

Universidad Central de Venezuela

Facultad de Ciencias

Escuela de Computación



**Sistema de Gestión de Pasantías para la
Escuela de Computación**

Trabajo Especial de Grado

Presentado ante la ilustre

Universidad Central de Venezuela

Por los Bachilleres

Félix D. Montes P.

V-17.402.860

Juan C. Florville V.

V-17.960.697

Para optar por el título de

Licenciado en Computación

Tutores:

Prof. Antonio Silva

Profa. Ana Morales

Caracas, Mayo 2013

Acta

Quienes suscriben, miembros del Jurado designado por el Consejo de Escuela de Computación, para examinar el Trabajo Especial de Grado presentado por los Bachilleres Félix D. Montes Pérez C.I.: 17.402.860 y Juan Carlos Florville C.I.: 17.960.697, titulado: "Sistema de Gestión de Pasantías para la Escuela de Computación", a los fines de optar por el título de Licenciado en Computación, dejen constancia de lo siguiente:

Leído como fue, dicho trabajo por cada uno de los miembros del jurado, se fijó el día 20 de mayo de 2013 a las 02:00 pm en la Sala PBIII de la Escuela de Computación, para que sus autores lo defendieran en forma pública, mediante una presentación oral de su contenido, luego de lo cual respondieron las preguntas formuladas. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado, sin hacerse solidario con la opinión de los autores, decidió aprobarlo con una nota de _____ puntos.

En fé de lo cual se levanta la presente Acta, en Caracas a los 20 días del mes de Mayo del año dos mil trece.

Profa. Ana Morales (Tutora)

Prof. Antonio Silva (Tutor)

Profa. Zenaida Castillo (Jurado)

Prof. Carlos Acosta (Jurado)

Dedicatorias

Principalmente a dios por darme la salud y energía necesaria para poder lograr lo que hoy soy con tanto esfuerzo y dedicación. Para ti abuelita rosa.

A mis padres, de quienes tuve la dicha de tenerlos durante estos largos años de carrera, enseñándome, apoyándome y guiándome por el buen camino, por nunca desmayar y estar ahí siempre conmigo en todo momento, demostrarme cariño, valor y humildad para las cosas, enseñarme que las cosas se consiguen con perseverancia, nunca bajar la cabeza y dar frente a problemas que no tienen solución. Por todo su amor, ¡Gracias! ¡Los quiero mucho!

A mi hermana por demostrarme que el cariño entre hermanos es muy distinto a cualquiera y supera cualquier cosa. ¡Te Amo sis!

A ese amor tan especial que es Janeiry García, por todos tus consejos y ayudas que me demostraste antes y durante el desarrollo de este trabajo, por tu dedicación y tolerancia en momentos de decaimiento, por estar ahí a mi lado siempre y compartir tu futuro conmigo. Fuiste y serás ese ser que uno siempre quiere encontrar en la vida y llevar de la mano a pasear por el mundo. Te Amo ¡PRIN!

A mis compañeros y amigos por todos esos momentos de tristeza y gloria que vivimos juntos durante la carrera, por aquellos que aun están y los que no, por estar siempre presente en las buenas y malas y enseñarme el trabajo en equipo. ¡Los quiero!

A mi amigo y compañero de este trabajo, Juan Carlos Florville y a su familia que gracias a ellos se pudo lograr, al fin, la realización de un proyecto como trabajo especial de grado, sin ti mi pana esto no fuera posible, ¡gracias por tu sincera amistad!

A mis tutores, Antonio Silva y Ana Morales, por su dedicación y aporte fundamental para llevar a cabo la realización de este trabajo.

A Anibal Rojas y compañeros de trabajo, por todo su apoyo en estos momentos de culminación.

Con cariño... ¡GRACIAS!

Félix Montes

A Dios por darme la salud, voluntad, ánimo, ganas y en los momentos más difíciles la fuerza necesaria para poder completar una más de mis grandes metas. Junto a él sé que están mis abuelos y tíos, que me iluminaron, guiaron y cuidaron.

A mis padres los cuales sin ello no sería nada de lo que hoy soy. Pieza super fundamental para poder lograr esta meta. Su constancia, perseverancia, paciencia, cariño, amor y dedicación fueron algunas de las tantas cosas que hicieron por mí para poder lograrlo.

A mi hermana que fue guía y base en mis estudios, que me enseñó tantos valores, que me enseñó que en esta vida todo se puede, pero hay que lucharlo y dar lo mejor de uno para obtenerlo.

A ese ser tan único y especial que es Josimar Sanchez que durante todo este proceso de trabajo especial de grado, estuvo a mi lado dándome apoyo, fuerza, consejos y sobre todo mucho amor. Por permitirme estar a tu lado, por hacerme tan feliz y poder compartir este éxito contigo. !Te Amo!

A mi amigo y compañero de trabajo Félix Montes el cual fue participe en gran parte para que todo esto fuese posible. El único en verdad que puede entender por todo lo que pasamos para llegar hasta aquí, difícil pero no imposible. A su familia por tanta amabilidad, hospitalidad y cariño.

A mis amigos los cuales pusieron su grano de arena para que mis estudios fuesen inolvidables y únicos, para aprender y vivir experiencias que solo se dan rodeadas de gente como ustedes.

A nuestros tutores Antonio y Ana por brindarnos su apoyo, colaboración, paciencia y ayuda para guiarnos para poder lograr un trabajo óptimo y hacernos crecer profesionalmente.

A mis compañeros de trabajo, especialmente a Esteban Felipe, por su tiempo y ganas de apoyarme cada vez que necesitaba de una ayuda o de una idea “simpática” que programar.

Para todos ustedes... !GRACIAS!

Juan Carlos Florville.

Resumen

La Licenciatura de Computación de la Facultad de Ciencias de la Universidad Central de Venezuela ofrece a sus estudiantes la oportunidad de realizar Pasantías. Éstas están dirigidas por La Comisión de Pasantías de la Facultad, la cual requiere gestionar de mejor forma, la organización de sus procesos a través del uso de las Tecnologías de Información y Comunicación (TIC), con el fin de extender este beneficio a la comunidad estudiantil, docente y personal que la conforma.

Con el propósito de apoyar a la Comisión de Pasantías se plantea como objetivo de este Trabajo Especial de Grado el desarrollo de una aplicación web que automatice los procesos y actividades relacionados a la inscripción de pasantes, formalización, evaluación; como iniciativa que impulse el mejoramiento del sistema dentro de la Escuela de Computación.

La aplicación web se realizó, mediante la ayuda del *framework Ruby on Rails*, un sistema manejador de bases de datos *PostgreSQL* y el estudio y aplicación del método de desarrollo *AgilUs*; permitiendo así la creación de un sistema innovador, que optimice el proceso de interacción con el usuario; logrando así interfaces de usuario usables.

El resultado de las pruebas de aceptación determinó que el Sistema de Gestión de Pasantías cumple con las expectativas de la comunidad de la Escuela de Computación a la que está dirigido.

Palabras Claves: Sistema de Gestión, Pasantías, Aplicación Web, AgilUs.

Índice

Acta	2
Dedicatorias	3
Resumen	5
Índice.....	6
Índice de Figuras	9
Índice de Tablas	11
Introducción.....	12
CAPÍTULO I. Tema de la Investigación	14
1.1 Planteamiento del Problema.....	14
1.2 Objetivo General	16
1.3 Objetivos Específicos.....	16
1.4 Alcance	16
1.5 Metodología de Desarrollo.....	17
CAPÍTULO II. Marco Conceptual.....	18
2.1 Área de Estudio	18
2.1.1 Pasantías.....	18
2.1.2 Sistema de Pasantías	18
2.2 Área de Investigación	22
2.2.1 Aplicación Web.....	22
2.2.2 Arquitectura Cliente / Servidor	23
2.2.3 Servidor Web.....	24
2.2.4 Cliente Web	25
2.2.5 Tecnologías del lado del Cliente.....	25
2.2.6 Hoja de Estilo en Cascada (CSS).....	25
2.2.7 Lenguaje de Marcado de Hipertexto (HTML).....	26
2.2.8 AJAX.....	26
2.2.9 JavaScript.....	27
2.2.10 JQuery	29
2.2.11 Tecnologías del lado del Servidor	29
2.2.12 Ruby	29
2.2.13 Ruby On Rails (RoR).....	30

2.2.14	Active_admin	32
2.2.15	Modelo Vista Controlador (MVC).....	33
2.2.16	Base de Datos.....	34
2.2.17	Sistema Manejador de Base de Datos	35
2.2.18	PostgreSQL	35
2.2.19	Manejo de Versiones	36
CAPÍTULO III. Marco Metodológico		38
3.1	AgilUs.....	38
3.1.1	Principios:	38
3.1.2	Mejores prácticas:	39
3.1.3	Ciclo de vida.....	40
CAPÍTULO IV. Marco Aplicativo.....		43
4.1	Aplicación del método AgilUs.....	43
4.2	Etapas de requisitos.....	43
4.2.1	Tormenta de Ideas.....	43
4.2.2	Perfil de Usuarios	44
	• <i>Usuario Administrador</i>	44
	• <i>Usuario Pasante</i>	45
	• <i>Usuario Tutor</i>	46
4.2.3	Lista de Requerimientos Funcionales y No Funcionales	46
4.3	Análisis.....	47
4.3.1	Modelos de casos de uso	47
4.3.2	Modelos Objetos del Dominio.....	54
4.3.3	Guía de Estilo.....	56
4.3.4	Prototipo en Papel.....	58
4.3.5	Patrones de Interacción	61
4.3.6	Prototipaje.....	67
4.3.7	Entrega	81
	• <i>Prueba de Aceptación</i>	81
	• <i>Escenario de Uso</i>	82
Conclusiones		101

Trabajos Futuros	103
Bibliografía	104

Índice de Figuras

Figura 1 Diagrama de flujos de información (Postulación). Autores: Florville – Montes -2013 ..	19
Figura 2 Diagrama de flujos de información (Inscripción). Autores: Florville – Montes -2013....	20
Figura 3 Diagrama de flujos de procesos (Formalización). Autores: Florville – Montes -2013....	20
Figura 4 Diagrama de procesos de información (Informe FInal). Autores: Florville – Montes - 2013	21
Figura 5. Arquitectura Cliente – Servidor. Autor: Tanenbaum - 2002.....	24
Figura 6. JavaScript. Autor: Álvarez - 2009.	29
Figura 7. Modelo de Vista Controlador. Autor: Rafael - 2008.	34
Figura 8. El método AgilUs: etapas, actividades y artefactos. Autor: Acosta - 2010.....	41
Figura 9. Modelo de Casos de Uso nivel cero. Autores: Florville – Montes -2013.....	48
Figura 10. Modelo de Casos de Uso nivel uno. Autores: Florville – Montes -2013.....	49
Figura 11. Modelo de casos de uso nivel dos administrador. Autores: Florville – Montes -2013.	50
Figura 12. Modelo de casos de uso pasante. Autores: Florville – Montes -2013.....	52
Figura 13. Modelo de casos de uso Tutor. Autores: Florville – Montes -2013.....	53
Figura 14. Modelo de objeto del dominio del modelo de SIGEPAS. Autores: Florville – Montes - 2013.	55
Figura 15. Logotipo de SIGEPAS. Autores: Florville – Montes -2013.....	57
Figura 16. Colores implementados. Autores: Florville – Montes -2013.	57
Figura 17. Menú para el Sistema. Autores: Florville – Montes -2013.	58
Figura 18. Menú Modulo Administrador. Autores: Florville – Montes -2013.....	58
Figura 19. Diseño del Modelo de Datos para el sistema de SIGEPAS. Autores: Florville – Montes - 2013.	68
Figura 20. Prototipo de Interfaz. Iteración Cero. Autores: Florville – Montes -2013.....	69
Figura 21 Representación Gráfica de los procesos y funcionalidades – Administrador. Autores: Florville – Montes -2013	73
Figura 22 Representación Gráfica de los procesos y funcionalidades - Tutor Académico. Autores: Florville – Montes -2013	74

Figura 23 Representación Gráfica de los procesos y funcionalidades – Pasante. Autores: Florville – Montes -2013.....	74
Figura 24. Resultados de las pruebas de Aceptación. Autores: Florville – Montes -2013.	82
Figura 25. Escenario de Uso Ingresar. Autores: Florville – Montes -2013.....	83
Figura 26. Ingresar como Pasante. Autores: Florville – Montes -2013.	84
Figura 27. Acciones del Pasante.	84
Figura 28. Formulario para Formalizar Pasantía. Autores: Florville – Montes -2013.....	85
Figura 29. Comprobante de Formalización de Pasantías. Autores: Florville – Montes -2013.	86
Figura 30. Subir Formalización a la página. Autores: Florville – Montes -2013.	87
Figura 31. Ingresar como Administrador. Autores: Florville – Montes -2013.....	88
Figura 32. Menú de la vista del Administrador. Autores: Florville – Montes -2013.....	88
Figura 33. Asignar Tutor por el Administrador. Autores: Florville – Montes -2013.	89
Figura 34. Elegir Tutor para pasante por el Administrador. Autores: Florville – Montes -2013..	89
Figura 35 Lista de Estudiantes Inscritos. Autores: Florville - Montes 2013.....	90
Figura 36. Requisitos para ingresar como Tutor. Autores: Florville – Montes -2013.....	91
Figura 37. Vista una vez ingresado el Tutor. Autores: Florville – Montes -2013.....	91
Figura 38. Evaluar Pasante por Tutor. Autores: Florville – Montes -2013.	92
Figura 39. Mensaje una vez Evaluado el Pasante. Autores: Florville – Montes -2013.	93
Figura 40 Reporte de Pasantías. Autores: Florville – Montes 2013.	94
Figura 41 Reporte de Pasantías en formato PDF. Autores: Florville - Montes 2013.....	95
Figura 42 Reporte de Asignaciones. Autores: Florville - Montes 2013	96
Figura 43 Reporte de Asignaciones en formato PDF. Autores: Florville - Montes 2013	96
Figura 44 Lista de Tutores Activos. Autores: Florville - Montes 2013	97
Figura 45 Constancia Individual por Tutor en formato PDF. Autores: Florville - Montes 2013 ...	98
Figura 46 Constancia General por Tutor en formato PDF. Autores: Florville - Montes 2013	99
Figura 47 Descarga en archivo .zip del lote de constancias individuales por tutor. Autores: Florville - Montes 2013	100

Índice de Tablas

Tabla 1. Prototipo en Papel	58
Tabla 2. Prototipo en papel (Continuación Tabla 1)	59
Tabla 3 Prototipo en papel (Continuación Tabla 2)	60
Tabla 4. Patrones de Iteración – Publicar Informe.	61
Tabla 5 Patrones de iteración- Publicar Informe (Continuación Tabla 4)	62
Tabla 6 . Patrones de iteración- Publicar Informe (Continuación Tabla 5).....	63
Tabla 7. Patrones de Iteración - Envío de Correos.	63
Tabla 8 Patrones de Iteración - Envío de Correos (Continuación Tabla 7)	64
Tabla 9. Patrones de Iteración - Evaluar Pasante.	64
Tabla 10 Patrones de Iteración - Evaluar Pasante (Continuación Tabla 9).....	65
Tabla 11. Patrones de Iteración - Asignación de Tutor.....	65
Tabla 12 Patrones de Iteración - Asignación de Tutor (Continuación Tabla 11)	66
Tabla 13 Patrones de Iteración –Asignación de Tutor (Continuación Tabla 12)	67
Tabla 14. Heurísticas propuestas por Jakob Nielsen.	75
Tabla 15 Heurísticas propuestas por Jakob Nielsen (Continuación Tabla 14).....	76
Tabla 16 Heurísticas propuestas por Jakob Nielsen (Continuación Tabla 15).....	77
Tabla 17. Evaluación heurística.....	77
Tabla 18 Evaluación Heurística (Continuación Tabla 17).....	78
Tabla 19 Evaluación Heurística (Continuación Tabla 18).....	79
Tabla 20 Evaluación Heurística (Continuación Tabla 19).....	80
Tabla 21. Encuesta para el Sistema de Gestión de Pasantías.....	81

Introducción

La Comisión de Pasantías (COMPAS) de la Escuela de Computación de la Facultad de Ciencias de la Universidad Central de Venezuela, es la encargada de planificar, supervisar, asesorar y controlar los programas de pasantías, así como es la responsable de mantener una participación activa de estudiantes y profesores en empresas, industrias e instituciones, de tal manera que los estudiantes desarrollen nuevos proyectos donde apliquen los conocimientos adquiridos en sus estudios y de esta manera se fortalezcan las relaciones entre la Universidad y el sector productivo, de investigación, de innovación, de desarrollo; que permita fomentar el espíritu emprendedor de la comunidad estudiantil.

La principal tarea de la COMPAS es asesorar y ayudar a estudiantes que se encuentren cursando sus pasantías en la escuela de computación, así como también a los profesores que forman parte de esta comunidad como tutores académicos. A su vez es la encargada de atender las solicitudes de formalización de pasantías, generar reportes y constancias y es la representante de la Facultad de Ciencias ante el consejo central de pasantías de la UCV.

Actualmente existe una comunicación deficiente entre el pasante y el tutor, además de que la COMPAS no posee un buen sistema que permita que los estudiantes estén informados acerca de las pasantías. A través de su página web, la COMPAS ofrece cartas de postulación y formalización de pasantías de manera obsoleta, el sistema se encuentra desarrollado bajo una versión de *Ruby on Rails 2.3.4* y muestra deficiencias en algunos de sus procesos, como asignar tutor, asignar nota, evaluar informe, aceptar formalización, reportes, entre otros. Además, el modelado de la base de datos no representa persistencia ni integridad de la data con respecto a la información del pasante y su relación con el semestre.

La interfaz de usuario no es usable para el pasante ni para el administrador, a su vez no existe un modulo tutor, que permita al tutor académico asignado llevar a cabo una gestión en línea adecuado y realizar tareas como: la corrección de informes y asignar nota. La División de Control de Estudios y los tutores en algunos casos exigen reportes, dichos reportes no pueden ser emitidos por el sistema que actualmente utiliza la COMPAS.

A través de esta investigación se persigue una solución optimizada del manejo del *Módulo Administrativo* y del *Módulo del Pasante*, además la incorporación de un nuevo *Módulo Tutor*, como una iniciativa que impulse el mejoramiento de Gestión de la COMPAS, utilizando para ello las últimas y más innovadoras tecnologías que existen en el campo de desarrollo web.

Este trabajo está estructurado en 4 capítulos, los cuales se generan producto de la revisión documental, que permite aportar elementos importantes que apoyan este Trabajo Especial de Grado (TEG), estos capítulos son:

- Capítulo 1. Tema de la Investigación: este capítulo está relacionado con la problemática existente en la Facultad de Ciencias referente a la Gestión de COMPAS. Igualmente se presenta la justificación de la solución al problema planteado, además del objetivo general, los objetivos específicos y el alcance del TEG.
- Capítulo 2. Marco Conceptual: en este capítulo se estudian las diversas áreas de investigación referente al problema actual por parte de la COMPAS de la Facultad de Ciencias de la UCV, para dar a conocer la información en base a sus funcionalidades y procesos para la comunidad estudiantil y profesorado. De igual manera se exponen las tecnologías utilizadas en el desarrollo del sistema, indicando el motivo por el cual fueron seleccionadas.
- Capítulo 3. Marco Metodológico: en éste se estudia el método ágil de desarrollo *AguilUs*, su ciclo de vida y la adaptación realizada para llevar a cabo el desarrollo de este TEG.
- Capítulo 4. Análisis e Interpretación de los Resultados: a partir de los resultados obtenidos de la implementación del método de desarrollo *AguilUs*.

CAPÍTULO I. Tema de la Investigación

1.1 Planteamiento del Problema

Con el transcurso del tiempo la tecnología avanza, las empresas e instituciones como la Facultad de Ciencias de la UCV se sienten con la necesidad y responsabilidad de adquirir tecnologías para el mejoramiento de sus sistemas y a la vez de sus procedimientos, con el fin de dar versatilidad a la divulgación de la información y fortalecer el sistema de comunicación de la UCV, de tal manera que la información pueda llegar de manera más rápida, eficiente y eficaz a toda la comunidad universitaria.

Las aplicaciones web ofrecen numerosas ventajas en cualquier ámbito de trabajo donde se apliquen, ya que por medio de estas se permiten divulgar, mantener y mejorar la estructura de información que se muestra en el sitio web.

En la actualidad las pasantías representan una gran metodología de enseñanza, ya que prepara y forma al estudiante para ejercer su profesión dentro del campo laboral, por ende, representa una opción de inserción al empleo. Así pues, la COMPAS de la Escuela de Computación tuvo la iniciativa de desarrollar una aplicación web llamada *Sistema de Gestión de Pasantías (SIGEPAS)* que permita planificar, supervisar, asesorar y controlar los programas dirigidos a los estudiantes y profesores de la Escuela de Computación.

El *Sistema de Gestión de Pasantías* actual consta de dos módulos: *Módulo Administrativo* y en el *Módulo Pasante*.

El *Módulo Administrativo* actual está basado en una versión de *Ruby on Rails 2.3.4*, la cual se encuentra fuera de actualización, presenta una interfaz obsoleta, poco eficiente en el desarrollo de sus funciones y no existe ningún tipo de relación con el *Módulo Pasante*. El *Módulo Pasante* sólo consta de dos acciones; generar cartas de postulación en un formato PDF y formalización de pasantías.

El *Sistema de Gestión de Pasantías* actual no contempla actividades que puedan estar relacionadas con las responsabilidades del Tutor Académico, tales como, evaluar informes finales de pasantías desde una aplicación web, si no, que el pasante tiene que dirigirse a la

Escuela de Computación hacer la entrega personalmente al tutor académico y esperar que sea evaluado y posteriormente firmado.

Por lo anteriormente descrito, surgen las siguientes interrogantes:

¿De qué manera beneficiaría la creación de un nuevo Sistema de Comisión de Pasantías?, ¿Cómo sería la integración de un nuevo modulo para los tutores? y ¿Cómo funcionaría la aplicación web con la integración de estos tres módulos?

A través del desarrollo de este TEG se pretendió desarrollar una aplicación web con la búsqueda de una mayor confiabilidad y calidad de los servicios ofrecidos por parte de la COMPAS a la comunidad de la Escuela de Computación de la UCV.

Además justifica la posibilidad de mejorar los módulos, *Módulo Pasante* y *Módulo Administrativo*, de manera tal que los mismos sean desarrollados bajo lenguajes de alto nivel como *Ruby* en su versión más actualizada, también con uso de librerías que ayuden a un desarrollo ágil de todas las funcionalidades que contemplan la gestión de pasantías, tales como: envío de correos con contraseña al tutor y pasante, envío de correos personalizados a cada pasante de manera más rápida sin la necesidad de un proveedor de correos como *Gmail*, *Hotmail*, entre otros, generar archivos en formato PDF para la creación de cartas de postulación y constancias de formalización, como también reportes por parte del administrador de cantidad de pasantes por semestres, cantidad de pasantes que retiraron, cantidad de pasantes aprobados, que tutor fue asignado a que pasante, cargar informes finales, entre otros.

