el formulario áreas.

Elaborado por: Morales Carlos, 2023.

**Tareas: Gestionar asignaturas.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T008.	<b>Clave de la historia:</b> HU006.
<b>Denominación de la tarea:</b> Diseño del formulario asignaturas.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 24/11/22	<b>Fecha fin:</b> 28/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.	

Tabla 37. Tarea – Diseño del formulario asistencia.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T009.	<b>Clave de la historia:</b> HU006.
<b>Denominación de la tarea:</b> Listar asignaturas.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 24/11/22	<b>Fecha fin:</b> 28/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se listar todas las asignaturas a utilizarse en el sistema con sus nombres y descripción.	

Tabla 38. Tarea – Listar asignaturas.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T010.	<b>Clave de la historia:</b> HU006.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario asignaturas.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.



<b>Fecha inicio:</b> 24/11/22	<b>Fecha fin:</b> 28/11/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario asignaturas para crear, editar y validar los datos correspondientes que se ingresan posterior ingresar las calificaciones de este taller.	

Tabla 39. Tarea – Funcionalidad para el formulario asignaturas.

Elaborado por: Morales Carlos, 2023.

**Tareas: Gestionar años lectivos.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T011.	<b>Clave de la historia:</b> HU007.
<b>Denominación de la tarea:</b> Diseño del formulario años lectivos.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 29/11/22	<b>Fecha fin:</b> 1/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.	

Tabla 40. Tarea – Diseño del formulario años lectivos.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T012.	<b>Clave de la historia:</b> HU007.
<b>Denominación de la tarea:</b> Listar años lectivos.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 29/11/22	<b>Fecha fin:</b> 1/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	

**Descripción:** Se listar todos los años lectivos a utilizarse en el sistema con sus nombres, descripción y un buscador desarrollado con Livewire.

Tabla 41. Tarea – Listar años lectivos.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T013.	<b>Clave de la historia:</b> HU007.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario años lectivos.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 29/11/22	<b>Fecha fin:</b> 1/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario años lectivos para crear, editar y validar los datos correspondientes que se ingresan.	

Tabla 42. Tarea – Funcionalidad para el formulario años lectivos.

Elaborado por: Morales Carlos, 2023.

**Tareas: Gestionar cursos.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T014.	<b>Clave de la historia:</b> HU008.
<b>Denominación de la tarea:</b> Diseño del formulario cursos.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 2/12/22	<b>Fecha fin:</b> 8/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.	

Tabla 43. Tarea – Diseño del formulario cursos.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T015.	<b>Clave de la historia:</b> HU008.
<b>Denominación de la tarea:</b> Listar cursos.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 2/12/22	<b>Fecha fin:</b> 8/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se listar todos los cursos a utilizarse en el sistema con sus nombres y descripción con un buscador empleando Livewire para una mejorar búsqueda.	

Tabla 44. Tarea – Listar cursos.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T016.	<b>Clave de la historia:</b> HU008.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario cursos.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 2/12/22	<b>Fecha fin:</b> 8/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario cursos para crear, editar y validar los datos correspondientes que se ingresan.	

Tabla 45. Tarea – Funcionalidad para el formulario cursos.

Elaborado por: Morales Carlos, 2023.

**Tareas: Gestionar matrículas de estudiantes.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T017.	<b>Clave de la historia:</b> HU009.
<b>Denominación de la tarea:</b> Diseño del formulario matrículas de estudiantes.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 9/12/22	<b>Fecha fin:</b> 19/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	

**Descripción:** Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.

Tabla 46. Tarea – Diseño del formulario matrículas de estudiantes.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T018.	<b>Clave de la historia:</b> HU009.
<b>Denominación de la tarea:</b> Listar matrículas de estudiantes.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 9/12/22	<b>Fecha fin:</b> 19/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se listar todas las matrículas de estudiantes con su respectivo curso en el sistema con sus nombres y descripción con un buscador empleando Livewire para una mejorar búsqueda.	

Tabla 47. Tarea – Listar matrículas de estudiantes.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T019.	<b>Clave de la historia:</b> HU009.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario matrículas de estudiantes.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 9/12/22	<b>Fecha fin:</b> 19/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario matrículas de estudiantes para crear, editar y validar los datos correspondientes.	

Tabla 48. Tarea – Funcionalidad para el formulario matrículas de estudiantes.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T020.	<b>Clave de la historia:</b> HU009.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario importar Excel.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 9/12/22	<b>Fecha fin:</b> 19/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario importar archivo Excel con un listado de matrículas de estudiantes validar los datos correspondientes que se ingresan.	

Tabla 49. Tarea – Funcionalidad para el formulario importar Excel.

Elaborado por: Morales Carlos, 2023.

**Tareas: Gestionar asignación de asignaturas a profesores.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T021.	<b>Clave de la historia:</b> HU010.
<b>Denominación de la tarea:</b> Diseño de formulario asignación de asignaturas a profesores.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 20/12/22	<b>Fecha fin:</b> 23/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.	

Tabla 50. Tarea – Diseño de formulario asignación de asignaturas a profesores.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T022.	<b>Clave de la historia:</b> HU010.

<b>Denominación de la tarea:</b> Listar asignaturas a profesores.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 20/12/22	<b>Fecha fin:</b> 23/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se listar todas las asignaturas a su respectivo docente con su respectivo curso en el sistema con un buscador empleando Livewire para una mejorar búsqueda.	

Tabla 51. Tarea – Listar asignaturas a profesores.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T023.	<b>Clave de la historia:</b> HU010.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario asignación de asignaturas a profesores.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 20/12/22	<b>Fecha fin:</b> 23/12/22
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario asignar las asignaturas que impartirán en un curso los docentes de la Unidad Educativa.	

Tabla 52. Tarea – Funcionalidad para el formulario asignación de asignaturas a profesores.

Elaborado por: Morales Carlos, 2023.

**Tareas: Registrar calificaciones.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T024.	<b>Clave de la historia:</b> HU011.
<b>Denominación de la tarea:</b> Diseño del formulario insumo.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 26/12/22	<b>Fecha fin:</b> 9/1/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	

**Descripción:** Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.

Tabla 53. Tarea – Diseño del formulario talleres.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T025.	<b>Clave de la historia:</b> HU011.
<b>Denominación de la tarea:</b> Listar los insumos.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 26/12/22	<b>Fecha fin:</b> 9/1/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se listar todas las tares con sus nombres descripción y fecha.	

Tabla 54. Tarea – Listar talleres.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T026.	<b>Clave de la historia:</b> HU003
<b>Denominación de la tarea:</b> Funcionalidad para el formulario insumos.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 26/12/22	<b>Fecha fin:</b> 9/1/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario talleres para crear, editar y validar los datos correspondientes que se ingresan posterior ingresar las calificaciones de este taller.	

Tabla 55. Tarea – Funcionalidad para el formulario talleres.

Elaborado por: Morales Carlos, 2023.

**Tareas: Registrar asistencia.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T027.	<b>Clave de la historia:</b> HU004.
<b>Denominación de la tarea:</b> Diseño del formulario asistencia.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 10/1/23	<b>Fecha fin:</b> 20/1/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.	

Tabla 56. Tarea – Diseño del formulario asistencia.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T028.	<b>Clave de la historia:</b> HU004.
<b>Denominación de la tarea:</b> Listar asistencia.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 10/1/23	<b>Fecha fin:</b> 20/1/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se listar todas las asistencias con sus nombres descripción y fecha.	

Tabla 57. Tarea – Listar asistencia.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T029.	<b>Clave de la historia:</b> HU004.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario asistencia.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.



<b>Fecha inicio:</b> 10/1/23	<b>Fecha fin:</b> 20/1/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se da funcionalidad al formulario asistencia para crear, editar y validar los datos correspondientes que se ingresan posterior ingresar las calificaciones de este taller.	

Tabla 58. Tarea – Funcionalidad para el formulario asistencia.

Elaborado por: Morales Carlos, 2023.

**Tareas: Ver calificaciones actuales de estudiantes.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T030.	<b>Clave de la historia:</b> HU011.
<b>Denominación de la tarea:</b> Diseño de formulario ver calificaciones.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 23/1/23	<b>Fecha fin:</b> 26/1/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se empleo Bootstrap 5 para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.	

Tabla 59. Tarea – Diseño de formulario ver calificaciones.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T031.	<b>Clave de la historia:</b> HU011.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario ver calificaciones.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 23/1/23	<b>Fecha fin:</b> 26/1/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	

**Descripción:** Se ha implementado una nueva funcionalidad en el formulario para que los estudiantes puedan visualizar con detalle sus calificaciones por asignatura, incluyendo el nombre del insumo, la fecha y la calificación obtenida en la Unidad Educativa. Esto permitirá a los estudiantes tener una mejor comprensión de su desempeño académico y facilitará su seguimiento y mejora en el futuro.

Tabla 60. Tarea – Funcionalidad para el formulario ver calificaciones.

Elaborado por: Morales Carlos, 2023.

**Tareas: Descargar reporte de calificaciones.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T032.	<b>Clave de la historia:</b> HU012.
<b>Denominación de la tarea:</b> Diseño informe calificaciones por asignatura.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 27/1/23	<b>Fecha fin:</b> 3/2/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se empleo Bootstrap para el diseño de los formularios que permiten visualizar las calificaciones de un estudiante en particular, por asignatura en el sistema, debido a su sistema de rejillas altamente adaptable a cualquier dispositivo, lo que proporciona una experiencia de usuario óptima y consistente en diferentes dispositivos.	

Tabla 61. Tarea – Diseño informe calificaciones por asignatura.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T033.	<b>Clave de la historia:</b> HU012.
<b>Denominación de la tarea:</b> Descargar reporte calificaciones por asignatura.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 27/1/23	<b>Fecha fin:</b> 3/2/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	

**Descripción:** se presentar al usuario un informe de las calificaciones de todos los estudiantes con la información requerida por el docente con una versión resumida y extendida con los promedios de cada insumo.

Tabla 62. Tarea – Funcionalidad para el formulario ver calificaciones.

Elaborado por: Morales Carlos, 2023.

**Tareas: Descargas reporte de asistencia.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T034.	<b>Clave de la historia:</b> HU013.
<b>Denominación de la tarea:</b> Diseño reporte de asistencia.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 6/2/23	<b>Fecha fin:</b> 10/2/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se utiliza Bootstrap para el diseño de los formularios para descargar en un documento PDF el reporte de la asistencia de los estudiantes de una específica asignatura con la información requerida por el docente del sistema.	

Tabla 63. Tarea – Diseño informe reportes de asistencia.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T035.	<b>Clave de la historia:</b> HU013.
<b>Denominación de la tarea:</b> Descargar reporte de asistencia.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 6/2/23	<b>Fecha fin:</b> 10/2/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se presentar al usuario un informe de la asistencia de todos los estudiantes por asignatura con la información requerida por el docente en un documento en formato PDF o Excel.	

Tabla 64. Tarea – Funcionalidad para el formulario ver calificaciones.

Elaborado por: Morales Carlos, 2023.

**Tareas: Consultar calificaciones de estudiante.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T036.	<b>Clave de la historia:</b> HU014.
<b>Denominación de la tarea:</b> Diseño de formulario consultar calificaciones de estudiantes.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 13/2/23	<b>Fecha fin:</b> 16/2/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se utiliza Bootstrap para el diseño de los formularios para consultar las calificaciones de los estudiantes de una específica asignatura con la información requerida por el docente del sistema porque este un tiene un poderoso sistema de rejillas adaptable al tamaño de cualquier dispositivo.	

Tabla 65. Tarea – Diseño de formulario consultar calificaciones de estudiantes.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T037.	<b>Clave de la historia:</b> HU014.
<b>Denominación de la tarea:</b> Funcionalidad de formulario consultar calificaciones de estudiantes.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 13/2/23	<b>Fecha fin:</b> 16/2/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se presentar al usuario un informe de la asistencia de todos los estudiantes por asignatura con la información requerida por el docente.	

Tabla 66. Tarea – Funcionalidad para el formulario ver calificaciones.

Elaborado por: Morales Carlos, 2023.

**Tareas: Auditoria por cabio o borrado de los registros de calificaciones.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T038.	<b>Clave de la historia:</b> HU015.

<b>Denominación de la tarea:</b> Implementar auditoría de cambios en registros de calificaciones en los insumos.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 17/2/23	<b>Fecha fin:</b> 22/2/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se desarrollará una función que permita capturar los datos al momento de cambiar o eliminar un registro en las calificaciones de los estudiantes.	

Tabla 67. Tarea – Implementar auditoría de cambios en registros de calificaciones en los insumos.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T039.	<b>Clave de la historia:</b> HU015.
<b>Denominación de la tarea:</b> Diseño de interfaz de auditoría de cambios.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 17/2/23	<b>Fecha fin:</b> 22/2/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se una interfaz intuitiva i fácil de usar para que el usuario observe los cambios que se realizaron en las calificaciones de los insumos. El listado debe contener información detallada del usuario que realizo el cambio, la dirección ip, la calificación antigua y la calificación nueva.	

Tabla 68. Tarea – Diseño de interfaz de auditoría de cambios.

Elaborado por: Morales Carlos, 2023.

**Tareas: Notificaciones de estudiantes con asistencia insuficiente.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T040.	<b>Clave de la historia:</b> HU016.
<b>Denominación de la tarea:</b> Implementar notificaciones de asistencia insuficiente.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 23/2/23	<b>Fecha fin:</b> 27/2/23

<b>Responsable:</b> Carlos Luis Morales Gutama.
<b>Descripción:</b> Se desarrolla una funcionalidad que permita la detección de estudiantes con un porcentaje de asistencia insuficiente y envío de notificaciones al docente correspondiente.

Tabla 69. Tarea – Implementar notificaciones de asistencia insuficiente.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T041.	<b>Clave de la historia:</b> HU016.
<b>Denominación de la tarea:</b> Diseño de informe de notificaciones de asistencia insuficiente.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 23/2/23	<b>Fecha fin:</b> 27/2/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se creará un diseño de informe para mostrar la información de las notificaciones de asistencia insuficiente generadas por el sistema. El informe debe ser fácil de leer, identificando claramente la información requerida.	

Tabla 70. Tarea – Diseño de informe de notificaciones de asistencia insuficiente

Elaborado por: Morales Carlos, 2023.

### **Tareas: Generar certificado de promoción.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T042.	<b>Clave de la historia:</b> HU017.
<b>Denominación de la tarea:</b> Diseño de formulario generación de certificados de promoción	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 28/2/23	<b>Fecha fin:</b> 6/3/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	

**Descripción:** Se utiliza Bootstrap para el diseño de los formularios para descargar en un documento PDF el reporte de la asistencia de los estudiantes de una específica asignatura con la información requerida por el docente del sistema.

Tabla 71. Tarea – Diseño de informe de notificaciones de asistencia insuficiente.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T043.	<b>Clave de la historia:</b> HU017.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario generación de certificados de promoción.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 28/2/23	<b>Fecha fin:</b> 6/3/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se desarrollará una funcionalidad que permita la generación automática de certificados de promoción para los estudiantes en documentos en formato PDF.	

Tabla 72. Tarea – Funcionalidad para el formulario generación de certificados de promoción.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T044.	<b>Clave de la historia:</b> HU017.
<b>Denominación de la tarea:</b> Diseño del certificado de promoción.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 28/2/23	<b>Fecha fin:</b> 6/3/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Crear un diseño de certificado de promoción que incluya el nombre del estudiante, el curso y la fecha de promoción y el promedio de cada asignatura. El certificado debe ser fácil de leer y tener un diseño atractivo y profesional.	

Tabla 73. Tarea – Diseño del certificado de promoción.

Elaborado por: Morales Carlos, 2023.

**Tareas: Generar boleta de calificaciones.**

<b>Tarea</b>	
<b>Clave de la historia:</b> T045.	<b>Clave de la historia:</b> HU018.
<b>Denominación de la tarea:</b> Diseño de formulario generación boleta de calificaciones.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 3.
<b>Fecha inicio:</b> 7/3/23	<b>Fecha fin:</b> 13/3/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se utiliza Bootstrap para el diseño del formulario que permiten generar una boleta de calificaciones de los estudiantes, ya que este framework cuenta con un sistema de rejillas adaptable al tamaño de cualquier dispositivo.	

Tabla 74. Tarea – Diseño de formulario generación boleta de calificaciones.

Elaborado por: Morales Carlos, 2023.

<b>Tarea</b>	
<b>Clave de la historia:</b> T046.	<b>Clave de la historia:</b> HU018.
<b>Denominación de la tarea:</b> Funcionalidad para el formulario generación boleta de calificaciones.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 5.
<b>Fecha inicio:</b> 7/3/23	<b>Fecha fin:</b> 13/3/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Se desarrollará una funcionalidad que permita la generación automática una boleta de calificaciones de cada estudiante en documentos en formato PDF.	

Tabla 75. Tarea – Funcionalidad para el formulario generación boleta de calificaciones.

Elaborado por: Morales Carlos, 2023.



<b>Tarea</b>	
<b>Clave de la historia:</b> T047.	<b>Clave de la historia:</b> HU018.
<b>Denominación de la tarea:</b> Diseño de boletas de calificaciones.	
<b>Tipo de tarea:</b> Desarrollador.	<b>Puntos estimados:</b> 4.
<b>Fecha inicio:</b> 7/3/23	<b>Fecha fin:</b> 13/3/23
<b>Responsable:</b> Carlos Luis Morales Gutama.	
<b>Descripción:</b> Crear un diseño de certificado de promoción que incluya el nombre del estudiante, el curso y el promedio de cada asignatura. La boleta debe ser fácil de leer y tener un diseño atractivo y profesional.	

Tabla 76. Tarea – Diseño del certificado de promoción.

Elaborado por: Morales Carlos, 2023.

### 3.3.5 Fase 3: Iteraciones

Después de la creación y evaluación de las historias de usuario, en la metodología XP, el siguiente paso es llevar a cabo las iteraciones. En cada iteración, se planifica, diseña, codifica y prueba cada historia de usuario seleccionada. Después de cada iteración, se revisa el progreso del proyecto y se determina si es necesario hacer ajustes en la planificación para la siguiente iteración. Este proceso se repite hasta que todas las historias de usuario estén completas y se alcance el objetivo del proyecto. De esta manera, se logra un enfoque iterativo e incremental que permite un desarrollo flexible y una respuesta efectiva a los cambios en los requisitos del proyecto.

Las historias de usuario corresponden a la siguiente iteración.

N° iteración	Historia de usuario	Tiempo estimado	
		Días	Horas
1	Formulario de ingreso al sistema	6	30
	Crear y gestionar perfiles de usuarios.	7	35
	Crear y gestionar áreas.	4	20

	Crear y gestionar asignatura.	3	15
--	-------------------------------	---	----

Tabla 77. Historias de usuario iteración 1.

Elaborado por: Morales Carlos, 2023.

## Tarjetas CRC

Las tarjetas CRC son un conjunto de tarjetas estándar. Cada tarjeta representa una clase específica y muestra las responsabilidades que esa clase tiene, así como los colaboradores que interactuarán con ella.

<b>Ingreso a la aplicación</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
El ingreso a la aplicación se realiza mediante las credenciales de usuario, seguido de la verificación de información en la base de datos.	Autenticación. Capa de acceso de datos.
<b>Observaciones:</b>	

Tabla 78. Tarjeta CRC ingreso a la aplicación.

Elaborado por: Morales Carlos, 2023.

<b>Gestionar roles</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar y gestionar los roles y permisos de los usuarios del sistema.	Autenticación Capa de acceso a datos
<b>Observaciones:</b>	

Tabla 79. Tarjeta CRC gestionar roles.

Elaborado por: Morales Carlos, 2023.

<b>Gestionar áreas</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar y gestionar las áreas que tiene la unidad educativa.	Capa de acceso a datos. CRUD áreas
<b>Observaciones:</b> Se involucra la tabla área.	

Tabla 80. Tarjeta CRC gestionar áreas.

Elaborado por: Morales Carlos, 2023.

<b>Gestionar asignaturas</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar y gestionar las asignaturas que tiene la unidad educativa.	Capa de acceso a datos. CURD asignaturas.
<b>Observaciones:</b> Se involucra la tabla área y asignaturas.	

