

**UNIVERSIDAD CATÓLICA SANTO TORIBIO DE MOGROVEJO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN**



**PLATAFORMA COLABORATIVA PARA CONTRIBUIR AL PROCESO
DE DESARROLLO DE SOFTWARE**

**TESIS PARA OPTAR EL TÍTULO DE
INGENIERO DE SISTEMAS Y COMPUTACIÓN**

AUTOR (A)

CAROL JANET GUERRERO GUTIERREZ

ASESOR (A)

Mgr. RICARDO DAVID IMÁN ESPINOZA

Chiclayo, 2019

DEDICATORIA

A Dios por ser mi luz y guía en cada paso que he dado en este trayecto de realización
universitaria.

A mis padres por ser mi apoyo y fortaleza, quienes en los altibajos siempre creyeron en la
capacidad de su hija.

A mis hermanos y sobrinos por ser mis incondicionales, motivación e inspiración diaria.

Al Ing. Anthony Luis Quepuy Damián por su guía y motivación constante durante la
realización de este proyecto.

AGRADECIMIENTOS

A mi familia por siempre estar a mi lado.

A mi asesor de tesis Mgtr. Ricardo David Imán Espinoza quien con su guía y tolerancia me permitió encontrar el camino adecuado en el alcance de esta meta.

A mi profesor Mgtr. Marlon Eugenio Vílchez Rivas quien me enseñó que a pesar de todo siempre debo dar lo mejor de mí.

A la División de Tecnologías de Información del Hospital Regional Lambayeque por permitirme realizar la investigación en su institución.

A la familia Quepuy Damián por estar a mi lado en este proceso y demostrarme que la única limitante para alcanzar nuestros objetivos es uno mismo.

RESUMEN

La presente tesis tiene como objetivo contribuir al proceso de desarrollo de software mediante una plataforma colaborativa desarrollado en el stack MEAN con metodología Scrum, además de una extensión para la consulta al repositorio desde un plugin desarrollado en Java. La metodología empleada es tecnológica aplicada, al implementar una plataforma colaborativa para apoyar el desarrollo de proyectos informáticos, y por contrastación de Hipótesis se realizó una investigación de tipo experimental de Pre test – Post test. La población consta del personal de la División de Tecnologías de Información de un hospital de la región de Lambayeque. Con la aplicación del software se pudo reducir las fuentes de búsqueda de solución, también realizar la búsqueda de incidencias con un tiempo promedio de consulta de 5 segundos, registrar 239 métodos y consulta frecuentes, 159 incidencias asignadas por proyecto correspondiente y 133 soluciones por incidencias, lo que permite tener la información en un solo lugar y al alcance de la necesidad. El desarrollo de esta investigación ha permitido contribuir a disminuir en un 75% la cantidad de fuentes de búsqueda de solución a incidencias, disminuir en un 99.91% el tiempo promedio de búsqueda de solución a incidencias, aumentar en un 75% la cantidad de métodos y consultas registrados en un repositorio, aumentar en un 100% la cantidad de incidencias registradas por sistema desarrollado en la entidad, aumentar en un 100% la cantidad de soluciones guardadas en una biblioteca de software definitivo.

PALABRAS CLAVE: Desarrollo de software, Incidencia, MEAN Stack, Java.

ABSTRACT

This thesis aims to collaborate in the software development process through groupware developed in MEAN stack with Scrum methodology, as well as an extension for consultation in the repository from a plugin developed in Java. The methodology used is applied technological, by implementing a groupware to support the development of computer projects, and by contrasting hypotheses an investigation of experimental of Pretest - Posttest was carried out. The demo consists of the staff of the Information Technology Division of a hospital in the Lambayeque region. With the application of the software it was possible to reduce the sources of search of solution, also carry out the search of incidences with an average time of consultation of 5 seconds, register 239 methods and frequent consultation, 159 incidents assigned by corresponding project and 133 solutions for incidents, which allows having the information in one place and within reach of the need. The development of this research has enabled to reduce the number of search sources for solutions to incidents by 75%, decrease by 99.91% average search time for solving incidents, heightens by 75% the number of methods and queries registered in a repository, heightens in 100% the number of incidents registered by the system developed in the entity, heightens in 100% the number of solutions stored in a definitive software library.

KEYWORDS: Software development, Incidence, MEAN Stack, Java.

ÍNDICE

I.	INTRODUCCIÓN.....	14
II.	MARCO TEÓRICO	20
2.1.	ANTECEDENTES.....	20
2.1.1.	ANTECEDENTES INTERNACIONALES.....	20
2.1.2.	ANTECEDENTES NACIONALES.....	22
2.1.3.	ANTECEDENTES LOCALES.....	24
2.2.	BASES TEÓRICO CIENTÍFICAS	26
2.2.1.	PLATAFORMA COLABORATIVA	26
2.2.2.	PROYECTO DE SOFTWARE	27
2.2.2.1.	Ciclo de Vida del Software	28
2.2.2.2.	Desarrollo de Software	29
2.2.3.	METODOLOGÍA SCRUM.....	34
2.2.3.1.	Scrum Team	34
2.2.3.2.	Eventos de Scrum.....	35
2.2.3.3.	Artefacto de Scrum	36
2.2.3.4.	Ventajas de la Metodología Scrum	37
2.2.4.	BIBLIOTECA DE INFRAESTRUCTURA DE TECNOLOGÍAS DE INFORMACIÓN	38
2.2.4.2.	Gestión del Servicio.....	39
2.2.4.3.	Catálogo de Servicios	40
2.2.4.4.	Programa de Mejora del Servicio	40
2.2.4.5.	Gestión de Incidencias	40
2.2.4.6.	Métricas	42
2.2.4.7.	Base de Datos de la Gestión de la Configuración	42
2.2.5.	SISTEMA DE GESTIÓN DE BASE DE DATOS NO RELACIONALES.....	43
2.2.6.	MEAN STACK.....	45
2.2.6.1.	MongoDB	47
2.2.6.2.	Express.JS.....	49
2.2.6.3.	Angular.JS	50
2.2.6.4.	Node.Js	51
III.	METODOLOGÍA	53
3.1.	TIPO Y NIVEL DE INVESTIGACIÓN	53
3.1.1.	TIPO DE INVESTIGACIÓN.....	53
3.1.2.	NIVEL DE INVESTIGACIÓN	53

3.2.	DISEÑO DE INVESTIGACIÓN	53
3.3.	POBLACIÓN, MUESTRA Y MUESTREO	53
3.3.1.	POBLACIÓN	53
3.3.2.	MUESTRA.....	53
3.3.3.	MUESTREO	54
3.4.	CRITERIOS DE SELECCIÓN.....	54
3.5.	OPERACIONALIZACIÓN DE VARIABLES.....	54
3.5.1.	VARIABLES.....	54
3.5.1.1.	Variable independiente.....	54
3.5.1.2.	Variable dependiente	54
3.5.2.	INDICADORES	55
3.6.	TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS	56
3.7.	PROCEDIMIENTOS	56
3.8.	PLAN DE PROCESAMIENTO Y ANÁLISIS DE DATOS	58
3.9.	MATRIZ DE CONSISTENCIA	59
3.10.	CONSIDERACIONES ÉTICAS.....	60
3.11.	GESTIÓN DE RIESGOS.....	60
3.11.1.	ETAPA DE ANÁLISIS	61
3.11.2.	ETAPA DE DISEÑO.....	63
3.11.3.	ETAPA DE DESARROLLO.....	65
3.11.4.	ETAPA DE PRUEBAS	68
IV.	RESULTADOS.....	71
4.1.	EN BASE A LA METODOLOGÍA UTILIZADA	71
4.1.1.	PRODUCT BACKLOG	71
4.1.2.	CONSTRUYENDO EL PRODUCT BACKLOG	71
4.1.3.	PRIORIZANDO EL PRODUCT BACKLOG	72
4.1.4.	IDENTIFICANDO LA COMPLEJIDAD.....	72
4.1.5.	ASIGNANDO EL VALOR POR CADA STORY POINTS PARA CADA USER STORY.....	73
4.1.6.	DURACIÓN EN DÍA DEL SPRINT	74
4.1.7.	EL USER STORY MÁS REPRESENTATIVO	74
4.1.8.	USER STORY ATENDIDOS POR UN SPRINT.....	75
4.1.9.	NÚMERO TOTAL DE SPRINTS	75
4.1.10.	TIEMPO TOTAL DE ENTREGA	76
4.1.11.	ELABORACIÓN Y AGRUPACIÓN DE SPRINTS.....	76
4.1.12.	DESARROLLO DE SPRINTS.....	77
4.1.12.1.	SPRINT 1	77

4.1.12.2.SPRINT 2	95
4.1.12.3.SPRINT 3	105
4.1.13. ETAPA DE PRUEBAS.....	117
4.1.13.1.PLAN DE PRUEBA POR SPRINT.....	117
4.1.13.2.PRUEBA DE ACEPTACIÓN	120
4.1. EN BASE A LOS OBJETIVOS DE LA INVESTIGACIÓN	124
4.1.1. DISMINUCIÓN DE LA CANTIDAD DE FUENTES DE BÚSQUEDA DE SOLUCIÓN DE INCIDENCIAS.....	124
4.1.2. DISMINUCIÓN EL TIEMPO PROMEDIO DE BÚSQUEDA DE SOLUCIÓN A INCIDENCIAS.....	126
4.1.3. AUMENTO DE LA CANTIDAD DE MÉTODOS Y CONSULTAS REGISTRADOS EN UN REPOSITORIO.....	127
4.1.4. AUMENTO DE LA CANTIDAD DE INCIDENCIAS REGISTRADAS POR SISTEMA DESARROLLADO EN LA ENTIDAD.....	129
4.1.5. AUMENTO DE LA CANTIDAD DE SOLUCIONES GUARDADAS EN UNA BIBLIOTECA DE SOFTWARE DEFINITIVO.....	131
V. DISCUSIÓN	133
5.1. INDICADOR 1: CANTIDAD DE FUENTES DE CONSULTA DE SOLUCIÓN	133
5.2. INDICADOR 2: TIEMPO PROMEDIO DE BÚSQUEDA DE SOLUCIÓN A INCIDENCIAS	135
5.3. INDICADOR 3: CANTIDAD DE MÉTODOS Y CONSULTAS ALMACENADAS EN UN REPOSITORIO	136
5.4. INDICADOR 4: CANTIDAD DE INCIDENCIAS REGISTRADAS POR SISTEMA DESARROLLADO 137	
5.5. INDICADOR 5: CANTIDAD DE SOLUCIONES POR INCIDENCIA REGISTRADAS EN EL REPOSITORIO	138
VI. CONCLUSIONES.....	139
VII. RECOMENDACIONES.....	140
VIII. LISTA DE REFERENCIAS.....	141
IX. ANEXOS	145
ANEXO N° 01. ESTRUCTURA DE ENTREVISTA A JEFATURA DE LA DIVISIÓN DE TECNOLOGÍAS DE INFORMACIÓN	145
ANEXO N° 02. ESTRUCTURA DE ENTREVISTA AL PERSONAL RESPONSABLE DEL DESARROLLO Y MONITOREO DEL SOFTWARE EN LA DIVISIÓN DE TECNOLOGÍAS DE INFORMACIÓN	147

ANEXO N° 03. ESTRUCTURA DE ACTA DE INCIDENCIA DE REFERENCIA.....	149
ANEXO N° 04. DEFINICIÓN DE PRIORIDADES ASIGNADAS A INCIDENCIAS	150
ANEXO N°05. ESTRUCTURA DE ENTREVISTA DE PRUEBA DE SOFTWARE	151
ANEXO N°06. ESTRUCTURA DE ENTREVISTA DE APROBACIÓN DE SOFTWARE.	152
ANEXO N°07. MANUAL DE USUARIO	153
ANEXO N°08. MANUAL TECNICO.....	183

ÍNDICE DE TABLAS

TABLA I. VENTAJAS Y DESVENTAJAS DE NoSQL [51] [52]	44
TABLA II. DIFERENCIAS ENTRE SQL Y NoSQL [50].....	45
TABLA III. COMPARACIÓN DE SINTAXIS EN CONSULTAS DE SQL Y NoSQL [53]	45
TABLA IV. INDICADORES	55
TABLA V. TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS	56
TABLA VI. MATRIZ DE CONSISTENCIA	59
TABLA 7. REFERENCIAS DURANTE EL ANÁLISIS	60
TABLA 8. MATRIZ DE RIESGOS DE LA ETAPA DE ANÁLISIS.....	61
TABLA 9. MATRIZ DE SALVAGUARDA DE RIESGOS EN LA ETAPA DE ANÁLISIS.....	62
TABLA 10. MATRIZ DE RIESGOS DE LA ETAPA DE DISEÑO	63
TABLA 11. MATRIZ DE SALVAGUARDA DE RIESGOS EN LA ETAPA DE DESARROLLO	64
TABLA 12. MATRIZ DE RIESGOS DE LA ETAPA DE DESARROLLO	65
TABLA 13. MATRIZ DE SALVAGUARDA DE RIESGOS EN LA ETAPA DE DESARROLLO	67
TABLA 14. MATRIZ DE RIESGOS DE LA ETAPA DE PRUEBAS.....	68
TABLA 15. MATRIZ DE SALVAGUARDA DE RIESGOS EN LA ETAPA DE PRUEBAS	70
TABLA 16. PRODUCTO BACKLOG.....	71
TABLA 17. CONSTRUYENDO EL PRODUCTO BACKLOG	71
TABLA 18. PRIORIZANDO EL PRODUCT BACKLOG	72
TABLA 19. IDENTIFICANDO LA COMPLEJIDAD	73
TABLA 20. ASIGNANDO EL VALOR POR CADA STORY POINTS PARA CADA USER STORY	73
TABLA 21. DURACIÓN EN DÍAS DE SPRINT.....	74
TABLA 22. USER STORY MÁS REPRESENTATIVO	74
TABLA 23. USER STORY ATENDIDOS POR SPRINT.....	75
TABLA 24. NÚMERO TOTAL DE SPRINTS	75
TABLA 25. TIEMPO TOTAL DE ENTREGA	76
TABLA 26. ELABORACIÓN Y AGRUPACIÓN DE LOS SPRINTS	76
TABLA 27. PILA DE SPRINT 1.....	78
TABLA 28. HISTORIA DE USUARIO DE INICIO DE SESIÓN.....	79
TABLA 29. HISTORIA DE USUARIO DE INCIDENCIA.....	80
TABLA 30. HISTORIA DE USUARIO DE ROLES	81
TABLA 31. TAREA DE USUARIO CREAR INICIO DE SESIÓN	81
TABLA 32. TAREA DE USUARIO CREAR AUTENTICACIÓN DE PERSONA	82
TABLA 33. TAREA DE USUARIO REALIZAR INTERFAZ GRÁFICA DE INICIO DE SESIÓN WEB.....	82
TABLA 34. TAREA DE USUARIO REALIZAR INTERFAZ GRÁFICA DE INICIO DE SESIÓN PLUGIN ...	83
TABLA 35. TAREA DE USUARIO CREAR ESQUEMA INCIDENCIAS	83

TABLA 36. TAREA DE USUARIO CREAR, REGISTRAR, EDITAR, LISTAR Y ELIMINAR INCIDENCIAS	84
TABLA 37. TAREA DE USUARIO REALIZAR INTERFAZ GRÁFICA DE INCIDENCIA PLUGIN.....	84
TABLA 38. TAREA DE USUARIO REALIZAR INTERFAZ GRÁFICA DE INCIDENCIAS WEB	85
TABLA 39. TAREA DE USUARIO CREAR ESQUEMA ROLES.....	85
TABLA 40. TAREA DE USUARIO CREAR, REGISTRAR, EDITAR, LISTAR Y ELIMINAR ROLES	86
TABLA 41. TAREA DE USUARIO REALIZAR INTERFAZ GRÁFICA DE ROLES WEB	86
TABLA 42. ACTORES INVOLUCRADOS CON EL SPRINT	87
TABLA 43. PILA DEL SPRINT 2.....	95
TABLA 44. HISTORIA DE USUARIO DE PERSONA	96
TABLA 45. HISTORIA DE USUARIO DE PROYECTO	97
TABLA 46. TAREA DE USUARIO CREAR ESQUEMA PERSONAS	98
TABLA 47. TAREA DE USUARIO CREAR, EDITAR, LISTAR Y ELIMINAR PERSONAS	98
TABLA 48. TAREA DE USUARIO REALIZAR INTERFAZ GRÁFICA WEB DE PERSONAS	99
TABLA 49. TAREA DE USUARIO CREAR ESQUEMA PROYECTOS	99
TABLA 50. TAREA DE USUARIO CREAR, REGISTRAR Y LISTAR PROYECTOS.....	100
TABLA 51. TAREA DE USUARIO REALIZAR INTERFAZ GRÁFICA WEB DE PROYECTO.....	100
TABLA 52. PILA DEL SPRINT 3.....	105
TABLA 53. HISTORIA DE USUARIO DE SOLUCIÓN.....	106
TABLA 54. HISTORIA DE USUARIO DE INTEGRANTE.....	107
TABLA 55. TAREA DE USUARIO CREAR ESQUEMA SOLUCIÓN	108
TABLA 56. TAREA DE USUARIO CREAR, REGISTRAR Y EDITAR SOLUCIÓN.....	109
TABLA 57. TAREA DE USUARIO CREAR ESQUEMA INTEGRANTES.....	110
TABLA 58. TAREA DE USUARIO CREAR, REGISTRAR, EDITAR, LISTAR Y ELIMINAR INTEGRANTE	111
TABLA 59. PLANIFICACIÓN DE PRUEBAS DE SPRINT 1	117
TABLA 60. PLANIFICACIÓN DE PRUEBAS DE SPRINT 2	118
TABLA 61. PLANIFICACIÓN DE PRUEBAS DE SPRINT 3	119
TABLA 62. CANTIDAD DE FUENTES DE CONSULTA DE SOLUCIÓN	133
TABLA 63. TIEMPO PROMEDIO DE BÚSQUEDA DE SOLUCIÓN A INCIDENCIAS	135
TABLA 64. TIEMPOS DE CONSULTA POR FUENTE.....	135
TABLA 65. CANTIDAD DE MÉTODOS Y CONSULTAS ALMACENADAS EN UN REPOSITORIO.....	136
TABLA 66. CANTIDAD TOTAL DE MÉTODOS Y CONSULTAS FRECUENTES	136
TABLA 67. CANTIDAD DE INCIDENCIAS REGISTRADAS POR PROYECTO	137
TABLA 68. CANTIDAD DE SOLUCIONES POR INCIDENCIA REGISTRADAS EN EL REPOSITORIO...	138

ÍNDICE DE FIGURAS

FIG. 1. MATRIZ 2X2 DE GROUPWARE ESPACIO TEMPORAL [22].....	27
FIG. 2. CICLO DE VIDA EN CASCADA [26]	28
FIG. 3. PLATAFORMA JAVA 2. EDICIÓN ESTÁNDAR V 5.0 [33].....	30
FIG. 4. ENTORNO DE DESARROLLO INTEGRADO NETBEANS FUENTE: ELABORACIÓN PROPIA ..	31
FIG. 5. MODELO DE IMPLEMENTACIÓN DE UN SISTEMA MVC UTILIZANDO API REST [42]	33
FIG. 6. PROCESO SCRUM [44]	37
FIG. 7. CICLO DE VIDA DE ITIL [46].....	39
FIG. 8. CICLO DE GESTIÓN DEL SERVICIO [46].....	40
FIG. 9. PROCESO DE GESTIÓN DE INCIDENCIAS [46]	41
FIG. 10. DEPENDENCIAS DE LA BASE DE DATOS DE LA GESTIÓN DE LA CONFIGURACIÓN [47].	42
FIG. 11. DEFINICIÓN DE UN ELEMENTO CON JSON [49].....	43
FIG. 12. ARQUITECTURA DEL STACK MEAN [57].....	46
FIG. 13. NIVELES DE INTERÉS DE LAS TECNOLOGÍAS MEAN EN EL PERÚ [58]	46
FIG. 14. ARQUITECTURA DE CLÚSTERES FRAGMENTADOS EN MONGODB [60].....	48
FIG. 15. CONEXIÓN A BASE DE DATOS LOCAL DE MONGODB EN JAVA Y JAVASCRIPT.	49
FIG. 16. LÍNEAS DE COMANDO PARA INSTALACIÓN DE EXPRESS [63].....	49
FIG. 17. ARQUITECTURA DE ANGULAR.JS [64].....	50
FIG. 18. GENERADOR DE CLASES EN ANGULAR [64]	51
FIG. 19. ARQUITECTURA DE SERVIDOR NODE.JS [67].....	52
FIG. 20. CREACIÓN DE SERVIDOR WEB CON NODE.JS [68].....	52
FIG. 21. DIAGRAMA DE CASO DE USO DE SPRINT 1.....	88
FIG. 22. DISEÑO LÓGICO DE BASE DE DATOS DE SPRINT 1	89
FIG. 23. DISEÑO FÍSICO DE LA BASE DE DATOS DE SPRINT 1	89
FIG. 24. BOCETO DE INTERFAZ WEB DE INICIO DE SESIÓN	90
FIG. 25. BOCETO DE INTERFAZ PLUGIN DE INICIO DE SESIÓN	90
FIG. 26. BOCETO DE INTERFAZ WEB DE GESTIONAR INCIDENCIA	91
FIG. 27. BOCETO DE INTERFAZ PLUGIN DE GESTIÓN DE INCIDENCIAS	91
FIG. 28. BOCETO DE INTERFAZ WEB DE GESTIONAR ROL	92
FIG. 29. INTERFAZ WEB DE INICIO DE SESIÓN	92
FIG. 30. INTERFAZ PLUGIN DE INICIO DE SESIÓN	93
FIG. 31. INTERFAZ WEB DE GESTIÓN DE INCIDENCIAS	93
FIG. 32. INTERFAZ PLUGIN DE GESTIÓN DE INCIDENCIAS	94
FIG. 33. INTERFAZ WEB DE GESTIÓN DE ROLES	94
FIG. 34. DIAGRAMA DE CASO DE USO DE SPRINT 2.....	101
FIG. 35. DISEÑO LÓGICO DE BASE DE DATOS DE SPRINT 2.....	101
FIG. 36. DISEÑO FÍSICO DE BASE DE DATOS DE SPRINT 2	102

FIG. 37. BOCETO INTERFAZ WEB DE GESTIÓN DE PERSONAS	102
FIG. 38. BOCETO INTERFAZ WEB DE GESTIÓN DE PROYECTOS.....	103
FIG. 39. INTERFAZ WEB DE GESTIONAR PERSONA	103
FIG. 40. INTERFAZ WEB DE GESTIONAR PROYECTO	104
FIG. 41. DIAGRAMA DE CASO DE USO DE SPRINT 3.....	112
FIG. 42. DISEÑO LÓGICO DE BASE DE DATOS DE SPRINT 3	113
FIG. 43. DISEÑO FÍSICO DE BASE DE DATOS DE SPRINT 3	113
FIG. 44. BOCETO INTERFAZ WEB DE GESTIÓN DE INTEGRANTES	114
FIG. 45. BOCETO INTERFAZ WEB DE GESTIÓN DE SOLUCIÓN	114
FIG. 46. BOCETO PLUGIN DE GESTIÓN DE SOLUCIÓN	115
FIG. 47. INTERFAZ WEB DE GESTIONAR INTEGRANTE.....	115
FIG. 48. INTERFAZ WEB DE GESTIONAR SOLUCIÓN.....	116
FIG. 49. INTERFAZ PLUGIN DE GESTIONAR SOLUCIÓN	116
FIG. 50. BÚSQUEDA EN INTERFAZ WEB DE INCIDENCIAS.....	124
FIG. 51. BÚSQUEDA EN INTERFAZ PLUGIN DE SOLUCIONES.....	124
FIG. 52. BUSCADOR DE INCIDENCIAS EN WEB	125
FIG. 53. BUSCADOR DE INCIDENCIAS EN PLUGIN.....	125
FIG. 54. REGISTRO DE MÉTODO O CONSULTA EN INTERFAZ WEB	127
FIG. 55. REGISTRO DE MÉTODO O CONSULTA EN INTERFAZ PLUGIN	128
FIG. 56. BOTÓN DE AÑADIR INCIDENCIA EN WEB	128
FIG. 57. OPCIÓN DE AÑADIR INCIDENCIAS EN PLUGIN.....	128
FIG. 58. INTERFAZ WEB DE REGISTRO DE INCIDENCIA POR PROYECTO	129
FIG. 59. INTERFAZ PLUGIN DE REGISTRO DE INCIDENCIA POR PROYECTO.....	129
FIG. 60. COMBO EN WEB DE PROYECTOS A ELEGIR	130
FIG. 61. COMBO EN PLUGIN DE PROYECTOS A ELEGIR.....	130
FIG. 62. INTERFAZ WEB DE DETALLE DE INCIDENCIA	131
FIG. 63. INTERFAZ PLUGIN DE DETALLE DE INCIDENCIA	132
FIG. 64. BOTÓN DE AÑADIR SOLUCIÓN EN WEB.....	132
FIG. 65. OPCIÓN DE AÑADIR SOLUCIÓN EN PLUGIN	132

I. INTRODUCCIÓN

En la actualidad las empresas cuentan con herramientas tecnológicas que les permite gestionar su información de manera eficiente y segura en el momento que lo necesitan y sin tiempos de espera excesivos, por tanto, el uso de tecnologías de información de calidad es importante para su buen desarrollo y posicionamiento en el mercado actual. Por eso, las empresas de desarrollo de sistemas o proyectos informáticos, acorde a los requerimientos de sus clientes, se ven en la necesidad de brindar productos de calidad que permita a las empresas agregar valor a la información que manejan. Sin embargo, los tiempos de entrega del producto son cada vez más cortos y la tecnología, entornos de desarrollo y requerimientos de usuarios muy cambiantes, generando que el desarrollo de un producto de calidad sea cada vez una gran tarea.

Muchas veces, por no contar con un mecanismo formal para la gestión y/o retroalimentación de un proceso (o incidencia), se genera que la solución a los problemas se pueda localizar en cualquier parte, desde una fuente en internet, notas guardadas por un compañero o un comentario dentro del código fuente, como menciona Zaval [1], los datos se encuentran en constante movimiento y son interpretados para volverse activos generadores de conocimiento mediante la información adquirida en la organización, se capitaliza la experiencia y los procesos desarrollados por el personal, por eso, el software no se elabora como cualquier producto industrial porque está dirigido a la operación de la organización, y si esta es compleja, la elaboración del software también lo será.

Según el informe del Caos del 2015, publicado por Standish Group, basado en el estudio de 50.000 proyectos a nivel mundial, 19% fracasaron y el 50% no se asegura ni el fracaso ni el éxito, mostrando la poca influencia de metodologías, ciclos de vida, etc., además, los proyectos pequeños lograban el mayor porcentaje de éxito a comparación de los proyectos grandes, con más de un 62% demostrando que la complejidad de un proyecto es relativa a su magnitud [2], así mismo el 74% concluye tiempo después de lo anticipado, y de ellos, un 59% eleva su costo y solo logran el 69% de los requerimientos funcionales [3].

Agregado a ello el estudio realizado por PMI (Pulse of the Profession) basado en más de 2.000 profesionales de la gestión de proyectos determina varias causas principales del fracaso en la realización de proyectos, resaltando la falta de implicación de los interesados y la pobre comunicación por parte de los involucrados, por lo que se debe mejorar los procesos, estandarizándolos y formalizándolos, además de reconocer y desarrollar las habilidades de las personas para el adecuado desarrollo y las buenas prácticas.

"Lo primero es conocer que hay un problema, lo segundo es conocer cuál es el problema y lo tercero es resolverlo" [4].

La consultora Gartner, mencionada en la revista Virtual Pro [5], estima que durante el 2017 el mercado mundial de software ha crecido un 7,2%, generando un volumen total de ventas de US\$ 357.000 millones, con ventas valorizadas en US\$ 943.000 millones por servicios de tecnología informática, además Cimas, mencionado en la revista Gestión [6] expresa que, de las empresas latinoamericanas el 54% considera desarrollar su propio software, mientras que el 30% prefiere el servicio de outsourcing y un 15% opta por software empaquetado.

El Perú cuenta con gran potencial en el mercado de desarrollo de software, ya que se dedica a cubrir necesidades que requieren los empresarios para agilizar o simplificar sus procesos, pero el Gerente comercial de Ofisis, Gustavo Mendoza mencionado en la revista Gestión [7], expone la reducción de personas emprendedoras en el desarrollo de software, ya que no alcanza ni el 1% del mercado. Ofisis es una firma que representa una de las principales empresas que ofrece el servicio de desarrollo de software en el Perú y cuenta con más de 35000 clientes activos en Latinoamérica y más de 600 profesionales especializados, es así como las empresas desarrolladoras de software tienen gran cantidad de personal que comparten software, gestionan proyectos y necesitan agilizar tiempos para ofrecer el mejor servicio.

Existen dos tipos de software con los que puede contar una organización, genérico y personalizado, el software genérico es un sistema desarrollado de forma aislada al consumidor y el software personalizado o hecho a medida, es un sistema que cumple los requerimientos de una organización en específica [10]. Por otro lado,

existen organizaciones que desarrollan su propio software y una de ellas es el Hospital Regional Lambayeque, que cuenta con la División de Tecnologías de la Información, la misma que es responsable de dar soporte informático y velar por su buen funcionamiento (Anexo 08), así mismo del desarrollo, implementación y monitoreo de los doce (12) sistemas con los que cuenta la entidad, de ahí 3 son genéricos y 9 personalizados (Anexo 12).

El desarrollo de un software se realiza en base a los requerimiento de algún área usuaria del hospital, y durante la codificación se sigue la metodología XP (Anexo 10 y Anexo 11), y en el ciclo de vida del software, específicamente la fase de desarrollo se lleva a cabo de la siguiente manera; durante el desarrollo de un software colaboran mínimo dos personas(Anexo 10 y Anexo 11), quienes son responsables de la codificación siguiendo un marco de trabajo, utilizando un lenguaje de programación y un gestor de base de datos. Este software puede presentar incidencias, ya sean por alguna falla presentada, algún error mostrado, un nuevo requerimiento o una mejora a realizar, lo que implica una modificación al sistema (Anexo 9, Anexo 10 y Anexo11), también se presentan métodos (en la codificación a nivel de lenguaje de programación) y consultas frecuentes (en la codificación a nivel de base de datos).

El desarrollador tanto interno (cualquiera de los dos trabajadores encargados) como externo (trabajador de apoyo, ya sea practicante, trabajador temporal o alguien más de la división) durante la codificación de un sistema realiza comentarios en el código fuente según su criterio, respecto a un método frecuente o si ha corregido algún método, comenta el método inicial y deja operativo el nuevo método (Anexo 12), en otros casos, cuando no comenta código, solo realiza modificaciones en los métodos y lo guarda como tal, lo que, en dos(2) de cada cinco(5) modificaciones del método en un sistema implantado ocasionan falla o error y se desconoce quién realizó la modificación y en donde la realizó (Anexo 12), además, durante la codificación, de cada cinco(5) métodos o consultas frecuentes, solo se registran dos(2) en cualquier editor de texto, o es comentado en el mismo código fuente del sistema, por lo que cuando quiere volver a revisar el método que recuerda haber guardado, realiza una búsqueda de archivo por

archivo, pero muchas veces no recuerda con que nombre ha guardado el archivo o donde lo ha guardado específicamente(Anexo 12).

Cuando el desarrollador guarda algún método o consulta frecuente en algún editor de texto y al momento que se le presenta una incidencia a la que no puede dar solución en el acto, empieza el proceso de búsqueda de una solución a nivel de líneas de código, donde, al encontrar una solución, esta es copiada en el código fuente y se adecuan las variables a la solución requerida por el sistema, es por ello que se busca un método o una consulta que dé solución a una problemática igual o similar.

Los desarrolladores tienen 4 tipos de búsqueda, que en el peor de los casos puede tomar horas, hasta días encontrar una solución (Anexo 9, Anexo 10 y Anexo 11), los cuales no son secuenciales, pero tienden a darse siempre, una forma de búsqueda en los comentarios del código fuente del sistema en desarrollo, que toma un tiempo de quince(15) a veinte(20) minutos, mientras se recorre la codificación de todo el sistema buscando una solución, otra forma de búsqueda, es mediante archivos editados con líneas de código que se encuentren en el mismo sistema, archivos ubicados en otros sistemas o archivos editados en bloc o Word que se encuentran en otras carpetas, correo o redes sociales, el realizar este procedimiento toma un tiempo de veinte(20) a treinta minutos (30).

Otro tipo de búsqueda es la consulta al compañero, cuando este se encuentra en la oficina se le realiza la consulta respecto a código, y el mismo, busca, muestra y comparte la solución hallada por algún medio de comunicación como correo o redes sociales, con el desarrollador, si el compañero no se encuentra en la oficina, debe esperarlo, si se encuentra de vacaciones o de permiso, debe realizar una llamada telefónica para consultarle y esperar su respuesta, pero en el peor de los casos, si ya no labora en la institución deberá ver cualquier medio de comunicación para realizar la consulta lo cual toma un tiempo correspondiente de veinte(20) minutos a tres(3) días, y otra forma de búsqueda es mediante internet que toma un tiempo de veinte(20) minutos a una(1) hora(Anexo 9, Anexo 10 y Anexo 11- Anexo 12).

Respecto a los equipos para un proyecto de software, durante el desarrollo se muestra un trabajo individual por parte de los desarrolladores, es decir, resuelven situaciones a criterio propio, lo cual ocasiona que, por cada vez que un sistema es modificado por un nuevo desarrollador, se debe volver a revisar toda la codificación para una nueva edición (Anexo 12), debido a que no se evidencia registros de incidencias durante el desarrollo de software, ni registro de quien fue la persona que resolvió una incidencia presentada, además de no contar con un repositorio de incidencias, métodos y consultas frecuentes, ni con una herramienta tecnológica apropiada para su gestión(Anexo 12). Se ha podido observar que se cuenta con un registro de incidencias a nivel de servicio, que se da en fichas técnicas (Anexo 10) diversos cuadernillos, post-it y pizarra, más no se efectúa una retroalimentación durante el desarrollo de software a nivel de programación (Anexo 12).

Al realizar el análisis de la situación problemática de la empresa, se plantea la siguiente pregunta ¿Qué herramienta tecnológica permitirá contribuir en el proceso de desarrollo de software? obteniendo como respuesta, con la implementación de esta plataforma colaborativa se contribuirá al proceso de desarrollo de software. Como objetivo general se tiene; Desarrollar una plataforma colaborativa como herramienta tecnológica para contribuir al proceso de desarrollo de software, y como objetivos específicos se tiene; a. Disminuir la cantidad de fuentes de búsqueda de solución de incidencias.; b. Disminuir el tiempo promedio de búsqueda de solución a incidencias.; c. Aumentar la cantidad de métodos y consultas registrados en un repositorio.; d. Aumentar la cantidad de incidencias registradas por sistema desarrollado en la entidad.; e. Aumentar la cantidad de soluciones guardadas en una biblioteca de software definitivo.

El tema de tesis propuesto se justifica socialmente, porque permitirá a todos los integrantes de un proyecto organizar, centralizar y compartir información respecto a todas las incidencias presentadas en el desarrollo de software, agilizar procesos, mejorar tiempos de desarrollo y facilitar la comunicación entre desarrolladores y así poder mejorar la calidad del software. En la parte tecnológica, se realizará una plataforma colaborativa basada en MEAN Stack que gestionará un repositorio de incidencias de desarrollo de software, la misma que será consultada a través del

IDE del desarrollador, en este caso NetBeans, mediante la descarga e instalación de un plugin hecho en JAVA brindado por la misma plataforma, además los integrantes de un proyecto podrán compartir y acceder a sus respectivos repositorios en línea.

En la parte económica, al implementar la plataforma colaborativa se podrá reducir tiempos, mejorar el esfuerzo y brindar recursos necesarios para el éxito del desarrollo lo cual permitirá mejorar las estimaciones de costo y llevar a cabo el proyecto de manera eficiente. Mientras que, en el aspecto científico, este proyecto busca colaborar con futuras investigaciones enfocadas a la mejora del desarrollo de software y gestión de la información, además tiene aporte científico ya que son muy pocas las aplicaciones existentes para la gestión de incidencias en el desarrollo de software y la consulta de las mismas desde el entorno de desarrollo integrado.

II. MARCO TEÓRICO

2.1. Antecedentes

2.1.1. Antecedentes internacionales

Briede, Cabello, Pérez y Arriagada [11], narran la problemática de controlar, seguir y evaluar proyectos en el taller de diseño industrial. Se aplicó la metodología de desarrollo incremental, logrando obtener como solución una plataforma colaborativa para la gestión de proyectos. El valor agregado de esta investigación es realizar una dinámica iterativa para corrección de diferencias y errores. Finalmente, los autores concluyeron que realizar una plataforma colaborativa permite la generación de un sistema de registro y seguimiento de proyectos, además la unificación y uniformización de formato de almacenamiento, seguimiento y retroalimentación mediante una herramienta digital se facilita el control, seguimiento y evaluación de proyectos. Se tomó en consideración esta investigación ya que plantea los lineamientos necesarios para el desarrollo de una plataforma colaborativa, como el análisis del proceso de diseño, estudio y análisis de plataformas existentes, diseño y desarrollo de plataforma, característica del software y objetivo del software, los cuales permitirán gestionar y compartir información de proyectos, definiendo cantidad de actividades, entregas parciales y finales, la inclusión de entornos generales y específicos para los usuarios y la opción de poder realizar comentarios por todos los miembros del proyecto e incluso invitados.

Quintero [12], narra la problemática existente en la gestión de los servicios de TI. Se aplicó la metodología Biblioteca de Infraestructura de Tecnologías de Información (o ITIL, por sus siglas en inglés), logrando mejorar la calidad de los servicios ofrecidos por el área de TI para la problemática en la gestión de los servicios de TI. El valor agregado de esta investigación es la realización de una herramienta de software basado en el Gestor

Gratuito de Equipos Informáticos que permita el soporte adecuado y oportuno de la gestión de la mesa de servicio. Finalmente, la autora concluyó que ITIL es un marco de trabajo adecuado para la gestión de servicios TI dentro de una empresa, porque se pueden definir métricas que permiten el monitoreo permanente de la calidad del servicio y el incremento de la productividad en la gestión de servicios y procesos. Se tomó en consideración esta investigación ya que narra los problemas en la gestión de incidencias que se presentan en el área de Tecnologías de Información, como no contar con una base de datos de la gestión de la configuración, no existe una gestión adecuada y estandarizada, carece de validaciones y no cuenta con una herramienta para gestionar su conocimiento. Así como el uso de una herramienta informática que permita su gestión siguiendo los parámetros del marco planteado como categorías y nivel de impacto, además de registrar y asociar incidentes a cada elemento, conocer la hora de registro y asignación de usuarios, añadir solución a la base de conocimientos y el cierre de incidencia.

Aldabari [13], narra la problemática presentada en la construcción y funcionamiento de páginas web. Se aplicó la metodología programación extrema o Xtreme Programming (XP por sus siglas en inglés), logrando obtener una solución web basado en tecnología full-stack MEAN.js para la problemática presentada en la construcción y funcionamiento de páginas. El valor agregado de esta investigación es que, para la producción, la aplicación fue soportada en una nube pública de desarrollo de aplicaciones de software libre, Openshift. Finalmente, el autor concluyó que MEAN.js permite el desarrollo asíncrono, actualizar datos en tiempo real, recarga automática de página y creación de una web single-page. Se tomó en consideración esta investigación ya que permite apreciar el desempeño de una metodología ágil en el desarrollo de una aplicación web basada en MEAN stack, al mostrar las ventajas del uso de una tecnología emergente, puesto que se puede usar tanto del lado del servidor como del lado del cliente, permite el uso de un

único lenguaje en toda la aplicación, admite el uso de la arquitectura modelo vista controlador (MVC) y la referencia de datos se basa en la utilización de objetos JavaScript Object Notation (JSON por sus siglas en inglés).

2.1.2. Antecedentes nacionales

Gómez [14], narra la problemática presentada en la gestión de incidentes y organización de eventos. Se aplicó la metodología Biblioteca de Infraestructura de Tecnologías de Información (ITIL, por sus siglas en inglés), logrando aplicar los lineamientos del marco referencial de las mejores prácticas de ITIL v3.0 para la problemática presentada en la gestión de incidentes y organización de eventos. El valor agregado de esta investigación es el uso de la herramienta de software como CA Unicenter Service Desk para apoyar la implementación de la metodología. Finalmente, el autor concluyó que implementando ITIL se mejora la relación con los usuarios, se desarrolla procedimientos estandarizados y se logra el cumplimiento de objetivos, además de la reducción de tiempos de atención de incidencias y gestión de problemas. Se tomó en consideración esta investigación ya que se definen procesos para la gestión de incidencias y organización de atención de eventos, al reportar, clasificar y atender las incidencias, definir su prioridad, incluir soluciones temporales, roles en base a los implicados en el proceso y métricas para la medición de su cumplimiento progresivo.

Ocola [15], narra la problemática presentada en el desarrollo y mantenimiento de software en la fábrica de software. Se aplicó la metodología de ingeniería de software, logrando realizar una metodología en base a todas las metodologías ágiles y tradicionales para mejorar el desarrollo y mantenimiento de software en la fábrica de software. El valor agregado de esta investigación es que puede ser aplicada a proyectos de gran o pequeño alcance, además de realizar una combinación de las metodologías Scrum, XP y RUP para la creación de una nueva metodología. Finalmente, el autor

concluyó que las empresas desarrolladoras de software deben considerar la metodología adecuada para el desarrollo de software y el éxito de una metodología depende del acoplamiento con los miembros del equipo de trabajo. Se tomó en consideración esta investigación ya que usa la ingeniería de software para la fusión de tres metodologías y enfatiza los atributos clave a resaltar y tomar en cuenta en la aplicación de una metodología, aplicando métodos, técnicas y usando las herramientas necesarias para la construcción de un software de calidad, como Integración de modelos de madurez de capacidades (CMMI por sus siglas en inglés), Método de desarrollo e sistemas dinámicos (DSDM por sus siglas en inglés) y la ISO/IEC 12207, técnicas de análisis de complejidad, funcionalidad ingeniería inversa, reingeniería por partes y reestructuración, contemplando todas las etapas de desarrollo de software y facilitar el desarrollo y mantenimiento del sistema.

Egusquiza [16], narra la problemática presentada en el proceso de desarrollo de software. Se aplicó la metodología Scrum, logrando obtener un modelo de buenas prácticas que describe las actividades del ciclo de vida del desarrollo de software para mejorar el proceso de desarrollo de software. El valor agregado de la investigación es la realización de una aplicación web que permite medir el grado de cumplimiento y facilita el seguimiento de los proyectos de software. Se tomó en consideración este artículo ya que utiliza la metodología Scrum para el desarrollo del proyecto, incluyendo la evaluación del ciclo de vida del desarrollo de software y un plan de implementación de un modelo de mejora, como la asignación de roles (Equipo de desarrollo, dueño del producto y scrum master) y tareas (como evaluar el estado del ciclo de vida del desarrollo de software, elaborar propuesta de mejora para el desarrollo del ciclo de vida y diseño del plan de implementación del modelo de mejora) por cada miembro del proyecto, identificar la fuente de información (usuario), modificación o actualización de los componentes del software, actualización y corrección de errores y aprobación de resultados.

2.1.3. Antecedentes locales

Cornetero y Rojas [17] narran la problemática que existe en la planificación operativa de eventos. Se aplicó la metodología Xtreme Programming, logrando obtener un sistema colaborativo para mejorar la planificación operativa de eventos. El valor agregado de esta investigación es que varios usuarios pueden participar en el proceso de planificación de eventos desde el sistema colaborativo, además del hecho de compartir documentos y realizar comentarios, así como poder realizar invitaciones a cualquier usuario del sistema y recordatorios por medio de correo electrónico. Finalmente, los autores concluyeron que el sistema colaborativo disminuyó el tiempo de planificación, aumentó el tiempo de anticipación de recordatorio, se aumentó la cantidad de participantes, además permitió asignar materiales, difundir y controlar eventos. Se tomó en consideración este artículo ya que realiza y define criterios analíticos para una matriz de priorización (con criterios como igualmente importante, significativamente importante, excesivamente importante, significativamente menos importante, excesivamente menos importante), se realiza la identificación de tareas (como solo poder ver, registrar y modificar, realizar asignaciones y generar reportes) y asignación de tareas a los usuarios, además de la interacción de los participantes en el flujo de información entre las actividades.

Gonzales [18], narra la problemática presentada en la gestión de incidencias del área del Centro de Sistemas de Información. Se aplicó la metodología ITIL, logrando obtener la implementación del marco de trabajo ITIL v3.0 para mejorar la gestión de incidencias del área del Centro de Sistemas de Información. El valor agregado de esta investigación es la incorporación de herramientas y controles mediante el uso de software basado en el marco. La autora concluyó que la implementación del marco de trabajo permitió el aumento de incidencias resueltas, se redujo el tiempo de atención y solución, además del aumento de satisfacción del usuario. Se tomó en

consideración esta investigación ya que muestra aplicaciones referenciales como CMDBuild (modela y administra base de datos de activos informáticos), GLPI (solución de software abierto para la gestión de inventario), OTRS (herramienta de código abierto que incluye una mesa de ayuda para servicios de tecnologías de información), SysAid (solución de software integral de servicio de asistencia de tecnologías de información) e EasyVista (solución de software integral que cubre la gestión de servicios de tecnologías de información), además de mostrar estructuras con roles, responsabilidades y métricas para la gestión de eventos, incidencias y problemas.

Delgado [19] narra la problemática presentada en el servicio de atención al cliente del área del Centro de Sistemas de Información. Se aplicó la metodología ITIL, logrando la implementación del marco de trabajo para mejorar la atención al cliente del área del Centro de sistemas de Información. El valor agregado de esta investigación es la incorporación de herramientas y controles mediante el uso de software basado en el marco. El autor concluyó que la incorporación de herramientas basadas en ITIL mejora los tiempos de solución de problemas de TI, además disminuyó el índice de llamadas por problemas e incrementó los niveles de satisfacción de usuarios de los servicios de TI. Se tomó en consideración esta investigación ya que brinda formatos con ítems que permite la estructuración de actas de incidencias(ANEXO N° 03. **ESTRUCTURA DE ACTA DE INCIDENCIA DE REFERENCIA**), como la identificación de roles (gestor de demanda, gestor de catálogo de servicio, gestor de nivel de servicio, gestor de disponibilidad, gestor de capacidad, gestor de cambio, gestor de incidencia, gestor de problemas), alcance de procesos (como gestión, planificación, soporte, configuración, validación y evaluación), asignación de usuario, además de indicadores basados en ITIL.

