

UNIVERSIDAD TÉCNICA DEL NORTE
Facultad de Ingeniería en Ciencias Aplicadas
Carrera de Software



**ELABORACIÓN DE UNA GUÍA METODOLÓGICA PARA LA GESTIÓN DE
PROYECTOS DE SOFTWARE UTILIZANDO BITBUCKET Y SCRUM PARA
APOYAR LOS PROYECTOS DE DESARROLLO**

Autor:

Córdova Morillo Dayana Lisbeth

Director:

MSc. Xavier Mauricio Rea Peñafiel

Ibarra – Ecuador

2026



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	040184680-3		
APELLIDOS Y NOMBRES:	CÓRDOVA MORILLO DAYANA LISBETH		
DIRECCIÓN:	EL OLIVO – DR. CRISTÓBAL TOBAR SUBIA 7		
EMAIL:	dlcordovam@utn.edu.ec – dayalisbeth16@gmail.com		
TELÉFONO FIJO:	-----	TELÉFONO O MÓVIL:	0997875275

DATOS DE LA OBRA	
TÍTULO:	ELABORACIÓN DE UNA GUÍA METODOLÓGICA PARA LA GESTIÓN DE PROYECTOS DE SOFTWARE UTILIZANDO BITBUCKET Y SCRUM PARA APOYAR LOS PROYECTOS DE DESARROLLO.
AUTOR (ES):	CÓRDOVA MORILLO DAYANA LISBETH
FECHA: DD/MM/AAAA	19/02/2026
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	☒ PREGRADO ☐ POSGRADO
TÍTULO POR EL QUE OPTA:	INGENIERÍA EN SOFTWARE
ASESOR /DIRECTOR:	MSC. XAVIER REA

2. CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 19 días del mes de febrero de 2026

EL AUTOR:

Firma:

Nombre: Dayana Lisbeth Córdova Morillo



CERTIFICACIÓN DEL DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Ibarra, a los 19 días del mes de febrero de 2026

MSc. Xavier Rea.

DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

CERTIFICA:

Haber revisado el presente informe final del trabajo de Integración Curricular, el mismo que se ajusta a las normas vigentes de la Universidad Técnica del Norte; en consecuencia, autorizo su presentación para los fines legales pertinentes.

MSc. Xavier Rea
C.C.: 1002485744

DEDICATORIA

Este trabajo va dedicado especialmente a mi bebé Dome, por ser ese pequeño ser que me inspira y me mantiene en pie cada día. Gracias por salvarme la vida mi pequeña.

Se lo dedico también a la Daiita que pensó que jamás lo lograría. ¡LO HICIMOS MI NIÑA!

Dayana Lisbeth Córdova Morillo

AGRADECIMIENTO

Expreso mi más profundo agradecimiento a Dios, por escucharme siempre, por no abandonarme en los momentos difíciles y por darme la fortaleza, sabiduría y perseverancia necesarias para culminar este camino tan exigente.

Agradezco a mi familia por el apoyo incondicional que me han brindado a lo largo de mi vida y durante mi formación académica. De manera especial, a mis abuelos, por cuidarme desde pequeña y por inculcarme valores que han sido fundamentales para convertirme en la persona que soy ahora. A mi madre, por ser un ejemplo de fortaleza, esfuerzo y dedicación; gracias a su trabajo y sacrificio he podido alcanzar cada una de mis metas, este logro es fruto de su amor y entrega.

Agradecer también a mis docentes de carrera que contribuyeron a mi formación profesional mediante la transmisión de sus conocimientos y experiencias. De manera especial, al Ing. Mauricio Rea, director de esta tesis, por su pasión al compartir su conocimiento y por su valiosa guía durante el desarrollo de este trabajo. Al Ing. Antonio Quiña, asesor de tesis, por ser una mano amiga desde el inicio hasta el final de mi formación académica. Asimismo, al Ing. MacArthur Ortega, coordinador de carrera, por su disposición y ayuda oportuna cada vez que la necesité. Su apoyo fue fundamental y lo agradezco infinitamente.

Finalmente, agradezco a mis compañeros y amigos de carrera, por su apoyo constante y por nunca dejarme sola durante este proceso. Cada momento compartido, palabra de ánimo y ayuda brindada hicieron que este camino fuera menos difícil, ya que siempre conté con su compañía. De manera especial, agradezco a Wily, por su apoyo desinteresado y por su constante presión para que logre culminar este trabajo.

Dayana Lisbeth Córdova Morillo

RESUMEN

Este trabajo de titulación se estructura en tres capítulos, en el cual se desarrolla el proceso para realizar el trabajo de integración curricular: “Elaboración de una guía metodológica para la gestión de proyectos de software utilizando Bitbucket y Scrum para apoyar los proyectos de desarrollo”.

En la Introducción se presenta el planteamiento del problema, los objetivos generales y específicos, alcance, metodología y justificación.

En el Capítulo 1 se presenta el marco conceptual que fundamenta la investigación. En este apartado se abordan los principales referentes teóricos relacionados con la gestión de proyectos de software, las metodologías ágiles con énfasis en Scrum, sus principios, roles, eventos y artefactos, así como los conceptos vinculados a las herramientas de gestión y el uso de Bitbucket como herramienta de colaboración. Este capítulo proporciona la base teórica necesaria para sustentar el desarrollo de la propuesta y garantizar su coherencia académica y técnica.

En el Capítulo 2 se describe el diseño de la propuesta metodológica. Se analiza la metodología ágil Scrum sus principios, roles, eventos y artefactos como base para su aplicación en el ámbito académico. A partir de ello, se estructura una guía para la gestión de proyectos que integra Scrum y Bitbucket, incluyendo fases de trabajo, definición de roles, organización de repositorios y buenas prácticas adaptadas al contexto formativo.

En el Capítulo 3 se presenta la validación de la guía mediante una prueba de concepto en un entorno académico. Se evalúa su usabilidad con la escala System Usability Scale (SUS) y su funcionalidad con base en una norma ISO de calidad del software. Finalmente, se exponen y analizan los resultados obtenidos.

En este documento se incluyen las conclusiones, recomendaciones, referencias bibliográficas y anexos que respaldan el desarrollo de la investigación.

ABSTRACT

This degree project is structured into three chapters, in which the process for carrying out the curricular integration work entitled: “Development of a methodological guide for software project management using Bitbucket and Scrum to support development projects” is presented.

The Introduction outlines the problem statement, the general and specific objectives, scope, methodology, and justification of the study.

Chapter 1 presents the conceptual framework that supports the research. This section addresses the main theoretical references related to software project management, agile methodologies with emphasis on Scrum, including its principles, roles, events, and artifacts, as well as concepts related to management tools and the use of Bitbucket as a collaboration platform. This chapter provides the theoretical foundation necessary to support the development of the proposal and ensure its academic and technical coherence.

Chapter 2 describes the research process and the design of the methodological proposal. The agile methodology Scrum is analyzed, including its principles, roles, events, and artifacts, as the basis for its application in the academic environment. Based on this analysis, a project management guide integrating Scrum and Bitbucket is structured, including work phases, role definition, repository organization, and best practices adapted to the educational context.

Chapter 3 presents the validation of the guide through a proof of concept conducted in an academic environment. Its usability is evaluated using the System Usability Scale (SUS), and its functionality is assessed based on an ISO software quality standard. Finally, the results obtained are presented and analyzed.

This document also includes conclusions, recommendations, bibliographic references, and annexes that support the development of the research.

ÍNDICE DE CONTENIDOS

INTRODUCCIÓN.....	1
Tema	1
Problema	1
Planteamiento del problema	1
Objetivos.....	2
Objetivo General.....	2
Objetivos Específicos	2
Alcance	2
Metodología	4
Justificación	5
1. CAPÍTULO 1	6
Marco Teórico	6
1.1. Guías metodológicas	6
1.1.1. ¿Qué es una guía metodológica?	6
1.1.2. Elementos de una guía metodológica.....	6
1.1.3. Importancia de las guías metodológicas.....	8
1.1.4. Aplicaciones de las guías metodológicas	9
1.2. Ciclo de vida del software	9
1.2.1. Definición	9
1.2.2. Modelos del ciclo de vida del software	10
1.2.3. Fases del ciclo de vida del software	15
1.2.4. Riesgos en el ciclo de vida del software	16
1.3. Metodologías ágiles en el desarrollo del software	17
1.3.1. Definición de metodologías ágiles en el contexto del desarrollo de software	17
1.3.2. Principios de las metodologías ágiles	18

1.3.3.	Beneficios y desventajas de utilizar metodologías ágiles para desarrollar software	20
1.3.4.	Aplicaciones de las metodologías ágiles en la industria del desarrollo de software	22
1.3.5.	Metodología ágil SCRUM	23
1.4.	Herramientas de gestión para proyectos de desarrollo de software	24
1.4.1.	Definición	24
1.4.2.	Beneficios del uso de herramientas de gestión de proyectos en el desarrollo de software	25
1.4.3.	Comparación de herramientas de desarrollo de software	26
1.4.4.	Normas ISO en la gestión de proyectos de desarrollo de software.....	32
2.	CAPÍTULO 2.....	35
2.1.	Elaboración de la guía metodológica.....	35
2.1.1.	Introducción.....	35
2.1.2.	Objetivo	35
2.1.3.	Alcance	35
2.1.4.	Terminología	36
2.1.5.	Responsables.....	36
2.1.6.	Desarrollo.....	37
2.1.7.	Conclusiones.....	96
3.	CAPÍTULO 3.....	98
3.	Validación de resultados.....	98
3.1.	Aplicación de la guía metodológica	98
3.1.1.	Fase 1: Pre – Juego	98
3.1.2.	Fase 2: Juego.....	99
3.1.3.	Fase 3: Post – Juego.....	101
3.2.	Elaboración de cuestionario	101
3.2.1.	Preguntas de usabilidad	102
3.2.2.	Preguntas de funcionalidad	103

3.3. Resultados obtenidos.....	104
3.3.1. Evaluación de usabilidad.....	104
3.3.2. Evaluación de funcionalidad	108
3.3.3. Interpretación de resultados	111
CONCLUSIONES.....	126
RECOMENDACIONES.....	127
BIBLIOGRAFÍA	128
4. ANEXOS.....	133
Desarrollo de la prueba de concepto.....	133
Fase 1. Pre – Juego	133
Fase 2. Juego o Desarrollo.....	184
Fase 3. Post – Juego.....	226

ÍNDICE DE FIGURAS

Figura 1. <i>Árbol de problemas</i>	2
Figura 2. <i>Alcance del proyecto</i>	4
Figura 3. <i>Página de creación del workspace</i>	39
Figura 4. <i>Formulario inicial de creación el workspace</i>	40
Figura 5. <i>Página de visualización del workspace</i>	41
Figura 6. <i>Formulario de creación del proyecto</i>	42
Figura 7. <i>Página de creación del repositorio</i>	43
Figura 8. <i>Formulario de creación del repositorio</i>	45
Figura 9. <i>Formulario avanzado de creación del repositorio</i>	46
Figura 10. <i>Página de visualización del repositorio</i>	47
Figura 11. <i>Página inicial de Eclipse</i>	48
Figura 12. <i>Sección de Git de Eclipse</i>	49
Figura 13. <i>Formulario de selección de repositorio</i>	49
Figura 14. <i>Repositorio de Bitbucket</i>	50
Figura 15. <i>Página de clonación del repositorio de Bitbucket</i>	50
Figura 16. <i>Formulario de obtención de origen del repositorio en Eclipse</i>	51
Figura 17. <i>Selección de la rama del repositorio</i>	51
Figura 18. <i>Selección de la ubicación del proyecto en local</i>	52
Figura 19. <i>Proyecto cargado en Eclipse</i>	52
Figura 20. <i>Selección de la vista de Git</i>	53
Figura 21. <i>Ubicación de la opción de commit del repositorio</i>	53
Figura 22. <i>Formulario de commit del repositorio</i>	54
Figura 23. <i>Ubicación del botón para ejecutar el commit en Eclipse</i>	54
Figura 24. <i>Opción push para el commit previo en Eclipse</i>	55
Figura 25. <i>Formulario de configuración para el push en Eclipse</i>	55
Figura 26. <i>Push exitoso en Eclipse</i>	56

Figura 27. <i>Repositorio actualizado en Bitbucket</i>	56
Figura 28. <i>Opción Jira Issues</i>	57
Figura 29. <i>Opciones de integración con Jira</i>	57
Figura 30. <i>Formulario de vinculación de Jira</i>	58
Figura 31. <i>Selección del plan de Jira que se usará en el repositorio</i>	58
Figura 32. <i>Confirmación de Jira vinculado al repositorio</i>	59
Figura 33. <i>Plantillas del proyecto de Jira</i>	59
Figura 34. <i>Configuración del nombre del proyecto</i>	60
Figura 35. <i>Configuración completada con éxito</i>	60
Figura 36. <i>Página de visualización de Jira Issues</i>	61
Figura 37. <i>Selección del repositorio en Jira</i>	61
Figura 38. <i>Jira solicitando el acceso al repositorio</i>	62
Figura 39. <i>Página para aceptar la vinculación de Jira al repositorio</i>	62
Figura 40. <i>Página de confirmación de la vinculación entre Jira y el repositorio</i>	63
Figura 41. <i>Opción para enlazar el espacio con Bitbucket Cloud</i>	63
Figura 42. <i>Jira solicitando acceder al espacio de trabajo</i>	64
Figura 43. <i>Visualización de Jira Issues</i>	64
Figura 44. <i>Sección Manage linked Projects</i>	65
Figura 45. <i>Gestión de los proyectos de Jira</i>	65
Figura 46. <i>Proyecto de Jira vinculado con Bitbucket</i>	66
Figura 47. <i>Gestionar los permisos y roles de usuarios</i>	66
Figura 48. <i>Añadir miembros a equipo</i>	67
Figura 49. <i>Colaboradores de equipo</i>	67
Figura 50. <i>Invitar a nuevos usuarios</i>	68
Figura 51. <i>Añadir usuarios o grupos</i>	68
Figura 52. <i>Formulario para añadir usuarios</i>	69
Figura 53. <i>Usuarios añadidos a Jira</i>	69

Figura 54. <i>Sección tablero de Jira</i>	70
Figura 55. <i>Añadir usuario invitado al proyecto</i>	70
Figura 56. <i>Ingreso de correo de colaborador</i>	71
Figura 57. <i>Usuario añadido correctamente</i>	71
Figura 58. <i>Tablero Scrum para crear el Sprint Backlog</i>	72
Figura 59. <i>Eliminar Sprint predeterminado</i>	72
Figura 60. <i>Agregar nuevo Sprint al tablero</i>	73
Figura 61. <i>Opción para editar el Sprint</i>	73
Figura 62. <i>Formulario de edición de tablero de Sprint</i>	74
Figura 63. <i>Crear una nueva incidencia</i>	74
Figura 64. <i>Selección de tipo de incidencia</i>	75
Figura 65. <i>Descripción de una nueva incidencia</i>	75
Figura 66. <i>Designar usuario a incidencia</i>	76
Figura 67. <i>Encargado de la nueva incidencia</i>	76
Figura 68. <i>Edición de la descripción de una tarea</i>	77
Figura 69. <i>Añadir y guardar la descripción de una tarea</i>	77
Figura 70. <i>Visualización de una tarea en el Sprint</i>	78
Figura 71. <i>Modificación de la Issue en el backlog</i>	79
Figura 72. <i>Product backlog de Jira</i>	80
Figura 73. <i>Creación del Sprint planning</i>	81
Figura 74. <i>Opción para iniciar el Sprint</i>	83
Figura 75. <i>Formulario de configuración de inicio de Sprint</i>	84
Figura 76. <i>Tablero de monitoreo de Sprints</i>	85
Figura 77. <i>Gestión de actividades de los Sprints</i>	86
Figura 78. <i>Visualización del Issue</i>	87
Figura 79. <i>Completar el Sprint</i>	87
Figura 80. <i>Gestión de incidentes pendientes</i>	88

Figura 81.	<i>Revisión de los incidentes pendientes del Sprint anterior</i>	89
Figura 82.	<i>Sprint retrospective</i>	90
Figura 83.	<i>Resumen de las actividades</i>	91
Figura 84.	<i>Resumen de los incidentes completados y pendientes</i>	91
Figura 85.	<i>Incidentes completados</i>	92
Figura 86.	<i>Repositorio actualizado</i>	93
Figura 87.	<i>Sección de commits del repositorio</i>	94
Figura 88.	<i>Sección de Tags del repositorio</i>	95
Figura 89.	<i>Formulario de creación de Tags</i>	95
Figura 90.	<i>Commit etiquetado</i>	96
Figura 91.	<i>Repositorio creado en Bitbucket con estructura del sistema</i>	99
Figura 92.	<i>Gestión de historias de usuario y planificación de Sprints en Jira</i>	100
Figura 93.	<i>Vista funcional del sistema: Gestión y seguimiento de tesis</i>	100
Figura 94.	<i>Confirmación de versión etiquetada al finalizar un Sprint</i>	101
Figura 95.	<i>Resultados de la pregunta 1</i>	112
Figura 96.	<i>Resultados de la pregunta 2</i>	113
Figura 97.	<i>Resultados de la pregunta 3</i>	114
Figura 98.	<i>Resultados de la pregunta 4</i>	115
Figura 99.	<i>Resultados de la pregunta 5</i>	116
Figura 100.	<i>Resultados de la pregunta 6</i>	117
Figura 101.	<i>Resultados de la pregunta 7</i>	118
Figura 102.	<i>Resultados de la pregunta 8</i>	119
Figura 103.	<i>Resultados de la pregunta 9</i>	120
Figura 104.	<i>Resultados de la pregunta 10</i>	121
Figura 105.	<i>Resultados de la pregunta 11</i>	122
Figura 106.	<i>Resultados de la pregunta 12</i>	123
Figura 107.	<i>Resultados de la pregunta 13</i>	124

Figura 108. <i>Resultados de la pregunta 14</i>	124
Figura 109. <i>Resultados de la pregunta 15</i>	125
Figura 110. <i>Opción projects del panel de navegación</i>	133
Figura 111. <i>Formulario de creación de proyecto</i>	134
Figura 112. <i>Vista con los repositorios del proyecto</i>	134
Figura 113. <i>Formulario de importación de un repositorio</i>	135
Figura 114. <i>Formulario avanzado de importación de un repositorio</i>	135
Figura 115. <i>Pantalla de espera de importación de repositorio</i>	136
Figura 116. <i>Repositorio importado</i>	136
Figura 117. <i>Vista de GIT en eclipse</i>	137
Figura 118. <i>Pantalla del repositorio para realizar la clonación</i>	138
Figura 119. <i>Ventana para clonar el repositorio de Bitbucket</i>	138
Figura 120. <i>Formulario de ingreso de credenciales en Eclipse</i>	139
Figura 121. <i>Selección de la rama del proyecto</i>	139
Figura 122. <i>Ubicación del repositorio en local</i>	140
Figura 123. <i>Repositorios clonados dentro de eclipse</i>	140
Figura 124. <i>Opción de importación de proyectos</i>	141
Figura 125. <i>Ventana de importación del proyecto</i>	141
Figura 126. <i>Proyecto importado en Eclipse</i>	142
Figura 127. <i>Ventana de gestión de roles en Bitbucket</i>	143
Figura 128. <i>Sección de gestión de usuarios en Bitbucket</i>	143
Figura 129. <i>Formulario para ingresar los usuarios al proyecto</i>	144
Figura 130. <i>Usuarios añadidos al proyecto</i>	144
Figura 131. <i>Vista principal del repositorio de Bitbucket</i>	145
Figura 132. <i>Usuarios dentro del proyecto</i>	145
Figura 133. <i>Gestión de permisos del rol dentro del repositorio</i>	146
Figura 134. <i>Listado de usuarios con sus respectivos permisos</i>	146

Figura 135. <i>Creación del proyecto en Jira</i>	148
Figura 136. <i>Selección de un tipo de proyecto</i>	149
Figura 137. <i>Descripción del tipo de proyecto y selección de gestión del proyecto</i>	149
Figura 138. <i>Formulario de creación del proyecto scrum en Jira</i>	150
Figura 139. <i>Vista principal del proyecto en Jira</i>	150
Figura 140. <i>Proyecto en Bitbucket</i>	151
Figura 141. <i>Sección para conectar con Jira Issues</i>	151
Figura 142. <i>Opción para gestionar proyectos vinculados</i>	152
Figura 143. <i>Vista de gestión de proyectos vinculados con Jira</i>	152
Figura 144. <i>Jira Issues vinculados al repositorio de Bitbucket</i>	153
Figura 145. <i>Proyecto de Jira con los Issues</i>	153
Figura 146. <i>Sección de Sprints vacía</i>	154
Figura 147. <i>Segundo Sprint vacío</i>	154
Figura 148. <i>Tercer Sprint Vacío</i>	154
Figura 149. <i>Cuarto Sprint vacío</i>	154
Figura 150. <i>Quinto Sprint vacío</i>	155
Figura 151. <i>Sexto Sprint vacío</i>	155
Figura 152. <i>Issues en el primer tablero Sprint</i>	155
Figura 153. <i>Issues en el segundo tablero Sprint</i>	155
Figura 154. <i>Issues en el tercer tablero Sprint</i>	156
Figura 155. <i>Issues en el cuarto tablero Sprint</i>	156
Figura 156. <i>Issues en el quinto tablero Sprint</i>	156
Figura 157. <i>Issues en el quinto tablero Sprint</i>	156
Figura 158. <i>Issue de gestión de tesis</i>	167
Figura 159. <i>Issue de Gestión de avances</i>	168
Figura 160. <i>Issue de gestión de resoluciones</i>	168

Figura 161. <i>Issue de filtros de búsquedas</i>	169
Figura 162. <i>Issue de gestión de docentes</i>	169
Figura 163. <i>Issue de gestión de estudiantes</i>	170
Figura 164. <i>Issue de gestión de carreras</i>	170
Figura 165. <i>Issue de gestión de facultades</i>	171
Figura 166. <i>Issue de visualización de tesis</i>	171
Figura 167. <i>Issue de visualización de avances</i>	172
Figura 168. <i>Visualización de resoluciones</i>	172
Figura 169. <i>Issue de Reportes</i>	173
Figura 170. <i>Issue de visualización de docentes</i>	173
Figura 171. <i>Issue de visualización de estudiantes</i>	174
Figura 172. <i>Issue de visualización de carreras</i>	174
Figura 173. <i>Issue de visualización de facultades</i>	175
Figura 174. <i>Issue de reportes</i>	175
Figura 175. <i>Issue #1 con estimación de 30 horas</i>	176
Figura 176. <i>Issue #2 con estimación de 20 horas</i>	176
Figura 177. <i>Issue #3 con estimación de 10 horas</i>	177
Figura 178. <i>Issue #4 con estimación de 2 horas</i>	177
Figura 179. <i>Issue #5 con estimación de 20 horas</i>	178
Figura 180. <i>Issue #6 con estimación de 6 horas</i>	178
Figura 181. <i>Issue #7 con estimación de 6 horas</i>	179
Figura 182. <i>Issue #8 con estimación de 3 horas</i>	179
Figura 183. <i>Issue #9 con estimación de 2 horas</i>	180
Figura 184. <i>Issue #10 con estimación de 2 horas</i>	180
Figura 185. <i>Issue #11 con estimación de 2 horas</i>	181
Figura 186. <i>Issue #12 con estimación de 5 horas</i>	181
Figura 187. <i>Issue #13 con estimación de 2 horas</i>	182

Figura 188. <i>Issue #14 con estimación de 2 horas</i>	182
Figura 189. <i>Issue #15 con estimación de 2 horas</i>	183
Figura 190. <i>Issue #16 con estimación de 2 horas</i>	183
Figura 191. <i>Issue #17 con estimación de 2 horas</i>	184
Figura 192. <i>Edición de la información Sprint #1</i>	184
Figura 193. <i>Edición de la información Sprint #2</i>	185
Figura 194. <i>Edición de la información Sprint #3</i>	185
Figura 195. <i>Edición de la información Sprint #4</i>	186
Figura 196. <i>Edición de la información Sprint #5</i>	186
Figura 197. <i>Edición de la información Sprint #6</i>	187
Figura 198. <i>Tablero Sprint 1</i>	189
Figura 199. <i>Inicio del Sprint #1</i>	190
Figura 200. <i>Tareas del Sprint 1 sin modificaciones</i>	190
Figura 201. <i>Avances del Sprint 1</i>	191
Figura 202. <i>Finalización del Issue #1</i>	191
Figura 203. <i>Finalización del Issue #2</i>	192
Figura 204. <i>Finalización del Issue #3</i>	192
Figura 205. <i>Avance del Issue #4</i>	193
Figura 206. <i>Finalización del Sprint 1</i>	193
Figura 207. <i>Resumen sencillo del Sprint 1</i>	194
Figura 208. <i>Resumen avanzado del Sprint 1</i>	194
Figura 209. <i>Tareas completadas en el Sprint 1</i>	195
Figura 210. <i>Tareas completadas en el Sprint 1</i>	195
Figura 211. <i>Issues retrasados del del Sprint 1 en el Sprint 2</i>	196
Figura 212. <i>Tablero Sprint 2</i>	197
Figura 213. <i>Inicio del Sprint #2</i>	197
Figura 214. <i>Progreso del Sprint 2</i>	198

Figura 215. <i>Sprint retrospective del Sprint 2</i>	198
Figura 216. <i>Finalización del Issue #5</i>	199
Figura 217. <i>Finalización del Issue #6</i>	199
Figura 218. <i>Finalización del Issue #7</i>	200
Figura 219. <i>Finalización del Issue #8</i>	200
Figura 220. <i>Finalización del Sprint 2</i>	201
Figura 221. <i>Resumen sencillo del Sprint 2</i>	201
Figura 222. <i>Resumen avanzado del Sprint 2</i>	202
Figura 223. <i>Tareas completadas en el Sprint 2</i>	202
Figura 224. <i>Comentarios del Sprint 2</i>	203
Figura 225. <i>Tablero Sprint 3</i>	204
Figura 226. <i>Inicio del Sprint #3</i>	204
Figura 227. <i>Progreso del Sprint 3</i>	205
Figura 228. <i>Sprint retrospective del Sprint 3</i>	205
Figura 229. <i>Finalización del Issue #9</i>	206
Figura 230. <i>Finalización del Sprint 3</i>	206
Figura 231. <i>Resumen sencillo del Sprint 3</i>	207
Figura 232. <i>Tareas completadas en el Sprint 3</i>	207
Figura 233. <i>Comentarios del Sprint 3</i>	208
Figura 234. <i>Tablero Sprint 4</i>	209
Figura 235. <i>Inicio del Sprint #4</i>	209
Figura 236. <i>Progreso del Sprint 4</i>	210
Figura 237. <i>Sprint retrospective del Sprint 4</i>	210
Figura 238. <i>Finalización del Issue #10</i>	211
Figura 239. <i>Finalización del Issue #11</i>	211
Figura 240. <i>Finalización del Sprint 4</i>	212
Figura 241. <i>Resumen sencillo del Sprint 4</i>	212

Figura 242. <i>Tareas completadas en el Sprint 4</i>	213
Figura 243. <i>Comentarios del Sprint 4</i>	213
Figura 244. <i>Tablero Sprint 5</i>	214
Figura 245. <i>Inicio del Sprint #5</i>	215
Figura 246. <i>Progreso del Sprint 5</i>	215
Figura 247. <i>Sprint retrospective del Sprint 5</i>	216
Figura 248. <i>Finalización del Issue #13</i>	216
Figura 249. <i>Finalización del Issue #14</i>	217
Figura 250. <i>Finalización del Sprint 5</i>	217
Figura 251. <i>Resumen sencillo del Sprint 5</i>	218
Figura 252. <i>Resumen avanzado del Sprint 5</i>	218
Figura 253. <i>Tareas completadas en el Sprint 5</i>	219
Figura 254. <i>Comentarios del Sprint 5</i>	219
Figura 255. <i>Tablero Sprint 6</i>	220
Figura 256. <i>Inicio del Sprint #6</i>	221
Figura 257. <i>Progreso del Sprint 6</i>	221
Figura 258. <i>Sprint retrospective del Sprint 6</i>	222
Figura 259. <i>Finalización del Issue #15</i>	222
Figura 260. <i>Finalización del Issue #16</i>	223
Figura 261. <i>Finalización del Issue #4</i>	223
Figura 262. <i>Finalización del Issue #12</i>	224
Figura 263. <i>Finalización del Sprint 6</i>	224
Figura 264. <i>Resumen sencillo del Sprint 6</i>	225
Figura 265. <i>Resumen avanzado del Sprint 6</i>	225
Figura 266. <i>Tareas completadas en el Sprint 6</i>	226
Figura 267. <i>Commit inicial desde Eclipse</i>	226
Figura 268. <i>Cambios del Sprint 1 subidos a Bitbucket</i>	227

Figura 269. <i>Creación de la etiqueta del Sprint 1</i>	227
Figura 270. <i>Commit del Sprint 1 etiquetado</i>	228
Figura 271. <i>Commits del Sprint 1 etiquetados</i>	228
Figura 272. <i>Commit del Sprint 2 subido desde eclipse</i>	229
Figura 273. <i>Cambios del Sprint 2 subidos a Bitbucket</i>	229
Figura 274. <i>Creación de la etiqueta del Sprint 2</i>	230
Figura 275. <i>Commits del Sprint 2 etiquetados</i>	230
Figura 276. <i>Commits actualizados luego del Sprint 2</i>	231
Figura 277. <i>Commits del Sprint 3 subidos</i>	231
Figura 278. <i>Cambios del Sprint 3 subidos a Bitbucket</i>	232
Figura 279. <i>Creación de la etiqueta del Sprint 3</i>	232
Figura 280. <i>Commits actualizados después del Sprint 3</i>	233
Figura 281. <i>Commit del Sprint 4</i>	233
Figura 282. <i>Cambios del Sprint 4 subidos a Bitbucket</i>	234
Figura 283. <i>Creación de la etiqueta del Sprint 4</i>	234
Figura 284. <i>Commit del Sprint 4 etiquetados</i>	235
Figura 285. <i>Commits actualizados luego del Sprint 4</i>	235
Figura 286. <i>Commit del Sprint 5</i>	236
Figura 287. <i>Cambios del Sprint 5 subidos a Bitbucket</i>	236
Figura 288. <i>Creación de la etiqueta del Sprint 5</i>	237
Figura 289. <i>Commit del Sprint 5 etiquetado</i>	237
Figura 290. <i>Commits actualizados después del Sprint 5</i>	238
Figura 291. <i>Commits del Sprint 6</i>	238
Figura 292. <i>Commits del Sprint 6 subidos en Bitbucket</i>	239
Figura 293. <i>Creación de la etiqueta del Sprint 6</i>	239
Figura 294. <i>Commits del Sprint 6 subidos a Bitbucket</i>	240
Figura 295. <i>Etiquetas actualizadas después del Sprint 6</i>	240

ÍNDICE DE TABLAS

Tabla 1. <i>Modelos del ciclo de vida del software - Cascada</i>	11
Tabla 2. <i>Modelos del ciclo de vida del software - Espiral</i>	11
Tabla 3. <i>Modelos del ciclo de vida del software – Iterativo e incremental</i>	12
Tabla 4. <i>Modelos del ciclo de vida del software – Modelo en “V”</i>	13
Tabla 5. <i>Modelos del ciclo de vida del software - Ágil</i>	13
Tabla 6. <i>Modelos del ciclo de vida del software - DevOps</i>	14
Tabla 7. <i>Comparativa entre herramientas de desarrollo de software</i>	30
Tabla 8. <i>Terminología</i>	36
Tabla 9. <i>Preguntas de usabilidad</i>	102
Tabla 10. <i>Resultados de usabilidad</i>	105
Tabla 11. <i>Resultados de las preguntas impares de usabilidad</i>	106
Tabla 12. <i>Resultados de las preguntas pares de usabilidad</i>	107
Tabla 13. <i>Resultados de funcionalidad</i>	108
Tabla 14. <i>Resultados de la pregunta 11</i>	110
Tabla 15. <i>Resultados de la sección de adecuación funcional</i>	111
Tabla 16. <i>Definición de los roles de scrum</i>	142
Tabla 17. <i>Product Backlog</i>	147
Tabla 18. <i>Historia de usuario Gestión de tesis</i>	157
Tabla 19. <i>Historia de usuario Gestión de avances</i>	157
Tabla 20. <i>Historia de usuario Gestión de resoluciones</i>	158
Tabla 21. <i>Historia de usuario Gestión de docentes</i>	159
Tabla 22. <i>Historia de usuario Gestión de estudiantes</i>	159
Tabla 23. <i>Historia de usuario Gestión de carreras</i>	160
Tabla 24. <i>Historia de usuario Gestión de facultades</i>	161
Tabla 25. <i>Historia de usuario Visualización de tesis</i>	161
Tabla 26. <i>Historia de usuario Visualización de avances</i>	162

Tabla 27. <i>Historia de usuario Visualización de resoluciones</i>	163
Tabla 28. <i>Historia de usuario Visualización de docentes</i>	163
Tabla 29. <i>Historia de usuarios Visualización de estudiantes</i>	164
Tabla 30. <i>Historia de usuario Visualización de carreras</i>	164
Tabla 31. <i>Historia de usuario Visualización de facultades</i>	165
Tabla 32. <i>Historia de usuario Filtros de búsqueda</i>	166
Tabla 33. <i>Historia de usuario Reportes</i>	166
Tabla 34. <i>Sprint backlog</i>	187
Tabla 35. <i>Prototipo Sprint 1</i>	188
Tabla 36. <i>Prototipo Sprint 2</i>	196
Tabla 37. <i>Prototipo Sprint 3</i>	203
Tabla 38. <i>Prototipo Sprint 4</i>	208
Tabla 39. <i>Prototipo sprint 5</i>	214
Tabla 40. <i>Prototipo Sprint 6</i>	220

INTRODUCCIÓN

Tema

Elaboración de una guía metodológica para la gestión de proyectos de software utilizando Bitbucket y Scrum para apoyar los proyectos de desarrollo.

Problema

Alto índice de ineficacia en la gestión de proyectos de software.

Planteamiento del problema

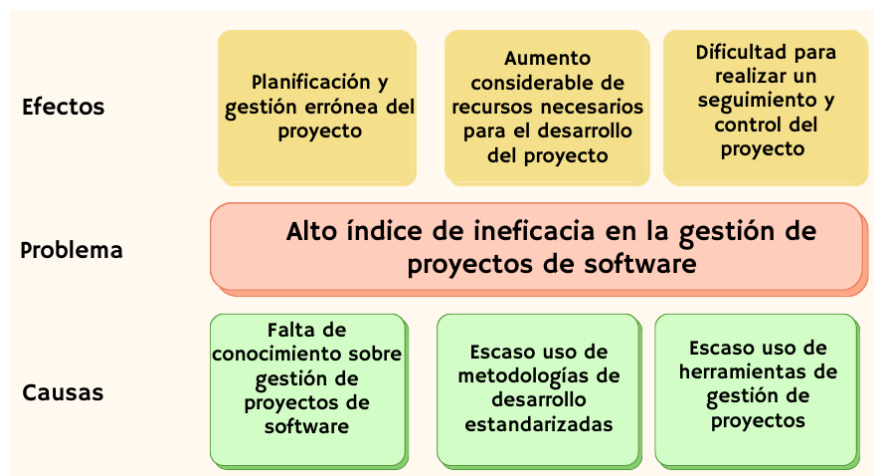
Según el “Reporte Chaos” perteneciente al Standish Group, en el periodo de 2015 a 2020, solo el 35% de los proyectos de software han sido exitosos, el 46% enfrentaron dificultades y el 19% fracasaron (Alvarez, 2023). Se ha visto que el desarrollo del software no se gestiona de una manera adecuada, ni formal, esto puede afectar su ciclo, ya que los desarrolladores ven menos llamativo el proceso completo que implica el desarrollar y gestionar un software; en la carrera de Software de la Universidad Técnica del Norte, se ha apreciado una desorganización en los grupos de trabajo, en consecuencia, los estudiantes no pueden cumplir plenamente con los objetivos de sus proyectos (Rochina, 2023a).

El gestionar un proyecto de software es un completo desafío debido a la falta de recursos que fusionen metodologías y herramientas para realizar esta tarea, lo que impide que se planifique correctamente, asimismo, el escaso uso de métodos estandarizados limita la capacidad para organizar las tareas necesarias para cumplir con el proyecto asignado, además la falta de adopción de herramientas de gestión de proyectos dificulta la comunicación dentro del equipo de trabajo, dando como resultado errores que impliquen el desperdicio de recursos.

En base a lo anterior, es necesario que exista una guía metodológica que incluya el uso de una metodología de desarrollo junto a una herramienta de gestión de proyectos, esta ayudará a que un equipo de trabajo pueda planificar, comunicarse y sobre todo seguir paso a paso el progreso de desarrollo, cumpliendo a cabalidad con las asignaciones y organización de los implementos necesarios, para gestionar los recursos y riesgos, así los miembros del equipo estarán enfocados en un mismo objetivo.

En la figura 1, se aprecia las causas y efectos del problema que se ha percibido.

Figura 1.
Árbol de problemas



Nota. Elaboración Propia.

Objetivos

Objetivo General.

Elaborar una guía metodológica para la gestión de proyectos de software utilizando Bitbucket y Scrum para apoyar los proyectos de desarrollo.

Objetivos Específicos

- Elaborar un marco conceptual sobre gestión de proyectos de software.
- Desarrollar una guía metodológica para la gestión de proyectos utilizando la herramienta Bitbucket basado en el marco de trabajo SCRUM.
- Validar la guía metodológica mediante una prueba de concepto.

Alcance

El presente proyecto tiene la finalidad de elaborar una guía metodológica que servirá especialmente a estudiantes de la carrera de Software. Para esto se iniciará con la elaboración de un marco conceptual para tener conocimiento profundo acerca del tema a desarrollarse, utilizando herramientas de Office como Microsoft Word y Mendeley Citation para gestionar y organizar citas y bibliografía.

Además, se realizará la guía metodológica combinando Bitbucket y la metodología ágil Scrum, cada principio de Scrum irá alineado con las funcionalidades de la herramienta, como la gestión de repositorios, control de versiones de código y la integración continua, esto logrará

que los equipos de desarrollo colaboren fácilmente promoviendo la agilidad en la entrega del producto y asegurando un control eficiente dentro del proceso.

También, se desarrollará un sistema informático para seguimiento de trabajos de integración curricular (básico) como prueba de concepto. Para la validación del sistema se realizará una encuesta en base a la escala de usabilidad del sistema SUS (System Usability Scale) y la característica de adecuación funcional de la ISO/IEC 25010, mediante Microsoft Forms.

Se validará la guía mediante la prueba de concepto, para ello se utilizarán las tecnologías detalladas a continuación:

- Base de datos: PostgreSQL
- IDE: Eclipse
- Frontend: JSF Primefaces
- Backend: Java EJB

El gestor web contará con los siguientes módulos:

- Módulo de gestión de trabajos de integración curricular
- Módulo de seguridades

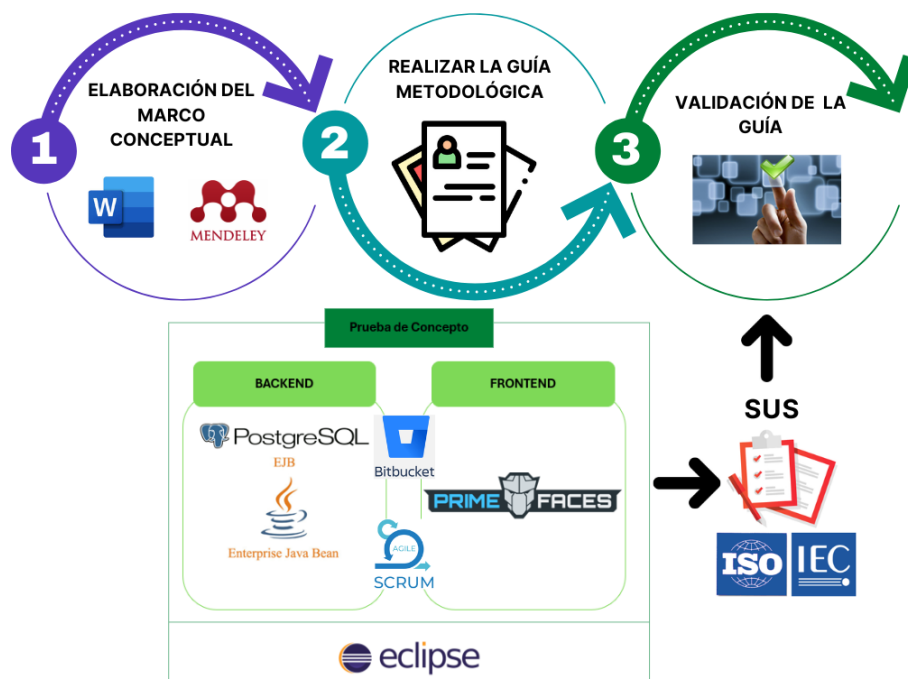
Las funciones que tendrá el sistema son:

- Gestión de datos de estudiantes
- Gestión de datos de comisión asesora y resoluciones
- Gestión de datos de resoluciones HCD
- Gestión de proyecto de integración curricular (tema, fechas, director, asesor)
- Reportes

Para la gestión del proyecto se utilizarán herramientas que brinda Bitbucket y se utilizará Scrum como marco de trabajo para el seguimiento de tareas y tiempos de entrega del sistema.

Figura 2.

Alcance del proyecto



Nota. Elaboración Propia.

Metodología

Con el propósito de cumplir el objetivo uno, se realizará una revisión documental con la finalidad de desarrollar el marco conceptual sobre lo que implica la gestión de proyectos y las herramientas que se utilizarán en el proyecto, con esto se logrará situar la investigación y sustentarla teórica y conceptualmente gracias a las investigaciones previamente realizadas por otros investigadores e investigadoras (Arnau & Sala, 2020). Los documentos obtenidos pertenecerán a diferentes bases de datos bibliográficas como Scopus, Google Académico, IEEE, etc., que serán recopilaciones de publicaciones científicas y técnicas con contenido temático, como artículos de revistas, libros, tesis, congresos, etc., que tienen como objetivo recopilar todos los trabajos bibliográficos posibles en un determinado campo del conocimiento (Universidad de Extremadura, 2022).

Para el segundo objetivo se realizará la guía metodológica empleando un formato preexistente utilizado en la tesis de (Cabascango, 2022a). Con el fin de asegurar que la guía contenga las herramientas y flujos de trabajo que Bitbucket proporciona para la gestión de proyectos, se realizará un estudio detallado de sus funcionalidades y características, así también se implementará Scrum para una mejor organización en el proyecto.

Finalmente, para cumplir con el objetivo tres, se realizará una prueba de concepto misma que será el desarrollo de un proyecto básico de software en el que se utilizará Bitbucket y Scrum, siguiendo las indicaciones de la guía metodológica. Una vez finalizado el desarrollo del software, se validará la usabilidad de este usando la prueba SUS y la funcionalidad bajo la característica de la adecuación funcional de la ISO/IEC 25010, y los resultados obtenidos de la encuesta validarán la guía metodológica.

Justificación

La investigación se enfocará en dos de los Objetivos de Desarrollo Sostenible; en el 4, que propone educación de calidad, basándose especialmente en la meta 4.7, que explica los conocimientos que los estudiantes adquieren para el desarrollo sostenible, la educación y el estilo de vida sostenibles, para mediante esto minimizar los problemas que puedan presentarse en el contexto educativo, así como también en el objetivo 9, tomando en cuenta su meta 9.3 la cual habla sobre facilitar el acceso a las empresas para hacer uso de tecnologías limpias, eficientes y sostenibles y su meta 9.5 que trata sobre mejorar el acceso a tecnologías para países en desarrollo (ONU, 2022), orientándose en esta justificación este trabajo de titulación solucionará la problemática de la ineficacia en la gestión de proyectos que enfrentan los estudiantes de la carrera de software de la Universidad Técnica del Norte.

Justificación tecnológica. – El uso de herramientas tecnológicas en el desarrollo de software es muy importante, pues ayuda a agilizar diferentes procesos y por ende ahorrar tiempo y recursos indispensables. La guía metodológica que se desarrollará en el presente proyecto permitirá a estudiantes y docentes aprovechar al máximo las capacidades que tiene la tecnología para ayudar a reducir errores y aumentar la eficiencia al gestionar proyectos de desarrollo, garantizando la calidad del producto que será entregado al usuario final.

Justificación educativa. – La guía metodológica propuesta ayudará a que los estudiantes aprendan a utilizar herramientas que se encuentran a su disposición para la buena gestión de proyectos, de esta manera se promoverá el aprendizaje práctico y eficiente, para obtener aptitudes que les ayuden a desenvolverse de manera adecuada dentro de un equipo de trabajo siendo organizados y llevando un control de las actividades que deberán realizar, gracias a esto tendrán una idea de cómo será el mercado laboral al que se enfrentarán una vez culminada su preparación académica dentro de la universidad.

CAPÍTULO 1

Marco Teórico

En este capítulo se presenta la fundamentación teórica que sustenta el proyecto a desarrollarse. Se definen conceptos clave relacionados con las guías metodológicas y su importancia de implementación en proyectos, además, se obtendrá información sobre las fases del ciclo de vida del software, lo cual permitirá contextualizar las etapas en las que interviene la metodología propuesta. Se abordará también las metodologías ágiles en el desarrollo del software y las herramientas de proyectos de software que servirán como base para la elaboración de la guía metodológica.

1.1. Guías metodológicas

1.1.1. *¿Qué es una guía metodológica?*

Es un manual técnico en el cual se detallan procesos, actividades o prácticas que describen cómo hacer algo siguiendo pasos en su secuencia lógica, este documento debe basarse en una experiencia que se sabe que funciona e incluya claves para que las actividades sean realizadas exitosamente (Huertas, 2024).

Según Soto (2021), una guía metodológica se define como un instrumento que ofrece pautas técnicas y normativas para facilitar un proceso específico, principalmente en entornos educativos, esta herramienta debe contar con objetivos claros y actividades concretas para alcanzarlos, además de definir el qué y el cómo, la guía contempla la evaluación y adaptación constante basándose en la experiencia práctica y el contexto en el que se aplica.

1.1.2. *Elementos de una guía metodológica*

Para realizar una guía metodológica existen pasos propuestos y utilizados por diferentes autores.

Según Váscones (2022), una guía metodológica incluye algunos elementos que aseguran su efectividad y aplicabilidad dentro de un contexto específico, estos son:

- **Objetivos:** establecer las metas que se pretende alcanzar con la guía.
- **Alcance:** especificación de los temas y campos que se abordarán.

- Terminología: listado explicativo de expresiones especializadas y nociones importantes empleadas en el escrito.
- Responsables: incluye a los individuos o grupos responsables de ejecutar las tareas sugeridas.
- Descripción de las actividades: exposición minuciosa de las frases, tareas y actividades a realizar.
- Formatos y diagramas de flujo: recursos gráficos que simplifican la interpretación de los métodos y la organización de la guía.

Cabascango (2022), desarrolló una guía metodológica utilizando Microsoft Azure Devops y el marco de trabajo Scrum en la cual definió los siguientes elementos como parte de la estructura de su guía:

- Resumen: ofrece una visión general concisa del contenido de la guía, proporciona al lector una idea general de la relevancia y el enfoque del trabajo.
- Introducción: esta sección establece el contexto del documento.
- Desarrollo: se centra en la elaboración de la guía metodológica, se detallan actividades a desarrollarse, incluyendo instrucciones precisas, para que los usuarios de la guía tengan una referencia concreta de lo que deben hacer en cada paso.
- Conclusiones: presentan las reflexiones y aprendizajes durante la elaboración y aplicación de la guía.

Rochina (2023), a diferencia de los dos autores anteriormente mencionados utiliza elementos adicionales, tales como:

- Introducción: es la sección inicial del documento, explica la relevancia y propósitos del trabajo.
- Objetivo: se establecen los propósitos que se pretenden alcanzar con la guía.
- Alcance: definición de los contenidos y áreas que cubre la guía metodológica.
- Audiencia: identificación de los destinatarios de la guía

- Fases de implementación: incluye etapas de Scrum que ayudan a estructurar la aplicación del proceso.
- Conclusión: sintetiza los resultados y evaluar si se cumplieron los objetivos planteados.

Aunque diferentes autores tienen sus propias ideas sobre qué debe incluir una guía metodológica, todos están de acuerdo en que sus elementos son importantes para que funcione bien en situaciones reales, cada paso de la guía, desde explicar qué se quiere lograr hasta detallar cómo hacerlo, ayuda a que sea más fácil de entender y usar, una buena guía metodológica ayuda a las personas a manejar proyectos y obtener buenos resultados en su trabajo.

1.1.3. Importancia de las guías metodológicas

Una guía metodológica es un instrumento que une eficazmente la teoría y la práctica dentro de cualquier ámbito, es importante porque une lo que se aprende mediante documentos como libros, artículos, etc., con lo que se realiza en el día a día, además las guías brindan consejos claros basados en investigaciones confiables, lo que ayuda a tomar las mejores decisiones dentro de un proyecto, también, hacen que sea fácil de entender ideas complicadas usando recursos visuales y explicaciones detalladas, permitiendo la comprensión y aplicación de métodos necesarios, así se transforma el conocimiento en acciones útiles mejorando la calidad de un trabajo y obteniendo los mejores resultados (Johnston et al., 2019).

El uso de una guía metodológica es crucial para el éxito de un proyecto, esta herramienta establece un marco que fomenta la colaboración y ayuda a alcanzar los objetivos propuestos, al proporcionar una dirección adecuada, facilita la gestión y supervisión de las distintas etapas del mismo, ya sean estas consecutivas, cíclicas o solapadas, una metodología bien definida permite una administración eficiente de los recursos y mejora la comunicación entre los miembros del equipo, este enfoque estructurado contribuye a minimizar riesgos y potenciar la productividad en la práctica, la aplicación de una guía metodológica es beneficiosa en cualquier ámbito, ya que ofrece una estructura clara que impulsa una ejecución exitosa del proyecto, este documento se convierte en un pilar fundamental para lograr resultados positivos en cualquier iniciativa, independientemente de su naturaleza o sector (Cardona & Valencia, 2021).

1.1.4. Aplicaciones de las guías metodológicas

Las guías metodológicas son herramientas de gran versatilidad que encuentran aplicación en múltiples campos del conocimiento y la práctica profesional. En el ámbito educativo, por ejemplo, De la Torre (2023), desarrolló una guía metodológica destinada al aprendizaje en niños con Trastorno por Déficit de Atención (TDA). Este trabajo se fundamentó en una amplia recopilación de estudios e investigaciones, cuyo objetivo fue identificar estrategias pedagógicas adecuadas y destacar la importancia de personalizarlas según las necesidades individuales de cada niño con TDA. De manera complementaria, Bolaños & Calahorrano (2023), diseñaron una guía metodológica centrada en la enseñanza basada en la inteligencia emocional, enfatizando que las emociones desempeñan un rol esencial en el fomento de la creatividad y el desarrollo integral del estudiante.

Por otra parte, las guías metodológicas son también fundamentales en diversas industrias, siendo el desarrollo de software un ejemplo destacado. En este ámbito, Rochina, (2023b), propuso una guía metodológica para la gestión de proyectos de software que combina la metodología Scrum con la plataforma GitHub. Esta propuesta facilitó el proceso de desarrollo de software, resultando en altos niveles de satisfacción entre los usuarios de la guía. Asimismo, Obando (2021), elaboró una guía metodológica para la creación e implementación de un Equipo de Respuesta ante Incidentes de Seguridad Informática (CSIRT). Este enfoque, centrado en la adopción de buenas prácticas y políticas de seguridad, permitió garantizar la integridad de la información y mejorar el registro detallado de las amenazas de seguridad identificadas durante la investigación.

En síntesis, las guías metodológicas son instrumentos ampliamente utilizados en distintas industrias y disciplinas. Su alcance no se limita exclusivamente al desarrollo de software, sino que abarca diversas áreas de estudio y práctica profesional, consolidándose como herramientas esenciales para la mejora continua y la optimización de procesos.

1.2. Ciclo de vida del software

1.2.1. Definición

El ciclo de vida del software en sus siglas SDLC (Software Development Life Cycle), es un marco que establece una serie de etapas fundamentales para crear sistemas informáticos que cumplan altos estándares de calidad y respondan efectivamente a las expectativas del usuario final, este ciclo contempla diferentes fases interconectadas como el análisis de requisitos, diseño, implementación, pruebas y mantenimiento, es importante tomar en cuenta que si se organiza adecuadamente estas fases se minimiza posibles defectos y

garantiza que se comprenda plenamente lo que el cliente busca antes de proceder con la siguiente fase, la flexibilidad del SDLC permite su aplicación tanto en metodologías convencionales como en marcos de trabajo ágiles, adaptándose a las características y requerimientos específicos (Alazzawi et al., 2023).

Para Gondim do Amaral et al. (2023), al definir el SDLC se refiere a todo el proceso de desarrollar un software desde su concepción hasta su retiro, recalando que el éxito de un sistema está relacionado con planear correctamente todo el ciclo de vida junto con una buena elección de arquitectura, esto para no enfrentarse a problemas durante su implementación, además menciona la importancia de tener un análisis de requisitos claro y conciso desde el inicio para no afectar el ciclo, mantener una buena comunicación y calidad durante el desarrollo del sistema.

Al mencionar el SDLC abarca el proceso completo de construir una aplicación informática, que es intrínsecamente complejo en el cual participan múltiples elementos, también incluye fases que pueden ser afectadas por distintas problemáticas que perjudican la capacidad que tiene la aplicación para responder a las necesidades identificadas, es por esto que se debe buscar la manera de gestionar de manera efectiva cada etapa para asegurar un resultado favorable del producto final (Moreno-Rodríguez, 2020).

1.2.2. Modelos del ciclo de vida del software

Los modelos del SDLC son métodos organizados que se emplean para planificar, desarrollar, implementar y mantener software, surgen para abordar la complejidad y la demanda de calidad en el desarrollo, por lo que analizar cada uno de estos modelos es fundamental para elegir el más adecuado que se acople a cada proyecto específico, considerando aspectos como los requisitos, la gestión de riesgos y la intervención de usuarios, una elección apropiada facilitará la entrega de software en el tiempo determinado y con la calidad requerida, el uso de estos modelos es importante en la industria del desarrollo, ya que orientan el proceso de creación de aplicaciones y ayudan a cumplir con las expectativas del cliente (Kyeremeh, 2019).

Tabla 1.*Modelos del ciclo de vida del software - Cascada*

Nombre	Cascada
Descripción	Modelo de desarrollo secuencial donde cada fase se completa antes de iniciar la siguiente, con requisitos definidos desde el inicio.
Ventajas	• Planificación clara del proyecto. • Control preciso de tiempos y costos. • Documentación bien definida
Desventajas	• Baja flexibilidad ante cambios. • Detección tardía de errores. • Retroalimentación limitada del usuario.
En qué casos se utiliza	El modelo en cascada se emplea en ciclos de desarrollo donde se favorecen enfoques metódicos y detallados, particularmente en proyectos cuyos requisitos se encuentran bien definidos desde el inicio y donde se prefiere un desarrollo más estructurado.

Nota. Adaptado de Saravanos & Curinga (2023) y Delgado & Díaz (2020)

Tabla 2.*Modelos del ciclo de vida del software - Espiral*

Nombre	Espiral
Descripción	Modelo iterativo que combina el desarrollo incremental con la gestión de riesgos en cada ciclo del proyecto.
Ventajas	• Gestión temprana de riesgos. • Alta interacción con el cliente. • Flexibilidad ante cambios.

Desventajas	• Difícil de planificar. • Requiere experiencia en gestión de riesgos. • Puede aumentar costos.
En qué casos se utiliza	Este modelo es adecuado para proyectos con requisitos inciertos, proyectos complejos a gran escala dentro de entornos con presupuestos limitados y adaptaciones continuas y para el desarrollo de sistemas críticos, ya que facilita el desarrollo de prototipos, la gestión de cambios de requisitos y las iteraciones antes de finalizar cada fase.

Nota. Adaptado de Delgado & Díaz (2020)

Tabla 3.
Modelos del ciclo de vida del software – Iterativo e incremental

Nombre	Iterativo e incremental
Descripción	Modelo que desarrolla el software en incrementos sucesivos, permitiendo retroalimentación constante del cliente.
Ventajas	• Entrega temprana de funcionalidades. • Adaptación a cambios. • Mejora progresiva del producto.
Desventajas	• Gestión compleja del proyecto. • Dependencia de la participación del cliente.
En qué casos se utiliza	Es eficaz para proyectos con requisitos cambiantes y alta incertidumbre, ya que permite desarrollar prototipos consecutivos mientras el equipo de adapta a las necesidades del cliente.

Nota. Adaptado de Bravo (2019)

Tabla 4.*Modelos del ciclo de vida del software – Modelo en “V”*

Nombre	Modelo en “V”
Descripción	Modelo de verificación y validación derivado del enfoque en cascada, donde cada fase de desarrollo se relaciona directamente con una fase de pruebas correspondiente, asegurando trazabilidad ante requisitos y validación.
Ventajas	• Detección temprana de defectos. • Alta trazabilidad entre requisitos y pruebas. • Mejora la calidad mediante validación estructurada.
Desventajas	• Baja flexibilidad ante cambios en requisitos. • Requiere alta documentación. • Puede aumentar tiempos si se presentan cambios durante el proceso.
En qué casos se utiliza	Adecuado para proyectos con requisitos bien definidos y alta exigencia de calidad, especialmente en entornos regulados. Prioriza la verificación temprana, la documentación detallada y la trazabilidad entre requisitos y pruebas, por encima de la flexibilidad.

Nota. Adaptado de Regulwar et al. (2021)

Tabla 5.*Modelos del ciclo de vida del software - Ágil*

Nombre	Ágil
Descripción	Enfoque iterativo e incremental que prioriza la adaptabilidad al cambio, la colaboración continua con el cliente y la entrega frecuente de software funcional a través de ciclos cortos de desarrollo.
Ventajas	• Alta flexibilidad ante cambios de requisitos.

	• Entrega continua de valor. • Mayor participación con el cliente.
Desventajas	• Menor énfasis en documentación formal. • Requiere alta comunicación. • Difícil de aplicar sin experiencia previa.
En qué casos se utiliza	Efectivo en proyectos con requerimientos variables o inicialmente ambiguos, cuando se busca una entrega rápida del producto y una comunicación fluida con el cliente, resulta adecuado para iniciativas innovadoras que exigen la creación de prototipos y una estrecha colaboración con los usuarios finales.

Nota. Adaptado de Huss et al. (2023)

Tabla 6.
Modelos del ciclo de vida del software - DevOps

Nombre	DevOps
Descripción	Enfoque que integra las actividades de desarrollo y operaciones con el objetivo de mejorar la colaboración, automatizar procesos y acelerar la entrega continua de software de alta calidad.
Ventajas	• Reducción de tiempos de carga. • Mejora la calidad mediante automatización. • Mayor colaboración entre equipos.
Desventajas	• Requiere cambio cultural organizacional. • Dependencia de herramientas especializadas. • Necesita personal con múltiples competencias.

En qué casos se utiliza	Resulta ideal para organizaciones que requieren actualizar frecuentemente su software y adaptarse con rapidez a las condiciones cambiantes del mercado, es particularmente efectivo para desarrollar aplicaciones en la nube y proyectos que exigen una constante integración y despliegue de nuevas funcionalidades.
-------------------------	---

Nota. Adaptado de Felipe & Núñez (2022)

1.2.3. Fases del ciclo de vida del software

Las fases del SDLC son etapas fundamentales que guían el desarrollo de un producto de software desde su concepción hasta su mantenimiento, estas son interdependientes y pueden variar dependiendo del modelo específico adoptado, las principales fases incluyen la planificación, análisis de requisitos, diseño, implementación, pruebas y mantenimiento, cada una de ellas juega un papel crucial en garantizar que el software cumpla con los requisitos establecidos y se desarrolle de manera óptima (P. Alvarez & Cueva, 2023).

Cada fase se detalla brevemente por Alzaid & Khalfan (2022), de la siguiente manera:

- **Planificación:** este periodo inicial se dedica a tres aspectos clave, determinar las metas del proyecto, definir el alcance y asignar los recursos disponibles, así mismo, la participación del liderazgo ejecutivo es vital para asegurar la obtención de toda la información que permitirá un desarrollo exitoso.
- **Análisis de requisitos:** en esta fase se analizan y documentan los requisitos del usuario junto con las funcionalidades necesarias del sistema, resulta vital para asegurar que exista concordancia entre las aspiraciones del cliente y lo que el sistema puede ofrecer.
- **Diseño:** la elaboración de la estructura del sistema ocurre en esta etapa, especificando en detalle cómo operará cada parte del software, un diseño adecuado facilita la optimización de recursos cuando se lleva a cabo la implementación.
- **Implementación:** durante esta fase de codificación, se materializa el producto de software empleando los lenguajes y herramientas apropiadas, se desarrollan los algoritmos y se integran los diferentes componentes requeridos para que el sistema pueda operar según lo planificado.

- Pruebas: implica someter el software a múltiples pruebas con el fin de identificar y corregir cualquier error o fallo, garantizando así que el producto funcione según lo previsto, esta etapa de validación resulta fundamental para asegurar la funcionalidad y calidad del software antes de su puesta en producción.
- Mantenimiento: tras la implementación del software, se requiere un trabajo de mantenimiento constante para solucionar errores que puedan aparecer y aplicar mejoras o actualizaciones, según sea necesario, al realizar esta acción se busca extender la vida útil del software y garantizar su adaptación a las nuevas demandas que vayan surgiendo.

1.2.4. Riesgos en el ciclo de vida del software

El desarrollo de software se compone de múltiples etapas, cada una de ellas presenta distintos desafíos y amenazas que podrían comprometer el resultado final del proyecto. Para garantizar que el software se entregue puntualmente y cumpla con los estándares de calidad esperados, resulta fundamental reconocer y administrar adecuadamente estos posibles obstáculos. A continuación, se detallan algunos de los riesgos identificados durante el ciclo de vida del software, los cuales han sido reportados por distintos autores:

Riesgos asociados a los modelos y definición del proyecto (Kyeremeh, 2019):

- Rigidez de modelos: los modelos tradicionales de desarrollo, como cascada, representa una limitación significativa, dado que estos métodos postergan la entrega de software funcional hasta etapas avanzadas del proyecto, debido a esto, los problemas tienden a manifestarse tardíamente, lo que implica correcciones más costosas y complejas.
- Requisitos mal definidos: cuando los requisitos no están completamente definidos o presentan ambigüedades, existe el riesgo de interpretaciones erróneas que se desemboquen en un producto inadecuado para el usuario.

Bilal et al. (2020), identifica los siguientes riesgos relacionados con la gestión y el equipo de trabajo:

- Falta de participación del usuario: la falta de participación de los usuarios durante la construcción del software puede derivar en una solución que no se

ajusta a sus necesidades concretas, elevando la probabilidad de que el cliente no quede satisfecho con el resultado.

- Problemas de gestión de costos y tiempo: es común que las iniciativas de desarrollo de software experimentan desviaciones en sus presupuestos y cronogramas, lo que puede comprometer su rentabilidad y sostenibilidad financiera.
- Falta de habilidades en el equipo: La falta de personal capacitado y recursos apropiados puede resultar en un deterioro significativo tanto en la calidad del proceso de desarrollo como en la ejecución del proyecto.

Para Khan et al. (2024), es importante examinar también los riesgos de seguridad en el ciclo de desarrollo del software, como se resumen a continuación:

- Identificación de requisitos inseguros: la falta de un plan adecuado para identificar los requisitos de seguridad desde las etapas tempranas puede resultar en vulnerabilidades significativas.
- Validación de salida: el no realizar validaciones adecuadas de los datos de entrada y salida es una fuente común de ataques, como la inyección SQL.

Es importante que se integren consideraciones de seguridad en todas las fases que abarca el SDLC, lo que toma en cuenta no solo la identificación y diseño de requisitos seguros, sino también la implementación de buenas prácticas de prueba efectivas, de esta manera los equipos de desarrollo enfocan sus esfuerzos en los riesgos más críticos que pueden impactar en la seguridad del producto.

1.3. Metodologías ágiles en el desarrollo del software

1.3.1. Definición de metodologías ágiles en el contexto del desarrollo de software

Las metodologías ágiles en el desarrollo de software son enfoques diseñados para facilitar la creación y entrega de productos, basándose en principios como la colaboración, la adaptabilidad, la escalabilidad y la entrega continua de valor, se distinguen por su habilidad para ajustarse de manera ágil a los cambios del mercado y responder eficazmente a las demandas de los clientes, lo cual es fundamental en un entorno altamente dinámico, estas son herramientas clave en el desarrollo de sistemas modernos, que permiten abordar desafíos de gestión y entrega de manera más efectiva. (Merchán et al., 2024).

El desarrollo de estas metodologías responde directamente a la necesidad de romper con los modelos tradicionales que son a menudo rígidos y no permiten ajustes fáciles en función de las necesidades cambiantes de los usuarios, como el modelo en cascada, ofreciendo soluciones más versátiles que facilita la incorporación de modificaciones según evolucionan las expectativas de los usuarios y las condiciones de la industria (Jaramillo Alba et al., 2024).

Scrum, Extreme Programming (XP), Dynamic Systems Development Method (DSDM), Feature-Driven Development (FDD), Adaptive Software Development (ASD) y Kanban, representan las metodologías ágiles más reconocidas, si bien cada una representa características propias, comparten un enfoque común basado en ciclos de trabajo iterativos, comunicación estrecha con el cliente a lo largo del ciclo de vida del software y capacidad de adaptación ante la evolución de los requisitos del proyecto, esto facilita un entorno donde la mejora continua es la norma (Tetteh, 2024).

1.3.2. Principios de las metodologías ágiles

Los principios de las metodologías ágiles están encapsulados en el Manifiesto Ágil, creado en 2001 por expertos del sector tecnológico, marcó un cambio fundamental en la gestión de proyectos de software, su propósito fundamental es promover un enfoque más dinámico que prioriza la adaptabilidad, el trabajo colaborativo y la capacidad de respuesta ante cambios durante el desarrollo (Kakar, 2023).

Este manifiesto se fundamenta en cuatro valores principales, los cuales se describen a continuación:

- Individuos e interacciones sobre procesos y herramientas: se destaca la interacción directa y la colaboración en equipo como factores determinantes para lograr los objetivos del proyecto.
- Software funcional sobre documentación extensiva: se prioriza la entrega de software funcional sobre el exceso de documentación administrativa.
- Colaboración con el cliente sobre negociación de contratos: se busca mantener una interacción cercana con los clientes para comprender a profundidad sus requisitos y expectativas.

- Respuesta ante el cambio sobre seguir un plan rígido: el documento impulsa la capacidad de ajustar y modificar requisitos durante todo el ciclo de desarrollo, sin restricciones por la fase del proyecto.

De acuerdo con Zapata Jaramillo & Yepes Palacio (2024), los 12 principios del manifiesto ágil son:

- La máxima prioridad es satisfacer al cliente: se busca entregar un software valioso de manera temprana y sostenida para garantizar la satisfacción del cliente.
- Aceptación del cambio: los cambios en los requisitos son bienvenidos en cualquier momento del desarrollo, buscando mantener la ventaja competitiva del cliente.
- Entrega frecuente de software funcional: es importante realizar entregas periódicas de versiones operativas del software, ya sea en semanas o meses.
- Colaboración diaria entre negocios y desarrolladores: entre el equipo de desarrollo y los stakeholders debe haber colaboración constante durante todo el proyecto.
- Construir proyectos en torno a individuos motivados: es fundamental ofrecer a los miembros del equipo un entorno adecuado y el apoyo necesario, confiando en su capacidad para realizar su trabajo de manera efectiva.
- Comunicación cara a cara: es la forma más eficiente de intercambiar información dentro de un equipo.
- El software en funcionamiento es la principal medida del progreso: se prioriza la entrega de software que funcione como la principal forma de medir éxito.
- Fomento de un desarrollo sostenible: patrocinadores, desarrolladores y usuarios deben trabajar a un ritmo sostenible que puedan mantener de forma indefinida.
- Atención continua a la excelencia técnica y al buen diseño: se promueve la mejora continua en la calidad del software y en su diseño.

- Simplicidad es esencial: es importante minimizar el trabajo innecesario, centrándose en las tareas que verdaderamente generan valor.
- Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados: los equipos que se autogestionan son más efectivos y producen mejores resultados.
- A intervalos regulares, el equipo debe reflexionar sobre cómo volverse más efectivo: el equipo debe ajustar sus prácticas y procesos basándose en las lecciones aprendidas, con el objetivo de optimizar su eficacia.

1.3.3. Beneficios y desventajas de utilizar metodologías ágiles para desarrollar software

1.3.3.1. Beneficios de utilizar metodologías ágiles

Flores et al. (2021), destacan múltiples beneficios de adoptar metodologías ágiles en el desarrollo de software, estas se detallan a continuación:

- Mejora la flexibilidad y adaptabilidad: los enfoques ágiles facilitan una rápida modificación de los objetivos del proyecto según las demandas del cliente, generando como resultado un producto que responde de manera más precisa a las expectativas y requerimientos del usuario final.
- Aumenta la calidad del producto: mediante un seguimiento continuo del cliente y la realización de evaluaciones periódicas durante todo el ciclo de desarrollo, se logra una detección y corrección anticipada de posibles errores, incrementando sustancialmente la eficacia y confiabilidad del software.
- Incrementa la productividad y la eficiencia: La interacción permanente y el intercambio constante de información entre los integrantes del equipo de desarrollo permite una coordinación precisa, lo que resulta en un incremento sustancial de la eficiencia y el desempeño del grupo.
- Fomenta la transparencia y visibilidad: se impulsa la transparencia organizacional mediante encuentros diarios y mecanismos que permiten una visión clara del avance del proyecto, lo que agiliza la detección y resolución de obstáculos de manera inmediata.

- Mejora la satisfacción del cliente: Al suministrar actualizaciones funcionales de manera constante, se facilita que los clientes experimenten resultados palpables en períodos más cortos, lo que fortalece su participación y genera una percepción positiva del proceso de desarrollo.
- Fomenta la mejora continua: se contemplan espacios de evaluación periódicos que permiten al equipo examinar críticamente su trabajo previo, identificar áreas de desarrollo y promover una filosofía de mejoramiento continuo.

1.3.3.2. Desventajas de utilizar metodologías ágiles

Existen varias desventajas y desafíos que se enfrenta al utilizar metodologías ágiles, Chahal (2023), las define de la siguiente manera:

- Resistencia al cambio: la transición hacia enfoques ágiles suele enfrentar oposición por parte de los equipos y de los actores organizacionales que están acostumbrados a modelos tradicionales de gestión de proyectos, lo que puede generar desconfianza y dificultar la adopción efectiva de nuevas prácticas de trabajo colaborativo.
- Falta de experiencia interna: muchas organizaciones carecen de profesionales con formación y experiencia en metodologías ágiles, como Scrum Masters o entrenadores ágiles, lo que limita la correcta implementación del enfoque y puede provocar interpretaciones erróneas o aplicaciones incompletas del marco ágil.
- Cambio cultural organizacional: la adopción de metodologías ágiles implica una transformación profunda de la cultura organizativa, ya que promueve estructuras más flexibles, colaborativas y autoorganizadas, las cuales pueden entrar en conflicto con sistemas jerárquicos y modelos de gestión tradicionales.
- Integración con procesos existentes: la incorporación de prácticas ágiles junto a otros enfoques de gestión previamente establecidos, como modelos tradicionales o marcos híbridos, puede generar inconsistencias metodológicas si no se gestionan adecuadamente, afectando la coherencia y efectividad de los procesos.
- Dificultad para escalar: aunque las metodologías ágiles han demostrado ser efectivas en equipos pequeños, su implementación en organizaciones de

mayor tamaño o en proyectos complejos presenta desafíos relacionados con la coordinación, la comunicación y la alineación entre múltiples equipos y áreas de trabajo.

1.3.4. Aplicaciones de las metodologías ágiles en la industria del desarrollo de software

Para los autores Estrada et al. (2021), la metodología Scrum transformó el proceso de desarrollo de software al introducir Sprints cortos que permitieron a los equipos mantener un enfoque dinámico y flexible, esta estrategia mejoró significativamente la eficiencia y productividad, facilitando una rápida adaptación a los cambios de requisitos y optimizando el rendimiento del equipo de trabajo, las revisiones y retrospectivas implementadas al término de cada Sprint fomentaron una cultura de mejora continua, permitiendo a los equipos aprender de manera sistemática y ajustar constantemente sus estrategias, esta dinámica de retroalimentación resultó fundamental para elevar la calidad del software, facilitando la entrega de productos más funcionales y optimizados de manera más ágil.

Según Pazmiño (2024), adoptó la metodología Kanban para optimizar la planificación y entrega de productos software en un entorno gubernamental, mediante el uso de tableros visuales, los equipos lograron identificar y resolver eficientemente los obstáculos que impedían un flujo de trabajo continuo y eficaz, la representación del trabajo mediante tarjetas en columnas de estado permitió una gestión óptima de tiempo y recursos , garantizando la alineación del equipo con los objetivos propuestos y proporcionando transparencia en el progreso del proyecto, a estrategia de entrega continua y mejora incremental facilitó la integración y prueba ágil de cambios, generando retroalimentación frecuente de los usuarios finales. Este enfoque permitió realizar ajustes oportunos, asegurando que el producto final se alinea completamente con las expectativas de los stakeholders.

El autor Muñoz (2023), adoptó DevOps para optimizar el proceso de desarrollo de software, implementando una estrategia de integración y entrega continua (CI/CD) que incrementa significativamente la velocidad y calidad de los lanzamientos de producto, la implementación de integración continua permitió la consolidación frecuente del código en un repositorio compartido, fomentando una colaboración más estrecha entre los desarrolladores, la automatización de pruebas e integraciones optimizó la validación del código, minimizando significativamente el tiempo destinado a correcciones y mejoras, el utilizar esta metodología posibilitó una monitorización y retroalimentación continua que permitió al equipo identificar y resolver problemas de rendimiento instantáneamente, esta estrategia mejoró la estabilidad y eficiencia del software, garantizando aplicaciones completamente optimizadas.

1.3.5. Metodología ágil SCRUM

Scrum es una metodología ágil que transforma la manera en que los equipos colaboran durante el desarrollo de productos, este sistema permite que los grupos de trabajo comprendan mejor su dinámica interna, estableciendo bases sólidas para la comunicación y el trabajo conjunto, a través de revisiones periódicas y ajustes continuos, Scrum garantiza la evolución constante del proceso, resultando en productos más exitosos y clientes más satisfechos (Hernández & Beltrán, 2020).

1.3.5.1. Roles de Scrum

Kadenic et al. (2023), define los siguientes roles de Scrum:

Product Owner: es responsable de gestionar y priorizar el Product Backlog, asegurando que los requisitos y actividades del equipo maximicen el valor del producto. Asimismo, actúa como enlace principal entre el equipo de desarrollo y las partes interesadas.

Scrum Master: es responsable de implementar y fortalecer los procedimientos del equipo según los lineamientos de la metodología Scrum. Su función incluye promover una comunicación eficaz y el trabajo colaborativo entre los miembros, garantizando que se cumplan los eventos de Scrum y resolviendo cualquier obstáculo que pueda impactar negativamente en el desempeño del grupo.

Development team: Los integrantes del equipo de desarrollo tienen la responsabilidad de elaborar cada componente de una versión funcional y aprovechable del producto en cada ciclo de trabajo.

1.3.5.2. Fases de Scrum

Scrum divide el trabajo en etapas específicas que ayudan al equipo a ser más eficiente, esta estructura permite que los proyectos avancen de forma ordenada y clara, facilitando que se alcancen todos los objetivos propuestos. Según Estrada, Núñez, et al. (2021), estas fases principales son:

Planificación del Sprint: El equipo define qué tareas del Product Backlog trabajará durante el Sprint y construye el Sprint Backlog.

Ejecución del Sprint: Se desarrollan, prueban y completan las tareas planificadas para cumplir el objetivo del Sprint.

Revisión del Sprint: Se presenta el incremento del producto a los interesados y se recoge retroalimentación para futuras mejoras.

Retrospectiva del Sprint: El equipo reflexiona sobre su desempeño, identifica mejoras y propone acciones para el siguiente Sprint.

1.3.5.3. Artefactos de Scrum

De acuerdo con Garcia et al. (2022), el éxito de un proyecto Scrum depende en gran medida de instrumentos clave denominados artefactos, su importancia radica en que proporcionan transparencia al avance del trabajo, posibilitan un seguimiento efectivo y dan la flexibilidad necesaria para implementar modificaciones cuando se requiera, estos artefactos se resumen de la siguiente manera:

Product Backlog: Es la lista priorizada de requisitos y tareas necesarias para desarrollar el producto, la cual se actualiza continuamente y es gestionada principalmente por el Product Owner junto al equipo.

Sprint Backlog: Es el conjunto de elementos seleccionados del Product Backlog que el equipo trabajará durante un Sprint, y solo el equipo de desarrollo puede modificarlo.

Incremento: Es el resultado de las funcionalidades completadas en uno o varios Sprints, que deben estar terminadas, verificadas y listas para su uso.

1.4. Herramientas de gestión para proyectos de desarrollo de software

1.4.1. Definición

Las herramientas de gestión en el desarrollo de software son aplicaciones y metodologías diseñadas para optimizar la planificación, monitoreo y supervisión de proyectos tecnológicos, estas soluciones resultan fundamentales para gestionar aspectos críticos como el alcance, tiempo, presupuesto y calidad del proyecto, permitiendo a los equipos de desarrollo estructurar y ejecutar sus iniciativas de manera más eficiente, entre sus componentes principales se encuentran herramientas como diagramas de Gantt, estructuras de diseño del trabajo y diagramas de red, que facilitan la visualización, organización y seguimiento de los proyectos de software (Assalaarachchi et al., 2024).

Las herramientas de gestión en desarrollo de software surgen para optimizar la eficiencia y calidad de proyectos tecnológicos, estas soluciones permiten a los equipos planificar y controlar aspectos clave del desarrollo, mejorando la rapidez mediante diferentes metodologías, su valor principal radica en mejorar la satisfacción del cliente, potenciar la colaboración interdisciplinaria y adaptarse a diversas arquitecturas de software, existen herramientas como, GitHub, Jira, Microsoft Teams, Miro, entre otras que buscan impulsar la interacción entre miembros, mientras el Desarrollo de Software Adaptativo (ASD) promueve una comunicación más efectiva y productos de alta calidad, las herramientas de visualización como UML y BPMN mantiene a los integrantes alineados con los objetivos, consolidándose como elementos esenciales para una gestión ágil y colaborativa en el desarrollo de software (Nikitashin et al., 2024).

1.4.2. Beneficios del uso de herramientas de gestión de proyectos en el desarrollo de software

En un mundo digital en constante evolución, las herramientas de gestión de proyectos han surgido como recursos fundamentales para equipos de desarrollo, estas soluciones tecnológicas transforman la colaboración profesional, potenciando la eficacia y agilidad de los grupos de trabajo en proyectos cada vez más complejos. Choudhury et al., (2020) menciona beneficios de utilizar estas herramientas detallados a continuación:

- Automatización del ciclo de desarrollo: los ingenieros pueden simplificar su trabajo al automatizar las diferentes etapas del desarrollo de software, desde el momento inicial de planificación hasta la implementación y seguimiento del código, esta herramienta ayuda a minimizar errores y agilizar cada fase del proceso.
- Integración continua: permiten a los equipos descomponer proyectos grandes en tareas más pequeñas y manejables, al trabajar simultáneamente en diferentes frentes y luego integrar todo el trabajo, los equipos logran una mayor eficiencia y aceleran significativamente los tiempos de desarrollo.
- Colaboración asíncrona: impulsan la colaboración sin límites de tiempo, permitiendo que los miembros del equipo trabajen desde diferentes partes del mundo sin necesidad de coincidir simultáneamente, esta flexibilidad es clave en entornos laborales remotos, ya que permite integrar talento diverso y eliminar la barrera de las reuniones presenciales tradicionales.