Tabla 81. Tarjeta CRC gestionar asignaturas.

Elaborado por: Morales Carlos, 2023.

<b>Gestionar años lectivos</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar y gestionar los años lectivos que tendrá la unidad educativa.	Capa de acceso a datos. CURD años lectivos.
<b>Observaciones:</b> Se involucra la tabla años_lectivos.	

Tabla 82. Tarjeta CRC gestionar años lectivos.

Elaborado por: Morales Carlos, 2023.

<b>Gestionar cursos</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar y gestionar los cursos que tendrá la unidad educativa.	Capa de acceso a datos. CURD cursos.
<b>Observaciones:</b> Se involucra la tabla cursos.	

Tabla 83. Tarjeta CRC gestionar cursos.

Elaborado por: Morales Carlos, 2023.

<b>Gestionar matricula estudiantes</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar y gestionar las matrículas de los estudiantes en los cursos de la unidad educativa.	Capa de acceso a datos. CURD matriculas.
<b>Observaciones:</b> Se involucra las tablas estudiantes, cursos.	

Tabla 84 Tarjeta CRC matriculas estudiantes.

Elaborado por: Morales Carlos, 2023.

<b>Gestionar asignaturas de profesores</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar y gestionar asignaturas que los docentes imparte un curso dentro de la unidad educativa.	Capa de acceso a datos. CURD asignación de asignaturas.
<b>Observaciones:</b> Se involucra las tablas users, cursos, asignaturas.	

Tabla 85 Tarjetas CRC asignaturas de profesores.

Elaborado por: Morales Carlos, 2023.

<b>Registrar calificaciones</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar y gestionar el registro de las calificaciones de los estudiantes por parte de los profesores.	Capa de acceso a datos. CURD registro calificaciones.
<b>Observaciones:</b> Se involucra las tablas estudiantes, insumos, detalle insumos.	

Tabla 86 Tarjeta CRC registro de calificaciones.

Elaborado por: Morales Carlos, 2023.

<b>Registrar asistencia</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar y gestionar el registro de la asistencia de los estudiantes por parte de los profesores.	Capa de acceso a datos. CURD registro asistencia.
<b>Observaciones:</b> Se involucra las tablas estudiantes, horarios, asistencia.	

Tabla 87. Tarjeta CRC registro de asistencia.

Elaborado por: Morales Carlos, 2023.

<b>Consultar calificaciones</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar las calificaciones de cada insumo calificado por cada asignatura	Capa de acceso a datos.
<b>Observaciones:</b> Se involucra las tablas estudiantes, las tablas de insumos y sus respectivos detalles.	

Tabla 88 Tarjeta CRC consultar calificaciones

Elaborado por: Morales Carlos, 2023.

<b>Generar reportes</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Genera reportes de calificaciones, asistencia, boletas de calificaciones y certificados de promoción.	Capa de acceso a datos.
<b>Observaciones:</b> Se involucra las tablas estudiantes, las tablas de insumos y sus respectivos detalles, horarios, asistencia, cursos, asignaturas y áreas.	

Tabla 89 Tarjeta CRC generar reportes.

Elaborado por: Morales Carlos, 2023.

<b>Notificaciones</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Visualizar y eliminar las notificaciones de los estudiantes con un porcentaje de inasistencia superior al 10%.	Capa de acceso a datos.
<b>Observaciones:</b> Se involucra las tablas horarios y asistencia.	

Tabla 90 Tarjeta CRC notificaciones.

Elaborado por: Morales Carlos, 2023.

## **Codificación**

### **HU001: Formulario de registro e ingreso al sistema**

Se instalan los complementos y herramientas necesarios para el desarrollo del sistema web. Se inicia el proyecto utilizando Laragon y luego se procede a conectar la aplicación con la base de datos MySQL.

### **Diseño de base de datos.**

El diseño de la base de datos se realizó según la información proporcionada por los profesores de la unidad educativa Huachi Grande, con el fin de cumplir con todos sus requerimientos y necesidades.

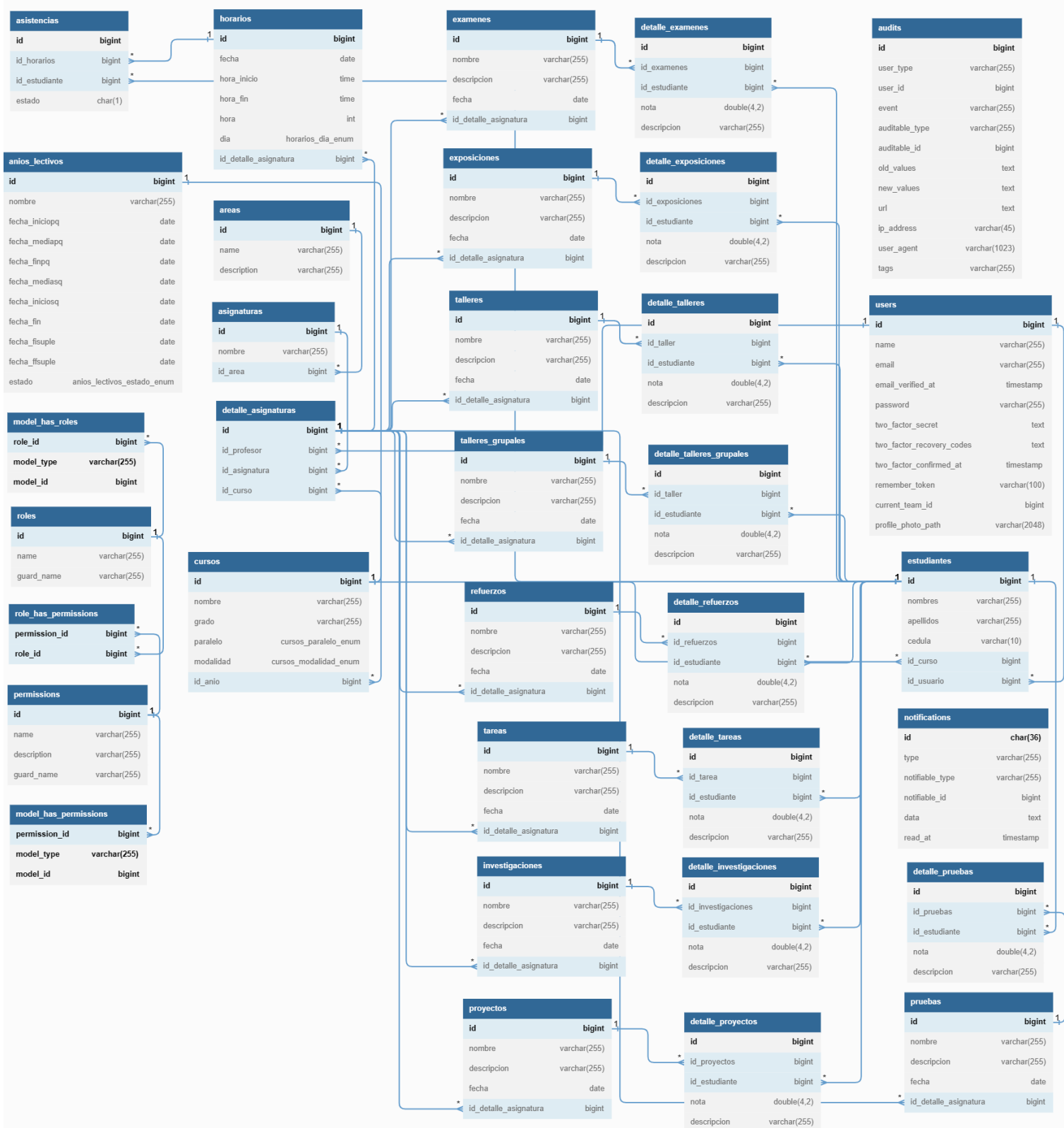
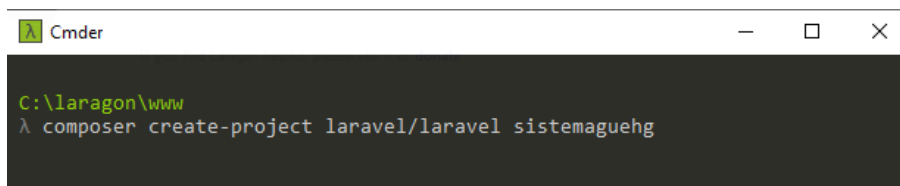


Figura 6. Diseño de base de datos.

Elaborado por: Morales Carlos, 2023.



Para crear un proyecto en Laravel se abre la terminal de Laragon se ingresa el comando **composer create-project laravel/laravel (nombre proyecto)** esperamos que composer descargue todas las dependencias necesarias para el proyecto.



```
C:\laragon\www
λ composer create-project laravel/laravel sistemaguehg
```

Figura 7. Creación del proyecto en terminal Laragon.

Elaborado por: Morales Carlos, 2023.



Figura 8. Página de inicio del sistema para la Unidad Educativa.

Elaborado por: Morales Carlos, 2023.

El diseño se realiza utilizando Bootstrap 5, que incluirá una página de inicio con el logo de la unidad educativa, permitiendo que los usuarios puedan registrarse e iniciar sesión en el sistema.

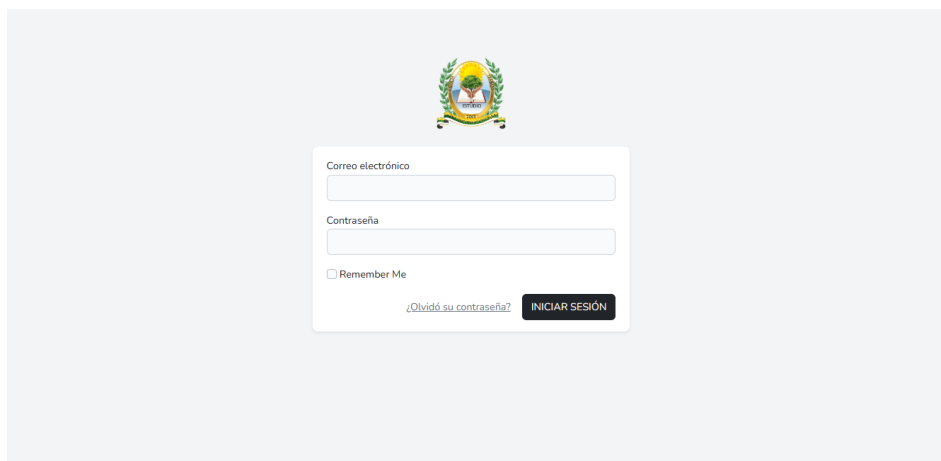


Figura 9. Formulario de ingresos al sistema.

Elaborado por: Morales Carlos, 2023.

El diseño se realiza utilizando Bootstrap 5 e incluirá un formulario para que los usuarios puedan ingresar al sistema utilizando su correo electrónico y contraseña.

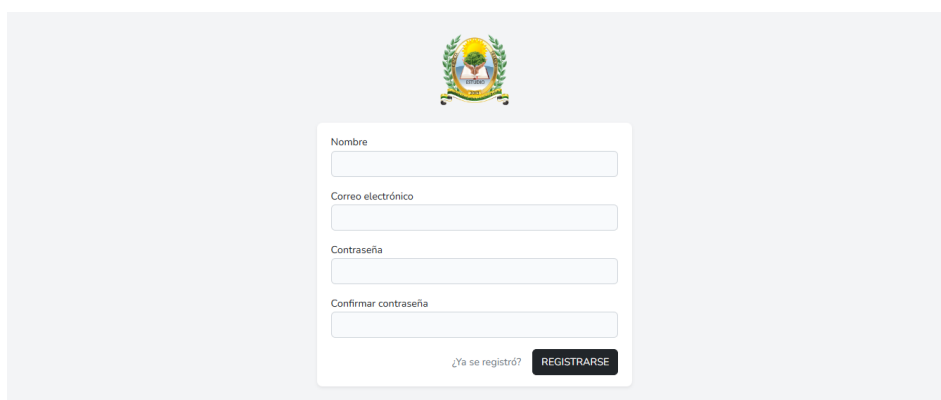


Figura 10. Formulario de registro al sistema.

Elaborado por: Morales Carlos, 2023.

El diseño se realiza utilizando Bootstrap 5 e incluirá un formulario para que los usuarios puedan registrarse en el sistema utilizando nombre, su correo electrónico y contraseña que deberá ser confirmada para el registro.

## HU002: Crear y gestionar perfiles de usuarios.

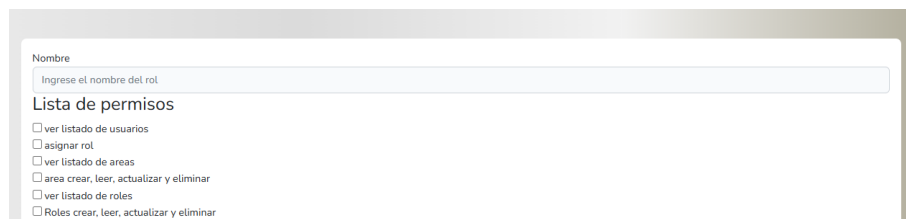


ID	Role		
1	Admin	Editar	Eliminar
2	Profesor	Editar	Eliminar

Figura 11. Listar roles del sistema.

Elaborado por: Morales Carlos, 2023.

Se diseñó una interfaz utilizando Bootstrap que incluye una tabla que muestra los roles existentes en el sistema, cada uno con su respectivo botón de edición y eliminación.



Nombre  
Ingrese el nombre del rol

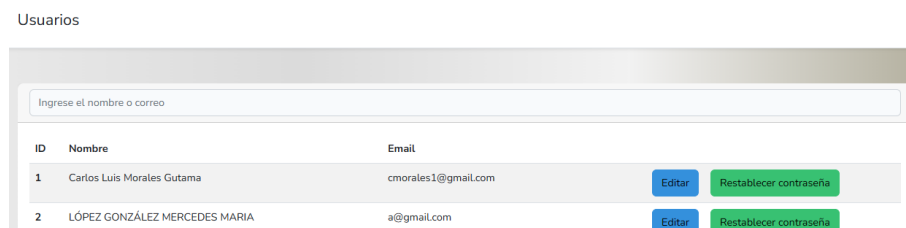
Lista de permisos

- ver listado de usuarios
- asignar rol
- ver listado de areas
- area crear, leer, actualizar y eliminar
- ver listado de roles
- Roles crear, leer, actualizar y eliminar

Figura 12. Formulario administrar rol y otorgar permisos.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 para definir un rol y seleccionar una lista de permisos que se le otorgarán al mismo.

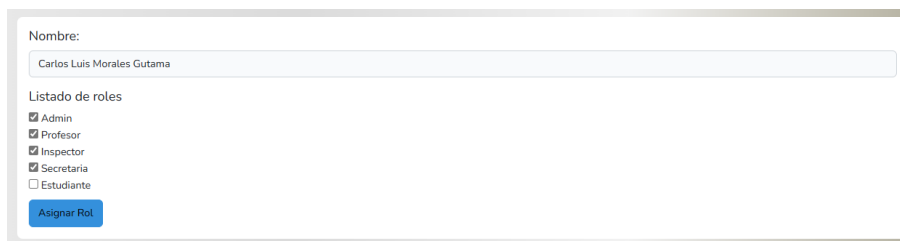


ID	Nombre	Email		
1	Carlos Luis Morales Gutama	cmorales1@gmail.com	Editar	Restablecer contraseña
2	LÓPEZ GONZÁLEZ MERCEDES MARIA	a@gmail.com	Editar	Restablecer contraseña

Figura 13. Lista de usuarios.

Elaborado por: Morales Carlos, 2023.

Se creo una tabla con Bootstrap 5 con la lista de usuarios con un buscador por nombre y correo con un botón para editar y asignar roles al usuario.



Nombre:  
Carlos Luis Morales Gutama

Listado de roles

- Admin
- Profesor
- Inspector
- Secretaria
- Estudiante

Asignar Rol

Figura 14. Formulario para asignar roles y permisos.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 en el que se incluyen los nombres de los usuarios y una lista de casillas de verificación que representan los roles disponibles en el sistema. De esta manera, se pueden asignar los roles correspondientes al usuario seleccionado.

### HU003: Gestionar áreas.



Áreas Nuevo Área

Ingrese el nombre del área

ID	Nombre	Descripción		
1	LENGUA Y LITERATURA		Editar	Eliminar
2	EDUCACIÓN CULTURA Y ARTÍSTICA		Editar	Eliminar

Figura 15. Lista de áreas.

Elaborado por: Morales Carlos, 2023.

Se ha creado una tabla utilizando Bootstrap 5 que contiene las áreas del sistema y sus respectivas descripciones. Además, se ha incluido un botón de "Editar" y otro de "Eliminar" para cada registro, lo que permite modificar o eliminar los datos de cada

área. También se ha agregado un buscador para facilitar la búsqueda de registros en la tabla.

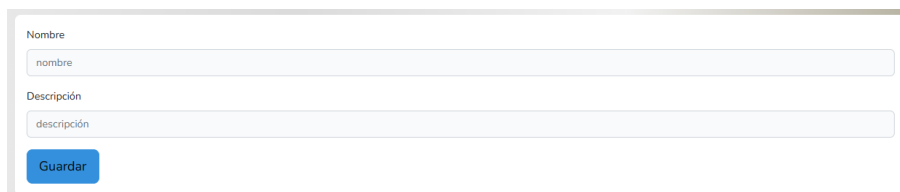
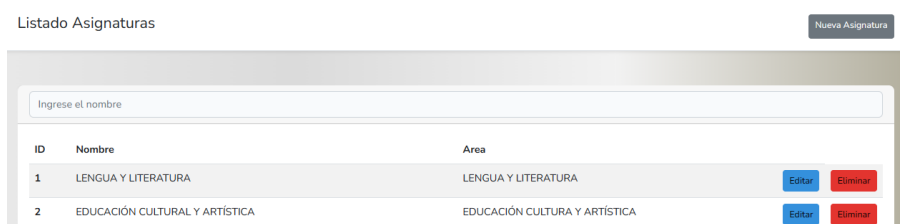


Figura 16. Formulario administrar áreas.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 que incluye dos campos importantes: "Nombre" y "Descripción". Este formulario se utiliza para crear las áreas que se utilizan en el sistema. De esta manera, se pueden agregar nuevas áreas al sistema y proporcionar una descripción para cada una de ellas.

#### **HU004: Gestionar asignatura.**



ID	Nombre	Area		
1	LENGUA Y LITERATURA	LENGUA Y LITERATURA	Editar	Eliminar
2	EDUCACIÓN CULTURAL Y ARTÍSTICA	EDUCACIÓN CULTURA Y ARTÍSTICA	Editar	Eliminar

Figura 17. Lista de asignaturas.

Elaborado por: Morales Carlos, 2023.

Se ha creado una tabla utilizando Bootstrap 5 que contiene las asignaturas del sistema y su respectiva área a la que pertenece. Además, se ha incluido un botón de "Editar" y otro de "Eliminar" para cada registro, lo que permite modificar o eliminar los datos de

cada área. También se ha agregado un buscador para facilitar la búsqueda de registros en la tabla.

Figura 18. Formulario asignatura.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 que incluye dos campos importantes: "Nombre" y "Área". Este formulario se utiliza para crear las áreas que se utilizan en el sistema. De esta manera, se pueden agregar nuevas asignaturas al sistema y una lista para seleccionar el área a la pertenece la asignatura.

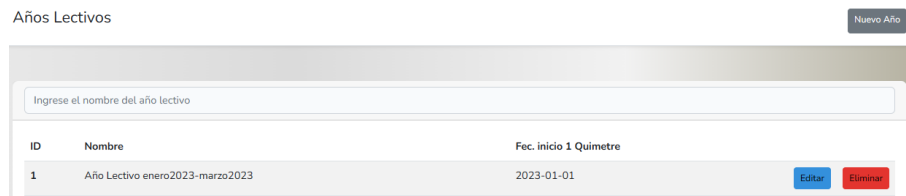
Las historias de usuario corresponden a la siguiente iteración.

N° iteración	Historia de usuario	Tiempo estimado	
		Días	Horas
2	Gestionar años lectivos.	3	15
	Gestionar cursos.	5	25
	Gestionar matrículas de estudiantes.	7	35
	Gestionar asignación de asignaturas a profesores.	4	20

Tabla 91. Historias de usuario iteración 2.