2.2. Bases Teórico Científicas

2.2.1. Plataforma Colaborativa

También conocido como Groupware (GW), Bravo [20] menciona que una plataforma colaborativa es aquel software que soporta el trabajo grupal, sirviendo como un nuevo componente de comunicación, orientado a la colaboración en el proceso del trabajo en grupo, que permite aumentar la eficacia de la comunicación (al percibir el mismo concepto en la emisión y recepción de la información), de la coordinación (al establecer coherentes enlaces entre actividades) y de la cooperación (al participar intencional y coordinadamente).

Mandviwalla y Olfman citados en [21] , identificaron diversas áreas funcionales en los requerimientos genéricos para basar un modelo de groupware que son las siguientes:

- Soporte a tareas/trabajos múltiples.
- Soporte a diferentes métodos de trabajo.
- Memoria grupal.
- Administración de grupo.
- Diferentes métodos de interacción comunicativa.
- Soporte a diferentes comportamientos.
- Gestión de frontera permeables.
- Ajustable a contexto y necesidades del grupo.

Según Ellis, Gibbs y Rein en [22] existen dos clases de Plataformas Colaborativas:

- A. Orientado al espacio – tiempo; El sistema mejora la comunicación y colaboración de un grupo que se encuentra cara a cara o que se encuentran ubicados en diferentes lugares, mediante la interacción en tiempo real o asíncrona.

	Mismo Tiempo	Diferente Tiempo
Mismo Lugar	Interacción cara a cara	Interacción asíncrona
Diferente Tiempo	Interacción distribuida síncrona	Interacción distribuida asíncrona

Fig. 1. Matriz 2x2 de Groupware espacio temporal [22]

B. Orientado a la funcionalidad a nivel de aplicación; Se encuentra basado en las características que cubre un dominio de groupware, brindando una idea general de lo que abarca una plataforma colaborativa.

2.2.2. Proyecto de Software

Guerra y Bedini [23] explica que, un proyecto es un grupo de esfuerzos con limitaciones de tiempo y objetivos definidos, para lo cual se debe realizar diversos arreglos entre especialidades y recursos, por ello, cuando los objetivos de un proyecto son alcanzados es cuando el proyecto está completo.

Los proyectos de software tienen impacto directo e indirecto en la organización, ya que pueden encontrarse relacionados con otros proyectos, volverse obsoletos o contar con la participación de distinto personal durante su desarrollo. Por ello, Arellano [24] comenta que, al realizar un proyecto de software se obtiene compromisos con clientes y superiores por lo que se debe prever y aceptar aquello que se pueda cumplir y así poder desempeñar con lo planificado, ya que si la proyección no es realista existirán retrasos y pérdida de tiempo y dinero, por eso mismo, los proyectos de software deben pasar barreras como especificaciones escasas por parte de usuarios, restricciones temporales durante el desarrollo y equipo humano sin entrenamiento adecuado.

2.2.2.1. Ciclo de Vida del Software

El ciclo de vida del software es el proceso que conlleva la realización de diversas actividades y resultados en relación a un proyecto de software y lo que se requiera, desde su concepción hasta su desmantelamiento, y aunque existen diversas modalidades, todas tienen en común tres fases bien definidas, las cuales son planificación, implementación o desarrollo y producción o mantenimiento [25]-[18].

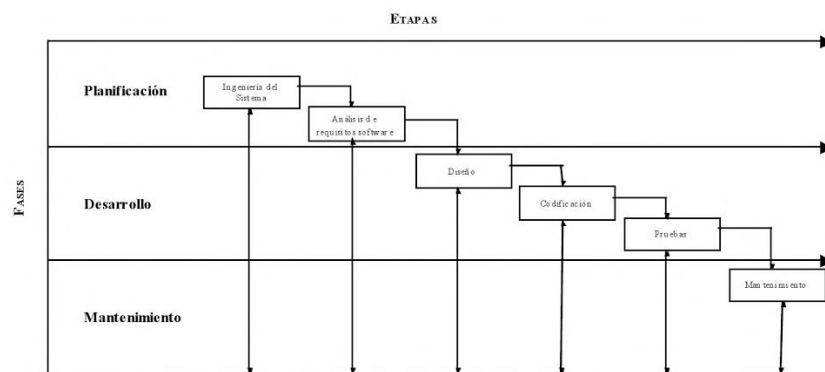


Fig. 2. Ciclo de vida en cascada [26]

Planificación. – Se planea detalladamente la gestión de un proyecto a nivel de tiempo y economía [25]. Esta fase contiene las etapas de Ingeniería del sistema y análisis de requisitos del software [26]. Es aquí donde se define las restricciones del software entre el Clientes e Ingenieros [27].

Implementación o Desarrollo. – Se detallan todas las actividades que implican la realización del producto [25]. Esta fase contiene las etapas de Diseño, Codificación y Pruebas [26].

Producción o Mantenimiento. – Se define el proyecto y se muestra al cliente y usuario final, cubriendo los requerimientos y evaluando el funcionamiento correcto [25]. Esta fase contiene la etapa de Mantenimiento [26]. En esta etapa se explota al software para

realizar los cambios necesarios, ya sea detectar o corregir errores, o realizar mejoras [28], las cuales al ser modificados y adaptados a los cambios requeridos por el cliente y el mercado permiten la evolución del software [27].

2.2.2.2. Desarrollo de Software

Un sistema de software es construido mediante un proceso de desarrollo formal o empírico [29], es decir, el desarrollo de software consiste en proyectar e implementar una solución de manera técnica acorde con la arquitectura elaborada y que cubra los requisitos brindados [30], es aquí donde se inicia con la codificación de algoritmos y estructura de datos, los cuales han sido predefinidos, todo esto en un apropiado lenguaje de programación con un sistema determinado como gestor de base de datos [25].

Los propósitos del desarrollo de software según Noriega [30] son:

- La transformación de requisitos en un proyecto de lo que ha de ser el sistema.
- La adaptación del proyecto a un entorno de implementación.
- La construcción del sistema incrementalmente mediante la generación de “builds”.
- La comprobación del trabajo, según las especificaciones, de las unidades técnicas que se usan para construir el sistema.

También explica que las tareas del desarrollo de software son:

- Integrar y crear build.
- Proyectar la solución.
- Implementar las pruebas de desarrollo.
- Implementar la solución.
- Ejecutar las pruebas de desarrollador.

Gómez y Moraleta [28] explican que, no necesariamente es normal que un software posea errores, pero que, si es susceptible a los mismos, aunque lo que no se puede admitir es un producto de

software con errores recurrentes por lo cual, este debe ser rechazado.

i. Java

Tecnología usada para el desarrollo de aplicaciones [31], la cual se encuentra en su plataforma Java 2 (Figura 3), es decir en su segunda versión. Los diversos programas de esta tecnología se ejecutan en una variedad de entornos mediante la plataforma JVM (Java Virtual Machine por sus siglas en inglés), el cual traduce el código Java en instrucciones específicas para cada dispositivo [32].

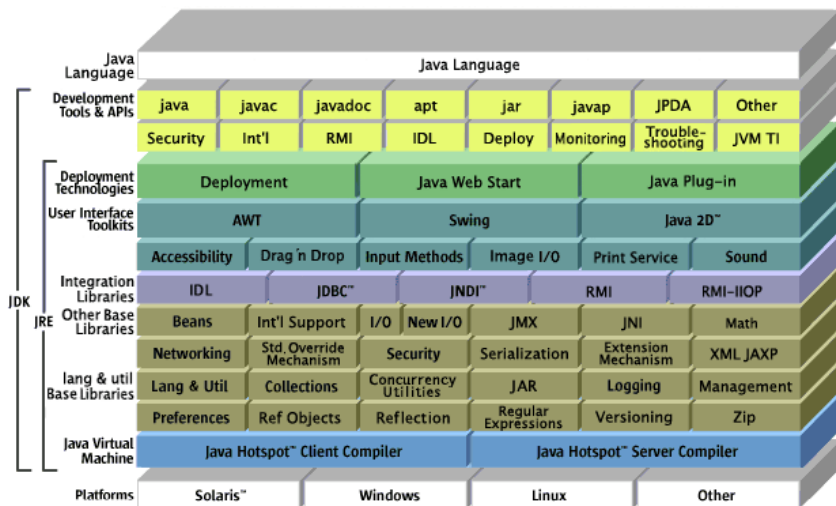


Fig. 3. Plataforma Java 2. Edición Estándar V 5.0 [33]

Dean y Dean [34] explican que, Java como lenguaje de programación, utiliza reglas definidas de sintaxis (palabras, gramática y puntuación), por lo que si el código fuente usa una sintaxis errónea (como olvidar un punto y coma, tratar a mayúsculas y minúsculas por igual o escribir incorrectamente una palabra) se presentará más de un inconveniente al intentar ejecutar el código fuente en un computador, esta sintaxis debe ser precisa y se encuentra aplicada en las instrucciones del código fuente, las cuales son llamadas por los programadores como *Sentencias Java*.

ii. Entorno de Desarrollo Integrado

También llamado IDE (por sus siglas en inglés, Integrated Development Environment), según Ávila [35], es una aplicación de software que brinda servicios integrales para el desarrollo de software a programadores, el cual está compuesto básicamente de un editor de código fuente, herramientas de compilación y un depurador, mientras que los modernos tienen ciertas características como ayuda on-line que permite el acceso del programador a la sintaxis y funcionamiento del lenguaje de programación en el que se encuentra codificando, además de un navegador de clases, explorador de objetos y la opción de herramientas de documentación en línea. El objetivo de un entorno es lograr que la productividad del programador se maximice, para lo cual se le brinda componentes específicos e interfaces de usuarios similares que faciliten su uso, además de reducir esfuerzos en su instalación y configuración.

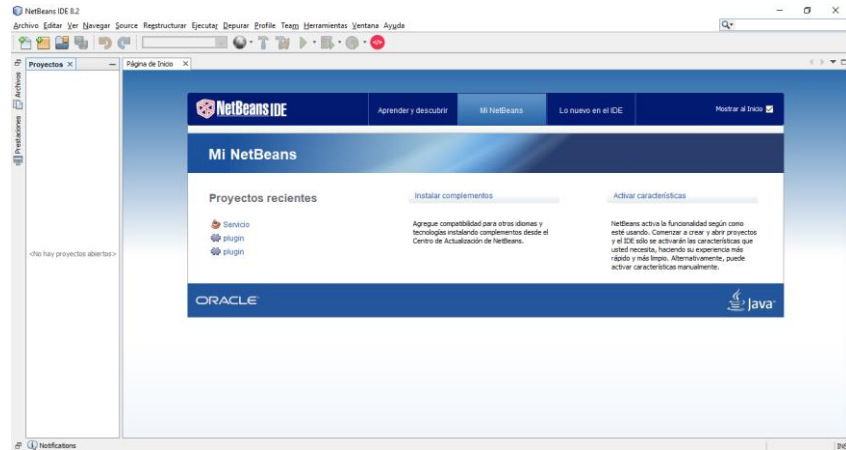


Fig. 4. Entorno de desarrollo integrado NetBeans
Fuente: Elaboración propia

Eslava [36] alude que, las características de un buen entorno de desarrollo son las siguiente:

- Resaltado de sintaxis como variables, funciones, textos, comentarios, etc., ya que permite más legibilidad y un mejor reconocimiento del código.

- Completado de código que permite al desarrollador reducir tiempo al momento de la codificación.
- Búsqueda de código que permite la reutilización de funciones o métodos.
- Resaltado de errores y advertencias que permitirá evitar los inconvenientes durante las pruebas.
- Refactorización y generación de código para reescribirlo y volverlo más sólido, formateado y con nomenclatura estándar.
- Depuración de código mediante el pintado de variables en los posibles lugares que se considera la existencia de un error.

iii. Plugin

Llamada también Complemento, Gris [37] y Peláez [38] mencionan que es una aplicación, un componente adicional que interactúa con otra aplicación y así aportarle una nueva función específica y determinada para enriquecerla. La aplicación principal ejecuta este complemento y se encuentra relacionados mediante un API.

Peláez comenta que las funciones de un plugin son las siguientes:

- Que la aplicación principal pueda extender sus funciones por desarrolladores exteriores.
- Que su tamaño sea menor de la aplicación principal.
- Que su código fuente sea separado a causa de incompatibilidad de licencias.

iv. Interfaz de Programación de Aplicaciones

Peláez [38] y Osorio [39] refieren que un API (por sus siglas en inglés de Application Programming Interface), es un compilado de métodos, funciones, procedimientos y subrutinas brindados por una biblioteca para ser usado como capa de abstracción por otro software. Además, Merino [40] expresa que las API tiene un gran valor ya que permiten utilizar las funciones ya existentes de la infraestructura de otro software, lo que concede la reutilización de

un código probado y correctamente funcional cuando es de código abierto, pero cuando no, se puede añadir funcionalidades sin necesidad de conocer el proceso interno de codificación.

API REST

Es un principio de arquitectura, estilo y patrones que sirve de guía para el diseño de interfaces [41], el modelo API REST (por sus siglas en inglés de Representational State Transfer) ha sido propulsado por el enfoque de consumir datos remotos en el desarrollo de aplicaciones, el mismo que no es dependiente de un acceso directo a una base de datos, además de independizar a los servidores de tu tecnología, lo que permite el intercambio de información en estándares como XML o JSON [42].

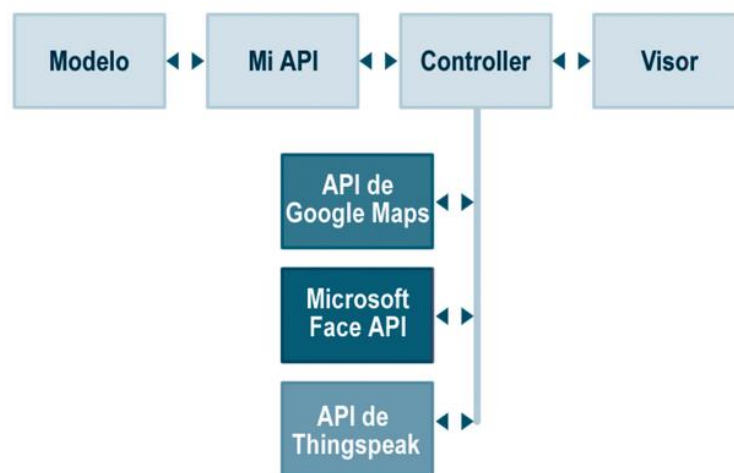


Fig. 5. Modelo de implementación de un sistema MVC utilizando API REST [42]

Luna, Peña e Iacono sostienen que las características de un API REST son las siguientes:

- Separación de acciones entre el cliente y el servidor que permite portar una interfaz desarrollada hacia otro estándar o plataforma sin necesidad de interactuar con los datos definidos y así todos los componentes podrían evolucionar de forma independiente.

- Visibilidad, fiabilidad y escalabilidad, para que el desarrollo escale un producto de software mediante la separación de componentes, lo que permite migrar datos a otros servidores o cambiar la base de datos mediante peticiones remotas.
- Independización de plataformas y lenguajes, ya que se adapta a cualquier tipo de sintaxis o plataformas con la que se trabaje.

2.2.3. Metodología Scrum

Schwaber y Sutherland [43], manifiestan que es un marco de trabajo que permite el uso de varios procesos y técnicas de gestión de productos y trabajo con la finalidad de mejorar continuamente al producto, equipo y entorno laboral, basado en equipos scrum, roles, eventos, artefactos y reglas incorporadas mediante el uso de un enfoque iterativo e incremental para la optimización de la predicción y control de riesgos, lo que vuelve al equipo flexible y adaptativo.

2.2.3.1. Scrum Team

El equipo Scrum está conformado por el dueño del producto, el equipo de desarrollo y un scrum master, este equipo presenta entregables de forma iterativa e incremental por lo que es autoorganizado y multifuncional, es decir, realiza su trabajo de la mejor manera y sin necesidad de una dirección externa, además de tener las competencias requeridas para no depender de personas que no son del equipo.

- A. Product Owner; Es el dueño del producto y único responsable de gestionar el Product Backlog, ordenando y expresando de forma clara los elementos para que se efectúe un trabajo óptimo por parte del equipo de desarrollo.
- B. Development Team; Es el equipo de desarrollo y responsable de la elaboración y entrega incremental de un producto “finalizado” por cada sprint, por lo que deben ser autoorganizados, multifuncionales, independientes y

funcionar como un todo, sin un rango determinado ni títulos individuales por habilidades específicas. El tamaño ideal para un equipo de desarrollo es de 3 a 9 integrantes, esto permite agilidad y carga significativa de trabajo para el desarrollo potencial de un producto.

- C. Scrum Master; Es el líder del equipo y responsable de las coordinaciones entre las personas externas y el equipo Scrum. El Scrum Master asegura el entendimiento de los objetivos, alcance y dominio para el dueño del producto, guía al equipo de desarrollo y planifica la implementación y adopción de Scrum en la organización.

2.2.3.2. Eventos de Scrum

Son bloques de tiempo o time-boxes con duración fija y un tiempo máximo, los cuales tienen la finalidad de regular y minimizar las reuniones innecesarias o que no han sido acordadas con anticipación.

- A. Sprint; Bloque de tiempo o time-box con una duración máxima de un mes en el que se da una creación incremental de producto temporalmente acabado que pueda ser usable y tenga un despliegue potencial.
- B. Sprint Planning; La Planificación de Sprint permite es cuando todo el Equipo Scrum realiza planificación del trabajo a realizar durante el sprint. Aquí se responden preguntas como: ¿Qué se puede hacer en este Sprint? ¿Cómo se logrará cumplir con el objetivo?
- C. Sprint Goal; El objetivo del Sprint es cuando se establece una meta por Sprint, la cual debe ser alcanzada con la implementación de la Lista de Producto. Aquí se responde a la pregunta: ¿Por qué se construye el incremento?

- D. Daily Scrum; El Scrum Diario es cuando el equipo de desarrollo se reúne cada día del Sprint con un tiempo máximo de 15 minutos, para la evaluación del progreso hacia el objetivo y la planificación del trabajo para el día siguiente. Aquí se podría responder las siguientes preguntas: ¿Qué hice ayer que ayudó al Equipo de Desarrollo a lograr el Objetivo del Sprint? ¿Qué haré hoy para ayudar al Equipo de Desarrollo a lograr el Objetivo del Sprint? ¿Veó algún impedimento que evite que el Equipo de Desarrollo o yo logremos el Objetivo del Sprint?

- E. Sprint Review; La Revisión del Sprint se desarrolla al final de cada Sprint para realizar la inspección del Incremento y su adaptación a la Lista de Producto. La reunión se da entre el Equipo Scrum y los interesados invitados por el Dueño del Producto, quien es el único que explica los elementos finalizados y no finalizados de la Lista de Producto, El Equipo de Desarrollo explica los inconvenientes presentados durante el Sprint y realiza una demostración del mismo.

- F. Sprint Retrospective; La Retrospectiva de Sprint sirve para realizar una autoinspección del Equipo Scrum y para crear un plan de mejoras para el siguiente Sprint. Su propósito es: Inspeccionar el progreso del Sprint en cuanto personas, relaciones, procesos y herramientas; Identificar y ordenan elementos importantes que salieron bien y sus posibles mejoras; Crear un plan de implementación de mejoras en el desempeño del trabajo del Equipo Scrum.

2.2.3.3. Artefacto de Scrum

Son las formas de trabajo o valor útiles para realizar la inspección y adaptación con transparencia y oportunidad de asegurar su entendimiento.

- A. Product Backlog; La lista de Producto es definida por el Dueño del Producto y contiene todos los requerimientos del producto, lo que permite cualquier cambio del mismo, esta lista enumera características, funcionalidades, requisitos, mejoras y correcciones, las cuales tienen atributos como descripción, orden, estimación y valor.
- B. Sprint Backlog; La Lista de Pendientes del Sprint es realizada por el Equipo de Desarrollo y son los elementos para el Sprint seleccionados de la Lista de Producto junto a un plan de entrega incremental del producto y el logro del objetivo.
- C. Incremento; Es la unión de todos los elementos finalizados de la Lista de Producto por cada Sprint, es entregado por el Equipo de Desarrollo y debe ser utilizable sin inconveniente alguno.

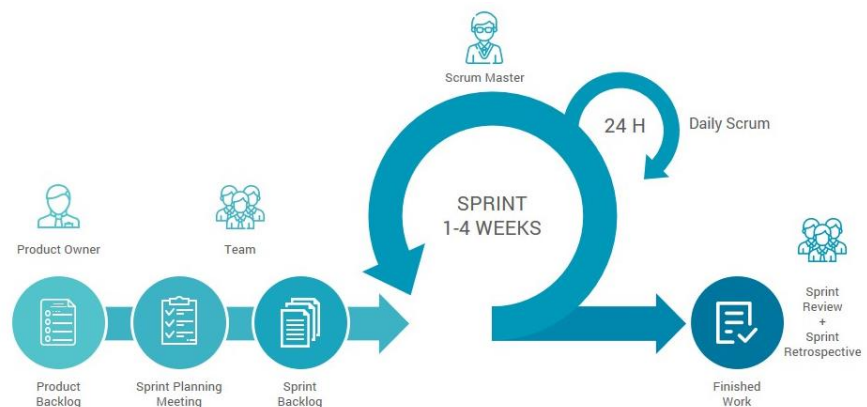


Fig. 6. Proceso Scrum [44]

2.2.3.4. Ventajas de la Metodología Scrum

Ruiz [45], expone que las principales ventajas del uso de SCRUM son:

- A. Es simple ya que no se necesita mucho tiempo para ser aprendido.

- B. Es operativa y flexible ya que a medida que se desarrolla el sprint, se respeta el trabajo del equipo de desarrollo y se permite la realización de cambios.
- C. Permite minimizar riesgos al implementarse ciclos cortos de desarrollo.
- D. Permite obtener resultados a corto plazo, ya que se proporciona historias de usuarios funcionales por cada sprint finalizado.
- E. Es óptima, ya que permite realizar ordenar la pila en base a prioridades.

2.2.4. Biblioteca de Infraestructura de Tecnologías de Información

Ríos [46], manifiesta que ITIL (por sus siglas en inglés de Information Technology Infrastructure Library) es una librería de buenas prácticas para gestionar los servicios de Tecnologías de la información (TI) y así mantener la alineación con los objetivos del negocio. El ciclo de vida de ITIL está conformado por:

- A. Estrategia de Servicios (Service Strategy – SE); Se diseña el plan de acción a implementar.
- B. Diseño de Servicio (Service Design); Se desarrollan conceptos como arquitectura proceso, políticas y documentación.
- C. Operaciones de Servicio (Service Operation – SO); Se muestran las mejores prácticas a realizar para lograr brindar un nivel de servicio acorde con los requerimientos del cliente.
- D. Mejora continua de Servicios (Continual Service Improvement – CSI); Se da la mejora continua tanto interna como externa durante el crecimiento y desarrollo del servicio mediante el análisis de sus procesos.
- E. Transición de servicios (Service Transition – ST); Se define la transición de servicios como, gestión de la configuración y servicio de activos, la planificación de la transición y de apoyo, gestión y despliegue de los Servicios TI, Gestión del Cambio,

Gestión del Conocimiento y, responsabilidades y funciones de las personas participantes.



Fig. 7. Ciclo de Vida de ITIL [46]

2.2.4.2. Gestión del Servicio

Consiste en acordar con el cliente el registrar todos los acontecimientos bajo un marco de referencia con la finalidad de realizar un Servicio TI de calidad, para lo cual se debe llevar un proceso documentado de planificación, seguimiento, acuerdos, informes, revisiones, verificaciones y validación. Es así que se va realizado un ciclo de mejora, en base a las fases de gestión.

- A. Planificación; abarca el catálogo de servicios, requisitos de nivel de servicio, hojas de especificación del servicio y plan de calidad del servicio.
- B. Realización; abarca el acuerdo de nivel de servicio, acuerdo de nivel de operación y contratos de soporte.
- C. Verificación; es parte del seguimiento y abarca incidencias y problemas detectados, quejas y reclamaciones de los usuarios, disponibilidad del servicio, tiempos de respuesta, escalas de capacidad y control de proveedores externos.

- D. Validación; también es parte del seguimiento y abarca lo mismo que la verificación.
- E. Mejora; abarca las mejoras a realizar, los plazos de prioridad, costes y responsabilidades con el objetivo de mantener una revisión constante del servicio.

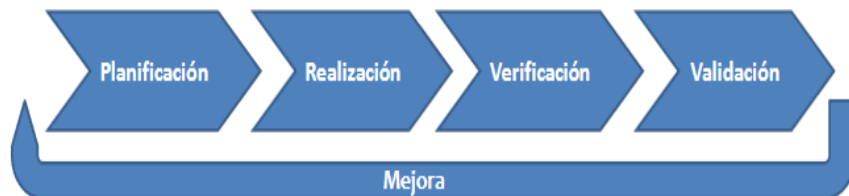


Fig. 8. Ciclo de Gestión del Servicio [46]

2.2.4.3. Catálogo de Servicios

Es la identificación de la cartera de servicios de TI, la cual sirve para describir los servicios ofrecidos por la empresa y sea entendible al cliente y así conocer los requerimientos que puede cubrir. [46]

2.2.4.4. Programa de Mejora del Servicio

Es el reflejo de las mejoras que se van a realizar, con la indicación de plazos, prioridades, costes y responsabilidades, con la finalidad de ser presentados en un informe con el siguiente esquema base:

- Proceso o Servicio tratado
- Responsable del proceso o servicio
- Propietario de la mejora
- Autorización de la dirección
- Descripción de la iniciativa
- Origen de la medida
- Detalle de resultados
- Situación actual

2.2.4.5. Gestión de Incidencias

Ríos [46] agrega que es la solución de incidentes registrados con el objetivo de una pronta restauración del servicio alterado, logrando

mejorar la satisfacción del cliente, generar más conocimiento y mejorar el rendimiento del servicio. El proceso para la gestión de incidencias es el siguiente:

- A. Recepción y registro; Se debe agregar los servicios afectados, posibles causas, nivel de prioridad, impacto, recursos asignados para su resolución y estado de la incidencia.
- B. Clasificación; Se debe clasificar a la incidencia por su urgencia, impacto y prioridad de solución en la organización y realizar las asignaciones según acuerdos.
- C. Comparación; Se debe realizar una búsqueda de incidencias similares en la base de datos (BBDD), si no es encontrada, se debe realizar una fase interna de investigación y diagnóstico de la incidencia.
- D. Seguimiento; Se debe conocer el estado de la incidencia respecto a su solución.
- E. Resolución; Se debe realizar un almacenamiento y actualización de la base de datos una vez resuelta una incidencia.
- F. Cierre; Se debe realizar acciones de cierre y finalizar el proceso mediante la comunicación al usuario y actualización de la base de datos de la gestión de la configuración.

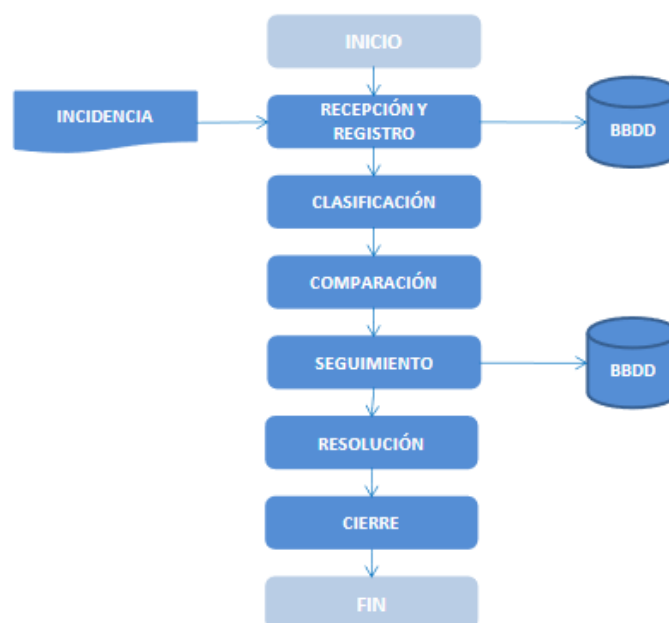


Fig. 9. Proceso de gestión de incidencias [46]

2.2.4.6. Métricas

Sirven para controlar, medir y analizar los objetivos del servicio, la gestión del servicio y sistemas asociados. Se pueden usar métricas como, número de Problemas identificados causados por las faltas de capacidad, número de Cambios realizados para rectificar los problemas de falta de capacidad, volumen de incidentes, pausas, paradas, o fallos en el Servicio por exceso de peticiones, % de inversión en componentes para infraestructura en años anteriores, Etc. [46]

2.2.4.7. Base de Datos de la Gestión de la Configuración

La CMDB (por sus siglas en inglés de Configuration Management Data Base) según Mora [47], sirve para almacenar información de todos los componentes de TI y su impacto, es decir, los atributos necesarios como tipo de elementos, estado, asignación y sistemas relacionados. Es importante como referente en la toma de decisiones, por lo que se debe mantener actualizado de forma continua y automática, ya que tiene información centralizada a la que puede acceder los participantes de TI.

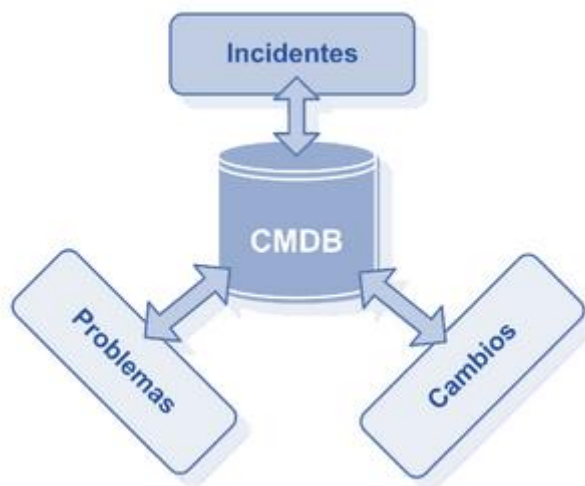


Fig. 10. Dependencias de la Base de Datos de la Gestión de la configuración [47]

2.2.5. Sistema de Gestión de Base de Datos no Relacionales

Llamada también NoSQL (por sus siglas en inglés de Not Only SQL), según Joyanes [48] es una base de datos que no sigue el modelo entidad – relación y puede manipular rápidamente gran volumen de datos, es usado como soporte por Facebook, Twitter, LinkedIn, Google, Yahoo.

Además, menciona características como:

- No utiliza SQL como lenguaje de consulta.
- No utiliza tablas para el almacenamiento de datos.
- Almacena gran volumen de datos con acceso rápido.
- Distribuye y manipula datos no estructurados.
- No pierde rendimiento por su escalabilidad lineal.
- Carece de un esquema predeterminado.

```
{
  Nombre: "Tipos de Base de Datos NoSQL ",
  Tipo: "Articulo",
  Categorías: [{
    Titulo: "Orientada a Documentos",
    Letras: 29
  },
  {
    Titulo: "Orientada a Columnas",
    Lineas: 50
  }]
}
```

Fig. 11. Definición de un elemento con JSON [49]

Mientras que Amazon [50] indica que los tipos de base de datos NoSQL son:

- Clave-valor; son base de datos divisibles y permite el escalado lineal, este tipo es aplicado en IoT, juegos y tecnología publicitaria.
- Documento; son base de datos intuitivas presentadas en un documento JSON conservando datos mediante el uso de un formato similar.

- Grafos; son base de datos que facilitan la creación y ejecución de aplicaciones que funcionan con un gran volumen de datos conectados, este tipo es aplicado en redes sociales, motores de recomendación, descubrimiento de fraude y gráficos de conocimiento.
- En memoria; son base de datos que facilitan tablas de clasificación, análisis en tiempo real y tiendas de sesión, lo cual permite adquirir altos picos de tráfico y requiere tiempos de respuestas de microsegundos.

La implementación de una base de datos no relacional tiene sus pro y contra como se muestra en la *Tabla I*, además de diferir con una base de datos relacional en características como las que se muestra en la *Tabla II* y en la sintaxis de sus consultas como se muestra en la *Tabla III*.

Tabla I. Ventajas y desventajas de NoSQL [51] [52]

Ventajas	Desventajas
- Estructura simple y flexible sin necesidad de un esquema.	- Ausencia de herramientas para reportes de análisis y pruebas de rendimiento.
- Puede incluir almacén de columnas, documentos, valores clave, gráficos, XML y objetos.	- Incompatibilidad con instrucciones SQL.
- Brinda una clave por cada valor en la base de datos.	- Ausencia de estándares para migración en NoSQL.
- No requieren costo de licencia y se puede ejecutar en software de bajos recursos.	- La tecnología se encuentra en maduración.
- Escalado y distribución horizontal en la carga de nodos.	- Inconveniente para datos interconectados.
- Se puede crear una base de datos sin necesidad de un modelo detallado.	

Tabla II. Diferencias entre SQL y NoSQL [50]

Características	SQL	NoSQL
Cargas de trabajo óptimas	Procesa transacciones OLTP altamente coherentes y transaccionales y procesamiento analítico OLAP.	Diseñada bajo una serie de patrones de acceso a datos para OLTP y realiza análisis de datos semiestructurados.
Modelo de datos	Realiza la normalización de datos en tablas y la relación entre ellas.	Brinda modelo variado de datos.
Propiedades ACID (atomicidad, coherencia, aislamiento y durabilidad)	Requiere ejecución absoluta, acoplamiento de esquema, ejecución de transacciones por separado y requiere capacidad de recuperación.	Brinda flexibilidad para algunas propiedades, lo que ocasiona baja latencia y alto rendimiento.
Rendimiento	Depende del subsistema del disco.	Depende del tamaño del clúster.
Escalado	Es de forma ascendente.	Es mediante arquitectura distribuida.
Api	Solicita el almacenamiento y recuperación mediante consultas.	Permite el almacenamiento y recuperación de estructuras en memoria.

Tabla III. Comparación de sintaxis en consultas de SQL y NoSQL [53]

Consulta	SQL	NoSQL
Crear base de datos	Use db_name	CREATE DATABASE db_name
Eliminar base de datos	db. dropDatabase	DROP DATABASE db_name
Insertar datos	db. Db_name. insert ({{title: "NoSQL"}})	INSERT INTO table_name VALUES ("SQL")
Actualizar datos	db. Db_name. update ({{title: "NoSQL"}}, {\$set: {{title: "NewNoSQL"}}})	UPDATE table_name SET title = 'NewNoSQL'

2.2.6. MEAN Stack

Es un marco Full Stack de JavaScript compuesto por MongoDB, Express.js, Angular.js y Node.js, es decir, es la combinación de marcos de trabajo, base de datos y tecnologías adicionales, las cuales sirven para la creación de aplicaciones web robustas, rápidas y sostenibles lo cual brinda la opción de implementar soluciones de

código abierto legible y simple [54] [55]. Anteriormente, desarrollar software en JavaScript, solo se podía realizar del lado del cliente, pero en la actualidad este lenguaje de programación también puede ser desarrollado por el lado de servidores y base de datos, lo que permite desarrollar toda una aplicación web con un solo lenguaje. [56]

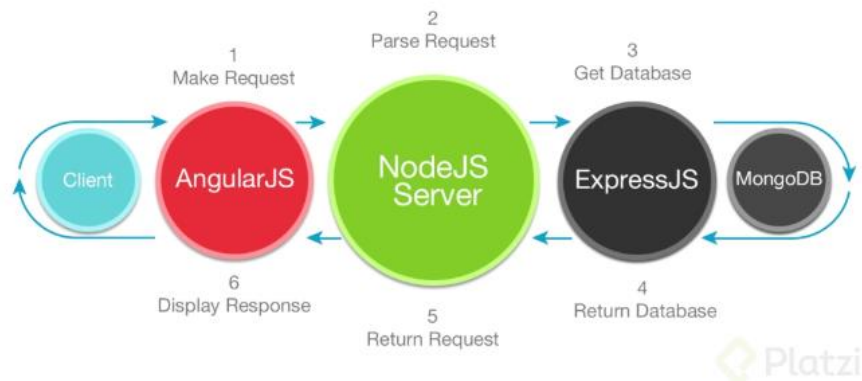


Fig. 12. Arquitectura del Stack MEAN [57]

Al realizar una consulta en Google Trends (Fig. 13) se puede observar la popularidad incremental en los últimos cinco años que van teniendo las tecnologías del stack en Perú, además de su uso en diversas empresas conocidas internacionalmente como Microsoft, PayPal, Netflix, Cisco, Uber, Yahoo, LinkedIn, eBay, Walmart, Medium, IBM, Nasa, Mozilla y Trello.

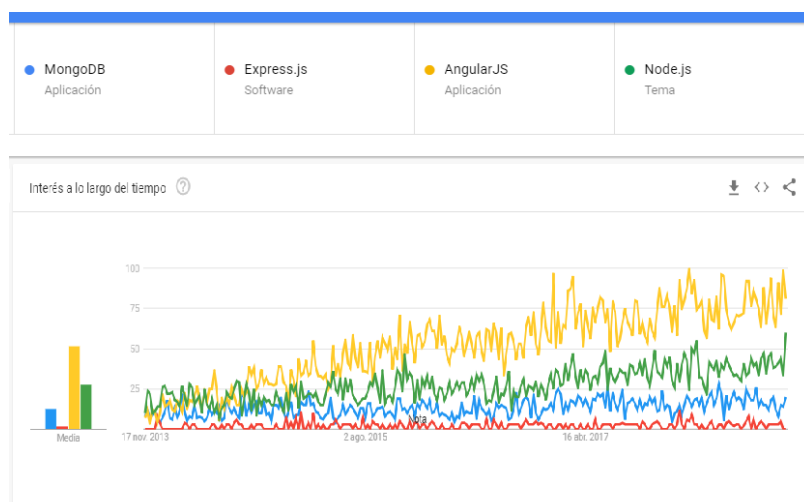


Fig. 13. Niveles de interés de las Tecnologías MEAN en el Perú [58]

Inubo [59] muestra casos de éxito en aplicaciones como:

- Mozilla; aplicación de navegación web con más de 100 millones de usuarios a nivel mundial, declaró que los equipos lograron trabajar rápido y ser productivos, además de lograr tener una memoria con poder de soporte de un millón de usuarios.
- Netflix; aplicación de servicio de streaming y contenidos de video con más de 117 millones de usuarios a nivel mundial, declaró que disminuyó sus tiempos de compilación, pudieron habilitar la personalización de usuario y mejoraron su tiempo de carga en un 70%.
- PayPal; aplicación de pagos en línea con más de 200 millones de usuarios a nivel mundial, declaró que su aplicación se desarrolló 2 veces más rápido con 33% menos código.
- LinkedIn; aplicación social de perfil de negocios con más de 450 millones de usuarios a nivel mundial, declaró que logró la eficiencia de rendimiento y escala reduciendo de 15 servidores a 4, doblando la capacidad de tráfico y logrando un funcionamiento de 2 a 10 veces más rápido por lado del cliente.

2.2.6.1. MongoDB

Gestor de base datos NoSQL licenciado por GNU AGPL (por sus siglas en inglés de Affero General Public License) v.3.0, es multiplataforma y de código abierto, orientado a la documentación y esquemas libres [56], es usado por empresas como Cisco, eBay, Adobe, Amazon, VMware, EA, Facebook, Bosch, Nokia, Telefónica y Google.

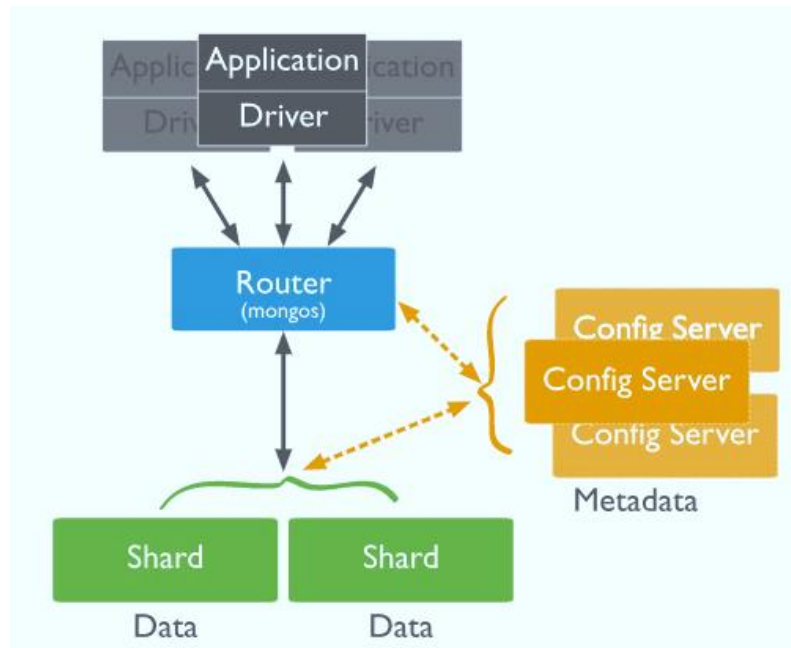


Fig. 14. Arquitectura de clústeres fragmentados en MongoDB [60]

Características de MongoDB [61]

- Almacenamiento de datos en documento de Notación de Objetos de JavaScript (JSON por sus siglas en inglés de JavaScript Object Notation) que permite la variación de la estructura de sus datos y de los campos de un documento a otro
- Facilidad de trabajo con datos al permitir la asignación de objetos en el código de la aplicación en desarrollo.
- Facilidad de acceso y análisis de datos mediante indexación, consultas ad hoc y agregación en tiempo real.
- Facilidad de uso por ser una base de datos distribuida en su Core con distribución geográfica incorporada, con gran disponibilidad, de fácil uso y escalado horizontal.
- Facilidad de conexión al brindar controladores para más de diez lenguajes de programación.


```

JAVA.js
1 // 1. Conectar a la instancia de MongoDB que se ejecuta en localhost
2 MongoClient mongoClient = new MongoClient();
3
4 // Acceso a la base de datos llamada 'prueba'
5 MongoDBDatabase database = mongoClient.getDatabase("prueba");
6
7 // Acceso colección llamada 'incidencias'
8 MongoCollection collection = database.getCollection("incidencias");

JavaScript.js
1 // 1. Conectar a la instancia de MongoDB que se ejecuta en localhost
2 // Conexión URL
3 var url = 'mongodb://localhost:27017/prueba';
4
5 co(function*() {
6   const db = yield MongoClient.connect(url);
7   console.log("Conectado exitosamente al servidor");
8
9   yield insertDocuments(db, null);
10  yield findDocuments(db, null);
11  yield indexCollection(db, null);
12  yield aggregateDocuments(db, null);
13
14  db.close();
15 }).catch(err => console.log(err));

```

Fig. 15. Conexión a base de datos local de MongoDB en Java y JavaScript.
Fuente: Elaboración propia.

2.2.6.2. Express.JS

Framework (marco de trabajo) de desarrollo flexible, rápido y asíncrono, basado en Sinatra, para Apis y aplicaciones web basados en Node.JS, es de código abierto lanzado bajo la licencia MIT y parte del backend(desarrollo del lado del servidor) del stack MEAN, además de ser base de varios otros marcos como Feathers, ItemAPI, KeystoneJs, Kraken, LoopBack, Dinoloop y Kites. [56] [62]

```

//Crear directorio para guardar aplicación
$ mkdir miaplicacion
//Volverlo directorio principal
$ cd myaplicacion
//Utilizar el siguiente comando para crear un archivo package.json
$ npm init
//Ingresar la siguiente excepción con el nombre del archivo principal
entry point: (index.js)
//Instalar express en el directorio principal y guardarlo en la lista de dependencia
$ npm install express --save

```

Fig. 16. Líneas de comando para instalación de Express [63]

2.2.6.3. Angular.JS

Framework de desarrollo creado por Google orientado a JavaScript, es de código abierto y permite facilitar el desarrollo de aplicaciones web Spa (web de una sola página), brindando herramientas que permitan trabajar de forma sencilla y optimizar el trabajo. Proporciona un marco arquitectónico MVC (modelo - vista - controlador) y MVVM (modelo – vista - vista - modelo) del lado del cliente, ya que es parte frontend (desarrollo del lado del cliente) de la pila de MEAN. [56]

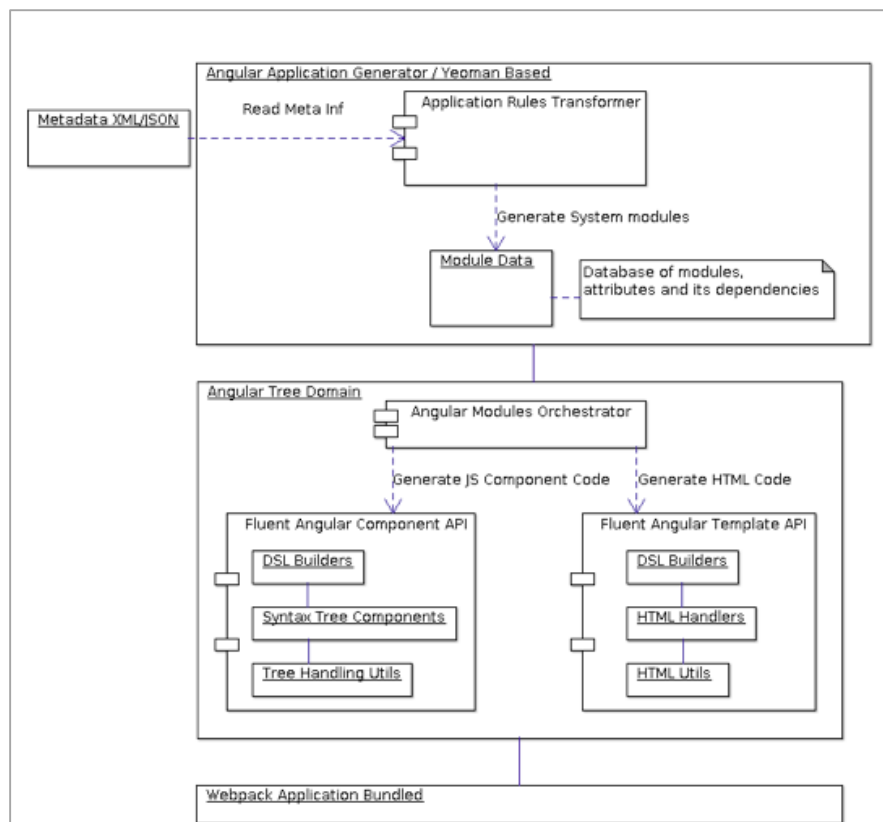


Fig. 17. Arquitectura de Angular.JS [64]

Características de Angular.JS [65]

- Arquitectura MVC que facilita una vista fluida y dinámica.
- Construcción de interfaz de usuario mediante el uso de HTML (siglas en inglés de Hyper Text Markup Language).
- Objetos funcionales y autosuficientes basados en modelo POJO (siglas en inglés de Plain Old Java Object).

