

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

**CARRERA
INGENIERÍA DE SISTEMAS**

**Trabajo de titulación previo a la obtención del título de:
Ingenieros de Sistemas**

**TEMA:
ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DEL MÓDULO DE REGISTRO DE
UN WORKFLOW PARA EL SEGUIMIENTO DE PASANTÍAS DE LA
CARRERA DE INGENIERÍA DE SISTEMAS - SEDE QUITO.**

**AUTORES:
JHON ALEXANDER CASTILLO REIMUNDO
JAIRO DANIEL FLORES GALLEGOS**

**TUTOR:
RODRIGO EFRAÍN TUFIÑO CÁRDENAS**

Quito, marzo del 2018

CESIÓN DE DERECHOS DE AUTOR

Nosotros Castillo Reimundo Jhon Alexander con documento de identificación No. 1718900184 y Flores Gallegos Jairo Daniel con documento de identificación No. 1721397113, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación con el tema: "ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DEL MÓDULO DE REGISTRO DE UN WORKFLOW PARA EL SEGUIMIENTO DE PASANTÍAS DE LA CARRERA DE INGENIERÍA DE SISTEMAS - SEDE QUITO.", mismo que ha sido desarrollado para optar por el título de INGENIEROS DE SISTEMAS en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



**JHON ALEXANDER
CASTILLO REIMUNDO**

CI: 1718900184



**JAIRO DANIEL
FLORES GALLEGOS**

CI: 1721397113

Quito, marzo del 2018

DECLARATORIA DE COAUTORÍA DEL TUTOR

Yo, declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación, con el tema: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DEL MÓDULO DE REGISTRO DE UN WORKFLOW PARA EL SEGUIMIENTO DE PASANTÍAS DE LA CARRERA DE INGENIERÍA DE SISTEMAS - SEDE QUITO., realizado por Jhon Alexander Castillo Reimundo y Jairo Daniel Flores Gallegos, obteniendo un producto que cumple con los requisitos estipulados por la Universidad Politécnica Salesiana para ser considerado como trabajo final de titulación.

Quito, marzo del 2018



RODRIGO EFRAÍN TUFIÑO CÁRDENAS

CI: 1717646390

Dedicatoria

Dedico este trabajo a mis padres Juan Castillo y Lucia Reimundo, que siempre me han apoyado estando en las buenas y malas, siempre guiándome por el camino correcto dándome consejos invaluable de vida, en todo este tiempo siempre han sido lo mejor para mí y la mejor forma de reeditar su esfuerzo y cariño hacia mí es demostrando ser un gran profesional y una mejor persona. A mis hermanas Priscila Castillo y Paola Castillo por ser una ayuda constante semestre a semestre, ofreciendo su sabiduría y consejos en los momentos que más lo necesite.

A mi madrina Carmen Logacho desde lo más profundo de mi corazón le dedico este trabajo, usted se merece esto y mucho más, es el gran artífice de la persona que hoy soy.

Jhon Alexander Castillo Reimundo

Dedicado a mi padre Eduardo Flores que ha sido el pilar de mis logros gracias a él he podido ser un hombre de bien y lleno de éxitos en mi vida. A mi esposa Leslie Castro que me ha brindado su apoyo incondicional en mi educación superior. A mi tía Gladis Burbano quien día a día me ayudó en mi formación y dándome la alimentación. A mi madre Marlene Gallegos quien aportó con su ayuda incondicional en mi vida y más aún cuando más la necesitaba. A mi hermano Dario Perez que me ha inspirado a conseguir mis sueños.

Jairo Daniel Flores Gallegos

Agradecimiento

Agradezco a mi madrina Carmen Logacho, nada de esto hubiera sido posible sin usted, gracias por todo su amor y cariño brindado hacia mí, le agradezco con todo mi corazón por todo lo que hizo y lo que represento para mí.

A mi amigo Jairo Flores, el gusto de haber sido amigos y compartido todo nuestro camino universitario, por todos estos años, finalmente todo nuestro esfuerzo ha dado resultado.

Jhon Alexander Castillo Reimundo

Agradezco a mi tutor Rodrigo Tufiño por su ayuda prestada dentro de todo el proceso de mi trabajo de titulación. También un agradecimiento a Bryan Torres por la mutua ayuda en la etapa de desarrollo. A mi amigo Jhon Castillo con quien he compartido la vida universitaria y juntos hemos alcanzado el objetivo más anhelado.

Jairo Daniel Flores Gallegos

ÍNDICE

1. Marco Metodológico	5
1.1. Workflow	5
1.1.1. Historia.....	6
1.1.2. Características de un Workflow	7
1.1.3. Administración Workflow	8
1.1.4. Funciones del Workflow:.....	8
1.1.5. Ventajas Workflow:	9
1.1.6. Conceptos de Base	10
1.1.7. Tipos de sistemas workflow	13
1.2. Lenguaje de Modelado Unificado (UML)	14
1.2.1. Bloques de construcción básicos.....	16
1.2.2. Mecanismos comunes	17
1.2.3. Especificaciones.....	17
1.2.4. Adornos	17
1.2.5. Divisiones comunes	17
1.3. Scrum	18
1.3.1. Definición.....	18
1.3.2. Eventos de Scrum.....	19
1.3.3. El equipo scrum.....	21
1.3.4. Artefactos de scrum	22
1.4. JSF	22
1.5. Pasantía	23
1.5.1. Las responsabilidades.....	24
1.5.2. Evaluación y acreditación	25
1.5.3. Prácticas preprofesionales.....	25
1.5.4. Las responsabilidades.....	26
1.5.5. Evaluación y acreditación	26
1.5.6. Ley Educación Superior Art. 87.....	27
1.6. XP	27

2. Análisis y Diseño.....	28
2.1. Análisis.....	28
2.1.1. Introducción	28
2.1.2. Propósito	28
2.1.3. Alcance.....	29
2.1.4. Audiencia	29
2.1.5. Definiciones, acrónimos y abreviaturas	29
2.1.6. Visión general	30
2.1.7. Requerimientos de interfaz externa.....	32
2.1.8. Módulos globales del producto	33
2.1.9. Requerimientos específicos.....	34
2.2. Diseño	43
2.2.1. Diagrama de casos de uso	43
2.2.2. Esquema de clases.....	45
2.2.3. Diagrama conceptual de base de datos.....	46
2.2.4. Interfaz abstracta	48
3. Construcción y Pruebas.....	55
3.1. Herramientas	55
3.1.1. Hibernate	55
3.1.2. HQL	55
3.1.3. Sb Admin 2	56
3.1.4. Bootsfaces	56
3.1.5. IText	56
3.1.6. Java Mail API 1.4.....	57
3.1.7. PostgreSQL 9.4	57
3.1.8. PgAdmin III	57
3.1.9. GitHub.....	57
3.1.10. Glassfish.....	58
3.1.11. Java JDK 8u151	58
3.2. Modelo Físico de la base de datos.....	58
3.3. Diccionario de Datos.....	61
3.4. Historias de Usuario	66

3.4.1. Módulo de registro de pasantía	67
3.4.2. Módulo de validación de documentación	73
3.5. Estructura de paquetes y clases	79
3.5.1. Paquete: com.sppp.beans	79
3.5.2. Paquete: com.sppp.DAO	82
3.5.3. Paquete: com.sppp.utils	83
3.5.4. Paquete: com.sppp.mailing	83
3.5.5. Paquete: com.sppp.classes	84
3.5.6. Paquete: com.sppp.servlets	84
3.6. Código	84
3.6.1. Web.xml configuraciones Web	85
3.6.2. Configuración de Hibernate.	85
3.6.3. Código de control de usuario por perfil.	86
3.6.4. Configuración del Servlet <code>Download.java</code> para subir y bajar archivos.....	87
3.6.5. Configuración de la clase de mensajería instantánea (email).	89
3.6.6. Almacenamiento y recuperación de datos.....	89
3.7. Pruebas	91
3.7.1. Pruebas unitarias	91
3.7.2. Pruebas de estrés	93
3.7.3. Pruebas contra requerimientos	98
4. Implementación	101
4.1. Configuración de la base de datos.....	101
4.2. Configurar el Firewall de Windows.....	101
4.3. Descripción del Hardware y Software	102
4.3.1. Características del hardware	102
4.3.2. Características del software.....	103
4.4. Configuración del servidor Oracle GlassFish Server.....	103
4.5. Carga de datos a la base	104
4.6. Diagrama de despliegue	109
4.7. Resultados	109
4.7.1. Acceso al sistema	110
4.7.2. Módulo de registro de pasantía	110

CONCLUSIONES.....	119
RECOMENDACIONES.....	121
LISTA DE REFERENCIAS	122

ÍNDICE DE TABLAS

Tabla 1. Clasificación de Workflow por funciones	9
Tabla 2. Tipo de tareas	10
Tabla 3. Tipo de orden de actividades	11
Tabla 4. Tipos de Workflow según actividades	13
Tabla 5. Tipo de procesos	14
Tabla 6. Tipos de elementos	16
Tabla 7. Tipo de relaciones	16
Tabla 8. Tipos de diagrama.....	16
Tabla 9. Tipos de divisiones comunes	17
Tabla 10. Cuadro RF 1: Aplicativo JAVA para carga de datos a la base de datos	34
Tabla 11. Cuadro RF 2: Ingreso al SPPP por el estudiante.....	35
Tabla 12. Cuadro RF 3: Inicio del proceso por el estudiante.....	35
Tabla 13. Cuadro RF 4: Generar oficio para empresa.	36
Tabla 14. Cuadro RF 5: Generar carta de aceptación	36
Tabla 15. Cuadro RF 6: Generar la carta compromiso	37
Tabla 16. Cuadro RF 7: Generar código de resolución automáticamente	37
Tabla 17. Cuadro RF 8: Solicitud para inicio de actividades en la empresa.....	38
Tabla 18. Cuadro RF 9: Guía o sección de ayuda.....	38
Tabla 19. Cuadro RF 10: Estado del flujo.....	38
Tabla 20. Cuadro RF 11: Subir documentos al Sistema	39
Tabla 21. Cuadro RF 12: Envío de notificaciones a estudiantes vía email.....	39
Tabla 22. Cuadro RF 13: Campo de observaciones.....	40
Tabla 23. Cuadro RF 14: Validación de documentos subidos por el alumno.....	40
Tabla 24. Cuadro RF 15: Validación de información mediante checkbox	40
Tabla 25. Cuadro RNF 1: Portabilidad.	41
Tabla 26. Cuadro RNF 2: Interfaz gráfica del usuario GUI.....	41
Tabla 27. Cuadro RNF 3: Sistema Responsive. (Achig & Torres, 2018).....	41
Tabla 28. Cuadro RNF 4: Mensajes de error en el sistema (POP-UP). (Achig & Torres, 2018)	42
Tabla 29. Cuadro RNF 5: Tiempo de respuesta. (Achig & Torres, 2018).....	42
Tabla 30. Cuadro RNF 6: Seguridad. (Achig & Torres, 2018).....	42
Tabla 31. Cuadro RNF 7: Concurrencia	42
Tabla 32. Cuadro RNF 8: Disponibilidad. (Achig & Torres, 2018)	43
Tabla 33. Cuadro de la tabla datos	61
Tabla 34. Cuadro de la tabla detalle pasantía.....	61
Tabla 35. Cuadro de la tabla empresa	62
Tabla 36. Cuadro de la tabla encargado de la empresa	62
Tabla 37. Cuadro de la tabla estudiante	62
Tabla 38. Cuadro de la tabla formato.....	63
Tabla 39. Cuadro de la tabla pasantía	63
Tabla 40. Cuadro de la tabla perfil.....	64

Tabla 41. Cuadro de la tabla periodo	64
Tabla 42. Cuadro de la tabla preguntas	65
Tabla 43. Cuadro de la tabla proceso	65
Tabla 44. Cuadro de la tabla respuesta.....	65
Tabla 45. Cuadro de la tabla tipo pregunta	65
Tabla 46. Cuadro de la tabla tutor	66
Tabla 47. Cuadro de la tabla usuario.....	66
Tabla 48. Paquete com.sppp.beans	80
Tabla 49. Paquete: com.sppp.DAO.....	82
Tabla 50. Paquete: com.sppp.utils.....	83
Tabla 51. Paquete: com.sppp.mailing	83
Tabla 52. Paquete: com.sppp.classes	84
Tabla 53. Paquete: com.sppp.servlets	84
Tabla 54. Web.xml.....	85
Tabla 55. Gestor de base de datos	85
Tabla 56. Configuración de mapeo	86
Tabla 57. Enrutar perfil	86
Tabla 58. Analizar pertenencia de path.....	87
Tabla 59. Ruteo a página principal del perfil.....	87
Tabla 60. Configuración servlet para archivos	87
Tabla 61. Obtener un File de un Path.....	88
Tabla 62. Descargar un archivo	88
Tabla 63. Envío email	89
Tabla 64. Guardar objeto en base de datos	89
Tabla 65. Recuperar datos de la base como objetos.....	91
Tabla 66. Métodos Clase DashboardGestBean.....	92
Tabla 67. Métodos Clase DetalleEstudianteBean	92
Tabla 68. Métodos Clase WizardCCValidacion	92
Tabla 69. Métodos Clase Mailing	93
Tabla 70. Estadísticas JMeter 50 Usuarios	93
Tabla 71. Estadísticas JMeter 100 Usuarios	95
Tabla 72. Estadísticas JMeter 300 Usuarios	97
Tabla 73. Prueba contra requerimiento gestor	98
Tabla 74. Prueba contra requerimiento estudiante.....	99
Tabla 75. Prueba contra requerimiento administrador.....	100
Tabla 76. Configuración de la base	101
Tabla 77. Características del hardware	102
Tabla 78. Características del software	103
Tabla 79. Campos Excel	104
Tabla 80. Campos Excel	106

ÍNDICE DE FIGURAS

Figura 1. Tendencia de Sistema.	7
Figura 2. Actividades y artefactos.....	19
Figura 3. Workflow de registro pasantía.....	30
Figura 4. Caso de uso módulo carga información.....	44
Figura 5. Caso de uso módulo de registro pasantía.....	45
Figura 6. Caso de uso módulo validación documentos.....	45
Figura 7. Esquema de clases principales.....	46
Figura 8. Diagrama conceptual de base de datos.	47
Figura 9. Ventana global para todos los perfiles.....	48
Figura 10. Proceso pasantía y datos empresa.....	49
Figura 11. Documentación requerida.....	49
Figura 12. Llenar carta de compromiso.	50
Figura 13. Solicitud inicio actividades en empresa.....	50
Figura 14. Lista solicitudes inicio proceso.....	51
Figura 15. Revisa y valida la información.	52
Figura 16. Listado pendiente carta compromiso.	52
Figura 17. Validar carta compromiso.....	53
Figura 18. Listado pendiente documentos.	53
Figura 19. Validar documentos PDF.....	54
Figura 20. Aprobación inicio actividades.	54
Figura 21. Modelo físico BD.	60
Figura 22. Login.....	67
Figura 23. Login.....	67
Figura 24. Inicio proceso.....	68
Figura 25. Inicio proceso.....	68
Figura 26. Carta.....	69
Figura 27. Carta.....	69
Figura 28. Dashboard.	71
Figura 29. Dashboard.	71
Figura 30. Descarga.	72
Figura 31. Descarga.	72
Figura 32. Colores.....	73
Figura 33. Indicador.	73
Figura 34. Vistos.	74
Figura 35. Vistos.	74
Figura 36. Vistos.	74
Figura 37. Enviar.....	74
Figura 38. Gestor.....	76
Figura 39. Gestor.....	76
Figura 40. Gestor.....	76
Figura 41. Descarga.	77

Figura 42. Validar.	77
Figura 43. Validar proceso.	78
Figura 44. lista proceso individual.	78
Figura 45. Diagrama de paquetes.	79
Figura 46. Diagrama de clases.	80
Figura 47. Estadísticas por cada solicitud.	94
Figura 48. Tiempo de Respuesta.	94
Figura 49. Estadísticas por cada solicitud.	95
Figura 50. Tiempo de Respuesta.	96
Figura 51. Estadísticas por cada solicitud.	97
Figura 52. Tiempo de Respuesta.	98
Figura 53. Carga Excel.	105
Figura 54. Formato.	106
Figura 55. Campos.	107
Figura 56. Formato.	107
Fuente 57. Archivos.	108
Figura 58. Programa S3P Cargar BD.	108
Figura 59. Diagrama de despliegue.	109
Figura 60. Pantalla de acceso.	110
Figura 61. Dashboard estudiante.	111
Figura 62. Formulario de ingreso de datos.	111
Figura 63. Pantalla informativa.	112
Figura 64. Errores de campos.	113
Figura 65. Descargar/Subir documentos.	113
Figura 66. Subir documentos corregidos.	114
Figura 67. Dashboard gestor.	114
Figura 68. Validar formularios.	115
Figura 69. Tablas dinámicas.	116
Figura 70. Validar documentos.	116
Figura 71. Validar inicio pasantía.	117
Figura 72. Buscador.	118

Resumen

En el presente documento se presenta en detalle las herramientas, el desarrollo e implantación de una aplicación Web, que permite el registro de pasantías o prácticas preprofesionales a los alumnos de la Universidad Politécnica Salesiana Campus Sur de la carrera de Ingeniería de Sistemas. Se desea satisfacer las necesidades de alumnos y docentes encargados del proceso, disminuir el tiempo de respuesta entre los procesos manuales que se manejan.

Las tecnologías que han permitido llevar a cabo la problemática mencionada con anterioridad son NetBeans, Hibernate, JSF, HQL, JavaScript entre otras, las cuales se las detalla en el documento.

El proceso se basa en un workflow, donde a través de diversas etapas el estudiante ingresa información para cada proceso, mientras que un gestor será el encargado de validarla, una vez aprobados todos los datos ingresados por el estudiante para cada etapa, el estudiante puede proseguir a la siguiente etapa, en caso de que uno de los datos haya sido negado, el estudiante tiene la tarea de corregir el dato erróneo y volver a ingresarlo hasta su posterior validación.

El sistema web ha sido diseñado de acorde a los requisitos solicitados en las entrevistas a los clientes, cumpliendo todos los requisitos funcionales y no funcionales, validados en diversas pruebas realizadas al sistema, con las cuales se verificó que el sistema cumple con normas de calidad que permitirán su fácil proceso de interacción con diversos tipos de usuarios y su ligera interfaz responsive adaptable a la pantalla desde donde se lo quiera usar.

Abstract

In this document we present in detail the tools, development and implementation of a Web application, which allows the registration of internships or preprofessional practices to the students of the Universidad Politécnica Salesiana Campus Sur of the Systems Engineering degree.

It is desired to satisfy the needs among the students and the teachers in charge of the process, to reduce the response time between the manual processes that are handled.

The technologies that have allowed to carry out the problems are NetBeans, Hibernate, JSF, HQL, JavaScript among others, which are detailed in the document.

The process is based on a workflow, through which through various stages the student enters the necessary information for each process, while a manager will be responsible for validating the information entered by the student, once the manager has approved all the data entered by the student for that stage, the student can proceed to the next stage, in case one of the data has been denied, the student has the task of correcting the wrong data and re-enter it until its subsequent validation.

The web system has been designed according to the requirements requested in the interviews with the clients, fulfilling all the functional and non-functional requirements, validated in various tests carried out on the system, with which it was verified that the system complies with quality standards allowing an easy process of interaction with different types of users and their light responsive interface adaptable to the screen from where they want to use it.

INTRODUCCIÓN

Esta sección contiene una revisión de proyectos similares, los cuales ayudan a la comprensión del tema y dan una idea clara, con el fin de usar las mejores prácticas y propuestas que otras personas han usado en la construcción y desarrollo. Adicional se cuenta con los objetivos planteados, los cuales serán cumplidos a cabalidad.

Antecedentes

Hoy en día el mundo es cambiante y es esa diversidad de situaciones, etapas, momentos que cada día lo hace interesante y a su vez se vuelve más desafiante en cada una de las etapas de vida, es así como avanza la tecnología apuntando a mejorar el estilo de vida la cual intenta hacer de lo difícil y lo complejo algo más simplificado. Desde este punto de vista es como nace una propuesta de minimizar los procesos que se realizan a mano, y las confusiones que a veces evitan que fluya de mejor manera.

Los docentes y estudiantes de la Universidad Politécnica Salesiana Campus Sur de la carrera de Ingeniería de Sistemas necesitan un sistema informático que sea capaz de agilizar el proceso de pasantías y prácticas pre profesionales, mejorar su tiempo de respuesta, facilite la forma en cómo se lo maneja y permita entender las tareas a realizar en cada una de sus etapas.

Es así como empieza un arduo trabajo de búsqueda con el fin de encontrar el método o la forma óptima para crear una aplicación Web, basados en los trabajos que se presenta a continuación, tomando las ideas relevantes, las mejores propuestas es como se ha desarrollado la aplicación Web.

El BPM que se aplica a este proceso de prácticas preprofesionales ha permitido tener un mejor enfoque al momento de automatizar el proceso, ya que se sigue un flujo de procesos, las cuales permiten analizar cada una de las tareas que se llevan a cabo en dicha etapa. (Campoverde Rengel & Torres Basantes, 2016).

La administración el control y seguimiento que propone este trabajo, da una idea clara de cómo sintetizar y minimizar el control de un proceso. (Mena Arce & Páez Estrella, 2011).

El trabajo que presentan permite una noción clara de la automatización informática de los procesos, donde el principal oponente es el cliente, debido a la resistencia al cambio. (Chasipanta Llulluna & Ramos Villacis, 2015).

El desarrollo de Workflow que propone este trabajo permite crear un enfoque sistemático del proceso manual, permitiendo transformar las etapas que sean convenientes en un sistema informático y dejar las partes o procesos que funcionan bien siendo manuales. (Becerra Mejora & Sosa Naranjo, 2011).

Problema

El registro de pasantías para los estudiantes de la carrera de Ingeniería de Sistemas es un proceso bastante demorado y difícil de comprender para los estudiantes ya que conlleva varios formatos de entrega y recepción tanto en la empresa como en la Universidad.

Al ser un proceso manual, éste es lento y puede llegar a presentar errores de algún documento, ya que pasa por varias personas encargadas del área de pasantías, generando pérdida de tiempo tanto para el estudiante como para el gestor de

revisar dicha documentación, hasta realizar varios cambios para ir ajustando la solicitud hasta que sea aceptada.

La pérdida de documentos es un problema habitual ya que se maneja varias solicitudes a la vez. Un solo papel que no se encuentre en la documentación puede causar el rechazo de la solicitud.

Otro inconveniente que se presenta al personal encargado es la gran cantidad de papeles con solicitudes e ir pasando a cada gestor de acuerdo con el avance de dicho proceso.

La Universidad Politécnica Salesiana al momento cuenta con varios formatos en Excel de cómo se debería realizar los pasos para aprobar dicha pasantía; de momento los formatos de Excel aún no llegan a implementarse por completo dado que es un cambio total de paradigma y la adaptación de este no es fácil para ninguno de los involucrados debido a la falta de conocimiento de los diversos formatos, como llenarlos y el orden en que estos deben ser entregados para su respectiva aprobación.

Justificación del tema

En vista a la problemática antes mencionada, se considera oportuno la creación de un sistema informático que automatice el proceso de pasantías de la carrera de Ingeniería de Sistemas de la UPS Campus Sur, con el fin de minimizar la documentación física y disminuir los costes en los recursos que son asignados para el proceso.

Con el sistema se busca satisfacer las necesidades de los docentes encargados del proceso, personal administrativo y alumnos, ya que, para todas las partes interesadas, el proceso será fácil de usar y rápido en respuesta o cambios que necesite realizar.

Cabe mencionar que se mejora la distribución de trabajo porque es más rápido de realizar, se aumenta la facilidad del proceso ya que se cuenta con todos los formatos de

la documentación que se debe presentar en cada etapa del proceso y se reduce el tiempo en el que se tramita la documentación de cada alumno al crear la solicitud de pasantías.

Objetivo general

Optimizar el proceso para la solicitud y registro de pasantías estudiantiles en la Carrera de Ingeniería de Sistemas – Sede Quito.

Objetivos específicos

Analizar el proceso usado para la generación, aprobación y seguimiento para la solicitud de pasantía.

Diseñar una aplicación Web que cumpla los requerimientos especificados previamente para el proyecto.

Desarrollar la aplicación Web utilizando herramientas de desarrollo basadas en Java.

Implementar la aplicación Web sobre la infraestructura de la Universidad.

Capítulo 1

1. Marco Metodológico

Esta sección presenta una investigación a profundidad de los temas que influirán en las siguientes etapas. Se menciona información relevante a como se basa el proyecto en su estructura, los medios a utilizar, las metodologías a investigar para obtener un resultado que favorezca las necesidades planteadas.

1.1. Workflow

Workflow es un conjunto de tareas a realizar en un orden específico, para conseguir una meta o un objetivo central del negocio, las diversas tareas deben ser completadas y satisfechas para mediante ese proceso seguir a la tarea siguiente y completar el flujo de trabajo. Se tomará como referencia la definición propia de la Workflow Management Coalition (WfMC).

WfMC, 1996:

“La automatización de procesos de negocios, en su totalidad o en parte, en función de cómo sus documentos, información o tareas son pasadas de un participante a otro para realizar su tarea de acuerdo a un conjunto de reglas”.

(Coalition, 1996).

Para una mejor comprensión de Workflow se debe entender el concepto de proceso de negocio:

Un proceso de negocio son varias actividades que se relacionan entre sí para obtener un resultado ya asignado o previamente definido. Cada proceso posee entradas, funciones y salidas.

Fundamentado en lo previamente escrito Workflow es un sistema que facilita la automatización de varios procesos de negocio definidos para modelar y gestionar los mismos con el objetivo de llegar a una meta decidida o previamente analizada por la empresa.

1.1.1. Historia

Se originó como una herramienta para implementar una mejoría en los tiempos y costos que abarcaban los diferentes procesos de los negocios, siempre asegurando o certificando que las tareas sean realizadas consistentemente para seguir mejorando la calidad de los diversos procesos.

Su crecimiento se dio conforme las tecnologías de la información iban avanzando, con la introducción de nuevas topologías de redes, el correo electrónico, siendo una de las primeras aproximaciones de como automatizar procesos de negocio, desde la tarea más simple hasta la tarea más compleja y vital dentro de la empresa, se basó en que las computadoras tenían la opción de realizar mejor ciertas tareas automáticamente como la búsqueda de información, vencimiento de plazos, cálculos matemáticos o el post proceso después de que una tarea se realizó correctamente para avanzar al siguiente paso en la secuencia correcta.

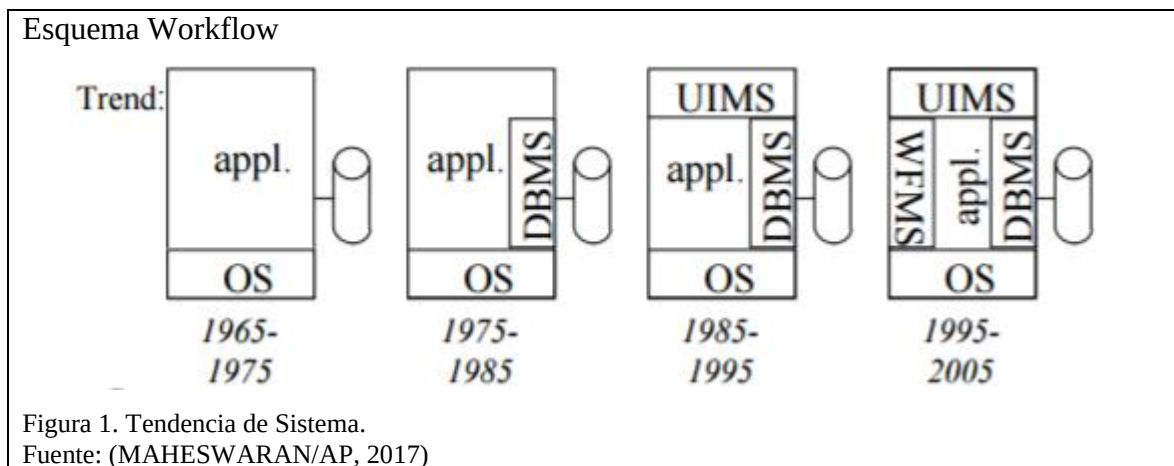
La tendencia a través del tiempo muestra como los sistemas se han ido adaptando a nuevos conceptos y prácticas con las tecnologías y métodos más actuales.

En los años 1965-1975 todo el sistema consistía en un sistema operativo y la aplicación.

En los años 1975-1985 al sistema se le incorpora una base de datos dentro de la capa de aplicación con lo cual se observa un nuevo sistema con mejores aplicativos.

En los años 1985-1995 se incluye una capa de interfaz para el usuario, con lo cual el usuario cuenta con una interacción del aplicativo.

En los años 1995-2005 la tendencia cambia nuevamente para incorporar un sistema de administración de workflow junto a la aplicación y la base de datos.



1.1.2. Características de un Workflow

Las características del Workflow son esenciales para su creación, modelamiento, ejecución e implementación, permite diferenciarlos de otros tipos de sistema de información. Las características son:

Tareas: Se deberá completar en su totalidad, no se puede dejar a medias y en caso de contener un error se deberá volver a realizarla desde el principio.

Personas: Las personas son las encargadas de realizar ciertas tareas mediante instrucciones secuenciales o un orden específico asignado. Pueden ser realizadas por agentes automatizadas asumiendo el papel de personas.

Herramientas: Son llamadas desde la aplicación Workflow, o invocadas por las personas. Las tareas no realizan ciertos procesos que son validadas o ayudadas por estas herramientas.