- Visibilidad de contribuciones: ofrecen una visibilidad completa de las contribuciones de cada miembro del equipo, promoviendo una comunicación transparente y fluida, al hacer visibles los aportes de todos, se garantiza que cada integrante esté alineado con los objetivos del proyecto y se previene la duplicación innecesaria de esfuerzos.

1.4.3. Comparación de herramientas de desarrollo de software

Las herramientas de desarrollo de software son muy variadas ya sea por sus funciones como por el contexto en el que se les puede dar uso, para este proyecto de investigación se seleccionaron principalmente 5 herramientas muy destacadas en el ámbito del desarrollo de software, entre ellas están Github, Gitlab, BitBucket, Trello y Asana.

1.4.3.1. GitHub

GitHub se posiciona como una de las plataformas más completas para el desarrollo de software colaborativo, integrando herramientas de control de versiones, gestión de proyectos, automatización de flujos de trabajo y seguridad avanzada, entre sus principales funcionalidades destacan: GitHub Copilot, que utiliza inteligencia artificial para sugerir líneas de código completas; GitHub Actions, que automatiza flujos CI/CD; GitHub Codespaces, entornos de desarrollo en la nube; y GitHub Advanced Security, con capacidades para detectar y remediar vulnerabilidades con tecnologías de análisis estático (CodeQL) y escaneo de secretos (GitHub, 2024).

Desde el punto de vista económico, GitHub promueve la optimización de recursos mediante automatización y herramientas integradas que reducen la dependencia de software adicional, mejoran la productividad y disminuyen los tiempos de entrega. Su modelo de licenciamiento flexible permite a las empresas escalar operaciones de manera eficiente, mientras asegura la protección de la cadena de suministro de software, GitHub también fomenta la participación de la comunidad a través de iniciativas como GitHub Sponsors, facilitando el financiamiento de proyectos de código abierto (GitHub, 2024).

1.4.3.2. GitLab

GitLab es una plataforma integral de DevSecOps impulsada por inteligencia artificial (IA) que centraliza las funciones clave del ciclo de vida del software: planificación, desarrollo, seguridad, implementación y monitoreo. La herramienta destaca por su capacidad de consolidar múltiples procesos en una única plataforma,

eliminando la necesidad de herramientas adicionales como Jira, Jenkins, GitHub o Snyk, entre sus principales características se incluyen la planificación de proyectos, gestión de código fuente, automatización de pruebas, análisis de seguridad, cumplimiento normativo y monitoreo en tiempo real. Su uso de IA, a través de funciones como generación automática de pruebas, resúmenes y análisis de código, mejora significativamente la productividad y reduce los tiempos de desarrollo.

Desde un punto de vista económico, el mismo autor señala que GitLab promete una reducción de costos operativos, con un ahorro estimado de \$150,000 anuales en consolidación de herramientas y procesos. Además, permite disminuir el tiempo de incorporación de nuevos proyectos en un promedio de 20 horas, estas ventajas la posicionan como una solución eficiente para empresas que buscan optimizar su flujo de trabajo mientras reducen su inversión en licencias y herramientas de terceros (GitLab, 2024).

1.4.3.3. Bitbucket

Bitbucket (2024), una plataforma integral desarrollada por Atlassian, optimiza el ciclo de vida del desarrollo de software al integrar control de versiones, CI/CD (integración y entrega continua) y colaboración en código en un único entorno. Su enfoque se centra en acelerar la productividad mediante herramientas de inteligencia artificial, como descripciones automáticas de pull requests y sugerencias de edición generativa, además de fomentar la colaboración a través de integraciones nativas con Jira y Confluence.

Entre sus características destacadas se incluyen mecanismos de seguridad avanzada, como cifrado en tránsito y en reposo, permisos granulares y control de despliegue, así como soporte para proyectos de cualquier escala. Asimismo, la plataforma ofrece planes adaptados a distintos tamaños de equipos, desde opciones gratuitas para pequeñas organizaciones hasta versiones Premium con mayor almacenamiento, minutos de compilación y funciones impulsadas por inteligencia artificial.

Adicionalmente, Bitbucket Cloud permite migraciones desde Data Center sin costos iniciales, garantizando continuidad y escalabilidad para equipos que buscan aprovechar las ventajas de la nube, además, con su integración en la plataforma Atlassian, Bitbucket se posiciona como líder en herramientas DevOps según el

Cuadrante Mágico de Gartner, asegurando a las organizaciones un entorno robusto y eficiente para la entrega de software de alta calidad (Bitbucket, 2024).

Bitbucket en Scrum

Bitbucket es una plataforma de alojamiento remoto que permite gestionar y controlar versiones del código fuente, facilitando el trabajo colaborativo, la administración de repositorios y el control de accesos. Estas funciones ayudan a organizar el desarrollo, registrar cambios y mantener un control estructurado del código durante los Sprints (Chakraborty & Aithal, 2022).

Según Shekar, (2024), Bitbucket ofrece varios beneficios como el control de acceso mediante permisos específicos para proteger el código, la gestión de flujos de trabajo que mejora la coordinación del equipo, y el uso de *pull requests* que facilitan la revisión colaborativa del código. Además, permite automatizar procesos mediante su API REST, integrarse con herramientas como Jira y Jenkins para implementar CI/CD, ofrecer diferentes opciones de despliegue (Server, Data Center y Cloud) y contar con Pipelines, que automatiza la construcción, prueba e implementación del software.

1.4.3.4. Trello

Trello (2024) es una herramienta de productividad diseñada para facilitar la organización de tareas y la colaboración en equipo, mediante una interfaz intuitiva basada en tableros, listas y tarjetas, esta plataforma permite a los equipos gestionar proyectos de forma visual y ágil, moviendo tarjetas entre listas para reflejar el estado de las tareas y adaptándose a diferentes flujos de trabajo, desde los más simples hasta los más complejos.

De acuerdo con la misma fuente, Trello incorpora funciones como vistas de cronograma, calendario, tabla y mapa, las cuales permiten planificar y supervisar proyectos de manera eficiente, mejorando la visibilidad de plazos y tareas. Asimismo, la automatización mediante Butler contribuye a reducir el trabajo manual, mientras que las integraciones con herramientas como Slack y Google Drive optimizan la colaboración entre equipos. Además, la plataforma se ofrece en diversos planes de precios, que van desde una opción gratuita básica hasta soluciones avanzadas para empresas, lo que la convierte en una herramienta versátil y adaptable a distintos contextos organizacionales.

1.4.3.5. Asana

Asana (2024) es una plataforma de gestión del trabajo que permite vincular tareas con objetivos estratégicos y automatizar flujos de trabajo, facilitando una mayor eficiencia y colaboración dentro de las organizaciones. La herramienta favorece la coordinación entre equipos y departamentos mediante una interfaz visual intuitiva, la cual simplifica la planificación, el seguimiento y la ejecución de proyectos.

De acuerdo con la misma fuente, Asana integra inteligencia artificial a través de la tecnología *Work Graph*®, la cual contribuye a identificar desafíos, optimizar la productividad y ofrecer visibilidad en tiempo real del progreso de los objetivos. Asimismo, la plataforma permite la creación de flujos de trabajo personalizados y la generación de informes que facilitan la detección de cuellos de botella.

Adicionalmente, Asana se caracteriza por contar con seguridad de nivel empresarial y cumplir con normativas globales de privacidad. Gracias a más de 300 integraciones y a la disponibilidad de distintos planes —desde opciones gratuitas hasta versiones avanzadas con gestión de recursos, portafolios y automatización a gran escala—, la plataforma se adapta a organizaciones de diversos tamaños y sectores, apoyando una gestión del trabajo más eficiente y alineada con los objetivos estratégicos.

En resumen, existen diversas herramientas de gestión de proyectos de desarrollo de software, al tomar en cuenta las herramientas seleccionadas se puede generar la tabla 7

Tabla 7.
Comparativa entre herramientas de desarrollo de software

Característica	GitHub	GitLab	Bitbucket	Trello	Asana
Tipo de herramienta	Control de versiones y colaboración de código	Control de versiones y DevOps	Control de versiones y repositorios de código	Gestión de tareas y proyectos	Gestión de proyectos y colaboración
Interfaz de usuario	Intuitiva y centrada en desarrollo	Completa y flexible, centrada en DevOps	Similar a GitHub, orientada a repositorios	Tableros estilo Kanban	Visual y orientada a proyectos
Automatización	Acciones de GitHub, integraciones CI/CD	CI/CD y DevOps integrados, automatización avanzada	Pipelines de Bitbucket	Limitada (Power-Ups)	Avanzada (IA y Work Graph®)
Integraciones	Amplia (GitHub Marketplace)	Extensa (GitLab Marketplace)	Integración con herramientas Atlassian	Más de 250 integraciones (Power-Ups)	Más de 300 integraciones
Gestión de proyectos	Issues, proyectos, y GitHub Projects	Issues, epics, y tableros Kanban	Issues, proyectos y tableros de tareas	Tableros de tareas y proyectos	Tableros de proyectos y gestión de tareas
Gestión de código	Repositorios Git, control de versiones	Repositorios Git, CI/CD y DevOps	Repositorios Git, control de versiones	No aplica	No aplica
Colaboración en equipo	Pull requests, revisión de código	Merge requests, revisión de código	Pull requests, revisión de código	Colaboración mediante tarjetas	Comentarios y asignación de tareas
Planificación de tareas	Issues y proyectos	Issues, epics, y tableros de planificación	Issues y tableros de tareas	Tableros Kanban y listas de tareas	Tareas, sub-tareas y proyectos
Precios	Gratuito (planes de pago para características avanzadas)	Gratuito y planes de pago (Self-hosted y SaaS)	Gratuito y planes de pago (integración con Atlassian)	Plan gratuito y de pago (Power-Ups)	Plan gratuito y de pago
CI/CD	GitHub Actions	GitLab CI/CD	Bitbucket Pipelines	No aplica	No aplica
Usos comunes	Desarrollo de software, código colaborativo	Desarrollo de software, CI/CD, DevOps	Integración con Jira y desarrollo de software	Tareas sencillas, proyectos ágiles	Proyectos de equipo, gestión de campañas

Seguridad y cumplimiento	Opciones avanzadas de seguridad en planes pagos	Seguridad y cumplimiento en planes avanzados	Seguridad y cumplimiento con Atlassian	Básica (seguridad de tablero)	Alta, cumplimiento de normativas globales
Escalabilidad	Ideal para proyectos de todos los tamaños	Escalable, ideal para equipos grandes	Ideal para proyectos pequeños a grandes	Ideal para equipos pequeños a medianos	Escalable para equipos medianos a grandes
Informes y analíticas	Estadísticas de repositorios y contribuciones	Métricas avanzadas de CI/CD y proyectos	Informes y estadísticas de repositorios	Básicos (con Power-Ups)	Avanzados con paneles de informes y objetivos
Interoperabilidad	Integración con otras herramientas de desarrollo	Integración con herramientas DevOps y CI/CD	Integración con herramientas de Atlassian	Integraciones con herramientas de terceros	Integraciones amplias y API

Nota. Adaptado de GitHub (2024), GitLab (2024), Bitbucket (2024), Trello (2024) y Asana (2024)

1.4.4. Normas ISO en la gestión de proyectos de desarrollo de software

Las normas ISO (Organización Internacional de Normalización) representan un conjunto de regulaciones internacionales creadas por especialistas para garantizar la calidad, eficiencia y seguridad en diversos ámbitos, estos estándares globales proporcionan lineamientos técnicos que abarcan múltiples sectores, desde procesos de gestión organizacional hasta aspectos relacionados con la protección de datos y prácticas sostenibles, el objetivo principal de estas normas es crear un sistema de referencia compartido que facilite las transacciones globales y el entendimiento mutuo entre empresas y países, al incorporar estos estándares, las organizaciones pueden demostrar su compromiso con la mejora permanente y la calidad, lo que resulta en un incremento de la confianza del público y una diferenciación estratégica en su sector (Bahrudin, 2023). Las normas ISO son un referente global que establece criterios de calidad en diversos sectores como tecnología, alimentación, salud y medio ambiente, aunque su adopción es voluntaria, muchas organizaciones las implementan para mejorar sus procesos, cumplir requisitos de mercado y diferenciarse competitivamente, demostrando un compromiso con la excelencia y la innovación.

Según Soares et al. (2024), las normas ISO juegan un papel fundamental en el desarrollo de software, estas normativas ofrecen beneficios estratégicos para las organizaciones, entre los que destacan:

- **Establecimiento de Normas de Calidad:** las normas ISO configuran un sistema de evaluación que permite medir de manera sistemática la calidad del software, mediante la definición de requisitos específicos que abarcan aspectos como el funcionamiento, la confiabilidad, la experiencia de usuario y los protocolos de seguridad.
- **Mejora continua:** la aplicación de estos lineamientos incentiva una estrategia de perfeccionamiento continuo en el desarrollo de aplicaciones, asegurando su capacidad de adaptación y vigencia ante las transformaciones tecnológicas y las nuevas necesidades del entorno.
- **Mejor comunicación:** establecen un vocabulario estandarizado que optimiza el entendimiento mutuo entre los diferentes actores involucrados en un proyecto de software, mejorando significativamente la comunicación y la cooperación.

- Cumplimiento y certificación: es una práctica habitual que las empresas emplean las certificaciones ISO como medio para demostrar su compromiso con los estándares del sector, lo cual se convierte en un factor determinante para construir la confianza de los clientes y ampliar la aceptación de sus soluciones tecnológicas en el mercado.
- Minimización de riesgos: mediante la implementación de normativas que garantizan calidad y seguridad, las empresas pueden mitigar los peligros asociados a la creación de software, evitando inconvenientes como errores operativos, vulneraciones de sistemas y la consecuente insatisfacción de los clientes.

Así, estos estándares no solo contribuyen a asegurar la calidad del software, sino que también respalda su desarrollo y mantenimiento de forma eficiente y sostenible. Mas allá de ser un requisito formal, se configuran como una filosofía de trabajo que impulsa la innovación, garantiza la satisfacción del cliente y promueve una cultura de excelencia en el ámbito del desarrollo de software.

1.4.4.1. Norma ISO/IEC 25010

De acuerdo con Syamaidzar & Sofyan (2024), el estándar internacional ISO/IEC 25010, parte del marco SQuaRE (Requisitos y Evaluación de la Calidad de Sistemas y Software), está dedicado a la evaluación de la calidad del software, este estándar establece un modelo de calidad compuesto por ocho características principales, cada una con sus subcaracterísticas, esenciales para evaluar el rendimiento, la estabilidad y la efectividad del software.

- Adecuación funcional: evalúa si el software proporciona las funciones adecuadas que cumplen con los requisitos establecidos.
- Eficiencia de desempeño: se refiere a la capacidad del software para funcionar de manera efectiva, gestionando adecuadamente los recursos y respondiendo en tiempos apropiados.
- Compatibilidad: indica la capacidad del software para operar en diferentes entornos y con otros sistemas, facilitando la interoperabilidad.

- Usabilidad: evalúa cuán fácil y satisfactorio es usar el software para los usuarios finales, considerando aspectos como la estética de la interfaz y la accesibilidad.
- Fiabilidad: se enfoca en la capacidad del software para mantener un rendimiento estable y confiable bajo condiciones establecidas, incluyendo aspectos como la recuperación de errores y la disponibilidad.
- Seguridad: se centra en cómo el software protege la información y controla el acceso a sus funciones, garantizando la confidencialidad e integridad de los datos.
- Mantenibilidad: indica cuán fácil es modificar el software para corregir errores, mejorar su rendimiento o adaptarlo a nuevos requisitos.
- Portabilidad: evalúa la facilidad con la que el software puede ser transferido de un entorno a otro, adaptándose a nuevos sistemas o plataformas (Syamaidzar & Sofyan, 2024).

Esta norma es especialmente valiosa ya que proporciona un enfoque integral para medir y mejorar la calidad de los sistemas de software. Syamaidzar & Sofyan (2024) realizaron un análisis donde se ha constatado que el uso de ISO/IEC 25010 permite obtener una comprensión más profunda de la calidad global de los sistemas, lo que resulta beneficioso para las instituciones que buscan cumplir con estándares elevados de calidad en sus sistemas informáticos.

CAPÍTULO 2

2.1. Elaboración de la guía metodológica

2.1.1. Introducción

Una vez realizada una investigación exhaustiva sobre los conceptos y prácticas relacionadas con el marco de trabajo Scrum y las herramientas de gestión de código, se presenta a continuación una guía metodológica de procedimientos y métodos para aplicar Scrum con la herramienta Bitbucket en el desarrollo de proyectos de software.

Esta guía está diseñada para aprovechar las capacidades de Bitbucket como herramienta principal en la gestión de proyectos de software, integrándola con el marco de trabajo Scrum, el documento proporciona una secuencia detallada de pasos para implementar las actividades esenciales de Scrum mediante Bitbucket, enfatizando cómo esta sinergia potencia la optimización de procesos y fomenta una colaboración más eficiente entre los miembros del equipo.

El propósito de esta propuesta es proporcionar una guía práctica que permita a los equipos de desarrollo implementar Scrum de manera eficiente, aprovechando las funcionalidades de Bitbucket para gestionar de forma efectiva las tareas, el código y la colaboración del equipo. A través de un enfoque didáctico, la guía abarca desde las etapas iniciales de un proyecto hasta su finalización y entrega, brindando instrucciones claras y aplicables que facilitan la organización, la comunicación y el seguimiento del progreso del proyecto.

Este documento puede ser de gran utilidad para cualquier persona que desee iniciar en la gestión de proyectos utilizando la herramienta Bitbucket bajo el marco de trabajo Scrum, beneficiando tanto al proyecto como a los equipos involucrados en el desarrollo.

2.1.2. Objetivo

Crear una guía metodológica para aplicar Scrum con Bitbucket para mejorar la gestión de proyectos.

2.1.3. Alcance

Proporcionar una guía metodológica que sirva como herramienta de referencia para desarrolladores que busquen gestionar sus proyectos con el marco de trabajo Scrum,

aprovechando las funcionalidades de Bitbucket para optimizar la colaboración del equipo y el control de procesos de desarrollo.

2.1.4. Terminología

Se proporciona un listado de términos propios de Bitbucket, junto con sus equivalencias dentro del marco de trabajo Scrum. Esto permite establecer una relación clara entre los componentes de la herramienta y los elementos fundamentales de Scrum, como se detalla en la Tabla 8.

Tabla 8.
Terminología

Término	Significado
Sprint	Iteración de duración fija en la que el equipo Scrum desarrolla un conjunto de funcionalidades planificadas.
Product Backlog	Lista priorizada de requisitos y funcionalidades del producto.
Sprint Backlog	Tareas seleccionadas del Product Backlog
Repositorio	Espacio centralizado donde se almacena y gestiona el código fuente del proyecto.
Commit	Registro de un cambio realizado en el código fuente dentro del repositorio
Pull request	Solicitud para revisar e integrar cambios de código en una rama principal del repositorio.

Nota. Elaboración propia

2.1.5. Responsables

Autora: Dayana Córdova

Director: MSc. Mauricio Rea

2.1.6. Desarrollo

2.1.6.1. Fase 1. Pre-Juego

Configuración inicial de Git y vinculación con Bitbucket

Configuración global de Git

Antes de iniciar un proyecto bajo el marco de trabajo Scrum utilizando Bitbucket, es necesario configurar los datos globales de Git. Esta configuración permite que cada contribución realizada en el repositorio sea identificable y rastreable, lo cual resulta fundamental en entornos de desarrollo colaborativo. Para ello, desde una línea de comandos (cmd, consola o terminal), se ejecuta el comando correspondiente:

```
git config --global user.name "Usuario"
```

Este comando establece el nombre que se asociará a cada commit realizado. Es importante utilizar un nombre que represente al miembro del equipo.

```
git config --global user.email " usuario@ejemplo.com"
```

Este comando registra el correo electrónico asociado al usuario en Git. Este correo debe coincidir con el utilizado en la cuenta de Bitbucket.

Inicializar un repositorio local

El siguiente paso consiste en inicializar un repositorio local dentro del directorio del proyecto, lo cual habilita el uso del control de versiones. Para ello, se ejecuta el siguiente comando:

```
git init
```

Este comando crea un repositorio vacío en la carpeta actual. Es importante que su ejecución se realice en el directorio principal del proyecto, con el fin de asegurar una correcta gestión del control de versiones.

Agregar archivos al repositorio

Para que Git comience a rastrear los archivos del proyecto, es necesario añadirlos al área de preparación.

```
git add .
```

Este comando mueve todos los archivos del proyecto al área de preparación (staging area), preparándolos para ser confirmados.

Realizar el primer Commit

Un commit representa un punto en la historia del proyecto, guardando el estado de los archivos.

```
git commit -m "Primer commit"
```

Es importante escribir un mensaje claro y descriptivo. Por ejemplo, el mensaje del commit puede incluir el propósito del cambio o la relación con un ítem del Backlog.

1. Creación de un repositorio en Bitbucket

El siguiente paso consiste en establecer un espacio de trabajo (*workspace*) en Bitbucket, el cual funciona como el contenedor principal para la gestión de proyectos y repositorios. Este espacio permite organizar el código fuente, las ramas y los elementos asociados al proyecto, además de facilitar la colaboración entre los miembros del equipo Scrum.

Acceso a la Plataforma de Bitbucket

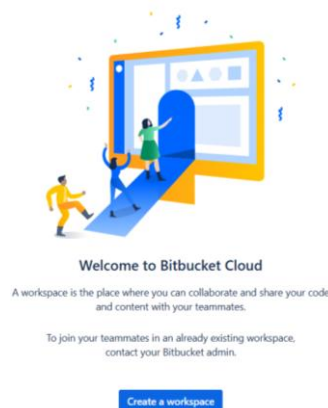
- Se accede al portal de Bitbucket utilizando una cuenta previamente configurada.
- Una vez dentro de la plataforma, se localiza la pantalla de bienvenida de Bitbucket Cloud, en la cual se presenta la opción para crear un nuevo espacio de trabajo (*workspace*).

Creación del espacio de trabajo

Se selecciona el botón “**Create a workspace**”.

Figura 3.

Página de creación del workspace



Nota. Elaboración Propia.

Configuración del espacio de trabajo en Bitbucket

Para continuar con la creación del espacio de trabajo en Bitbucket, se configura la información clave que permitirá identificar el proyecto y su contexto.

Selección del correo electrónico

En el formulario que se muestra en pantalla, se confirma el correo electrónico que será asociado al espacio de trabajo. Este correo se utiliza para la gestión de notificaciones y la administración del proyecto. En el presente ejemplo, se emplea el correo **usuario@ejemplo.com**.

Definición del nombre del espacio de trabajo

En el campo correspondiente, se asigna un nombre único al espacio de trabajo. En el presente ejemplo, se utiliza el nombre **testproject1501**.

El nombre del espacio de trabajo debe ser único y solo puede contener letras minúsculas, números, guiones y guiones bajos. Este identificador resulta fundamental para la construcción de la URL del repositorio y para la correcta organización de los proyectos dentro de Bitbucket.

Aceptación de términos y condiciones

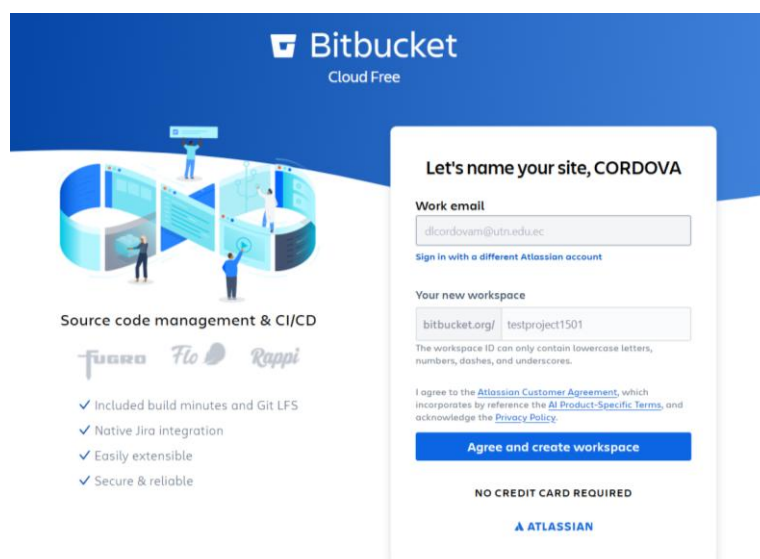
Posteriormente, se revisan y aceptan los acuerdos de uso mediante la selección de la casilla que confirma la conformidad con los términos del cliente de Atlassian y la política de privacidad.

Creación del espacio de trabajo

Finalmente, se selecciona el botón “**Agree and create workspace**” para completar el proceso. Este paso valida la información ingresada y genera el espacio de trabajo en la cuenta asociada.

Figura 4.

Formulario inicial de creación el workspace



The image shows the Bitbucket Cloud Free workspace creation interface. On the left, there's a blue header with the Bitbucket logo and 'Cloud Free' text. Below it, a 3D illustration shows people working in a modern office space. Text below the illustration reads 'Source code management & CI/CD' and lists logos for 'Fugro', 'Flo', and 'Rappi'. A list of features follows: 'Included build minutes and Git LFS', 'Native Jira integration', 'Easily extensible', and 'Secure & reliable'. On the right, a white form titled 'Let's name your site, CORDOVA' contains a 'Work email' field with the value 'dcordova@utn.edu.ec' and a 'Sign in with a different Atlassian account' link. Below this is a 'Your new workspace' field with the value 'bitbucket.org/ testproject1501'. A note states: 'The workspace ID can only contain lowercase letters, numbers, dashes, and underscores.' There is a checkbox for 'I agree to the Atlassian Customer Agreement, which incorporates by reference the At Product-Specific Terms, and acknowledge the Privacy Policy.' A blue button labeled 'Agree and create workspace' is at the bottom of the form. Below the button, it says 'NO CREDIT CARD REQUIRED' and 'A ATlassian'.

Nota. Elaboración Propia.

Creación de un nuevo proyecto en Bitbucket

Para iniciar la gestión de un proyecto dentro del espacio de trabajo previamente creado, se siguen los siguientes pasos:

Acceso a Bitbucket

Se accede al sitio web oficial de Bitbucket a través de la dirección **bitbucket.org** utilizando un navegador web. Es necesario que la sesión se encuentre iniciada con la cuenta que contiene el espacio de trabajo previamente creado.

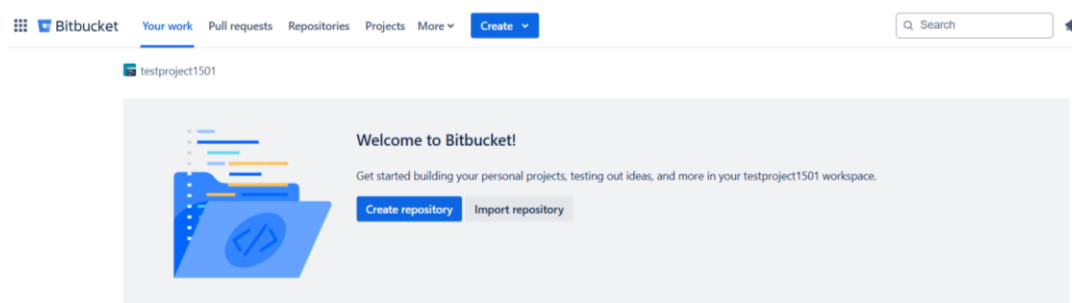
Navegación a la sección de proyectos

Desde la página principal del espacio de trabajo, se localiza y selecciona la pestaña **“Projects” (Proyectos)** ubicada en el menú superior. Esta acción dirige a la sección destinada a la gestión de proyectos.

Creación de un nuevo proyecto

En la sección de proyectos, se selecciona el botón **“Create a project” (Crear un proyecto)**. Esta acción despliega un formulario en el cual se definen los detalles del nuevo proyecto, tales como el nombre, la clave, la descripción y la configuración de privacidad.

Figura 5.
Página de visualización del workspace



Nota. Elaboración Propia.

Configuración del proyecto en Bitbucket

Una vez en el formulario de creación del proyecto, se completan los campos correspondientes con la información requerida.

Nombre del proyecto

En el campo correspondiente, se ingresa el nombre del proyecto que se desea crear. En el presente ejemplo, se utiliza el nombre **“testProject1501”**, el cual constituirá la identificación principal del proyecto dentro del espacio de trabajo.

Alias del proyecto

Se define un alias único y breve que represente al proyecto. En el presente ejemplo, se utiliza **“TP150”** como identificador abreviado. Este alias facilita la identificación rápida del proyecto en distintas vistas y reportes dentro de la plataforma.

Descripción del proyecto

Se incluye una descripción breve pero informativa que resuma el propósito u objetivos del proyecto. Esta descripción permite a los colaboradores comprender el contexto y la finalidad del proyecto.

Tipo de privacidad

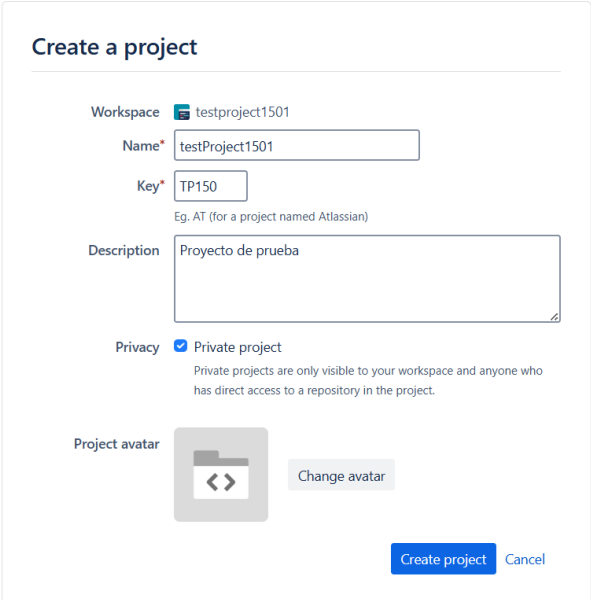
Se selecciona la configuración de privacidad según las necesidades del proyecto. En el presente ejemplo, se opta por la opción **“Privado”**, con el fin de restringir el acceso únicamente a los miembros autorizados del equipo.

Confirmación

Finalmente, se selecciona el botón **“Create project”** (Crear proyecto) para completar la configuración. A continuación, Bitbucket registra el proyecto con las configuraciones especificadas.

Figura 6.

Formulario de creación del proyecto



Create a project

Workspace  testproject1501

Name*

Key*

Eg. AT (for a project named Atlassian)

Description

Privacy ☒ Private project
Private projects are only visible to your workspace and anyone who has direct access to a repository in the project.

Project avatar  [Change avatar](#)

[Create project](#) [Cancel](#)

Nota. Elaboración Propia.

Creación de un repositorio en el proyecto

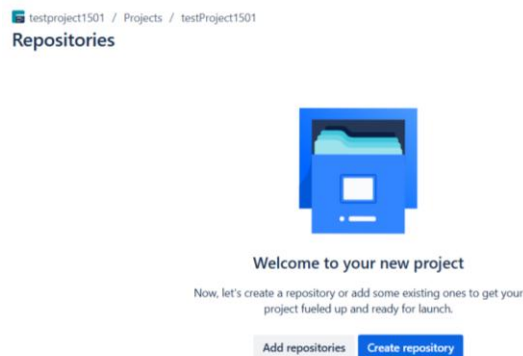
Una vez completada la creación del proyecto, Bitbucket redirige automáticamente a la página de descripción general del mismo. En esta sección se gestionan los componentes y configuraciones asociadas al proyecto.

Crear un repositorio

Para iniciar la creación de un nuevo repositorio dentro del proyecto, se localiza y selecciona el botón **“Create repository”** (Crear repositorio), el cual generalmente se encuentra en una posición destacada dentro de la página.

Figura 7.

Página de creación del repositorio



Nota. Elaboración Propia.

Formulario de creación de repositorio en Bitbucket

Una vez seleccionado el botón **“Create repository”**, se despliega un formulario en el que se configura la información básica del nuevo repositorio. A continuación, se describen los campos y opciones que deben completarse:

Seleccionar el proyecto

El formulario solicita seleccionar el proyecto al que se asociará el repositorio. En el presente ejemplo, se elige **“testProject2718”** dentro de la lista desplegable correspondiente.

Nombre del repositorio

En el campo correspondiente, se especifica el nombre del repositorio. En el presente ejemplo, se utiliza “**testP001**” como identificador del repositorio.

El nombre asignado debe ser único dentro del proyecto seleccionado, con el fin de evitar conflictos en la gestión de los repositorios.

Nivel de acceso

Dado que el proyecto ha sido configurado como privado, el repositorio hereda automáticamente este nivel de privacidad. Por lo tanto, no es necesario realizar configuraciones adicionales relacionadas con el acceso.

En cuanto al archivo **README (opcional)**, el formulario permite incluir un archivo inicial en el repositorio. Para ello, se puede optar por una de las siguientes alternativas:

- Utilizar una plantilla prediseñada.
- Incorporar un tutorial básico.
- No incluir un archivo README en esta etapa.

Nombre de la rama por defecto

En este campo, se define el nombre de la rama principal del repositorio. En el presente ejemplo, se establece “**main**” como denominación de la rama principal.

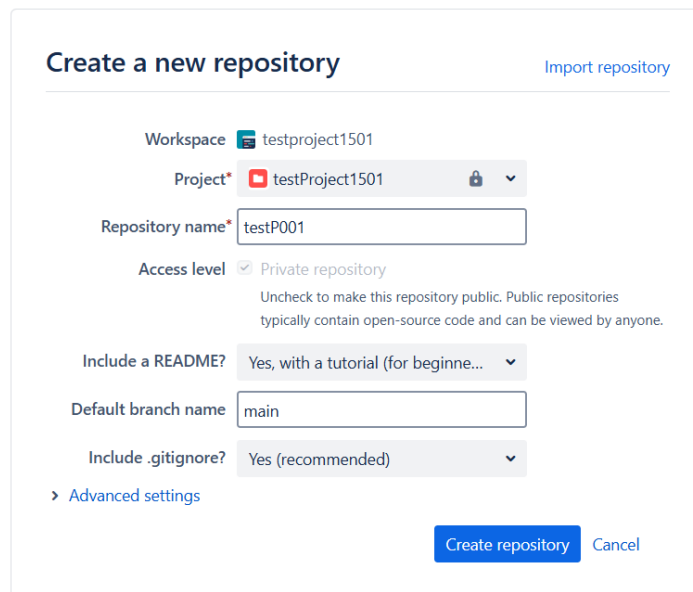
En relación con el archivo .gitignore (opcional), el formulario permite incluirlo con el fin de excluir determinados archivos o directorios del control de versiones. Este archivo puede configurarse mediante plantillas específicas, según el tipo de proyecto y las necesidades del desarrollo.

Finalización del formulario

Una vez completada la configuración correspondiente, se selecciona el botón “**Create Repository**” para finalizar el proceso de creación. Si no se requieren modificaciones adicionales, esta acción redirige al repositorio recién creado.

Figura 8.

Formulario de creación del repositorio



The screenshot shows the 'Create a new repository' form in Bitbucket. At the top right is a link for 'Import repository'. The form fields are: 'Workspace' (testproject1501), 'Project*' (testProject1501 with a lock icon), 'Repository name*' (testP001), 'Access level' (checked 'Private repository' with a note: 'Uncheck to make this repository public. Public repositories typically contain open-source code and can be viewed by anyone.'), 'Include a README?' (Yes, with a tutorial (for beginne...)), 'Default branch name' (main), and 'Include .gitignore?' (Yes (recommended)). There is a link for '> Advanced settings' and two buttons at the bottom right: 'Create repository' and 'Cancel'.

Nota. Elaboración Propia.

Formulario avanzado para la creación de repositorios en Bitbucket

El formulario avanzado presenta opciones adicionales que permiten personalizar el repositorio según las necesidades del proyecto. Si bien estas configuraciones no son obligatorias, pueden resultar útiles dependiendo de los requerimientos específicos del desarrollo. A continuación, se describen las opciones disponibles:

Añadir una descripción al proyecto

En este campo, se puede incorporar una breve descripción que explique el propósito o alcance del proyecto. Este texto permite a los colaboradores comprender el contexto del repositorio.

Ejemplo: “Repositorio para el proyecto de prueba con funciones básicas de Bitbucket”.

Habilitar forks

- **Forks:** corresponden a copias del repositorio que permiten a los usuarios realizar modificaciones o experimentar de manera independiente, sin afectar el repositorio principal.

- Esta funcionalidad puede habilitarse o deshabilitarse según las necesidades del proyecto.

Seleccionar el lenguaje de programación

- El formulario permite elegir el lenguaje principal del proyecto, como Java, Python, JavaScript, entre otros.
- Esta opción no afecta el funcionamiento del repositorio, pero ayuda a categorizarlo para la organización y búsqueda en Bitbucket.

Opcionalidad de estas configuraciones

- Todas estas opciones son opcionales y pueden omitirse si no se consideran necesarias al momento de crear el repositorio.
- El formulario permite proceder directamente a la creación del repositorio sin completar estas configuraciones adicionales.

Figura 9.

Formulario avanzado de creación del repositorio

The image shows the 'Create repository' form in Bitbucket with the 'Advanced settings' section expanded. The form contains the following elements:

- Repository name:** A text input field containing 'testP001'.
- Access level:** A radio button selection with 'Private repository' selected. Below it, a note states: 'Uncheck to make this repository public. Public repositories typically contain open-source code and can be viewed by anyone.'
- Include a README?:** A dropdown menu with the option 'Yes, with a tutorial (for beginne...)' selected.
- Default branch name:** A text input field containing 'main'.
- Include .gitignore?:** A dropdown menu with the option 'Yes (recommended)' selected.
- Advanced settings:** A section header with a downward arrow, indicating it is expanded.
- Description:** A large text area for entering a repository description.
- Forking:** A dropdown menu with the option 'Allow only private forks' selected.
- Language:** A dropdown menu with the option 'Select language...' selected.
- Buttons:** At the bottom right, there are two buttons: 'Create repository' (in blue) and 'Cancel'.

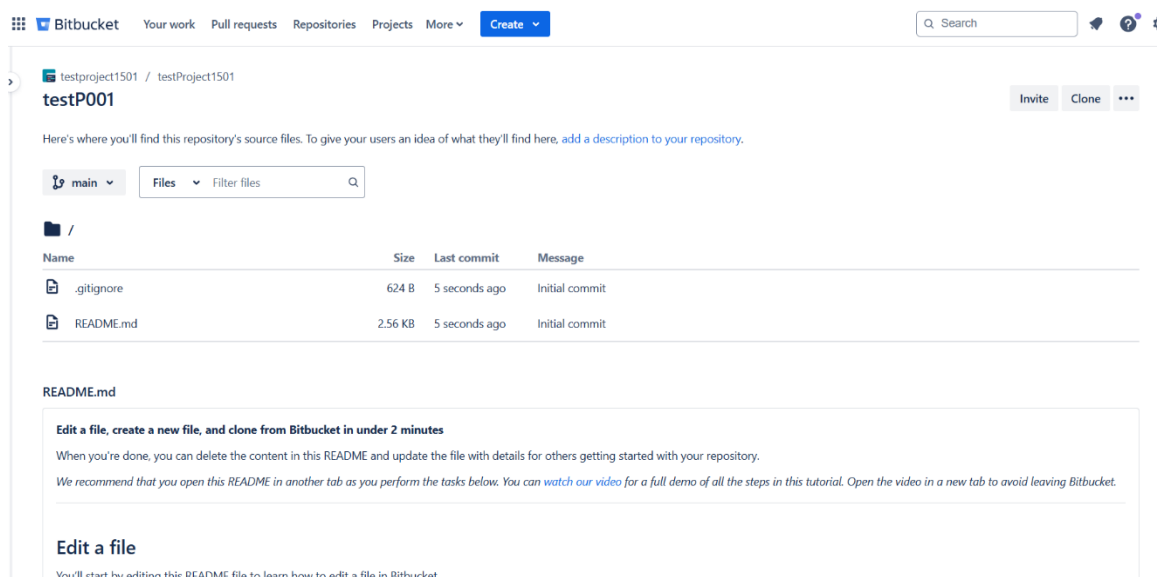
Nota. Elaboración Propia.

Una vez finalizada la creación del repositorio, se visualiza una pantalla como la presentada en la Figura 10, donde se muestran los elementos principales asociados al nuevo repositorio.

- Se presenta el nombre del repositorio recién creado (en este caso, “**testP001**”) y su asociación con el proyecto correspondiente (por ejemplo, “**testProject1501**”).
- En caso de haberse incluido un archivo README, este se muestra dentro del repositorio.
- Desde esta interfaz es posible acceder a las opciones y comandos necesarios para comenzar a trabajar con el repositorio de forma local, como la clonación mediante Git.

Figura 10.

Página de visualización del repositorio



Nota. Elaboración Propia.

Conectar el repositorio local con Bitbucket

Para facilitar la colaboración entre los miembros del equipo Scrum, es necesario vincular el repositorio local con el repositorio remoto alojado en Bitbucket. Para ello, se ejecuta el siguiente comando:

```
git remote add origin https://usuario@Bitbucket.org/usuario/repositorio.git
```

En el comando presentado, los términos **usuario** y **repositorio** corresponden a datos de ejemplo y deben reemplazarse por la información asociada a la cuenta y al proyecto configurados en Bitbucket.

Subir el código al repositorio remoto

Finalmente, los cambios realizados en el repositorio local se envían al repositorio remoto en Bitbucket, con el fin de que estén disponibles para el resto del equipo.

```
git push -u origin main
```

Este comando envía los cambios a la rama principal del repositorio (ya sea **main** o **master**) y establece “**origin**” como el repositorio remoto predeterminado.

Es importante considerar cuál es la rama principal configurada en el repositorio, a fin de ajustar el comando según corresponda.

Asignación de roles de Scrum

Antes de iniciar la implementación de Scrum, es necesario definir las funciones específicas contempladas por este marco de trabajo.

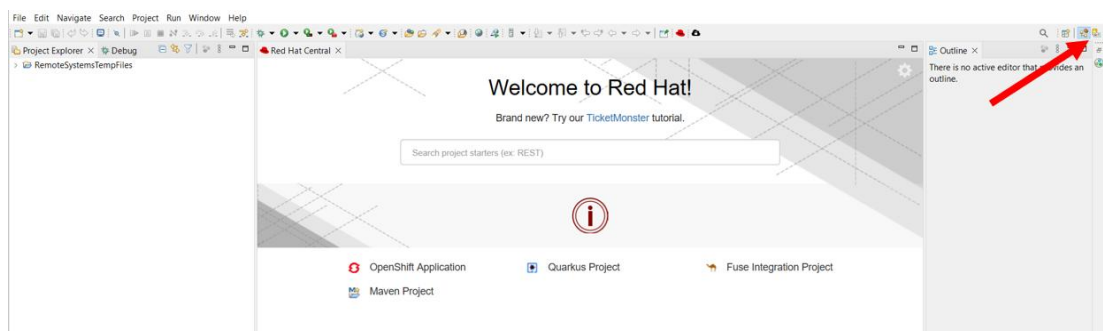
La distribución de responsabilidades se estructura en torno a tres roles fundamentales: el Product Owner (Propietario del Producto), el Scrum Master (Facilitador Scrum) y el Equipo de Desarrollo (Development Team).

2. Subir el repositorio a Bitbucket utilizando Eclipse

En el entorno de Eclipse, se localiza y selecciona la vista correspondiente a Git, ubicada en la esquina superior derecha de la interfaz principal. Esta acción permite acceder a las funcionalidades necesarias para la gestión de repositorios y operaciones de control de versiones directamente desde el entorno de trabajo.

Figura 11.

Página inicial de Eclipse

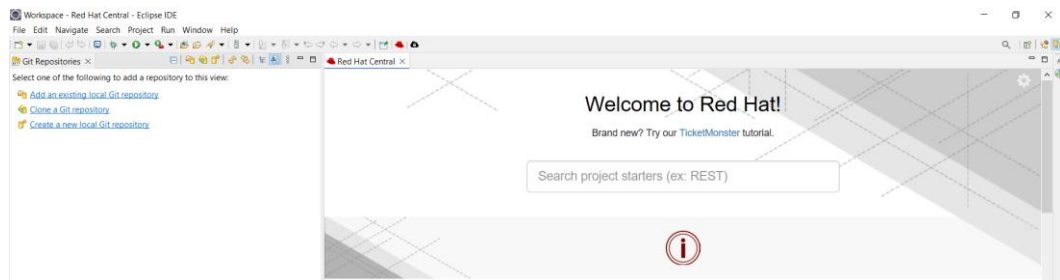


Nota. Elaboración Propia.

Una vez seleccionada la vista de Git en Eclipse, la interfaz presenta los paneles y opciones correspondientes a la gestión de repositorios. A continuación, se selecciona la opción denominada “Clone a Git Repository”, la cual permite iniciar el proceso de clonación de un repositorio remoto hacia el entorno local de desarrollo.

Figura 12.

Sección de Git de Eclipse

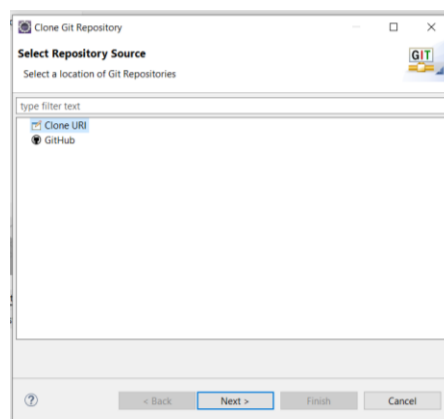


Nota. Elaboración Propia.

En la ventana que se despliega tras seleccionar la opción de clonación, se elige la alternativa denominada “**Clone URI**”, la cual permite establecer la conexión con el repositorio remoto mediante la dirección uniforme de recursos (URI) proporcionada por la plataforma.

Figura 13.

Formulario de selección de repositorio

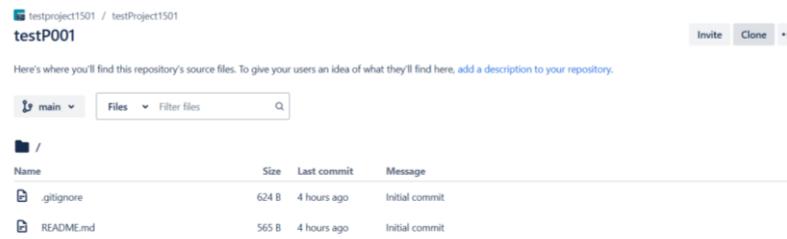


Nota. Elaboración Propia.

Dentro de la plataforma Bitbucket, se accede al repositorio previamente creado y, desde su interfaz principal, se selecciona la opción “**Clone**”. Esta funcionalidad permite obtener la URL necesaria para realizar la clonación del repositorio en el entorno local de desarrollo.

Figura 14.

Repositorio de Bitbucket

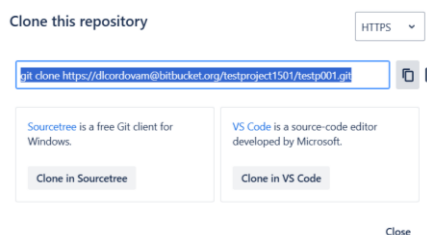


Nota. Elaboración Propia.

A continuación, se copia el vínculo o URL proporcionado por Bitbucket al seleccionar la opción “**Clone**”, el cual contiene la dirección necesaria para establecer la conexión con el repositorio remoto desde el entorno local. Esta URL se utiliza posteriormente en el proceso de clonación mediante Eclipse u otra herramienta compatible con Git.

Figura 15.

Página de clonación del repositorio de Bitbucket

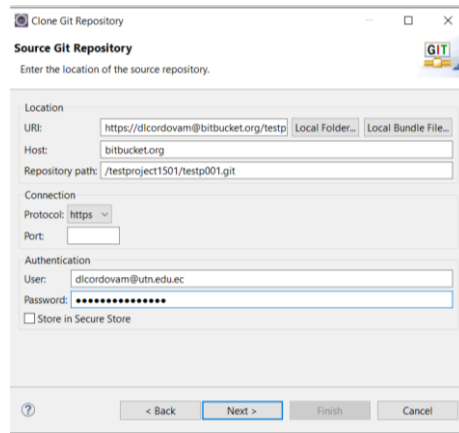


Nota. Elaboración Propia.

En el entorno de desarrollo Eclipse, se completa la información solicitada en el formulario de clonación, incorporando la URL del repositorio previamente copiada desde Bitbucket. En el campo correspondiente al host se especifica **bitbucket.org** y, posteriormente, se ingresan las credenciales de acceso asociadas a la cuenta para establecer la autenticación con el repositorio remoto.

Figura 16.

Formulario de obtención de origen del repositorio en Eclipse

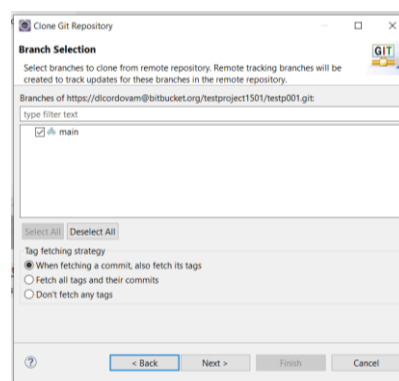


Nota. Elaboración Propia.

En la siguiente etapa del asistente de clonación, se presenta una lista con las ramas disponibles del repositorio remoto; en este caso, se selecciona la rama denominada *main*. Una vez realizada esta selección, se continúa con el proceso haciendo clic en el botón *Next* para avanzar a la configuración del directorio local.

Figura 17.

Selección de la rama del repositorio



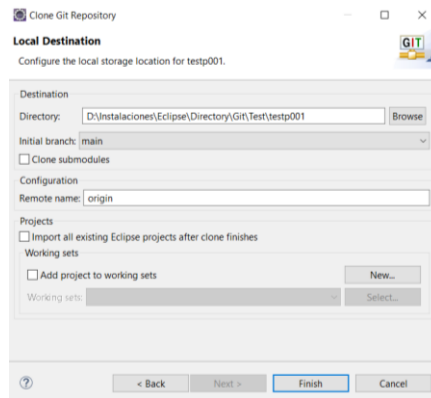
Nota. Elaboración Propia.

Finalmente, tras verificar los parámetros configurados durante el proceso, se procede a seleccionar la opción *Finish*, lo cual inicia la clonación del repositorio alojado en Bitbucket

hacia el entorno local. Esta acción permite disponer del proyecto de forma local para su posterior gestión y desarrollo.

Figura 18.

Selección de la ubicación del proyecto en local

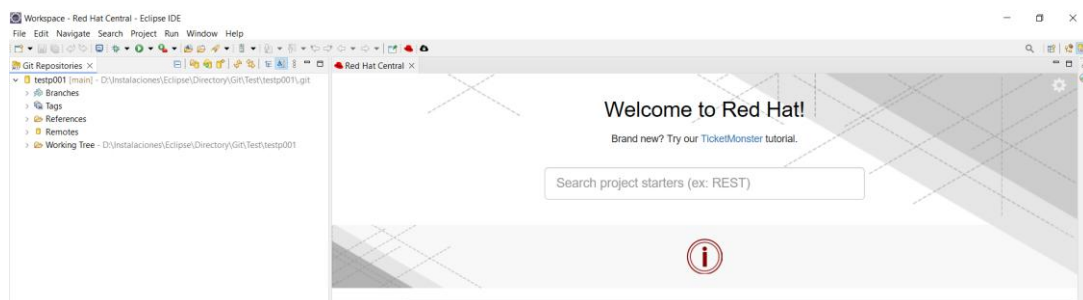


Nota. Elaboración Propia.

En el entorno de desarrollo Eclipse, se localiza y selecciona la vista correspondiente a Git, ubicada en la esquina superior derecha de la interfaz. Esta vista permite acceder de manera integrada a las funcionalidades necesarias para la administración de repositorios y el control de versiones.

Figura 19.

Proyecto cargado en Eclipse

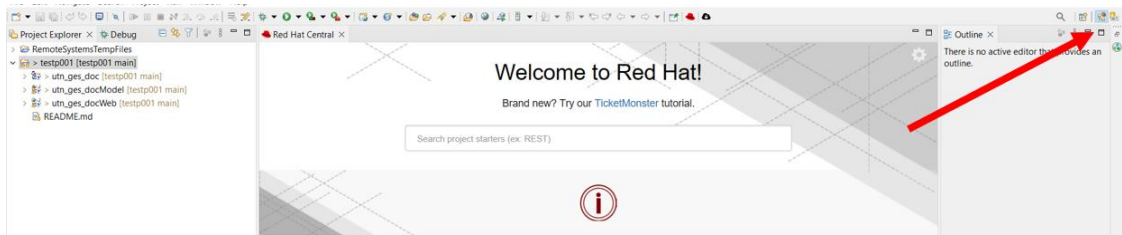


Nota. Elaboración Propia.

Una vez realizados los cambios correspondientes en el proyecto, se regresa al entorno de desarrollo Eclipse y se selecciona la vista de Git. Esta acción permite visualizar y gestionar las modificaciones efectuadas, facilitando su seguimiento y control mediante el sistema de control de versiones.

Figura 20.

Selección de la vista de Git

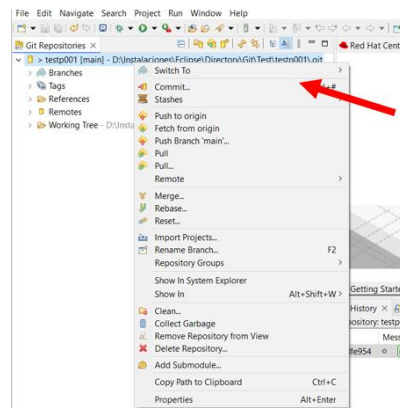


Nota. Elaboración Propia.

Se realiza clic derecho sobre el repositorio alojado en Bitbucket y, dentro del menú contextual que se despliega, se selecciona la opción denominada **“Commit”**, la cual permite registrar los cambios locales efectuados en los archivos del proyecto.

Figura 21.

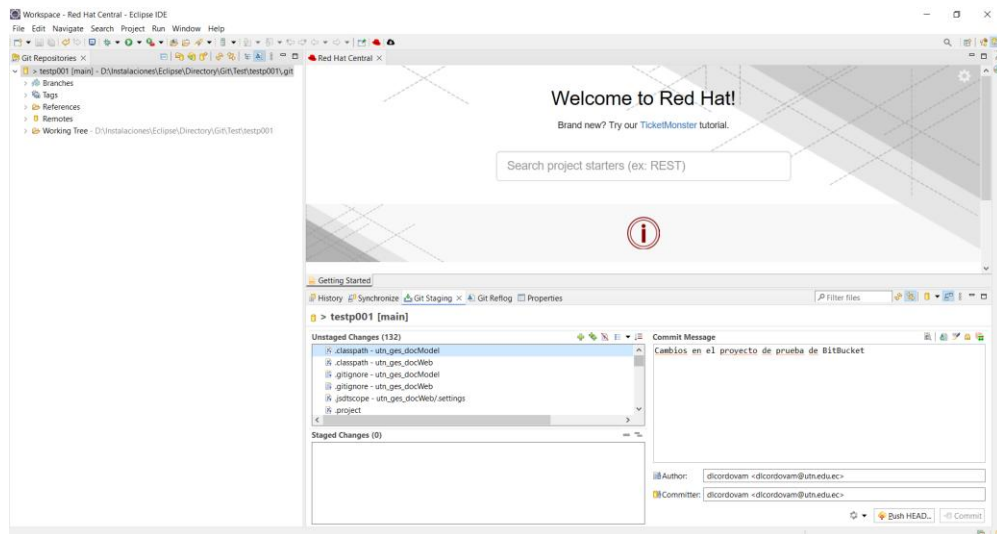
Ubicación de la opción de commit del repositorio



Nota. Elaboración Propia.

En la ventana que se despliega, se introduce un mensaje descriptivo que identifique claramente los cambios realizados, el cual acompañará al commit. Posteriormente, se seleccionan los íconos identificados con el signo “+” para incorporar dichos cambios al commit, asegurando su correcto registro en el historial del repositorio.

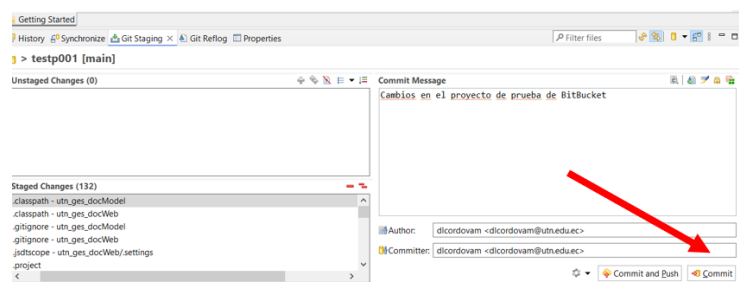
Figura 22.
Formulario de commit del repositorio



Nota. Elaboración Propia.

Una vez añadidos los archivos modificados, estos quedan asociados al *commit* correspondiente. Posteriormente, se selecciona la opción **“Commit”** para registrar formalmente los cambios en el historial local del repositorio, garantizando su trazabilidad y control dentro del proyecto.

Figura 23.
Ubicación del botón para ejecutar el commit en Eclipse

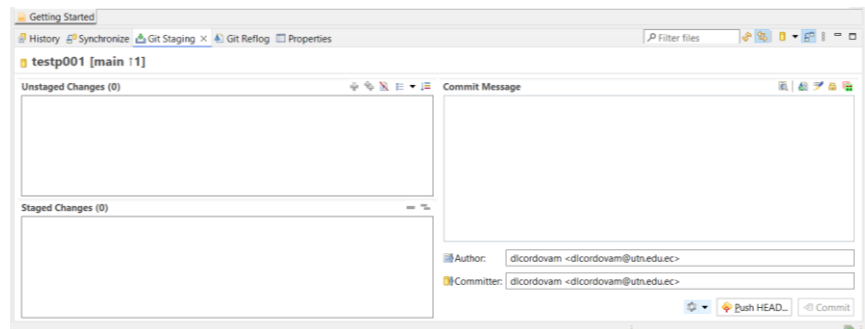


Nota. Elaboración Propia.

Después de completar el commit, se procede a seleccionar la opción **“Push”**, la cual permite enviar los cambios registrados en el repositorio local hacia el repositorio remoto en Bitbucket. Esta acción garantiza que las modificaciones queden disponibles para los demás miembros del equipo y se mantenga la sincronización del proyecto.

Figura 24.

Opción push para el commit previo en Eclipse

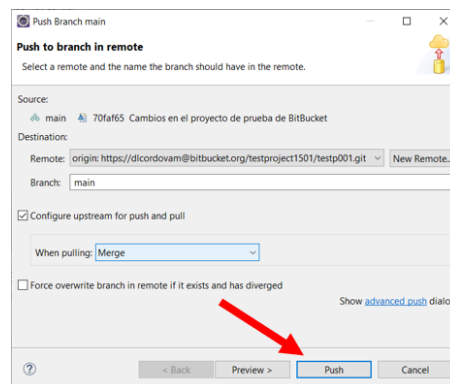


Nota. Elaboración Propia.

Una vez ejecutado el proceso de *push*, se selecciona la opción correspondiente según la operación requerida; en este caso, se elige **“Merge”** para integrar los cambios con la rama remota. Posteriormente, se confirma la operación mediante la selección nuevamente de la opción **“Push”**, lo que permite enviar oficialmente los cambios al repositorio remoto.

Figura 25.

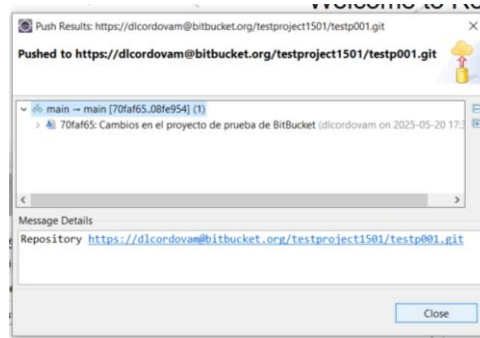
Formulario de configuración para el push en Eclipse



Nota. Elaboración Propia.

Finalmente, una vez completado el envío de los cambios al repositorio remoto, se selecciona la opción **“Close”** para cerrar la ventana correspondiente al proceso de *push*. Con ello, se concluye la operación y se retorna a la vista principal del entorno de desarrollo.

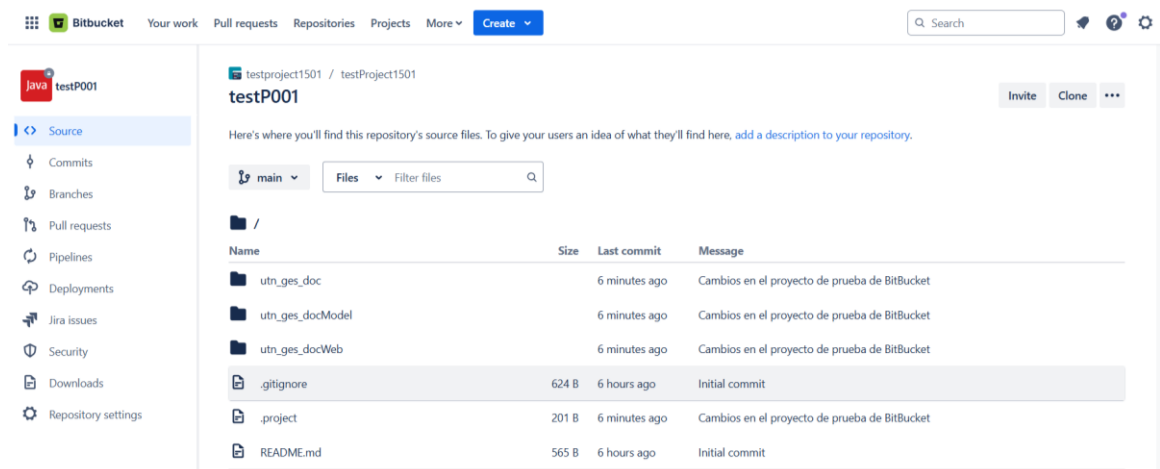
Figura 26.
Push exitoso en Eclipse



Nota. Elaboración Propia.

Para confirmar que el proceso se ha realizado correctamente, se accede al repositorio en Bitbucket y se verifica que los cambios efectuados se encuentren reflejados en la rama correspondiente.

Figura 27.
Repositorio actualizado en Bitbucket



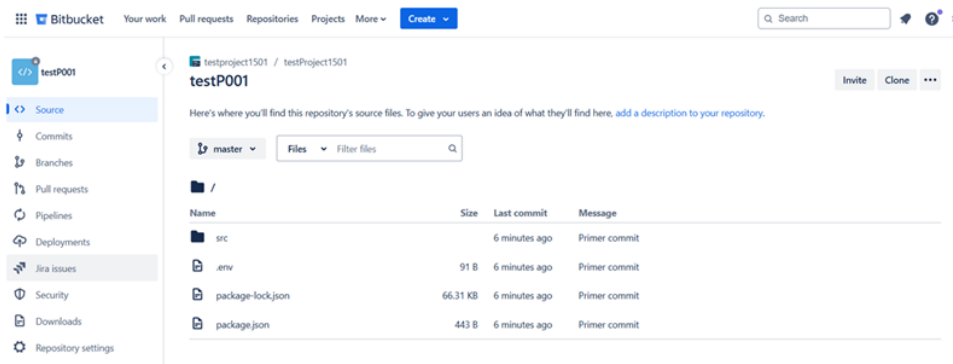
Nota. Elaboración Propia.

3. Conexión con Jira

Para establecer la integración con Jira desde el repositorio en Bitbucket, se despliega el panel lateral izquierdo mediante la selección de la flecha ">". Una vez expandido, se localiza y selecciona la opción "**Jira Issues**". Esta funcionalidad permite vincular los *issues* creados en Jira con el repositorio, facilitando la gestión, trazabilidad y seguimiento de las tareas dentro del proyecto.

Figura
Opción Jira Issues

28.

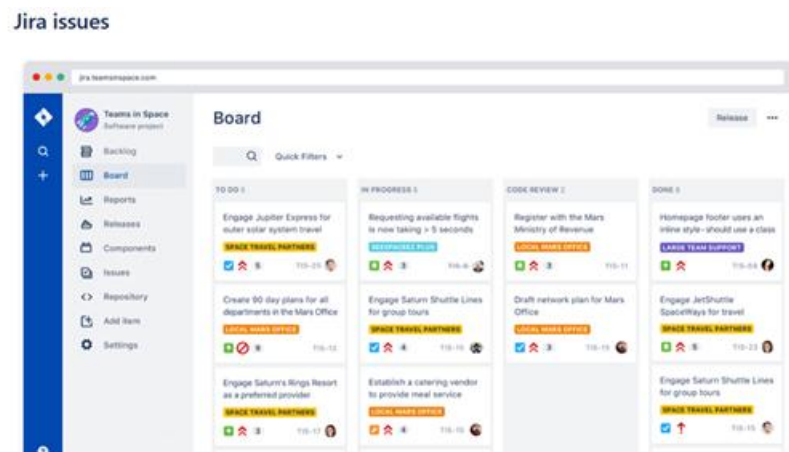


Nota. Elaboración propia

Una vez dentro de la sección “**Jira Issues**”, se visualiza una pantalla que presenta diversas opciones relacionadas con la integración de Jira. En esta etapa, se selecciona la opción “**Try Jira Free**” para habilitar las funcionalidades de Jira vinculadas al repositorio. Este paso permite activar una versión gratuita de la herramienta, facilitando la gestión y seguimiento de las tareas del proyecto.

Figura
Opciones de integración con Jira

29.



Nota. Elaboración propia

El sistema solicita la creación de un nuevo sitio asociado a la plataforma Jira, utilizando una dirección personalizada con la extensión **atlassian.net**. En el presente ejemplo, se configura el sitio con un nombre referencial, el cual formará parte de la URL del entorno de Jira.

Una vez definido el nombre del sitio, se selecciona la opción **“Continue”** para continuar con la configuración y establecer la vinculación del entorno de Jira con el repositorio en Bitbucket.

Figura 30.

Formulario de vinculación de Jira



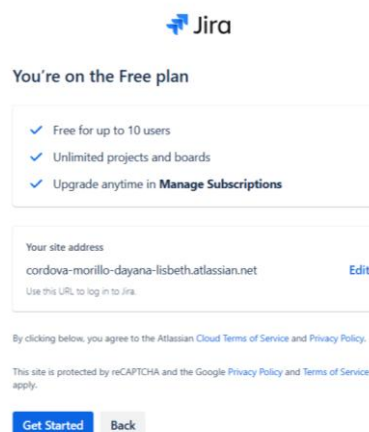
The screenshot shows the Jira logo at the top, followed by the heading 'Add Jira to a site'. Below this is a form titled 'Create a new site'. It contains a text input field with the value 'cordova-morillo-dayana-lisbeth' and a dropdown menu showing '.atlassian.net'. Below the input field is a small note: 'Try something familiar like your company or team name.' At the bottom of the form is a blue button labeled 'Continue'.

Nota. Elaboración Propia.

En esta etapa, se selecciona la opción correspondiente al plan gratuito ofrecido por la plataforma, el cual proporciona las funcionalidades esenciales para la gestión del proyecto. Posteriormente, se selecciona el botón **“Get Started”** para iniciar la configuración del entorno de Jira con las características del plan elegido.

Figura 31.

Selección del plan de Jira que se usará en el repositorio



The screenshot shows the Jira logo at the top, followed by the heading 'You're on the Free plan'. Below this is a box containing three bullet points: 'Free for up to 10 users', 'Unlimited projects and boards', and 'Upgrade anytime in [Manage Subscriptions](#)'. Below this box is another box titled 'Your site address' containing the text 'cordova-morillo-dayana-lisbeth.atlassian.net' and an 'Edit' link. Below this box is a small note: 'Use this URL to log in to Jira.' At the bottom of the page, there is a line of text: 'By clicking below, you agree to the Atlassian Cloud [Terms of Service](#) and [Privacy Policy](#).' Below this is another line of text: 'This site is protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.' At the bottom are two buttons: 'Get Started' (blue) and 'Back' (grey).

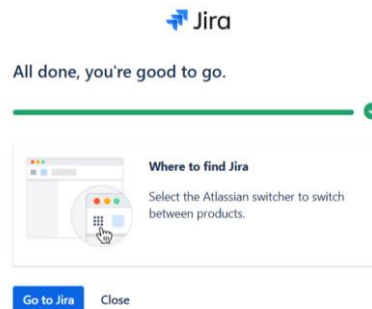
Nota. Elaboración Propia.

El sistema requiere unos minutos para completar la creación del sitio en Jira. Una vez finalizado el proceso, se muestra una notificación que confirma su correcta configuración. En ese momento, se selecciona la opción **“Go to Jira”** para acceder al entorno de gestión del

proyecto. Este paso permite iniciar la configuración y el uso de las herramientas proporcionadas por Jira para la organización y el seguimiento de tareas.

Figura 32.

Confirmación de Jira vinculado al repositorio



Nota. Elaboración Propia.

Entre las plantillas disponibles en Jira, se selecciona la opción “**Scrum**” para configurar el proyecto bajo este marco de trabajo ágil. Posteriormente, se elige “**Siguiente**” para continuar con la configuración inicial, habilitando las funcionalidades necesarias para la planificación y seguimiento del proyecto.

Figura 33.

Plantillas del proyecto de Jira



Nota. Elaboración Propia.

Durante la configuración inicial en Jira, se asigna un nombre al proyecto que mantenga coherencia con el repositorio previamente creado en Bitbucket. En el presente ejemplo, se utiliza **“testP001”**, garantizando consistencia e identificación entre ambas plataformas. Posteriormente, se selecciona la opción **“Comenzar”** para finalizar la configuración y habilitar las herramientas de gestión del proyecto.

Figura
Configuración del nombre del proyecto

34.

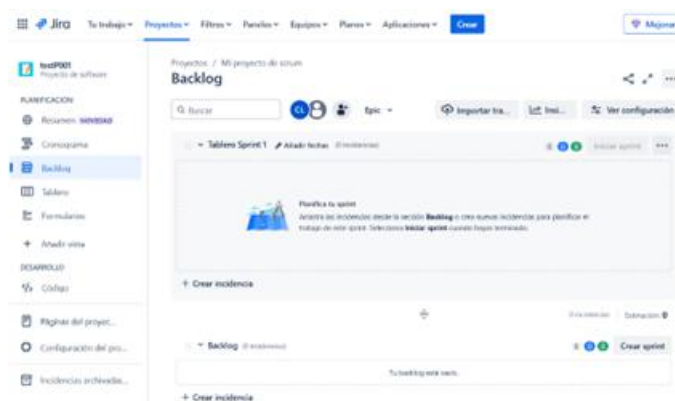


Nota. Elaboración propia

Una vez completada correctamente la configuración, el sistema muestra una pantalla de confirmación. Desde esta interfaz se accede a las principales funcionalidades de Jira y a la gestión de las tareas asociadas al proyecto.

Figura
Configuración completada con éxito

35.



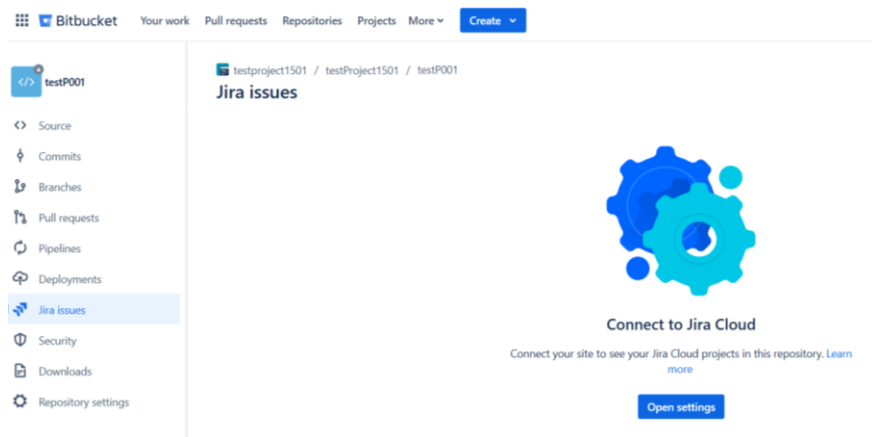
Nota. Elaboración propia

Una vez creado el proyecto en Jira, se retorna al repositorio en Bitbucket. Desde la sección **“Jira Issues”**, se selecciona la opción **“Open Settings”** para acceder a los ajustes

de integración entre el repositorio y el proyecto de Jira. Este procedimiento permite gestionar la vinculación de tareas y optimizar la sincronización del flujo de trabajo.

Figura 36.

Página de visualización de Jira Issues

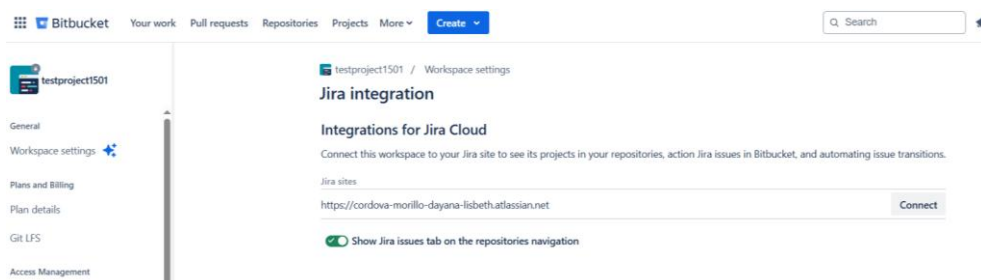


Nota. Elaboración Propia.

Al acceder a la página de configuración, se visualiza el sitio previamente creado en Atlassian. En esta sección, se verifica que el sitio sea el correcto y se selecciona la opción **“Connect”** para vincular el repositorio de Bitbucket con el proyecto en Jira, habilitando la integración entre ambas herramientas.

Figura 37.

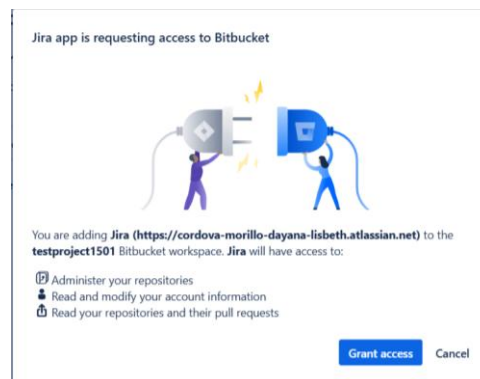
Selección del repositorio en Jira



Nota. Elaboración Propia.

Una vez seleccionada la opción **“Connect”**, se muestra una ventana emergente que solicita la confirmación para otorgar los permisos necesarios. En esta instancia, se selecciona **“Grant Access”** para autorizar la integración entre Bitbucket y Jira, permitiendo el intercambio de información y la habilitación de funcionalidades colaborativas entre ambas plataformas.

Figura 38.
Jira solicitando el acceso al repositorio



Nota. Elaboración Propia.

Posteriormente, se presenta una pantalla de confirmación final para conceder los permisos requeridos. En esta etapa, se selecciona el botón **"Otorgar Acceso"**, completando la integración entre el repositorio en Bitbucket y el proyecto en Jira y garantizando su funcionamiento coordinado.

Figura 39.
Página para aceptar la vinculación de Jira al repositorio

La cuenta de Bitbucket está solicitando acceso a Jira



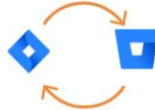
Nota. Elaboración Propia.

Una vez completada la configuración, el sistema muestra una notificación que confirma la correcta conexión de Jira con el espacio de trabajo en Bitbucket. Este mensaje indica que la integración entre ambas plataformas se encuentra activa. Finalmente, se selecciona el botón **"Entendido"** para retornar al entorno principal del repositorio con la integración establecida.

Figura 40.

Página de confirmación de la vinculación entre Jira y el repositorio

Espacio de trabajo de Bitbucket conecta...



El espacio de trabajo testproject1501 está conectado

En relación con los repositorios actuales y futuros, Jira hará lo siguiente:

☒ [Habilitar confirmaciones inteligentes](#) para todos los repositorios

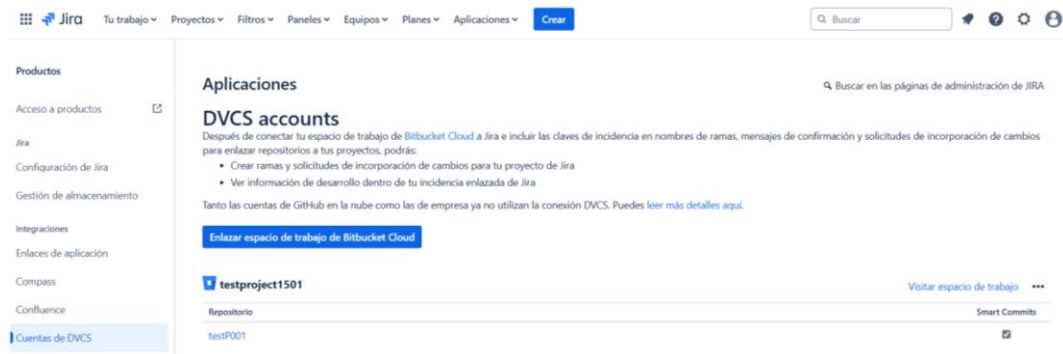
Entendido

Nota. Elaboración Propia.

A continuación, se visualiza el espacio de trabajo en Jira con las herramientas disponibles y la integración correspondiente. En esta pantalla, se selecciona la opción **“Enlazar espacio de trabajo de Bitbucket Cloud”** para completar la conexión entre el repositorio de Bitbucket y el entorno de Jira.

Figura 41.

Opción para enlazar el espacio con Bitbucket Cloud

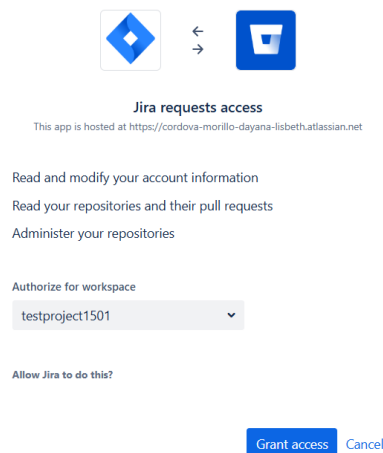


Nota. Elaboración Propia.

En esta etapa, Jira solicita los permisos necesarios para acceder al espacio de trabajo en Bitbucket. Se muestra una ventana emergente en la que se selecciona el botón **“Grant Access”** para autorizar la conexión. Este procedimiento garantiza que Jira cuente con los permisos adecuados para interactuar con el espacio de trabajo y los repositorios asociados.

Figura 42.

Jira solicitando acceder al espacio de trabajo

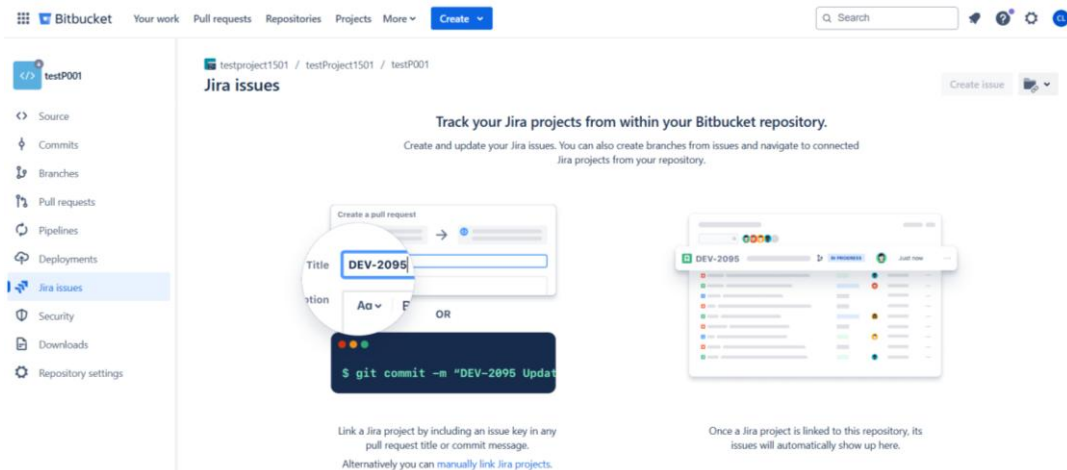


Nota. Elaboración Propia.