De esta manera se genera robustez en la aplicación, lo que se traduce en mayor confiabilidad y en mayor calidad de servicio ofrecido por la Comisión de Pasantías a la comunidad de la Escuela de Computación, así como disminución en tiempos de respuesta en consultas de datos, manteniendo la integridad de los mismos y fortaleciendo el proceso de gestión de pasantías dentro de la Facultad de Ciencias hasta el arribo de nuevas tendencias tecnológicas dentro del desarrollo web.

1.2 Objetivo General

El objetivo general de este Trabajo Especial de Grado es Desarrollar un Sistema de Gestión de Pasantías profesionales para la Escuela de Computación de la Facultad de Ciencias que permita planificar, supervisar, asesorar, controlar y evaluar las actividades indispensables de cada pasante, de los tutores y en general de la comunidad estudiantil interesada.

1.3 Objetivos Específicos

A continuación se muestra una lista de los objetivos específicos de este Trabajo Especial de Grado:

1. Levantar requerimientos del sistema de gestión actual.
2. Desarrollar un Módulo Administrativo.
3. Desarrollar funcionalidades para la gestión de e-mails.
4. Desarrollar un Módulo Pasante.
5. Desarrollar un Módulo Tutor.
6. Desarrollar funcionalidades para la gestión de noticias, ofertas y documentos de pasantías.
7. Desarrollar módulos de reportes y constancias.
8. Realizar pruebas de aceptación por parte de la comisión de pasantías para conocer su opinión sobre la aplicación.

1.4 Alcance

El alcance de esta investigación abarca los siguientes puntos:

- Desarrollo de los módulos de administración, pasante y tutor.
- Establecimiento de plantillas de correos que permitan su reutilización dentro del sistema.
- Generación de reportes y constancias a formato PDF por parte del administrador y tutor.
- Evaluación del informe final del pasante, por parte del tutor académico a través del sistema.

1.5 Metodología de Desarrollo

Para este TEG se utilizó la implementación del método de desarrollo *AgilUs*, la cual consta de cuatro etapas de desarrollo: requisitos, análisis, prototipaje y entrega. En el Capítulo III. Marco Metodológico, se explican detalladamente cada una ellas.

CAPÍTULO II. Marco Conceptual

2.1 Área de Estudio

Esta investigación se enmarca dentro del área del proceso de Gestión de Pasantías de la Escuela de Computación de la Facultad de Ciencias de la UCV. Por lo que en el siguiente capítulo se darán a conocer algunos conceptos relevantes que se tomaron en cuenta para la realización y desarrollo del sistema de gestión, entre ellos tenemos:

2.1.1 Pasantías

Las pasantías son una forma de aprendizaje que comprende lapsos de permanencia y participación activa de estudiantes y profesores en las empresas, industrias e instituciones relacionadas a su carrera universitaria. Las pasantías le permiten a las empresas desarrollar proyectos donde los estudiantes apliquen los conocimientos adquiridos en los estudios y generar nuevos proyectos de investigación y desarrollo, lo que da a conocer la buena práctica que el estudiante ha adquirido durante su carrera y a su vez formar al estudiante a incursionar en un ámbito laboral de agrado para su bienestar y desarrollo profesional.

2.1.2 Sistema de Pasantías

El sistema de pasantías es una aplicación web que va dirigida a la comunidad estudiantil y a las empresas interesadas en el capital intelectual que ofrece la Escuela de Computación, a través de este sitio se pueden conocer las ofertas laborales del mercado para los estudiantes que se encuentran en búsqueda de pasantías.

Mediante esta aplicación web actual los estudiantes de la Escuela de Computación una vez inscrito y con los créditos requeridos para optar a pasantías, pueden tener acceso a diversas ofertas laborales, generar cartas de postulación dirigidas a las empresas o instituciones donde quieran desarrollar su pasantías, leer noticias de actualidad, descargar documentos de interés con respecto a las normativas de las mismas en la Escuela de Computación en general, tener la posibilidad de formalizar sus pasantías, generar una constancias de formalización en formato PDF, y así dirigirse a la comisión para luego ser firmada y sellada. Luego de formalizar las pasantías la COMPAS debe asignarles un tutor académico a cada pasante para que el tutor

correspondiente pueda evaluarlo a través de un informe final que el pasante debe entregar. Para así el pasante obtener una calificación definitiva en la materia.

En la Figura 1, se muestra a través de un diagrama de flujos de información los procesos que interactúan para la generación de cartas de postulación.

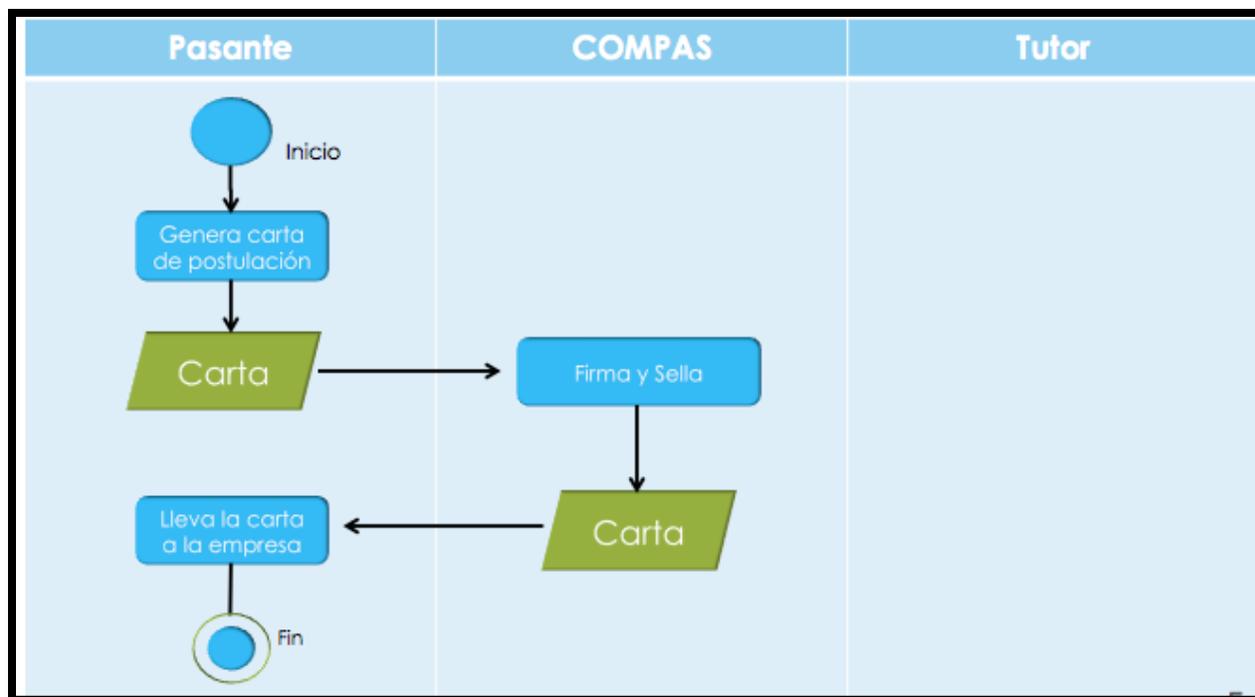


Figura 1 Diagrama de flujos de información (Postulación). Autores: Florville – Montes -2013

En la Figura 2 y en la Figura 3, se muestra a través de un diagrama de flujos de información los procesos que interactúan para los procesos de inscripción y formalización por parte del estudiante.

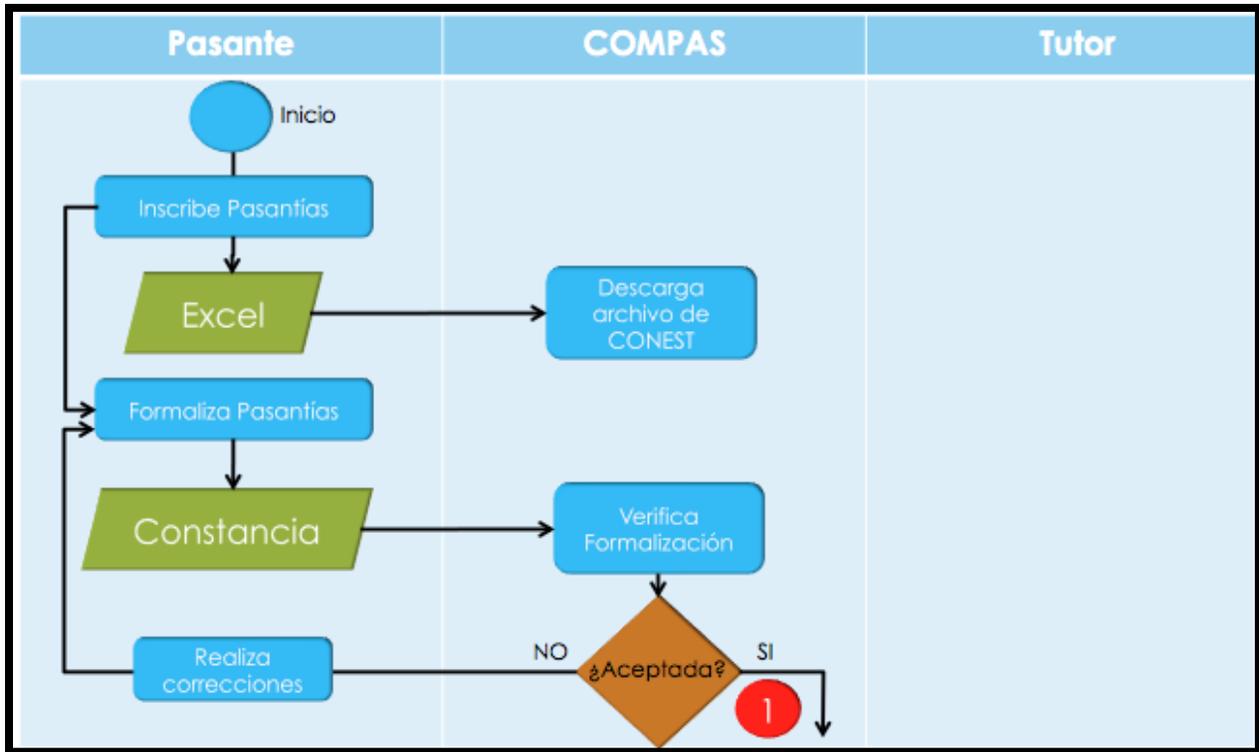


Figura 2 Diagrama de flujos de información (Inscripción). Autores: Florville – Montes -2013

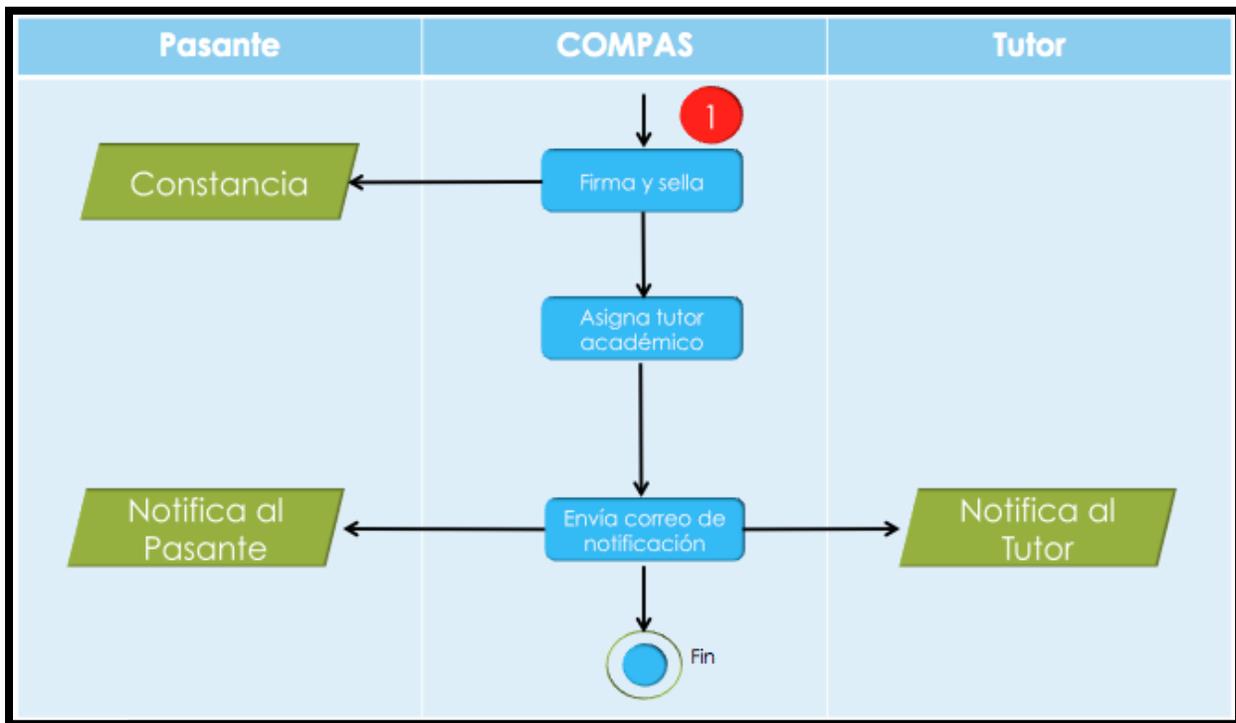


Figura 3 Diagrama de flujos de procesos (Formalización). Autores: Florville – Montes -2013

En la Figura 4, se muestra el flujo de procesos de información para la evaluación del pasante a través del tutor académico correspondiente y la entrega de la calificación definitiva a la División de Control de Estudios (CDE).

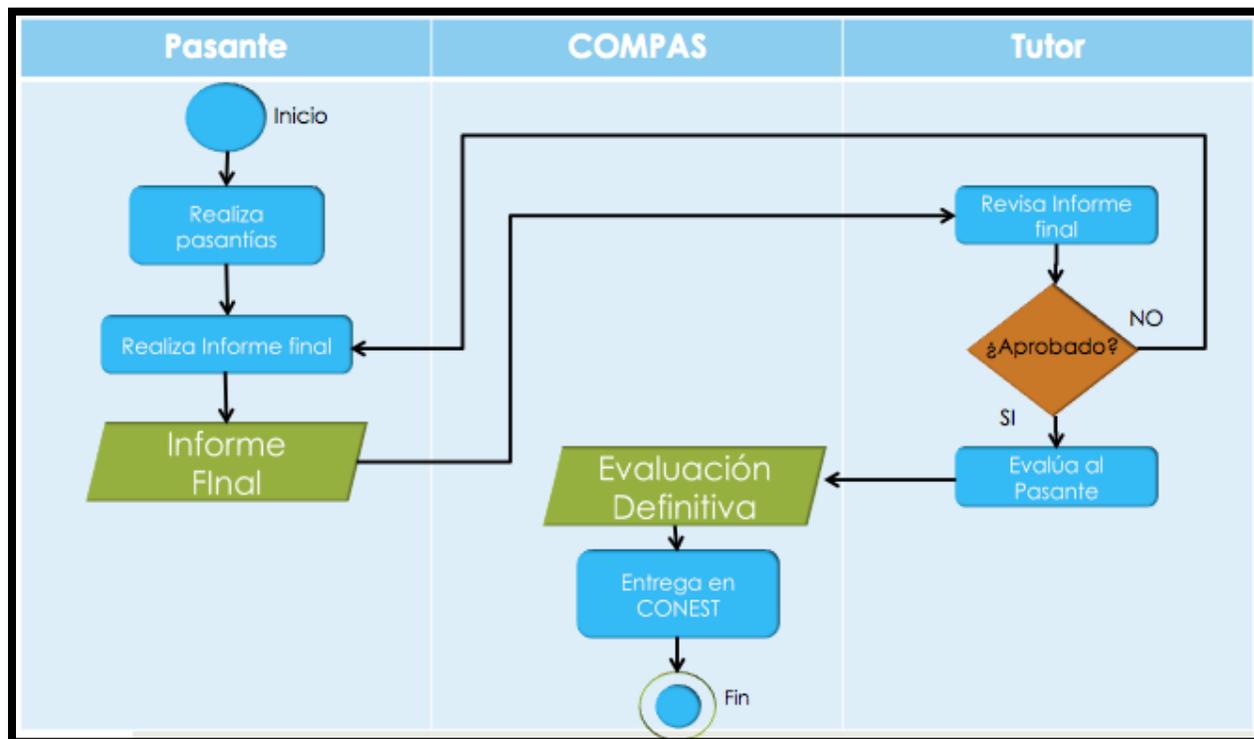


Figura 4 Diagrama de procesos de información (Informe Final). Autores: Florville – Montes -2013

La aplicación web actual que alberga la COMPAS está desarrollada bajo el lenguaje de programación *Ruby*, utilizando un *framework* de desarrollo sobre *Rails 2.3.4*.

El Sistema de Pasantías actual se divide en Módulo Administrativo y en el Módulo Pasante.

- **Módulo Administrativo**

Este módulo permite supervisar, asesorar y controlar los programas de pasantías a través de una aplicación web, otorgando ofertas, documentos y noticias referentes a las pasantías dentro de la Facultad de Ciencias de la UCV. Se convierte en un medio de comunicación de información para el estudiante que necesite asesoramiento.

- ***Modulo de Sistema al Pasante***

A través de este módulo el pasante tiene la posibilidad de generar cartas de postulaciones que necesite por cualquier requisito que le exija la empresa o institución en donde va a realizar la pasantía, así como también, le permite al pasante realizar la formalización de la pasantía indicando toda la información por parte de la empresa, tutor organizacional, tutor académico, tiempo dedicado realizando la pasantía, cantidad de horas laboradas, entre otras, para luego generar una constancia en formato PDF que indique que el estudiante realizó la pasantías, cumpliendo todos los requisitos exigidos por la normativa de la Facultad de Ciencias de la UCV.

Dadas estas definiciones se puede decir que hoy en día existen una gran variedad de tecnologías, las cuales podemos utilizar para crear diversas aplicaciones web que sirvan de ayuda para el sistema en desarrollo, las cuales para hacer el uso adecuado de dichas tecnologías, es necesario conocer algunos conceptos. A continuación se nombran y definen algunos de ellos.

2.2 Área de Investigación

Esta investigación se ubica dentro del área de Aplicaciones con Tecnologías Internet, por que se basa en un conjunto de páginas que interactúan entre sí. El objetivo del sistema es ilustrar, presentar y divulgar contenido de información, apoyándose en Bases de Datos asociadas, demostrando la integridad y veracidad de los datos. Con recursos en servidores web fiables, que permiten la administración de contenido y el procesamiento de la información proporcionada a los usuarios finales del sistema.

A continuación se describen los conceptos asociados al desarrollo de sistemas en tecnologías web.

2.2.1 Aplicación Web

Se denomina a una aplicación web, a un conjunto de aplicaciones o programas que los usuarios utilizan para poder acceder a un servidor web, a través de Internet o una Intranet mediante un navegador web, es decir, son todos esos mecanismos que permiten la búsqueda, acceso a la información o comunicación de un usuario con el servidor web. (Valencia, 2008)

La aplicación web ha llegado hacer uno de los más simples mecanismos de acceso a la red, por su simplicidad en cuanto a mantenimiento y actualizaciones de las cuales no requieren ser distribuidas e instaladas para los miles de usuarios conectados a la red. Son muchas las empresas, que en la actualidad, han llevado esta tecnología a sus clientes para obtener una comunicación activa entre el cliente y la información, permitiéndole así al usuario acceso a los datos de manera interactiva. Existen distintas aplicaciones web para diferentes funcionalidades y soluciones, que han llevado a grandes empresas a nivel mundial a posicionarse en los primeros lugares por su flujo de información y comunicación con los clientes a través de la web.

2.2.2 Arquitectura Cliente / Servidor

La Arquitectura cliente/servidor se basa en el procesamiento compartido de la información por medio de un conjunto de procesadores, en el cual múltiples clientes, distribuidos en cualquier parte solicitan requerimientos a uno o más servidores centrales por medio de una red amplia como internet o compartida como intranet. Existen 3 elementos fundamentales sobre los cuales se desarrollan e implementan los sistemas cliente/servidor (Tanenbaum, 2002): El proceso del clientes que es quien inicia el dialogo.

- El proceso servidor que es quien espera pasivamente las peticiones de servicio hechas por los clientes.
- El sistema intermedio que corresponde a la interfaz gráfica, que provee la conectividad entre el cliente y el servidor para poder intercambiar mensajes.

En la Figura 5 se muestra la Arquitectura Cliente-Servidor. En ella Se observan los componentes de arquitectura y el proceso de petición y respuestas.

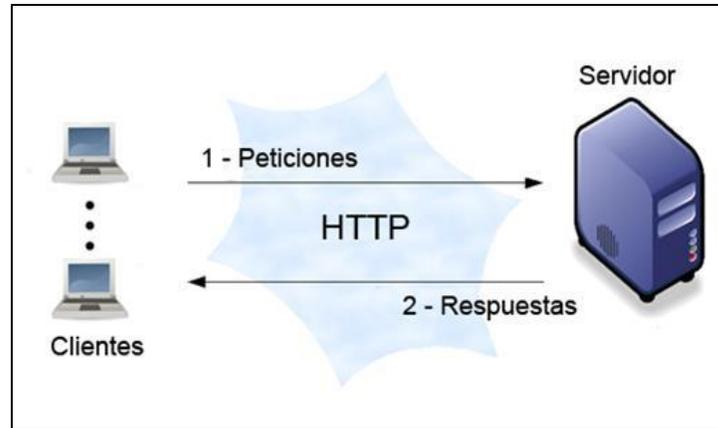


Figura 5. Arquitectura Cliente – Servidor. Autor: Tanenbaum - 2002.

La Arquitectura cliente/servidor proporciona un intercambio de información entre clientes y servidores obteniendo ciertas ventajas, destacándose:

- Centralización del Control: la información, los recursos y la integridad de los datos a solicitar, ubicados en el servidor, son controlados de forma que un cliente externo o defectuoso pueda dañar el sistema.
- Interfaz gráfica que resalte en cuanto a la manipulación de la data.
- Topología Distribuida: es caracterizada por lograr satisfacer a clientes ubicados en cualquier parte.

2.2.3 Servidor Web

Un servidor web es un Software o programa que sirve para atender y responder a las diferentes peticiones de los navegadores, proporcionando los recursos que soliciten usando el protocolo *HTTP* (*Hypertext Transfer Protocol* – Protocolo de Transferencias de peticiones y respuestas por parte de un servidor y un cliente) o el protocolo *HTTPS* (*Hypertext Transfer Protocol Secure* - la versión cifrada y autenticada de *HTTP*) (Piñol, 2003).