Elaborado por: Morales Carlos, 2023.

## HU005: Gestionar años lectivos.



ID	Nombre	Fec. inicio 1 Quimetre		
1	Año Lectivo enero2023-marzo2023	2023-01-01	Editar	Eliminar

Figura 19. Lista de años lectivos.

Elaborado por: Morales Carlos, 2023.

Se ha creado una tabla utilizando Bootstrap 5 que contiene los años lectivos del sistema y sus respectivas descripciones. Además, se ha incluido un botón de "Editar" y otro de "Eliminar" para cada registro, lo que permite modificar o eliminar los datos de cada área. También se ha agregado un buscador para facilitar la búsqueda de registros en la tabla.

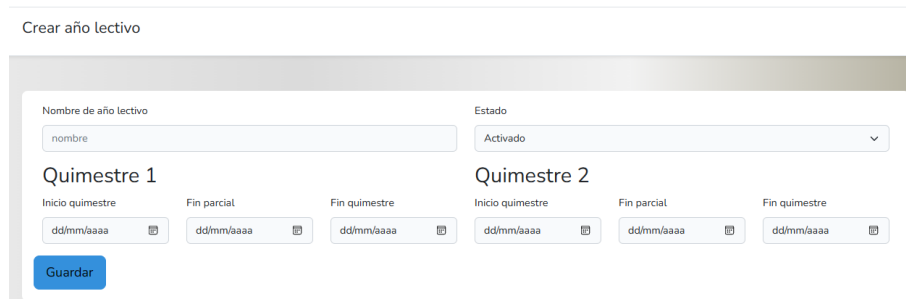


Figura 20. Formulario para administrar los años lectivos.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 que incluye campos importantes: "Nombre" y las fechas que delimitan los quimestres. Este formulario se utiliza para administrar los años lectivos que se utilizan en el sistema.

## HU006: Gestionar cursos.



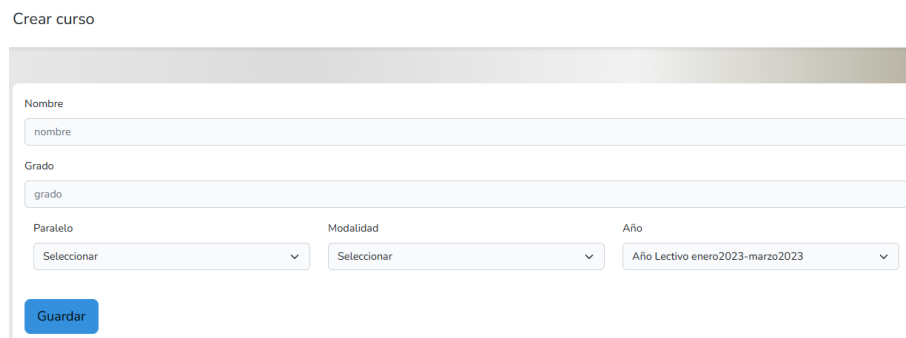
The screenshot shows a web interface for managing courses. At the top right, there is a button labeled "Nuevo Curso". Below it is a search bar with the placeholder text "Ingrese el nombre o grado". The main content is a table with the following columns: ID, Nombre, Grado, Paralelo, and Modalidad. There are two rows of data. Each row has "Editar" and "Eliminar" buttons to its right.

ID	Nombre	Grado	Paralelo	Modalidad		
1	curso 1	7mo de básica	B	MATUTINA	Editar	Eliminar
2	curso 2	6to de básica	C	MATUTINA	Editar	Eliminar

Figura 21. Lista de cursos.

Elaborado por: Morales Carlos, 2023.

Se ha creado una tabla utilizando Bootstrap 5 que contiene los cursos del sistema y con su respectiva información necesaria. Además, se ha incluido un botón de "Editar" y otro de "Eliminar" para cada registro, lo que permite modificar o eliminar los datos de cada área. También se ha agregado un buscador para facilitar la búsqueda de registros en la tabla.



The screenshot shows a form titled "Crear curso". It contains the following fields: "Nombre" (text input with placeholder "nombre"), "Grado" (text input with placeholder "grado"), "Paralelo" (dropdown menu with "Seleccionar" and a downward arrow), "Modalidad" (dropdown menu with "Seleccionar" and a downward arrow), and "Año" (dropdown menu with "Año Lectivo enero2023-marzo2023" and a downward arrow). At the bottom left of the form is a blue "Guardar" button.

Figura 22. Formulario para administrar cursos.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 que incluye campos importantes: "Nombre", "Grado", "Paralelo", "Modalidad" y "Año lectivo". Este formulario se



utiliza para crear los cursos que se utilizan en el sistema. De esta manera, se pueden agregar un nuevo curso al sistema.

### HU007: Gestionar matrículas de estudiantes.



ID	Nombres	Apellidos	Cedula	Curso	Grado		
1	MERCEDES MARIA	LÓPEZ GONZÁLEZ	19001	curso 1	7mo de básica	Editar	Eliminar
2	BYRON DAVID	CEVALLOS TRUILLLO	19002	curso 2	6to de básica	Editar	Eliminar
3	GONZALO LUIS	BALCAZAR CAMPOVERDE	19003	curso 1	7mo de básica	Editar	Eliminar

Figura 23. Lista estudiantes matriculados

Elaborado por: Morales Carlos, 2023.

Se ha creado una tabla utilizando Bootstrap 5 que contiene los estudiantes matriculados en un curso del sistema y con su respectiva información necesaria. Además, se ha incluido un botón de "Editar" y otro de "Eliminar" para cada registro, lo que permite modificar o eliminar los datos de cada área. También se ha agregado un buscador para facilitar la búsqueda de registros en la tabla.

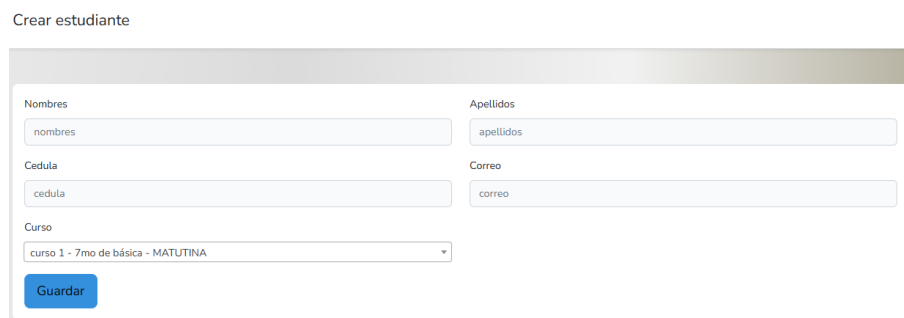
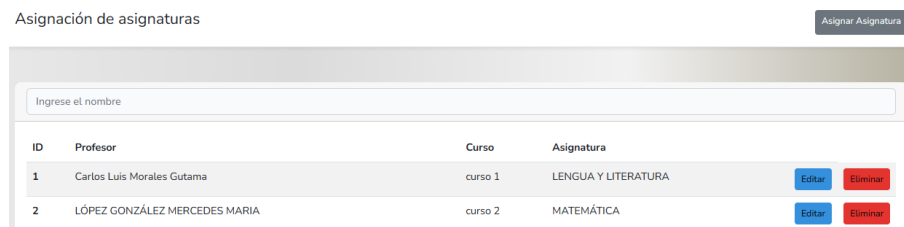


Figura 24. Formulario para administrar a los estudiantes matriculados.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 que incluye campos importantes: "Nombres", "Apellidos", "Cedula", "Email" y "Curso". Este formulario se utiliza para matricular a los estudiantes en un curso del sistema. De esta manera, se pueden agregar un nuevo estudiante a un curso en específico y crear su cuenta de usuario para posteriormente consultar sus calificaciones por asignatura.

#### **HU008: Gestionar asignación de asignaturas a profesores.**



ID	Profesor	Curso	Asignatura		
1	Carlos Luis Morales Gutama	curso 1	LENGUA Y LITERATURA	Editar	Eliminar
2	LÓPEZ GONZÁLEZ MERCEDES MARIA	curso 2	MATEMÁTICA	Editar	Eliminar

Figura 25. Lista de asignación de asignaturas a docentes.

Elaborado por: Morales Carlos, 2023.

Se ha creado una tabla utilizando Bootstrap 5 que contiene las asignaciones de asignaturas a docentes del sistema y con su respectiva información necesaria. Además, se ha incluido un botón de "Editar" y otro de "Eliminar" para cada registro, lo que permite modificar o eliminar los datos de cada área. También se ha agregado un buscador para facilitar la búsqueda de registros en la tabla.

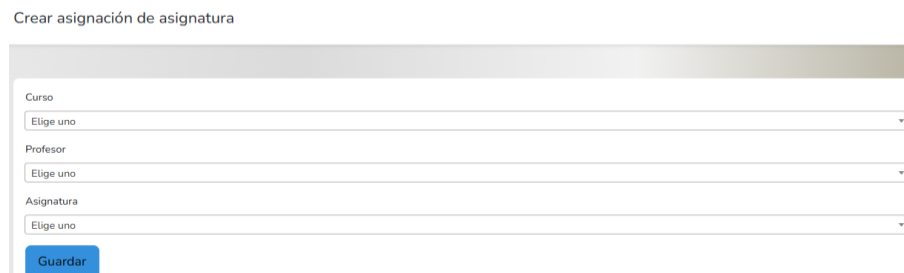


Figura 26. Formulario para administrar la asignación de asignaturas a docentes.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 que incluye tres campos importantes: "Curso" y "Asignatura". Este formulario se utiliza para asignar a los docentes las asignaturas que se utilizan en el sistema. De esta manera, se pueden agregar una nueva asignación al sistema.

Las historias de usuario corresponden a la siguiente iteración.

N° iteración	Historia de usuario	Tiempo estimado	
		Días	Horas
3	Registrar calificaciones.	11	55
	Registrar asistencia.	9	45
	Ver calificaciones parciales	4	20
	Descargar reporte de calificaciones.	6	30
	Descargas reporte de asistencia.	5	25

Tabla 92. Historias de usuario iteración 3.

Elaborado por: Morales Carlos, 2023.

### HU009: Registrar calificaciones.

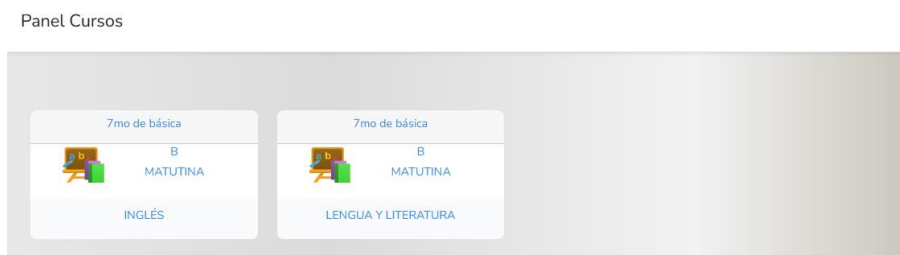


Figura 27. Tarjetas de asignaturas de docente

Elaborado por: Morales Carlos, 2023.

Se ha desarrollado una vista mediante la utilización de Bootstrap 5, que presenta de manera visualmente atractiva y organizada la información de cada asignatura asignada a un docente en el curso. La interfaz se compone de tarjetas que incluyen información detallada sobre el grado, modalidad y nombre de cada asignatura, permitiendo una rápida identificación de la información requerida por el docente.

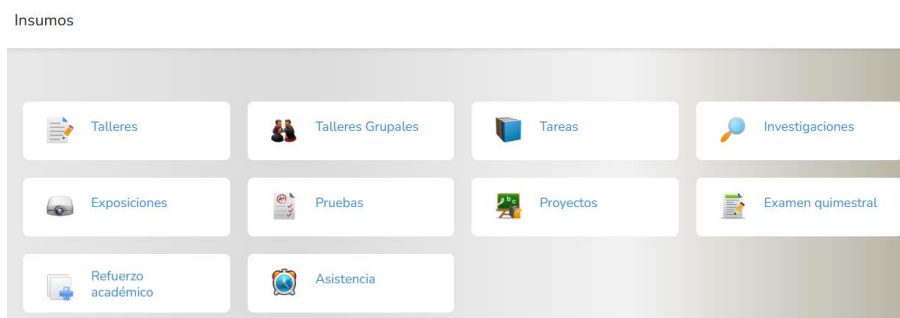


Figura 28. Tarjetas de insumos y asistencia.

Elaborado por: Morales Carlos, 2023.

Se ha desarrollado una interfaz de usuario mediante la utilización de Bootstrap 5, que presenta de manera visualmente atractiva y organizada la información de cada insumo que puede administrar un docente en el curso. La interfaz se compone de tarjetas que incluyen información detallada sobre el nombre de cada insumo, permitiendo una rápida identificación de la información requerida por el docente.

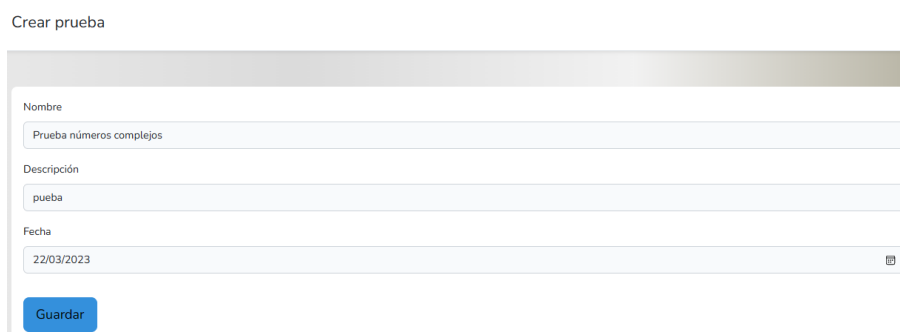
The image shows a form titled "Crear prueba". It has three input fields: "Nombre" with the value "Prueba números complejos", "Descripción" with the value "pueba", and "Fecha" with the value "22/03/2023". There is a "Guardar" button at the bottom left.

Figura 29. Formulario administrar insumo.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 que incluye tres campos importantes: "Nombres", "Descripción" y "Fecha". Este formulario se utiliza para administrar los insumos en un curso del sistema. De esta manera, se pueden agregar una nueva insumo a una asignatura en específico, para posteriormente ingresar su calificación a cada estudiante.

Prueba números complejos			pueba	2023-03-22
Estudiantes	Nota	Descripción		
BALCAZAR CAMPOVERDE GONZALO LUIS	<input type="text"/>	<input type="text"/>		
CÁRDENAS MOLINA JAIME EDUARDO	<input type="text"/>	<input type="text"/>		
CHUCHUCA SERRANO JAIME VICENTE	<input type="text"/>	<input type="text"/>		
COBOS VELOZ ALEX RUBEN	<input type="text"/>	<input type="text"/>		

Figura 30. Formulario para administrar las calificaciones por insumo.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 para facilitar la asignación de notas y descripciones a los estudiantes después de haber creado un nuevo insumo para una asignatura. El formulario incluye una lista de los estudiantes matriculados en la asignatura correspondiente, lo que permite una rápida selección del estudiante a calificar. Además, el formulario incluye campos para asignar la nota y una descripción si es necesario, lo que ayuda a mantener un registro detallado de las calificaciones y la progresión académica de los estudiantes.

Pruebas Nueva prueba

ID	Nombre	Descripción	Fecha	
1	Prueba números complejos	pueba	2023-03-22	<input type="button" value="Editar"/> <input type="button" value="Notas"/> <input type="button" value="Eliminar"/>

Figura 31. Lista de insumo creado.


Elaborado por: Morales Carlos, 2023.

Se diseñó una tabla con Bootstrap 5 para listar de manera organizada todos los insumos creados en el curso, incluyendo información como la descripción y fecha de creación.

Se incluyeron tres botones para cada insumo, el botón "Editar" para modificar la información, el botón "Ver notas" para acceder a la lista de estudiantes con las respectivas notas y el botón "Eliminar" para eliminar el insumo en caso de ser necesario. Con esta tabla, los usuarios pueden visualizar rápidamente los insumos del curso y realizar las acciones necesarias para su gestión de manera eficiente con un buscador para realizar búsquedas más rápidas.

### HU010: Registrar asistencia.

Crear horario



Fecha inicio: dd/mm/aaaa  
Fecha fin: dd/mm/aaaa  
Hora inicio: --:--  
Hora fin: --:--  
Dia: Lunes

Guardar

Figura 32. Formulario para crear horarios de sesiones.

Elaborado por: Morales Carlos, 2023.

Se ha creado un formulario utilizando Bootstrap 5 que incluye cinco campos importantes: "Fecha inicio", "Fecha fin", "Hora inicio", "Hora fin" y "Dia". Este formulario se utiliza para crear las sesiones de las clases de una asignatura de un curso del sistema. De esta manera, se pueden agregar unas nuevas sesiones para llevar un registro de la asistencia de estudiantes en un curso en específico.

Horarios Nuevo horario

Ingrese la fecha

	fecha	horas		
<input type="checkbox"/>	2023-04-13	2	Asistencia	Eliminar
<input type="checkbox"/>	2023-04-20	2	Asistencia	Eliminar
<input type="checkbox"/>	2023-04-27	2	Asistencia	Eliminar

Horarios anteriores

	fecha	horas		
<input type="checkbox"/>	2023-04-06	2	Asistencia	Eliminar

Figura 33. Lista de sesiones para tomar asistencia.

Elaborado por: Morales Carlos, 2023.

Se diseñó una tabla con Bootstrap 5 para tomar la asistencia de los estudiantes en diferentes sesiones, donde se incluyen la fecha y hora correspondiente. Además, se agregaron casillas de verificación para eliminar múltiples registros y un botón para tomar la asistencia y eliminar cada registro individualmente.

Estudiantes	Estado
BALCAZAR CAMPOVERDE GONZALO LUIS	<input type="button" value="Presente"/> <input type="button" value="Atraso"/> <input type="button" value="Justificado"/> <input type="button" value="Injustificado"/>
CÁRDENAS MOLINA JAIME EDUARDO	<input type="button" value="Presente"/> <input type="button" value="Atraso"/> <input type="button" value="Justificado"/> <input type="button" value="Injustificado"/>
CHUCHUCA SERRANO JAIME VICENTE	<input type="button" value="Presente"/> <input type="button" value="Atraso"/> <input type="button" value="Justificado"/> <input type="button" value="Injustificado"/>

Figura 34. Formulario para registrar asistencia.

Elaborado por: Morales Carlos, 2023.

Se diseñó un formulario con Bootstrap 5 que incluye una lista de estudiantes y botones para registrar su asistencia, incluyendo opciones de presente, atraso, justificado e injustificado.

#### **HU011: Ver calificaciones actuales de estudiantes.**

Asignatura	Calificaciones
LENGUA Y LITERATURA	<input type="button" value="Calificaciones"/>
MATEMÁTICA	<input type="button" value="Calificaciones"/>
FILOSOFÍA	<input type="button" value="Calificaciones"/>
INGLÉS	<input type="button" value="Calificaciones"/>

Figura 35. Vista lista de asignaturas de estudiante.

Elaborado por: Morales Carlos, 2023.

Se diseñó una tabla con Bootstrap 5 que incluye una lista de asignaturas de un estudiante y un botón para ver el registro de calificaciones.

Talleres			
Nombre	Descripción	Fecha	Nota
Taller 1	números complejos	2023-01-04	5.8
taller 2	ejercicios en clases	2023-01-14	7

Talleres Grupales			
Nombre	Descripción	Fecha	Nota
taller grupat 1	ejercicios en clases	2023-01-19	6

Figura 36. Vista ver calificaciones estudiantes.

Elaborado por: Morales Carlos, 2023.

Se ha diseñado una interfaz gráfica utilizando Bootstrap 5, que incluye una lista detallada del registro de calificaciones que muestra el Descripción de la tarea, su descripción, fecha y calificación correspondiente.

**HU012: Descargar reporte de calificaciones.**

Figura 37. Reporte calificaciones versión extendida.

Elaborado por: Morales Carlos, 2023.

Se genera un archivo Excel con los promedios por separado para cada uno de los insumos de las calificaciones de una asignatura.