1.1.3. Administración Workflow

Un Workflow Management System – WFMS (Sistema de administración de Workflow) es un paquete software que puede ser usado para apoyar la definición, administración y ejecución de procesos del flujo de trabajo, apoya un set específico de procesos de negocio a través de la ejecución computarizada de procesos.

Soporta la integración de aplicaciones existentes, los sistemas de workflow se asegura una integración global de personas y programas en el framework de un proceso de negocio, durante este tiempo, los procesos de negocio son analizados y coordinados. (processmaker, 2000).

1.1.4. Funciones del Workflow:

El sistema de Workflow provee las siguientes funciones:

Asignación de tareas al personal.

Aviso al personal de tareas pendientes.

Colaboración de los empleados en tareas comunes.

Optimización de recursos alineándolos a la estrategia de la empresa.

Automatizar varias secuencias de los diversos procesos.

Agilizar un proceso y proveer un mejor servicio al cliente.

Establecer controles y seguimientos de procesos escogidos.

Se dispone también con funciones más generales y específicas:

Tabla 1. Clasificación de Workflow por funciones

Funciones		Descripción
En tiempo de construcción	de	Se lo utiliza para definir un proceso de negocio, se lo puede representar ya sea gráficamente o en un lenguaje semiformal
En tiempo de ejecución		Todo proceso es interpretado por el software responsable de controlar la operación del proceso o ejecutar una actividad específica
De interacción (humanos/aplicaciones)		Se basa en las operaciones humanas y las cuales se realizan en conjunto con herramientas computacionales, por lo cual este tipo de procesos deben soportar este tipo de interacción o incluir otras aplicaciones

Nota: Esta tabla contiene información acerca de los diversos workflows.

1.1.5. Ventajas Workflow:

Reducción de tiempo en tareas complejas.

Consistencia y confiabilidad.

Documentación de procesos.

Reducción de trámites o pasos innecesarios en los procesos.

Prevención de pérdida de información u alteración del orden.

Control total de todas las actividades según las personas asignadas a estas.

Recordatorios de la presencia de tareas nuevas o incompletas.

Garantía de que las actividades sigan el orden específico establecido.

Identificación de los recursos disponibles.

Evitar tiempos muertos o sin actividades

1.1.6. Conceptos de Base

Tarea

La definición de un workflow se la realiza a través de identificar las diversas tareas que lo componen, es una unidad lógica de trabajo. Una tarea se deberá completar en su totalidad, no se puede dejar a medias y en caso de contener un error se deberá volver a realizarla desde el principio.

Existen 3 tipos de tareas:

Tabla 2. Tipo de tareas

Tarea	Descripción
Manuales	Se lleva a cabo por una o más personas sin usar aplicaciones
Automáticas	El proceso es hecho por una aplicación o un programa, es llevada a cabo sin intervención alguna de personas
Semiautomáticas	Se da cuando una computadora realiza una tarea y es complementada por una persona

Nota: Esta tabla contiene información acerca de las diversas tareas.

Proceso

Abarca en si un conjunto de tareas para llegar a una meta o un objetivo, las cuales deben ser cumplidas en un orden específico, el orden de una tarea varía de acuerdo con sus características.

Se definen varios procesos y se tiene en cuenta procesos establecidos y definidos con anterioridad, es la unidad lógica de trabajo que es llevado como si fuera un todo por un solo recurso.

Ej.- Consultar saldo de una cuenta sin ingresar el número de la cuenta.

Orden

Las diversas actividades con las que se cuenta pueden ser ejecutadas siguiendo una secuencia ya establecida a través de ciertas ramas.

Se tiene cuatro instrucciones para describir el orden en que se ejecutaran actividades dentro de un proceso:

Tabla 3. Tipo de orden de actividades

Tipo de orden	Descripción
Secuencial	Son ejecutadas una tras otra las tareas
Paralelo	Dos tareas realizadas al mismo tiempo
Selectivo	Se debe tomar una decisión entre varias opciones
Iteración	Es necesario ejecutar la tarea más de una vez

Nota: Esta tabla contiene información acerca del orden de las actividades.

Roles

El rol distingue la capacidad que tienen las personas de manejar el sistema, diferentes roles incluyen diferentes funciones. Las personas pueden tener uno o más roles según como se haya diseñado y modelado el Workflow.

Rutas

Es el camino que seguirá la información o los pasos a seguir en los procesos. Es un elemento clave para que las tareas se ejecuten en el orden correcto. Se distinguen varios tipos de rutas como:

Rutas fijas: La información ira siempre por un único camino.

Rutas condicionales: Para elegir un camino se tomará en cuenta una condición a cumplir.

Rutas Ad-Hoc: El usuario toma la decisión de cual ruta se seguirá en el proceso.

Documentos-Data

Al momento de la ejecución del Workflow se realiza un intercambio de información o almacenamiento de esta, esta puede ser desde archivos, imágenes hasta llegar a incluir registros en una base de datos. Se los clasifica en:

Datos de Control: Se la utiliza para determinar el estado de una actividad o tarea.

Datos Relevantes: Datos analizados o generados por el sistema, los cuales decidirán la ruta a seguir en el proceso de negocio.

Eventos

Es la interrupción que se le da a un proceso, mediante el cual se informa al usuario las razones de esta interrupción, puede ser disparada por el usuario voluntariamente o estar implícita. Una interrupción puede causar que se siga una ruta distinta en el proceso.

Plazos

Son tiempos previamente definidos para ciertas actividades o tareas, a las cuales se les puede asignar un evento en caso de que cumpla el plazo o el mismo expire para generar una ruta distinta en el proceso según la evaluación previamente tomada.

Políticas

Indican como deben llevarse a cabo las decisiones y bajo qué parámetros tomarlas. En caso de no acatar las políticas queda bajo la responsabilidad de la persona las acciones que realice, así como posibles sanciones.

1.1.7. Tipos de sistemas workflow

Dada la inmensa variedad de negocios, el Workflow se ha dividido en varios tipos según el valor del proceso a manejar. A continuación, se lo presenta según el tipo de actividades y su redundancia en el proceso:

Tabla 4. Tipos de Workflow según actividades

Workflow	Descripción
Workflow de producción	Se las llama así a todas las tareas que se ejecuten repetidamente y no varíen en el transcurso de tiempo, generalmente ya están definidas. Tienden a estar bien estructuradas y manejan gran cantidad de datos
Workflow de colaboración	Se transmite información o documentos a personas para que estas realicen actividades sobre ellas, donde las personas estructuran el proceso del negocio con lo cual el tratamiento del documento puede conllevar a diversas rutas
Workflow administrativo	Se encarga de medir las actividades o procesos del negocio, mediante la generación de reportes.

Nota: Esta tabla contiene información acerca de las actividades en workflows.

De acuerdo con el tipo de procesos que ejecutan:

Tabla 5. Tipo de procesos

Procesos	Descripción
Sistemas basados en imágenes	Permiten controlar el flujo de documentos, archivos en un negocio mediante la transformación de documentos escritos en imágenes.
Sistemas basados en documentos	Se controla el flujo de formas de un negocio con documentos, los cuales deben describir un giro empresarial, estos sistemas pueden notificar cuando la actividad se ha ejecutado
Sistemas basados en coordinación	Se basan en los aspectos que posee una relación con coordinación de un aspecto humano en el proceso de negocio

Nota: Esta tabla contiene información acerca de los diversos procesos de sistemas.

1.2. Lenguaje de Modelado Unificado (UML)

UML es un lenguaje de modelado o notación gráfica, basado en los métodos aplicados para expresar un diseño al desarrollo de toda clase de sistemas. (Fowler, 1999). Utilizado para visualizar, especificar, construir y documentar todos los componentes de un sistema de software orientado a objetos. Aplicado a sistemas de grandes empresas, hasta aplicaciones de servicios Web.

Acorde al proyecto que se va a realizar, se lo describe como un sistema complejo, el cual requiere abarcar distintas vistas de la arquitectura adaptándolo a la evolución a lo largo del ciclo de vida de su desarrollo, nos indicar como crear y leer modelos bien formados.

Dichas vistas sirven como ayuda o guía para visualizar o hacer más formal su arquitectura, siendo accesible para otros, donde UML no influye o está asociado a la metodología de desarrollo y es independiente del proceso.

Los procesos de la empresa deberán ser maduros ‘bien definidos’, el cual orienta de una forma clara qué componentes son necesarios crear, así como los procesos a llevar a cabo y el personal que los gestionará. (Martin, Fowler, & Kendall, 1997).

El éxito de un UML:

Fácil de aprender.

Tiene aspectos y características bien definidas que pueden ser verificados y comprobados, en caso de realizar un análisis o validación a profundidad de este, para determinar su madurez. En síntesis, es posible generar un código de alta calidad partiendo de un UML maduro.

Es legible para la lectura a las personas, mientras que los computadores pueden mostrar su estructura de forma sencilla, dando lugar a un sin número de herramientas que soportan este lenguaje.

La notación UML es grafica con varios tipos de diagramas divididos en cuatro categorías:

Diagramas estructurales.

Funcionales.

Interacción.

Comportamiento.

En UML se puede modelar software o sistemas como una colección o varias colecciones de objetos que interactúan entre sí. A continuación, se presenta la estructura básica de UML.

1.2.1. Bloques de construcción básicos

Los elementos con lo que se construye los distintos modelos con sus relaciones y diagramas.

Elementos: componentes del modelado.

Tabla 6. Tipos de elementos

Tipos	Descripción
Elementos estructurales	Se definen como una clase, interfaz, caso de uso, componente o nodo.
Elementos de comportamiento	Compuesto por interacciones, actividades.
Elementos de agrupación	Agrupar los elementos semánticamente relacionados en paquetes.
Elementos de anotación	Captura información que se puede anexar al modelo.

Nota: Esta tabla contiene información acerca de tipos de elementos de modelado.

Relaciones: especifican la relación semántica entre dos o más elementos.

Tabla 7. Tipo de relaciones

Tipos	Descripción
Dependencia	Indica que el elemento de origen tiene un elemento destino, y depende de éste, cualquier cambio en el elemento destino puede afectar al primero.
Asociación	Es como se vincula un objeto con otro.
Contención	El elemento origen contiene al elemento destino.
Realización	El elemento origen garantiza llevar a cabo lo especificado por el elemento destino.

Nota: Esta tabla contiene información acerca de los tipos de relaciones en modelos.

Diagramas: son una colección de elementos que permiten visualizar lo que hará el sistema, son vistas o ventanas distintas del modelo.

Tabla 8. Tipos de diagrama

Tipos	Descripción	Subtipos
Diagramas de	Usan un modelado	• Diagrama de clase.

estructura	estático, describiendo que tipos de objetos se reaccionan entre sí.	• Diagrama de componentes. • Diagrama de estructura compuesta. • Diagrama de objetos. • Diagrama de paquete. • Diagrama de despliegue.
Diagramas de comportamiento	Describe la interacción entre objetos para conseguir una funcionalidad.	• Diagrama de actividad. • Diagrama de interacción. • Diagrama de casos de uso. • Diagrama máquina de estado.

Nota: Esta tabla contiene información acerca de los diversos tipos de diagramas.

1.2.2. Mecanismos comunes

La capacidad de conseguir ciertos objetivos específicos aplicados en forma repetida en diferentes contextos.

1.2.3. Especificaciones

Son descripciones textuales de la semántica de un elemento, formando su esqueleto semántico y dotándolo de sentido.

1.2.4. Adornos

Todo elemento del modelado tiene un símbolo asociado al que se puede añadir un sin número de adornos para hacer visible algún aspecto.

1.2.5. Divisiones comunes

Son formas particulares de pensar sobre el mundo.

Tabla 9. Tipos de divisiones comunes

Tipos		Descripción
Clasificador	/	La idea o noción abstracta de un tipo de elemento es

instancia	el clasificador y los elementos específicos son la instancia.
Interfaz / implementación	Separar que hace un elemento de como lo hace.

Nota: Esta tabla contiene información acerca de la noción abstracta de un elemento.

1.3. Scrum

1.3.1. Definición

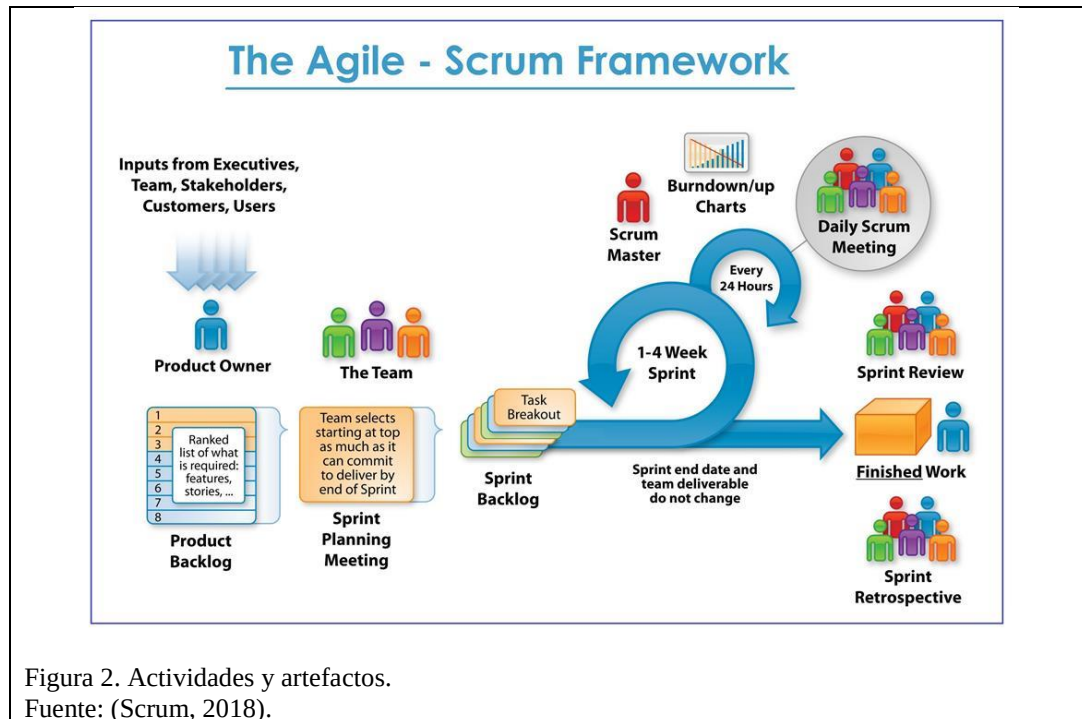
Es una metodología ágil para para gestionar el desarrollo de software y flexible en adaptación, auto gestión e innovación, donde las personas pueden emplear problemas complejos adaptativos.

Es un marco de trabajo que emplea varias técnicas y procesos, su marco de trabajo consiste en equipos, roles, eventos, artefactos y reglas las cuales están asociadas a cada evento de trabajo.

“El enfoque de Scrum es iterativo e incremental con el fin de optimizar el requerimiento y controlar el riesgo”. (Sutherland, Julio 2013) .

El control de los procesos es empírico, donde el conocimiento procede de la experiencia y a su vez la toma de decisiones basadas en lo que se conoce, desglosándose en tres pilares.

Scrum roles



Transparencia: el proceso debe ser claro, tener una visión completa del alcance del proyecto.

Inspección: aquí se visualiza el avance hacia un objetivo, puede darse el caso de que exista variaciones las cuales deberán ser alineadas al proyecto, pero la inspección será acordada en citas específicas ya que no deben interferir en el trabajo.

Adaptación: si el proceso se desvía de los deberá ser reajustado cuanto antes para evitar desviaciones mayores. Los eventos formales de Scrum dentro del Sprint con el fin de realizar la inspección y adaptación son:

1.3.2. Eventos de Scrum

El objetivo es minimizar las reuniones no definidas, dichos eventos tienen una duración máxima, por ende, el inicio y finalización de un Sprint no puede

acortarse o alargarse. Cada hito de Scrum es una oportunidad formal para inspeccionar o adaptar algún aspecto.

Sprint: es un bloque de tiempo con duración máxima de 4 semanas para presentar un hito, el Sprint está formado por:

Planificación del sprint.

Los scrum diarios.

La revisión del sprint.

Retrospectiva del sprint.

Cancelación de un Sprint: puede cancelarse antes que el bloque de tiempo culmine, donde el dueño del proyecto tiene la oportunidad de cancelar el sprint o a su vez bajo la influencia de los interesados. La cancelación de este consume recursos.

Reunión de planificación de sprint: aquí se planifica todo el trabajo a realizar de cada sprint, con una duración máxima de ocho horas, donde los asistentes deberán entender el propósito al que se quiere llegar.

Scrum Diario: reunión de quince minutos para sincronizar las actividades del equipo de desarrollo, inspeccionando el trabajo realizado desde el último scrum, haciendo una proyección. También se evalúa el progreso hacia el objetivo del sprint.

Revisión del Sprint: los interesados y el equipo de desarrollo colaboran acerca de lo que se llevó a cabo con el fin de determinar los próximos aspectos a realizar.

Retrospectiva del Sprint: reunión mínima de tres horas, sirve para inspeccionar al equipo para crear un plan de mejoras que pueden ser adaptadas al siguiente sprint.

1.3.3. El equipo scrum

Dueño del producto: responsable que maximiza el trabajo del equipo de desarrollo.

A su vez gestiona la lista del producto (Product backlog), el cual consta de:

- Ordenar los objetivos y misiones de la mejor manera posible.

- Optimizar el valor del trabajo del equipo de desarrollo.

- Asegura que la lista es entendible para todos y va enfocada al próximo trabajo que el equipo de desarrollo debe realizar.

Equipo de desarrollo: son aquellos profesionales que realizan el avance del proyecto, con el fin de ponerlo en producción al final de cada sprint. Cada uno de los integrantes cuenta con habilidades necesarias para continuar con el alcance del proyecto.

Scrum master: se asegura que el equipo de desarrollo trabaja ajustándose a la teoría, prácticas y reglas de scrum. A su vez ayuda a las personas externas al

equipo a entender las interacciones que pueden ser de ayuda o no con el equipo de desarrollo. (Schwaber, k, & Beedle, 2002).

1.3.4. Artefactos de scrum

Lista de producto: lista ordenada de lo que podría ser necesario en el producto, evoluciona a medida que el producto y su entorno también lo hacen, es decir, es una lista dinámica.

Consta de todas las características, funcionalidades, requisitos mejoras y correcciones sobre el producto. Los elementos iniciales de la lista son los que están más claros, mientras que los últimos aún faltan por pulir.

Lista de pendientes: es un conjunto de elementos de la lista del producto, es una predicción hecha por el equipo de desarrollo sobre las futuras funcionalidades a ser entregables. El objetivo de la lista es hacer visible el trabajo necesario para alcanzar el objetivo.

1.4. JSF

Es un componente basado en un framework MVC, el cual está construido en lo más alto de Servlet API y provee componentes vía taglib, los cuales pueden ser usados en JSP o cualquier tecnología con base Java tales como Facelets. Los Facelets son más cómodos y prácticos de usar con JSF que con JSP ya que provee de componentes adicionales compuestos, para realizar el modelado de un HTML sin redundancia en el código, mientras que JSP solo provee <jsp:include> por lo cual el usuario se ve forzado a crear sus propios componentes a su manera en Raw Java. Desde la incorporación de JSF 2.0, se ha declarado JSP como obsoleto para usar con facelets.

Como es un MVC (Modelo – Vista- Controlador) framework, JSF provee los FacesServlet como el único controlador para los pedidos-respuestas (request-response). Toma todo los estándares y tediosos HTTP pedidos/respuestas y los maneja junto a entradas de usuarios (inputs), validando/convirtiendo, poniéndolos en objetos del modelo, invocando a acciones y renderizando la respuesta. (Oracle, Introduction to Javaserer Faces - What is JSF?, 2005).

De esta manera se determina básicamente con un JSP o Facelets (Xhtml) páginas para la vista y una clase JavaBean como modelo. Los componentes JSF son usados para juntar la vista con el modelo y el FacesServlet usa el componente JSF a modo de un árbol con raíces para hacer el resto del trabajo.

Al momento de trabajar con JSF, se termina con una página XHTML como vista y una clase JavaBean como modelo y esto incrementa velozmente el desarrollo. El verdadero poder de Java Server Faces está en su modelo de componente interfaz-usuario, donde las aplicaciones están plenamente construidas desde colecciones de componentes que pueden renderizarse en diferentes maneras para múltiples tipos de clientes. La UI de JSF ofrece a los desarrolladores el construir interfaces de usuario usando componentes preconstruidos en vez de iniciarlos desde cero. Los componentes vienen en varias formas desde las más simples hasta las más complejas como dataTable los cuales pueden representar un conjunto de datos, ya sean estáticos o dinámicos desde una base de datos. (IA, 2014).

1.5. Pasantía

Breve historia, las nuevas modificaciones legislativas en el año de 1992 que fueron orientadas a flexibilizar las relaciones laborales, crearon el Sistema Nacional de

Pasantías Educativas en base al decreto 340/92. Donde se define a la pasantía a la extensión orgánica del sistema educativo a instituciones de carácter público o privado para la realización por parte de los alumnos y docentes de prácticas relacionadas con su educación y formación. (Adamini, 2011)

Según el Art. 19 del reglamento de vinculación con la colectividad.

Se determinan PASANTÍAS, las que efectúen los estudiantes de la Universidad Politécnica Salesiana, en industrias, empresas e instituciones afines con la carrera cursada, para aplicar, fortalecer y retroalimentar los conocimientos, técnicas y desarrollar destrezas en el campo laboral.

Las responsabilidades de las instituciones, empresas y/o industrias estarán expresadas en el convenio específico. La pasantía no significa compromiso laboral alguno, y no genera obligaciones contractuales de ninguna naturaleza entre la universidad, el estudiante y la empresa.

En los artículos 23 al 30 se menciona que la duración de pasantías será establecida por el consejo académico de la facultad, la interrupción injustificada de la misma se considera abandono y su plazo máximo hasta aprobar la pasantía es de un año. (SOCIEDAD, 2012).

1.5.1. Las responsabilidades

La Universidad Politécnica Salesiana entre los artículos 27 al 29 está obligada a formalizar los acuerdos con las instituciones, industrias y/o empresa, realizar jornadas de preparación para los estudiantes previo a la realización de la pasantía

a su vez designa tutores que dan seguimiento y evalúan dicha actividad. A su vez menciona que no significa compromiso laborar alguno.

El tutor entre los artículos 30 y 31 es seleccionado por Dirección de Carrera, es responsable del seguimiento. Vela por lo establecido en el convenio correspondiente, supervisa el desarrollo del pasante evaluando y emitiendo informes a las instancias correspondientes.

Los estudiantes en el artículo 32 deberán cumplir las actividades presentando un informe semanal de las actividades desarrolladas al tutor con el visto bueno del responsable directo en la empresa. (SOCIEDAD, 2012).

1.5.2. Evaluación y acreditación

Artículo 33 se evaluará de acorde al formato de evaluación del tutor y del representante de la empresa donde se realiza la pasantía y una autoevaluación del estudiante.

Artículo 34 es estudiante presenta un informe de la actividad asignada emitido por el representante legal de la empresa, un informe del profesor tutor y el informe personal de actividades.

1.5.3. Prácticas preprofesionales

Según el Art. 35 del reglamento de vinculación con la colectividad.

Las prácticas preprofesionales son espacios académicos establecidos en las propuestas curriculares de las carreras para fortalecer la formación profesional y humana de los estudiantes. Las prácticas preprofesionales no significan

compromiso laboral alguno y no generan obligaciones contractuales de ninguna naturaleza entre la universidad, el estudiante y los centros de prácticas.

1.5.4. Las responsabilidades

La Universidad Politécnica Salesiana de acorde al artículo 39 designa un tutor responsable, establece criterios para la selección de centros, instituciones para las prácticas. Elabora instrumentos de seguimiento y evaluación de los estudiantes para someterlos al Consejo Académico.

El profesor responsable según el artículo 41 realiza la planificación semestral para presentar informes periódicos o semestrales sobre su desarrollo, organiza reuniones periódicas con los docentes supervisores para enviar la información necesaria a la Coordinación de Vinculación con la Colectividad para la emisión de certificados de aprobación.

El supervisor según el artículo 42 asesora y anima a los estudiantes visitando los centros de prácticas para calificar lo realizado por el estudiante con el fin de elaborar un informe sobre el desarrollo y cumplimiento.

El estudiante según el artículo 43 asistir puntualmente a los centros de práctica, informar constantemente sobre su desarrollo y entregar los documentos e informes exigidos por la Carrera. (Salesiana, 2012).

1.5.5. Evaluación y acreditación

Para la evaluación y acreditación de las prácticas preprofesionales se remitirán a los instructivos elaborados por cada Carrera.

1.5.6. Ley Educación Superior Art. 87

Como requisito previo a la obtención del título, los y las estudiantes deberán acreditar servicios a la comunidad mediante prácticas o pasantías pre profesionales, debidamente monitoreadas, en los campos de su especialidad, de conformidad con los lineamientos generales definidos por el Consejo de Educación Superior. Dichas actividades se realizarán en coordinación con organizaciones comunitarias, empresas e instituciones públicas y privadas relacionadas con la respectiva especialidad. (CEMLAD, 2013).

1.6. XP

Conocida como programación extrema, propuesto por Kent Beck en 1999. Se caracteriza por su manera de trabajar, XP no es a plazo fijo y cuenta con historias de usuario las cuales van evidenciando los cambios realizados en la realimentación continua entre los desarrolladores y el cliente.

Fomenta el trabajo en equipo y el co-aprendizaje conocido como programación por parejas en el mismo ordenador, está sometida un alto índice de cambios por su forma de trabajo en los cuales se tiene requisitos imprecisos, y se lo recomienda su aplicación en proyectos de corto plazo (Beck, 2001).

Capítulo 2

2. Análisis y Diseño

De acorde a la recolección de datos, bajo los antecedentes y propuestas, entre otros factores que influyen para obtener datos del análisis del sistema Web, con la finalidad de entender las necesidades y crear un diseño enfocado a las problemáticas planteados donde se obtiene los requerimientos funcionales y no funcionales. En esta sección se aprecia la estructura del Workflow, como base del análisis realizado, el cual describe las etapas de cada proceso y las tareas que puede haber. La siguiente presenta los casos de uso por cada módulo, el diagrama conceptual de la base,

2.1.Análisis

2.1.1. Introducción

A continuación, se presenta el documento de especificación de requerimientos de software, el cual contiene el módulo de carga de información, módulo de registro de pasantía y módulo de validación de documentación, que forman parte del sistema de pasantías y prácticas pre profesionales y depende de los módulos de visitas y módulo de validación de pasantías que están realizados por la tesis de Karen Achig y Bryan Torres (Achig & Torres, 2018).

2.1.2. Propósito

Desarrollar e implementar el sistema de pasantías y prácticas pre profesionales versión 1.0, de la Universidad Politécnica Salesiana Campus Sur para la carrera de Ingeniería de Sistemas, para agilizar su tiempo en respuesta disponible en cualquier instante.

2.1.3. Alcance

Mediante el uso de las tecnologías de la información diseñar una plataforma Web para mejorar el control y manejo de pasantías y prácticas pre profesionales, donde los usuarios involucrados controlen el estado del proceso como: requisitos, validaciones y aprobaciones de requerimientos establecidos bajo los criterios que manejan para la continuidad del proceso.

2.1.4. Audiencia

Todos los involucrados tienen relación directa o indirecta e influyen con el proceso del desarrollo del sistema, algunos involucrados son los desarrolladores del sistema, el coordinador del ciclo de vida del sistema, los docentes encargados de administrar el sistema, administrativos, encargados del mantenimiento del sistema y alumnos.

2.1.5. Definiciones, acrónimos y abreviaturas

SPPP

Sistema de pasantías y prácticas pre profesionales.

GUI

Graphical user interface.

UPS

Universidad Politécnica Salesiana.

2.1.6.3.Funciones de producto

Permitir el ingreso de datos a los diversos formatos en todas las etapas para el inicio de la pasantía o práctica pre profesional.

Enviar notificaciones automáticas a estudiantes/administrativos según la operación realizada.

Mostrar el estado del proceso en el cual se encuentre el proceso de cada estudiante.

Almacenar en los repositorios los documentos subidos por el estudiante de su proceso de pasantía o práctica pre profesional

Generar un código único de acorde al proceso que el estudiante este realizando

Mostrar al administrativo/docente la información llenada por los estudiantes en los formatos para su respectiva aprobación/negación.

2.1.6.4.Tipo de usuario y características

Estudiante

Ingresar al sistema.

Llenar los formularios necesarios para su proceso de pasantía o práctica pre profesional. Recibir notificaciones según el estado de su proceso.

Subir documentos de los formatos aprobados

Administrativo, Docente o Gestor.

Ingresar al sistema.

Aprobar o negar los formatos llenados por los estudiantes.

Verificar los estados en que se encuentre el estudiante.

Ver informes y documentos del estado de la pasantía o práctica pre profesional en consejo de carrera.

2.1.6.5. Ambiente de operación

El SPPP operará en un servidor Windows establecido en la Universidad Politécnica Salesiana Campus Sur, deberá coexistir con una base de datos Postgresql y una aplicación paralela en Java.

2.1.6.6. Dependencia

El SPPP depende de la hoja de cálculo que contiene la información de todos los alumnos de la carrera de ingeniería de sistemas de la Universidad Politécnica Salesiana Campus Sur, dicho documento es entregado por parte de la dirección de carrera.

2.1.7. Requerimientos de interfaz externa

Interfaces de usuario:

La interfaz tendrá configuración *responsive*, la cual permite a la misma adaptarse a los diferentes dispositivos tanto móviles como de escritorio, con una interfaz limpia, clara e intuitiva.