En la plataforma de Bitbucket, se accede a la sección **“Jira Issues”**. Desde allí, se selecciona la opción **“Manually link projects”** para establecer de forma manual la vinculación entre el repositorio de Bitbucket y el proyecto previamente configurado en Jira.

Figura 43.

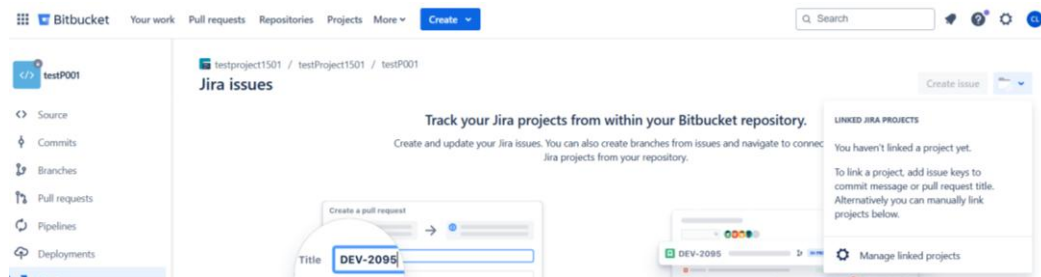
Visualización de Jira Issues



Nota. Elaboración Propia.

En la esquina superior derecha de la pantalla, se despliega un cuadro de diálogo en el que se selecciona la opción **“Manage linked Projects”**. Esta acción permite gestionar los proyectos vinculados, facilitando la sincronización entre el repositorio de Bitbucket y el proyecto correspondiente en Jira.

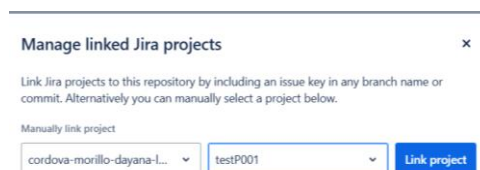
Figura 44.
Sección Manage linked Projects



Nota. Elaboración Propia.

A continuación, se seleccionan el sitio de Atlassian y el proyecto previamente creado. Una vez identificados ambos elementos, se elige la opción **“Link Project”** para establecer la vinculación entre el repositorio de Bitbucket y el proyecto en Jira.

Figura 45.
Gestión de los proyectos de Jira

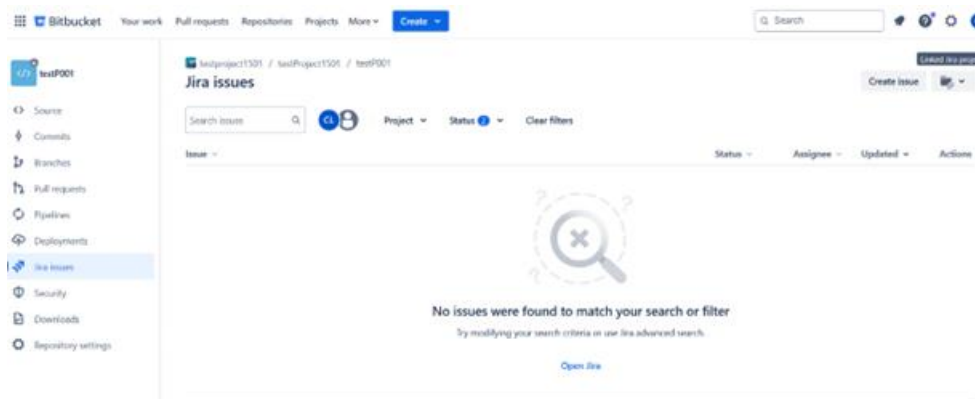


Nota. Elaboración Propia.

Una vez completados los pasos anteriores, el proyecto de Jira queda vinculado con el repositorio de Bitbucket. Esta integración permite sincronizar ambos entornos, facilitando la gestión centralizada de tareas, el seguimiento del progreso y la colaboración dentro del equipo.

Figura 46.

Proyecto de Jira vinculado con Bitbucket



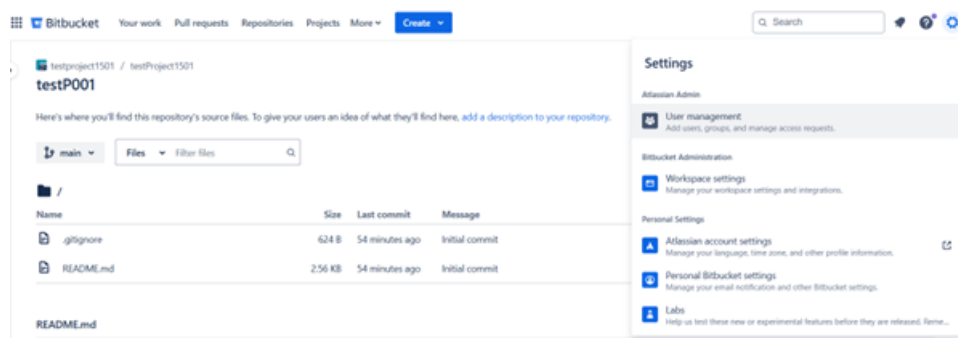
Nota. Elaboración propia

4. Añadir usuarios al proyecto

Para incorporar nuevos usuarios al proyecto, se accede al repositorio correspondiente en Bitbucket. En la esquina superior derecha de la interfaz, se selecciona el ícono de engranaje para ingresar a la configuración del repositorio. Posteriormente, en el menú desplegable, se elige la opción **“User Management”**, desde donde se gestionan los permisos y roles de los usuarios que participarán en el proyecto.

Figura 47.

Gestionar los permisos y roles de usuarios



Nota. Elaboración propia

En la pantalla correspondiente, se encuentra disponible la opción para añadir miembros al equipo del proyecto. Para ello, se introduce el correo electrónico del usuario en el campo destinado a la invitación y se selecciona la opción **“Añadir a miembros del**

equipo”, enviando la solicitud y otorgando los permisos correspondientes para su participación en el proyecto.

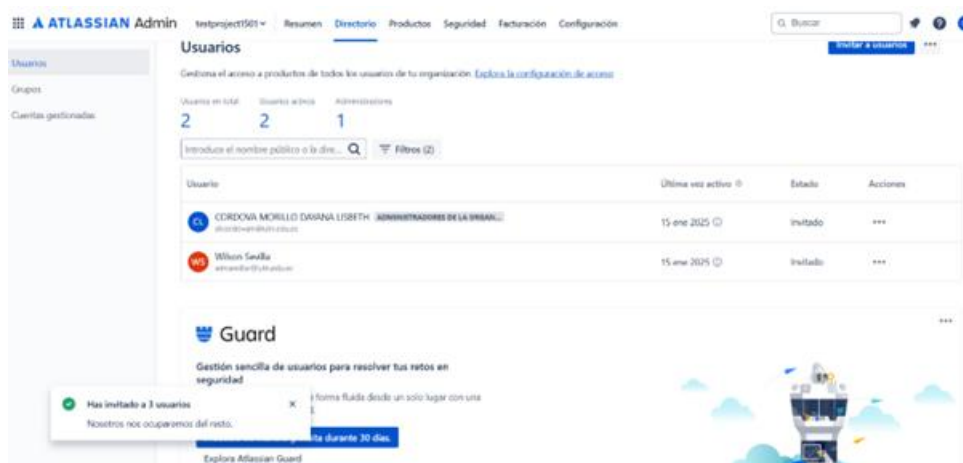
Figura 48.
Añadir miembros a equipo



Nota. Elaboración propia

En el sitio de Atlassian, dentro de la sección correspondiente al espacio de trabajo (*Workspace*), se visualiza un listado de los usuarios que forman parte del equipo. Esta lista incluye información básica, como el nombre y el correo electrónico asociado, facilitando la gestión y supervisión de los colaboradores activos en el proyecto.

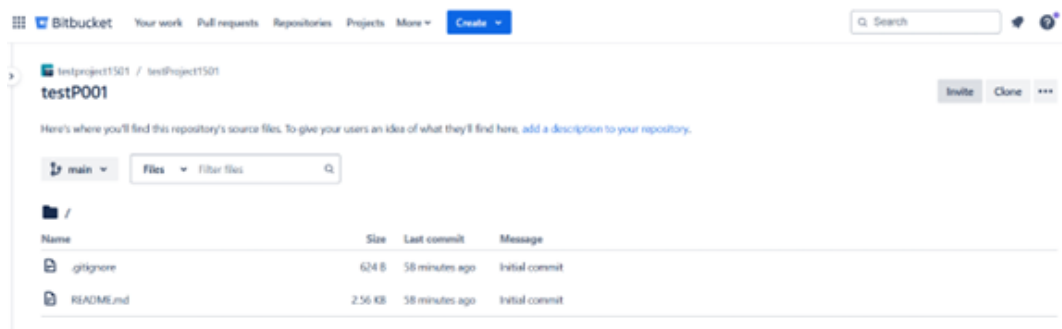
Figura 49.
Colaboradores de equipo



Nota. Elaboración propia

En la plataforma Bitbucket, se accede al repositorio en el que se desea añadir nuevos colaboradores. Una vez dentro, se selecciona el botón **“Invite”**, generalmente ubicado en una posición destacada de la interfaz. Esta acción despliega un formulario desde el cual se gestiona la incorporación de usuarios al equipo de trabajo asociado al repositorio.

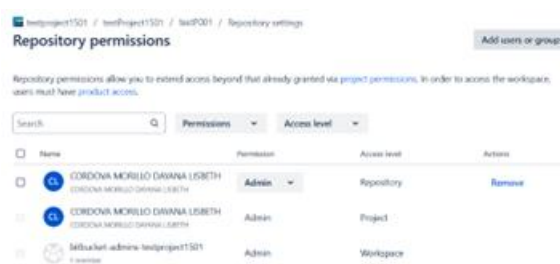
Figura 50.
Invitar a nuevos usuarios



Nota. Elaboración propia

Al seleccionar el botón **“Invite”** en el repositorio de Bitbucket, se despliega una pantalla con opciones relacionadas con la gestión de usuarios. En esta sección, se elige la opción **“Add users or groups”**, que permite incorporar nuevos colaboradores o equipos de trabajo al repositorio, asignándoles los permisos correspondientes para su participación en el proyecto.

Figura 51.
Añadir usuarios o grupos



Nota. Elaboración propia

Una vez accedido a la opción **“Add users or groups”**, se selecciona el usuario previamente invitado al repositorio. En el campo correspondiente a los permisos, se asigna el nivel de acceso según las funciones que desempeñará en el proyecto. En el presente ejemplo, se elige la opción **“Write”**, otorgando permisos de escritura que permiten realizar contribuciones al código del repositorio. Finalmente, se confirma la configuración para aplicar los cambios.

Figura 52.
Formulario para añadir usuarios

Nota. Elaboración Propia.

Una vez completados los pasos anteriores, la interfaz muestra la confirmación de que el usuario ha sido incorporado al repositorio. En esta vista se visualizan el nombre del usuario, los permisos asignados (en este caso, **“Write”**) y otros detalles relacionados con su participación en el proyecto. Esta confirmación indica que el usuario cuenta con los accesos necesarios para colaborar en el desarrollo del repositorio.

Figura 53.
Usuarios añadidos a Jira

Name	Permission	Access level	Actions
CORDOVA MORILLO DAYANA LISBETH	Admin	Repository	Remove
CORDOVA MORILLO DAYANA LISBETH	Admin	Project	
Wilson Sevilla	Write	Repository	Remove
bitbucket-admins-testproject1501	Admin	Workspace	

Access granted

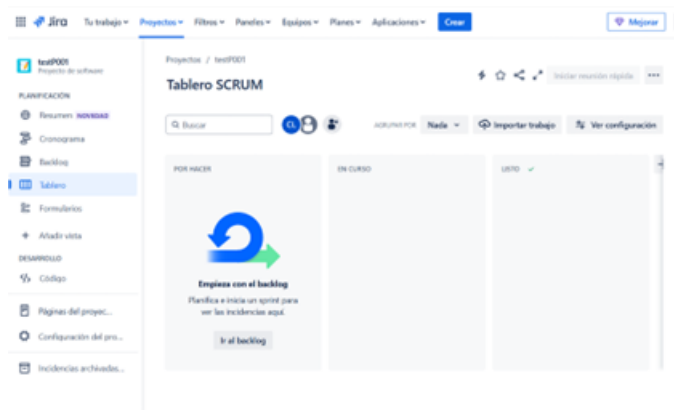
"Wilson Sevilla" was successfully added.

Nota. Elaboración Propia.

Una vez incorporado el usuario al repositorio en Bitbucket, resulta necesario integrarlo también en el proyecto correspondiente en Jira, a fin de que pueda participar en la gestión de tareas y actividades.

Para ello, se accede al proyecto en Jira y, desde el menú principal, se ingresa a la sección **“Tablero”**. Posteriormente, se selecciona la opción **“Ir al Backlog”** para continuar con la configuración de la participación del usuario en el proyecto.

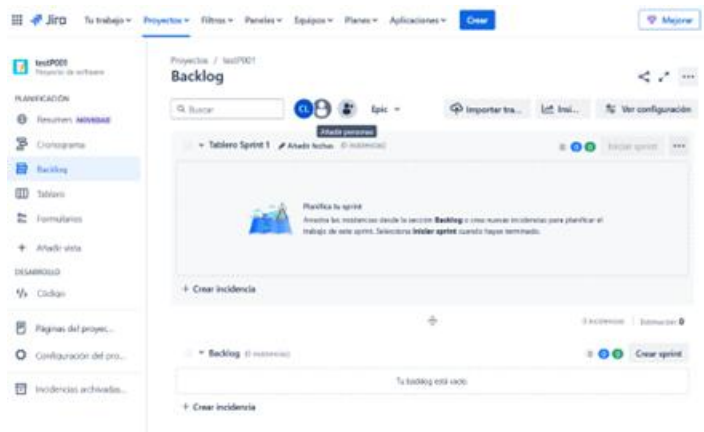
Figura 54.
Sección tablero de Jira



Nota. Elaboración propia

Para incorporar al usuario invitado en el proyecto de Jira, se accede al **Backlog** correspondiente. En la parte superior de la pantalla, se selecciona el ícono ubicado junto al área del perfil y, en el menú desplegable, se elige la opción **“Añadir Usuarios”** para gestionar la incorporación al proyecto.

Figura 55.
Añadir usuario invitado al proyecto

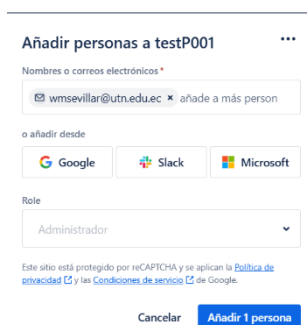


Nota. Elaboración propia

Una vez seleccionada la opción **“Añadir Usuarios”**, se abre una ventana en la que se ingresa la dirección de correo electrónico del usuario invitado. Posteriormente, se selecciona el botón **“Añadir persona”** para completar la invitación, permitiendo que el usuario se

incorpore al proyecto en Jira con los permisos correspondientes para colaborar en las tareas del equipo.

Figura 56.
Ingreso de correo de colaborador

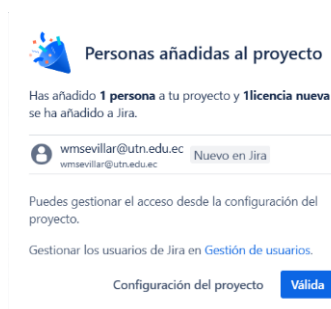


The screenshot shows the 'Añadir personas a testP001' interface in Jira. It features a search bar for email addresses with the text 'Nombres o correos electrónicos *' and a dropdown menu showing 'wmsevilla@utn.edu.ec'. Below this, there are buttons for 'o añadir desde' with logos for Google, Slack, and Microsoft. A 'Role' dropdown menu is set to 'Administrador'. At the bottom, there are 'Cancelar' and 'Añadir 1 persona' buttons. A small disclaimer at the bottom states: 'Este sitio está protegido por reCAPTCHA y se aplican la [Política de privacidad](#) y las [Condiciones de servicio](#) de Google.'

Nota. Elaboración propia

Una vez añadido el usuario, se muestra una pestaña que confirma su incorporación al proyecto. En esta sección, se selecciona la opción **“Valida”** para concluir el proceso de integración y completar la configuración de acceso para los nuevos miembros en Jira.

Figura 57.
Usuario añadido correctamente



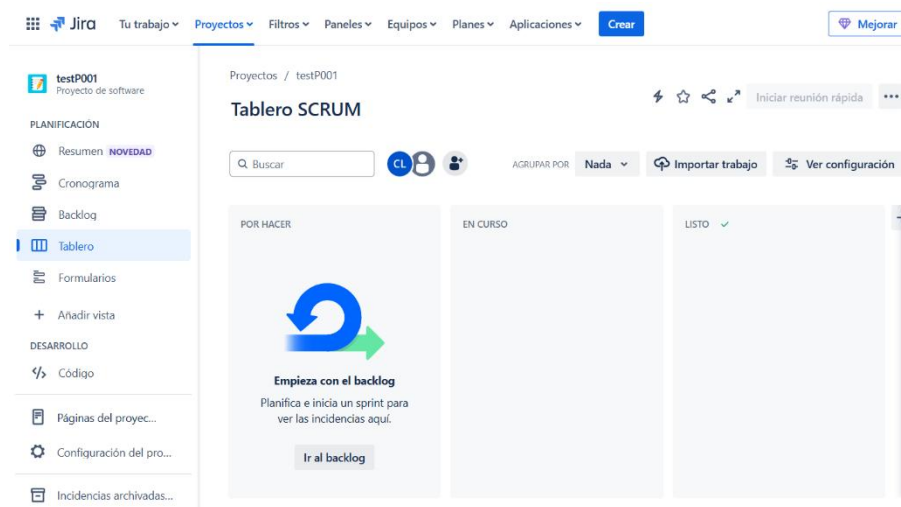
The screenshot shows the 'Personas añadidas al proyecto' confirmation screen. It displays a message: 'Has añadido 1 persona a tu proyecto y 1 licencia nueva se ha añadido a Jira.' Below this, there is a list of added users with the email 'wmsevilla@utn.edu.ec' and the status 'Nuevo en Jira'. At the bottom, there is a link to 'Gestionar los usuarios de Jira en Gestión de usuarios.' and a 'Configuración del proyecto' button with a 'Valida' button next to it.

Nota. Elaboración propia

5. Creación del Sprint Backlog

Para iniciar la creación del **Sprint Backlog**, se accede al proyecto en Jira y, desde la barra de navegación, se ingresa a la sección **“Tablero”**. Posteriormente, se selecciona la opción **“Ir al Backlog”** para visualizar y gestionar los elementos pendientes del proyecto. Este paso permite organizar las tareas que serán consideradas en el siguiente Sprint.

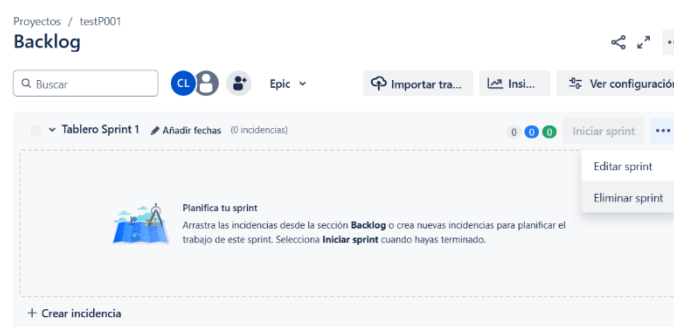
Figura 58.
Tablero Scrum para crear el Sprint Backlog



Nota. Elaboración propia

En el tablero del proyecto en Jira, se identifica el **Sprint 1** que aparece de manera predeterminada. Desde el menú de opciones representado por los tres puntos ubicados en la esquina superior derecha del Sprint, se selecciona la opción “**Eliminar Sprint**” para proceder con su eliminación. Este paso permite reorganizar la planificación o iniciar un nuevo Sprint sin elementos preconfigurados.

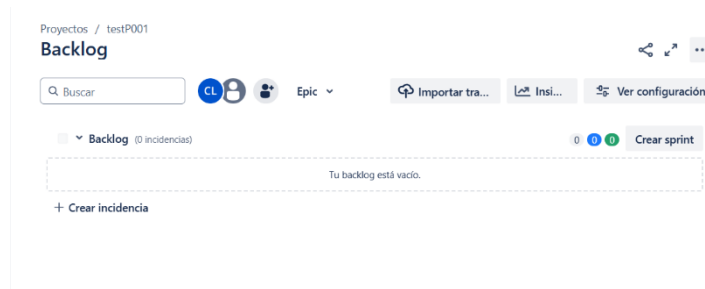
Figura 59.
Eliminar Sprint predeterminado



Nota. Elaboración propia

Para iniciar la planificación de los Sprints del proyecto, se selecciona el botón “**Crear Sprint**”. Esta acción agrega un nuevo Sprint al tablero, en el cual se definen las tareas y objetivos específicos que formarán parte del ciclo de trabajo correspondiente.

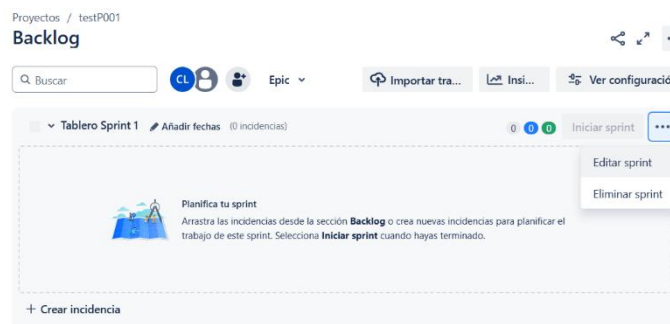
Figura 60.
Agregar nuevo Sprint al tablero



Nota. Elaboración propia

Una vez creado el nuevo Sprint, este se visualiza en la sección correspondiente del proyecto. Para ajustar sus detalles, se accede al menú representado por los tres puntos ubicados en la esquina superior derecha y se selecciona la opción **“Editar Sprint”**. Esta acción permite modificar el nombre, la duración y otros parámetros relevantes conforme a las necesidades del proyecto.

Figura 61.
Opción para editar el Sprint



Nota. Elaboración propia

En el tablero del Sprint, se accede a la opción de edición para personalizar sus detalles. En esta sección se pueden realizar los siguientes ajustes:

- **Nombre del Sprint:** se define una denominación que identifique claramente el propósito o ciclo de trabajo.
- **Duración del Sprint:** se establece el número de días que abarcará la iteración.
- **Meta del Sprint:** se especifica un objetivo claro que oriente el trabajo del equipo durante el período definido.

La fecha de finalización se calcula automáticamente en función de la duración seleccionada. Una vez configurados estos parámetros, se selecciona la opción “**Actualizar**” para guardar los cambios.

Figura 62.

Formulario de edición de tablero de Sprint

Editar sprint: Tablero Sprint 1

Los campos obligatorios están marcados con un asterisco *

Nombre del sprint *
Sprint 001

Duración
3 semanas

Fecha de inicio
por ejemplo: 31/12/2018 por ejemplo: 13:00

Fecha de finalización
por ejemplo: 14/01/2019 por ejemplo: 13:00

Meta del sprint
Completar las seguridades del proyecto

Cancelar Actualizar

Nota. Elaboración Propia.

Una vez configurado el Sprint, se procede a añadir las incidencias correspondientes. Para ello, se selecciona el botón “**Crear Incidencia**”, ubicado en la esquina inferior izquierda del tablero del Sprint. Esta acción permite registrar los detalles específicos de cada incidencia, como su descripción, prioridad y otra información relevante para su seguimiento y resolución dentro del marco del Sprint.

Figura 63.

Crear una nueva incidencia

Jira Tu trabajo Proyectos Filtros Paneles Equipos Planes Aplicaciones Crear Mejorar Q

Proyectos / testP001

Backlog

Buscar CL Epic Importar tra... Ins... Ver configuración

Sprint 001 Añadir fechas 0 incidencias Iniciar sprint ...

Completar las seguridades del proyecto

Planifica tu sprint
Arrastra las incidencias desde la sección Backlog o crea nuevas incidencias para planificar el trabajo de este sprint. Selecciona Iniciar sprint cuando hayas terminado.

+ Crear incidencia

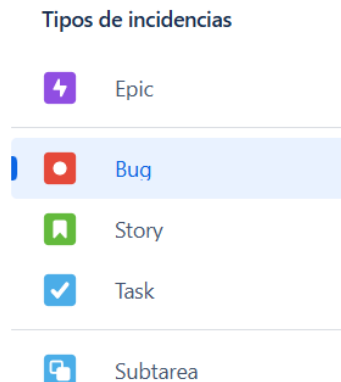
0 Incidencias Estimación: 0

Nota. Elaboración propia

En esta sección, se define el tipo de incidencia que se desea crear, seleccionando entre las opciones disponibles, como **Tarea**, **Bug** o **Historia**. Posteriormente, se ingresa la descripción correspondiente a la actividad a realizar. Una vez completada la información, se confirma la creación de la incidencia mediante la tecla “**Enter**”, tras lo cual esta se incorpora automáticamente al tablero del Sprint para su gestión y seguimiento dentro del proyecto.

Figura 64.

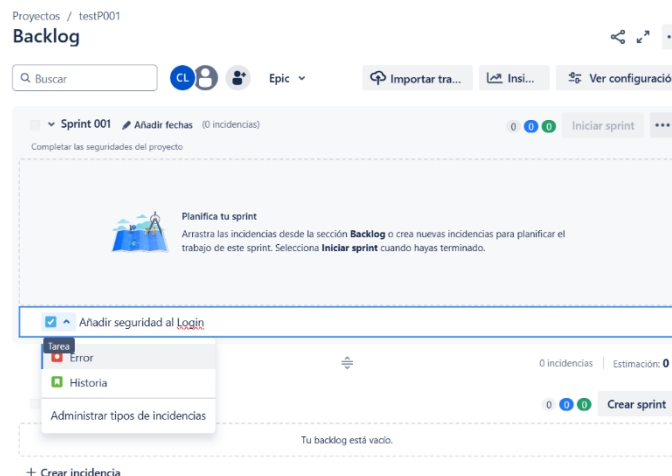
Selección de tipo de incidencia



Nota. Elaboración Propia.

Figura 65.

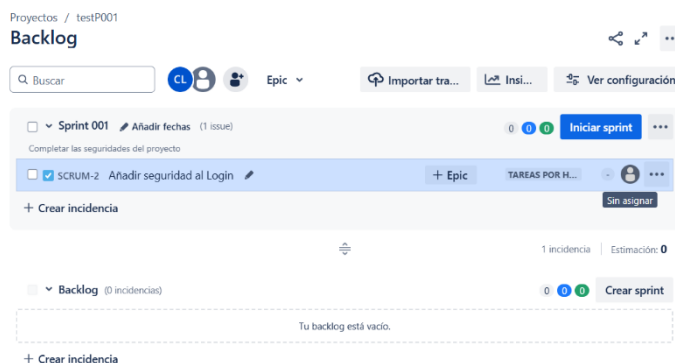
Descripción de una nueva incidencia



Nota. Elaboración propia

Una vez creada la incidencia, se asigna al usuario responsable de su ejecución. Para ello, en el detalle de la incidencia se selecciona la opción “**Sin asignar**”, desde donde se define el miembro del equipo encargado de su desarrollo.

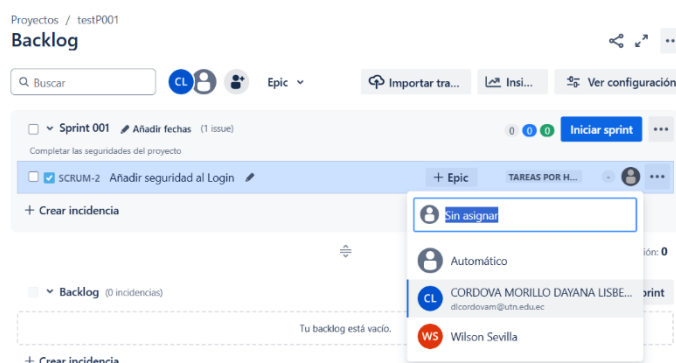
Figura 66.
Designar usuario a incidencia



Nota. Elaboración propia

En el listado desplegable, se selecciona el usuario responsable de ejecutar la actividad correspondiente. Esta asignación garantiza que cada incidencia cuente con un encargado específico, facilitando el seguimiento y la gestión eficiente de las tareas dentro del Sprint.

Figura 67.
Encargado de la nueva incidencia



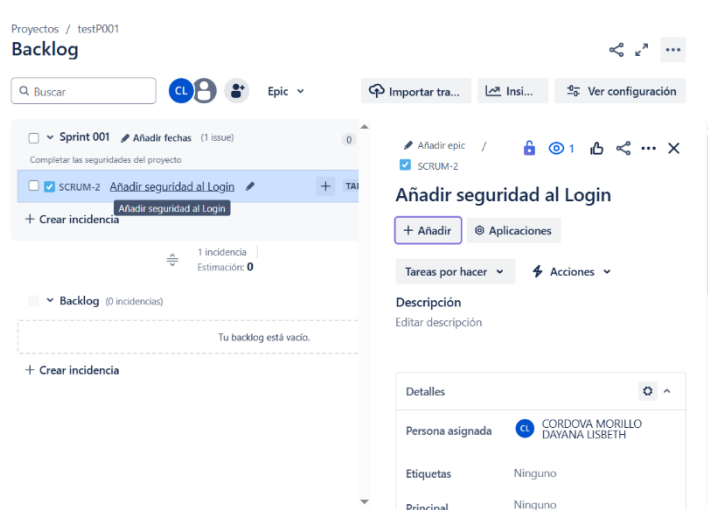
Nota. Elaboración propia

Para agregar información detallada a la tarea, se accede al título correspondiente dentro del Sprint, lo que despliega una ventana con los detalles asociados.

En dicha ventana, se selecciona la sección “**Editar Descripción**” y se incorpora el contenido necesario para especificar el objetivo o alcance de la tarea. Esta descripción puede incluir instrucciones, criterios de aceptación u otros elementos relevantes para su correcta ejecución.

Finalmente, se confirma la edición para que la información quede registrada y asociada a la tarea dentro del Sprint.

Figura 68.
Edición de la descripción de una tarea

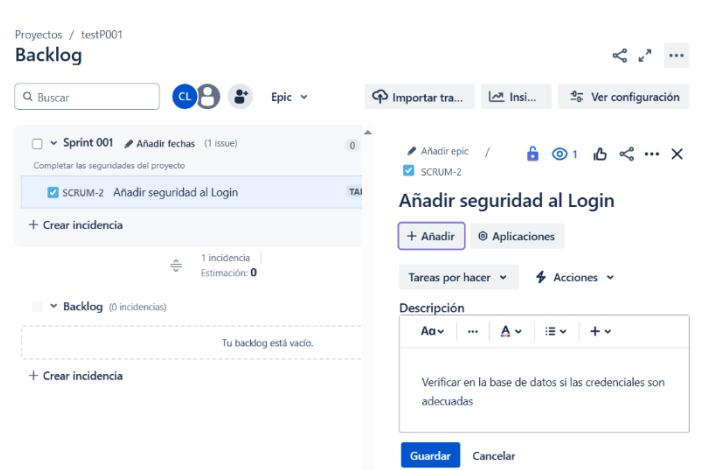


Nota. Elaboración propia

Para añadir y guardar la descripción de una tarea, se accede a la ventana correspondiente y se localiza la sección **“Descripción”**. En este apartado, se incorpora la información detallada relacionada con la actividad, incluyendo el objetivo, las expectativas y cualquier otro elemento relevante para su ejecución.

Una vez completada la información, se selecciona la opción **“Guardar”** para registrar los cambios realizados

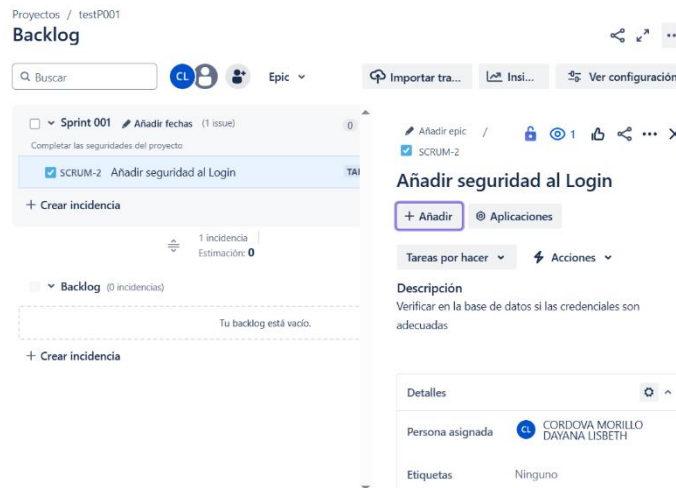
Figura 69.
Añadir y guardar la descripción de una tarea



Nota. Elaboración propia

Una vez completado el proceso, la tarea se visualiza en el tablero con la descripción incorporada, el usuario asignado y los detalles actualizados. De esta manera, el Sprint refleja el estado y la información relevante para el seguimiento y ejecución de la actividad.

Figura 70.
Visualización de una tarea en el Sprint



Nota. Elaboración propia

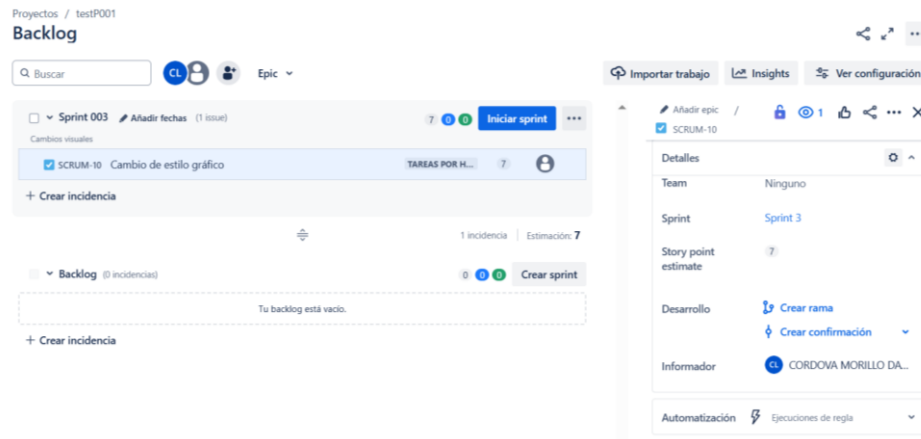
Para asignar un nivel de prioridad o peso a un *Issue* en Jira, se accede a la ventana de detalles correspondiente mediante la selección del título desde el tablero o el *Backlog* del proyecto.

En la parte inferior de la interfaz se localiza el campo “**Estimación de puntos de historia**”, donde se define el valor asociado al esfuerzo, complejidad o relevancia de la actividad. Este valor permite estimar la carga de trabajo y establecer la priorización del *Issue* dentro del Sprint o del *Backlog* general.

Una vez asignado el puntaje correspondiente, se guardan los cambios para registrar la actualización en el sistema.

Figura 71.

Modificación de la Issue en el backlog



Nota. Elaboración Propia.

2.1.6.2. Fase 2. Juego

6. Creación del Product Backlog

Para construir el **Product Backlog** en Jira, es necesario seguir un proceso similar al descrito en la sección 5 sobre la **Creación del Sprint Backlog** , pero adaptado para abarcar todos los elementos necesarios del proyecto. A continuación, se detallan los pasos:

Repetir el proceso de creación de Sprints:

- Siga los pasos indicados en la sección correspondiente para crear nuevos Sprints.
- Para cada Sprint, se agregan los elementos o tareas correspondientes, asegurándose de que reflejan las necesidades y prioridades del proyecto.

Completar los detalles de cada tarea:

- Al crear cada tarea, se define su descripción, nivel de prioridad y responsable asignado.
- Se incluye estimaciones de esfuerzo, cuentos como los puntos de *historia* (Story points), para facilitar la planificación del trabajo.

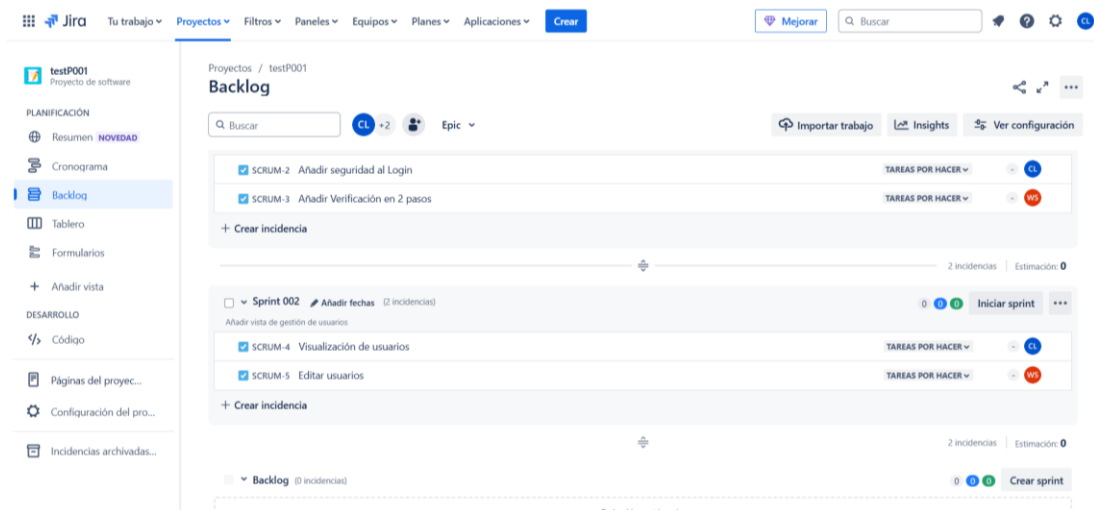
Incluir todos los Sprints requeridos:

El proceso se desarrolla hasta que el **Product Backlog** incluya todos los elementos necesarios para la ejecución completa del proyecto.

Posteriormente, cada Sprint debe configurarse adecuadamente, incorporando las tareas priorizadas y alineadas con los objetivos establecidos.

Este procedimiento garantiza que el **Product Backlog** esté completamente configurado, proporcionando una visión clara y detallada de todas las actividades y objetivos del proyecto dentro del marco Scrum.

Figura 72.
Product backlog de Jira



Nota. Elaboración Propia.

7. Sprint planning

Para llevar a cabo la planificación de los Sprints, se deben seleccionar y organizar los elementos previamente definidos en el Product Backlog. Estos elementos, registrados como tareas o historias, servirán como base para estructurar el trabajo que se desarrollará en cada iteración. A continuación, se describen los pasos para configurar dichas tareas:

Creación de la tarea de planificación:

- Acceda a la sección de Issues en Jira y cree una nueva tarea etiquetada como "Sprint Planning". Esta etiqueta permitirá identificar claramente el propósito de la tarea dentro del proyecto.

- Asigne un nombre descriptivo que relacione la tarea con el Sprint correspondiente, por ejemplo, "Sprint Planning - Sprint 1".

Adjuntar documentos útiles:

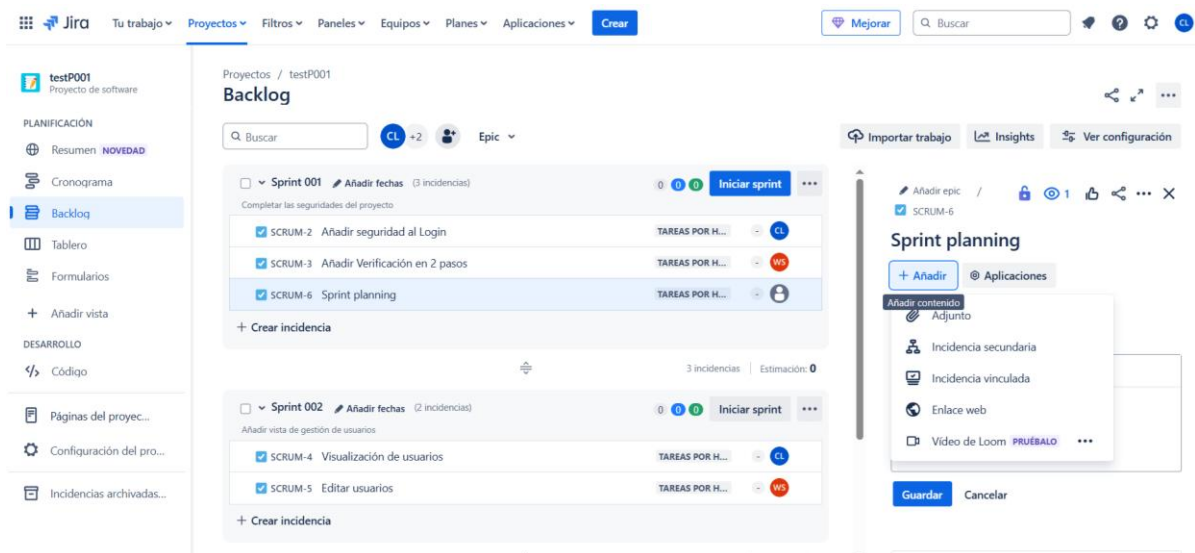
- Dentro de la tarea recién creada, localice el botón **"Añadir"** y haga clic en él para adjuntar cualquier documento relevante que facilite la planificación del Sprint.
- Los documentos pueden incluir especificaciones, diagramas, historias de usuario, o cualquier recurso necesario para clarificar y organizar las actividades del Sprint.

Repetir para todos los Sprints:

Este procedimiento se replica para cada Sprint planificado a partir del Product Backlog, garantizando que cada iteración cuente con una tarea de planificación específica y los recursos necesarios para su adecuada ejecución.

Figura 73.

Creación del Sprint planning



Nota. Elaboración Propia.

8. Inicio de las Iteraciones del Sprint

Para iniciar las iteraciones correspondientes a un Sprint en Jira, se ejecutan las acciones necesarias desde el tablero del proyecto, permitiendo activar el ciclo de trabajo definido y dar comienzo al desarrollo de las tareas planificadas.

Acceso al proyecto en Jira:

Para gestionar las iteraciones del Sprint, se accede a la plataforma Jira y se selecciona el proyecto correspondiente. Posteriormente, se ingresa a la sección **Backlog**, donde se visualizan los Sprints programados y las tareas asociadas a cada uno.

Selección del Sprint:

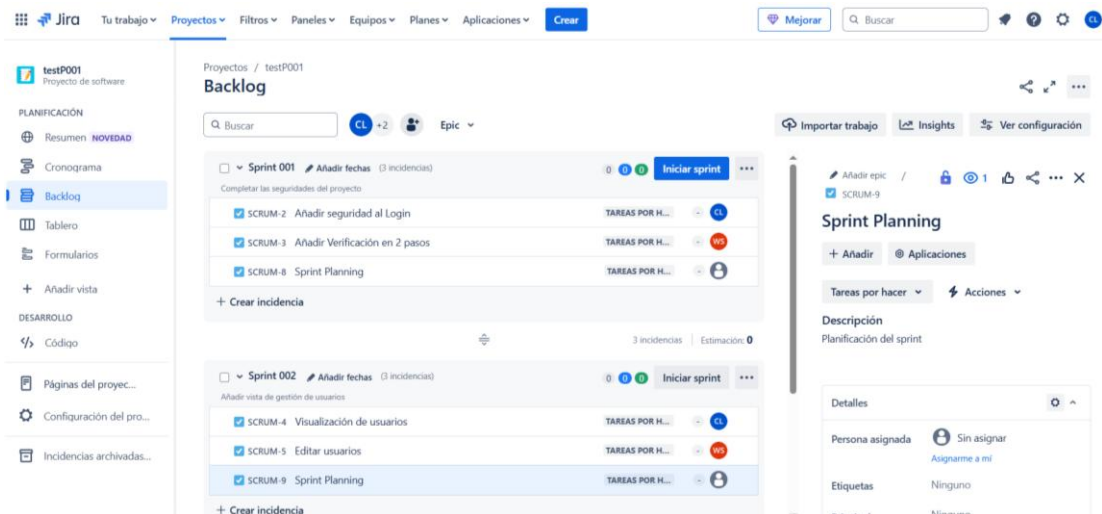
Se identifica el Sprint programado para su inicio dentro del Backlog del proyecto. Posteriormente, se verifica que las tareas y configuraciones necesarias se encuentren correctamente definidas, incluyendo las historias de usuario, los puntos de historia y la asignación de responsables.

Inicio del Sprint:

Para activar el Sprint, se selecciona la opción **“Iniciar Sprint”** correspondiente a la iteración planificada. Esta acción formaliza el comienzo del ciclo de trabajo y habilita la gestión de las tareas en el tablero del equipo, ya sea bajo la vista Scrum o Kanban, según la configuración del proyecto.

Este proceso marca el inicio formal de la iteración del Sprint, permitiendo al equipo enfocarse en el cumplimiento de los objetivos establecidos dentro del marco temporal definido. Asimismo, facilita el seguimiento de las tareas en curso y el monitoreo del progreso hacia la meta del Sprint.

Figura 74.
Opción para iniciar el Sprint



Nota. Elaboración Propia.

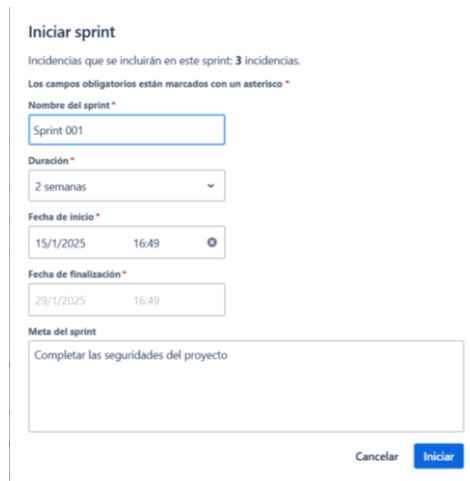
Al seleccionar la opción **“Iniciar Sprint”**, se despliega una ventana emergente que solicita la confirmación de la información relacionada con la iteración. En esta etapa, se revisan los datos proporcionados, que incluyen:

- Nombre del Sprint: debe corresponder al ciclo de trabajo que se desea iniciar.
- Duración del Sprint: se verifica que el período establecido (fecha de inicio y fecha de finalización) esté alineado con la planificación del proyecto.
- Meta del Sprint: se comprueba que el objetivo se encuentre claramente definido y refleje las prioridades del ciclo de trabajo.

Una vez validada la información, se selecciona la opción **“Iniciar”** para activar formalmente el Sprint, habilitando la ejecución de las tareas planificadas por el equipo.

Figura 75.

Formulario de configuración de inicio de Sprint



El formulario, titulado "Iniciar sprint", contiene la siguiente información y campos:

- Encabezado: "Iniciar sprint".
- Subencabezado: "Incidencias que se incluirán en este sprint: 3 incidencias."
- Nota: "Los campos obligatorios están marcados con un asterisco *".
- Campo "Nombre del sprint *": Un input de texto con el valor "Sprint 001".
- Campo "Duración *": Un menú desplegable con el valor "2 semanas".
- Campo "Fecha de inicio *": Un selector de fecha y hora con el valor "15/1/2025 16:49".
- Campo "Fecha de finalización *": Un selector de fecha y hora con el valor "29/1/2025 16:49".
- Campo "Meta del sprint": Un área de texto con el contenido "Completar las seguridades del proyecto".
- Botones de acción: "Cancelar" y "Iniciar".

Nota. Elaboración Propia.

Al iniciar el Sprint, se visualiza un tablero interactivo que refleja las actividades programadas para el ciclo de trabajo. Este tablero ofrece una representación organizada de las tareas asignadas, proporcionando una visión general del progreso de la iteración.

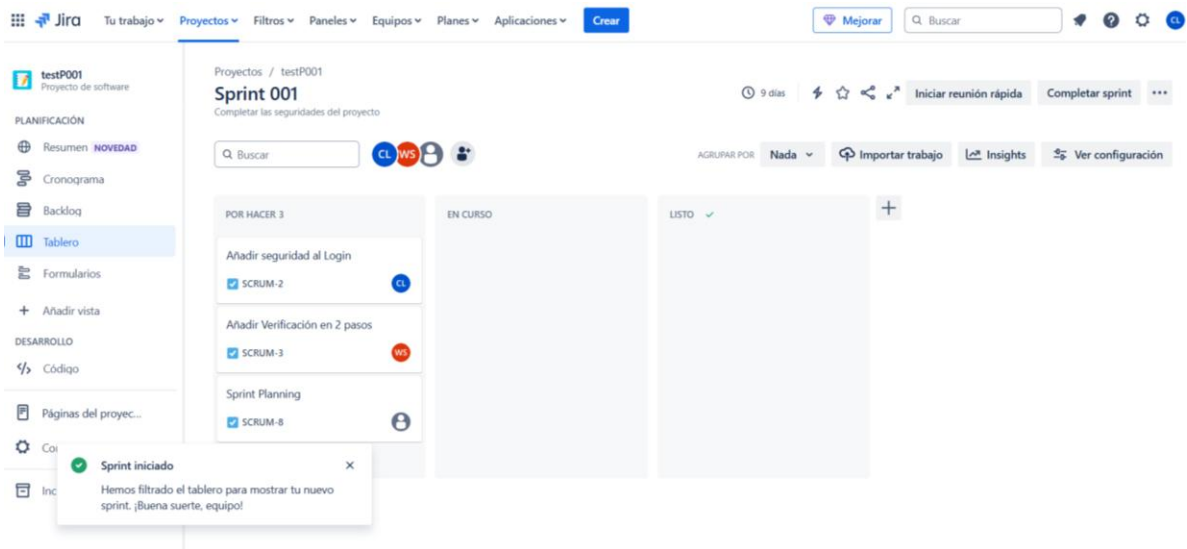
Las actividades se agrupan en columnas que representan distintos estados, como **“Por hacer”**, **“En progreso”** y **“Finalizado”**, lo que permite identificar con claridad las tareas pendientes, en desarrollo o completadas.

Cada incidencia incluye información relevante, como el responsable asignado, el nivel de prioridad, los puntos de historia y el estado actual. El equipo puede actualizar el estado de las tareas conforme avanza el desarrollo.

De esta manera, el tablero constituye una herramienta clave para el monitoreo del Sprint, garantizando una gestión transparente y eficiente dentro del equipo.

Figura 76.

Tablero de monitoreo de Sprints



Nota. Elaboración Propia.

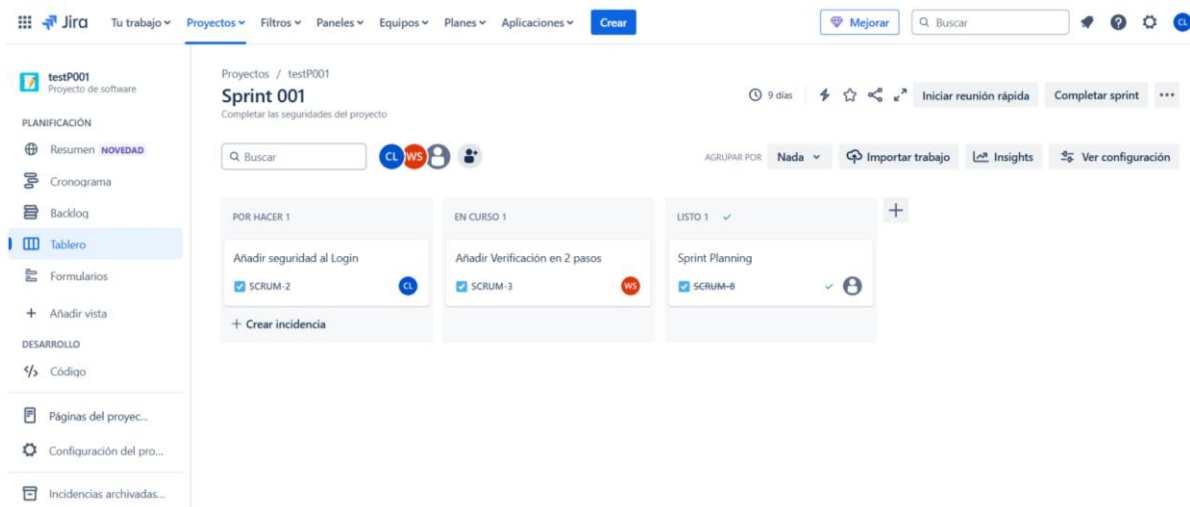
El tablero del Sprint incorpora una funcionalidad interactiva que permite gestionar el estado de las actividades de manera visual e intuitiva. Cada tarea se representa mediante un recuadro que puede desplazarse entre las columnas del tablero para reflejar su estado actual.

La actualización del estado se realiza moviendo la actividad hacia la columna correspondiente, como **“En progreso”** o **“Finalizado”**, lo que permite mantener una representación dinámica del avance del Sprint.

Al realizar estos cambios, el tablero se actualiza automáticamente, proporcionando una visualización precisa del progreso en tiempo real. Este mecanismo favorece la transparencia, la colaboración y el seguimiento continuo de las tareas dentro del equipo.

Figura 77.

Gestión de actividades de los Sprints



Nota. Elaboración Propia.

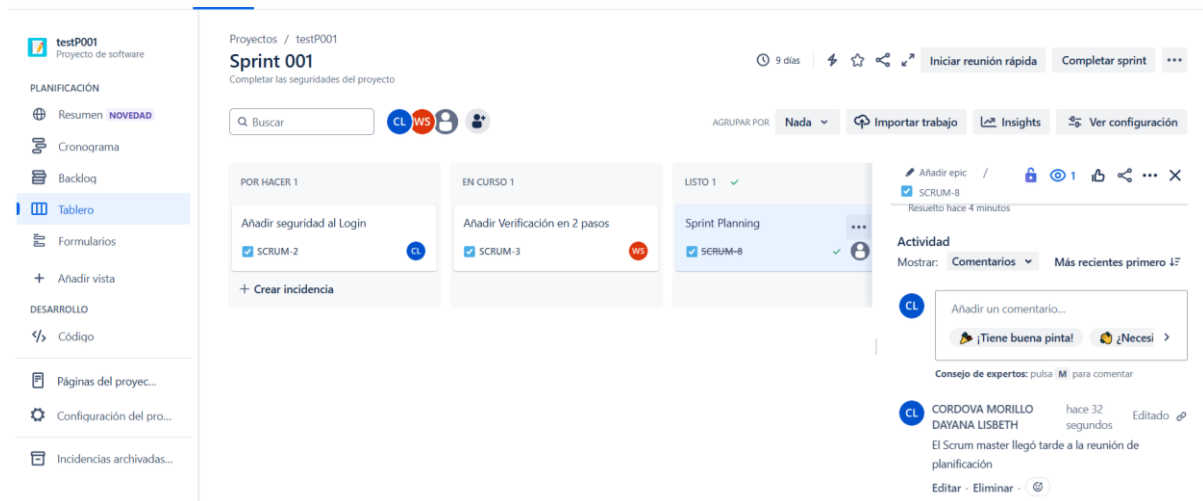
Al completar una tarea dentro del Sprint, resulta pertinente documentar cualquier incidencia o eventualidad ocurrida durante su desarrollo. Este registro se realiza mediante la incorporación de un comentario en la sección correspondiente del *Issue*, garantizando que la información relevante quede almacenada para futuras referencias.

Para ello, se accede a la tarea desde el tablero o el *Backlog* y, en la ventana de detalles, se localiza la sección destinada a comentarios. En este espacio se incorpora una descripción clara y precisa de la incidencia, especificando su naturaleza, la forma en que fue gestionada y cualquier impacto generado en la tarea o en el Sprint.

Una vez redactado el comentario, se registra la información en el sistema. Este procedimiento fortalece la trazabilidad de las incidencias y contribuye a la mejora continua del proyecto, proporcionando insumos relevantes para análisis retrospectivos y ajustes en Sprints posteriores.

Figura 78.

Visualización del Issue



Nota. Elaboración Propia.

9. Cierre de Sprints

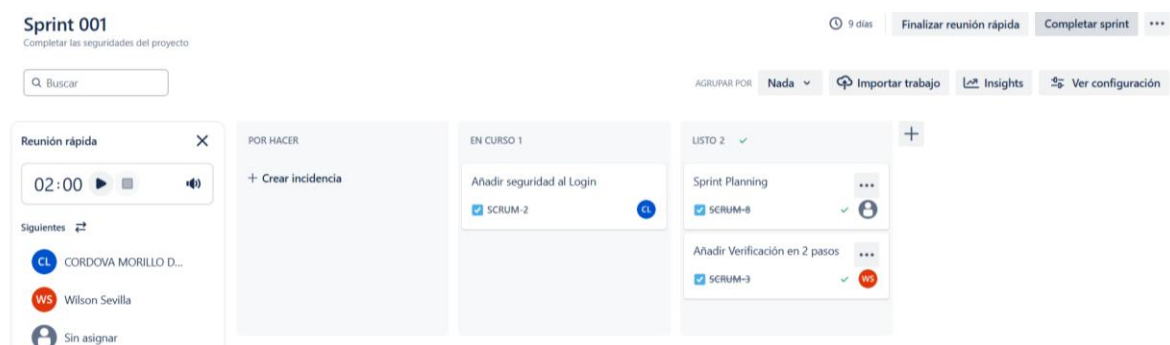
El cierre del Sprint constituye una etapa fundamental dentro del marco Scrum, ya que marca la conclusión formal del ciclo de trabajo y permite evaluar los resultados obtenidos.

Previo a su finalización, se verifica que las actividades completadas se encuentren en la columna “**Listo**” del tablero, reflejando el cumplimiento de los objetivos establecidos para la iteración.

Una vez confirmada esta condición, se selecciona la opción “**Completar Sprint**”, con lo cual se formaliza el cierre del ciclo en el sistema.

Figura 79.

Completar el Sprint



Nota. Elaboración Propia.

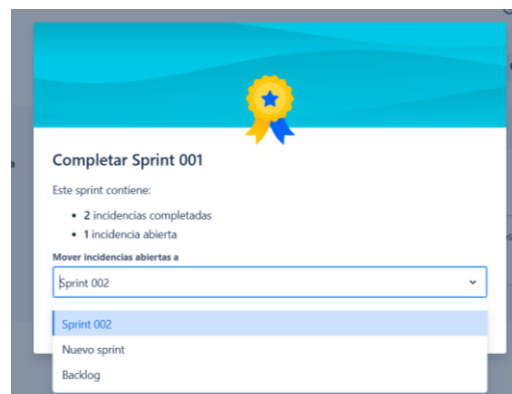
Al finalizar un Sprint, se gestionan las incidencias que no fueron completadas durante la iteración. Estas actividades se trasladan al siguiente Sprint, permitiendo su continuidad y asegurando su futura ejecución.

Para ello, se identifican las tareas que permanecen en estados incompletos, como **“Por hacer”** o **“En progreso”**, verificando que su información se encuentre actualizada y correctamente documentada. Posteriormente, dichas incidencias se asignan al siguiente Sprint planificado dentro del *Backlog*, garantizando su priorización en el nuevo ciclo de trabajo.

Una vez trasladadas las incidencias al siguiente Sprint, se inicia el nuevo ciclo de trabajo conforme al procedimiento establecido para la activación del Sprint en Jira.

Figura 80.

Gestión de incidentes pendientes



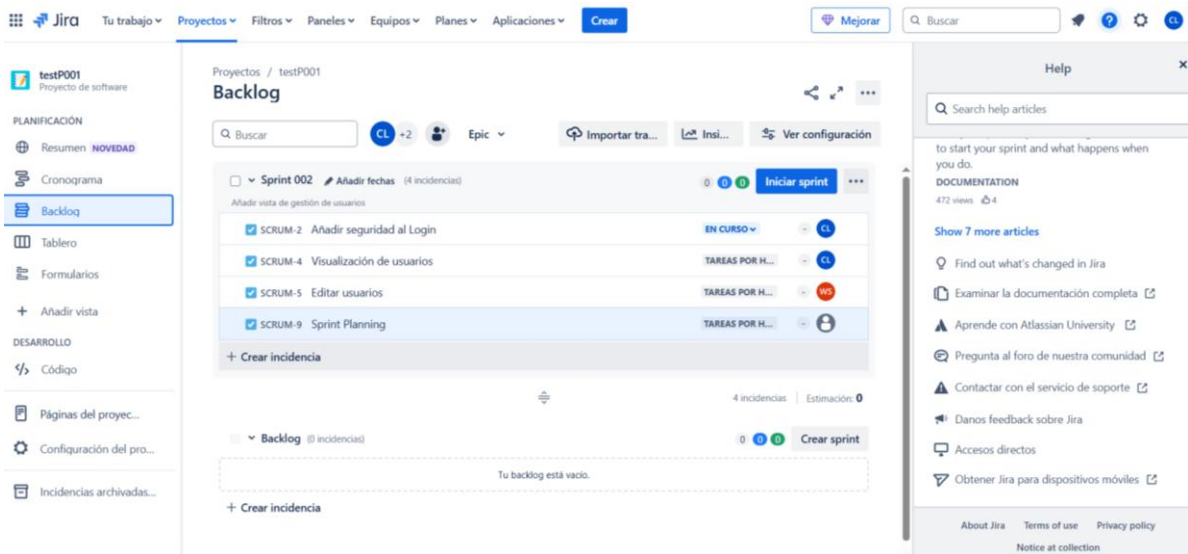
Nota. Elaboración Propia.

Al revisar el *Backlog*, se observa que las incidencias pendientes del Sprint anterior han sido correctamente trasladadas al siguiente. Estas actividades se integran con las nuevas tareas planificadas, conformando un conjunto consolidado que será gestionado durante la iteración.

Esta organización permite mantener claramente identificadas tanto las tareas pendientes como las nuevas, asegurando su alineación con los objetivos del proyecto y favoreciendo una gestión estructurada dentro del marco Scrum.

Figura 81.

Revisión de los incidentes pendientes del Sprint anterior



Nota. Elaboración Propia.

10. Sprint retrospective

La retrospectiva del Sprint constituye una etapa clave dentro del marco Scrum, ya que permite al equipo reflexionar sobre el ciclo de trabajo finalizado, identificar oportunidades de mejora y fortalecer las prácticas que resultaron efectivas.

Para desarrollar esta actividad en Jira, se realizan las siguientes acciones:

Acceso al Sprint correspondiente:

Se accede al tablero o al *Backlog* del proyecto en Jira y se identifica el Sprint que ha finalizado o que se encuentra en curso.

Creación de la incidencia de retrospectiva:

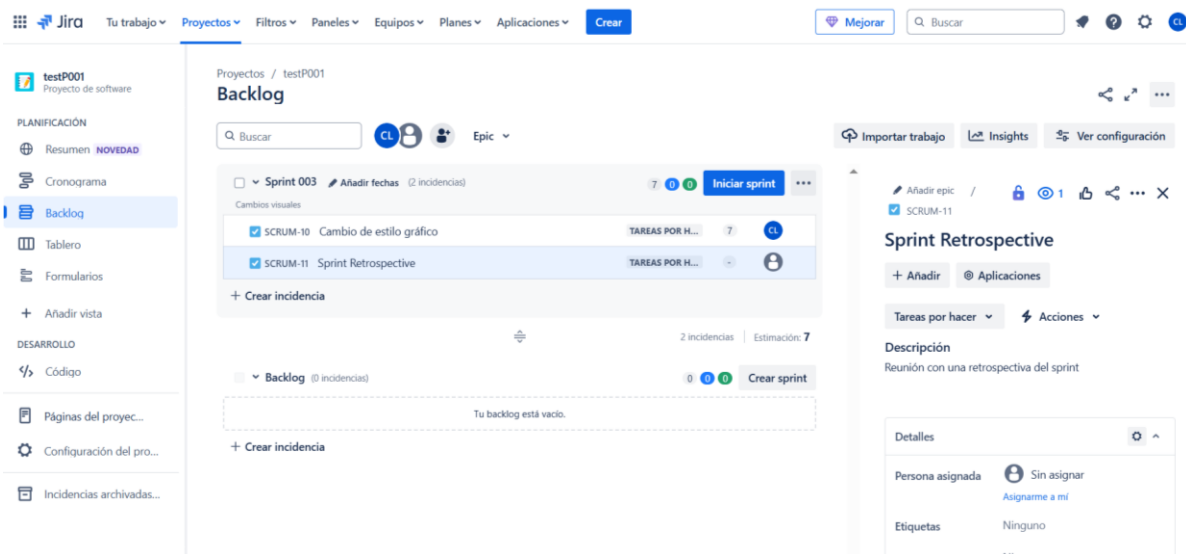
Dentro del Sprint seleccionado, se crea una nueva incidencia denominada “**Sprint Retrospective**”, la cual funcionará como espacio para el registro de observaciones y acuerdos.

Registro de resultados

En la incidencia creada se documentan los aspectos positivos del Sprint, las dificultades identificadas y las acciones de mejora propuestas para la siguiente iteración.

Este procedimiento permite sistematizar los resultados de la retrospectiva y facilita la implementación de mejoras en Sprints posteriores, fortaleciendo el principio de mejora continua propio del marco Scrum.

Figura 82.
Sprint retrospective



Nota. Elaboración Propia.

Para complementar el análisis de la retrospectiva del Sprint, se accede al apartado **“Resumen”** en Jira, el cual proporciona una vista general del estado del proyecto y facilita la evaluación del flujo de trabajo desarrollado durante la iteración.

Acceso al apartado de Resumen:

Desde el menú principal de Jira, se selecciona la sección **“Resumen”** correspondiente al proyecto en curso. Este apartado presenta un panel con información clave sobre el progreso y las actividades registradas.

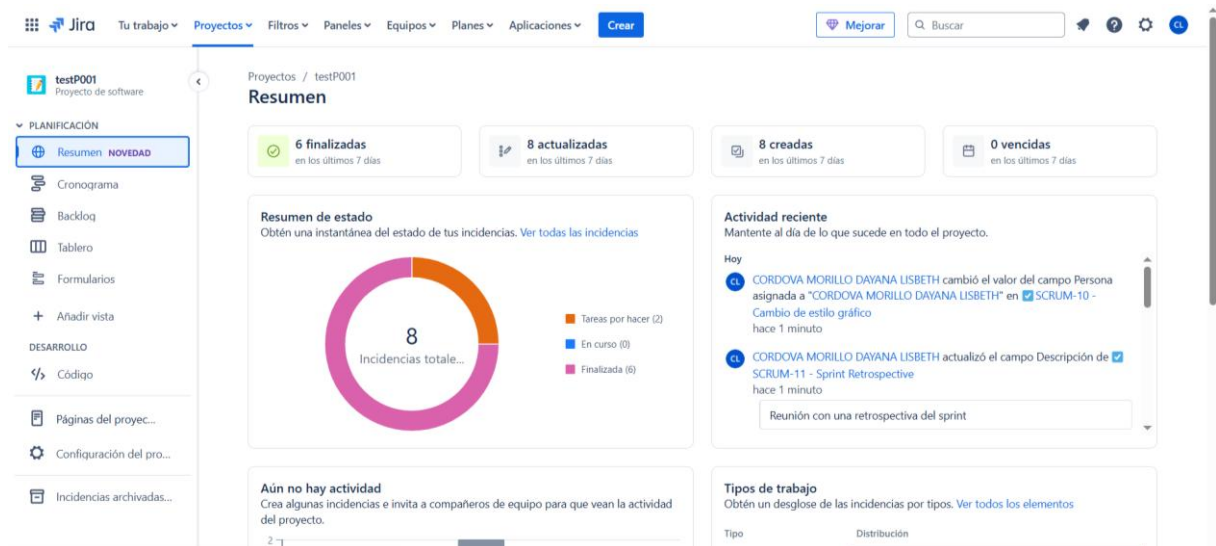
Análisis de actividades finalizadas y pendientes:

En esta vista se identifican las secciones que detallan el estado de las actividades, incluyendo tareas finalizadas, actualizadas, creadas o vencidas durante el Sprint. La revisión

de estos datos permite analizar el flujo de trabajo, detectar oportunidades de mejora y evaluar el grado de cumplimiento de los objetivos establecidos.

Figura 83.

Resumen de las actividades



Nota. Elaboración Propia.

Figura 84.

Resumen de los incidentes completados y pendientes

The screenshot shows the Jira 'Lista' (List) page for project testP001. The table displays the following data:

	Tipo	#	Clave	Resumen	Estado	Sprint	Persona asignada	Fecha
☐	☒		SCRUM-9	Sprint Planning	FINALIZADA	Sprint 002		
☐	☒		SCRUM-5	Editar usuarios	FINALIZADA	Sprint 002	Wilson Sevilla	
☐	☒		SCRUM-4	Visualización de usuarios	FINALIZADA	Sprint 002	CORDOVA MORILLO...	
☐	☒		SCRUM-2	Añadir seguridad al Login	FINALIZADA	Sprint 002	CORDOVA MORILLO...	
☐	☒		SCRUM-8	Sprint Planning	FINALIZADA	Sprint 001		
☐	☒		SCRUM-3	Añadir Verificación en 2 pasos	FINALIZADA	Sprint 001	Wilson Sevilla	
☐	☒		SCRUM-10	Cambio de estilo gráfico	TAREAS POR HAC...	Sprint 3	CORDOVA MORILLO...	
☐	☒		SCRUM-11	Sprint Retrospective	TAREAS POR HAC...	Sprint 003		

Nota. Elaboración Propia.

Figura 85.
Incidentes completados

	Tipo	# Clave	Resumen	Estado	Sprint	Persona asignada	Fecha
☐	☒	SCRUM-9	Sprint Planning	FINALIZADA	Sprint 002		
☐	☒	SCRUM-5	Editar usuarios	FINALIZADA	Sprint 002	Wilson Sevilla	
☐	☒	SCRUM-4	Visualización de usuarios	FINALIZADA	Sprint 002	CORDOVA MORILLO...	
☐	☒	SCRUM-2	Añadir seguridad al Login	FINALIZADA	Sprint 002	CORDOVA MORILLO...	
☐	☒	SCRUM-8	Sprint Planning	FINALIZADA	Sprint 001		
☐	☒	SCRUM-3	Añadir Verificación en 2 pasos	FINALIZADA	Sprint 001	Wilson Sevilla	

Nota. Elaboración Propia.

2.1.6.3. Fase 3. Post-Juego

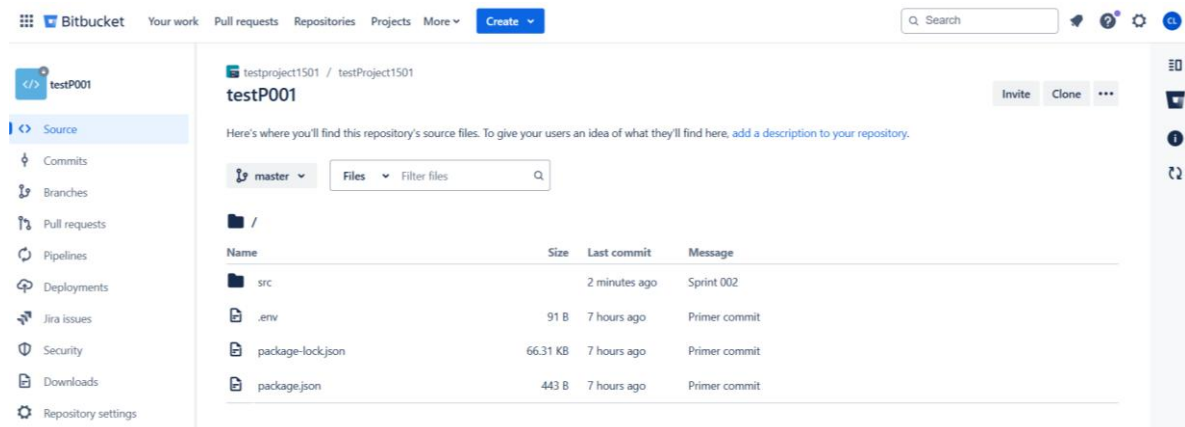
11. Etiquetado de una versión de lanzamiento (Tag)

El etiquetado de una versión de lanzamiento constituye una práctica fundamental para identificar y registrar hitos relevantes dentro del desarrollo del proyecto. Las etiquetas (*tags*) permiten asociar estados específicos del código con versiones determinadas, facilitando la trazabilidad y el control de versiones.

- Se ingresa al repositorio correspondiente dentro de la plataforma Bitbucket.
- Desde el menú lateral izquierdo, se accede a la sección “**Commits**”, donde se visualiza el historial de confirmaciones realizadas en el repositorio, incluyendo información como mensajes de *commit*, autores y fechas.

El acceso a esta sección constituye el paso previo para la creación y gestión de etiquetas asociadas a versiones específicas del proyecto.

Figura 86.
Repositorio actualizado



Nota. Elaboración Propia.

Para etiquetar una versión específica en Bitbucket, se debe identificar previamente el *commit* correspondiente dentro de la rama de trabajo. Para ello, se desarrollan las siguientes acciones:

Selección de la rama

Dentro de la sección “**Commits**”, se utiliza el menú desplegable para seleccionar la rama del repositorio en la que se desea trabajar, por ejemplo, “**master**” o “**main**”, según la configuración del proyecto.

Identificación del commit

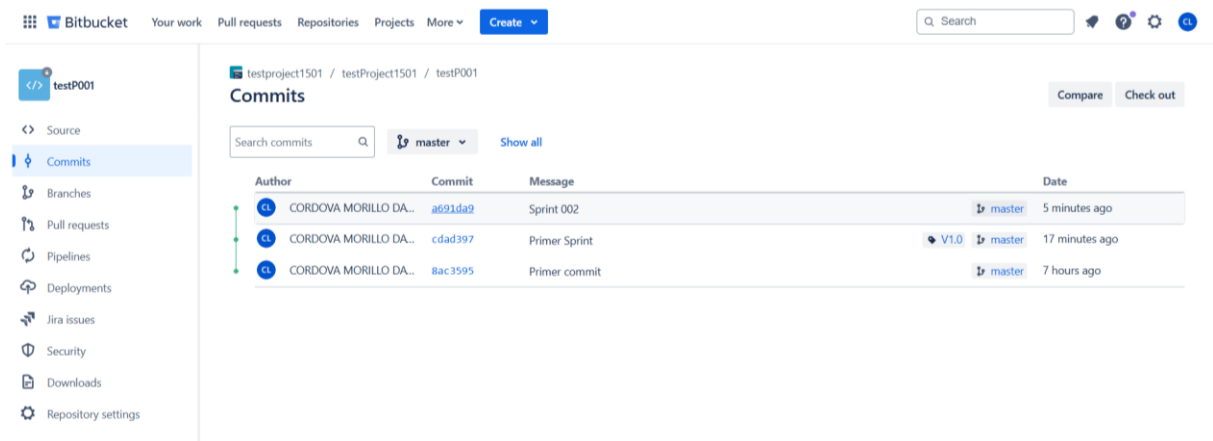
Se revisa el listado de confirmaciones hasta localizar el *commit* específico que se desea asociar a la versión.

Acceso a los detalles del commit

Una vez identificado, se selecciona el ID del *commit*, el cual corresponde a un identificador único que permite acceder a la información detallada de esa confirmación.

Este procedimiento garantiza la correcta identificación del punto exacto del código que será etiquetado, fortaleciendo la organización y la trazabilidad de las versiones del proyecto.

Figura 87.
Sección de commits del repositorio



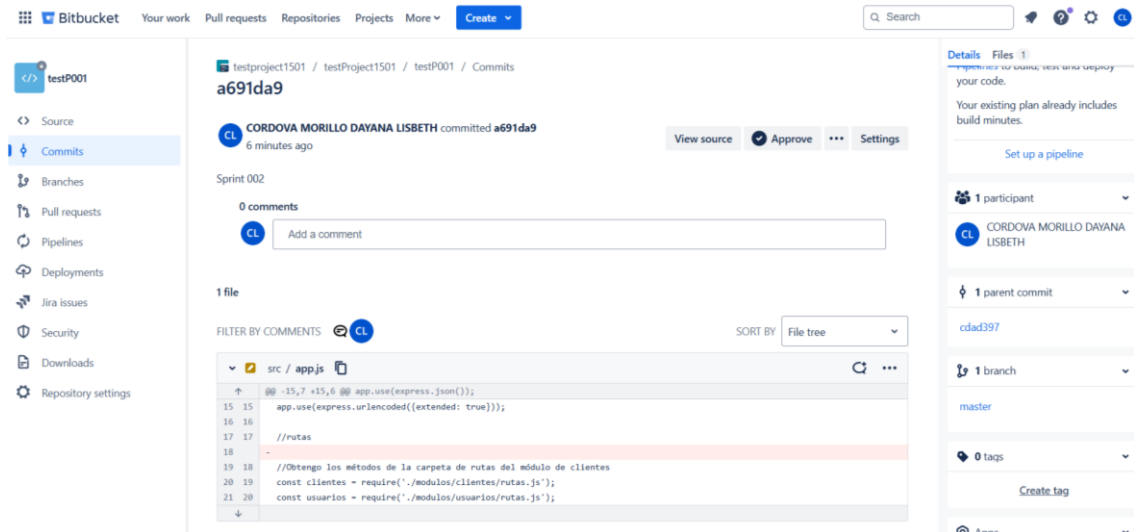
Nota. Elaboración Propia.

Una vez accedido a los detalles del *commit* seleccionado, se procede a la creación de una etiqueta (*tag*) con el fin de marcar una versión de lanzamiento del proyecto, para ello se desarrollan las siguientes acciones:

- En la página de detalles del *commit*, se identifica la sección denominada **“Tags”**, donde se gestionan las etiquetas asociadas.
- Dentro de esta sección, se selecciona la opción correspondiente para generar una nueva etiqueta vinculada al *commit* elegido.

Este procedimiento permite asociar formalmente una versión específica a un estado determinado del repositorio, fortaleciendo la gestión y trazabilidad de las versiones del proyecto.

Figura 88.
Sección de Tags del repositorio



Nota. Elaboración Propia.

- En el campo correspondiente, se establece un nombre descriptivo que represente claramente la versión del proyecto. Se recomienda utilizar un formato estandarizado, como **v1.0.0**, para identificar una versión inicial estable o un hito relevante.
- En el campo de descripción, se incorpora información relevante sobre la versión asociada a la etiqueta. Esta puede incluir objetivos alcanzados, funcionalidades implementadas o correcciones realizadas durante el Sprint.
- Una vez configurados el título y la descripción, se confirma la creación del **tag**, quedando registrada la versión en el repositorio.

Figura 89.
Formulario de creación de Tags

Create tag

♦ tag "V1.0" already exists

Tag name *

V1.0.0

Description

Lanzamiento de la versión 1.0 de las rutas del proyecto

Cancel Create tag

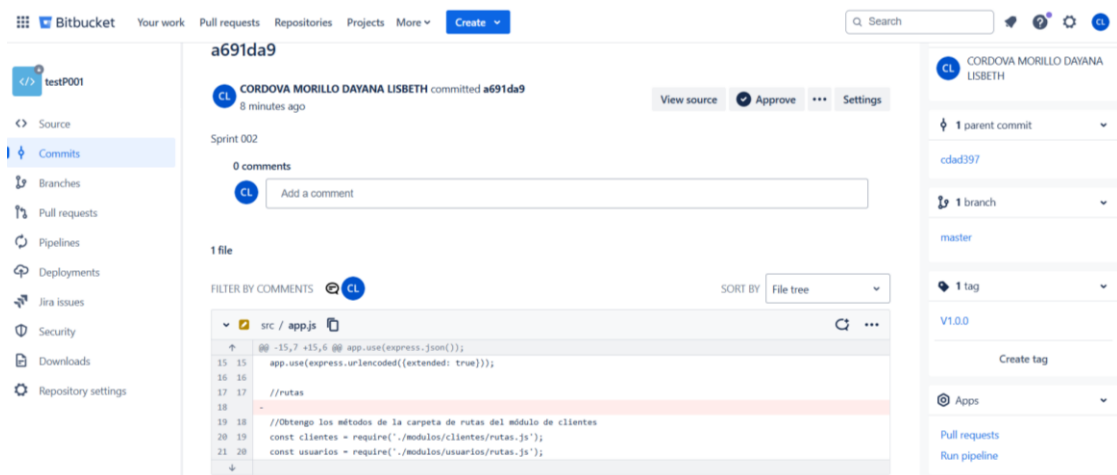
Nota. Elaboración Propia.