Un servidor web básico cuenta con un esquema de funcionamiento muy simple, basado en ejecutar infinitamente el siguiente bucle:

- Espera peticiones en el puerto *TCP* indicado (el estándar por defecto para *HTTP* es el 80).
- Recibe una petición

- Busca el recurso
- Envía el recurso utilizando la misma conexión por la que recibió petición
- Vuelve al segundo punto

Para la realización de este Trabajo Especial de Grado se utilizó la plataforma de servicio web gratuito *HEROKU*, el cual permitía alojar la aplicación web para que el personal de la COMPAS realizara las distintas pruebas de aceptación o detalles a través de internet que pudieran servir para obtener un producto de calidad.

2.2.4 Cliente Web

Un cliente web es conocido generalmente como el navegador que se utiliza en el equipo del usuario. También puede hacer referencia a los componentes y aplicaciones que permitan que mejoren el navegador para mantener optimizado las peticiones hechas por los usuarios. El cliente web puede implicar toda la máquina del usuario como también a un dispositivo portátil que proporciona acceso a la Web (PCMAG, 2003).

2.2.5 Tecnologías del lado del Cliente

Son tecnologías que proporcionan al cliente agilidad y velocidad en la respuesta del servidor, permitiendo a los lenguajes como *HTML* (Hypertext Markup Language), *JavaScript*, entre otros ser directamente interpretado por el navegador, dando así el apoyo para la visualización de la página web. A continuación se describen algunas tecnologías utilizadas para el desarrollo del sistema de Comisión de Pasantías.

2.2.6 Hoja de Estilo en Cascada (CSS)

Es una tecnología que permite reparar el contenido de la presentación de páginas web de una manera más exacta, pudiendo incorporar colores, imágenes formato de texto, fondos, entre otros a los documentos web escritos en *HTML* y *XML*, así también, es posible definir estilos en un archivo externo a las páginas y de querer cambiar alguno de ellos, automáticamente se actualizarían todas las páginas vinculadas del sitio web (Alvarez, 2008).

Actualmente se encuentra la versión *CSS 3.0* en la cual consiste en la incorporación de nuevos mecanismos para mantener un mayor control sobre el estilo con el que se muestran los

elementos de las páginas, sin tener que recurrir a trucos o parches, que a menudo complican el código de la web. Por ejemplo, en versiones anteriores era necesario usar imágenes de fondos o *JavaScript* para crear bordes, a diferencia de la versión 3 que solo es necesario usar una propiedad como *border-radius* para definir los bordes, con un color de fondo. Además existe la posibilidad de declarar varios fondos de imagen a un elemento de la página; lo que antes era solo posible creando varios elementos anidados y colocando un fondo en cada uno, entre otras mejoras que puede ofrecer *CSS 3.0* (Alvarez, 2008).

2.2.7 Lenguaje de Marcado de Hipertexto (HTML)

Es un lenguaje que indica al navegador donde colocar cada texto, cada imagen o cada video y la forma que tendrán estos al ser colocados en la página mediante etiquetas. Por ejemplo, `` significa que se escriba en negrita (bold) o `<p>` significa un párrafo, `<a>` es un enlace, etc. Casi todas las etiquetas tienen su correspondiente etiqueta de cierre, que indica que a partir de ese punto no debe de afectar la etiqueta.

HTML 5 pretende proporcionar una plataforma con la que desarrollar aplicaciones web más parecidas a las aplicaciones de escritorio, donde su ejecución dentro de un navegador no implique falta de recursos o facilidades para resolver las necesidades reales de los desarrolladores. Para ello, se están creando unas *APIs* que permitan trabajar con cualquiera de los elementos de la página y realizar acciones que hasta hoy era necesario realizar por medio de tecnologías accesorias. Por ejemplo, *API* para hacer *Drag&Drop* (Mediante eventos), *API* para trabajar Off-Line (Permite descargar todos los contenidos necesarios y trabajar en local), *API* de Geoposicionamiento para dispositivos que lo soporten, entre otros (Alvarez, 2008).

2.2.8 AJAX

AJAX (JavaScript Asíncrono y XML), no es una tecnología independiente, sino es la técnica que detalla el desarrollo web para aplicaciones interactivas. Estas aplicaciones se ejecutan del lado del cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano (Nerfites, 2009). Entre estas están:

- *HTML* o *XHTML* + *CSS* para crear una aplicación basada en estándares.
- *DOM* para la creación y manipulación dinámica de la aplicación.

-
- *XML* y *JSON* para el intercambio y la manipulación de la información que será mostrada en la aplicación.
 - Objeto *XMLHttpRequest* para el intercambio asíncrono de información.
 - *JavaScript* para combinar todas las tecnologías anteriores.

Esto hace posible la creación de aplicaciones muchos más rápidas y con mejor respuesta a las acciones del usuario, ya que cada acción ejecutada por el usuario genera una petición al servidor; una vez procesada esta, se crea una nueva página para el usuario. *Ajax* posibilita que el usuario nunca se encuentre frente a una ventana del navegador vacía esperando una respuesta tardía del servidor.

2.2.9 JavaScript

El programa en *JavaScript* tiene la particularidad de que está insertado dentro del mismo documento *HTML*, que lo presenta al usuario y no es por ello un programa aparte. Permite crear aplicaciones similares a las *CGI* (*Common Gateway Interface*). El *CGI* es un mecanismo que se ha utilizado en los servidores web para implementar las páginas web activas. El funcionamiento de los *CGI* es el siguiente: lee los datos provenientes de un formulario de una página web, procesa la información y lo escribe sobre el canal de salida estándar que es la pantalla del ordenador (Alvarez, 2009).

El programa en *JavaScript* reconoce eventos, que no son más que acciones de *JavaScript* (uno de los eventos es *onClick*, que se refiere, que al momento de pulsar con el cursor sobre la zona a la cual está asociada ese evento realice una determinada acción), creados por el usuario, definiendo así un sistema interactivo. Se puede por ello crear formularios que verifiquen la validez de la información e interpreten esta en el mismo programa contenido en el documento *HTML*, sin necesidad de comunicación por la red. También se permite por medio de un código *JavaScript* realizar acciones particulares como ejecutar un archivo de audio, ejecutar una venta, entre otros.

El lenguaje *HTML* permite dotar las páginas web de una atractiva información visual, no obstante le falta cierto grado de interactividad para el usuario. Para llenar el vacío existente entre la sencillez y la poca interactividad de *HTML*, y el grado de interactividad de *Java*, las

empresas de software se dispusieron a desarrollar órdenes fáciles de utilizar. De esta manera *Netscape Communications* introdujo el lenguaje *LiveScript*, poco después la empresa *Sun Microsystems* (creadora del lenguaje Java), se unió a *Netscape* para conseguir que el *LiveScript* fuese adoptado como el lenguaje estándar de internet para la escritura de ordenes acerca de la web. Puesto que el *LiveScript* tenía muchas semejanzas con *Java*, el lenguaje fue renombrado *JavaScript*.

JavaScript es un lenguaje compacto, y basado en objeto, diseñado para el desarrollo de aplicaciones cliente servidor a través de internet. En una aplicación cliente para un navegador, la sentencia *JavaScript* contenida en un documento *HTML* puede reconocer y responder a eventos generados por el usuario como *clicks* del mouse, información de formularios y navegación de documento a documento.

Por ejemplo se puede escribir una función *JavaScript* que verifique que la información introducida por el usuario sea correcta. Un documento *HTML* con *JavaScript* contenido es capaz de interpretar la información insertada por el usuario, verificar que sea correcta y alertar al usuario de que no lo sea. *JavaScript* es un lenguaje de programación dirigido a los programadores de páginas web.

JavaScript fue diseñado para ser un lenguaje de elaboración de *scripts* que pudieran incrustarse en archivos *HTML*. No es compilado si no que a través de ello es interpretado y leído por el navegador como código fuente.

Es necesario resaltar que hay dos tipos de *JavaScript*: por un lado está el que se ejecuta en el cliente, este es el *JavaScript* propiamente dicho, aunque técnicamente se denomina *Navigator JavaScript*. Pero también existe un *JavaScript* que se ejecuta en el servidor, y se denomina *SS JavaScript (Server - Side)*, que permite a los *scripts* ejecutarse del lado del servidor para ser mezclado directamente con el contenido web. Cada vez que un documento contiene código *JavaScript* a ser ejecutado del lado del servidor y es solicitado por el cliente, el servidor ejecuta el *script* o *scripts* del documento, y devuelve el documento resultante (este puede generarse parcialmente estático y parcialmente dinámico).

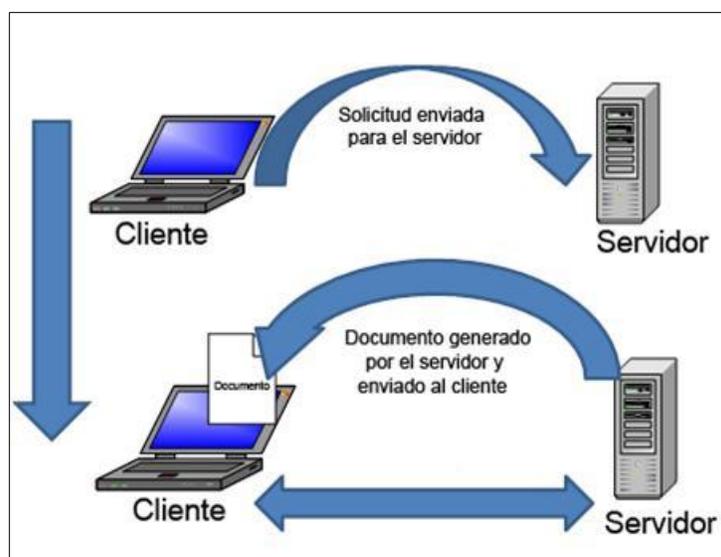


Figura 6. JavaScript. Autor: Álvarez - 2009.

2.2.10 JQuery

JQuery es una librería de *JavaScript* que permite simplificar la manera de poder interactuar con los documentos en *HTML*. Entre muchas de sus funciones, permite agregar interacción con la tecnología *AJAX* en páginas web y blogs, crear y desarrollar animaciones y manejar eventos, proporciona una serie de funcionalidades basadas en *JavaScript* que de otra manera requerirían de mucho más código como por ejemplo: menú desplegable utilizando listas, *drag and drop*, efectos de los métodos *SHOW* y *HIDE*, búsquedas dinámicas en la base de datos, entre otros.

Con las funciones propias de *JQUERY* se logra grandes resultados en menos tiempo y escribiendo menos (Alvarez, Desarrollo web, 2009).

2.2.11 Tecnologías del lado del Servidor

Son aquellas tecnologías que son reconocidas, ejecutadas e interpretadas por el propio servidor y que se envían de vuelta gracias a una petición hecha por el cliente en un formato comprensible para él. A continuación se describen algunas tecnologías.

2.2.12 Ruby

Ruby es un lenguaje de programación de alto nivel, reflexivo y orientado a objetos, donde su sintaxis viene inspirada en lenguajes como *Python* y *Perl*. *Ruby* es un lenguaje de programación

interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre (Mazt, 2003).

En *Ruby*, todo es un objeto. Se le puede asignar propiedades y acciones a toda información y código. La orientación a objetos es un paradigma de programación que facilita la creación de software de calidad por sus factores que potencian el mantenimiento, la extensión y la reutilización del software generado bajo este paradigma.

La programación orientada a objetos trata de amoldarse al modo de pensar del hombre y no al de la máquina. Esto es posible gracias a la forma racional con la que se manejan las abstracciones que representan las entidades del dominio del problema, y a propiedades como la jerarquía o el encapsulamiento.

En muchos lenguajes, los números y otros tipos primitivos no son objetos. *Ruby* tiene mucho en común del lenguaje *Smalltalk*¹ pudiendo poner métodos y variables de instancia a todos sus tipos de datos. Esto facilita el uso de *Ruby*, porque las reglas que se aplican a los objetos son aplicables a todo *Ruby*.

Ruby es considerado un lenguaje flexible, ya que permite a sus usuarios alterarlo libremente. Las partes esenciales de *Ruby* pueden ser quitadas o redefinidas a placer. Se puede agregar funcionalidad a partes ya existentes. *Ruby* intenta no restringir al desarrollador.

2.2.13 Ruby On Rails (RoR)

Rails es un *framework* para el desarrollo de aplicaciones web, software libre por naturaleza, está basado en el patrón de diseño Modelo Vista Controlador (*MVC*), presenta una gran ventaja cuando se desarrolla un software para distintos dispositivos, ya que los cambios requeridos son mínimos y sencillos. Trata de combinar la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros entornos de trabajo y con mínimas configuraciones.

Uno de los principales objetivos de *RoR* es la reutilización de código para no perder tiempo programando algo que ya está creado. Además, el *framework* también está pensado para que

¹ *Smalltalk* fue el primer sistema puro de objetos. Todo en *Smalltalk* es un objeto y toda la computación es desarrollada mediante mensajes que son enviados entre los objetos.

trabajar con bases de datos *SQL* sea de lo más sencillo. Sin embargo, es importante conocer las convenciones de programación que existen en *Rails*, por ejemplo: el nombre de las tablas de las bases de datos. *Rails* recomienda nombrar las tablas con el nombre de la clase que la representa en plural y en minúsculas (la clase 'Pasante' si tuviera una tabla se debería llamar 'pasantes'). Como este hay varios convenios más que a simple vista parecen inútiles pero luego permiten ahorrar una cantidad de tiempo impresionante cuando luego queramos hacer la documentación o editar código, etc.

RoR tiene algunas filosofías que es recomendable mencionar, como son:

- No lo repitas (del ingles Don't repeat yourself, DRY)
- Convención sobre configuración
- Uso de patrones de diseño
- Generación de código (HELPERS)
- Menos código, menos errores
- Tests integrados (Unitarios y Funcionales)

Para desarrollo y pruebas, se utiliza *Mongrel* o *WEBrick*, incluido con Ruby. Para utilizar *Rails* en servidores en producción se está extendiendo el uso de *Passenger*, una ventaja de *mod_rails* para *Apache* desarrollado en 2008 por la empresa holandesa Phusion. Otras opciones para producción son *Nginx*, *Mongrel*, *HEROKU*, *Apache*, *lighttpd* con *FastCGI* o alguna combinación de ambos (por ejemplo utilizando *Apache* como proxy para los procesos *Mongrel*). Sobre *Apache*, *mod_ruby* puede mejorar considerablemente el rendimiento, aunque su uso no se recomienda porque no es seguro utilizar múltiples aplicaciones *RoR* sobre *Apache* (Santos, 2006).

RoR trata de combinar la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros *frameworks* y con un mínimo de configuración. Se distribuye a través de *RubyGems*, que es el formato oficial de paquete y canal de distribución de librerías y aplicaciones *Ruby* (Tate, 2007).

Como parte de las librerías y aplicaciones que distribuye *RubyGems*, se encuentra *Active_admin*, la cual brinda un nuevo estilo de interfaz para el administrador, fácil manejo de

funcionalidades y manipulación de datos del sistema. Siendo esta una vista a parte de la página principal donde el administrador tiene todo los privilegios de la aplicación, pudiendo exportar a formato PDF constancias y reportes, así como también, la generación de nuevas claves de acceso para pasantes y tutores, y publicación de información como: noticas, ofertas y documentos.

Es importante destacar que para la implementación del envío de correos electrónicos *Rails* aporta un mecanismo propio de configuración de correo el cual puede ser modificado editando el archivo "*app/config/environments/production.rb*", aquí puede modificar el nombre del correo de la persona o institución de quien envía, como también el del servidor de correo.

2.2.14 Active_admin

Es una librería de *Ruby* que permite la generación de una interfaz anexa a la de la aplicación, teniendo como objetivo poder administrar todo el contenido de manera más dinámica y eficaz, además, brindando apoyo del manejo de la información. Tiene como objetivo hacer que las tareas por parte del administrador sean fáciles y seguras de implementar.

Dentro de las funcionalidades más resaltantes que ofrece *Active_admin* se tienen:

- Se establecen filtros de búsqueda por cada campo de cada tabla de la base de datos.
- Implementación de una función de encriptación *MD5* para claves de acceso al sistema.
- Permite la descarga en formatos CVS, XML Y JSON del contenido de una vista desplegada en tablas o bien sea de una tabla en particular de la base de datos.
- Es posible implementar botones de acciones que permitan la ejecución de una acción o función que facilite las funcionalidades del sistema.
- Fácil distribución del contenido y manejo de la información.
- Aporta una vista global, denominada "Inicio", donde permite la visualización de temas o datos relevantes que se necesiten mostrar al ingresar al sistema.

2.2.15 Modelo Vista Controlador (MVC)

El Modelo Vista Controlador es un patrón de diseño arquitectónico que divide de una aplicación web, las reglas, la interfaz del usuario y lógica de control en tres elementos: el modelo, la vista y el controlador (Rafael, 2008).

- **Modelo:** es el elemento que representa los datos y las reglas de negocio con la cual se van a trabajar en el sistema. En el modelo se encuentra toda la información requerida por el controlador hecha por un usuario a través de una vista. De existir un cambio en la vista para poder representar el modelo, no es necesario modificar esta capa.
- **Vista:** Es la representación del contenido del modelo a través de una interfaz gráfica. Le da el poder al usuario de manipular la información.
- **Controlador:** Es el gestor de peticiones hechas por los usuarios a través de la vista o interfaz grafica, invoca los procedimientos al modelo para poder mostrar datos, es el encargado de interpretar la información suministrada por un usuario.

Este patrón en la actualidad ha llevado a desarrollar aplicaciones web complejas de manera más sencilla por su patrón de tres componentes, lo cual los hace independientes uno de los otros, permitiendo así una programación más detallada de los datos, la interfaz y los métodos. En la Figura 7 se pueden observar la interacción entre los tres componentes. Al tener el proceso separado en estas partes, es posible, disponer de varias interfaces gráficas distintas que utilice el mismo modelo, gracias a esto se puede utilizar la aplicación web en distintas interfaces gráficas como; para la web, para dispositivos móviles, PDA, entre otros. Esto reduce significativamente el proceso de desarrollo, pruebas y actualizaciones de una aplicación compleja.

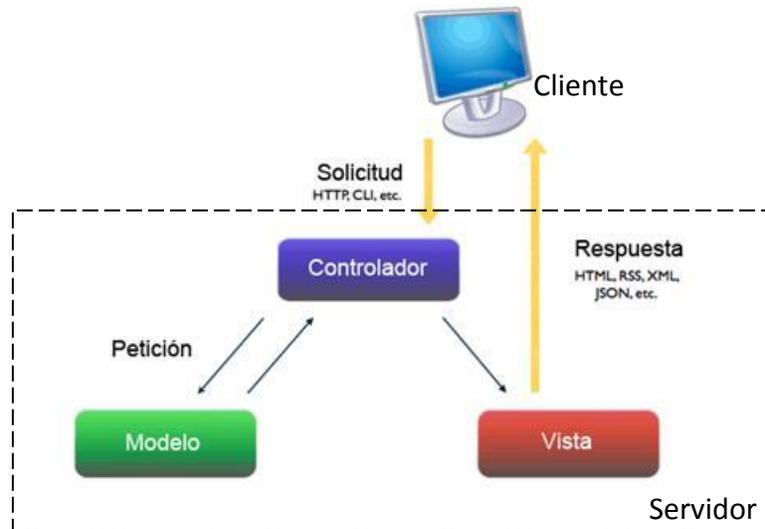


Figura 7. Modelo de Vista Controlador. Autor: Rafael - 2008.

2.2.16 Base de Datos

Una Base de Datos es un repositorio centralizado de datos relacionados lógicamente entre sí, que permiten a los usuarios o una empresa resguardar su información de manera que permitan almacenar y organizar hechos ó eventos y restituirlos a conveniencia (Valdés, 2007).

A su vez, es una colección de información organizada de forma que un software específico pueda seleccionar rápidamente los fragmentos de datos que necesite. Una base de datos es un sistema de archivos electrónico. Entre las principales características de los sistemas de base de datos podemos mencionar:

- Independencia lógica y física de los datos
- Redundancia mínima
- Acceso concurrente por parte de múltiples usuarios
- Integridad de los datos
- Consultas complejas optimizadas
- Seguridad de acceso y auditoría
- Respaldo y recuperación
- Acceso a través de lenguajes de programación estándar

2.2.17 Sistema Manejador de Base de Datos

Un Sistema Manejador de Base de Datos es un software que tiene como propósito general facilitar el proceso de definir, construir y manipular bases de datos que se utilizan para diferentes tipos de aplicaciones (Rodríguez, 2006).

Entre sus principales procesos destacamos:

- Definir la base de datos significa la declaración de:
 - Los tipos de datos
 - La estructura
 - Las restricciones de los datos a ser almacenados en la base de datos.
- Crear o construir la base de datos: es el proceso de almacenar los datos en algún medio de almacenamiento, esto es controlado por el Sistema Manejador de Base de Datos (SMBD).
- Manipular una base de datos incluye funciones como:
 - Consultar la base de datos para obtener algunos datos específicos.
 - Actualizar la base de datos para reflejar cambios.
 - Generar reportes de los datos.
 - Eliminar algunos datos.

2.2.18 PostgreSQL

Es una herramienta de manejo de bases de datos relacional que ofrece una potencia sustancial al incorporar los conceptos de clases, herencia, tipos y funciones, en una vía en la que los usuarios pueden extender fácilmente el sistema (*The PostgreSQL Global Development Group, 2005*).

De acuerdo a esto, el mismo es un SMBD objeto-relacional basado en *POSTGRES* versión 4.2 desarrollado en la Universidad de Berkeley por el Departamento de Ciencias de la Computación. *POSTGRES* fue el pionero de muchos conceptos que aparecieron en algunos SGBD comerciales mucho tiempo después.

PostgreSQL es código abierto descendiente del código fuente original de Berkeley, el mismo soporta una gran parte del estándar SQL 2003 y ofrece modernas características tales como:

- Consultas complejas
- Claves foráneas
- Disparadores
- Vistas
- Integridad transaccional
- Control de concurrencia multiversión

También, *PostgreSQL* puede ser extendido de varias maneras por los usuarios, por ejemplo añadiendo nuevos tipos de datos, funciones, operadores, funciones agregadas, métodos indexados.

2.2.19 Manejo de Versiones

Se llama manejo de versiones a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo. Una versión, revisión o edición de un producto, es el estado en el que se encuentra dicho producto en un momento dado de su desarrollo o modificación. Aunque un sistema de control de versiones puede realizarse de forma manual, es muy aconsejable disponer de herramientas que faciliten esta gestión dando lugar a los llamados sistemas de control de versiones o SVC (del inglés System Version Control) (Version Control with Subversion. O'Reilly, 2004).