Interfaces de hardware:

La compatibilidad del SPPP trabajara en arquitecturas de 32 y 64 bit.

Interfaces de software:

El SPPP funcionará en navegadores como Edge, Chrome, algunos componentes del SPPP variaran su conducta si se trabaja en otros navegadores.

Interfaces de comunicación:

El sistema estará diseñado para operar bajo el protocolo HTTP.

2.1.8. Módulos globales del producto

2.1.8.1. Módulo de carga de información

Este módulo se basa en un paso previo, se habilita a los alumnos pre seleccionados bajo un criterio interno de la Universidad Politécnica Salesiana la cual provee de un listado de los alumnos del periodo actual. El listado se ingresa en la base de datos del SPPP donde un gestor actualiza la base.

2.1.8.2. Módulo de registro de pasantía

Sigue un flujo que va de acorde al avance del estudiante en cada proceso. La aprobación de la documentación y requerimientos que realice un alumno es validada o rechazada por un gestor el cual valida dicha solicitud.

Ingreso a la plataforma de pasantías y prácticas pre profesionales por el estudiante.

Inicio del proceso de pasantías y prácticas pre profesionales por el estudiante.

Generar oficio para la empresa.

Generar carta de aceptación.

Generar la carta de compromiso.

Generar código de resolución automáticamente.

Solicitud para inicio de actividades en la empresa.

Guía o sección de ayuda.

Estado del flujo.

Subir documentos al sistema.

2.1.8.3. Módulo de validación de documentación

Se basa en cada etapa del flujo, cada proceso se valida de acorde a los pasos establecidos para validar una cierta información, el gestor aprueba o niega dicha información, la cual una vez sea aprobada avanza al siguiente proceso, hasta culminar el mismo.

Envío de notificaciones a estudiantes vía email.

Campo de observaciones.

Validación de documentos subidos por el alumno.

Validación de información mediante checkbox.

2.1.9. Requerimientos específicos

2.1.9.1. Requerimientos funcionales

Módulo de carga de información.

Cuadro RF 1: Aplicativo JAVA para carga de datos a la base de datos.

Tabla 10. Cuadro RF 1: Aplicativo JAVA para carga de datos a la base de datos

Descripción Una persona asignada por cuenta de la Carrera de Ingeniería de Sistemas deberá cargar un archivo CSV RF 3.1.8.1			
Entrada	Fuente	Salida	Restricción
Archivo CSV.	Aplicación en JAVA.	Carga de datos a en la base.	

Proceso	La persona asignada ingresa a la aplicación en JAVA, deberá ingresar el archivo CSV delimitado por comas de acorde a un formato pre establecido, donde se mencionan los campos necesarios para la correcta carga a la base.
---------	---

Nota: Esta tabla contiene información acerca del requisito funcional del aplicativo Java.

Módulo de registro de pasantía

Cuadro RF 2: Ingreso a la plataforma de pasantías y prácticas pre profesionales por el estudiante.

Tabla 11. Cuadro RF 2: Ingreso al SPPP por el estudiante

Descripción El estudiante tiene acceso para ingresar al sistema de pasantías y prácticas pre profesionales RF 3.1.8.2			
Entrada	Fuente	Salida	Restricción
Correo institucional. Clave predeterminada por el número de cédula.	Interfaz de Ingreso (Login)	Usuario con acceso al sistema.	No cumple los requisitos previos para aplicar al proceso de pasantías o prácticas pre profesionales, por ende, el sistema niega el acceso.
Proceso	El estudiante accede a la plataforma de pasantías y prácticas pre profesionales, en el módulo de ingreso registra sus credenciales, las cuales el sistema la comprueba para permitir su acceso.		

. Nota: Esta tabla contiene información acerca del ingreso al portal.

Cuadro RF 3: Inicio del proceso de pasantías y prácticas pre profesionales por el estudiante.

Tabla 12. Cuadro RF 3: Inicio del proceso por el estudiante

Descripción El estudiante ingresa los datos al formulario para el inicio de pasantías. RF 3.1.8.2			
Entrada	Fuente	Salida	Restricción

Datos alumno: Teléfono fijo, teléfono celular, actividad. Datos empresa: Nombre, RUC, persona a quien va dirigido el oficio, título profesional.	Formulario de solicitud de inicio de actividades.	Notificación de solicitud aprobada o negada.	No puede realizarse dos veces la misma solicitud, los datos deben ser llenados en su totalidad.
Proceso	El estudiante accede al formulario para el inicio de pasantías e ingresa toda la información solicitada en el mismo.		

Nota: Esta tabla contiene información acerca del inicio del proceso por el estudiante.

Cuadro RF 4: Generar oficio para empresa.

Tabla 13. Cuadro RF 4: Generar oficio para empresa.

Descripción Oficio generado automáticamente que será entregado en la empresa. RF 3.1.8.2			
Entrada	Fuente	Salida	Restricción
El gestor aprueba la solicitud de inicio de pasantías.	El sistema de pasantías y prácticas pre profesionales.	Oficio para la empresa.	
Proceso	Una vez que el gestor aprueba el formulario de inicio de pasantías se genera un oficio el cual será descargado por el estudiante en la plataforma.		

Nota: Esta tabla contiene información acerca de cómo se genera oficio para la empresa.

Cuadro RF 5: Generar carta de aceptación.

Tabla 14. Cuadro RF 5: Generar carta de aceptación

Descripción Carta generada automáticamente que será firmada y sellada en la empresa. RF 3.1.8.2			
Entrada	Fuente	Salida	Restricción
Oficio para la empresa.	Carta de aceptación.	Carta de aceptación firmada y sellada por la empresa.	
Proceso	La empresa firma y sella el formato que se encuentra en la plataforma de pasantías y prácticas pre profesionales.		

Nota: Esta tabla contiene información acerca de cómo se genera la carta de aceptación.

Cuadro RF 6: Generar la carta compromiso.

Tabla 15. Cuadro RF 6: Generar la carta compromiso

Descripción	La carta será generada automáticamente en la cual el estudiante deberá completar unos campos adicionales en la plataforma de pasantías y prácticas pre profesionales. RF 3.1.8.2		
Entrada	Fuente	Salida	Restricción
Actividad principal. Objeto de la actividad académica. Nombre programa. Actividades que desarrollar. Resultados de la actividad académica. Entregables de actividad académica.	Carta compromiso.	La carta de compromiso completa y su código único generado automáticamente.	
Proceso			

Nota: Esta tabla contiene información acerca de cómo se genera la carta de compromiso.

Cuadro RF 7: Generar código de resolución automáticamente.

Tabla 16. Cuadro RF 7: Generar código de resolución automáticamente

Descripción	Se crea el código al momento de que el estudiante solicita el inicio de actividades en la empresa. RF 3.1.8.2		
Entrada	Fuente	Salida	Restricción
Al aprobar carta de aceptación.	Solicitud de inicio de actividades generada por el alumno.	Se genera un código único de acorde a la actividad realizada, ejemplo 1234-123-12345-12-12.	No existirán códigos repetidos.

Proceso	Se consulta en la base de datos y de acorde al próximo registro se genera el código.
---------	--

Nota: Esta tabla contiene información acerca de la generación de código de resolución.

Cuadro RF 8: Solicitud para inicio de actividades en la empresa.

Tabla 17. Cuadro RF 8: Solicitud para inicio de actividades en la empresa

Descripción	Se genera automáticamente el formato solicitud resolución. RF 3.1.8.2		
Entrada	Fuente	Salida	Restricción
No aplica.	Formato de solicitud de resolución.	Formato de solicitud de resolución con los datos completos y un docente asignado.	
Proceso	El estudiante ingresa a la plataforma de pasantías y prácticas pre profesionales, genera la solicitud de inicio de actividades en la empresa, donde el sistema automáticamente genera un docente para que le realice las visitas, cabe mencionar que dicho docente es seleccionado tentativamente.		

Nota: Esta tabla contiene información acerca de la solicitud de inicio de actividades.

Cuadro RF 9: Guía o sección de ayuda.

Tabla 18. Cuadro RF 9: Guía o sección de ayuda

Descripción	Se detalla todos los pasos del procedimiento y preguntas frecuentes. RF 3.1.8.2		
Entrada	Fuente	Salida	Restricción
	Plataforma de pasantías y prácticas pre profesionales.	Ayuda complementaria en una pestaña de la plataforma.	
Proceso			

Nota: Esta tabla contiene información acerca de la guía de ayuda para usuarios.

Cuadro RF 10: Estado del flujo.

Tabla 19. Cuadro RF 10: Estado del flujo

Descripción	El gestor o el estudiante podrán observar en que parte del proceso se encuentra de manera individual. RF 3.1.8.2		
Entrada	Fuente	Salida	Restricción
Acorde al estado en el que se encuentra	Plataforma de pasantías y prácticas pre profesionales.	Presentación que indica tanto al estudiante como al gestor el estado actual en el que se encuentra.	No se puede volver a un estado anterior.

el estudiante.			
Proceso			

Nota: Esta tabla contiene información acerca de cómo será el estado de flujo.

Cuadro RF 11: Subir y descargar documentos al Sistema

Tabla 20. Cuadro RF 11: Subir documentos al Sistema

Descripción	El estudiante puede subir documentos que avalen el RF 3.1.8.2 cumplimiento de los diversos procesos y etapas		
Entrada	Fuente	Salida	Restricción
Documentos o formatos llenados y aprobados para la pasantía	Plataforma de pasantías y prácticas pre profesionales.	Documentos guardados en el servidor	Peso máximo del documento 2Mb
Proceso	Cuando el estudiante tenga todos los documentos necesarios para iniciar el proceso de pasantía aprobados, debe subir a la plataforma los documentos en formato PDF con un peso máximo por cada uno de 2Mb. Si el formato consta de 2 hojas las mismas deberán estar en un solo PDF		

Nota: Esta tabla contiene información acerca de cómo subir documentos al aplicativo.

Módulo de validación de documentación

Cuadro RF 12: Envío de notificaciones a estudiantes vía email.

Tabla 21. Cuadro RF 12: Envío de notificaciones a estudiantes vía email

Descripción	Envío automático de notificaciones por medio de mensajería electrónica, el cual se genera en tiempo real. RF 3.1.8.3		
Entrada	Fuente	Salida	Restricción
El gestor aprueba o niega la solicitud o proceso en el sistema	Solicitudes realizadas por el estudiante	Notificación en el correo personal del estudiante	No aplica
Proceso	El proceso de validación de pasantías y prácticas pre profesionales cuenta con varias etapas, las cuales son aprobadas o negadas, una vez que el gestor revisa la documentación, datos en el sistema se genera automáticamente una notificación hacia el estudiante de forma automática que va de acuerdo a la revisión realizada.		

Nota: Esta tabla contiene información acerca de cómo se enviará notificaciones.

Cuadro RF 13: Campo de observaciones.

Tabla 22. Cuadro RF 13: Campo de observaciones

Descripción	Este campo es el para el gestor de validar, aprobar los diversos formularios generados por cada estudiante. RF 3.1.8.3		
Entrada	Fuente	Salida	Restricción
Los formularios ingresados por los estudiantes.	Plataforma de pasantías y prácticas pre profesionales.	Un campo de observaciones en cada etapa o estado del flujo a seguir.	Para el estudiante este es un campo de solo lectura.
Proceso	De acorde a los literales que sean incorrectos, el gestor si es necesario escribirá la información pertinente para que el alumno vea y corrija.		

Nota: Esta tabla contiene información acerca de cómo escribir observaciones.

Cuadro RF 14: Validación de documentos subidos por el alumno.

Tabla 23. Cuadro RF 14: Validación de documentos subidos por el alumno

Descripción	El gestor revisa que los documentos subidos por el alumno sean correctos. RF 3.1.8.3		
Entrada	Fuente	Salida	Restricción
Documentos escaneados referentes al proceso de pasantía.	Plataforma de pasantías y prácticas pre profesionales.	Notificación automática vía email al estudiante de aprobado o negado.	No se podrá observar los documentos si no están subidos en formato .PDF
Proceso	El gestor revisa la documentación subida por el alumno en una interfaz para observar que sean los correctos y no exista ninguna anomalía. Posteriormente se procede a aprobar o negar para continuar con el proceso.		

Nota: Esta tabla contiene información acerca del proceso de validación

Cuadro RF 15: Validación de información mediante checkbox.

Tabla 24. Cuadro RF 15: Validación de información mediante checkbox

Descripción	El gestor valida la información realiza una serie de vistos en cada cuadro de validación. RF 3.1.8.3		
Entrada	Fuente	Salida	Restricción

Datos que sube el alumno	Plataforma de pasantías y prácticas pre profesionales.	Notificación automática vía email al estudiante de aprobado o negado.	El gestor no puede editar la información.
Proceso	En cada etapa del proceso que sea necesaria la validación de información, el gestor revisa el formulario pre cargado con los datos, si el formulario no completa todos los vistos, se envía una notificación de rechazado al estudiante para que vuelva a modificar el campo no marcado, si todos los campos son marcados pasa al siguiente estado.		

Nota: Esta tabla contiene información acerca de cómo validar los diversos campos.

2.1.9.2.Requerimientos no funcionales

Cuadro RNF 1: Portabilidad.

Tabla 25. Cuadro RNF 1: Portabilidad.

Descripción	El sistema será portable para las plataformas UNIX, Windows.
Razón	Se necesita que la aplicación Web funcione en cualquier S.O. Linux o Windows.
Criterio de medición	Instalación correcta en los dos sistemas operativos.

Nota: Esta tabla contiene información acerca de la portabilidad del sistema.

Cuadro RNF 2: Interfaz gráfica del usuario GUI.

Tabla 26. Cuadro RNF 2: Interfaz gráfica del usuario GUI.

Descripción	La interfaz gráfica mostrará todo lo que el usuario necesite.
Razón	Una interfaz gráfica generada correctamente permite agilizar y manejar de una forma adecuada los procesos.
Criterio de medición	Pruebas con usuarios.

Nota: Esta tabla contiene información acerca de cómo será la interfaz de usuario.

Cuadro RNF 3: Sistema Responsive

Tabla 27. Cuadro RNF 3: Sistema Responsive. (Achig & Torres, 2018)

Descripción	Para una correcta visualización el sistema será adaptativo a cualquier plataforma o dispositivo.
Razón	La interfaz tiene que ser minimalista, intuitiva y responsive.
Criterio de medición	Pruebas de adaptabilidad en diferentes dispositivos como: computadores de escritorio, laptops, móviles y tablets.

Nota: Esta tabla contiene información acerca del sistema y su adaptabilidad en diversas pantallas.

Cuadro RNF 4: Mensajes de error en el sistema (POP-UP)

Tabla 28. Cuadro RNF 4: Mensajes de error en el sistema (POP-UP).
(Achig & Torres, 2018)

Descripción	El sistema proporcionará mensajes de error informativos para el usuario final.
Razón	El sistema apoyará al usuario con mensajes informativos.
Criterio de medición	Pruebas con el usuario.

Nota: Esta tabla contiene información acerca de los mensajes de error a mostrarse.

Cuadro RNF 5: Tiempo de respuesta

Tabla 29. Cuadro RNF 5: Tiempo de respuesta. (Achig & Torres, 2018)

Descripción	La rapidez con la que el sistema realiza una acción o petición que ejecuta el usuario.
Razón	El sistema realizará acciones o peticiones que realiza el usuario en un máximo de 2 segundos.
Criterio de medición	En la actualidad Google considera que el tiempo de respuesta de un servidor es de 0.2 segundos (200 milisegundos). Se usará la herramienta PageSpeed Insights se obtendrá el valor de respuesta.

Nota: Esta tabla contiene información acerca de los tiempos de respuesta esperados.

Cuadro RNF 6: Seguridad

Tabla 30. Cuadro RNF 6: Seguridad. (Achig & Torres, 2018)

Descripción	La información almacenada en la base de datos será de acceso restringido para usuarios no registrados y usuarios sin privilegios.
Razón	Los respaldos de la base de datos, así como la información que se va almacenando son únicamente accesibles por una persona delegada como administrador, quien tendrá acceso a las dos plataformas.
Criterio de medición	Pruebas de integridad de datos

Nota: Esta tabla contiene información acerca de la seguridad a ser implementada.

Cuadro RNF 7: Concurrencia

Tabla 31. Cuadro RNF 7: Concurrencia

Descripción	Promedio de usuarios que ingresan al sistema de pasantías y prácticas pre profesionales.
Razón	El sistema soportará un mínimo de 100 usuarios a la vez, sin demoras en el mismo.
Criterio de	Se estima que cada semestre existe un máximo de 200 estudiantes que

medición	podrían usar el sistema de pasantías y prácticas pre profesionales.
----------	---

Nota: Esta tabla contiene información acerca de cuantos usuarios concurrentes debe soportar.

Cuadro RNF 8: Disponibilidad.

Tabla 32. Cuadro RNF 8: Disponibilidad. (Achig & Torres, 2018)

Descripción	La aplicación debe soportar estar disponible los siete días de la semana, las veinte y cuatro horas al día, de acorde al estándar IT su disponibilidad será de un 90% anual.
Razón	Se necesita una aplicación donde su disponibilidad sea permanente.
Criterio de medición	Tiempo en las que un sistema es capaz de realizar sus funciones

Nota: Esta tabla contiene información acerca del tiempo activo del sistema

2.2.Diseño

La presente sección de diseño del software está compuesta de los diagramas de casos de uso, diagrama de clases, diagrama de base de datos en la cual se puede observar su estructura, la presentación de las interfaces a ser desarrolladas. Para el diseño de la interfaz se usó la aplicación Pencil Project (Project, 2017).

Para los diagramas de caso de uso se usó la aplicación StarUml (StarUml, 2014).

2.2.1. Diagrama de casos de uso

Permite observar de forma gráfica el comportamiento que tendrán los involucrados o actores dentro de la plataforma.

2.2.1.1. Diagrama de caso de uso para el módulo de carga de información

En el siguiente diagrama se presenta los actores involucrados que van a interactuar con la aplicación. Los actores son la Dirección de Carrera de Ingeniería de Sistemas y una persona delegada ha de ser seleccionada para realizar este proceso.

Caso de uso

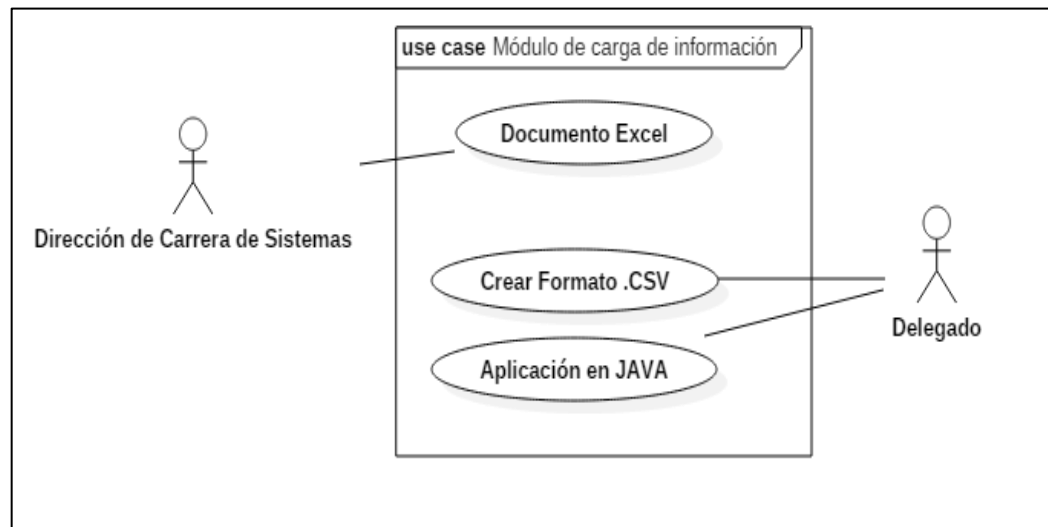


Figura 4. Caso de uso módulo carga información.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

2.2.1.2. Diagrama de caso de uso para el módulo de registro de pasantía o práctica preprofesional

En el siguiente diagrama se presentan los actores involucrados que van a interactuar con la aplicación. Los actores son el estudiante y un gestor.

Caso de uso

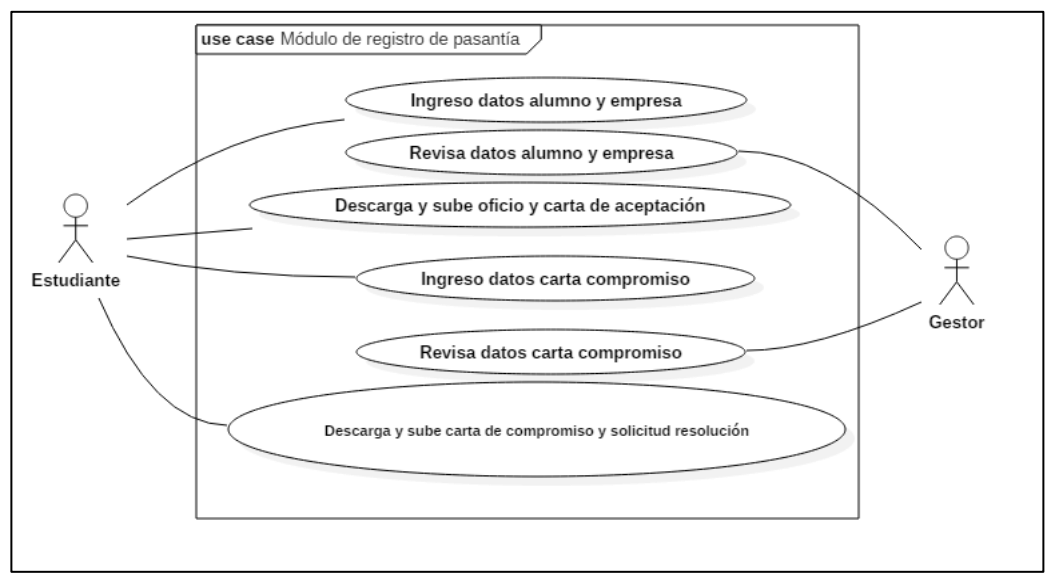
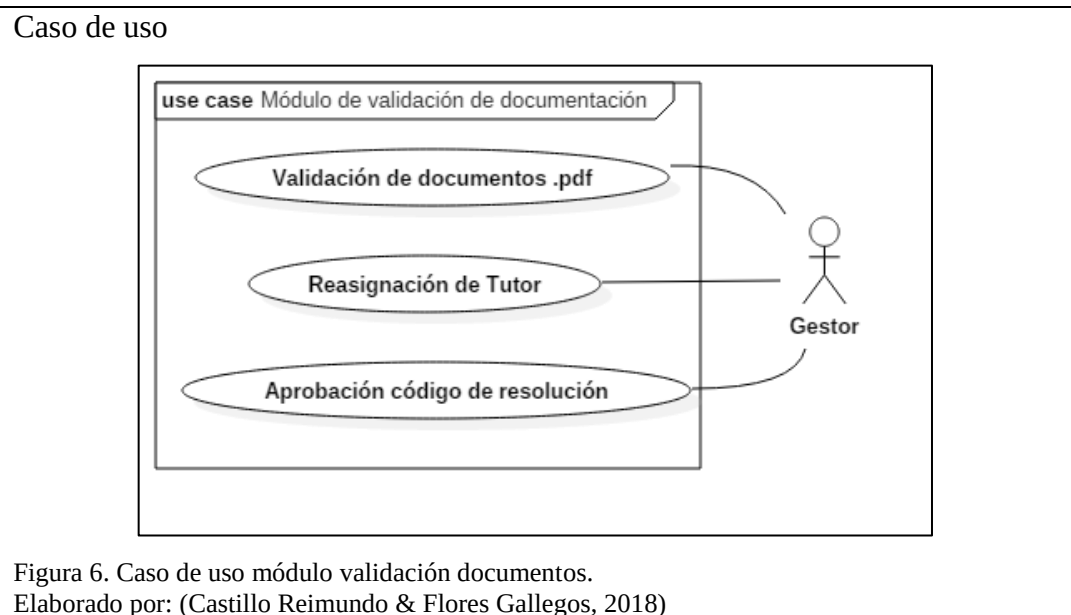


Figura 5. Caso de uso módulo de registro pasantía.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

2.2.1.3. Diagrama de caso de uso para el módulo de validación de documentación

En el siguiente diagrama se presentan los actores involucrados que van a interactuar con la aplicación. El actor es el gestor.



2.2.2. Esquema de clases

A continuación, se presentan las clases con una explicación de por qué se realizó de dicha manera.

Esquema de clases

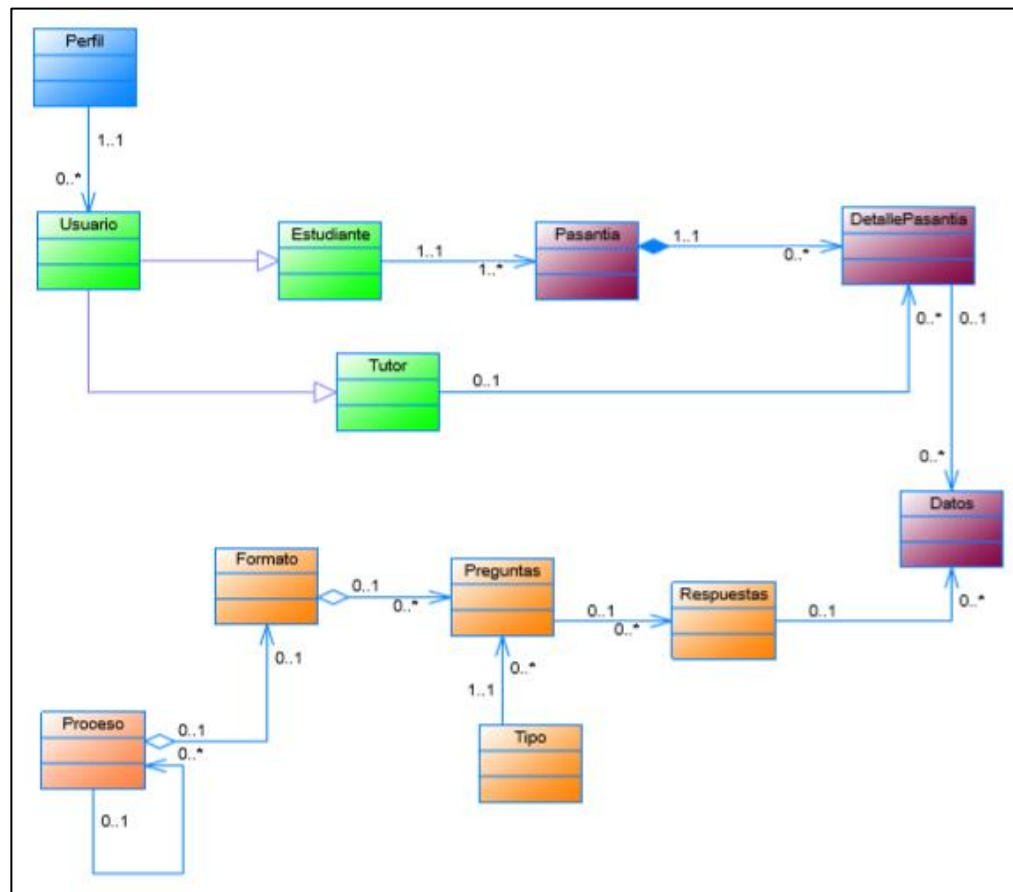


Figura 7. Esquema de clases principales.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

2.2.3. Diagrama conceptual de base de datos

En el diagrama se puede observar todos los campos y cardinalidades, basado en los requerimientos planteados por los usuarios. (Achig & Torres, 2018).