- Requiere de poca codificación por lo que brinda directivas para el controlador.
- Realización de paginado y filtración de matriz de datos respecto a sus parámetros.
- Aplicación unificada mediante inyección de dependencias, que permite el adecuado funcionamiento y administración de control y alcance.
- Modificación y manipulación del DOM (siglas en inglés de Document Object Model) mediante la vista para el comportamiento y actualización de datos.

```

/**
 * Declaration of the AST object handlers.
 */
var ClassSyntaxTree = require('./astmodel/ClassSyntaxTree');
var RequireSyntaxTree = require('./astmodel/RequireSyntaxTree');
var Exp module export=
  (property) export=: typeof ClassBuilder
  (property) export=: typeof ClassBuilder
module.exports = class ClassBuilder {

  constructor(className) {
    this.classST = new ClassSyntaxTree(className);

    this.className = className;

    this._bodyClassBuilder = null;
  }

  addRequire(variable, module) {
    var requireST = new RequireSyntaxTree(variable, module);
    this.classST.addRequireSyntaxTree(requireST);
    return this;
  }
}

```

Fig. 18. Generador de clases en Angular [64]

2.2.6.4. Node.Js

Entorno de ejecución de JavaScript por parte del backend, de código abierto con arquitectura por eventos y sin bloqueo, desarrollado basado en el uso del motor JavaScript V8 para Chrome, lo que permite tener un escenario de hasta un millón de conexiones simultáneas en un servidor con un consumo de dos megabytes de memoria en un sistema usual de ocho gigabytes de RAM. [56] [66]

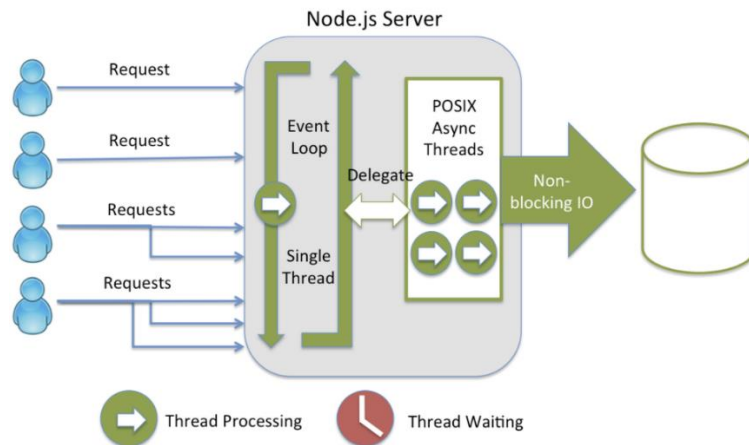


Fig. 19. Arquitectura de servidor Node.JS [67]

Características de Node.JS [66]

- Permite que los datos JSON funcionen sin conversión de datos ni resistencia para coincidencias.
- Permite el manejo de conexiones concurrentes, reconociendo el comportamiento del cliente antes de que los datos pasen a la base de datos.
- Permite el streaming de datos por procesamiento de archivos mientras aún están cargando, mediante secuencia de datos y procesamientos en línea.
- Permite manejar diversidad de conexiones simultáneas sin bloqueo, lo que permite distribuir distintos servicios, en distintos tiempos de respuesta y recopilar gran cantidad de datos en distintos puntos de origen.
- Permite el monitoreo del Dashboard (Interfaz gráfica de usuario) mediante la recopilación en tiempo real de estadísticas de usuario con sockets bidireccionales.

```
const http = require('http');
const hostname = '127.0.0.1';
const port = 3000;
const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hola Carol\n');});
server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);});
```

Fig. 20. Creación de servidor web con Node.JS [68]

III. METODOLOGÍA

3.1. Tipo y nivel de investigación

3.1.1. Tipo de investigación

Tecnológica aplicada, ya que se pretende implementar una plataforma colaborativa para apoyar el desarrollo de proyectos informáticos mediante la administración de repositorios de incidencias, manejo de perfiles, consultas desde el IDE de trabajo e interacción entre los participantes del proyecto.

3.1.2. Nivel de investigación

Cuasi experimental, porque se realiza pre Y post test, utilizando herramientas de recolección de datos, que permitan comparar los cambios que la aplicación de la propuesta generaría.

3.2. Diseño de investigación

Por contrastación de Hipótesis la investigación es de tipo experimental de Pre test – Post test, ya que se realizará un control de los indicadores antes y después de implementar la plataforma colaborativa.

G: O1 X O2

X: Variable Experimental: Plataforma Colaborativa

O1: Aplicación del Pre – Test: Proceso de desarrollo de Software.

O2: Aplicación del Post – Test: Proceso de desarrollo de Software.

3.3. Población, muestra y muestreo

3.3.1. Población

Trabajadores de la División de Tecnologías de Información

3.3.2. Muestra

La población constituyó la muestra.

3.3.3. Muestreo

No se aplicó técnica de muestreo ya que la población constituyó la muestra.

3.4. Criterios de selección

Se eligió al personal de la Unidad de Desarrollo de Sistemas, el cual está conformado por tres (3) Ingenieros de Sistemas, ya que solo ellos son quienes efectúan el proceso de desarrollo de todos los sistemas con los que cuenta la entidad.

3.5. Operacionalización de variables

Las variables que se han utilizado como elementos básicos en el desarrollo de la hipótesis están identificadas de la siguiente manera:

3.5.1. Variables

3.5.1.1. Variable independiente

Plataforma colaborativa

3.5.1.2. Variable dependiente

Proceso de desarrollo de software

3.5.2. Indicadores

Tabla IV. Indicadores

Objetivo específico	Indicador(es)	Definición conceptual	Unidad de medida	Instrumento	Definición operacional
Disminuir la cantidad de fuentes de búsqueda de solución de incidencias	Cantidad de fuentes de consulta de solución.	Indica la cantidad de fuentes de consulta de solución para una incidencia.	Número	Entrevista Observación	Número de fuentes de consulta actualmente.
Disminuir el tiempo promedio de búsqueda de solución a incidencias	Tiempo promedio de búsqueda de solución a incidencias	Indica el tiempo de búsqueda de solución para una incidencia.	Minutos	Entrevista Observación	HES – HIB HIB: Hora inicial de búsqueda. HES: Hora que encuentra solución.
Aumentar la cantidad de métodos y consultas registrados.	Cantidad de métodos y consultas almacenadas en un repositorio de soluciones.	Indica la cantidad de métodos y consultas almacenadas en un repositorio	Número	Entrevista Observación	$(\sum \text{MCR}) / 2$ MCR = Total de métodos y consultas registrados actualmente.
Aumentar la cantidad de incidencias registradas por sistema desarrollado en la entidad.	Cantidad de soluciones por incidencia registradas en el repositorio.	Indica la cantidad de incidencias resueltas registradas en el repositorio.	Número	Entrevista Observación	$(\sum \text{SR}) / \text{TI}$ SR = Total de soluciones registradas actualmente. TI = Total de incidencias.
Aumentar la cantidad de soluciones guardadas en una biblioteca de software definitivo.	Cantidad de incidencias registradas por sistema desarrollado.	Indica la cantidad de incidencias registradas por sistema desarrollado.	Número	Entrevista Observación	$(\sum \text{SG}) / \text{TS}$ SG = Total de soluciones guardadas actualmente. TS = Total de sistemas.

3.6. Técnicas e instrumentos de recolección de datos

A continuación, en la siguiente tabla se muestra las técnicas e instrumentos que fueron útiles para la recolección de datos.

Tabla V. Técnicas e instrumentos de recolección de datos

Técnicas	Instrumentos	Elementos de la población	Propósito
Control de tiempos	Observación	Trabajadores de la Unidad de Desarrollo de Sistemas	Conocer el proceso de gestión de los indicadores y tener una visión clara y directa del problema y la causas de los mismos
Cuestionario	Entrevista	Trabajadores de la Unidad de Desarrollo de Sistemas	Tener un dominio más claro del negocio, conocer sus procesos y los inconvenientes presentados por narración de sus actores

3.7. Procedimientos

La metodología utilizada para el desarrollo de la tesis es la metodología SCRUM y las etapas a seguir son las siguientes:

- A. Desarrollo del Product Backlog; Se definió el Product Backlog e Historia del proyecto.
- B. Construcción del Product Backlog; Se listó los requerimientos funcionales y no funcionales.
- C. Priorización del Product Backlog; Se definió las prioridades del Product Backlog por cada User Story.
- D. Identificación de la complejidad; Se identificó la complejidad del Product Backlog para los User Story.
- E. Asignación de valor por cada Story Point para cada User Story; Se determinó el esfuerzo del Product Backlog aplicando Planing Poker por cada User Story.
- F. Duración en día del Sprint; Se definió los días que van a demorar para la presentación de cada Sprint.

- G. Extracción del User Story más representativo; Se listó el Product Backlog y luego se eligió el User Story más representativo en base a prioridades y esfuerzos.
- H. Listar los User Story atendidos por un Sprint; En función del User Story más representativo se evaluó cuantos User Story pueden ser atendidos en un sprint con duración de 15 días, para lo cual se sumó esfuerzos y se extrajo la cantidad máxima.
- I. Calcular el número total de Sprints; Se realizó el cálculo del total de los Sprints en el proyecto, esto se obtuvo dividiendo el número total de User Points entre el número de User Point máximo en un sprint.
- J. Calcular el tiempo total de entrega; Se calculó los días estimados para la realización del proyecto mediante la multiplicación del número total de Sprint por los días de duración de un Sprint.
- K. Elaborar y agrupar los Sprint; Se identificó los User Story a desarrollar por cada sprint, tomando en cuenta la prioridad inicial.
- L. Desarrollo de los Sprint; Se desarrolló cada Sprint de manera individual en los tiempos definidos.
- i. Sprint N; Se implementó un total de 3 Sprints para el proyecto y por cada uno, se realizan las siguientes etapas de desarrollo:
 - Etapa de planificación; Se definió los ajustes realizados en el Sprint anterior (cuando es el primer Sprint solo se detalla las herramientas a utilizar) y se realizó la pila del Sprint, las historias y tareas de usuario.
 - Etapa de análisis; Se definieron los actores y se realizó los diagramas de caso de uso.
 - Etapa de diseño; Se realizaron los diseños de Base de Datos y de Interfaz de usuario.

- Etapa de desarrollo; Se mostró las interfaces de usuarios ya desarrolladas.

3.8. Plan de procesamiento y análisis de datos

Las etapas a seguir para el procesamiento de datos son las siguientes:

1. Aplicar las técnicas de control de tiempos y cuestionarios, mediante la observación y la entrevista.
2. Extraer los datos de la hoja de observación y de las respuestas de las entrevistas.
3. Procesar y analizar datos usando como herramienta la aplicación de oficina Microsoft Excel.
4. Emitir resultados mediante gráficos representativos.

3.9. Matriz de consistencia

Tabla VI. Matriz de Consistencia

PROBLEMA	OBJETIVOS	HIPÓTESIS	VARIABLES
<u>PROBLEMA GENERAL</u>	<u>OBJETIVO GENERAL</u>	<u>HIPÓTESIS GENERAL</u>	<u>VARIABLES DE ESTUDIO</u>
¿Qué herramienta tecnológica permitirá contribuir en el proceso de desarrollo de software?	Desarrollar una plataforma colaborativa como herramienta tecnológica para contribuir al proceso de desarrollo de software	La implementación de una plataforma colaborativa, contribuirá en el proceso de desarrollo de software	VARIABLE INDEPENDIENTE Plataforma colaborativa VARIABLE DEPENDIENTE Proceso de Desarrollo de Software
<u>PROBLEMAS ESPECÍFICOS</u>	<u>OBJETIVOS ESPECÍFICOS</u>	<u>HIPÓTESIS ESPECÍFICA</u>	<u>INDICADORES</u>
Cuatro fuentes de consulta de solución de incidencias.	Disminuir la cantidad de fuentes de búsqueda de solución de incidencias.	La implementación de una plataforma colaborativa, disminuirá el número de fuentes de búsqueda de solución de incidencias.	Cantidad de fuentes de consulta de solución.
Sesenta y seis minutos de tiempo promedio para la búsqueda de una solución a incidencia.	Disminuir el tiempo promedio de búsqueda de solución a incidencias.	La implementación de una plataforma colaborativa, disminuirá el tiempo promedio de búsqueda de solución a incidencias.	Tiempo promedio de búsqueda de solución a incidencias.
Sin evidencia de métodos y consultas registrados en un repositorio.	Aumentar la cantidad de métodos y consultas registrados en un repositorio.	La implementación de una plataforma colaborativa, aumentará la cantidad de métodos y consultas registrados en un repositorio.	Cantidad de métodos y consultas almacenadas en un repositorio de soluciones.
Sin evidencia de incidencias registradas por sistema desarrollado en la entidad.	Aumentar la cantidad de incidencias registradas por sistema desarrollado en la entidad.	La implementación de una plataforma colaborativa, aumentará la cantidad de incidencias registradas por sistema desarrollado en la entidad.	Cantidad de soluciones por incidencia registradas en el repositorio.
Sin evidencia de soluciones guardadas en una biblioteca de software definitivo.	Aumentar la cantidad de soluciones guardadas en una biblioteca de software definitivo.	La implementación de una plataforma colaborativa, aumentará la cantidad de soluciones guardadas en una biblioteca de software definitivo.	Cantidad de incidencias registradas por sistema desarrollado.

3.10. Consideraciones éticas

El proyecto de Investigación es realizado con fines académicos, por lo cual se garantiza:

- ✓ Que la metodología de recolección de datos no representa riesgo alguno para los involucrados siendo acordes al principio de no maleficencia.
- ✓ Reserva y confidencialidad respecto a toda la información a la que se tendrá acceso de ser aprobado y autorizado el perfil o proyecto de investigación.
- ✓ No informar, publicar, registrar o comunicar, total o parcialmente, por cualquier medio, el contenido de los documentos recibidos, reservando el derecho de utilizar los datos brindados con fines netamente científicos salvaguardando la integridad, privacidad y anonimato de los involucrados.
- ✓ Adoptar las medidas de bioseguridad necesarias con la diligencia debida, para evitar que toda o parte de la información sean observadas, reproducidas o manipuladas por personas no autorizadas al desarrollo del perfil o proyecto de investigación aprobado y autorizado.

Se asumirá la responsabilidad de las consecuencias legales y administrativas por el incumplimiento de estas medidas, falta ética o mala conducta en investigación antes y durante de la ejecución del mismo.

3.11. Gestión de Riesgos

Se ha realizado el análisis de los posibles eventos que se pueden presentar durante la elaboración del software para el proyecto de investigación. Los riesgos potenciales por cada fase de la metodología a desarrollar han sido identificados y medido en una matriz de riesgos mediante niveles de probabilidad, niveles de impacto y niveles de riesgo, y luego se presenta un plan de mitigación en niveles de riesgo muy alto y alto por cada amenaza, tomando como guía las plantillas adaptadas por Araque [69].

Tabla 7. Referencias durante el análisis

NOMBRE DEL PROYECTO	SIGLAS O ABREVIATURA	TIPO DE RESPUESTA
Plataforma Colaborativa para contribuir al proceso de desarrollo de Software.	HW: Hardware SW: Software	Evitar; Mitigar; Transferir; Evitar

3.11.1. Etapa de análisis

A. Matriz de riesgos

Tabla 8. Matriz de riesgos de la etapa de análisis

CÓDIGO DEL RIESGO	DESCRIPCIÓN DEL RIESGO	FASE AFECTADA	CAUSA RAÍZ	ENTREGABLES AFECTADOS	ESTIMACIÓN PROBABILIDAD	OBJETIVO AFECTADO	ESTIMACIÓN IMPACTO	PROBABILIDAD POR IMPACTO	NIVEL DE RIESGO
R-001	Requerimientos incompletos o ambiguos	Análisis	Los requerimientos no se definieron de forma clara y completa	Documento de análisis detallado por Sprint	5	Alcance	4	20	Muy Alto
						Tiempo	5	25	
						Costo	5	25	
						Calidad	4	20	
						Total probabilidad por Impacto		90	
R-002	Falta de acompañamiento de los usuarios en el levantamiento de requerimientos	Análisis	Usuarios que no colaboran o no se comprometen con la definición de los requerimientos	Documento de análisis detallado por Sprint	4	Alcance	5	20	Alto
						Tiempo	4	16	
						Costo	5	20	
						Calidad	4	16	
						Total probabilidad por Impacto		72	
R-003	Retraso en la especificación de requerimientos	Análisis	Las reuniones con el cliente para el levantamiento de requerimientos se posponen. Las especificaciones de las interfaces esenciales no están a tiempo.	Documento de análisis detallado por Sprint	3	Alcance	3	9	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	3	9	
						Total probabilidad por Impacto		42	
R-004	Incorporación continua de nuevos requerimientos	Análisis	El cliente no tiene claridad de lo que desea. Necesidad nueva por parte del mercado, del gobierno o del negocio.	Documento de análisis detallado por Sprint	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	3	9	
						Total probabilidad por Impacto		45	

R-005	Modificación continua de requerimientos.	Análisis	Actualización necesaria debido a una deficiente definición de requerimientos inicialmente.	Documento de análisis detallado por Sprint	4	Alcance	3	12	Medio
						Tiempo	3	12	
						Costo	3	12	
						Calidad	3	12	
						Total probabilidad por Impacto		48	
R-006	Modificaciones incorrectas de las especificaciones	Análisis	Actualización incorrecta de los requerimientos debido a la ausencia de un estudio detallado previo.	Documento de análisis detallado por Sprint	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	4	12	
						Total probabilidad por Impacto		48	
R-007	Entendimiento errado de los requerimientos	Análisis	El ingeniero de requerimientos entiende y documenta de manera equivocada las necesidades expuestas por el cliente.	Documento de análisis detallado por Sprint	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	4	12	
						Total probabilidad por Impacto		48	

B. Matriz de salvaguarda de riesgos

Tabla 9. Matriz de salvaguarda de riesgos en la etapa de análisis

CÓDIGO DEL RIESGO	AMENAZA / OPORTUNIDAD	DESCRIPCIÓN DEL RIESGO	FASE	NIVEL DE RIESGO	TIPO DE RESPUESTA	RESPONSABLE	PLAN DE MITIGACIÓN
R-001	Amenaza	Requerimientos incompletos o ambiguos	Análisis	Muy Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Capacitar a usuarios sobre la lógica del negocio. ➤ Aclarar con usuarios lo que se desea lograr con el proyecto. ➤ Realizar lista de preguntas sobre temas pocos claros en reuniones previas. ➤ Incorporar nuevos requerimientos o cambios necesarios de forma clara y completa para cumplir funcionalidad requerida.
R-002	Amenaza	Falta de acompañamiento de los usuarios en el levantamiento de requerimientos	Análisis	Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Reuniones periódicas con el cliente ➤ Participación del usuario en la definición de requerimientos. ➤ Compromiso y responsabilidad por parte de los usuarios.

3.11.2. Etapa de diseño

A. Matriz de riesgos

Tabla 10. Matriz de riesgos de la etapa de diseño

CÓDIGO DEL RIESGO	DESCRIPCIÓN DEL RIESGO	FASE AFECTADA	CAUSA RAÍZ	ENTREGABLES AFECTADOS	ESTIMACIÓN PROBABILIDAD	OBJETIVO AFECTADO	ESTIMACIÓN IMPACTO	PROBABILIDAD POR IMPACTO	NIVEL DE RIESGO
R-008	Incorrecta definición y estructuración de los datos establecidos.	Diseño	Definición pobre de tipos de datos e integridad, y poco entendimiento sobre la relación o dependencia de los mismos	Documento de diseño detallado por Sprint	3	Alcance	3	9	Medio
						Tiempo	3	9	
						Costo	3	9	
						Calidad	3	9	
						Total probabilidad por Impacto		36	
R-009	Diseño de interfaces incompleto	Diseño	Desconocimiento de todas las interfaces que pueden afectar la solución	Documento de diseño detallado por Sprint	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	3	9	
						Total probabilidad por Impacto		45	
R-010	Subestimación del tamaño de la aplicación.	Diseño	Al realizar el diseño se subestima el software con respecto a las necesidades del cliente.	Documento de diseño detallado por Sprint	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	5	15	
						Calidad	3	9	
						Total probabilidad por Impacto		48	
R-011	Falta de Especificación de la arquitectura lógica	Diseño	No se Define adecuadamente las interconexiones y recursos lógicos entre módulos del sistema de manera apropiada para su diseño detallado y administración.	Documento de diseño detallado por Sprint	5	Alcance	4	20	Muy Alto
						Tiempo	5	25	
						Costo	5	25	
						Calidad	4	20	
						Total probabilidad por Impacto		90	

R-012	Falta de Especificación de la arquitectura física	Diseño	No se define correctamente el conjunto de dispositivos físicos que se va utilizar para que la arquitectura lógica funcione correctamente.	Documento de diseño detallado por Sprint	3	Alcance	4	12	Alto
						Tiempo	4	12	
						Costo	5	15	
						Calidad	4	12	
						Total probabilidad por Impacto		51	
R-013	Desconocimiento de la lógica de negocio	Diseño	Mala interpretación y/o interpretación superficial de los requisitos para hacer el diseño detallado del sistema	Documento de diseño detallado por Sprint	3	Alcance	5	15	Alto
						Tiempo	4	12	
						Costo	5	15	
						Calidad	4	12	
						Total probabilidad por Impacto		54	

B. Matriz de salvaguarda de riesgos

Tabla 11. Matriz de salvaguarda de riesgos en la etapa de desarrollo

CÓDIGO DEL RIESGO	AMENAZA / OPORTUNIDAD	DESCRIPCIÓN DEL RIESGO	FASE	NIVEL DE RIESGO	TIPO DE RESPUESTA	RESPONSABLE	PLAN DE MITIGACIÓN
R-011	Amenaza	Falta de Especificación de la arquitectura lógica	Diseño	Muy Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Definir la arquitectura lógica correcta y más eficiente con base a la especificación de requerimientos ➤ Utilizar modelos, vistas y diagramas para el diseño de la arquitectura lógica
R-012	Amenaza	Falta de Especificación de la arquitectura física	Diseño	Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Tener la última versión aprobada del documento de especificación de requerimientos ➤ Definir la arquitectura física correcta y más eficiente con base a la especificación de requerimientos. ➤ Realizar el diseño de la arquitectura física teniendo en cuenta la arquitectura que posee el cliente ➤ Utilizar modelos, vistas y diagramas para el diseño de la arquitectura física
R-013	Amenaza	Desconocimiento de la lógica de negocio	Diseño	Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Facilitación de documentación sobre la lógica del negocio de la División de Tecnologías de Información ➤ Reuniones para la aclaración de dudas sobre temas puntuales

3.11.3. Etapa de desarrollo

A. Matriz de riesgos

Tabla 12. Matriz de riesgos de la etapa de desarrollo

CÓDIGO DEL RIESGO	DESCRIPCIÓN DEL RIESGO	FASE AFECTADA	CAUSA RAÍZ	ENTREGABLES AFECTADOS	ESTIMACIÓN PROBABILIDAD	OBJETIVO AFECTADO	ESTIMACIÓN IMPACTO	PROBABILIDAD POR IMPACTO	NIVEL DE RIESGO
R-014	Bajo rendimiento de la herramienta CASE	Desarrollo	Las herramientas CASE que se utilizan como apoyo no tienen el rendimiento y las funcionalidades esperadas	Implementación del Software	3	Alcance	3	9	Medio
						Tiempo	4	12	
						Costo	3	9	
						Calidad	4	12	
						Total probabilidad por Impacto		42	
R-015	Manejo inadecuado en liberación de versiones	Desarrollo	Despliegue incompleto de versión de la aplicación. El no despliegue de la última versión de la aplicación. Despliegue de versión con direccionamiento equivocada a bases de datos.	Implementación del Software	3	Alcance	3	9	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	3	9	
						Total probabilidad por Impacto		42	
R-016	Falta de documentación en código fuente	Desarrollo	Limitación del tiempo. Aplicación de malas prácticas de desarrollo y ausencia de revisiones	Implementación del Software	3	Alcance	2	6	Bajo
						Tiempo	3	9	
						Costo	2	6	
						Calidad	3	9	
						Total probabilidad por Impacto		30	
R-017	Modificación cronograma actividades	Desarrollo	Actividades no contempladas. Adición de nuevas actividades. Complejidad del desarrollo de actividades no estimadas. Retrasos en la ejecución de actividades por imprevistos indirectos.	Implementación del Software	4	Alcance	3	12	Alto
						Tiempo	4	16	
						Costo	4	16	
						Calidad	3	12	
						Total probabilidad por Impacto		56	

R-018	No disponibilidad de hardware y/o software.	Desarrollo	El hardware y/o software esencial no es entregado a tiempo.	Implementación del Software	4	Alcance	5	20	Alto
						Tiempo	5	20	
						Costo	4	16	
						Calidad	4	16	
						Total probabilidad por Impacto		72	
R-019	El Software es complejo de implementar	Desarrollo	El desarrollo de la aplicación tiene un nivel alto de complejidad. El modelado del sistema realizado en la fase de diseño no fue tan clara y específica.	Implementación del Software	4	Alcance	4	16	Alto
						Tiempo	4	16	
						Costo	4	16	
						Calidad	4	16	
						Total probabilidad por Impacto		64	
R-020	Compleja la integración de Implementación del software. módulos del software	Desarrollo	Al codificar y comenzar la integración se hace evidente que la especificación está incompleta o contiene requisitos contradictorios o hay falencias en el diseño del software.	Implementación del Software	5	Alcance	4	20	Muy Alto
						Tiempo	5	25	
						Costo	5	25	
						Calidad	4	20	
						Total probabilidad por Impacto		90	
R-021	Retiro de personal con conocimiento y experiencia	Desarrollo	Al ser las únicas personas que maneja ciertos temas específicos y/o complejos al irse generan retraso en el curso de las tareas.	Implementación del Software	4	Alcance	3	12	Medio
						Tiempo	3	12	
						Costo	3	12	
						Calidad	3	12	
						Total probabilidad por Impacto		48	
R-022	No hay buena comunicación y/o sinergia en el equipo.	Desarrollo	La comunicación entre el personal del área de desarrollo no es el más óptimo y así mismo su sinergia no es la más eficaz para el cumplimiento de objetivos en común.	Implementación del Software	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	4	12	
						Total probabilidad por Impacto		48	
R-023	Falta de conocimiento y Experiencia sobre las tareas asignadas y las herramientas a utilizar.	Desarrollo	El personal no es idóneo o no tiene la experticia necesaria para el rol asignado.	Implementación del Software	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	4	12	
						Total probabilidad por Impacto		48	

						Total probabilidad por Impacto	48		
R-024	Pérdida de Backus	Desarrollo	Pérdida de la copia de seguridad de la versión de software actual causado por virus o por remplazo de versión sin sacar la copia previamente.	Implementación del Software	3	Alcance	3	9	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	3	9	
						Total probabilidad por Impacto		42	

B. Matriz de salvaguarda de riesgos

Tabla 13. Matriz de salvaguarda de riesgos en la etapa de desarrollo

CÓDIGO DEL RIESGO	AMENAZA / OPORTUNIDAD	DESCRIPCIÓN DEL RIESGO	FASE	NIVEL DE RIESGO	TIPO DE RESPUESTA	RESPONSABLE	PLAN DE MITIGACIÓN
R-017	Amenaza	Modificación cronograma actividades	Desarrollo	Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Tamaño del módulo relativamente pequeño, para minimizar el impacto al hacer un cambio, corrupción de error o un rediseño. ➤ Independencia modular, permite de manera más fácil y flexible trabajar con ellos. ➤ Realizar pruebas de componentes.
R-018	Amenaza	No disponibilidad de hardware y/o software.	Desarrollo	Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Realizar a tiempo los documentos de diseño y contextualizar. ➤ Tener claro el alcance del desarrollo de la aplicación. ➤ Realizar una buena planeación de recursos, tareas y tiempos para evitar posibles desfases. ➤ Incluir en la planeación un tiempo racional por si ocurren imprevistos indirectos.
R-019	Amenaza	El Software es complejo de implementar	Desarrollo	Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Definir el plan de desarrollo de software los requisitos tanto de equipos físicos como herramientas software necesario para la codificación de la aplicación. ➤ Gestionar por parte del equipo de desarrollo con anterioridad los recursos HW y SW para poder ejecutarse las tareas planeadas. ➤ Tener plan alternativo por si llegan a fallar algunos de los recursos software y/ hardware.
R-020	Amenaza	Compleja la integración de Implementación del software. módulos del software	Desarrollo	Muy Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Realizar reutilización de software. ➤ Realizar pruebas unitarias.

3.11.4. Etapa de pruebas

A. Matriz de riesgos

Tabla 14. Matriz de riesgos de la etapa de pruebas

CÓDIGO DEL RIESGO	DESCRIPCIÓN DEL RIESGO	FASE AFECTADA	CAUSA RAÍZ	ENTREGABLES AFECTADOS	ESTIMACIÓN PROBABILIDAD	OBJETIVO AFECTADO	ESTIMACIÓN IMPACTO	PROBABILIDAD POR IMPACTO	NIVEL DE RIESGO
R-025	Alcance de las pruebas No definido completamente.	Pruebas	No hay suficientes recursos y/o ingresan demasiado tarde	Aseguramiento de la calidad del software	5	Alcance	4	20	Muy Alto
						Tiempo	5	25	
						Costo	5	25	
						Calidad	5	25	
						Total probabilidad por Impacto		95	
R-026	Documentación de requisitos insuficiente, desactualizada, contradictoria o ambigua.	Pruebas	Los casos de prueba no quedan cubiertos en su totalidad, debido a que pueden existir cambios y/o mejoras que no se encuentran actualizados a la fecha.	Aseguramiento de la calidad del software	5	Alcance	5	25	Muy Alto
						Tiempo	5	25	
						Costo	5	25	
						Calidad	5	25	
						Total probabilidad por Impacto		100	
R-027	Real izar pruebas en ambiente desarrollo	Pruebas	Alta Inestabilidad del ambiente Funcionalidades probadas, pero no certificadas al 100%.	Aseguramiento de la calidad del software	4	Alcance	4	16	Alto
						Tiempo	4	16	
						Costo	5	20	
						Calidad	5	20	
						Total probabilidad por Impacto		72	
R-028	No se real iza completitud en las pruebas.	Pruebas	El conjunto de pruebas real izadas no son lo suficientes para garantizar la calidad del software esto sucede por omisión y/o por falta de tiempo.	Aseguramiento de la calidad del software	4	Alcance	4	16	Alto
						Tiempo	4	16	
						Costo	4	16	
						Calidad	4	16	
						Total probabilidad por Impacto		64	

R-029	No se realiza priorización en la ejecución de las pruebas	Pruebas	No se le da prioridad para probar las funcionalidades más importantes y complejas del software. Al final se descubre defectos bloqueantes los cuales necesitan tiempo para ser solucionados.	Aseguramiento de la calidad del software	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	4	12	
						Total probabilidad por Impacto		48	
R-030	Demoras excesivas en la reparación de defectos encontrados en las pruebas	Pruebas	Solución de defectos no priorizada por parte de los desarrolladores lo cual retrasa las pruebas.	Aseguramiento de la calidad del software	3	Alcance	3	9	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	3	9	
						Total probabilidad por Impacto		42	
R-031	Problemas de disponibilidad con el ambiente de pruebas.	Pruebas	Problemas con el alistamiento, adecuación y estabilización del ambiente donde se ejecutan las pruebas, afectando cronogramas y retrasando el inicio de cada ciclo.	Aseguramiento de la calidad del software	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	4	12	
						Total probabilidad por Impacto		48	
R-032	Retraso Testing debido a nuevos errores después de despliegues	Pruebas	Retrabajo causado por despliegues los cuales dañan funcionalidades ya exitosas, generando así nuevos defectos.	Aseguramiento de la calidad del software	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	4	12	
						Total probabilidad por Impacto		48	
R-033	Pobre Productividad	Pruebas	Tiempos muertos en subfases iniciales de la fase de pruebas que no se pueden recuperar por entregas tardías de desarrollo.	Aseguramiento de la calidad del software	3	Alcance	2	6	Bajo
						Tiempo	3	9	
						Costo	2	6	
						Calidad	2	6	
						Total probabilidad por Impacto		27	
R-034	No hay suficientes recursos y/o ingresan demasiado tarde	Pruebas	Ingreso tardío del personal a los roles necesitados. Personal reducido donde se necesitan más de los asignados.	Aseguramiento de la calidad del software	3	Alcance	4	12	Medio
						Tiempo	4	12	
						Costo	4	12	
						Calidad	4	12	
						Total probabilidad por Impacto		48	

B. Matriz de salvaguarda de riesgos

Tabla 15. Matriz de salvaguarda de riesgos en la etapa de pruebas

CÓDIGO DEL RIESGO	AMENAZA / OPORTUNIDAD	DESCRIPCIÓN DEL RIESGO	FASE	NIVEL DE RIESGO	TIPO DE RESPUESTA	RESPONSABLE	PLAN DE MITIGACIÓN
R-025	Amenaza	Alcance de las pruebas No definido completamente.	Pruebas	Muy Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Contextualizar el desarrollo ➤ Aclarar dudas respecto al desarrollo
R-026	Amenaza	Documentación de requisitos insuficiente, desactualizada, contradictoria o ambigua.	Pruebas	Muy Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Se debe entregar la última versión de la documentación la cual debe estar actualizada hasta la fecha. ➤ Se debe informar y explicar los cambios realizados. ➤ Se debe entregar las nuevas versiones de los documentos
R-027	Amenaza	Realizar pruebas en ambiente desarrollo	Pruebas	Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Realizar pruebas por alguien que no sea el desarrollador
R-028	Amenaza	No se realiza completitud en las pruebas.	Pruebas	Alto	Salvaguarda	Tesista	<ul style="list-style-type: none"> ➤ Usar técnica de pruebas y buenas prácticas para cubrimiento total de pruebas. ➤ Asignar analistas de calidad con conocimientos y experiencia sobre el tema ➤ Realizar una óptima planeación de ejecución y actividades ➤ Realizar pruebas de usuario

IV. RESULTADOS

4.1. En base a la metodología utilizada

4.1.1. Product Backlog

Se elabora el Product Backlog del proyecto y su historia de usuario en base a ello.

Tabla 16. Producto Backlog

PRODUCT BACKLOG	HISTORIA
PLATAFORMA COLABORATIVA	Construir una plataforma colaborativa que contribuya al proceso de desarrollo de Software

4.1.2. Construyendo el Product Backlog

El Product Backlog lista los requerimientos funcionales, construidos y dados a conocer por el Propietario o Product Owner, se realiza una lista panorámica de todas las tareas a efectuar durante la implementación del proyecto acorde con lo que se desea cumplir y con proyección a la realización de sprints, además como agregado es relevante mencionar los requerimientos no funcionales.

Tabla 17. Construyendo el producto Backlog

PRODUCT BACKLOG
Gestionar Inicio de Sesión
Gestionar Personas
Gestionar Proyecto
Gestionar Roles
Gestionar Integrantes
Gestionar Incidencia
Gestionar Solución

Requerimientos no funcionales

- El sistema web será desarrollado en Stack MEAN.
- El plugin será desarrollado en lenguaje Java e instalado en el IDE NetBeans.

- La aplicación web debe comunicarse con el IDE de desarrollo.
- La aplicación protegerá la información de acuerdo a los permisos de acceso.
- La aplicación web podrá ser abierta desde cualquier navegador.
- Se presentará manuales de usuario para la aplicación web y el plugin.

4.1.3. Priorizando el Product Backlog

El Product Owner define las priorizaciones del Product Backlog, lo mismo que permite determinar el punto de partida del proyecto, es decir, desde que sprint se dará inicio al desarrollo del sistema. Se ha priorizado inicio de sesión ya que el usuario debe estar registrado al sistema para interactuar con él.

Tabla 18. Priorizando el Product Backlog

PRODUCT BACKLOG	PRIORIDAD
Gestionar Inicio de Sesión	1
Gestionar Personas	2
Gestionar Proyecto	3
Gestionar Roles	4
Gestionar Integrantes	5
Gestionar Incidencia	6
Gestionar Solución	7

4.1.4. Identificando la Complejidad

Se identifica la complejidad para los User Story, los cuales se desarrollarán mediante historias de usuario agrupados por sprint. A pesar de que Gestionar Solución es el de menor prioridad, es el sprint de mayor complejidad por los requerimientos solicitados y las dependencias generadas.

Tabla 19. Identificando la Complejidad

PRODUCT BACKLOG	PRIORIDAD	COMPLEJIDAD
Gestionar Inicio de Sesión	1	1
Gestionar Personas	2	2
Gestionar Proyecto	3	2
Gestionar Roles	4	2
Gestionar Integrantes	5	3
Gestionar Incidencia	6	4
Gestionar Solución	7	5

4.1.5. Asignando el valor por cada Story Points para cada User Story

Aplicando el Planing Poker se determina el esfuerzo del User Story y se asignan los valores de esfuerzo, dónde la gestión de personas, incidencias y solución demanda mayor esfuerzo en el aspecto de implementación ya que su desarrollo requiere de diversas funcionalidades complejas.

Tabla 20. Asignando el valor por cada Story Points para cada User Story

PRODUCT BACKLOG	PRIORIDAD	ESFUERZO
Gestionar Inicio de Sesión	1	5
Gestionar Personas	2	8
Gestionar Proyecto	3	5
Gestionar Roles	4	3
Gestionar Integrantes	5	5
Gestionar Incidencia	6	8
Gestionar Solución	7	8
TOTAL, STORY POINTS		42

4.1.6. Duración en día del Sprint

Coordinando con el Product Owner, se acordó realizar reuniones periódicas de informes de avances y observaciones presentadas durante el desarrollo del producto.

Tabla 21. Duración en días de Sprint

SPRINT	DÍAS
	15

Al realizarse un desarrollo acelerado, un Sprint será presentado cada 15 días a la Jefatura de División de Tecnologías de Información, lo cual es necesario para la solicitud de nuevos requerimientos y modificaciones.

4.1.7. El User Story más representativo

Se elige User Story que genera mayor valor al proyecto, es decir el más representativo.

Tabla 22. User Story más representativo

PRODUCT BACKLOG	PRIORIDAD	ESFUERZO
Gestionar Inicio de Sesión	1	5
Gestionar Personas	2	8
Gestionar Proyecto	3	5
Gestionar Roles	4	3
Gestionar Integrantes	5	5
Gestionar Incidencia	6	8
Gestionar Solución	7	8
TOTAL, STORY POINTS		42

4.1.8. User Story atendidos por un Sprint

En función del User Story más representativo se evalúa cuantos User Story pueden ser atendidos en un sprint con duración de 15 días, que suman esfuerzos y se extrae la cantidad máxima

Tabla 23. User Story atendidos por Sprint

PRODUCT BACKLOG	ESFUERZO
Gestionar Inicio de Sesión	5
Gestionar Incidencia	8
Gestionar Roles	3
TOTAL, DE ESFUERZO	16

En consecuencia, el proyecto se realizará con una velocidad de 16 User Points/15 días.

4.1.9. Número total de Sprints

Se realiza el cálculo del total de Sprints en el proyecto se realiza dividiendo el número total de User Points entre el número de User Point máximo en un sprint.

Tabla 24. Número total de Sprints

Número total de Story Points	42
Número máximo de Story Point en un sprint	16
Número total de Sprints	3

4.1.10. Tiempo total de entrega

La cantidad de días estimados para la realización del proyecto se calcula multiplicando el número total de sprint por los días de duración de un sprint.

Tabla 25. Tiempo total de entrega

Número total de Sprints	Número de días por Sprint	Total de días estimados para el proyecto
3	15	45
DURACIÓN EN MESES		1.5

4.1.11. Elaboración y agrupación de Sprints

Se identifica los User Story a desarrollar por cada sprint, tomando en cuenta la prioridad inicial.

Tabla 26. Elaboración y agrupación de los Sprints

SPRINTS	PRODUCT BACKLOG	ESFUERZO
Sprint 1	Gestionar Inicio de Sesión	5
	Gestionar Incidencia	8
	Gestionar Roles	3
Sprint 2	Gestionar Personas	8
	Gestionar Proyecto	5
Sprint 3	Gestionar Solución	8
	Gestionar Integrantes	5
TOTAL, STORY POINTS		42

4.1.12. Desarrollo de Sprints

4.1.12.1. Sprint 1

I. Etapa de planificación

A. Ajustes del Sprint anterior

En el Sprint 1 no es necesario realizar ajustes, porque aquí se inicia el desarrollo del proyecto, por esto mismo, se detalla las herramientas a utilizar durante el desarrollo a continuación:

➤ Software

- Visual Paradigm: Modelamiento de diagramas de caso de uso.
- DbSchema: Modelamiento de Base de Datos NoSQL
- Studio 3T: Interfaz gráfica para la gestión de Base de Datos NoSQL.

➤ Stack

- MEAN: Stack para desarrollo web basado en JavaScript (JS), conformado por cuatro tecnologías (MongoDB, Express.js, Angular.js y Node.js).

➤ Librería

- HTML: Hypertext Markup Language, lenguaje de marcado
- CSS: Cascading Style Sheet, lenguaje de estilo

➤ Lenguaje de Programación

- Java

➤ IDE

- Visual Studio Code: Editor de código fuente usado para el desarrollo del API.
- Sublime text: Editor de código usado para el desarrollo de la Web.
- NetBeans: Editor de código usado para el desarrollo del Plugin.

B. Pila del Sprint

Tabla 27. Pila de Sprint 1

SPRINTS	PRODUCT BACKLOG
Sprint 1	Gestionar Inicio de Sesión
	Gestionar Incidencia
	Gestionar Roles

C. Historias de usuario

Tabla 28. Historia de Usuario de Inicio de Sesión

HISTORIA DE USUARIO			
Número	1	Usuario	Cliente
Nombre de historia	Inicio sesión	Riego en desarrollo	Bajo
Prioridad en negocio	Alta	Sprint asignada	1
Puntos estimados	5	Dependencia para su desarrollo	Persona
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero acceder al sistema mediante un inicio de sesión para que mi acceso sea seguro.		
Criterio de aceptación	<ul style="list-style-type: none"> - Se accederá ingresando correo y contraseña. - Los atributos “correo” y “contraseña” serán llamados del esquema persona. - Se realizará una interfaz gráfica en la web y el plugin para el logueado del usuario. - en el plugin una vez que accedió el usuario se habilitará el ícono del plugin en el IDE. 		

Tabla 29. Historia de Usuario de Incidencia

HISTORIA DE USUARIO			
Número	2	Usuario	Cliente
Nombre de historia	Incidencia	Riego en desarrollo	Alto
Prioridad en negocio	Alta	Sprint asignada	2
Puntos estimados	8	Dependencia para su desarrollo	Persona, Tipo de registro
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero registrar mis incidencias para conocer cuáles con las que cuento al momento de realizar el proceso de desarrollo de software.		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como "incidencias". - Sus atributos serán: titulo, descripcion, fecha_creacion, hora_creacion, persona_registra, comentario, tipo_incidencia, persona_asignada, estado, asignación, prioridad. - Se creará una interfaz gráfica en la web y el plugin para su gestión. - La incidencia podrá ser agregada, editada, listada y eliminada mediante la interfaz web y plugin. 		

Tabla 30. Historia de Usuario de Roles

HISTORIA DE USUARIO			
Plata	3	Usuario	Cliente
Nombre de historia	Roles	Riego en desarrollo	Bajo
Prioridad en negocio	Alta	Sprint asignada	1
Puntos estimados	3	Dependencia para su desarrollo	Ninguna
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero clasificar a los integrantes de mis proyectos por su rol en el mismo.		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como “roles”. - Sus atributos serán: nombre_rol - Se creará una interfaz gráfica web para su gestión. 		

D. Tareas de usuario

Tabla 31. Tarea de Usuario *Crear Inicio de Sesión*

Nº de Tarea	1.1	Nº Historia de Usuario	1
Nombre de la Tarea	Crear inicio de sesión	Tiempo Estimado en Horas	4
Fecha Inicio	27/04/2018	Fecha Fin	27/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	<ul style="list-style-type: none"> a. Se accederá ingresando correo y contraseña. b. Los atributos “correo” y “contraseña” serán llamados del esquema persona. 		

Tabla 32. Tarea de Usuario *Crear autenticación de persona*

Nº de Tarea	1.2	Nº Historia de Usuario	1
Nombre de la Tarea	Crear autenticación de persona	Tiempo Estimado en Horas	4
Fecha Inicio	27/04/2018	Fecha Fin	27/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	a. La autenticación será basada en token. b. Añadir autenticación token en registrar personas.		

Tabla 33. Tarea de Usuario *Realizar interfaz gráfica de inicio de sesión web*

Nº de Tarea	1.3	Nº Historia de Usuario	1
Nombre de la Tarea	Realizar interfaz gráfica de inicio de sesión web	Tiempo Estimado en Horas	1
Fecha Inicio	2/05/2018	Fecha Fin	2/05/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Consiste en diseñar e implementar el formulario necesario para la gestión adecuada del inicio de sesión, utilizando Angular.		

Tabla 34. Tarea de Usuario *Realizar interfaz gráfica de inicio de sesión plugin*

Nº de Tarea	1.4	Nº Historia de Usuario	1
Nombre de la Tarea	Realizar interfaz gráfica de inicio de sesión plugin	Tiempo Estimado en Horas	2
Fecha Inicio	16/05/2018	Fecha Fin	16/05/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Consiste en diseñar e implementar el formulario necesario para la gestión adecuada del inicio de sesión, utilizando Java.		

Tabla 35. Tarea de Usuario *Crear esquema incidencias*

Nº de Tarea	2.1	Nº Historia de Usuario	2
Nombre de la Tarea	Crear esquema incidencias	Tiempo Estimado en Horas	1
Fecha Inicio	26/04/2018	Fecha Fin	26/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	<p>- Atributos:</p> <ul style="list-style-type: none"> i. Título ii. Descripción iii. Fecha_creacion iv. Hora_creación v. Comentario vi. Tipo_incidencia vii. Estado viii. Asignación ix. Prioridad x. Persona_registra xi. Persona_asignada xii. Id_proyecto 		

Tabla 36. Tarea de Usuario *Crear, registrar, editar, listar y eliminar incidencias*

N° de Tarea	2.2	N° Historia de Usuario	2
Nombre de la Tarea	Crear registrar, editar, listar y eliminar incidencias.	Tiempo Estimado en Horas	4
Fecha Inicio	29/04/2018	Fecha Fin	29/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	<p>a. Solo se eliminará definitivamente las incidencias que no tenga descripción.</p> <p>b. La búsqueda se realizará mediante el nombre de la incidencia.</p> <p>c. El usuario puede registrar una incidencia con o sin solución.</p> <p>d. La persona asignada, es aquella persona asignada a dar solución a la incidencia.</p>		

Tabla 37. Tarea de Usuario *Realizar interfaz gráfica de incidencia plugin*

N° de Tarea	2.3	N° Historia de Usuario	2
Nombre de la Tarea	Realizar interfaz gráfica de incidencia plugin	Tiempo Estimado en Horas	2
Fecha Inicio	2/05/2018	Fecha Fin	2/05/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Consiste en diseñar e implementar el formulario necesario para la gestión adecuada de la incidencia, utilizando Java.		