Para este trabajo de investigación se utilizó el siguiente:

- ***GIT***

Git es un sistema de control de versiones diseñado para manejar proyectos muy grandes con velocidad y eficiencia, pero igual de apropiado para repositorios pequeños; es especialmente popular con la comunidad *Open Source*, sirviendo como plataforma de desarrollo para proyectos como el Kernel Linux, *Ruby on Rails*, WINE o X.org.

Git cae en la categoría de herramientas de manejo de código fuente distribuido, similar por ejemplo a *Mercurial* o *Bazaar*. Cada directorio de trabajo de *Git* es un repositorio completo con historial y capacidades totales de tracking de revisiones, independiente de acceso de red o un servidor central. Aún así, *Git* es extremadamente rápido y eficiente con el espacio.

Git es un proyecto *Open Source* cubierto por la *GNU General Public License v2*. Originalmente escrito por *Linus Torvalds* y mantenido por *Junio C Hamano* (<http://picandocodigo.net>, 2008).

CAPÍTULO III. Marco Metodológico

3.1 AgilUs

AgilUs es un método de desarrollo ágil, resultado de una de las líneas de investigación desarrolladas en el Centro de Ingeniería de Software y Sistemas (ISYS) de la Escuela de Computación, Universidad Central de Venezuela. Se basa en el concepto de usabilidad. Se fundamenta en el análisis centrado en el usuario y en la participación de especialistas, con el objetivo de evolucionar el software, a fin de que este alcance el mayor grado de usabilidad una vez culminado su desarrollo. *AgilUs* es un método de desarrollo iterativo e incremental que pone el mayor peso del desarrollo en la consecución de la usabilidad. Se centra en que la construcción y desarrollo de las interfaces de usuario no debe ser una adición estética que se da al final del desarrollo del sistema sino, muy por el contrario, el desarrollo de interfaces de usuario debe guiar las decisiones en Ingeniería de Software. En *AgilUs* son los usuarios, y no el cliente ni los programadores quienes guían el desarrollo del proyecto (Acosta, 2010).

El Método *AgilUs* busca proporcionar un conjunto de actividades organizadas para construir la usabilidad en el diseño de interfaces de usuario durante el desarrollo de un producto de software. El proceso de desarrollo de software engloba las actividades de requisitos, análisis, prototipaje y entrega; así como las evaluaciones de usabilidad correspondientes a cada etapa del proceso. Se realizan en ciclos iterativos hasta alcanzar el producto final. En cada etapa del proceso de desarrollo de software, se incluyen actividades propias para la construcción de la usabilidad.

3.1.1 Principios:

AgilUs centra el desarrollo de software en los siguientes principios:

- Integra la Interacción Humano Computador (IHC) y la Ingeniería de Software (IS). IS y IHC son complementarias, no son disciplinas excluyentes. Un diseño centrado en el usuario impacta positivamente en la calidad del software.
- La usabilidad debe considerarse desde el principio del desarrollo. Si la IS y la IHC son complementarias y no excluyentes, y si la usabilidad aumenta la calidad del software,

entonces es conveniente incluir la usabilidad desde el principio en el desarrollo como uno de los requisitos para impactar positivamente la calidad del producto final.

- La usabilidad determina la utilidad. Un software se considera útil en la medida que pueda ser usado a fin de producir resultados, en forma eficiente, intuitiva y satisfactoria para los usuarios.
- El usuario determina la usabilidad, la cual no es una propiedad abstracta. Un software sólo será considerado usable en un contexto y por un tipo de usuario específico. El objetivo es lograr que todos los usuarios del software encuentren usables las tareas que pueden realizar.

3.1.2 Mejores prácticas:

Algunas de las “buenas prácticas” del desarrollo de software que se aplican en *AgilUs*, las cuales están enfocadas en satisfacer las demandas del usuario y el desarrollo iterativo e incremental, procurando la usabilidad en cada paso del proceso de desarrollo, son las siguientes:

- Diseño centrado en el usuario (DCU). El DCU es un enfoque de diseño y desarrollo que se centra en los deseos, limitaciones y necesidades de los usuarios finales de un software. En las técnicas de DCU es relevante que los desarrolladores realicen pruebas constantes para verificar el curso que lleva el desarrollo del sistema y su interfaz de usuario. De este modo, el usuario guía indirecta pero influyentemente el proceso de desarrollo del sistema. La diferencia fundamental entre éste y otros enfoques de diseño es que en el DCU se procura construir el sistema para adaptarse, a través de su interfaz, a cómo el usuario desea trabajar, en lugar de forzar al usuario a cambiar su modo de trabajar para adaptarse a lo que los desarrolladores consideraron apropiado.
- Diseño basado en prototipos. El desarrollo de software en *AgilUs* está guiado por la construcción de prototipos de alta fidelidad y la evaluación de los mismos por los usuarios y por especialistas en usabilidad. Se entiende entonces que, tras una inspección inicial, los desarrolladores producen un primer prototipo, los especialistas y el usuario lo evalúan, los analistas preguntan directamente al usuario sus opiniones sobre el desarrollo, y con esa retroalimentación, los desarrolladores se disponen a producir un

siguiente prototipo. Este ciclo continúa hasta que se tiene un producto listo para la entrega, cuando las evaluaciones de usabilidad, requerimientos y calidad del software están completamente satisfechas.

- Desarrollo ágil, iterativo e incremental. Una de las máximas del desarrollo iterativo e incremental y del Manifiesto Ágil es la simplicidad. Se recomienda entonces desarrollar el sistema más simple que satisfaga las necesidades actuales de los usuarios, preparándose para cambios futuros. El desarrollo por incrementos permite proveer resultados sin necesidad de esclarecer todo los requisitos de una vez al inicio del desarrollo. La iteratividad permite regresar a etapas anteriores una vez recibida la retroalimentación producto de las evaluaciones realizadas.
- Usabilidad como atributo de la calidad. Como se indica en el estándar ISO/IEC 9126-1, la usabilidad es considerada un atributo de la calidad interna y externa del software, y *AgilUs* hace énfasis en la producción de software usable, siguiendo la recomendación de este estándar internacional.
- Interacción continua con el usuario, propiciando un intercambio cara a cara. Naturalmente, para *AgilUs* la presencia constante y participativa del usuario es fundamental. El equipo de desarrollo sólo puede tomar decisiones tras realizar evaluaciones de usabilidad, y la usabilidad del sistema sólo puede ser determinada por el usuario.

3.1.3 Ciclo de vida

El ciclo de vida de *AgilUs* hace énfasis en la importancia del usuario y sus evaluaciones. Está basado en el refinamiento iterativo e incremental de prototipos de alta fidelidad hasta que se convierten en el producto final para entrega. Este producto *final* puede ser posteriormente modificado a través de un mantenimiento correctivo y/o evolutivo, que no está contemplado como parte del método.

En cada etapa del desarrollo se incluyen actividades para la construcción de la usabilidad. Se busca proporcionar una manera de proceder organizadamente para construir la usabilidad

durante el desarrollo de un producto. El ciclo de vida engloba la definición de requisitos, análisis, prototipaje y entrega.

La Figura 8 muestra un diagrama de la relación entre cada una de las etapas del ciclo de vida de *AgilUs*, con las actividades que se realizan y artefactos que se generan en cada etapa.



Figura 8. El método AgilUs: etapas, actividades y artefactos. Autor: Acosta - 2010.

A continuación se describen las etapas de este método:

- **Requisitos:** se realiza el análisis global del problema a solucionar, se estudian productos similares existentes, se genera un perfil de usuario, y se define la lista de requerimientos a desarrollar. Esta etapa es importante en el desarrollo del software, ya que un mal análisis de requisitos traería como consecuencia un software que no cumple con las necesidades del usuario.
- **Análisis:** se lleva a cabo el análisis de la solución a desarrollar, se emplean diagramas de casos de uso y modelo de objetos del dominio, siguiendo la notación UML (*Unified*

Modeling Language - Lenguaje Unificado de Modelado), para definir las funcionalidades que tendrá el producto a desarrollar.

- Prototipaje: se implementa un prototipo rápido de la interfaz de usuario a partir de los patrones de interacción, el cual va evolucionando hasta convertirse en el producto final, se genera la guía de estilo, y se realizan evaluaciones de usabilidad apropiadas a esta etapa: las evaluaciones heurísticas y las listas de comprobación.
- Entrega: se aplican las pruebas al sistema para certificar que la aplicación desarrollada sea un software usable y sin errores, finalmente se pone en producción la aplicación.

Las metodologías ágiles se deben aplicar en proyectos donde exista incertidumbre, donde el entorno es volátil, cuando los requisitos no se conocen con exactitud, mientras que las metodologías tradicionales obligan al cliente a tomar las decisiones al inicio del proyecto.

CAPÍTULO IV. Marco Aplicativo

Para el logro exitoso de los objetivos planteados en el Capítulo I, se decidió utilizar una adaptación del método de desarrollo *AgilUs*.

4.1 Aplicación del método AgilUs

En las siguientes secciones del documento se describen las etapas que comprenden el método *AgilUs* y las iteraciones llevadas a cabo en cada una de ellas para el desarrollo de la aplicación.

4.2 Etapa de requisitos

En esta etapa se tomaron en cuenta las tormentas de ideas hechas por los desarrolladores, como del personal que labora en la COMPAS, tomando en cuenta tanto los perfiles de usuario, como la lista de requerimientos funcionales y no funcionales, los cuales son artefactos propuestos en el método *AgilUs*.

4.2.1 Tormenta de Ideas

La tormenta de ideas se efectuó entre el grupo de desarrollo y los integrantes de la COMPAS de la Escuela de Computación, y consistió en intercambiar ideas sobre el problema en cada uno de los módulos y sus principales funcionalidades. A continuación se muestran las ideas que fueron tomadas en cuenta del proceso creativo:

- Introducir el concepto de “Sistema”, cuya función es agrupar la gestión e interacción entre los módulos de administrador, pasante y tutor.
- Diseñar y elaborar las vistas de administrador y principal de SIGEPAS.
- Diseñar y elaborar las vistas para pasante y tutor.
- Introducir el concepto de “Funciones”, ya que por parte del administrador es la persona que más funcionalidades debe tener durante su desarrollo.
- Generar los reportes, constancias y cartas en formato PDF para su posterior impresión.
- Diseñar un medio directamente desde el sistema para el envío de correos con plantillas reutilizables para los pasantes y tutores.

-
- Implementar una función de envíos de correos masivos, es decir, a todos los pasantes activos (estudiantes cursando pasantías en el semestre actual), sin necesidad de seleccionar uno a uno.
 - Aplicar una función para que el pasante pueda enviar el informe final de pasantías a través del sistema y el tutor asignado pueda evaluarlo.
 - Implementar un método en donde el tutor pueda visualizar en formato PDF el informe final del pasante, donde pueda corregir y posteriormente evaluar. Así mismo, que el tutor pueda visualizar el documento de formalización de pasantías y por último generar constancias.
 - Elaborar y definir el modelo de base de datos, tomando en cuenta la creación o eliminación de nuevas tablas o relaciones.

4.2.2 Perfil de Usuarios

En el desarrollo de la aplicación web para SIGEPAS se identificaron y caracterizaron los siguientes perfiles de usuarios, que accederán o manipularán el contenido del mismo.

- ***Usuario Administrador***

Este usuario es el encargado del *Módulo Administrativo* de la aplicación, el cual tiene como tarea la responsabilidad de mantener una participación activa de estudiantes y profesores en empresas, industrias e instituciones, a través de manejo de información, publicación de ofertas, noticias y documentos que ayuden al estudiante en su desarrollo como pasante. También, como a su vez el coordinador, es el encargado de dirigir, supervisar y controlar los programas de pasantías de la Escuela de Computación de la Facultad de Ciencias, tanto para los estudiantes en curso como para los profesores que serán los tutores académicos.

El administrador tiene la potestad de inscribir un nuevo estudiante a parte de los asignados por la División Control de Estudios (DCE) que hayan aprobado todos los requisitos obligatorios, como también poder evaluarlos en caso tal de que el tutor asignado haya perdido el derecho. A su vez, el administrador puede generar reportes informativos referentes a pasantes, tutores e información general del semestre, generar

constancias para pasantes que indique que fueron completadas las horas de labor en dicha empresa o institución, como también para el tutor académico indicando que cumplió como tutor para el estudiante durante su desarrollo en dicho período. Así mismo, éste módulo debe permitir la generación de reportes en cuanto a cantidad de estudiantes en un semestre seleccionado y cantidad de pasantes retirados y/o aprobados. Por último el administrador también tendrá la potestad de retirar y reprobar pasantes a través del sistema de gestión automatizado.

- ***Usuario Pasante***

El pasante ingresa a la aplicación web en busca de información como; noticias, ofertas de empleos para pasantías, documentos de normativas de pasantías, teléfonos o correos electrónicos de personas que pudieran asesorarlo en este desarrollo obligatorio.

Este perfil de usuario, a través de SIGEPAS, puede generar cartas de postulación que serán dirigidas a las empresas o instituciones en las cuales desee ejercer la labor de pasante y en las cuales tengan la disponibilidad de realizarlo, como a su vez, tienen la obligación de formalizar dicha pasantía una vez aceptado como pasante dentro de la organización o institución, indicando cierta información de parte de la organización o institución donde el pasante está por realizarla, información del tutor organizacional e información general de la pasantía, dejando por escrito y firmado que el estudiante realizará las pasantías en dicha empresa, durante cierto tiempo, dando fecha de inicio y fecha final del periodo, entre otros datos importantes; para que así, le sea asignado un tutor académico y quede como constancia de que el pasante ha formalizado su labor obligatoria.

Por otra parte es necesario la revisión del informe final, en tal sentido se implementaron las funcionalidades de poder cargar el informe en formato PDF al sistema, para que el tutor académico a través del sistema pueda descargárselo y así poder corregir y evaluar.

Una vez finalizada la pasantía el pasante debe subir dos planillas; Desempeño del Pasante y Control de Pasantías, ambas llenadas y firmadas por el tutor organizacional y

tutor académico, donde quede constancias de la labor ejercida por el pasante dentro de la organización y su desempeño durante su labor.

- ***Usuario Tutor***

El tutor académico, es el asesor directo de cada pasante dentro de la universidad, es importante mencionar que a cada estudiante se le asigne un tutor académico que lo asesore, lo supervise y evalúe durante su labor obligatoria. Este tipo de usuario tiene el derecho de evaluar el informe final, como también, servir de representante académico frente a empresas o instituciones que requieran de esta importante labor, es por ello, que el rol que cumple el tutor académico es tan importante como los antes mencionados.

La funcionalidad que debe tener el tutor académico a través del sistema de gestión de pasantías es descargar el informe final, leerlo y por último evaluarlo, como también ver la planilla de Desempeño del Pasante, Control de Pasantías para su verificación final y la visualización de la formalización, para que así el tutor tenga conocimiento de las pasantías realizadas.

4.2.3 Lista de Requerimientos Funcionales y No Funcionales

Basándose en el estudio de los antecedentes expuestos en el Capítulo I, y los artefactos descritos hasta el momento, se definen los requerimientos funcionales y no funcionales.

Los requerimientos funcionales son:

- Desarrollar las vistas del proyecto.
- Exportar en formato PDF los reportes, constancias y cartas de postulación.
- Desarrollar un mecanismo para envíos de correos masivos o selectivos a pasantes y tutores.
- Diseñar e implementar las estructuras de datos para almacenar la información de pasantes según el semestre en curso, como también de los anteriores.

Los requerimientos no funcionales son:

-
- Brindar una interfaz de usuario intuitiva e interactiva para todos los perfiles de usuario y la página principal de SIGEPAS.
 - Proporcionar un código fuente legible y entendible, ya que puede estar sujeto a cambios, bien sea por nuevas funcionalidades o corrección de errores.
 - Disponibilidad, ya que el administrador podrá consultar información de semestres anteriores o todos aquellos que no estén activos.

4.3 Análisis

En esta etapa se analizó la solución requerida para el desarrollo del sistema, tomando en consideración los siguientes artefactos: modelo de casos de uso, modelo objeto del dominio, guía de estilo, prototipo en papel y patrones de interacción.

4.3.1 Modelos de casos de uso

Un modelo de casos de uso describe la secuencia de las interacciones que se desarrollarán entre actores y el sistema, en respuesta a un evento que inicia un actor (Acosta, 2010). En el sistema de Comisión de Pasantías se definieron los actores que interactúan con esta aplicación, uno de los actores se denominó como *Administrador*, el cual desempeñará su labor desde el módulo administrativo para coordinar, supervisar y evaluar los programas de pasantías dentro de la Escuela de Computación de la Facultad de Ciencias. El segundo actor se denominó como *Pasante* y son todos aquellos estudiantes que hayan cumplido con todos los requisitos exigidos por la División de Control de Estudios (DCE), para poder cursar las pasantías, los cuales tienen la posibilidad de acceder al sistema para poder cargar su informe final, planilla de Desempeño del Pasante y Control de Pasantía, generar cartas de postulación y formalizar pasantías. Por último se denominó al tercer actor como *Tutor*, y son todos aquellos profesores y profesoras que fueron asignados a cada pasante para servir de ayuda y soporte durante su labor, tienen la posibilidad de acceder al sistema para evaluar al pasante correspondiente.

El diagrama cuenta con tres niveles de análisis, los cuales se muestran a continuación en las siguientes figuras.

- El nivel cero muestra los actores que interactúan con SIGEPAS, como se muestra en la Figura 9.

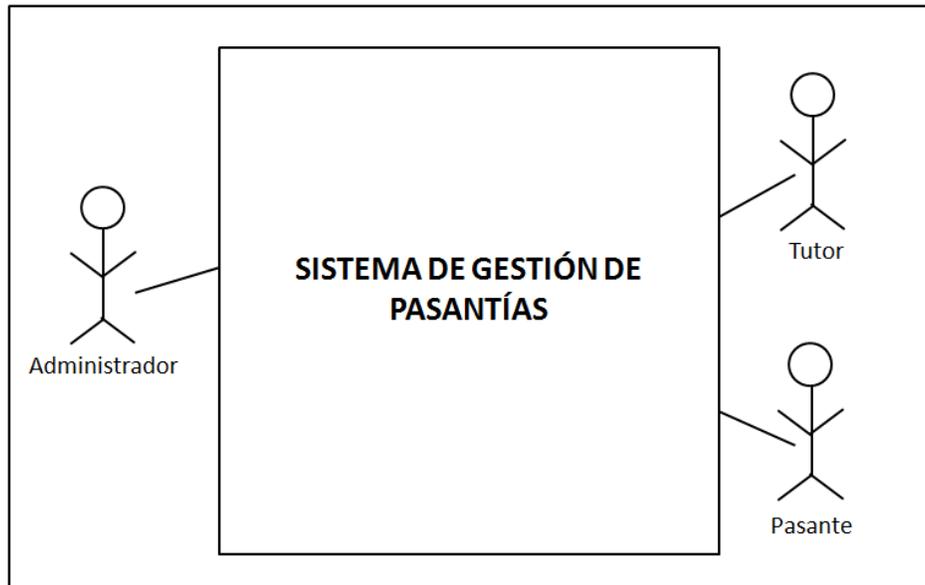


Figura 9. Modelo de Casos de Uso nivel cero. Autores: Florville – Montes -2013.

Nombre: Sistema de Gestión de Pasantías (SIGEPAS).

Actores: Administrador, Pasante y Tutor.

Acción: Coordinar, controlar, supervisar y asesorar los procesos y funcionalidades para llevar a cabo las pasantías en la Escuela de Computación de la Facultad de Ciencias.

Pre-condición: se requiere de un usuario administrador, de un usuario pasante y de un usuario tutor para completar SIGEPAS.

Post-condición: El usuario administrador debe asegurarse de que cada uno de los pasantes haya obtenido una nota calificativa.

- El nivel uno muestra la interacción de los actores con las funcionalidades más generales del sistema, como se observa en la Figura 10.

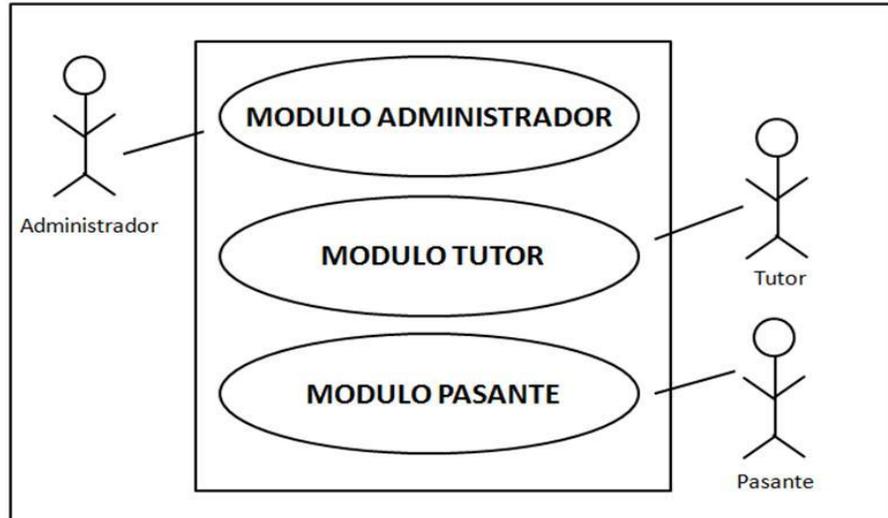


Figura 10. Modelo de Casos de Uso nivel uno. Autores: Florville – Montes -2013.

Nombre: Módulo Administrador.

Actor: Administrador.

Acción: funcionalidad que permite al administrador coordinar, supervisar y evaluar los programas de pasantías. En el caso de módulo de pasante y tutor administrar información de interés para ambos.

Pre-condición: se requiere que el usuario haya ingresado al sistema de Comisión de Pasantías a través de su correo electrónico y clave.

Post-condición: el usuario puede acceder a funcionalidades de exportar a formato PDF, así como evaluar y pre-visualizar datos referentes a pasantes y tutores.

Nombre: Módulo Pasante.

Actor: estudiante (cursando pasantías).

Acción: funcionalidad que permite al usuario subir informes al sistema, generar cartas de postulación y formalizar pasantías.

Pre-condición: se requiere que el usuario haya aprobado los requisitos de la División de Control de Estudios (DCE) para cursar pasantías, en tal caso puede ingresar al modulo de pasantes a través de su cédula y clave proporcionada por el sistema vía correo electrónico.

Post-condición: el usuario puede formalizar pasantías y cargar el informe final al sistema para que el tutor pueda evaluarlo, como también puede cargar las planillas de Desempeño del Pasante y Control de Pasantías, generar cartas de postulación y visualizar información general.