Diagrama conceptual

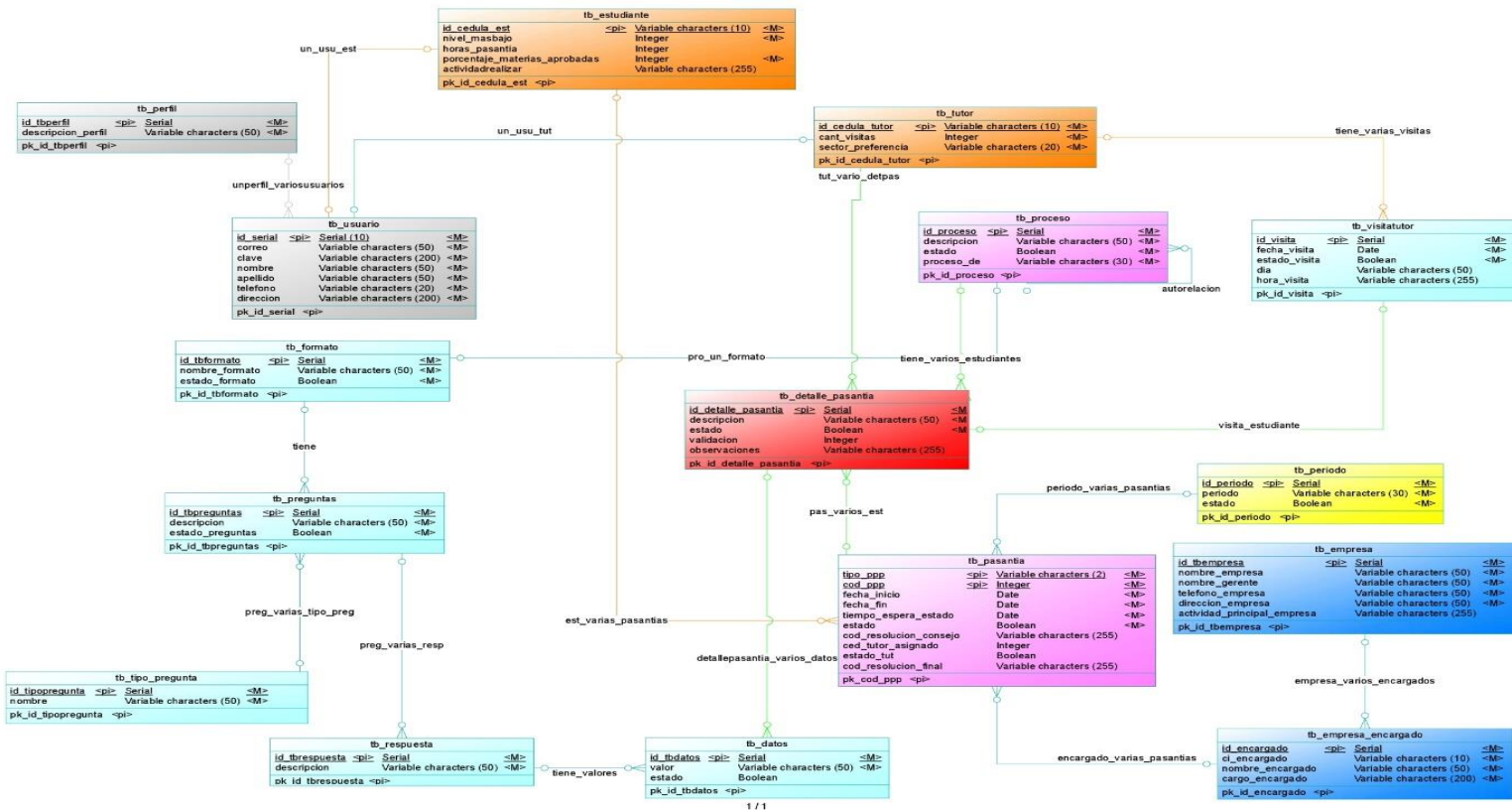


Figura 8. Diagrama conceptual de base de datos.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

2.2.4. Interfaz abstracta

Cada una de las interfaces presentadas, cuenta con todos los campos que se observarán en el producto final, las ventanas siguen un orden específico mientras el proceso avanza.

2.2.4.1. Ventana global para todos los perfiles

Ingreso, autenticación para todos los usuarios y perfiles.

Interfaz

Diagrama de la interfaz de login para la plataforma. El formulario está centrado y contiene los siguientes elementos:

- Título: "Ingreso a la plataforma"
- Etiqueta: "Usuario" sobre un campo de entrada rectangular.
- Etiqueta: "Contraseña" sobre un campo de entrada rectangular.
- Botón: "Ingresar" (rectangular, color verde).
- Enlace: "[Olvide mi contraseña](#)" (texto azul).

Figura 9. Ventana global para todos los perfiles.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

2.2.4.2. Alumno

Llena la solicitud de inicio del proceso de pasantía y datos de la empresa.

Interfaz

SOLICITUD INICIO PROCESO PASANTIA

Datos del alumno:

Correo para notificaciones:

Telefono celular:

Telefono fijo:

☒ Pasantia
 ☐ Practica pre profesional

Actividad bien definida: Indicar las actividades a realizar dentro de la empresa.

Datos de la empresa:

RUC:

Nombre de la empresa:

Telefono:

Direccion:

Nombre de la persona a quien va dirigido el oficio:

Titulo profesional:

Observaciones:

Fechas tentativas de inicio y fin de las actividades que va a realizar el estudiante

Fecha inicio

AUG - 2016

S	M	T	W	T	F	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Fecha fin

AUG - 2016

S	M	T	W	T	F	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Mensaje: Si los datos ingresados son aprobados, la proxima vez que ingrese al sistema, usted pasara automaticamente a otra pantalla.

Enviar

Figura 10. Proceso pasantía y datos empresa.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

El alumno descarga, completa y sube la documentación requerida.

SECCION DE DOCUMENTACION A COMPLETAR

1 Descargue el oficio para la empresa

Descargar

2 Descargue el formato de carta de aceptacion que debe llenar la empresa

[Descargar enlace del formato a seguir](#)

SECCION DE CARGA DE ARCHIVOS

3 Subir el oficio sellado por la empresa en formato .pdf

Subir PDF

4 Subir la carta de aceptacion sellada por la empresa en formato .pdf

Subir PDF

Observaciones:

He subido los dos archivos PDF

Figura 11. Documentación requerida.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

El alumno procede a llenar los campos faltantes en la carta de compromiso, dichos campos se los presenta con fondo blanco.

Interfaz

CARTA DE COMPROMISO

Información general:

Tipo Código unico

Nombre de la empresa:

Dirección de la empresa:

Información alumno

Correo del estudiante: Ciclo que cursa

Objeto de la actividad académica

Nombre del programa

Área que requiere la actividad académica

Responsable del área

Lista de actividades previstas:

Lista de resultados previstos:

Lista de productos entregables:

Observaciones:

Fecha inicio

Fecha fin

Horas a la semana

Total horas

Tutor asignado

Figura 12. Llenar carta de compromiso.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

El alumno descarga la solicitud de inicio de actividades en la empresa y sube la documentación requerida.

Interfaz

SOLICITUD INICIO ACTIVIDADES EMPRESA

1) Descargar la carta compromiso

2) Subir la carta compromiso firmada y sellada por la empresa en pdf

3) Subir la solicitud inicio actividades firmada por el alumno en pdf

Figura 13. Solicitud inicio actividades en empresa.

2.2.4.3. Gestor

El gestor observa la lista de solicitudes para inicio del proceso de pasantía que están pendientes a revisar.

Interfaz

LISTADO SOLICITUD INICIO PROCESO PASANTIA			
Name	Email Address	Mobile Number	Action
Lisa	lisa@gmail.com	98763123	<button>Read</button>
Tom	tom@gmail.com	93813412	<button>Read</button>

Figura 14. Lista solicitudes inicio proceso.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

El gestor revisa y valida la información de forma individual de inicio de proceso de pasantías.

Interfaz

SOLICITUD INICIO PROCESO PASANTIA

Estudiante: Nombre1 Nombre2 Apellido1 Apellido2

Datos del alumno:

Correo para notificaciones:

☐
☒

Telefono celular:

☐
☒

Telefono fijo:

☐
☒

Actividad bien definida: Indicar las actividades a realizar dentro de la empresa.

☐
☒

Datos de la empresa:

RUC:

☐
☒

Nombre de la empresa:

☐
☒

Telefono:

☐
☒

Dirección:

☐
☒

Nombre de la persona a quien va dirigido el oficio:

☐
☒

Título profesional:

☐
☒

En caso de enviar a corregir, poner aquí sus observaciones:

Fechas tentativas de inicio y fin de las actividades que va a realizar el estudiante

Fecha inicio

AUG - 2015

S

M

T

W

T

F

Sa

Fecha fin

AUG - 2015

S

M

T

W

T

F

Sa

Enviar a corregir

Aprobar

Figura 15. Revisa y valida la información.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

El gestor observa la lista de la carta de compromiso que están pendientes a revisar.

Interfaz

LISTADO PENDIENTE CARTA DE COMPROMISO			
Name	Email Address	Mobile Number	Action
Lisa	lisa@gmail.com	98763123	Read
Tom	tom@gmail.com	93813412	Read

Figura 16. Listado pendiente carta compromiso.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

El gestor revisa y valida la carta de compromiso de forma individual.

Interfaz

52

Estudiante: Nombre1 Nombre2 Apellido1 Apellido2

CARTA DE COMPROISO

Informacion general:

Tipo

PA / PP

Codigo unico

1234

Nombre de la empresa:

Direccion de la empresa:

Informacion alumno

Correo del estudiante:

Ciclo que cursa

✓

Fecha inicio

AUG - 2016

S

M

T

W

T

F

Sa

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

1

2

3

4

5

6

7

8

9

10

Fecha fin

AUG - 2016

S

M

T

W

T

F

Sa

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

1

2

3

4

5

6

7

8

9

10

Objeto de la actividad academica

✓

Nombre del programa

✓

Area que requiere la actividad academica

✓

Responsable del area

✓

Lista de actividades previstas.

✓

Lista de resultados previstos

✓

Lista de productos entregables

✓

En caso de enviar a corregir, poner aqui sus observaciones:

Enviar a corregir

Aprobar

Figura 17. Validar carta compromiso.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

El gestor observa la lista de documentación del proceso de pasantía que están pendientes a revisar.

Interfaz

LISTADO PENDIENTE DE VALIDACION DOCUMENTACION

Name	Email Address	Mobile Number	Action
Lisa	lisa@gmail.com	98763123	Read
Tom	tom@gmail.com	93813412	Read

Figura 18. Listado pendiente documentos.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

El gestor revisa y valida la documentación de forma individual.

Interfaz

VALIDACION DE DOCUMENTACION

Estudiante: N N A A

Aqui puede observar el oficio y la carta de aceptacion sellados por la empresa

1 Lorem Ipsum Dolor

En caso de enviar a corregir, poner aqui sus observaciones:

Enviar a corregir Aprobar

Figura 19. Validar documentos PDF.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

El gestor observa la lista de solicitud de inicio de actividades que están pendientes a revisar y aprueba de manera individual a cada alumno.

Interfaz

APROBAR INICIO ACTIVIDADES EN CONSEJO

En caso de cambiar el tutor asignado seleccione de la lista

Reasignacion tutor

Ingrese el cod de resolucion que se dio al alumno en consejo de carrera

Aprobar por consejo

Enviar

Figura 20. Aprobación inicio actividades.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Capítulo 3

3. Construcción y Pruebas

En este capítulo observa los segmentos más importantes del proceso de construcción del sistema, las herramientas involucradas para su desarrollo, las ideas plasmadas en imagen que se han convertido en objetos tangibles, a su vez se observa la forma en cómo ha ido evolucionando hasta llegar a la etapa de pruebas.

3.1.Herramientas

3.1.1. Hibernate

Es un framework que permite transferir los objetos de una aplicación Java a una base de datos relacional en forma de objetos, ésta se denota por sus siglas ORM. Sus principales desarrolladores son Redhat y Redhat Developers, y por ende es una aplicación de negocio de código abierto, su fundador se llama Mr. Gavin King quien lo creó en el 2002, como una alternativa a los EJB. Se usa esta herramienta ya que permite la manipulación de los datos de la base de una forma fácil, además que por detrás maneja y administra el pool de conexiones, así como las peticiones de transacciones que los usuarios hacen al interactuar con la aplicación. (Xia, 2009).

3.1.2. HQL

Se presenta al lenguaje de Hibernate (Hibernate Query Language), en su documentación permite saber la estructura de una consulta simple y juntar consultas adicionales con parámetros establecidos como es el caso de “JOIN”

que permite hacer búsqueda entre dos tablas con valores específicos (Hibernate, 2004).

3.1.3. Sb Admin 2

La empresa Twitter en 2010 es quien crea el framework Bootstrap con lanzamiento oficial en agosto 19 del 2011. Esta a su vez sirvió para el desarrollo del tema “Sb Admin 2”, su lanzamiento fue el 14 de enero del 2014 desarrollado por David Miller. Tema de código abierto, fácil de manipular para construcción de aplicaciones Web. Se usa esta herramienta para aplicarla a la vista del MVC de la aplicación (Miller, 2014).

3.1.4. Bootsfaces

Es un framework basado en el Twitter Bootstrap de código abierto, desarrollada por el apoyo de la comunidad formado por profesionales de IT. Se usa esta herramienta como complemento de Bootstrap, que contiene componentes mejorados, como los que se usan para cargar datos en una tabla partiendo de un objeto o de una lista de objetos dentro de Java Server Faces (D'Urzo, 2013).

3.1.5. IText

El primer lanzamiento fue el 14 de febrero del 2000, además que se basa en un flujo de trabajo de documentación. La librería PDF de versión 5.0.5 utilizada en el desarrollo de la aplicación, es de código abierto y permite a la aplicación generar los diferentes reportes PDF en tiempo real, de manera dinámica y agiliza el proceso para los usuarios (Itext, 2000).

3.1.6. Java Mail API 1.4

Desarrollada por Oracle, el 21 de enero del 2011, funciona a partir de un JAVA SE 6. Se la usa para el envío automático de notificaciones email, con un proveedor GMAIL, ya que permite una configuración sencilla para el trabajo en conjunto de las herramientas (Oracle, JavaMail 1.4.4, 2011).

3.1.7. PostgreSQL 9.4

Es un sistema de administración de base de datos relacional, proviene de la Universidad de California en Berkeley, cuenta con más de dos décadas de desarrollo, su proyecto se llama “The Berkeley POSTGRES Project”, sus inicios parten del año 1986 liderado por el profesor Michael Stonebraker. Esta herramienta es el núcleo del sistema, ya que es el motor de toda búsqueda, petición y acción que se quiera realizar (Stonebraker, 1986).

3.1.8. PgAdmin III

Es el complemento de PostgreSQL, ya que éste es el gestor gráfico de la base, es un sistema de administración diseñado para los sistemas Unix y Windows. El grupo a cargo se llama “The pgAdmin Development Team”, desarrollado en C++ y usa las librerías wxWidgets (Team, 2002).

3.1.9. GitHub

Es una herramienta que permite el trabajo en equipo, cuando de desarrollo de software se habla, creada en octubre del 2007, en la ciudad de San Francisco, sus tipos de servicio son de código abierto y corporativo. Su funcionalidad principal es unir automáticamente el código de desarrollo de una o varias aplicaciones,

donde los participantes tendrán actualizados los cambios de los demás en tiempo real (Chris Wanstrath, 2007).

3.1.10. Glassfish

Permite arrancar la aplicación y es eficiente para soportar gran cantidad de usuario, se optó por este servidor ya que permite ejecutar la mensajería instantánea más rápido que Tomcat. (Oracle, Glassfish, 2015).

3.1.11. Java JDK 8u151

Java permite compilar el código e implementarlo a una máquina virtual la cual permite su ejecución, la versión utilizada es estable, ayuda a la optimización en el tiempo de ejecución al invocar a otras desarrolladas en Java. (Oracle, JDK 8u151 Update Release Notes, 2017).

3.2. Modelo Físico de la base de datos

A continuación, se presenta el modelo físico de la base de datos, el cual describe toda la estructura de almacenamiento de los datos y la relación que hay entre ellos. La tabla central de la base es la tabla de color rojo “tb_detalle_pasantia” de aquí es como se controla todo el workflow dentro del proceso.

Las tablas de color plomo representan a los usuarios y el perfil de cada uno los cuales se asocian al inicio de sesión. Las tablas de color naranja representan a datos específicos que necesita el usuario estudiante y tutor. Las tablas de color rosado permiten controlar el estado del flujo, y el proceso al que pertenece dicho estado. Las tablas azules describen los datos de la empresa y de varios empleados de la empresa. Las tablas de

color celeste contienen los datos de los tutores y datos ingresados por el alumno de acorde a la pasantía de cada estudiante.

Modelo DB

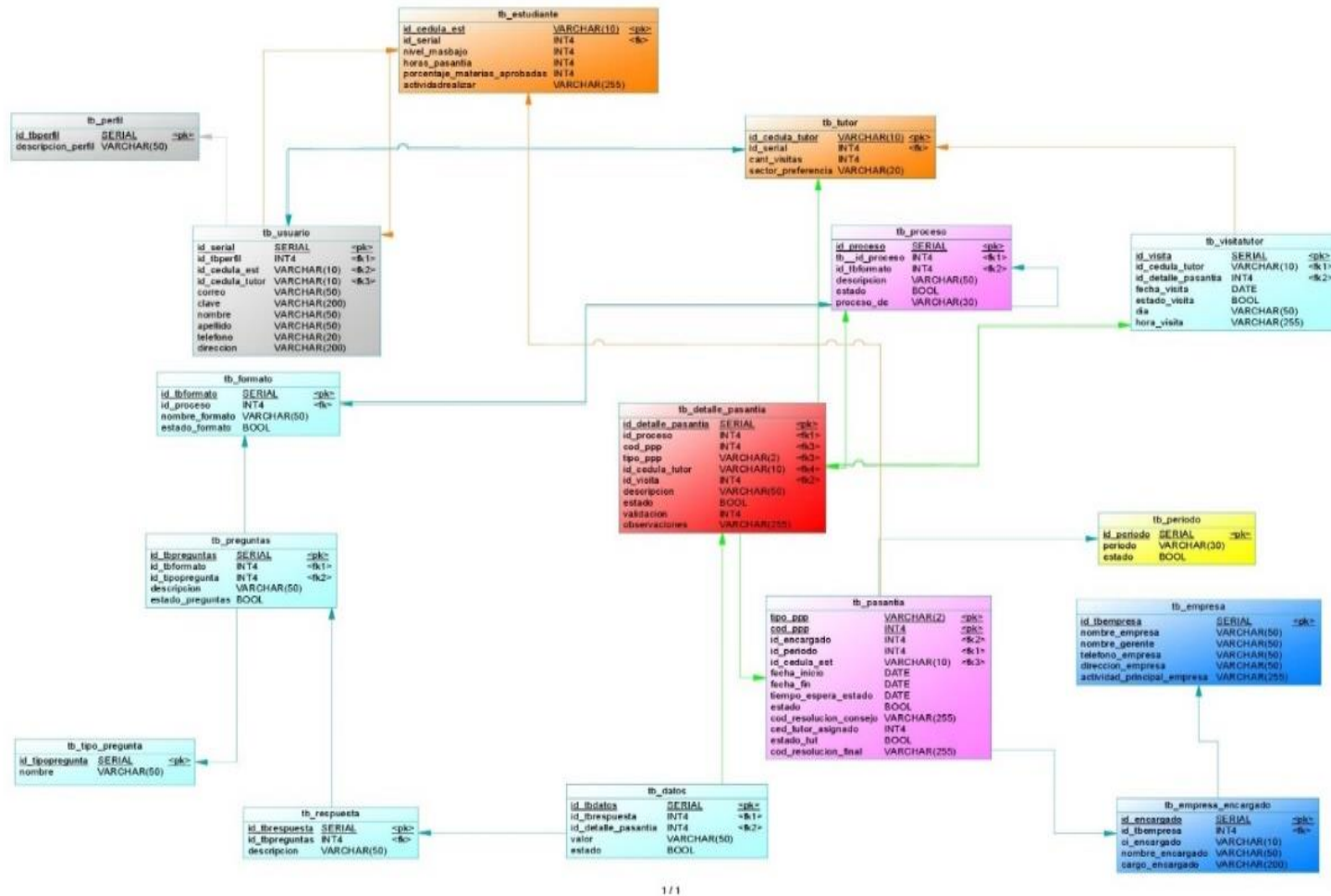


Figura 21. Modelo físico BD.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018) (Achig & Torres, 2018)

3.3.Diccionario de Datos

A continuación, se describe el diccionario de datos.

Cuadro de la tabla datos

Tabla 33. Cuadro de la tabla datos

Tabla tb_datos			
Esta tabla contiene las respuestas a los datos que se le ha pedido ingresar al estudiante			
Campo	Tipo	Observación	Mandatorio
id_tbdatos	Bigserial	Clave única	PK
valor_datos	character varying(255)	Contiene la respuesta	
estado	Boolean	Valor por defecto	
iddetallepasantia	Integer	Valor del detalle pasantía al que pertenece	FK
id_tbrespuesta	Bigint	Valor de nombre de la respuesta	FK

Nota: Esta tabla contiene información acerca de los campos de la tabla datos

Cuadro de la tabla detalle pasantía

Tabla 34. Cuadro de la tabla detalle pasantía

Tabla tb_detalle_pasantia			
Tabla central de la base, su finalidad es obtener el estado del proceso de cada uno de los alumnos a lo largo del flujo			
Campo	Tipo	Observación	Mandatorio
iddetallepasantia	Serial		PK
descripcion	character varying(255)	Se describe el nombre del proceso en el que se encuentra	
estado	Boolean	Permite saber si un proceso es activo o inactivo	
validacion	Integer	Permite saber el tipo de acción que se debe tomar para un proceso	FK
id_proceso	Bigint	Indica el proceso al que pertenece en el flujo	FK
tipo_ppp	character varying(255)	Indica el tipo de actividad	FK
cod_ppp	Integer	Indica el código de la actividad	FK
cedula_tut	Bigint	Se agrega la identificación del tutor asignado	FK
id_visita	Bigint	Es el id de la visita del estudiante	FK
observacion	character varying(255)	Observaciones vistas por el alumno	FK

Nota: Esta tabla contiene información acerca de los campos de la tabla detalle pasantia

Cuadro de la tabla empresa

Tabla 35. Cuadro de la tabla empresa

Tabla tb_empresa			
Contiene los datos de la empresa			
Campo	Tipo	Observación	Mandatorio
id_empresa	Bigint	Ruc de la empresa	PK
nombre_empresa	character varying(255)	Nombre de la empresa o institución	
nombre_gerente	character varying(255)	Nombres y apellidos del gerente	
telefono_empresa	character varying(255)	Número de contacto de la empresa o institución	
direccion_empresa	character varying(255)	Dirección de la empresa o institución	
actividad_principal_empresa	character varying(255)	Describe el giro del negocio	

Nota: Esta tabla contiene información acerca de los campos de la tabla empresa

Cuadro de la tabla encargado de la empresa

Tabla 36. Cuadro de la tabla encargado de la empresa

Tabla tb_empresa_encargado			
Contiene datos de varios encargados que pueden existir en una empresa			
Campo	Tipo	Observación	Mandatorio
id_encargado	Bigserial	Se genera automáticamente un contador	PK
ci_encargado	Bigint	Cédula de ciudadanía del encargado	
nombre_encargado	character varying(255)	Nombres de la persona a cargo del alumno	
cargo_encargado	character varying(255)	Cargo o profesión que ocupa dentro de la empresa	
id_empresa	Bigint	Recibe el id de la empresa a la que pertenece el encargado	FK

Nota: Esta tabla contiene información acerca de los campos de la tabla encargado.

Cuadro de la tabla estudiante

Tabla 37. Cuadro de la tabla estudiante

Tabla tb_estudiante			
Se almacena información necesaria del alumno			
Campo	Tipo	Observación	Mandatorio
cedula_est	Bigint	Se la toma de la tabla usuario,	PK

		aquí se la recibe como clave primaria	
ultimonivel	Integer	Nivel más bajo del alumno	Obligatorio
horaspasantia	Integer	Horas que realizara en una pasantía o práctica preprofesional el alumno	
porcentajemateriasaprobadas	Integer	Número que representa el total de materias aprobadas	Obligatorio
actividadrealizar	character varying(255)	Detalla las actividades a realiza	

Nota: Esta tabla contiene información acerca de los campos de la tabla estudiante.

Cuadro de la tabla formato

Tabla 38. Cuadro de la tabla formato

Tabla tb_formato			
Contiene el nombre de los formatos que se debe llenar dentro del proceso			
Campo	Tipo	Observación	Mandatorio
id_tbformato	Bigserial	Enumerador automático	PK
nombre_formato	character varying(255)	Se ingresa el nombre de cada formato a manejar dentro del proceso	
estado_formato	Boolean	Se define los formatos activos e inactivos	

Nota: Esta tabla contiene información acerca de los campos de la tabla formato.

Cuadro de la tabla pasantía

Tabla 39. Cuadro de la tabla pasantía

Tabla tb_pasantia			
Contiene la información necesaria de la pasantía			
Campo	Tipo	Observación	Mandatorio
tipo_ppp	character varying(255)	Contiene el tipo de pasantía o práctica preprofesional	PK
cod_ppp	Integer	Contiene el número que se asigna a un tipo, este número se lo maneja por detrás como un serial	PK
fechainicio	Date	Almacena la fecha de inicio de la actividad a realizar por el alumno	
fechafin	Date	Almacena la fecha fin de la actividad a realizar por el alumno	
tiempoesperaestado	Integer	Almacena el dato numérico de días de espera	
estado	Boolean	Si el estado es verdad quiere decir que hay pasantías pendientes, una vez en falso quiere decir que el alumno ya a culminado con el flujo	

cod_resolucion_consejo	character varying(255)	Se agrega el código una vez que el estudiante inicie las actividades en la empresa o institución	
ced_tutor_asignado	Bigint	Se asigna una cédula de un tutor de acorde a su carga	
estado_tut	Boolean	Se lo pone en estado verdadero para que un tutor vea que tiene visitas pendientes	
cod_resolucion_final	character varying(255)	Se pone el código final cuando el alumno envió a validar su pasantía o práctica preprofesional	
id_encargado	Bigint	Se asigna el serial del encargado que está asociado al estudiante	FK
cedula	Bigint	Esta es la cédula del estudiante, quien está realizando la actividad	FK
id_periodo	Bigint	Aquí se obtiene el serial del periodo habilitado o periodo en curso	FK

Nota: Esta tabla contiene información acerca de los campos de la tabla pasantía.

Cuadro de la tabla perfil

Tabla 40. Cuadro de la tabla perfil

Tabla tb_perfil			
Se define todos los perfiles que puede existir en la aplicación			
Campo	Tipo	Observación	Mandatorio
id_tbperfil	Serial	Identificador del perfil	PK
descripcion_perfil	character varying(255)	Nombre del perfil	

Nota: Esta tabla contiene información acerca de los campos de la tabla perfil

Cuadro de la tabla periodo

Tabla 41. Cuadro de la tabla periodo

Tabla tb_periodo			
Describe el periodo en actual de cada semestre			
Campo	Tipo	Observación	Mandatorio
id_periodo	Bigserial	Valor auto incremental	PK
periodo	character varying(255)	Describe el número de cada periodo	
estado_periodo	Boolean	Establece el periodo habilitado	

Nota: Esta tabla contiene información acerca de los campos de la tabla periodo.

Cuadro de la tabla preguntas

Tabla 42. Cuadro de la tabla preguntas

Tabla tb_preguntas			
Se ingresan todas las preguntas para todos los formatos			
Campo	Tipo	Observación	Mandatorio
id_tbpreguntas	Bigserial	Auto incremental	PK
descripcion	character varying(255)	Se describe a la pregunta	
estado_preguntas	Boolean	Define si la pregunta está activa o no	
id_tipopregunta	Bigint	Se hace una relación a qué tipo de pregunta pertenece	FK
id_tbformato	Bigint	Se hace una relación al formato que pertenece	FK

Nota: Esta tabla contiene información acerca de los campos de la tabla preguntas.

Cuadro de la tabla proceso

Tabla 43. Cuadro de la tabla proceso

Tabla tb_proceso			
Aquí se describe el flujo que seguirá todo el proceso			
Campo	Tipo	Observación	Mandatorio
id_proceso	Bigserial	Auto incremental	PK
descripcion	character varying(255)	Nombre de cada proceso	
estado	Boolean	Indica los procesos habilitados	
proceso_de	Bigint	Auto relación al proceso que se deriva como un sub proceso	FK
formato_id	Bigint	Indica a que formato pertenece un proceso	FK

Nota: Esta tabla contiene información acerca de los campos de la tabla proceso.

Cuadro de la tabla respuesta

Tabla 44. Cuadro de la tabla respuesta

Tabla tb_respuesta			
Es la descripción de la respuesta, y se indica la asociación que tiene con cada pregunta			
Campo	Tipo	Observación	Mandatorio
id_tbrespuesta	Bigserial	Auto incremental	PK
descripcion_resp	character varying(255)	Se describe la respuesta	
id_tbpreguntas	Bigint	Asociación con tabla preguntas	FK

Nota: Esta tabla contiene información acerca de los campos de la tabla respuesta.

Cuadro de la tabla tipo pregunta

Tabla 45. Cuadro de la tabla tipo pregunta

Tabla tb_tipopregunta	
Se define los diferentes tipos de preguntas	

Campo	Tipo	Observación	Mandatorio
id_tipopregunta	Bigserial	Auto incremental	PK
tipo	character varying(255)	Define el tipo de campo que se usa para una pregunta dentro de un formulario	

Nota: Esta tabla contiene información acerca de los campos de la tabla tipo pregunta.

Cuadro de la tabla tutor

Tabla 46. Cuadro de la tabla tutor

Tabla tb_tutor			
Se puede observar los tutores de visitas y la cantidad de las mismas			
Campo	Tipo	Observación	Mandatorio
cedula_tut	Bigint	Auto incremental	PK
cant_visitas	Integer	Se aumentará de acorde a las visitas concertadas de cada tutor	Obligatorio
sector_preferencia	character varying(255)		Obligatorio

Nota: Esta tabla contiene información acerca de los campos de la tabla tutor

Cuadro de la tabla usuario

Tabla 47. Cuadro de la tabla usuario

Tabla tb_usuario			
Se tiene los datos primordiales del usuario			
Campo	Tipo	Observación	Mandatorio
id_usuario	Bigint	Cédula de ciudadanía	PK
correo	character varying(255)	Email de referencia usuario	
clave	character varying(255)	Clave del usuario	
nombre	character varying(255)	Apellidos del usuario	
apellido	character varying(255)	Nombres del usuario	
telefono	character varying(255)	Teléfono del usuario	
direccion	character varying(255)	Dirección del usuario	
id_perfil	Integer	Perfil del usuario	Obligatorio

Nota: Esta tabla contiene información acerca de los campos de la tabla usuario

3.4.Historias de Usuario

Se utilizó XP como desarrollo iterativo, se procede a indicar como ha ido evolucionando el software en las iteraciones realizadas durante su desarrollo. Se utilizaron historias de usuario para cada parte funcional del sistema.