Una vez completado el proceso de creación de la etiqueta, el *commit* asociado muestra de forma visible la información correspondiente al *tag* registrado. Esta visualización permite identificar con facilidad el punto exacto del historial que ha sido marcado como versión de lanzamiento.

El nombre de la etiqueta aparece destacado junto al ID del *commit*, indicando que dicho estado del repositorio ha sido formalmente versionado. Asimismo, la descripción incorporada durante la configuración permanece disponible para consulta, proporcionando contexto adicional sobre los cambios, funcionalidades o correcciones asociadas a esa versión.

Este mecanismo facilita la identificación de hitos relevantes dentro del desarrollo y fortalece la gestión y trazabilidad de versiones en el proyecto.

Figura 90.
Commit etiquetado



Nota. Elaboración Propia.

2.1.7. Conclusiones

La guía metodológica desarrollada en este capítulo permitió estructurar de manera sistemática la implementación del marco Scrum mediante la integración de Bitbucket y Jira como herramientas de apoyo para la gestión de proyectos de software.

Bitbucket proporciona un entorno colaborativo orientado al control de versiones y la administración del código fuente, mientras que Jira facilita la planificación, organización y seguimiento de las tareas dentro de cada Sprint. La articulación entre ambas plataformas fortalece la trazabilidad entre los requerimientos definidos y los cambios realizados en el repositorio, asegurando coherencia en el desarrollo del proyecto.

La integración propuesta no solo optimiza la administración del código, sino que también mejora la coordinación del equipo, al vincular incidencias, Sprints y versiones de lanzamiento dentro de un flujo de trabajo ágil estructurado. Asimismo, el uso de etiquetas de versión y la documentación de incidencias contribuyen al control de configuraciones y a la mejora continua del proceso iterativo.

En este sentido, la guía metodológica presentada ofrece un procedimiento claro, organizado y replicable para la implementación de Scrum en entornos académicos, permitiendo a los equipos de desarrollo gestionar el ciclo de vida del software de manera eficiente, transparente y alineada con buenas prácticas profesionales.

CAPÍTULO 3

3. Validación de resultados

Con el objetivo de validar la guía metodológica para gestionar proyectos de software utilizando Bitbucket y el marco de trabajo Scrum, se realizó una prueba de concepto que consistió en el desarrollo de un sistema básico denominado “Gestión de documentos UTN”. El desarrollo se llevó a cabo siguiendo rigurosamente los lineamientos definidos en la guía, permitiendo poner en práctica sus fases y herramientas propuestas. Se validó la usabilidad mediante el cuestionario SUS y la funcionalidad bajo la característica de adecuación funcional de la norma ISO/IEC 25010.

3.1. Aplicación de la guía metodológica

3.1.1. Fase 1: Pre – Juego

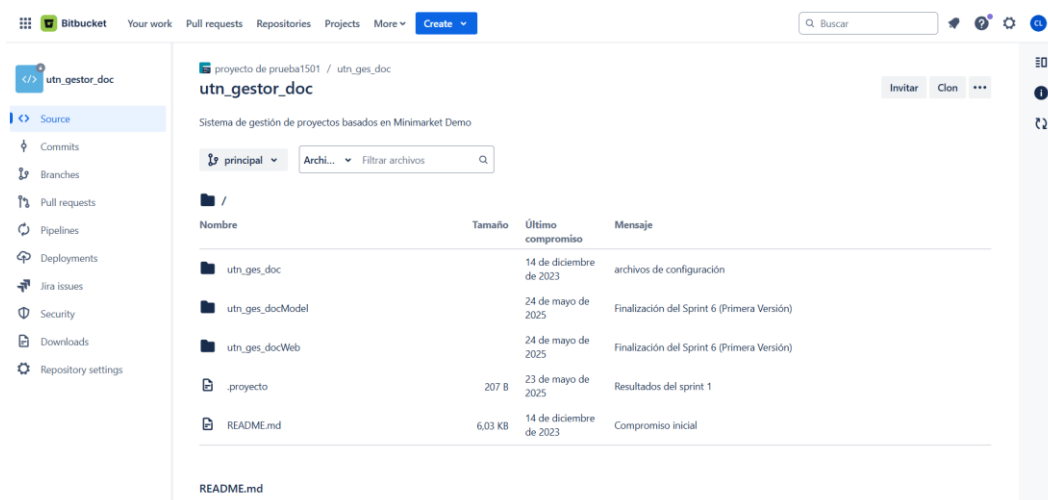
Durante esta fase se establecieron las bases técnicas y organizativas para la implementación del proyecto. Se configuró Git a nivel local y se vinculó el repositorio con Bitbucket, donde posteriormente se creó el repositorio remoto dentro del espacio de trabajo correspondiente. Asimismo, se realizaron las configuraciones necesarias para habilitar el control de versiones colaborativo.

En el ámbito metodológico, se definieron los roles del marco Scrum, asignando las funciones de Product Owner, Scrum Master y equipo de desarrollo. Posteriormente, se configuraron los permisos de acceso de los colaboradores en el repositorio y se importó la base del sistema mediante la opción “Import repository”, garantizando la disponibilidad inicial del proyecto.

De igual manera, se configuró el entorno de desarrollo Eclipse para clonar y administrar el repositorio remoto, realizando pruebas iniciales de commit y push con el fin de verificar la correcta sincronización entre el entorno local y la plataforma remota. Finalmente, se integró Bitbucket con Jira, logrando la vinculación entre el código fuente y la planificación ágil del proyecto.

Esta fase permitió comprobar la viabilidad operativa de la guía metodológica en un entorno real de desarrollo, asegurando la correcta articulación entre herramientas y marco de trabajo.

Figura 91.
Repositorio creado en Bitbucket con estructura del sistema



Nota. Elaboración Propia.

3.1.2. Fase 2: Juego

Esta Fase constituyó el núcleo del desarrollo del proyecto, en la cual se aplicaron de manera directa los principios del marco Scrum mediante iteraciones (Sprints) gestionadas desde la plataforma Jira.

Se definió el Product Backlog a partir de historias de usuario que contemplaban funcionalidades relacionadas con la gestión de tesis, avances, resoluciones, docentes, estudiantes, carrera y facultades, así como la visualización de datos, filtros de búsqueda y generación de reportes PDF. A cada historia se le asignó un nivel de prioridad y estimación de esfuerzo, permitiendo organizar el trabajo de manera estructurada.

Las historias fueron distribuidas en Sprints planificados en Jira; en total, se desarrollaron 6 iteraciones. En cada una se ejecutaron las actividades correspondientes al ciclo Scrum: planificación del Sprint, desarrollo de tareas, revisión de resultados y retrospectiva, incluyendo replanificación cuando fue necesario.

Durante las iteraciones se implementaron componentes funcionales y visuales del sistema, tales como formularios, tablas dinámicas, filtros, exportaciones a PDF y conexiones entre tablas de datos. Paralelamente, se utilizó el control de versiones en Bitbucket para gestionar los cambios y etiquetar las versiones entregables al finalizar cada Sprint, fortaleciendo la trazabilidad y el control del desarrollo.

Figura 92.
Gestión de historias de usuario y planificación de Sprints en Jira

Project: Systema_utm_ges_doc

Search: Buscar en la lista | Filter: [C]

	Tipo	Clave	Resumen	Estado	Sprint	Persona asignada	Fecha de vi
☐	> ☒	SUGD-2	Gestión de tesis	FINALIZADA	Sprint 1	CORDOVA MORILL...	
☐	> ☒	SUGD-3	Gestión de avances	FINALIZADA	Sprint 1	CORDOVA MORILL...	
☐	> ☒	SUGD-4	Gestión de resoluciones	FINALIZADA	Sprint 1	CORDOVA MORILL...	
☐	> ☒	SUGD-12	Visualización de docentes	FINALIZADA	Sprint 5	CORDOVA MORILL...	
☐	> ☒	SUGD-13	Visualización de estudiantes	FINALIZADA	Sprint 5	CORDOVA MORILL...	
☐	> ☒	SUGD-10	Visualización de avances	FINALIZADA	Sprint 4	CORDOVA MORILL...	
☐	> ☒	SUGD-11	Visualización de resoluciones	FINALIZADA	Sprint 4	CORDOVA MORILL...	
☐	> ☒	SUGD-5	Gestión de docentes	FINALIZADA	Sprint 2	CORDOVA MORILL...	
☐	> ☒	SUGD-6	Gestión de estudiantes	FINALIZADA	Sprint 2	CORDOVA MORILL...	

Nota. Elaboración Propia.

Figura 93.
Vista funcional del sistema: Gestión y seguimiento de tesis

Regresar | Registro de anteproyectos | Sesiones | Carreras | Facultades | Docentes | Estudiantes | Periodos | dlcordovam@utm.edu.ec

Seguimiento de tesis - Secretario | Seleccionar columnas | Exportar a PDF

<< < 1 > >>

Título	Nombre Estudiante	Cédula Estudiante	Director	Carrera	Porcentaje de avance de tesis	Acciones
Dasarrollo de una guía metodológica para gestión de proyectos utilizando Bitbucket y Scrum para apoyar los proyectos de desarrollo	Lisbeth Córdova	0401846802	Mauricio Rea	Telecomunicaciones	Último porcentaje de avance: 76%	<button>Editar</button> <button>Eliminar</button>
Desarrollo de un sistema contable	Wilson Sevilla	1025478127	Mauricio Rea	Telecomunicaciones	Sin avances registrados	<button>Editar</button> <button>Eliminar</button>
Desarrollo de videojuego	Jorge Jaramillo	1025478651	Carpio Pineda	Telecomunicaciones	Sin avances registrados	<button>Editar</button> <button>Eliminar</button>

Nota. Elaboración Propia.

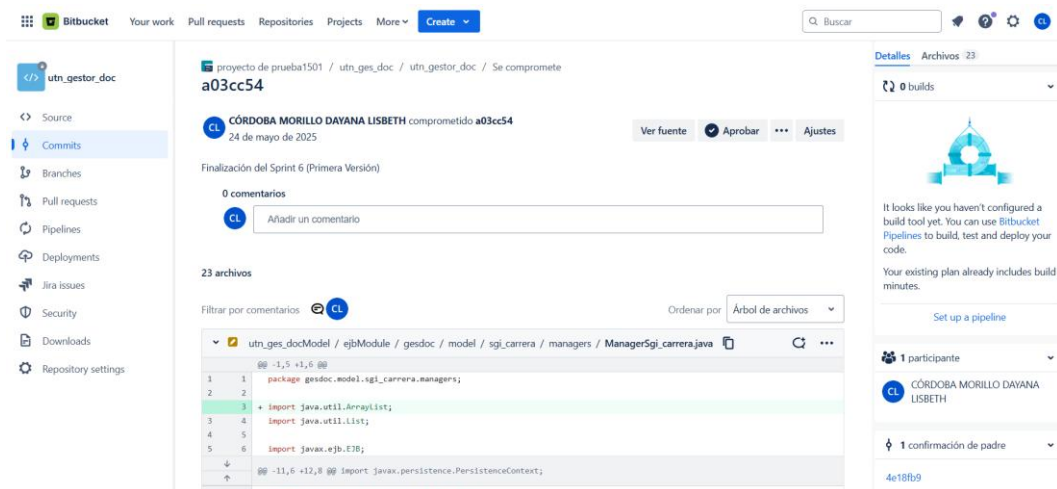
3.1.3. Fase 3: Post – Juego

En esta fase se consolidaron los resultados del desarrollo. Cada versión entregable fue documentada mediante etiquetas (tags) en Bitbucket, representando formalmente el cierre de cada Sprint. Asimismo, se verificó que los cambios implementados se encontraran correctamente reflejados y disponibles en el repositorio remoto.

De manera complementaria, se recopilaron los resultados obtenidos durante las iteraciones y se ejecutaron retrospectivas orientadas a la mejora continua. Los registros del proyecto fueron actualizados y los entregables organizados por Sprint, garantizando una estructura clara de avance.

Finalmente, se aseguró la trazabilidad del trabajo realizado mediante la vinculación entre commits y tareas gestionadas en Jira, permitiendo mantener coherencia entre la planificación ágil y el control de versiones.

Figura 94.
Confirmación de versión etiquetada al finalizar un Sprint



Nota. Elaboración Propia.

El detalle completo de la prueba de concepto, incluyendo capturas de configuración, evidencias de implementación y registros del proceso, se encuentra disponible en el apartado de **ANEXOS** del presente documento.

3.2. Elaboración de cuestionario

Con el propósito de evaluar la usabilidad y la funcionalidad del sistema desarrollado, se elaboró un cuestionario mediante la herramienta Microsoft Forms, estructurado en dos secciones y conformado por un total de 15 preguntas.

La primera sección incluyó 10 ítems correspondientes a la Escala de Usabilidad del Sistema (SUS), orientados a medir la percepción del usuario respecto a la facilidad de uso. La segunda sección estuvo compuesta por 5 preguntas fundamentadas en los principios de la norma ISO/IEC 25010, específicamente enfocadas en la evaluación de la adecuación funcional del sistema.

Esta combinación metodológica permitió obtener una valoración integral de la calidad del software, integrando tanto la experiencia del usuario como el grado de cumplimiento de requisitos funcionales establecidos.

3.2.1. Preguntas de usabilidad

De acuerdo con Vlachogianni & Tselios (2021), la encuesta SUS (System Usability Scale) es una herramienta ampliamente reconocida por su validez y confiabilidad en la evaluación de la usabilidad percibida en sistemas y productos tecnológicos. El instrumento está compuesto por diez ítems formulados de manera alternada entre afirmaciones positivas y negativas, los cuales se valoran mediante una escala de Likert de cinco puntos que va desde “Totalmente en desacuerdo” hasta “Totalmente de acuerdo”. Debido a su versatilidad y efectividad, incluso cuando se aplica en muestras reducidas, ha sido adoptada como uno de los métodos más confiables para medir la experiencia del usuario.

Con base a lo anterior, la primera sección del cuestionario incluyó 10 preguntas fundamentadas en la versión adaptada al español propuesta por B. Obando (2025), la cual conserva la estructura original del instrumento, manteniendo la alternancia entre afirmaciones positivas y negativas, tal como se presenta en la tabla 9:

Tabla 9.
Preguntas de usabilidad

Nro	Pregunta
1	¿Creo que me gustaría usar este sistema con frecuencia?
2	¿Me pareció que el sistema tiene un nivel de complejidad mayor al necesario?
3	¿Considero que el sistema fue fácil de usar?
4	¿Creo que necesitaría ayuda de un técnico para poder usar el sistema?
5	¿Considero que las funciones del sistema están bien integradas?

-
- | | |
|----|--|
| 6 | ¿Creo que el sistema tiene demasiadas inconsistencias? |
| 7 | ¿Considero que la mayoría de las personas aprenderían a usar el sistema rápidamente? |
| 8 | ¿Tuve dificultades para utilizar el sistema debido a su complejidad? |
| 9 | ¿Me siento seguro/a al usar el sistema? |
| 10 | ¿Necesité aprender muchas cosas antes de poder usar el sistema? |
-

Nota. Adaptado de B. Obando (2025)

3.2.2. Preguntas de funcionalidad

La norma ISO/IEC 25010 fue utilizada para el análisis de la adecuación funcional del sistema "Gestión de Documentos UTN", con el propósito de evaluar si las funcionalidades implementadas responden de manera adecuada a las necesidades operativas de la institución. Su aplicación permitió determinar el grado de cumplimiento de los requerimientos definidos, verificando que la solución proporcione resultados coherentes, relevantes y completos en relación con sus objetivos.

Esta evaluación contribuye a validar la calidad funcional desde una perspectiva centrada en el usuario, alineando el desempeño del sistema con los estándares internacionales de la calidad del software.

En la segunda sección del cuestionario se incluyeron preguntas orientadas a medir la adecuación funcional conforme a los criterios establecidos por la norma ISO/IEC 25010. A continuación, se presentan los ítems considerados para dicha evaluación.

¿El sistema cubre todas las necesidades esperadas?

Motivación: Esta pregunta permite evaluar la completitud funcional, verificando que se hayan implementado todas las operaciones necesarias para responder adecuadamente a los procesos documentales de la institución.

¿El software tiene validaciones necesarias para los campos de ingreso de información?

Motivación: Esta cuestión está orientada a medir la fiabilidad y consistencia de los datos capturados, asegurando que existan mecanismos que prevengan errores y protejan la integridad del repositorio documental.

¿Las funciones disponibles facilitan la realización de tareas y objetivos planteados?

Motivación: Con este ítem se analiza la adecuación funcional en términos de apoyo operativo, verificando que las funcionalidades implementadas contribuyan a agilizar y simplificar las actividades del usuario final.

¿Existen funciones que considero innecesarias o que dificultan el uso del sistema?

Motivación: Esta pregunta permite identificar posibles funcionalidades redundantes o contraproducentes, contribuyendo a optimizar la pertinencia y eficiencia del sistema, eliminando o ajustando elementos que entorpezcan el flujo de trabajo.

¿El sistema satisface tus necesidades esperadas?

Motivación: Este ítem proporciona una valoración global de aceptación y cumplimiento funcional, permitiendo contrastar las expectativas iniciales con el desempeño real del sistema.

3.3. Resultados obtenidos

3.3.1. Evaluación de usabilidad

Los resultados del cuestionario se presentan en la tabla 10 donde, en la primera columna se detallan las opciones de respuesta correspondiente a la escala de Likert utilizada. Las columnas numeradas del 1 al 10 representan cada uno de los ítems incluidos en la sección de evaluación de usabilidad del instrumento.

Tabla 10.
Resultados de usabilidad

Pregunta	1	2	3	4	5	6	7	8	9	10
Totalmente en desacuerdo	0	4	0	10	0	10	1	14	0	11
En desacuerdo	2	13	0	11	2	12	1	12	1	13
Neutral	4	9	3	8	5	7	2	2	6	3
De acuerdo	14	5	18	2	14	3	16	4	18	4
Totalmente de acuerdo	12	1	11	1	11	0	12	0	7	1
Total	32	32	32	32	32	32	32	32	32	32

Nota. Elaboración propia

3.3.1.1. Análisis de resultados

Con el propósito de calcular la puntuación del cuestionario SUS, se aplicó el procedimiento estandarizado propuesto por sus autores, el cual se desarrolla en las siguientes etapas:

1. **Ajuste de las respuestas:** En las preguntas impares (afirmaciones con orientación positiva), al valor seleccionado por el participante en la escala de 1 a 5 se le restará 1. En las preguntas pares (afirmaciones con orientación negativa), el valor seleccionado se restó 5.
2. **Cálculo de la puntuación parcial:** Se sumaron los valores resultantes de los ajustes realizados en las diez preguntas del cuestionario, obteniendo así un puntaje intermedio.
3. **Escalado a la escala 0–100:** El puntaje intermedio obtenido se multiplicó 2,5, con el fin de transformar el resultado a una escala de 0 y 100, donde 100 representa la máxima percepción de usabilidad.

Ajuste de respuestas

Para ilustrar el procedimiento de ajuste, se tomó como ejemplo el cálculo correspondiente a las preguntas 1 y 2 del cuestionario.

Pregunta Impar

En las preguntas con orientación positiva, se aplicó la siguiente fórmula:

$$\Sigma (\text{valor obtenido} - 1)$$

Para la pregunta 1, el cálculo fue el siguiente:

$$12(5 - 1) + 14(4 - 1) + 4(3 - 1) + 2(2 - 1) + 0(1 - 1)$$

$$48 + 42 + 8 + 2 + 0 = 100$$

Este procedimiento se replicó en cada una de las preguntas impares del instrumento, presentándose sus resultados en la sección siguiente:

Tabla 11.
Resultados de las preguntas impares de usabilidad

Pregunta	Resultado
1	100
3	104
5	98
7	101
9	95

Nota. Elaboración propia

Ahora se realiza el procedimiento para las preguntas pares utilizando la pregunta 2 como ejemplo:

Pregunta Par

Para las preguntas con orientación negativa (preguntas pares), se aplicó la siguiente fórmula:

$$\Sigma (5 - \text{valor obtenido})$$

Como ejemplo, para la Pregunta 2, el cálculo fue el siguiente:

$$4(5 - 1) + 13(4 - 1) + 9(3 - 1) + 5(2 - 1) + 1(1 - 1)$$

$$16 + 39 + 18 + 5 + 0 = 78$$

Este mismo procedimiento se aplicó a cada una de las preguntas pares del cuestionario, cuyos resultados se presentan a continuación:

Tabla 12.
Resultados de las preguntas pares de usabilidad

Pregunta	Resultado
2	78
4	91
6	93
8	100
10	93

Nota. Elaboración propia

Cálculo de la puntuación SUS

Cálculo de la puntuación parcial

Se procedió a sumar los valores ajustados obtenidos en cada una de las diez preguntas del cuestionario:

$$100 + 78 + 104 + 91 + 98 + 93 + 101 + 100 + 95 + 93 = 953$$

Obtención del promedio por participante

Dado que la encuesta fue aplicada a 32 participantes, el total acumulado se dividió para el número de encuestados:

$$\frac{953}{32} = 29.78$$

Escalado a la puntuación SUS (0 – 100)

Finalmente, el valor promedio obtenido se multiplicó por 2.5, conforme al procedimiento estándar del método SUS:

$$29.78 * 2.5 = 74.45$$

El sistema obtuvo una puntuación SUS de 74,45 sobre 100, valor que se sitúa por encima del promedio estándar internacional (68). Este resultado indica un nivel de usabilidad “Bueno”, según la clasificación propuesta en estudios de interpretación de la escala SUS. En consecuencia, puede afirmarse que el sistema presenta una adecuada aceptación por parte de los usuarios, evidenciando facilidad de uso, coherencia en la interacción y cumplimiento satisfactorio de sus funciones principales. Si bien existen oportunidades de mejora, no se identifican problemas críticos que comprometan la experiencia del usuario.

3.3.2. Evaluación de funcionalidad

Los resultados correspondientes a la segunda sección del cuestionario se presentan en la Tabla 13. En la primera fila se detallan las opciones de respuesta disponibles, mientras que las filas numeradas del 11 al 15 representan las preguntas formuladas con base en los criterios por la norma ISO/IEC 25010.

Esta sección tuvo como finalidad evaluar la adecuación funcional del sistema, desde una perspectiva técnica, analizando el grado en que las funcionalidades implementadas responden a los requerimientos definidos.

Tabla 13.
Resultados de funcionalidad

	Pregunta	Totalmente en desacuerdo	En desacuerdo	Neutral	De acuerdo	Totalmente de acuerdo
11	¿El sistema cubre todas las necesidades esperadas?	0	1	6	10	15
12	¿El software Tiene validaciones necesarias para los campos de ingreso de información?	2	3	1	15	11
13	¿Las funciones disponibles facilitan la realización de tareas y objetivos planteados?	0	0	6	14	12
14	¿Existen funciones que considero innecesarias o que dificultan el uso del sistema?	7	10	9	5	1
15	¿El sistema satisface tus necesidades esperadas?	0	0	6	13	13

3.3.2.1. Análisis de los resultados

Para el análisis de los resultados obtenidos en la segunda sección del cuestionario, se calcularon promedios ponderados a partir de las respuestas emitidas por los participantes.

Debido a que algunas preguntas fueron formuladas en sentido negativo —como “¿Existen funciones que considero innecesarias o que dificultan el uso del sistema?”—, se procedió a invertir los valores asignados a dichos ítems. Este ajuste permitió unificar la dirección de la escala de medición, de modo que valores más altos representaran una percepción más favorable del sistema en todos los casos.

Para ello, se incorporó una columna adicional de puntaje ajustado, la cual sirvió como base para el cálculo del promedio final de funcionalidad. Dando como resultado los siguientes valores

Para determinar el nivel de adecuación funcional del sistema, se aplicó el cálculo del promedio ponderado mediante la siguiente ecuación:

$$x = \frac{\sum_{i=1}^n f_i * x_i}{\sum_{i=1}^n f_i}$$

Donde:

x = Promedio ponderado de la adecuación funcional

x_i = valor numérico de la opción de respuesta

f_i = frecuencia correspondiente a cada valor

n = número de categorías de respuesta

$$\sum f_i = \text{total de respuestas}$$

Para ejemplificar la aplicación de esta ecuación, se tomó como referencia la Pregunta 11, cuyos datos se detallan a continuación:

Tabla 14.
Resultados de la pregunta 11

Opción	Valor	Frecuencia
Totalmente en desacuerdo	1	0
En desacuerdo	2	1
Neutral	3	6
De acuerdo	4	10
Totalmente de acuerdo	5	15

Nota. Elaboración propia

Reemplazando los valores en la fórmula queda de la siguiente manera:

$$x = \frac{(1 * 0) + (2 * 1) + (3 * 6) + (4 * 10) + (5 * 15)}{32}$$

$$x = \frac{0 + 2 + 18 + 40 + 75}{32}$$

$$x = \frac{135}{32}$$

$$x = 4,21$$

A partir de este procedimiento, la fórmula fue aplicada a cada una de las preguntas correspondientes a la sección de adecuación funcional. Asimismo, se incorporaron los valores ajustados en los ítems formulados en sentido negativo. Como resultado, se obtuvieron los siguientes promedios finales:

Tabla 15.*Resultados de la sección de adecuación funcional*

Literal	Promedio	Puntaje ajustado
11	4,21875	4,21875
12	3,9375	3,9375
13	4,1875	4,1875
14	2,46875	2,53125
15	4,21875	4,21875

Nota. Elaboración propia

El promedio general obtenido fue de 3,81 en una escala de 1 a 5, lo que evidencia una percepción favorable respecto a la adecuación funcional del sistema. Este valor se sitúa por encima del punto medio de la escala y cercano al nivel de acuerdo, lo que indica que las funcionalidades implementadas responden adecuadamente a los requerimientos establecidos. En términos porcentuales, el sistema alcanza aproximadamente un 76% de valoración funcional, reflejando un nivel satisfactorio de cumplimiento de los criterios definidos en la norma ISO/IEC 25010.

3.3.3. Interpretación de resultados

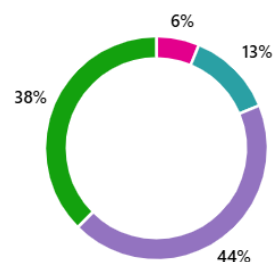
A continuación, se presenta la interpretación de los resultados obtenidos en las 32 encuestas aplicadas para evaluar la usabilidad y funcionalidad del sistema “Gestión de documentos UTN”:

Pregunta 1:

Figura 95.
Resultados de la pregunta 1

1. ¿Creo que me gustaría usar este sistema con frecuencia?

● Totalmente en desacuerdo	0
● En desacuerdo	2
● Neutral	4
● De acuerdo	14
● Totalmente de acuerdo	12



Nota. Elaboración Propia.

Análisis

El valor ajustado obtenido para esta pregunta fue de 100 puntos. Al aplicar el procedimiento de cálculo establecido por la escala SUS, el resultado corresponde a 7,81 sobre 10 en la escala proporcional.

Interpretación

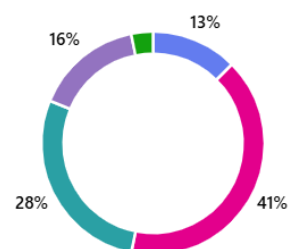
El resultado evidencia una tendencia positiva hacia el uso frecuente del sistema. La mayoría de los participantes seleccionó las opciones “De acuerdo” y “Totalmente de acuerdo”, lo que refleja una disposición favorable hacia la adopción continua de la plataforma en su entorno de trabajo. Al situarse por encima del punto medio de la escala, se confirma un nivel adecuado de aceptación en términos de intención de uso recurrente, aspecto fundamental en la evaluación de la usabilidad.

Pregunta 2:

Figura 96.
Resultados de la pregunta 2

2. ¿Me pareció que el sistema tiene un nivel de complejidad mayor al necesario?

● Totalmente en desacuerdo	4
● En desacuerdo	13
● Neutral	9
● De acuerdo	5
● Totalmente de acuerdo	1



Nota. Elaboración Propia.

Análisis

La suma de los valores ajustados para esta pregunta fue de 78 puntos. Al dividir este resultado entre los 32 participantes se obtuvo un promedio de 2,4375. Posteriormente, al aplicar el factor de conversión del método SUS ($\times 2,5$), el puntaje equivalente fue de 6,09 sobre 10.

Interpretación

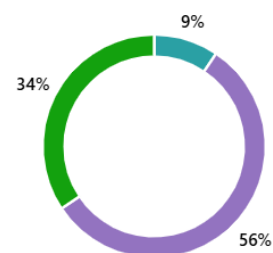
Al tratarse de un ítem formulado en sentido negativo, una puntuación más alta refleja menor percepción de complejidad. El valor obtenido (6,09) indica que la mayoría de los participantes no considera que el sistema sea excesivamente complejo. No obstante, al situarse ligeramente por encima del punto medio, sugiere que existe un pequeño grupo de usuarios que percibió cierto nivel de dificultad o sobrecarga cognitiva durante su uso.

Pregunta 3:

Figura 97.
Resultados de la pregunta 3

3. ¿Considero que el sistema fue fácil de usar?

● Totalmente en desacuerdo	0
● En desacuerdo	0
● Neutral	3
● De acuerdo	18
● Totalmente de acuerdo	11



Nota. Elaboración Propia.

Análisis

La suma de los valores ajustados para esta pregunta fue de 104 puntos. Al dividir este resultado entre los 32 participantes se obtuvo un promedio de 3,25. Posteriormente, al aplicar el factor de conversión del método SUS ($\times 2,5$), se obtuvo un puntaje equivalente a 8,12 sobre 10.

Interpretación

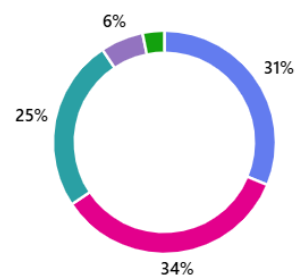
El valor obtenido evidencia una percepción altamente positiva respecto a la facilidad de uso del sistema. La mayoría de los participantes manifestó estar de acuerdo o totalmente de acuerdo con la afirmación, lo que indica que la interfaz y los procesos de interacción resultan claros e intuitivos. Este puntaje, considerablemente superior al punto medio de la escala, confirma que el sistema presenta un nivel adecuado de usabilidad en términos de simplicidad y comprensión para el usuario.

Pregunta 4:

Figura 98.
Resultados de la pregunta 4

4. ¿Creo que necesitaría ayuda de un técnico para poder usar el sistema?

● Totalmente en desacuerdo	10
● En desacuerdo	11
● Neutral	8
● De acuerdo	2
● Totalmente de acuerdo	1



Nota. Elaboración Propia.

Análisis

La suma de los valores ajustados para esta pregunta fue de 91 puntos. Al dividir este resultado entre los 32 participantes se obtuvo un promedio de 2,84. Posteriormente, al aplicar el factor de conversión del método SUS ($\times 2,5$), se obtuvo un puntaje equivalente a 7,11 sobre 10.

Interpretación

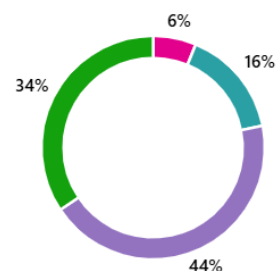
Dado que se trata de un ítem formulado en sentido negativo, una puntuación más alta indica mayor desacuerdo con la necesidad de asistencia técnica. El valor obtenido (7,11) refleja que, en términos generales, los usuarios consideran que el sistema puede utilizarse sin requerir apoyo constante de un técnico. No obstante, al situarse ligeramente por debajo de otros ítems relacionados con la facilidad de uso, sugiere que un grupo reducido de participantes percibió cierta necesidad de orientación inicial o acompañamiento puntual.

Pregunta 5:

Figura 99.
Resultados de la pregunta 5

5. ¿Considero que las funciones del sistema están bien integradas?

● Totalmente en desacuerdo	0
● En desacuerdo	2
● Neutral	5
● De acuerdo	14
● Totalmente de acuerdo	11



Nota. Elaboración Propia.

Análisis

La suma de los valores ajustados para esta pregunta fue de 98 puntos. Al dividir este resultado entre los 32 participantes se obtuvo un promedio de 3,06. Posteriormente, al aplicar el factor de conversión del método SUS ($\times 2,5$), se obtuvo una puntuación equivalente a 7,66 sobre 10.

Interpretación

El resultado obtenido refleja una percepción positiva respecto a la integración de las funciones del sistema. La mayoría de los participantes manifestó estar de acuerdo o totalmente de acuerdo con la afirmación, lo que indica que las funcionalidades son percibidas como coherentes y correctamente articuladas entre sí. Aunque no alcanza los valores más altos observados en otros ítems, el puntaje evidencia un nivel adecuado de cohesión entre los módulos del sistema, sin fragmentaciones significativas en la experiencia de uso.

Pregunta 6:

Figura 100.

Resultados de la pregunta 6

6. ¿Creo que el sistema tiene demasiadas inconsistencias?



Nota. Elaboración Propia.

Análisis

La puntuación ajustada obtenida para esta pregunta fue de 93 puntos. Al dividir este valor entre los 32 participantes se obtiene un promedio parcial de 2,91, el cual, al multiplicarse por el factor de escalado del método SUS ($\times 2,5$), equivale a 7,27 sobre 10.

Interpretación

Al tratarse de un enunciado formulado en sentido negativo, una puntuación de 7,27 indica que la mayoría de los participantes manifestó desacuerdo con la existencia de inconsistencias en el sistema. Esto sugiere que los usuarios perciben coherencia en el diseño, uniformidad en los elementos visuales y consistencia en el comportamiento funcional del sistema. No obstante, al no alcanzar valores superiores a 8, se puede inferir que aún existen aspectos menores susceptibles de mejora.

Pregunta 7:

Figura 101.

Resultados de la pregunta 7

7. ¿Considero que la mayoría de las personas aprenderían a usar el sistema rápidamente?



Nota. Elaboración Propia.

Análisis

La puntuación ajustada obtenida fue de 101 puntos. Al dividir este valor entre los 32 participantes se obtiene un promedio parcial de 3,16, el cual, al aplicarse el factor de escalado del método SUS ($\times 2,5$), equivale a 7,89 sobre 10.

Interpretación

Este resultado refleja una percepción favorable respecto a la facilidad de aprendizaje del sistema. La mayoría de los participantes manifestó estar de acuerdo o totalmente de acuerdo con la afirmación, lo que indica que el sistema presenta una curva de aprendizaje accesible. Esto sugiere que la estructura de navegación, la organización de los elementos y la lógica funcional permiten que nuevos usuarios puedan familiarizarse con el sistema en un periodo relativamente corto, favoreciendo su adopción institucional y escalabilidad.

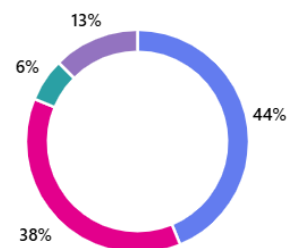
Pregunta 8:

Figura 102.

Resultados de la pregunta 8

8. ¿Tuve dificultades para utilizar el sistema debido a su complejidad?

● Totalmente en desacuerdo	14
● En desacuerdo	12
● Neutral	2
● De acuerdo	4
● Totalmente de acuerdo	0



Nota. Elaboración Propia.

Análisis

La puntuación ajustada obtenida fue de 100 puntos. Al dividir este valor entre los 32 participantes se obtiene un promedio parcial de 3,125. Al aplicar el factor de escalado del método SUS ($\times 2,5$), el resultado equivale a 7,81 sobre 10.

Interpretación

Al tratarse de un ítem formulado en sentido negativo, una puntuación de 7,81 indica un desacuerdo general con la afirmación de que la complejidad del sistema haya dificultado su uso. La mayoría de los participantes manifestó estar en desacuerdo o totalmente en desacuerdo, lo que sugiere que el sistema no presenta barreras significativas de uso. Aunque un grupo reducido expresó haber experimentado cierta dificultad, el resultado global evidencia que la complejidad percibida no representa un obstáculo relevante para la interacción con el sistema.

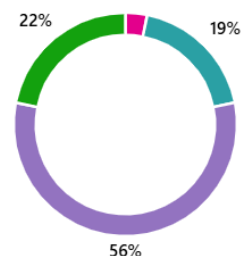
Pregunta 9:

Figura 103.

Resultados de la pregunta 9

9. ¿Me siento seguro/a al usar el sistema?

● Totalmente en desacuerdo	0
● En desacuerdo	1
● Neutral	6
● De acuerdo	18
● Totalmente de acuerdo	7



Nota. Elaboración Propia.

Análisis

La puntuación ajustada obtenida fue de 95 puntos. Al dividir este valor entre los 32 participantes se obtiene un promedio parcial de 2,97. Posteriormente, al aplicar el factor de escalado del método SUS ($\times 2,5$), el resultado equivale a 7,42 sobre 10.

Interpretación

El resultado de 7,42 evidencia una percepción favorable en cuanto a la sensación de seguridad durante el uso del sistema. La mayoría de los participantes manifestó estar de acuerdo o totalmente de acuerdo con la afirmación, lo que indica que la interacción con la plataforma genera confianza y control en el usuario. Este aspecto es relevante dentro de la usabilidad, ya que la percepción de seguridad influye directamente en la aceptación, continuidad de uso y satisfacción general del sistema.

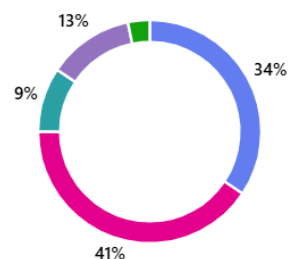
Pregunta 10:

Figura 104.

Resultados de la pregunta 10

10. ¿Necesité aprender muchas cosas antes de poder usar el sistema?

● Totalmente en desacuerdo	11
● En desacuerdo	13
● Neutral	3
● De acuerdo	4
● Totalmente de acuerdo	1



Nota. Elaboración Propia.

Análisis

La puntuación ajustada obtenida fue de 93 puntos. Al dividir este valor entre los 32 participantes se obtiene un promedio parcial de 2,91. Posteriormente, al aplicar el factor de escalado del método SUS ($\times 2,5$), el resultado equivale a 7,27 sobre 10.

Interpretación

Al tratarse de un ítem formulado en sentido negativo, un resultado de 7,27 indica un desacuerdo general con la afirmación planteada. Esto sugiere que los participantes no consideraron necesario realizar un aprendizaje complejo o prolongado para utilizar el sistema. En consecuencia, se evidencia que la curva de aprendizaje es adecuada y que el sistema presenta un nivel de accesibilidad favorable para nuevos usuarios.

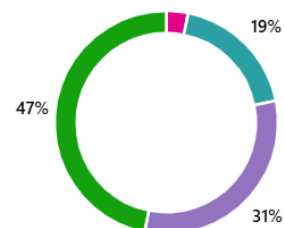
Pregunta 11:

Figura 105.

Resultados de la pregunta 11

11. El sistema cubre todas las funciones necesarias especificadas

● Totalmente en desacuerdo	0
● En desacuerdo	1
● Neutral	6
● De acuerdo	10
● Totalmente de acuerdo	15



Nota. Elaboración Propia.

Análisis

El promedio ponderado calculado para esta pregunta fue de 4,22 en una escala de 1 a 5. Este valor se sitúa claramente por encima del punto medio (3), lo que indica una tendencia predominante hacia las respuestas “De acuerdo” y “Totalmente de acuerdo”.

Interpretación

El resultado evidencia que los usuarios perciben que el sistema cumple de manera adecuada con las necesidades operativas planteadas. En términos de la norma ISO/IEC 25010, esto refleja un alto nivel de completitud funcional, indicando que las funcionalidades implementadas responden a los requerimientos establecidos para la gestión documental.

Pregunta 12:

Figura 106.

Resultados de la pregunta 12

12. El software tiene validaciones necesarias para los campos de ingreso de información.



Nota. Elaboración Propia.

Análisis

El promedio ponderado fue de 3,94, valor que también supera el punto medio de la escala. Aunque ligeramente inferior a la pregunta anterior, mantiene una tendencia favorable.

Interpretación

Este resultado indica que los usuarios consideran que el sistema incorpora validaciones adecuadas en el ingreso de datos. Desde la perspectiva de calidad funcional, esto se relaciona con la **corrección funcional y consistencia de datos**, contribuyendo a preservar la integridad del repositorio documental.

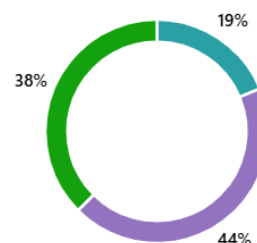
Pregunta 13:

Figura 107.

Resultados de la pregunta 13

13. Las funciones disponibles facilitan la realización de las tareas y objetivos planteados

● Totalmente en desacuerdo	0
● En desacuerdo	0
● Neutral	6
● De acuerdo	14
● Totalmente de acuerdo	12



Nota. Elaboración Propia.

Análisis

El promedio obtenido fue de 4,19, reflejando una valoración alta dentro de la escala.

Interpretación

Los participantes perciben que las funcionalidades del sistema apoyan eficazmente el cumplimiento de sus tareas. Esto demuestra una adecuada pertinencia funcional, ya que las herramientas implementadas no solo existen, sino que contribuyen de manera efectiva a los objetivos institucionales.

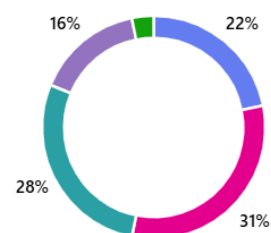
Pregunta 14:

Figura 108.

Resultados de la pregunta 14

14. Existen funciones que considero innecesarias o que dificultan el uso del sistema

● Totalmente en desacuerdo	7
● En desacuerdo	10
● Neutral	9
● De acuerdo	5
● Totalmente de acuerdo	1



Nota. Elaboración Propia.

Análisis

Al tratarse de una pregunta formulada en sentido negativo, se aplicó el ajuste correspondiente para mantener coherencia en la dirección de la escala. El puntaje ajustado obtenido fue de 2,53.

Interpretación

El resultado indica que existe una percepción moderada respecto a la presencia de funcionalidades que podrían considerarse innecesarias o mejorables. Aunque no representa un nivel crítico, sí sugiere oportunidades de optimización en ciertos elementos del sistema para mejorar su eficiencia operativa y simplificar la experiencia del usuario.

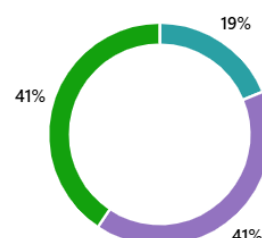
Pregunta 15:

Figura 109.

Resultados de la pregunta 15

15. El sistema satisface tus necesidades esperadas

● Totalmente en desacuerdo	0
● En desacuerdo	0
● Neutral	6
● De acuerdo	13
● Totalmente de acuerdo	13



Nota. Elaboración Propia.

Análisis

El promedio fue de 4,22, ubicándose nuevamente en un nivel alto dentro de la escala de valoración.

Interpretación

Este resultado confirma una percepción positiva general del sistema. La mayoría de los participantes considera que el sistema satisface sus expectativas, lo que respalda el cumplimiento del criterio de adecuación funcional establecido en la norma ISO/IEC 25010.

CONCLUSIONES

En el Capítulo 1 se estableció un sólido andamiaje teórico y metodológico que sustenta la presente investigación. Se revisaron en profundidad las principales metodologías de desarrollo de software, haciendo hincapié en los enfoques ágiles y en el marco de trabajo Scrum, así como en las plataformas colaborativas de gestión de código, con Bitbucket como caso ejemplar. Esta fundamentación permitió reconocer la importancia de diseñar procesos estructurados que garanticen la trazabilidad de cada artefacto del proyecto, favorezcan la planificación adaptativa e iterativa y promuevan la colaboración continua entre los miembros del equipo. De este modo, el marco conceptual aportó la base necesaria para la elaboración de una guía metodológica coherente con las mejores prácticas de la industria y alineada con los objetivos de calidad y eficiencia del desarrollo de software.

La guía metodológica propuesta en el Capítulo 2 integra de manera articulada las funcionalidades de Bitbucket con las ceremonias y roles definidos en Scrum. Se describieron tres fases claramente delimitadas —Pre-juego, Juego y Post-juego—, junto con las actividades específicas asociadas a cada una de ellas. Asimismo, se detalló el uso coordinado de herramientas de control de versiones, planificación de tareas y seguimiento de Sprints, definiendo flujos de trabajo replicables y adaptables. Esta estructura metodológica facilita a los equipos de desarrollo un recorrido ordenado, incremental y flexible, capaz de responder eficazmente a los cambios de requerimientos y de optimizar la visibilidad y el control del proyecto.

En el Capítulo 3, la aplicación de la guía metodológica permitió desarrollar el sistema “Gestión de Documentos UTN” bajo un enfoque ágil basado en Scrum, integrando Bitbucket y Jira para la planificación, el control de versiones y la trazabilidad del proyecto, lo que facilitó la organización del trabajo mediante Sprints y el seguimiento continuo del avance. La validación evidenció resultados favorables, obteniendo una puntuación de 74,45 en la escala

System Usability Scale (SUS), valor superior al promedio internacional de 68, lo que lo ubica en un nivel de usabilidad “Bueno” y refleja una adecuada aceptación por parte de los usuarios. Asimismo, la evaluación de la adecuación funcional, basada en la norma ISO/IEC 25010, alcanzó un promedio de 3,81 sobre 5 (aproximadamente 76%), indicando un cumplimiento satisfactorio de los requisitos establecidos. En conjunto, estos resultados confirman la efectividad de la guía metodológica propuesta y su contribución a mejorar la organización y la calidad del desarrollo de proyectos en el contexto académico.

RECOMENDACIONES

A modo de recomendaciones, se sugiere emplear el marco teórico desarrollado como referencia inicial para quienes se inicien en la gestión de proyectos de software, enriqueciendo continuamente su contenido con nuevos enfoques ágiles y herramientas tecnológicas emergentes.

Se recomienda, además, la incorporación de esta guía metodológica en entornos formativos —tanto académicos como profesionales—, y la realización de pruebas adicionales en diversos tipos de proyectos para ampliar su aplicabilidad, ajustar sus fases según las particularidades de cada contexto y validar su escalabilidad.

Asimismo, promover este enfoque como modelo base en las carreras vinculadas al desarrollo tecnológico fortalecerá la formación práctica de los estudiantes y su adopción de metodologías ágiles con herramientas reales del entorno profesional.

BIBLIOGRAFÍA

- Alazzawi, A., M.Yas, Q., & Rahmatullah, B. (2023). *A Comprehensive Review of Software Development Life Cycle methodologies: Pros, Cons, and Future Directions*. <https://doi.org/10.52866/ijcsm.2023.04.04.0>
- Alvarez, A. (2023). Digital transformation and risk management in projects. *Proceedings from the International Congress on Project Management and Engineering, November*. <https://doi.org/10.61547/3484>
- Alvarez, P., & Cueva, S. (2023). Desarrollo de un Marco de Gestión del ciclo de vida de un producto software utilizando modelos de innovación de procesos: Revisión de Literatura. *CEDAMAZ*, 13(1), 122–125. <https://doi.org/10.54753/cedamaz.v13i1.1261>
- Alzaid, A., & Khalfan, A. (2022). Understanding Top Management Involvement in SDLC Phases. *Journal of Software*, 87–120. <https://doi.org/10.17706/jsw.17.3.87-120>
- Arnau, L., & Sala, J. (2020). *La revisión de la literatura científica: Pautas, procedimientos y criterios de calidad*. https://ddd.uab.cat/pub/recdoc/2020/222109/revliltcie_a2020.pdf
- Asana. (2024). *Asana*. <https://asana.com/es>
- Assalaarachchi, L., Liyanage, P., & Hewagamage, C. (2024). The Need for Integrating Cloud-Based Project Management Tools in IT Curricula: Insights from Sri Lanka's Software Industry. *KDU Journal of Multidisciplinary Studies*, 6(2), 219–229. <https://doi.org/10.4038/kjms.v6i2.142>
- Bahrudin, M. (2023). Analisis Komparasi ISO 11620:2014 dan ISO 11620:2023 sebagai Standar Indikator Kinerja Perpustakaan. *Media Pustakawan*, 30(2), 171–184. <https://doi.org/10.37014/medpus.v30i2.4897>
- Bilal, M., Gani, A., Liaqat, M., Bashir, N., & Malik, N. (2020). Risk Assessment Across Life Cycle Phases for Small and Medium Software Projects. In *Journal of Engineering Science and Technology* (Vol. 15, Issue 1). <https://www.researchgate.net/publication/339089423>
- Bitbucket. (2024). *Bitbucket*. <https://Bitbucket.org/>
- Bolaños, R., & Calahorrano, C. (2023). *Guía metodológica como herramienta de la enseñanza basada en la inteligencia emocional para despertar la creatividad, en el aula en la Unidad Educativa "Daniel Reyes" 2023*. <http://repositorio.utn.edu.ec/handle/123456789/14725>
- Bravo, A. (2019). *Análisis comparativo del modelo de ciclo de vida y de dirección en ocho proyectos de diferentes sectores*. <https://oa.upm.es/65543/>
- Cabascango, E. P. (2022a). *Elaboración de una guía metodológica para la gestión de proyectos de software usando Microsoft Azure DevOps y el marco de trabajo Scrum*. <http://repositorio.utn.edu.ec/handle/123456789/13048>

- Cabascango, E. P. (2022b). *Elaboración de una guía metodológica para la gestión de proyectos de software usando Microsoft Azure DevOps y el marco de trabajo Scrum*.
- Cardona, S., & Valencia, R. (2021). *Propuesta de diseño metodológico para la gestión de proyectos en el sector de construcción, bajo la metodología del PMI dentro del grupo de procesos de inicio y planificación*. <https://repository.eafit.edu.co/server/api/core/bitstreams/9a1668d0-6c18-42c4-a5ce-310cdcdd924d/content>
- Chahal, S. (2023). Agile Methodologies for Improved Product Management. *Journal of Business and Strategic Management*, 8(4), 79–94. <https://doi.org/10.47941/jbsm.1439>
- Chakraborty, S., & Aithal, P. S. (2022). A Practical Approach To GIT Using Bitbucket, GitHub and SourceTree. *International Journal of Applied Engineering and Management Letters (IJAEML) A Refereed International Journal of Srinivas University*, 6(2), 2581–7000. <https://doi.org/10.5281/zenodo.7262771>
- Choudhury, P., Crowston, K., Dahlander, L., Minervini, M. S., & Raghuram, S. (2020). GitLab: work where you want, when you want. *Journal of Organization Design*, 9(1). <https://doi.org/10.1186/s41469-020-00087-8>
- De la Torre, A. (2023). *Guía metodológica como herramienta de aprendizaje en los niños con trastornos por déficit de atención del sexto año de básica, del CECIB “Luis Ulpiano de la Torre” de la ciudad de Otavalo, 2023*. <http://repositorio.utn.edu.ec/handle/123456789/14691>
- Delgado, L. de la C., & Díaz, L. (2020). Modelos de Desarrollo de Software. *Revista Cubana de Ciencias Informáticas*, 15(1). <http://rcci.uci.cu> Pág.37-51 Editorial "Ediciones Futuro"
- Estrada, M., Núñez, J., Saltos, P., & Cunuhay, W. (2021). *Revisión Sistemática de la Metodología Scrum para el Desarrollo de Software*. 7, 434–447. <https://doi.org/10.23857/dc.v7i4.2429>
- Estrada, M., Núñez, J., Saltos, P., & Cunuhay, W. (2021). *Revisión Sistemática de la Metodología Scrum para el Desarrollo de Software*. 7, 434–447. <https://doi.org/10.23857/dc.v7i4.2429>
- Felipe, A., & Núñez, F. de J. (2022). DevOps: un vistazo rápido. In *Publicación semestral* (Vol. 10, Issue 19). <https://repository.uaeh.edu.mx/revistas/index.php/huejutla/issue/archive>
- Flores, F., Sanhueza, V., Valdés, H., & Reyes, L. (2021). Metodologías ágiles: un análisis de los desafíos organizacionales para su implementación. *Revista Científica*, 43(1), 38–49. <https://doi.org/10.14483/23448350.18332>
- Garcia, L. A., Oliveira Jr, E., & Morandini, M. (2022). Tailoring the Scrum framework for software development: Literature mapping and feature-based support. In *Information and Software Technology* (Vol. 146). Elsevier B.V. <https://doi.org/10.1016/j.infsof.2021.106814>
- GitHub. (2024). *GitHub*. <https://github.com/>

GitLab. (2024). *GitLab*. <https://about.gitlab.com/>

Gondim do Amaral, T., Macedo da Silva, L., & Almeida Silva, V. (2023). Stein: proposta de um sistema ERP para construção civil. *Ambiente Construído*, 23(1), 197–211. <https://doi.org/10.1590/s1678-86212023000100658>

Hernández, E., & Beltrán, C. (2020). *SCRUM, Un enfoque práctico de metodología ágil para la ingeniería de software*. <https://revistas.udistrital.edu.co/index.php/tia/article/view/15702>

Huertas, R. (2024). *Desarrollo de una guía metodológica para mejorar la documentación del capítulo técnico de los trabajos de grado de la carrera de software que utilicen la metodología Scrum*. <http://repositorio.utn.edu.ec/handle/123456789/15812>

Huss, M., Herber, D. R., & Borky, J. M. (2023). An Agile Model-Based Software Engineering Approach Illustrated through the Development of a Health Technology System. *Software*, 2(2), 234–257. <https://doi.org/10.3390/software2020011>

Jaramillo Alba, J. A., Pulla Vasquez, T. E., Sarmiento Rios, S. J., & Guzmán Hidalgo, C. L. (2024). Evolución de las Metodologías de Desarrollo de Software en la Gestión Académica. *Ciencia Latina Revista Científica Multidisciplinar*, 8(4), 2468–2475. https://doi.org/10.37811/cl_rcm.v8i4.12490

Johnston, A., Kelly, S. E., Hsieh, S., Skidmore, B., & Wells, G. A. (2019). Systematic reviews of clinical practice guidelines : a methodological guide. *Journal of Clinical Epidemiology*, 108, 64–76. <https://doi.org/10.1016/j.jclinepi.2018.11.030>

Kadenic, M. D., Koumaditis, K., & Junker-Jensen, L. (2023). Mastering scrum with a focus on team maturity and key components of scrum. *Information and Software Technology*, 153. <https://doi.org/10.1016/j.infsof.2022.107079>

Kakar, A. (2023). A Rhetorical Analysis of the Agile Manifesto on its 20th Anniversary. *Journal of the Southern Association for Information Systems*, 10(1), 20–29. <https://doi.org/10.17705/3JSIS.00030>

Khan, R. A., Khan, S. U., Akbar, M. A., & Alzahrani, M. (2024). Security risks of global software development life cycle: Industry practitioner's perspective. *Journal of Software: Evolution and Process*, 36(3). <https://doi.org/10.1002/smr.2521>

Kyeremeh, K. (2019). *Overview of system development life cycle models*. <https://ssrn.com/abstract=3448536>

Makani, S. T. (2023). STREAMLINING AWS SERVERLESS DEPLOYMENTS: A BITBUCKET PIPELINE APPROACH. In *Article in International Journal of Computer Applications*. <https://iaeme.com/Home/journal/IJCA>

Merchán, N., Palma, E., & Poma, D. (2024). Comparación de metodologías ágiles para el desarrollo de software. *Journal Scientific MQRInvestigar*, 8(1), 5052–5074. <https://doi.org/10.56048/MQR20225.8.1.2024.4985-5052-5074>

- Moreno-Rodríguez, R. (2020). CASE jMDA de arquitectura dirigida por modelos para Sistemas de Información. *Revista Cubana de Ciencias Informáticas*, 14(4). [http://rcci.uci.cuPág.210-223Editorial"EdicionesFuturo"](http://rcci.uci.cuPág.210-223Editorial)
- Muñoz, C. (2023). *DevOps en el desarrollo de software: integración y entrega continua*. <https://doi.org/10.23857/pc.v8i9.6038>
- Nikitashin, M., Kaluža, M., & Werber, B. (2024). Analysis of Methodologies and Tools for Software Development in Different Architectures. *2024 47th ICT and Electronics Convention, MIPRO 2024 - Proceedings, 1999–2007*. <https://doi.org/10.1109/MIPRO60963.2024.10569545>
- Obando, B. (2025). *APLICACIÓN MÓVIL DE E-COMMERCE CON PAGO EN LÍNEA, DOBLE FACTOR DE AUTENTICACIÓN Y FACTURACIÓN AUTOMÁTICA PARA LA EMPRESA "DIGITALPC" APLICANDO LA NORMA ISO/IEC 27002:2022*. <https://repositorio.utn.edu.ec/handle/123456789/16839>
- Obando, C. M. (2021). *Diseño de una guía metodológica para la implementación de un CSIRT con la finalidad de asegurar la integridad de la información mediante el planteamiento de buenas prácticas y políticas de seguridad*. <http://repositorio.utn.edu.ec/handle/123456789/11371>
- ONU. (2022). *Objetivos de Desarrollo Sostenible Manual de referencia Sindical sobre la Agenda 2030 para el Desarrollo Sostenible*. https://www.ilo.org/wcmsp5/groups/public/@ed_dialogue/@actrav/documents/publication/wcms_569914.pdf
- Pazmiño, C. (2024). *Un gestor de tareas basado en Kanban para la planificación operativa: un enfoque ágil para la gestión y entrega continua de un producto software*. <http://bibdigital.epn.edu.ec/handle/15000/25969>
- Regulwar, G. B., Gulhane, V. S., Jawandhiya, P. M., & Tugnayat, R. M. (2021). Variations in V Model for Software Development. *International Journal of Advanced Research in Computer Science*, 1(2). <https://www.researchgate.net/publication/348488626>
- Rochina, W. (2023a). *Desarrollo de una guía metodológica para la gestión de proyectos de software con la herramienta GitHub y el marco de trabajo SCRUM*. <http://repositorio.utn.edu.ec/handle/123456789/14598>
- Rochina, W. (2023b). *Desarrollo de una guía metodológica para la gestión de proyectos de software con la herramienta GitHub y el marco de trabajo SCRUM*.
- Rochina, W. (2023c). *Desarrollo de una guía metodológica para la gestión de proyectos de software con la herramienta Github y el marco de trabajo Scrum*. <http://repositorio.utn.edu.ec/handle/123456789/14598>
- Saravanos, A., & Curinga, M. X. (2023). Simulating the Software Development Lifecycle: The Waterfall Model. *Applied System Innovation*, 6(6). <https://doi.org/10.3390/asi6060108>
- Shekar, A. (2024). *Bitbucket Server to Bitbucket Cloud migration*. <https://urn.fi/URN:NBN:fi:amk-202404025509>

- Soares, W., Gomes, H., Moreira, M., Joventino, E., & Magalhães, F. (2024). Software evaluation on infant development to support teaching and professional training. *Revista Latino-Americana de Enfermagem*, 32. <https://doi.org/10.1590/1518-8345.7248.4284>
- Soto, C. (2021). *Guía metodológica para desarrollar el liderazgo pedagógico en directivos de una institución educativa privada del cercado de Lima*.
- Syamaidzar, N., & Sofyan, H. (2024). Quality Evaluation of Academic Information Systems with ISO / IEC 25010 Standards (Case Study: ABC University). *Jurnal Informatika Dan Teknologi Informasi*, 21(2), 158–172. <https://doi.org/10.31515/telematika.v21i2>
- Tetteh, S. G. (2024). Empirical Study of Agile Software Development Methodologies: A Comparative Analysis. *Asian Journal of Research in Computer Science*, 17(5), 30–42. <https://doi.org/10.9734/ajrcos/2024/v17i5436>
- Trello. (2024). *Trello*. <https://trello.com/es>
- Universidad de Extremadura. (2022). *Guías de la biblioteca de la UEX*.
- Váscones, E. A. (2022). *Elaboración de una guía metodológica para la gestión de proyectos de software utilizando la herramienta Gitlab y la metodología Scrum para fortalecer los proyectos de desarrollo de software de los estudiantes de la carrera de Ingeniería de Software*.
- Vlachogianni, P., & Tselios, N. (2021). Perceived usability evaluation of educational technology using the System Usability Scale (SUS): A systematic review. *Journal of Research on Technology in Education*, 54(3), 392–409. <https://doi.org/10.1080/15391523.2020.1867938>
- Warkim, W., Muslim, M. H., Harvianto, F., & Utama, S. (2020). Penerapan Metode SCRUM dalam Pengembangan Sistem Informasi Layanan Kawasan. *Jurnal Teknik Informatika Dan Sistem Informasi*, 6(2). <https://doi.org/10.28932/jutisi.v6i2.2711>
- Zapata Jaramillo, C. M., & Yepes Palacio, D. E. (2024). Representation of some principles of the Agile Manifesto in the Semat Essence Kernel. *Scientia et Technica*, 29(01), 41–48. <https://doi.org/10.22517/23447214.24241>

ANEXOS

Desarrollo de la prueba de concepto

Fase 1. Pre – Juego

1.1 Inicio del proyecto

Creación del proyecto

Para llevar a cabo la creación de un nuevo proyecto en Bitbucket, primero accedemos a la plataforma con nuestras credenciales y nos dirigimos a la opción **"Projects"** en el panel de navegación.

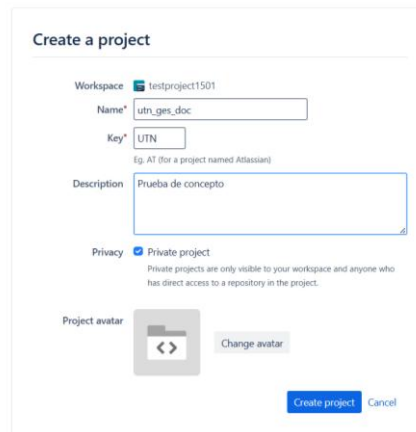
Figura 110.
Opción projects del panel de navegación



Nota. Elaboración Propia.

Procedemos a crear el proyecto asignándole el nombre **"utn_ges_doc"**, el cual servirá como identificador dentro del espacio de trabajo en Bitbucket. Durante este proceso, es importante establecer la configuración de privacidad del proyecto, por lo que se selecciona la opción **"Privado"**, asegurando así que únicamente los usuarios autorizados puedan acceder al contenido del repositorio y colaborar en su desarrollo.

Figura 111.
Formulario de creación de proyecto



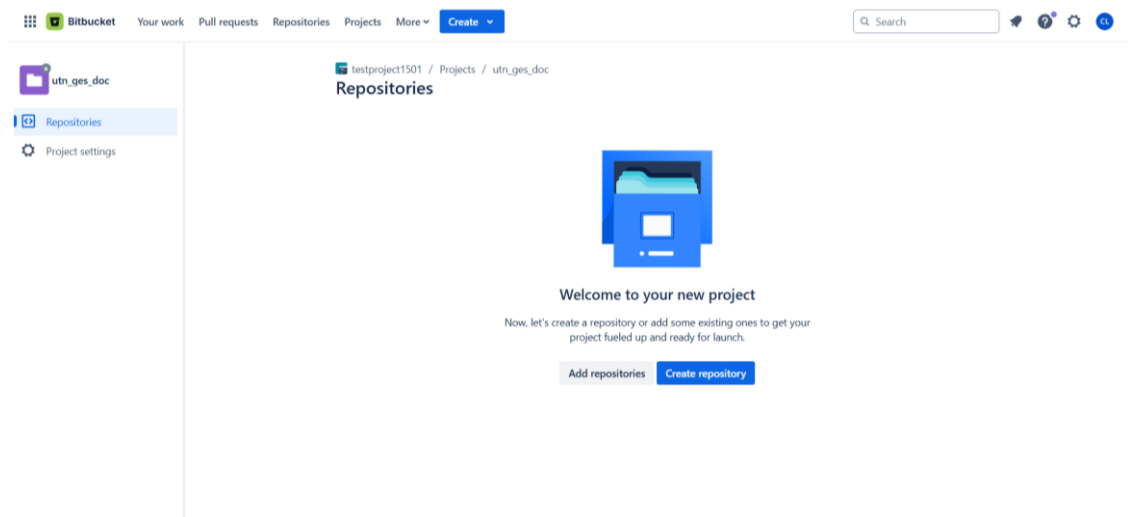
The screenshot shows the 'Create a project' form in Bitbucket. The form is titled 'Create a project' and contains the following fields and options:

- Workspace:** A dropdown menu showing 'testproject1501'.
- Name:** A text input field containing 'utn_ges_doc'.
- Key:** A text input field containing 'UTN'. Below it, a small text says 'Eg. AT (for a project named Atlassian)'.
- Description:** A text area containing 'Prueba de concepto'.
- Privacy:** A checkbox labeled 'Private project' which is checked. Below it, a small text says 'Private projects are only visible to your workspace and anyone who has direct access to a repository in the project.'
- Project avatar:** A placeholder image with a '<>' icon and a 'Change avatar' button.
- Buttons:** 'Create project' (blue) and 'Cancel' (grey).

Nota. Elaboración Propia.

Una vez creado el proyecto, procedemos a la **creación del repositorio** dentro del entorno de Bitbucket. Para ello, accedemos al proyecto previamente generado (en este caso, **utn_ges_doc**) y seleccionamos la opción **"Create repository"**.

Figura 112.
Vista con los repositorios del proyecto

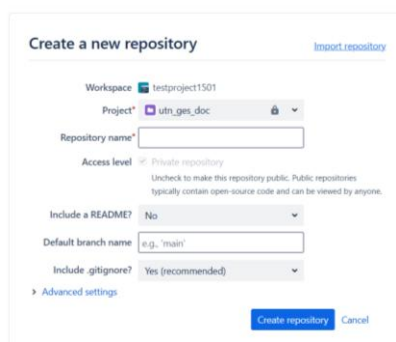


Nota. Elaboración Propia.

Durante el proceso de creación del repositorio, se debe seleccionar la opción denominada **“Import repository”**, la cual permite incorporar un proyecto existente desde una ubicación remota.

Figura 113.

Formulario de importación de un repositorio

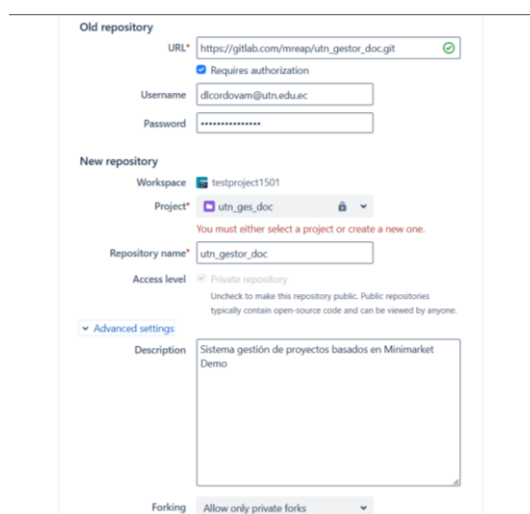


Nota. Elaboración Propia.

A continuación, se deben completar los campos solicitados, ingresando la URL del repositorio de origen, las credenciales de acceso correspondientes, el proyecto al cual estará asociado el repositorio, así como también el nombre asignado al nuevo repositorio y una descripción breve que resuma su propósito o contenido. Esta información es esencial para garantizar una importación adecuada y ordenada.

Figura 114.

Formulario avanzado de importación de un repositorio



Nota. Elaboración Propia.

Si el procedimiento de importación se ha ejecutado correctamente, el sistema mostrará una pantalla de confirmación en la cual se indica que el repositorio está siendo importado.

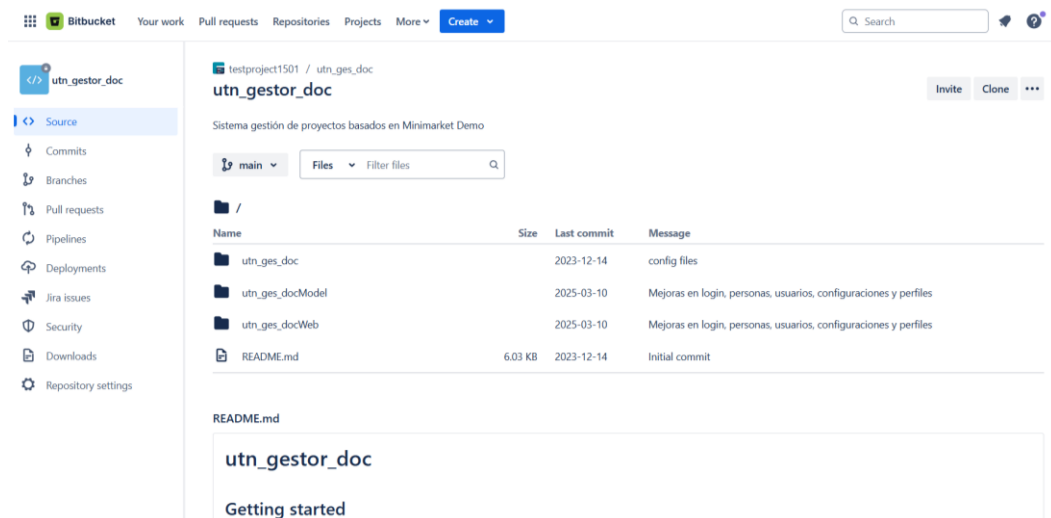
Figura 115.
Pantalla de espera de importación de repositorio



Nota. Elaboración Propia.

Una vez completado el proceso de importación, el repositorio se visualizará con su estructura y archivos correctamente integrados en la interfaz de Bitbucket.

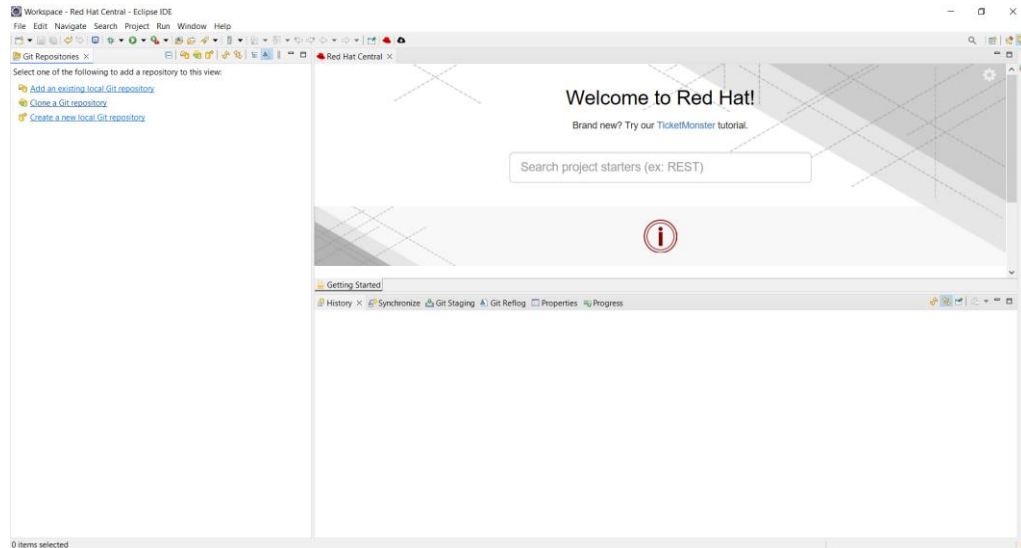
Figura 116.
Repositorio importado



Nota. Elaboración Propia.

Dentro del entorno de desarrollo Eclipse, se accede a la vista denominada **GIT**, la cual permite gestionar repositorios de control de versiones. Desde esta vista, se selecciona la opción correspondiente a la **clonación de un proyecto**, con el fin de obtener una copia local del repositorio remoto para su edición y seguimiento.

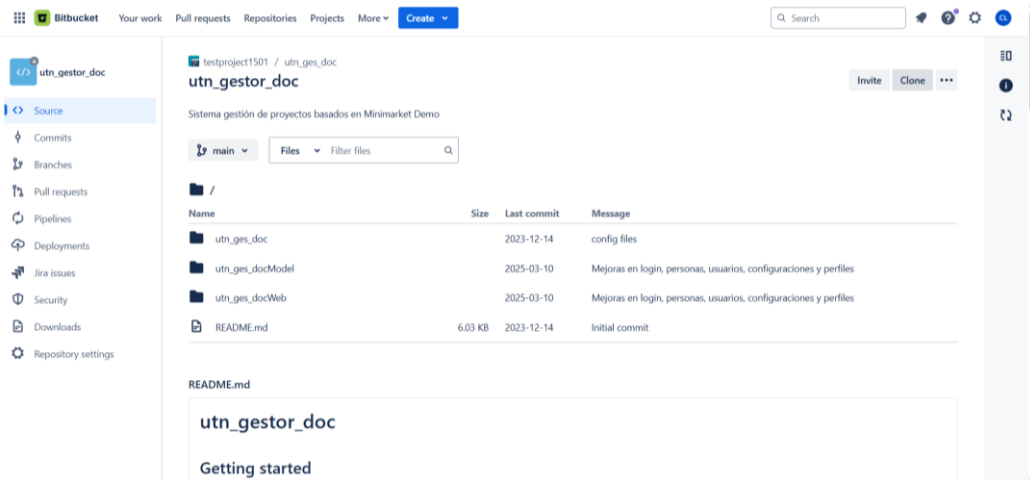
Figura 117.
Vista de GIT en eclipse



Nota. Elaboración Propia.

Nos dirigimos al repositorio correspondiente dentro de la plataforma Bitbucket y, una vez allí, seleccionamos la opción "**Clone**", la cual proporciona la URL necesaria para realizar la clonación del repositorio. Esta acción permite vincular el proyecto remoto con el entorno local de desarrollo.

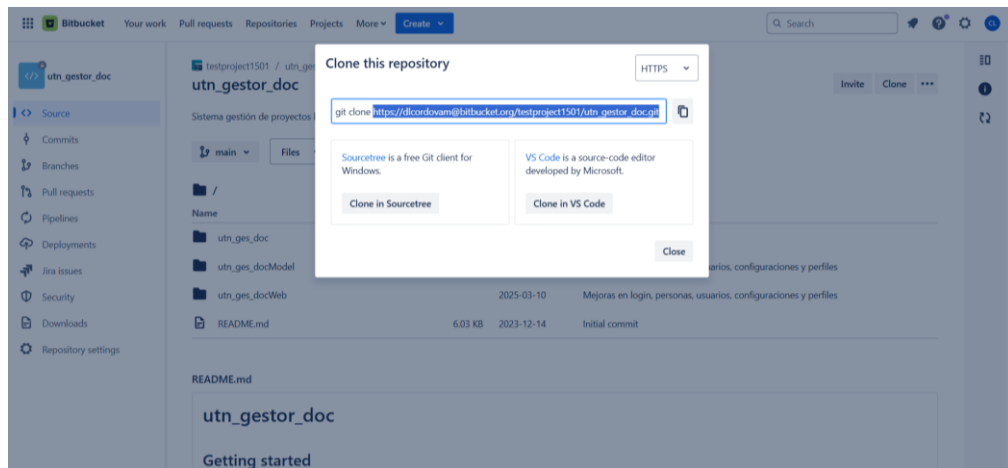
Figura 118.
Pantalla del repositorio para realizar la clonación



Nota. Elaboración Propia.

Copiamos el enlace que nos proporciona Bitbucket.

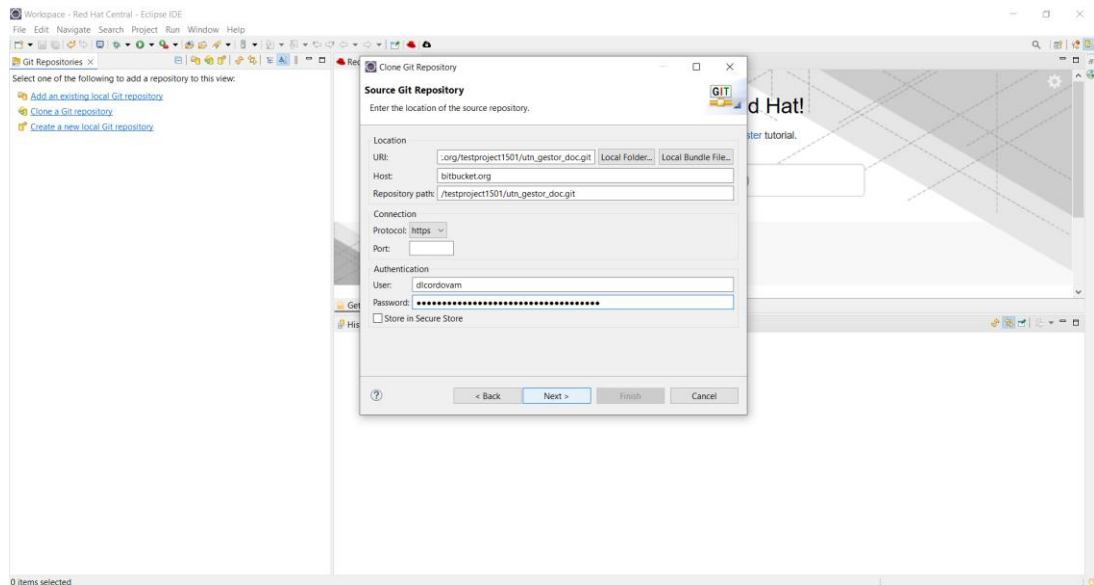
Figura 119.
Ventana para clonar el repositorio de Bitbucket



Nota. Elaboración Propia.

Utilizamos el enlace de clonación proporcionado por Bitbucket con el propósito de establecer la vinculación entre el repositorio remoto y el entorno de desarrollo Eclipse.

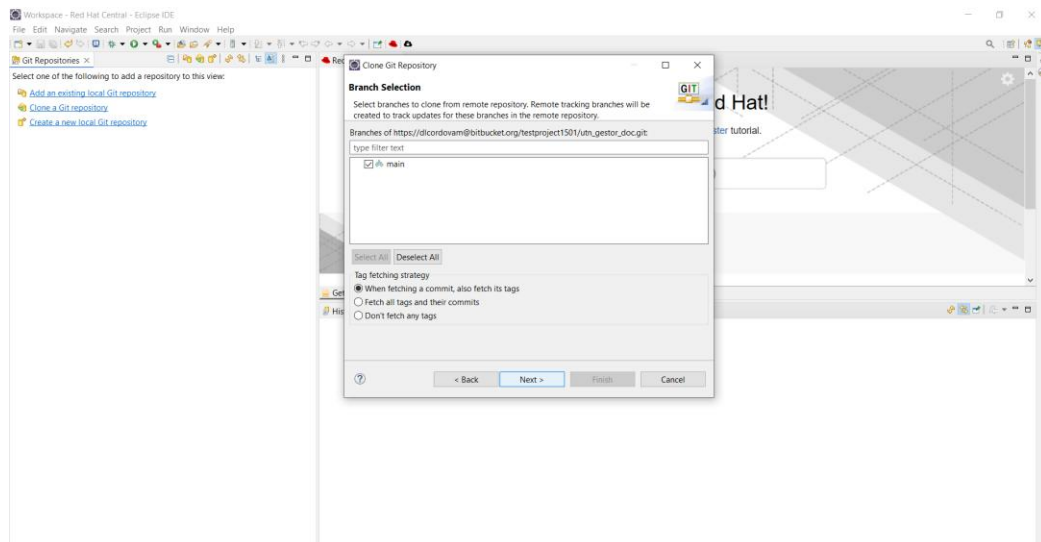
Figura 120.
Formulario de ingreso de credenciales en Eclipse



Nota. Elaboración Propia.