Nombre: Módulo Tutor.

Actor: profesor o profesora asignado a un pasante.

Acción: funcionalidad que permite al usuario poder evaluar al pasante asignado.

Pre-condición: Se requiere que el tutor haya ingresado al sistema mediante su correo electrónico y clave para visualizar información referente a su pasante, debe tener un pasante asignado.

Post-condición: El usuario puede descargar el informe final y las planillas de Desempeño del Pasante y Control de Pasantía, y evaluarlo a través del sistema.

- El nivel dos, desglosa los casos de uso generales, para estudiar en detalle cada una de las acciones que proveen al usuario. En la Figura 11 se visualiza el caso de uso detallado para el módulo administrativo.

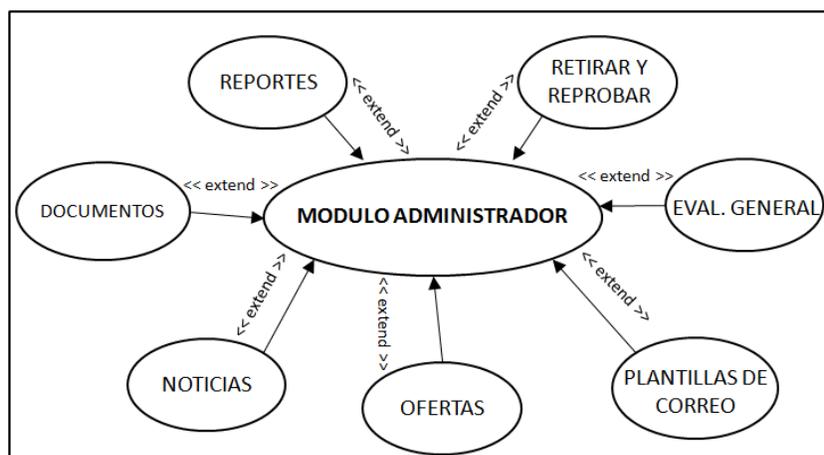


Figura 11. Modelo de casos de uso nivel dos administrador. Autores: Florville – Montes -2013.

Nombre: Reportes.

Actor: Administrador.

Acción: funcionalidad que permite al usuario exportar a formato PDF reportes y constancias a los tutores.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su correo electrónico y clave.

Nombre: Documentos.

Actor: Administrador.

Acción: funcionalidad que permite al usuario subir documentos al sistema para ser vistos y posiblemente descargarlos en la página principal del sistema.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su correo electrónico y clave.

Nombre: Noticias.

Actor: Administrador.

Acción: funcionalidad que permite al usuario publicar noticias referentes a las pasantías.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su correo electrónico y clave.

Nombre: Ofertas.

Actor: Administrador.

Acción: funcionalidad que permite al usuario publicar ofertas de empresas que buscan pasantes para cumplir con sus pasantías.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su correo electrónico y clave.

Nombre: Plantillas de correo.

Actor: Administrador.

Acción: funcionalidad que permite al usuario crear plantillas de correo para ser reutilizadas a futuro.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su correo electrónico y clave.

Nombre: Evaluación General.

Actor: Administrador.

Acción: funcionalidad que permite al usuario evaluar al pasante.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su correo electrónico y clave, y el tutor asignado a dicho pasante no se encuentre habilitado para realizar la acción.

Nombre: Retirar y Reprobar.

Actor: Administrador.

Acción: funcionalidad que permite al usuario retirar y reprobar el pasante.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su correo electrónico y clave, luego el administrador puede retirar al pasante o reprobarlo según sea el caso.

En la Figura 12 se visualiza el caso de uso detallado para el módulo pasante.

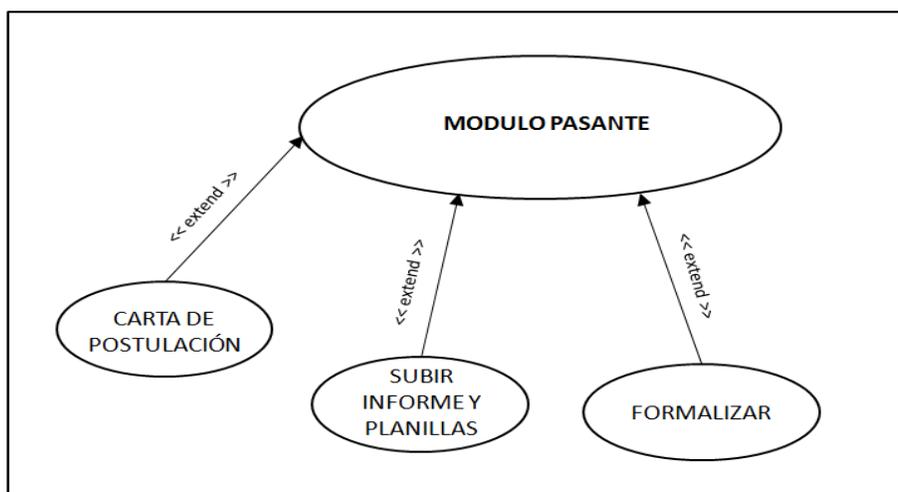


Figura 12. Modelo de casos de uso pasante. Autores: Florville – Montes -2013.

Nombre: Carta de Postulación.

Actor: Pasante.

Acción: funcionalidad que permite al usuario generar cartas de postulaciones para ser presentadas a las empresas en busca de pasantías.

Pre-condición: no requiere ingresar al sistema para generar cartas de postulación.

Nombre: Subir Informe final.

Actor: Pasante.

Acción: funcionalidad que permite al usuario enviar el informe final al sistema.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su cédula y clave.

Nombre: Formalizar.

Actor: Pasante.

Acción: funcionalidad que permite al usuario formalizar las pasantías y generarla en formato PDF para obtener una constancia.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su número de cédula y clave.

Por último en la Figura 13 se visualiza el caso de uso detallado para el módulo Tutor.

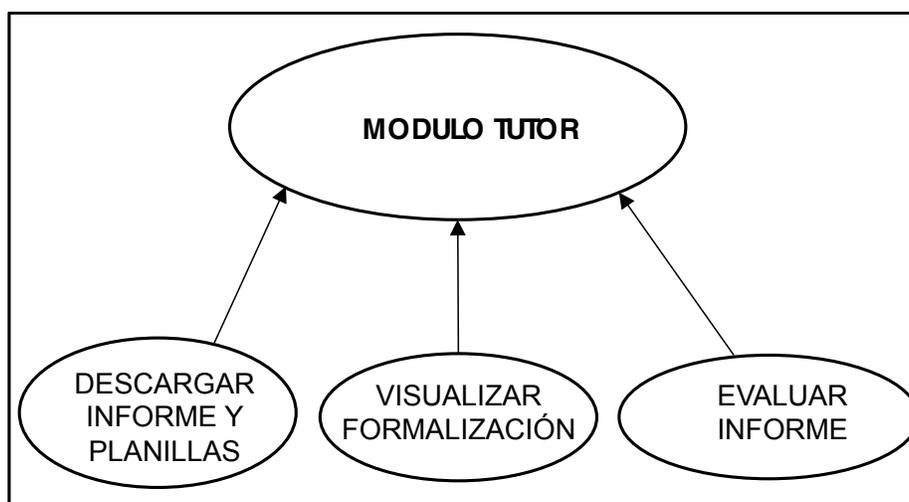


Figura 13. Modelo de casos de uso Tutor. Autores: Florville – Montes -2013.

Nombre: Descarga de Informe y Planillas.

Actor: Profesor Asignado.

Acción: funcionalidad que permite al usuario poder descargar el informe final y planillas de Desempeño del Pasante y Control de Pasantía del pasante.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su correo electrónico y clave.

Nombre: Visualizar Formalización.

Actor: Profesor Asignado.

Acción: funcionalidad que permite al usuario poder visualizar la formalización del pasante.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su correo electrónico y clave.

Nombre: Evaluar informe.

Actor: Profesor Asignado.

Acción: funcionalidad que permite al usuario poder evaluar al pasante asignado visualizando el informe final enviado por el pasante.

Pre-condición: se requiere que el usuario haya ingresado al sistema mediante su correo electrónico y clave.

4.3.2 Modelos Objetos del Dominio

El Modelo Objeto del Dominio, es aquel que describe los objetos del dominio del sistema y las relaciones estáticas que lo conforman (Acosta, 2010). En la Figura 14 se presenta el modelo de objeto del dominio de SIGEPAS.

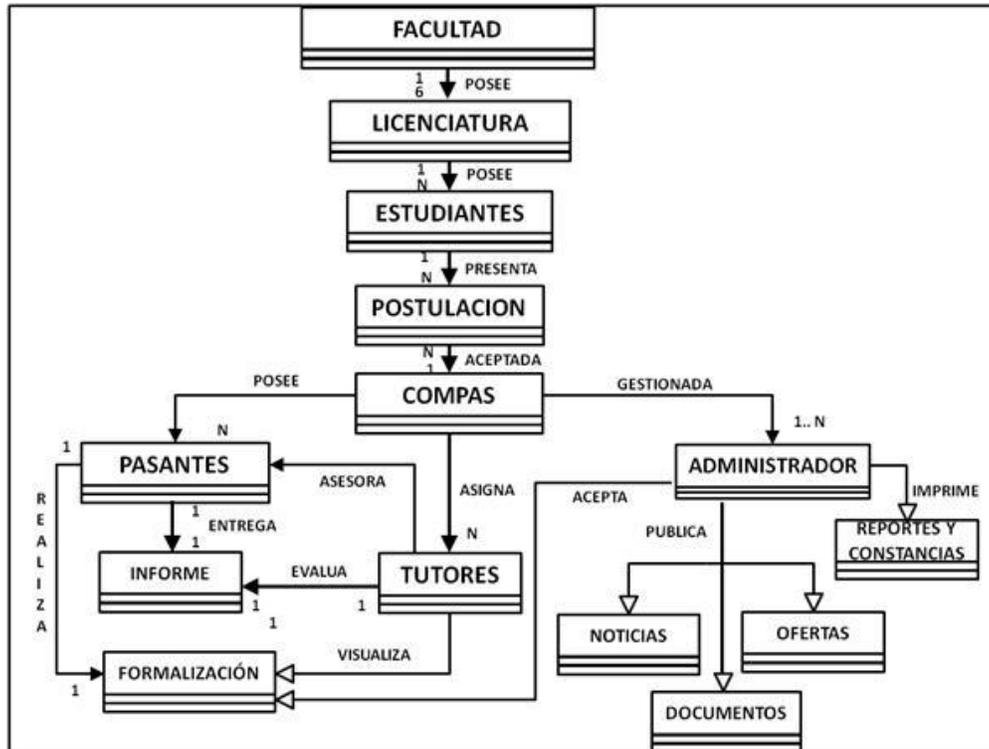


Figura 14. Modelo de objeto del dominio del modelo de SIGEPAS. Autores: Florville – Montes -2013.

Para el SIGEPAS se enumeran los siguientes objetos:

1. Facultad: es una institución docente donde se imparten estudios superiores, en este contexto se entiende por Facultad de Ciencias de la UCV.
2. Licenciatura: son el conjunto de estudios necesarios para conseguir el título universitario.
3. Estudiantes: es el conjunto de personas que optan por un título universitario.
4. Postulación: carta que puede ser genera por el pasante para ser entrega a las empresas indicando que se postula como pasante.
5. COMPAS: ente encargado de coordinar, supervisar, evaluar y publicar todos los programas de pasantías dentro de la Facultad de Ciencias.
6. Pasantes: conjunto de estudiantes que se encuentran cursando pasantías.
7. Administrador: persona encargada del sistema de Gestión de Pasantías (SIGEPAS).

8. Tutor: profesor asignado a cada pasante para servir de soporte y ayuda durante su curso en las pasantías.
9. Informe: documento final que es requerido de manera obligatoria al culminar las pasantías.
10. Formalizar: proceso obligatorio por parte del pasante al momento de determinar donde realizará las pasantías.

Una vez enumerados los objetos del dominio se puede establecer la relación existente entre ellos:

La Facultad de Ciencias posee Licenciaturas, que poseen Estudiantes que se postulan como pasantes a la COMPAS para cursar sus pasantías, una vez aceptada, bajo ciertas condiciones implementadas en el reglamento de la Escuela de Computación debe formalizarla, luego de esto, se le asigna un tutor académico al pasante, el cual debe evaluar el informe final y ver la formalización entregada por el pasante y asesorarlo en cualquier inconveniente o ayuda que solicite. Como caso aparte el Administrador tiene el derecho de aceptar la formalización y evaluar el informe final en caso de que el pasante no tenga tutor académico asignado o éste no se encuentre disponible para su evaluación.

4.3.3 Guía de Estilo

La guía de estilo recoge las normativas y patrones básicos relacionados con el aspecto de una interfaz para su aplicación en el desarrollo de pantallas dentro de un entorno concreto (Acosta, 2010). A continuación se desglosa la guía de estilo utilizada en el desarrollo de la página principal de la COMPAS.

- ***Logo***

El logo que se muestra en la Figura 15 es utilizado como cabecera de la página principal de SIGEPAS, el cual tiene unas dimensiones de 800 x 100 px.



Figura 15. Logotipo de SIGEPAS. Autores: Florville – Montes -2013.

- **Colores Principales**

En la Figura 16 se aprecia la relación de los principales colores de la página principal, están descritos con su color hexadecimal y sus valores en RGB.

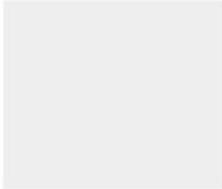
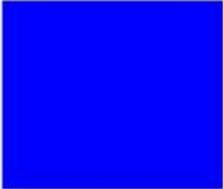
			
RGB 238, 238, 238 #EEEEEE	RGB 0, 0, 0 #000000	RGB 255, 255, 255 #FFFFFF	RGB 0, 0, 255 #0000FF

Figura 16. Colores implementados. Autores: Florville – Montes -2013.

- **Tipografía**

La fuente utilizada dentro de sistema fue “Helvetica Neue”, variando el tamaño del texto entre 13px y 35px dependiendo de la relevancia del mismo.

- **Uso de menú para el sistema**

El formato del menú se va a repetir tanto para la página principal (módulo de pasante y tutor), como para el del administrador, variando sus opciones según la vista en donde se encuentre el usuario. En la Figura 17 se refleja el menú para la página principal.



Figura 17. Menú para el Sistema. Autores: Florville – Montes -2013.

En la Figura 18 se refleja el menú para el modulo del administrador.



Figura 18. Menú Modulo Administrador. Autores: Florville – Montes -2013.

4.3.4 Prototipo en Papel

En esta sección se presenta de manera secuencial los diferentes prototipos iniciales de las interfaces de usuario, a través de las cuales el usuario debe navegar, para así poder generar cartas de postulación, formalizar su pasantía con la COMPAS y exportarlos en formato PDF, como a su vez, el formato de reportes, correos y constancias por parte del administrador. En la Tabla 1 se puede observar los prototipos de las distintas interfaces.

Tabla 1. Prototipo en Papel

DESCRIPCIÓN	PROTOTIPO EN PAPEL
<p>Para poder generar la carta de postulación el usuario no requiere ingresar al sistema, por lo que se le coloca la opción en la página principal a través de un botón que especifique “Carta de Postulación”.</p>	

Tabla 2. Prototipo en papel (Continuación Tabla 1).

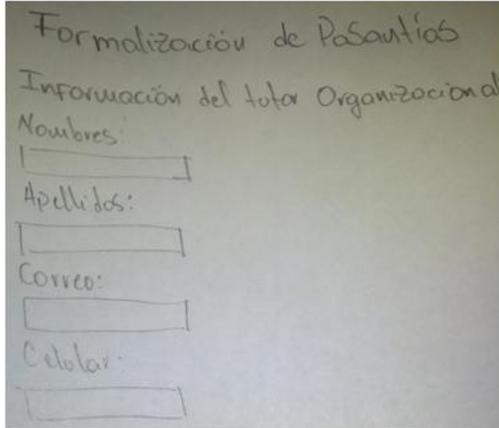
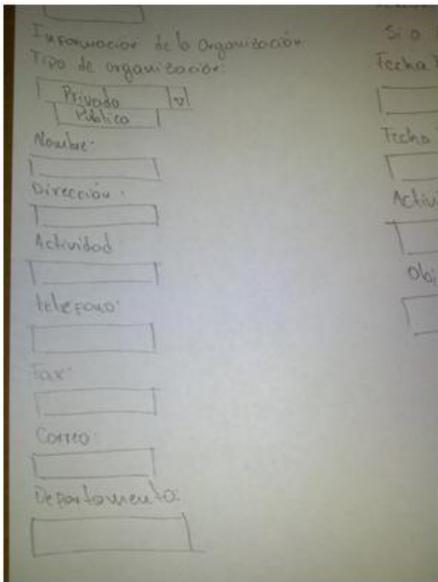
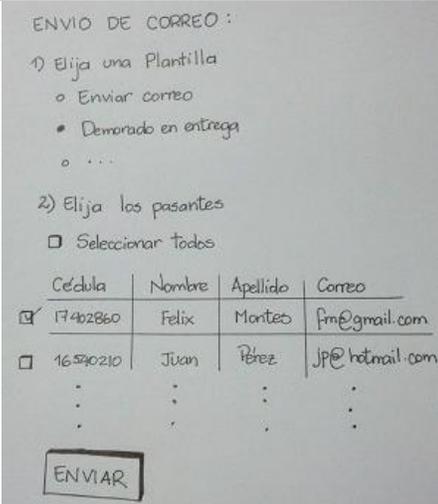
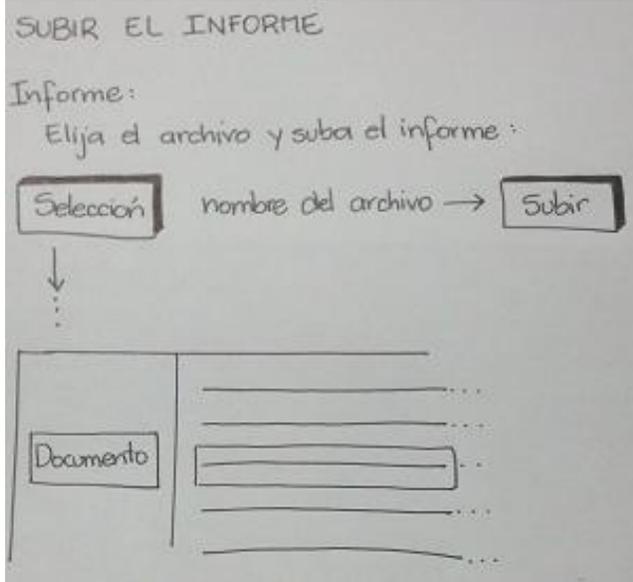
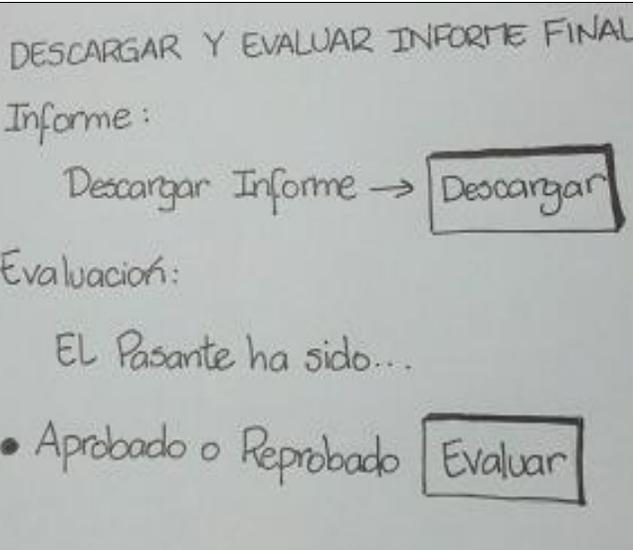
DESCRIPCIÓN	PROTOTIPO EN PAPEL																									
<p>En el siguiente paso, una vez que el usuario ingrese al sistema tiene el derecho de formalizar la pasantía llenando los campos; Información del tutor Organizacional, Información de la Organización e Información sobre la Pasantía, lo cual el sistema al formalizar le genera una constancia la cual puede ser exportada en formato PDF para su posterior verificación con el administrador.</p>	 																									
<p>En la siguiente vista se muestran como es la selección de los pasantes a los cuales el administrador necesite mandarles un correo, eligiendo una plantilla de correo para su contenido.</p>	 <p>ENVIO DE CORREO :</p> <p>1) Elija una Plantilla</p> <ul style="list-style-type: none"> <input type="radio"/> Enviar correo <input checked="" type="radio"/> Demorado en entrega <input type="radio"/> ... <p>2) Elija los pasantes</p> <p><input type="checkbox"/> Seleccionar todos</p> <table border="1" data-bbox="922 1627 1349 1795"> <thead> <tr> <th></th> <th>Cédula</th> <th>Nombre</th> <th>Apellido</th> <th>Correo</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>174b2860</td> <td>Felix</td> <td>Montes</td> <td>fm@gmail.com</td> </tr> <tr> <td><input type="checkbox"/></td> <td>16570210</td> <td>Juan</td> <td>Pérez</td> <td>jp@hotmail.com</td> </tr> <tr> <td></td> <td>⋮</td> <td>⋮</td> <td>⋮</td> <td>⋮</td> </tr> <tr> <td></td> <td>⋮</td> <td>⋮</td> <td>⋮</td> <td>⋮</td> </tr> </tbody> </table> <p><input type="button" value="ENVIAR"/></p>		Cédula	Nombre	Apellido	Correo	<input checked="" type="checkbox"/>	174b2860	Felix	Montes	fm@gmail.com	<input type="checkbox"/>	16570210	Juan	Pérez	jp@hotmail.com		⋮	⋮	⋮	⋮		⋮	⋮	⋮	⋮
	Cédula	Nombre	Apellido	Correo																						
<input checked="" type="checkbox"/>	174b2860	Felix	Montes	fm@gmail.com																						
<input type="checkbox"/>	16570210	Juan	Pérez	jp@hotmail.com																						
	⋮	⋮	⋮	⋮																						
	⋮	⋮	⋮	⋮																						

Tabla 3 Prototipo en papel (Continuación Tabla 2)

DESCRIPCIÓN	PROTOTIPO EN PAPEL
<p>Para este prototipo en papel se implementó la vista de subir el informe al sistema para que el tutor asignado pueda evaluarlo</p>	 <p>The prototype is titled "SUBIR EL INFORME". It includes the instruction "Informe: Elija el archivo y suba el informe:". Below this, there is a box labeled "Selección" followed by the text "nombre del archivo" and an arrow pointing to a box labeled "Subir". A vertical dashed line with three dots is positioned below the "Selección" box. At the bottom, there is a box labeled "Documento" next to a list of horizontal lines representing a document list.</p>
<p>Por último se implementó la vista para cuando el tutor evalúe el informe final del pasante, pudiendo descargar el archivo subido por el pasante y determinar su nota final directamente desde el sistema.</p>	 <p>The prototype is titled "DESCARGAR Y EVALUAR INFORME FINAL". It includes the instruction "Informe: Descargar Informe" followed by an arrow pointing to a box labeled "Descargar". Below this, it says "Evaluación: EL Pasante ha sido...". At the bottom, there is a bullet point "• Aprobado o Reprobado" followed by a box labeled "Evaluar".</p>

En base a este prototipo inicial, se realizaron las diferentes interfaces de usuario que poseen los módulos de administrador, pasante y tutor, gracias a que se plasmaron de forma visual las interacciones necesarias para llevar a cabo el proceso de funcionalidades de la COMPAS.