3.4.1. Módulo de registro de pasantía

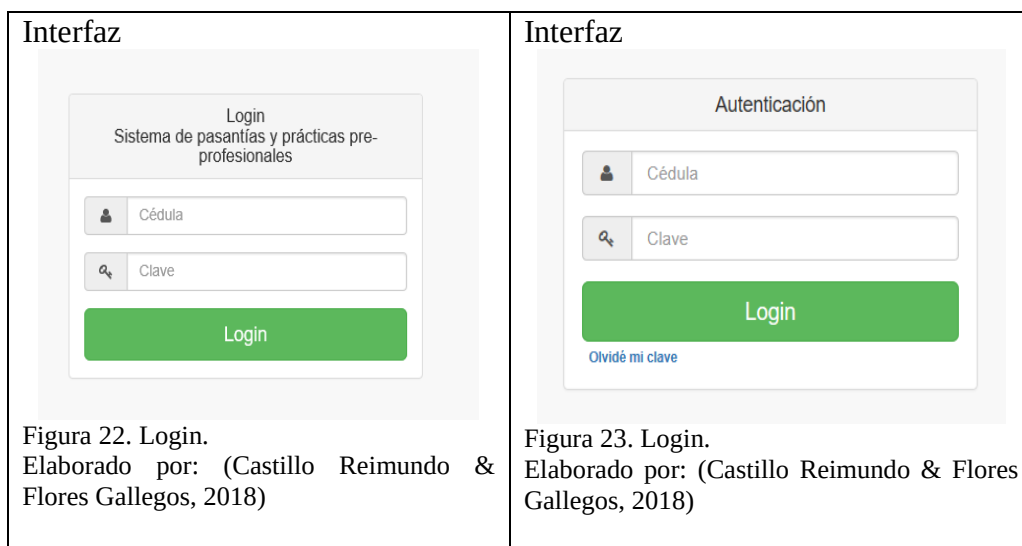
3.4.1.1. Primera y segunda iteración Login

El desarrollo se enfoca en mantener una interfaz simple e intuitiva para el usuario, con acciones fáciles a realizar.

Para el portal de acceso se realizó un diseño limpio y claro para permitir el ingreso de credenciales, usuario y contraseña.

La segunda iteración

Se procede a cambiar el diseño de acceso al sistema con una nueva leyenda, colocando un enlace en caso de pérdida de contraseña y adaptándolo con iconos más generales.



3.4.1.2. Primera y segunda iteración Wizard datos básicos

Se procede a realizar la construcción de las interfaces para el ingreso del usuario, así como el de los wizards para los formularios de ingreso de datos.

El wizard para el formulario de ingreso de datos básicos se realizó con un carrusel para evitar mostrar campos demasiados extensos, en vez de eso se divide el formulario en dos páginas a las cuales el usuario accede dando clic en los botones adelante y atrás para después proceder a su envío de datos al gestor en cargo.

La segunda iteración

En el caso de los wizards se procede a eliminar los botones de atrás/adelante en caso de que no redirijan a ninguna página, adjuntando otro panel para la confirmación del envío de datos. Es implementado un botón de búsqueda según el RUC de la empresa, el cual nos permite llenar automáticamente los campos de la empresa en el formulario de datos básicos.

Interfaz

Solicitud Inicio Proceso de Pasantia

Datos Estudiante - pág 1 de 2

Nombre:

Apellido:

Teléfono: *

Correo: *

Nivel más Bajo:

Actividad a Realizar: *

Tipo de Práctica

☒ Pasantia

☐ Práctica Pre Profesional

Fecha Tentativa Inicio: *

Fecha Tentativa Fin: *

Figura 24. Inicio proceso.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Interfaz

Solicitud Inicio Proceso de Pasantia

Datos Empresa - pág 2 de 2

Buscar Empresa Por RUC (Opcional):

RUC: *

Nombre Empresa: *

Nombres y Apellidos Gerente Empresa: *

Telefono Empresa: *

Dirección: *

Actividad Principal que Ejerce Empresa: *

Cedula Encargado: *

Nombres y Apellidos Encargado: *

Cargo / Profesión Encargado: *

pág 2/3

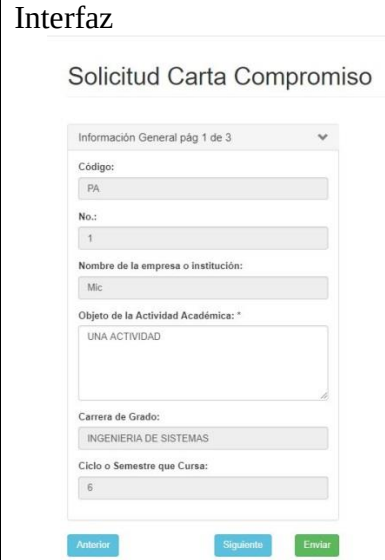
Figura 25. Inicio proceso.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

3.4.1.3. Tercera y cuarta iteración

Para el formulario de carta compromiso se procede a realizar un wizard que permita ingresar los datos necesarios o requeridos por el usuario, hay que notar algunos datos para este formulario vienen previamente cargados del proceso anterior por lo cual no son editables.

Cuarta iteración

Al botón finalizar se lo coloca en un wizard dando a notar al usuario que el ingreso de datos ha finalizado.

Interfaz	Interfaz
 Figura 26. Carta. Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)	 Figura 27. Carta. Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

3.4.1.4. Tercera y cuarta iteración

El menú para el estudiante se realiza con los enlaces más frecuentes a usar, se lista el estado del proceso y se pinta de verde cada proceso realizado.

Cuarta iteración

El menú fue modificado para retirar los enlaces que no se estén utilizando, mostrar el nombre del usuario actual y corregir las redirecciones de los enlaces.

Se incluye en el dashboard para el estudiante un acceso directo al proceso actual que tenga que realizar.

Un mapa de estado se implementa para mostrar al usuario en qué estado se encuentra su proceso actualmente y cuantas tareas le faltan para completar su inicio de pasantía.

Interfaz

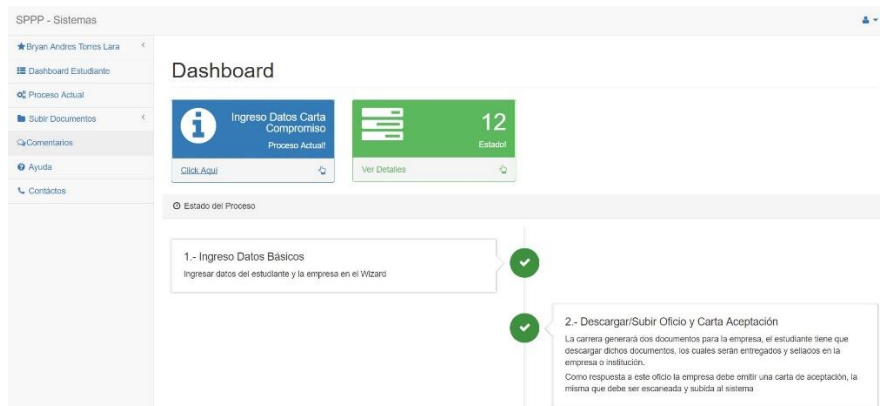


Figura 28. Dashboard.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Interfaz

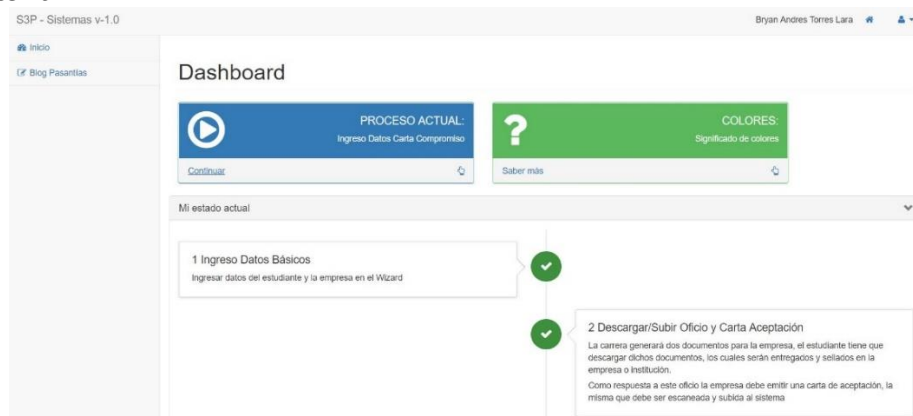


Figura 29. Dashboard.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

3.4.1.5. Quinta y sexta iteración

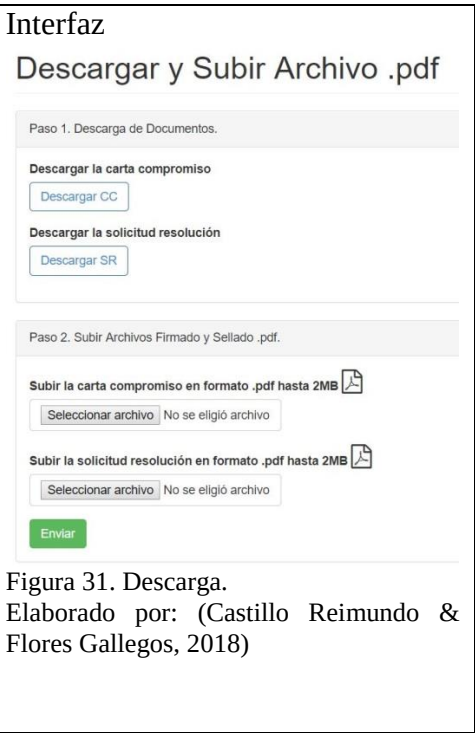
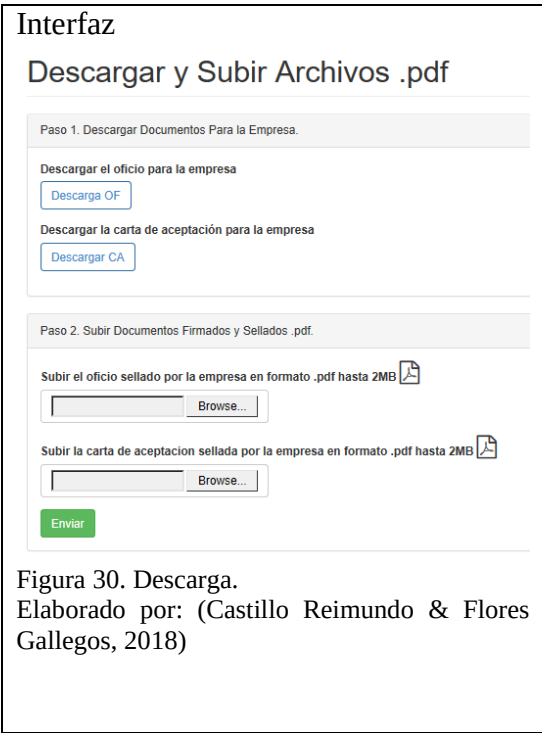
Se procede a desarrollar las interfaces de subida y descarga de archivos, en la cual se dispone de botones que permiten descargar y cargar archivos al servidor que son necesarios para validar y continuar el proceso de pasantías.

Sexta iteración

Para solicitar la interfaz de solicitud de inicio de pasantía, se diseña una página con botones que permitan descargar el archivo con los datos ya previamente

ingresados por el usuario, el documento está ya formateado y listo para ser firmado por el estudiante.

Después de descargar el archivo y firmar el presente documento, se ha colocado un botón para subir el oficio ya firmado y listo para ser validado por el gestor.



3.4.1.6. Séptima y octava iteración

Al mapa de estado del flujo de trabajo del estudiante, se le asigna un color según el estado que tenga el proceso, se solicita color azul para el proceso actual a realizar, color amarillo para un proceso en estado de validación, color plomo para procesos aun no alcanzados y color verde para los procesos ya aprobados por el estudiante.

Octava iteración

Si el gestor ha establecido que algún dato ingresado por el estudiante no es correcto, este campo indica al estudiante en color rojo para identificar que el mismo es incorrecto.

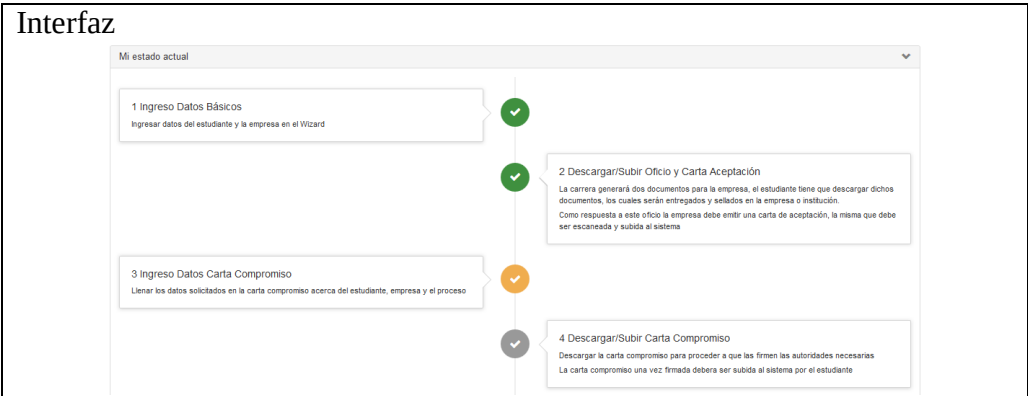


Figura 32. Colores.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Interfaz

Datos Estudiante - pág 1 de 2

Nombre:
JAIRO DANIEL

Apellido:
FLORES GALLEGOS

Teléfono: *
0983466689

Correo: *
jairdean@hotmail.com

Nivel más Bajo:
6

Figura 33. Indicador.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

3.4.2. Módulo de validación de documentación

3.4.2.1. Primera y segunda iteración

Permitir la carga de los datos de un estudiante predeterminado y mostrarlos en un formulario similar al cual se ingresaron, implementar checkbox para que el gestor pueda validar si la información de ese campo esta correcta.

Segunda iteración

Se complementa el wizard de validación con las demás pantallas necesarias para los datos ingresados por el estudiante. Los botones de adelante/siguiente solo son llamados si existe una pantalla previa o después en el panel.

Para observaciones de un proceso fallido se coloca un campo de observación, el gestor puede modificarlo o escribir en el de ser necesario para que el estudiante tenga una mejor comprensión del error cometido al ingresar sus datos.

Interfaz

Solicitud Inicio Proceso de Pasantia

Datos Estudiante

Nombre:

Karen Vanessa

Apellido:

Achig Guachamin

Teléfono: *

321

Correo: *

kuarencio14@hotmail.com

Nivel más Bajo:

5

Actividad a Realizar: *

arides

Tipo de Práctica

Pasantía

Práctica Pre Profesional

Fecha Tentativa Inicio: *

Mon Jan 15 00:00:00 COT 2018

Fecha Tentativa Fin: *

Sat Jan 20 00:00:00 COT 2018

Anterior

Siguiente

Enviar

Figura 34. Vistos.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Interfaz

Solicitud Inicio Proceso de Pasantia

Datos Estudiante

Nombre:

Karen Vanessa

Apellido:

Achig Guachamin

Teléfono: *

321

Correo: *

kuarencio14@hotmail.com

Nivel más Bajo:

5

Actividad a Realizar: *

arides

Tipo de Práctica

Pasantía

Práctica Pre Profesional

Fecha Tentativa Inicio: *

Mon Jan 15 00:00:00 COT 2018

Fecha Tentativa Fin: *

Sat Jan 20 00:00:00 COT 2018

Anterior

Siguiente

Enviar

Figura 35. Vistos.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Interfaz

Solicitud Inicio Proceso de Pasantia

Datos Empresa - pág. 2 de 2

RUC: *

172008283001

Nombre Empresa: *

VIMENWORK S.A.

Nombre y Apellidos Gerente Empresa: *

MAURICIO LOYOLA

Teléfono: *

3066549

Dirección: *

CC EL RECORO LOCAL 12

Actividad Principal que Ejerce Empresa: *

DESARROLLO IOS

Cédula Encargado: *

176447864

Nombre y Apellidos Encargado: *

MICHAEL JACKSON

Cargo Encargado: *

DEVELOPER

Observaciones al Estudiante Acerca de su Proceso

Anterior

Siguiente

Figura 36. Vistos.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Interfaz

Solicitud Inicio Proceso de Pasantia

Guardar Datos Inicio Proceso

NOTA

Si está seguro de los datos validados/negados, proceda con el envío

Enviar

pág 3/3

Anterior

Figura 37. Enviar.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

3.4.2.2. Tercera y cuarta iteración

El dashboard para el gestor tendrá contadores para cada tipo de proceso enviado a validar por los estudiantes, mientras que en la pantalla principal una tabla mostrará los últimos 10 procesos en estado de validación, con datos anexos como la cédula del estudiante, tipo de proceso y un enlace para realizar su posterior validación.

Cuarta iteración

Corrección al menú del gestor, para eliminar enlaces obsoletos, cambiar iconos que representen mejor la acción que realizan los tablonos y colocar etapas del proceso de pasantía necesarias de ser validadas.

Según el tablón elegido, se genera una tabla que contiene los datos mandados a validar solo pertenecientes a ese proceso, pueden ser validaciones de datos básicos, carta compromiso, solicitud de inicio de pasantía y validación de pdf.

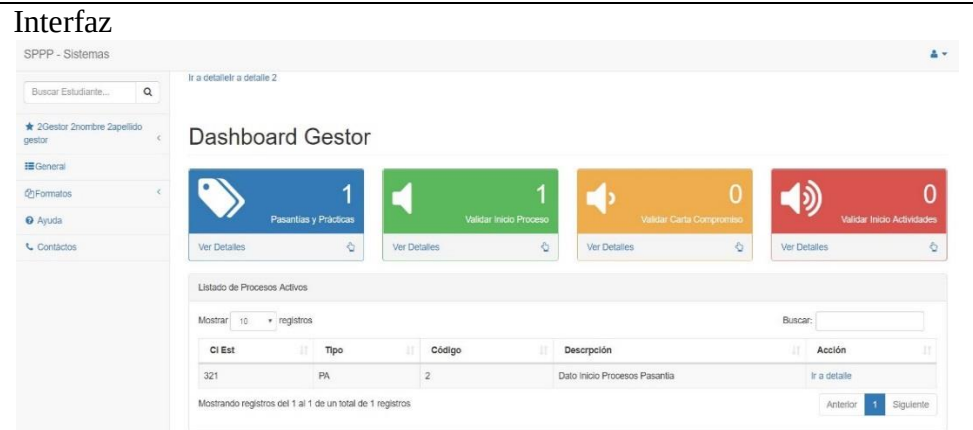


Figura 38. Gestor.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

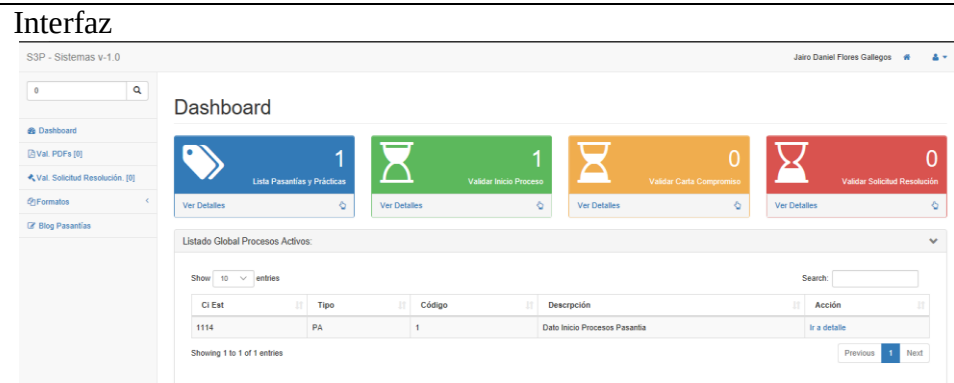


Figura 39. Gestor.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)



Figura 40. Gestor.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

3.4.2.3. Quinta y sexta iteración

Construcción de una interfaz que permita validar el contenido subido al servidor en formato pdf por parte de los estudiantes, colocando botones que permiten la

descarga de los archivos, incluir checkbox en la parte derecha para marcar con un visto los pdfs correctamente subidos por parte del usuario, incluyendo una observación en la parte inferior en caso de ser necesaria una explicación adicional.

Sexta iteración

Elaboración de una interfaz para colocar el código de resolución de inicio de la pasantía asignado por consejo de carrera, incluir una paleta de opciones para asignar un tutor en caso de que se decida reasignar un tutor para la pasantía.

Interfaz

Validar Documentos .pdf Previos a Solicitud de Inicio PA / PP

Documentos Subidos por el Alumno.

Oficio de la Empresa	Descarga OF-E	☐
Carta de Aceptación Empresa	Descarga CA-E	☐
Carta Compromiso Empresa	Descarga CO-E	☐
Solicitud Inicio / Resolución	Descarga SR-E	☐
Observación		

Validar

Figura 41. Descarga.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Interfaz

Reasignar Tutor

33nombre 33apellido

Reasignar

Ingreso Número Resolución

Código de Resolución *

12345-678-9123-45-67

Aprobar

Figura 42. Validar.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

3.4.2.4. Séptima y octava iteración

A la interfaz para asignar código de resolución, agregar una tabla de información general de la pasantía del estudiante escogido.

Octava iteración

Implementar un buscador de estudiantes, que permita ingresar la cédula del estudiante para buscar todos los detalles de la pasantía, incluyendo en una tabla todos los procesos atravesados hasta el punto actual.



Figura 43. Validar proceso.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

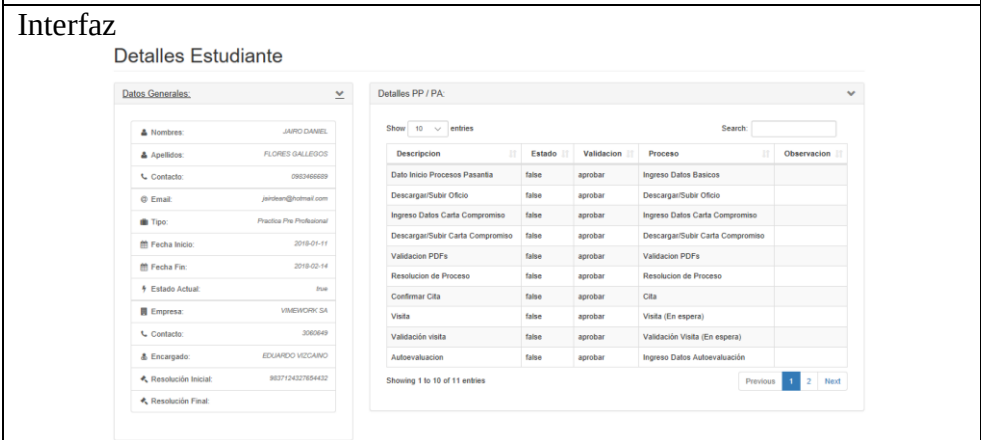


Figura 44. lista proceso individual.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

3.5. Estructura de paquetes y clases

La aplicación SPPP se ha desarrollado en JSF, con un patrón MVC (Modelo-Vista-Controlador) para poder identificar y separar los diversos componentes de esta en los paquetes necesarios según la función que cumplan las diversas clases.

Estructura paquetes

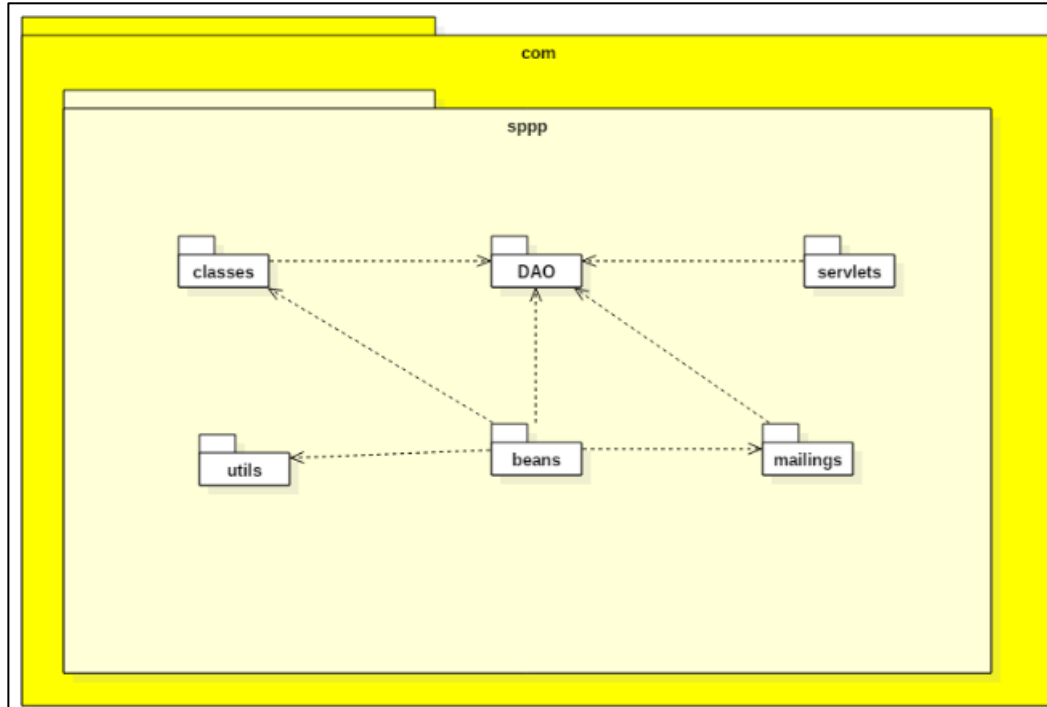


Figura 45. Diagrama de paquetes.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

3.5.1. Paquete: com.sppp.beans

Es el paquete principal de la aplicación contiene los backing beans que permiten interactuar con la parte de la vista, tanto para enviar datos como para recibirlos.

Contiene las entidades que van a ser modeladas en el sistema con sus respectivos ficheros hbm.xml para dar configuraciones adicionales al momento de ser creadas en la base de datos.

Diagrama clases

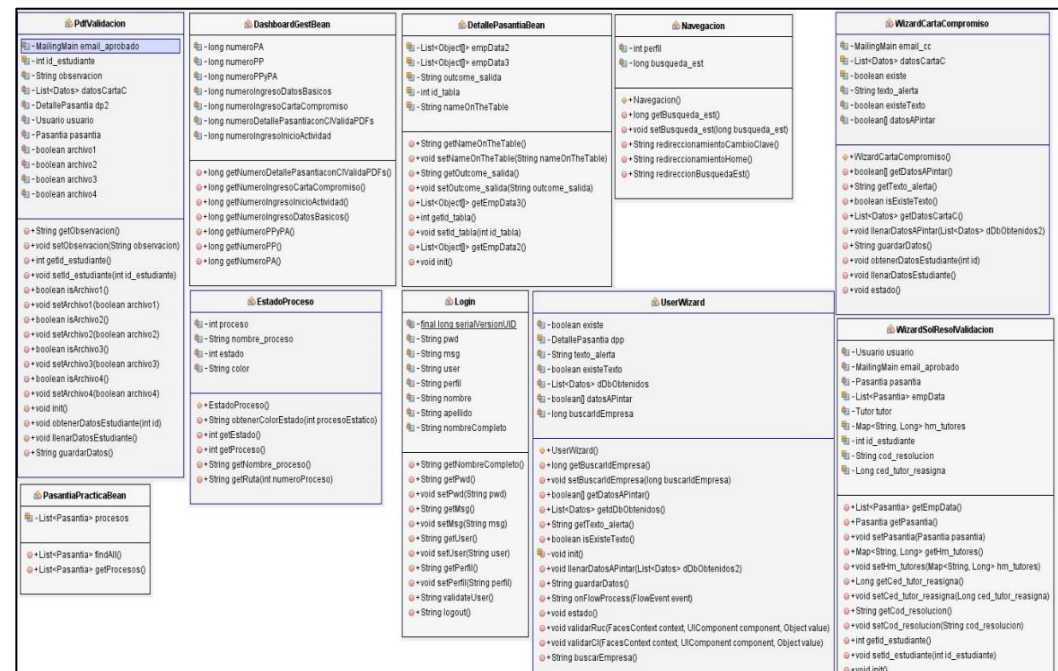


Figura 46. Diagrama de clases.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Tabla 48. Paquete com.sppp.beans

Clase: DashboardGestBean

Permite obtener el número de diversos procesos realizados por los estudiantes que se han mandado a validar, manda a ejecutar los métodos de consulta y retribuye el dato a la vista.

Clase: DetalleEstudianteBean

Es una clase que recibe un id de un estudiante, y manda a realizar las consultas necesarias para obtener los datos generales respecto a ese estudiante, información como pasantía y detalle de la pasantía es retribuida para su posterior uso en vistas.

Clase: DetallePasantiaBean

Esta clase recibe un id que le permite identificar qué tipo de tabla va a renderizar, de acuerdo al id recibido manda a pedir los datos de un proceso en específico.

Clase: EstadoProceso

Toma el id guardado en la sesión actual, con el id obtenido busca el proceso y estado actual de ese proceso para el usuario actual. Según el estado establece el color será renderizado en el mapa del flujo de trabajo en la vista.

Clase: Login

Obtiene las credenciales ingresadas por el usuario, manda a realizar el proceso de validación, según el usuario y perfil obtenido genera una redirección al dashboard del usuario.

Clase: Navegación

Busca el perfil del usuario actual en el sistema, según este perfil obtenido genera paths de redirecciones según sea de donde se lo invoque.

Clase: PasantiaPracticaBean

Esta clase permite obtener datos de todas las pasantías registradas al momento.

Clase: PdfEstudiante

Obtiene los datos de los archivos pdfs que han sido validados/negados por el gestor incluyendo observaciones en caso de haberlas, permite el registro y subida de nuevos pdfs para completar el proceso en este paso.