Tabla 38. Tarea de Usuario *Realizar interfaz gráfica de incidencias web*

Nº de Tarea	2.4	Nº Historia de Usuario	2
Nombre de la Tarea	Realizar interfaz gráfica de incidencia web	Tiempo Estimado en Horas	1
Fecha Inicio	18/05/2018	Fecha Fin	18/05/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Consiste en diseñar e implementar el formulario necesario para la gestión adecuada de incidencia, utilizando Angular.		

Tabla 39. Tarea de Usuario *Crear esquema roles*

Nº de Tarea	3.1	Nº Historia de Usuario	3
Nombre de la Tarea	Crear esquema roles	Tiempo Estimado en Horas	1
Fecha Inicio	26/04/2018	Fecha Fin	26/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	- Atributos: i. Nombre		

Tabla 40. Tarea de Usuario *Crear, registrar, editar, listar y eliminar roles*

Nº de Tarea	3.2	Nº Historia de Usuario	3
Nombre de la Tarea	6.2. Crear registrar, editar, listar y eliminar roles	Tiempo Estimado en Horas	4
Fecha Inicio	30/04/2018	Fecha Fin	30/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	a. Solo el jefe de proyectos podrá gestionar los roles.		

Tabla 41. Tarea de Usuario *Realizar interfaz gráfica de roles web*


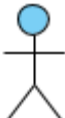
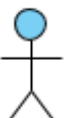
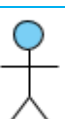


Nº de Tarea	3.3	Nº Historia de Usuario	3
Nombre de la Tarea	Realizar interfaz gráfica de roles web	Tiempo Estimado en Horas	1
Fecha Inicio	19/05/2018	Fecha Fin	19/05/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Consiste en diseñar e implementar el formulario necesario para la gestión adecuada de roles, utilizando Angular.		



II. Etapa de análisis

A. Actores

En esta etapa se identifica y describe a los actores involucrados con el Sprint, pero en este caso se describirá a todos los actores involucrados con el proyecto, los cuales son:

Tabla 42. Actores involucrados con el Sprint

ACTOR	DESCRIPCIÓN
 Jefe de Proyecto	Persona responsable de la gestión y planificación del proyecto, controlar los avances, asignar el trabajo, realizar seguimientos y realizar las coordinaciones con su equipo y el usuario final.
 Analista	Persona responsable del análisis del sistema, especifica requisitos, realiza análisis general y detallado, propone mejoras y soluciones respecto a los requerimientos del cliente.
 Arquitecto	Persona responsable de estructurar el sistema, convierte los requisitos definidos por el analista en una solución técnica y realiza el seguimiento del desarrollo para verificar la estructura generada.
 Diseñador	Persona responsable del diseño del software, se cerciora que el sistema cumpla con las características de un sistema amigable con el usuario final.
 Desarrollador	Persona responsable de implementar componentes y funcionalidades mediante la codificación, ubica y corrige defectos, documenta el código e integra subsistemas implementados y su contenido durante el desarrollo.
 Tester	Persona responsable de la gestión y el desarrollo de las pruebas a realizar en el sistema, debe asegurar el funcionamiento correcto del sistema construido.

 Especialista de Calidad	Persona responsable de instaurar un plan de calidad, debe observar el proceso de desarrollo, examinar los resultados y garantizar que el sistema esté libre de defectos.
 Administrador de Base de Datos	Persona responsable de administrar la base de datos, crea, gestiona, implementa y brinda soporte a la base de datos de la organización.

B. Diagrama de caso de uso

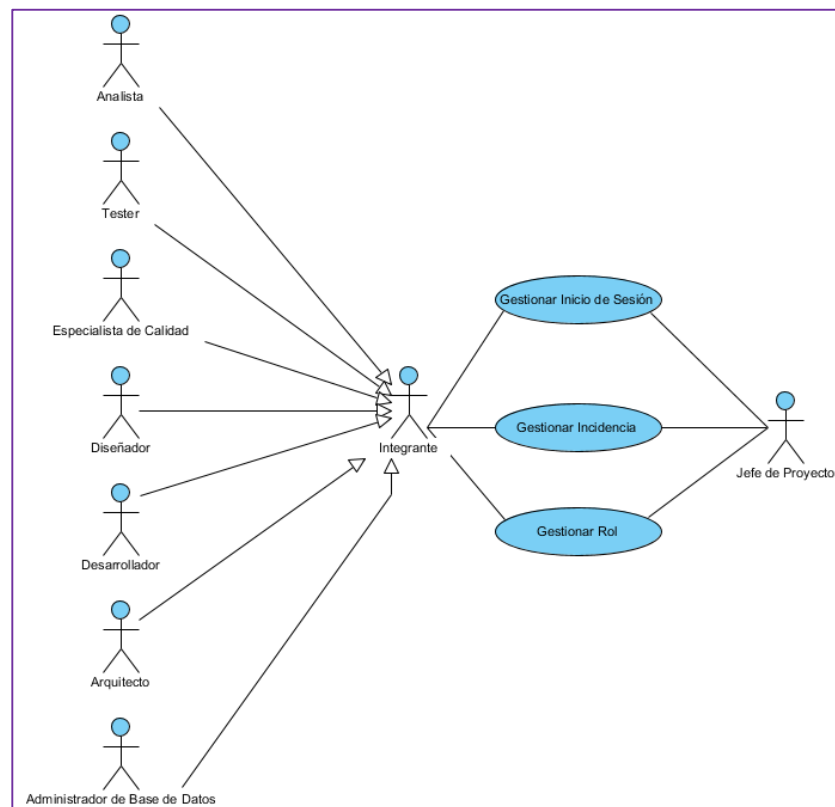


Fig. 21. Diagrama de Caso de Uso de Sprint 1

III. Etapa de diseño

A. Diseño de Base de Datos

a. Diseño Lógico

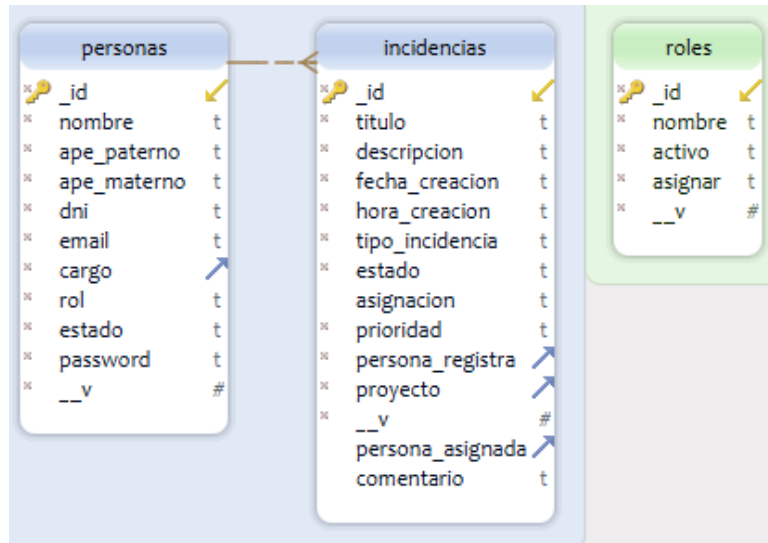


Fig. 22. Diseño lógico de Base de Datos de Sprint 1

b. Diseño Físico

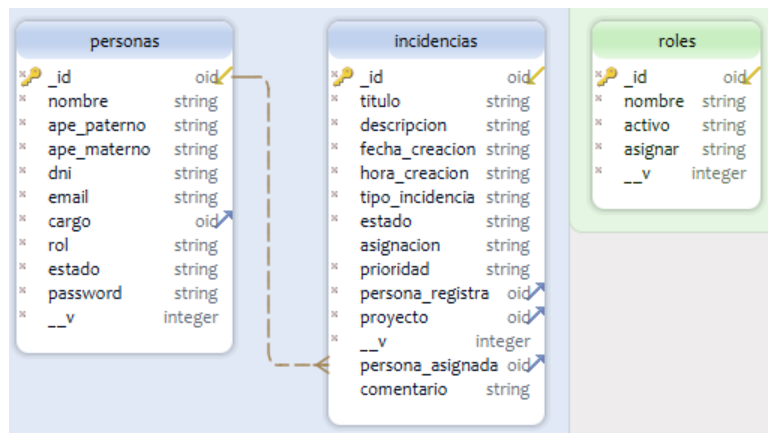


Fig. 23. Diseño Físico de la Base de Datos de Sprint 1

B. Diseño de Interfaz de Usuario

a. Inicio de Sesión

➤ Interfaz Web

Boceto de interfaz web para el inicio de sesión del usuario.

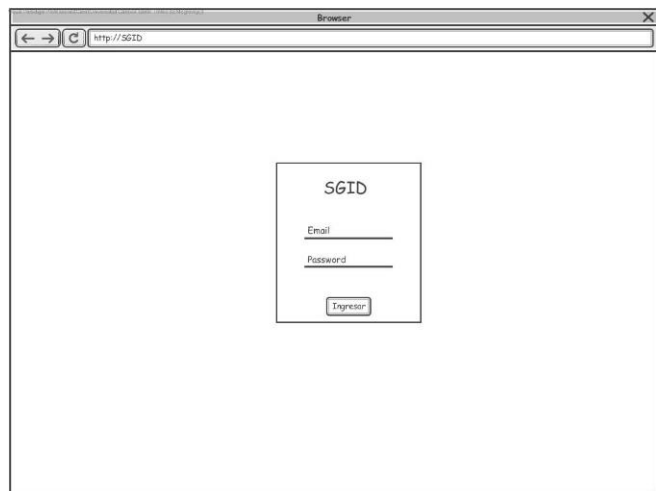


Fig. 24. Boceto de interfaz web de Inicio de Sesión

➤ Interfaz Plugin

Boceto de interfaz de Inicio de Sesión desde el plugin en NetBeans.

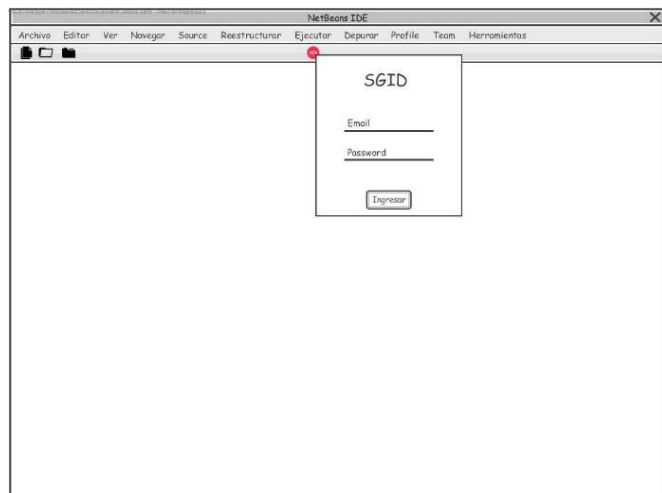


Fig. 25. Boceto de interfaz plugin de Inicio de Sesión

b. Gestionar Incidencia

➤ Interfaz Web

Boceto de interfaz web para la gestión de incidencias.

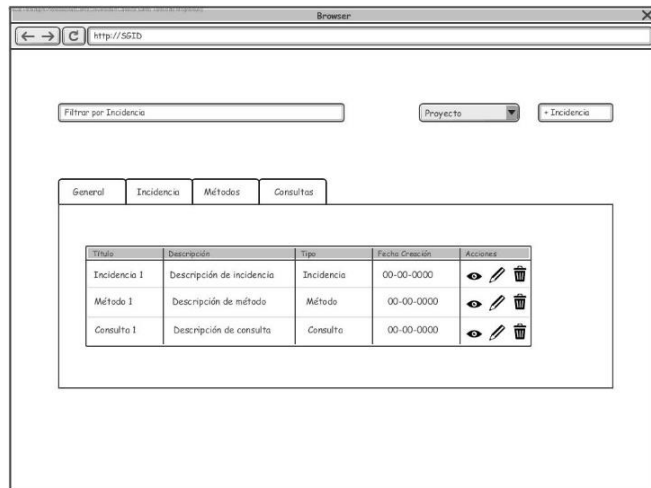


Fig. 26. Boceto de interfaz web de Gestionar Incidencia

➤ Interfaz Plugin

Boceto de interfaz para la gestión de incidencias con el plugin en NetBeans.

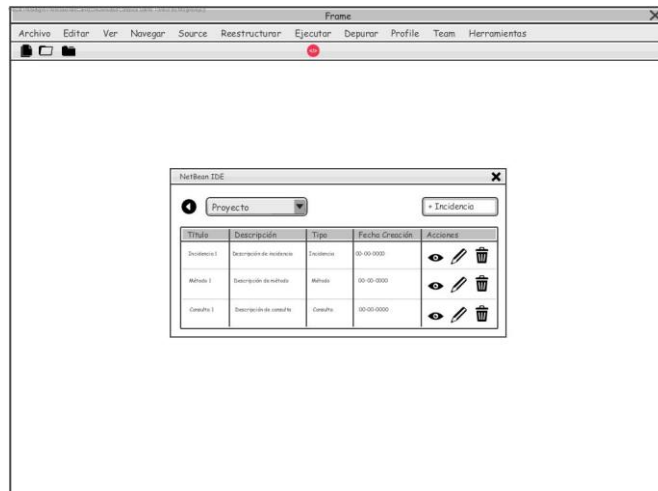


Fig. 27. Boceto de interfaz plugin de Gestión de Incidencias

c. Gestionar Rol

➤ Interfaz Web

Boceto de interfaz web para la gestión de roles en el proceso de desarrollo.

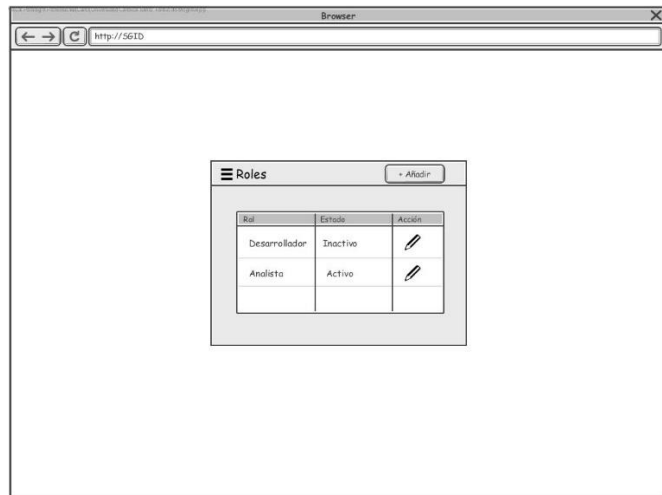


Fig. 28. Boceto de interfaz web de Gestionar Rol

IV. Etapa de desarrollo

C. Desarrollo de las interfaces de usuario

a. Inicio de Sesión

➤ Interfaz Web

Con esta interfaz se puede realizar el inicio de sesión del usuario en la aplicación web, mediante el ingreso de su correo y contraseña.



Fig. 29. Interfaz web de inicio de sesión

➤ Interfaz Plugin

Con esta interfaz se puede realizar el inicio de sesión del usuario con el plugin en NetBeans, mediante el ingreso de su correo y contraseña.

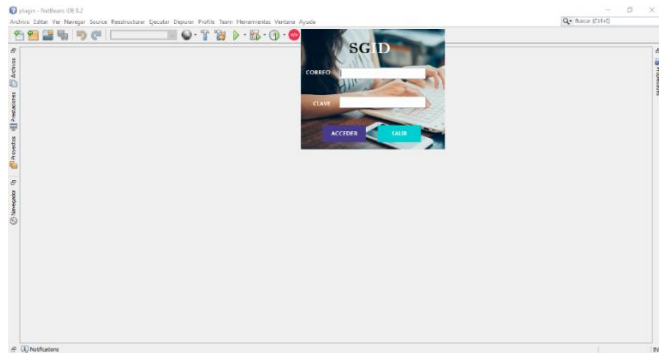


Fig. 30. Interfaz plugin de inicio de sesión

b. Gestión de Incidencias

➤ Interfaz Web

Con esta interfaz se puede gestionar las incidencias en la aplicación web, realizar acciones como agregar, listar, editar, ver detalles y eliminar.

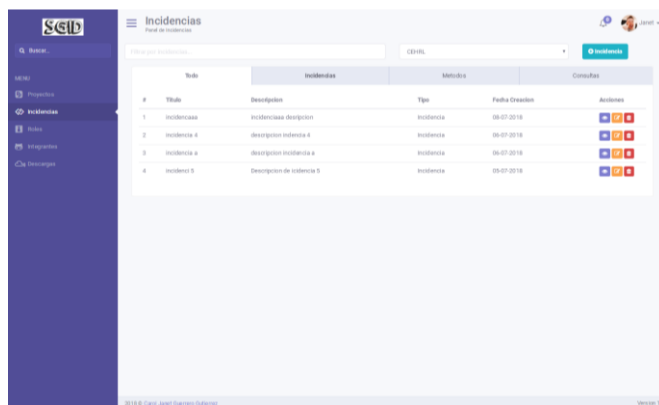


Fig. 31. Interfaz web de Gestión de Incidencias

➤ **Interfaz Plugin**

Con esta interfaz se puede gestionar las incidencias desde NetBeans, realizar acciones como agregar, listar, editar, ver detalles y eliminar



Fig. 32. Interfaz plugin de Gestión de Incidencias

c. **Gestión de Roles**

➤ **Interfaz Web**

Con esta interfaz se puede gestionar los roles en la aplicación web, realizar acciones como agregar, listar, editar y cambiar de estado.

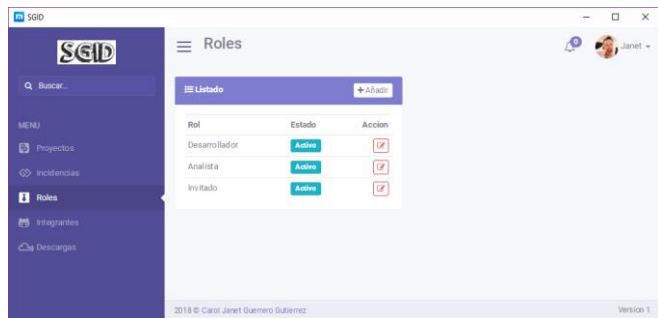


Fig. 33. Interfaz web de Gestión de Roles

4.1.12.2. Sprint 2

I. Etapa de planificación

A. Ajustes del Sprint anterior

Se procedió a realizar la funcionalidad de concurrencia para el registro de inicio de sesión, incidencia y roles. Además, se ha realizado el agregado de datos a la tabla roles y cargo.

B. Pila del Sprint

Tabla 43. Pila del Sprint 2

SPRINTS	PRODUCT BACKLOG
Sprint 2	Gestionar Personas
	Gestionar Proyecto

C. Historias de usuario

Tabla 44. Historia de usuario de Persona

HISTORIA DE USUARIO			
Número	4	Usuario	Cliente
Nombre de historia	Persona	Riego en desarrollo	Alto
Prioridad en negocio	Alta	Sprint asignada	1
Puntos estimados	8	Dependencia para su desarrollo	Ninguna
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero registrarme como usuario del software para tener acceso a sus funciones.		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como "personas" - Sus atributos serán: nombre_persona, correo, apellido_paterno, apellido_materno, contraseña, dni, estado, cargo - Todos los datos son obligatorios - Se realizará una interfaz gráfica web para el registro de la persona. - Se creará una interfaz web para la modificación de sus datos. - La persona mediante su cuenta podrá darse de baja - Dar de baja a una persona será un cambio de estado "Activo" a "Inactivo". - La búsqueda se realizará mediante el nombre o correo de la persona - Solo el jefe de proyecto podrá buscar y listar usuarios activos e inactivos que han participado en sus proyectos. 		

Tabla 45. Historia de usuario de Proyecto

HISTORIA DE USUARIO			
Número	5	Usuario	Cliente
Nombre de historia	Proyecto	Riego en desarrollo	Bajo
Prioridad en negocio	Alta	Sprint asignada	3
Puntos estimados	5	Dependencia para su desarrollo	Persona
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero registrar mis proyectos para llevar un control de sus incidencias.		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como “proyectos” - Sus atributos serán: nombre, fecha_creación, hora_creacion, descripcion, persona_registra. - El atributo “persona_registra” será llamado del esquema “persona” - Se creará una interfaz gráfica web para su gestión. - El usuario que registra el proyecto se vuelve automáticamente el jefe del proyecto. - El jefe de proyecto podrá buscar personas e integrarlas al proyecto. - El jefe de proyecto deberá asignarle un rol al integrante agregado al proyecto. - La lista de integrantes se deberá visualizar al momento de acceder al proyecto. - La lista de incidencias se deberá visualizar al momento de acceder al proyecto. 		

D. Tareas de usuario

Tabla 46. Tarea de usuario *Crear esquema personas*

N° de Tarea	4.1	N° Historia de Usuario	4
Nombre de la Tarea	Crear esquema personas	Tiempo Estimado en Horas	1
Fecha Inicio	26/04/2018	Fecha Fin	26/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	<p>- Atributos:</p> <ul style="list-style-type: none"> i. nombre ii. ape_paterno iii. ape_materno iv. dni v. correo vi. contraseña vii. cargo viii. estado 		

Tabla 47. Tarea de usuario *Crear, editar, listar y eliminar personas*

N° de Tarea	4.2	N° Historia de Usuario	4
Nombre de la Tarea	Crear editar, listar y eliminar personas con autenticación.	Tiempo Estimado en Horas	4
Fecha Inicio	28/04/2018	Fecha Fin	28/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	<ul style="list-style-type: none"> a. Eliminar es cuando la persona mediante su cuenta podrá darse de baja b. Dar de baja a una persona será un cambio de estado "Activo" a "Inactivo". c. La búsqueda se realizará mediante el nombre o correo de la persona 		

Tabla 48. Tarea de usuario *Realizar interfaz gráfica web de personas*

Nº de Tarea	4.3	Nº Historia de Usuario	4
Nombre de la Tarea	Realizar interfaz gráfica de persona web	Tiempo Estimado en Horas	1
Fecha Inicio	15/05/2018	Fecha Fin	15/05/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Consiste en diseñar e implementar el formulario necesario para la gestión adecuada de la persona, utilizando Angular.		

Tabla 49. Tarea de usuario *Crear esquema proyectos*

Nº de Tarea	5.1	Nº Historia de Usuario	5
Nombre de la Tarea	Crear esquema proyectos	Tiempo Estimado en Horas	1
Fecha Inicio	26/04/2018	Fecha Fin	26/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	- Atributos: i. Nombre ii. Fecha_creacion iii. Hora_creacion iv. Descripción v. Persona_registra		

Tabla 50. Tarea de usuario *Crear, registrar y listar proyectos*

Nº de Tarea	5.2	Nº Historia de Usuario	5
Nombre de la Tarea	Crear registrar, editar y listar proyectos	Tiempo Estimado en Horas	4
Fecha Inicio	28/04/2018	Fecha Fin	28/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	<p>a. El usuario que registra el proyecto se vuelve automáticamente el jefe del proyecto.</p> <p>b. El jefe de proyecto podrá buscar personas e integrarlas al proyecto.</p> <p>c. La lista de incidencias se deberá visualizar al momento de acceder al proyecto.</p> <p>d. El atributo “persona_registra” será llamado del esquema “personas”.</p>		

Tabla 51. Tarea de usuario *Realizar interfaz gráfica web de proyecto*

Nº de Tarea	5.3	Nº Historia de Usuario	5
Nombre de la Tarea	Realizar interfaz gráfica de proyecto web	Tiempo Estimado en Horas	1
Fecha Inicio	17/05/2018	Fecha Fin	17/05/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Consiste en diseñar e implementar el formulario necesario para la gestión adecuada del proyecto, utilizando Angular.		

II. Etapa de análisis

A. Diagrama de caso de uso

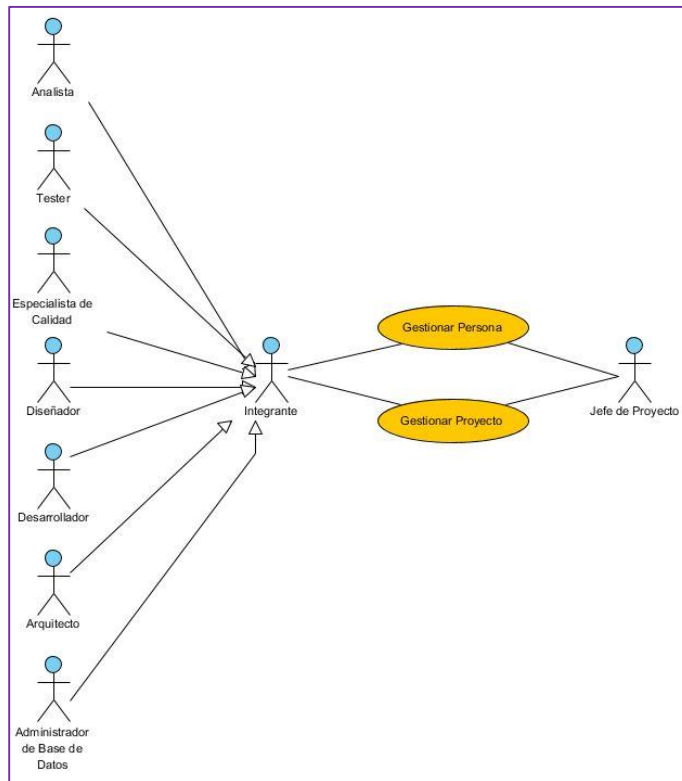


Fig. 34. Diagrama de caso de uso de Sprint 2

III. Etapa de diseño

B. Diseño de Base de Datos

a. Diseño Lógico

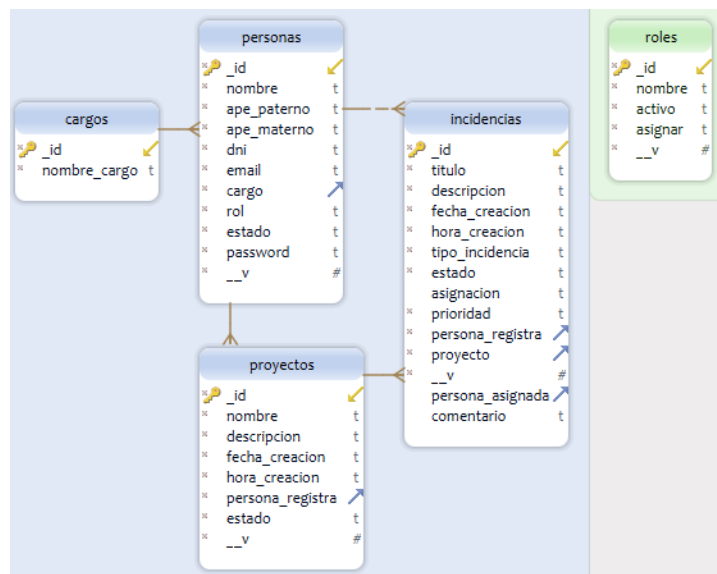


Fig. 35. diseño lógico de base de datos de Sprint 2

b. Diseño Físico

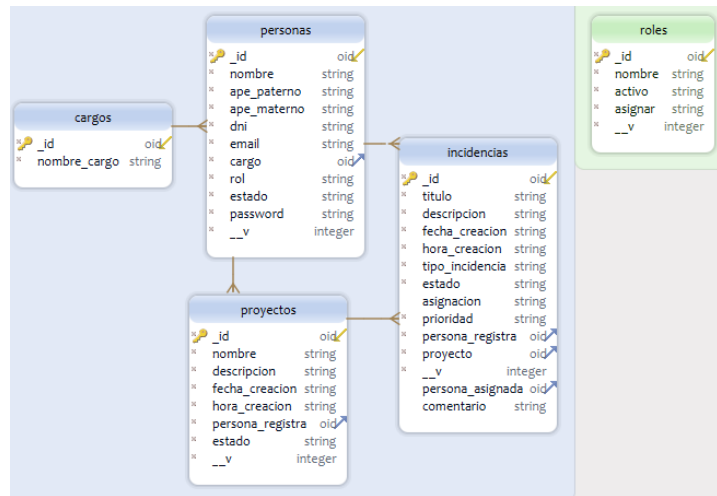


Fig. 36. Diseño físico de base de datos de Sprint 2

C. Diseño de interfaz de usuario

a. Gestionar Persona

➤ Interfaz Web

Boceto de interfaz web para la gestión de personas.

Boceto de interfaz web de Gestión de Personas. Muestra un formulario con los siguientes campos:

- Nombre
- Ap. Paterno
- Ap. Materno
- DNI
- Email
- Password
- Cargo (lista desplegable)
- Registrar

Fig. 37. Boceto interfaz web de Gestión de Personas

b. Gestionar proyecto

➤ Interfaz Web

Boceto de interfaz web para la gestión de proyectos.

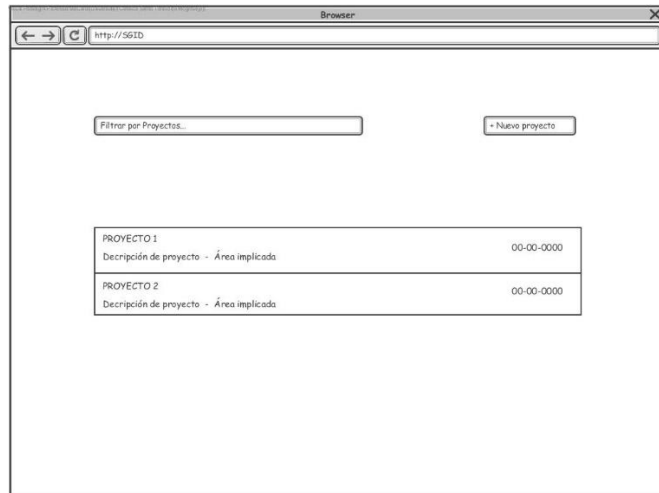


Fig. 38. Boceto interfaz web de Gestión de Proyectos

IV. Etapa de desarrollo

A. Diseño de interfaz de usuario

a. Gestionar Persona

➤ Interfaz Web

Con esta interfaz se puede realizar el registro de un nuevo usuario en la aplicación web, mediante el ingreso de su nombre, apellidos, DNI, correo, contraseña y cargo.

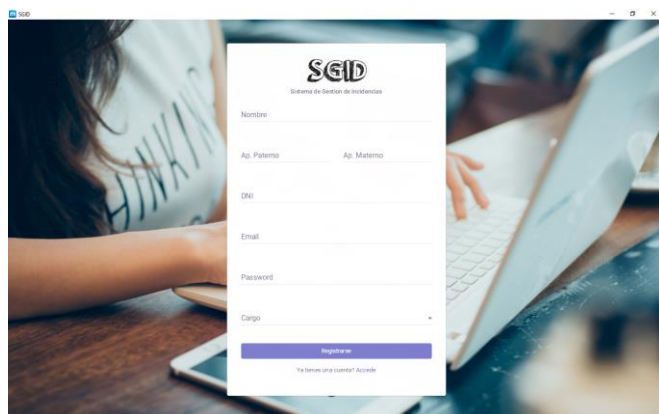


Fig. 39. Interfaz web de Gestionar Persona

b. Gestionar Proyecto

➤ Interfaz Web

Con esta interfaz se puede gestionar los proyectos en la aplicación web, realizar acciones como agregar, listar, editar, ver detalles y eliminar.

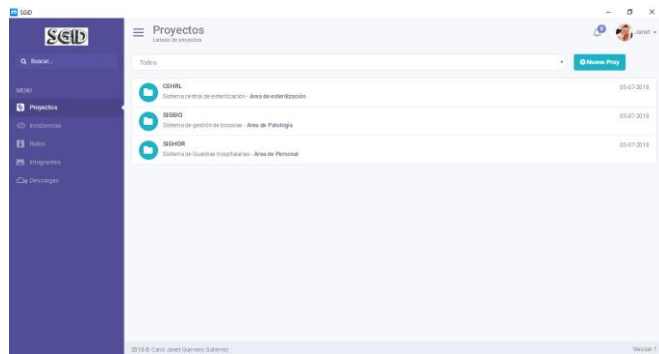


Fig. 40. Interfaz web de Gestionar Proyecto

4.1.12.3. Sprint 3

I. Etapa de planificación

A. Ajustes del Sprint anterior

En el Sprint anterior se creó la tabla cargos, porque se considera mostrar módulos basado en los mismos. La aplicación tendrá dos cargos; jefe de DTI y Operario. Se procedió a realizar la funcionalidad de concurrencia para el registro de persona y proyecto.

B. Pila del Sprint

Tabla 52. Pila del Sprint 3

SPRINTS	PRODUCT BACKLOG
Sprint 3	Gestionar Solución
	Gestionar Integrantes

C. Historias de usuario

Tabla 53. Historia de usuario de Solución

HISTORIA DE USUARIO			
Número	6	Usuario	Cliente
Nombre de historia	Solución	Riego en desarrollo	Alto
Prioridad en negocio	Alta	Sprint asignada	2
Puntos estimados	8	Dependencia para su desarrollo	Persona, Incidencia
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero registrar mis soluciones para ...		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como "solución". - Sus atributos serán: tipo_descripcion, fecha_creacion, hora_creacion, nombre_archivo, descripción, tipo_solucion, comentario, persona_registra, estado. - Se gestionará desde la interfaz gráfica web y plugin de incidencia. - Los tipos de descripción son dos, "texto" y "código". - Los tipos de solución son dos, "temporal" y "final". - Los estados de solución son dos, "resuelto" y "no resuelto". - Cuando el tipo de descripción sea código, debe abrirse la opción de "Nombre de archivo" para su registro. - El usuario puede registrar una incidencia con o sin solución. - Los atributos "fecha_creacion", "hora_creacion" y "persona_registra" se tomarán del sistema. - Solo cuando el tipo de solución es "final" el estado de solución será "resuelto". - El atributo "persona_registra" será tomado del esquema persona. - El atributo "id_proyecto" será tomado del esquema proyecto. 		

Tabla 54. Historia de usuario de Integrante

HISTORIA DE USUARIO			
Número	7	Usuario	Cliente
Nombre de historia	Integrante	Riego en desarrollo	Bajo
Prioridad en negocio	Media	Sprint asignada	3
Puntos estimados	5	Dependencia para su desarrollo	Persona, Proyecto, Roles
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero agregar personas a mis proyectos como integrantes para que puedan registrar incidencias que se presenten en el proceso de desarrollo.		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como "integrante" - Sus atributos serán: id_persona, id_proyecto, id_rol, permiso, fecha_registro, estado. - En la búsqueda y listado del usuario, para registrarlo como integrante y agregarlo a un proyecto, se mostrarán aquellas personas que se encuentren activas y disponibles. - Al momento de agregar un integrante al proyecto, se podrá seleccionar su rol, pero por defecto este será desarrollador. - El rol será asignado por el jefe de proyecto. - Se podrá cambiar el rol de un integrante en un mismo proyecto. - La lista de integrantes será por proyecto al que pertenecen. - La fecha y hora de registro se tomará del sistema. - El jefe de proyecto puede asignar a otra persona su rol como jefe de proyecto. - Al asignar a otra persona como jefe de proyecto su rol cambiara por defecto a "Desarrollador". - El "id_persona" será llamado del esquema "persona" - El "id_proyecto" será llamado del esquema "proyecto" - El "id_rol" será llamado del esquema "rol" - Los tipos de "permiso" serán dos, lectura y escritura - Al eliminar un integrante de un proyecto cambiará de estado de "activo" a "inactivo". 		

D. Tareas de usuario

Tabla 55. Tarea de usuario *Crear esquema solución*

Nº de Tarea	6.1	Nº Historia de Usuario	6
Nombre de la Tarea	Crear esquema solución	Tiempo Estimado en Horas	1
Fecha Inicio	26/04/2018	Fecha Fin	26/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	<p>- Atributos:</p> <ul style="list-style-type: none"> i. Tipo_descripcion ii. Fecha_creacion iii. Hora_creacion iv. Nombre_archivo v. Descripción vi. Tipo_solucion vii. Comentario viii. Estado ix. Persona_registra x. Id_incidencia 		

Tabla 56. Tarea de usuario *Crear, registrar y editar solución*

Nº de Tarea	6.2	Nº Historia de Usuario	6
Nombre de la Tarea	Crear registrar y editar solución	Tiempo Estimado en Horas	4
Fecha Inicio	29/04/2018	Fecha Fin	29/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	<p>a. Los tipos de descripción son dos, "texto" y "código".</p> <p>b. Los tipos de solución son dos, "temporal" y "final".</p> <p>c. Los estados de solución son dos, "resuelto" y "no resuelto".</p> <p>d. Cuando el tipo de descripción sea código, debe abrirse la opción de "Nombre de archivo" para su registro.</p> <p>e. El usuario puede registrar una incidencia con o sin solución.</p> <p>f. Los atributos "fecha_creacion", "hora_creacion" y "persona_registra" se tomarán del sistema.</p> <p>g. Solo cuando el tipo de solución es "final" el estado de solución será "resuelto".</p> <p>h. El atributo "persona_registra" será tomado del esquema persona.</p> <p>i. El atributo "id_incidencia" será tomado del esquema "incidencias".</p>		

Tabla 57. Tarea de usuario *Crear esquema Integrantes*

N° de Tarea	7.1	N° Historia de Usuario	7
Nombre de la Tarea	Crear esquema de integrantes	Tiempo Estimado en Horas	1
Fecha Inicio	26/04/2018	Fecha Fin	26/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	- Atributos: i. permiso ii. fecha_registro iii. hora de registro iv. estado v. id_persona vi. id_proyecto vii. id_rol		

Tabla 58. Tarea de usuario *Crear, registrar, editar, listar y eliminar integrante*

Nº de Tarea	7.2	Nº Historia de Usuario	7
Nombre de la Tarea	Crear registrar, editar, listar y eliminar integrante	Tiempo Estimado en Horas	4
Fecha Inicio	30/04/2018	Fecha Fin	30/04/2018
Responsable	Carol Janet Guerrero Gutiérrez		
Descripción	<p>a. Los tipos de "permiso" serán dos, "lectura" y "escritura".</p> <p>b. Al eliminar un integrante de un proyecto cambiará de estado de "activo" a "inactivo".</p> <p>c. La lista de integrantes será por proyecto al que pertenecen.</p> <p>d. La fecha y hora de registro se tomará del sistema.</p> <p>e. El jefe de proyecto puede asignar a otra persona su rol como jefe de proyecto.</p> <p>f. Al asignar a otra persona como jefe de proyecto su rol cambiara por defecto a "Desarrollador".</p> <p>g. El "id_persona" será llamado del esquema "persona".</p> <p>h. El "id_proyecto" será llamado del esquema "proyecto".</p> <p>i. El "id_rol" será llamado del esquema "rol".</p>		

II. Etapa de análisis

A. Diagrama de caso de uso

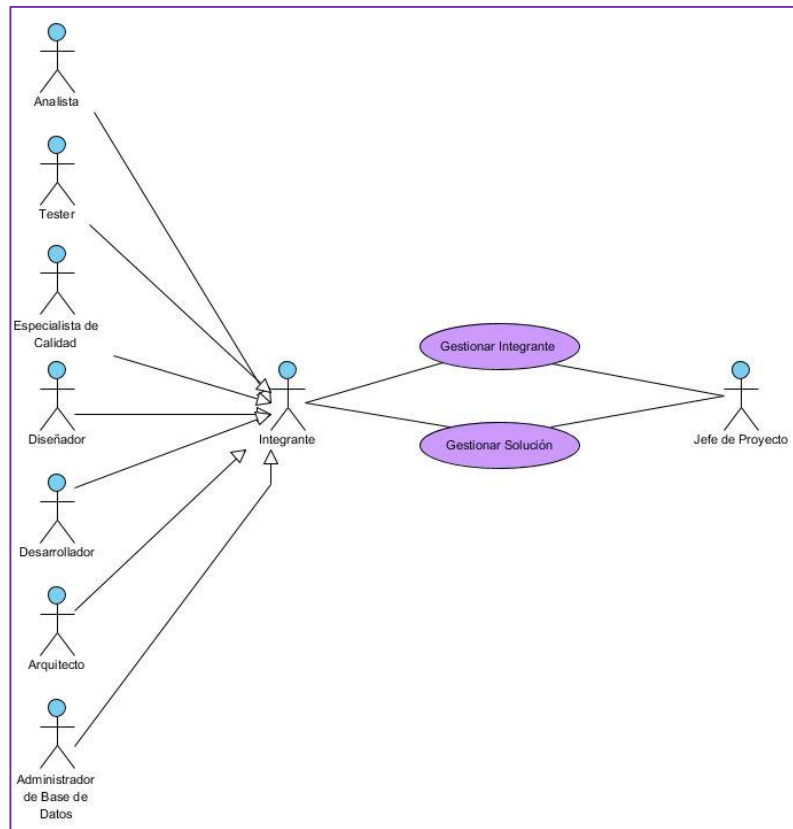


Fig. 41. Diagrama de caso de uso de Sprint 3

III. Etapa de diseño

A. Diseño de Base de Datos

a. Diseño Lógico

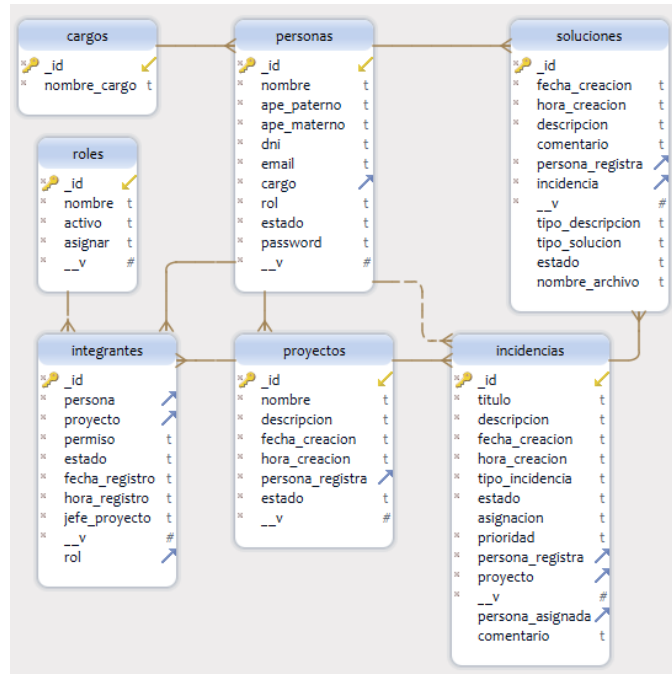


Fig. 42. Diseño lógico de base de datos de Sprint 3

b. Diseño Físico

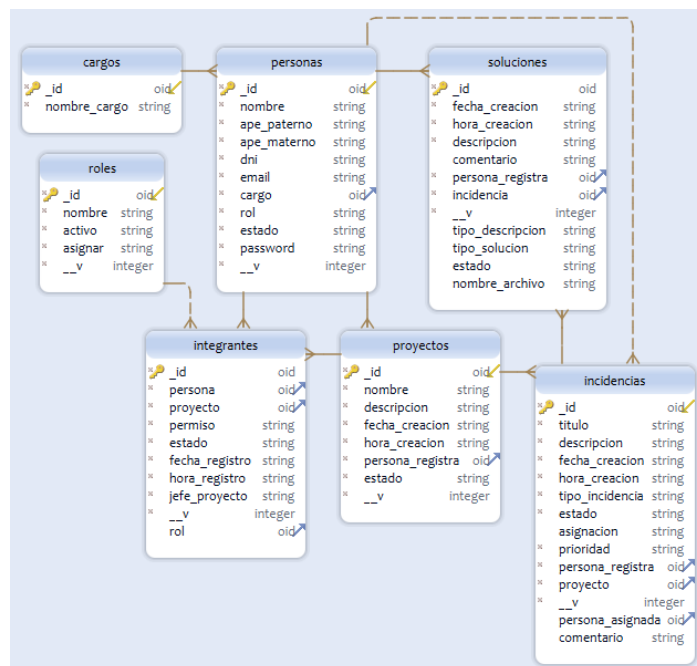


Fig. 43. Diseño físico de base de datos de Sprint 3

B. Diseño de interfaz de usuario

a. Gestionar Integrante

➤ Interfaz Web

Boceto de interfaz web para la gestión de integrantes.

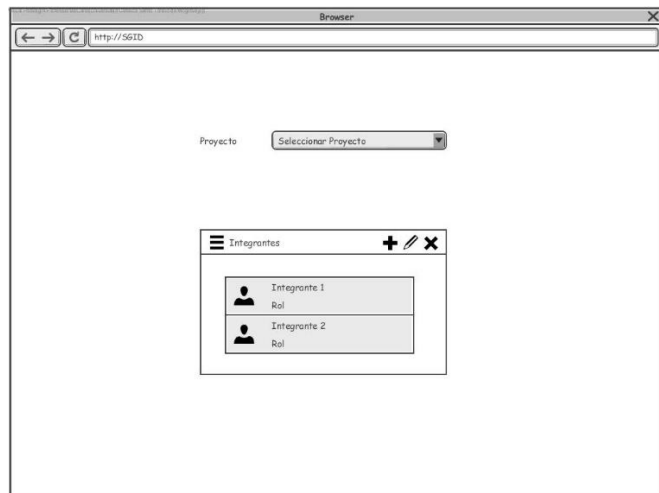


Fig. 44. Boceto interfaz web de Gestión de Integrantes

b. Gestionar Solución

➤ Interfaz Web

Boceto de interfaz web para la gestión de soluciones.

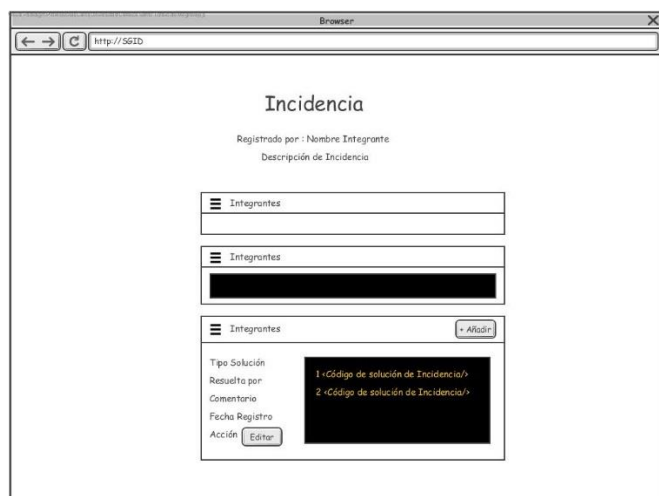


Fig. 45. Boceto interfaz web de Gestión de Solución

➤ Interfaz Plugin

Boceto de interfaz plugin para la gestión de solución.