Seleccionamos la rama “main”.

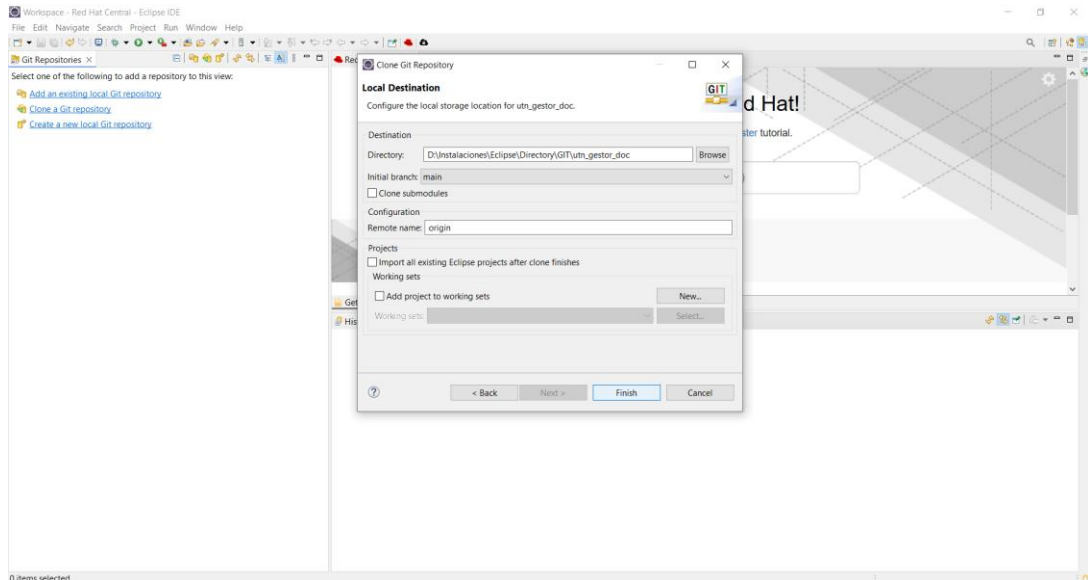
Figura 121.
Selección de la rama del proyecto



Nota. Elaboración Propia.

Procedemos a definir el directorio local en el sistema de archivos donde se almacenará el repositorio clonado.

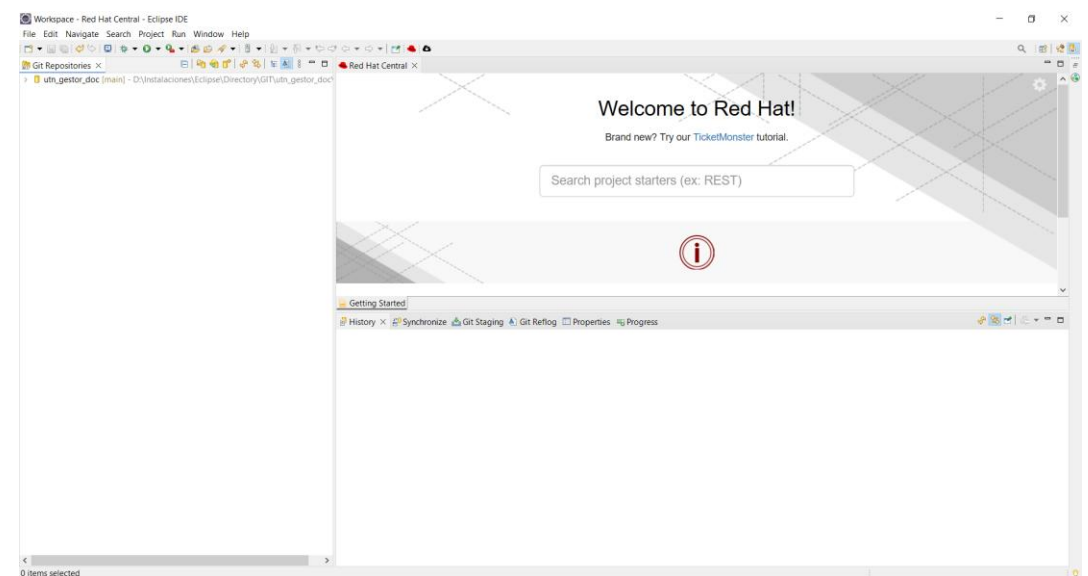
Figura 122.
Ubicación del repositorio en local



Nota. Elaboración Propia.

Una vez completado el proceso de clonación, el repositorio se presenta visualmente dentro del entorno de desarrollo Eclipse, permitiendo al usuario acceder y gestionar los archivos y carpetas que conforman el proyecto de manera integrada y eficiente.

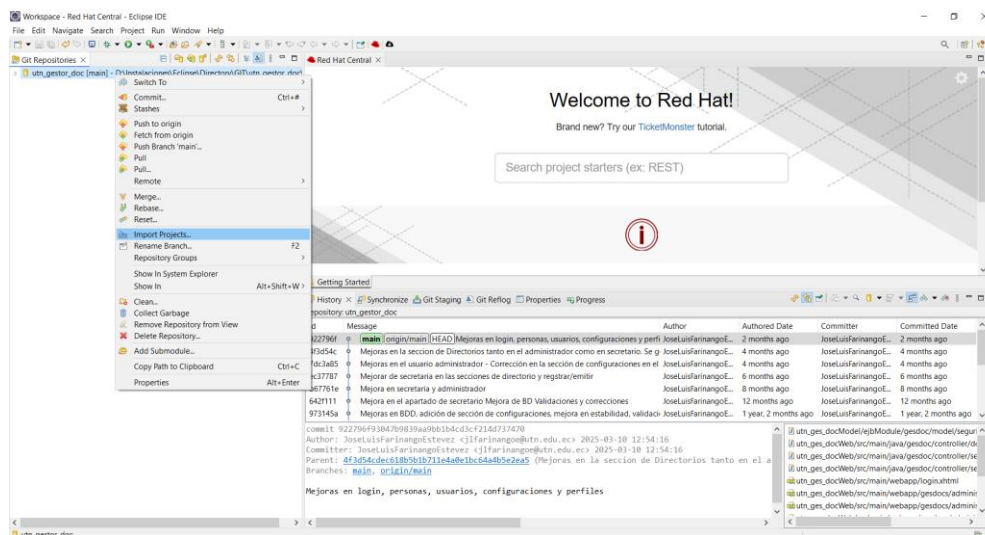
Figura 123.
Repositorios clonados dentro de eclipse



Nota. Elaboración Propia.

Dentro del entorno de Eclipse, nos dirigimos al repositorio previamente clonado y seleccionamos la opción **"Importar proyectos"**, con el fin de integrar formalmente los archivos del repositorio al espacio de trabajo.

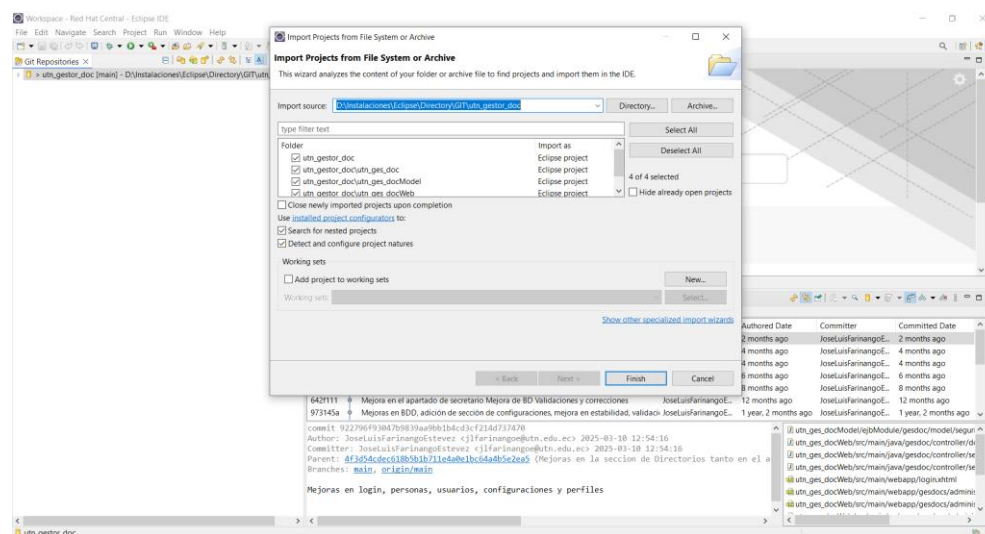
Figura 124.
Opción de importación de proyectos



Nota. Elaboración Propia.

En la ventana de importación, procedemos a seleccionar las carpetas que contienen la estructura del proyecto proveniente de Bitbucket.

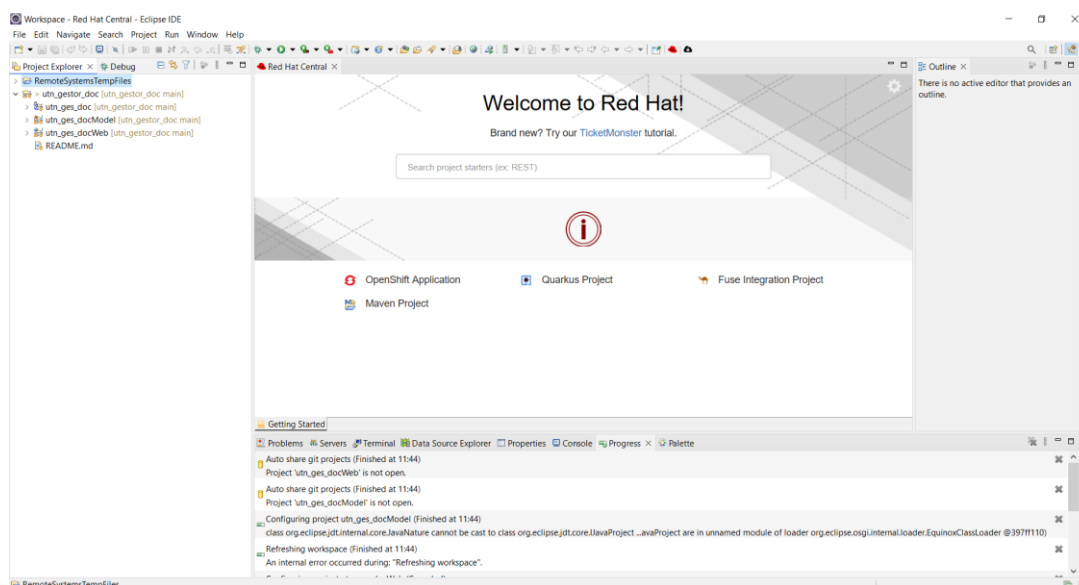
Figura 125.
Ventana de importación del proyecto



Nota. Elaboración Propia.

Una vez finalizado el proceso de importación, la estructura del proyecto debe visualizarse de manera organizada dentro del entorno de desarrollo Eclipse. Esta presentación confirma que los archivos han sido integrados correctamente y que el proyecto está listo para ser gestionado localmente.

Figura 126.
Proyecto importado en Eclipse



Nota. Elaboración Propia.

1.2 Definición de roles de Scrum

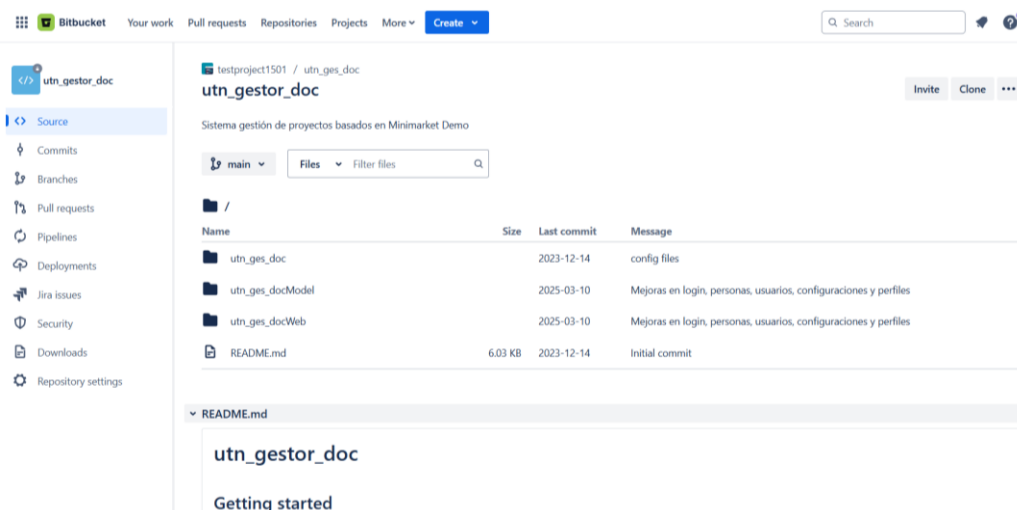
Tabla 16.
Definición de los roles de scrum

Rol	Nombre
Product Owner	MSc. Mauricio Rea
Scrum Máster	Dayana Córdova
Developer Team	Dayana Córdova

Nota. Elaboración propia

Agregar los miembros y roles al repositorio de Bitbucket

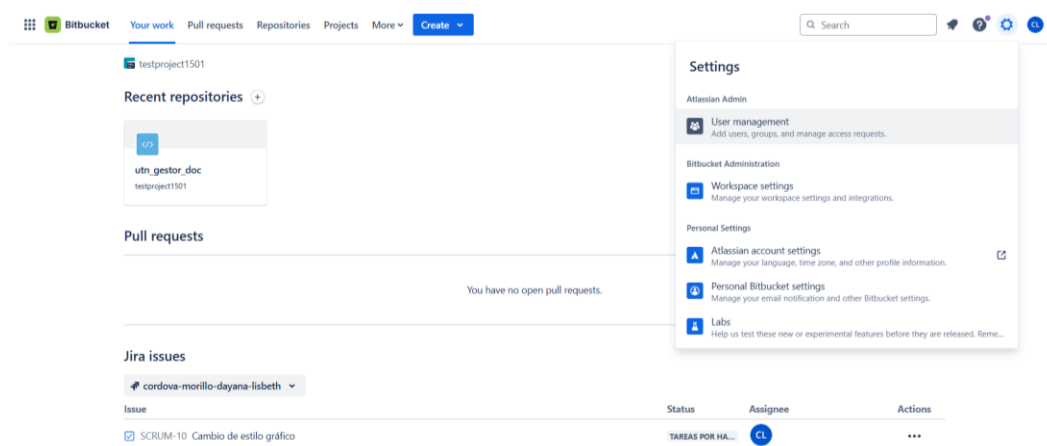
Figura 127.
Ventana de gestión de roles en Bitbucket



Nota. Elaboración Propia.

Accedemos al entorno de trabajo del repositorio en Bitbucket y nos dirigimos a la sección **"Settings"** o configuración. Dentro de este apartado, seleccionamos la opción correspondiente a la **gestión de usuarios**, desde donde es posible administrar el acceso y participación de los distintos colaboradores en el proyecto.

Figura 128.
Sección de gestión de usuarios en Bitbucket



Nota. Elaboración Propia.

En la sección de gestión de usuarios, procedemos a **ingresar las direcciones de correo electrónico** de los miembros que formarán parte del equipo.

Figura 129.
Formulario para ingresar los usuarios al proyecto

Administrador / testproject1501

Usuarios

Gestiona el acceso a productos de todos los usuarios de tu organización. [Explora la configuración de acceso](#)

[Invitar a usuarios](#) ...

Detrás de cada éxito hay un equipo
Añade a tu equipo y empezad a crear cosas increíbles juntos

Dirección de correo electrónico

Dirección de correo electrónico

Dirección de correo electrónico

[Añade a miembros del equipo](#)

Este sitio está protegido por reCAPTCHA y se aplican la [Política de privacidad](#) y las [Condiciones del servicio](#) de Google.

Introduce el nombre público o la dirección... Filtros (2)

Usuario	Última vez activo	Estado	Acciones

Nota. Elaboración Propia.

Una vez que se ha agregado correctamente un nuevo usuario al repositorio, este se visualiza en la tabla ubicada en la parte inferior de la sección de gestión de usuarios. Dicha tabla muestra el listado completo de miembros que forman parte del **workspace**, junto con sus roles y niveles de acceso correspondientes.

Figura 130.
Usuarios añadidos al proyecto

Administrador / testproject1501

Usuarios

Gestiona el acceso a productos de todos los usuarios de tu organización. [Explora la configuración de acceso](#)

[Invitar a usuarios](#) ...

Usuarios en total: 2, Usuarios activos: 2, Administradores: 1

Introduce el nombre público o la dirección... Filtros (1)

Usuario	Última vez activo	Estado	Acciones
CORDOVA MORILLO DAYANA LISBETH ADMINISTRADORES DE LA ORGANIZACIÓN dcordovaam@utn.edu.ec	15 ene 2025	Invitado	...
Mauricio Rea mrea@utn.edu.ec	21 may 2025	Invitado	...

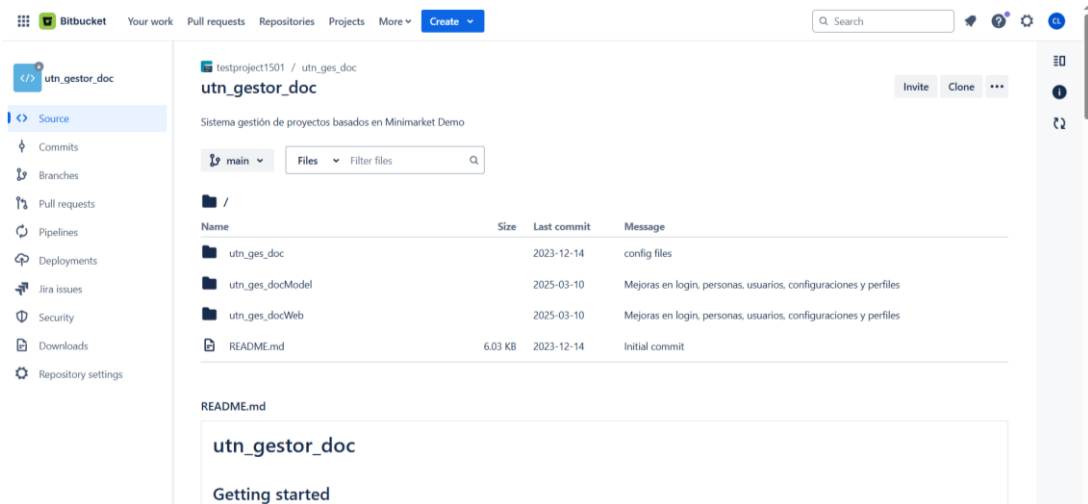
Guard

Gestión sencilla de usuarios para resolver tus retos en

Nota. Elaboración Propia.

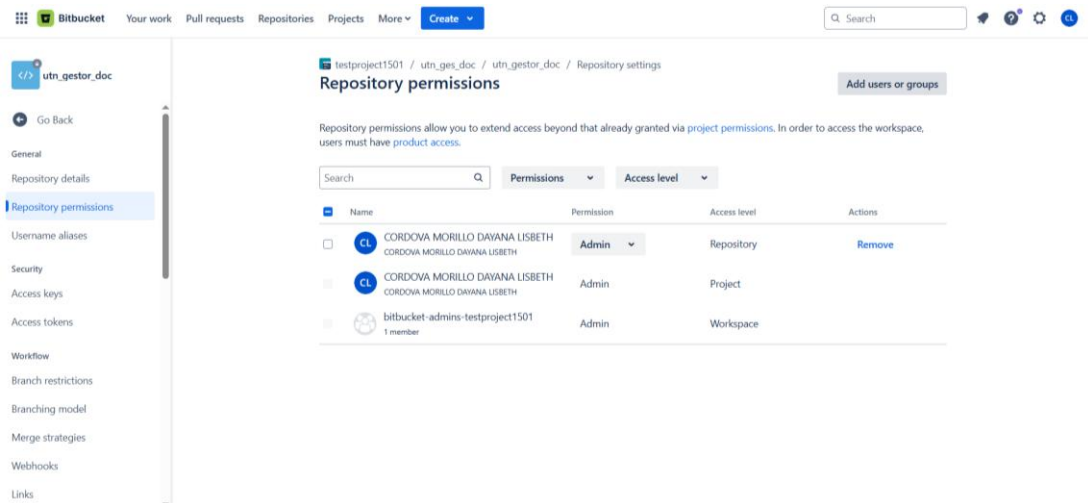
Regresamos a la vista principal del repositorio en Bitbucket y seleccionamos la opción **"Invite"**, ubicada generalmente en la parte superior o en el menú de configuración.

Figura 131.
Vista principal del repositorio de Bitbucket



Nota. Elaboración Propia.

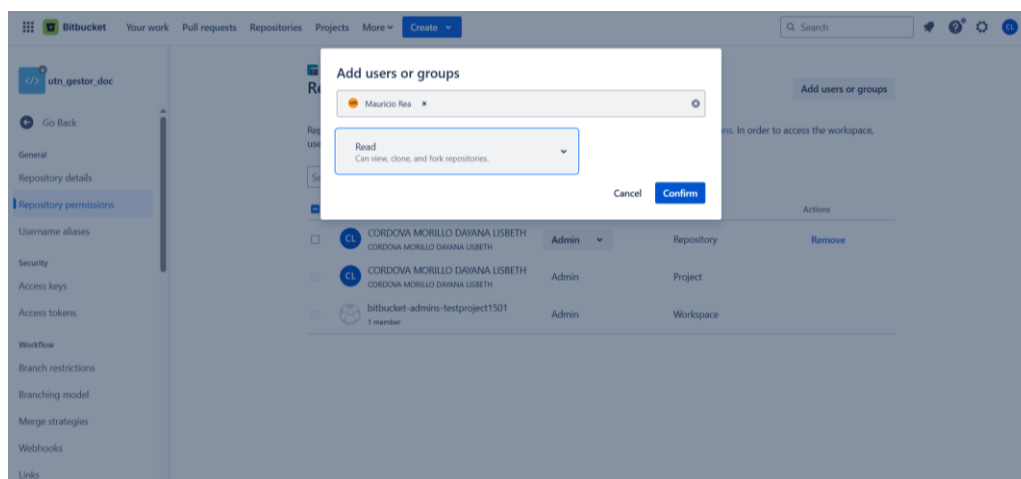
Figura 132.
Usuarios dentro del proyecto



Nota. Elaboración Propia.

Seleccionamos al miembro del equipo que deseamos incorporar al repositorio y, como parte del proceso de invitación, procedemos a asignarle los **permisos correspondientes** según su rol dentro del proyecto.

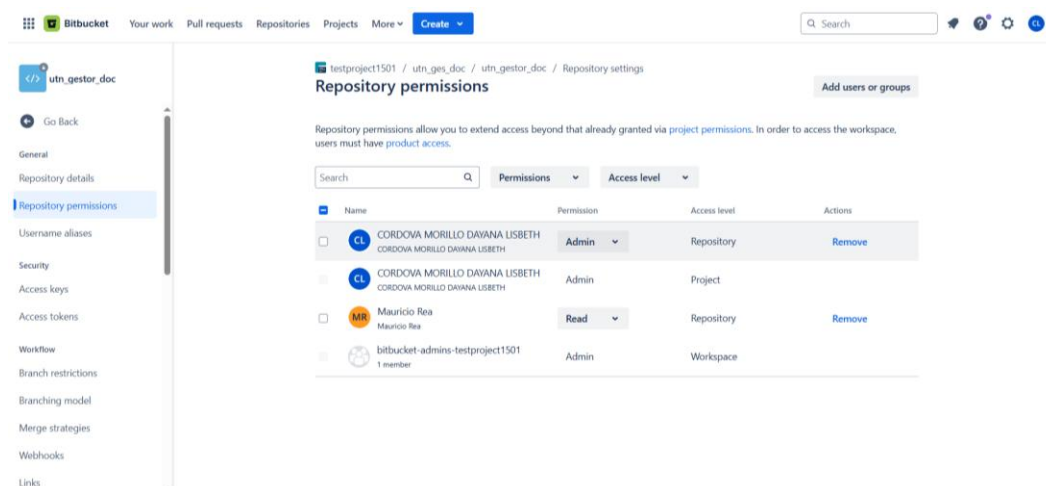
Figura 133.
Gestión de permisos del rol dentro del repositorio



Nota. Elaboración Propia.

Una vez completado el proceso de asignación, se visualiza un listado de los usuarios asociados al repositorio, junto con los **permisos específicos** que les han sido otorgados.

Figura 134.
Listado de usuarios con sus respectivos permisos



Nota. Elaboración Propia.

Definición del product backlog

Tabla 17.
Product Backlog

ID	Historia / Ítem	Tipo	Prioridad	Estimación (pts)	Estado	Notas / Criterios de Aceptación
G001	Gestión de tesis	Historia de Usuario	Alta	9	Hecho	Capacidad de visualizar, editar y crear tesis dentro del sistema
G002	Gestión de avances	Historia de Usuario	Alta	9	Hecho	Capacidad de visualizar y crear avances dentro de las tesis
G003	Gestión de resoluciones	Historia de Usuario	Alta	9	Hecho	Capacidad de visualizar, editar y crear resoluciones según el tipo de resolución dentro del sistema
G004	Gestión de docentes	Historia de Usuario	Alta	9	Hecho	Capacidad de visualizar, editar y crear docentes dentro del sistema
G005	Gestión de estudiantes	Historia de Usuario	Alta	9	Hecho	Capacidad de visualizar, editar y crear estudiantes dentro del sistema
G006	Gestión de carreras	Historia de Usuario	Alta	7	Hecho	Capacidad de visualizar, editar y crear carreras dentro del sistema
G007	Gestión de Facultades	Historia de Usuario	Alta	5	Hecho	Capacidad de visualizar, editar y crear facultades dentro del sistema
V001	Visualización de tesis	Historia de Usuario	Alta	4	Hecho	Visualización de la información de las tesis del sistema
V002	Visualización de avances	Historia de Usuario	Alta	4	Hecho	Visualización de la información de los avances del sistema
V003	Visualización de resoluciones	Historia de Usuario	Alta	4	Hecho	Visualización de la información de las resoluciones del sistema
V004	Visualización de docentes	Historia de Usuario	Alta	4	Hecho	Visualización de la información de los docentes del sistema
V005	Visualización de estudiantes	Historia de Usuario	Alta	4	Hecho	Visualización de la información de los estudiantes del sistema
V006	Visualización de carreras	Historia de Usuario	Alta	4	Hecho	Visualización de la información de las carreras del sistema

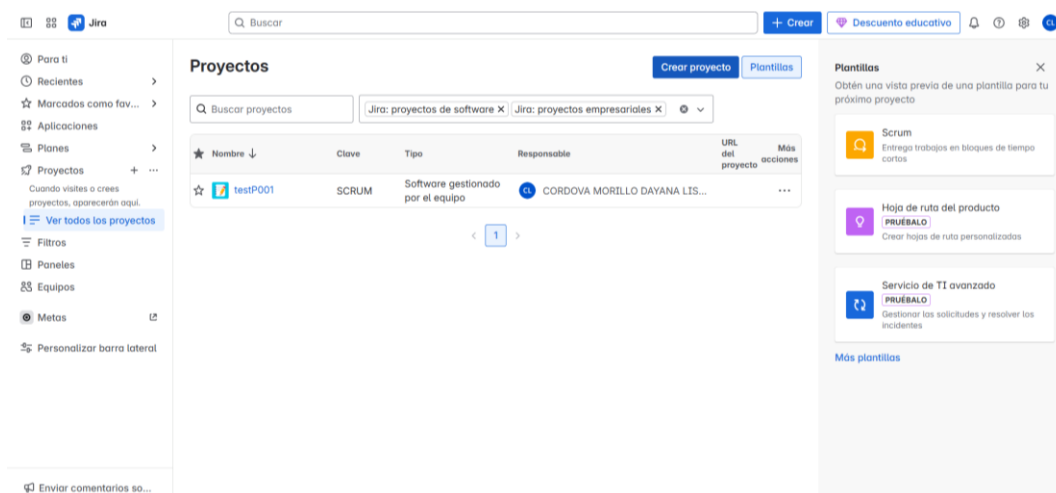
V007	Visualización de Facultades	Historia de Usuario	Alta	4	Hecho	Visualización de la información de las facultades del sistema
V008	Filtros de búsqueda	Historia de Usuario	Baja	2	Hecho	Visualización y gestión de tesis utilizando filtros de búsqueda
V009	Reportes	Historia de Usuario	Alta	7	Hecho	Generación de reportes en PDF de las tablas con la información necesaria para el sistema

Nota. Elaboración propia

Creación del proyecto en Jira

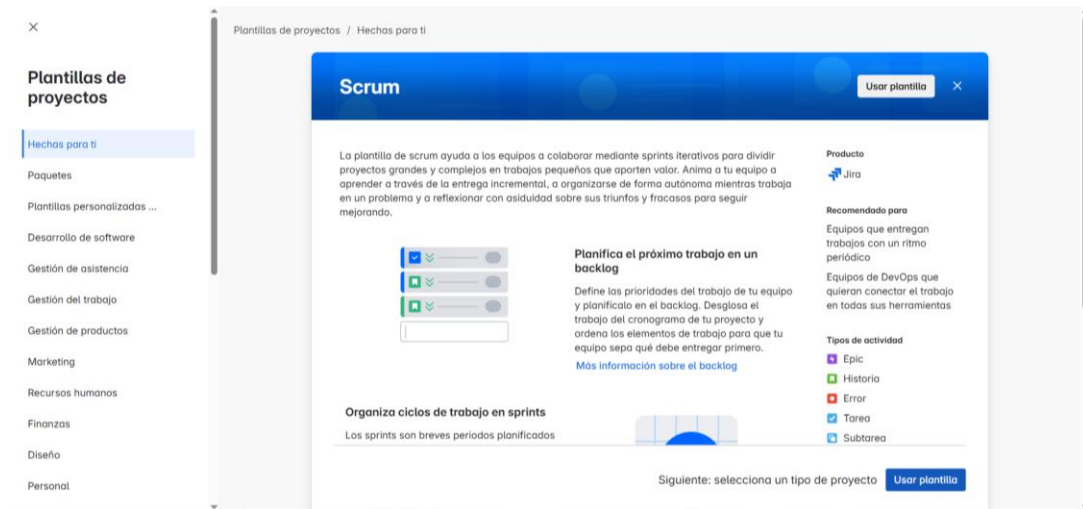
Para crear un nuevo proyecto en Jira, primero accedemos a la plataforma con nuestras credenciales y nos dirigimos al panel principal. Allí seleccionamos la opción **"Create project"**, donde se nos solicitará elegir un tipo de proyecto (por ejemplo, Scrum, Kanban o gestión de incidencias), además de ingresar un nombre y una clave identificadora para el proyecto. Finalmente, confirmamos la creación para establecer un espacio de trabajo organizado y adaptado a las necesidades del equipo.

Figura 135.
Creación del proyecto en Jira



Nota. Elaboración Propia.

Figura 136.
Selección de un tipo de proyecto



Nota. Elaboración Propia.

Figura 137.
Descripción del tipo de proyecto y selección de gestión del proyecto



Nota. Elaboración Propia.

Figura 138.
Formulario de creación del proyecto scrum en Jira

← Volver a los tipos de proyectos

Los campos obligatorios están marcados con un asterisco *

Nombre*
Systema_utm_ges_doc

Clave*
SUGD

Permisos de proyectos *

☒ Abierto | GRATUITO
Cualquiera puede editar este proyecto.

☐ Limitado | PLAN STANDARD
Elige quién puede editar este proyecto.

☐ Privado | PLAN STANDARD
Elige quién puede comentar y editar este proyecto.

Elige quién puede ver, editar y comentar
Con los permisos del proyecto, controla qué personas pueden acceder y lo que pueden hacer.
[Pruébalo gratis durante 14 días](#)

Tipo
Cambiar tipo

Gestionados por el equipo
Controla tus propios procesos y prácticas de trabajo en un espacio independiente.

Cancelar **Siguiente**

Nota. Elaboración Propia.

Con la finalización de estos pasos, el proyecto queda formalmente creado dentro de la plataforma Jira.

Figura 139.
Vista principal del proyecto en Jira

Para ti
Recientes
Marcados como fav...
Aplicaciones
Panes
Proyectos
Recientes
Systema_utm_ges_doc
Ver todos los proyectos
Filtros
Paneles
Equipos
Metas
Personalizar barra lateral

Proyectos
Systema_utm_ges_doc
Resumen Cronograma Backlog Tablero Calendario Lista Formularios Metas Todas las actividades Código More

Filtro

0 finalizadas en los últimos 7 días

0 actualizadas en los últimos 7 días

0 creadas en los últimos 7 días

0 vencen pronto en los próximos 7 días

Resumen de estado
The status overview for this project will display here after you [create some work items](#)

0 Actividades en...

Tareas por hacer: 0
En curso: 0
Finalizada: 0

Aún no hay actividad
Create a few work items and invite some teammates to your project to see your project activity.

Aún no hay actividad
Obtén una visión global de cómo se prioriza el trabajo. [Cómo gestionar las](#)

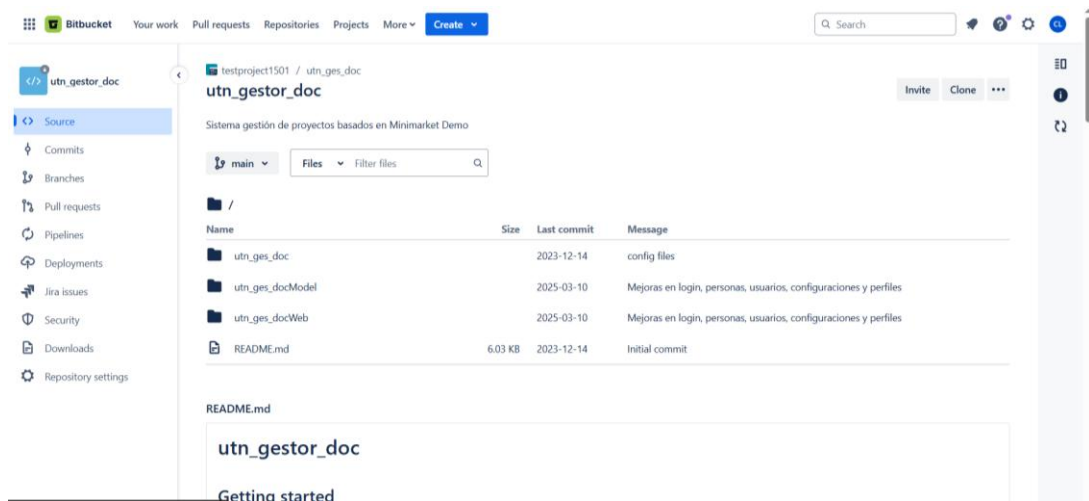
Tipos de trabajo
Create some work items to view a breakdown of total work by work type. [What](#)

Enviar comentarios so...

Nota. Elaboración Propia.

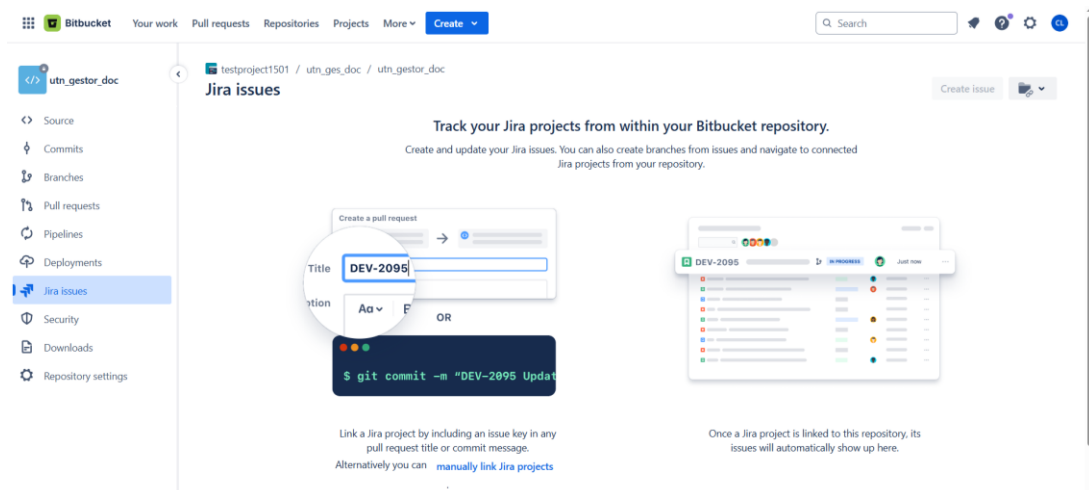
Vinculación del proyecto con Jira

Figura 140.
Proyecto en Bitbucket



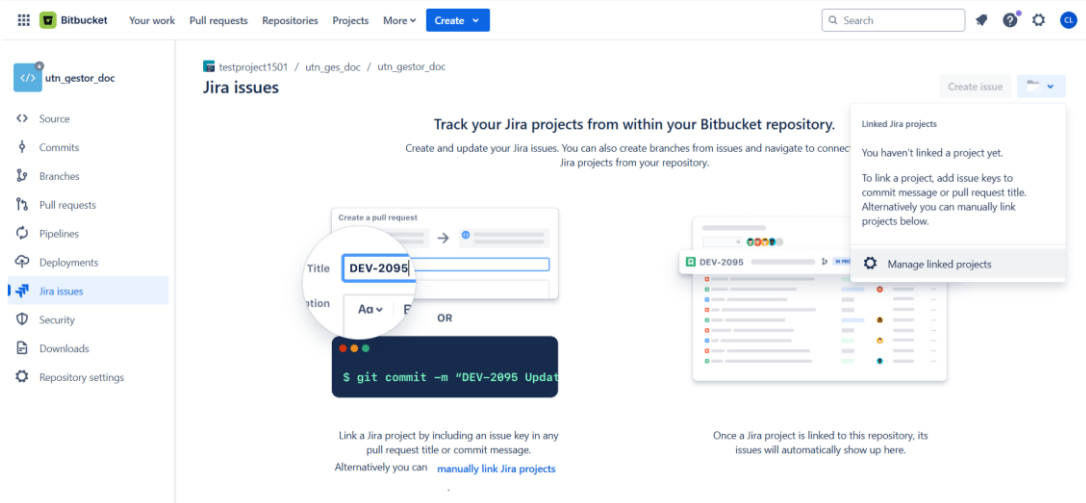
Nota. Elaboración Propia.

Figura 141.
Sección para conectar con Jira Issues



Nota. Elaboración Propia.

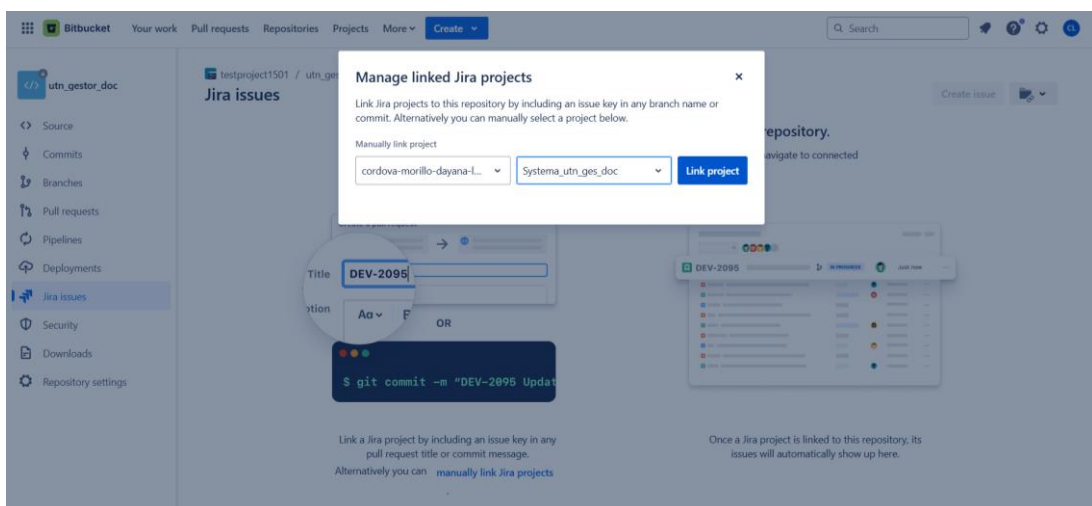
Figura 142.
Opción para gestionar proyectos vinculados



Nota. Elaboración Propia.

Procedemos a seleccionar el proyecto recientemente creado dentro de Jira y, a continuación, realizamos la vinculación con el proyecto correspondiente en la otra plataforma. Este enlace establece una conexión directa que facilita la sincronización y el seguimiento conjunto de las actividades entre ambos entornos.

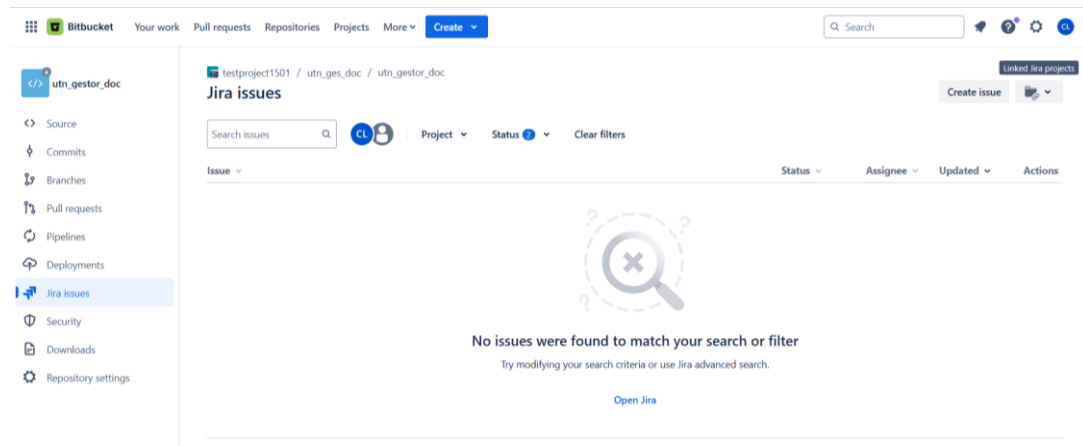
Figura 143.
Vista de gestión de proyectos vinculados con Jira



Nota. Elaboración Propia.

Una vez completado el proceso de integración, se confirma que **Jira ha sido vinculado exitosamente con Bitbucket**, lo cual permite una comunicación fluida entre ambas plataformas.

Figura 144.
Jira Issues vinculados al repositorio de Bitbucket

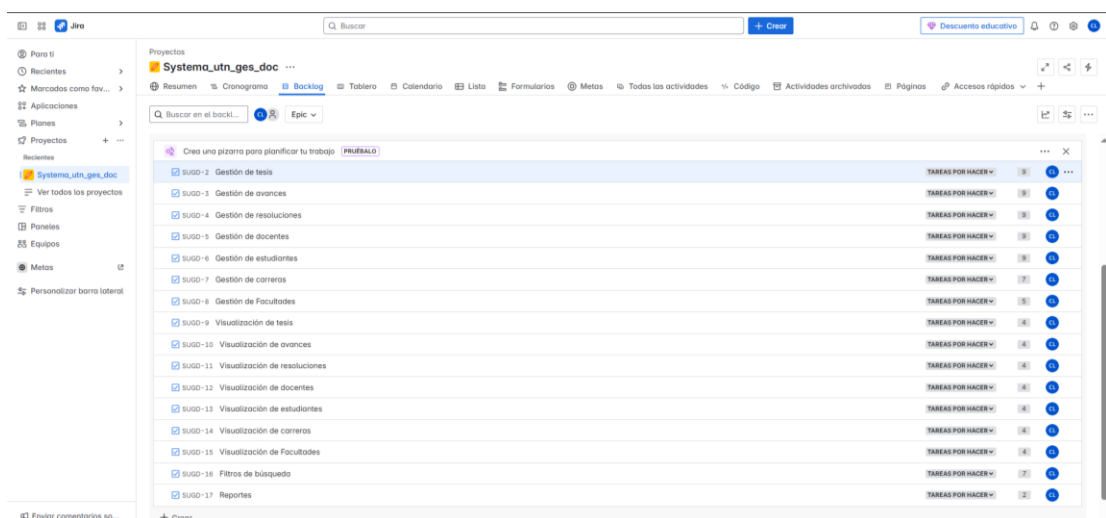


Nota. Elaboración Propia.

Creación de Issues en el Product Backlog

Los **issues** representan elementos de trabajo registrados en el **Product Backlog** dentro de Jira, tales como historias de usuario, tareas o errores. Estos se organizan y priorizan según el valor que aportan al producto, sirviendo como base para la planificación y ejecución de los Sprints en el marco de trabajo ágil.

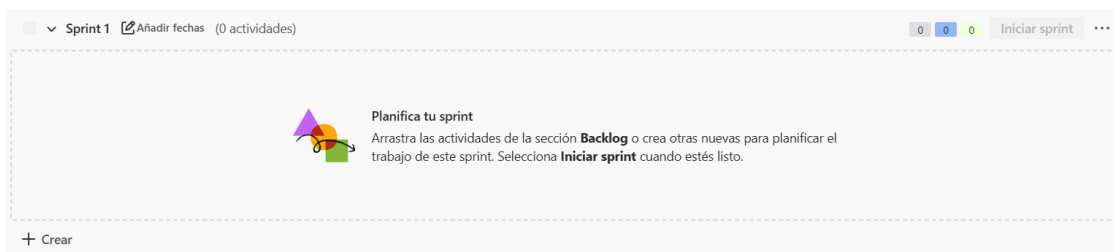
Figura 145.
Proyecto de Jira con los Issues



Nota. Elaboración Propia.

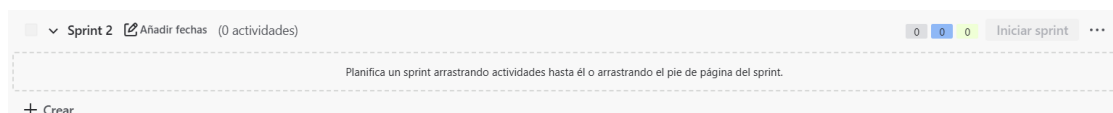
1.3 Creación de los Sprints

Figura 146.
Sección de Sprints vacía



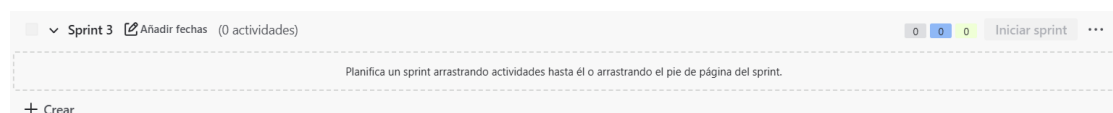
Nota. Elaboración Propia.

Figura 147.
Segundo Sprint vacío



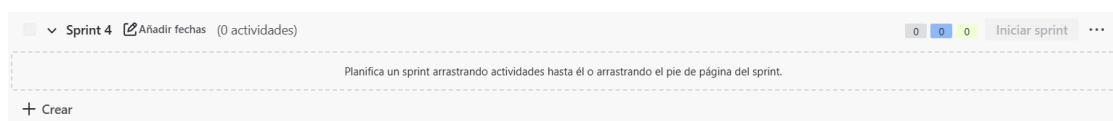
Nota. Elaboración Propia.

Figura 148.
Tercer Sprint Vacío



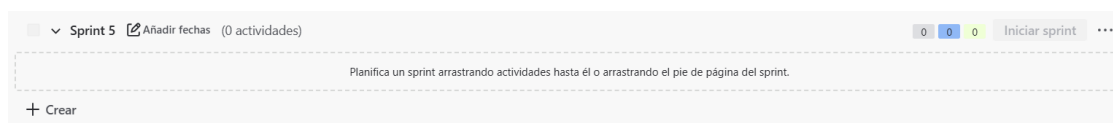
Nota. Elaboración Propia.

Figura 149.
Cuarto Sprint vacío



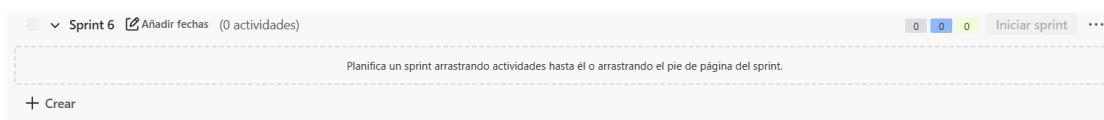
Nota. Elaboración Propia.

Figura 150.
Quinto Sprint vacío



Nota. Elaboración Propia.

Figura 151.
Sexto Sprint vacío



Nota. Elaboración Propia.

Asignación de tareas a los Sprints

Figura 152.
Issues en el primer tablero Sprint

☐ Tablero Sprint 1 Añadir fechas (4 actividades) 34 0 0 Iniciar sprint ...			
☒	SUGD-2	Gestión de tesis	TAREAS POR HACER 9 CL
☒	SUGD-3	Gestión de avances	TAREAS POR HACER 9 CL
☒	SUGD-4	Gestión de resoluciones	TAREAS POR HACER 9 CL
☒	SUGD-16	Filtros de búsqueda	TAREAS POR HACER 7 CL
+ Crear			

Nota. Elaboración Propia.

Figura 153.
Issues en el segundo tablero Sprint

☐ Tablero Sprint 2 Añadir fechas (4 actividades) 30 0 0 Iniciar sprint ...			
☒	SUGD-5	Gestión de docentes	TAREAS POR HACER 9 CL
☒	SUGD-6	Gestión de estudiantes	TAREAS POR HACER 9 CL
☒	SUGD-7	Gestión de carreras	TAREAS POR HACER 7 CL
☒	SUGD-8	Gestión de Facultades	TAREAS POR HACER 5 CL
+ Crear			

Nota. Elaboración Propia.

Figura 154.
Issues en el tercer tablero Sprint

☐ Tablero Sprint 3	Añadir fechas (1 actividad)	4	0	0	Iniciar sprint	...
☒ SUGD-9	Visualización de tesis	TAREAS POR HACER	4	CL		
+ Crear						

Nota. Elaboración Propia.

Figura 155.
Issues en el cuarto tablero Sprint

☐ Tablero Sprint 4	Añadir fechas (3 actividades)	10	0	0	Iniciar sprint	...
☒ SUGD-10	Visualización de avances	TAREAS POR HACER	4	CL		
☒ SUGD-11	Visualización de resoluciones	TAREAS POR HACER	4	CL		
☒ SUGD-17	Reportes	TAREAS POR HACER	2	CL		
+ Crear						

Nota. Elaboración Propia.

Figura 156.
Issues en el quinto tablero Sprint

☐ Tablero Sprint 5	Añadir fechas (2 actividades)	8	0	0	Iniciar sprint	...
☒ SUGD-12	Visualización de docentes	TAREAS POR HACER	4	CL		
☒ SUGD-13	Visualización de estudiantes	TAREAS POR HACER	4	CL		
+ Crear						

Nota. Elaboración Propia.

Figura 157.
Issues en el quinto tablero Sprint

☐ Tablero Sprint 6	Añadir fechas (2 actividades)	8	0	0	Iniciar sprint	...
☒ SUGD-14	Visualización de carreras	TAREAS POR HACER	4	CL		
☒ SUGD-15	Visualización de Facultades	TAREAS POR HACER	4	CL		
+ Crear						

Nota. Elaboración Propia.

Definición de historias de usuario

Tabla 18.

Historia de usuario Gestión de tesis

Código: G001	Usuario: Secretario
Nombre historia: Gestión de tesis	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 1
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero ver, crear y manipular la información de las tesis como:	
- Título - Nombre del estudiante que desarrolla la tesis - Cédula del estudiante - Descripción de la tesis - Información del Director - Información del Asesor 1 - Información del Asesor 2 - Si el estudiante está matriculado - Si el estudiante está retirado - Fecha de presentación del anteproyecto - Nombre de la empresa auspiciante - Carrera a la que pertenece la tesis - Estado del tema - Historial de avance del tema	
Observación:	
Pueden mostrarse y ocultarse columnas en la vista	

Nota. Elaboración propia

Tabla 19.

Historia de usuario Gestión de avances

Código:	Usuario: Secretario
Nombre historia: Gestión de avances	

Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 1
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero ver, crear y manipular la información de los avances de la tesis como:	
- Porcentaje de avance - Fecha de avance - Descripción del avance	
Observación:	
De ser posible mostrar los avances dependiendo en la vista de la tesis	
Nota. Elaboración propia	

Tabla 20.
Historia de usuario Gestión de resoluciones

Código:	Usuario: Secretario
Nombre historia: Gestión de resoluciones	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 1
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero ver, crear y manipular la información de las diferentes resoluciones como:	
- Número de Resolución - Fecha de resolución - Tesis que se aprobaron en esa resolución - Tipo de resolución (Comisión asesora o HCD)	
Observación:	

Se pueden mostrar las sesiones de la comisión asesora y HCD en la misma tabla

Nota. Elaboración propia

Tabla 21.

Historia de usuario Gestión de docentes

Código:	Usuario: Secretario
Nombre historia: Gestión de docentes	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 2
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero ver, crear y manipular la información de los docentes como:	
- Nombres del docente - Apellidos del docente - Cédula del docente - Carreras a las que imparte clases - Visualizar tesis en las que está trabajando como director	
Observación:	
Únicamente se mostrarán las tesis en las que el docente es director	

Nota Elaboración propia

Tabla 22.

Historia de usuario Gestión de estudiantes

Código:	Usuario: Secretario
Nombre historia: Gestión de estudiantes	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:

Programador responsable: Dayana Córdova

Descripción:

Como usuario quiero ver, crear y manipular la información de los estudiantes como:

- Nombres del estudiante
- Apellidos del estudiante
- Cédula del estudiante
- Carrera en la que está el estudiante

Observación:

En la vista se mostrará la carrera y la facultad en la que está el estudiante

Nota. Elaboración propia

Tabla 23.

Historia de usuario Gestión de carreras

Código:	Usuario: Secretario, Coordinador
Nombre historia: Gestión de carreras	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 2
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero ver, crear y manipular la información de las carreras como:	
- Nombre de la carrera - Facultad de la carrera	
Observación:	
Ninguna	

Nota. Elaboración propia

Tabla 24.*Historia de usuario Gestión de facultades*

Código:	Usuario: Secretario, Coordinador
Nombre historia: Gestión de facultades	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 2
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero ver, crear y manipular la información de las facultades como:	
- Nombre de la facultad	
Observación:	
Ninguna	

Nota. Elaboración propia

Tabla 25.*Historia de usuario Visualización de tesis*

Código: G001	Usuario: Coordinador
Nombre historia: Visualización de tesis	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 3
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero visualizar la información de las tesis como:	
- Título - Nombre del estudiante que desarrolla la tesis	

-
- Cédula del estudiante
 - Descripción de la tesis
 - Información del Director
 - Información del Asesor 1
 - Información del Asesor 2
 - Si el estudiante está matriculado
 - Si el estudiante está retirado
 - Fecha de presentación del anteproyecto
 - Nombre de la empresa auspiciante
 - Carrera a la que pertenece la tesis
 - Estado del tema
 - Historial de avance del tema

Observación:

Pueden mostrarse y ocultarse columnas en la vista

Nota. Elaboración propia

Tabla 26.

Historia de usuario Visualización de avances

Código:	Usuario: Coordinador		
Nombre historia: Visualización de avances			
Prioridad en negocio:		Riesgo de desarrollo:	
Alta		Alta	
Puntos estimados:		Integración asignada:	
9		Sprint 4	
Programador responsable: Dayana Córdova			
Descripción:			
Como usuario quiero visualizar la información de los avances de la tesis como:			
- Porcentaje de avance - Fecha de avance - Descripción del avance			

Observación:

De ser posible mostrar los avances dependiendo en la vista de la tesis

Nota. Elaboración propia

Tabla 27.*Historia de usuario Visualización de resoluciones*

Código:	Usuario: Coordinador
Nombre historia: Visualización de resoluciones	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 4
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero visualizar la información de las diferentes resoluciones como:	
- Número de Resolución - Fecha de resolución - Tesis que se aprobaron en esa resolución - Tipo de resolución (Comisión asesora o HCD)	
Observación:	
Se pueden mostrar las sesiones de la comisión asesora y HCD en la misma tabla	

Nota. Elaboración propia

Tabla 28.*Historia de usuario Visualización de docentes*

Código:	Usuario: Coordinador
Nombre historia: Visualización de docentes	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 5
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero visualizar la información de los docentes como:	

-
- Nombres del docente
 - Apellidos del docente
 - Cédula del docente
 - Carreras a las que imparte clases
 - Visualizar tesis en las que está trabajando como director

Observación:

Únicamente se mostrarán las tesis en las que el docente es director

Nota. Elaboración propia

Tabla 29.

Historia de usuarios Visualización de estudiantes

Código:	Usuario: Coordinador
Nombre historia: Visualización de estudiantes	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 5
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero visualizar la información de los estudiantes como:	
- Nombres del estudiante - Apellidos del estudiante - Cédula del estudiante - Carrera en la que está el estudiante	
Observación:	
En la vista se mostrará la carrera y la facultad en la que está el estudiante	

Nota. Elaboración propia

Tabla 30.

Historia de usuario Visualización de carreras

Código:	Usuario: Coordinador, Coordinador
Nombre historia: Visualización de carreras	

Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 6
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero visualizar la información de las carreras como:	
- Nombre de la carrera - Facultad de la carrera	
Observación:	
Ninguna	

Nota. Elaboración propia

Tabla 31.
Historia de usuario Visualización de facultades

Código:	Usuario: Coordinador, Coordinador
Nombre historia: Visualización de facultades	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
9	Sprint 6
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero visualizar la información de las facultades como:	
- Nombre de la facultad	
Observación:	
Ninguna	

Nota. [Elaboración propia

Tabla 32.*Historia de usuario Filtros de búsqueda*

Código:	Usuario: Secretario, Coordinador
Nombre historia: Filtros de búsqueda	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Baja
Puntos estimados:	Integración asignada:
	Sprint 1
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero poder filtrar los resultados en las tablas según criterios de búsqueda.	
Observación:	
Se pueden filtrar la información dentro de las tablas.	

Nota. Elaboración propia

Tabla 33.*Historia de usuario Reportes*

Código:	Usuario: Secretario, Coordinador
Nombre historia: Reportes	
Prioridad en negocio:	Riesgo de desarrollo:
Alta	Alta
Puntos estimados:	Integración asignada:
7	Sprint 4
Programador responsable: Dayana Córdova	
Descripción:	
Como usuario quiero poder realizar reportes utilizando las tablas y los filtros	

Observación:

El reporte debe ser en formato .pdf

Nota. Elaboración propia

Definición de actividades para los Issues

Issue #1

Figura 158.
Issue de gestión de tesis

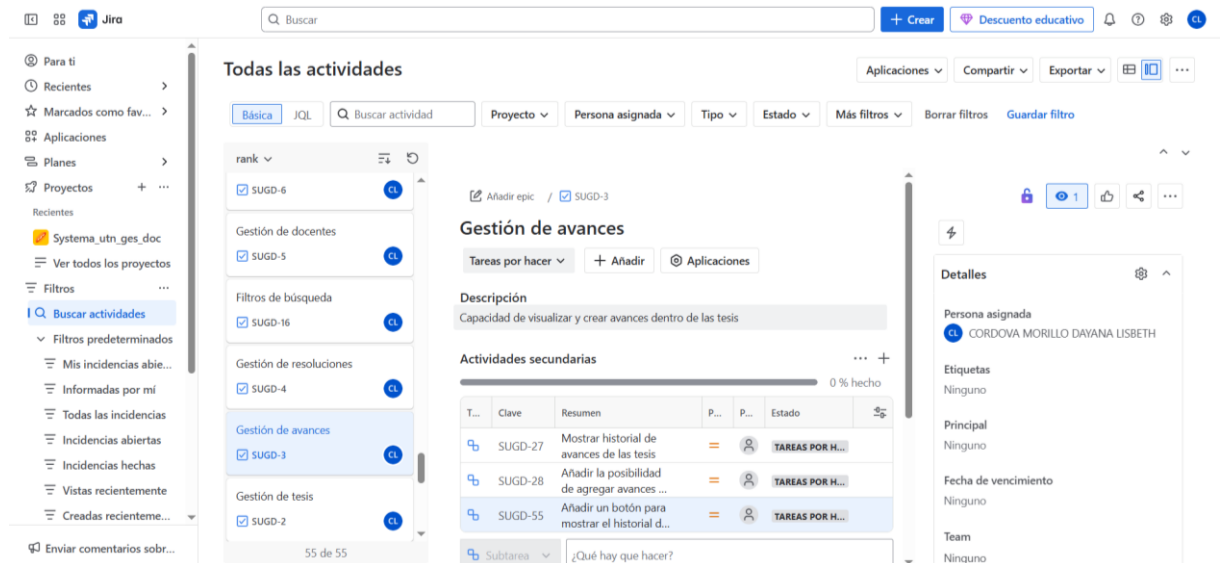
The screenshot displays the Jira web interface for an issue titled 'Gestión de tesis'. The interface is divided into several sections:

- Left Sidebar:** Contains navigation links such as 'Para ti', 'Recientes', 'Marcados como fav...', 'Aplicaciones', 'Planes', 'Proyectos', and 'Filtros'. The 'Filtros' section is expanded, showing 'Buscar actividades' and 'Filtros predeterminados'.
- Top Bar:** Includes a search bar, a '+ Crear' button, and a 'Descuento educativo' button.
- Main Content Area:**
 - Todas las actividades:** A section with filters for 'Básica', 'JQL', and 'Buscar actividad'. It includes dropdowns for 'Proyecto', 'Persona asignada', 'Tipo', and 'Estado'. A 'Guardar filtro' button is also present.
 - Actividades secundarias:** A table listing activities with columns for 'T...', 'Clave', 'Resumen', 'P...', 'P...', and 'Estado'. The table shows three activities: SUGD-24, SUGD-25, and SUGD-26, all with a status of 'TAREAS POR H...'.
- Right Sidebar:** Contains details for the issue, including 'Persona asignada' (CORDOVA MORILLO DAYANA LISBETH), 'Etiquetas' (Ninguno), 'Principal' (Ninguno), 'Fecha de vencimiento' (Ninguno), and 'Team' (Ninguno).

Nota. Elaboración Propia.

Issue #2

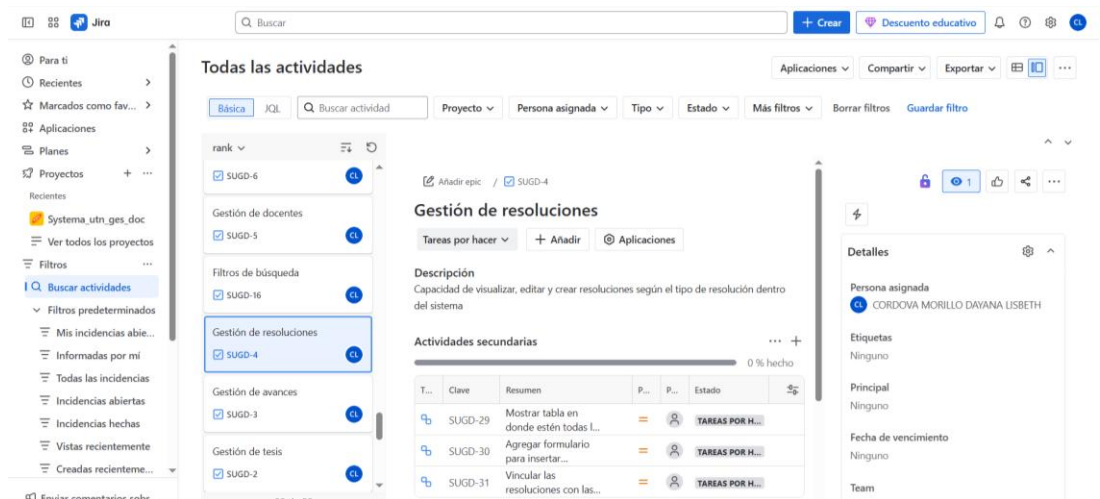
Figura 159.
Issue de Gestión de avances



Nota. Elaboración Propia.

Issue #3

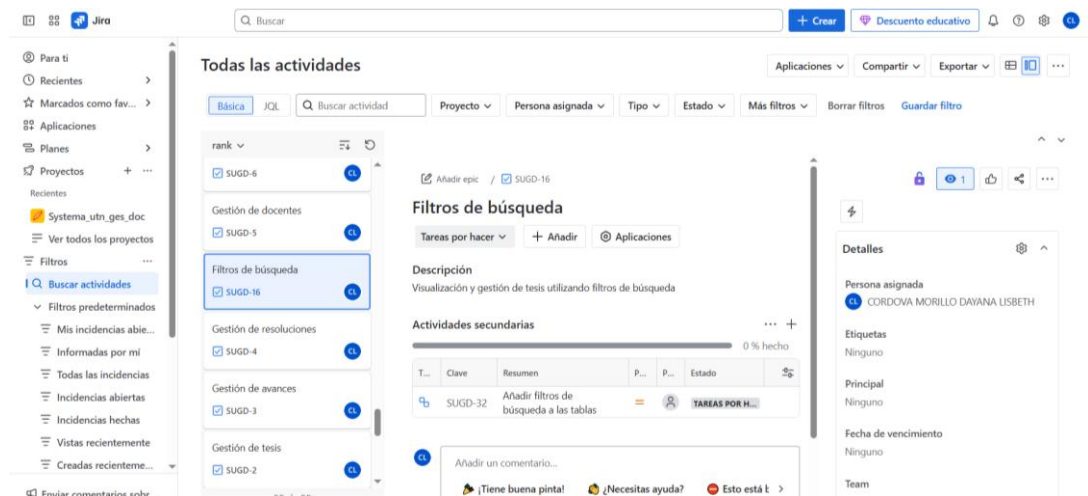
Figura 160.
Issue de gestión de resoluciones



Nota. Elaboración Propia.

Issue #4

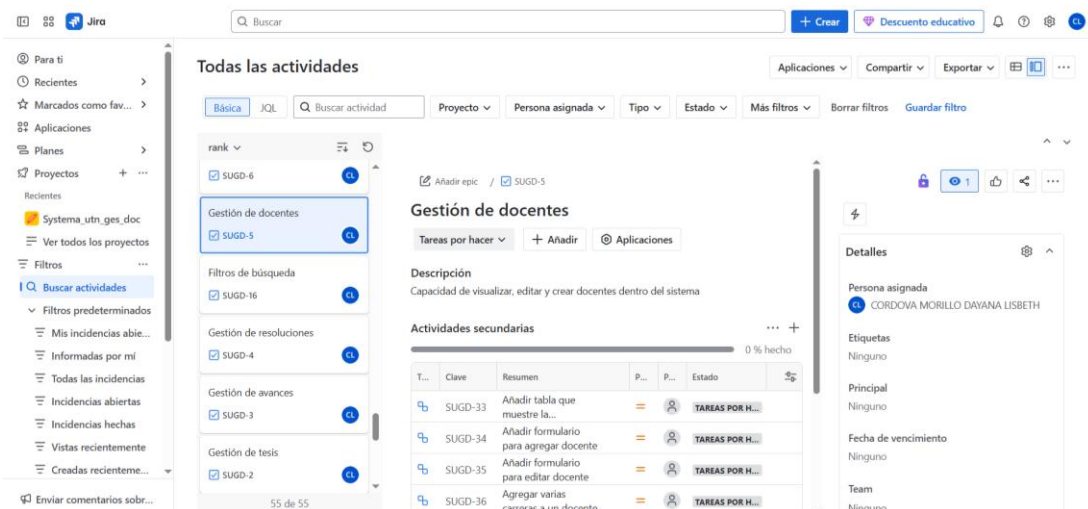
Figura 161.
Issue de filtros de búsquedas



Nota. Elaboración Propia.

Issue #5

Figura 162.
Issue de gestión de docentes



Nota. Elaboración Propia.

Issue #6

Figura 163.
Issue de gestión de estudiantes

The screenshot shows a Jira issue page for 'Gestión de estudiantes' (SUGD-6). The left sidebar contains navigation links like 'Para ti', 'Recientes', 'Marcados como fav...', 'Aplicaciones', 'Planes', 'Proyectos', and 'Filtros'. The main content area is titled 'Todas las actividades' and includes a search bar and filters. The issue description states: 'Capacidad de visualizar, editar y crear estudiantes dentro del sistema'. The 'Actividades secundarias' table lists tasks such as 'Añadir tabla para visualizar estudiantes', 'Añadir formulario para agregar...', and 'Añadir formulario para editar...'. The right sidebar shows 'Detalles' including 'Persona asignada' (CORDOVA MORILLO DAYANA LISBETH) and 'Etiquetas'.

T...	Clave	Resumen	P...	P...	Estado
	SUGD-37	Añadir tabla para visualizar estudiantes			TAREAS POR H...
	SUGD-38	Añadir formulario para agregar...			TAREAS POR H...
	SUGD-39	Añadir formulario para editar...			TAREAS POR H...

Nota. Elaboración Propia.

Issue #7

Figura 164.
Issue de gestión de carreras

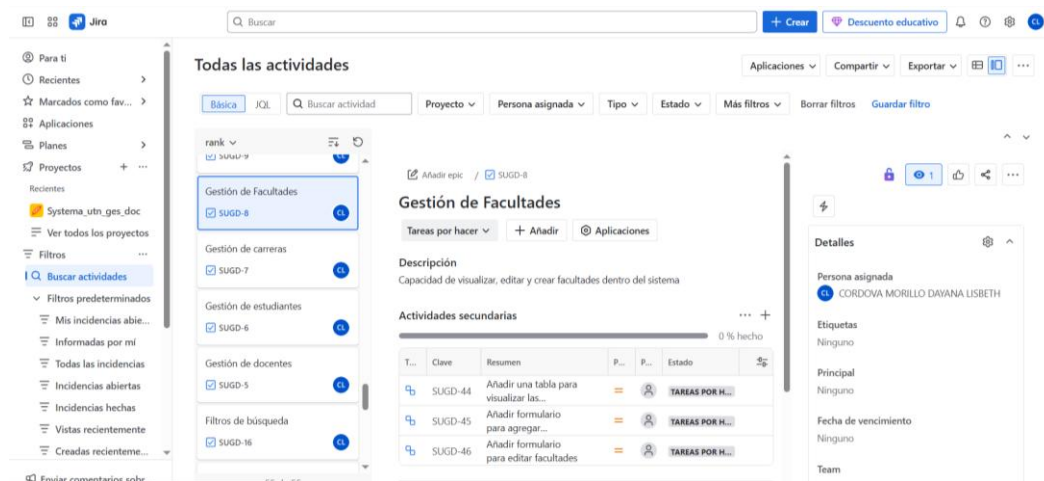
The screenshot shows a Jira issue page for 'Gestión de carreras' (SUGD-7). The left sidebar contains navigation links like 'Para ti', 'Recientes', 'Marcados como fav...', 'Aplicaciones', 'Planes', 'Proyectos', and 'Filtros'. The main content area is titled 'Todas las actividades' and includes a search bar and filters. The issue description states: 'Capacidad de visualizar, editar y crear carreras dentro del sistema'. The 'Actividades secundarias' table lists tasks such as 'Añadir tabla que muestre las carreras', 'Añadir formulario para agregar carreras', 'Añadir formulario para editar carreras', and 'Vincular una carrera'. The right sidebar shows 'Detalles' including 'Persona asignada' (CORDOVA MORILLO DAYANA LISBETH) and 'Etiquetas'.

T...	Clave	Resumen	P...	P...	Estado
	SUGD-40	Añadir tabla que muestre las carreras			TAREAS POR H...
	SUGD-41	Añadir formulario para agregar carreras			TAREAS POR H...
	SUGD-42	Añadir formulario para editar carreras			TAREAS POR H...
	SUGD-43	Vincular una carrera			TAREAS POR H...

Nota. Elaboración Propia.

Issue #8

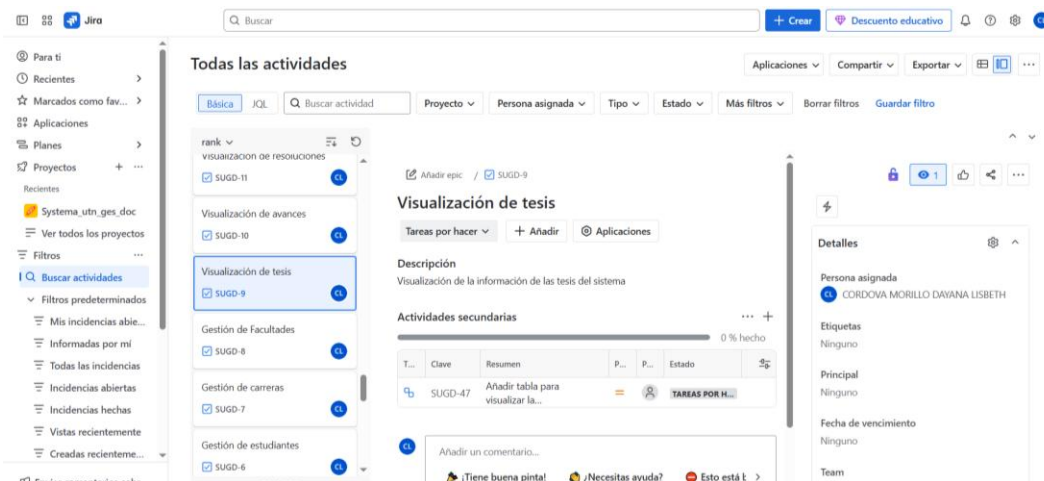
Figura 165.
Issue de gestión de facultades



Nota. Elaboración Propia.

Issue #9

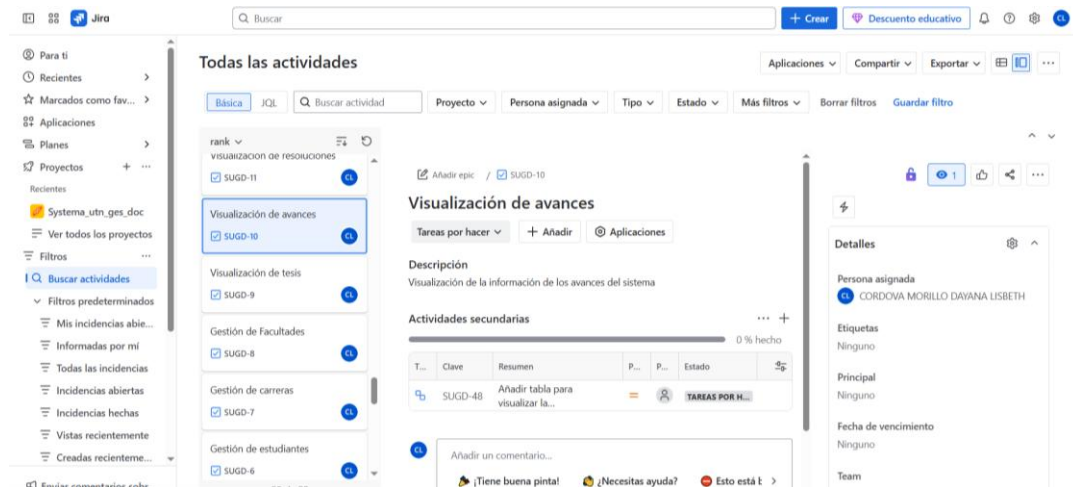
Figura 166.
Issue de visualización de tesis



Nota. Elaboración Propia.

Issue #10

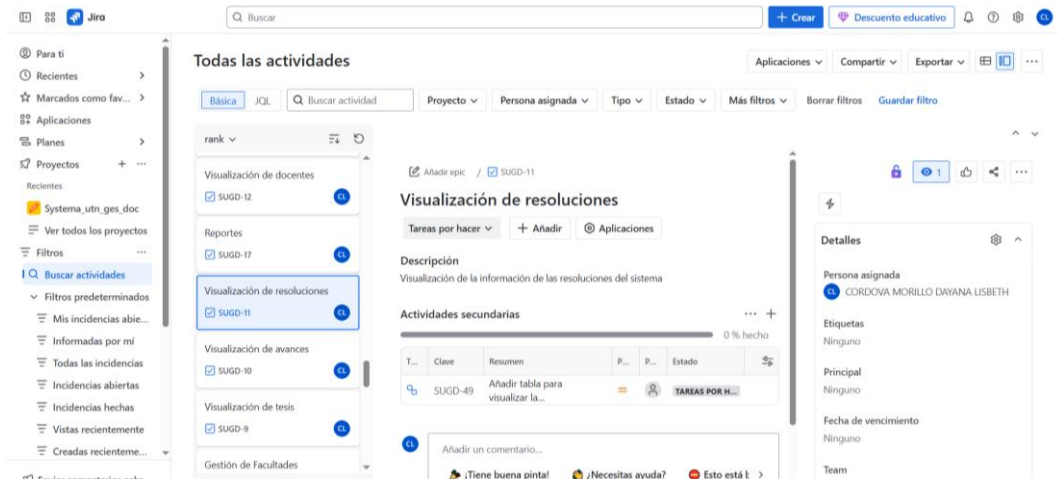
Figura 167.
Issue de visualización de avances



Nota. Elaboración Propia.

Issue #11

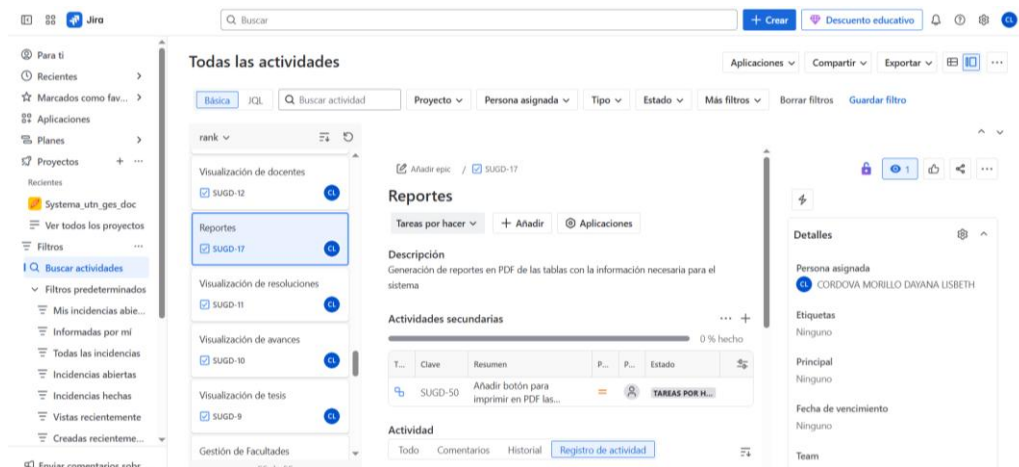
Figura 168.
Visualización de resoluciones



Nota. Elaboración Propia.

Issue #12

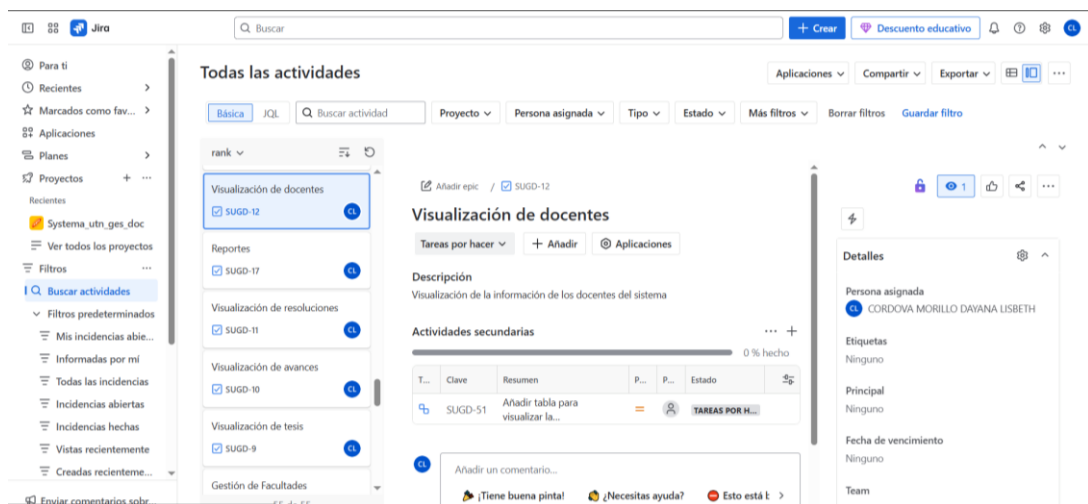
Figura 169.
Issue de Reportes



Nota. Elaboración Propia.