NÓMINA	Primer Cuatrimestre					Segundo Cuatrimestre				
	Prom. parcial	Prom. parcial	Promed. Pr	Exa. Pr	Exa. Pr	Prom. parcial	Prom. parcial	Promed. Pr	Exa. Pr	Exa. Pr
BALCAZAR CAMPOVERDE GONZALO LUIS	5	0	0	0	0	5	0	0	0	0
CÁRDENAS MOLINA JAMIE EDUARDO	3	2	0	0	0	4	1	0	0	0
CHUCHUCA SERRANO JAMIE VICENTE	1	2	2	0	0	2	0	1	2	40
COBOS VELOZ ALEX RUBEN	2	2	1	0	0	2	0	2	1	30
GARCÍA GARCÍA DANILLO FERNANDO	3	1	0	1	20	4	1	0	0	0

Figura 38. Reporte calificaciones versión resumida.

Elaborado por: Morales Carlos, 2023.

Se genera un archivo Excel con los promedios finales de una asignatura.

### HU013: Descargas reporte de asistencia.

**Unidad Educativa Huachi Grande**

ASIGNATURA: INGLÉS

PROFESOR: Carlos Luis Morales Gutama

CURSO: curso 1

GRADO: 7mo de básica

PARALELO: B

NÓMINA	PRIMER CUATRIMESTRE					SEGUNDO CUATRIMESTRE				
	P	A	J	I	%I	P	A	J	I	%I
BALCAZAR CAMPOVERDE GONZALO LUIS	5	0	0	0	0	5	0	0	0	0
CÁRDENAS MOLINA JAMIE EDUARDO	3	2	0	0	0	4	1	0	0	0
CHUCHUCA SERRANO JAMIE VICENTE	1	2	2	0	0	2	0	1	2	40
COBOS VELOZ ALEX RUBEN	2	2	1	0	0	2	0	2	1	30
GARCÍA GARCÍA DANILLO FERNANDO	3	1	0	1	20	4	1	0	0	0

Figura 39. Reporte de asistencia estudiantes en documentos en formato PDF.

Elaborado por: Morales Carlos, 2023.

Se genera un documento en formato PDF que incluye los nombres de los estudiantes con el número de asistencias presentes, atrasos, justificados e injustificados, así como

el porcentaje de faltas injustificadas. Si el porcentaje de faltas injustificadas supera el 10%, se marca en rojo para que el docente pueda identificar fácilmente a los estudiantes con asistencia insuficiente. El objetivo de este informe es proporcionar al docente una herramienta clara y efectiva para monitorear la asistencia de los estudiantes.

N° iteración	Historia de usuario	Tiempo estimado	
		Días	Horas
4	Consultar calificaciones de estudiante.	4	20
	Auditoria por cabio o borrado de los registros de calificaciones	4	20
	Notificaciones de estudiantes con asistencia insuficiente.	3	15
	General certificado de promoción.	5	25
	Generar boleta de calificaciones.	5	25

Tabla 93. Historias de usuario iteración 4.

Elaborado por: Morales Carlos, 2023.

#### **HU014: Consultar calificaciones de estudiante.**

Calificaiones BALCAZAR CAMPOVERDE GONZALO LUIS

Talleres			
Nombre	Descripción	Fecha	Nota
Taller 1	números complejos	2023-01-04	7.55
taller 2	ejercicios en clases	2023-01-14	9.6

Talleres Grupales			
Nombre	Descripción	Fecha	Nota
taller grupal 1	ejercicios en clases	2023-01-19	6

Figura 40. Vista ver calificaciones estudiantes.

Elaborado por: Morales Carlos, 2023.

Se ha diseñado una vista utilizando Bootstrap 5, que incluye una lista detallada del registro de calificaciones que muestra el Descripción de la tarea, su descripción, fecha y calificación correspondiente.

### HU15: Auditoría por cambio o borrado de los registros de calificaciones.

Tabla Auditoria

18 taller 2 Estudiante: MARLENE ESTEFANIA NOVILLO JARA

Profesor	Evento	Nota antigua	Nota nueva	Ip	Agente	Tipo Insumo	Id	Fecha
Carlos Luis Morales Gutama	Actualizado	{"nota":5}	{"nota":"9.5"}	127.0.0.1	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML like Gecko) Chrome/111.0.0.0 Safari/537.36 Edg/111.0.1661.54	Taller	<a href="#">info</a>	2023-03-31 00:57:33
Carlos Luis Morales Gutama	Actualizado	{"nota":6.99}	{"nota":"8.65"}	127.0.0.1	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML like Gecko) Chrome/111.0.0.0 Safari/537.36 Edg/111.0.1661.54	Taller	<a href="#">info</a>	2023-03-31 00:56:50

Figura 41. Vista tabla auditoria de calificaciones.

Elaborado por: Morales Carlos, 2023.

Se ha creado una tabla que contiene información sobre los cambios y eliminaciones de calificaciones, que incluye los datos necesarios para identificar tanto la nota actual como la antigua, la dirección IP y la información del usuario que realizó el cambio, como su ubicación. Esta tabla ha sido diseñada para garantizar la transparencia y la responsabilidad en el proceso de gestión de calificaciones.

### HU016: Notificaciones de estudiantes con asistencia insuficiente.

Notificaciones

No leídas	Leídas
<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <a href="#">Has recibido una notificación de curso 1</a> </div>	<div style="background-color: #dc3545; color: white; width: 20px; height: 20px; margin: 0 auto; display: flex; align-items: center; justify-content: center;"> <span style="font-size: 10px;">x</span> </div>

Figura 42. Vista notificaciones leídas y no leídas.

Elaborado por: Morales Carlos, 2023.

Se ha desarrollado una tabla con Bootstrap 5 para listar las notificaciones del usuario, tanto las leídas como las no leídas. Esta tabla ha sido diseñada con el objetivo de mejorar la experiencia del usuario al visualizar sus notificaciones.

Notificaciones



Figura 43. Vista de notificación de estudiantes con inasistencia.

Elaborado por: Morales Carlos, 2023.

Se ha diseñado una tarjeta con Bootstrap 5 con el propósito de mostrar los nombres de los estudiantes que presentan un porcentaje de inasistencias superior al 10%. El uso de esta tarjeta permitirá identificar de manera rápida y visual a aquellos estudiantes que presentan problemas de asistencia, lo que facilitará el seguimiento y la intervención por parte de los docentes.

## HU017: Generar certificado de promoción.

**CERTIFICADO DE PROMOCIÓN**

El Rector(a) Director (a) de la Institución Educativa:

**UNIDAD EDUCATIVA HUACHI GRANDE**  
Año Lectivo enero2023-marzo2023

De conformidad con lo prescrito en el Art. 197 del Reglamento General a la Ley Orgánica de Educación Intercultural y demás normativas vigentes, certifica que el/la estudiante

**BALCAZAR CAMPOVERDE GONZALO LUIS**

del CUARTO AÑO DE EDUCACIÓN GENERAL BÁSICA  
CÓDIGO ESTUDIANTE No 19003, obtuvo las siguientes calificaciones durante el presente año lectivo.

AREA	ASIGNATURA	PROMEDIO ANUAL	
		CALIFICACIÓN CUANTITATIVA	CALIFICACIÓN CUALITATIVA
LENGUA Y LINGÜÍSTICA	LENGUA Y LINGÜÍSTICA	1,2	NA
MATEMÁTICA	MATEMÁTICA	0	NA
CIENCIAS SOCIALES	FILOSOFÍA	4,00	PA
LENGUA EXTRANJERA	INGLÉS	0	NA
Promedio		1,365	NA

No cumple la nota mínima de 7  
Para certificar suscribe esta director/a - Rector/a del Plantel.  
Dado y firmado  
AMBATO, TUNGURAHUA, el martes, 4 de abril de 2023 23:28

---


Rector(a) Director(a)
Secretario(a)
Docente Tutor

Figura 44. Certificado de promoción.

Elaborado por: Morales Carlos, 2023.

Se genera un certificado en un documento en formato PDF que detalla el promedio final de los estudiantes en cada asignatura.

### HU018: Generar boleta de calificaciones.



**UNIDAD EDUCATIVA HUACHI GRANDE**  
Año Lectivo enero2023-marzo2023

---

Datos Informativos del estudiante

Nombre: BALCAZAR CAMPOVERDE GONZALO LUIS Grado /Paralelo: B Jornada: M

cedula: 19803

**REPORTE ANUAL DE CALIFICACIONES**

Area	Asignatura	Primer Quimestre								Segundo Quimestre								PROMEDIO ANUAL	
		P1	P2	PROM	PROM 80%	EXAM	EXAM 20%	QUIM	EC	P1	P2	PROM	PROM 80%	EXAM	EXAM 20%	QUIM	EC	PA	EC
LENGUA Y LITERATURA	LENGUA Y LITERATURA	6	0	3	2.4	-	0	2.4	NA	0	0	0	0	-	0	0	NA	1.2	NA
MATEMÁTICA	MATEMÁTICA	0	0	0	0	-	0	0	NA	0	0	0	0	-	0	0	NA	0	NA
CIENCIAS SOCIALES	FILOSOFÍA	7.55	5.5	6.53	5.22	-	0	5.22	PA	0	6	3	2.4	-	0	2.4	NA	3.81	NA
LENGUA EXTRANJERA	INGLÉS	0	0	0	0	-	0	0	NA	0	0	0	0	-	0	0	NA	0	NA

COMPORTAMIENTO	ESCALA CUALITATIVA	ESCALA CUANTITATIVA
A Muy satisfactorio	DA Domina los aprendizajes requeridos	9.00 - 10.00
B Satisfactorio	AA Alcanza los aprendizajes requeridos	7.00 - 8.99
C Poco Satisfactorio	PA Próximo a alcanzar los aprendizajes requeridos	4.01 - 6.99
D Marginal	NA No alcanza los aprendizajes requeridos	<=4
E Insatisfactorio		

Figura 45. Boleta de calificaciones.

Elaborado por: Morales Carlos, 2023.

Se genera un informe de calificaciones en la que se incluye el promedio de los estudiantes en cada asignatura durante los dos parciales de cada quimestre, así como su desempeño en los exámenes, proyectos y refuerzo académico. Este documento abarca todas las asignaturas que estudia el estudiante y se entregará a su padre en futuras reuniones.

## HU019: Generar reporte de calificaciones concentrado.



### UNIDAD EDUCATIVA HUACHI GRANDE

#### CONCENTRADO DE CLIFICACIONES POR PARALELO

ESTUDIANTES	LENGUA Y LITERATURA			MATEMÁTICA			FILOSOFÍA			INGLÉS			Prom. Anual
	Quim 1.	Quim 2.	Prom Anual	Quim 1.	Quim 2.	Prom Anual	Quim 1.	Quim 2.	Prom Anual	Quim 1.	Quim 2.	Prom Anual	
BALCAZAR CAMPOVERDE GONZALO LUIS	2.4	0	1.2	0	0	0	0	0	0	5.63	2.4	4.02	1.31
CÁRDENAS MOLINA JAIME EDUARDO	3.2	0	1.6	0	0	0	0	0	0	5.59	2.8	4.2	1.45
CHUCHUCA SERRANO JAIME VICENTE	3	0	1.5	0	0	0	0	0	0	5.64	1.6	3.62	1.28
COBOS VELOZ ALEX RUBEN	2.6	0	1.3	0	0	0	0	0	0	6.66	2	4.33	1.41

Figura 46. Reporte concentrado de calificaciones por paralelo.

Elaborado por: Morales Carlos, 2023.

Se genera un informe en un documento en formato PDF que incluye la información de los promedios de calificaciones de cada parcial, quimestre y anual de cada asignatura de cada paralelo.

### 3.3.6 Fase 4: Producción

Durante la fase de producción, es esencial verificar que se hayan cumplido todos los requerimientos del proyecto y probar todas las funcionalidades. En la metodología XP, se da una gran importancia a la satisfacción del cliente, y las historias de usuario son una herramienta fundamental para lograr este objetivo. Por lo tanto, la evaluación de las historias de usuario puede ayudar a tomar decisiones informadas sobre la implementación y puesta en producción de las funcionalidades del sistema.

Prueba de aceptación	
Clave: PA001.	Historia de usuario: HU001.

<b>Nombre:</b> Registro usuario.
<b>Descripción:</b> Usuario puede registrarse correctamente en el sistema.
<b>Condición de ejecución:</b> Ninguna.
<b>Interfaz:</b> El usuario completa los campos obligatorios del formulario de registro, incluyendo contraseña y correo electrónico. El usuario hace clic en el botón de registro.
<b>Resultado esperado:</b> El usuario puede acceder al sistema con el correo electrónico y la contraseña proporcionados durante el registro.
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.

Tabla 94. Prueba de aceptación – Registro usuario.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA002.	<b>Historia de usuario:</b> HU001.
<b>Nombre:</b> Inicio de sesión.	
<b>Descripción:</b> ingreso de usuario al sistema.	
<b>Condición de ejecución:</b> Ninguna.	
<b>Interfaz:</b> El usuario completa los campos obligatorios del formulario de inicio de sesión, incluyendo contraseña y correo electrónico. El usuario hace clic en el botón de ingresar.	
<b>Resultado esperado:</b> El usuario puede acceder al sistema con el correo electrónico y la contraseña proporcionados durante el registro.	
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.	

Tabla 95. Prueba de aceptación – Inicio de sesión.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA003.	<b>Historia de usuario:</b> HU002.
<b>Nombre:</b> Lista de usuarios.	

<b>Descripción:</b> La lista de usuarios estará disponible para ser visualizada por el administrador.
<b>Condición de ejecución:</b> El usuario debe haber iniciado sesión en el sistema y poseer rol de administrador.
<b>Interfaz:</b> El usuario realiza búsquedas en la tabla de usuarios. El usuario selecciona la opción "Editar" para actualizar los roles de un usuario. El usuario marca los roles a otorgar al usuario seleccionado y guardar.
<b>Resultado esperado:</b> El usuario tiene acceso a las funciones de los roles que posee.
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.

Tabla 96. Prueba de aceptación – Lista de usuarios.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA004.	<b>Historia de usuario:</b> HU002.
<b>Nombre:</b> Crear rol.	
<b>Descripción:</b> La lista de usuarios estará disponible para ser visualizada por el administrador.	
<b>Condición de ejecución:</b> El usuario debe haber iniciado sesión en el sistema y poseer rol de administrador.	
<b>Interfaz:</b> El usuario realiza búsquedas en la tabla de usuarios. El usuario selecciona la opción "Editar" para actualizar los roles de un usuario. El usuario marca los roles a otorgar al usuario seleccionado y guardar.	
<b>Resultado esperado:</b> El usuario tiene acceso a las funciones de los roles que posee.	
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.	

Tabla 97. Prueba de aceptación – Crear rol.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA005.	<b>Historia de usuario:</b> HU003.
<b>Nombre:</b> Gestión de áreas.	



<b>Descripción:</b> Verificar que el sistema permite la gestión adecuada de las asignaturas.
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión en el sistema
<b>Interfaz:</b> El administrador del sistema accede a la sección de gestión de áreas. El administrador crea una nueva área y le asigna un nombre y una descripción. El administrador edita el nombre y la descripción de un área existente. El administrador elimina un área existente del sistema. El administrador busca un área en específico por el nombre.
<b>Resultado esperado:</b> El sistema debe mostrar un mensaje de confirmación para cada acción realizada. La nueva área creada debe aparecer en la lista de áreas del sistema. El nombre y la descripción del área editada deben ser actualizados correctamente. El área eliminada no debe aparecer en la lista de áreas del sistema. El sistema debe mostrar una lista de las áreas que coinciden con el criterio de búsqueda.
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.

Tabla 98. Prueba de aceptación – Gestión de áreas.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA006.	<b>Historia de usuario:</b> HU004.
<b>Nombre:</b> Gestión de asignaturas.	
<b>Descripción:</b> Verificar que el sistema permite la gestión adecuada de las asignaturas.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión en el sistema	
<b>Interfaz:</b> El administrador del sistema accede a la sección de gestión de asignaturas. El administrador crea una nueva área y le asigna un nombre y un área. El administrador edita el nombre y el área de la asignatura existente. El administrador elimina una asignatura existente del sistema. El administrador busca una asignatura en específico por el nombre.	

<p><b>Resultado esperado:</b></p> <p>El sistema debe mostrar un mensaje de confirmación para cada acción realizada.</p> <p>La nueva asignatura creada debe aparecer en la lista de asignaturas del sistema.</p> <p>El nombre y la descripción del área editada deben ser actualizados correctamente.</p> <p>El área eliminada no debe aparecer en la lista de asignaturas del sistema.</p> <p>El sistema debe mostrar una lista de las asignaturas que coinciden con el criterio de búsqueda.</p>
<p><b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.</p>

Tabla 99. Prueba de aceptación – Gestión de asignaturas.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA007.	<b>Historia de usuario:</b> HU005.
<b>Nombre:</b> Gestión de años lectivos.	
<b>Descripción:</b> Verificar que el sistema permite la gestión adecuada de los años lectivos.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión en el sistema	
<p><b>Interfaz:</b></p> <p>El administrador del sistema accede a la sección de gestión de años lectivos.</p> <p>El administrador crea un nuevo año lectivo y le asigna un nombre y fechas correspondientes.</p> <p>El administrador edita el nombre y fechas correspondientes de año lectivo existente.</p> <p>El administrador elimina un año lectivo existente del sistema.</p> <p>El administrador busca un año lectivo en específico por el nombre.</p>	
<p><b>Resultado esperado:</b></p> <p>El sistema debe mostrar un mensaje de confirmación para cada acción realizada.</p> <p>La nueva asignatura creada debe aparecer en la lista de años lectivos del sistema.</p> <p>El nombre y las fechas del año lectivo editado deben ser actualizados correctamente.</p> <p>El año lectivo eliminada no debe aparecer en la lista de asignaturas del sistema.</p> <p>El sistema debe mostrar una lista de los años lectivos que coinciden con el criterio de búsqueda.</p>	
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.	

Tabla 100. Prueba de aceptación – Gestión de años lectivos.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA008.	<b>Historia de usuario:</b> HU006.
<b>Nombre:</b> Gestión de cursos.	
<b>Descripción:</b> Verificar que el sistema permite la gestión adecuada de los cursos.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión en el sistema.	
<b>Interfaz:</b> El administrador del sistema accede a la sección de gestión de cursos. El administrador crea un nuevo curso y le asigna un nombre, grado, paralelo, modalidad y año lectivo correspondientes. El administrador edita el nombre, grado, paralelo, modalidad y año lectivo correspondientes de curso existente. El administrador elimina un curso existente del sistema. El administrador busca un curso en específico por el nombre.	
<b>Resultado esperado:</b> El sistema debe mostrar un mensaje de confirmación para cada acción realizada. El nuevo curso creado debe aparecer en la lista de cursos del sistema. El nombre, grado, paralelo, modalidad y año lectivo del curso editado deben ser actualizados correctamente. El curso eliminado no debe aparecer en la lista de asignaturas del sistema. El sistema debe mostrar una lista de los cursos que coinciden con el criterio de búsqueda.	
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.	

Tabla 101. Prueba de aceptación – Gestión de cursos.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA009.	<b>Historia de usuario:</b> HU007.
<b>Nombre:</b> Gestión matrículas de estudiantes.	
<b>Descripción:</b> Verificar que el sistema permite la gestión adecuada de la matrícula de los estudiantes.	