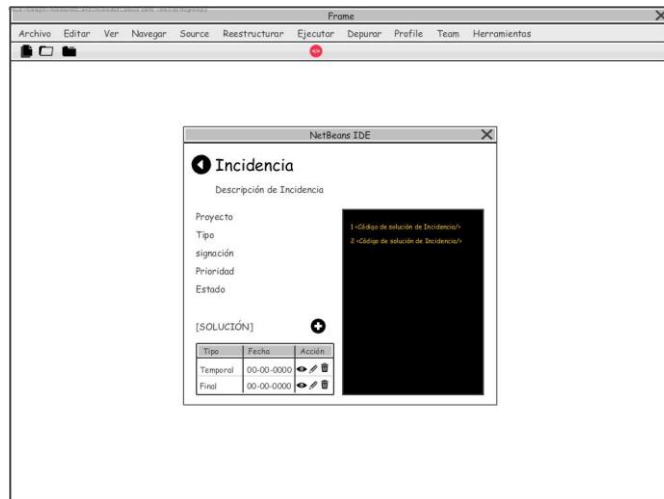


Fig. 46. Boceto plugin de Gestión de solución

IV. Etapa de desarrollo

A. Desarrollo de interfaz de usuario

a. Gestionar Integrante

➤ Interfaz Web

Con esta interfaz se puede gestionar los Integrantes del proyecto en la aplicación web, realizar acciones como agregar integrante por proyecto, listar, editar, ver detalles y quitar a integrante de un proyecto.

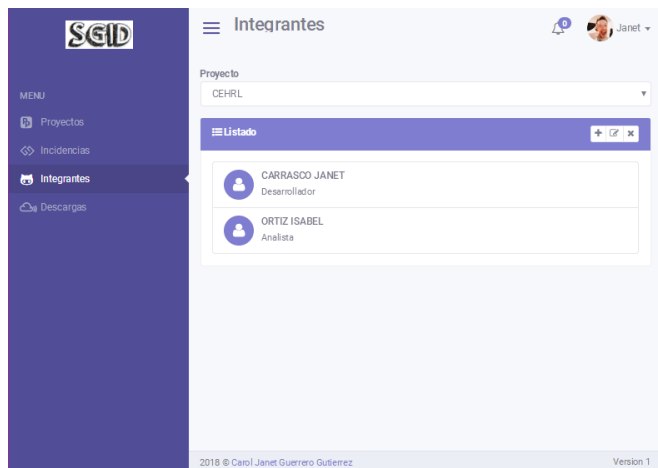


Fig. 47. Interfaz web de Gestionar Integrante

b. Gestionar Solución

➤ Interfaz Web

Con esta interfaz se puede gestionar las soluciones por cada incidencia en la aplicación web, realizar acciones como agregar, editar y ver detalles.

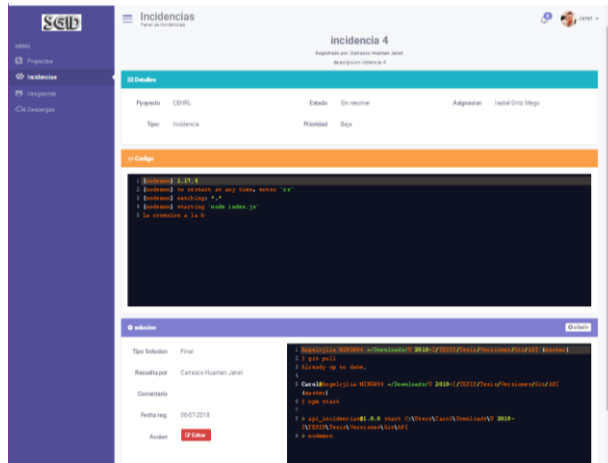


Fig. 48. Interfaz web de Gestionar Solución

➤ Interfaz plugin

Con esta interfaz se puede gestionar las soluciones por cada incidencia en NetBeans, realizar acciones como agregar, editar y ver detalles.

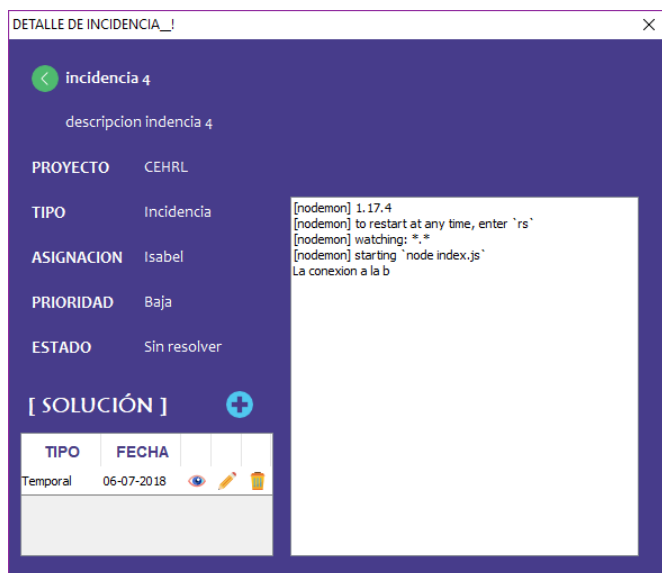


Fig. 49. Interfaz plugin de Gestionar Solución

4.1.13. Etapa de Pruebas

4.1.13.1. Plan de prueba por Sprint

I. Sprint 1

Tabla 59. Planificación de pruebas de Sprint 1

HISTORIA DE USUARIO	ETAPA DE EVALUACIÓN	CRITERIOS DE EVALUACIÓN	ESTADO DE EVALUACIÓN
Gestionar Inicio de Sesión	Análisis	Revisión de historia de usuario.	Aprobado
	Diseño	Revisión de prototipos de interfaz de usuario.	Aprobado
	Desarrollo	Revisión del funcionamiento en respuesta del servidor de la interfaz.	Aprobado
	Sistema	Revisión de respuestas del software mediante casos de prueba.	Aprobado
	Usuario	El sistema debe loguear al usuario correctamente.	Aprobado
Gestionar Incidencias	Análisis	Revisión de historia de usuario.	Aprobado
	Diseño	Revisión de prototipos de interfaz de usuario.	Aprobado
	Desarrollo	Revisión del funcionamiento en respuesta del servidor de la interfaz.	Aprobado
	Sistema	Revisión de respuestas del software mediante casos de prueba.	Aprobado
	Usuario	El sistema debe registrar, listar, modificar y eliminar incidencias.	Aprobado
Gestionar Roles	Análisis	Revisión de historia de usuario.	Aprobado
	Diseño	Revisión de prototipos de interfaz de usuario.	Aprobado
	Desarrollo	Revisión del funcionamiento en respuesta del servidor de la interfaz.	Aprobado
	Sistema	Revisión de respuestas del software mediante casos de prueba.	Aprobado
	Usuario	El sistema debe registrar, listar, modificar e inhabilitar roles.	Aprobado

II. Sprint 2

Tabla 60. Planificación de pruebas de Sprint 2

HISTORIA DE USUARIO	ETAPA DE EVALUACIÓN	CRITERIOS DE EVALUACIÓN	ESTADO DE EVALUACIÓN
Gestionar Personas	Análisis	Revisión de historia de usuario.	Aprobado
	Diseño	Revisión de prototipos de interfaz de usuario.	Aprobado
	Desarrollo	Revisión del funcionamiento en respuesta del servidor de la interfaz.	Aprobado
	Sistema	Revisión de respuestas del software mediante casos de prueba.	Aprobado
	Usuario	El sistema debe registrar personas.	Aprobado
Gestionar Proyecto	Análisis	Revisión de historia de usuario.	Aprobado
	Diseño	Revisión de prototipos de interfaz de usuario.	Aprobado
	Desarrollo	Revisión del funcionamiento en respuesta del servidor de la interfaz.	Aprobado
	Sistema	Revisión de respuestas del software mediante casos de prueba.	Aprobado
	Usuario	El sistema debe registrar, listar, modificar y eliminar proyectos.	Aprobado

III. Sprint 3

Tabla 61. Planificación de pruebas de Sprint 3

HISTORIA DE USUARIO	ETAPA DE EVALUACIÓN	CRITERIOS DE EVALUACIÓN	ESTADO DE EVALUACIÓN
Gestionar Solución	Análisis	Revisión de historia de usuario.	Aprobado
	Diseño	Revisión de prototipos de interfaz de usuario.	Aprobado
	Desarrollo	Revisión del funcionamiento en respuesta del servidor de la interfaz.	Aprobado
	Sistema	Revisión de respuestas del software mediante casos de prueba.	Aprobado
	Usuario	El sistema debe registrar, listar, modificar y eliminar soluciones.	Aprobado
Gestionar Integrantes	Análisis	Revisión de historia de usuario.	Aprobado
	Diseño	Revisión de prototipos de interfaz de usuario.	Aprobado
	Desarrollo	Revisión del funcionamiento en respuesta del servidor de la interfaz.	Aprobado
	Sistema	Revisión de respuestas del software mediante casos de prueba.	Aprobado
	Usuario	El sistema debe agregar, listar, modificar e inhabilitar integrantes.	Aprobado

4.1.13.2.Prueba de aceptación

I. Sprint 1

PRUEBA DE ACEPTACIÓN			
Número de prueba	1	Número de historia de usuario	1
Nombre de la prueba	Prueba de aceptación de inicio de sesión		
Objetivo	Verificar que se efectúe la verificación del usuario para su acceso al sistema		
ítem	Precondición	Entrada	Resultado Esperado
1.1	Usuario registrado y activo	Correo y contraseña	El sistema lo dirige a la interfaz principal.
1.2	Usuario ingresa datos incorrectos	Correo y contraseña	El sistema muestra el siguiente mensaje: "Dato incorrecto"
1.3	Usuario registrado pero inactivo	Correo y contraseña	El sistema muestra el siguiente modal de alerta: "Usuario inactivo"
Evaluación de la prueba	Prueba satisfactoria		

PRUEBA DE ACEPTACIÓN			
Número de prueba	2	Número de historia de usuario	2
Nombre de la prueba	Prueba de aceptación de incidencia		
Objetivo	Verificar que se efectúe el registro, listado, modificación y eliminación de incidencia.		
ítem	Precondición	Entrada	Resultado Esperado
2.1	Realizar registro de incidencia	Usuario llena campos del formulario incidencia.	El sistema muestra un mensaje de confirmación; "Registrado correctamente".
2.2	Listar incidencias registradas	Usuario lista los registros de las incidencias.	El sistema muestra el listado de las incidencias.
2.3	Modificar incidencia registrada	Usuario edita campos del formulario incidencia.	El sistema muestra un mensaje de alerta; "Se actualizó correctamente".
2.4	Eliminar incidencia registrada	Usuario elimina incidencia registrada no deseada.	El sistema muestra un mensaje de confirmación; "¿Está seguro de eliminar esta incidencia?".

Evaluación de la prueba		Prueba satisfactoria	
PRUEBA DE ACEPTACIÓN			
Número de prueba	3	Número de historia de usuario	3
Nombre de la prueba	Prueba de aceptación de roles		
Objetivo	Verificar que se efectúe		
ítem	Precondición	Entrada	Resultado Esperado
3.1	Realizar registro de rol	Usuario llena campos del formulario rol.	El sistema muestra un mensaje de confirmación; "Registrado correctamente".
3.2	Listar roles registrados	Usuario lista los registros de los roles.	El sistema muestra el listado de los roles.
3.3	Modificar rol registrado	Usuario edita campos del formulario rol.	El sistema muestra un mensaje de alerta; "Se actualizó correctamente".
3.4	Inhabilitar rol registrado	Usuario inhabilita rol registrado no necesario.	El sistema muestra un mensaje de confirmación; "¿Está seguro de inhabilitar este rol?".
Evaluación de la prueba		Prueba satisfactoria	

II. Sprint 2

PRUEBA DE ACEPTACIÓN			
Número de prueba	4	Número de historia de usuario	4
Nombre de la prueba	Prueba de aceptación de personas		
Objetivo	Verificar que se efectúe		
ítem	Precondición	Entrada	Resultado Esperado
4.1	Realizar el registro de persona	Usuario llena campos del formulario registro.	El sistema muestra un mensaje de confirmación; "Registrado correctamente".
4.2	Modificar persona registrada	Usuario edita campos del formulario perfil.	El sistema muestra un mensaje de alerta; "Se actualizó correctamente".
Evaluación de la prueba		Prueba satisfactoria	

PRUEBA DE ACEPTACIÓN			
Número de prueba	5	Número de historia de usuario	5
Nombre de la prueba	Prueba de aceptación de proyectos		
Objetivo	Verificar que se efectúe el registro, listado, modificación y eliminación de proyectos.		
ítem	Precondición	Entrada	Resultado Esperado
5.1	Realizar registro de proyecto	Usuario llena campos del formulario proyecto.	El sistema muestra un mensaje de confirmación; "Registrado correctamente".
5.2	Listar proyectos registrados	Usuario lista los registros de los proyectos.	El sistema muestra el listado de los proyectos.
5.3	Modificar proyecto registrado	Usuario edita campos del formulario proyecto.	El sistema muestra un mensaje de alerta; "Se actualizó correctamente".
5.4	Eliminar proyecto registrado	Usuario elimina proyecto registrado no deseado.	El sistema muestra un mensaje de confirmación; "¿Está seguro de eliminar esta incidencia?".
Evaluación de la prueba	Prueba satisfactoria		

III. Sprint 3

PRUEBA DE ACEPTACIÓN			
Número de prueba	6	Número de historia de usuario	6
Nombre de la prueba	Prueba de aceptación de solución		
Objetivo	Verificar que se efectúe el registro, listado y modificación de solución.		
ítem	Precondición	Entrada	Resultado Esperado
6.1	Realizar registro de solución	Usuario llena campos del formulario solución.	El sistema muestra un mensaje de confirmación; "Registrado correctamente".
6.2	Listar soluciones registradas	Usuario lista los registros de las soluciones.	El sistema muestra el listado de las soluciones.
6.3	Modificar solución registrada	Usuario edita campos del formulario solución.	El sistema muestra un mensaje de alerta; "Se actualizó correctamente".
Evaluación de la prueba	Prueba satisfactoria		

PRUEBA DE ACEPTACIÓN			
Número de prueba	7	Número de historia de usuario	7
Nombre de la prueba	Prueba de aceptación de integrantes		
Objetivo	Verificar que se efectúe el agregado, la edición, modificación de permisos e inhabilitación de integrante.		
ítem	Precondición	Entrada	Resultado Esperado
7.1	Realizar el agregado de integrante	Usuario selecciona integrante.	El sistema muestra listado con nuevo integrante.
7.2	Listar integrantes agregados	Usuario selecciona proyecto.	El sistema muestra el listado de los integrantes.
7.3	Modificar permisos de integrante	Usuario edita permisos del formulario integrantes.	El sistema muestra un mensaje de alerta; "Se actualizó correctamente".
7.4	Inhabilitar integrante agregado	Usuario inhabilita integrante no necesario.	El sistema cambia el estado a "Inactivo".
Evaluación de la prueba	Prueba satisfactoria		

4.1. En base a los objetivos de la investigación

4.1.1. Disminución de la cantidad de fuentes de búsqueda de solución de incidencias.

A. Objetivo

Disminuir la cantidad de fuentes de búsqueda de solución de incidencias.

B. Software

El software permite realizar búsquedas mediante la interfaz de incidencias (Fig. 50) en la aplicación web y la interfaz de buscar incidencia (Fig. 51) en el plugin.

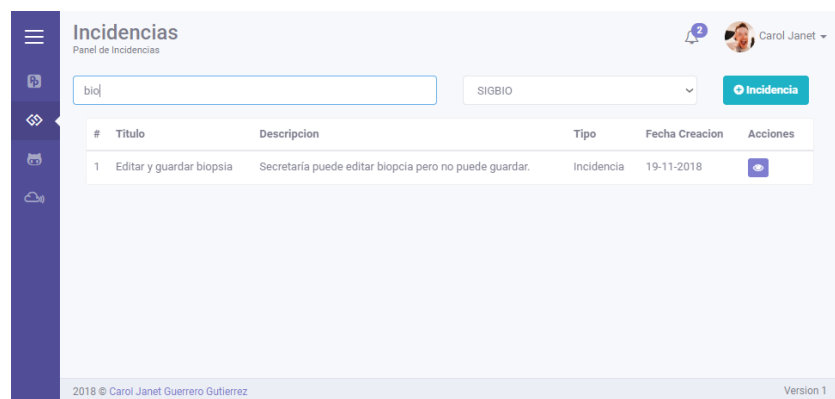


Fig. 50. Búsqueda en interfaz web de incidencias

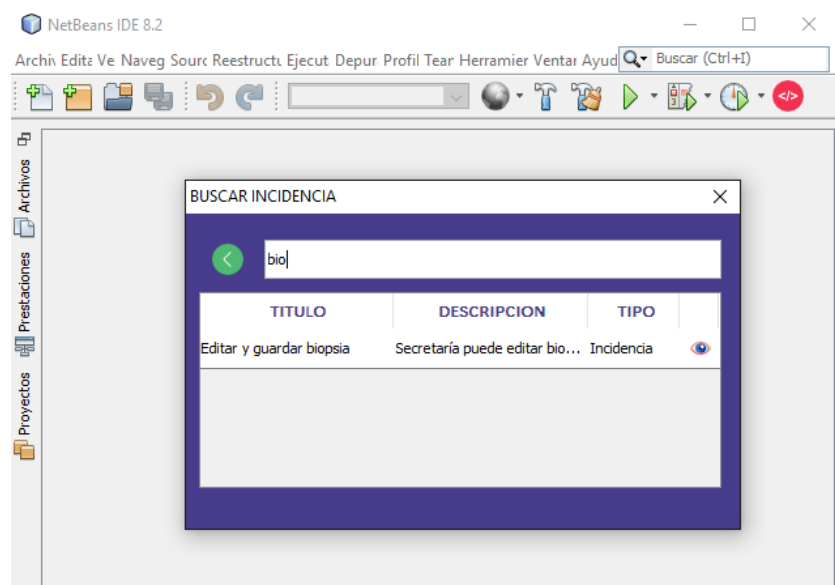


Fig. 51. Búsqueda en interfaz plugin de soluciones

C. Resultado

El software permite acceder en una misma ubicación, a todo tipo de incidencias (incidencia, método, consulta) de cualquier proyecto, registradas por el mismo usuario o compartidas por otros. Para lograr esto tiene que acceder a las interfaces de incidencias en la aplicación web (*Fig. 50*) y la interfaz de buscar incidencia en el plugin (*Fig. 51*). En la web se debe escribir la palabra clave (*Fig. 52*. Buscador de incidencias en web *Fig. 52*) y a medida que realiza la escritura se va desarrollando la búsqueda, mientras que en el IDE se debe escribir la palabra clave o parte de ella (*Fig. 53*) y presionar la tecla *Enter*, para que se realice la búsqueda.

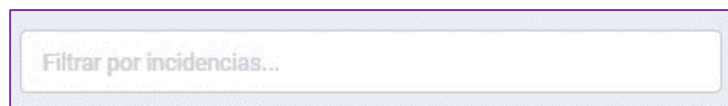


Fig. 52. Buscador de incidencias en web



Fig. 53. Buscador de incidencias en plugin

Con ambas interfaces se puede realizar la búsqueda en un repositorio de incidencias, mostrando el título, descripción y tipo de incidencia como referente, así tener una noción previa del contenido de la incidencia, la finalidad es, acceder en una misma ubicación, a todo tipo de incidencias (incidencia, método, consulta) de cualquier proyecto, registradas por el mismo usuario o compartidas por otros.

4.1.2. Disminución el tiempo promedio de búsqueda de solución a incidencias.

A. Objetivo

Disminuir el tiempo promedio de búsqueda de solución a incidencias.

B. Software

El software permite realizar búsquedas mediante la interfaz de incidencias (*Fig. 50*) en la aplicación web y la interfaz de buscar incidencia (*Fig. 51*) en el plugin.

C. Resultado

El software permite realizar la búsqueda de incidencias (incidencias, métodos, consultas) en un solo lugar. Para lograr esto tiene que acceder a las interfaces de incidencias en la aplicación web (*Fig. 50*) y la interfaz de buscar incidencia en el plugin (*Fig. 51*). En la web se debe escribir la palabra clave (*Fig. 52*). Buscador de incidencias en web (*Fig. 52*) y a medida que realiza la escritura se va desarrollando la búsqueda, mientras que en el IDE se debe escribir la palabra clave o parte de ella (*Fig. 53*) y presionar la tecla *Enter*, para que se realice la búsqueda. Con ambas interfaces se puede realizar la búsqueda en un repositorio de incidencias, mostrando el título, descripción y tipo de incidencia como referente, así poder realizar búsquedas instantáneas y en corto plazo.

4.1.3. Aumento de la cantidad de métodos y consultas registrados en un repositorio.

A. Objetivo

Aumentar la cantidad de métodos y consultas registrados en un repositorio.

B. Software

El software permite registrar incidencias mediante la interfaz de Añadir una incidencia (*Fig. 54*) en la interfaz web y Registro de Incidencia (*Fig. 55*) en la interfaz del plugin.

The screenshot shows a web interface for adding an incident. At the top, there is a purple header with a computer icon and the text "Añadir un incidencia" and "Gestion de proyectos". Below the header, the form contains the following elements:

- Proyecto:** SIGHOR
- Título:** A text input field.
- Descripcion:** A larger text input field.
- Tipo:** Three radio buttons: "Incidencia", "Metodo" (which is selected), and "Consulta".
- Imagen:** A button labeled "Examinar..." followed by the text "No se ha seleccionado ningún archivo."
- Solucion:** A large, dark text area for entering the solution.

At the bottom of the form, there are two buttons: a red "Cancelar" button and a purple "Guardar" button.

Fig. 54. Registro de método o consulta en interfaz web

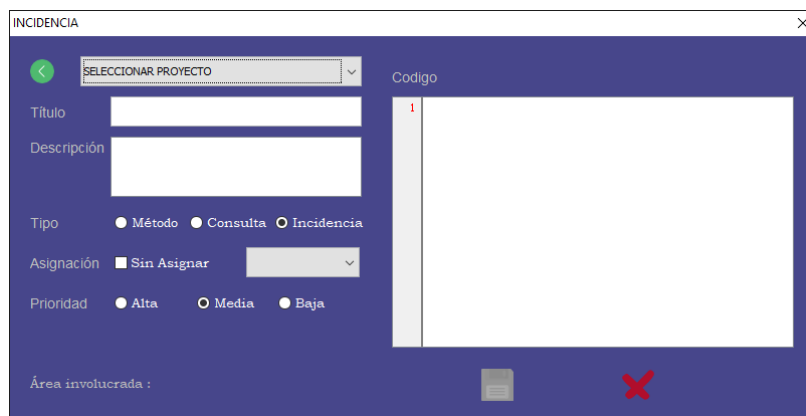


Fig. 55. Registro de método o consulta en interfaz plugin

C. Resultado

El software permite realizar el registro de métodos y consultas frecuentes, teniendo todo en un solo lugar y al alcance de la necesidad, con acceso disponible en la web e IDE. Para lograr esto tiene que acceder a las interfaces de registro de la web (Fig. 54) y plugin (Fig. 55). En la web se debe acceder al módulo de incidencias y luego seleccionar el botón de añadir incidencias (Fig. 56), mientras que en el IDE se debe seleccionar la opción de nuevo (Fig. 57) desde el menú principal.



Fig. 56. Botón de añadir incidencia en web

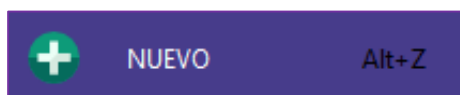


Fig. 57. Opción de añadir incidencias en plugin

Con ambas interfaces se puede realizar el registro de métodos o consultas frecuentes y así tenerlas en una sola ubicación.

4.1.4. Aumento de la cantidad de incidencias registradas por sistema desarrollado en la entidad.

A. Objetivo

Aumentar la cantidad de incidencias registradas por sistema desarrollado en la entidad.

B. Software

El software permite registrar incidencias por proyecto mediante la interfaz de Incidencias (Fig. 58 Fig. 50) en la aplicación web y la interfaz de agregar incidencia (Fig. 59 Fig. 51) en el plugin.

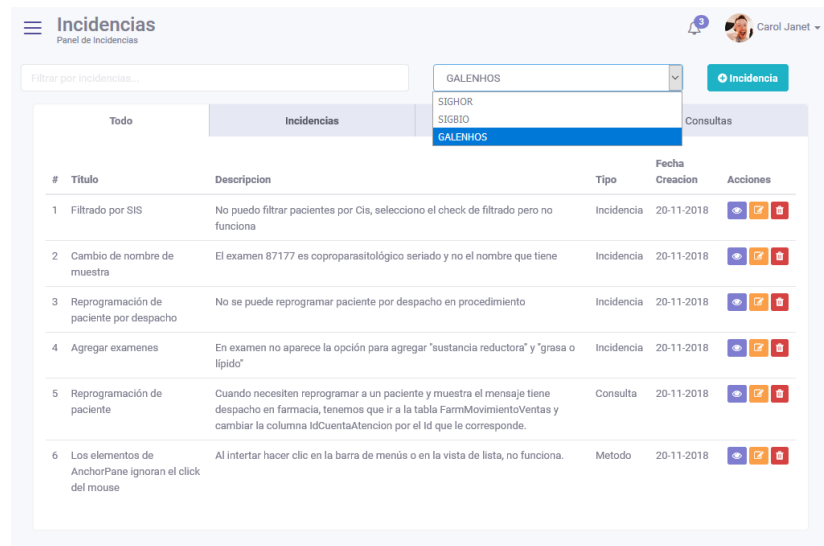


Fig. 58. Interfaz web de registro de incidencia por proyecto

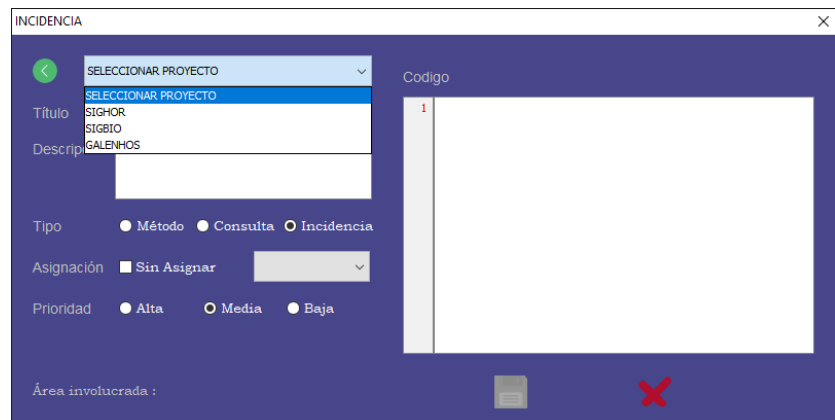


Fig. 59. Interfaz plugin de registro de incidencia por proyecto

C. Resultado

Antes de la aplicación del software no se contaba con incidencias registradas por sistema desarrollado, luego de aplicar el software se puede realizar el registro de 159 incidencias por cada proyecto desarrollado, lo que permite poder clasificarlas y atenderlas según prioridad de las mismas, para lograr esto tiene que acceder a las interfaces de registro de la web (Fig. 58Fig. 54) y plugin (Fig. 59Fig. 55). En la web se debe acceder a la interfaz de incidencia y seleccionar el proyecto (Fig. 60) al que pertenece la incidencia y luego se selecciona el botón añadir, mientras que en el IDE se debe acceder a la interfaz de registro de incidencia y ahí seleccionar el proyecto (Fig. 61) a la cual pertenece.

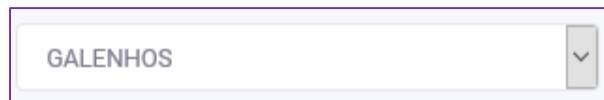


Fig. 60. Combo en web de proyectos a elegir



Fig. 61. Combo en plugin de proyectos a elegir

Con ambas interfaces se puede realizar el registro de incidencias por cada proyecto donde se presenta.

4.1.5. Aumento de la cantidad de soluciones guardadas en una biblioteca de software definitivo.

A. Objetivo

Aumentar la cantidad de soluciones guardadas en una biblioteca de software definitivo.

B. Software

El software permite registrar soluciones mediante la interfaz de Detalle de la incidencia en la interfaz web (Fig. 62) e interfaz plugin (Fig. 63).

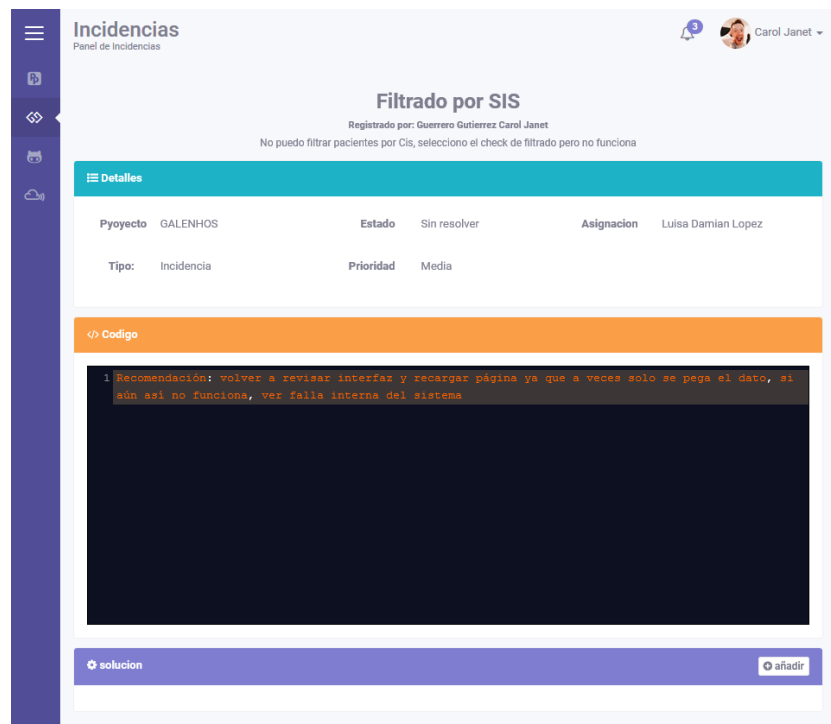


Fig. 62. Interfaz web de detalle de incidencia



Fig. 63. Interfaz plugin de detalle de incidencia

C. Resultado

El software permite realizar el registro de la solución de cada incidencia, logrando tenerlas en un solo lugar y al alcance de la necesidad, ya que su acceso está disponible tanto en la web como en el IDE. Para lograr esto tiene que acceder a las interfaces de registro de la web (Fig. 62Fig. 54) y plugin (Fig. 63Fig. 55). En la web se debe acceder al detalle de la incidencia y luego seleccionar el botón de añadir solución (Fig. 64), mientras que en el IDE se debe seleccionar la opción de agregar (Fig. 65Fig. 57) desde el detalle de la incidencia.



Fig. 64. Botón de añadir solución en web



Fig. 65. Opción de añadir solución en plugin

Con ambas interfaces se puede realizar el registro de soluciones por incidencia y así tenerlas en una sola ubicación.

V. DISCUSIÓN

El proceso de desarrollo es parte fundamental del ciclo de vida del software y el uso de herramientas adecuadas permite contribuir al proceso es post de su mejora, como menciona Egusquiza [16] concluyendo que adoptar herramientas tecnológicas durante el proceso de desarrollo de software es de gran importancia y aporta un valor agregado al mismo. Es por ello que al usar una plataforma colaborativa se puede contribuir al proceso de desarrollo donde, se encuentren involucrados más de un participante y se busque la mejora continua, como lo afirman Briede, Cabello, Pérez y Arriaga [11] en Chile y Cornetero y Rojas en Lambayeque [17] demostrando lo viable que es la construcción de una plataforma colaborativa en el logro de la comunicación activa entre participantes, retroalimentación y control de procesos

Se ha realizado un análisis de los resultados de la plataforma colaborativa durante el proceso de desarrollo de software en la unidad de desarrollo de la División de Tecnologías de Información. El enfoque estuvo orientado en base a un análisis cuantitativo de los indicadores (Tabla IV) expuestos en el capítulo III, y su evaluación Pre Test y Post Test mediante observación y entrevistas.

5.1. Indicador 1: Cantidad de fuentes de consulta de solución

A. Resultado de Pre-test y Post-test

Tabla 62. Cantidad de fuentes de consulta de solución

Indicador	Pre Test (O₁)	Post Test (O₂)	Diferencia (O₂ – O₁)
Cantidad de fuentes de consulta de solución	4	1	-3

Antes de la aplicación del software, la búsqueda de solución se realizaba mediante las siguientes fuentes:

- i. Comentarios de código fuente; Se realizaba la búsqueda de comentarios realizados en líneas de código entre todos los archivos de uno o más sistemas, mediante el acceso y revisión de archivo por archivo hasta encontrar o no la solución.
- ii. Archivos de procesamiento de texto; Se realizaba la búsqueda en mensajes, correos, documentos Word y block de notas guardados con un nombre referente, sin descripción y en diferentes ubicaciones, mediante el acceso y revisión de archivo por archivo hasta encontrar o no la solución.
- iii. Consulta a compañero; Cuando este se encuentra en la oficina se le realizaba la consulta respecto a código, y el mismo, busca, muestra y comparte la solución hallada por algún medio de comunicación como correo o redes sociales, con el desarrollador, si el compañero no se encuentra en la oficina, debe esperarlo, si se encuentra de vacaciones o de permiso, debe realizar una llamada telefónica para consultarle y esperar su respuesta, pero en el peor de los casos, si ya no labora en la institución, deberá ver cualquier medio de comunicación para realizar la consulta.
- iv. Búsqueda en Internet; Se realizaba la búsqueda en un navegador mediante la descripción del problema o palabras clave, en lugares como foros, páginas de desarrollo, comentarios y blogs teniendo que leer todo el texto previo o ver si la solución era correcta a través de prueba y error hasta encontrar o no la solución.

Luego de aplicar el software se obtuvo una variación del 75%, reduciendo a 1 la fuente de búsqueda de solución, de tal manera que la fuente de consulta es un repositorio central.

5.2. Indicador 2: Tiempo promedio de búsqueda de solución a incidencias

A. Resultado de Pre-test y Post-test

Tabla 63. Tiempo promedio de búsqueda de solución a incidencias

Indicador	Pre Test (O₁)	Post Test (O₂)	Diferencia (O₂ – O₁)
Tiempo promedio de búsqueda de solución a incidencias	5280s	5s	-5275

Antes de la aplicación del software, el tiempo promedio de búsqueda de solución de incidencias era de ochenta y ocho minutos, como se muestran a continuación:

Tabla 64. Tiempos de consulta por fuente

Nº	Nombre	Tiempo (minutos)	
1	Ing. Baldera	Mínimo	10
		Máximo	510
		Promedio	120
2	Ing. Díaz	Mínimo	10
		Máximo	220
		Promedio	63
3	Ing. Quepuy	Mínimo	10
		Máximo	300
		Promedio	80

Luego de aplicar el software se obtuvo una variación del 99.91%, es decir, se puede realizar la búsqueda de incidencias en un solo lugar con un tiempo promedio de consulta de 5 segundos.

5.3. Indicador 3: Cantidad de métodos y consultas almacenadas en un repositorio

A. Resultado de Pre-test y Post-test

Tabla 65. Cantidad de métodos y consultas almacenadas en un repositorio

Indicador	Pre Test (O₁)	Post Test (O₂)	Diferencia (O₂ – O₁)
Cantidad de métodos y consultas almacenadas en un repositorio	60	239	179

Antes de la aplicación del software se contaba con 60 métodos y consultas registrados de forma dispersa, como se muestra a continuación:

Tabla 66. Cantidad total de métodos y consultas frecuentes

N°	Nombre	Cantidad
1	Ing. Baldera	25
2	Ing. Díaz	20
3	Ing. Quepuy	15

Luego de aplicar el software se logró una variación del 75%, es decir, se pudo realizar el registro de 239 métodos y consulta frecuentes, teniendo todo en un solo lugar y al alcance de la necesidad, con acceso disponible en la web e IDE.

5.4. Indicador 4: Cantidad de incidencias registradas por sistema desarrollado

A. Resultado de Pre-test y Post-test

Tabla 67. Cantidad de incidencias registradas por proyecto

Indicador	Pre Test (O₁)	Post Test (O₂)	Diferencia (O₂ – O₁)
Cantidad de incidencias registradas por sistema desarrollado	0	159	159

Antes de la aplicación del software no se contaba con incidencias registradas por sistema desarrollado, luego de aplicar el software se logró una variación del 100%, es decir, se pudo realizar el registro de 159 incidencias por cada proyecto desarrollado, lo que permite poder clasificarlas y atenderlas según prioridad de las mismas.

5.5. Indicador 5: Cantidad de soluciones por incidencia registradas en el repositorio

A. Resultado de Pre-test y Post-test

Tabla 68. Cantidad de soluciones por incidencia registradas en el repositorio

Indicador	Pre Test (O₁)	Post Test (O₂)	Diferencia (O₂ – O₁)
Cantidad de soluciones por incidencia registradas en el repositorio	0	133	133

Antes de la aplicación del software no se contaba con soluciones guardadas en una biblioteca de software definitivo, luego de aplicar el software se logró una variación del 100%, es decir, se pudo realizar el registro de 133 soluciones por incidencia, lo que permite tenerlas en un solo lugar y al alcance de la necesidad, ya que su acceso está disponible tanto en la web como en el IDE.

Con la aplicación de la plataforma colaborativa se puede contribuir mediante la gestión de incidencias al proceso de desarrollo de software, como se demuestra en los resultados de los indicadores analizados con anterioridad (Tabla 62, Tabla 63, Tabla 65, Tabla 67 y Tabla 68), para lo cual se usó de base referente la Biblioteca de Infraestructura de Tecnologías de Información, así como lo hicieron Gómez [14], Delgado [19], Quintero [12] y Gonzales [18], quienes aplicaron la metodología ITIL para mejorar los procesos mediante la gestión de incidencias, logrando la reducción de tiempos en su atención y solución e incrementando la productividad mediante la definición de métricas adecuadas.

VI. CONCLUSIONES

- 1.** Antes de la aplicación del software, la búsqueda de solución se realizaba mediante 4 fuentes de consulta, después de aplicar el software se logró disminuir en un 75% la cantidad de fuentes de búsqueda de solución a incidencias, obteniendo solo una fuente, en un repositorio central.
- 2.** Antes de la aplicación del software, el tiempo promedio de búsqueda de solución de incidencias era de 88 minutos, después de aplicar el software se logró disminuir en un 99.91% el tiempo promedio de búsqueda de solución a incidencias obteniendo un tiempo promedio de 5 segundos para una búsqueda.
- 3.** Antes de la aplicación del software se contaba con 60 métodos y consultas registrados de forma dispersa, después de aplicar el software se logró aumentar en un 75% la cantidad de métodos y consultas registradas en un repositorio, obteniendo 179 métodos y consultas al alcance de todos los integrantes del proyecto relacionado.
- 4.** Antes de la aplicación del software no se contaba con incidencias registradas por sistema desarrollado, después de aplicar el software se logró aumentar en un 100% la cantidad de incidencias registradas por sistema desarrollado en la entidad, obteniendo 159 incidencias al alcance de todos los integrantes del proyecto relacionado.
- 5.** Antes de la aplicación del software no se contaba con soluciones guardadas en una biblioteca de software definitivo, después de aplicar el software se logró aumentar en un 100% la cantidad de soluciones guardadas en una biblioteca de software definitivo, obteniendo 133 soluciones al alcance de todos los integrantes del proyecto relacionado.

VII. RECOMENDACIONES

- 1.** Ampliar el uso del plugin para otros editores de código, es decir, elaborar plugin que sean compatibles con otros editores de código diferentes de NetBeans, como Sublime Text, Visual Studio Code, Atom y Brackets.
- 2.** Realizar captura automática de errores en el IDE de NetBeans, es decir, trabajar de la mano con el compilador de NetBeans y que el plugin capture automáticamente errores de ejecución y compilación
- 3.** Realizar asignaciones automáticas por cargas de trabajo, es decir, establecer secuencias de asignación configurables que se permita o esté aceptado para diversas circunstancias presentadas en el equipo de trabajo.
- 4.** Ampliar la plataforma con nuevos módulos para otras áreas de la División de Tecnologías de Información, es decir, gestionar incidencias de soporte técnico y data center a nivel de personal de la división y poder anexarlo una mesa de servicio a nivel de usuario.

VIII. LISTA DE REFERENCIAS

- [1] J. Zaval, «¿Por Qué Fracasan los Proyectos de Software?; Un Enfoque Organizacional,» 2004. [En línea]. Available: <https://claroline.ucaribe.edu.mx/claroline/claroline/backends/download.php?url=L3Bvci1xdWUtZmFsbGFuLWxvcy1wcm95LWRILXNvZnQucGRm&cidReset=true&cidReq=NI0215>. [Último acceso: Junio 2017].
- [2] J. Gómez, «El LaboratoriodelasTI,» 16 Mayo 2016. [En línea]. Available: <http://www.laboratorioti.com/2016/05/16/informe-del-caos-2015-chaos-report-2015-bien-mal-fueron-los-proyectos-ano-2015/>. [Último acceso: Junio 2017].
- [3] ASPgems, «ASPgems Soluciones Ágiles,» 21 Enero 2014. [En línea]. Available: <https://aspgems.com/blog/leticia/el-61-de-los-proyectos-de-desarrollo-de-software-fracasan>. [Último acceso: Junio 2017].
- [4] J. Gómez, «El Laboratorio de las TI,» Enero 2015. [En línea]. Available: <http://www.laboratorioti.com/2015/01/21/esta-es-la-clave-del-fracaso-de-tu-proyecto-evitala/>. [Último acceso: Julio 2017].
- [5] VirtualPro, «VIRTUALPRO Procesos Industriales,» 24 Abril 2017. [En línea]. Available: <https://www.revistavirtualpro.com/noticias/las-empresas-de-software-de-america-latina-y-el-caribe-apuestan-a-la-insercion-exportadora->. [Último acceso: Junio 2017].
- [6] Gestión, «Gestión,» Gestión, Junio 2014. [En línea]. Available: <http://gestion.pe/empresas/54-empresas-latinoamerica-opta-desarrollar-software-propio-2101350>. [Último acceso: Junio 2017].
- [7] Gestión, «Gestión,» Gestión, Abril 2016. [En línea]. Available: <http://gestion.pe/tecnologia/peru-tiene-potencial-desarrollo-software-dedicado-al-sector-empresarial-2158494>. [Último acceso: Junio 2017].
- [8] Gestión, «Gestión,» Gestión, Marzo 2017. [En línea]. Available: <http://gestion.pe/economia/pymes-cierran-mas-contratos-software-gestion-procesos-2184310>. [Último acceso: Junio 2017].
- [9] ProChile, «ProChile Ministerio de Relaciones Exteriores,» Septiembre 2013. [En línea]. Available: <http://www.prochile.gob.cl/documento-biblioteca/estudio-de-mercado-servicios-de-software-peru/>. [Último acceso: Junio 2017].
- [10] I. Sommerville, Ingeniería del software, 9na ed., Madrid: Pearson, 2011.
- [11] J. Briede, M. Cabello, C. Pérez y A. Arriagada, «Plataforma Colaborativa para la Gestión de Proyectos de Diseño Industrial,» *Formación Universitaria*, vol. 9, n° 3, pp. 61-73, 2016.
- [12] L. F. Quintero Gómez, *Modelo basado en ITIL para la Gestión de los Servicios de TI en la Cooperativa de Caficultores de Manizales*, Univ. Autónoma de Manizales, 2015.
- [13] I. Aldabari Carrillo, *Aplicación web de gestión financiera personal desarrollada con MEAN.JS*, Universidad Politécnica de Madrid, 2016.
- [14] J. R. Gómez Álvarez, «Implantación de los procesos de gestión de incidentes y gestión de problemas según ITIL v3.0 en el área de tecnologías de información de una entidad financiera,» Pontificia Universidad Católica del Perú, Lima, 2012.
- [15] R. p. Ocola Agüero, «Metodología de desarrollo y mantenimiento de software para un fábrica de software,» Universidad Nacional de Ingeniería, Lima, 2012.

- [16] H. J. Egusquiza Herrada, «Modelo de mejora del ciclo de vida del desarrollo de software con referencia a las ISO/IEC 29110 Caso: Mype Holinsys,» Universidad de Lima, Lima, 2016.
- [17] M. A. Cornetero Mendoza y K. M. Rojas Villarue, «sistema colaborativo para mejorar el proceso de planificación operativa de eventos,» Universidad Católica Santo Toribio de Mogrovejo, Chiclayo, 2015.
- [18] J. A. Gonzales Flores, «Implementación del marco de trabajo ITIL v3.0 para el proceso de gestión de incidencias en el área del centro de sistemas de información de la Gerencia Regional de Salud de Lambayeque,» Universidad Católica Santo Toribio de Mogrovejo, Chiclayo, 2015.
- [19] A. H. Delgado Chávarry, «Implementación del marco de trabajo ITIL para apoyar la gestión de los servicios del centro de sistemas de información en la Gerencia Regional de Salud,» Universidad Católica Santo Toribio de Mogrovejo, Chiclayo, 2015.
- [20] J. Bravo Rodríguez, «Sistema de interacción persona - computador,» Universidad de Castilla La Mancha, Cuenca, 2001.
- [21] J. Tramullas, P. Garrido Picazo y A. I. Sánchez Casabón, «Groupware y software social: Propuesta de marco de evaluación analítica para herramientas de software libre,» *El profesional de la información*, vol. 20, n° 4, pp. 465 - 473, 2011.
- [22] C. A. Ellis, S. J. Gibbs y G. Rein, «Groupware: some issues and experiences,» *Magazine Communications of the ACM*, vol. 34, n° 1, pp. 39 - 58, Jan 1991.
- [23] L. Guerra G. y A. Bedini G., Gestión de proyectos de software, Valparaíso: UTFSM, 2005.
- [24] E. Arellano Alameda, Proyéctica: Buenas prácticas para la gestión de proyectos de software, Madrid: Editorial Universitaria Ramón Areces, 2016.
- [25] D. Cantone, Implementación y Debugging, España: M.P.Ediciones, 2006.
- [26] F. Alonso, L. Martínez y J. Sogovia, Introducción a la Ingeniería de software, Madrid: Delta Publicaciones, 2005.
- [27] I. Sommerville, Ingeniería del software, Madrid: Pearson Educación, 2005.
- [28] S. R. Gómez Palomo y E. Moraleda Gil, Aproximación a la Ingeniería del Software, Madrid: Ramón Areces S.A., 2015.
- [29] D. Ramos Cardozzo, Desarrollo de Software: Requisitos, Estimaciones y Análisis, España: IT Campus Academy, 2016.
- [30] R. Noriega Martínez, El proceso de Desarrollo de Software, España: IT Campus Academy, 2017.
- [31] Oracle Corporation, «Java,» Oracle, 2015. [En línea]. Available: https://www.java.com/es/about/whatis_java.jsp?bucket_value=desktop-firefox63-windows10-64bit&in_query=no. [Último acceso: 2018].
- [32] A. Durango, Á. Arias y J. Gracia, Curso de programación con Java, España: IT Campus Academy, 2016.
- [33] Java, «Java,» Sun Microsystems, 2014. [En línea]. Available: <http://www.cs.mun.ca/java-api-1.5/>. [Último acceso: 2018].
- [34] J. S. Dean y R. H. Dean, Introducción a la programación con Java, México: Mc Graw Hill, 2009.
- [35] J. L. Ávila Jiménez, El ciclo de vida del desarrollo de aplicaciones, España: Elearning, 2016.