Clase: PdfValidacion

Obtiene los archivos pdfs subidos por el estudiante y permite su descarga/visualización para que el gestor decida si están correctos, guarda la validación de los documentos incluyendo notas de observaciones.

Clase: UserWizard

Permite obtener y establecer datos del primer formulario que el estudiante debe llenar, consta de un método guardar que permite insertar los datos si no existen o actualizarlos en caso de que ya existen. Retribuye los datos si estos ya existen.

Clase: WizardCC

Es una clase abstracta que tiene atributos en común, que comparten dos clases una dedicada para estudiantes y otra para gestores.

Clase: WizardCCValidacion

Hereda atributos de WizardCC, permite obtener datos previamente mandados a validar por el estudiante, tiene un método guardar que permite el almacenamiento de los datos que fueron considerados correctos y de los datos que fueron considerados incorrectos.

Clase: WizardCartaCompromiso

Hereda atributos de WizardCC, permite establecer y obtener datos para este formulario, contiene un método guardar que almacena los datos insertados en los campos de los formularios, realiza la comprobación de si ya existen datos para actualizarlos, caso contrario los inserta por primera vez.

Clase: WizardDB

Es una clase abstracta con atributos en común los cuales son compartidos para formularios de estudiante y gestor.

Clase: WizardDBValidacion

Hereda atributos de WizardDB, permite obtener datos ingresados

previamente por el estudiante para ser validados por el gestor. El método guardar permite almacenar los campos que fueron marcados como erróneos/correctos.

Clase: WizardSolResolValidacion

Permite actualizar el tutor asignado a la pasantía e ingresar el código de resolución de inicio de pasantía, con lo cual ejecuta el cambio de proceso para empezar a realizar el seguimiento de esta.

Nota: Esta tabla contiene información acerca de las clases dentro del paquete beans.

3.5.2. Paquete: com.sppp.DAO

Tabla 49. Paquete: com.sppp.DAO

Clase: DatosDAO

Consultas HQL para obtener registros de la tabla datos, realizar el ingreso, actualización de estos.

Clase: DetallePasantiaDAO

Consultas HQL para obtener, actualizar, ingresar registros de la tabla detalle pasantía.

Clase: EmpresaDAO

Consultas HQL para obtener, actualizar, ingresar registros de la tabla empresa.

Clase: EncargadoDAO

Consultas HQL para obtener, actualizar, ingresar registros de la tabla detalle pasantía.

Clase: EstudianteDAO

Consultas HQL para obtener, actualizar, ingresar registros de la tabla estudiante.

Clase: LoginDAO

Consulta HQL para obtener un usuario validando sus credenciales ingresadas.

Clase: PasantiaDAO

Consultas HQL para obtener, actualizar, ingresar registros de la tabla pasantía.

Clase: PasantiaPracticaDAO

Consultas HQL generales ya sea para una pasantía o una práctica pre profesional.

Clase: PeriodoDAO

Consultas HQL para obtener, actualizar, ingresar registros de la tabla periodo.

Clase: ProcesoDAO

Consultas HQL para obtener, actualizar, ingresar registros de la tabla proceso.

Clase: UsuarioDAO

Consultas HQL para obtener, actualizar, ingresar registros de la tabla usuario.

Clase: WizardDAO

Permite guardar datos básicos del wizard inicial para el estudiante.

Nota: Esta tabla contiene información acerca de las clases dentro del paquete DAO.

3.5.3. Paquete: com.sppp.utils

Tabla 50. Paquete: com.sppp.utils

Clase: Cls_OutOfSystem

Esta clase recibe el email como parámetro para encontrar la contraseña del usuario y enviarla por email.

Clase: HibernateUtil

Permite la comunicación con Hibernate generando solo un SessionFactory.

Clase: LoginFilter

Filtra todas las peticiones al sistema, para bloquear el ingreso a páginas de un perfil a usuarios con perfil diferente y en caso de ser correcto los redirecciona a sus dashboard iniciales.

Clase: PasswdReplacement

Permite cambiar la clave del usuario y hacer la redirección a la interfaz principal del usuario según su perfil.

Clase: PasswordRestoration

Permite el acceso a los métodos de la clase Cls_OutOfSystem para encontrar la contraseña del usuario.

Clase: SessionUtils

Permite obtener de una manera menos complicada los datos de la sesión actual en el sistema y los atributos guardados en la misma.

Clase: UploadFile

Esta clase permite el subir archivo al servidor, el archivo escogido por los estudiantes es creado en una carpeta identificada por su número de cédula, todos los archivos subidos son almacenados en esta carpeta.

Clase: Validaciones

Recibe un parámetro de tipo long en su método de validación de cédula, para encontrar que el número ingresado sea en efecto el de una cédula correcta y devuelva un true/false.

Nota: Esta tabla contiene información acerca de las clases dentro del paquete utils.

3.5.4. Paquete: com.sppp.mailing

Tabla 51. Paquete: com.sppp.mailing

Clase: MailingMain

Permite el envío de correos electrónicos según sea el caso que se necesite, puede ser para la corrección de datos de un estudiante, así como la asignación de una nueva tarea a algún gestor.

Nota: Esta tabla contiene información acerca de las clases dentro del paquete mailing.

3.5.5. Paquete: com.sppp.classes

Tabla 52. Paquete: com.sppp.classes

Clase: AlmacenamientoPDF Genera los archivos PDFS, a través de un parámetro que permite elaborar un tipo específico de PDF con información previamente ingresada del estudiante.
Clase: ChangePassword Permite cambiar la contraseña en el sistema de los usuarios que envíen como parámetro su id, contraseña actual y nueva contraseña.
Clase: ClsPerfilNotation A través de un parámetro de perfil, se regresa en una cadena el nombre de ese perfil.
Clase: ListaDocentesAdministrativos Permite establecer y obtener los datos de los docentes administrativos que se encuentran incluidos dentro del proceso de pasantía.
Clase: Paths Ubicaciones en el servidor donde se almacenarán todos los registros de los usuarios que ingresan a la aplicación.

Nota: Esta tabla contiene información acerca de las clases dentro del paquete classes.

3.5.6. Paquete: com.sppp.servlets

Tabla 53. Paquete: com.sppp.servlets

Clase: Download Permite la descarga y carga de archivos pdf para el perfil estudiante, recibe como parámetros un archivo y un id del estudiante.
Clase: DownloadGestP1 Permite solo la descarga de archivos en formato pdf para el perfil gestor, recibe como parámetros un id del estudiante y el tipo de archivo pdf a mostrar.

Nota: Esta tabla contiene información acerca de las clases dentro del paquete servlets.

3.6.Código

A continuación, se presenta el código más relevante de la aplicación, a su vez se podrá apreciar las configuraciones y comandos relevantes.

3.6.1. Web.xml configuraciones Web.

Se describe la configuración del despliegue de la aplicación Web.

El siguiente código permite seleccionar un tema que no sea Bootstrap, ya que se está utilizando Sb Admin 2. Se deshabilita la llamada al bootstrap-theme.css. Hay que eliminar los componentes CSS asociados a Bootstrap, y el bsf.css el cual es un componente de BootsFaces.

Tabla 54. Web.xml

<pre><context-param> <param-name>BootsFaces_THEME</param-name> <param-value>custom</param-value> </context-param> <context-param> <param-name>BootsFaces_USETHEME</param-name> <param-value>>false</param-value> </context-param> <context-param> <param-name>net.bootsfaces.get_bootstrap_from_cdn</param-name> <param-value>>true</param-value> </context-param></pre>
--

Nota: Esta tabla contiene información acerca de parte del código de web.xml

3.6.2. Configuración de Hibernate.

El siguiente código describe la configuración para el gestor de base de datos.

Update: Permite actualizar las tablas de la base y es la función normal de la misma, en caso de crearla por primera vez, su argumento seria la palabra “create”.

Tabla 55. Gestor de base de datos

<pre><property name="hibernate.dialect">org.hibernate.dialect.PostgreSQLDialect</property> <property name="hibernate.connection.driver_class">org.postgresql.Driver</property> <property name="hibernate.connection.url">jdbc:postgresql://localhost:5432/db_sp3</property> > <property name="hibernate.connection.username">postgres</property> <property name="hibernate.connection.password">****</property> <property name="hbm2ddl.auto">update</property> <property name="show_sql">true</property> <property name="format_sql">true</property> <property name="hibernate.current_session_context_class">thread</property></pre>

Nota: Esta tabla contiene información acerca de parte del código del gestor de base de datos.

Se procede a llamar a cada uno de los BEAN que se ha mapeado en la aplicación Web.

Tabla 56. Configuración de mapeo

<mapping resource="com/sppp/beans/perfil.hbm.xml"/>
<mapping resource="com/sppp/beans/usuario.hbm.xml"/>
<mapping resource="com/sppp/beans/estudiante.hbm.xml"/>
<mapping resource="com/sppp/beans/proceso.hbm.xml"/>
<mapping resource="com/sppp/beans/detallePasantia.hbm.xml"/>
<mapping resource="com/sppp/beans/pasantia.hbm.xml"/>
<mapping resource="com/sppp/beans/encargado.hbm.xml"/>
<mapping resource="com/sppp/beans/empresa.hbm.xml"/>
<mapping resource="com/sppp/beans/datos.hbm.xml"/>
<mapping resource="com/sppp/beans/respuesta.hbm.xml"/>
<mapping resource="com/sppp/beans/preguntas.hbm.xml"/>
<mapping resource="com/sppp/beans/tipoPregunta.hbm.xml"/>
<mapping resource="com/sppp/beans/formato.hbm.xml"/>
<mapping resource="com/sppp/beans/periodoTutor.hbm.xml"/>
<mapping resource="com/sppp/beans/horarioTutor.hbm.xml"/>
<mapping resource="com/sppp/beans/tutor.hbm.xml"/>
<mapping resource="com/sppp/beans/visitaTutor.hbm.xml"/>
<mapping resource="com/sppp/beans/periodo.hbm.xml"/>

Nota: Esta tabla contiene información acerca de parte del código de la configuración del mapeo.

3.6.3. Código de control de usuario por perfil.

Se presenta como se filtra un usuario al momento de ingresar a la aplicación con sus respectivas credenciales, para este caso se toma como ejemplo solo al perfil estudiante en la clase LoginFilter.java.

Tabla 57. Enrutar perfil

<pre> if(obj1.perfilChooser(perfil).equals("Estudiante")){ //Para acceder a paginas de estudiante o publicas if(reqURI.indexOf("/user/administrador/") >= 1 reqURI.indexOf("/user/consejo/") >= 1 reqURI.indexOf("/user/coordinador/") >= 1 reqURI.indexOf("/user/gestores/") >= 1 reqURI.indexOf("/user/secretaria/") >= 1 reqURI.indexOf("/user/tutor/") >= 1){ resp.sendRedirect(reqt.getContextPath() + "/faces/user/estudiantes/dashboard_est.xhtml");} if(reqURI.indexOf("/user/estudiantes") >= 0 reqURI.indexOf("/public/") >=0 reqURI.contains("javax.faces.resource")){ if (reqURI.indexOf("/public/login.xhtml") >= 0){ resp.sendRedirect(reqt.getContextPath() + "/faces/user/estudiantes/dashboard_est.xhtml"); }else{ chain.doFilter(request, response);}}} </pre>

Nota: Esta tabla contiene información acerca de parte del código de como enrutar un perfil.

Se adjunta el método de selección de perfil, de acorde a cada sesión en la clase

Cls_PerfilNotation.java.

Tabla 58. Analizar pertenencia de path

```
public String perfilChooser(int recive_perfil) {String profile_name="0";  
if (recive_perfil==1){return profile_name="Estudiante";  
}else if(recive_perfil==2){return profile_name="Gestor";  
}else if(recive_perfil==3){return profile_name="Tutor";  
}else if(recive_perfil==4){return profile_name="Consejo";  
}else if(recive_perfil==5){return profile_name="Secretaria";  
}else if(recive_perfil==6){return profile_name="Coordinador";  
}else if(recive_perfil==7){return profile_name="Administrador";}return  
profile_name;}  

```

Nota: Esta tabla contiene información acerca de parte del código del path al que se conecta.

Ahora a cada sesión se la valida, para saber a qué path pertenece y tiene acceso

en la clase Login.java.

Tabla 59. Ruteo a página principal del perfil

```
if (obj1.perfilChooser(perfil).equals("Estudiante")){  
return "/user/estudiantes/dashboard_est.xhtml";  
}else if (obj1.perfilChooser(perfil).equals("Gestor")){  
return "/user/gestores/dashboard_gestor.xhtml";  
}else if (obj1.perfilChooser(perfil).equals("Tutor")){  
return "/user/tutor/tutor.xhtml";  
}else if (obj1.perfilChooser(perfil).equals("Consejo")){  
return "/user/consejo/dashboard_consejo.xhtml";  
}else if (obj1.perfilChooser(perfil).equals("Secretaria")){  
return "/user/secretaria/dashboard_secretaria.xhtml";  
}else if (obj1.perfilChooser(perfil).equals("Coordinador")){  
return "/user/coordinador/coordinator.xhtml";  
}else if (obj1.perfilChooser(perfil).equals("Administrador")){  
return "/user/administrador/administrator.xhtml";}  
return "/public/login";  

```

Nota: Esta tabla contiene información acerca de parte del código de como redireccionar al dashboard.

3.6.4. Configuración del Servlet Download.java para subir y bajar archivos.

Líneas esenciales del servlet

Tabla 60. Configuración servlet para archivos

```
@WebServlet(name = "Download", urlPatterns = {"/user/estudiantes/download/*"})  
  
@Override  
protected void doGet(HttpServletRequest request, HttpServletResponse  
response)  
throws ServletException, IOException {  

```

```
String filename = request.getPathInfo().substring(1);
Paths obtener_path= new Paths();
File file = new File(obtener_path.local_path()+"/"+id, filename);
response.setHeader("Content-Type", getServletContext().getMimeType(filename));
response.setHeader("Content-Length", String.valueOf(file.length()));
response.setHeader("Content-Disposition", "inline; filename=\"\" + filename +
\"\"");
Files.copy(file.toPath(), response.getOutputStream());}
```

Nota: Esta tabla contiene información acerca de cómo se manejan los archivos.

Las líneas que se presenta a continuación permiten subir y bajar archivos desde y hacia el servidor, su almacenamiento se lo hace en un directorio establecido dentro del mismo, dichos métodos se encuentran en la clase UploadFile.java.

Líneas para subir un archivo al servidor

Tabla 61. Obtener un File de un Path

```
String local_path = directorio.local_path();
String nombre;
String nombreSinExt, extension;
Long tamano;
int punto;

try {
    InputStream input = file.getInputStream();
    nombre = getFilename(file);
    tamano = getFile().getSize();
    //CODIGO PARA OBTENER LA EXTENSION DEL ARCHIVO
    punto = nombre.indexOf(".");
    nombreSinExt = nombre.substring(0, punto);
    extension = nombre.substring(punto, nombre.length());
    System.out.println(nombreSinExt+" "+extension+" "+tamano);

    Files.copy(input, new File(local_path + id, "1" + extension).toPath(),
    StandardCopyOption.REPLACE_EXISTING);}
```

Nota: Esta tabla contiene información acerca de parte del código de como obtener dirección de un archivo.

Líneas para descargar un archivo

Tabla 62. Descargar un archivo

```
FacesContext facesContext = FacesContext.getCurrentInstance();
ExternalContext context = facesContext.getExternalContext();
HttpServletRequest request = (HttpServletRequest) context.getRequest();
HttpServletResponse response = (HttpServletResponse) context.getResponse();
//mando a llamar a un PDF que se ha creado con anterioridad y esta listo a
descargar
response.sendRedirect(request.getContextPath()
"/faces/user/estudiantes/download/" + opcion + ".pdf");
```

Nota: Esta tabla contiene información acerca de parte del código de descarga de archivo.

3.6.5. Configuración de la clase de mensajería instantánea (email).

Tabla 63. Envío email

```
private void deliverMail(String AddRecipientDestination, String AddSubject,String
contentMessage){
try {
    Properties props = System.getProperties();
    props.setProperty("mail.transport.protocol", "smtp");
    props.setProperty("mail.host", "smtp.gmail.com");
    props.put("mail.smtp.auth", "true");
    props.put("mail.smtp.port", "465");
    props.put("mail.debug", "true");
    props.put("mail.smtp.socketFactory.port", "465");

    props.put("mail.smtp.socketFactory.class","javax.net.ssl.SSLSocketFactory");
    props.put("mail.smtp.socketFactory.fallback", "false");

    Session emailSession = Session.getInstance(props,new
    javax.mail.Authenticator() {
        protected PasswordAuthentication getPasswordAuthentication() {return new
        PasswordAuthentication("emailaddress", "password");
        }
    });
    emailSession.setDebug(true);
    Message message = new MimeMessage(emailSession);
    message.setFrom(new InternetAddress(username));
    message.setRecipients(Message.RecipientType.TO, InternetAddress.parse(AddRecip
    ientDestination));
    message.setSubject(AddSubject);
    message.setText(contentMessage);

    Transport transport = emailSession.getTransport("smtps");
    transport.connect("smtp.gmail.com", username, password);
    transport.sendMessage(message, message.getAllRecipients());
} catch (MessagingException e) {}
```

Nota: Esta tabla contiene información acerca de parte del código de envío de emails.

3.6.6. Almacenamiento y recuperación de datos.

En el siguiente cuadro se aprecia el código que permite guardar los datos desde el Wizard de carta compromiso hacia la base, este método pertenece a la clase WizardCartaCompromiso.java. Al momento de guardar los datos se obtienen las variables que han sido llenadas por el alumno, se obtiene una información de que la data sea actualizada en caso de ya existir, y por último se actualiza el estado del proceso, permitiendo interactuar entre alumno y gestor.

Tabla 64. Guardar objeto en base de datos

```
public String guardarDatos(){
    String redireccion = "revision_window";
    //Recolectando la informacion
    pasantia.getTipo_ppp();
    pasantia.getCod_ppp();
    empresa.getNombre_empresa();
```



```

        empresa.getDireccion_empresa();
        empresa.getTelefono_empresa();
        empresa.getActividad_principal_empresa();

        //Recolectando info estudiante
        usuario.getApellido();
        usuario.getNombre();
        usuario.getEstudiante().getUltimoNivel();

        usuario.getEstudiante().getHorasPasantia();
        pasantia.getFechaInicio();
        pasantia.getFechaFin();

        encargado.getNombre_encargado();
        //Nombre Gerente
        empresa.getNombre_gerente();

        List<String> resp = new LinkedList<>();
        resp.add(carrera_grado); //0
        resp.add(tipo_actividad_academica); //1
        resp.add(objetoDeLaActividad); //2
        resp.add(horarioPrevisto); //3
        resp.add(nombrePrograma); //4
        resp.add(areaAcademica); //5
        resp.add(actividadPrevista); //6
        resp.add(resultadosPrevistos); //7
        resp.add(productosEntregables); //8
        resp.add(cargo); //9
        resp.add(delegadoUPS); //10
        resp.add(cargoDelegado); //11
        resp.add(telefonoDelegado); //12
        resp.add(lugarFecha); //13
        //Obtener idDetallePasantia
        DetallePasantia dp2 = new DetallePasantia();
        DetallePasantiaDAO dpDAO = new DetallePasantiaDAO();
        dp2 = dpDAO.findDetallePasantia(pasantia.getTipo_ppp(),
pasantia.getCod_ppp());
        //Guardar en la tb_datos;
        DatosDAO obj = new DatosDAO();
        pasantia.setCed_tutor_asignado(objen_ci_tutor);
        // si no existe se guarda caso contrario se actualiza
        if(existe){
            obj.actualizarDatosCartaCompromiso(usuario, empresa, encargado, pasantia, dp2,
            resp);
        }else{
            obj.datosGuardar(usuario, empresa, encargado, pasantia, dp2, resp,getTutorXXX()
            );
        }
        //Mando a estado validar el registro en la DB
        dp2.setValidacion(EnumEstado.validar);
        dpDAO.actualizarDetallePasantia(dp2);
        return redireccion;
    }

```

Nota: Esta tabla contiene información acerca de parte del código de como persistir un objeto.

En el siguiente cuadro se aprecia el código que permite obtener los datos desde el Wizard de carta compromiso hacia la base, estos métodos se encuentran en la clase WizardCartaCompromiso.java. Influyen tres métodos, los dos que se indican son los más importantes, ya que el método

llenarDatosEstudiante() obtiene los datos necesarios que pertenecen al Wizard, a su vez se obtiene los datos de la pasantía, mientras que otro método permite saber los datos que son de validar del alumno.

Tabla 65. Recuperar datos de la base como objetos

```
public void obtenerDatosEstudiante(int id) {
    DatosDAO dDAO = new DatosDAO();
    datosCartaC = dDAO.datosPorDetallePasantia(id);
    //Lleno el vector auxiliar de Pintar
    llenarDatosAPintar(datosCartaC);

    llenarDatosEstudiante();
}

public void llenarDatosEstudiante(){
    lugarFecha = datosCartaC.get(28).getValor_datos();

    objetoDeLaActividad = datosCartaC.get(11).getValor_datos();
    horarioPrevisto = datosCartaC.get(15).getValor_datos();
    nombrePrograma = datosCartaC.get(16).getValor_datos();
    areaAcademica = datosCartaC.get(17).getValor_datos();
    actividadPrevista = datosCartaC.get(19).getValor_datos();
    resultadosPrevistos = datosCartaC.get(20).getValor_datos();
    productosEntregables = datosCartaC.get(21).getValor_datos();
}
```

Nota: Esta tabla contiene información acerca de parte del código de recuperación de datos.

3.7.Pruebas

A continuación, se presenta una serie que influyen en el rendimiento de la aplicación Web, adicional se indica las pruebas contra requerimientos, estas pruebas han sido verificadas y criticadas por el usuario final, dando así un mayor enfoque al cumplimiento de los requisitos de la aplicación, las pruebas unitarias asegurando que su funcionamiento sea correcto, por último, se presenta las pruebas de estrés con la ayuda de Jmeter.

3.7.1. Pruebas unitarias

Las pruebas unitarias permiten evaluar la respuesta de un método conforme a los parámetros asignados previamente, el método realizar sus instrucciones

normalmente hasta devolver una respuesta la cual será comparada con un dato previamente asignado para validar el correcto funcionamiento del método.

Para ejecutar pruebas unitarias en NetBeans utilizaremos JUnit que viene ya precargado en el IDE, se escoge las clases que serán puestas a prueba y se crea una clase Test en un paquete de pruebas, esta clase permite al usuario establecer parámetros que pueden ser cargados antes de ser ejecutada una prueba o antes de ejecutar toda la clase de pruebas.

Las pruebas se han realizado en métodos que requieren más recursos, así como parámetros y tiempos para obtener los datos necesarios que permitirán continuar con la ejecución del método hasta obtener una respuesta.

Tabla 66. Métodos Clase DashboardGestBean

N.-	Función	Entrada	Tiempo	Resp.
1	TestGetNumeroPPyPA	Ninguna	2.01	Ok
2	testGetNumeroIngresoInicioActividad	Ninguna	0.008	Ok
3	testGetNumeroPA	Ninguna	0.005	Ok
4	testGetNumeroPP	Ninguna	0.001	Ok
5	testGetNumeroIngresoDatosBasicos	Ninguna	0.001	Ok

Nota: Esta tabla contiene información acerca de datos de pruebas realizadas a métodos.

Tabla 67. Métodos Clase DetalleEstudianteBean

N.-	Función	Entrada	Tiempo (s)	Resp.
1	testNombreProceso	Id_proceso	0.008	Ok
2	testGetPasantia	Ninguna	0.001	Ok
3	testGetTipo	Ninguna	0.001	Ok
4	testGetUsuario	Ninguna	0.001	Ok
5	testGetListaDetalles	Ninguna	0.001	Ok
6	testGetEncargado	Ninguna	0.001	Ok

Nota: Esta tabla contiene información acerca de datos de pruebas realizadas a métodos

Tabla 68. Métodos Clase WizardCCValidacion

N.-	Función	Entrada	Tiempo	Resp.
1	testGuardarDatos	Ninguna	2.259	Ok

Nota: Esta tabla contiene información acerca de datos de pruebas realizadas a métodos

Tabla 69. Métodos Clase Mailing

N.-	Función	Entrada	Tiempo	Resp.
1	testMensajes	tipo_mensaje, AddRecipientDestination, observaciones	2.835	Ok

Nota: Esta tabla contiene información acerca de datos de pruebas realizadas a métodos

3.7.2. Pruebas de estrés

El aplicativo Web fue probado con diversos escenarios para comprobar como maneja la carga de usuarios y los tiempos en que realiza cada uno de los procesos, siendo establecido que el aplicativo deberá soportar 100 usuarios concurrentes y que el tiempo de respuesta del aplicativo no deberá superar los 3 segundos.

La herramienta Apache JMeter permite configurar los diversos escenarios con varios usuarios, en esta aplicación se configuró un grupo de usuarios que ingresarán a la pantalla de acceso del sistema, ingresarán sus credenciales y el aplicativo responderá desplegando la página principal del usuario con perfil establecido.

Prueba 1

Se establecen 50 usuarios de carga en la aplicación que ingresarán a la página principal del sistema con sus credenciales para acceder al tablón personal. Se busca obtener los tiempos de respuesta del servidor con el ingreso simultáneo de usuarios, para obtener valores en respuesta.

Tabla 70. Estadísticas JMeter 50 Usuarios

Descripción	Ejemp.	Prom(ms)	Min(ms)	Max(ms)	Error %	Pedidos/seg.
Login Request	50	114	18	243	0.00	42/seg
Autenticación	50	169	64	309	0.00	37.8/seg

Nota: Esta tabla contiene información acerca de datos de pruebas con 50 usuarios.

La siguiente ilustración indica que el acceso de los usuarios fue exitoso, dando a conocer que el sistema no tuvo caídas al momento de la prueba de estrés.

Acceso de usuarios

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	23.09.36.685	Grupo de Hilos 1-1	Initial Login Request	18	✓	5048	0	18	0
2	23.09.36.705	Grupo de Hilos 1-2	Initial Login Request	19	✓	5048	0	19	0
3	23.09.36.725	Grupo de Hilos 1-3	Initial Login Request	24	✓	5052	0	24	0
4	23.09.36.709	Grupo de Hilos 1-1	Auth	64	✓	26981	0	64	0
5	23.09.36.746	Grupo de Hilos 1-4	Initial Login Request	32	✓	5050	0	32	0
6	23.09.36.766	Grupo de Hilos 1-5	Initial Login Request	31	✓	5055	0	31	0
7	23.09.36.728	Grupo de Hilos 1-2	Auth	77	✓	26981	0	77	0
8	23.09.36.751	Grupo de Hilos 1-3	Auth	67	✓	26991	0	67	0
9	23.09.36.807	Grupo de Hilos 1-7	Initial Login Request	46	✓	5050	0	46	0
10	23.09.36.789	Grupo de Hilos 1-6	Initial Login Request	79	✓	5050	0	79	0
11	23.09.36.828	Grupo de Hilos 1-8	Initial Login Request	37	✓	5048	0	37	0
12	23.09.36.800	Grupo de Hilos 1-5	Auth	80	✓	26986	0	80	0
13	23.09.36.781	Grupo de Hilos 1-4	Auth	119	✓	26986	0	119	0
14	23.09.36.849	Grupo de Hilos 1-9	Initial Login Request	56	✓	5048	0	56	0
15	23.09.36.855	Grupo de Hilos 1-7	Auth	74	✓	26981	0	72	0
16	23.09.36.867	Grupo de Hilos 1-6	Auth	72	✓	26986	0	71	0
17	23.09.36.886	Grupo de Hilos 1-11	Initial Login Request	60	✓	5050	0	60	0
18	23.09.36.906	Grupo de Hilos 1-12	Initial Login Request	64	✓	5050	0	64	0
19	23.09.36.867	Grupo de Hilos 1-8	Auth	105	✓	26986	0	105	0
20	23.09.36.866	Grupo de Hilos 1-10	Initial Login Request	127	✓	5052	0	127	0
21	23.09.36.847	Grupo de Hilos 1-14	Initial Login Request	55	✓	5050	0	55	0
22	23.09.36.908	Grupo de Hilos 1-9	Auth	112	✓	26986	0	111	0
23	23.09.36.968	Grupo de Hilos 1-15	Initial Login Request	68	✓	5050	0	68	0
24	23.09.36.987	Grupo de Hilos 1-16	Initial Login Request	54	✓	5048	0	54	0

Figura 47. Estadísticas por cada solicitud.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Se observa que, a través de las dos acciones, no se dio ningún error al realizar las peticiones o envíos de credenciales, el promedio del tiempo que demora un usuario en realizar un login request es de 114 (ms) y una autenticación es de 169 (ms).

Diagrama

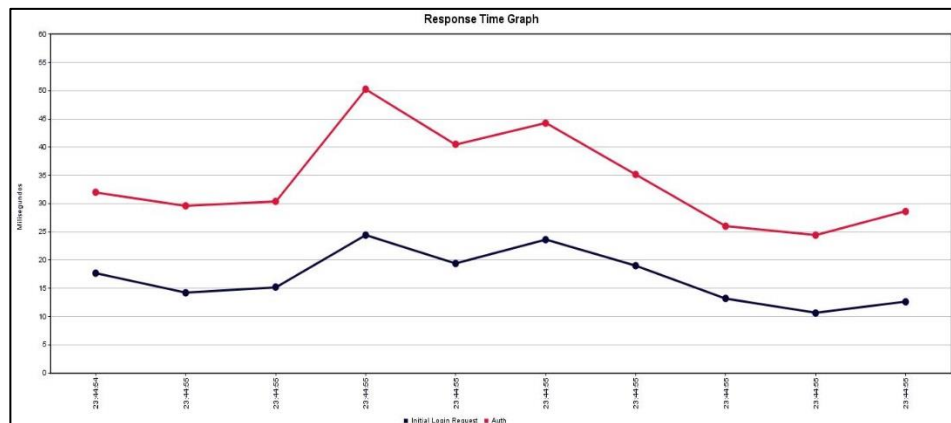


Figura 48. Tiempo de Respuesta.