<p><b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión en el sistema y tener un rol con los permisos adecuados.</p>
<p><b>Interfaz:</b></p> <p>El administrador del sistema accede a la sección de gestión de matrícula de estudiantes.</p> <p>El administrador crea un nuevo estudiante y le asigna un curso correspondiente.</p> <p>El administrador edita el nombre, apellidos, cedula, correo y curso correspondientes de estudiante existente.</p> <p>El administrador elimina un estudiante matriculado en curso existente del sistema.</p> <p>El administrador busca un estudiante matriculado en un curso en específico por los nombres del estudiante.</p>
<p><b>Resultado esperado:</b></p> <p>El sistema debe mostrar un mensaje de confirmación para cada acción realizada.</p> <p>La nueva matricula creada debe aparecer en la lista de matrículas del sistema.</p> <p>Los datos de la matriculas editado deben ser actualizados correctamente.</p> <p>La matrícula del estudiante eliminado no debe aparecer en la lista de matrículas del sistema.</p> <p>El sistema debe mostrar una lista del estudiante matriculado que coinciden con el criterio de búsqueda.</p>
<p><b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.</p>

Tabla 102. Prueba de aceptación – Gestión matrículas de estudiantes.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA010.	<b>Historia de usuario:</b> HU007.
<b>Nombre:</b> Importar matrículas de estudiantes.	
<b>Descripción:</b> Verificar que el sistema permite la importar un archivo Excel adecuadamente la matrícula de los estudiantes.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión en el sistema y tener un rol de profesor en el sistema.	
<b>Interfaz:</b>	

<p>El usuario prepara un archivo Excel con las matrículas de los estudiantes en un curso, siguiendo el formato requerido por el sistema.</p> <p>El usuario accede al formulario de importación de matrículas.</p> <p>El usuario confirma la importación y el sistema realiza la matrícula de los estudiantes en el curso.</p>
<p><b>Resultado esperado:</b></p> <p>El sistema debe validar el archivo Excel e importar los datos matriculando a los estudiantes en un curso determinados y registrar al estudiante en el sistema de ser necesario.</p>
<p><b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.</p>

Tabla 103. Prueba de aceptación – Importar matrículas de estudiantes.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA011.	<b>Historia de usuario:</b> HU008.
<b>Nombre:</b> Gestión de asignación de asignaturas a profesores.	
<b>Descripción:</b> Verificar que el sistema permite la gestión adecuada de las asignaturas.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión en el sistema	
<p><b>Interfaz:</b></p> <p>El administrador del sistema accede a la sección de gestión de asignación de asignaturas a profesores.</p> <p>El administrador crea una nueva asignación de asignaturas a profesores y le asigna un curso y una asignatura.</p> <p>El administrador edita el curso, profesor y asignatura de una asignación de asignaturas existente.</p> <p>El administrador elimina una asignación de asignaturas existente del sistema.</p> <p>El administrador busca una asignación de asignaturas a profesores en específico por el nombre del profesor.</p>	
<b>Resultado esperado:</b> El sistema debe mostrar un mensaje de confirmación para cada acción realizada.	

<p>La nueva asignación de asignaturas a profesores creada debe aparecer en la lista de asignación de asignaturas del sistema.</p> <p>El nombre y la descripción de la asignación de asignaturas a profesores editada deben ser actualizados correctamente.</p> <p>La asignación de asignaturas eliminada no debe aparecer en la lista de asignación de asignaturas a profesores del sistema.</p> <p>El sistema debe mostrar una lista de la asignación de asignaturas que coinciden con el criterio de búsqueda.</p>
<p><b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.</p>

Tabla 104. Prueba de aceptación – Gestión de asignación de asignaturas a profesores.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA012.	<b>Historia de usuario:</b> HU009.
<b>Nombre:</b> Gestión de calificaciones.	
<b>Descripción:</b> Verificar que el sistema permite la gestión adecuada de las calificaciones.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de profesor en el sistema	
<p><b>Interfaz:</b></p> <p>El usuario del sistema accede a la sección de gestión de calificaciones.</p> <p>El usuario crea una nueva insumo le asigna un nombre, descripción y fecha y posterior asignar calificaciones a todos los estudiantes del curso.</p> <p>El usuario edita el nombre y la descripción del insumo.</p> <p>El usuario editar la calificación de un estudiante.</p> <p>El usuario elimina un insumo del sistema.</p> <p>El usuario busca un insumo en específico por el nombre.</p>	
<p><b>Resultado esperado:</b> El sistema debe mostrar un mensaje de confirmación para cada acción realizada.</p> <p>El nuevo insumo creada debe aparecer en la lista de insumos del sistema.</p> <p>El nombre, descripción y fecha del insumo editada deben ser actualizados correctamente.</p>	

<p>La calificación del estudiante debe actualizarse correctamente</p> <p>El insumo eliminado no debe aparecer en la lista de insumos del sistema.</p> <p>El sistema debe mostrar una lista de insumos que coinciden con el criterio de búsqueda.</p>
<p><b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.</p>

Tabla 105. Prueba de aceptación – Gestión de calificaciones.

Fuente: Elaborado por el autor.

<p><b>Prueba de aceptación</b></p>	
<p><b>Clave:</b> PA013.</p>	<p><b>Historia de usuario:</b> HU010.</p>
<p><b>Nombre:</b> Gestión de asistencia.</p>	
<p><b>Descripción:</b> Verificar que el sistema permite la gestión adecuada de la asistencia.</p>	
<p><b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de profesor en el sistema.</p>	
<p><b>Interfaz:</b></p> <p>El usuario del sistema accede a la sección de gestión de asistencia.</p> <p>El usuario crea lista de horario le asigna un rango de fechas, rango de horas, y día.</p> <p>El usuario toma asistencia a los estudiantes.</p> <p>El usuario editar la asistencia de un estudiante.</p> <p>El usuario elimina un horario del sistema.</p> <p>El usuario elimina selecciona múltiples horarios del sistema.</p> <p>El usuario busca un horario en específico por la fecha.</p>	
<p><b>Resultado esperado:</b> El sistema debe mostrar un mensaje de confirmación para cada acción realizada.</p> <p>La lista de horarios debe aparecer en la lista de insumos del sistema.</p> <p>La asistencia del estudiante debe actualizarse correctamente.</p> <p>Los horarios eliminados no deben aparecer en la lista de horarios del sistema.</p> <p>El sistema debe mostrar una lista de insumos que coinciden con el criterio de búsqueda.</p>	
<p><b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.</p>	

Tabla 106. Prueba de aceptación – Gestión de asistencia.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA014.	<b>Historia de usuario:</b> HU011.
<b>Nombre:</b> Mostrar calificaciones actuales de estudiantes.	
<b>Descripción:</b> Verificar que el sistema permite mostrar calificaciones actuales de estudiantes.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de estudiante en el sistema.	
<b>Interfaz:</b> El usuario del sistema accede a la sección de ver calificaciones actuales. El usuario selecciona el curso posterior la asignatura a consultar sus calificaciones.	
<b>Resultado esperado:</b> Se listan todos los insumos con su respectivo nombre, fecha y calificación.	
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.	

Tabla 107. Prueba de aceptación – Mostrar calificaciones actuales de estudiantes.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA015.	<b>Historia de usuario:</b> HU012.
<b>Nombre:</b> Descargar reporte de calificaciones por asignatura.	
<b>Descripción:</b> Verificar que el sistema permite descargar reporte de calificaciones.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de profesor en el sistema.	
<b>Interfaz:</b> El usuario del sistema accede a la sección de reportes opción generar reporte. El usuario selecciona la asignatura y versión de reporte extendida o resumida de las calificaciones.	
<b>Resultado esperado:</b> Se descargar un reporte con en formato Excel con los promedios de los insumos.	
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.	

Tabla 108. Prueba de aceptación – Descargar reporte de calificaciones por asignatura.

Fuente: Elaborado por el autor.



<b>Prueba de aceptación</b>	
<b>Clave:</b> PA016.	<b>Historia de usuario:</b> HU013.
<b>Nombre:</b> Descargar reporte de asistencia por asignatura.	
<b>Descripción:</b> Verificar que el sistema permite descargar reporte de asistencia.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de profesor en el sistema.	
<b>Interfaz:</b> El usuario del sistema accede a la sección de reportes opción generar reporte. El usuario selecciona la asignatura y formato de reporte en un documento en formato PDF o Excel.	
<b>Resultado esperado:</b> Se descargar un reporte con en un documento en formato PDF o Excel con la asistencia y porcentaje de inasistencia.	
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.	

Tabla 109. Prueba de aceptación – Descargar reporte de asistencia por asignatura.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA017.	<b>Historia de usuario:</b> HU014.
<b>Nombre:</b> Consultar calificaciones de estudiante.	
<b>Descripción:</b> Esta prueba de aceptación tiene como objetivo verificar que el usuario pueda consultar las calificaciones de un estudiante específico en el sistema.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de profesor, secretaria o inspector en el sistema.	
<b>Interfaz:</b> El usuario accede al formulario de consulta de calificaciones. El usuario selecciona el estudiante para el que desea consultar las calificaciones. El sistema muestra las calificaciones del estudiante para cada insumo en la asignatura correspondiente.	
<b>Resultado esperado:</b> El usuario debe poder acceder al formulario de consulta de calificaciones.	

El sistema debe mostrar las calificaciones del estudiante seleccionado correctamente.
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.

Tabla 110. Prueba de aceptación – Consultar calificaciones de estudiante.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA018.	<b>Historia de usuario:</b> HU015.
<b>Nombre:</b> Auditoría por cambio o borrado de los registros de calificaciones.	
<b>Descripción:</b> Esta prueba de aceptación tiene como objetivo verificar que el sistema registre los cambios y borrados de las calificaciones de los estudiantes para llevar un registro de auditoría.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de profesor en el sistema.	
<b>Interfaz:</b> El usuario modifica la calificación de un estudiante en un insumo. El usuario elimina la calificación de un estudiante en un insumo. El sistema registra el cambio de calificación en la tabla de auditoría con el usuario que realizó la modificación y la fecha, dirección IP y hora de la modificación.	
<b>Resultado esperado:</b> El sistema debe registrar el cambio de calificación en la tabla de auditoría con el usuario que realizó la modificación o eliminación del registro.	
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.	

Tabla 111. Prueba de aceptación – Auditoría por cambio o borrado de los registros de calificaciones.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA019.	<b>Historia de usuario:</b> HU016.
<b>Nombre:</b> Notificaciones de estudiantes con asistencia insuficiente.	
<b>Descripción:</b> Esta prueba de aceptación tiene como objetivo verificar que el sistema envíe notificaciones a los docentes listando los estudiantes que tienen una asistencia insuficiente en una asignatura específica.	

<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de profesor en el sistema.
<b>Interfaz:</b> El usuario registra la asistencia de una asignatura los últimos 7 días antes de terminar el primer y segundo quimestre.
<b>Resultado esperado:</b> El sistema debe enviar una notificación con una lista de todos los estudiantes con un porcentaje de inasistencia superior al 10% en el primero o segundo quimestre.
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.

Tabla 112. Prueba de aceptación – Notificaciones de estudiantes con asistencia insuficiente.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA020.	<b>Historia de usuario:</b> HU017.
<b>Nombre:</b> Descargar certificado de promoción.	
<b>Descripción:</b> Verificar que el sistema permite descargar certificado de promoción.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de secretaria en el sistema.	
<b>Interfaz:</b> El usuario del sistema accede a la sección de reportes opción reporte anuales. El usuario selecciona el año lectivo posterior un curso. El usuario clic sobre el botón certificado en el nombre del estudiante.	
<b>Resultado esperado:</b> Se descargar un certificado en un documento en formato PDF con el promedio final en cada asignatura.	
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.	

Tabla 113. Prueba de aceptación – Descargar certificado de promoción.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA021.	<b>Historia de usuario:</b> HU018.
<b>Nombre:</b> Generar boleta de calificaciones.	
<b>Descripción:</b> Verificar que el sistema permite descargar certificado de promoción.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de secretaria en el sistema.	
<b>Interfaz:</b> El usuario del sistema accede a la sección de reportes opción reporte anuales. El usuario selecciona el año lectivo posterior un curso. El usuario clic sobre el botón generar en el nombre del estudiante.	
<b>Resultado esperado:</b> Se descargar un reporte en un documento en formato PDF con las calificaciones promedios de los dos parciales de cada quimestre de cada asignatura de un estudiante en específico.	
<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.	

Tabla 114. Prueba de aceptación – Generar boleta de calificaciones.

Fuente: Elaborado por el autor.

<b>Prueba de aceptación</b>	
<b>Clave:</b> PA022.	<b>Historia de usuario:</b> HU019.
<b>Nombre:</b> Generar reporte concentrado de calificaciones.	
<b>Descripción:</b> Verificar que el sistema permite descargar reporte concentrado de un determinado curso y paralelo.	
<b>Condición de ejecución:</b> Usuario debe tener iniciada la sesión y tener rol de secretaria, Inspector en el sistema.	
<b>Interfaz:</b> El usuario del sistema accede a la sección de reportes opción reporte concentrado. El usuario selecciona el año lectivo posterior un curso. El usuario clic sobre el botón generar en el nombre del curso.	
<b>Resultado esperado:</b> Se descargar un reporte en un documento en formato PDF con las calificaciones promedios de los quimestres de cada asignatura de un curso y párralelo en específico.	

<b>Evaluación de la prueba:</b> Se pudo llevar a cabo la prueba de manera exitosa.
--

Tabla 115. Prueba de aceptación – Generar boleta de calificaciones.

Fuente: Elaborado por el autor.

### **3.3.7 Fase 5: Mantenimiento**

El sistema debe cumplir con los requerimientos establecidos y el desarrollador será responsable de garantizar su correcto funcionamiento a lo largo del tiempo. Para ello, se seguirá la metodología XP, lo que implica la generación de una nueva iteración para asegurar la integración y el cumplimiento de los requisitos. En otras palabras, es necesario establecer un proceso de mejora y optimización constante del sistema para garantizar su rendimiento óptimo a largo plazo.

## **CAPITULO IV. CONCLUSIONES Y RECOMENDACIONES**

### **4.1 Conclusiones**

- La estrategia de utilizar cuadros comparativos demostró ser eficaz para evaluar y seleccionar las tecnologías y metodologías de desarrollo ágil más adecuadas para el sistema web de gestión de calificaciones y asistencia de la unidad educativa Huachi Grande. A partir de los requerimientos del sistema, se optó por implementar la metodología XP, lo que permitió una ejecución ordenada y en línea con las historias de usuario. La utilización de estas técnicas de evaluación y planificación contribuyó significativamente a la asignación y cumplimiento de tareas en un cronograma establecido.
- A través de la interacción con el personal docente de la Unidad Educativa Huachi Grande, por medio de reuniones y observación de los procesos, se lograron identificar las necesidades de la institución. Esta información se convirtió en requerimientos funcionales claros y precisos para el desarrollo de la solución. Para asegurar una experiencia de usuario satisfactoria, se utilizó herramientas

apropiadas para crear una aplicación web dinámica y responsiva, capaz de adaptarse a cualquier dispositivo. De esta manera, se logró desarrollar una solución acorde a las necesidades de la institución y que brinda una experiencia de usuario agradable.

- Se ha seleccionado la metodología de desarrollo XP (Programación Extrema) como enfoque adecuado para el desarrollo del sistema web de gestión de calificaciones y asistencia de estudiantes en la Unidad Educativa. Esta metodología brinda una estructura sólida que garantiza la efectividad y precisión en la gestión de calificaciones y asistencia. Al seguir los principios y prácticas de XP, se podrá alcanzar los objetivos establecidos, permitiendo una gestión óptima de la información de los estudiantes y mejorando el proceso educativo en general.

## **4.2 Recomendaciones**

- Se recomienda utilizar navegadores actualizados para asegurar el correcto funcionamiento del sistema, ya que además de brindar mayor velocidad en la compilación y ejecución, también garantiza la compatibilidad con las últimas tecnologías y actualizaciones de seguridad, lo que reduce los riesgos de errores y vulnerabilidades en la aplicación.
- Se sugiere que se brinde capacitación al cuerpo docente para asegurar que puedan utilizar correctamente el sistema web desarrollado, dada su importancia como principales usuarios del mismo. Es recomendable establecer políticas claras para el uso del sistema y asignar un administrador responsable de su gestión y mantenimiento. Esto garantizará que el sistema esté siempre disponible y en óptimas condiciones para su uso.

## Bibliografía

- [1] K. D. R. Cedeño Vargas, “Diseño e implementación de un sistema web de control de matrícula y calificaciones para el colegio Rashid Torbay ‘Sismarashid’ en el cantón Playas, provincia del Guayas.”
- [2] D. X. Anrrango Benavides, “Sistema web para la gestión de matrículas y calificaciones de la unidad educativa fiscomisional Fray Bartolomé de las Casas Salasaca.”
- [3] A. V. Álvarez Cisneros, “Sistema Web Para La Gestión De La Pensión Diferenciada En La Unidad Educativa González Suárez,” 2018.
- [4] R. Villafuerte and A. Rodríguez, “Implementación de aplicación web para la gestión de matrículas y calificaciones, en la escuela de educación básica Enrique Fierro.”
- [5] E. M. Rojas *et al.*, “Sistema de Gestión Académica a través del desarrollo de Modelo-Vista-Controlador.”
- [6] Claro, “Las TICs han revolucionado los procedimientos de transmisión de la información,” May 23, 2019.
- [7] Oxford Languages, “Educación.”
- [8] F. García, S. Juárez, and L. Salgado, “Gestión escolar y calidad educativa,” *Revista Cubana de Educación Superior*, vol. 37, p. 3, 2018.
- [9] “Sistema de gestión de calificaciones - planteamiento.” <https://fundamentosingenieriasoftwareitsx2014.weebly.com/> (accessed Feb. 26, 2023).
- [10] “ASP.NET overview | Microsoft Learn.” <https://learn.microsoft.com/en-us/aspnet/overview> (accessed Mar. 09, 2023).
- [11] O.-A. Tekniikan, J. A. Liikenteen Ala T E K I J Ä / T :, and S. Rinnevuori, “Sistema de gestión de contenidos Laravel 8 para una página web,” 2020, Accessed: Feb. 26, 2023. [Online]. Available: <https://www.theseus.fi/handle/10024/355367>

- [12] Z. Subecz, “Web-development with Laravel framework,” *Gradus*, vol. 8, no. 1, 2021, doi: 10.47833/2021.1.csc.006.
- [13] Laravel LLC, “Laravel - The PHP Framework For Web Artisans,” *Laravel LLC*. 2019.
- [14] D. A. Ocaña Lema, “Implementación de un sistema de inventario informático y generación de código qr, para la facultad de mecánica en La Escuela Superior Politécnica De Chimborazo, desarrollado en laravel y gestor de base de datos MySql en el período 2020,” 2021, Accessed: Feb. 26, 2023. [Online]. Available: <https://sangabrielriobamba.edu.ec/tesis/sistemas/tesis27.pdf>
- [15] P. Palacios and A. Nicole, “ANÁLISIS COMPARATIVO DE METODOLOGÍAS PARA LA AUTOMATIZACIÓN DE PROCESOS CON METODOLOGÍA BPM VS. METODOLOGÍA TRADICIONAL DE DESARROLLO DE SOFTWARE”.
- [16] P. P. Reyes Cuba and R. P. Marín Apaza, “Aplicación web empleando la metodología XP para la gestión académica del instituto de Informática de la Universidad Nacional del Altiplano Puno - 2019,” 2021.
- [17] J. Patricia, Z. Gamboa, C. Alexandra, and L. Arreaga, “Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software,” *dialnet.unirioja.es*, vol. 3, no. 10, pp. 20–33, 2018, Accessed: Feb. 21, 2023. [Online]. Available: <https://dialnet.unirioja.es/servlet/articulo?codigo=6777227>
- [18] E. Bautista-Villegas, “Metodologías ágiles XP y Scrum, empleadas para el desarrollo de páginas web, bajo MVC, con lenguaje PHP y framework Laravel,” *Revista Amazonía Digital*, vol. 1, no. 1, 2022, doi: 10.55873/rad.v1i1.168.
- [19] F. José and G. Peñalvo, “Procesos y métodos de la Ingeniería Web Procesos y Métodos de Modelado para la Ingeniería Web y Web Semántica.” [Online]. Available: <http://grial.usal.eshttp://twitter.com/frangpContenidos>



- [20] S. C. Iza Poaquiza, “Aplicación web para la gestión administrativa de la comunidad Angahuana alto de la parroquia Santa Rosa.”
- [21] Sinnaps.com, “Metodología XP o Programación Extrema: ¿Qué es y cómo aplicarla?,” *Sinnaps.com*, 2020.
- [22] C. Chen, T. Su, G. Meng, Z. Xing, and Y. Liu, “From UI design image to GUI skeleton: A neural machine translator to bootstrap mobile GUI implementation,” *Proceedings - International Conference on Software Engineering*, pp. 665–676, May 2018, doi: 10.1145/3180155.3180240.
- [23] L. Fattorini, M. Marcheselli, C. Pisani, and L. Pratelli, “Design-based properties of the nearest neighbor spatial interpolator and its bootstrap mean squared error estimator,” *Biometrics*, vol. 78, no. 4, pp. 1454–1463, Dec. 2022, doi: 10.1111/BIOM.13505.
- [24] F. J. Lopez-Pellicer, R. Béjar, M. A. Latre, J. Noguerras-Iso, and J. Zarazaga-Soria, “GitHub como herramienta docente”, Accessed: Feb. 28, 2023. [Online]. Available:  
<http://subversion.apache.org/3http://mercurial.selenic.com/4http://git-scm.com/>

## **Anexos**

### **Codificación.**

#### **HU001: Formulario de ingreso al sistema.**

El diseño se realizó con Bootstrap 5 para todos los formularios y la página de inicio del sistema, con ayuda del motor de plantillas Blade para simplificar el código mediante la creación de componentes.

```

<x-guest-layout>
  <x-jet-authentication-card>
    <x-slot name="logo">
      <x-jet-authentication-card-logo />
    </x-slot>

    <div class="card-body">
      <x-jet-validation-errors class="mb-3 rounded-0" />

      @if (session('status'))
        <div class="alert alert-success mb-3 rounded-0" role="alert">
          {{ session('status') }}
        </div>
      @endif

      <form method="POST" action="{{ route('login') }}">
        @csrf
        <div class="mb-3">
          <x-jet-label value="{{ __('Email') }}" />

          <x-jet-input class="{{ $errors->has('email') ? 'is-invalid' : '' }}" type="email" name="email"
            :value="old('email')" required />
          <x-jet-input-error for="email"></x-jet-input-error>
        </div>

        <div class="mb-3">
          <x-jet-label value="{{ __('Password') }}" />

          <x-jet-input class="form-control{{ $errors->has('password') ? 'is-invalid' : '' }}" type="password"
            name="password" required autocomplete="current-password" />
        </div>
      </form>
    </div>
  </x-jet-authentication-card>
</x-guest-layout>

```

Figura 47. Diseño de formulario de ingreso al sistema.