Issue #13

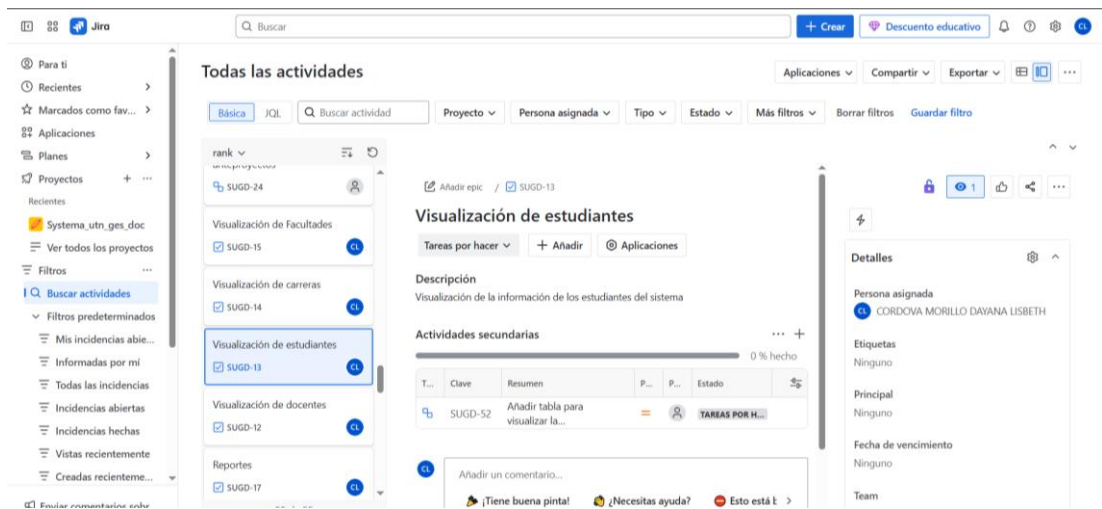
Figura 170.
Issue de visualización de docentes



Nota. Elaboración Propia.

Issue #14

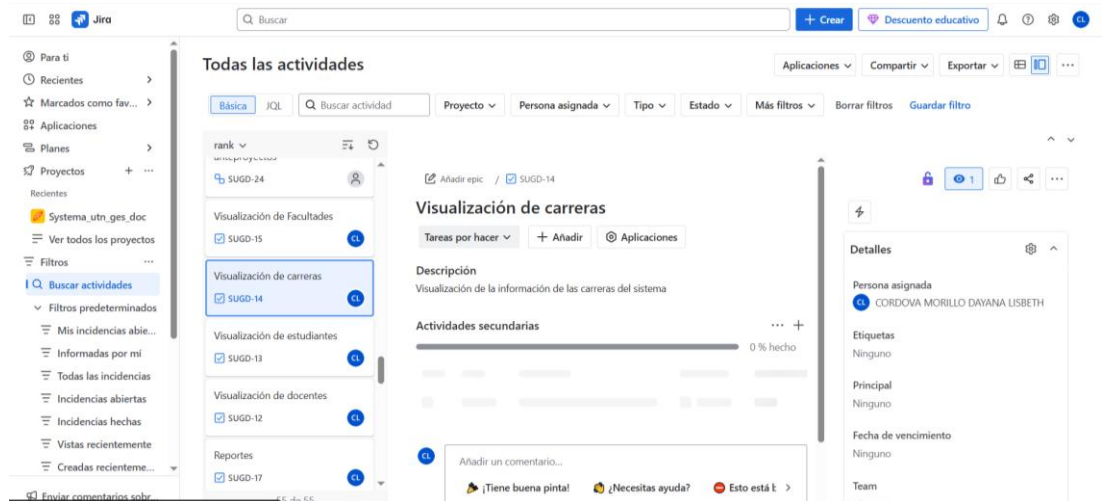
Figura 171.
Issue de visualización de estudiantes



Nota. Elaboración Propia.

Issue #15

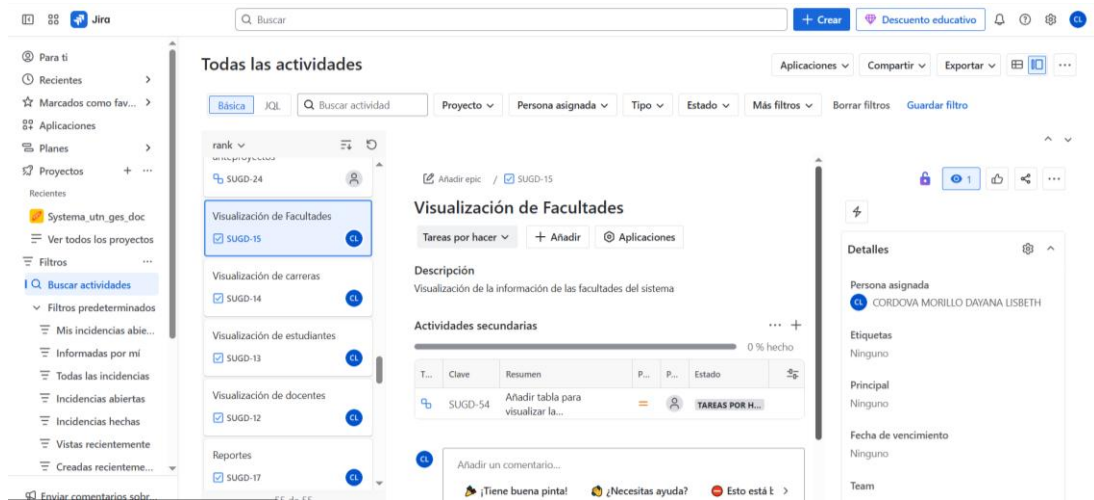
Figura 172.
Issue de visualización de carreras



Nota. Elaboración Propia.

Issue #16

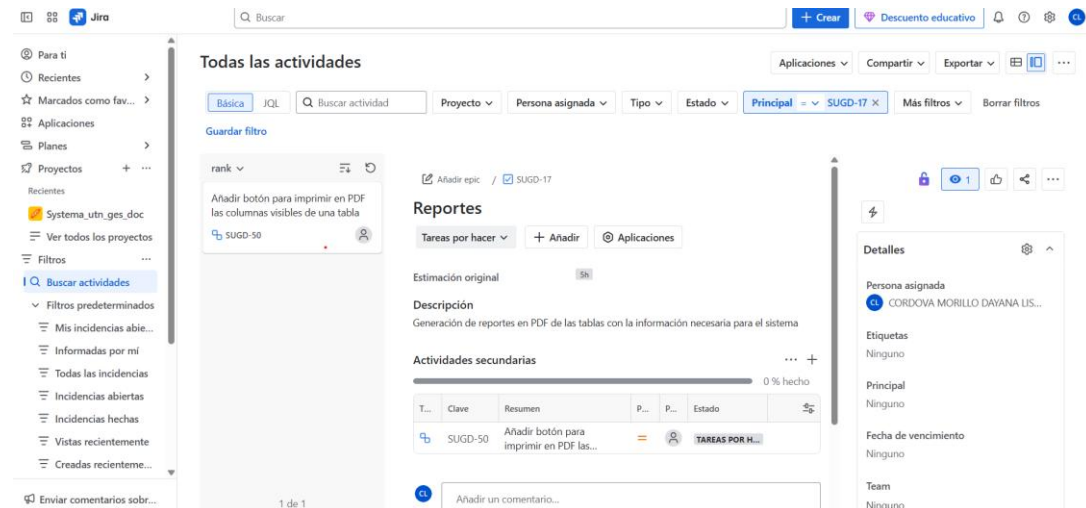
Figura 173.
Issue de visualización de facultades



Nota. Elaboración Propia.

Issue #17

Figura 174.
Issue de reportes



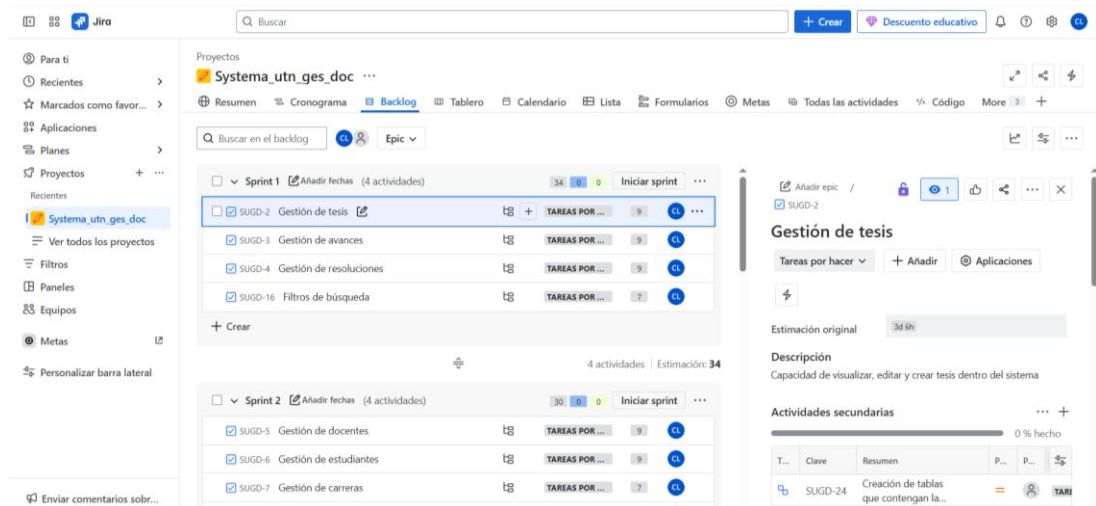
Nota. Elaboración Propia.

Time Tracking de los Issues

Issue # 1 con estimación de 30 horas

Figura 175.

Issue #1 con estimación de 30 horas

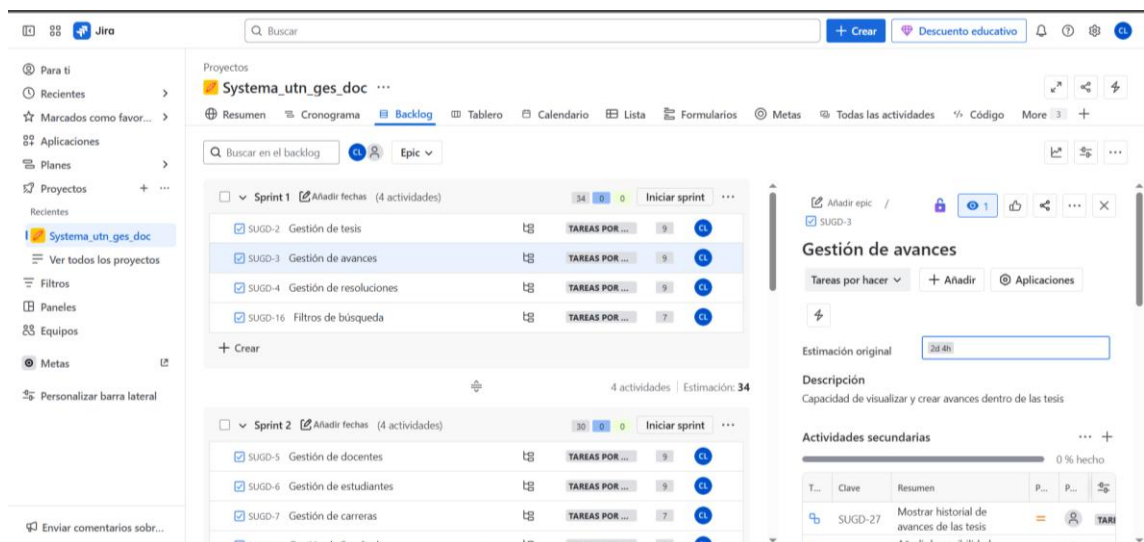


Nota. Elaboración Propia.

Issue #2 con estimación de 20 horas

Figura 176.

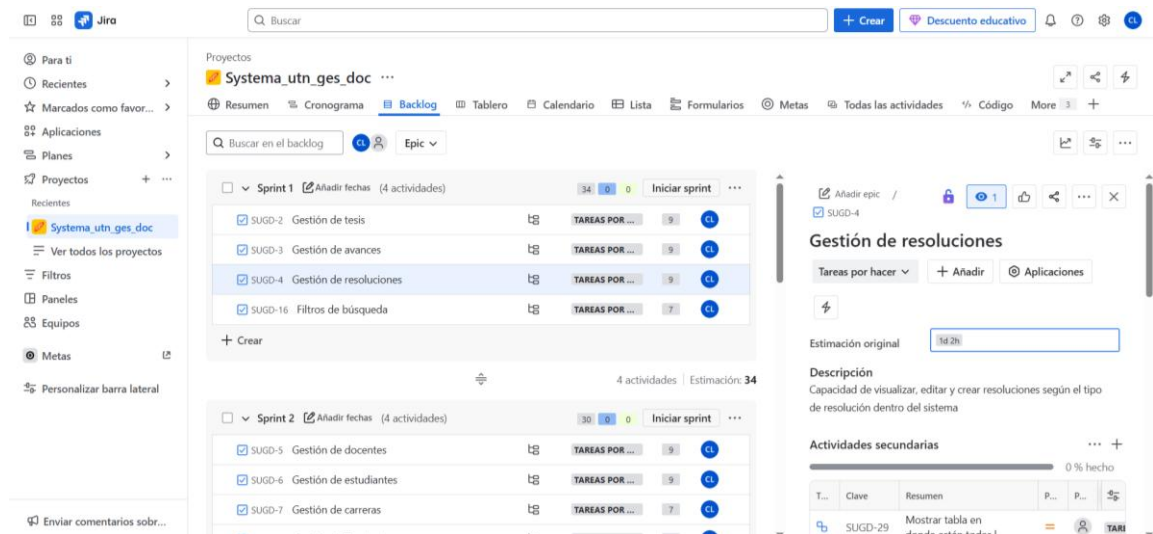
Issue #2 con estimación de 20 horas



Nota. Elaboración Propia.

Issue #3 con estimación de 10 horas

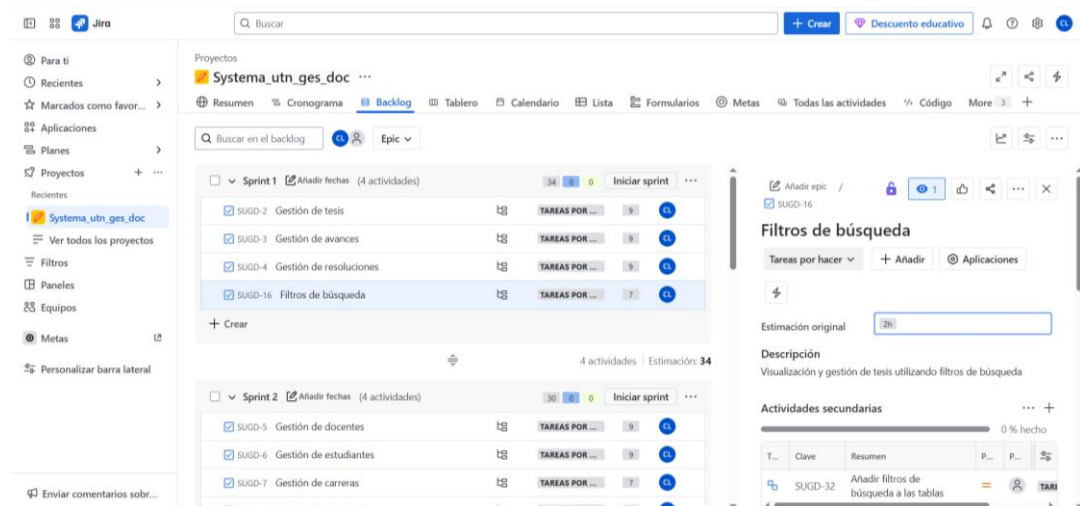
Figura 177.
Issue #3 con estimación de 10 horas



Nota. Elaboración Propia.

Issue #4 con estimación de 2 horas

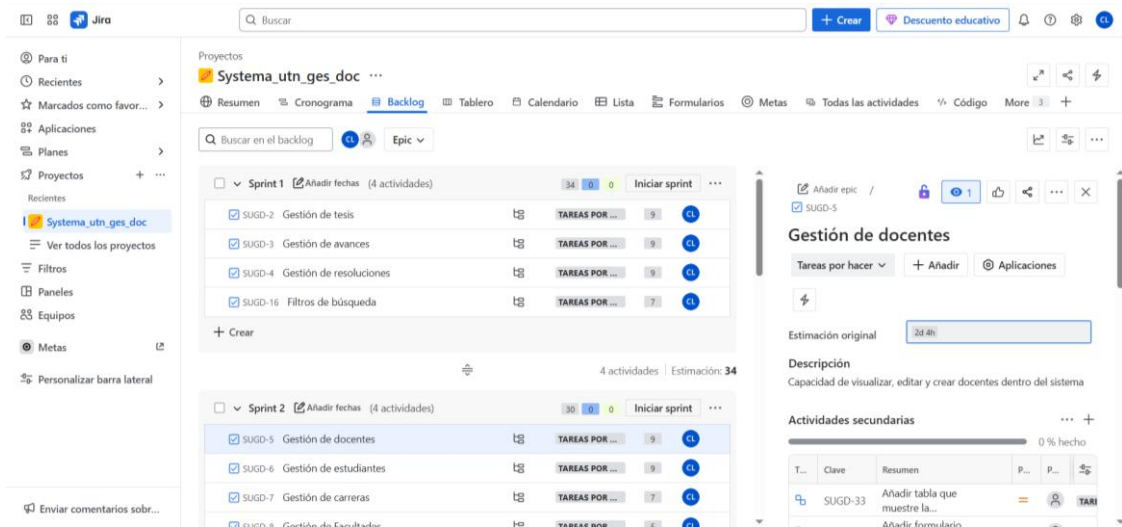
Figura 178.
Issue #4 con estimación de 2 horas



Nota. Elaboración Propia.

Issue #5 con estimación de 20 horas

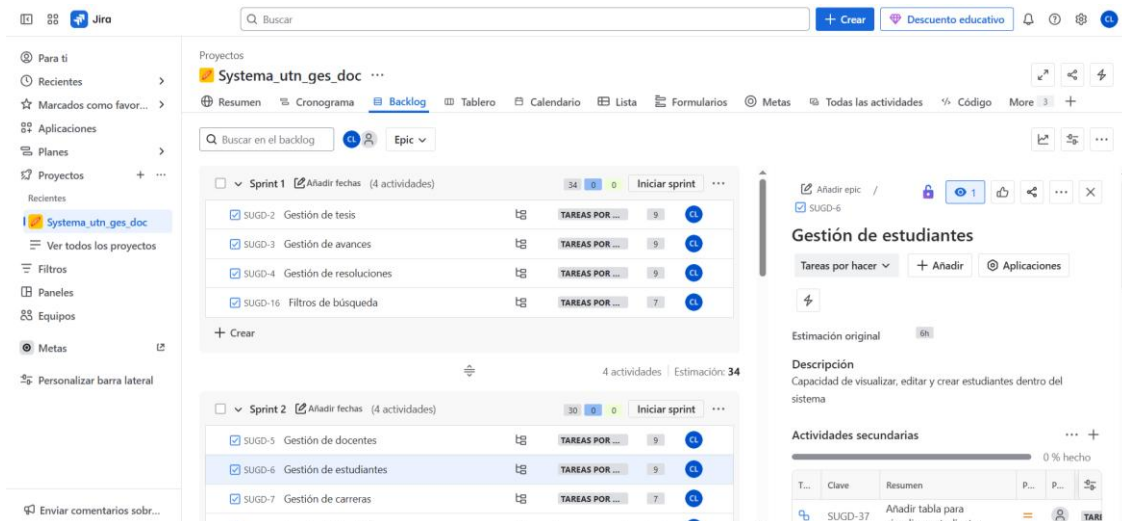
Figura 179.
Issue #5 con estimación de 20 horas



Nota. Elaboración Propia.

Issue #6 con estimación de 6 horas

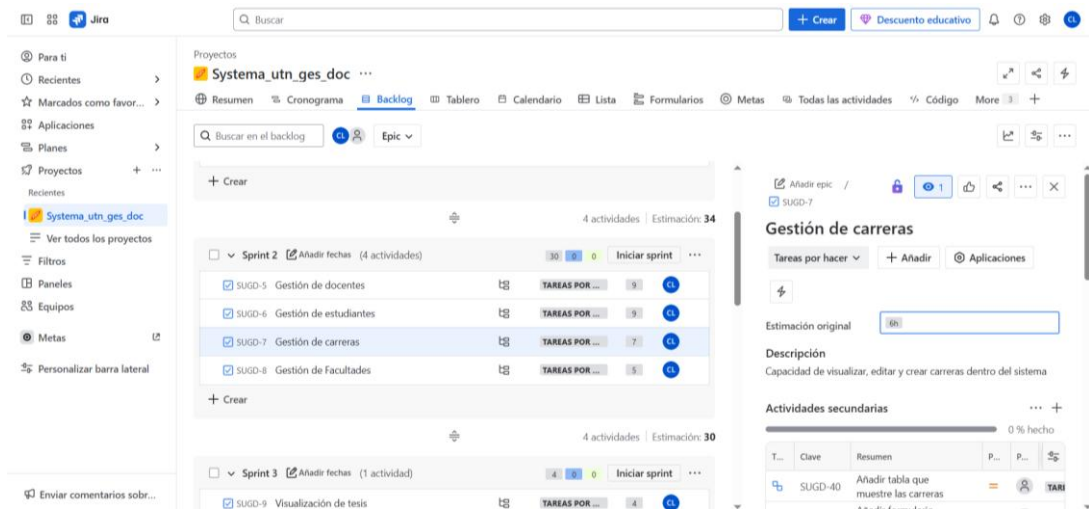
Figura 180.
Issue #6 con estimación de 6 horas



Nota. Elaboración Propia.

Issue #7 con estimación de 6 horas

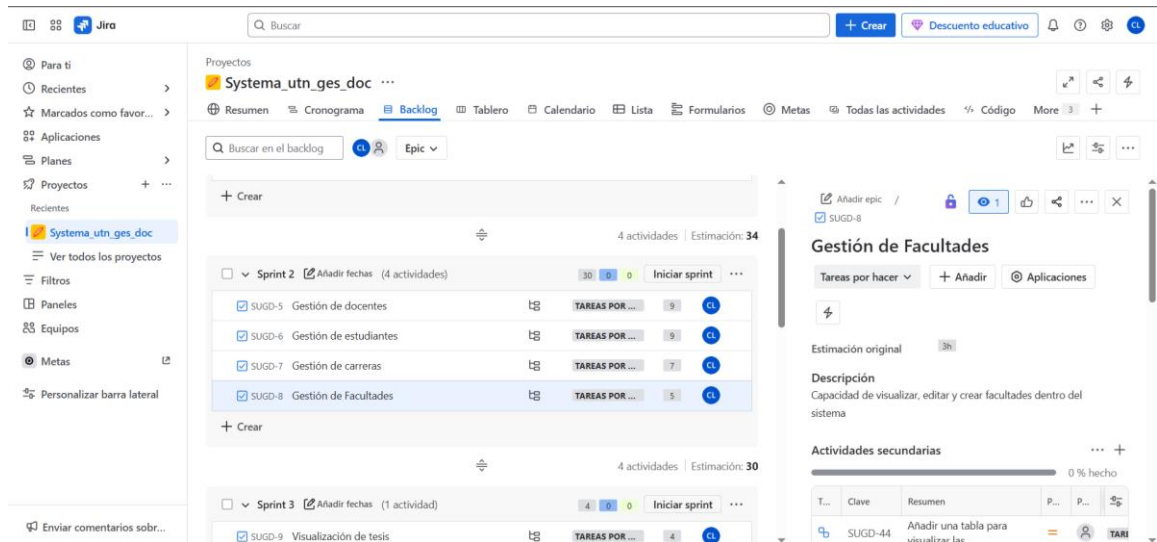
Figura 181.
Issue #7 con estimación de 6 horas



Nota. Elaboración Propia.

Issue #8 con estimación de 3 horas

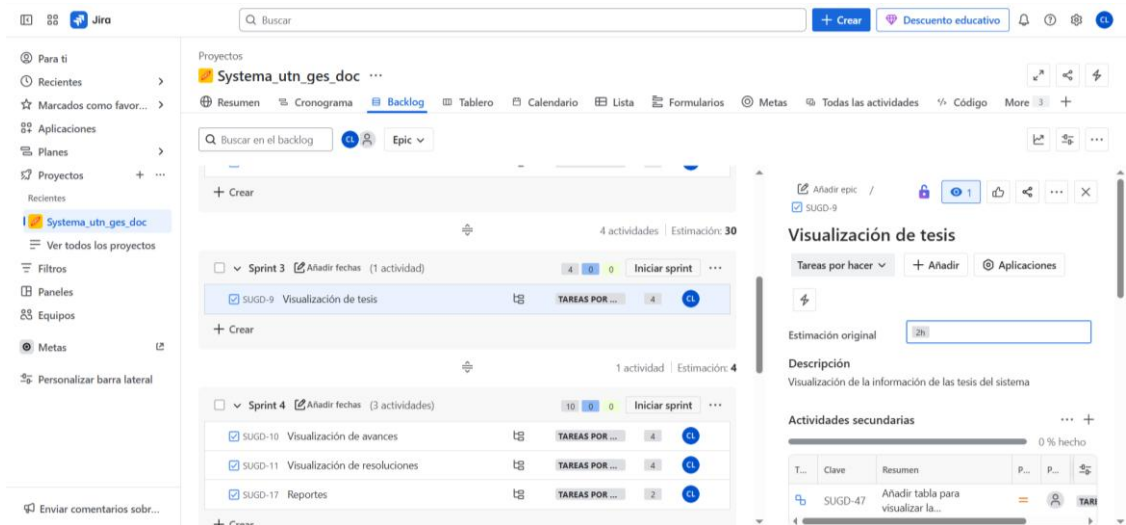
Figura 182.
Issue #8 con estimación de 3 horas



Nota. Elaboración Propia.

Issue #9 con estimación de 2 horas

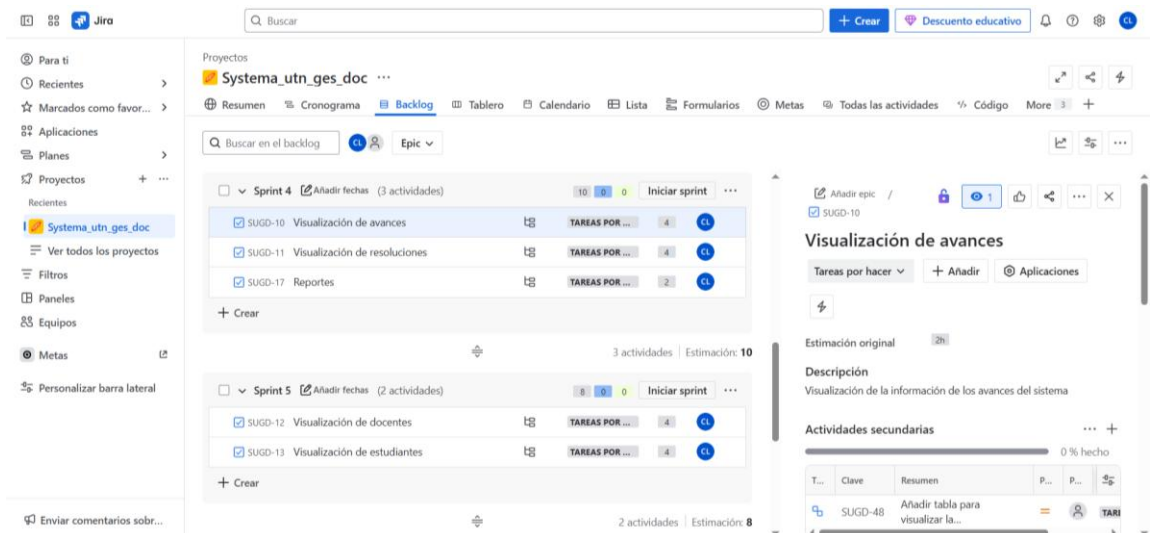
Figura 183.
Issue #9 con estimación de 2 horas



Nota. Elaboración Propia.

Issue #10 con estimación de 2 horas

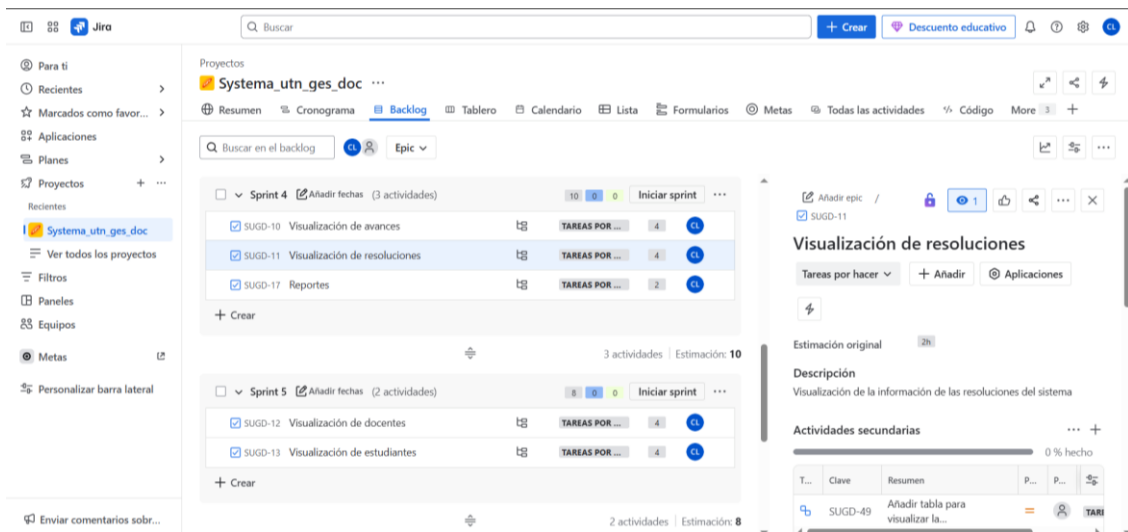
Figura 184.
Issue #10 con estimación de 2 horas



Nota. Elaboración Propia.

Issue #11 con estimación de 2 horas

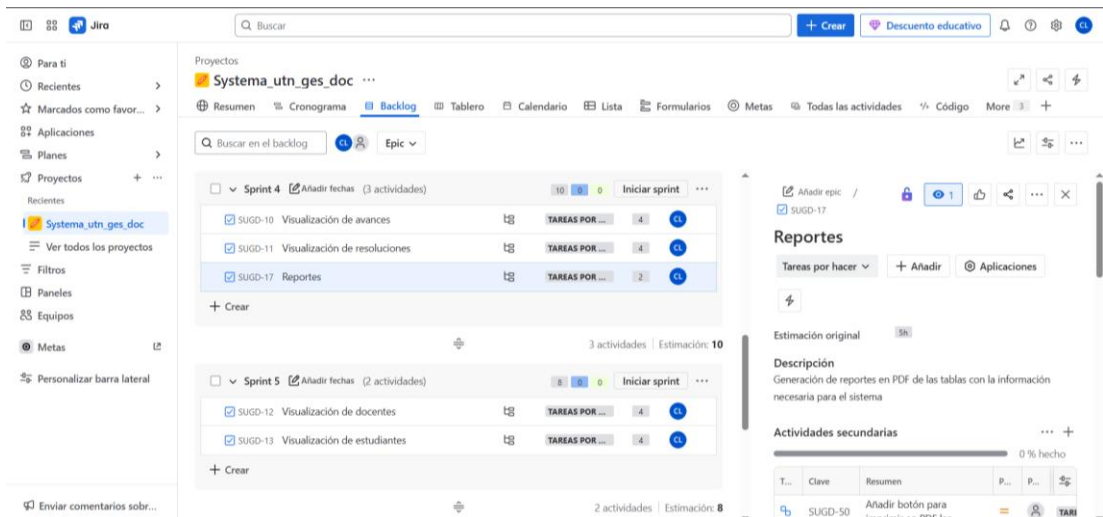
Figura 185.
Issue #11 con estimación de 2 horas



Nota. Elaboración Propia.

Issue #12 con estimación de 5 horas

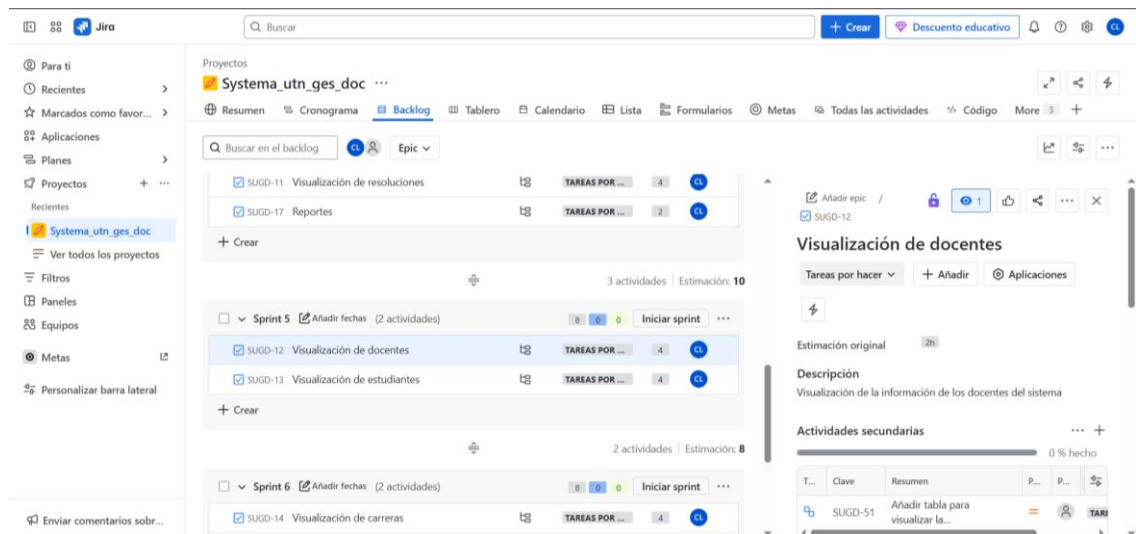
Figura 186.
Issue #12 con estimación de 5 horas



Nota. Elaboración Propia.

Issue #13 con estimación de 2 horas

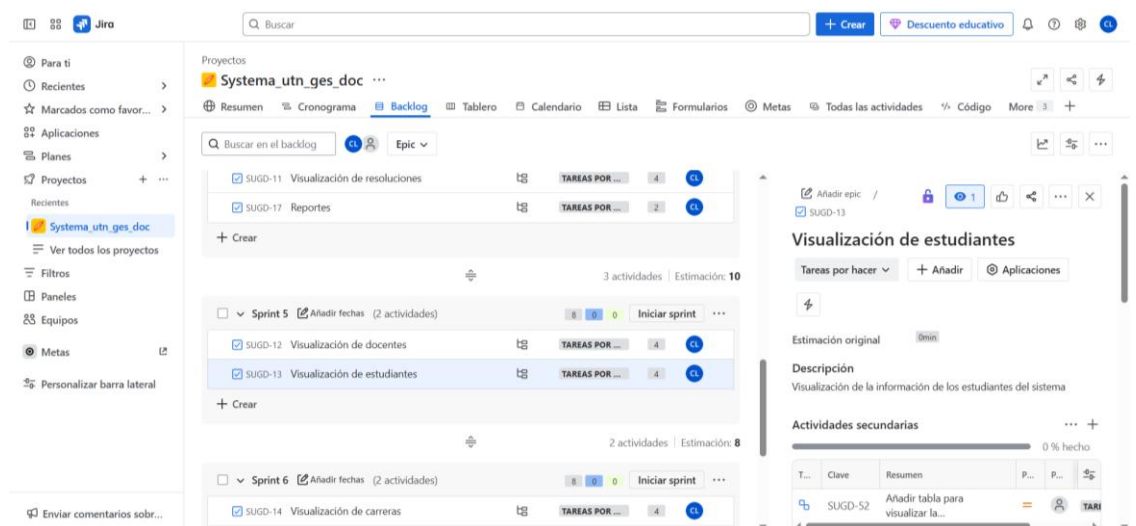
Figura 187.
Issue #13 con estimación de 2 horas



Nota. Elaboración Propia.

Issue #14 con estimación de 2 horas

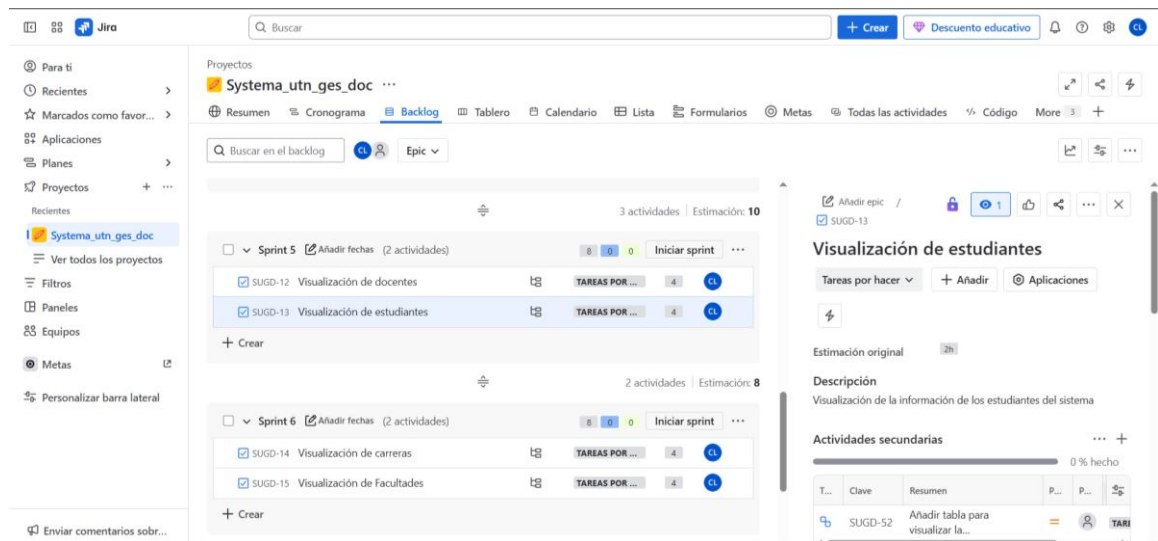
Figura 188.
Issue #14 con estimación de 2 horas



Nota. Elaboración Propia.

Issue #15 con estimación de 2 horas

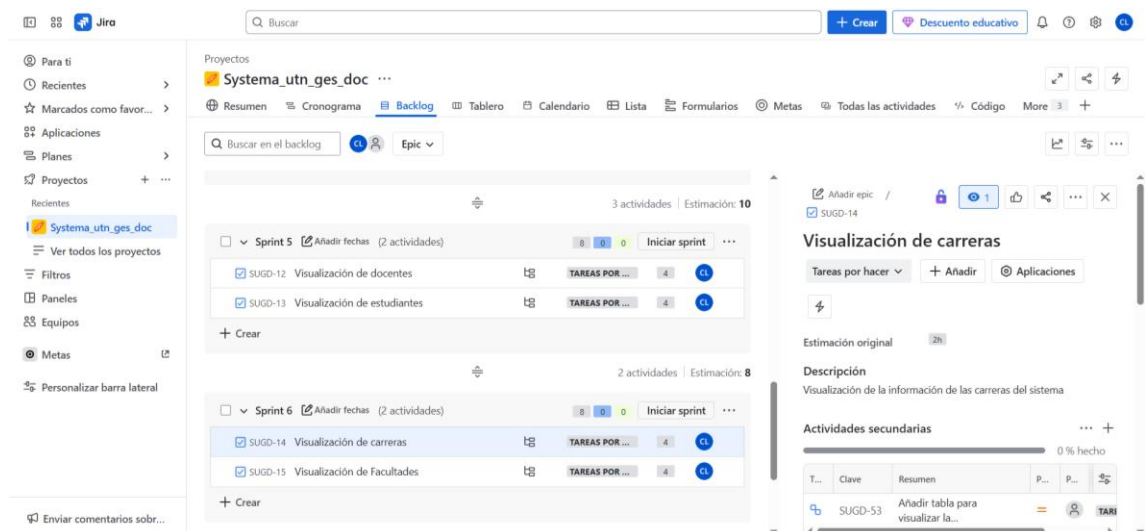
Figura 189.
Issue #15 con estimación de 2 horas



Nota. Elaboración Propia.

Issue #16 con estimación de 2 horas

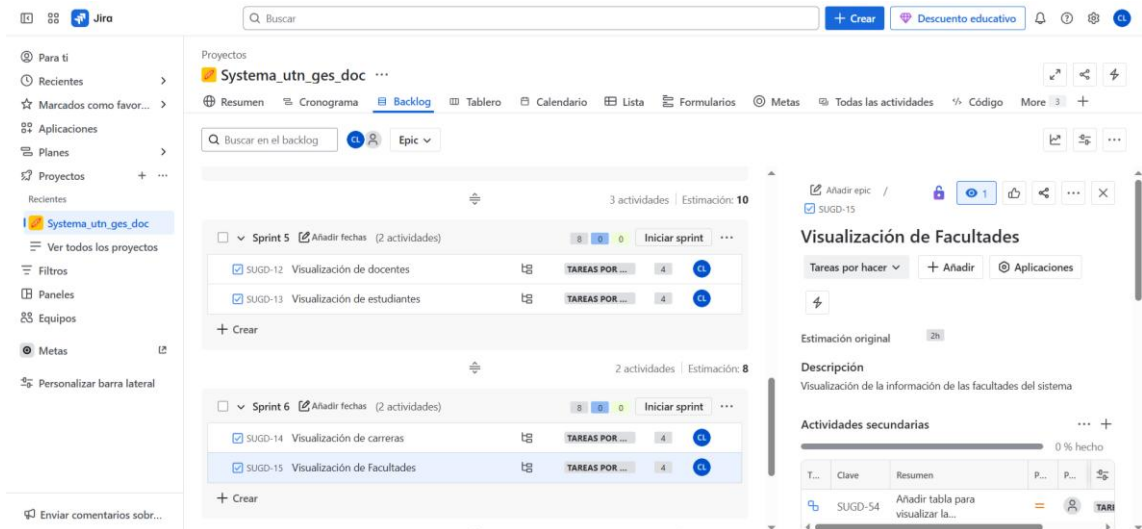
Figura 190.
Issue #16 con estimación de 2 horas



Nota. Elaboración Propia.

Issue #17 con estimación de 2 horas

Figura 191.
Issue #17 con estimación de 2 horas

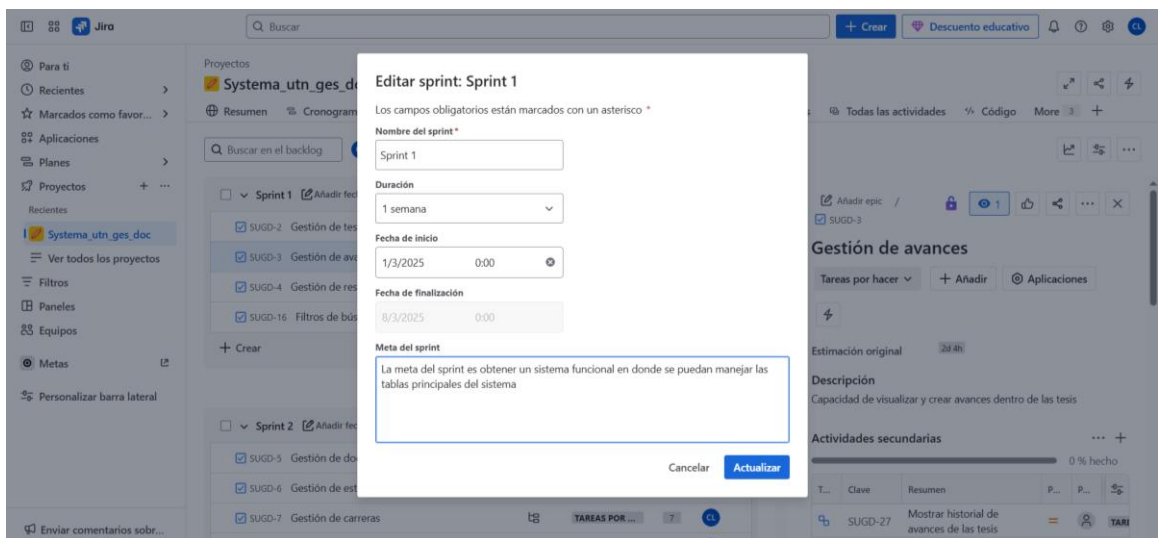


Nota. Elaboración Propia.

Fase 2. Juego o Desarrollo

2.1 Sprint Planning

Figura 192.
Edición de la información Sprint #1



Nota. Elaboración Propia.

Figura 193.
Edición de la información Sprint #2

Editar sprint: Sprint 2

Los campos obligatorios están marcados con un asterisco *

Nombre del sprint *
Sprint 2

Duración
1 semana

Fecha de inicio
10/3/2025 0:00

Fecha de finalización
17/3/2025 0:00

Meta del sprint
En esta iteración se plantea modificar tablas menos complejas

Cancelar Actualizar

Nota. Elaboración Propia.

Figura 194.
Edición de la información Sprint #3

Editar sprint: Sprint 3

Los campos obligatorios están marcados con un asterisco *

Nombre del sprint *
Sprint 3

Duración
1 semana

Fecha de inicio
13/3/2025 0:00

Fecha de finalización
20/3/2025 0:00

Meta del sprint
En este sprint se plantea iniciar con las vistas para el Coordinador

Cancelar Actualizar

Nota. Elaboración Propia.

Figura 195.
Edición de la información Sprint #4

Editar sprint: Sprint 4

Los campos obligatorios están marcados con un asterisco *

Nombre del sprint*
Sprint 4

Duración
1 semana

Fecha de inicio
14/3/2025 0:00

Fecha de finalización
21/3/2025 0:00

Meta del sprint
Se plantea la visualización de más tablas y de la generación de reportes en PDF

Cancelar Actualizar

Nota. Elaboración Propia.

Figura 196.
Edición de la información Sprint #5

Editar sprint: Sprint 5

Los campos obligatorios están marcados con un asterisco *

Nombre del sprint*
Sprint 5

Duración
1 semana

Fecha de inicio
17/3/2025 0:00

Fecha de finalización
24/3/2025 0:00

Meta del sprint
Se plantea añadir la visualización de tablas de estudiantes y docentes para el Coordinador

Cancelar Actualizar

Nota. Elaboración Propia.

Figura 197.
Edición de la información Sprint #6

Editar sprint: Sprint 6

Los campos obligatorios están marcados con un asterisco *

Nombre del sprint*
Sprint 6

Duración
1 semana

Fecha de inicio
18/3/2025 0:00

Fecha de finalización
25/3/2025 0:00

Meta del sprint
Finalización de la primera versión del programa

Cancelar Actualizar

Nota. Elaboración Propia.

2.2 Sprint Backlog

Tabla 34.
Sprint backlog

ID	Historia / Ítem	Estimación (horas)	Sprint Asignado	Fecha
G001	Gestión de tesis	30	Sprint 1	01/03/2025 hasta 08/03/2025
G002	Gestión de avances	20	Sprint 1	01/03/2025 hasta 08/03/2025
G003	Gestión de resoluciones	10	Sprint 1	01/03/2025 hasta 08/03/2025
G004	Gestión de docentes	10	Sprint 2	10/03/2025 hasta 17/03/2025
G005	Gestión de estudiantes	6	Sprint 2	10/03/2025 hasta 17/03/2025
G006	Gestión de carreras	6	Sprint 2	10/03/2025 hasta 17/03/2025
G007	Gestión de Facultades	3	Sprint 2	10/03/2025 hasta 17/03/2025

V001	Visualización de tesis	2	Sprint 3	13/03/2025 hasta 20/03/2025
V002	Visualización de avances	2	Sprint 4	14/03/2025 hasta 21/03/2025
V003	Visualización de resoluciones	2	Sprint 4	14/03/2025 hasta 21/03/2025
V004	Visualización de docentes	2	Sprint 5	17/03/2025 hasta 24/03/2025
V005	Visualización de estudiantes	2	Sprint 5	17/03/2025 hasta 24/03/2025
V006	Visualización de carreras	2	Sprint 6	18/03/2025 hasta 25/03/2025
V007	Visualización de Facultades	2	Sprint 6	18/03/2025 hasta 25/03/2025
V008	Filtros de búsqueda	2	Sprint 1	01/03/2025 hasta 08/03/2025
V009	Reportes	5	Sprint 4	14/03/2025 hasta 21/03/2025

Nota. Elaboración propia

2.3 Iteraciones

Iteración 1

Inicio del Sprint 1

Prototipo del Sprint 1

Tabla 35.
Prototipo Sprint 1

Sprint 1		
Historia de usuario	Tarea	Horas
Gestión de tesis	Creación de tablas que contengan la información requerida para los anteproyectos	10

Gestión de avances	Creación de vista para registro de anteproyectos	10
	Formulario de edición de tesis dentro de la tabla	10
	Mostrar historial de avances de las tesis	8
Gestión de resoluciones	Añadir la posibilidad de agregar avances a las tesis	10
	Añadir un botón para mostrar el historial de avances de un proyecto	2
	Mostrar tabla en donde estén todas las resoluciones	3
Filtros de búsqueda	Agregar formulario para insertar resoluciones	4
	Vincular las resoluciones con las tesis aprobadas	3
	Añadir filtros de búsqueda a las tablas	2

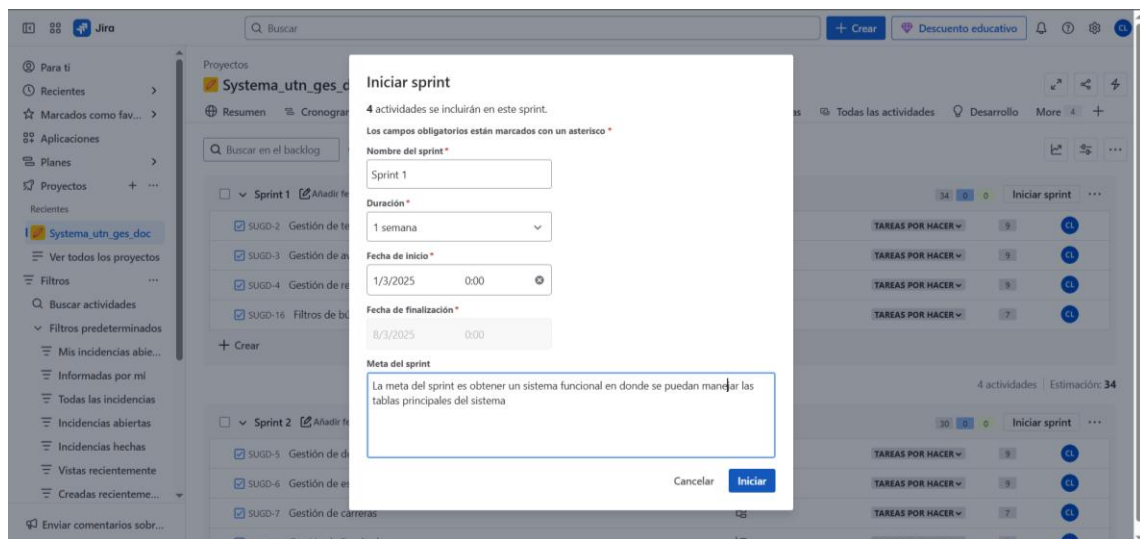
Nota. Elaboración propia

Figura 198.
Tablero Sprint 1



Nota. Elaboración Propia.

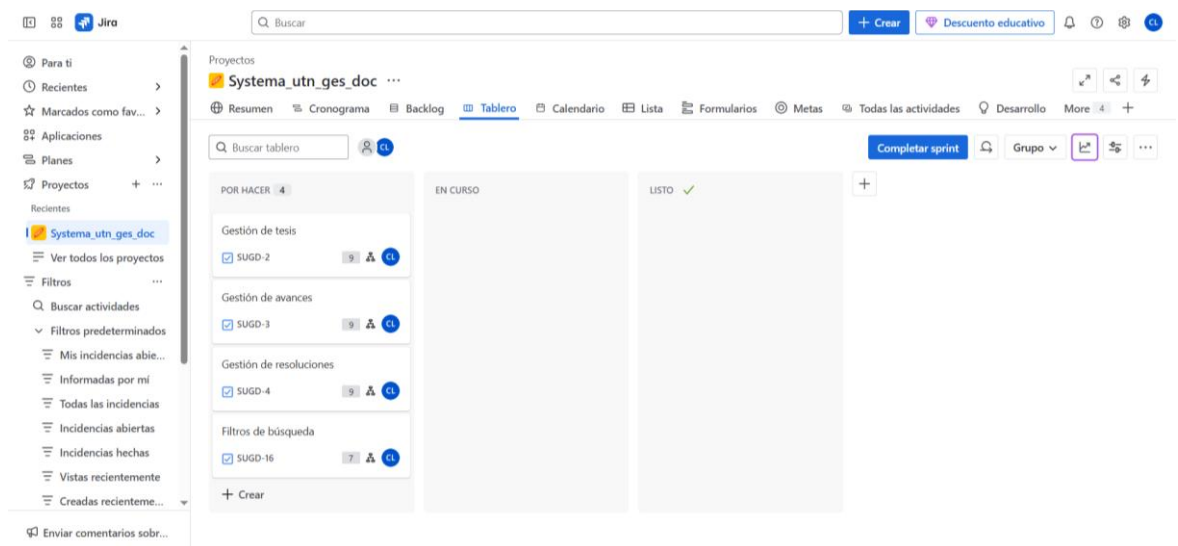
Figura 199.
Inicio del Sprint #1



Nota. Elaboración Propia.

Seguimiento del progreso de las tareas

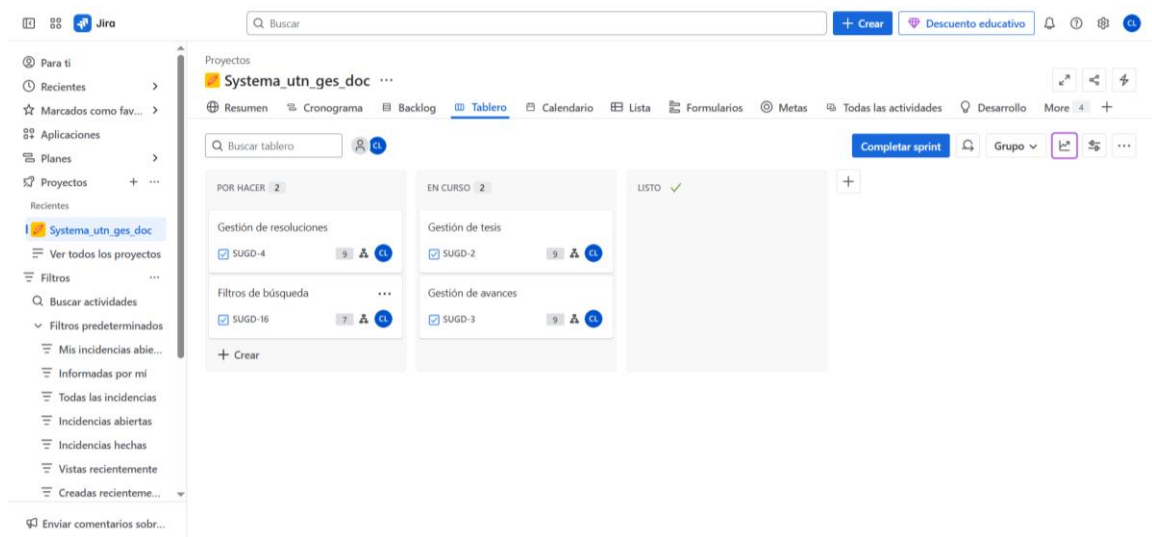
Figura 200.
Tareas del Sprint 1 sin modificaciones



Nota. Elaboración Propia.

Visualización del avance del Sprint

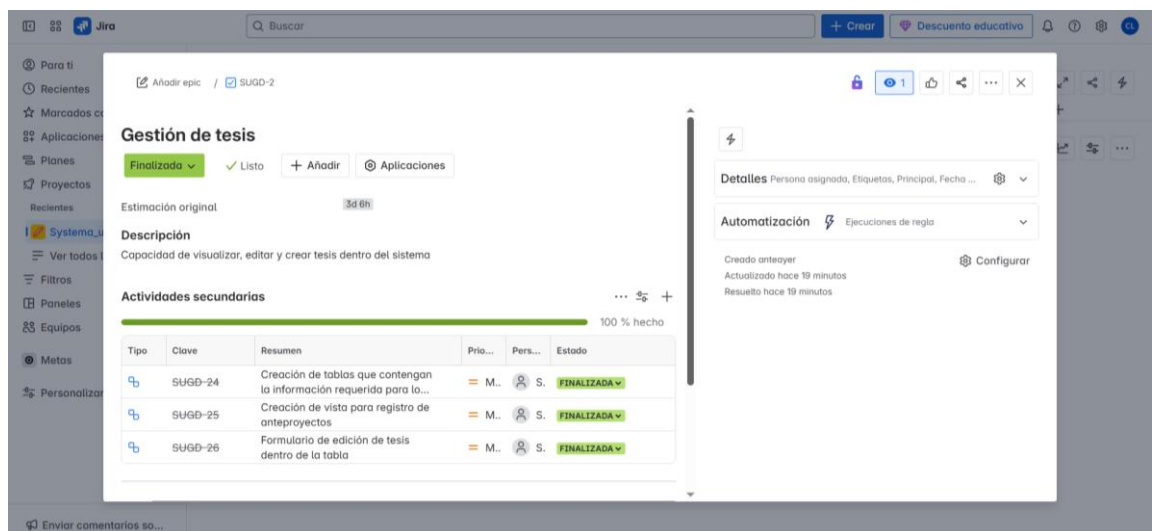
Figura 201.
Avances del Sprint 1



Nota. Elaboración Propia.

Finalización del Issue #1

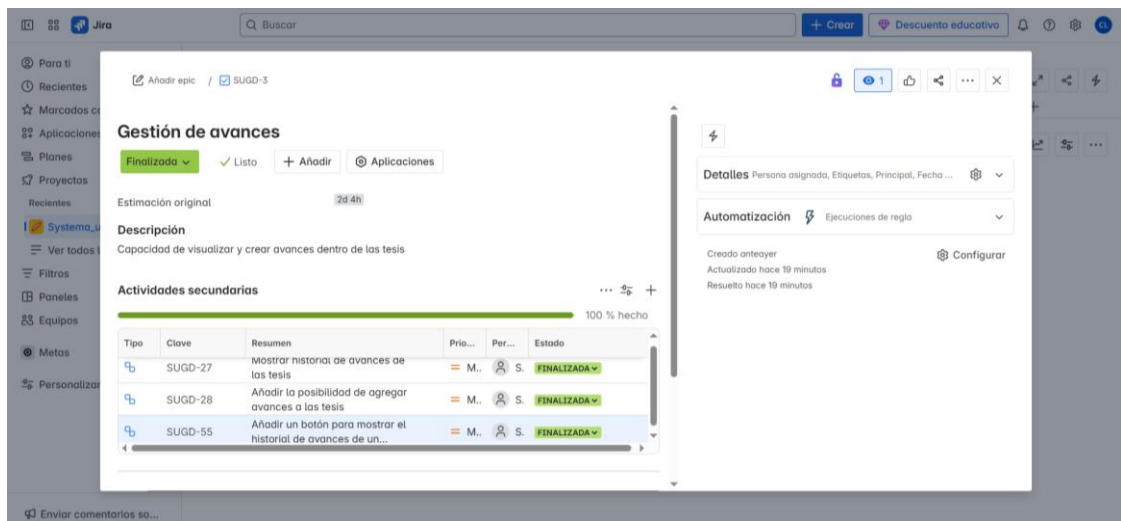
Figura 202.
Finalización del Issue #1



Nota. Elaboración Propia.

Finalización del Issue #2

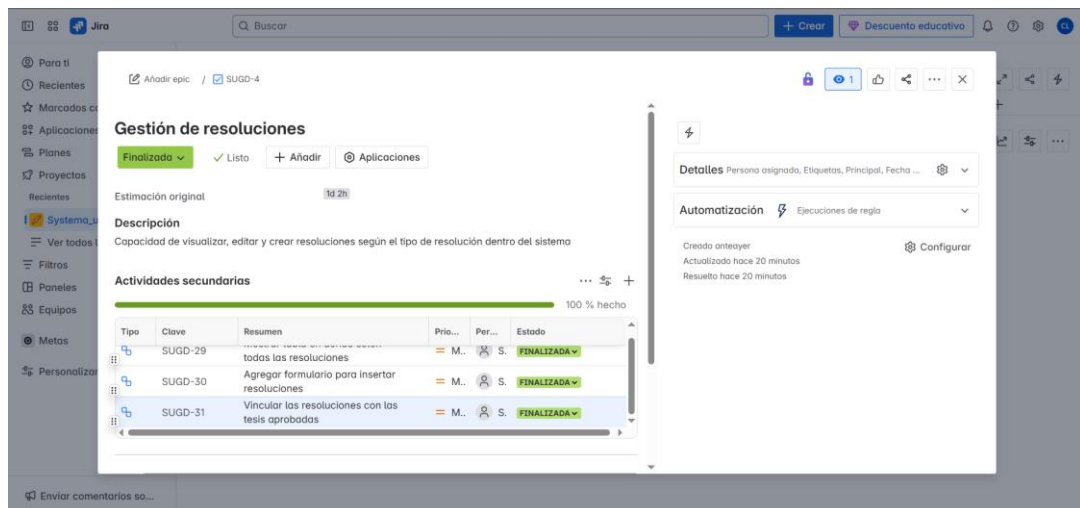
Figura 203.
Finalización del Issue #2



Nota. Elaboración Propia.

Finalización del Issue #3

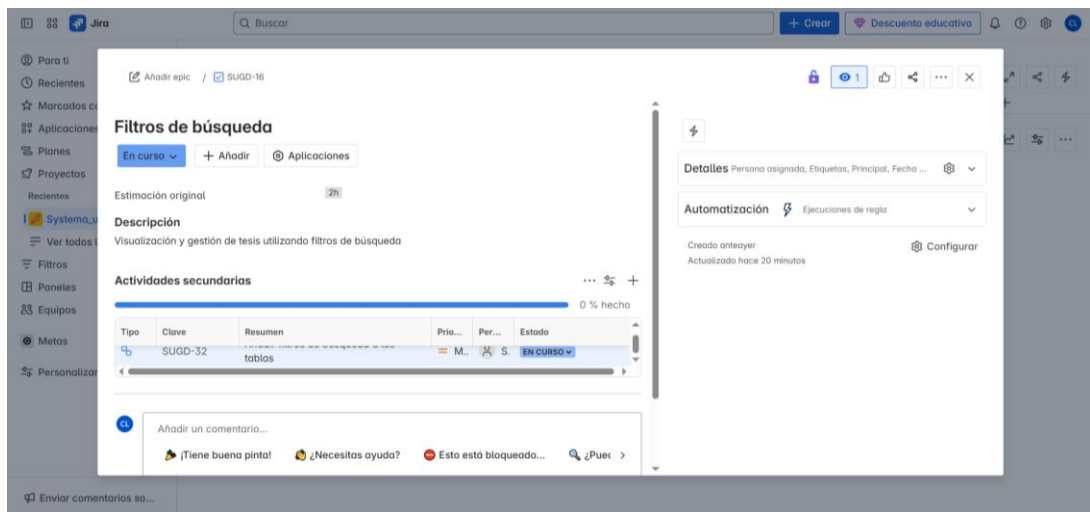
Figura 204.
Finalización del Issue #3



Nota. Elaboración Propia.

Avance del Issue #4

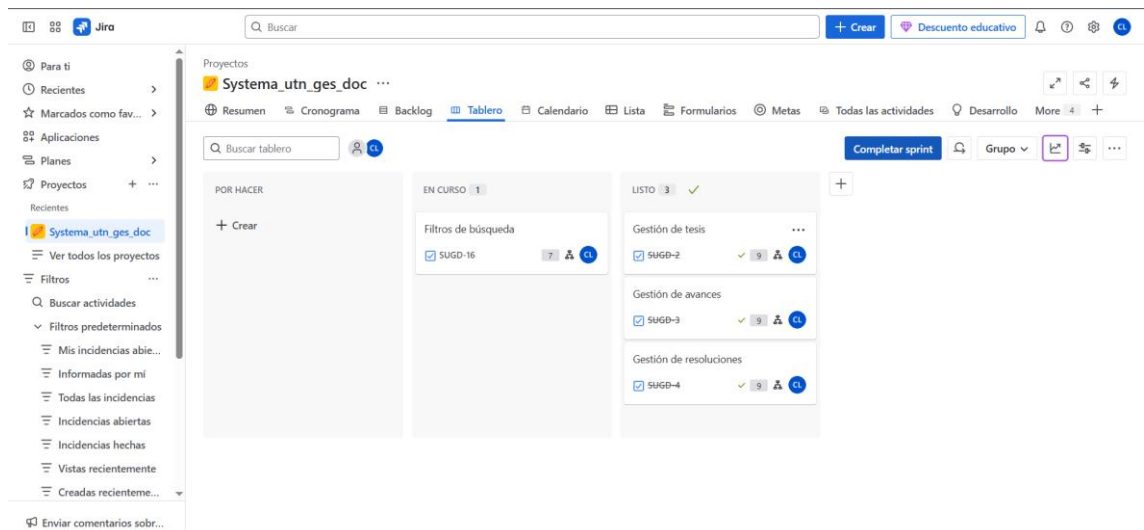
Figura 205.
Avance del Issue #4



Nota. Elaboración Propia.

Cierre del Sprint

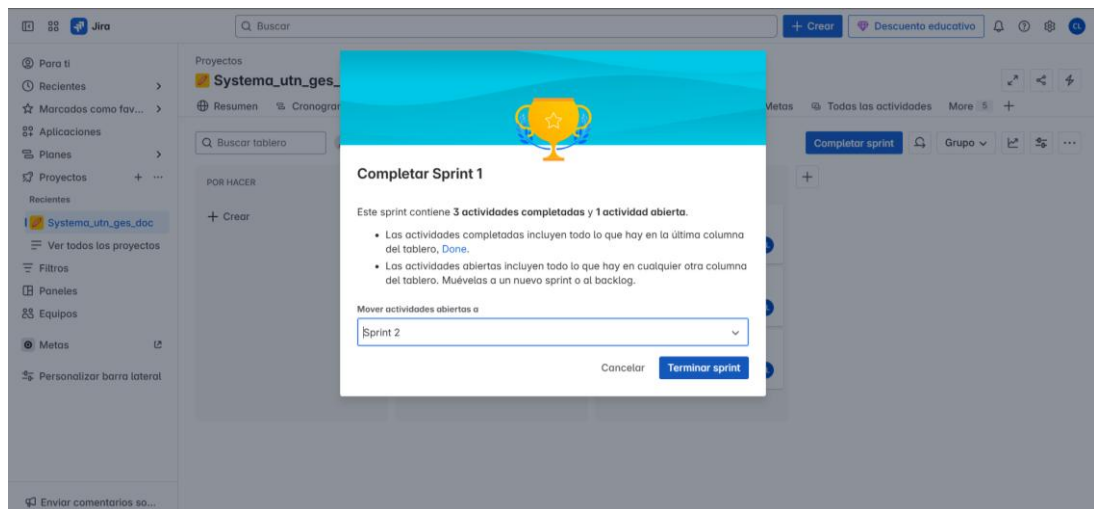
Figura 206.
Finalización del Sprint 1



Nota. Elaboración Propia.

Gestión de las actividades faltantes enviadas al siguiente Sprint

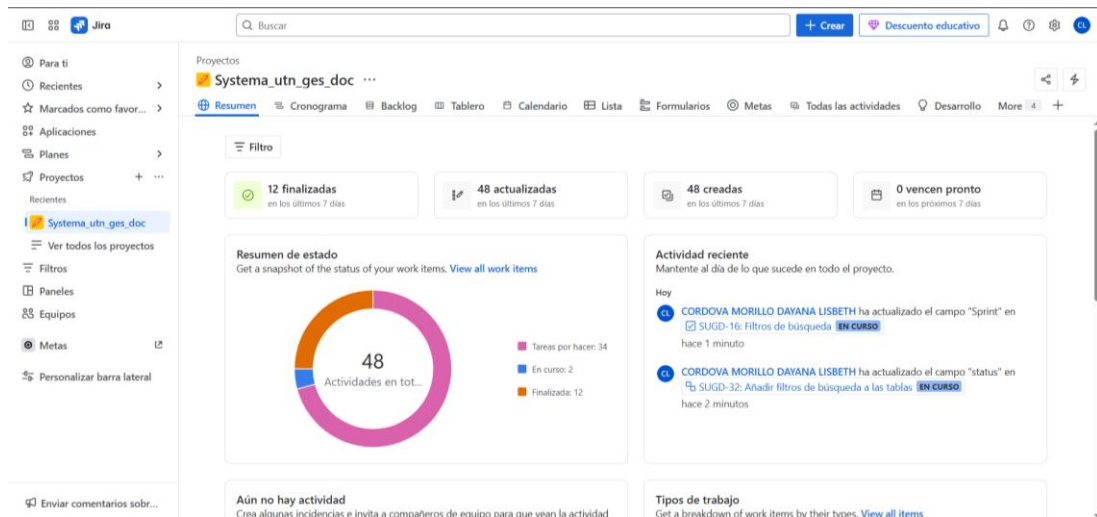
Figura 207.
Resumen sencillo del Sprint 1



Nota. Elaboración Propia.

Sprint retrospective

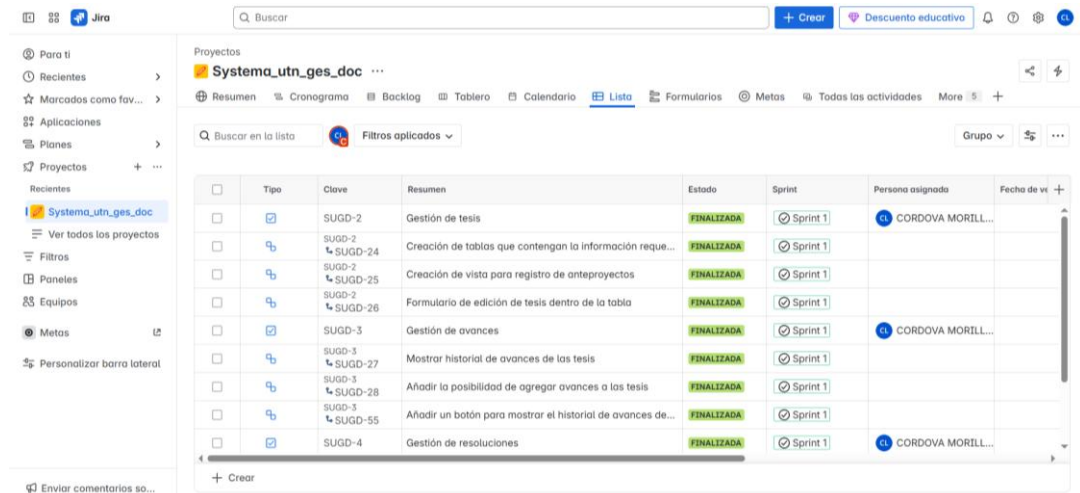
Figura 208.
Resumen avanzado del Sprint 1



Nota. Elaboración Propia.

Actividades finalizadas

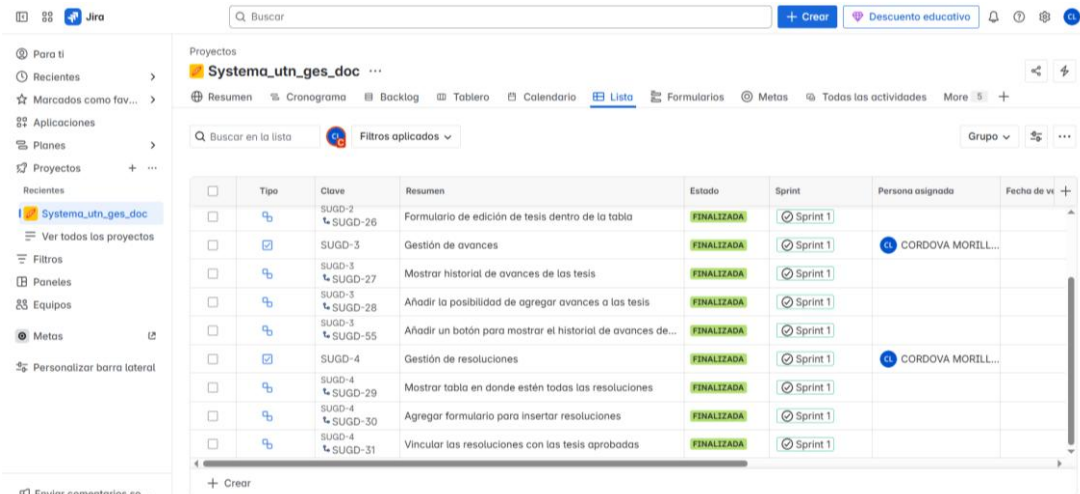
Figura 209.
Tareas completadas en el Sprint 1



	Tipo	Clave	Resumen	Estado	Sprint	Persona asignada	Fecha de vi
☐	☒	SUGD-2	Gestión de tesis	FINALIZADA	Sprint 1	CORDOVA MORELL...	
☐	☒	SUGD-24	Creación de tablas que contengan la información requ...	FINALIZADA	Sprint 1		
☐	☒	SUGD-25	Creación de vista para registro de anteproyectos	FINALIZADA	Sprint 1		
☐	☒	SUGD-26	Formulario de edición de tesis dentro de la tabla	FINALIZADA	Sprint 1		
☐	☒	SUGD-3	Gestión de avances	FINALIZADA	Sprint 1	CORDOVA MORELL...	
☐	☒	SUGD-27	Mostrar historial de avances de las tesis	FINALIZADA	Sprint 1		
☐	☒	SUGD-28	Añadir la posibilidad de agregar avances a las tesis	FINALIZADA	Sprint 1		
☐	☒	SUGD-55	Añadir un botón para mostrar el historial de avances de...	FINALIZADA	Sprint 1		
☐	☒	SUGD-4	Gestión de resoluciones	FINALIZADA	Sprint 1	CORDOVA MORELL...	

Nota. Elaboración Propia.

Figura 210.
Tareas completadas en el Sprint 1

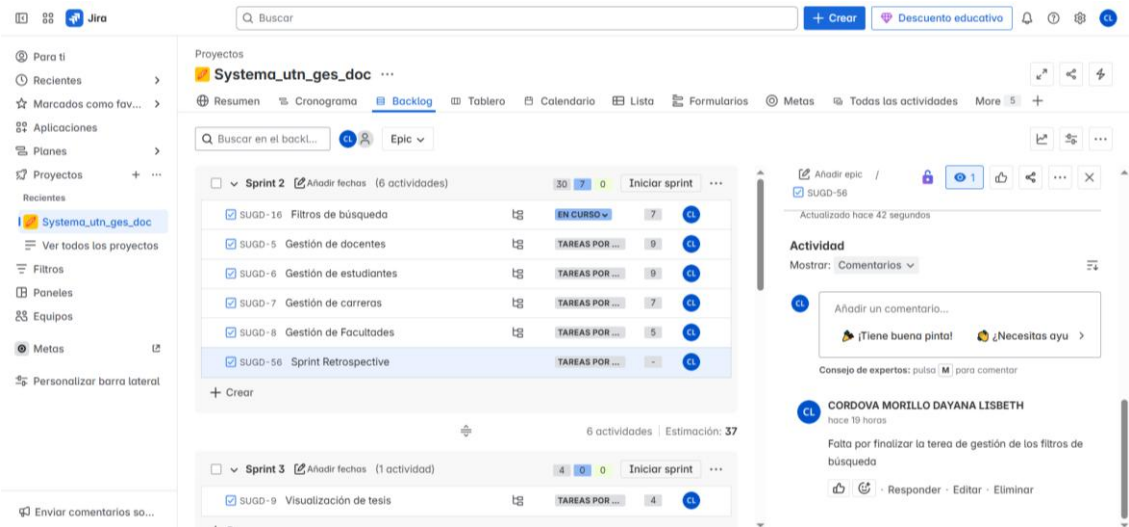


	Tipo	Clave	Resumen	Estado	Sprint	Persona asignada	Fecha de vi
☐	☒	SUGD-26	Formulario de edición de tesis dentro de la tabla	FINALIZADA	Sprint 1		
☐	☒	SUGD-3	Gestión de avances	FINALIZADA	Sprint 1	CORDOVA MORELL...	
☐	☒	SUGD-27	Mostrar historial de avances de las tesis	FINALIZADA	Sprint 1		
☐	☒	SUGD-28	Añadir la posibilidad de agregar avances a las tesis	FINALIZADA	Sprint 1		
☐	☒	SUGD-55	Añadir un botón para mostrar el historial de avances de...	FINALIZADA	Sprint 1		
☐	☒	SUGD-4	Gestión de resoluciones	FINALIZADA	Sprint 1	CORDOVA MORELL...	
☐	☒	SUGD-29	Mostrar tabla en donde estén todas las resoluciones	FINALIZADA	Sprint 1		
☐	☒	SUGD-30	Agregar formulario para insertar resoluciones	FINALIZADA	Sprint 1		
☐	☒	SUGD-31	Vincular las resoluciones con las tesis aprobadas	FINALIZADA	Sprint 1		

Nota. Elaboración Propia.

Comentarios del Sprint 1

Figura 211.
Issues retrasados del del Sprint 1 en el Sprint 2



Nota. Elaboración Propia.

Iteración 2

Inicio del Sprint 2

Prototipo del Sprint 2

Tabla 36.
Prototipo Sprint 2

Sprint 2		
Historia de usuario	Tarea	Horas
Gestión de docentes	Añadir tabla que muestre la información de los docentes	5
	Añadir formulario para agregar docente	5
	Añadir formulario para editar docente	5
	Agregar varias carreras a un docente	5
	Añadir tabla para visualizar estudiantes	2
Gestión de estudiantes	Añadir formulario para agregar estudiantes	2
	Añadir formulario para editar estudiantes	2
Gestión de carreras	Añadir tabla que muestre las carreras	1
	Añadir formulario para agregar carreras	1
	Añadir formulario para editar carreras	2
	Vincular una carrera con su facultad	2

Gestión de Facultades	Añadir una tabla para visualizar las facultades	1
	Añadir formulario para agregar facultades	1
	Añadir formulario para editar facultades	1

Nota. Elaboración propia

Figura 212.
Tablero Sprint 2

☐	▼ Sprint 2	✎ Añadir fechas (6 actividades)	30	7	0	Iniciar sprint	...
☒	SUGD-16	Filtros de búsqueda	EN CURSO	7		CL	
☒	SUGD-5	Gestión de docentes	TAREAS POR ...	9		CL	
☒	SUGD-6	Gestión de estudiantes	TAREAS POR ...	9		CL	
☒	SUGD-7	Gestión de carreras	TAREAS POR ...	7		CL	
☒	SUGD-8	Gestión de Facultades	TAREAS POR ...	5		CL	
☒	SUGD-56	Sprint Retrospective	TAREAS POR ...	-		CL	
+ Crear							

Nota. Elaboración Propia.