Prueba 2

Se establecen 100 usuarios que ingresarán a la página principal del sistema e ingresarán sus credenciales para acceder a su tablón personal.

Se desea observar la respuesta del servidor con una carga mayor a la establecida, con el fin de ver si es capaz de soportar un crecimiento de usuarios.

Tabla 71. Estadísticas JMeter 100 Usuarios

Descripción	Ejemp.	Prom(ms)	Min(ms)	Max(ms)	Error %	Pedidos/seg.
Login Request	100	338	20	629	0.00	61.7/seg
Autenticación	100	478	75	807	0.00	46/seg

Nota: Esta tabla contiene información acerca de datos de pruebas con 100 usuarios.

La siguiente ilustración permite evidenciar que, con la carga ingresada, el sistema funciona y no tiene caídas en su ejecución.

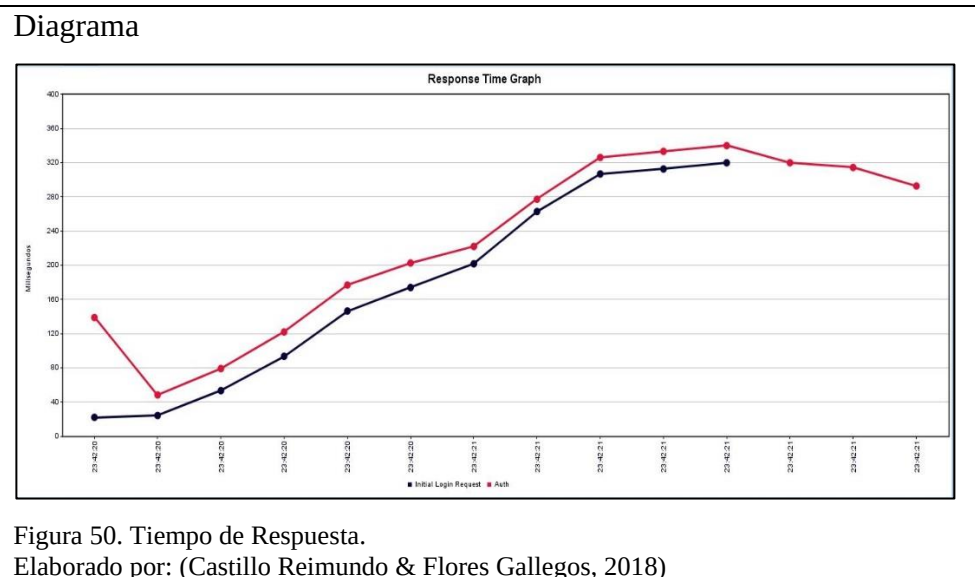
Carga de ingreso

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	23:27:29.108	Grupo de Hilos 1-1	Initial Login Request	35	✓	5048	0	35	0
2	23:27:29.118	Grupo de Hilos 1-2	Initial Login Request	26	✓	5048	0	26	0
3	23:27:29.126	Grupo de Hilos 1-3	Initial Login Request	20	✓	5050	0	20	0
4	23:27:29.137	Grupo de Hilos 1-4	Initial Login Request	21	✓	5048	0	21	0
5	23:27:29.149	Grupo de Hilos 1-5	Initial Login Request	25	✓	5050	0	25	0
6	23:27:29.161	Grupo de Hilos 1-6	Initial Login Request	50	✓	5050	0	50	0
7	23:27:29.151	Grupo de Hilos 1-3	Auth	82	✓	26989	0	81	0
8	23:27:29.148	Grupo de Hilos 1-1	Auth	85	✓	26981	0	84	0
9	23:27:29.162	Grupo de Hilos 1-4	Auth	75	✓	26981	0	73	0
10	23:27:29.148	Grupo de Hilos 1-2	Auth	94	✓	26989	0	93	0
11	23:27:29.199	Grupo de Hilos 1-10	Initial Login Request	70	✓	5048	0	70	0
12	23:27:29.189	Grupo de Hilos 1-9	Initial Login Request	83	✓	5050	0	83	0
13	23:27:29.179	Grupo de Hilos 1-8	Initial Login Request	97	✓	5050	0	97	0
14	23:27:29.169	Grupo de Hilos 1-7	Initial Login Request	120	✓	5052	0	120	0
15	23:27:29.220	Grupo de Hilos 1-12	Initial Login Request	88	✓	5050	0	88	0
16	23:27:29.177	Grupo de Hilos 1-5	Auth	135	✓	26981	0	135	0
17	23:27:29.210	Grupo de Hilos 1-11	Initial Login Request	106	✓	5042	0	106	0
18	23:27:29.232	Grupo de Hilos 1-13	Initial Login Request	102	✓	5050	0	102	0
19	23:27:29.214	Grupo de Hilos 1-8	Auth	128	✓	26981	0	128	0
20	23:27:29.250	Grupo de Hilos 1-15	Initial Login Request	94	✓	5045	0	94	0
21	23:27:29.260	Grupo de Hilos 1-16	Initial Login Request	87	✓	5048	0	87	0
22	23:27:29.240	Grupo de Hilos 1-14	Initial Login Request	117	✓	5050	0	117	0
23	23:27:29.270	Grupo de Hilos 1-17	Initial Login Request	112	✓	5050	0	112	0
24	23:27:29.281	Grupo de Hilos 1-18	Initial Login Request	116	✓	5048	0	116	0
25	23:27:29.273	Grupo de Hilos 1-10	Auth	135	✓	26981	0	135	0
26	23:27:29.282	Grupo de Hilos 1-19	Initial Login Request	126	✓	5052	0	126	0
27	23:27:29.279	Grupo de Hilos 1-8	Auth	158	✓	26989	0	157	0
28	23:27:29.292	Grupo de Hilos 1-7	Auth	158	✓	26989	0	157	0
29	23:27:29.311	Grupo de Hilos 1-21	Initial Login Request	149	✓	5048	0	149	0
30	23:27:29.301	Grupo de Hilos 1-20	Initial Login Request	169	✓	5048	0	169	0
31	23:27:29.311	Grupo de Hilos 1-12	Auth	184	✓	26981	0	184	0

Figura 49. Estadísticas por cada solicitud.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Estos son los parámetros establecidos con los cuales el sistema Web deberá trabajar a futuro. La respuesta es positiva con un error de 0.00% en las peticiones realizadas al servidor, el promedio de tiempo de respuesta para un login request es de 338 (ms), mientras que para la autenticación es de 478(ms) lo cual corrobora que las respuestas a las peticiones realizadas están en un tiempo correcto para el aplicativo Web. A pesar del incremento en comparación con la anterior prueba los tiempos no varían en gran medida lo cual permitirá a los usuarios tener una mejor usabilidad del sistema sin tiempos de espera largos en peticiones que realicen.



Prueba 3

Se establecen 300 usuarios concurrentes que ingresarán a la página principal del sistema e ingresarán sus credenciales para acceder a su tablón personal.

Esta prueba es solo para fines informativos de como reaccionaria el aplicativo Web.

Tabla 72. Estadísticas JMeter 300 Usuarios

Descripción	Ejemp.	Prom(ms)	Min(ms)	Max(ms)	Error %	Pedidos/seg.
Login Request	300	688	27	1390	0.00	121.6/seg
Autenticación	300	1181	71	1624	0.00	75.4/seg

Nota: Esta tabla contiene información acerca de datos de pruebas con 300 usuarios.

Se puede apreciar que la aplicación soporta una cantidad alta de usuarios, ésta prueba presenta que las respuestas del sistema son correctas y sin caídas.

Cantidad alta de usuarios

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	23:59:51.140	Grupo de Hilos 1-3	Initial Login Request	27	✓	5048	0	27	0
2	23:59:51.143	Grupo de Hilos 1-2	Initial Login Request	34	✓	5050	0	34	0
3	23:59:51.150	Grupo de Hilos 1-4	Initial Login Request	27	✓	5048	0	27	0
4	23:59:51.140	Grupo de Hilos 1-1	Initial Login Request	37	✓	5052	0	37	0
5	23:59:51.154	Grupo de Hilos 1-5	Initial Login Request	31	✓	5048	0	31	0
6	23:59:51.161	Grupo de Hilos 1-7	Initial Login Request	36	✓	5052	0	36	0
7	23:59:51.167	Grupo de Hilos 1-9	Initial Login Request	33	✓	5050	0	33	0
8	23:59:51.164	Grupo de Hilos 1-6	Initial Login Request	36	✓	5048	0	36	0
9	23:59:51.157	Grupo de Hilos 1-6	Initial Login Request	43	✓	5048	0	43	0
10	23:59:51.171	Grupo de Hilos 1-10	Initial Login Request	40	✓	5050	0	40	0
11	23:59:51.176	Grupo de Hilos 1-11	Initial Login Request	49	✓	5050	0	49	0
12	23:59:51.177	Grupo de Hilos 1-12	Initial Login Request	54	✓	5050	0	54	0
13	23:59:51.181	Grupo de Hilos 1-4	Auth	71	✓	26988	0	71	0
14	23:59:51.181	Grupo de Hilos 1-1	Auth	71	✓	26991	0	71	0
15	23:59:51.181	Grupo de Hilos 1-13	Initial Login Request	70	✓	5048	0	70	0
16	23:59:51.181	Grupo de Hilos 1-2	Auth	81	✓	26991	0	81	0
17	23:59:51.181	Grupo de Hilos 1-3	Auth	86	✓	26988	0	86	0
18	23:59:51.185	Grupo de Hilos 1-14	Initial Login Request	87	✓	5050	0	87	0
19	23:59:51.186	Grupo de Hilos 1-15	Initial Login Request	99	✓	5048	0	99	0
20	23:59:51.192	Grupo de Hilos 1-16	Initial Login Request	93	✓	5050	0	93	0
21	23:59:51.196	Grupo de Hilos 1-17	Initial Login Request	92	✓	5048	0	92	0
22	23:59:51.196	Grupo de Hilos 1-18	Initial Login Request	94	✓	5050	0	94	0
23	23:59:51.187	Grupo de Hilos 1-5	Auth	108	✓	26981	0	108	0
24	23:59:51.200	Grupo de Hilos 1-19	Initial Login Request	110	✓	5048	0	110	0
25	23:59:51.199	Grupo de Hilos 1-7	Auth	118	✓	26981	0	117	0
26	23:59:51.202	Grupo de Hilos 1-8	Auth	122	✓	26981	0	122	0
27	23:59:51.202	Grupo de Hilos 1-9	Auth	127	✓	26991	0	127	0
28	23:59:51.202	Grupo de Hilos 1-6	Auth	127	✓	26981	0	127	0
29	23:59:51.206	Grupo de Hilos 1-20	Initial Login Request	127	✓	5052	0	127	0
30	23:59:51.209	Grupo de Hilos 1-21	Initial Login Request	127	✓	5050	0	127	0

Figura 51. Estadísticas por cada solicitud.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

A continuación, se observa el tiempo de respuesta con la prueba de estrés, el sistema soporta la concurrencia de usuarios.

Diagrama

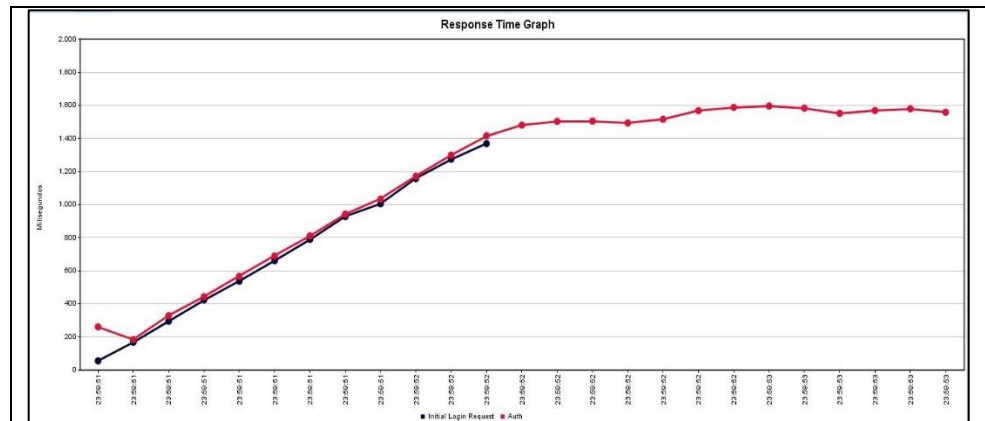


Figura 52. Tiempo de Respuesta.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

3.7.3. Pruebas contra requerimientos

Los usuarios finales verificaron la funcionalidad, características y el flujo de los procesos del sistema.

3.7.3.1. Requerimientos funcionales

Gestor

Tabla 73. Prueba contra requerimiento gestor

No.	Prueba	Ref RF	Acciones	Realizado	Observaciones
1	Generar código de resolución automáticamente	3.1.8.2	El código de aprobación se lo ingresa desde teclado al sistema	NO	El código lo genera un programa aparte, del cual solo una persona copia el valor a la aplicación
2	Estado del flujo	3.1.8.2	Ingresar el número de cédula del alumno del cual se desea saber su estado	SI	
3	Envío de notificaciones a estudiantes vía email	3.1.8.3	El envío es automático	SI	
4	Campo de observaciones	3.1.8.3	Ingresar las observaciones en el campo establecido	SI	
5	Validación de documentos	3.1.8.3	Accionar el botón de descarga para visualizar	SI	Solo funciona con archivos de

	subidos por el alumno		cada documento		extensión PDF
6	Validación de información mediante checkbox	3.1.8.3	Marcar o desmarcar campos a validar	SI	

Nota: Esta tabla contiene información acerca de pruebas a requerimientos pedidos.

Estudiante

Tabla 74. Prueba contra requerimiento estudiante

No.	Prueba	Ref RF	Acciones	Realizado	Observaciones
1	Ingreso a la plataforma de pasantías y prácticas pre profesionales por el estudiante	3.1.8.2	Ingresa la cédula. Ingresa la contraseña. Accionar el botón Login.	SI	
2	Inicio del proceso de pasantías y prácticas pre profesionales por el estudiante	3.1.8.2	Accionar el botón continuar.	SI	Las rutas son automáticas de acorde al proceso dentro del flujo.
3	Generar oficio para empresa	3.1.8.2	Accionar el botón descargar oficio.	SI	Las rutas son automáticas de acorde al proceso dentro del flujo.
4	Generar carta de aceptación	3.1.8.2	Accionar el botón descargar carta	SI	Las rutas son automáticas de acorde al proceso dentro del flujo.
5	Generar la carta compromiso	3.1.8.2	Accionar el botón descargar carta	SI	Las rutas son automáticas de acorde al proceso dentro del flujo.
6	Solicitud para inicio de actividades en la empresa	3.1.8.2	Accionar el botón descargar	SI	Es un formato sin validez legal.
7	Guía o sección de ayuda	3.1.8.2	Accionar el botón saber más	NO	Solo se muestra la ayuda de los colores que representan a cada proceso
8	Estado del flujo	3.1.8.2	Ingresar a la página principal	SI	
9	Subir documentos al Sistema	3.1.8.2	Accionar el botón descargar PDF	SI	La descarga es dinámica, va de acorde al proceso en el que se

					encuentra
--	--	--	--	--	-----------

Nota: Esta tabla contiene información acerca de pruebas a requerimientos pedidos.

Administrador

Tabla 75. Prueba contra requerimiento administrador

No.	Prueba	Ref RF	Acciones	Realizado	Observaciones
1	Aplicativo JAVA para carga de datos a la base de datos	3.1.8.1	Ingresar al aplicativo. Accionar el botón cargar csv. Buscar el archivo. Seleccionar el tipo de ingreso. Accionar el botón ejecutar.	SI	

Nota: Esta tabla contiene información acerca de pruebas a requerimientos pedidos.

Capítulo 4

4. Implementación

Este capítulo presenta una serie de configuraciones que describen como se debe correr la aplicación en el equipo establecido. Adicional se presenta los resultados en cuanto a su correcto funcionamiento.

4.1. Configuración de la base de datos

La versión del Postgresql 9.4 de 64 bit, para su instalación se requiere Microsoft Visual C++ 2013 x64

Detalles importantes:

Tabla 76. Configuración de la base

Nombre	Descripción
Puerto	5432
Nombre de la base de datos	db_sp3
Vacuum	Realizar un mantenimiento completo a la base cuando esté cargada con los datos iniciales que vienen con el Excel
Configurar el archivo pg_hba.conf	Agregar bajo la línea conexiones locales de IPV4 lo siguiente: host all all 0.0.0.0/0 md5
Reiniciar	Ir a los servicios de Windows, que se encuentran en el panel de control en la sección de herramientas administrativas, dentro del listado buscar el servicio de “postgresql-x64-9.4 – PostgreSQL Server 9.4” y reiniciarlo.

Nota: Esta tabla contiene información acerca de la configuración con parámetros específicos.

4.2. Configurar el Firewall de Windows

A continuación, se detalla los pasos a seguir para permitir el acceso al puerto 5432.

Aplastar la tecla Windows + R.

Ingresa el comando WF.msc

Selección y dar click derecho sobre las reglas de entrada, para agregar una nueva regla.

Seleccionar el círculo que dice Puerto, avanzar a la siguiente ventana.

Observar que estén seleccionados los botones de TPC y Especificar un puerto local, en el campo agregar solo el número 5432, avanzar a la siguiente ventana.

Observar que el primer botón esté seccionado, para permitir la conexión, avanzar a la siguiente ventana.

Seleccionar los campos de acorde al tipo de conexión, para este caso se debe seleccionar dominio y privado.

Por último, agregar un nombre para la regla y una descripción.

4.3.Descripción del Hardware y Software

Esta sección cuenta con las características del equipo donde se alojará la aplicación Web.

4.3.1. Características del hardware

Tabla 77. Características del hardware

Nombre	Descripción
Marca	DELL
Modelo	Optiplex 9020
Disco duro	500 GB
Procesador	Intel Core i7 basado en x64
Monitor	14" (no táctil)
Tipo de conexión	Cableada
Velocidad en Mbps	Subida: desconocida. Bajada: desconocida.

Nota: Esta tabla contiene información acerca de las características necesarias del hardware.

4.3.2. Características del software

Tabla 78. Características del software

Nombre	Descripción
Sistema operativo	Windows
Versión	Windows 8.1 Pro
Bit	64

Nota: Esta tabla contiene información acerca de las características necesarias del software.

4.4. Configuración del servidor Oracle GlassFish Server

La versión del servidor es 4.1 para el sistema operativo Windows. Se requiere que el archivo War esté listo.

Previo a su instalación se requiere Java JDK 8u151 el cual se lo puede obtener en la página principal de Oracle.

Ingresa al bin y usando la consola de Windows ejecutar “asadmin start-domain”.

Ingresa al navegador con la url “http://localhost:4848/”

De acuerdo a la ubicación de los archivos de GlassFish ir al siguiente path “C:\glassfish4\glassfish\domains\domain1\lib\applibs”.

Hay que copiar y pegar las siguientes librerías en el directorio mencionado

“antlr-2.7.7.jar,BootsFaces-OSP-1.1.3-dist-default.jar,c3p0-

0.9.2.1.jar,dom4j-1.6.1.jar,ehcache-core-2.4.3.jar,iText-5.0.5.jar,javassist-

3.18.1-GA.jar,jboss-logging-3.1.3.GA.jar,jboss-transaction-api_1.2_spec-

1.0.0.Final.jar,mail_1.4.jar,mchange-commons-java-0.2.3.4.jar,postgresql-

9.4.1209.jar,primefaces-5.0.jar,slf4j-api-1.6.1.jar,slf4j-simple-1.6.1.jar”.

En la ventana de configuración del servidor ingresar en la pestaña de aplicaciones, proceder a implementar la aplicación.

Escoger la primera opción de búsqueda de archivo y asignar el War, adicional agregar los nombres de las librerías como se muestra en los pasos iniciales.

Correr la aplicación, ésta se abre en una url distinta a la que comúnmente se usa.

Abrir la aplicación con el siguiente enlace “http://<ip>:8080/SPPP_App”.

La aplicación está lista para usar.

4.5.Carga de datos a la base

De acorde al Excel que provee la Dirección de Carrera de Sistemas, con el registro de los alumnos habilitados para realizar pasantías se procede a realizar los siguientes pasos para subir dicha lista a la base de datos.

- 1) Crear un documento de Excel en blanco, en la fila número uno agregar el siguiente nombre a cada columna.

Tabla 79. Campos Excel

id_usuario	correo	clave	nombre	apellido	telefono	id_perfil	genero
------------	--------	-------	--------	----------	----------	-----------	--------

Nota: Esta tabla contiene información acerca de campos necesarios en el documento Excel.

- 2) Asegurarse que para el siguiente paso se filtre únicamente los alumnos que tienen el 60% de los créditos aprobados y están en 6to nivel, según las normas vigentes.

Con los datos (Excel) de los alumnos copiar al nuevo documento solo los datos de las siguientes columnas.

Cédula.

Email institucional.

Cédula (repetir la columna con las cédulas).

Nombres.

Apellidos.

Celular.

Perfil. Esta columna debe tener el número uno en cada campo para cada estudiante de la lista.

Genero. Identifica el género de cada alumno, las palabras que representan a cada uno son “Femenino” y “MASCULINO”.

A continuación, se presenta una imagen como ejemplo.

Lista de carga

	A	B	C	D	E	F	G	H
1	id_usuario	correo	clave	nombre	apellido	telefono	id_perfil	genero
2	1725074650	eachigp@est.ups.edu.ec	1725074650	EDISON STEVEN	ACHIG PASQUEL	(09) 68219590	1	MASCULINO
3	1721041364	caguasr@est.ups.edu.ec	1721041364	CARLOS AUGUSTO	AGUAS RAMIREZ	(09) 95411607	1	MASCULINO
4	1726635996	laguayza@est.ups.edu.ec	1726635996	LUIS DAVID	AGUAYZA SALDAÑA	(09) 84896961	1	MASCULINO
5	1205621541	jaguilarc3@est.ups.edu.ec	1205621541	JESSICA LIZZETE	AGUILAR CÉLLERI	995554606	1	Femenino
6	1725600850	daguilarn@est.ups.edu.ec	1725600850	DANIEL ALEJANDRO	AGUILAR NOBLECILLA	999973714	1	MASCULINO
7	1721194478	salbaracin@est.ups.edu.ec	1721194478	SANTIAGO JAVIER	ALBARACIN ROMAN	(09) 98678493	1	MASCULINO
8	1714632542	naldaz@est.ups.edu.ec	1714632542	NELSON ISMAEL	ALDAZ CALISPA	998919910	1	MASCULINO
9	1721393229	ealemanc@est.ups.edu.ec	1721393229	ELIZABETH ESTEFANIA	ALEMAN CADENA	958871759	1	Femenino
10	1722939368	kalmachet@est.ups.edu.ec	1722939368	KLEBER GIOVANNI	ALMACHE TOALOMBO	0	1	MASCULINO

Figura 53. Carga Excel.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

En caso de que un dato numérico como el número de celular de algún estudiante esté vacío, poner el número cero. Si un dato como el correo está vacío, poner la palabra vacío. Guardar los datos en el formato CSV delimitado por comas en el disco local C con el nombre `s3p_usuarios.csv`

Formato datos

CSV (Comma delimited) (*.csv)

Figura 54. Formato.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

- 3) Crear un documento de Excel en blanco, en la fila número uno agregar el siguiente nombre a cada columna.

Tabla 80. Campos Excel

cedula_est	horaspasantia	ultimonivel	porcentajemateriasaprobadas
------------	---------------	-------------	-----------------------------

Nota: Esta tabla contiene información acerca de campos necesarios en el documento Excel.

- 4) En el campo de cedula, copiar las cédulas de los estudiantes del anterior archivo de Excel. Las tres columnas adicionales tendrán el mismo valor para cada alumno como se indica en la siguiente imagen.

Formato datos

	A	B	C	D	E
1	cedula_est	horaspasantia	ultimonivel	porcentajemateriasaprobadas	
2	1725197428	0	6	60	
3	1725501017	0	6	60	
4	1724059066	0	6	60	
5	1718699554	0	6	60	
6	1717481467	0	6	60	
7	1723405518	0	6	60	
8	1726876178	0	6	60	
9	1717851917	0	6	60	
10	1723252654	0	6	60	
11	1752154284	0	6	60	
12	1725516122	0	6	60	
13	1718332115	0	6	60	
14	17200353130	0	6	60	

Figura 55. Campos.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Guardar los datos en el formato CSV delimitado por comas en el disco local C con el nombre `s3p_estudiantes.csv`

Formato documento

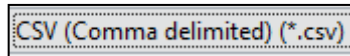
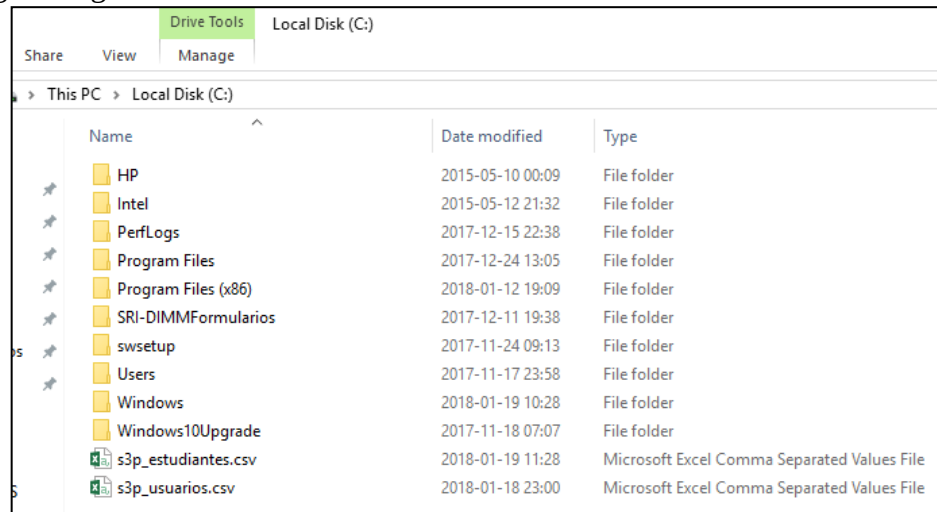


Figura 56. Formato.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

- 5) Los dos documentos de Excel deberán estar en el disco C similar a la siguiente imagen.

Lugar obligatorio



Fuente 57. Archivos.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

- 6) Ejecutar la aplicación de escritorio “S3P Cargar base de datos”.

Aplicativo JAVA

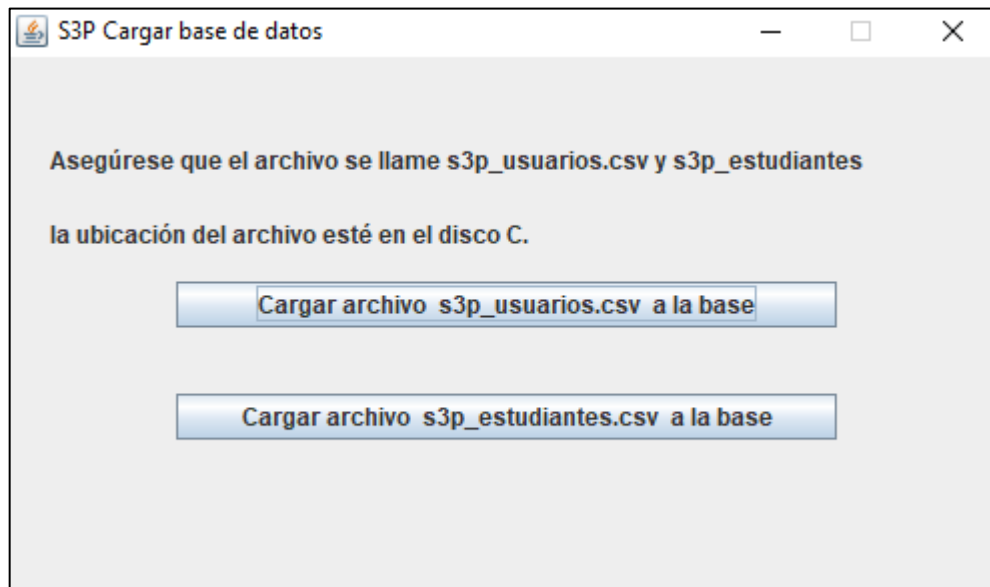


Figura 58. Programa S3P Cargar BD.