- [36] V. j. Eslava Muñoz, El nuevo PHP. Conceptos avanzados, España: Bubok Publishing, 2013.
- [37] M. Gris, Iniciación a Internet, Barcelona: ENI, 2014.
- [38] M. L. Peláez recios, Desarrollo de componente software en sistemas ERP - CRM, España: Elearning, 2015.
- [39] F. L. Osorio Rivera, Introducción a la programación en Java. un enfoque práctico, Colombia: Instituto Tecnológico Metropolitano, 2007.
- [40] M. Merino, «TICbeat,» Axel Springer España, 12 Julio 2014. [En línea]. Available: <http://www.ticbeat.com/tecnologias/que-es-una-api-para-que-sirve/>. [Último acceso: 2018].
- [41] A. De Vega Luna, R. De la Heras del Dedo, C. Domingo soriano, C. Santa Cecilia y S. De la Puente González, Historias de Developers, Barcelona: Lulu.com, 2013.
- [42] F. Luna, C. Peña Millahual y M. Iacono, Programación Web Full Stack. Web apps y plataformas amigables, Argentina: Redusers, 2018.
- [43] K. Schwaber y J. Sutherland, «La Guía de Scrum,» Noviembre 2017. [En línea]. Available: <https://www.scrumguides.org>. [Último acceso: 2018].
- [44] SlideSalad, «SlideSalad,» EBSAR Group, Noviembre 2018. [En línea]. Available: <https://www.slidesalad.com/product/scrum-process-google-slides-presentation-template/>. [Último acceso: 2018].
- [45] E. Ruiz Larrocha, Nuevas tendencias en los sistemas de información, Madrid: Centro de Estudios Ramon Areces, 2017.
- [46] S. Ríos Huercano, Manual de ITIL V3, Sevilla: Creative Commons, 2014.
- [47] J. J. Mora Pérez, Capacity planning IT: una aproximación práctica, Madrid: Creative Commons, 2012.
- [48] L. Joyanes Aguiler, Big Data, análisis de grandes volúmenes de datos en organizaciones., México: Alfaomega, 2016.
- [49] C. Caballero Gonzales y R. Montoya Cordero, Almacenamiento de la información e introducción a SGBD, Madrid: Paraninfo, 2016.
- [50] Amazon, «aws,» Amazon Web Services, 2016. [En línea]. Available: <https://aws.amazon.com/es/nosql/>. [Último acceso: 2018].
- [51] M. Altarad, «Pandorafms Monitoring Blog,» Pandorafms, 20 Abril 2017. [En línea]. Available: <https://blog.pandorafms.org/nosql-databases-the-definitive-guide/>. [Último acceso: 2018].
- [52] Apptunix, «Apptunix,» Apptunix, 28 Septiembre 2018. [En línea]. Available: <https://www.apptunix.com/sql-or-nosql-database/>. [Último acceso: 2018].
- [53] G. Chandra Deka, NoSQL Database for Storage and Retrieval of Data in Cloud, Estados Unidos: CRC Press, 2017.
- [54] Linnovate, «MEAN.IO,» Linnovate, 2014. [En línea]. Available: <http://mean.io/blog/>. [Último acceso: 2018].
- [55] Comunidad MEAN.JS, «MEAN.JS,» Creative Commons, 2014. [En línea]. Available: <http://meanjs.org/>. [Último acceso: 2018].
- [56] J. F. Cía, «BBVA API_Market,» BBVA, 08 Julio 2015. [En línea]. Available: <https://bbvaopen4u.com/es/actualidad/mean-el-cuarteto-de-desarrollo-full-stack-en-javascript-para-hacer-aplicaciones-web-0>. [Último acceso: 2018].

- [57] Y. L. Cortés Pineda, «Platzi,» platzi, 2017. [En línea]. Available: <https://platzi.com/blog/que-es-mean-full-stack-javascript/>. [Último acceso: 2018].
- [58] Google, «Google Trends,» Google, 2018. [En línea]. Available: https://trends.google.es/trends/explore?date=today%205-y&geo=PE&q=%2Fm%2F05z_r2n,%2Fm%2F0_v2szx,%2Fm%2F0j45p7w,%2Fm%2F0bbxf89. [Último acceso: 2018].
- [59] Inubo, «Inubo,» ciTECam, 10 Octubre 2017. [En línea]. Available: <http://inubo.es/noticia/los-10-mejores-ejemplos-de-aplicaciones-node-js-para-empresas>. [Último acceso: 2018].
- [60] MongoDB, «mongoDB,» MongoDB, 2008. [En línea]. Available: <https://docs.mongodb.com/manual/core/distributed-queries/>. [Último acceso: 2018].
- [61] MongoDB, «mongoDB,» MongoDB, 2008. [En línea]. Available: <https://www.mongodb.com/what-is-mongodb>. [Último acceso: 2018].
- [62] Express.js, «Express,» Node.js, 2017. [En línea]. Available: <http://expressjs.com/>. [Último acceso: 2018].
- [63] Express, «Express,» Node.js, 2017. [En línea]. Available: <http://expressjs.com/en/starter/installing.html>. [Último acceso: 2018].
- [64] J. Wingeter Rodrigues y L. Augusto Yamame, «InfoQ,» C4Media, 17 Mayo 2018. [En línea]. Available: <https://www.infoq.com/articles/angular-application-generator#anch159249>. [Último acceso: 2018].
- [65] Angular Minds, «Angular Minds,» Angular Minds, 29 Junio 2017. [En línea]. Available: <https://www.angularminds.com/blog/article/best-angularjs-features-for-web-application-development.html>. [Último acceso: 2018].
- [66] T. Capan, «Toptal,» Toptal, 2014. [En línea]. Available: <https://www.toptal.com/nodejs/why-the-hell-would-i-use-node-js>. [Último acceso: 2018].
- [67] U. Gascón, «Medium,» Medium, 24 Abril 2017. [En línea]. Available: <https://medium.com/@ulisesGascon/pensar-as%C3%ADncronamente-en-un-mundo-s%C3%ADncrono-8e25cfcafd83>. [Último acceso: 2018].
- [68] Node.js, «nodejs,» Fundación Node.js, 2016. [En línea]. Available: <https://nodejs.org/en/docs/guides/getting-started-guide/>. [Último acceso: 2018].
- [69] M. L. Araque Jimenez, «Gestión de riesgos en proyectos de software a desarrollar en empresa privada,» Universidad Militar nueva Granda, Bogotá, 2015.

IX. ANEXOS

ANEXO N° 01. ESTRUCTURA DE ENTREVISTA A JEFATURA DE LA DIVISIÓN DE TECNOLOGÍAS DE INFORMACIÓN



Facultad de Ingeniería
Escuela de Ingeniería de Sistemas y Computación

ENTREVISTA A JEFATURA DE LA DIVISIÓN DE TECNOLOGÍAS DE INFORMACIÓN.

Nombre: _____

Cargo: _____

Fecha: _____

PREGUNTAS:

1. ¿Qué función o funciones realiza la División de Tecnologías de Información en el Hospital Regional Lambayeque?
2. ¿Con cuántos empleados cuenta la división? ¿Qué funciones tienen?
3. ¿Cómo funciona el mecanismo de trabajo para la asignación de actividades en su división?
4. ¿Realiza algún control de tiempo y carga de trabajo para las actividades de la división?
5. ¿Qué criterios se emplean para medir y evaluar el desempeño?
6. ¿La creación e intercambio de conocimiento en la división siempre se realiza por medios formales?
7. ¿Realiza una medición del éxito del servicio brindado? ¿Cómo lo hace?

8. ¿Cuánto tiempo invierte en tomar una decisión respecto a las acciones de la división?
9. ¿Existen manuales de procedimientos, políticas o lineamientos de desempeño documentados oficial o no oficialmente?
10. ¿La División cuenta con algún mecanismo de retroalimentación?
11. ¿La división realiza o se encuentra proyectos tecnológicos?
12. ¿Qué tipo de proyectos son los que realiza en la división?
13. ¿Sus áreas cuentan con herramientas de apoyo para sus actividades?
14. ¿Cree que es conveniente mejorar el proceso de gestión de conocimientos mediante una herramienta tecnológica?
15. ¿Le gustaría contar con un sistema que gestione la base de conocimiento mediante el control de incidencias para su división?
16. ¿Con qué funcionalidades le gustaría contar en el sistema?
17. ¿Todo su equipo tiene acceso a todos los sistemas de hospital?
18. ¿Todo su equipo participa en el desarrollo de sistemas de esta entidad?
19. ¿El Hospital solicita algún informe de rendimiento de la división? ¿Con qué frecuencia?

**ANEXO N° 02. ESTRUCTURA DE ENTREVISTA AL PERSONAL
RESPONSABLE DEL DESARROLLO Y MONITOREO DEL SOFTWARE EN
LA DIVISIÓN DE TECNOLOGÍAS DE INFORMACIÓN**



Facultad de Ingeniería
Escuela de Ingeniería de Sistemas y Computación

ENTREVISTA AL PERSONAL RESPONSABLE DEL DESARROLLO Y MONITOREO
DEL SOFTWARE EN LA DIVISIÓN DE TECNOLOGÍAS DE INFORMACIÓN.

Nombre: _____

Cargo: _____

Fecha: _____

PREGUNTAS:

1. ¿Qué actividades desempeña usted en la División de Tecnologías de Información?
2. ¿Qué proyectos ha realizado, está planificando y se encuentra realizando?
3. ¿Considera que existe una relación entre la calidad del servicio brindado y sus incidentes presentados?
4. ¿Con qué tipo de información trabaja para el desempeño de sus actividades?
5. ¿Qué eventos o incidentes suelen presentarse durante la realización del servicio?
6. ¿Usted almacena sus eventos y/o incidencias durante el desempeño de sus actividades? ¿Dónde almacena sus conocimientos?
7. ¿Cuánto tiempo le toma en promedio como máximo y como mínimo buscar una solución para la incidencia presentada?

8. ¿Cuántas personas se involucran como máximo y como mínimo en la realización de un proyecto?
9. ¿Ha trabajado en proyectos donde alguno de los integrantes ha dejado el equipo?
10. ¿Ha sido necesario contactarlo(s) para alguna consulta respecto a alguna actividad realizada?
11. ¿Qué acciones realiza si esa persona no se encuentra disponible para ayudarlo en su consulta?
12. ¿Usted utiliza la información de proyectos pasados en nuevos proyectos?
13. ¿Emplea alguna técnica o procedimiento formal para gestionar sus eventos o incidencias? ¿Cuáles son?
14. ¿Ha tenido que recurrir a internet para dar solución a algún incidente presentado?
15. ¿Ha tenido que recurrir a internet para dar solución a algún incidente presentado, cuya solución ya conocía un compañero?
16. ¿Cuánto tiempo le toma en promedio, como máximo y mínimo, realizar sus consultas por internet?
17. ¿Considera que varios de sus proyectos tienen eventos o incidencias similares? ¿Puede mencionar algún ejemplo?
18. ¿El proceso de su actividad se encuentra basado en alguna metodología?
19. ¿Lleva un registro de eventos o incidencias presentadas en sus proyectos y actividades?
20. ¿Qué tipo de registro maneja? ¿Dónde almacena la información?
21. ¿Considera que llevar un registro de eventos o incidencias de sus proyectos le permitiría optimizar el tiempo durante el desarrollo de nuevas actividades?
22. ¿Utiliza alguna herramienta tecnológica para la gestión de sus incidencias de sus actividades o proyectos?

ANEXO N° 03. ESTRUCTURA DE ACTA DE INCIDENCIA DE REFERENCIA

Antecedente	Implementación del marco de trabajo ITIL para apoyar la gestión de los servicios del centro de sistemas de información en la Gerencia Regional de Salud [19]
Objetivo	Controlar y documentar incidencias.
Descripción	Estructura de acta de incidencias que se tomará a los atributos como referentes para el diseño de interfaz(es) de gestión de incidencias.

ACTA DE INCIDENCIAS N° _____

DATOS GENERALES:

• **Del Área de Atención:**

- Unidad Orgánica: _____

- Resp. Patrimonial: _____

- Resp. Funcional: _____

• **Del Área de Soporte Técnico:**

- Resp. De Soporte: _____

- Personal de Apoyo: _____

- Fecha: ____/____/20__

N°PC: _____ COD. PATRIMONIAL: _____ SIST. OP.: _____

DESCRIPCION DE SITUACION:

• **CPU** N° De Inventario: _____

Falla de S.O. Averías del Disco Duro Falla en Microprocesador

Problemas de Configuración Error de Software

Error de Conectividad Averías de Tarjeta de Red Fallas de Mainboard

• **MONITOR** N° De Inventario: _____

Falla Interna Error de Resolución

Averías de Botones Error de Conectividad

• **TECLADO Y/O MOUSE** N° De Inventario: _____

Error de Conectividad Avería Interna Desconfiguración

• **IMPRESORA** N° De Inventario: _____

Error de Conectividad Problema de Cola Desconfiguración de Impresión

Avería Interna

• **OTRO DISPOSITIVO:** _____ N° De Inventario: _____

• **OTRO MOTIVO DE ATENCIÓN:** _____

ESTADO

Concluido y operativo el dispositivo.

En espera de repuestos, para reparación.

Malogrado sin posibilidad de reparación.

TIPO DE ATENCIÓN

Solución.

Traslado a Soporte Técnico para revisión técnica.

ACCIONES REALIZADAS:

• **SOFTWARE:**

✓ Formateo de Disco Duro.....() ✓ Instalación de Antivirus Actualizable.()

✓ Instalación de programas Básicos.....() ✓ Tránsferencia de datos.....()

✓ Asignación de IP/grupo de trabajo/nombre de equipo.....()

✓ Contraseña de acceso al equipo.....Aprobación del CSI Si () No ()

✓ Instalación de los controladores de la Mainboard: Video () Red () Audio () Impresora ()

• **HARDWARE:**(Cambio de dispositivos)

✓ Pila de BIOS () Disco duro () ✓ Memoria RAM () Red ()

✓ Fuente () Video () ✓ Repotenciación de los Bancos de Memoria ()

SUGERENCIAS/RECOMENDACIONES/OBSERVACIONES

Limpieza externa del equipo de computo.....

.....

FIRMA DE CONFORMIDAD DEL USUARIO TECNICO/CSI

RESPONSABLE SOPORTE

**ANEXO N° 04. DEFINICIÓN DE PRIORIDADES ASIGNADAS A
INCIDENCIAS**

OBJETIVO	Definir la prioridad de una incidencia para su adecuada atención.										
DESCRIPCIÓN	Se busca priorizar adecuadamente las incidencias para su resolución de acuerdo a este criterio.										
HERRAMIENTA	Marco de trabajo ITIL V3.0.										
EVIDENCIA	<p align="center">$P = I \times U$</p> <p>Dónde:</p> <p align="center">P: Prioridad I: Impacto U: Urgencia</p> <p>Urgencia de atención: Área involucrada del proyecto Impacto de Incidencia: Bloqueo Crítico, Retraso Crítico, Retraso no crítico, Molestia</p>										
CONCLUSIÓN	<p>Se establece el siguiente rango para la clasificación de prioridades:</p> <table align="center"> <tr> <td></td> <td>Área involucrada</td> </tr> <tr> <td>Bloqueo Crítico</td> <td>Alta</td> </tr> <tr> <td>Retraso Crítico</td> <td>Media</td> </tr> <tr> <td>Retraso No Crítico</td> <td>Baja</td> </tr> <tr> <td>Molestia</td> <td>Baja</td> </tr> </table>		Área involucrada	Bloqueo Crítico	Alta	Retraso Crítico	Media	Retraso No Crítico	Baja	Molestia	Baja
	Área involucrada										
Bloqueo Crítico	Alta										
Retraso Crítico	Media										
Retraso No Crítico	Baja										
Molestia	Baja										

ANEXO N°05. ESTRUCTURA DE ENTREVISTA DE PRUEBA DE SOFTWARE



Facultad de Ingeniería
Escuela de Ingeniería de Sistemas y Computación

ENCUESTA AL PERSONAL RESPONSABLE DEL DESARROLLO Y MONITOREO DEL SOFTWARE EN LA DIVISIÓN DE TECNOLOGÍAS DE INFORMACIÓN.

Nombre: _____

Cargo: _____

Fecha: _____

Cuestionario:

INTERROGANTES	SI	NO
¿Usted se registró como usuario en el software SGID?	<input type="checkbox"/>	<input type="checkbox"/>
¿Usted accedió a la web del software SGID?	<input type="checkbox"/>	<input type="checkbox"/>
¿Usted descargó el plugin desde el software SGID?	<input type="checkbox"/>	<input type="checkbox"/>
¿Usted instaló el plugin del software SGID en NetBeans?	<input type="checkbox"/>	<input type="checkbox"/>
¿Usted accedió al plugin del software SGID?	<input type="checkbox"/>	<input type="checkbox"/>
¿Usted registró proyectos en el software SGID?	<input type="checkbox"/>	<input type="checkbox"/>
¿Usted registró incidencias en el software SGID?	<input type="checkbox"/>	<input type="checkbox"/>
¿Usted registró métodos en el software SGID?	<input type="checkbox"/>	<input type="checkbox"/>
¿Usted registró consultas en el software SGID?	<input type="checkbox"/>	<input type="checkbox"/>
¿Usted agregó integrantes por proyecto?	<input type="checkbox"/>	<input type="checkbox"/>
¿Usted registró soluciones en el software SGID?	<input type="checkbox"/>	<input type="checkbox"/>

ANEXO N°06. ESTRUCTURA DE ENTREVISTA DE APROBACIÓN DE SOFTWARE



Facultad de Ingeniería
Escuela de Ingeniería de Sistemas y Computación

ENTREVISTA AL PERSONAL RESPONSABLE DEL DESARROLLO Y MONITOREO DEL SOFTWARE EN LA DIVISIÓN DE TECNOLOGÍAS DE INFORMACIÓN.

Nombre: _____

Cargo: _____

Fecha: _____

Cuestionario:

INTERROGANTES	SI	NO
¿Considera usted que el software SGID redujo a uno las fuentes de búsqueda de solución de incidencias?	<input type="checkbox"/>	<input type="checkbox"/>
¿Considera usted que el software SGID brinda una interfaz donde se pueda realizar búsquedas de incidencias?	<input type="checkbox"/>	<input type="checkbox"/>
¿Considera usted que el software SGID brinda una interfaz donde se pueda visualizar el título, descripción y tipo de incidencia?	<input type="checkbox"/>	<input type="checkbox"/>
¿Considera usted que el software SGID brinda una interfaz donde se pueda realizar registros de incidencias (incidencia, método y consulta)?	<input type="checkbox"/>	<input type="checkbox"/>
¿Considera usted que el software SGID brinda una interfaz donde se pueda realizar registros de solución de cada incidencia?	<input type="checkbox"/>	<input type="checkbox"/>
¿Considera usted que el software SGID brinda una interfaz donde se pueda guardar la solución de cada incidencia?	<input type="checkbox"/>	<input type="checkbox"/>
¿Considera usted que el software SGID brinda una interfaz donde se pueda realizar registros de incidencia por cada sistema?	<input type="checkbox"/>	<input type="checkbox"/>

MANUAL

USUARIO



SISTEMA DE GESTIÓN DE INCIDENCIAS

El Sistema de Gestión de Incidencias de Desarrollo (SGID) es una plataforma colaborativa para contribuir al proceso de desarrollo de software, se centra en la gestión de métodos y consultas frecuentes y en la gestión de incidencias por proyecto al que pertenece.

El SGID es un entorno que brinda herramientas para gestión de proyectos, gestión de incidencias, gestión de integrantes y acceso a la descarga de plugin. Es así como se presenta este manual que facilite el uso de esta plataforma colaborativa mediante la web y desde el IDE de NetBeans.



DESCRIPCIÓN DE LA WEB

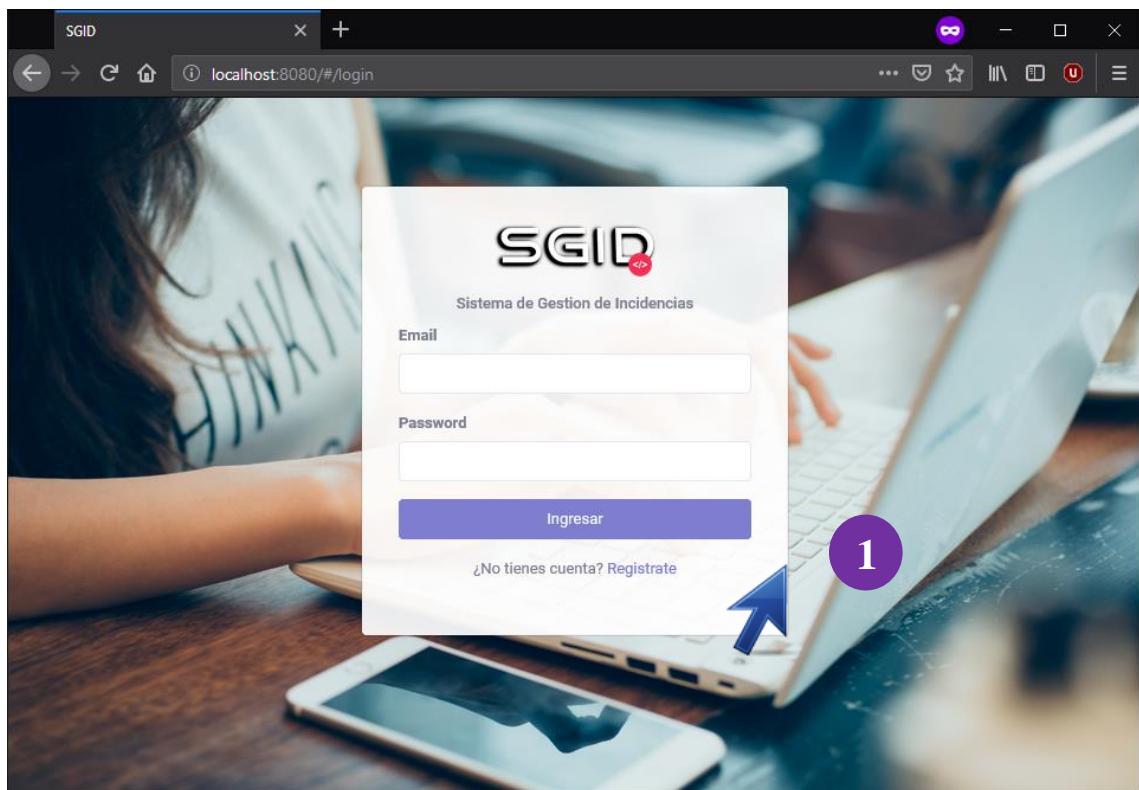
Para hacer uso de los servicios del sistema de Gestión de incidencias de Desarrollo y descargar el plugin de complemento para NetBeans tendrá que acceder inicialmente a la web.

Registrar Usuarios

Descripción: Con esta actividad el usuario aprenderá a registrarse.

Para registrarse en el SGID, seguir los siguientes pasos:

Ingresar al registro



Ingresar los datos solicitados

The image shows a registration form for the SGID (Sistema de Gestion de Incidencias). The form is titled '2 SGID Sistema de Gestion de Incidencias'. It contains the following fields and callouts:

- Ingresar su**: Callout for the 'Nombre' field.
- Ingresar su DNI**: Callout for the 'DNI' field.
- Ingresar su correo**: Callout for the 'Email' field.
- Ingresar sus**: Callout for the 'Apellido Paterno' and 'Apellido Materno' fields.
- Seleccionar su**: Callout for the 'Cargo' dropdown menu.
- 3**: A purple circle with the number 3 is located near the bottom right of the form.

At the bottom of the form, there is a blue button labeled 'Registrarse' and a link that says '¿No tienes una cuenta? Accede'.

Ingresar al sistema

Descripción: Con esta actividad el usuario aprenderá acceder al sistema. Para acceder al SGID, seguir los siguientes pasos:

The image shows a login form for the SGID (Sistema de Gestion de Incidencias). The form is titled '1 SGID Sistema de Gestion de Incidencias'. It contains the following fields and callouts:

- Ingresar su correo**: Callout for the 'Email' field.
- Ingresar su**: Callout for the 'Password' field.
- 2**: A purple circle with the number 2 is located near the bottom right of the form.

At the bottom of the form, there is a blue button labeled 'Ingresar' and a link that says '¿No tienes una cuenta? Regístrate'.

Entorno web

Descripción: Con esta actividad el usuario conocerá el entorno del sistema y las opciones que ofrece.

El entorno web ofrece las siguientes características:



Entorno de Proyecto

Descripción: Con esta actividad el usuario conocerá el entorno de Proyectos y las opciones que ofrece.

El entorno de proyectos ofrece las siguientes características:



Registro de Proyecto

Descripción: Con esta actividad el usuario aprenderá a registrar un proyecto. Para registrar el proyecto, seguir los siguientes pasos:

Seleccionar el botón de Nuevo Proyecto



Ingresar datos solicitados

The image shows a web form titled 'Añadir proyecto' (Add project) under the heading 'Gestión de proyectos'. The form contains several input fields: 'Título:' (Title), 'Descripción:' (Description), 'Áreas Involucradas:' (Involved Areas), and 'Rol:' (Role). The 'Rol:' field has a dropdown menu with the text 'Seleccionar un Rol' and a red error message below it: 'Seleccionar el rol que desempeñará en el proyecto'. At the bottom of the form, there are two buttons: 'Cerrar' (Close) and 'Guardar' (Save). A blue arrow points to the 'Guardar' button. The form is annotated with callouts: a purple circle with the number '1' is at the top left; a callout box says 'Ingresar título del' pointing to the title field; another callout box says 'Ingresar descripción del' pointing to the description field; a third callout box says 'Ingresar áreas involucradas' pointing to the areas field; a fourth callout box says 'Selección Rol*' pointing to the role dropdown; a fifth callout box says 'Permiso automático de asignación' pointing to the role field; and a purple circle with the number '2' is at the bottom right, near the 'Guardar' button.

* Los roles son creados con anticipación dependiendo del personal del área.

Detalle de Proyecto

Descripción: Con esta actividad el usuario aprenderá acceder al detalle del proyecto. Para acceder al proyecto, seguir los siguientes pasos:

Seleccionar le proyecto



Características del entorno de detalle de proyecto

Detalle proyecto

Nombre de proyecto: Proyecto ejemplo

Área involucrada: DIT?

Creador del proyecto: Creado por: Guerrero Gutierrez

Descripción del proyecto: Descripción del proyecto ejemplo

Opción de inhabilitar proyecto: Eliminar

Opción de editar proyecto: Editar

Añadir integrantes al proyecto: INTEGRANTES

Integrantes: Guerrero Gutierrez Carol - Jefe Proy, Analista - Puede asignar

#	Nombre	Descripcion
#	Incidencia	Descripción de incidencia
#	Método	Descripción de método
#	Consulta	Descripción de consulta

Listado de incidencias

Version 1

Editar proyecto

1 Añadir proyecto

Gestion de proyectos

Título: Proyecto ejemplo

Editar título de

Descripción: Descripción del proyecto ejemplo

Editar descripción de proyecto

Áreas involucradas: DITI

Editar áreas involucradas

Cerrar

Guardar **2**

Entorno de Incidencias

Descripción: Con esta actividad el usuario conocerá el entorno de Incidencias y las opciones que ofrece.

El entorno de incidencias ofrece las siguientes características:

The screenshot shows the 'Incidencias' application interface. A dark blue sidebar on the left contains a 'MENU' with options: 'Proyectos', 'Incidencias', 'Integrantes', 'Roles', and 'Descargas'. The main content area has a search bar labeled 'Buscador de' and a filter dropdown labeled 'Filtro de incidencia por'. A table lists incidents with columns: '#', 'Titulo', 'Descripcion', 'Tipo', 'Fecha Creacion', and 'Acciones'. A callout 'Añadir incidencia' points to a '+ Incidencia' button. Another callout 'Acciones por' points to the 'Acciones' column. A callout 'Listado de' points to the table header. The footer shows '2018 © Carol Janet Guerrero Gutierrez' and 'Version 1'.

Registro de Incidencia

Descripción: Con esta actividad el usuario aprenderá a registrar una incidencia. Para registrar la incidencia, seguir los siguientes pasos:

Seleccionar el botón de Incidencia



Ingresar datos solicitados

The screenshot shows a web form titled 'Añadir incidencia' with a purple header. A large purple circle with the number '1' is in the top left. The form contains several fields: 'Proyecto ejemplo' (Project), 'Nombre automático de' (Automatic name), 'Ingresar título' (Title), 'Ingresar' (Description), 'Ingresar imagen de referencia' (Reference image), 'Solución' (Solution), 'Sin asignación' (Assignment), 'Seleccionar' (Assign to), 'Seleccionar' (Priority), and 'Guardar' (Save). A blue arrow points to the 'Guardar' button, which is next to a purple circle with the number '2'. Callouts are purple boxes with white text and arrows pointing to the respective fields.

Detalle de Incidencia

Descripción: Con esta actividad el usuario aprenderá acceder al detalle de la incidencia. Para acceder a la incidencia, seguir los siguientes pasos:

Seleccionar el botón de detalle de incidencia

#	Título	Descripción	Tipo	Fecha Creacion	Acciones
#	Incidencia	Descripción de incidencia	Incidencia	03-12-2018	

Entorno de detalle de incidencia

The screenshot shows a web interface for incident details. At the top, there's a header with 'Incidencia' and 'Registrado por: Guerrero Gutierrez Carol'. Below this is a 'Descripción de incidencia' section. A teal bar labeled 'Detalles' contains a table with the following data:

Proyecto	Proyecto ejemplo	Estado	Sin resolver	Asignación	Sin asignacion
Tipo:	Incidencia	Prioridad	Baja		

Below the table is a 'Codigo' section with a dark background for code. At the bottom, there's a blue bar with a gear icon and the text 'solucion', and a button labeled 'añadir'.

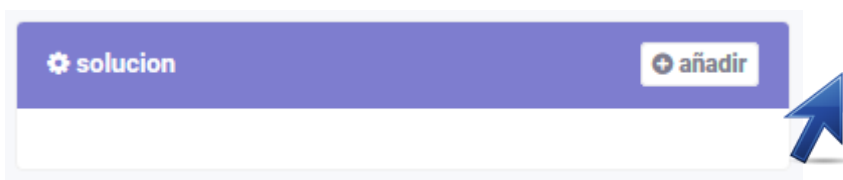
Callouts in the image point to the following elements:

- Usuario que realizó el
- Título
- Descripción
- Sección de detalles
- Líneas de código referenciales
- Sección de
- Añadir solución

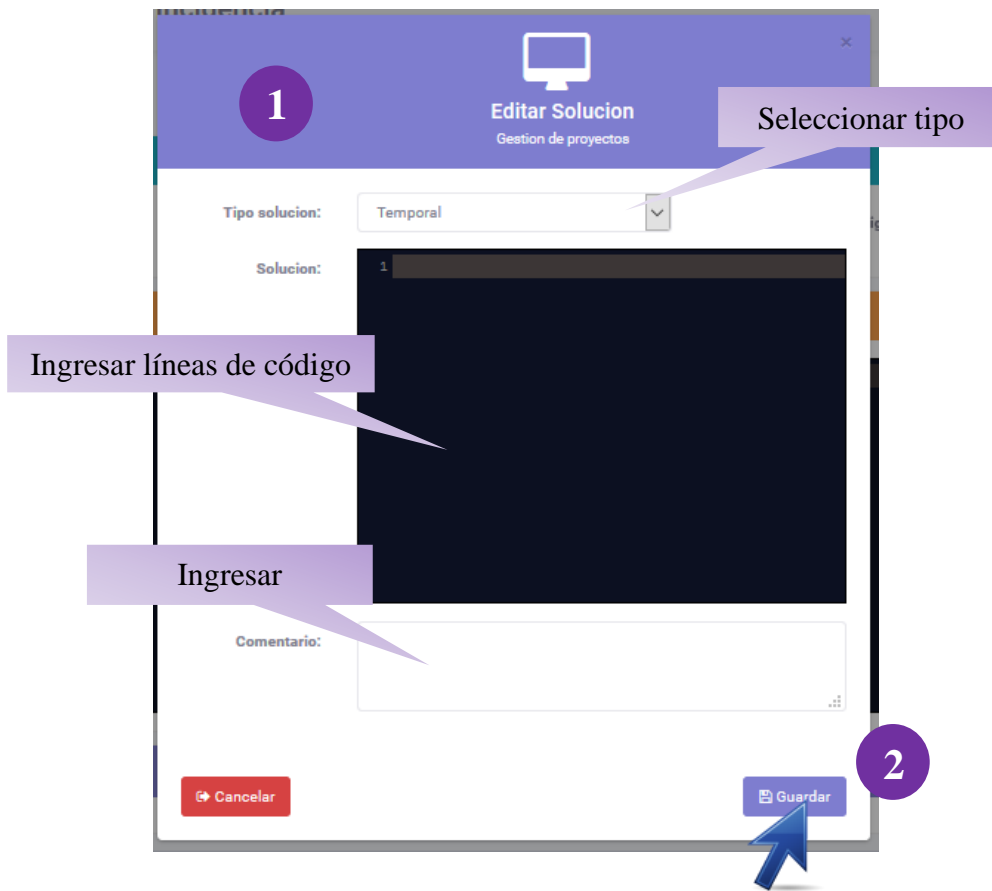
Registro de Solución

Descripción: Con esta actividad el usuario aprenderá a registra una solución. Para registrar la solución, seguir los siguientes pasos:

Seleccionar botón de añadir




Ingresar datos solicitados



Editar Incidencia

Descripción: Con esta actividad el usuario aprenderá a editar una incidencia. Para registrar el proyecto, seguir los siguientes pasos:

Seleccionar el botón de editar Incidencia

#	Titulo	Descripcion	Tipo	Fecha Creacion	Acciones
#	Incidencia	Descripción de incidencia	Incidencia	03-12-2018	

Ingresar datos solicitados

The image shows a web form titled "Editar incidencia" (Edit incident) under the heading "Gestion de proyectos" (Project Management). The form is annotated with callouts and numbered steps:

- 1** (Top left): A purple circle containing the number 1.
- Editar título** (Top left): Callout pointing to the "Titulo:" text input field.
- Editar descripción** (Top left): Callout pointing to the "Descripción:" text area.
- Cambiar imagen de referencia** (Top left): Callout pointing to the "Imagen:" field, which includes a "Examinar..." button and the text "No se ha seleccionado ningún archivo."
- Editar solución** (Middle left): Callout pointing to the "Solucion:" text area.
- Editar asignación** (Middle left): Callout pointing to the "Asignar a:" dropdown menu.
- Cambiar integrante** (Middle right): Callout pointing to the "Asignar a:" dropdown menu.
- Cambiar prioridad** (Bottom center): Callout pointing to the "Prioridad:" radio buttons (Baja, Media, Alta).
- 2** (Bottom right): A purple circle containing the number 2, with a blue arrow pointing to the "Guardar" (Save) button.

The form fields and controls include:

- Projecto: Proyecto ejemplo
- Titulo: Incidencia
- Descripción: Descripción de incidencia
- Tipo: Incidencia, Metodo, Consulta
- Imagen: Examinar... No se ha seleccionado ningún archivo.
- Solucion: [Empty text area]
- Sin asignacion: Sin asignar integrante
- Asignar a: [Dropdown menu]
- Prioridad: Baja, Media, Alta
- Buttons: Cancelar (red), Guardar (blue)

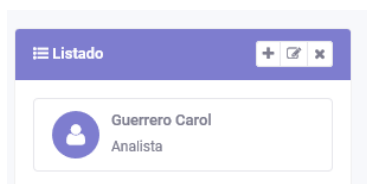
Entorno de Integrantes

Descripción: Con esta actividad el usuario conocerá el entorno de Integrantes y las opciones que ofrece.

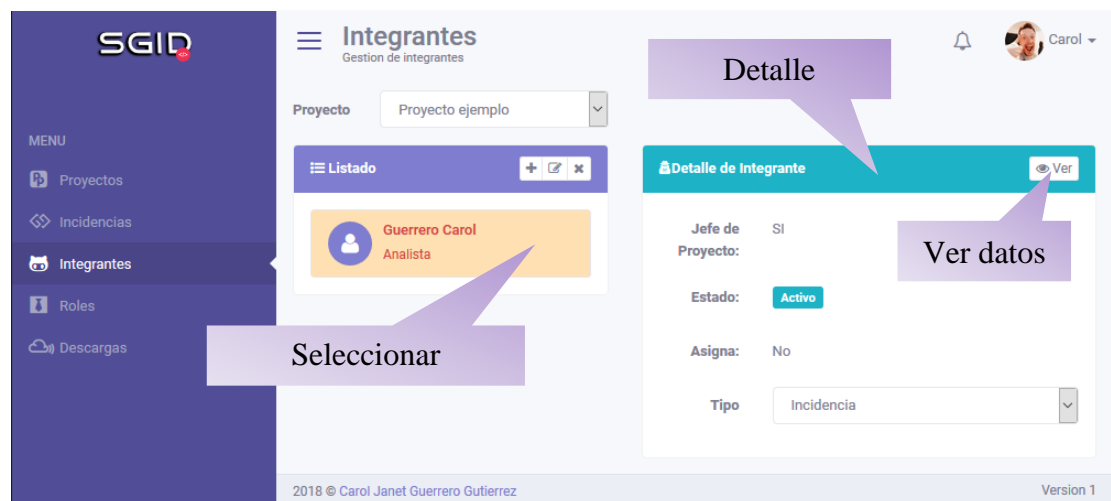
El entorno de integrantes ofrece las siguientes características:



Seleccionar integrante



Detalle de integrante

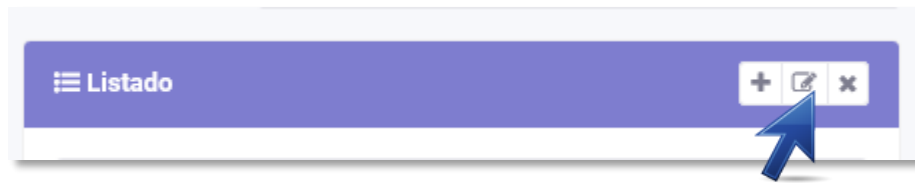


Agregar Integrantes

Descripción: Con esta actividad el usuario aprenderá a agregar Integrantes en su proyecto.

Para agregar al integrante, seguir los siguientes pasos:

Seleccionar botón de agregar



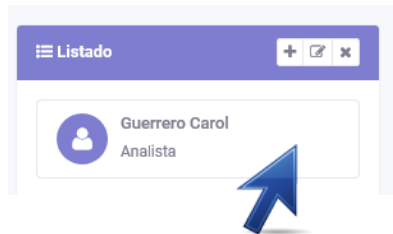
Ingresar datos solicitados



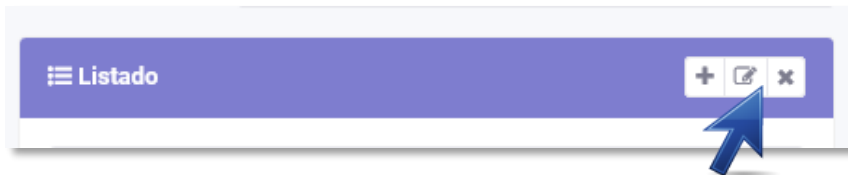
Editar Integrante

Descripción: Con esta actividad el usuario aprenderá a editar un integrante. Para editar al integrante, seguir los siguientes pasos:

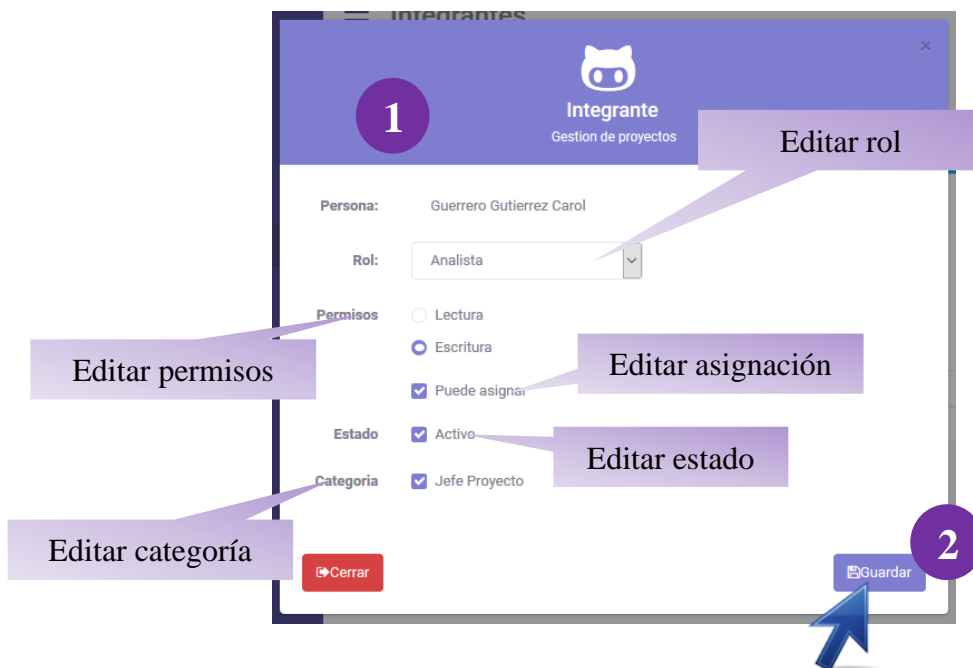
Seleccionar integrante



Seleccionar el botón de editar



Editar integrante



Entorno de Roles

Descripción: Con esta actividad el usuario conocerá el entorno de Roles y las opciones que ofrece.

El entorno de roles ofrece las siguientes características:

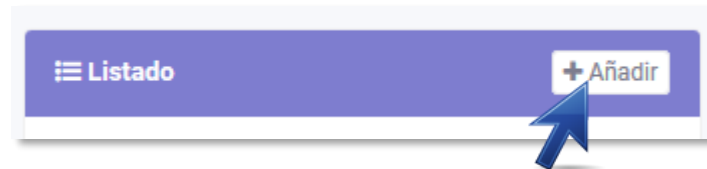
The screenshot shows the 'Roles' management interface in SGID. The interface includes a sidebar menu with options like 'Proyectos', 'Incidencias', 'Integrantes', 'Roles', and 'Descargas'. The main content area displays a table of roles with columns for 'Rol', 'Estado', and 'Acción'. Callouts highlight specific features: 'Añadir rol' points to the '+ Añadir' button, 'Editar rol' points to the edit icon in the 'Acción' column, 'Listado' points to the table header, and 'Estado' points to the 'Activo' status in the 'Estado' column.

Rol	Estado	Acción
Analista	Activo	[Edit]
Desarrollador	Activo	[Edit]
Arquitecto	Activo	[Edit]
Diseñador	Activo	[Edit]
Tester	Activo	[Edit]
Especialista de Calidad	Activo	[Edit]
Administrador de Base de Datos	Activo	[Edit]

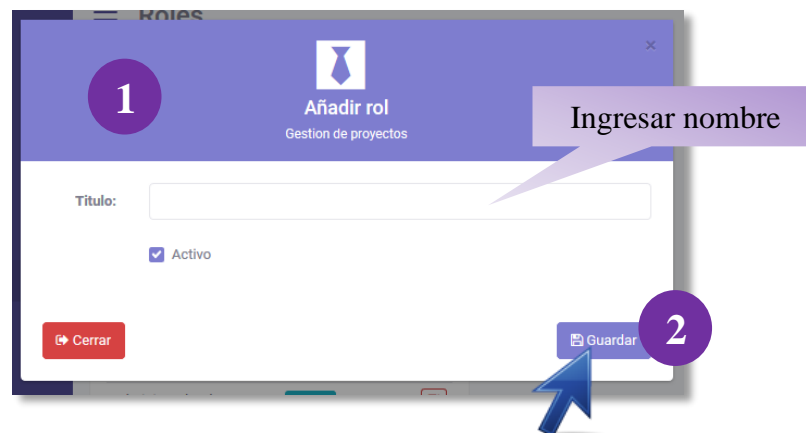
Agregar Roles

Descripción: Con esta actividad el usuario aprenderá a agregar Roles en su proyecto. Para agregar al rol, seguir los siguientes pasos:

Seleccionar botón añadir



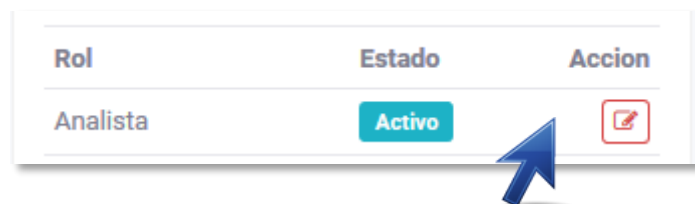
Ingresar datos solicitados



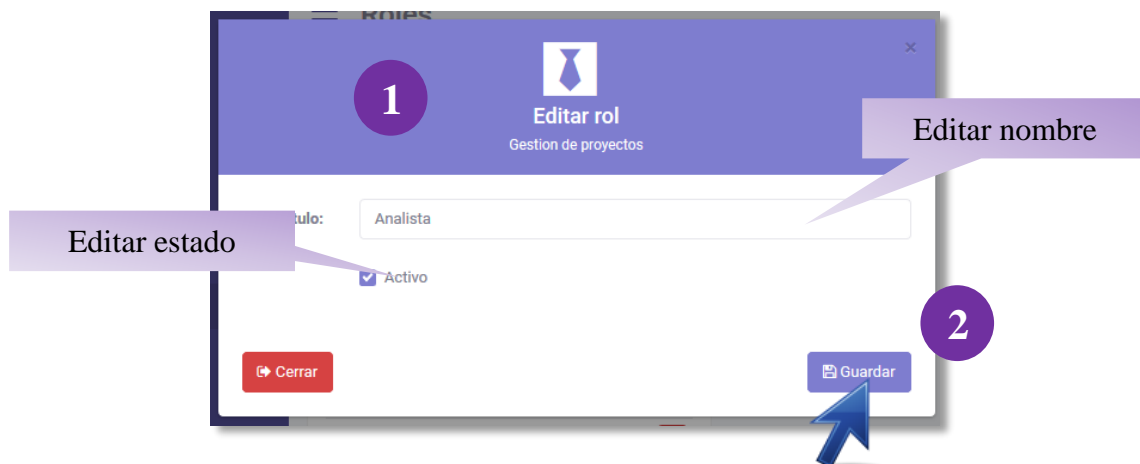
Editar Rol

Descripción: Con esta actividad el usuario aprenderá a editar un rol. Para editar el rol, seguir los siguientes pasos:

Seleccionar botón de editar



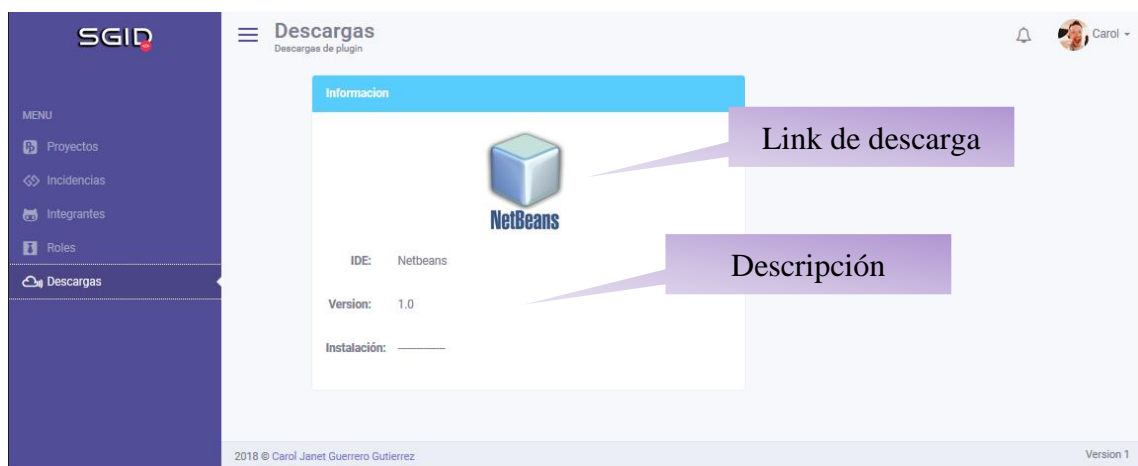
Editar rol



Entorno de Descarga

Descripción: Con esta actividad el usuario conocerá el entorno de Descargas y las opciones que ofrece.