Elaborado por: Morales Carlos, 2023.

```

<x-guest-layout>
  <x-jet-authentication-card>
    <x-slot name="logo">
      <x-jet-authentication-card-logo />
    </x-slot>

    <x-jet-validation-errors class="mb-3" />

    <div class="card-body">
      <form method="POST" action="{{ route('register') }}">
        @csrf

        <div class="mb-3">
          <x-jet-label value="{{ __('Name') }}" />

          <x-jet-input class="{{ $errors->has('name') ? 'is-invalid' : '' }}" type="text" name="name"
            :value="old('name')" required autofocus autocomplete="name" />
          <x-jet-input-error for="name"></x-jet-input-error>
        </div>

        <div class="mb-3">
          <x-jet-label value="{{ __('Email') }}" />

          <x-jet-input class="{{ $errors->has('email') ? 'is-invalid' : '' }}" type="email" name="email"
            :value="old('email')" required />
          <x-jet-input-error for="email"></x-jet-input-error>
        </div>

        <div class="mb-3">
          <x-jet-label value="{{ __('Password') }}" />

          <x-jet-input class="{{ $errors->has('password') ? 'is-invalid' : '' }}" type="password"
            name="password" required autocomplete="new-password" />
        </div>
      </form>
    </div>
  </x-jet-authentication-card>
</x-guest-layout>

```

Figura 48. Diseño de formulario de registro del sistema.

Elaborado por: Morales Carlos, 2023.

```

<!DOCTYPE html>
<html lang="{{ str_replace('_', '-', app()->getLocale()) }}">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="csrf-token" content="{{ csrf_token() }}">

  <title>{{ config('app.name', 'Laravel') }}</title>
  <link rel="shortcut icon" href="{{ asset('libro.ico') }}">
  <!-- Fonts -->
  <link href="https://fonts.googleapis.com/css?family=Nunito:400,600,700" rel="stylesheet">

  <!-- Styles -->
  <link rel="stylesheet" href="{{ asset('css/estilosmc.css') }}">

  @livewireStyles

  <!-- Scripts -->
  @vite(['resources/sass/app.scss', 'resources/js/app.js'])

  <script src="https://code.jquery.com/jquery-3.6.3.min.js"
    integrity="sha256-pvPw+upLPUjgMXY0G+800xUf+/Im1MZjXxxgOcBQBxU=" crossorigin="anonymous"></script>
  <link href="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/css/select2.min.css" rel="stylesheet" />
  <script src="https://cdn.jsdelivr.net/npm/select2@4.1.0-rc.0/dist/js/select2.min.js"></script>
</head>

<body class="fondo">
  {{-- <x-jet-banner /> --}}
  @livewire('navigation-menu')

  <!-- Page Heading -->

```

Figura 49. Diseño página de inicio.

Elaborado por: Morales Carlos, 2023.

## HU002: Gestionar perfiles de usuarios.

```

<x-app-layout>
  <x-slot name="header">
    <h2 class="h4 font-weight-bold">
      Usuarios
    </h2>
  </x-slot>

  @livewire('admin.users-index')
</x-app-layout>

```

Figura 50. Diseño de vista lista de usuarios.

Elaborado por: Morales Carlos, 2023.

```

<div class="card">
  <div class="card-header">
    <input type="text" wire:model="search" class="form-control" placeholder="Ingrese el nombre o correo">
  </div>

  @if ($users->count())
    <div class="card-body">
      <div class="table-responsive">
        <table class="table table-striped">
          <thead>
            <tr>
              <th scope="col">ID</th>
              <th scope="col">Nombre</th>
              <th scope="col">Email</th>
            </tr>
          </thead>
          <tbody>
            @foreach ($users as $user)
              <tr>
                <th scope="row">{{ $user->id }}</th>
                <td>{{ $user->name }}</td>
            </tr>
            @endforeach
          </tbody>
        </table>
      </div>
    </div>
  </if>
</div>

```

Figura 51. Componente Livewire listar y buscar usuarios.

Elaborado por: Morales Carlos, 2023.

```

class UsersIndex extends Component
{
    use WithPagination;

    public $search;
    // resetea la propiedad de buscar
    public function updatingSearch(){
        $this->resetPage();
    }
    protected $paginationTheme="bootstrap";
    public function render()
    {
        $users=User::where('name','LIKE','%'. $this->search . '%')
        ->orWhere('email','LIKE','%'. $this->search . '%')
        ->paginate();
        return view('livewire.admin.users-index',compact('users'));
    }
}

```

Figura 52. Función para listar y buscar usuarios.

Elaborado por: Morales Carlos, 2023.

```

public function edit(User $user)
{
    $roles= Role::all();
    return view('admin.users.edit', compact('user','roles'));
}

public function update(Request $request, User $user)
{
    $user->roles()->sync($request->roles);
    return redirect()->route('admin.users.edit',$user)->with('info','Se asigno rol correctamente');
}

```

Figura 53. Función para editar roles de usuarios.

Elaborado por: Morales Carlos, 2023.

```

7
public function store(Request $request)
{
    $request->validate([
        'name'=>'required'
    ]);

    $role = Role::create($request->all());
    $role->permissions()->sync($request->permissions);
    return redirect()->route('admin.roles.edit',$role)->with('info','El rol se creó con éxito');
}

```

Figura 54. Función para administrar roles y permisos de usuarios.

Elaborado por: Morales Carlos, 2023.

### HU003: Gestionar áreas.

El diseño se realizó con Bootstrap 5 para el formulario de crear y editar área, con ayuda del motor de plantillas Blade para simplificar el código mediante la creación de componentes y buscadores con Livewire.

```

<x-app-layout>
  <x-slot name="header">
    <div class="card">
      <div class="d-flex justify-content-between ">
        <h2 class="h4 font-weight-bold">
          Áreas
        </h2>
        <a href="{{ route('admin.areas.create') }}" class="btn btn-secondary btn-sm ">Nueva Área</a>
      </div>
    </div>
  </x-slot>
  @if (session('info'))
  <div class="alert alert-success" role="alert">
    <strong>{{ session('info') }}</strong>
  </div>
  @endif
  @livewire('admin.area-index')
</x-app-layout>

```

Figura 55. Diseño de vista lista de áreas.

Elaborado por: Morales Carlos, 2023.

```

<div class="card-header">
  <input type="text" wire:model="search" class="form-control" placeholder="Ingrese el nombre del área">
</div>

@if ($areas->count())
  <div class="card-body">
    <div class="table-responsive">
      <table class="table table-striped">
        <thead>
          <tr>
            <th scope="col">ID</th>
            <th scope="col">Nombre</th>
            <th scope="col">Descripción</th>
            <th colspan="2"></th>
          </tr>
        </thead>
        <tbody>
          @foreach ($areas as $area)
            <tr>
              <th scope="row">{{ $area->id }}</th>
              <td>{{ $area->name }}</td>
              <td>{{ $area->description }}</td>
              <td width="10px">
                <a class="btn btn-primary btn-sm"

```

Figura 56. Componente Livewire listar y buscar áreas.

Elaborado por: Morales Carlos, 2023.

```

public $search;
// resetea la propiedad buscar
public function updatingSearch()
{
  $this->resetPage();
}
protected $paginationTheme = "bootstrap";
public function render()
{
  $areas = Area::where('name', 'LIKE', '%' . $this->search . '%')
    ->orWhere('description', 'LIKE', '%' . $this->search . '%')
    ->paginate();
  return view('livewire.admin.area-index', compact('areas'));
}
protected $listeners = ['delete'];
public function delete(Area $area)
{
  $area->delete();
}
}

```

Figura 57. Función para listar, buscar y eliminar áreas.

Elaborado por: Morales Carlos, 2023.

```

<div class="card">
  <div class="card-body">
    <form @if ($area->id)
      action="{{ route('admin.areas.update',["area"=>$area]) }}"
    @else
      action="{{ route('admin.areas.store') }}"
    @endif method="POST">
    @if ($area->id)
      {{ method_field('PUT')}}
    @endif
    @csrf
    <label class="form-label">Nombre</label>
    <div class="input-group mb-3">
      <input type="text" class="form-control" placeholder="nombre" name="name" id="name"
        value="{{ old('name',$area->name) }}">
    </div>
    @error('name')
      <span class="text-warning">{{ $message }}</span>
    @enderror
  </div>
</div>

```

Figura 58. Formulario para crear y actualizar área.

Elaborado por: Morales Carlos, 2023.

```

public function store(AreaStoreRequest $request)
{
    $data= $request->all();
    Area::create($data);
    return redirect()->route('admin.areas.index')->with('info','El area se creó con éxito');
}

```

Figura 59. Función crear área.

Elaborado por: Morales Carlos, 2023.

```

*/
public function update(Request $request, Area $area)
{
    $data= $request->all();
    $area->fill($data);
    $area->save();
    return redirect()->route('admin.areas.index')->with('info','El area se actualizo con éxito');
}

```

Figura 60. Función actualizar área.

Elaborado por: Morales Carlos, 2023.

#### **HU004: Gestionar asignatura.**

El diseño se realizó con Bootstrap 5 para el formulario de crear y editar asignaturas, con ayuda del motor de plantillas Blade para simplificar el código mediante la creación de componentes y buscadores con Livewire.

```

<x-app-layout>
  <x-slot name="header">
    <div class="card">
      <div class="d-flex justify-content-between">
        <h2 class="h4 font-weight-bold">
          Listado Asignaturas
        </h2>
        <a href="{{ route('admin.asignaturas.create') }}" class="btn btn-secondary btn-sm ">Nueva Asignatura</a>
      </div>
    </div>
  </x-slot>
  @if (session('info'))
  <div class="alert alert-success" role="alert">
    <strong>{{ session('info') }}</strong>
  </div>
  @endif
  @livewire('admin.asignatura-index')
</x-app-layout>

```

Figura 61 Diseño de vista lista de asignaturas.

Elaborado por: Morales Carlos, 2023.

```

div>
  <div class="card">
    <div class="card-header">
      <input type="text" wire:model="search" class="form-control" placeholder="Ingrese el nombre">
    </div>
    @if ($asignaturas->count())
    <div class="card-body">
      <div class="table-responsive">
        <table class="table table-striped">
          <thead>
            <tr>
              <th scope="col">ID</th>
              <th scope="col">Nombre</th>
              <th scope="col">Area</th>
            </tr>
          </thead>
          <tbody>
            @foreach ($asignaturas as $asignatura)
            <tr>
              <th scope="row">{{ $asignatura->id }}</th>
              <td>{{ $asignatura->nombre }}</td>
              <td>{{ $asignatura->area->name }}</td>
              <td width="10px">
                <a class="btn btn-sm btn-primary"
                  href="{{ route('admin.asignaturas.edit', $asignatura) }}">Editar</a>
              </td>
            </tr>
            @endforeach
          </tbody>
        </table>
      </div>
    </div>
  </div>

```

Figura 62. Componente Livewire listar y buscar asignaturas.

Elaborado por: Morales Carlos, 2023.



```

public $search;
// resetea la propiedad buscar
public function updatingSearch()
{
    $this->resetPage();
}
protected $paginationTheme = "bootstrap";
public function render()
{
    $asignaturas = Asignatura::with('area')
        ->where('nombre', 'LIKE', '%' . $this->search . '%')->paginate();

    return view('livewire.admin.asignatura-index', compact('asignaturas'));
}

protected $listeners = ['delete'];
public function delete(Asignatura $asignatura)
{
    $asignatura->delete();
}

```

Figura 63. Función para listar, buscar y eliminar asignatura.

Elaborado por: Morales Carlos, 2023.

```

<x-app-layout>
  <x-slot name="header">
    <h2 class="h4 font-weight-bold">
      Crear asignatura
    </h2>
  </x-slot>

  <div class="card">
    <div class="card-body">
      <form>
        @if ($asignatura->id)
          action="{{ route('admin.asignaturas.update', ['asignatura'=>$asignatura]) }}"
        @else
          action="{{ route('admin.asignaturas.store') }}"
        @endif

        method="POST"
        @if ($asignatura->id)
          {{ method_field('PUT')}}
        @endif

        @csrf
        <label class="form-label">Nombre</label>
        <div class="input-group mb-3">
          <input type="text" class="form-control" placeholder="nombre" name="nombre" id="nombre"
            value="{{ old('name', $asignatura->nombre) }}">
        </div>
      </form>
    </div>
  </div>

```

Figura 64. Formulario para crear y actualizar asignaturas.

Elaborado por: Morales Carlos, 2023.

```

public function store(AsignaturaStoreRequest $request)
{
    $asignatura=Asignatura::create($request->validated());
    return redirect()->route('admin.asignaturas.index')->with('info','La asignatura se creó con éxito');
}

```

Figura 65. Función para crear asignatura.

Elaborado por: Morales Carlos, 2023.

```

7
public function update(Request $request, Asignatura $asignatura)
{
    $data= $request->all();
    $asignatura->fill($data);
    $asignatura->save();
    return redirect()->route('admin.asignaturas.index')->with('info','La asignatura se actualizo con éxito');
}

```

Figura 66. Función actualizar asignatura.

Elaborado por: Morales Carlos, 2023.

### HU005: Gestionar años lectivos.

El diseño se realizó con Bootstrap 5 para el formulario de crear y editar años lectivos, con ayuda del motor de plantillas Blade para simplificar el código mediante la creación de componentes y buscadores con Livewire.

```

<x-app-layout>
  <x-slot name="header">
    <div class="card">
      <div class="d-flex justify-content-between ">
        <h2 class="h4 font-weight-bold">
          Años Lectivos
        </h2>
        <a href="{{ route('admin.anios.create') }}" class="btn btn-secondary btn-sm ">Nuevo Año</a>
      </div>
    </div>
  </x-slot>
  @if (session('info'))
    <div class="alert alert-success" role="alert">
      <strong>{{ session('info') }}</strong>
    </div>
  @endif

```

Figura 67. Diseño de vista lista de años lectivos.

Elaborado por: Morales Carlos, 2023.

```

<div class="card">
  <div class="card-header">
    <input type="text" wire:model="search" class="form-control" placeholder="Ingrese el nombre del año lectivo">
  </div>

  @if ($anios->count())
    <div class="card-body">
      <div class="table-responsive">
        <table class="table table-striped">
          <thead>
            <tr>
              <th scope="col">ID</th>
              <th scope="col">Nombre</th>
              <th scope="col">Fec. inicio 1 Quimetre</th>
              <th colspan="2"></th>
            </tr>
          </thead>
          <tbody>
            @foreach ($anios as $anio)
              <tr>
                <th scope="row">{{ $anio->id }}</th>
                <td>{{ $anio->nombre }}</td>
                <td>{{ $anio->fecha_inicio }}</td>
                <td width="10px">
                  <a class="btn btn-primary btn-sm"
                    href="{{ route('admin.anios.edit', $anio) }}">Editar</a>
                </td>
              </tr>
            </tbody>
          </table>
        </div>
      </div>
    </div>
  </if>
</div>

```

Figura 68. Componente Livewire listar y buscar años lectivos.

Elaborado por: Morales Carlos, 2023.

```

public $search;
// resetea la propiedad buscar
public function updatingSearch()
{
    $this->resetPage();
}
protected $paginationTheme = "bootstrap";
public function render()
{
    $anios = AniosLectivo::where('nombre', 'LIKE', '%' . $this->search . '%')
        ->paginate();
    return view('livewire.admin.anio-index', compact('anios'));
}

protected $listeners = ['delete'];
public function delete(AniosLectivo $aniosLectivo)
{
    $aniosLectivo->delete();
}

```

Figura 69. Función para listar, buscar y eliminar años lectivos.

Elaborado por: Morales Carlos, 2023.

```

<x-app-layout>
  <x-slot name="header">
    <h2 class="h4 font-weight-bold">
      @if ($anhosLectivo->id)
        Editar año lectivo
      @else
        Crear año lectivo
      @endif
    </h2>
  </x-slot>

  <div class="card">
    <div class="card-body">
      <form
        @if ($anhosLectivo->id) action="{{ route('admin.anios.update', ['anio' => $anhosLectivo]) }}"
        @else
        action="{{ route('admin.anios.store') }}" @endif
        method="POST">
        @if ($anhosLectivo->id)
          {{ method_field('PUT') }}
        @endif
        @csrf

        <div class="container">
          <div class="row">
            <div class="col">
              <label class="form-label">Nombre</label>
              <div class="input-group mb-3">
                <input type="text" class="form-control" placeholder="nombre" name="nombre"

```

Figura 70. Formulario para crear y actualizar años lectivos.

Elaborado por: Morales Carlos, 2023.

```

public function store(Request $request)
{
    $data= $request->all();
    AniosLectivo::create($data);
    return redirect()->route('admin.anios.index')->with('info','El año se creó con éxito');
}

```

Figura 71. Función crear año lectivo.

Elaborado por: Morales Carlos, 2023.

```

public function update(Request $request, AniosLectivo $anio)
{
    $data= $request->all();
    $anio->fill($data);
    $anio->save();
    return redirect()->route('admin.anios.index')->with('info','El años Lectivo se actualizo con éxito');
}

```

Figura 72. Función actualizar año lectivo.

Elaborado por: Morales Carlos, 2023.

## HU006: Gestionar cursos.

El diseño se realizó con Bootstrap 5 para el formulario de crear y editar cursos, con ayuda del motor de plantillas Blade para simplificar el código mediante la creación de componentes y buscadores con Livewire.

```
<x-app-layout>
  <x-slot name="header">
    <div class="card">
      <div class="d-flex justify-content-between ">
        <h2 class="h4 font-weight-bold">
          Cursos
        </h2>
        <a href="{{ route('admin.cursos.create') }}" class="btn btn-secondary btn-sm ">Nuevo Curso</a>
      </div>
    </div>
  </x-slot>
  @if (session('info'))
  <div class="alert alert-success" role="alert">
    <strong>{{ session('info') }}</strong>
  </div>
  @endif
  @livewire('admin.curso-index')
</x-app-layout>
```

Figura 73. Diseño de vista lista de cursos.