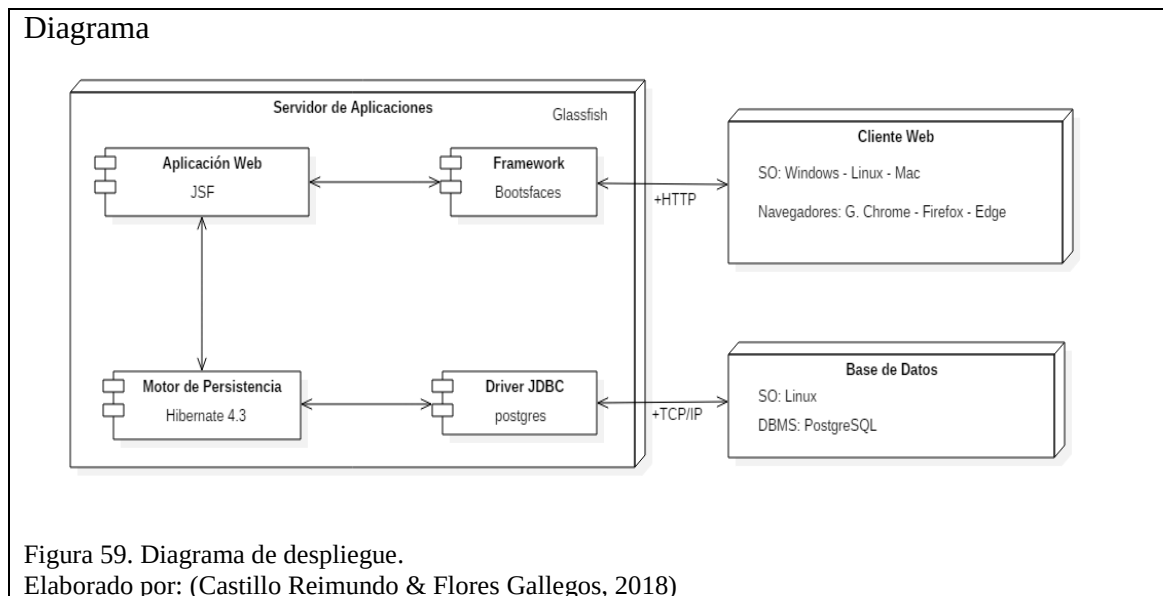
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

- 7) Accionar el primer botón, en caso de ser exitoso el almacenamiento un mensaje de confirmación aparecerá en la pantalla, el cual indicará que se puede continuar.

- 8) Accionar el segundo botón solo si apareció la confirmación cuando realizo la acción en el primer botón.
- 9) Ha finalizado el ingreso a la base, cerrar la aplicación de escritorio “S3P Cargar base de datos”.

4.6. Diagrama de despliegue

A continuación, se puede observar la arquitectura del sistema, se aprecia sus componentes de desarrollo y la estructura de la arquitectura.



4.7. Resultados

Con la etapa de desarrollo culminada, el sistema está listo para registrar pasantías de estudiante tomando en cuenta todo el flujo de procesos que el estudiante debe realizar, entre lo que se incluye el ingreso de datos en formularios, la subida y descarga de archivos, todos estos procesos están sujetos a validaciones por parte del gestor del sistema.

4.7.1. Acceso al sistema

Para poder ingresar al sistema, es necesario tener un navegador web en el cual se escribirá la dirección del sitio web y lo primero que se mostrará es la pantalla de ingreso.

Interfaz

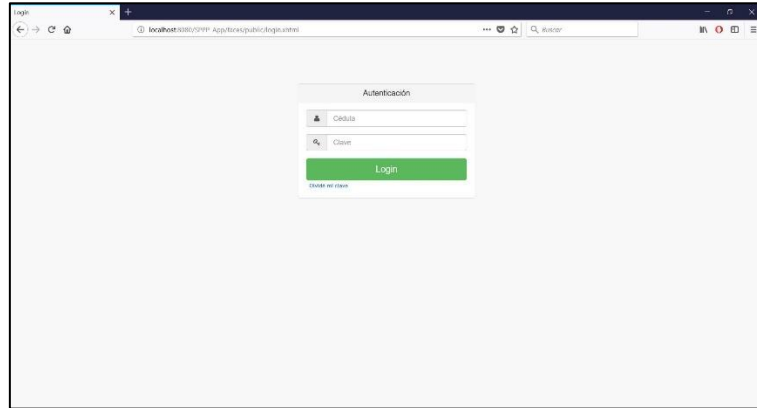


Figura 60. Pantalla de acceso.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

En esta pantalla se procede a colocar el usuario y contraseña, el sistema realiza una validación de los datos asignados y según el perfil del usuario, redireccionará al dashboard principal para empezar las tareas.

4.7.2. Módulo de registro de pasantía

Es la pantalla principal de todos los usuarios de perfil estudiante, contiene un tablón que muestra el proceso actual en el que están y dirige a otras pantallas para cumplirlo.

Se encuentra debajo de los tableros un mapa, donde el usuario puede ver en qué etapa del proceso se encuentra y qué estado tiene ese proceso.

Interfaz

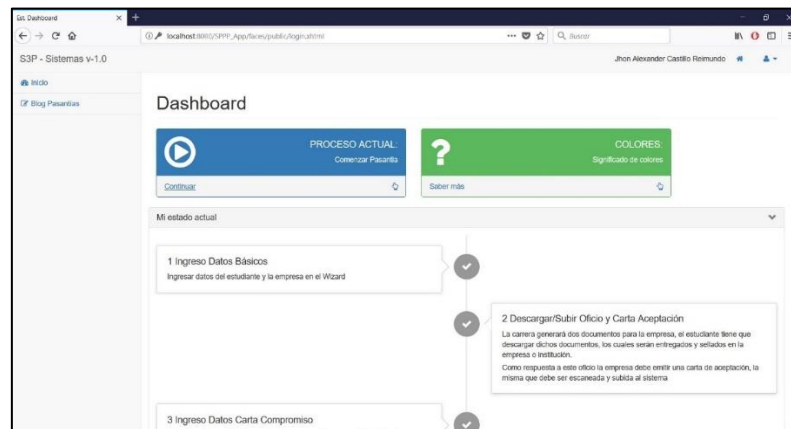


Figura 61. Dashboard estudiante.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Para el ingreso de datos, se existe un wizard para que el despliegue de los campos no sea tan extenso en una sola pantalla, para lo cual los campos se dividen en paneles a los cuales el usuario puede navegar dando click en los botones anterior o siguiente.

En este formulario los estudiantes ingresan datos que serán mandados a validar por un gestor, para comprobar que todo está correcto.

Interfaz

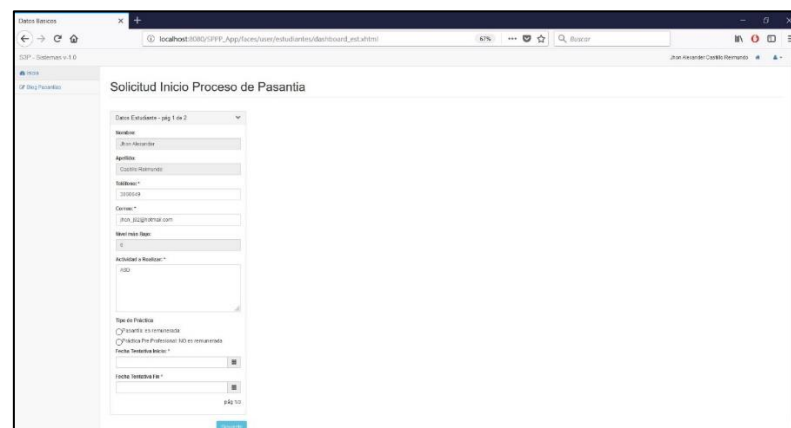


Figura 62. Formulario de ingreso de datos.

Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Esta pantalla se muestra cada vez que el usuario estudiante tenga un proceso en estado de validación, es un intermedio hasta que el gestor decida aprobar el proceso y continuar con el siguiente.



Si el gestor al momento de validar los datos decide marcar algún campo como incorrecto o dejar una observación, el estudiante deberá ingresar y corregir nuevamente estos datos.

Los errores son mostrados en rojo y la observación del gestor en la parte superior.



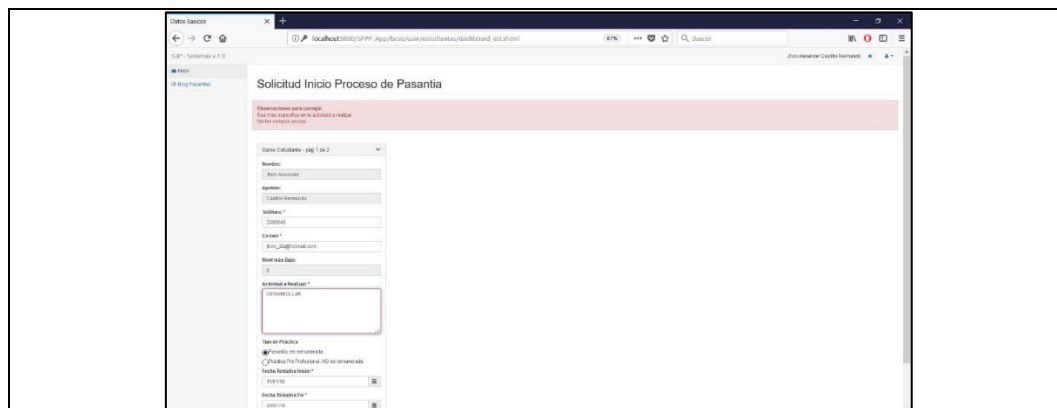


Figura 64. Errores de campos.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

El estudiante tiene la necesidad de descargar documentos para firmarlos y volverlos a subir a la plataforma, razón por la cual se implementó una pantalla que le sirva de ayuda para realizar este trámite.

Interfaz

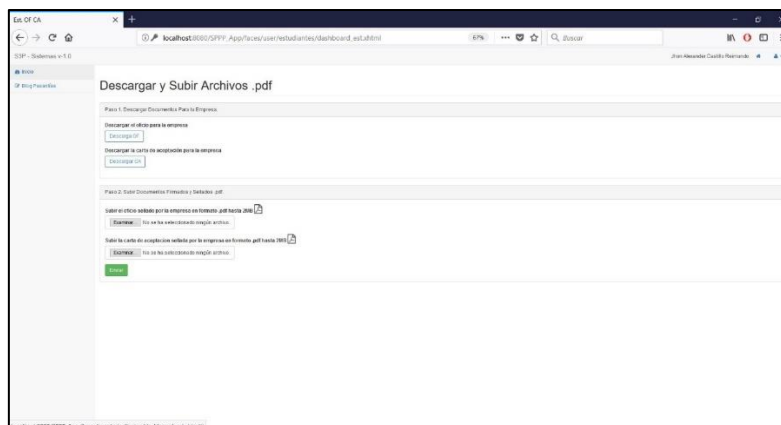


Figura 65. Descargar/Subir documentos.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Si el gestor al momento de revisar los pdfs subidos por el estudiante, decide que uno de ellos no se encuentra correcto o no cumple los requisitos solicitados, el estudiante deberá volver a subir ese documento, en esta pantalla se da la opción

al estudiante de que suba cualquier documento pdf que se encuentre negado por el gestor.

Interfaz

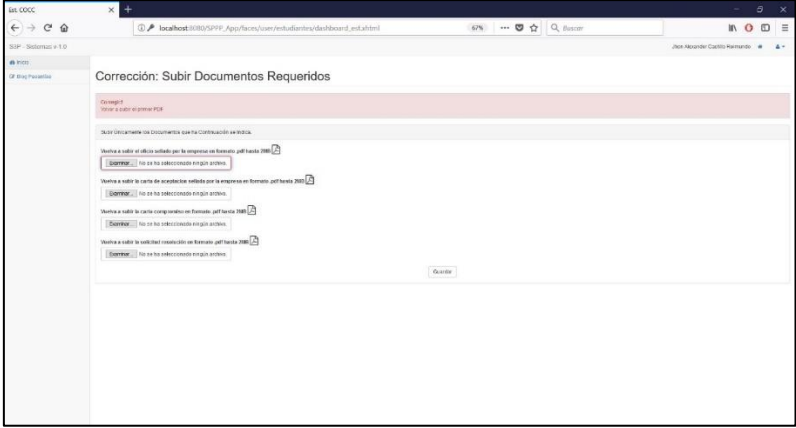


Figura 66. Subir documentos corregidos.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Es la pantalla principal para los usuarios de perfil gestor, contiene una tabla con los últimos 10 procesos enviados a validar, tableros con el número actual de procesos de un tipo en específico para ser validados, así como un menú interactivo para procesos más adelantados.

Interfaz

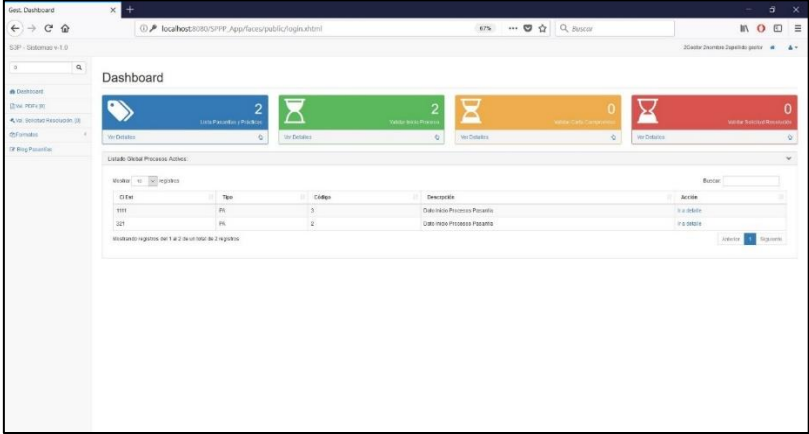


Figura 67. Dashboard gestor.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Para validar la información de un estudiante, en cualquiera de las tablas de procesos a validar dar click en ir a detalle y se desplegara un formulario con los datos previamente ingresados.

En los formularios se ha colocado vistos a lado de cada campo para que el gestor pueda decidir si la información ingresada concuerda con el campo pedido, al final del formulario se le agrega un campo observación al gestor para que pueda escribir sugerencias o notas al estudiante.

Interfaz

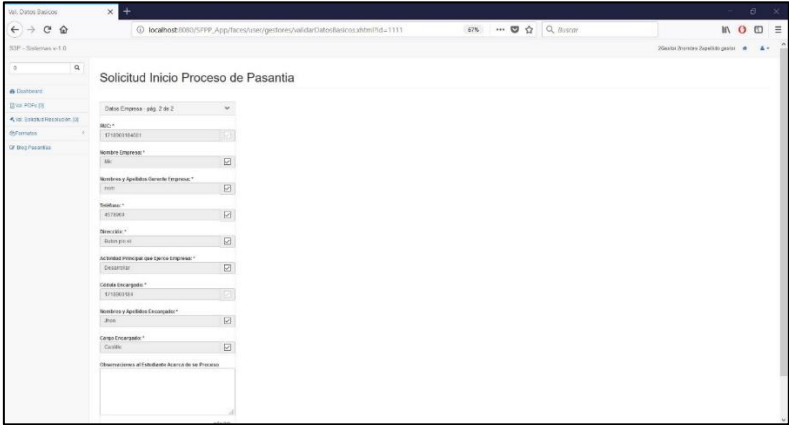


Figura 68. Validar formularios.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Existen diversos procesos que el gestor debe validar, para lo cual se ha implementado una pantalla que indica una tabla solo con datos de un proceso específico que el gestor escoja, esto sirve para cuando el gestor decida validar solo un tipo específico de proceso.

Interfaz

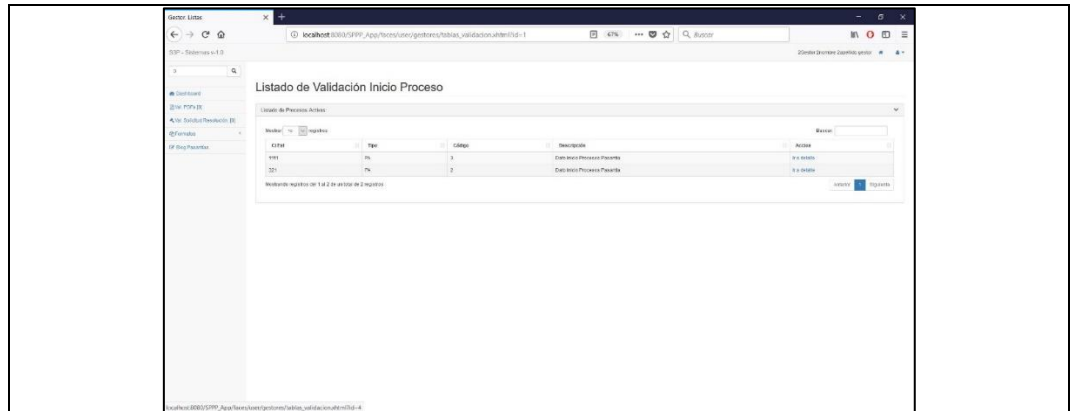


Figura 69. Tablas dinámicas.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

Para que el gestor pueda comprobar si los documentos subidos por el estudiante son correctos, se agrega esta pantalla en la cual el gestor puede observar todos los documentos subidos por un estudiante en específico y marcar con un visto aquellos que si cumplen las normas establecidas o escribir una observación en caso de que algún documento no se encuentre debidamente adaptado o subido.

Interfaz

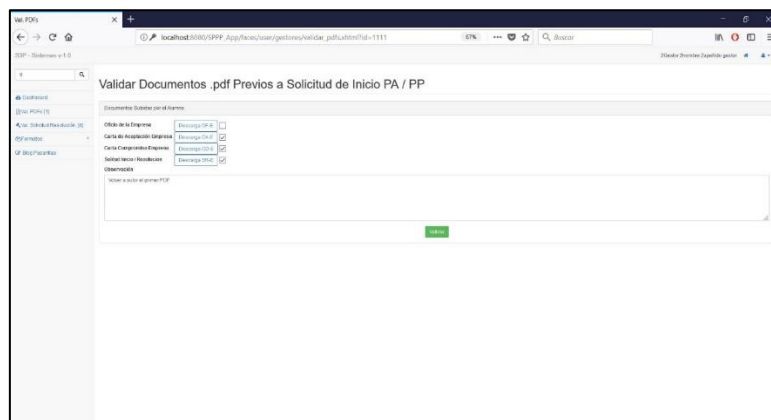
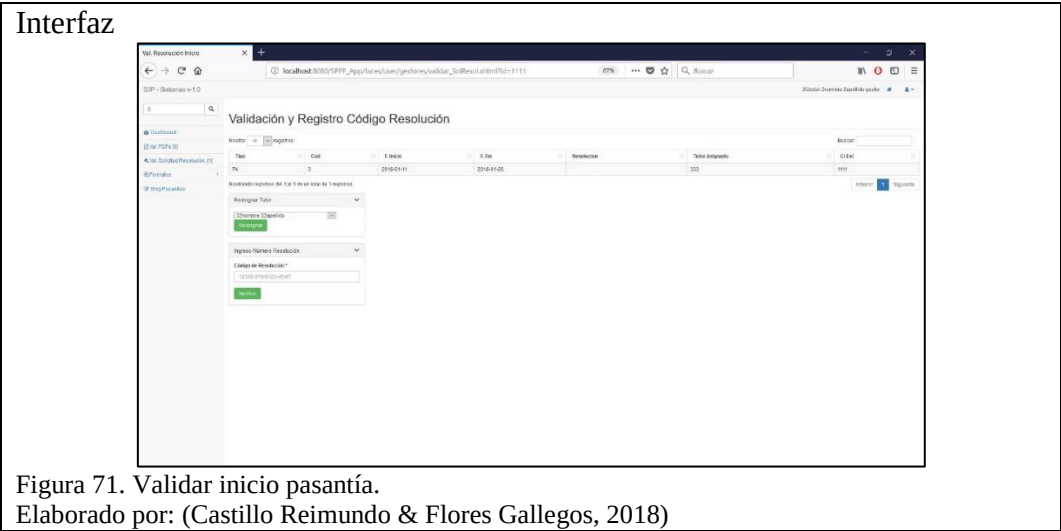


Figura 70. Validar documentos.
Elaborado por: (Castillo Reimundo & Flores Gallegos, 2018)

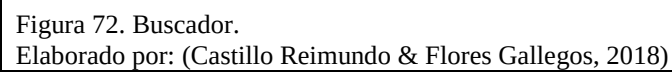
En esta pantalla se muestra una tabla en la parte superior con los datos de la pasantía y el estudiante, se implementa la opción de reasignación de un tutor para su pasantía si el gestor lo considera necesario y para finalizar con el proceso del estudiante el gestor debe colocar el código de resolución para ese inicio de pasantía.



Para que el gestor tenga acceso a la data de un estudiante en específico, se ha colocado un buscador en el dashboard del gestor, el cual mediante un número de cédula busca toda información relacionada con ese estudiante.

Se desplegará al lado izquierdo un panel con la información y datos propios del estudiante, así como de su proceso de pasantía, mientras que en el lado izquierdo se generará una tabla con todos los procesos atravesados por el estudiante hasta llegar al proceso actual.

Interfaz



CONCLUSIONES

- El lenguaje HQL es mejor en comparación al SQL común, ya que permite obtener el mismo resultado en un Objeto y a su vez dirigirse al atributo que se desea obtener si necesidad de especificar los campos deseados a obtener.
- El lenguaje JSF permite fácilmente establecer un patrón de diseño MVC, para separar las capas de modelo vista y controlador, logrando de esta manera que el mantenimiento del sistema sea mucho más fácil de hacer al igual que sus posteriores correctivos ya que el código se encuentra organizado en cada una de estas capas.
- El proceso para registrar una solicitud de pasantía debe mantener un orden concreto y tiempos establecidos, dado que al atravesar diferentes etapas las validaciones realizadas en previas etapas resultan tener una importancia fundamental y el realizarlos en desorden o con datos incorrectos puede generar inconsistencias en etapas futuras.
- Las entrevistas con usuarios de todos los perfiles, permitió tener con claridad cuál es la función de ellos dentro del proceso, lo cual facilita al momento de desarrollar establecer límites y tareas asignadas a cada uno de ellos, evitando acciones inadecuadas por parte de los mismos o procesos fuera de orden.
- Generar un sistema web, basados en un flujo de trabajo permite al código sea reusado para ciertas tareas, debido a que, si el flujo vuelve a un proceso anterior, la tarea vuelve a tomar parte del código de nuevo, lo cual se evita generando métodos parametrizados.

- La metodología XP utilizada en el desarrollo del sistema facilitó la construcción al realizar las diversas historias de usuarios ya que estas fueron concisas y bien detalladas, permitiendo al momento de iterar sobre ellas capturar errores y depurar de mejor manera los requisitos establecidos por el cliente.
- Definir las entidades dentro del sistema evita que el código sea desordenado y los métodos sean repetitivos, por lo cual es de gran importancia que entidades que comparten los mismos atributos hereden de una entidad padre estos atributos, evitando la repetición innecesaria de los mismos.
- La captura de requerimientos del cliente debe ser precisa y entendible, así como el proceso del flujo de trabajo, debido a que, si un paso del flujo es malinterpretado, el sistema web puede ejecutar las tareas en el orden erróneo o producir resultados distintos a los esperados por el cliente.
- La mensajería instantánea ayuda a que un sistema interactúe por sí solo con el usuario, mejora la comunicación en tiempo, facilita describir un mensaje debido a que se lo hace de forma global de acuerdo a un proceso.

RECOMENDACIONES

- Agregar un módulo de administración, el cual permita la gestión de cambios en menos tiempo.
- Diseñar un método que permita calcular el tiempo de espera máxima para un alumno dentro de cada proceso del Workflow.
- Integrar un sistema que permita la notificación SMS a los alumnos, para que se le haga saber lo antes posible de cualquier cambio.
- Realizar mejoras a la estructura de la base, para que se pueda obtener reportes de cada periodo.
- Agregar un módulo de eliminación de pasantía o práctica pre profesional.
- Integrar un nuevo wizard con todos los componentes como generación de PDF, para acoplar el proceso de extensión universitaria.

LISTA DE REFERENCIAS

- Achig, K. V., & Torres, B. A. (2018). *ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DEL MÓDULO DE CONTROL DE UN WORKFLOW PARA EL SEGUIMIENTO DE PASANTÍAS DE LA CARRERA DE INGENIERÍA DE SISTEMAS – SEDE QUITO*.
- Adamini, M. (Abril de 2011). Revista derecho y ciencias sociales. *ProQuest ebrary*.
- Becerra Mejoa, A. V., & Sosa Naranjo, P. F. (11 de Julio de 2011). *Repositorio Digital EPN*. Obtenido de <http://bibdigital.epn.edu.ec/handle/15000/3959>
- Beck, K. &. (2001). *Planning extreme programming*. Addison-Wesley Professional.
- Campoverde Rengel, A., & Torres Basantes, V. J. (2016). *Repositorio ESPE*. Obtenido de <http://repositorio.espe.edu.ec/handle/21000/12114>
- Castillo Reimundo, J. A., & Flores Gallegos, J. D. (2018). *ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DEL MÓDULO DE REGISTRO DE UN WORKFLOW PARA EL SEGUIMIENTO DE PASANTÍAS DE LA CARRERA DE INGENIERÍA DE SISTEMAS - SEDE QUITO*.
- CEMLAD. (2013). REGLAMENTO DE VINCULACIÓN CON LA COMUNIDAD . *REGLAMENTO DE VINCULACIÓN CON LA COMUNIDAD - CEMRAD*, 1-2.
- Chasipanta Llulluna, L. R., & Ramos Villacis, D. F. (2015). *Repositorio ESPE*. Obtenido de <http://repositorio.espe.edu.ec/handle/21000/10241>
- Chris Wanstrath, C. &-F. (2007). *Github*. Obtenido de Github: <https://github.com/about>
- Coalition, W. M. (1996). *Workflow Management Coalition*. Obtenido de Workflow Management Coalition: <http://www.wfmc.org/>
- D'Urzo, D. (2013). *BootsFaces*. Obtenido de BootsFaces: <https://www.bootsfaces.net/index.jsf>
- Fowler, M. &. (1999). *UML gota a gota*. Pearson Educación.
- Hibernate. (2004). *Hibernate*. Obtenido de Hibernate Community Documentation: <https://docs.jboss.org/hibernate/orm/3.3/reference/en/html/queryhql.html>
- IA, D. C. (26 de Junio de 2014). *Universidad de Alicante*. Obtenido de Titulo de experto universitario en desarrollo de aplicaciones y servicion con java EE: <http://www.jtech.ua.es/j2ee/publico/jsf-2012-13/sesion01-apuntes.html>
- Itext. (14 de Feabrero de 2000). *About iText Software*. Obtenido de About iText Software: <https://itextpdf.com/about>
- MAHESWARAN/AP, D. G. (10 de 4 de 2017). *SlidePLayer*. Obtenido de SlidePLayer: <http://slideplayer.com/slide/11894410/>

- Martin, Fowler, & Kendall. (1997). *UML Distilled*. Adison Wesley Longman Inc.,
- Mena Arce, J. D., & Páez Estrella, J. G. (Agosto de 2011). *Repositorio ESPE*. Obtenido de <http://repositorio.espe.edu.ec/handle/21000/4273>
- Miller, D. (14 de Enero de 2014). *New Theme - SB Admin 2*. Obtenido de Start Bootstrap: <https://startbootstrap.com/news/2014/01/14/new-theme-sb-admin-2/>
- Oracle. (Abril de 2005). *Introduction to Javaserer Faces - What is JSF?* Obtenido de Oracle Technology Network: <http://www.oracle.com/technetwork/topics/index-090910.html>
- Oracle. (21 de Enero de 2011). *JavaMail 1.4.4*. Obtenido de javamail: <http://www.oracle.com/technetwork/java/javamail/javamail144-1562675.html>
- Oracle. (24 de Septiembre de 2015). *Gassfish*. Obtenido de Galssfish: <https://javaee.github.io/glassfish/download>
- Oracle. (17 de oCTUBRE de 2017). *JDK 8u151 Update Release Notes*. Obtenido de Java™ SE Development Kit 8, Update 151 (JDK 8u151): <http://www.oracle.com/technetwork/java/javase/8u151-relnotes-3850493.html>
- processmaker. (2000). *Software de administración de workflows de ProcessMaker*. Obtenido de processmaker: <https://www.processmaker.com/es/products/workflow>
- Project, P. (9 de Febrero de 2017). *Pencil Project*. Obtenido de Pencil Project: <https://pencil.evolus.vn/>
- Salesiana, U. P. (2012). *Universidad Politecnica Salesiana/Normativa*. Obtenido de Normativa: http://www.ups.edu.ec/web/guest/normativa/-/document_library_display/u8OILw1nqXw9/viewf/7861224
- Schwaber, k, & Beedle. (2002). *Agile software development with Scrum (Vol. 1)*. Upper Saddle River: Prentice Hall.
- Scrum. (2018). *Scrum Framework image*. Obtenido de Scrum Framework image: http://metodologiascrum.readthedocs.io/en/latest/_images/reuniones.jpg
- SOCIEDAD, U. P. (17 de Octubre de 2012). *Reglamento de Vinculación con la Sociedad*. Obtenido de La UPS/Normativa/Normativa Interna/Reglamentos/Reglamento de Vinculación con la Sociedad: https://www.ups.edu.ec/web/guest/normativa/-/document_library_display/u8OILw1nqXw9/viewf/27799
- StarUml. (2014). *StarUml*. Obtenido de StarUml: <http://staruml.io/>
- Stonebraker, M. (1986). *A Brief History of PostgreSQL*. Obtenido de PostgreSQL: <https://www.postgresql.org/docs/9.4/static/history.html>

Sutherland, K. S. (Julio 2013). La guía de Scrum. *La Guía Definitiva de Scrum: Las Reglas del Juego*, 4.

Team, T. p. (2002). *pgAdmin*. Obtenido de Introduction: <https://www.pgadmin.org/docs/pgadmin3/1.22/introduction.html>

Vásquez Barrionuevo, H. O., & Estrella Rubio, P. P. (Agosto de 2008). *Repositorio Digital EPN*. Obtenido de <http://bibdigital.epn.edu.ec/handle/15000/873>

Xia, C. Y. (2009). Efficient implement of orm (object/relational mapping) use in j2ee framework: Hibernate. *In Computational Intelligence and Software Engineering*, 1'3.