4.3.5 Patrones de Interacción

Los patrones de interacción describen una solución exitosa a un problema recurrente concerniente a la interfaz de usuario, en el contexto dado (Acosta, 2010).

Las interfaces de usuarios del sistema para el módulo administrativo requieren de mucha interacción por parte del usuario, por lo cual cuenta con un gran número de patrones de interacción, los cuales son: Inscribir pasante mediante un archivo en formato XLS o de manera individual, crear semestre, eliminar semestre, modificar semestre, mandar por correo electrónico contraseña de acceso al pasante, mandar por correo electrónico contraseña de acceso al tutor, crear plantillas de correos, eliminar plantillas de correo, modificar plantillas de correo, mandar correos electrónicos selectivos o masivos a pasantes y tutores, publicar noticias, publicar ofertas, publicar documentos, asignar tutor a pasante, cambiar asignación de tutor, genera reportes en formato PDF, genera constancias en formato PDF, descarga informes de pasantes, descargar formalización de pasante, evaluar informes y evaluar formalización.

Para este estudio se decidió mostrar los patrones de diseños de publicar información, envío de correos, evaluar pasante y asignación de tutor.

Tabla 4. Patrones de Iteración – Publicar Informe.

Nombre	Publicar Información
Problema	El administrador del sistema necesita publicar noticias, documentos y ofertas.
Solución	En el módulo administrativo, se crearon tres funcionalidades por separado, para la creación de noticias, ofertas y un mecanismo para subir un archivo o documento al sistema, colocando titulo, descripción y contenido de cada uno.

Tabla 5 Patrones de iteración- Publicar Informe (Continuación Tabla 4)

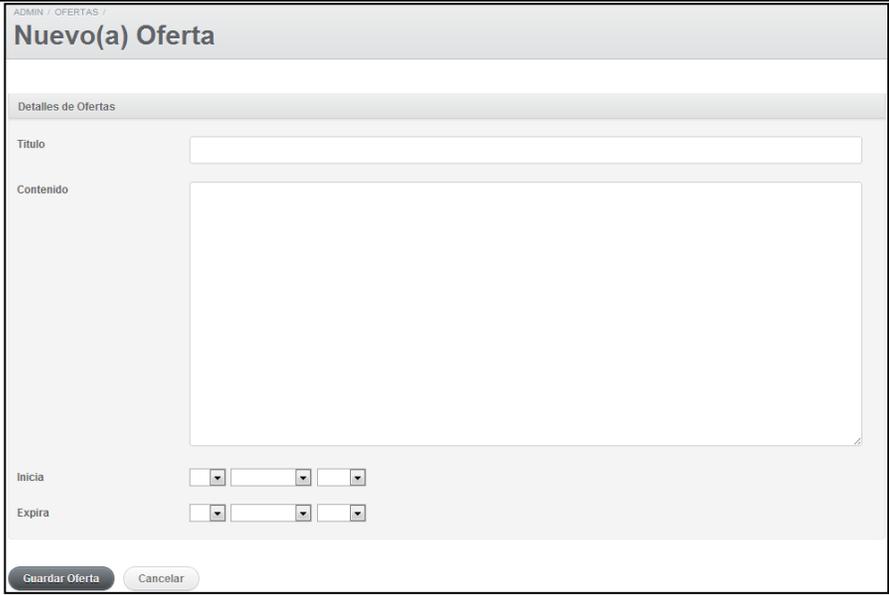
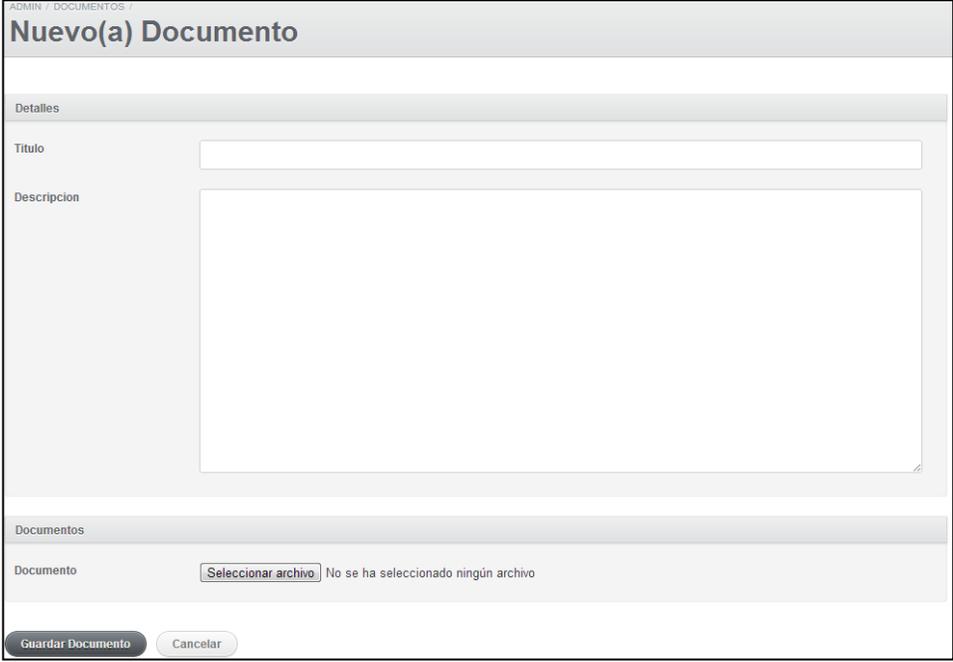
<p>Solución</p>	
	<p>Esta vista se repite tanto para la creación de ofertas y noticias.</p>  <p>Aquí se muestra la vista para publicar un nuevo documento, es aceptado tanto el formato PDF como .DOC (Microsoft Word).</p>
<p>Contexto</p>	<p>Aplicación Web</p>

Tabla 6 . Patrones de iteración- Publicar Informe (Continuación Tabla 5).

Usabilidad	Satisfacción del Usuario, reducción de tiempo a la hora de crear y publicar noticias, ofertas y documentos.
Fuerza	El usuario debe tener conocimiento básico del computador.
Consecuencias	El usuario debe tener precaución al momento de crear algunos de estos, ya que serán publicados automáticamente en el sistema al ser creados.
Patrones Relacionados	Crear planillas de correo e Inscribir Pasantes por medio de la lectura de un archivo en formato .XLS proporcionado por la División de Control de Estudios a través del sistema CONEST.

Tabla 7. Patrones de Iteración - Envío de Correos.

Nombre	Envíos de Correos									
Problema	El usuario administrador necesita de un mecanismo de envío de correos para cada o todos los pasantes y para cada o todos los tutores.									
Solución	<p>Se creó un método en el cual el usuario puede elegir a través de un “check box” el o los pasantes o tutores a quien desee enviarles el correo, eligiendo previamente una plantilla de correo.</p>  <p>The screenshot shows a web form with the following elements:</p> <ul style="list-style-type: none"> Title: Elija los pasantes a quien desee enviarle el correo: Checkbox: <input checked="" type="checkbox"/> Seleccionar Todos Table: <table border="1"> <thead> <tr> <th>Seleccionar</th> <th>Nombres</th> <th>Ap</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>Felix</td> <td>Mo</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>Antonio</td> <td>Silv</td> </tr> </tbody> </table> Button: Enviar Correo 	Seleccionar	Nombres	Ap	<input type="checkbox"/>	Felix	Mo	<input checked="" type="checkbox"/>	Antonio	Silv
Seleccionar	Nombres	Ap								
<input type="checkbox"/>	Felix	Mo								
<input checked="" type="checkbox"/>	Antonio	Silv								
Contexto	Aplicación web									
Usabilidad	El usuario elige el o los pasantes a quien desee enviarles el correo, bien sea predefiniendo el contenido del correo a través de una plantilla de correo creada anteriormente.									

Tabla 8 Patrones de Iteración - Envío de Correos (Continuación Tabla 7).

Fuerza	El usuario debe tener conocimiento de los pasos a seguir para el envío de correos.
Consecuencia	El administrador debe crear una plantilla de correos previo al envío, ya que de no hacerlo el sistema arrojará un error indicándole que debe crearla, de la misma manera arrojará un error si no selecciona al menos un pasante o tutor a quien va dirigido el correo.
Patrones Relacionados	Envío de correos a tutores.

Tabla 9. Patrones de Iteración - Evaluar Pasante.

Nombre	Evaluar Pasante
Problema	El tutor académico asignado al pasante debe tener un mecanismo de evaluación para el informe final, en el cual puede aprobar o reprobar.
Solución	<p>En vista “Pasantes”, en el módulo administrador, se muestra la lista de pasantes activos, en la cual en uno de sus campos “Evaluar Informe” el administrador podrá evaluar el informe del pasante, en caso tal de que el tutor académico no se encuentre disponible.</p> 

Tabla 10 Patrones de Iteración - Evaluar Pasante (Continuación Tabla 9).

Contexto	Aplicación Web
Usabilidad	El administrador despliega la lista de pasantes activos y aquel que aún no haya sido evaluado por su tutor académico, puede optar por evaluarlo, seleccionando el enlace de evaluar informe en el campo "Evaluar Informe" de la lista desplegada. Asignando la nota de Aprobado o Reprobado sea bien el caso.
Fuerza	El usuario debe tener conocimiento básico en descargar un archivo en formato PDF o .DOC.
Consecuencia	El pasante obtendrá su nota de evaluación de pasantías una vez que el usuario haya descargado el informe y evaluado respectivamente directamente desde el sistema.
Patrones Relacionados	Evaluar formalización, descargar informe y descargar formalización.

Tabla 11. Patrones de Iteración - Asignación de Tutor.

Nombre	Asignación de tutor
Problema	Luego de que el pasante haya formalizado la pasantía debe existir una funcionalidad por parte del modulo administrador el cual le asigne un tutor académico al pasante.
Solución	En vista "Pasantes", en el módulo administrador, se muestra la lista de pasantes activos, en la cual en uno de sus campos "Tutor" el administrador podrá asignarle el tutor académico a cada pasante. A continuación se presentan las vistas:

Tabla 12 Patrones de Iteración - Asignación de Tutor (Continuación Tabla 11).

Solución	<table border="1"> <thead> <tr> <th>Cedula</th> <th>Nombres y Apellidos</th> <th>Descargar Informe</th> <th>Evaluar Informe</th> <th>Descargar Formalizacion</th> <th>Evaluar Formalizacion</th> <th>Tutor</th> </tr> </thead> <tbody> <tr> <td>123456</td> <td>pese_respeito</td> <td>Sin informe</td> <td>No tiene informe</td> <td>Sin Formalizacion</td> <td>Sin Archivo</td> <td>No ha form</td> </tr> <tr> <td>123456789</td> <td>jane garcia</td> <td>Sin informe</td> <td>No tiene informe</td> <td>8W6kn4ZhLg</td> <td>Aceptada</td> <td>Asignar</td> </tr> <tr> <td>12345678</td> <td>juan perez</td> <td>Informe Final</td> <td>Evaluar</td> <td>GgMq61shPz</td> <td>Aceptada</td> <td>Asignar</td> </tr> </tbody> </table>	Cedula	Nombres y Apellidos	Descargar Informe	Evaluar Informe	Descargar Formalizacion	Evaluar Formalizacion	Tutor	123456	pese_respeito	Sin informe	No tiene informe	Sin Formalizacion	Sin Archivo	No ha form	123456789	jane garcia	Sin informe	No tiene informe	8W6kn4ZhLg	Aceptada	Asignar	12345678	juan perez	Informe Final	Evaluar	GgMq61shPz	Aceptada	Asignar
	Cedula	Nombres y Apellidos	Descargar Informe	Evaluar Informe	Descargar Formalizacion	Evaluar Formalizacion	Tutor																						
	123456	pese_respeito	Sin informe	No tiene informe	Sin Formalizacion	Sin Archivo	No ha form																						
	123456789	jane garcia	Sin informe	No tiene informe	8W6kn4ZhLg	Aceptada	Asignar																						
12345678	juan perez	Informe Final	Evaluar	GgMq61shPz	Aceptada	Asignar																							
<div style="border: 1px solid #ccc; padding: 10px;"> <p>ADMIN / PASANTE / 754 /</p> <h2 style="text-align: center;">Pasante - Tutor Academico</h2> <h3 style="text-align: center;">Asignación de Tutor al estudiante jane garcia</h3> <div style="display: flex; justify-content: space-between; align-items: center;"> Tutor <div style="border: 1px solid #ccc; padding: 2px;">Antonio Silva (3) ▼</div> </div> <div style="text-align: center; margin-top: 10px;"> <input type="button" value="Asignar"/> </div> </div>																													
<table border="1"> <thead> <tr> <th>Descargar Formalizacion</th> <th>Evaluar Formalizacion</th> <th>Tutor</th> </tr> </thead> <tbody> <tr> <td>Sin Formalizacion</td> <td>Sin Archivo</td> <td>No ha formalizado</td> </tr> <tr> <td>8W6kn4ZhLg</td> <td>Aceptada</td> <td>Antonio - Silva</td> </tr> <tr> <td>GgMq61shPz</td> <td>Aceptada</td> <td>Antonio - Silva</td> </tr> </tbody> </table>	Descargar Formalizacion	Evaluar Formalizacion	Tutor	Sin Formalizacion	Sin Archivo	No ha formalizado	8W6kn4ZhLg	Aceptada	Antonio - Silva	GgMq61shPz	Aceptada	Antonio - Silva																	
Descargar Formalizacion	Evaluar Formalizacion	Tutor																											
Sin Formalizacion	Sin Archivo	No ha formalizado																											
8W6kn4ZhLg	Aceptada	Antonio - Silva																											
GgMq61shPz	Aceptada	Antonio - Silva																											
Contexto	Aplicación Web																												
Usabilidad	El administrador despliega la lista de pasantes activos y aquel que aún no haya formalizado, no podrá asignarle un tutor académico, en caso contrario, de haber formalizado el pasante la pasantía, el administrador podrá asignarle un tutor.																												
Fuerza	El usuario debe tener conocimiento de los pasos a seguir para la asignación de tutor.																												
Consecuencia	El pasante tendrá un tutor académico que lo asesore durante su labor obligatoria, como su evaluación del informe final.																												

Tabla 13 Patrones de Iteración –Asignación de Tutor (Continuación Tabla 12).

Patrones Relacionados	Asignación de tutor académico, Evaluación de informe final.
----------------------------------	---

4.3.6 Prototipaje

En esta etapa del ciclo de vida del método de desarrollo *AguilUs*, surge la necesidad de llevar a cabo iteraciones, con las que los módulos, administrativo, pasante y tutor van madurando y entrelacionándose para convertirse en el sistema de Gestión de Pasantías SIGEPAS, que cumplen con todas las necesidades planteadas para cada usuario.

Es por ello que, en esta etapa se determina la plataforma de desarrollo, los prototipos ejecutables y la evaluación heurística que formaron parte de la implementación del sistema.

- ***Plataforma de desarrollo***

Según lo que dictamina el decreto 3390 Artículo 1º donde se especifica que La Administración Pública Nacional empleará prioritariamente Software Libre desarrollado con Estándares Abiertos, en sus sistemas, proyectos y servicios informáticos, la plataforma de desarrollo de SIGEPAS, utiliza un conjunto de estas tecnologías y herramientas de software libre.

- ***Preparación de la Base de Datos***

Para llevar a cabo el desarrollo del sistema, se hace necesario estudiar el modelo de datos actual del sistema COMPAS, para así rediseñar y completar su estructura como se muestra en la Figura 19, con el objetivo de cubrir las necesidades del nuevo sistema.

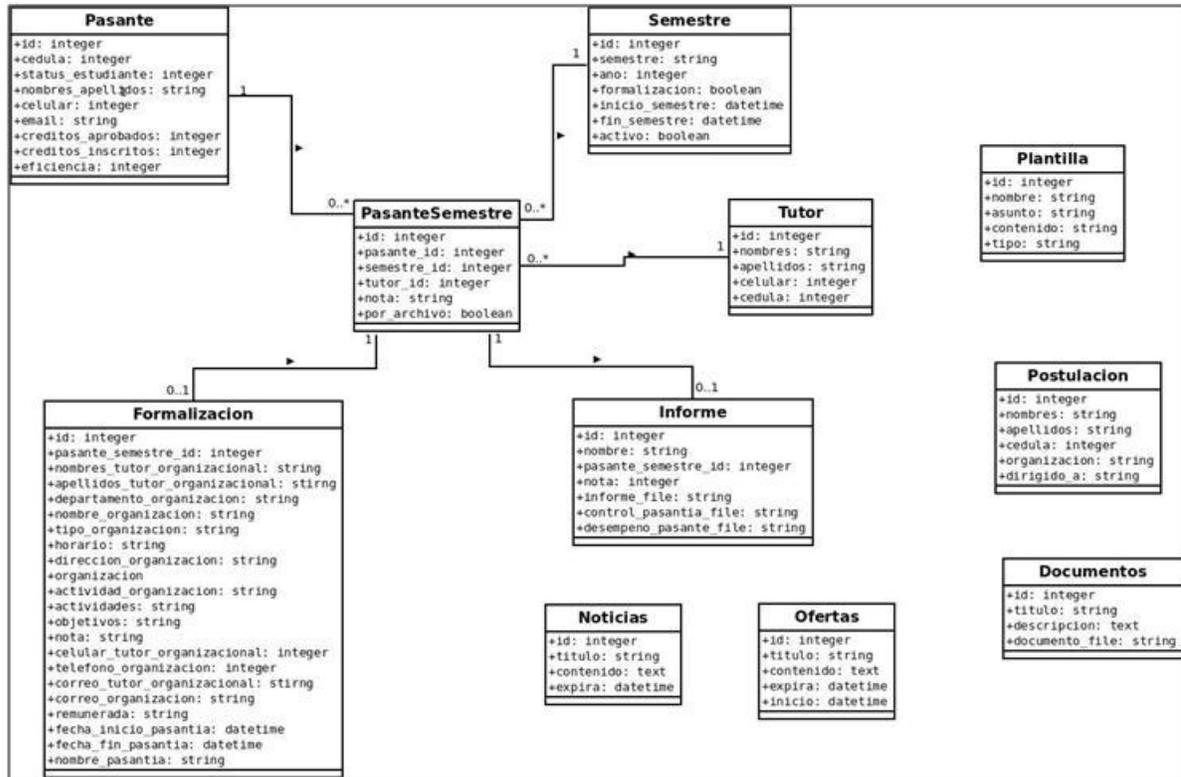


Figura 19. Diseño del Modelo de Datos para el sistema de SIGEPAS. Autores: Florville – Montes -2013.

- **Prototipo Ejecutable**

Un prototipo es una versión inicial de un sistema de software que se utiliza para demostrar los conceptos, probar las opciones de diseño y entender mejor el “problema” y su solución (SOMMERVILLE, 2000). El prototipo inicial de los módulos, administrativo, pasante y tutor, evolucionó a lo largo de su desarrollo por medio de iteraciones, las cuales realizaron una función a cumplir metas en vez de realizarlo en función a tiempo.

- **Iteración cero:**

En esta iteración la idea principal era definir los conceptos y funcionalidades actuales por parte del administrador, tutor y pasante dentro de SIGEPAS de la Facultad de Ciencias de la UCV, para que a su vez se discutieran, con los usuarios que la usan, las mejoras a realizar en cuento a nuevos mecanismos a implementar para el nuevo sistema en cuento a corrección de lectura de archivo en formato XLS para inscripción de nuevos

pasantes, envíos de correos selectivos a pasantes y tutores, plantillas de correos, entre otros.

- ***Iteración uno:***

Para esta iteración se comenzó la implementación de un diseño general para el módulo administrativo, módulo pasante y tutor, el cual se dividiría en tres sectores o espacios. Primero un menú principal colocado en la parte superior, el cual se repetiría constantemente en todas las vistas. Segundo, un espacio llamado contenido, donde se mostraran la información requerida por el usuario y por último, un pie de página donde se especificara los derechos de autor del sistema y a quien pertenece como se muestra en la Figura 20.



Figura 20. Prototipo de Interfaz. Iteración Cero. Autores: Florville – Montes -2013.

- ***Iteración dos:***

En esta iteración se enfocaron los esfuerzos en definir los ítems del menú principal, para el módulo pasante, tutor y administrador, a su vez en su despliegue de información y contenido de cada uno de ellos, en el cual se mostrará en todas las vistas del sistema. Para el contenido del menú para el modelo de pasante y tutor se tomaron en cuenta los mismo para la página principal, los cuales son: *Inicio*; el cual es un enlace a la vista principal del sistema o mejor dicho, página de bienvenida, *Documentos*; aquí se

mostrarán los distintos documentos descargables por el usuario, los cuales pueden ser normativas de las pasantías dentro de la Facultad de Ciencias, formatos de informes, entre otros documentos relevantes. Otro ítem a mostrar en el menú principal es: *Noticias*, el cual llevará al usuario a las noticias publicadas por el administrador, *Ofertas*, *Ayudas* y *Contactos*, mostrarán las ofertas publicadas por el administrador de las distintas empresas o instituciones que soliciten de un pasante, las ayudas que necesita el pasante para formalizar o generar una carta de postulación y usabilidad del sistema y una lista de contactos con correo electrónico, tanto el del administrador, subcoordinador y profesores a cargo que sirvan de ayuda a la comunidad de la Facultad de Ciencias respectivamente.

- ***Iteración tres:***

En esta iteración comienza el desarrollo de funcionalidades por parte del pasante como generar cartas de postulación, la cual no necesita del ingreso al sistema, mientras que la formalización de pasantías si la necesita, por lo que se definieron los formularios con el mínimo contenido necesario para la generación en formato PDF de ambas cartas y mecanismo de autenticación para el pasante, a través de su cédula de identidad y una clave encriptada aleatoria de acceso la cual recibirá mediante correo electrónico enviada por el administrador del sistema.

- ***Iteración cuatro:***

Para la siguiente iteración se tomaron en cuenta las sugerencias por parte del administrador en cuanto al desarrollo de un mecanismo óptimo para la publicación de noticias, ofertas, documentos y contactos que a su vez fuesen publicadas automáticamente en la página principal del sistema.