Elaborado por: Morales Carlos, 2023.

```
<input type="text" wire:model="search" class="form-control" placeholder="Ingrese el nombre o grado">
</div>

@if ($cursos->count())
<div class="card-body">
  <div class="table-responsive">
    <table class="table table-striped">
      <thead>
        <tr>
          <th scope="col">ID</th>
          <th scope="col">Nombre</th>
          <th scope="col">Grado</th>
          <th scope="col">Paralelo</th>
          <th scope="col">Modalidad</th>
          <th colspan="2"></th>
        </tr>
      </thead>
      <tbody>
        @foreach ($cursos as $curso)
          <tr>
            <th scope="row">{{ $curso->id }}</th>
            <td>{{ $curso->nombre }}</td>
            <td>{{ $curso->grado }}</td>
            <td>{{ $curso->paralelo }}</td>
            <td>
              @if ($curso->modalidad == 'M')
                MATUTINA
              </td>
          </tr>
        @endforeach
      </tbody>
    </table>
  </div>
</div>
@endif
```

Figura 74. Componente Livewire listar y buscar cursos.

Elaborado por: Morales Carlos, 2023.

```

public $search;
// resetea la propiedad buscar
public function updatingSearch()
{
    $this->resetPage();
}
protected $paginationTheme = "bootstrap";
public function render()
{
    $cursos = Curso::where('nombre', 'LIKE', '%' . $this->search . '%')
        ->orWhere('grado', 'LIKE', '%' . $this->search . '%')
        ->paginate();
    return view('livewire.admin.curso-index', compact('cursos'));
}
protected $listeners = ['delete'];
public function delete(Curso $curso)
{
    $curso->delete();
}

```

Figura 75. Función para listar, buscar y eliminar cursos.

Elaborado por: Morales Carlos, 2023.

```

<div class="card">
    <div class="card-body">
        <form
            @if ($curso->id) action="{{ route('admin.cursos.update', ['curso' => $curso]) }}"
            @else
            action="{{ route('admin.cursos.store') }}" @endif
            method="POST">
            @if ($curso->id)
                {{ method_field('PUT') }}
            @endif
            @csrf
            <label class="form-label">Nombre</label>
            <div class="input-group mb-3">
                <input type="text" class="form-control" placeholder="nombre" name="nombre" id="nombre"
                    value="{{ old('nombre', $curso->nombre) }}">
            </div>
            @error('nombre')
                <span class="text-warning">{{ $message }}</span>
            @enderror

```

Figura 76. Formulario para crear y actualizar cursos.

Elaborado por: Morales Carlos, 2023.

```

*/
public function store(CursoStoreRequest $request)
{
    $data = $request->all();
    Curso::create($data);
    return redirect()->route('admin.cursos.index')->with('info', 'El curso se creó con éxito');
}

```

Figura 77. Función crear curso.

Elaborado por: Morales Carlos, 2023.

```
public function update(Request $request, Curso $curso)
{
    $data = $request->all();
    $curso->fill($data);
    $curso->save();
    return redirect()->route('admin.cursos.index')->with('info', 'El curso se actualizo con éxito');
}
```

Figura 78. Función actualizar curso.

Elaborado por: Morales Carlos, 2023.

### HU007: Gestionar matrículas de estudiantes.

El diseño se realizó con Bootstrap 5 para el formulario de crear, editar e importar matriculas de los estudiantes, con ayuda del motor de plantillas Blade para simplificar el código mediante la creación de componentes y buscadores con Livewire.

```
<x-app-layout>
  <x-slot name="header">
    <div class="card">
      <div class="d-flex justify-content-between ">
        <h2 class="h4 font-weight-bold">
          Listado Estudiantes
        </h2>
        <a href="{{ route('admin.estudiantes.create') }}" class="btn btn-secondary btn-sm ">Nuevo Estudiante</a>
      </div>
    </div>
  </x-slot>
  @if (session('info'))
    <div class="alert alert-success" role="alert">
      <strong>{{ session('info') }}</strong>
    </div>
  @endif

  @livewire('admin.estudiante-index')

  <div class="row justify-content-center mt-4">
    <div class="col-md-6">
      <div class="card">
        <div class="card-header">Importar estudiantes</div>
        <div class="card-body">
```

Figura 79. Diseño de vista lista de matrículas.

Elaborado por: Morales Carlos, 2023.

```

<input type="text" wire:model="search" class="form-control" placeholder="Ingrese el nombre"
/>div>
@if ($estudiantes->count())
<div class="table-responsive">
<div class="card-body">
<table class="table table-striped">
<thead>
<tr>
<th scope="col">ID</th>
<th scope="col">Nombres</th>
<th scope="col">Apellidos</th>
<th scope="col">Cedula</th>
<th scope="col">Curso</th>
<th scope="col">Grado</th>
</tr>
</thead>
<tbody>
<tr>
<th scope="row">{{ $estudiante->id }}</th>
<td>{{ $estudiante->nombres }}</td>
<td>{{ $estudiante->apellidos }}</td>
<td>{{ $estudiante->cedula }}</td>

```

Figura 80. Componente Livewire listar y buscar matriculas.

Elaborado por: Morales Carlos, 2023.

```

public $search;

// resetea la propiedad buscar
public function updatingSearch(){
    $this->resetPage();
}
protected $paginationTheme="bootstrap";
protected $listeners = ['delete'];
public function render()
{
    $estudiantes=Estudiante::with('curso')
->where('nombres','LIKE','%'. $this->search . '%')
->orWhere('apellidos','LIKE','%'. $this->search . '%')->paginate();
    return view('livewire.admin.estudiante-index',compact('estudiantes'));
}

public function delete(Estudiante $estudiante)
{
    $estudiante->delete();
}

```

Figura 81. Función para listar, buscar y eliminar matriculas.

Elaborado por: Morales Carlos, 2023.



```

<div class="card">
  <div class="card-body">
    <form
      @if ($estudiante->id)
        action="{{ route('admin.estudiantes.update',["estudiante"=>$estudiante]) }}"
      @else
        action="{{ route('admin.estudiantes.store') }}"
      @endif
      method="POST">
      @if ($estudiante->id)
        {{ method_field('PUT')}}
      @endif
      @csrf
      <div class="row">
        <div class="col-md-6">
          <label class="form-label">Nombres</label>
          <div class="input-group mb-3">
            <input type="text" class="form-control" placeholder="nombres" name="nombres" id="nombres"
              value="{{ old('nombres',$estudiante->nombres) }}">
          </div>
          @error('nombres')
            <span class="text-warning">{{ $message }}</span>
          @enderror
          <label class="form-label">Cedula</label>

```

Figura 82. Formulario para crear y actualizar matricula.

Elaborado por: Morales Carlos, 2023.

```

public function store(EstudianteStoreRequest $request)
{
    // Primero, verificamos si el usuario ya existe
    $usuario = User::where('email', $request->id_usuario)->first();
    if (!$usuario) {
        // Si el usuario no existe, lo creamos
        $usuario = User::create([
            'name' => $request->nombres . ' ' . $request->apellidos,
            'email' => $request->id_usuario,
            'password' => bcrypt($request->cedula), // Esta es una contraseña temporal
        ]);
        $usuario->assignRole('Estudiante');

        // Ahora, creamos el estudiante y lo asociamos con el usuario
        $estudiante = Estudiante::create([
            'nombres' => $request->nombres,
            'apellidos' => $request->apellidos,
            'cedula' => $request->cedula,
            'id_curso' => $request->id_curso,
            'id_usuario' => $usuario->id,
        ]);
        // $estudiante = Estudiante::create($request->validated());
        return redirect()->route('admin.estudiantes.index')->with('info', 'El estudiante se creó con éxito');
    }
}

```

Figura 83. Función crear matricula.

Elaborado por: Morales Carlos, 2023.

```

public function generarexcel(Request $request)
{
    $file = $request->file('import_file');
    Excel::import(new UsersImport, $file);
    Excel::import(new EstudiantesImport, $file);
    return redirect()->route('admin.estudiantes.index')->with('info', 'Los estudiante se importaron con éxito');
}

```

Figura 84. Función importar matriculas desde Excel.

Elaborado por: Morales Carlos, 2023.

```

public function update(Request $request, Estudiante $estudiante)
{
    $data = $request->all();
    $estudiante->fill($data);
    $estudiante->save();
    return redirect()->route('admin.estudiantes.index')->with('info', 'El estudiante se actualizo con éxito');
}

```

Figura 85. Función actualizar matricula.

Elaborado por: Morales Carlos, 2023.

```

private $nombre = [];
public function model(array $row)
{
    $this->nombre = explode(" ", $row['nombres']);
    $estudiante = Estudiante::where('id_curso', '=', $this-> cursos[$row['curso']])->where('id_usuario', '=', $this-> usuario[$row['correo']])->first();
    if (!$estudiante) {
        return new Estudiante([
            'nombres' => $this->nombre[2] . ' ' . $this->nombre[3],
            'apellidos' => $this->nombre[0] . ' ' . $this->nombre[1],
            'cedula' => $row['cedula'],
            'id_curso' => $this->cursos[$row['curso']],
            'id_usuario' => $this->usuario[$row['correo']],
        ]);
    }
}

public function rules(): array
{
    return [
        '*.nombres' => [
            'required'
        ],
        '*.cedula' => [
            'required'
        ]
    ];
}

```

Figura 86. Importar matriculas Excel.

Elaborado por: Morales Carlos, 2023.

### **HU008: Gestionar asignación de asignaturas a profesores.**

El diseño se realizó con Bootstrap 5 para el formulario de crear y editar la asignación de asignaturas a profesores, con ayuda del motor de plantillas Blade para simplificar el código mediante la creación de componentes y buscadores con Livewire.

```

<x-app-layout>
  <x-slot name="header">
    <div class="card">
      <div class="d-flex justify-content-between ">
        <h2 class="h4 font-weight-bold">
          Asignación de asignaturas
        </h2>
        <a href="{{ route('admin.detalleasignaturas.create') }}" class="btn btn-secondary btn-sm ">Asignar Asignat
      </div>
    </div>
  </x-slot>
  @if (session('info'))
    <div class="alert alert-success" role="alert">
      <strong>{{ session('info') }}</strong>
    </div>
  @endif

  @livewire('admin.detalle-asignatura-index')
</x-app-layout>

```

Figura 87. Diseño de vista lista de asignación de asignaturas a profesor.

Elaborado por: Morales Carlos, 2023.

```

<div class="card-header">
  <input type="text" wire:model="search" class="form-control" placeholder="Ingrese el nombre">
</div>

@if ($detalleasignaturas->count())
  <div class="card-body">
    <div class="table-responsive">
      <table class="table table-striped">
        <thead>
          <tr>
            <th scope="col">ID</th>
            <th scope="col">Profesor</th>
            <th scope="col">Curso</th>
            <th scope="col">Asignatura</th>
          </tr>
        </thead>
        <tbody>
          @foreach ($detalleasignaturas as $detalleasignatura)
            <tr>
              <th scope="row">{{ $detalleasignatura->id }}</th>
              <td>{{ $detalleasignatura->profesor->name }}</td>
              <td>{{ $detalleasignatura->curso->nombre }}</td>
            </tr>
          @endforeach
        </tbody>
      </table>
    </div>
  </div>
</if>

```

Figura 88. Componente Livewire listar y buscar asignación de signaturas a profesores.

Elaborado por: Morales Carlos, 2023.

```

public $search;
// resetea la propiedad buscar
public function updatingSearch()
{
    $this->resetPage();
}
protected $paginationTheme = "bootstrap";
public function render()
{
    $detalleasignaturas = DetalleAsignatura::with('profesor')
        ->with('curso')->with('asignatura')->paginate();
    return view('livewire.admin.detalle-asignatura-index', compact('detalleasignaturas'));
}
protected $listeners = ['delete'];
public function delete(DetalleAsignatura $detalleAsignatura)
{
    $detalleAsignatura->delete();
}
}

```

Figura 89. Función para listar, buscar y eliminar asignación de asignaturas.

Elaborado por: Morales Carlos, 2023.

```

<div class="card">
  <div class="card-body">
    <form>
      @if ($detalleasignatura->id) action="{{ route('admin.detalleasignaturas.update', ['detalleasignatura' => $detalleasignatura->id]) }}" @endif
      @else
      action="{{ route('admin.detalleasignaturas.store') }}" @endif
      method="POST"
      @if ($detalleasignatura->id)
      {{ method_field('PUT') }}
      @endif
      @csrf
      <label class="form-label">Curso</label>
      <div class="input-group mb-3">
        <select class="form-select combo" aria-label="Default select example" id="id_curso" name="id_curso">
          <option selected>Elige uno</option>
          @foreach ($cursos as $curso)
            <option value="{{ $curso->id }}"
              {{ $detalleasignatura->id_curso == $curso->id ? 'selected' : '' }}>{{ $curso->nombre }}
          </option>
          @endforeach
        </select>
      </div>
      @error('id_curso')
      <span class="text-warning">{{ $message }}</span>
      <br>
    </form>
  </div>
</div>

```

Figura 90. Formulario para crear y actualizar asignación de asignaturas.

Elaborado por: Morales Carlos, 2023.

```

public function store(DetalleAsignaturaStoreRequest $request)
{
    $asignatura=DetalleAsignatura::create($request->validated());
    return redirect()->route('admin.detalleasignaturas.index')->with('info','La asignatura fue asignada');
}

```

Figura 91. Función crear asignación de asignatura.

Elaborado por: Morales Carlos, 2023.

```

public function update(Request $request, DetalleAsignatura $detalleasignatura)
{
    $data= $request->all();
    $detalleasignatura->fill($data);
    $detalleasignatura->save();
    return redirect()->route('admin.detalleasignaturas.index')->with('info','La asignación se actualizo con éxito');
}

```

Figura 92. Función actualizar asignación de asignatura.

Elaborado por: Morales Carlos, 2023.

### HU009: Registrar calificaciones.

El diseño se realizó con Bootstrap 5 para el formulario de crear y editar insumos y posterior asignar la calificación correspondiente, con ayuda del motor de plantillas Blade para simplificar el código mediante la creación de componentes y buscadores con Livewire.

```

<x-app-layout>
  <x-slot name="header">
    <div class="card">
      <div class="d-flex justify-content-between ">
        <h2 class="h4 font-weight-bold">
          Talleres
        </h2>
        <a href="{{ route('crearTaller',$id) }}" class="btn btn-secondary btn-sm ">Nueva Taller</a>
      </div>
    </div>
  </x-slot>
  @if (session('info'))
  <div class="alert alert-success" role="alert">
    <strong>{{ session('info') }}</strong>
  </div>
  @endif
  @livewire('profesor.taller-index',['id'=>$id])
</x-app-layout>

```

Figura 93. Diseño de vista lista de insumos.

Elaborado por: Morales Carlos, 2023.

```

<thead>
  <tr>
    <th scope="col">ID</th>
    <th scope="col">Nombre</th>
    <th scope="col">Descripción</th>
    <th scope="col">Fecha</th>
    <th colspan="3"></th>
  </tr>
</thead>
<tbody>
  @foreach ($taller as $taller)
    <tr>
      <th scope="row">{{ $taller->id }}</th>
      <td>{{ $taller->nombre }}</td>
      <td>{{ $taller->descripcion }}</td>
      <td>{{ $taller->fecha }}</td>
      <td width="10px">
        <a class="btn btn-primary btn-sm"
          href="{{ route('editarTaller', $taller) }}">Editar</a>
      </td>
    </tr>
  @endforeach

```

Figura 94. Componente Livewire listar y buscar insumo.

Elaborado por: Morales Carlos, 2023.

```

<div class="card">
  <div class="card-body">
    <form>
      @if ($taller->id) action="{{ route('actualizarTaller', ['taller' => $taller]) }}"
    @else
      action="{{ route('profesor.talleres.store') }}" @endif
      method="POST">
      @if ($taller->id)
        {{ method_field('PUT') }}
      @endif
      @csrf
      <label class="form-label">Nombre</label>
      <div class="input-group mb-3">
        <input type="text" class="form-control" placeholder="nombre" name="nombre" id="nombre"
          value="{{ old('nombre', $taller->nombre) }}">
      </div>
      @error('nombre')
        <span class="text-warning">{{ $message }}</span>
      @enderror
      <label class="form-label">Descripción</label>
      <div class="input-group mb-3">

```

Figura 95. Formulario para crear y actualizar insumo.

Elaborado por: Morales Carlos, 2023.

```

<div class="card">
  <form action="{{ route('profesor.talleresdetalle.store') }}" method="POST">
    <div class="table-responsive">
      <table class="table table-striped">
        <thead>
          <tr>
            <th scope="col">Estudiantes</th>
            <th scope="col" width="1px"></th>
            <th scope="col" width="1px"></th>
            <th scope="col">Nota</th>
            <th scope="col">Descripcion</th>
          </tr>
        </thead>
        <tbody>
          @foreach ($estudiantesArray as $estudiante)
            <tr>
              <th scope="row">{{ $estudiante->apellidos . ' ' . $estudiante->nombres }}
            </th>
            <td><input type="hidden" id="data_{{ $estudiante['id'] }}" name="data_{{ $estudiante['id'] }}" value="{{ $taller->id }}" width="1px" /></td>
          </tr>
        </tbody>
      </table>
    </div>
  </form>

```

Figura 96. Formulario registrar calificaciones.

Elaborado por: Morales Carlos, 2023.

```

<div class="card-body">
  <form @if ($det->id)
    action="{{ route('profesor.talleresdetalle.update', ['talleresdetalle'=>$det]) }}"
  @else
    action="{{ route('profesor.talleresdetalle.store') }}"
  @endif method="POST">
    @if ($det->id)
      {{ method_field('PUT')}}
    @endif
    @csrf

    <div class="input-group mb-3">
      <input type="hidden" class="form-control" placeholder="" name="id_taller" id="id_taller"
        value="{{ old('id_taller', $det->id_taller) }}" />
    </div>
    @error('id_taller')
      <span class="text-warning">{{ $message }}</span>
    @enderror

    <div class="input-group mb-3">
      <input type="hidden" class="form-control" placeholder="id_estudiante" name="id_estudiante"
        id="id_estudiante" value="{{ old('id_estudiante', $det->id_estudiante) }}" />
    </div>
    @error('id_taller')

```

Figura 97. Formulario para actualizar calificación.

Elaborado por: Morales Carlos, 2023.

```

public function store(TallerStoreRequest $request)
{
    $taller = Taller::create($request->validated());
    return redirect()->route('profesor.adicionarestudiantes', ['taller' => $taller->id])->with('info', 'El taller
}

```

Figura 98. Función crear insumo.

Elaborado por: Morales Carlos, 2023.

```

public function update(Request $request, Taller $taller)
{
    $data = $request->all();
    $taller->fill($data);
    $taller->save();
    return redirect()->route('TallerId', $taller->id_detalle_asignatura)->with('info', 'El taller se actualizo con éxito');
}

```

Figura 99. Función actualizar insumo.

Elaborado por: Morales Carlos, 2023.

```

public function store(Request $request)
{
    $var = $request->get('_token');

    foreach ($request->all() as $req) {
        if ($var != $req) {
            DetalleTaller::create([
                'id_taller' => $req['id_taller'],
                'id_estudiante' => $req['id_estudiante'],
                'nota' => $req['nota'],
                'descripcion' => $req['descripcion']
            ]);
        }
    }

    return redirect()->route('dashboard')->with('info', 'La nota se registro con éxito');
}

```

Figura 100. Función crear calificación de estudiantes.

Elaborado por: Morales Carlos, 2023.

```

public function update(Request $request, DetalleTaller $tallerdetalle)
{
    $data = $request->all();
    $tallerdetalle->fill($data);
    $tallerdetalle->save();
    return redirect()->route('profesor.adicionarestudiantes', $tallerdetalle->id_taller)->with('info', 'La nota se');
}

```

Figura 101. Función actualizar calificación estudiante.

Elaborado por: Morales Carlos, 2023.