Figura 213.
Inicio del Sprint #2

Jira

Para ti

Recientes

Marcados como fav...

Aplicaciones

Panes

Proyectos

Recientes

Systema_utm_ges...

Ver todos los proyectos

Filtros

Paneles

Equipos

Metas

Personalizar barra lateral

Proyectos

Systema_utm_ges...

Resumen

Cronograma

Buscar en el backl...

▼ Sprint 2

✎ Añadir

☒ SUGD-16

Filtros de

☒ SUGD-5

Gestión de

☒ SUGD-6

Gestión de

☒ SUGD-7

Gestión de

☒ SUGD-8

Gestión de

☒ SUGD-56

Sprint Ret

+ Crear

Iniciar sprint

6 actividades se incluirán en este sprint.

Los campos obligatorios están marcados con un asterisco *

Nombre del sprint *

Sprint 2

Duración *

1 semana

Fecha de inicio *

10/3/2025 0:00

Fecha de finalización *

17/3/2025 0:00

Meta del sprint

En esta iteración se plantea modificar tablas menos complejas

Cancelar Iniciar

Metas

Todas las actividades

More

5

+

Añadir epic

/

☒ SUGD-56

Actualizado hace 7 minutos

Actividad

Mostrar: Comentarios

Añadir un comentario...

¡Tiene buena pinta!

¿Necesitas ayu

Consejo de expertos: pulsa **M** para comentar

CORDOVA MORILLO DAYANA LISBETH

hace 16 horas

Falta por finalizar la tarea de gestión de los filtros de búsqueda

Responder

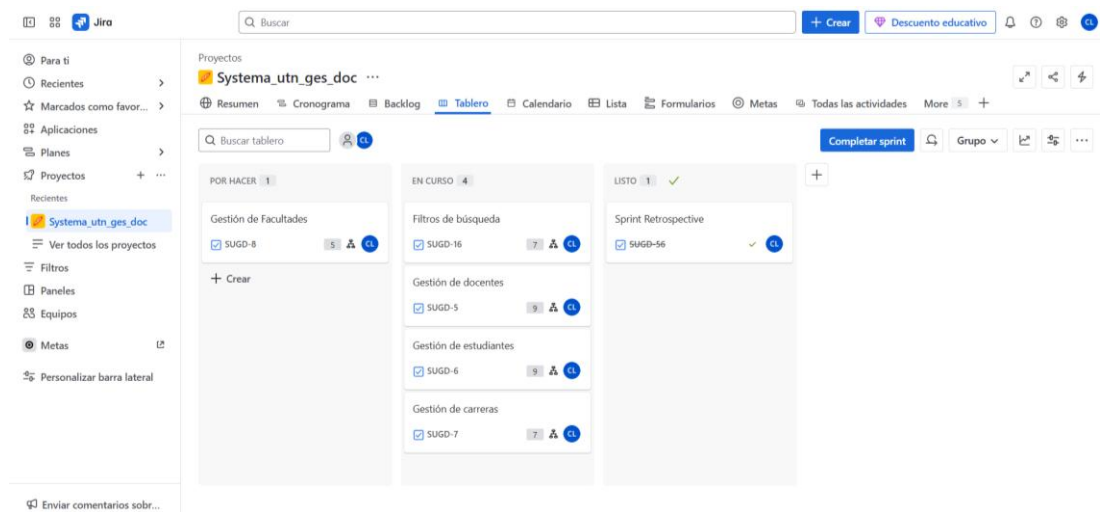
Editor

Eliminar

Nota. Elaboración Propia.

Reflejo del progreso del Sprint

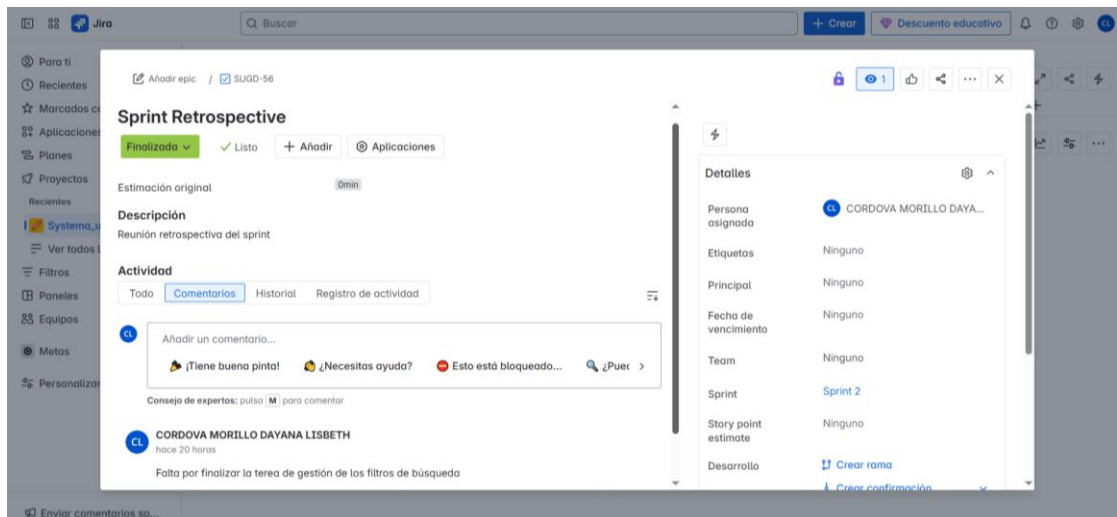
Figura 214.
Progreso del Sprint 2



Nota. Elaboración Propia.

Finalización del Sprint Retrospective

Figura 215.
Sprint retrospective del Sprint 2



Nota. Elaboración Propia.

Finalización del Issue #5

Figura 216.
Finalización del Issue #5

The screenshot shows the Jira interface for issue #5, titled "Gestión de docentes". The issue is marked as "Finalizada" (Completed) with a green checkmark. The description states: "Capacidad de visualizar, editar y crear docentes dentro del sistema". The "Actividades secundarias" section shows a progress bar at 100% completion. Below this, a table lists four tasks, all of which are marked as "FINALIZADA" (Completed).

Tipo	Clave	Resumen	Prio...	Pers...	Estado
	SUGD-33	Añadir tabla que muestre la información de los docentes	M.	S.	FINALIZADA
	SUGD-34	Añadir formulario para agregar docente	M.	S.	FINALIZADA
	SUGD-35	Añadir formulario para editar docente	M.	S.	FINALIZADA
	SUGD-36	Agregar varias carreras a un docente	M.	S.	FINALIZADA

The right sidebar shows details for the issue, including the assigned person (CORDOVA MORILLO DAYA...), labels (Ninguno), principal (Ninguno), date of completion (Ninguno), team (Ninguno), sprint (Sprint 2), story point estimate (9), and development (Crear rama, Crear confirmación).

Nota. Elaboración Propia.

Finalización del Issue #6

Figura 217.
Finalización del Issue #6

The screenshot shows the Jira interface for issue #6, titled "Gestión de estudiantes". The issue is marked as "Finalizada" (Completed) with a green checkmark. The description states: "Capacidad de visualizar, editar y crear estudiantes dentro del sistema". The "Actividades secundarias" section shows a progress bar at 100% completion. Below this, a table lists three tasks, all of which are marked as "FINALIZADA" (Completed).

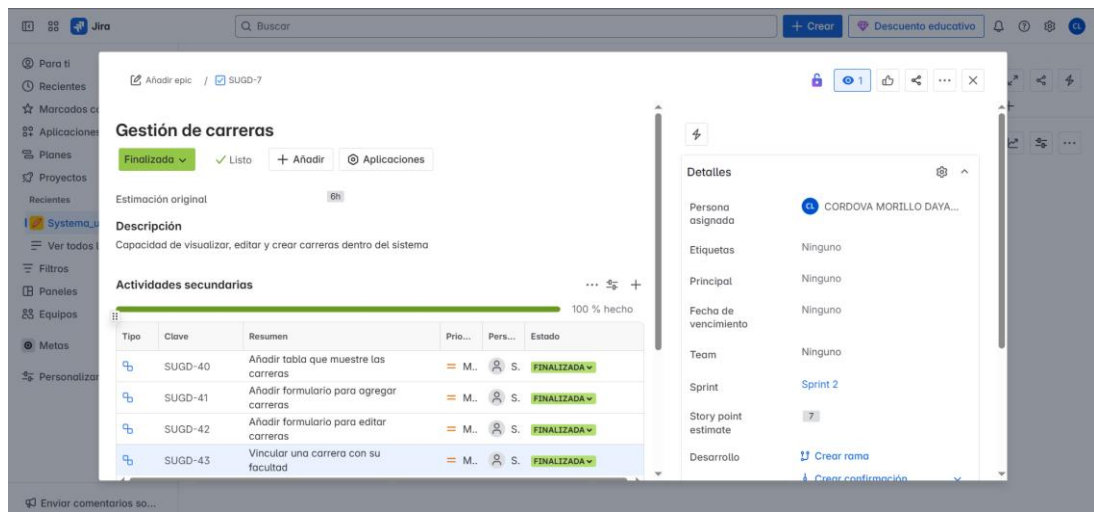
Tipo	Clave	Resumen	Prio...	Pers...	Estado
	SUGD-37	Añadir tabla para visualizar estudiantes	M.	S.	FINALIZADA
	SUGD-38	Añadir formulario para agregar estudiantes	M.	S.	FINALIZADA
	SUGD-39	Añadir formulario para editar estudiantes	M.	S.	FINALIZADA

The right sidebar shows details for the issue, including the assigned person (CORDOVA MORILLO DAYA...), labels (Ninguno), principal (Ninguno), date of completion (Ninguno), team (Ninguno), sprint (Sprint 2), story point estimate (9), and development (Crear rama, Crear confirmación).

Nota. Elaboración Propia.

Finalización del Issue #7

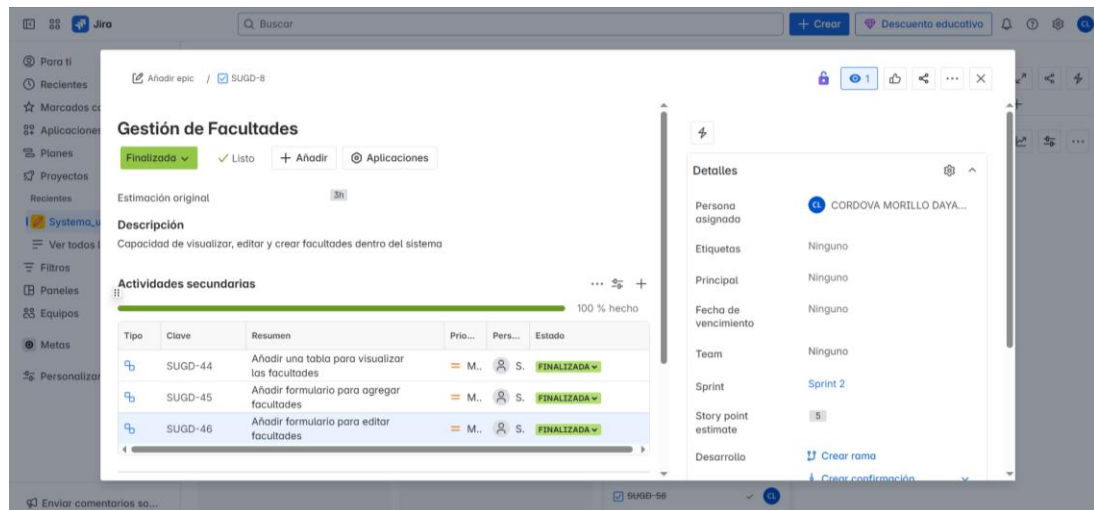
Figura 218.
Finalización del Issue #7



Nota. Elaboración Propia.

Finalización del Issue #8

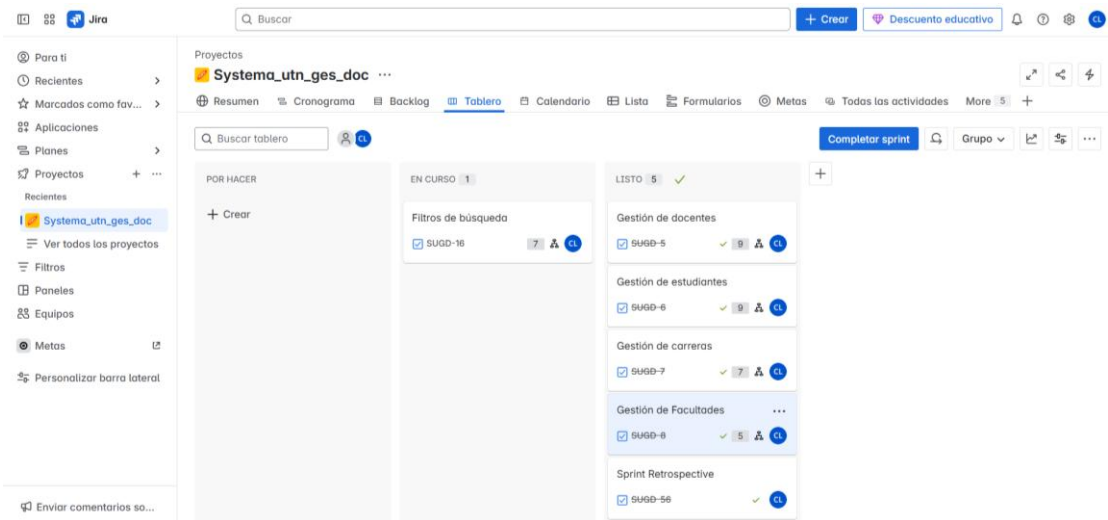
Figura 219.
Finalización del Issue #8



Nota. Elaboración Propia.

Cierre del Sprint

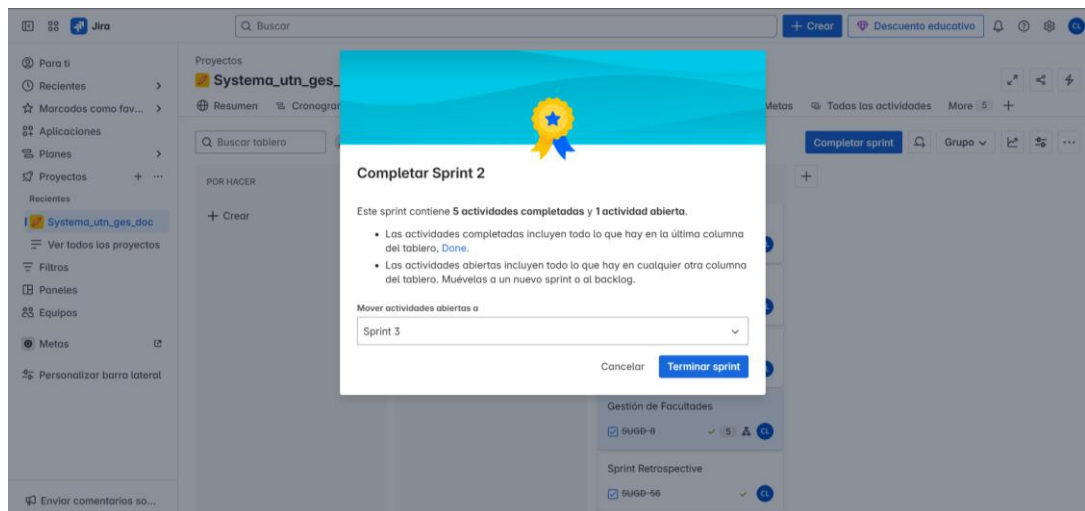
Figura 220.
Finalización del Sprint 2



Nota. Elaboración Propia.

Gestión de actividades faltantes enviadas al siguiente Sprint

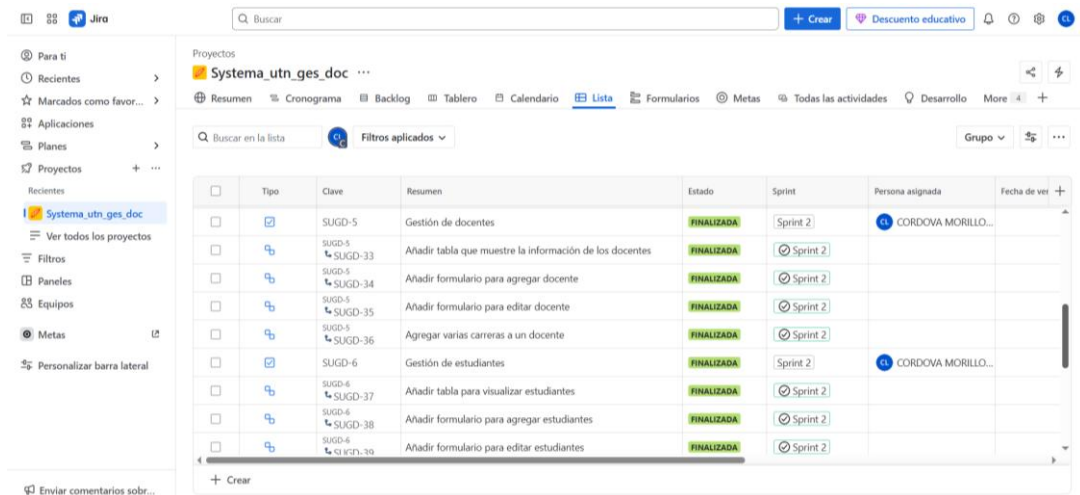
Figura 221.
Resumen sencillo del Sprint 2



Nota. Elaboración Propia.

Actividades finalizadas

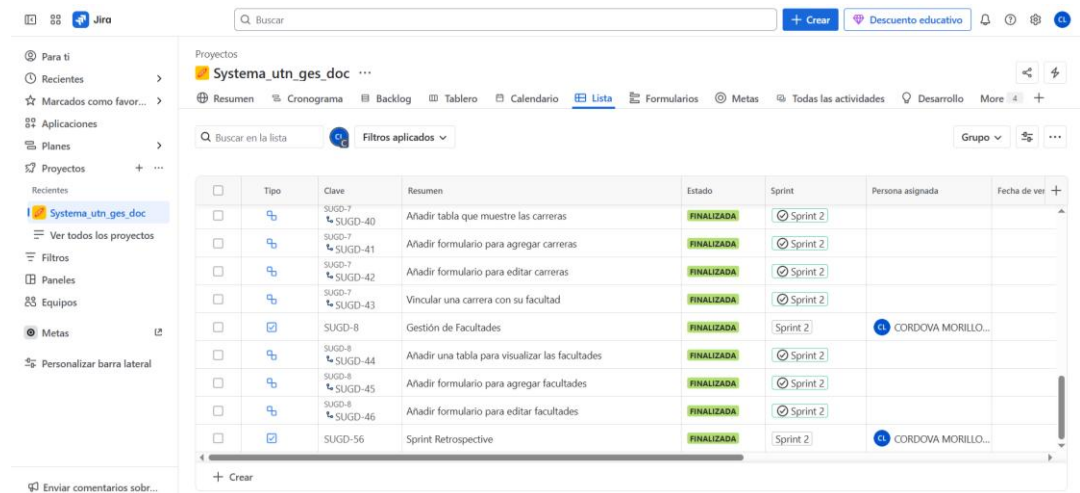
Figura 222.
Resumen avanzado del Sprint 2



	Tipo	Clave	Resumen	Estado	Sprint	Persona asignada	Fecha de ver
☐		SUGD-5	Gestión de docentes	FINALIZADA	Sprint 2	CORDOVA MORILLO...	
☐		SUGD-5 ↳ SUGD-33	Añadir tabla que muestre la información de los docentes	FINALIZADA	Sprint 2		
☐		SUGD-5 ↳ SUGD-34	Añadir formulario para agregar docente	FINALIZADA	Sprint 2		
☐		SUGD-5 ↳ SUGD-35	Añadir formulario para editar docente	FINALIZADA	Sprint 2		
☐		SUGD-5 ↳ SUGD-36	Agregar varias carreras a un docente	FINALIZADA	Sprint 2		
☐		SUGD-6	Gestión de estudiantes	FINALIZADA	Sprint 2	CORDOVA MORILLO...	
☐		SUGD-6 ↳ SUGD-37	Añadir tabla para visualizar estudiantes	FINALIZADA	Sprint 2		
☐		SUGD-6 ↳ SUGD-38	Añadir formulario para agregar estudiantes	FINALIZADA	Sprint 2		
☐		SUGD-6 ↳ SUGD-39	Añadir formulario para editar estudiantes	FINALIZADA	Sprint 2		

Nota. Elaboración Propia.

Figura 223.
Tareas completadas en el Sprint 2

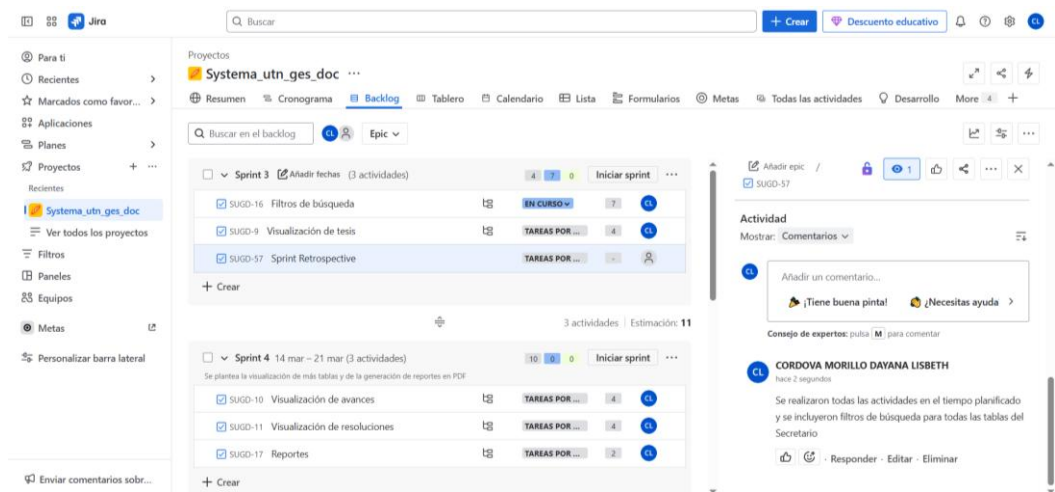


	Tipo	Clave	Resumen	Estado	Sprint	Persona asignada	Fecha de ver
☐		SUGD-7 ↳ SUGD-40	Añadir tabla que muestre las carreras	FINALIZADA	Sprint 2		
☐		SUGD-7 ↳ SUGD-41	Añadir formulario para agregar carreras	FINALIZADA	Sprint 2		
☐		SUGD-7 ↳ SUGD-42	Añadir formulario para editar carreras	FINALIZADA	Sprint 2		
☐		SUGD-7 ↳ SUGD-43	Vincular una carrera con su facultad	FINALIZADA	Sprint 2		
☐		SUGD-8	Gestión de Facultades	FINALIZADA	Sprint 2	CORDOVA MORILLO...	
☐		SUGD-8 ↳ SUGD-44	Añadir una tabla para visualizar las facultades	FINALIZADA	Sprint 2		
☐		SUGD-8 ↳ SUGD-45	Añadir formulario para agregar facultades	FINALIZADA	Sprint 2		
☐		SUGD-8 ↳ SUGD-46	Añadir formulario para editar facultades	FINALIZADA	Sprint 2		
☐		SUGD-56	Sprint Retrospective	FINALIZADA	Sprint 2	CORDOVA MORILLO...	

Nota. Elaboración Propia.

Comentarios del Sprint 2

Figura 224.
Comentarios del Sprint 2



Nota. Elaboración Propia.

Iteración 3

Inicio del Sprint 3

Prototipo del Sprint 3

Tabla 37.
Prototipo Sprint 3

	Sprint 3	
Historia de usuario	Tarea	Horas
Visualización de tesis	Añadir tabla para visualizar la información de las tesis	2

Nota. Elaboración propia

Figura 225.
Tablero Sprint 3

☐	▼ Sprint 3	📅 Añadir fechas (3 actividades)	4	7	0	Iniciar sprint	...
☒	SUGD-16	Filtros de búsqueda	🔗	EN CURSO	7	CL	
☒	SUGD-9	Visualización de tesis	🔗	TAREAS POR ...	4	CL	
☒	SUGD-57	Sprint Retrospective		TAREAS POR ...	-	👤	
+ Crear							

Nota. Elaboración Propia.

Figura 226.
Inicio del Sprint #3

Iniciar sprint

3 actividades se incluirán en este sprint.

Los campos obligatorios están marcados con un asterisco *

Nombre del sprint *

Sprint 3

Duración *

1 semana

Fecha de inicio *

13/3/2025 0:00

Fecha de finalización *

20/3/2025 0:00

Meta del sprint

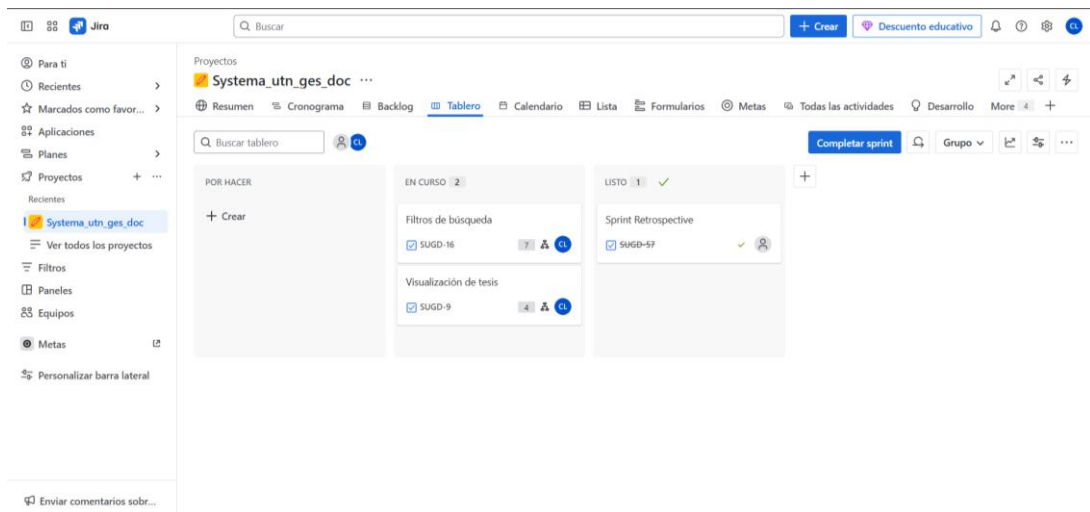
En este sprint se plantea iniciar con las vistas del coordinador

Cancelar Iniciar

Nota. Elaboración Propia.

Reflejo del progreso del Sprint

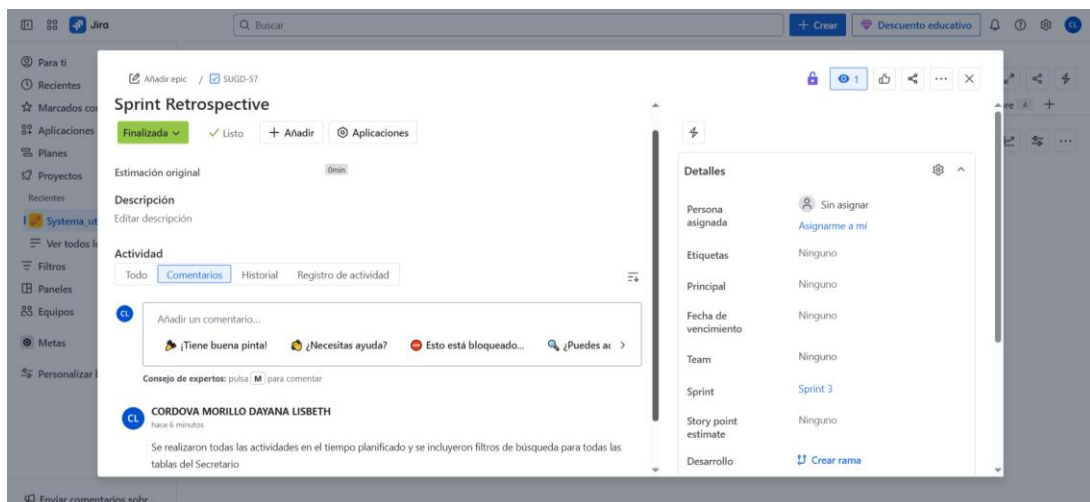
Figura 227.
Progreso del Sprint 3



Nota. Elaboración Propia.

Finalización del Sprint Retrospective

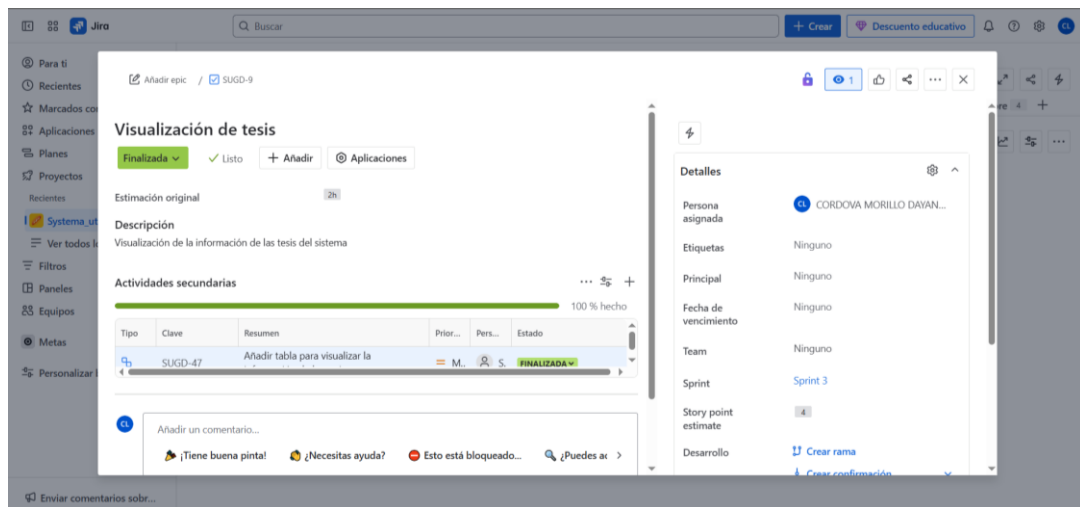
Figura 228.
Sprint retrospective del Sprint 3



Nota. Elaboración Propia.

Finalización del Issue #9

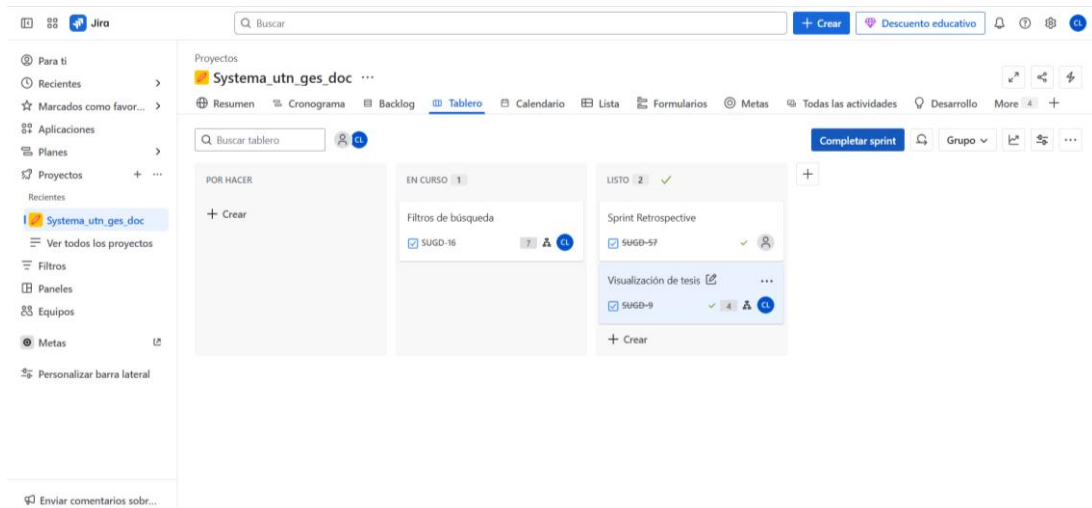
Figura 229.
Finalización del Issue #9



Nota. Elaboración Propia.

Cierre del Sprint

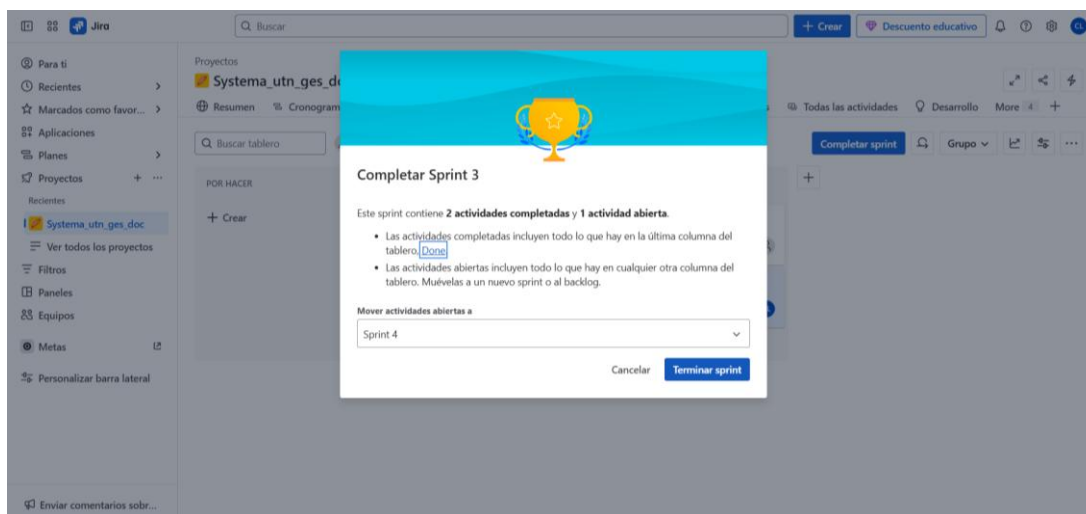
Figura 230.
Finalización del Sprint 3



Nota. Elaboración Propia.

Gestión de actividades faltantes enviadas al siguiente Sprint

Figura 231.
Resumen sencillo del Sprint 3



Nota. Elaboración Propia.

Actividades finalizadas

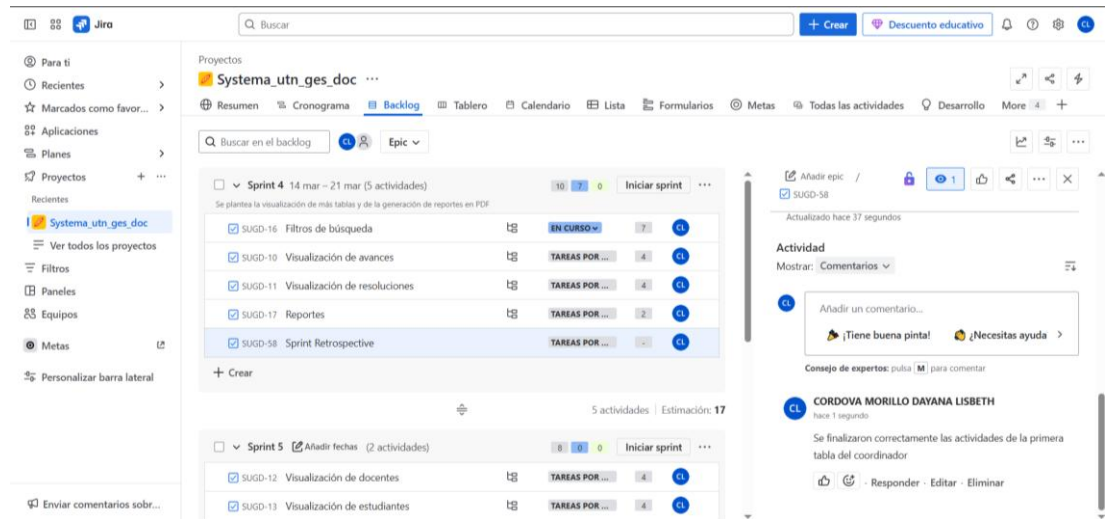
Figura 232.
Tareas completadas en el Sprint 3

	Tipo	Clave	Resumen	Estado	Sprint	Persona asignada	Fecha de ver
☐		SUGD-7 SUGD-43	Vincular una carrera con su facultad	FINALIZADA	Sprint 2		
☐		SUGD-8	Gestión de Facultades	FINALIZADA	Sprint 2	CL CORDOVA MORILLO...	
☐		SUGD-8 SUGD-44	Añadir una tabla para visualizar las facultades	FINALIZADA	Sprint 2		
☐		SUGD-8 SUGD-45	Añadir formulario para agregar facultades	FINALIZADA	Sprint 2		
☐		SUGD-8 SUGD-46	Añadir formulario para editar facultades	FINALIZADA	Sprint 2		
☐		SUGD-56	Sprint Retrospective	FINALIZADA	Sprint 2	CL CORDOVA MORILLO...	
☐		SUGD-57	Sprint Retrospective	FINALIZADA	Sprint 3		
☐		SUGD-9	Visualización de tesis	FINALIZADA	Sprint 3	CL CORDOVA MORILLO...	
☐		SUGD-8 SUGD-47	Añadir tabla para visualizar la información de las tesis	FINALIZADA	Sprint 3		

Nota. Elaboración Propia.

Comentarios del Sprint 3

Figura 233.
Comentarios del Sprint 3



Nota. Elaboración Propia.

Iteración 4

Inicio del Sprint 4

Prototipo del Sprint 4

Tabla 38.
Prototipo Sprint 4

Sprint 4		
Historia de usuario	Tarea	Horas
Visualización de avances	Añadir tabla para visualizar la información de los avances	2
Visualización de resoluciones	Añadir tabla para visualizar la información de las resoluciones	2
Reportes	Añadir botón para imprimir en PDF las columnas visibles de una tabla	5

Nota. Elaboración propia

Figura 234.
Tablero Sprint 4

☐ **Sprint 4** 14 mar – 21 mar (5 actividades) 10 7 0 Iniciar sprint ...

Se plantea la visualización de más tablas y de la generación de reportes en PDF

☒ SUGD-16	Filtros de búsqueda		EN CURSO ▾	7	
☒ SUGD-10	Visualización de avances		TAREAS POR ...	4	
☒ SUGD-11	Visualización de resoluciones		TAREAS POR ...	4	
☒ SUGD-17	Reportes		TAREAS POR ...	2	
☒ SUGD-58	Sprint Retrospective		TAREAS POR ...	-	

+ Crear

Nota. Elaboración Propia.

Figura 235.
Inicio del Sprint #4

Iniciar sprint

5 actividades se incluirán en este sprint.

Los campos obligatorios están marcados con un asterisco *

Nombre del sprint *

Sprint 4

Duración *

1 semana

Fecha de inicio *

14/3/2025 0:00

Fecha de inicio prevista: 14/3/2025, 0:00
La fecha de inicio de un sprint afecta a la velocidad y al alcance de los informes. [Obtén más información.](#)

Fecha de finalización *

21/3/2025 0:00

Meta del sprint

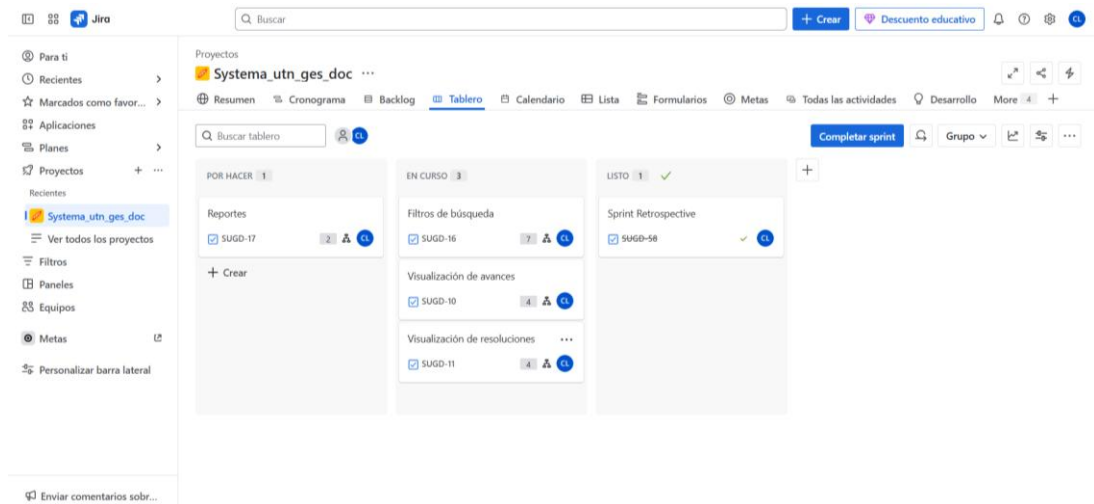
Se plantea la visualización de más tablas y de la generación de reportes en PDF

Cancelar Iniciar

Nota. Elaboración Propia.

Reflejo del progreso del Sprint

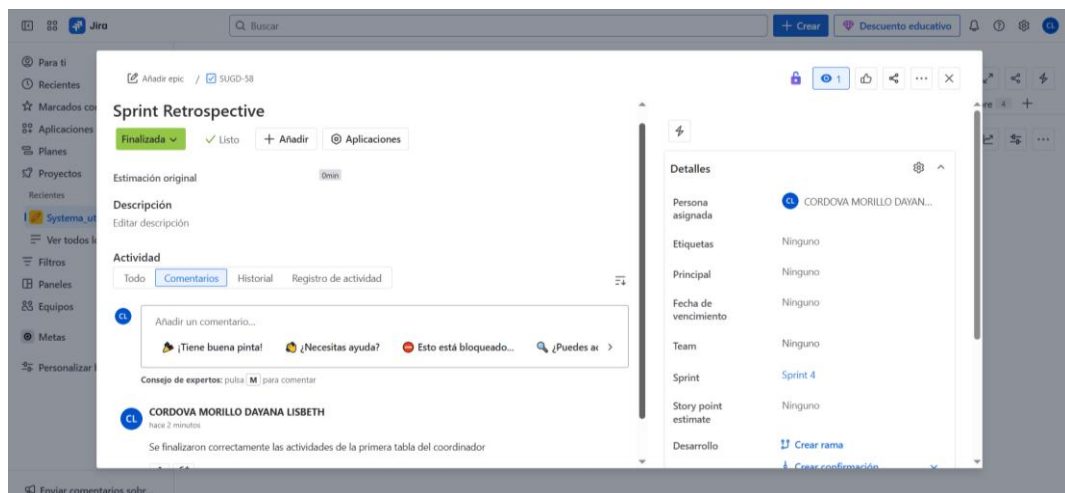
Figura 236.
Progreso del Sprint 4



Nota. Elaboración Propia.

Finalización del Sprint Retrospective

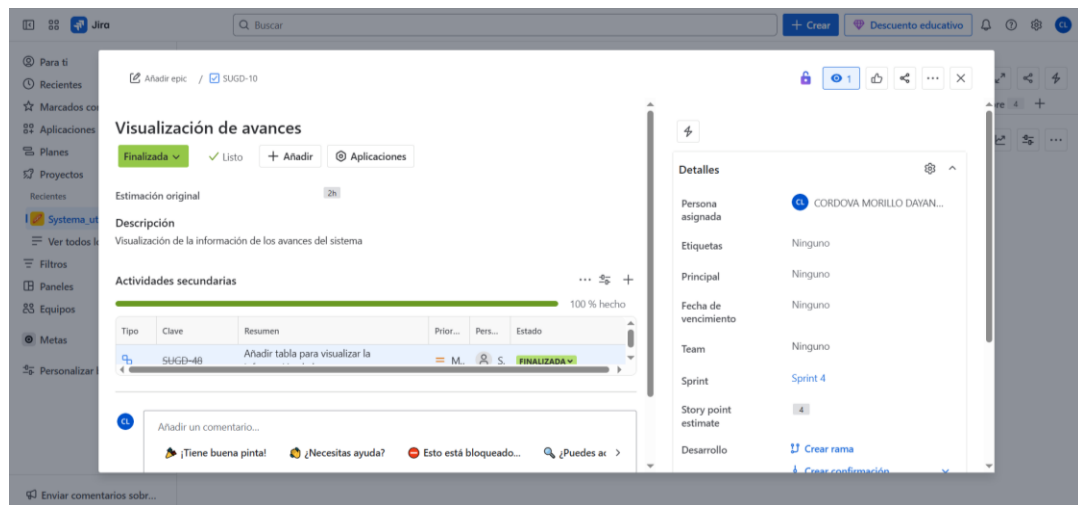
Figura 237.
Sprint retrospective del Sprint 4



Nota. Elaboración Propia.

Finalización del Issue #10

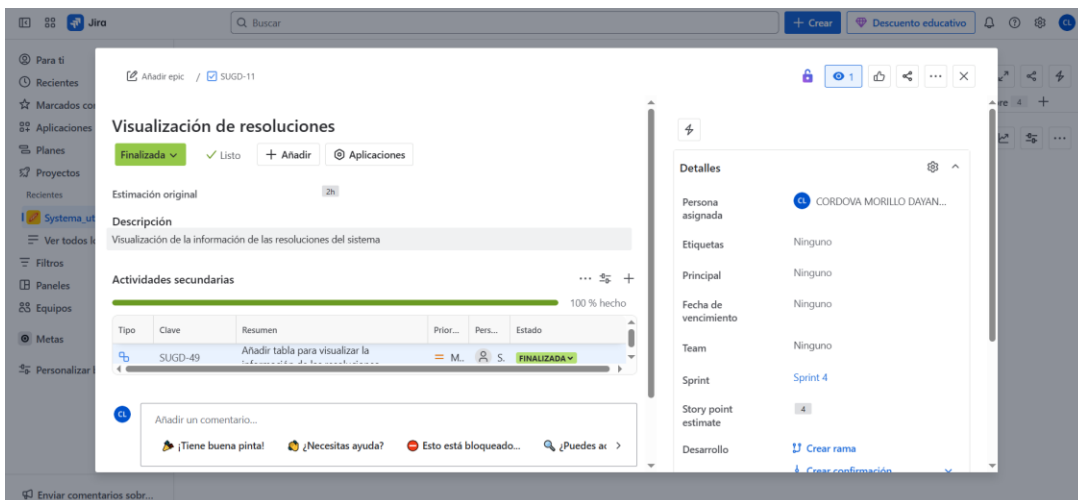
Figura 238.
Finalización del Issue #10



Nota. Elaboración Propia.

Finalización del Issue #11

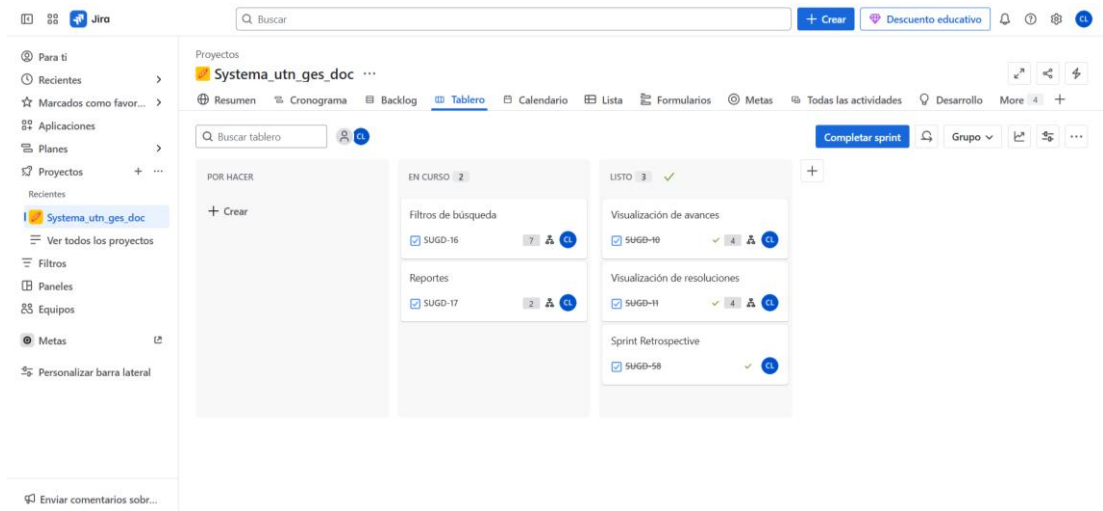
Figura 239.
Finalización del Issue #11



Nota. Elaboración Propia.

Cierre del Sprint

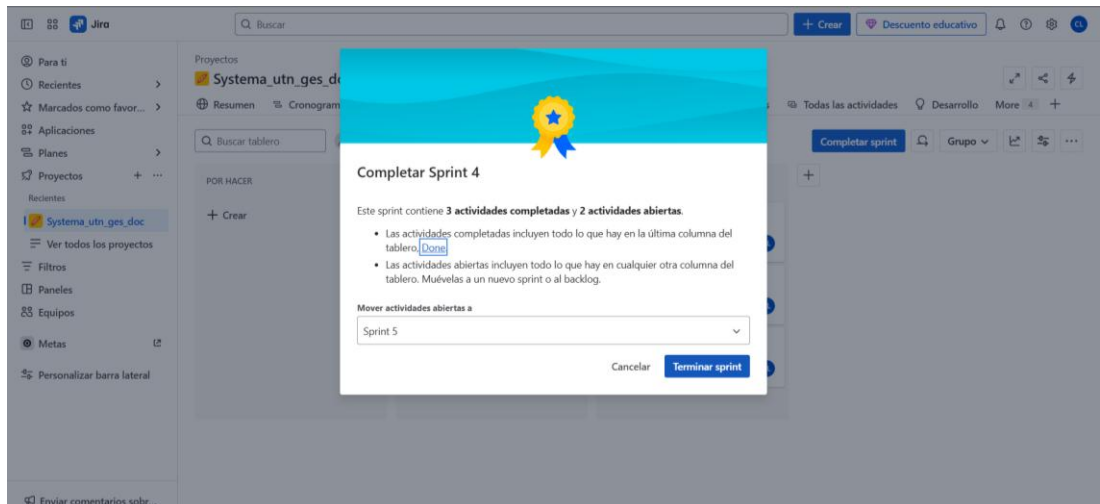
Figura 240.
Finalización del Sprint 4



Nota. Elaboración Propia.

Gestión de actividades faltantes enviadas al siguiente Sprint

Figura 241.
Resumen sencillo del Sprint 4



Nota. Elaboración Propia.

Actividades finalizadas

Figura 242.
Tareas completadas en el Sprint 4

The screenshot shows the Jira interface for the project 'Systema_utm_ges_doc'. The 'List' view is selected, displaying a table of tasks. The tasks are all marked as 'FINALIZADA' (Completed) and assigned to 'CORDOVA MORILLO...'. The tasks include Sprint Retrospectives and the addition of various tables for visualizing resolutions, advances, and theses.

	Tipo	Clave	Resumen	Estado	Sprint	Persona asignada	Fecha de ver
☐		SUGD-58	Sprint Retrospective	FINALIZADA	Sprint 4	CORDOVA MORILLO...	
☐		SUGD-11	Añadir tabla para visualizar la información de las resoluciones	FINALIZADA	Sprint 4	CORDOVA MORILLO...	
☐		SUGD-11	Visualización de resoluciones	FINALIZADA	Sprint 4	CORDOVA MORILLO...	
☐		SUGD-10	Añadir tabla para visualizar la información de los avances	FINALIZADA	Sprint 4	CORDOVA MORILLO...	
☐		SUGD-10	Visualización de avances	FINALIZADA	Sprint 4	CORDOVA MORILLO...	
☐		SUGD-9	Añadir tabla para visualizar la información de las tesis	FINALIZADA	Sprint 3	CORDOVA MORILLO...	
☐		SUGD-9	Visualización de tesis	FINALIZADA	Sprint 3	CORDOVA MORILLO...	
☐		SUGD-57	Sprint Retrospective	FINALIZADA	Sprint 3	CORDOVA MORILLO...	
☐		SUGD-56	Sprint Retrospective	FINALIZADA	Sprint 2	CORDOVA MORILLO...	

Nota. Elaboración Propia.

Figura 243.
Comentarios del Sprint 4

The screenshot shows the Jira interface for the project 'Systema_utm_ges_doc' in the 'Backlog' view. It displays tasks for Sprint 5 and Sprint 6. A comment is visible on the task 'SUGD-59 Sprint Retrospective'.

Sprint 5 (5 actividades)

- SUGD-16 Filtros de búsqueda: EN CURSO (7)
- SUGD-17 Reportes: EN CURSO (2)
- SUGD-12 Visualización de docentes: TAREAS POR... (4)
- SUGD-13 Visualización de estudiantes: TAREAS POR... (4)
- SUGD-59 Sprint Retrospective: TAREAS POR... (-)

Sprint 6 (2 actividades)

- SUGD-14 Visualización de carreras: TAREAS POR... (4)
- SUGD-15 Visualización de Facultades: TAREAS POR... (4)

Comment on SUGD-59:

¡Tiene buena pinta! ¡Necesitas ayuda >

Consejo de expertos: pulsa **M** para comentar

CORDOVA MORILLO DAYANA LISBETH hace 3 segundos
Se completaron las vistas: tablas planificadas en el sprint
Responder · Editar · Eliminar

CORDOVA MORILLO DAYANA LISBETH hace 1 minuto
Se inició con los reportes para las vistas de Secretario y Coordinador
Responder · Editar · Eliminar

Nota. Elaboración Propia.

Iteración 5

Inicio del Sprint 5

Prototipo del Sprint 5

Tabla 39.
Prototipo sprint 5

	Sprint 5	
Historia de usuario	Tarea	Horas
Visualización de docentes	Añadir tabla para visualizar la información de los docentes	2
Visualización de estudiantes	Añadir tabla para visualizar la información de los estudiantes	2

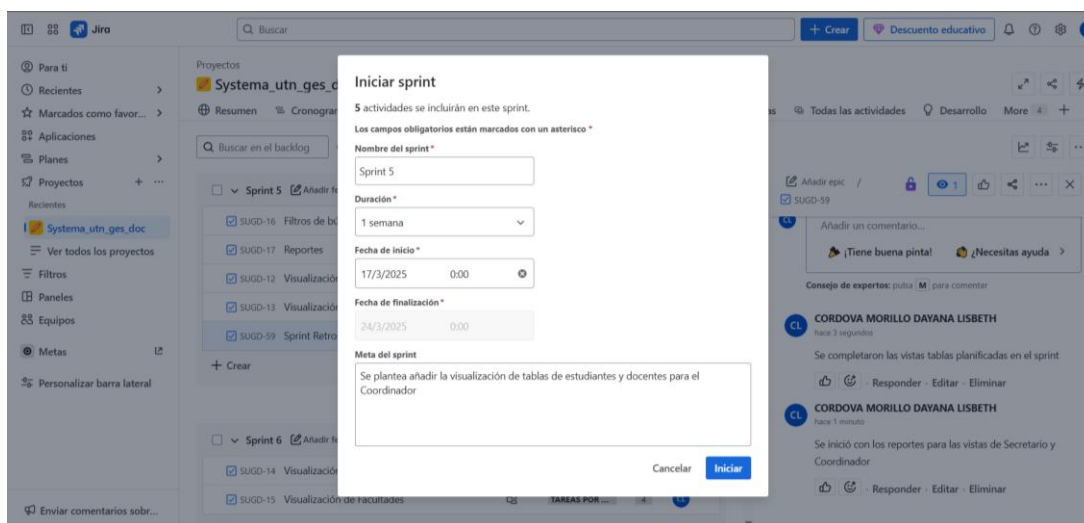
Nota. Elaboración propia

Figura 244.
Tablero Sprint 5

☐ Sprint 5 Añadir fechas (5 actividades)	8 9 0	Iniciar sprint	...
☒ SUGD-16 Filtros de búsqueda	EN CURSO	7	CL
☒ SUGD-17 Reportes	EN CURSO	2	CL
☒ SUGD-12 Visualización de docentes	TAREAS POR ...	4	CL
☒ SUGD-13 Visualización de estudiantes	TAREAS POR ...	4	CL
☒ SUGD-59 Sprint Retrospective	TAREAS POR ...	-	CL
+ Crear			

Nota. Elaboración Propia.

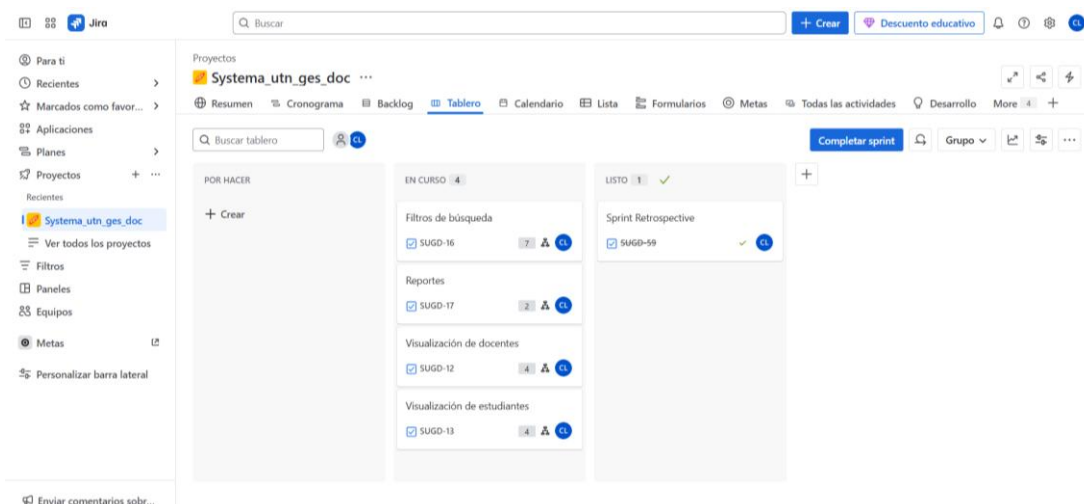
Figura 245.
Inicio del Sprint #5



Nota. Elaboración Propia.

Reflejo del progreso del Sprint

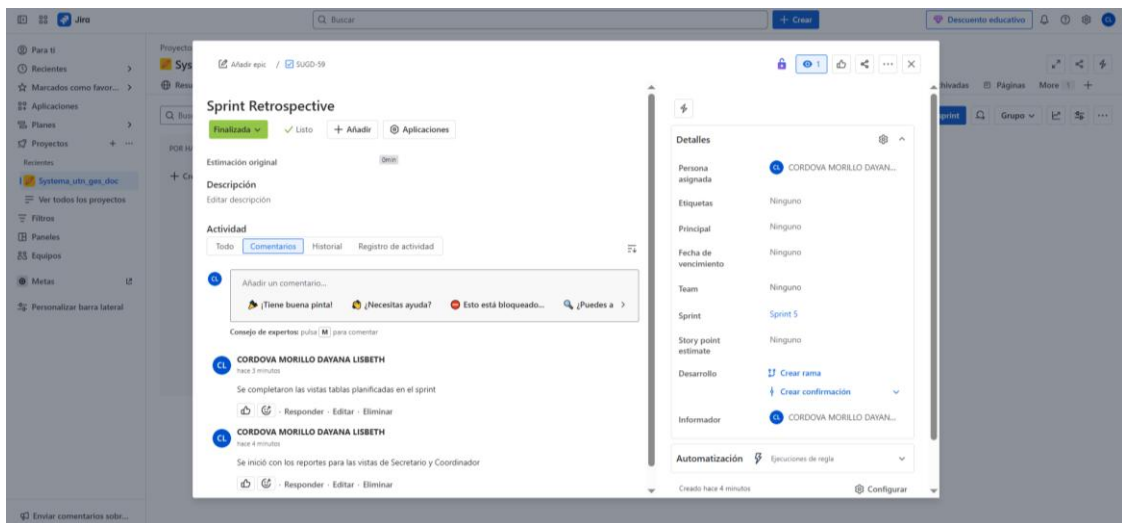
Figura 246.
Progreso del Sprint 5



Nota. Elaboración Propia.

Finalización del Sprint Retrospective

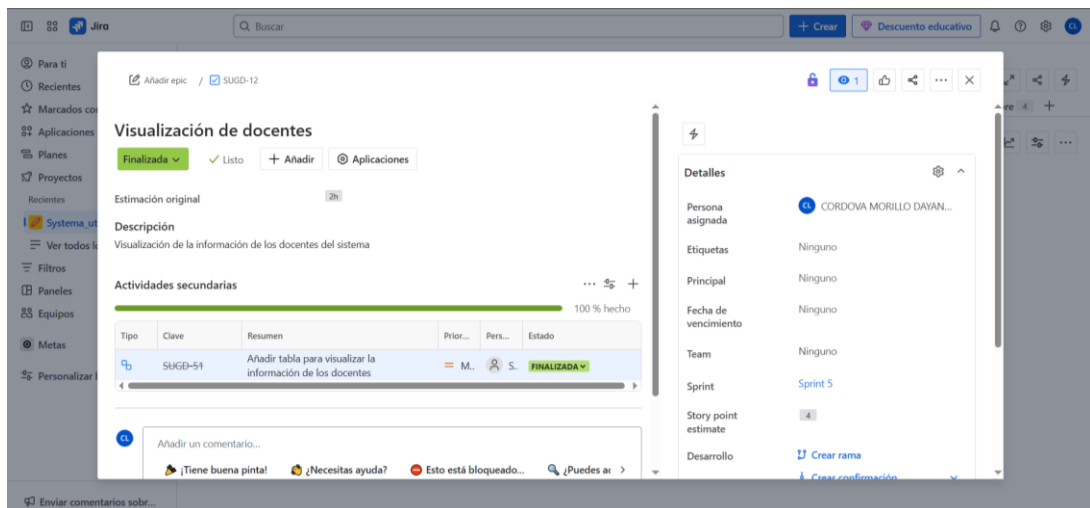
Figura 247.
Sprint retrospective del Sprint 5



Nota. Elaboración Propia.

Finalización del Issue #13

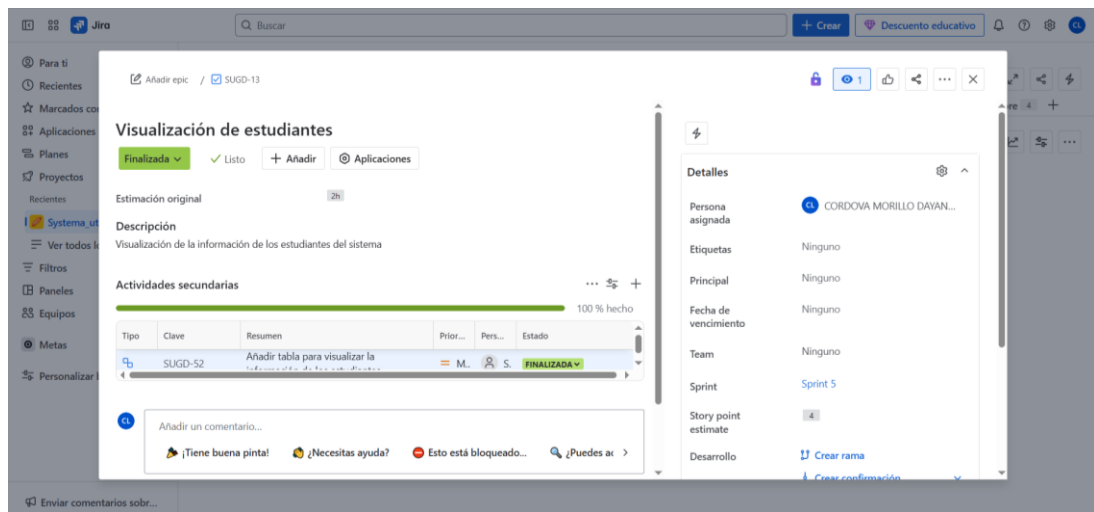
Figura 248.
Finalización del Issue #13



Nota. Elaboración Propia.

Finalización del Issue #14

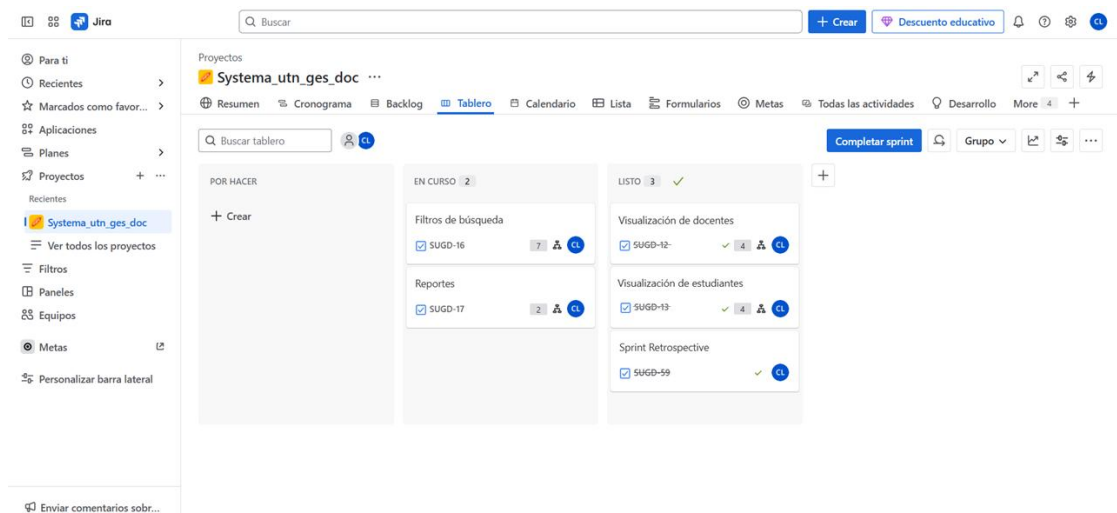
Figura 249.
Finalización del Issue #14



Nota. Elaboración Propia.

Cierre del Sprint

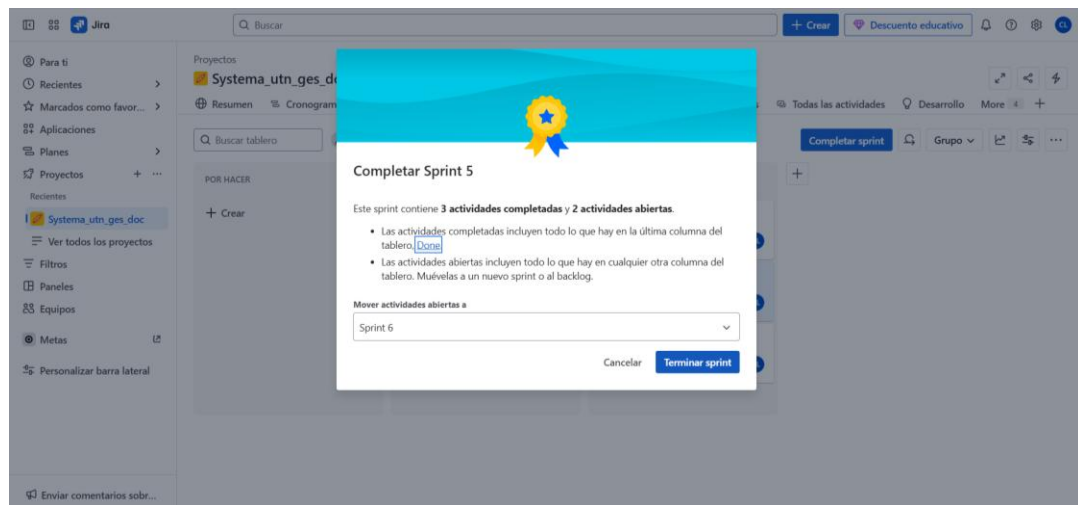
Figura 250.
Finalización del Sprint 5



Nota. Elaboración Propia.

Gestión de actividades enviadas al siguiente Sprint

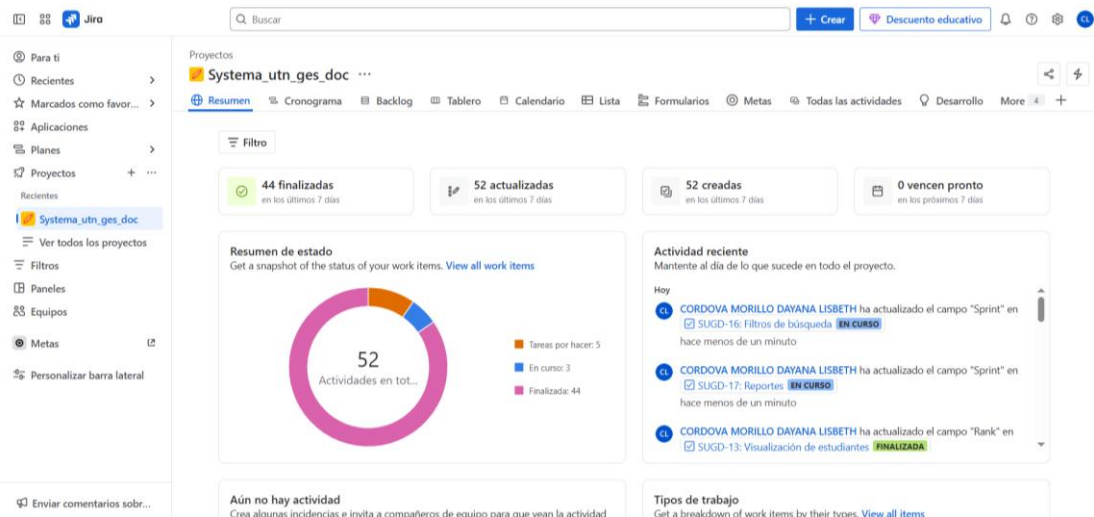
Figura 251.
Resumen sencillo del Sprint 5



Nota. Elaboración Propia.

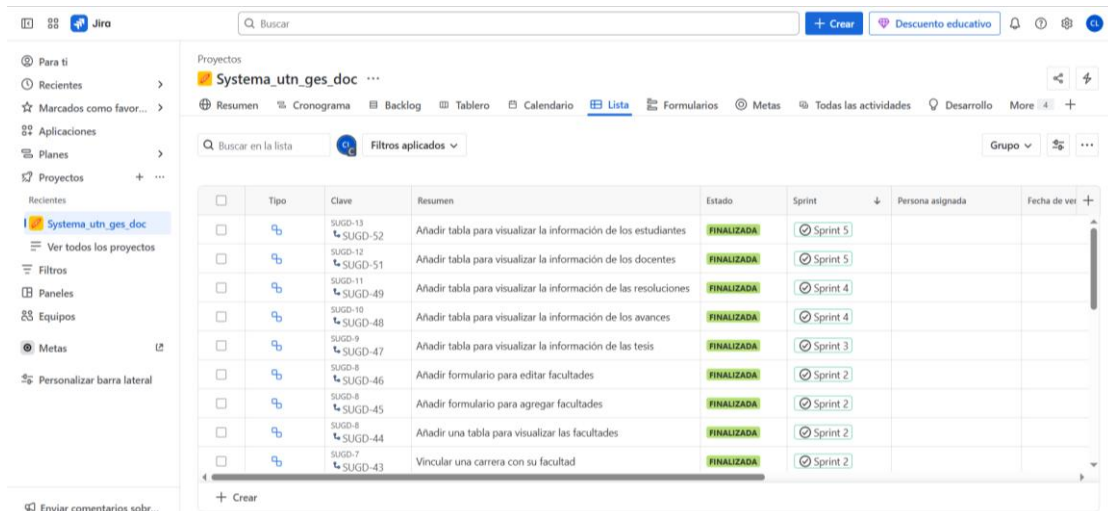
Actividades finalizadas

Figura 252.
Resumen avanzado del Sprint 5



Nota. Elaboración Propia.

Figura 253.
Tareas completadas en el Sprint 5

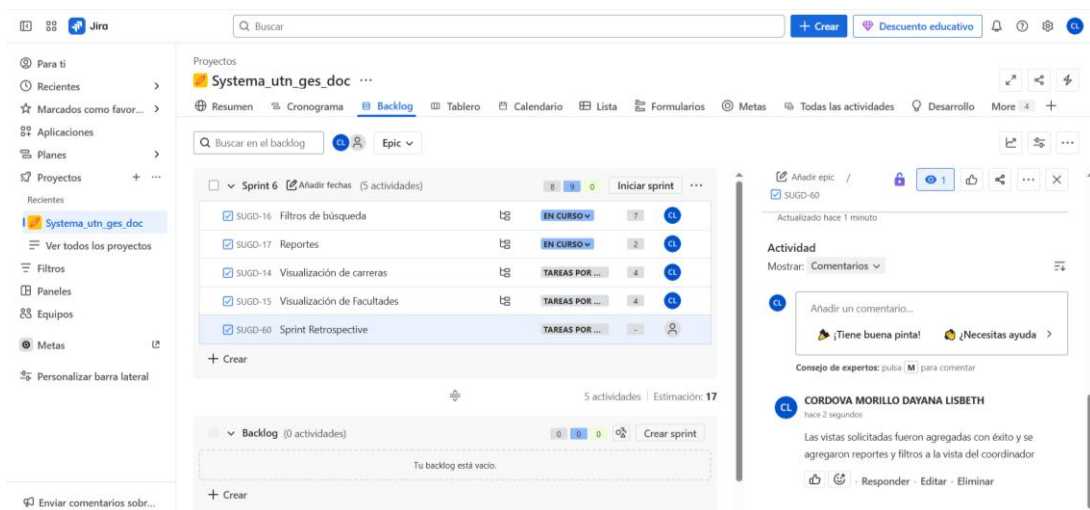


	Tipo	Clave	Resumen	Estado	Sprint	Persona asignada	Fecha de ver
☐	🔗	SUGD-13	Añadir tabla para visualizar la información de los estudiantes	FINALIZADA	🕒 Sprint 5		
☐	🔗	SUGD-52	Añadir tabla para visualizar la información de los docentes	FINALIZADA	🕒 Sprint 5		
☐	🔗	SUGD-51	Añadir tabla para visualizar la información de las resoluciones	FINALIZADA	🕒 Sprint 4		
☐	🔗	SUGD-49	Añadir tabla para visualizar la información de los avances	FINALIZADA	🕒 Sprint 4		
☐	🔗	SUGD-10	Añadir tabla para visualizar la información de las tesis	FINALIZADA	🕒 Sprint 3		
☐	🔗	SUGD-48	Añadir formulario para editar facultades	FINALIZADA	🕒 Sprint 2		
☐	🔗	SUGD-47	Añadir formulario para agregar facultades	FINALIZADA	🕒 Sprint 2		
☐	🔗	SUGD-8	Añadir una tabla para visualizar las facultades	FINALIZADA	🕒 Sprint 2		
☐	🔗	SUGD-45	Añadir una tabla para visualizar las facultades	FINALIZADA	🕒 Sprint 2		
☐	🔗	SUGD-8	Vincular una carrera con su facultad	FINALIZADA	🕒 Sprint 2		
☐	🔗	SUGD-44					
☐	🔗	SUGD-7					
☐	🔗	SUGD-43					

Nota. Elaboración Propia.

Comentarios del Sprint 5

Figura 254.
Comentarios del Sprint 5



Backlog (5 actividades)

- ☒ SUGD-16 Filtros de búsqueda **EN CURSO** 7
- ☒ SUGD-17 Reportes **EN CURSO** 2
- ☒ SUGD-14 Visualización de carreras **TAREAS POR COMPLETAR** 4
- ☒ SUGD-15 Visualización de Facultades **TAREAS POR COMPLETAR** 4
- ☒ SUGD-60 Sprint Retrospective **TAREAS POR COMPLETAR** 1

Actividad

Mostrar: Comentarios

Añadir un comentario...

¡Tiene buena pinta! ¡Necesitas ayuda?

Consejo de expertos: pulsa M para comentar

CORDOVA MORILLO DAYANA LISBETH
hace 2 segundos

Las vistas solicitadas fueron agregadas con éxito y se agregaron reportes y filtros a la vista del coordinador

Responder · Editar · Eliminar

Nota. Elaboración Propia.

Iteración 6

Inicio del Sprint 6

Prototipo del Sprint 6

Tabla 40.
Prototipo Sprint 6

Sprint 6		
Historia de usuario	Tarea	Horas
Visualización de carreras	Añadir tabla para visualizar la información de las carreras	2
Visualización de Facultades	Añadir tabla para visualizar la información de las facultades	2

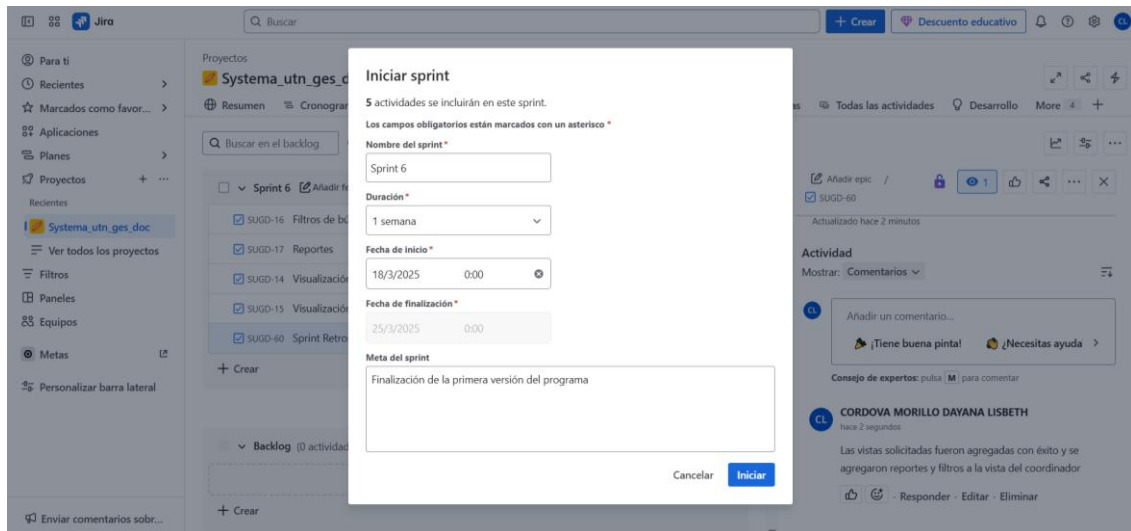
Nota. Elaboración propia

Figura 255.
Tablero Sprint 6

☐	▼ Sprint 6	✎ Añadir fechas (5 actividades)	8	9	0	Iniciar sprint	...
☒	SUGD-16	Filtros de búsqueda	🔖	EN CURSO ▼	7	CL	
☒	SUGD-17	Reportes	🔖	EN CURSO ▼	2	CL	
☒	SUGD-14	Visualización de carreras	🔖	TAREAS POR ...	4	CL	
☒	SUGD-15	Visualización de Facultades	🔖	TAREAS POR ...	4	CL	
☒	SUGD-60	Sprint Retrospective		TAREAS POR ...	-	👤	
+ Crear							

Nota. Elaboración Propia.

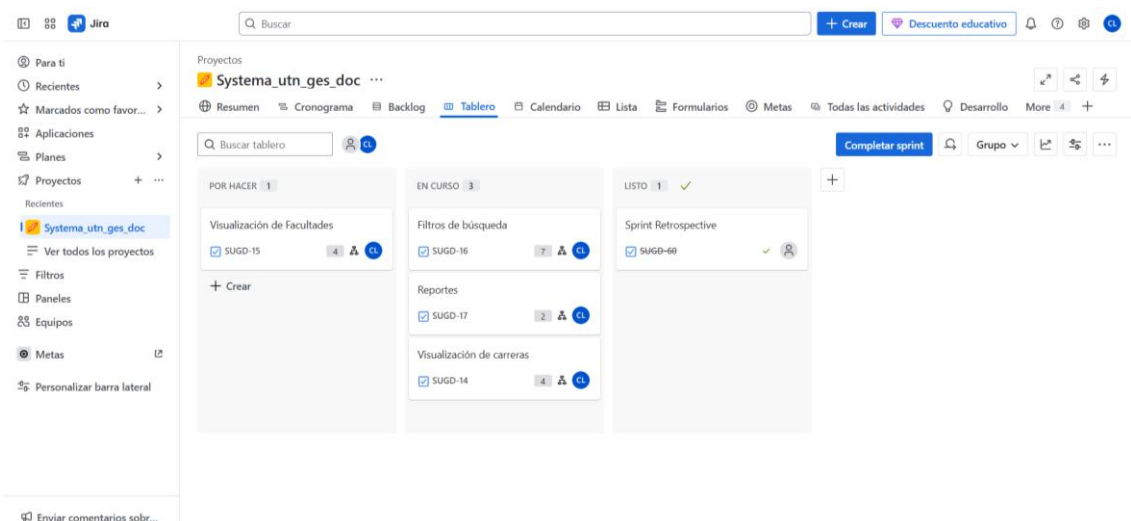
Figura 256.
Inicio del Sprint #6



Nota. Elaboración Propia.