El entorno de descarga ofrece las siguientes características:



DESCRIPCIÓN DEL PLUGIN

Para hacer uso del repositorio de los servicios del sistema de Gestión de incidencias de Desarrollo se puede descargar el plugin desde la web e instalarlo en NetBeans.

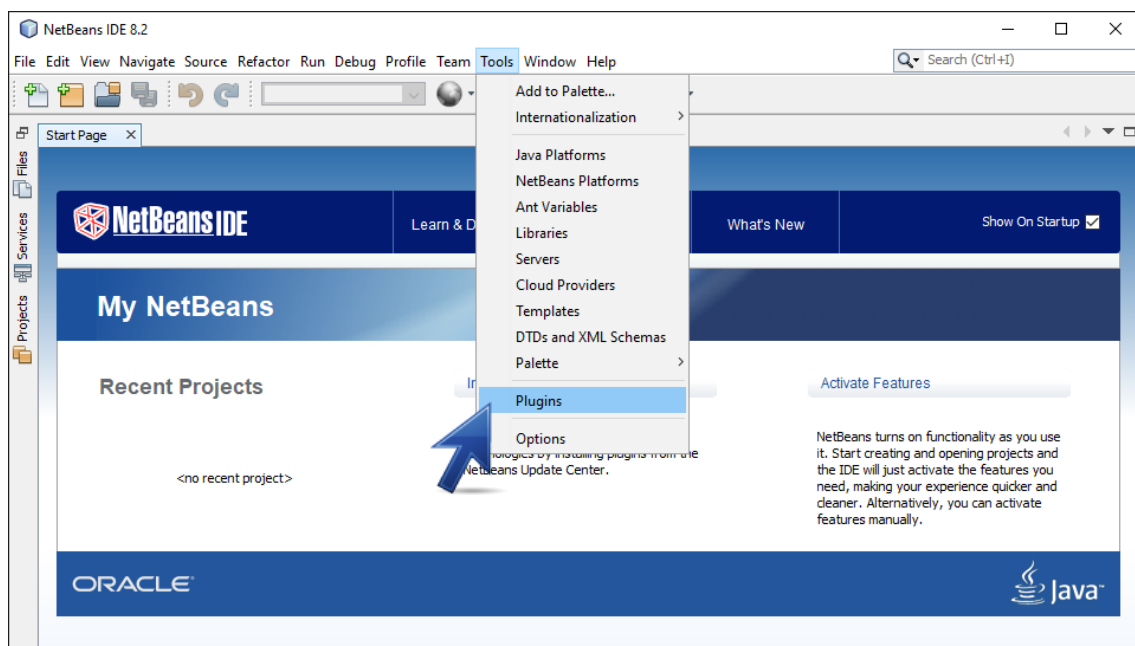
Instalar Plugin

Descripción: Con esta actividad el usuario aprenderá instalar el plugin SGID. Para instalar el plugin, seguir los siguientes pasos:

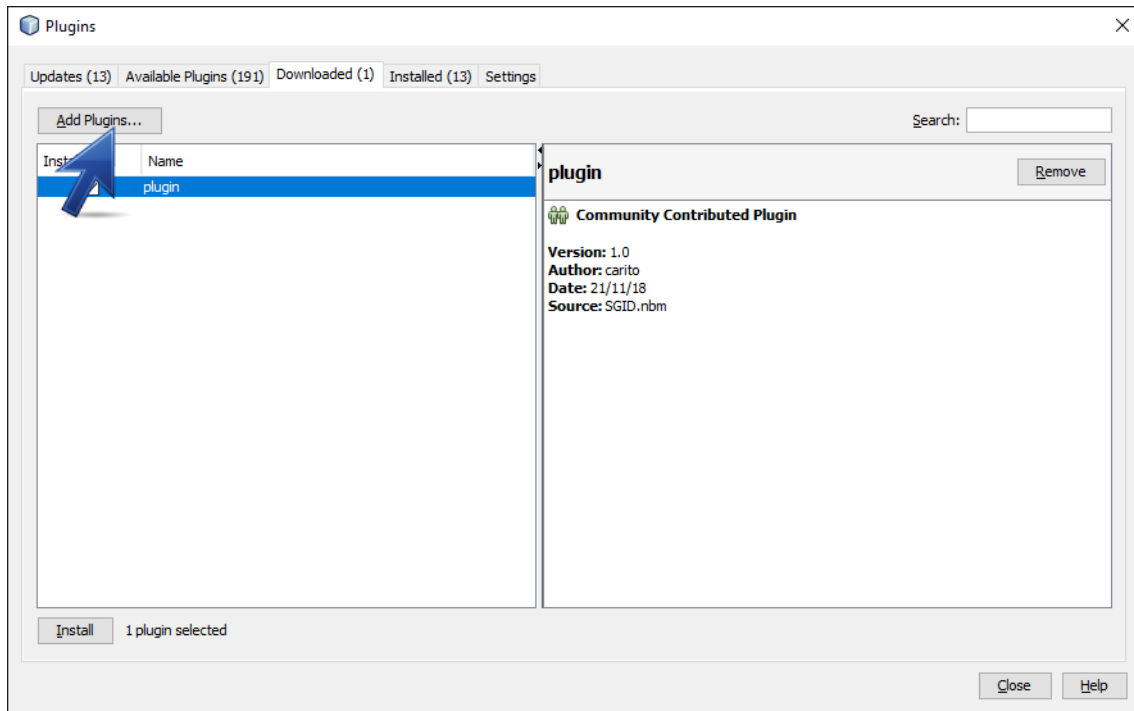
Descargar el plugin



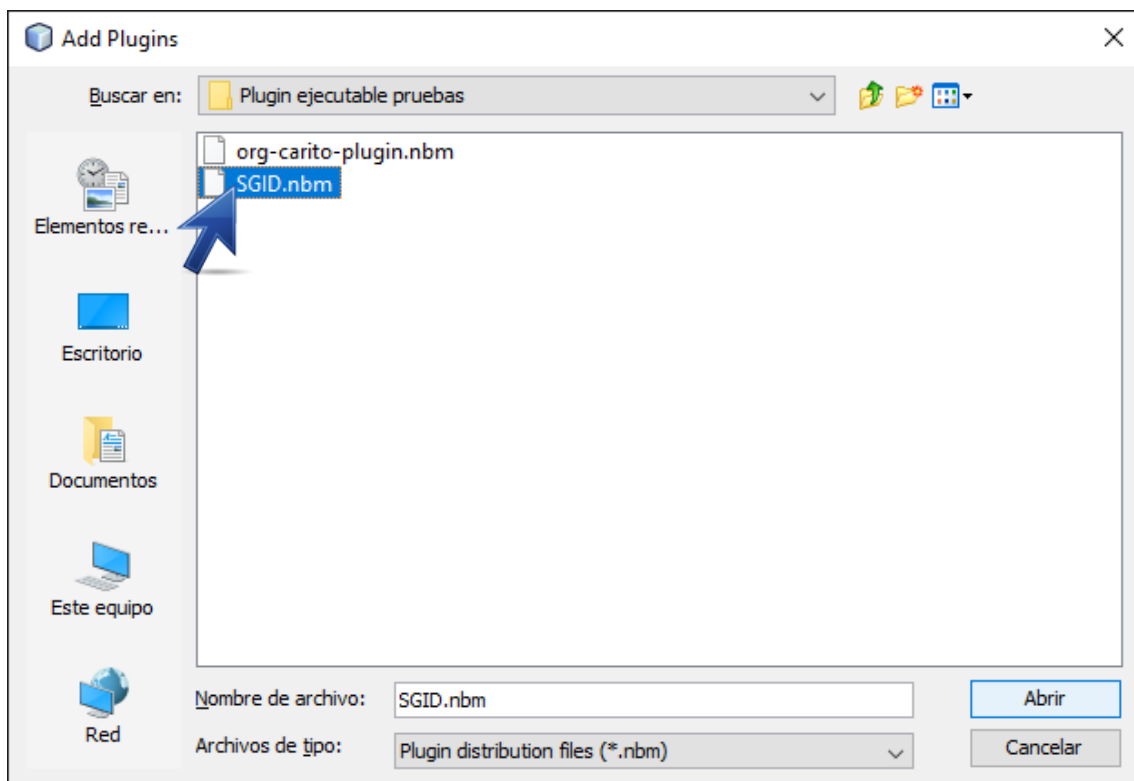
Seleccionar a herramientas en NetBeans y seleccionar la opción plugin



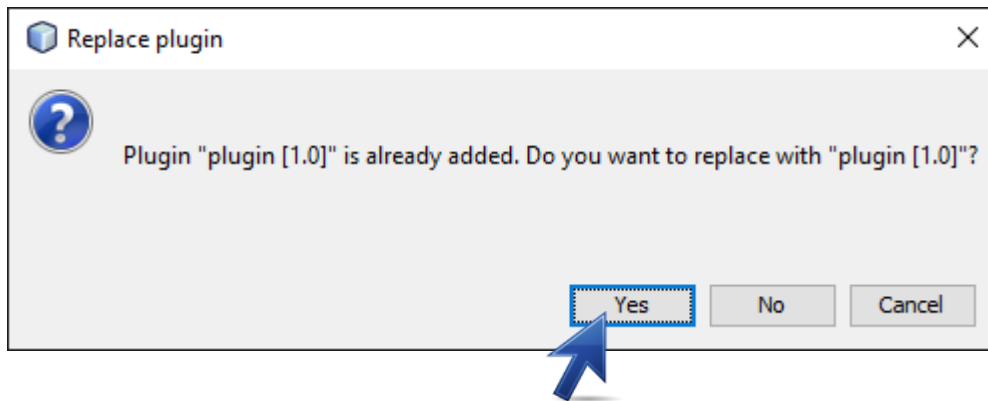
En la interfaz de plugin seleccionar agregar plugin



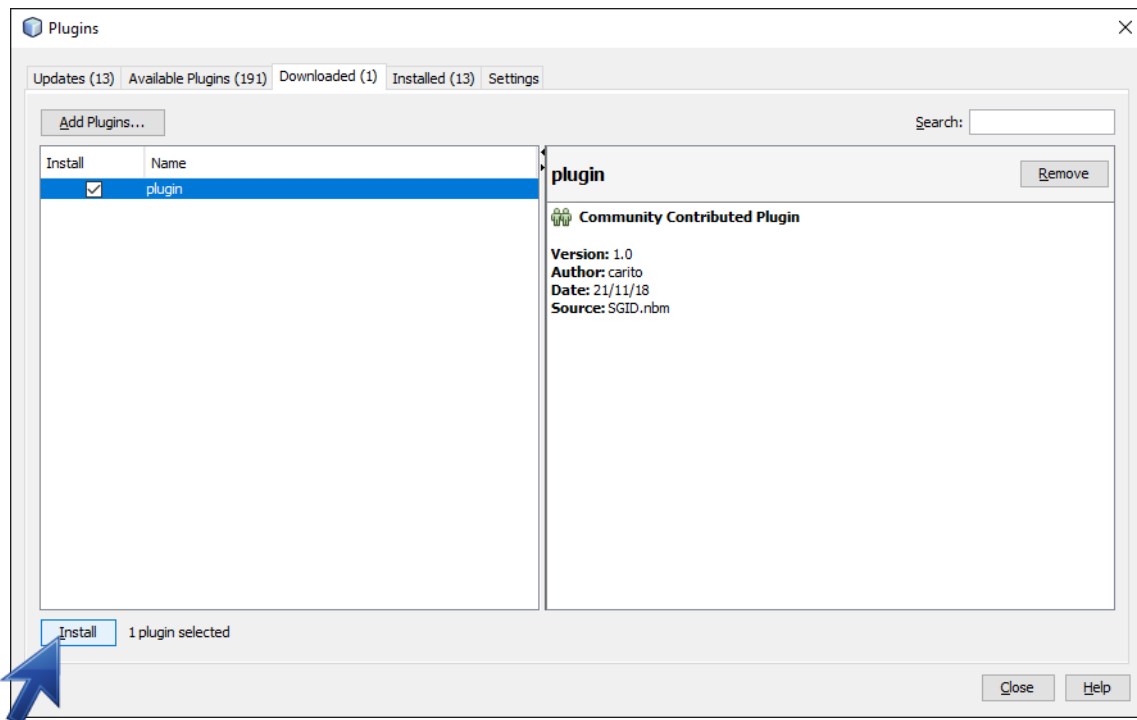
Seleccionar el ejecutable con extensión “. nbm”



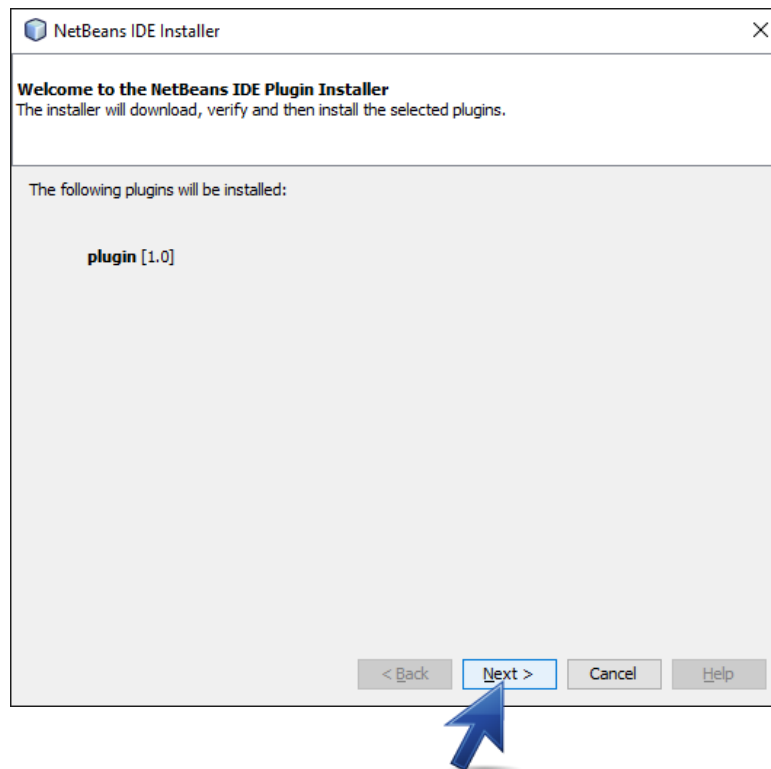
Aceptar



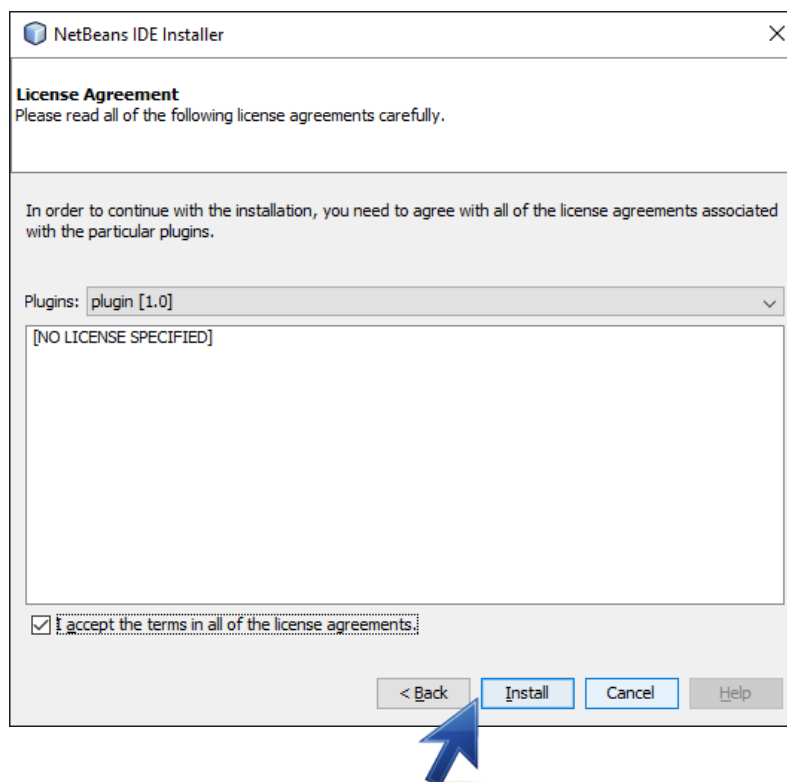
Instalar



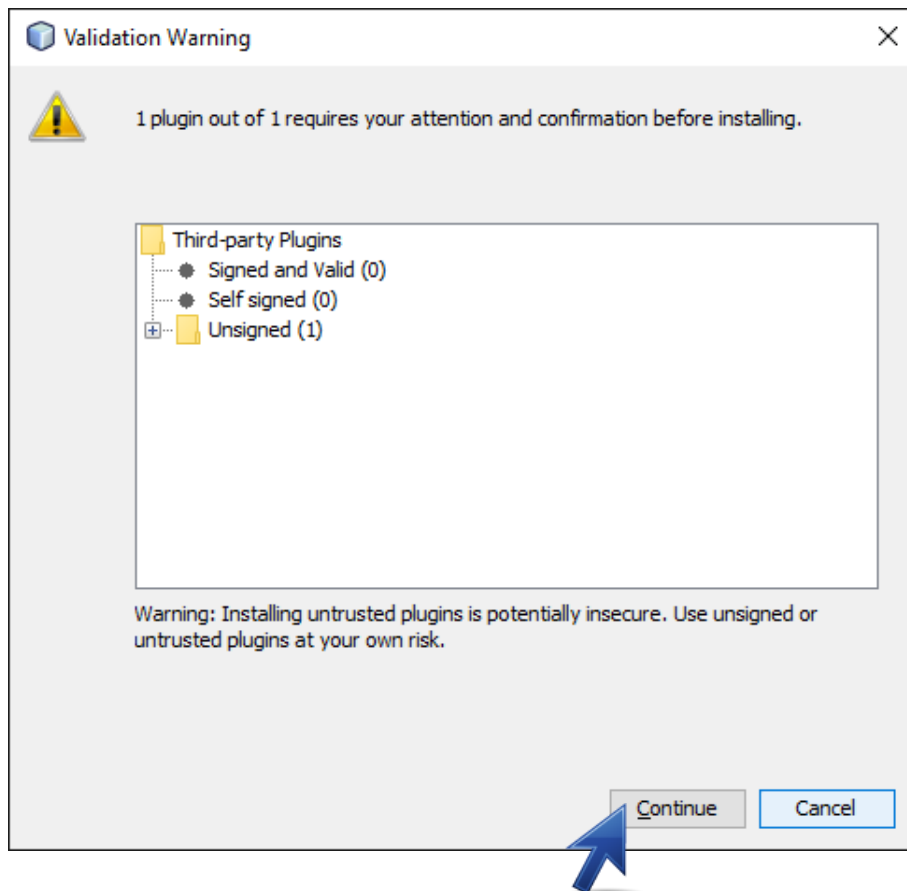
Seleccionar siguiente



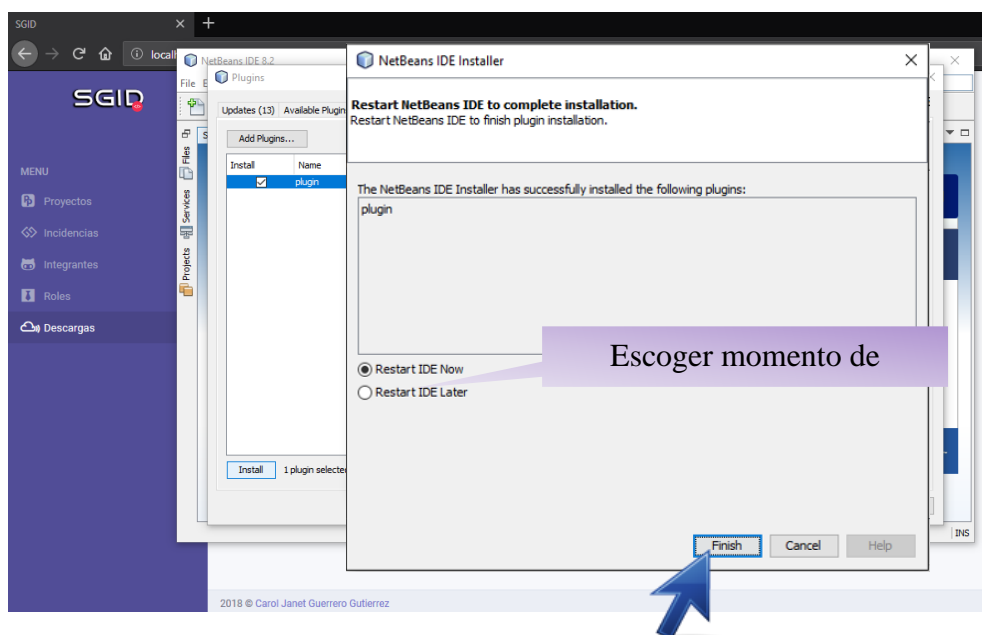
Aceptar los términos



Continuar

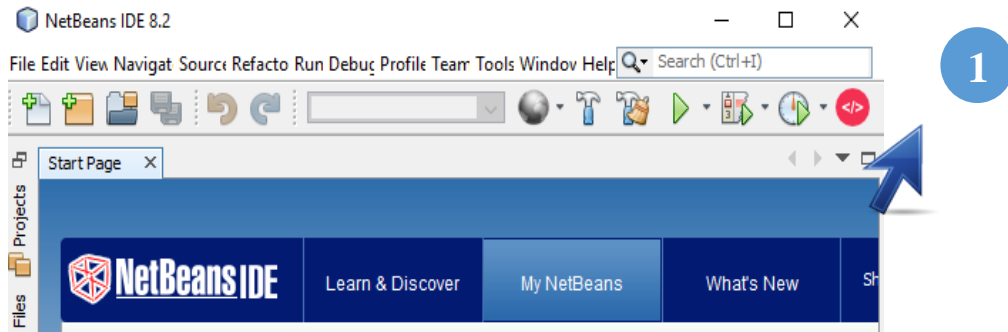


Reiniciar el IDE



Ingresar al Plugin

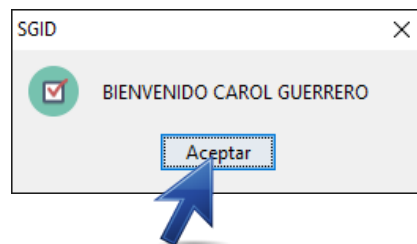
Descripción: Con esta actividad el usuario aprenderá a ingresar al plugin. Para acceder al SGID, seguir los siguientes pasos:
Seleccionar el ícono de la barra de herramientas de NetBeans



Ingresar su correo y contraseña



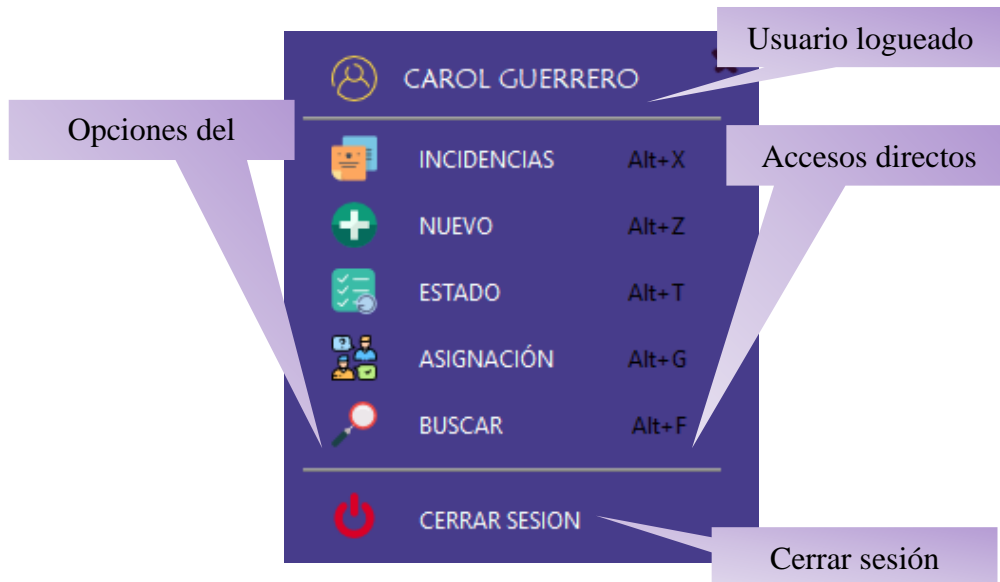
Aceptar



Menú principal

Descripción: Con esta actividad el usuario conocerá el menú principal y las opciones que ofrece.

El menú principal ofrece las siguientes características:



Descripción de opciones

Incidencias: Su selección permite visualizar el entorno de incidencias.

Nuevo: Su selección permite registrar nuevas incidencias.

Estado: Su selección permite visualizar el estado de las incidencias.

Asignación: Su selección permite visualizar las asignaciones realizadas a las incidencias.

Buscar: Su selección permite realizar búsquedas en el repositorio.






Cerrar sesión: Su selección permite salir del plugin.

Entorno de Incidencias

Descripción: Con esta actividad el usuario conocerá el entorno de incidencias y las opciones que ofrece.

El entorno de incidencias ofrece las siguientes características:

The screenshot shows a web interface titled "LISTAR INCIDENCIAS". At the top left, there is a dropdown menu labeled "Proyecto ejemplo" with a callout "Filtro por proyecto". At the top right, there is a button labeled "Añadir" with a plus icon and a callout "Añadir". Below the header is a table with the following columns: "INCIDENCIA", "DESCRIPCIÓN", "TIPO", "FECHA", and "Acciones**". The table contains three rows: "Incidencia", "Método", and "Consulta". The "Acciones**" column contains icons for viewing, editing, and deleting. A callout "Listado" points to the table area, and another callout "Acciones**" points to the action icons.

INCIDENCIA	DESCRIPCIÓN	TIPO	FECHA	Acciones**
Incidencia	Descripción de incidencia	Incidencia	03-12-2018	  
Método	Descripción de método	Método	03-12-2018	
Consulta	Descripción de consulta	Consulta	03-12-2018	

* Solo se muestra si tiene permisos de edición

** Las incidencias de tipo método y consulta no pueden ser editados ni eliminados desde el plugin, contrario a las de tipo incidencias.

Detalle de Incidencias

Descripción: Con esta actividad el usuario conocerá el detalle de la incidencia y las opciones que ofrece.

El entorno de detalle de incidencia ofrece las siguientes características:



Registro de Solución

Descripción: Con esta actividad el usuario aprenderá a registrar una solución. Para registrar la solución, seguir los siguientes pasos:

Seleccionar botón de añadir



Ingresar datos solicitados

Una captura de pantalla de una aplicación móvil con el título "SOLUCIONAR INCIDENCIA". El formulario tiene un fondo azul oscuro y un área blanca para el contenido. En la parte superior izquierda hay un botón de retroceso y el texto "Incidencia". A la derecha hay un círculo azul con el número "1". Debajo de eso, el texto "Solución" precede a un área de entrada de texto grande y vacía. Una etiqueta "Editar solución" apunta a esta área. Abajo de eso, el texto "Tipo" precede a un botón "Temporal" con un radio seleccionado. Una etiqueta "Seleccionar tipo" apunta a este botón. Debajo de eso, el texto "Comentario" precede a un campo de entrada de texto más pequeño. Una etiqueta "Editar comentario" apunta a este campo. En la parte inferior, hay un círculo azul con el número "2", un botón "GUARDAR" con un icono de un disco de almacenamiento, y un botón "SALIR" con un icono de una X roja. Una flecha azul apunta al botón "GUARDAR".

Registro de Incidencias

Descripción: Con esta actividad el usuario aprenderá a registrar una incidencia. Para registrar la incidencia, seguir los siguientes pasos:

Ingresar datos solicitados

The screenshot shows a form titled 'INCIDENCIA' with the following fields and callouts:

- SELECCIONAR PROYECTO** (dropdown menu) - Callout: **Seleccionar**
- Título** (text input) - Callout: **Editar título**
- Descripción** (text input) - Callout: **Editar**
- Tipo** (radio buttons: Método, Consulta, Incidencia) - Callout: **Seleccionar tipo**
- Asignación** (checkbox: Sin Asignar, dropdown) - Callout: **Seleccionar**
- Prioridad** (radio buttons: Alta, Media, Baja) - Callout: **Seleccionar**
- Código** (text input) - Callout: **Editar código**
- Área involucrada** (dropdown menu) - Callout: **Selección**

Numbered callouts: **1** points to the 'Código' field, and **2** points to the 'Área involucrada' dropdown.

Estado de Incidencias

Descripción: Con esta actividad el usuario conocerá el entorno de estado de incidencias y las opciones que ofrece.

El entorno de estado de incidencias ofrece las siguientes características:

The screenshot shows the 'ESTADO DE INCIDENCIAS_2' interface with the following elements:

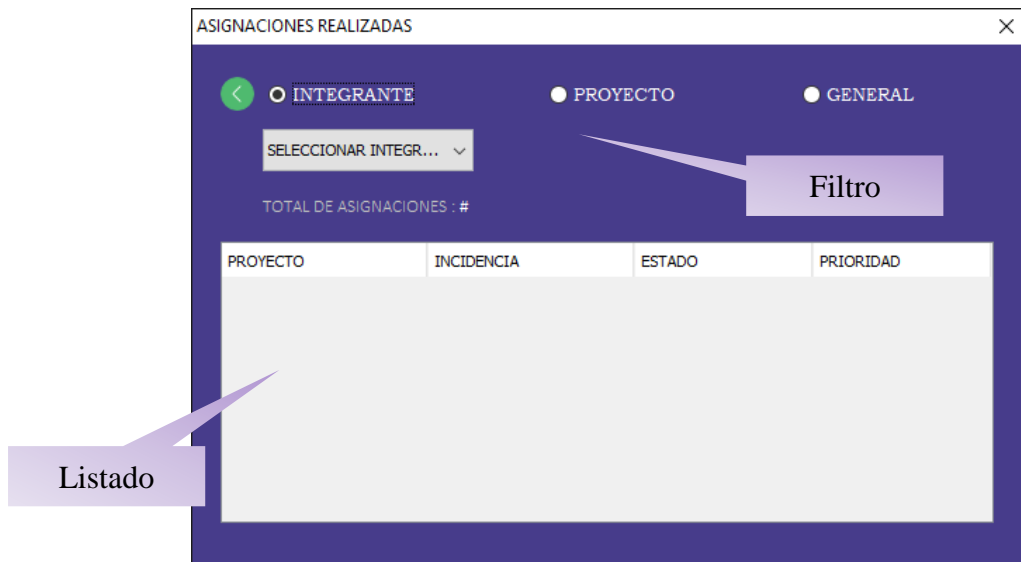
- Filtro** (radio buttons: NO FINALIZADO, FINALIZADO) - Callout: **Filtro**
- Listado** (table) - Callout: **Listado**

NOMBRE	ASIGNACIÓN	PRIORIDAD	FINALI...
Incidencia	---	Alta	NO

Asignación de Incidencias

Descripción: Con esta actividad el usuario conocerá las asignaciones realizadas y las opciones que ofrece.

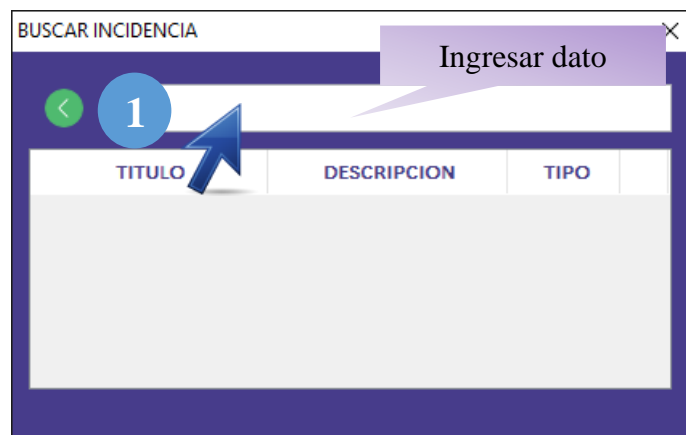
El entorno de asignaciones ofrece las siguientes características:



Buscar incidencia

Descripción: Con esta actividad el usuario aprenderá a buscar una incidencia. Para buscar la incidencia, seguir los siguientes pasos:

Ingresar dato a buscar y presionar Enter



MANUAL TECNICO



HERRAMIENTAS TECNOLÓGICAS

MONGO DB

Descripción:

MongoDB es una base de datos orientada a documentos. Esto quiere decir que, en lugar de guardar los datos en registros, guarda los datos en documentos. Estos documentos son almacenados en una representación binaria de JSON.

Versión:

V3.6.4

Requerimientos mínimos:

RAM: 2GB

Procesador: Core I3

Disco: 40 GB

Instalación:

Instalar el exec de mongod (en nuevas versiones se debe deseleccionar el opcion "install mongocompass")

Añadir la direccion del BIN al path global del sistema

C:\Program Files\MongoDB\Server\3.4\bin

Crear el directorio "data\db" y darle permisos de escritura para todos los usuarios

Ejecutar el comando: mongod en la terminal debera correr sin problemas

Dar permisos de escritura al direccion de instalacion de mongoDB

Crear una carpeta "log" dentro de la carpeta de instalacion de mongoDB

Crear en el directorio de mongoDB un archivo "mongod.cfg"

Abrir "mongod.cfg" y añadir : logpath=C:\Program Files\MongoDB\Server\3.6\log\mongod.log

Correr desde terminal con permisos de administrador:

mongod --config "C:\Program Files\MongoDB\Server\3.6\mongod.cfg" --install

Con esto el servicio aparecera en services.msc de windows

Probar desde terminal ingresar con la palabra "mongo"

Con esto se instala mongo y se genera como servicio de windows.


```
C:\WINDOWS\system32\cmd.exe - mongo v
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Anthony>mongo v
MongoDB shell version v3.6.4
connecting to: mongod://127.0.0.1:27017/v
MongoDB server version: 3.6.4
Server has startup warnings:
2018-12-03T07:54:03.746-0500 I CONTROL [initandlisten]
2018-12-03T07:54:03.746-0500 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.
2018-12-03T07:54:03.746-0500 I CONTROL [initandlisten] **      Read and write access to data and configuration is u
nrestricted.
2018-12-03T07:54:03.746-0500 I CONTROL [initandlisten]
2018-12-03T07:54:03.746-0500 I CONTROL [initandlisten] ** WARNING: This server is bound to localhost.
2018-12-03T07:54:03.746-0500 I CONTROL [initandlisten] **      Remote systems will be unable to connect to this ser
ver.
2018-12-03T07:54:03.746-0500 I CONTROL [initandlisten] **      Start the server with --bind_ip <address> to specify
which IP
2018-12-03T07:54:03.746-0500 I CONTROL [initandlisten] **      addresses it should serve responses from, or with --
bind_ip_all to
2018-12-03T07:54:03.747-0500 I CONTROL [initandlisten] **      bind to all interfaces. If this behavior is desired,
start the
2018-12-03T07:54:03.747-0500 I CONTROL [initandlisten] **      server with --bind_ip 127.0.0.1 to disable this warn
ing.
2018-12-03T07:54:03.747-0500 I CONTROL [initandlisten]
2018-12-03T07:54:03.747-0500 I CONTROL [initandlisten]
2018-12-03T07:54:03.747-0500 I CONTROL [initandlisten] ** WARNING: The file system cache of this machine is configured
to be greater than 40% of the total memory. This can lead to increased memory pressure and poor performance.
2018-12-03T07:54:03.747-0500 I CONTROL [initandlisten] See http://dochub.mongodb.org/core/wt-windows-system-file-cache
2018-12-03T07:54:03.747-0500 I CONTROL [initandlisten]
>
```

EXPRESS

Descripción:

Es un framework de desarrollo de aplicaciones web minimalista y flexible para Node.js". Está inspirado en Sinatra, además es robusto, rápido, flexible y muy simple. Entre otras características, ofrece Router de URL (Get, Post, Put), facilidades para motores de plantillas (Jade, EJS, JinJS ...), Middleware vía Connect y un buen test coverage

Versión:

4.16.3

Requerimientos mínimos:

RAM: 2GB

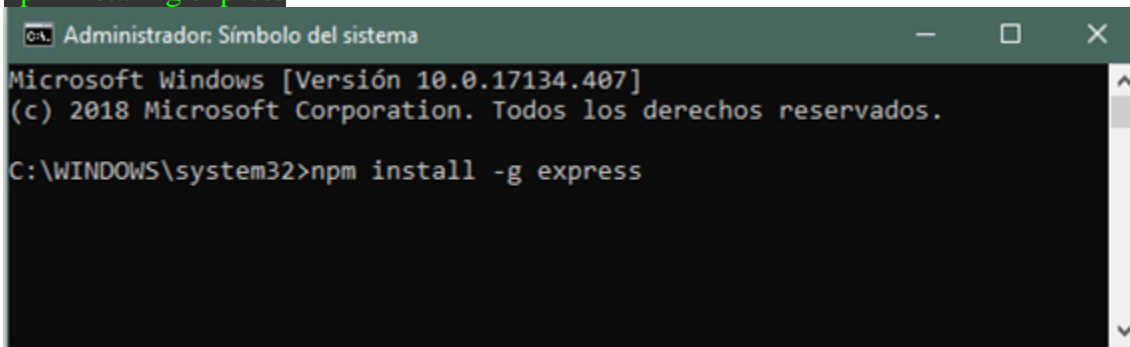
Procesador: Core I3

Disco: 40 GB

Instalación:

Abrir el terminal en la carpeta del proyecto y digitar:

```
npm install -g express
```



The image shows a screenshot of a Windows Command Prompt window titled "Administrador: Símbolo del sistema". The window content displays the following text: "Microsoft Windows [Versión 10.0.17134.407] (c) 2018 Microsoft Corporation. Todos los derechos reservados. C:\WINDOWS\system32>npm install -g express". The command "npm install -g express" is highlighted in green in the original image.

ANGULAR

Descripción:

Es un framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

Versión:

6.1.0

Requerimientos mínimos:

RAM: 2GB

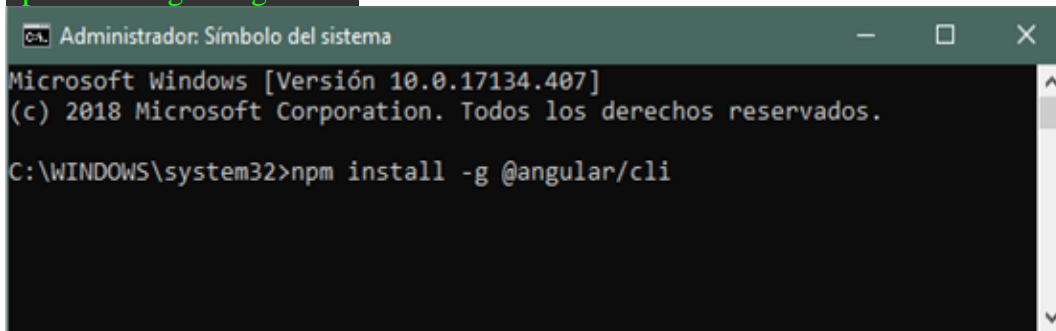
Procesador: Core I3

Disco: 40 GB

Instalación:

Abrir terminal y digitar.

```
npm install -g @angular/cli
```

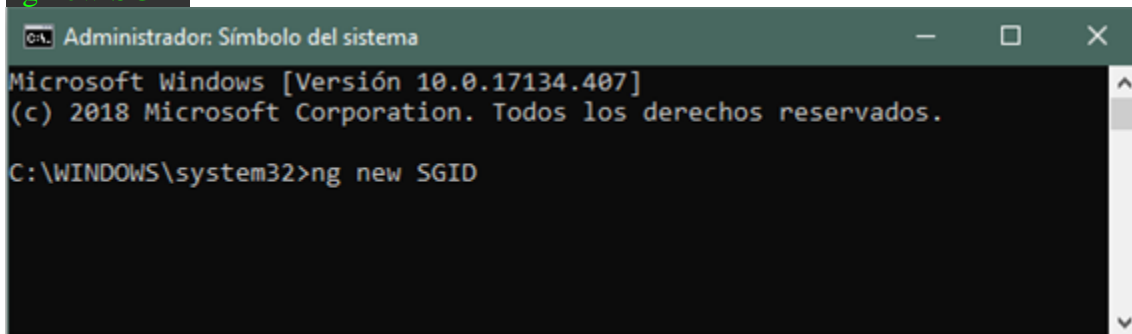


```
Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.17134.407]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\WINDOWS\system32>npm install -g @angular/cli
```

para iniciar un nuevo proyecto se deberá digitar:

```
ng new SGID
```



```
Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.17134.407]
(c) 2018 Microsoft Corporation. Todos los derechos reservados.

C:\WINDOWS\system32>ng new SGID
```

NODEJS

Descripción:

Node.js es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor basado en el lenguaje de programación ECMAScript, asíncrono, con I/O de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google.

Versión:

8.11.1

Requerimientos mínimos:

RAM: 2GB

Procesador: Core I3

Disco: 40 GB

Instalación:

Se instala un ejecutable que puede ser descargado en

<https://nodejs.org/en/#download>

node

CASA | ACERCA DE | DESCARGAS | DOCS | INVOLUCRARSE | SEGURIDAD | NOTICIAS | **FUNDACIÓN**

Node.js® es un tiempo de ejecución de JavaScript creado en el motor de JavaScript V8 de Chrome .

Noviembre de 2018 versiones de seguridad disponibles,
actualizar ahora

Descargar para Windows (x64)

10.14.1 LTS
recomendado para la mayoría de los usuarios

11.3.0
Características
actuales más
recientes

Otras descargas | Registro de cambios | Documentos API Otras descargas | Registro de cambios | Documentos API

JAVA

Descripción:

Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos, que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. El plugin se desarrolló en este lenguaje por compatibilidad con el IDE de NetBeans, el cual es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java.

Versión:

NetBeans IDE: 8.0 o superior

JDK: 7 o superior.

Requerimientos mínimos:

RAM: 2GB

Procesador: Core I3

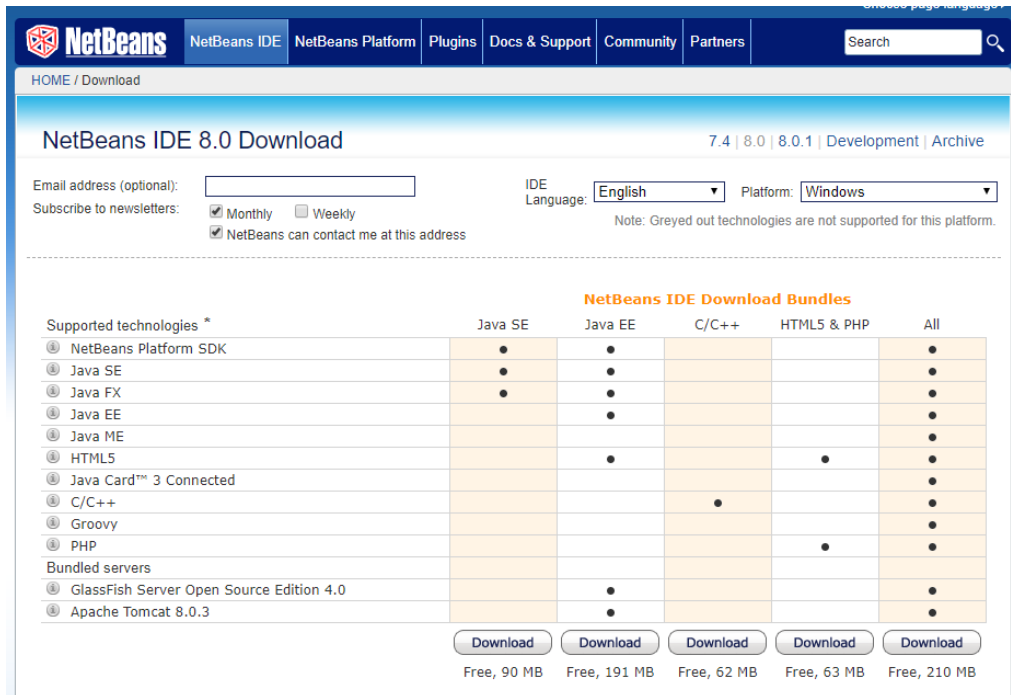
Disco: 1.5 GB

Instalación:

Se instalan de ejecutables que pueden ser descargados de las siguientes páginas:

NetBeans

<https://netbeans.org/downloads/8.0/>



Supported technologies *	Java SE	Java EE	C/C++	HTML5 & PHP	All
NetBeans Platform SDK	•	•			•
Java SE	•	•			•
Java FX	•	•			•
Java EE		•			•
Java ME		•			•
HTML5		•		•	•
Java Card™ 3 Connected				•	•
C/C++			•		•
Groovy					•
PHP				•	•
Bundled servers					
GlassFish Server Open Source Edition 4.0		•			•
Apache Tomcat 8.0.3		•			•

JDK

<https://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase7-521261.html>



ESPECIFICACIÓN DE REQUERIMIENTOS

HISTORIA DE USUARIO			
Número	1	Usuario	Cliente
Nombre de historia	Inicio sesión	Riego en desarrollo	Bajo
Prioridad en negocio	Alta	Sprint asignada	1
Puntos estimados	5	Dependencia para su desarrollo	Persona
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero acceder al sistema mediante un inicio de sesión para que mi acceso sea seguro.		
Criterio de aceptación	<ul style="list-style-type: none"> - Se accederá ingresando correo y contraseña. - Los atributos “correo” y “contraseña” serán llamados del esquema persona. - Se realizará una interfaz gráfica en la web y el plugin para el logueado del usuario. - en el plugin una vez que accedió el usuario se habilitará el ícono del plugin en el IDE. 		