### **HU010: Registrar asistencia.**

El diseño se realizó con Bootstrap 5 para el formulario de crear y editar la asistencia de los estudiantes. Se utilizó el motor de plantillas Blade para simplificar el código a través de la creación de componentes y se utilizaron buscadores con Livewire.



```

<x-app-layout>
  <x-slot name="header">
    <div class="card">
      <div class="d-flex justify-content-between">
        <h2 class="h4 font-weight-bold">
          Horarios
        </h2>
        <a href="{{ route('profesor.horarios.create',$id) }}" class="btn btn-secondary btn-sm">Nueva
      </div>
    </div>
  </x-slot>
  @if (session('info'))
  <div class="alert alert-success" role="alert">
    <strong>{{ session('info') }}</strong>
  </div>
  @endif
  @livewire('profesor.horario-index',['id'=>$id])
</x-app-layout>

```

Figura 102. Diseño de vista listas horario de asistencias.

Elaborado por: Morales Carlos, 2023.

```

<input type="text" wire:model="search" class="form-control" placeholder="Ingrese la fecha">
</div>
<div class="card">
  @if ($horarios->count())
  <div class="card-body">
    <div class="table-responsive">
      <table class="table table-striped">
        <thead>
          <tr>
            <th scope="col"></th>
            <th scope="col">fecha</th>
            <th scope="col">horas</th>
            <th colspan="3"></th>
          </tr>
        </thead>
        <tbody>
          @foreach ($horarios as $horario)
            <tr>
              @if ($horario->fecha == $date) style="background-color: rgb(102, 178, 255)"
              <th scope="row"><input type="checkbox" value="{{ $horario->id }}"
                id="{{ $horario->id }}" name="borraRegistros[]">
              </th>
              <td>{{ $horario->fecha }}</td>
              <td>{{ $horario->hora }}</td>
              <td width="10px">
                <a class="btn btn-warning btn-sm"
                  href="{{ route('profesor.horarios.adicionarestudiantes', $horario
                    Asistencia)</a>
              </td>
              <td width="10px">

```

Figura 103. Componente Livewire listar y buscar horarios de asistencia.

Elaborado por: Morales Carlos, 2023.

```

public $search;

public $selectedItems = [];

// resetea la propiedad buscar
public function updatingSearch()
{
    $this->resetPage();
}

protected $paginationTheme = "bootstrap";

public $valor;
public function mount($id)
{
    $this->valor = $id;
}

public function render()
{
    $valor = $this->valor;

    $horarios = Horario::whereDate('fecha', '>=', now()->format('Y-m-d'))
        ->where('id_detalle_asignatura', '=', $valor)
        ->where('fecha', 'LIKE', '%' . $this->search . '%')
        ->orderBy('fecha', 'asc') // Ordenar por fecha de manera ascendente
    ;
}

```

Figura 104. Función para listar, buscar y eliminar horarios de asistencia.

Elaborado por: Morales Carlos, 2023.

```

        <div class="card">
            <div class="card-body">
                <form
                    @if ($horario->id) action="{{ route('profesor.horarios.update', ['horario' => $horario]) }}"
                    @else
                    action="{{ route('profesor.horarios.store') }}" @endif
                    method="POST">
                    @if ($horario->id)
                        {{ method_field('PUT') }}
                    @endif
                    @csrf
                    <div class="row">
                        <div class="col">
                            <label class="form-label">Fecha inicio</label>
                            <div class="input-group mb-3">
                                <input type="date" class="form-control" name="fecha" id="fecha"

```

Figura 105. Formulario para crear y actualizar horario de asistencia.

Elaborado por: Morales Carlos, 2023.

```

<x-slot name="header">
  <h2 class="h4 font-weight-bold">
    Editar asistencia
  </h2>
</x-slot>

<div class="card">
  <div class="card-body">
    <form
      @if ($det->id) action="{{ route('profesor.asistencias.update', ['horario' => $det]) }}"
      @else
      action="{{ route('profesor.asistencias.store') }}" @endif
      method="POST">
      @if ($det->id)
        {{ method_field('PUT') }}
      @endif
      @csrf

      <div class="input-group mb-3">
        <input type="hidden" class="form-control" placeholder="" name="id_horarios" id="id_horarios"
          value="{{ old('id_horarios', $det->id_horarios) }}">
        </div>
      @error('id_horarios')
        <span class="text-warning">{{ $message }}</span>
      @enderror
    </form>
  </div>
</div>

```

Figura 106. Formulario actualizar asistencia.

Elaborado por: Morales Carlos, 2023.

```

public function deleteSelected()
{
    foreach ($this->selectedItems as $itemId) {
        // Eliminar el registro de la base de datos
        Horario::find($itemId)->delete();
    }

    // Limpiar la selección
    $this->selectedItems = [];
}

```

Figura 107. Función para eliminar múltiples registros de asistencia.

Elaborado por: Morales Carlos, 2023.

```

public function store(Request $request)
{
    $fecha_inicio = $request->get('fecha');
    $fecha_fin = $request->get('fechaf');
    $hora_inicio = $request->get('hora_inicio');
    $hora_fin = $request->get('hora_fin');
    $diaSeleciondo = $request->get('dia');
    $id_asignatura = $request->get('id_detalle_asignatura');

    $nHoras = date_create($hora_inicio);
    $nHorasf = date_create($hora_fin);
    $aux = date_diff($nHoras, $nHorasf);
    $minutes = $aux->days * 24 * 60;
    $minutes += $aux->h * 60;
    $minutes += $aux->i;
    $numeroHoras = abs($minutes / 40);

    $fechaInicio = strtotime($fecha_inicio);
    $fechaFin = strtotime($fecha_fin);
    //Recorro las fechas y con la función strtotime obtengo los lunes
    for ($i = $fechaInicio; $i <= $fechaFin; $i += 86400) {
        //Sacar el día de la semana con el modificador N de la función date
        $dia = date('N', $i);
        if ($dia == $diaSeleciondo) {
            Horario::create([
                'hora_inicio' => $request->get('hora_inicio'), 'hora_fin' => $request->get('hora_fin'), 'fec
            ]);
        }
    }
}

```

Figura 108. Función crear horarios de asistencia.

Elaborado por: Morales Carlos, 2023.

```

public function store(Request $request)
{
    $var = $request->get('_token');
    foreach ($request->all() as $req) {
        if ($var != $req) {
            Asistencia::create([
                'id_horarios' => $req['id_horarios'],
                'id_estudiante' => $req['id_estudiante'],
                'estado' => $req['estado']
            ]);
        }
    }
}

```

Figura 109. Función crear asistencia.

Elaborado por: Morales Carlos, 2023.

```

public function update(Request $request, Asistencia $horario)
{
    $data = $request->all();
    $horario->fill($data);
    $horario->save();
    return redirect()->route('profesor.horarios.adicionarestudiantes', $horario->id_horarios)->with('info',
}

```

Figura 110. Función actualizar asistencia.

Elaborado por: Morales Carlos, 2023.

## HU011: Ver calificaciones actuales de estudiante.

El diseño se realizó con Bootstrap 5 para el formulario de ver las calificaciones actuales de los estudiantes. Se utilizó el motor de plantillas Blade para simplificar el código a través de la creación de componentes y se utilizaron buscadores con Livewire.

```
<x-app-layout>
  <x-slot name="header">
    <div class="card">
      <div class="d-flex justify-content-between">
        <h2 class="h4 font-weight-bold">
          Calificaciones {{ $estudiante->apellidos . ' ' . $estudiante->nombres }}
        </h2>
      </div>
    </div>
  </x-slot>

  <div class="card">
    @if ($talleres->count())
      <h3>Talleres</h3>
      <div class="card-body">
        <div class="table-responsive">
          <table class="table table-striped">
            <thead>
              <tr>
                <th scope="col">Nombre</th>
                <th scope="col">Descripción</th>
                <th scope="col">Fecha</th>
                <th scope="col">Nota</th>
              </tr>
            </thead>
            <tbody>
              @foreach ($talleres as $taller)
                <tr>
                  <th>{{ $taller->nombre }}</th>
                  <th>{{ $taller->descripcion }}</th>
                  <th>{{ $taller->fecha }}</th>
                  <th>{{ $taller->nota }}</th>
                </tr>
              @endforeach
            </tbody>
          </table>
        </div>
      </div>
    @endif
  </div>
</x-app-layout>
```

Figura 111. Diseño de vista de calificaciones de insumo.

Elaborado por: Morales Carlos, 2023.

```
public function verAsignaturas($estudiante)
{
    $estu = Estudiante::where('id', '=', $estudiante->first());
    $rolEstudinte = User::whereHas('roles', function ($query) {
        $query->where('name', 'Estudiante');
    }->where('id', $estu->id_usuario)->first();
    if (!$rolEstudinte) {
    }

    $listaAsignaturas = DB::table('cursos as c')
        ->select('da.id', 'a.nombre', 'e.id as idest', 'e.apellidos', 'e.nombres')
        ->join('años_lectivos as al', 'c.id_año', '=', 'al.id')
        ->join('estudiantes as e', 'c.id', '=', 'e.id_curso')
        ->join('detalle_asignaturas as da', 'c.id', '=', 'da.id_curso')
        ->join('asignaturas as a', 'da.id_asignatura', '=', 'a.id')
        ->where('e.id', '=', $estudiante)
        ->where('al.estado', '=', 'A')
        ->get();

    return view('profesores.consulta.listaasignaturas', compact('listaAsignaturas'));
}
```

Figura 112. Función ver asignaturas.

Elaborado por: Morales Carlos, 2023.

```

public function verAsignaturasNotas($estudiante, $asignatura)
{
    $estudiante = Estudiante::where('id', $estudiante)->first();

    $talleres = DB::table('talleres as t')
        ->join('detalle_talleres as dt', 't.id', '=', 'dt.id_taller')
        ->join('detalle_asignaturas as da', 't.id_detalle_asignatura', '=', 'da.id')
        ->join('asignaturas as a', 'da.id_asignatura', '=', 'a.id')
        ->select('t.nombre', 't.descripcion', 't.fecha', 'dt.nota')
        ->where('dt.id_estudiante', '=', $estudiante->id)
        ->where('da.id', '=', $asignatura)
        ->paginate();

    $talleresg = DB::table('talleres_grupales as t')
        ->select('t.nombre', 't.descripcion', 't.fecha', 'dt.nota')
        ->join('detalle_talleres_grupales as dt', 't.id', '=', 'dt.id_taller')
        ->join('detalle_asignaturas as da', 't.id_detalle_asignatura', '=', 'da.id')
        ->join('asignaturas as a', 'da.id_asignatura', '=', 'a.id')
        ->where('dt.id_estudiante', '=', $estudiante->id)
        ->paginate();
}

```

Figura 113. Función ver calificaciones de insumos.

Elaborado por: Morales Carlos, 2023.

### HU012: Descargar reporte de calificaciones.

El diseño se realizó con Bootstrap 5 para el formulario descargar reporte de calificaciones por un curso y paralelo en específico en formato Excel en dos versiones una extendida y una resumida. Se utilizó el motor de plantillas Blade para simplificar el código a través de la creación de componentes y se utilizaron listas con Livewire para filtrar los cursos.

```

class CalificacionesExport implements FromView, ShouldAutoSize, WithStyles, WithDrawings
{
    protected $id;

    function __construct($id)
    {
        $this->id = $id;
    }
    /**
     * @return \Illuminate\Support\Collection
     */
    public function view(): View
    {
        $detalleAsignatura = DetalleAsignatura::where('id', '=', $this->id)->first();
        $aniolectivo = DB::select("SELECT a.nombre,a.fecha_inicio AS fipq,a.fecha_mediapq AS ffpp,a.fecha_finpq
        a.fecha_mediasq AS fisq,a.fecha_inicio AS ffsq,a.fecha_fin AS ffsq
        FROM años_lectivos a
        JOIN cursos c ON a.id=c.id_año
        WHERE c.id={$detalleAsignatura->id_curso}");
        $auxAnio = $aniolectivo[0];
    }
}

```

Figura 114. Función calificaciones por asignatura Excel.

Elaborado por: Morales Carlos, 2023.

```

public function styles(Worksheet $sheet)
{
    return [

        // Style the first row as bold text.

        1 => ['font' => ['size' => 20, 'bold' => true]],

        // Styling a specific cell by coordinate.
        'B2' => ['font' => ['italic' => true]],

        // Styling an entire column.
        // 'C' => ['font' => ['size' => 11, 'color' => ['rgb' => 'E11414']],
        9 => ['fill' => ['fillType' => Fill::FILL_SOLID, 'color' => ['rgb' => 'FAB30B']], 'font' => ['size' => 10],
        10 => ['fill' => ['fillType' => Fill::FILL_SOLID, 'color' => ['rgb' => 'F7EE2E']], 'font' => ['size' => 10]
    ];
}

```

Figura 115. Función ver calificaciones de insumos.

Elaborado por: Morales Carlos, 2023.

```

public function drawings()
{
    $drawing2 = new Drawing();
    $drawing2->setName('Logo');
    $drawing2->setDescription('escudo UEHG');
    $drawing2->setPath(public_path('/image/Escudo_uehg.png'));
    $drawing2->setHeight(80);
    $drawing2->setCoordinates('D1');

    return [$drawing2];
}

```

Figura 116. Función dibujar imagen en Excel.

Elaborado por: Morales Carlos, 2023.

### **HU013: Descargas reporte de asistencia.**

El diseño se realizó con Bootstrap 5 para el formulario descargar reporte de asistencia por un curso y paralelo en específico en dos formatos Excel y Documento en formato PDF los estudiantes con un porcentaje menor al 10% se marcará en rojo. Se utilizó el motor de plantillas Blade para simplificar el código a través de la creación de componentes y se utilizaron listas con Livewire para filtrar los cursos.

```

public function styles(Worksheet $sheet)
{
    return [
        // Style the first row as bold text.
        1 => ['font' => ['size' => 20, 'bold' => true]],

        // Styling a specific cell by coordinate.
        'B2' => ['font' => ['italic' => true]],

        // Styling an entire column.
        'C' => ['font' => ['size' => 11, 'color' => ['rgb' => '1C118E']],

        7 => ['fill' => ['fillType' => Fill::FILL_SOLID, 'color' => ['rgb' => '2668EE']]
    ];
}

```

Figura 117. Función estilos reporte asistencia.

Elaborado por: Morales Carlos, 2023.

```

protected $id;

function __construct($id)
{
    $this->id = $id;
}

public function view(): View
{
    $queryAsistencia = "SELECT e.id,a.estado,COUNT( a.estado), SUM(h.hora) as hora FROM estudiantes e
JOIN asistencias a ON e.id=a.id_estudiante
JOIN horarios h ON a.id_horarios=h.id
WHERE h.id_detalle_asignatura=?
AND a.estado=?
AND h.fecha BETWEEN ? AND ?
GROUP BY a.estado,e.id,h.hora
ORDER BY e.apellidos";
    $detalleAsignatura = DetalleAsignatura::where('id', '=', $this->id)->first();
    $anioLectivo = DB::select("SELECT a.nombre,a.fecha_inicio AS fipq,a.fecha_medio AS ffpq,a.fecha_fin AS ffpq
a.fecha_medio AS fisq,a.fecha_inicio AS fispq,a.fecha_fin AS ffsq
FROM años_lectivos a
JOIN cursos c ON a.id=c.id_anio
WHERE c.id={$detalleAsignatura->id_curso}");
    $auxAnio = $anioLectivo[0];

    $listaEstudiantes = DB::select("SELECT e.id , e.apellidos, e.nombres FROM estudiantes e
WHERE e.id_curso=(SELECT c.id from cursos c
WHERE c.id =(SELECT d.id curso FROM detalle asignaturas d

```

Figura 118. Función generar reporte asistencia.

Elaborado por: Morales Carlos, 2023.

#### **HU014: Consultar calificaciones de estudiante.**

El diseño se realizó con Bootstrap 5 para el formulario para consultar las calificaciones de un estudiante. Se utilizó el motor de plantillas Blade para simplificar el código a través de la creación de componentes y se utilizó un buscador con Livewire para filtrar la información.



```

public function verAsignaturasNotas($estudiante, $asignatura)
{
    $estudiante = Estudiante::where('id', $estudiante)->first();

    $talleres = DB::table('talleres as t')
        ->join('detalle_talleres as dt', 't.id', '=', 'dt.id_taller')
        ->join('detalle_asignaturas as da', 't.id_detalle_asignatura', '=', 'da.id')
        ->join('asignaturas as a', 'da.id_asignatura', '=', 'a.id')
        ->select('t.nombre', 't.descripcion', 't.fecha', 'dt.nota')
        ->where('dt.id_estudiante', '=', $estudiante->id)
        ->where('da.id', '=', $asignatura)
        ->paginate();

    $talleresg = DB::table('talleres_grupales as t')
        ->select('t.nombre', 't.descripcion', 't.fecha', 'dt.nota')
        ->join('detalle_talleres_grupales as dt', 't.id', '=', 'dt.id_taller')
        ->join('detalle_asignaturas as da', 't.id_detalle_asignatura', '=', 'da.id')
        ->join('asignaturas as a', 'da.id_asignatura', '=', 'a.id')
        ->where('dt.id_estudiante', '=', $estudiante->id)
        ->paginate();
}

```

Figura 119. Función ver calificaciones de insumos.

Elaborado por: Morales Carlos, 2023.

### HU015: Auditoria por cambio o borrado de los registros de calificaciones.

El diseño se realizó con Bootstrap 5 para el formulario para registrar el cambio de calificaciones o eliminación de calificación por parte del profesor. Se utilizó el motor de plantillas Blade para simplificar el código a través de la creación de componentes.

```

'resolvers' => [
    'ip_address' => OwenIt\Auditing\Resolvers\IpAddressResolver::class,
    'user_agent' => OwenIt\Auditing\Resolvers\UserAgentResolver::class,
    'url' => OwenIt\Auditing\Resolvers\UrlResolver::class,
],

/*
|-----
| Audit Events
|-----
|
| The Eloquent events that trigger an Audit.
|
*/

'events' => [
    //'created',
    'updated',
    'deleted',
    //'restored'
],

```

Figura 120. Configuración de los eventos auditar.

Elaborado por: Morales Carlos, 2023.

### HU016: Notificaciones de estudiantes con asistencia insuficiente.

El diseño se realizó con Bootstrap 5 para el formulario para leer las notificaciones de estudiantes con una asistencia insuficiente. Se utilizó el motor de plantillas Blade para simplificar el código a través de la creación de componentes.

```
$fechainicio1P = Carbon::createFromFormat('Y-m-d', $auxAnio->ffpq);
$fechainicio1P->subDays(7)->format('Y-m-d');
$fechainicio2P = Carbon::createFromFormat('Y-m-d', $auxAnio->ffsq);
$fechainicio2P->subDays(7)->format('Y-m-d');
$fecha = Carbon::now();

if ($fecha->between($fechainicio1P, $auxAnio->ffpq)|| $fecha->between($fechainicio2P,$auxAnio->ffsq)) {
    $message = [
        'sender_id' => $asignatura['id_curso'],
        'recipient_id' => Auth::id(),
        'body' => $this->calcularAsistencia($hcurso['id_detalle_asignatura']),
    ];

    $recipient = User::find(Auth::id());
    $recipient->notify(new MessageSent($message));
}
```

Figura 121. Función notificación con estudiantes con inasistencia superior 10%.

Elaborado por: Morales Carlos, 2023.

### HU018: Generar boleta de calificaciones.

El diseño se realizó con Bootstrap 5 para el formulario descargar documento en formato PDF con una boleta con las asignaturas y sus promedios por parcial y quimestre. Se utilizó el motor de plantillas Blade para simplificar el código a través de la creación de componentes.

```
function promedioFinal($listaEstudiantes)
{
    foreach ($listaEstudiantes as $aux) {
        $contadorPQ = 0;
        $examenPP = 0;
        $proyectoPP = 0;
        $refuerzoPP = 0;
        $contadorSQ = 0;
        $examenSP = 0;
        $proyectoSP = 0;
        $refuerzoSP = 0;

        if ($aux->PexamenesSPPQ != '-') {
            $examenPP = $aux->PexamenesSPPQ;
            $contadorPQ++;
        }
        if ($aux->PproyectoSPPQ != '-') {
            $proyectoPP = $aux->PproyectoSPPQ;
            $contadorPQ++;
        }
        if ($aux->PrefuerzoSPPQ != '-') {
            $refuerzoPP = $aux->PrefuerzoSPPQ;
            $contadorPQ++;
        }
    }
}
```

Figura 122. Función generar boleta de calificaciones.

Elaborado por: Morales Carlos, 2023.