Reflejo del progreso del Sprint

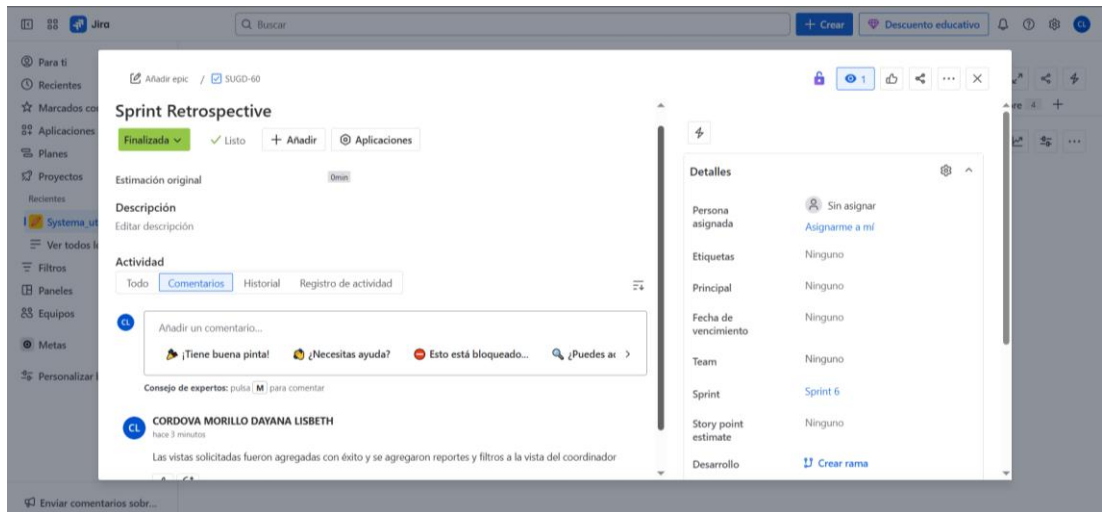
Figura 257.
Progreso del Sprint 6



Nota. Elaboración Propia.

Finalización del Sprint Retrospective

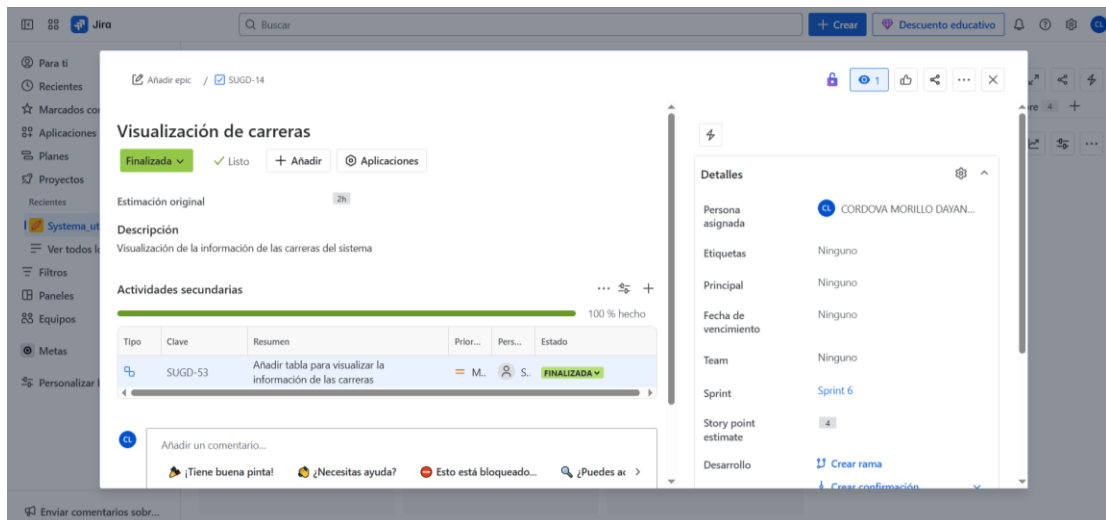
Figura 258.
Sprint retrospective del Sprint 6



Nota. Elaboración Propia.

Finalización del Issue #15

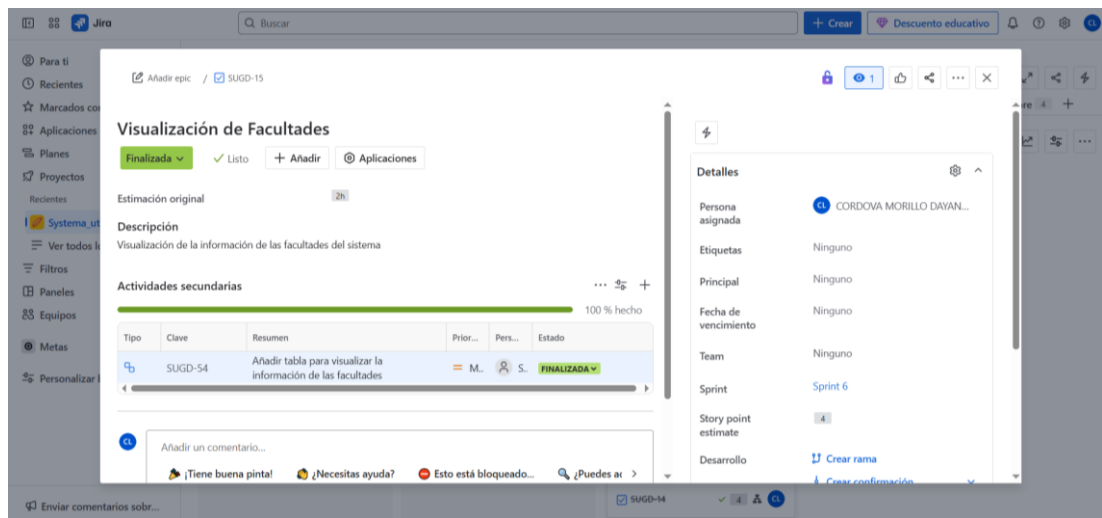
Figura 259.
Finalización del Issue #15



Nota. Elaboración Propia.

Finalización del Issue #16

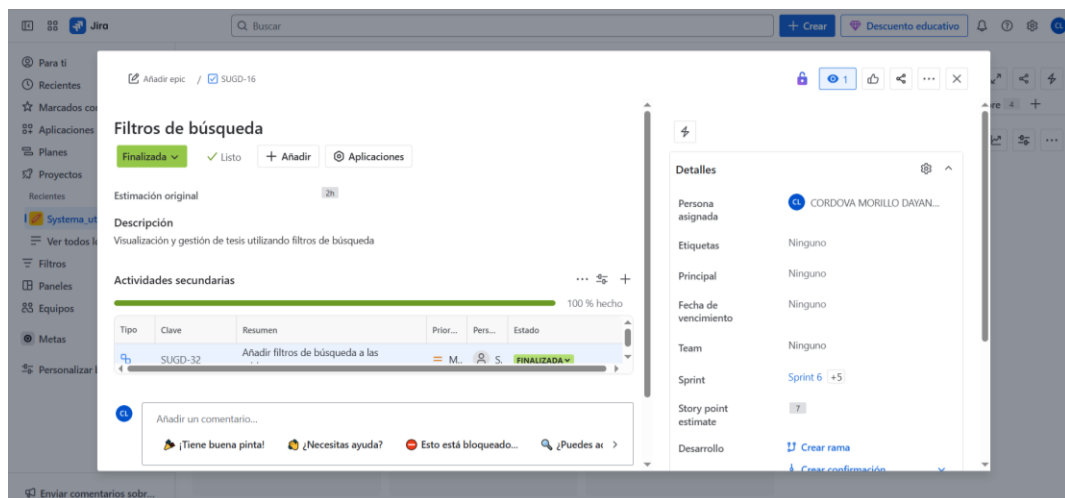
Figura 260.
Finalización del Issue #16



Nota. Elaboración Propia.

Finalización del Issue #4

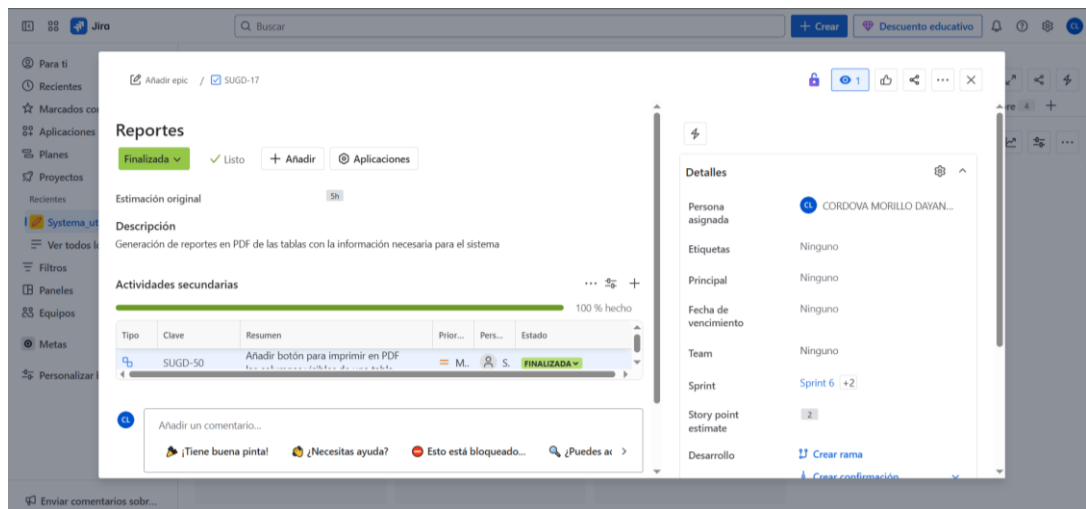
Figura 261.
Finalización del Issue #4



Nota. Elaboración Propia.

Finalización del Issue #12

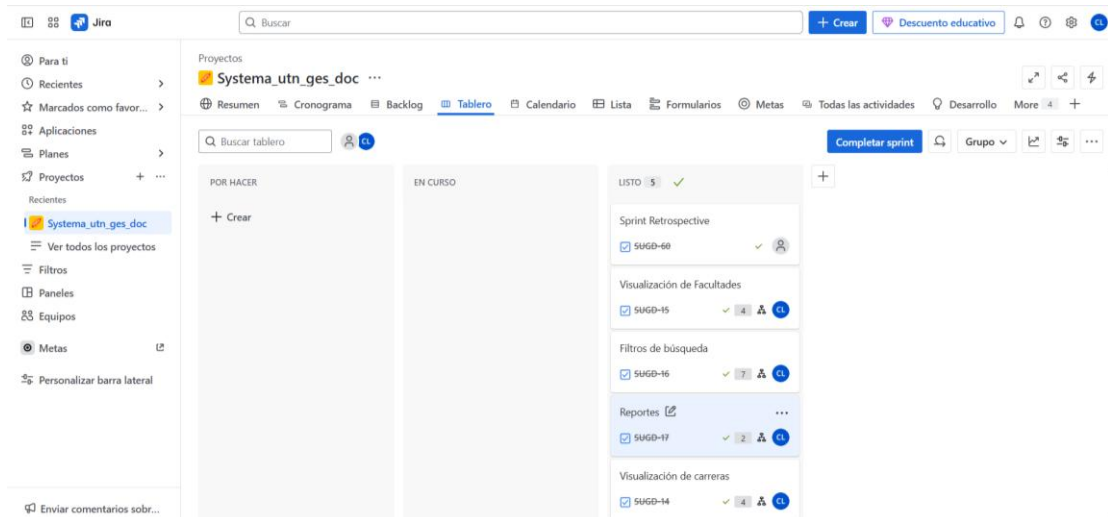
Figura 262.
Finalización del Issue #12



Nota. Elaboración Propia.

Cierre del Sprint

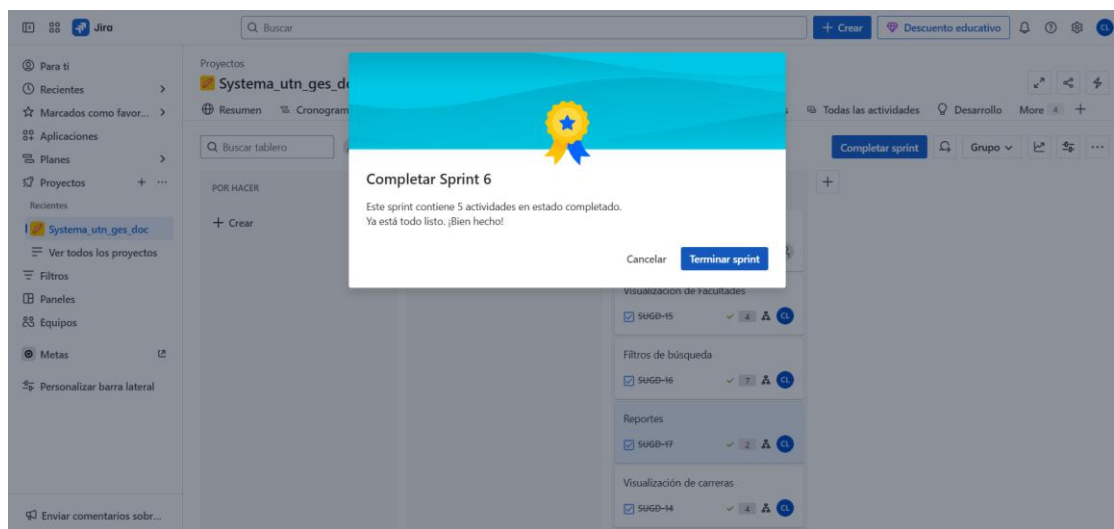
Figura 263.
Finalización del Sprint 6



Nota. Elaboración Propia.

Actividades finalizadas en el Sprint

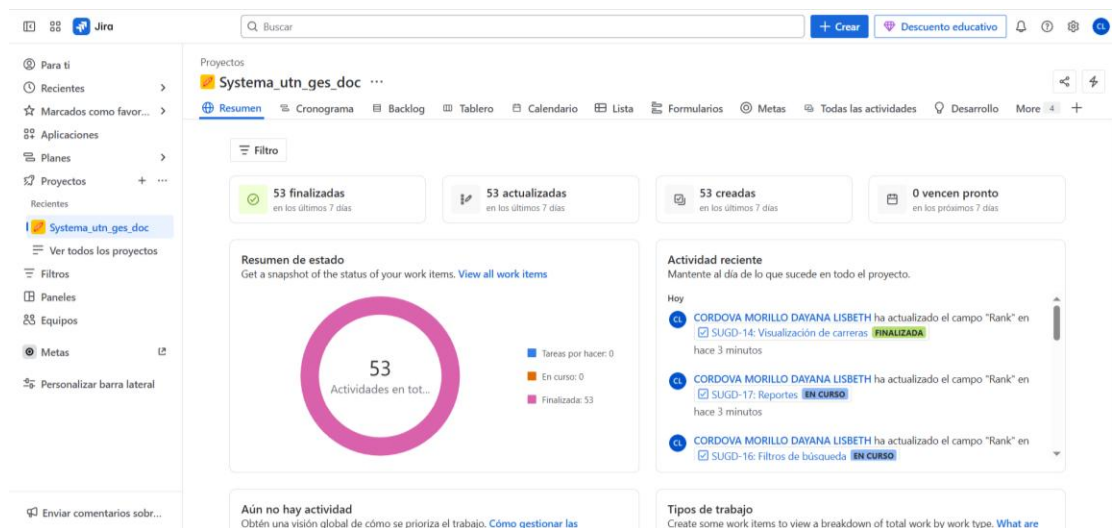
Figura 264.
Resumen sencillo del Sprint 6



Nota. Elaboración Propia.

Actividades finalizadas

Figura 265.
Resumen avanzado del Sprint 6



Nota. Elaboración Propia.

Figura 266.
Tareas completadas en el Sprint 6

	Tipo	Clave	Resumen	Estado	Sprint	Persona asignada	Fecha de ver
☐		SUGD-14	Añadir tabla para visualizar la información de las carreras	FINALIZADA	Sprint 6	CORDOVA MORILLO...	
☐		SUGD-14	Visualización de carreras	FINALIZADA	Sprint 6	CORDOVA MORILLO...	
☐		SUGD-17	Añadir botón para imprimir en PDF las columnas visibles de ...	FINALIZADA	Sprint 6	CORDOVA MORILLO...	
☐		SUGD-17	Reportes	FINALIZADA	Sprint 6	CORDOVA MORILLO...	
☐		SUGD-16	Añadir filtros de búsqueda a las tablas	FINALIZADA	Sprint 6	CORDOVA MORILLO...	
☐		SUGD-16	Filtros de búsqueda	FINALIZADA	Sprint 6	CORDOVA MORILLO...	
☐		SUGD-15	Añadir tabla para visualizar la información de las facultades	FINALIZADA	Sprint 6	CORDOVA MORILLO...	
☐		SUGD-15	Visualización de Facultades	FINALIZADA	Sprint 6	CORDOVA MORILLO...	
☐		SUGD-60	Sprint Retrospective	FINALIZADA	Sprint 6		

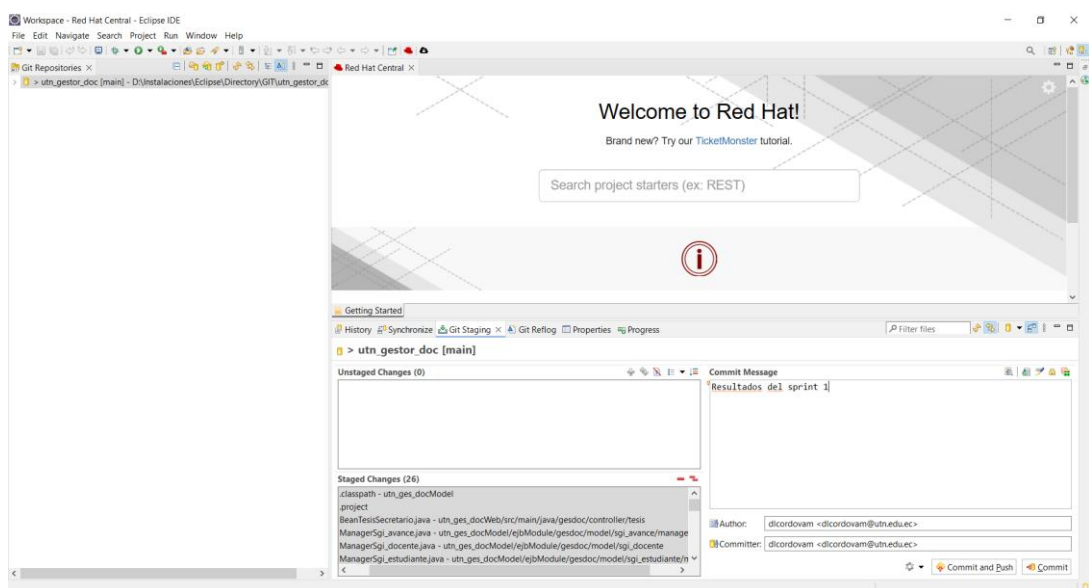
Nota. Elaboración Propia.

Fase 3. Post – Juego

Etiquetado de la versión de lanzamiento: Sprint 1

Commit subido desde eclipse

Figura 267.
Commit inicial desde Eclipse

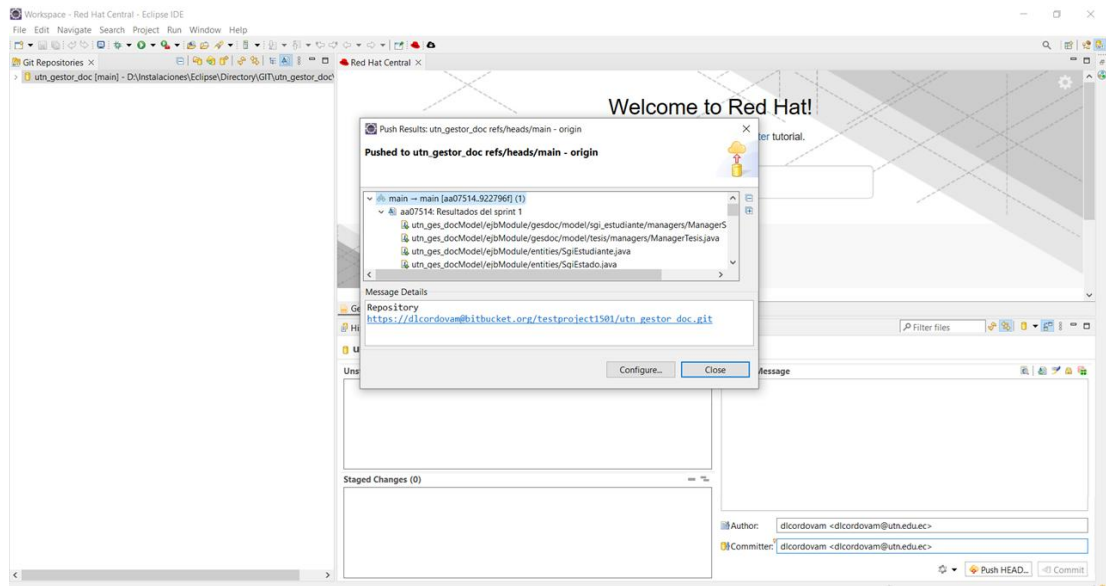


Nota. Elaboración Propia.

Cambios subidos a Bitbucket

Figura 268.

Cambios del Sprint 1 subidos a Bitbucket

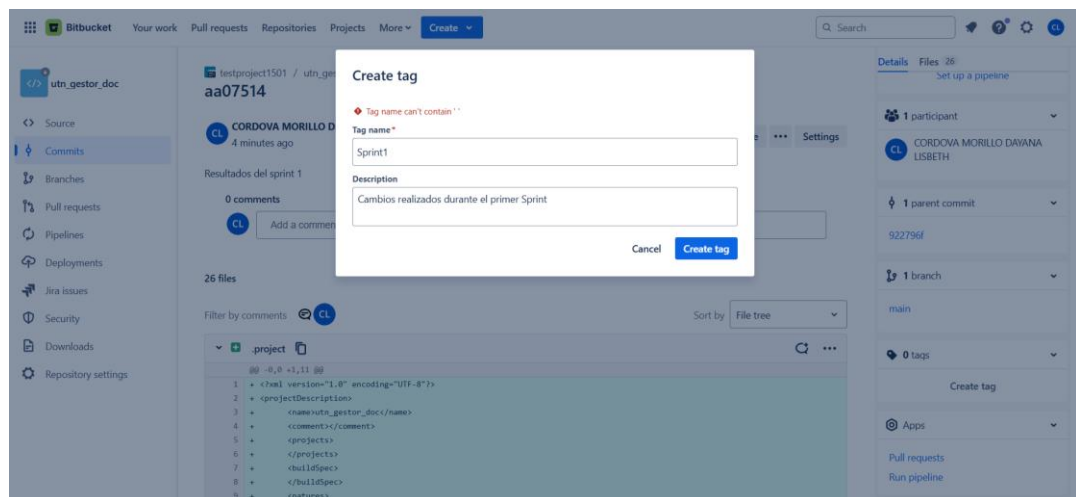


Nota. Elaboración Propia.

Creación de la etiqueta del Sprint 1

Figura 269.

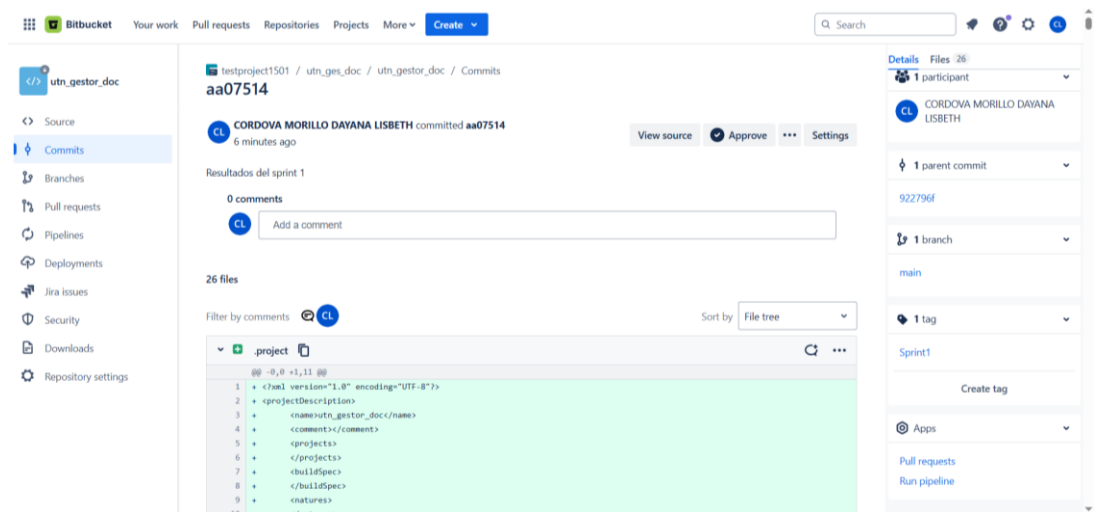
Creación de la etiqueta del Sprint 1



Nota. Elaboración Propia.

Commit etiquetado

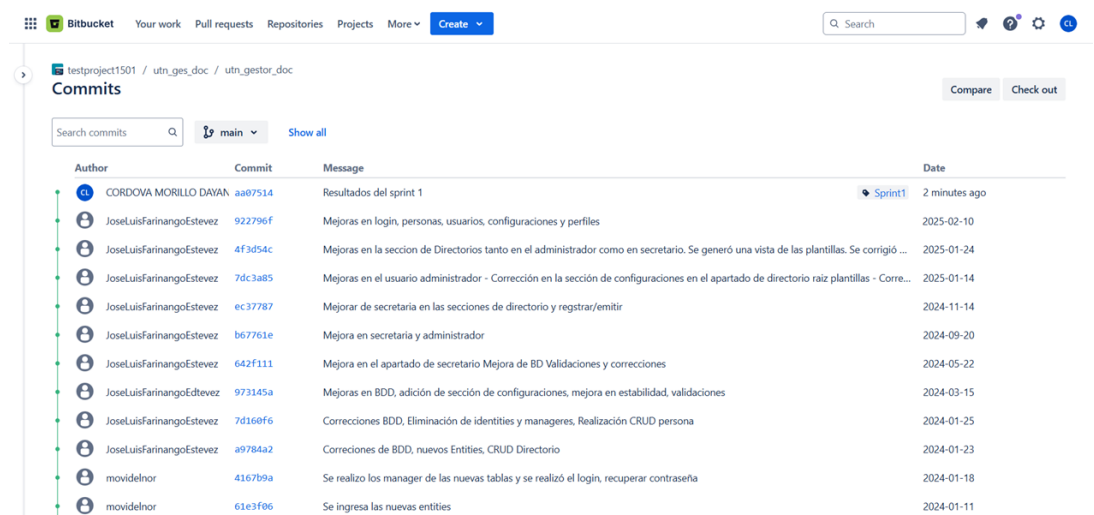
Figura 270.
Commit del Sprint 1 etiquetado



Nota. Elaboración Propia.

Commits actuales etiquetados

Figura 271.
Commits del Sprint 1 etiquetados



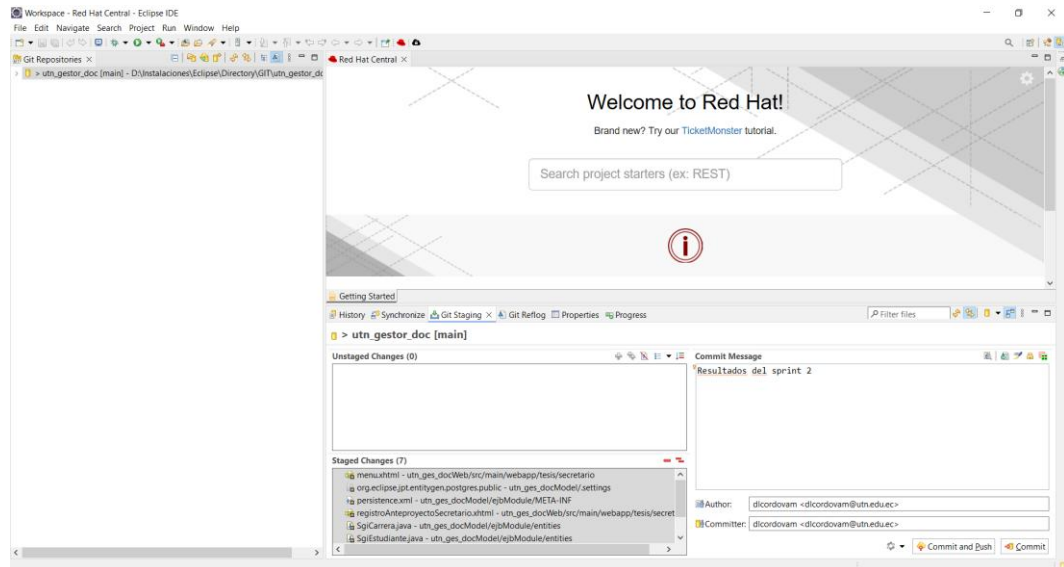
Nota. Elaboración Propia.

Etiquetado de la versión de lanzamiento: Sprint 2

Commit subido desde eclipse

Figura 272.

Commit del Sprint 2 subido desde eclipse

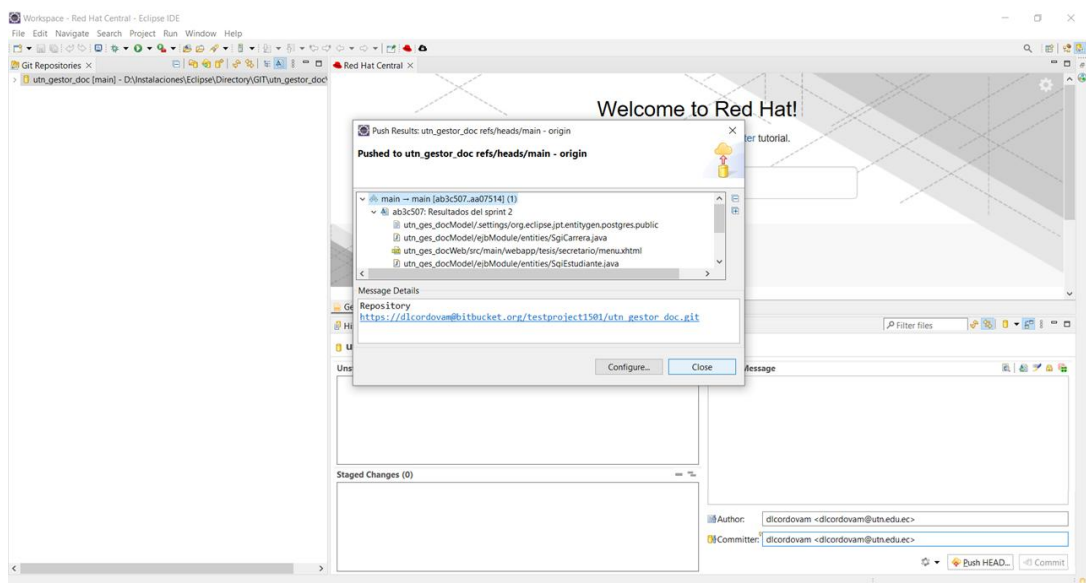


Nota. Elaboración Propia.

Cambios subidos a Bitbucket

Figura 273.

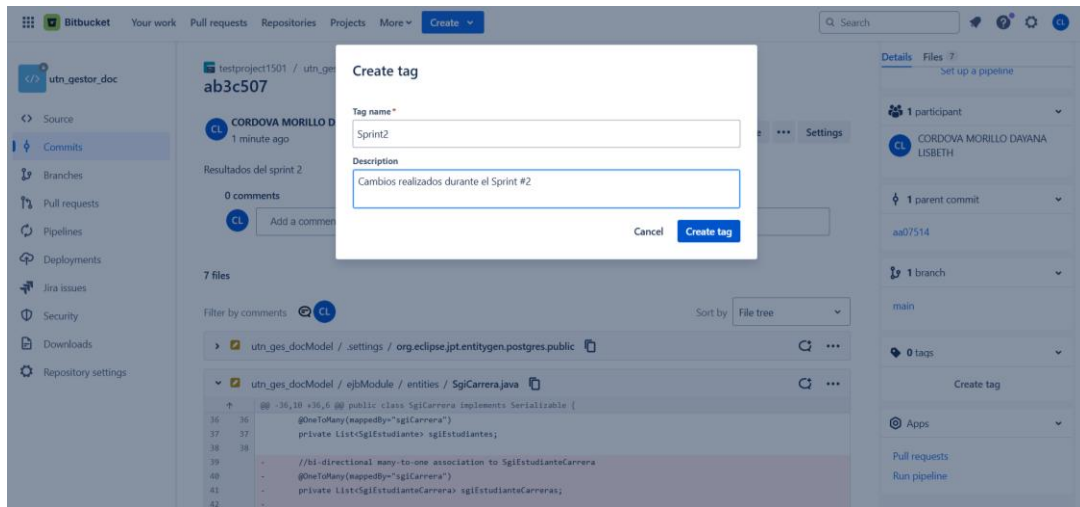
Cambios del Sprint 2 subidos a Bitbucket



Nota. Elaboración Propia.

Creación de la etiqueta del Sprint 2

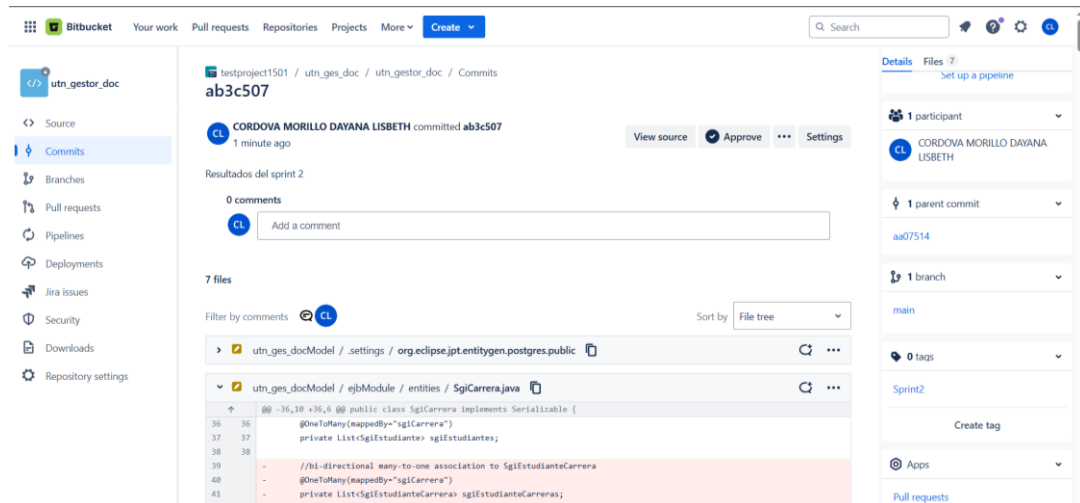
Figura 274.
Creación de la etiqueta del Sprint 2



Nota. Elaboración Propia.

Commit etiquetado

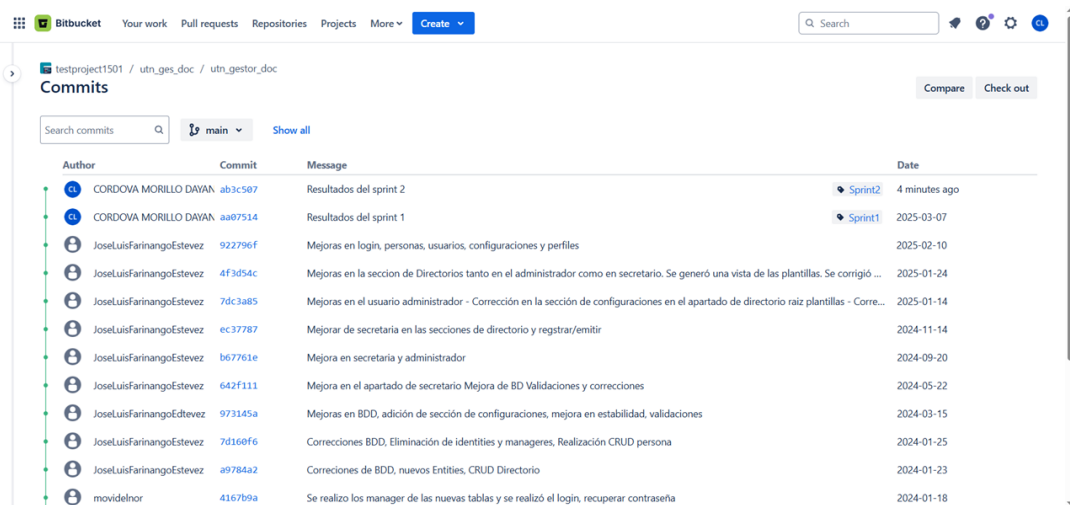
Figura 275.
Commits del Sprint 2 etiquetados



Nota. Elaboración Propia.

Commits actuales etiquetados

Figura 276.
Commits actualizados luego del Sprint 2



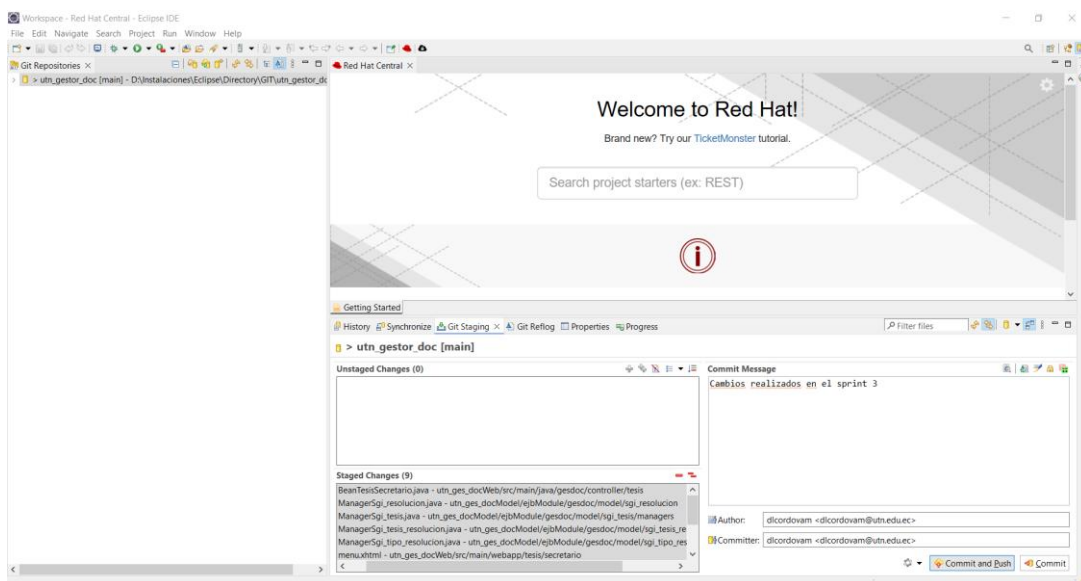
Author	Commit	Message	Date
CORDOVA MORILLO DAVAN	ab3c5b7	Resultados del sprint 2	4 minutes ago
CORDOVA MORILLO DAVAN	aa07514	Resultados del sprint 1	2025-03-07
JoseLuisFarinangoEstevez	922796f	Mejoras en login, personas, usuarios, configuraciones y perfiles	2025-02-10
JoseLuisFarinangoEstevez	4f3d54c	Mejoras en la seccion de Directorios tanto en el administrador como en secretario. Se generó una vista de las plantillas. Se corrigió ...	2025-01-24
JoseLuisFarinangoEstevez	7dc3a85	Mejoras en el usuario administrador - Corrección en la sección de configuraciones en el apartado de directorio raiz plantillas - Corre...	2025-01-14
JoseLuisFarinangoEstevez	ec37787	Mejorar de secretaria en las secciones de directorio y registrar/emitir	2024-11-14
JoseLuisFarinangoEstevez	b67761e	Mejora en secretaria y administrador	2024-09-20
JoseLuisFarinangoEstevez	642f111	Mejora en el apartado de secretario Mejora de BD Validaciones y correcciones	2024-05-22
JoseLuisFarinangoEstevez	973145a	Mejoras en BDD, adición de sección de configuraciones, mejora en estabilidad, validaciones	2024-03-15
JoseLuisFarinangoEstevez	7d160f6	Correcciones BDD, Eliminación de identities y managers, Realización CRUD persona	2024-01-25
JoseLuisFarinangoEstevez	a9784a2	Correcciones BDD, nuevos Entities, CRUD Directorio	2024-01-23
movidelnor	4167b9a	Se realizo los manager de las nuevas tablas y se realizó el login, recuperar contraseña	2024-01-18

Nota. Elaboración Propia.

Etiquetado de la versión de lanzamiento: Sprint 3

Commit subido desde eclipse

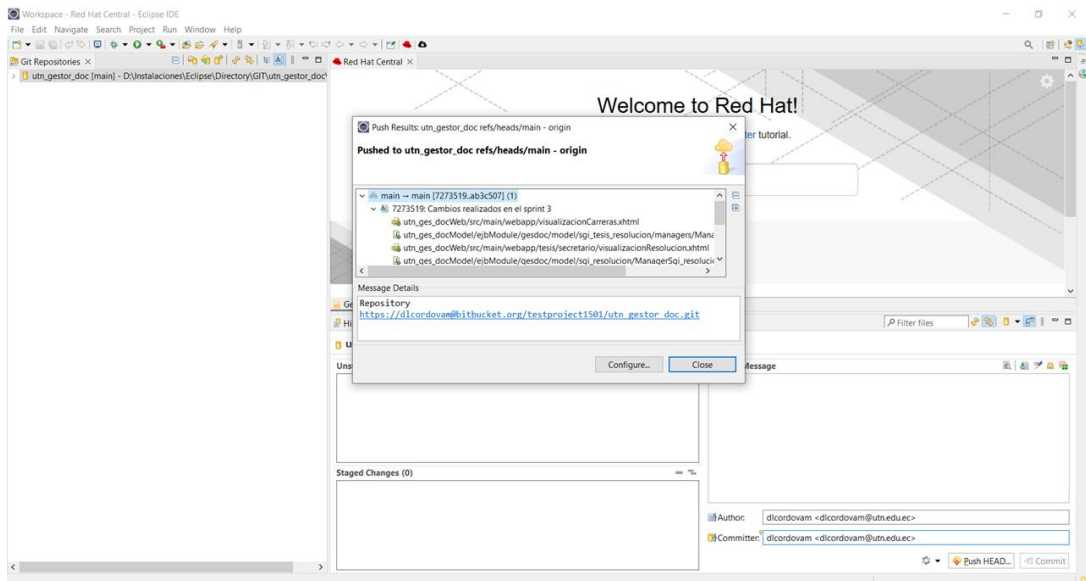
Figura 277.
Commits del Sprint 3 subidos



Nota. Elaboración Propia.

Cambios subidos a Bitbucket

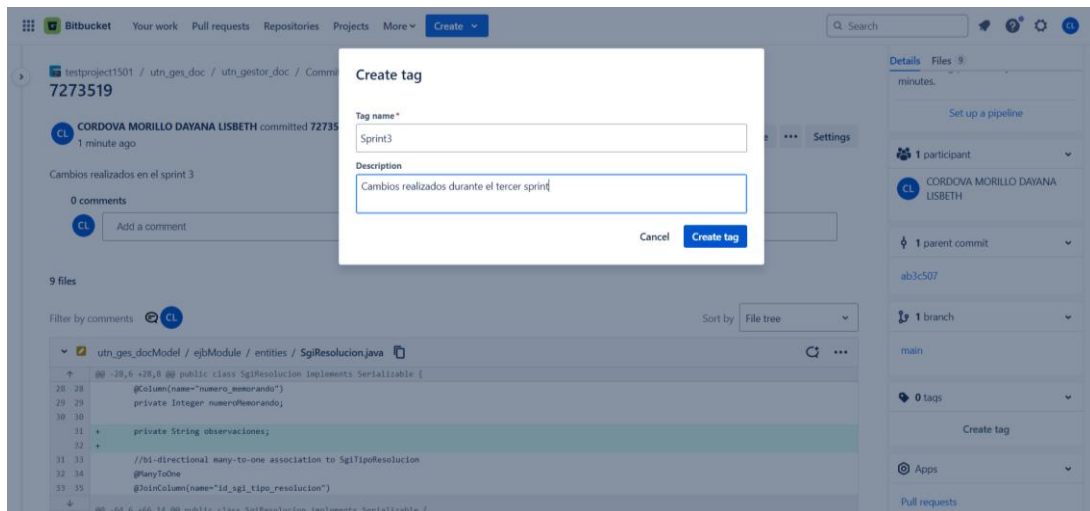
Figura 278.
Cambios del Sprint 3 subidos a Bitbucket



Nota. Elaboración Propia.

Creación de la etiqueta del Sprint 3

Figura 279.
Creación de la etiqueta del Sprint 3



Nota. Elaboración Propia.

Commits actuales etiquetados

Figura 280.
Commits actualizados después del Sprint 3

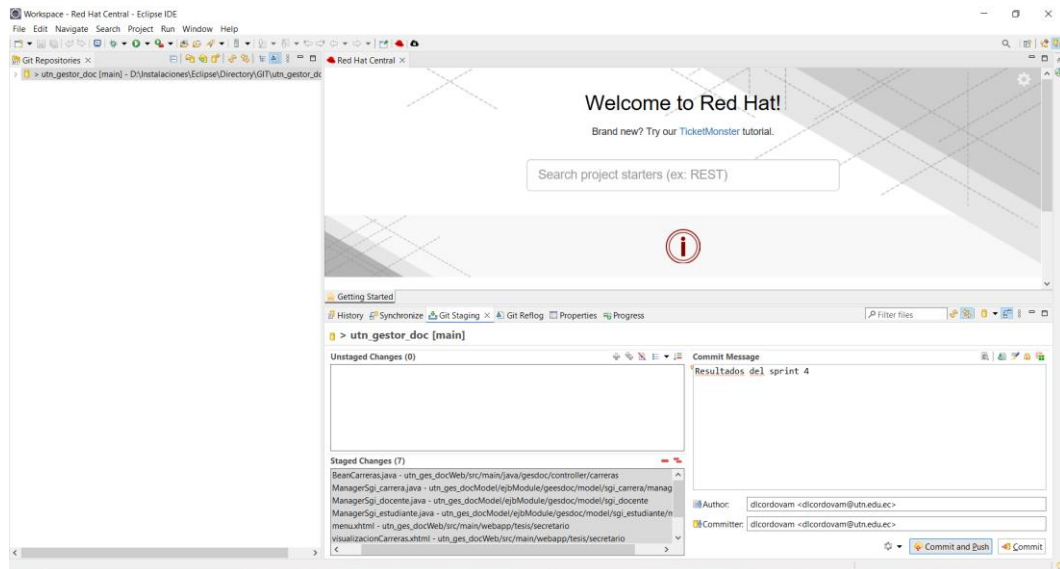
Author	Commit	Message	Date
CORDOVA MORILLO DAYAN	7273519	Cambios realizados en el sprint 3	Sprint3 2 minutes ago
CORDOVA MORILLO DAYAN	ab3c507	Resultados del sprint 2	Sprint2 2025-03-15
CORDOVA MORILLO DAYAN	aa07514	Resultados del sprint 1	Sprint1 2025-03-07
JoseLuisFarinangoEstevez	922796f	Mejoras en login, personas, usuarios, configuraciones y perfiles	2025-02-10
JoseLuisFarinangoEstevez	4f3d54c	Mejoras en la seccion de Directorios tanto en el administrador como en secretario. Se generó una vista de las plantillas. Se corrigió ...	2025-01-24
JoseLuisFarinangoEstevez	7dc3a85	Mejoras en el usuario administrador - Corrección en la sección de configuraciones en el apartado de directorio raiz plantillas - Corre...	2025-01-14
JoseLuisFarinangoEstevez	ec37787	Mejorar de secretaria en las secciones de directorio y registrar/emitir	2024-11-14
JoseLuisFarinangoEstevez	b67761e	Mejora en secretaria y administrador	2024-09-20
JoseLuisFarinangoEstevez	642f111	Mejora en el apartado de secretario Mejora de BD Validaciones y correcciones	2024-05-22
JoseLuisFarinangoEstevez	973145a	Mejoras en BDD, adición de sección de configuraciones, mejora en estabilidad, validaciones	2024-03-15
JoseLuisFarinangoEstevez	7d160f6	Correcciones BDD, Eliminación de identidades y managers, Realización CRUD persona	2024-01-25
JoseLuisFarinangoEstevez	a9784a2	Correcciones de BDD, nuevos Entites, CRUD Directorio	2024-01-23

Nota. Elaboración Propia.

Etiquetado de la versión de lanzamiento: Sprint 4

Commit subido desde eclipse

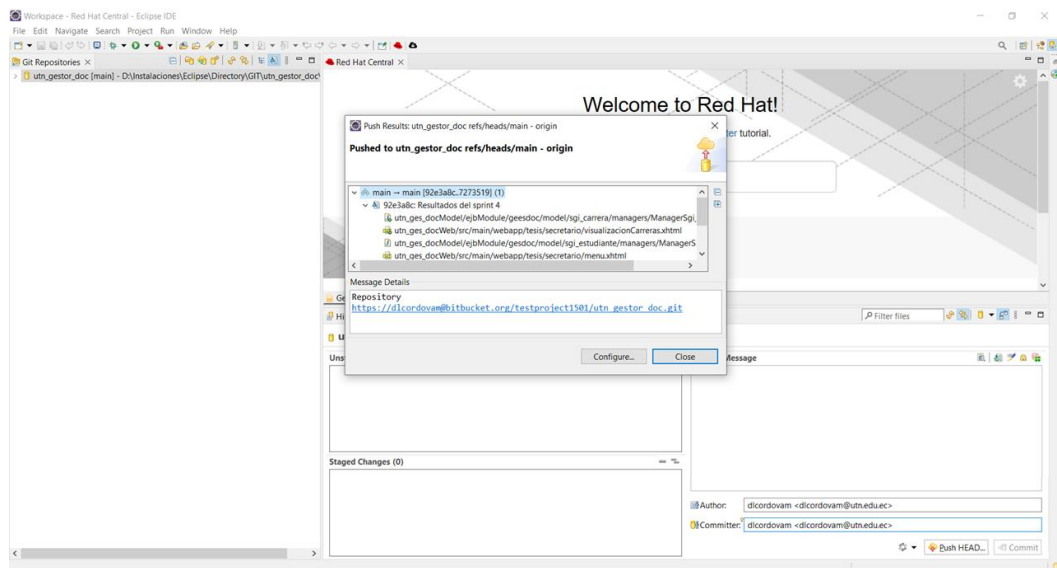
Figura 281.
Commit del Sprint 4



Nota. Elaboración Propia.

Cambios subidos a Bitbucket

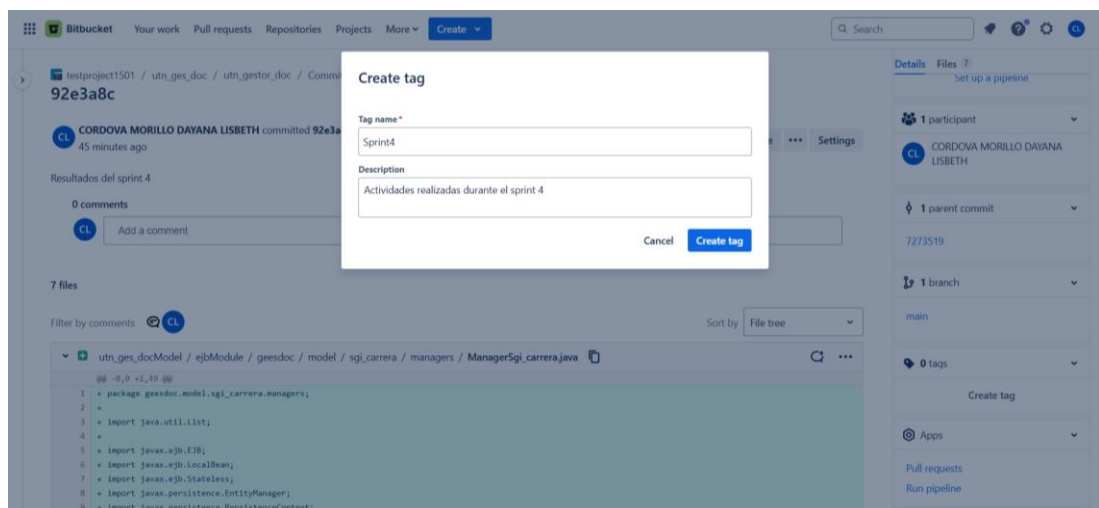
Figura 282.
Cambios del Sprint 4 subidos a Bitbucket



Nota. Elaboración Propia.

Creación de la etiqueta del Sprint 4

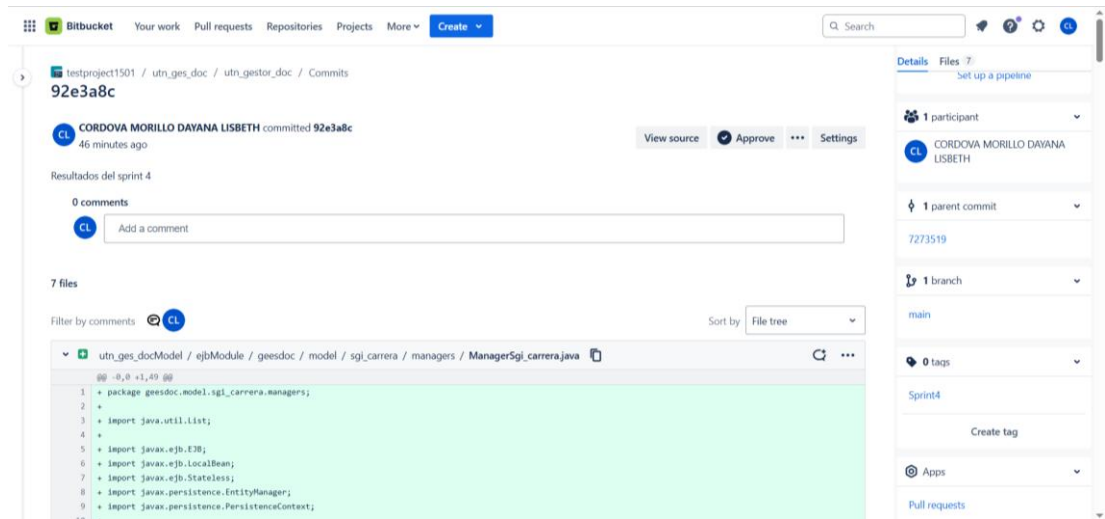
Figura 283.
Creación de la etiqueta del Sprint 4



Nota. Elaboración Propia.

Commit etiquetado

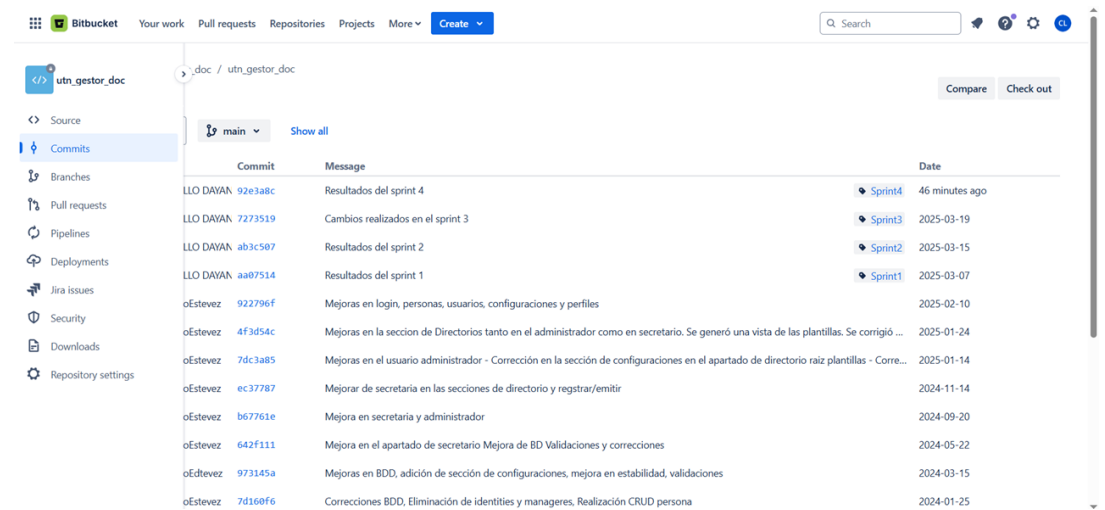
Figura 284.
Commit del Sprint 4 etiquetados



Nota. Elaboración Propia.

Commits actuales etiquetados

Figura 285.
Commits actualizados luego del Sprint 4

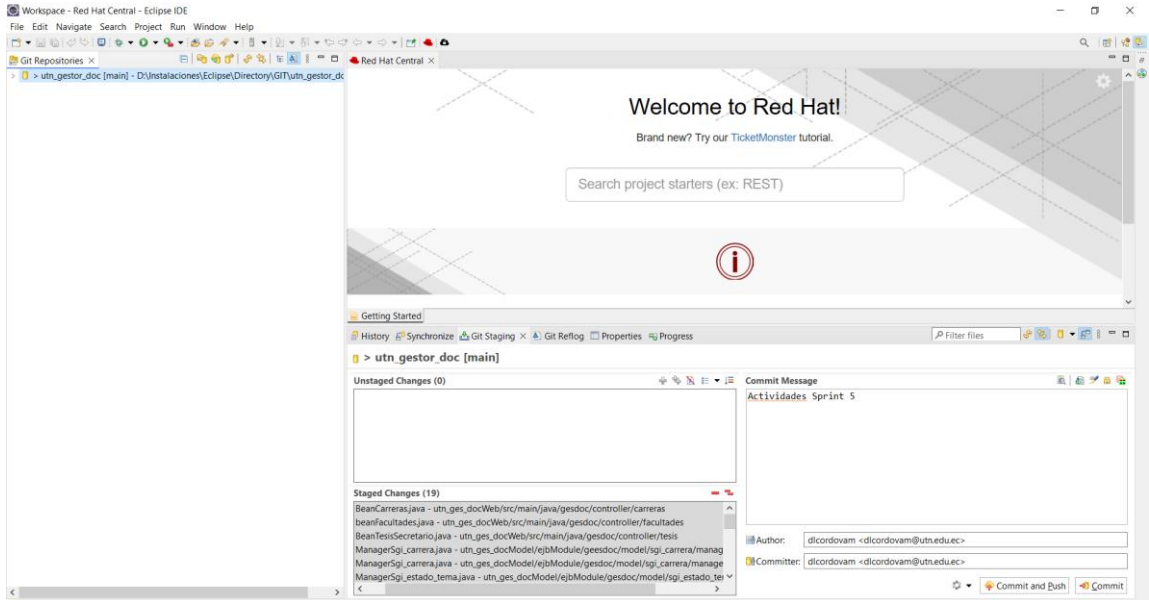


Nota. Elaboración Propia.

Etiquetado de la versión de lanzamiento: Sprint 5

Commit subido desde Eclipse

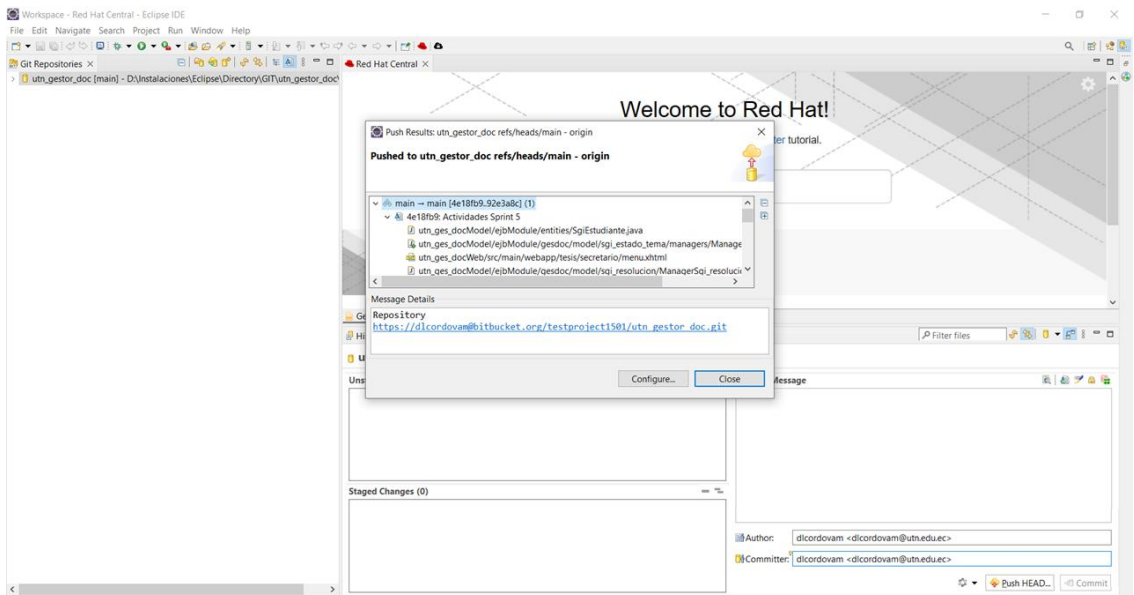
Figura 286.
Commit del Sprint 5



Nota. Elaboración Propia.

Cambios subidos a Bitbucket

Figura 287.
Cambios del Sprint 5 subidos a Bitbucket

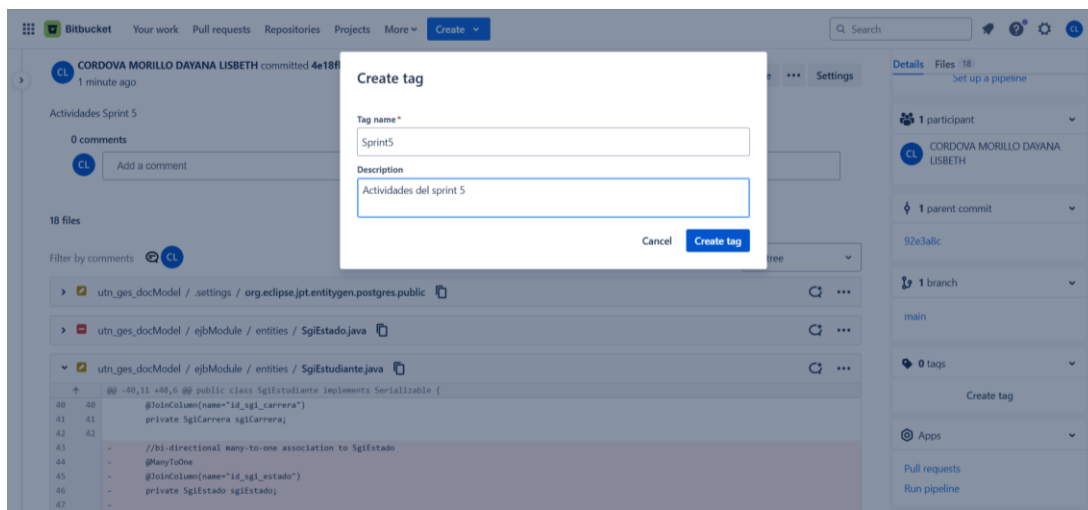


Nota. Elaboración Propia.

Creación de la etiqueta del Sprint 5

Figura 288.

Creación de la etiqueta del Sprint 5

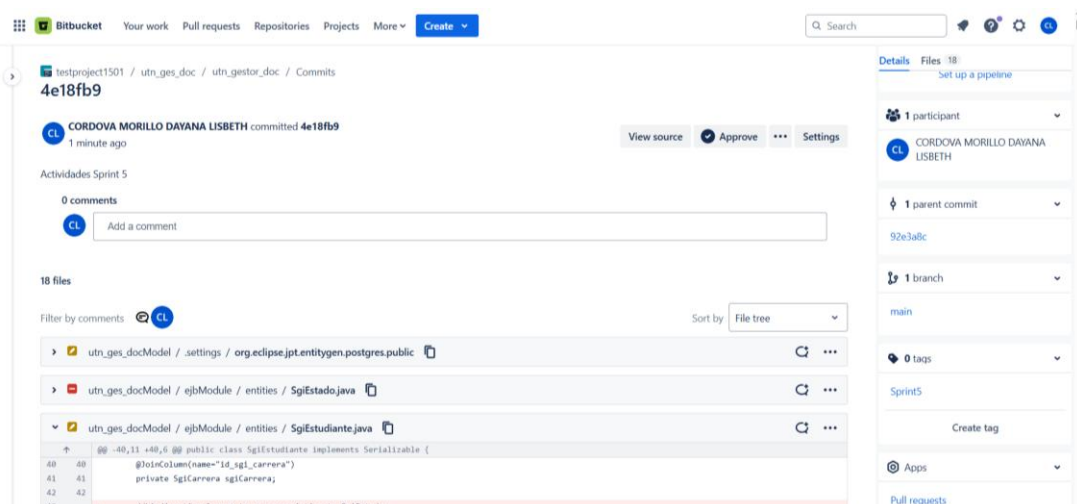


Nota. Elaboración Propia.

Commit etiquetado

Figura 289.

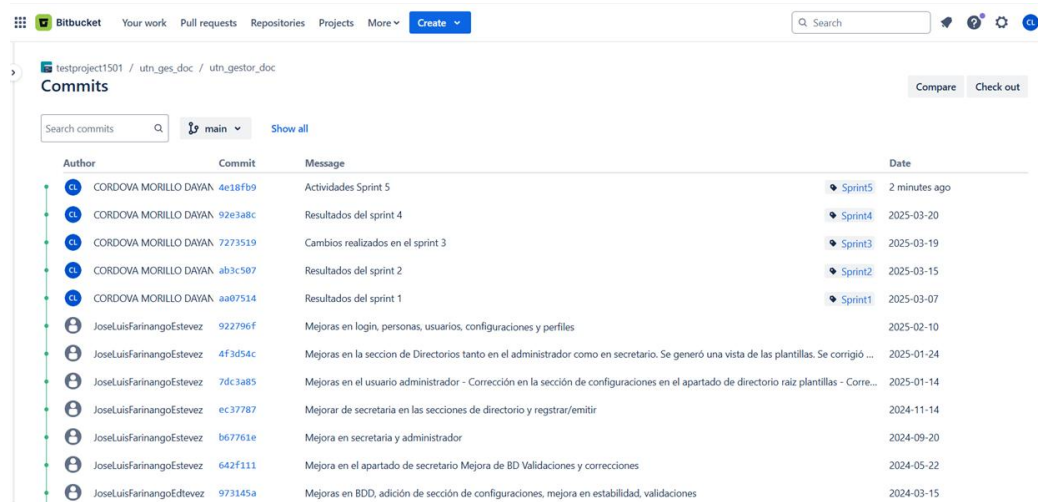
Commit del Sprint 5 etiquetado



Nota. Elaboración Propia.

Commits actuales etiquetados

Figura 290.
Commits actualizados después del Sprint 5



The screenshot shows the Bitbucket interface for the repository 'utn_ges'. The 'Commits' tab is active, displaying a list of commits. The commits are sorted by date, with the most recent at the top. The list includes commits from 'CORDOVA MORILLO DAVAN' and 'JoseLuisFarinangoEstevez'. The commits are labeled with 'Sprint' tags (Sprint1 through Sprint5) and include a 'Date' column.

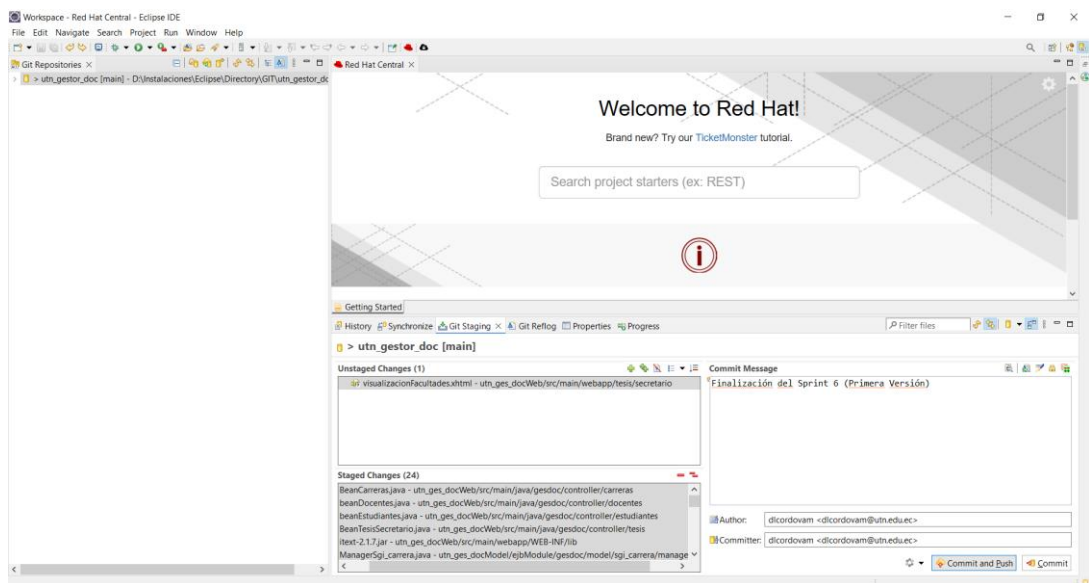
Author	Commit	Message	Date
CORDOVA MORILLO DAVAN	4e18fb9	Actividades Sprint 5	Sprint5 2 minutos ago
CORDOVA MORILLO DAVAN	92e3a8c	Resultados del sprint 4	Sprint4 2025-03-20
CORDOVA MORILLO DAVAN	7273519	Cambios realizados en el sprint 3	Sprint3 2025-03-19
CORDOVA MORILLO DAVAN	ab3c507	Resultados del sprint 2	Sprint2 2025-03-15
CORDOVA MORILLO DAVAN	aa07514	Resultados del sprint 1	Sprint1 2025-03-07
JoseLuisFarinangoEstevez	922796f	Mejoras en login, personas, usuarios, configuraciones y perfiles	2025-02-10
JoseLuisFarinangoEstevez	4f3d54c	Mejoras en la seccion de Directorios tanto en el administrador como en secretario. Se generó una vista de las plantillas. Se corrigió ...	2025-01-24
JoseLuisFarinangoEstevez	7dc3a85	Mejoras en el usuario administrador - Corrección en la sección de configuraciones en el apartado de directorio raiz plantillas - Corre...	2025-01-14
JoseLuisFarinangoEstevez	ec37787	Mejorar de secretaria en las secciones de directorio y registrar/emitir	2024-11-14
JoseLuisFarinangoEstevez	b67761e	Mejora en secretaria y administrador	2024-09-20
JoseLuisFarinangoEstevez	642f111	Mejora en el apartado de secretario Mejora de BD Validaciones y correcciones	2024-05-22
JoseLuisFarinangoEstevez	973145a	Mejoras en BDD, adición de sección de configuraciones, mejora en estabilidad, validaciones	2024-03-15

Nota. Elaboración Propia.

Etiquetado de la versión de lanzamiento: Sprint 6

Commit subido desde Eclipse

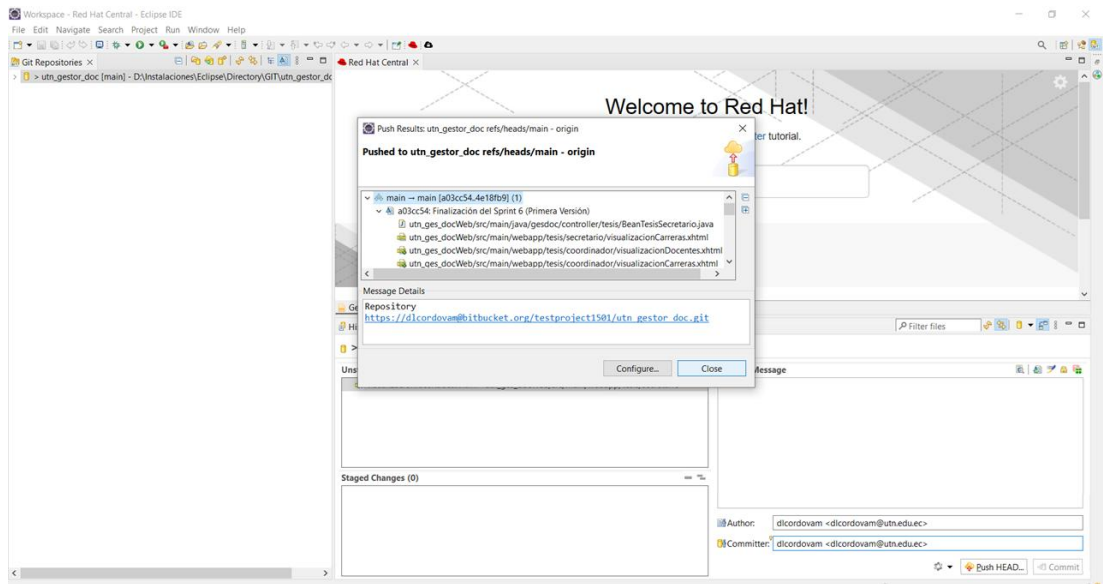
Figura 291.
Commits del Sprint 6



Nota. Elaboración Propia.

Cambios subidos a Bitbucket

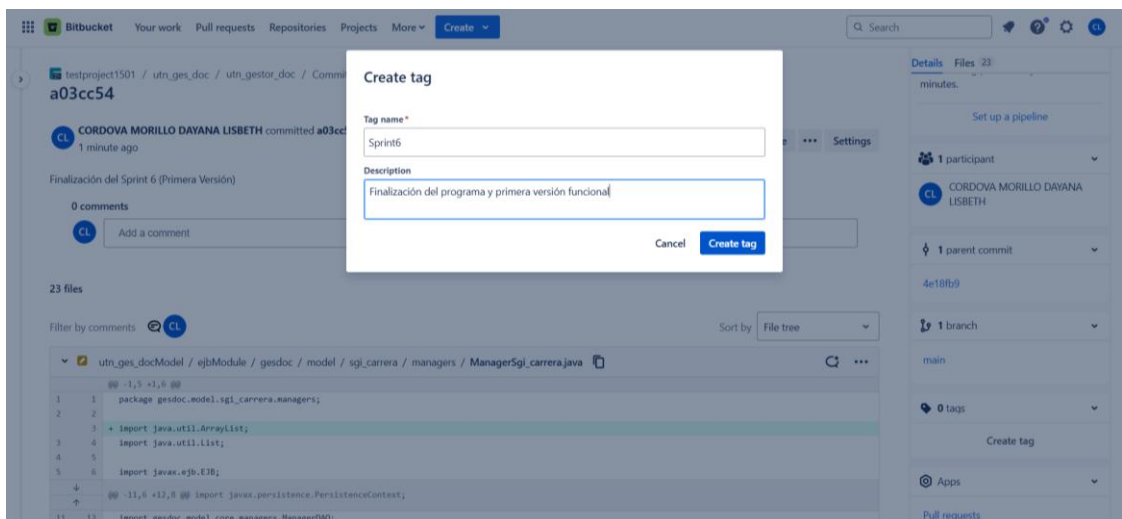
Figura 292.
Commits del Sprint 6 subidos en Bitbucket



Nota. Elaboración Propia.

Creación de la etiqueta del Sprint 6

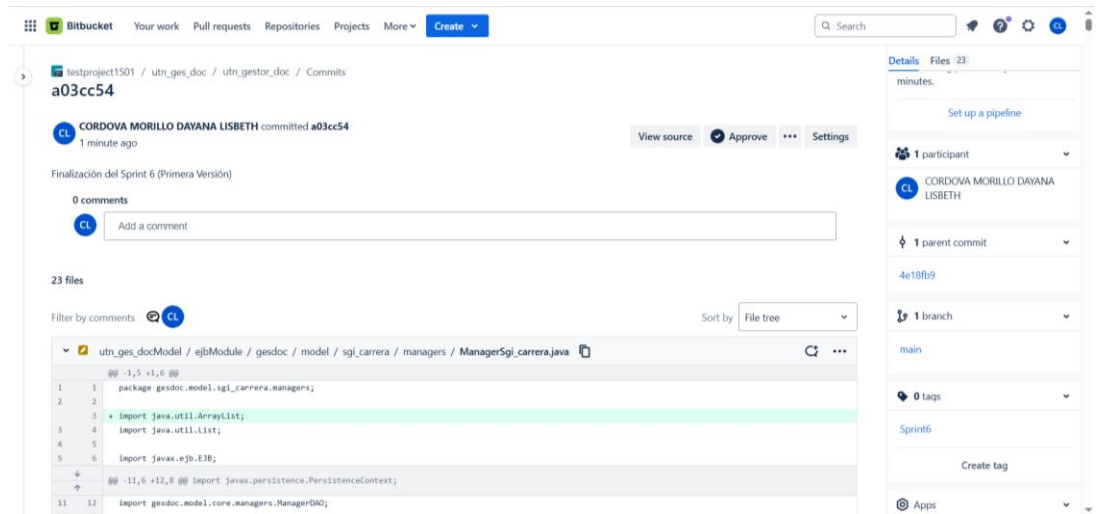
Figura 293.
Creación de la etiqueta del Sprint 6



Nota. Elaboración Propia.

Commit etiquetado

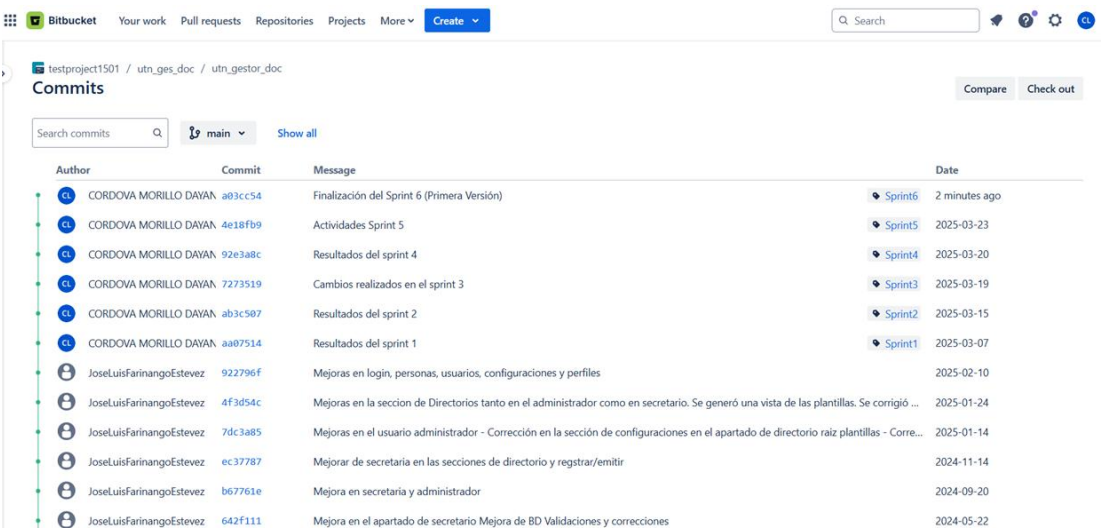
Figura 294.
Commits del Sprint 6 subidos a Bitbucket



Nota. Elaboración Propia.

Commits actuales etiquetados

Figura 295.
Etiquetas actualizadas después del Sprint 6



Nota. Elaboración Propia.