HISTORIA DE USUARIO			
Número	2	Usuario	Cliente
Nombre de historia	Incidencia	Riego en desarrollo	Alto
Prioridad en negocio	Alta	Sprint asignada	2
Puntos estimados	8	Dependencia para su desarrollo	Persona, Tipo de registro
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero registrar mis incidencias para conocer cuáles con las que cuento al momento de realizar el proceso de desarrollo de software.		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como” incidencias”. - Sus atributos serán: titulo, descripcion, fecha_creacion, hora_creacion, persona_registra, comentario, tipo_incidencia, persona_asignada, estado, asignación, prioridad. - Se creará una interfaz gráfica en la web y el plugin para su gestión. - La incidencia podrá ser agregada, editada, listada y eliminada mediante la interfaz web y plugin. 		

HISTORIA DE USUARIO			
Plata	3	Usuario	Cliente
Nombre de historia	Roles	Riego en desarrollo	Bajo
Prioridad en negocio	Alta	Sprint asignada	1
Puntos estimados	3	Dependencia para su desarrollo	Ninguna
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero clasificar a los integrantes de mis proyectos por su rol en el mismo.		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como “roles”. - Sus atributos serán: nombre_rol - Se creará una interfaz gráfica web para su gestión. 		

HISTORIA DE USUARIO			
Número	4	Usuario	Cliente
Nombre de historia	Persona	Riego en desarrollo	Alto
Prioridad en negocio	Alta	Sprint asignada	1
Puntos estimados	8	Dependencia para su desarrollo	Ninguna
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero registrarme como usuario del software para tener acceso a sus funciones.		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como "personas" - Sus atributos serán: nombre_persona, correo, apellido_paterno, apellido_materno, contraseña, dni, estado, cargo - Todos los datos son obligatorios - Se realizará una interfaz gráfica web para el registro de la persona. - Se creará una interfaz web para la modificación de sus datos. - La persona mediante su cuenta podrá darse de baja - Dar de baja a una persona será un cambio de estado “Activo” a “Inactivo”. - La búsqueda se realizará mediante el nombre o correo de la persona - Solo el jefe de proyecto podrá buscar y listar usuarios activos e inactivos que han participado en sus proyectos. 		

HISTORIA DE USUARIO			
Número	5	Usuario	Cliente
Nombre de historia	Proyecto	Riego en desarrollo	Bajo
Prioridad en negocio	Alta	Sprint asignada	3
Puntos estimados	5	Dependencia para su desarrollo	Persona
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero registrar mis proyectos para llevar un control de sus incidencias.		
Criterio de aceptación	<p>En la base de datos se registrará como “proyectos” Sus atributos serán: nombre, fecha_creación, hora_creacion, descripcion, persona_registra. El atributo “persona_registra” será llamado del esquema “persona” Se creará una interfaz gráfica web para su gestión. El usuario que registra el proyecto se vuelve automáticamente el jefe del proyecto. El jefe de proyecto podrá buscar personas e integrarlas al proyecto. El jefe de proyecto deberá asignarle un rol al integrante agregado al proyecto. La lista de integrantes se deberá visualizar al momento de acceder al proyecto. La lista de incidencias se deberá visualizar al momento de acceder al proyecto.</p>		

HISTORIA DE USUARIO			
Número	6	Usuario	Cliente
Nombre de historia	Solución	Riego en desarrollo	Alto
Prioridad en negocio	Alta	Sprint asignada	2
Puntos estimados	8	Dependencia para su desarrollo	Persona, Incidencia
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero registrar mis soluciones para ...		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como “solución”. - Sus atributos serán: tipo_descripcion, fecha_creacion, hora_creacion, nombre_archivo, descripción, tipo_solucion, comentario, persona_registra, estado. - Se gestionará desde la interfaz gráfica web y plugin de incidencia. - Los tipos de descripción son dos, "texto" y "código". - Los tipos de solución son dos, "temporal" y "final". - Los estados de solución son dos, “resuelto” y “no resuelto”. - Cuando el tipo de descripción sea código, debe aperturarse la opción de "Nombre de archivo" para su registro. - El usuario puede registrar una incidencia con o sin solución. - Los atributos “fecha_creacion”, “hora_creacion” y “persona_registra” se tomarán del sistema. - Solo cuando el tipo de solución es “final” el estado de solución será “resuelto”. - El atributo “persona_registra” será tomado del esquema persona. - El atributo “id_proyecto” será tomado del esquema proyecto. 		

HISTORIA DE USUARIO			
Número	7	Usuario	Cliente
Nombre de historia	Integrante	Riego en desarrollo	Bajo
Prioridad en negocio	Media	Sprint asignada	3
Puntos estimados	5	Dependencia para su desarrollo	Persona, Proyecto, Roles
Programador responsable	Carol Janet Guerrero Gutiérrez		
Descripción	Yo como cliente quiero agregar personas a mis proyectos como integrantes para que puedan registrar incidencias que se presenten en el proceso de desarrollo.		
Criterio de aceptación	<ul style="list-style-type: none"> - En la base de datos se registrará como "integrante" - Sus atributos serán: id_persona, id_proyecto, id_rol, permiso, fecha_registro, estado. - En la búsqueda y listado del usuario, para registrarlo como integrante y agregarlo a un proyecto, se mostrarán aquellas personas que se encuentren activas y disponibles. - Al momento de agregar un integrante al proyecto, se podrá seleccionar su rol, pero por defecto este será desarrollador. - El rol será asignado por el jefe de proyecto. - Se podrá cambiar el rol de un integrante en un mismo proyecto. - La lista de integrantes será por proyecto al que pertenecen. - La fecha y hora de registro se tomará del sistema. - El jefe de proyecto puede asignar a otra persona su rol como jefe de proyecto. - Al asignar a otra persona como jefe de proyecto su rol cambiara por defecto a "Desarrollador". - El "id_persona" será llamado del esquema "persona" - El "id_proyecto" será llamado del esquema "proyecto" - El "id_rol" será llamado del esquema "rol" - Los tipos de "permiso" serán dos, lectura y escritura - Al eliminar un integrante de un proyecto cambiará de estado de "activo" a "inactivo". 		

ACTORES DEL SISTEMA





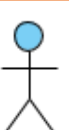
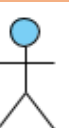
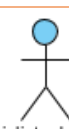
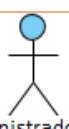
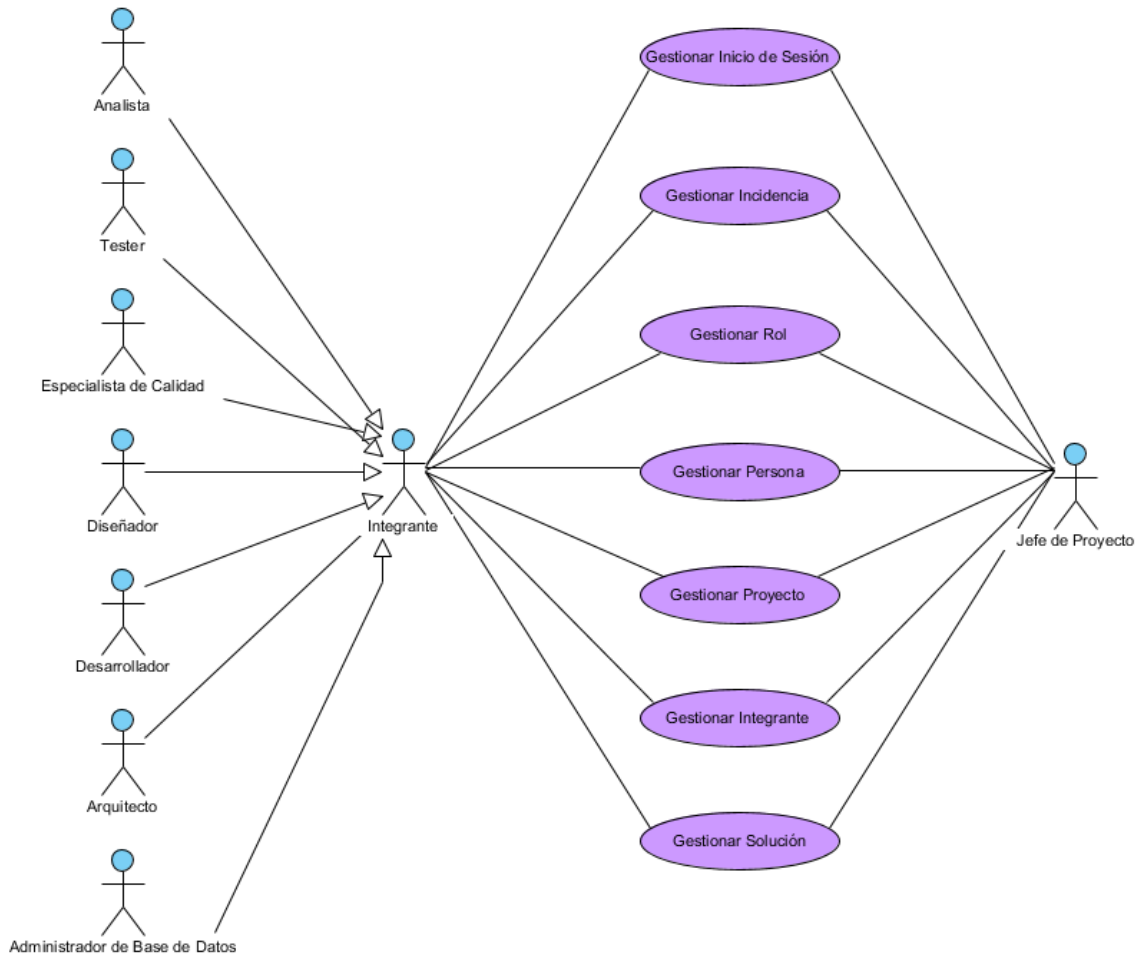
ACTOR	DESCRIPCIÓN
 <p>Jefe de Proyecto</p>	<p>Persona responsable de la gestión y planificación del proyecto, controlar los avances, asignar el trabajo, realizar seguimientos y realizar las coordinaciones con su equipo y el usuario final.</p>
 <p>Analista</p>	<p>Persona responsable del análisis del sistema, especifica requisitos, realiza análisis general y detallado, propone mejoras y soluciones respecto a los requerimientos del cliente.</p>
 <p>Arquitecto</p>	<p>Persona responsable de estructurar el sistema, convierte los requisitos definidos por el analista en una solución técnica y realiza el seguimiento del desarrollo para verificar la estructura generada.</p>
 <p>Diseñador</p>	<p>Persona responsable del diseño del software, se cerciora que el sistema cumpla con las características de un sistema amigable con el usuario final.</p>
 <p>Desarrollador</p>	<p>Persona responsable de implementar componentes y funcionalidades mediante la codificación, ubica y corrige defectos, documenta el código e integra subsistemas implementados y su contenido durante el desarrollo.</p>
 <p>Tester</p>	<p>Persona responsable de la gestión y el desarrollo de las pruebas a realizar en el sistema, debe asegurar el funcionamiento correcto del sistema construido.</p>
 <p>Especialista de Calidad</p>	<p>Persona responsable de instaurar un plan de calidad, debe observar el proceso de desarrollo, examinar los resultados y garantizar que el sistema esté libre de defectos.</p>
 <p>Administrador de Base de Datos</p>	<p>Persona responsable de administrar la base de datos, crea, gestiona, implementa y brinda soporte a la base de datos de la organización.</p>

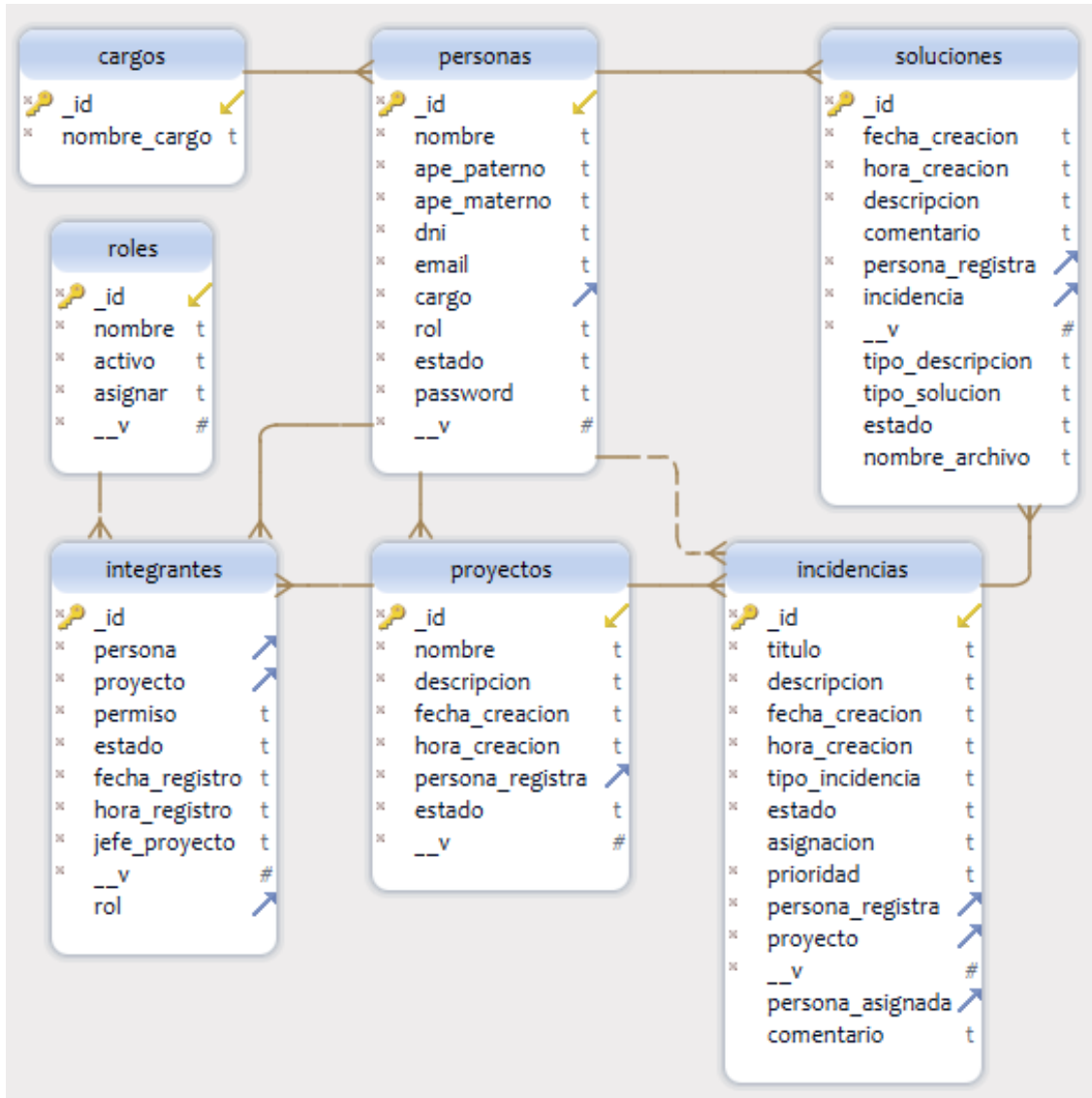
DIAGRAMA DE CASO DE USO



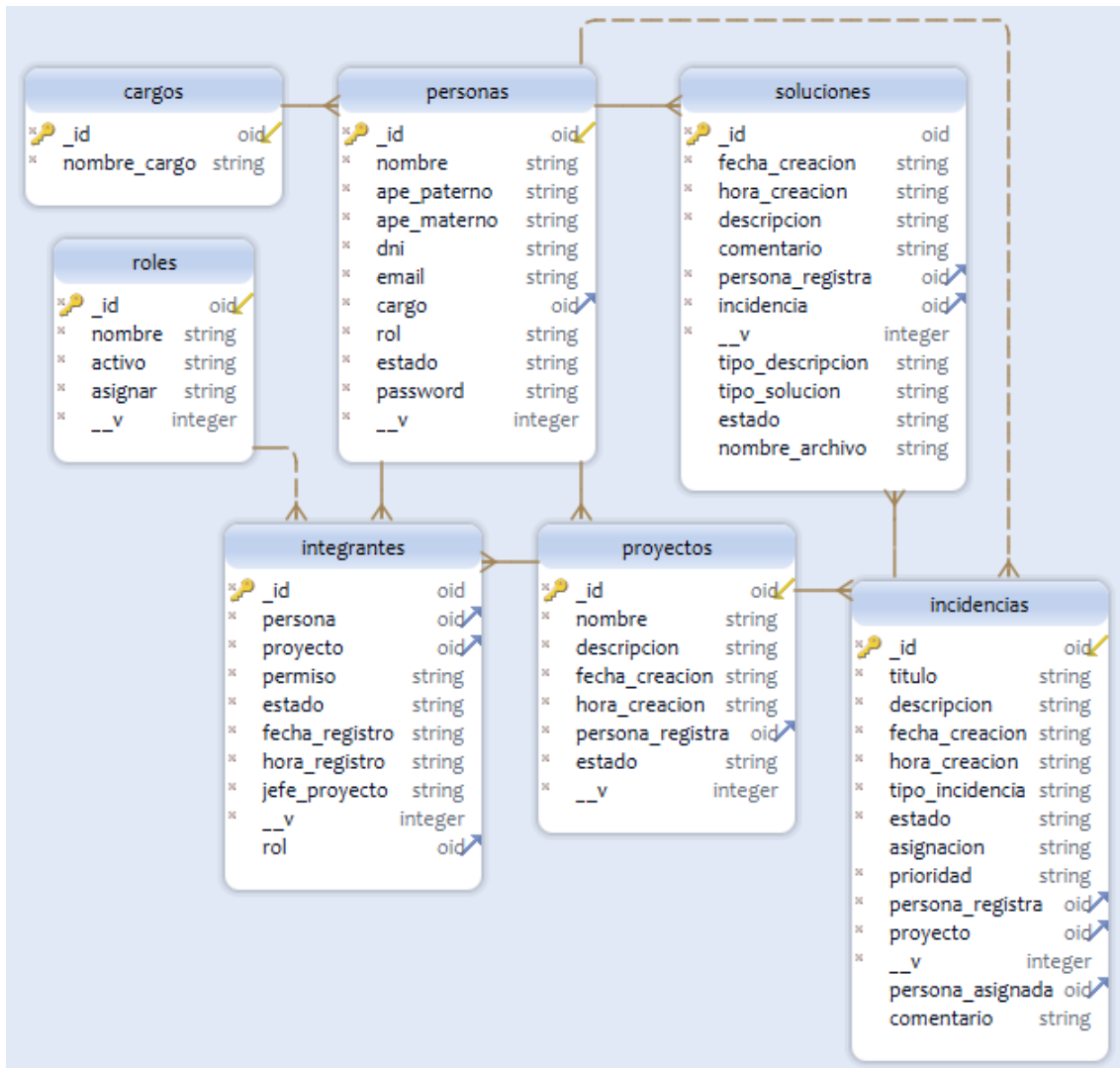
BASE DE DATOS

Modelo de Base de Datos

Lógico



Físico



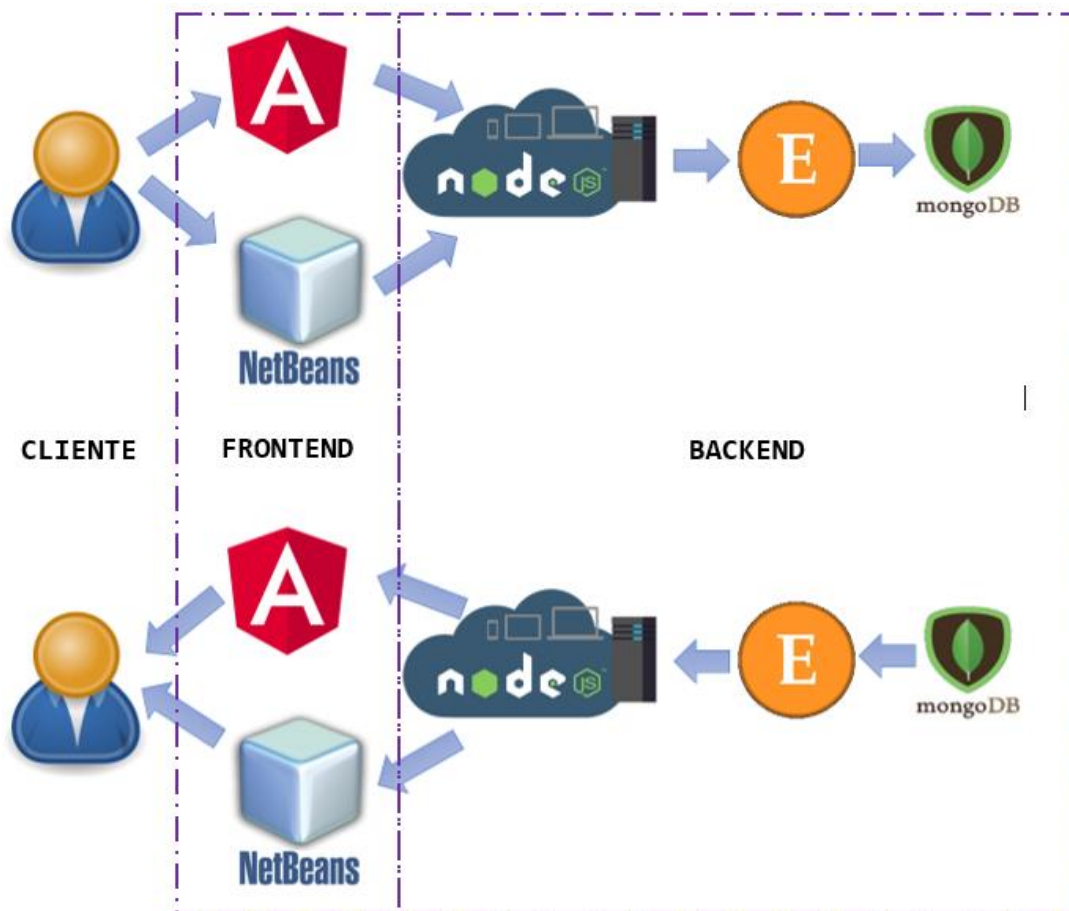
Diccionario de Base de Datos

TABLA	TITULO	DESCRIPCIÓN	VALORES
CARGOS	Id	Identificador del cargo	Asignado por defecto
	nombre_cargo	Nombre del cargo correspondiente al área	Jefatura Operario Practicante
ROLES	id	Identificador del rol	Asignado por defecto
	nombre	Nombre del rol correspondiente del actor	Asignado por jefe de área
	activo	Estado del rol	Activo: 1 Inactivo: 0
	asignar	Permiso de asignación	Asigna: 1 No asigna: 0
PERSONAS	id	Identificador de la persona	Asignado por defecto
	nombre	Nombre del usuario	Dato ingresado por el usuario
	ape_paterno	Apellido paterno del usuario	Dato ingresado por el usuario
	ape_materno	Apellido materno de usuario	Dato ingresado por el usuario
	dni	Documento nacional de identificación del usuario	Dato ingresado por el usuario
	email	Email del usuario	Dato ingresado por el usuario
	cargo	Cargo que desempeña el usuario	Jefatura Operario Practicante
	estado	Estado del usuario	Activo: 1 Inactivo: 0
password	Contraseña del usuario	Dato ingresado por el usuario	
INTEGRANTES	id	Identificador del integrante	Asignado por defecto
	persona	Identificador de la persona	Datos del usuario

	proyecto	Identificador del proyecto	Datos del proyecto al que pertenece
	permiso	Permisos de edición	Editar: 1 Leer: 0
	estado	Estado del integrante	Activo: 1 Inactivo: 0
	fecha_registro	Fecha en la que se registra al integrante	Asignados por el sistema
	hora_registro	Hora en la que se registra al integrante	Asignados por el sistema
	jefe_proyecto	Clasifica si es jefe o integrante del proyecto	Si: 1 No: 0
	rol	El rol que cumple dentro del proyecto	Id de roles
PROYECTOS	id	Identificador del proyecto	Asignado por defecto
	nombre	Nombre del proyecto	Dato ingresado por jefe del proyecto
	descripcion	Descripción de proyecto	Dato ingresado por jefe de proyecto
	fecha_creacion	Fecha en el que se crea el proyecto	Asignados por el sistema
	hora_creacion	Hora en el que se registra el proyecto	Asignados por el sistema
	persona_registra	Usuario que registra el proyecto	Id de persona
	estado	Estado del proyecto	Habilitado: 1 Inhabilitado: 0
INCIDENCIAS	id	Identificador de la incidencia	Asignado por defecto
	titulo	Título para referenciar incidencia	Ingresado por el usuario
	descripcion	Describe en que consiste la incidencia	Ingresado por el usuario
	fecha_creacion	Fecha en la que se registra la incidencia	Asignados por el sistema
	hora_creacion	Hora en la que se registra la incidencia	Asignados por el sistema

	tipo_incidencia	Tipo de la incidencia	Incidencia Método Consulta
	estado	Estado de la incidencia	Activo: 1 Inactivo: 0
	asignacion	Si tiene persona asignada	Si: 1 No: 0
	prioridad	La prioridad para su solución	Alta Media Baja
	persona_registra	El usuario que ha registrado la incidencia	Id del usuario
	proyecto	El proyecto al que corresponde	Id del proyecto
	persona_asignada	La persona a quien ha sido asignada la incidencia	Id de persona
	comentario	Comentario que realice el usuario	Dato ingresado por el usuario
SOLUCIONES	id	Identificador de la solución	Asignado por defecto
	fecha_creación	Fecha en la que se registra la solución	Asignados por el sistema
	hora_creacion	Hora en la que se registra la solución	Asignados por el sistema
	descripción	Descripción de la solución	Dato ingresado por el usuario
	comentario	Comentario de la solución	Dato ingresado por el usuario
	persona_registra	Usuario que registra el comentario	Id de la persona
	incidencia	Incidencia a la que pertenece la solución	Id de la incidencia
	estado	Estado de la solución	Activo: 1 Inactivo: 0

ARQUITECTURA DEL SOFTWARE



Descripción

La arquitectura del proyecto está dividida en 2 grandes grupos:

FRONT-END

Conformada por todo aquello que interactúa directamente con el usuario, IDE NetBeans o Pagina web. Dentro de este apartado se emplea lenguajes de estructura e datos como HTML, de maquetación como CSS o lenguaje de programación interpretado como JavaScript, todo esto a través del framework **ANGULAR**, desarrollada por Google, que permite crear single page Application, haciendo la navegación e interacción con un sistema web sea más eficiente, debido a que ya no debe renderizar todo los elementos HTML desde un servidor en backend, sino solo deberá interpretar la información que recibe en formato Json.

BACK-END

Este grupo está conformado por 2 secciones:

Servidor Web:

Al ser un entorno de ejecución de JavaScript y contar con un módulo http que permite generar un servidor web interno, haciendo que la comunicación entre frontend y backend sea más rápida y liviana.

Para dicha comunicación se generó un API REST, que es empleado en la actualidad para tecnologías móviles, pero se viene tomando como estándar, debido a que la transmisión de información es mucho más ligera y en un formato más ordenado, JSON.

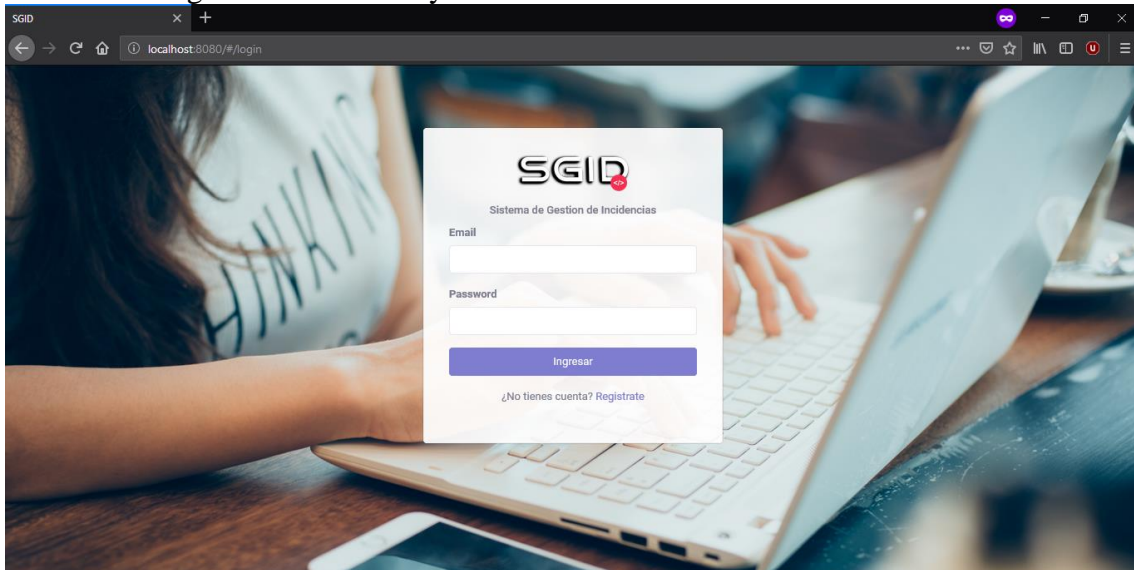
Base de Datos

Se empleó una base de datos No-SQL que permite ser flexible con la información almacenada, debido a que no se genera una estructura tradicional como en otros motores de datos, la información es almacenada en archivos Json, permitiendo que la información sea recogida de esta forma y la interpretación tanto en el API REST como la aplicación FRONT-END se realice de manera más fluida.

VISTA DE PANTALLAS PRINCIPALES WEB

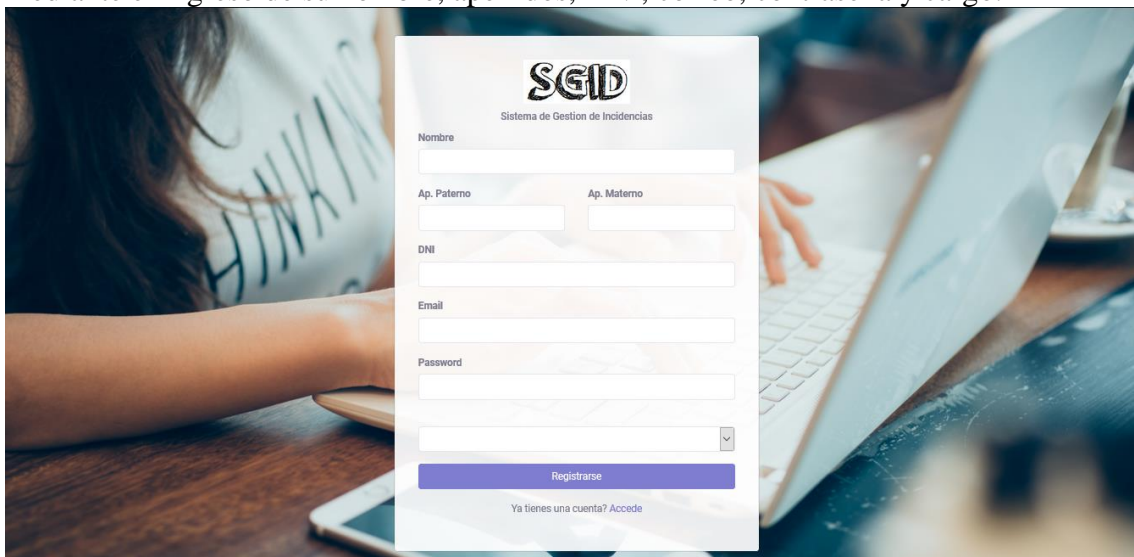
PANTALLA DE INICIO DE SESIÓN

Con esta interfaz se puede realizar el inicio de sesión del usuario en la aplicación web, mediante el ingreso de su correo y contraseña.



PANTALLA DE REGISTRO

Con esta interfaz se puede realizar el registro de un nuevo usuario en la aplicación web, mediante el ingreso de su nombre, apellidos, DNI, correo, contraseña y cargo.



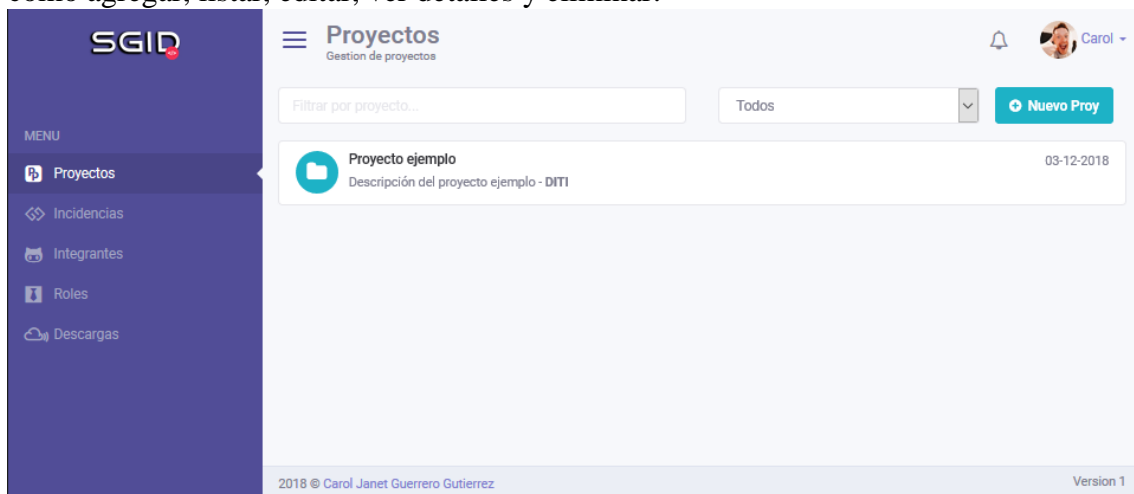
PANTALLA PRINCIPAL

Con esta interfaz se puede tener una visión general del entorno del sistema.



PANTALLA DE PROYECTO

Con esta interfaz se puede gestionar los proyectos en la aplicación web, realizar acciones como agregar, listar, editar, ver detalles y eliminar.



PANTALLA DE INCIDENCIAS

Con esta interfaz se puede gestionar las incidencias en la aplicación web, realizar acciones como agregar, listar, editar, ver detalles y eliminar.

The screenshot shows the 'Incidentes' (Incidents) management page. The interface includes a dark blue sidebar with a 'MENU' section containing 'Proyectos', 'Incidentes', 'Integrantes', 'Roles', and 'Descargas'. The main content area has a header with the 'Incidentes' title and 'Gestión de incidencias' subtitle. Below the header is a search bar labeled 'Filtrar por incidencias...' and a dropdown menu for 'Proyecto ejemplo'. A blue button labeled 'Incidencia' is positioned to the right. The main area features a table with columns for 'Todo', 'Incidencias', 'Metodos', and 'Consultas'. The table lists three items: 'Incidencia', 'Método', and 'Consulta', each with a title, description, type, creation date (03-12-2018), and a set of action icons (eye, edit, delete). The footer contains the copyright notice '2018 © Carol Janet Guerrero Gutierrez' and 'Version 1'.

PANTALLA DE INTEGRANTES

Con esta interfaz se puede gestionar los Integrantes del proyecto en la aplicación web, realizar acciones como agregar integrante por proyecto, listar, editar, ver detalles y quitar a integrante de un proyecto.

The screenshot shows the 'Integrantes' (Members) management page. The sidebar is identical to the previous page. The main content area has a header with the 'Integrantes' title and 'Gestión de integrantes' subtitle. Below the header is a dropdown menu for 'Proyecto' set to 'Proyecto ejemplo'. A blue button labeled 'Listado' with a plus icon and a close icon is present. Below this is a card for a member named 'Guerrero Carol' with the role 'Analista'. The footer contains the copyright notice '2018 © Carol Janet Guerrero Gutierrez' and 'Version 1'.

PANTALLA DE ROLES

Con esta interfaz se puede gestionar los roles en la aplicación web, realizar acciones como agregar, listar, editar y cambiar de estado.

SGID

MENU

- Proyectos
- Incidencias
- Integrantes
- Roles**
- Descargas

Roles
Gestión de roles

+ Añadir

Rol	Estado	Acción
Analista	Activo	<input checked="" type="checkbox"/>
Desarrollador	Activo	<input checked="" type="checkbox"/>
Arquitecto	Activo	<input checked="" type="checkbox"/>
Diseñador	Activo	<input checked="" type="checkbox"/>
Tester	Activo	<input checked="" type="checkbox"/>
Especialista de Calidad	Activo	<input checked="" type="checkbox"/>
Administrador de Base de Datos	Activo	<input checked="" type="checkbox"/>

2018 © Carol Janet Guerrero Gutierrez

Version 1

PANTALLA DE DESCARGAS

Con esta interfaz se puede realizar la descarga del plugin para su posterior instalación en el IDE de NetBeans.


SGID

MENU

- Proyectos
- Incidencias
- Integrantes
- Roles
- Descargas**

Descargas
Descargas de plugin

Información



NetBeans

IDE: Netbeans

Version: 1.0

Instalación: _____

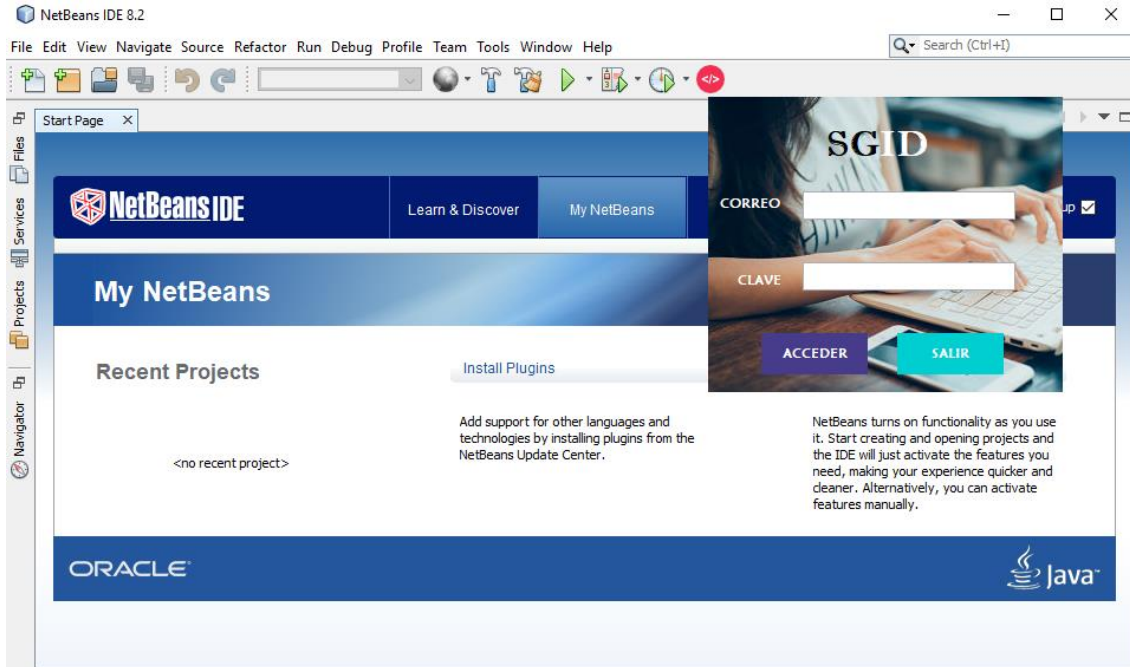
2018 © Carol Janet Guerrero Gutierrez

Version 1

VISTAS DE PANTALLAS PRINCIPALES PLUGIN

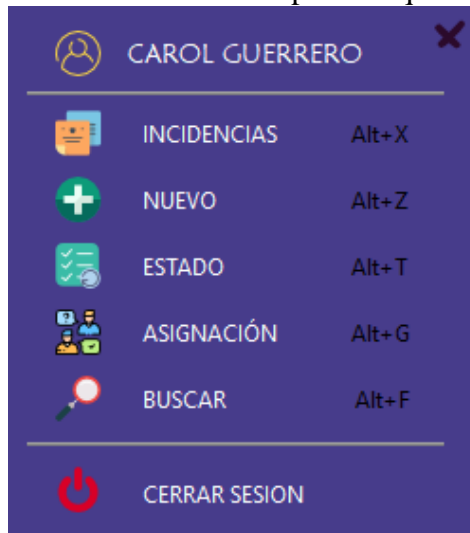
PANTALLA DE INICIO DE SESIÓN

Con esta interfaz se puede realizar el inicio de sesión del usuario con el plugin en NetBeans, mediante el ingreso de su correo y contraseña.



PANTALLA DE MENÚ PRINCIPAL

Con esta interfaz se puede acceder al menú de opciones que brinda el plugin.



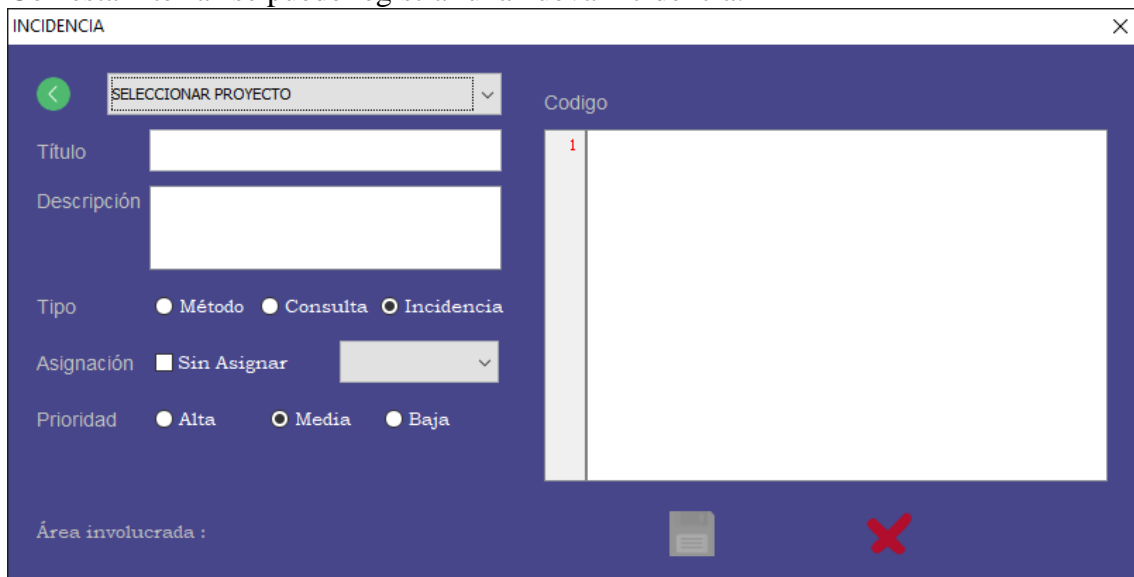
PANTALLA DE OPCIÓN INCIDENCIAS

Con esta interfaz se puede gestionar las incidencias desde NetBeans, realizar acciones como agregar, listar, editar, ver detalles y eliminar



PANTALLA DE OPCIÓN NUEVO

Con esta interfaz se puede registrar una nueva incidencia.



PANTALLA DE OPCIÓN ESTADO

Con esta interfaz se puede llevar un control del estado de las incidencias y así poder gestionar las no finalizadas, como considere el jefe de proyecto.



PANTALLA DE OPCIÓN ASIGNACIÓN

Con esta interfaz se puede llevar un control de las asignaciones realizadas por integrantes del equipo, proyecto o de forma general.



PANTALLA DE OPCIÓN BUSCAR

Con esta interfaz se puede realizar la búsqueda de incidencias desde el IDE del desarrollador.



The screenshot shows a mobile application window titled "BUSCAR INCIDENCIA". At the top left, there is a green circular button with a white left-pointing arrow. To its right is a white search input field. Below the search bar is a table with three columns: "TITULO", "DESCRIPCION", and "TIPO". The table body is currently empty, showing a light gray background.

TITULO	DESCRIPCION	TIPO
--------	-------------	------

GLOSARIO DE TÉRMINOS

Incidencia: Una incidencia es toda interrupción o reducción de la calidad no planificada del servicio. Pueden ser fallos o consultas reportadas por los usuarios, el equipo del servicio o por alguna herramienta de monitorización de eventos.

Método: Consiste generalmente de una serie de sentencias para llevar a cabo una acción, un juego de parámetros de entrada que regularán dicha acción o, posiblemente, un valor de salida (o valor de retorno) de algún tipo.

Consulta: Una consulta es el método para acceder a la información en las bases de datos. Con las consultas se puede modificar, borrar, mostrar y agregar datos en una base de datos, también pueden utilizarse como origen de registro para formularios.

Plugin: Aplicación que un programa informático, añade una funcionalidad adicional o una nueva característica al software que va a complementar.

IDE: Entorno de desarrollo interactivo, aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software.

JDK: EL Java Developer Kit es un software que provee herramientas de desarrollo para la creación de programas en Java. Puede instalarse en una computadora local o en una unidad de red.

Frontend: Son todas aquellas tecnologías que corren del lado del cliente, es decir, todas aquellas tecnologías que corren del lado del navegador web.

Backend: se encuentra del lado del servidor, es decir, se encarga de interactuar con bases de datos, verificar manejo de sesiones de usuarios y montar la página en un servidor.

TypeScript: Lenguaje de programación libre y de código abierto, es un superconjunto de JavaScript, que esencialmente añade tipado estático y objetos basados en clases.

Framework: Entorno de trabajo es una estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de software.

MVC: Modelo – Vista – Controlador, es un patrón de arquitectura de software que separa la lógica de la aplicación de la lógica de la vista en una aplicación.