Por medio de estas discusiones se comenzó a plantear las funcionalidades que lleva a cabo el administrador del sistema, como algunas otras nuevas a desarrollar, por ejemplo; envío de correos selectivos tanto para pasantes como tutores, seleccionando

previamente una plantilla de correo, creada por el administrador, la cual indicaría el contenido el mismo con la finalidad de ser reutilizadas.

Se definieron los mecanismos de autenticación para el módulo tutor, los cuales se definieron, el correo electrónico para el nombre de usuario y una clave encriptada aleatoria de acceso la cual recibirá mediante correo electrónico enviada por el administrador del sistema.

- ***Iteración cinco:***

En esta iteración se plantea la problemática actual referente a la lecturas de archivo en formato XLS proporcionado por la División de Control de Estudios, los cuales en su gran mayoría generan excepciones, impidiendo la lectura correcta del archivo y pérdida de tiempo para el administrador. Este archivo contiene información de todos los pasantes que se inscribieron y fueron aprobados para cursar pasantías en la Escuela de Computación de la Facultad de Ciencias durante el nuevo semestre en curso.

- ***Iteración seis:***

Fue en esta iteración en la que se definió la importancia de las relaciones entre pasante y el semestre activo, es decir semestre en curso. A partir de aquí se toma en cuenta que las acciones hechas por el pasante como cargar informes, cargar las planillas de desempeño del pasante y control de pasantías, cargar la formalización, están asociadas directamente al semestre al cual están cursando, esto por el simple hecho de que si se elimina un semestre se eliminará todo tipo de archivos y documentos asociado al pasante también, salvo la información general del estudiante, tutor asignado y de notas y evaluaciones adquiridas. Gracias a esta implementación se ahorra gran cantidad de espacio en disco para la BD, ya que no contendrán archivos ni documentos de semestre anteriores que ya no sirve su información.

- ***Iteración siete:***

En esta iteración se definieron las siguientes funcionalidades: El pasante aparte de subir el informe final, debe subir dos planillas; planilla de Desempeño del Pasante y Control de Pasantías, como otra funcionalidad, El administrador debe descargar en un archivo comprimido todos los documentos de todos los pasantes de un semestre, una vez descargado los archivos, deben ser borrados del sistema. Los archivos son: formalización, informe final, planilla desempeño del pasante y planilla de control de pasantías.

- ***Iteración ocho:***

Para esta iteración se definieron los últimos detalles de la aplicación, generar una búsqueda dinámica para los pasantes a través de su cedula de identidad, como también para los tutores. A su vez, un generador de constancias global para todos los tutores, lo cual optimizaría el proceso de generación de constancias individuales por cada tutor, ya que es un requisito indispensable para cada profesor que haya sido tutor académico. Por último se discutieron los estilos definitivos de cada vista con sus colores y presentación.

De esta manera se obtiene como resultado un nuevo sistema de gestión de pasantías para la Escuela de Computación de la Facultad de Ciencias, dividido en tres módulos, denominado SIGEPAS.

Luego de haber definido cada uno de los procesos y funcionalidades para cada módulo determinado por su tipo de usuario correspondiente, a continuación, se pueden observar todos los procesos y sus funcionalidades. En Figura 21, se representa de manera gráfica los procesos y funcionalidades para el módulo administrativo ejecutados por el usuario "Administrador".

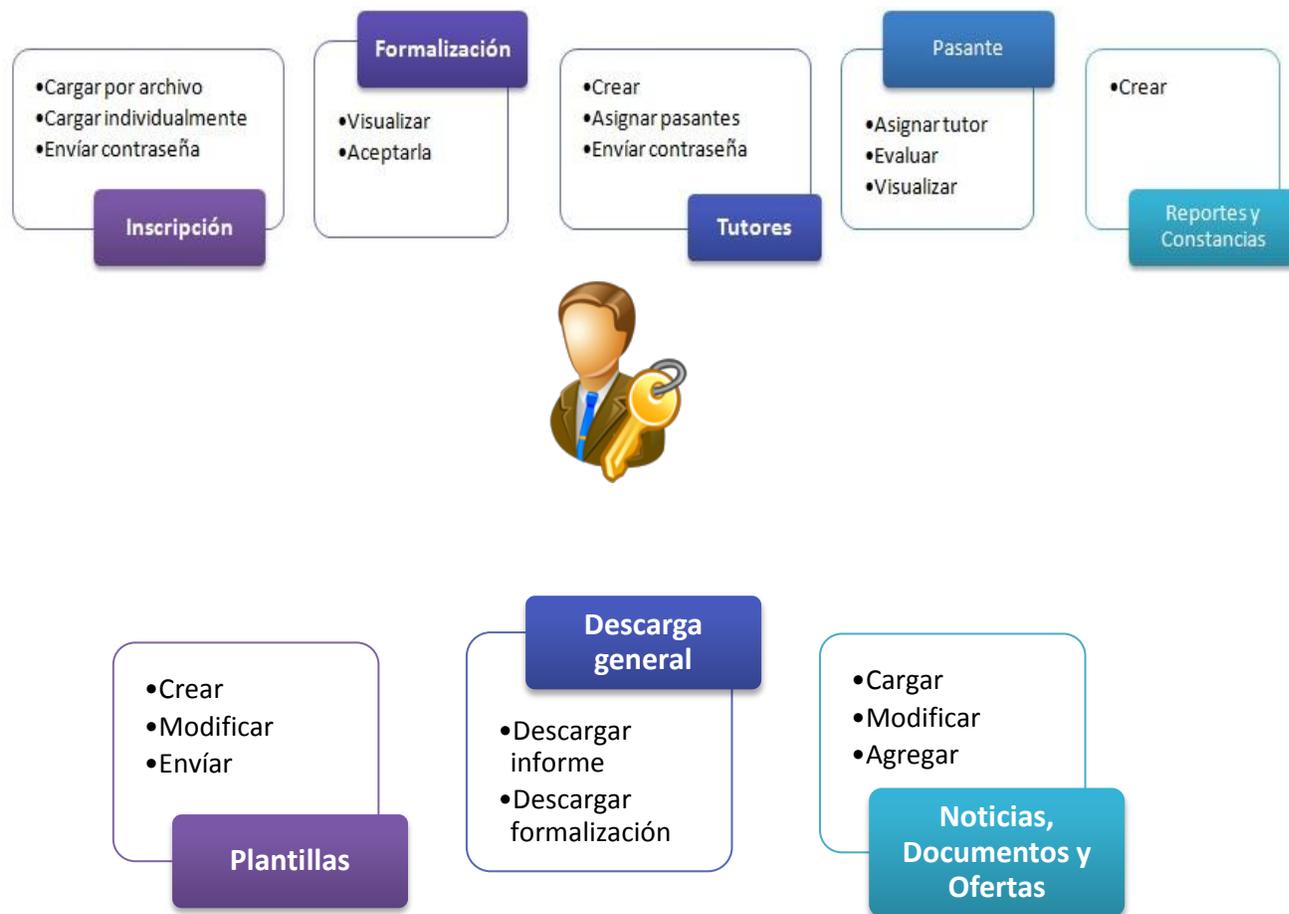


Figura 21 Representación Gráfica de los procesos y funcionalidades – Administrador. Autores: Florville – Montes -2013

En la figura 22, se puede observar una representación gráfica de los procesos y funcionalidades para el módulo tutor, ejecutados por un “Tutor Académico” dentro del nuevo sistema.



Figura 22 Representación Gráfica de los procesos y funcionalidades - Tutor Académico. Autores: Florville – Montes -2013

Por último, en la figura 23, se puede detallar todos los procesos y funcionalidades para el módulo pasante, los cuales pueden ser ejecutados por un usuario con rolo de "Pasante".



Figura 23 Representación Gráfica de los procesos y funcionalidades – Pasante. Autores: Florville – Montes -2013

Concluido con el desarrollo de todos los procesos, requerimientos, implementaciones, nuevas estructuras y estilos del nuevo sistema de gestión de pasantías, se opta por entregar la aplicación web al personal de la Comisión de Pasantías, como tambien a la comunidad docente interesada que forma parte de la Escuela de Computación, para que realicen pruebas de aceptación, rendimiento, ejecución, estilos, colores y resultados, a través de una propuesta realizada por Jakob Nielsen, llamada Evaluación Heurística.

- ***Evaluación Heurística***

Con el objetivo de detectar la mayor cantidad de problemas posibles se usó la técnica de evaluación heurística, donde los Profesores Antonio Silva Sprock y Ana Morales, como también el personal que labora en la Comisión de Pasantías evaluaron SIGEPAS, en su totalidad.

Una vez detectados los problemas, se evalúa y estudia cuál de las diez heurística propuestas por Jakob Nielsen se violan, posteriormente se califican de cero a cuatro siendo el primero el menos importante y el último siendo de carácter catastrófico. Las Heurísticas se describen en la Tabla 14.

Tabla 14. Heurísticas propuestas por Jakob Nielsen.

Prueba heurística	Descripción
H1	Visibilidad del estado del sistema. El sistema debe siempre mantener a los usuarios informados del estado del sistema, con una realimentación apropiada y en un tiempo razonable.

Tabla 15 Heurísticas propuestas por Jakob Nielsen (Continuación Tabla 14).

Prueba heurística	Descripción
H2	Utilizar el lenguaje de los usuarios. El sistema debe hablar el lenguaje de los usuarios, con las palabras, las frases y los conceptos familiares, en lugar de que los términos estén orientados al sistema. Utilizar convenciones del mundo real, haciendo que la información aparezca en un orden natural y lógico.
H3	Control y libertad para el usuario. Los usuarios eligen a veces funciones del sistema por error y necesitan a menudo una salida de emergencia claramente marcada, esto es, salir del estado indeseado sin tener que pasar por un diálogo extendido. Es importante disponer de deshacer y reparar.
H4	Consistencia y estándares. Los usuarios no deben tener que preguntarse si las diversas palabras, situaciones, o acciones significan la misma cosa. En general siga las normas y convenciones de la plataforma sobre la que está implementando el sistema.
H5	Prevención de errores. Es importante prevenir la aparición de errores que mejor que generar buenos mensajes de error.
H6	Minimizar la carga de la memoria del usuario. El usuario no debería tener que recordar la información de una parte del diálogo a la otra. Es mejor mantener objetos, acciones, y las opciones visibles que memorizar.
H7	Flexibilidad y eficiencia de uso. Las instrucciones para el uso del sistema deben ser visibles o fácilmente accesibles siempre que se necesiten. Los aceleradores no vistos por el usuario principiante, mejoran la interacción para el usuario experto de tal manera que el sistema puede servir para usuarios inexpertos y experimentados. Es importante que el sistema permita personalizar acciones frecuentes.

Tabla 16 Heurísticas propuestas por Jakob Nielsen (Continuación Tabla 15).

Prueba heurística	Descripción
H8	Los diálogos estéticos y diseño minimalista. No deben contener la información que sea inaplicable o se necesite raramente. Cada unidad adicional de la información en un diálogo compite con las unidades relevantes de la información y disminuye su visibilidad relativa.
H9	Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores. Que los mensajes de error se deben expresar en un lenguaje claro, se debe indicar exactamente el problema, y deben ser constructivos.
H10	Ayuda y documentación. Aunque es mejor si el sistema se puede usar sin documentación, puede ser necesario disponer de ayuda y documentación. Ésta tiene que ser fácil de buscar, centrada en las tareas del usuario, tener información de las etapas a realizar y que no sea muy extensa.

En la Tabla 17, se muestran los problemas encontrados en la evaluación heurística, la cual se les aplicó a dos personas miembros de la Comisión de Pasantías. De las problemáticas planteadas, se tomaron en cuenta las siguientes soluciones:

Tabla 17. Evaluación heurística.

N°	Problema	Heurística	Valoración	Solución
1	Colocar un método de descarga por lotes de constancias individuales y generales por tutor, por parte del administrador	H7	2	Se colocaron dos botones de descarga en la barra de menú de la vista de tutores para el administrador.
2	La asignación de tutor debe realizarse en un sólo paso, no en dos.	H3	2	En la lista de pasantes se agregó una nueva columna de asignación, la cual por cada

Tabla 18 Evaluación Heurística (Continuación Tabla 17).

N°	Problema	Heurística	Valoración	Solución
2		H3	2	pasante tiene un enlace que al momento de darle "click", si está "Aceptada", pasará a "No Aceptada", de igual forma lo contrario.
3	Debe existir una forma en la que en todas las vistas del administrador, se deba poder ver el nombre del semestre activo	H10	3	Se ubicó el nombre del semestre para todas las vistas del administrador en la parte superior derecha en letras grandes.
4	La evaluación del pasante debe hacerse en un solo paso, no es dos	H10	4	De igual forma que la asignación de tutor, se tomaron los mecanismos para evaluación del pasante.
5	La imagen que se encuentra como cabecera de la página principal, ya no se utiliza, buscar y modificar la nueva imagen para ser ubicada en la página principal como en todas las constancias y reportes.	H2	3	Y encontrada la imagen, se ubicó en la página principal de la página principal como en todas las constancias y reportes que genere la aplicación.
6	El campo apellido dentro de la lista de pasantes se encuentra vacío para todos los pasantes	H5	2	Se eliminó la columna de nombre y la columna apellidos en la lista de pasantes y se sustituyó por "Nombres y Apellidos".

Tabla 19 Evaluación Heurística (Continuación Tabla 18).

N°	Problema	Heurística	Valoración	Solución
7	La cédula del pasante no se ve reflejada por ningún lado en la lista de pasantes.	H10	3	Se agregó la columna "Cédula" como primera columna de la lista de pasantes.
8	El nombre del enlace para descargar la formalización del pasante en la lista de pasantes debe decir el código de la formalización	H3	3	Al momento de generación en formato PDF de la formalización del pasante se genera un código el cual es sustituido por el nombre del enlace.
9	El tutor tiene que poder ver la formalización del estudiante, para tener conocimiento de la labor que se encuentra realizando	H4	4	Se fijó una función dentro de la información del pasante en la vista del tutor académico, para visualizar la formalización en PDF.
10	Las fechas que se encuentran en las planillas, reportes y constancias están en ingles	H2	3	Se corrigieron los formatos de fecha para obtener el idioma adecuado.
11	Al momento de enviar una nueva clave para el pasante o tutor no hay ningún tipo de mensaje confirmación	H1	4	Se colocó un mensaje de confirmación al enviar un correo y clave, el cual debe aceptar para que se envíe, de lo contrario no se enviará.

Tabla 20 Evaluación Heurística (Continuación Tabla 19).

N°	Problema	Heurística	Valoración	Solución
12	El contenido del mensaje de los correos para envío de claves y cualquier información es muy simple.	H10	3	Se implementaron la creación de plantillas de correo para que el mismo usuario administrador colocase el contenido del correo.
13	Cada correo enviado por parte del sistema, viene emitido por "jcflorville@gmail.com"	H10	4	Para poder editar esta información es necesario la configuración de envíos de correo electrónico, proporcionando una nueva cuenta de correo, su clave y el proveedor de correo que desee, la ruta para editar este archivo es: "app/config/production.rb"
14	El pasante no puede modificar su formalización	H6	3	En la vista del pasante se le agregó una función para poder modificar su formalización, de ser así se generará un nuevo código.
15	Poder ingresar información del pasante en la creación de una plantilla de correo de manera intuitiva.	H4	4	Se definió una nomenclatura para ingresar datos del pasante, los cuales deben estar dentro de llaves Ej. {pasante.nombres_apellidos}.

4.3.7 Entrega

Una vez solucionadas las problemáticas que surgieron de la evaluación heurística se considera que el Sistema de Gestión de Pasantías está listo para ser presentado a los usuarios y se le apliquen pruebas de aceptación.

- ***Prueba de Aceptación***

Las pruebas de aceptación permiten conocer la opinión general de los usuarios (Acosta, 2010). Para SIGEPAS se realizó una encuesta con siete (7) preguntas, la cual se le aplicó a diez (10) personas que laboran en la COMPAS, como también a las personas que forman parte de la comunidad universitaria. En la Tabla 21 se muestra la encuesta empleada.

Tabla 21. Encuesta para el Sistema de Gestión de Pasantías.

N°	Pregunta	1	2	3	4	5	Observaciones
1	La generación de carta de postulación queda clara a simple vista.						
2	El formato de presentar las noticias, ofertas y documentos es agradable a la vista.						
3	Módulo Administrativo previene errores.						
4	Se hace fácil poder evaluar al pasante (Módulo Administrativo y Módulo Tutor).						
5	Se hizo fácil cumplir con la tarea asignada.						
6	Cumple con sus expectativas.						
7	Los reportes y constancias generadas por el sistema cumplen satisfactoriamente con sus necesidades.						

En la Figura 24. Resultados de las pruebas de Aceptación se muestran los resultados de las pruebas de aceptación realizadas al personal de la COMPAS de la Escuela de Computación de la Facultad de Ciencias, siendo el valor cinco (5) la puntuación más alta y siendo uno (1) la puntuación más baja.

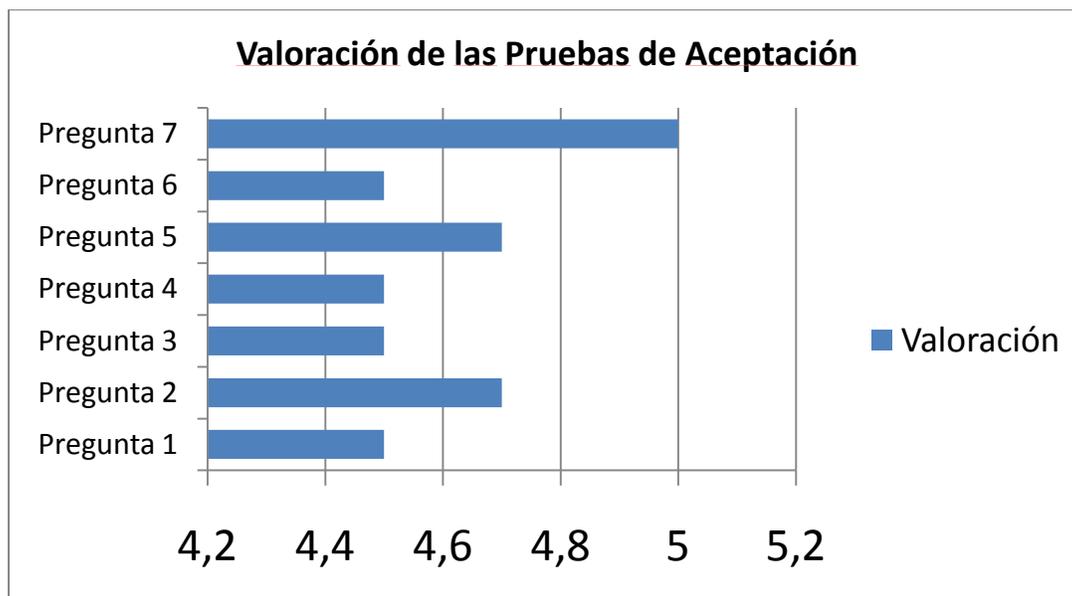


Figura 24. Resultados de las pruebas de Aceptación. Autores: Florville – Montes -2013.

Tomando en cuenta que los resultados arrojados en las pruebas de aceptación para cada uno de los problemas planteados, se puede concluir que se cumplen con las expectativas de los miembros de la COMPAS con respecto al sistema propuesto en este Trabajo Especial de Grado.

- ***Escenario de Uso***

Un escenario de uso es una descripción parcial y concreta del comportamiento de un sistema en una determinada situación. Es una descripción parcial, porque no necesita describir todas las características de las entidades involucradas, sólo se describe aquello que está relacionado con un comportamiento particular del sistema analizado.

Los escenarios que se describirán en esta investigación son:

- Formalización de la pasantía.
- Asignación del tutor académico para el pasante.

- Evaluación del informe final del pasante.

A continuación se describe cada una de las tareas antes mencionadas.

Escenario: Formalización de la pasantía

Para llevar a cabo esta tarea, es necesario que el pasante cuente con una contraseña de acceso, la cual obtendrá mediante la recepción de un correo electrónico una vez haya sido registrado por el administrador en el sistema de control de pasantías. El estudiante debe ingresar a la página de la aplicación (<http://www.ciens.ucv.ve/sigepas>), luego hacer *click* en la palabra “Ingresar” del menú ubicado en la parte superior de la pantalla, a continuación se desplegará un menú de opciones en el cual deberá seleccionar el link “Pasante”; como se muestra en el Figura 25.



Figura 25. Escenario de Uso Ingresar. Autores: Florville – Montes -2013.

Al seleccionar el link “Como pasante”, el sistema le solicitará el número de cédula y contraseña para darle acceso al mismo, como se muestra en la Figura 26.

Figura 26. Ingresar como Pasante. Autores: Florville – Montes -2013.

En Figura 27, el estudiante debe hacer “click” en el botón “Formalizar Pasantía”; esta acción le mostrara el formulario de formalización.

Figura 27. Acciones del Pasante.

El pasante deberá llenar cada uno de los campos mostrados en el formulario que se observa en la Figura 28, tales como información del tutor organizacional, información de la organización e información sobre la pasantía. Una vez que todos los campos estén llenos debe hacer “click” en el botón formalizar.

Webpage Screenshot

Facultad de Ciencias - SIGEPAS Inicio Documentos Noticias Oferta Ayuda Contactos Pasante - Juan Florvil... -

Editar Formalización de Pasantía

Por favor, edite el campo que requiera la modificación.

Información del tutor Organizacional

Nombres:

Apellidos:

Correo:

Celular:

Información sobre la pasantía

Tipo de Horario:

Remunerada: Si No

Fecha Inicio de las pasantías:

Fecha Finalización de las pasantías:

Nombre de la Pasantía (Nombre del Proyecto que usted realizará durante su pasantías):

Actividades:

Objetivos:

Información de la Organización

Tipo de Organización:

Nombre:

Dirección:

Actividad:

Teléfono:

Fax:

Correo:

Departamento:

Figura 28. Formulario para Formalizar Pasantía. Autores: Florville – Montes -2013.

Luego de suministrar la información solicitada en el formulario de formalización, se mostrará una vista previa en formato PDF de dicho formulario, donde deberá hacer “click” en el botón “Generar PDF” (de acuerdo a la Figura 29), el cual el pasante deberá imprimir, escanear y subir a la página de la aplicación, debidamente firmado por el tutor

organizacional y validado con el sello húmedo de la organización en la cual llevara a cabo la pasantía.

Webpage Screenshot

Facultad de Ciencias - SIGEPAS Inicio Documentos Noticias Oferta Ayuda Contactos Pasante - Juan Florvil...



UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS



Codigo: 3MhozEsj9B Caracas, 21 de Marzo de 2013

Comprobante de Formalización

Información del tutor Organizacional

Nombres: Josimar
Apellidos: Sanchez
Correo: josimarsanchez@gmail.com
Celular: 0412-7083299

Información del la Organización

Nombre: Josi Developers
Direccion: Plaza Venezuela
Actividad: Programador
Telefono: 0212-3728406
Fax: 0212-6373537
Correo: josidevelopers@gmail.com
Departamento: Plaza Venezuela
Tipo: publica

Información del la Pasantia

Tipo de Horario: mañana
Remunerada: NO
Fecha de Inicio: 10 de Octubre de 2012
Fecha de Fin: 06 de Febrero de 2013
Nombre de la Pasantia: Gestion de Pasantias
Actividades: Actividades
Objetivos: Objetivos

Firma del Pasante Firma del Tutor Organizacional

Generar en PDF Cancelar

© SIGEPAS 2013 - UCV - Facultad de Ciencias - Escuela de Computación

http://localhost:3000/pasantes/572/formalizacion

Figura 29. Comprobante de Formalización de Pasantías. Autores: Florville – Montes -2013.

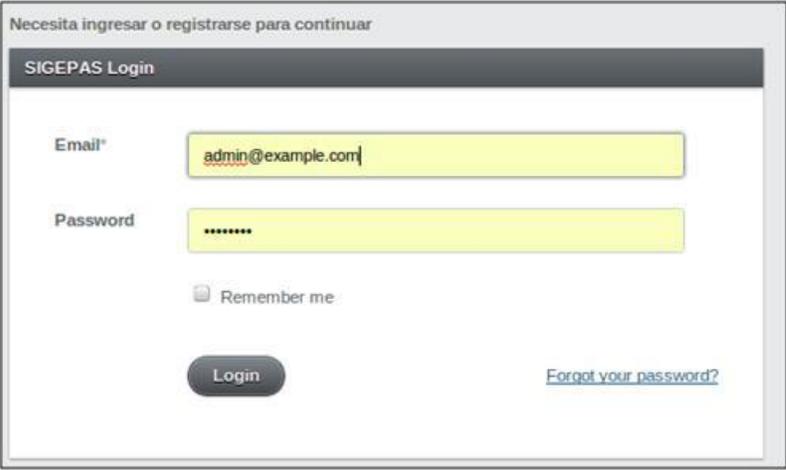
Para que el pasante pueda subir la formalización a la página de la aplicación, debe haber iniciado sesión en el sistema. Luego hacer “click” en el botón “Seleccionar archivo”, para adjuntar el archivo escaneado y luego hacer “click” en el botón “Subir”. (De acuerdo a la Figura 30).



Figura 30. Subir Formalización a la página. Autores: Florville – Montes -2013.

Escenario: Asignación del tutor académico para el pasante.

Esta actividad es ejecutada por el usuario con el rol de administrador en la aplicación, para ello es necesario que acceda a la misma mediante la URL: <http://www.ciens.ucv.ve/sigepas/admin>, y será mostrada la pantalla como en la Figura 31, donde debe iniciar sesión.



Necesita ingresar o registrarse para continuar

SIGEPAS Login

Email*

Password

Remember me

[Forgot your password?](#)

Figura 31. Ingresar como Administrador. Autores: Florville – Montes -2013.

Una vez iniciada sesión, el administrador podrá visualizar la pantalla como se muestra en la Figura 32, donde se le presenta un menú con varias opciones, allí deberá seleccionar la opción “Pasantes”, a continuación se desplegará un menú de opciones en el cual deberá seleccionar nuevamente la opción “Pasantes”.

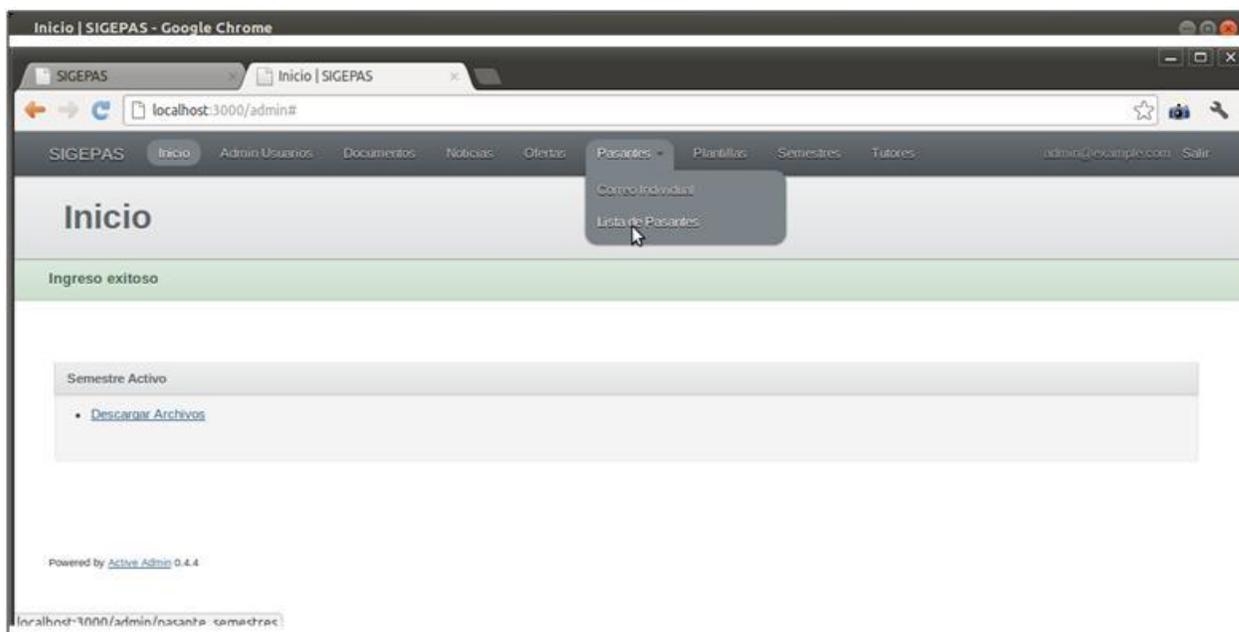


Figura 32. Menú de la vista del Administrador. Autores: Florville – Montes -2013.

Una vez seleccionada la opción “Pasantes”, tendrá acceso a una lista de pasantes inscritos en el semestre activo, tal como se observa en la Figura 33, en la séptima columna (tutor) deberá seleccionar “Asignar”.

Cedula	Nombres y Apellidos	Descargar Informe	Evaluar Informe	Descargar Formalizacion	Evaluar Formalizacion	Tutor	# Inscripciones	Nota	Reprobado	
17960697	Juan Florville	Sin informe	No tiene informe	3MozEsi9B	Aceptada	Asignar	1	No se puede calificar	-	Eliminar
17490037	Josi Sanchez	Sin informe	No tiene informe	Sin Formalizacion	Sin formalizar	No ha formalizado	1	No se puede calificar	-	Eliminar
17402860	Felix Montes	Sin informe	No tiene informe	Sin Formalizacion	Sin formalizar	No ha formalizado	1	No se puede calificar	-	Eliminar

Figura 33. Asignar Tutor por el Administrador. Autores: Florville – Montes -2013.

En la pantalla siguiente (de acuerdo la Figura 34), se le muestra la lista de tutores activos, donde se deberá seleccionar uno de ellos para el pasante previamente seleccionado.



Figura 34. Elegir Tutor para pasante por el Administrador. Autores: Florville – Montes -2013.

Una vez haya sido asignado el tutor académico para el pasante, se mostrará la pantalla con la lista de estudiantes inscritos para ese semestre, y en el séptima columna, se visualizará el nombre del tutor académico asignado como se muestra en la Figura 35.

Se modifico la informacion del Pasante.

Estudiantes Activos (3)

Cedula	Nombres y Apellidos	Descargar Informe	Evaluar Informe	Descargar Formalizacion	Evaluar Formalizacion	Tutor	# Inscripciones	Nota	Reprobado	
17960697	Juan Florville	Sin informe	No tiene informe	3MhozEs2B	Aceptada	Ana - Morales	1	No se puede calificar	-	Eliminar
17490037	José Sánchez	Sin informe	No tiene informe	Sin Formalizacion	Sin formalizar	No ha formalizado	1	No se puede calificar	-	Eliminar
17402860	Felix Montes	Sin informe	No tiene informe	Sin Formalizacion	Sin formalizar	No ha formalizado	1	No se puede calificar	-	Eliminar

Descargar: [CSV](#) [XML](#) [JSON](#)

Mostrando **todos(as) los(as) 3** Pasante Semestres

Powered by [ActiveAdmin](#) 0.4.4

Figura 35 Lista de Estudiantes Inscritos. Autores: Florville - Montes 2013

Escenario: Evaluación del informe final del pasante

Para llevar a cabo esta tarea, es necesario que el tutor cuente con una contraseña de acceso, la cual obtendrá mediante la recepción de un correo electrónico una vez haya sido registrado por el administrador en el sistema de gestión de pasantías. El tutor debe ingresar a la página de la aplicación (<http://compas.ciens.ucv.ve>), luego hacer “click” en la palabra “Ingresar” del menú ubicado en la parte superior de la pantalla, a continuación se desplegará un menú de opciones en el cual deberá seleccionar el link “Tutor”; como se muestra en la Figura 25.

Al seleccionar el link “Tutor”, el sistema le solicitará correo electrónico y contraseña para darle acceso al mismo, como se muestra en la Figura 36.

The screenshot shows the login interface for a tutor. At the top, there is a navigation bar with the text 'Facultad de Ciencias - SIGEPAS' on the left and a menu with 'Inicio', 'Documentos', 'Noticias', 'Oferta', 'Ayuda', 'Contactos', and 'Ingresar' on the right. The main heading is 'Ingresar como Tutor'. Below this, there are two input fields: 'Correo:' with the value 'jcfiorville@gmail.com' and 'Contraseña:' with masked characters '*****'. A blue 'Ingresar' button is positioned below the password field. At the bottom of the page, there is a copyright notice: '© SIGEPAS 2013 - UCV - Facultad de Ciencias - Escuela de Computación'.

Figura 36. Requisitos para ingresar como Tutor. Autores: Florville – Montes -2013.

Una vez el tutor inicie sesión, se mostrará la pantalla de la Figura 37, donde se observa la lista de pasantes asignados para dicho tutor según el semestre.

The screenshot displays the tutor's dashboard after login. The top navigation bar is identical to Figure 36, but includes a user profile icon and the text 'Prof(a) - Asilva' on the right. The main content area is titled 'Semestre(s):' and features a blue 'Constancia General' button. Below this, there is a section for 'I - 2013' which contains a table of assigned students. The table has two columns: 'Cédula' and 'Nombre y Apellidos'. One student is listed with the ID '17960697' and the name 'Juan Florville'. At the bottom, the same copyright notice as in Figure 36 is present.

Cédula	Nombre y Apellidos
17960697	Juan Florville

Figura 37. Vista una vez ingresado el Tutor. Autores: Florville – Montes -2013.

Al momento de seleccionar el pasante (haciéndole *click* en la cédula) para ver su información general, se puede observar si el pasante ha cargado el informe final, ha formalizado y ha cargado las planillas de desempeño del pasante y control de pasantías. Al tutor se le mostrará el nombre del archivo, al cual le puede hacer "*click*", para

descargarlo y visualizarlo para luego colocar la nota correspondiente, como se muestra en la Figura 38.

The screenshot shows the SIGEPAS web application interface for evaluating a student named Juan Florville. The page is titled "Pasante - Juan Florville" and is part of the "Facultad de Ciencias - SIGEPAS" system. The user is identified as "Prof(a) - Asilva".

The interface is divided into several sections:

- Formalización:** "Visualice la formalización del pasante, descargando el archivo en formato PDF." A blue button labeled "Generar en PDF" is visible.
- Informe:** "1.- Evalúe el informe final del pasante". Below this, it says "Informe_final.doc --> Informe_final.doc".
- Evaluación:** "2.- ¿De acuerdo a los criterios de evaluación, considera usted que el pasante ha...". There are two radio buttons: "Aprobado" (selected) and "Reprobado". A blue button labeled "Enviar Nota" is next to them.
- Documentos:** "Descargue la planilla de Desempeño del Pasante." Below this, it says "comprobante_563-967494.pdf --> comprobante_563-967494.pdf".
- Constancia:** "Descargue la planilla de Control de Pasantías." Below this, it says "17960697-F.pdf --> 17960697-F.pdf". A blue button labeled "Constancia" is visible.

At the bottom of the page, there is a "Volver a la lista" button and a footer: "© SIGEPAS 2013 - UCV - Facultad de Ciencias - Escuela de Computación".

Figura 38. Evaluar Pasante por Tutor. Autores: Florville – Montes -2013.

Luego de haber seleccionado la calificación correspondiente para la evaluación del informe final del pasante, se refleja dicha calificación como se puede ver en la Figura 39.



Figura 39. Mensaje una vez Evaluado el Pasante. Autores: Florville – Montes -2013.

Escenario: Generar reporte pasantes activos.

Para llevar a cabo este escenario de uso, el administrador debe ir a la vista de la lista de pasantes activos. En la parte superior derecha encontrará un botón que dice “Reporte de Pasantías”, al darle “click” se mostrará una tabla de los pasantes que se encuentran inscritos y quienes han formalizado sus pasantías como se muestra en la Figura 40.

ADMIN / PASANTE / Nuevo(a) Pasante I-2013

Reporte Pasante Activo

Estudiantes inscritos

Nro	CEDULA	NOMBRE
1	17960697	Juan Florville
2	17490037	Josi Sanchez
3	17402660	Felix Montes

Estudiantes formalizados

Nro	CEDULA	NOMBRE	ORGANIZACION	PRIVADA O PUBLICA
1	17960697	Juan Florville	Josi Developers	publica

Total inscritos: 3
 Total formalizados: 1
 Total publica: 1
 Total privada: 0

Figura 40 Reporte de Pasantías. Autores: Florville – Montes 2013.

En la Figura 41 se muestra en formato PDF el reporte completo de pasantías, el cual se destacan cuales son los estudiantes inscritos, mencionando su número de cédula y nombre. De igual manera para los estudiantes que ya han formalizados, mencionando su cédula, nombre completo, en que organización realizó las pasantías y si fue en una institución o empresa pública o privada.

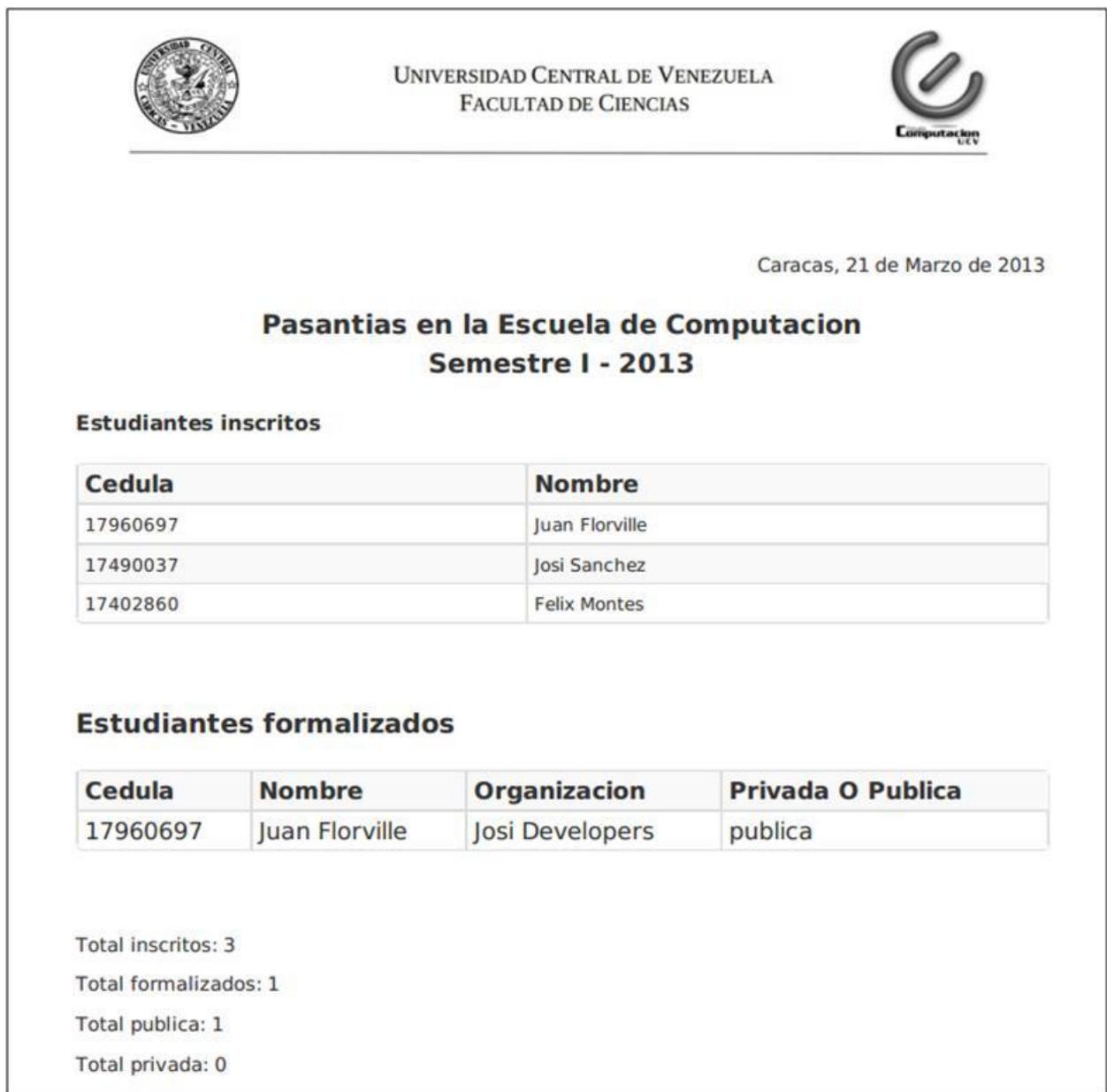


Figura 41 Reporte de Pasantías en formato PDF. Autores: Florville - Montes 2013

Escenario: Generar reporte de asignaciones.

Al igual que la generación de reporte de pasantes activos, existe un botón ubicado en la parte superior derecha de la vista de la lista de pasantes activos, que dice “Reporte de Asignaciones”, al darle “click” se mostrará una tabla de los pasantes que tienen tutor

académico asignado, además, del nombre del tutor y el correo. Como se muestra en la Figura 42 se observa la tabla en el modulo del administrador.

The screenshot shows the 'Lista Asignaciones' page in the SIGEPAS system. The page has a dark navigation bar with links like 'Inicio', 'Admin Usuarios', 'Documentos', 'Noticias', 'Ofertas', 'Pasantes', 'Planillas', 'Semestres', 'Tutores', and 'Salir'. The main header includes 'ADMIN /' and 'I-2013'. Below the header, there are two buttons: 'Crear Nuevo(a) Pasante' and 'Enviar Contraseña'. The main content area features a table with the following data:

Pasante	Correo Pasante	Tutor	Correo Tutor
17960697 Juan Florville	jcfiorville@gmail.com	Antonio Asilva	jcfiorville@gmail.com

Below the table, there are links for 'Generar en PDF' and 'Cancelar', and a footer note 'Powered by Active Admin 0.4.4'.

Figura 42 Reporte de Asignaciones. Autores: Florville - Montes 2013

En la Figura 43 se muestra la generación en formato PDF de la lista de asignaciones de los pasantes activos en el semestre en curso.

The PDF report is titled 'Pasantias en la Escuela de Computacion Semestre I - 2013'. It features the logos of the Universidad Central de Venezuela and the Facultad de Ciencias. The date is 'Caracas, 21 de Marzo de 2013'. Below the title, it says 'Lista de asignaciones' and includes a table with the following data:

Pasante	Correo Pasante	Tutor	Correo Tutor
17960697 Juan Florville	jcfiorville@gmail.com	Antonio Asilva	jcfiorville@gmail.com

Figura 43 Reporte de Asignaciones en formato PDF. Autores: Florville - Montes 2013

Escenario: Generar Constancias para Tutor.

Dentro del módulo administrativo existe una pestaña “Tutor”, aquí se muestra la lista de tutores activos. En una de las columnas de esta tabla, específicamente “Constancias”, se pueden generar las constancias, tanto individuales, generales o por lotes de cada uno de los tutores con sus respectivos pasantes asignados. Como se muestra en la Figura 44 se pueden observar la lista de tutores activos y los respectivos botones que llevan a la generación de constancias.

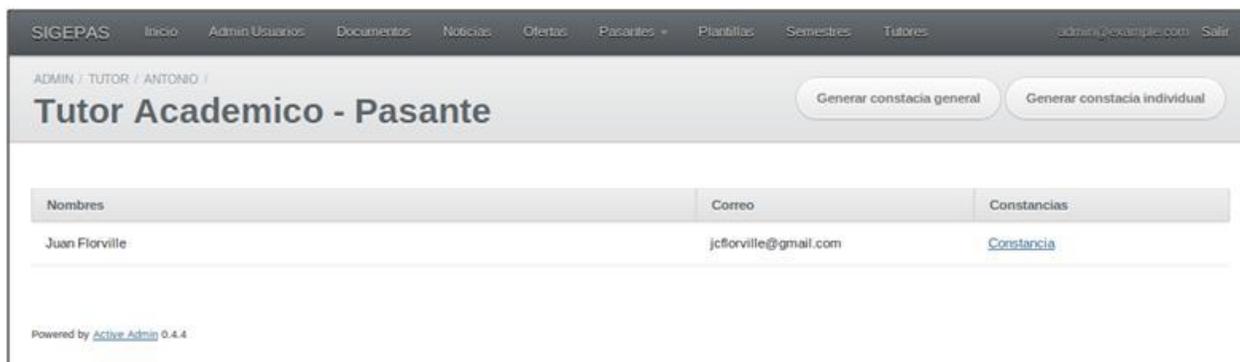


Figura 44 Lista de Tutores Activos. Autores: Florville - Montes 2013

En la Figura 45 se puede observar un ejemplo de la generación en formato PDF de la constancia individual por tutor, donde se especifica el nombre de la organización donde realizó las pasantías, como también el nombre del informe final. Vale destacar que esta generación en formato PDF cuenta con la firma y teléfonos del coordinador a cargo de la comisión de pasantías para ese momento.



UNIVERSIDAD CENTRAL DE VENEZUELA
FACULTAD DE CIENCIAS



Caracas, 21 de Marzo de 2013

A quien pueda interesar

Quien suscribe, Felix Montes, CI: 17402860, Coordinador de la Comisión de Pasantías de la Escuela de Computación de la Facultad de Ciencias de la Universidad Central de Venezuela, hace constar que la Prof(a). Antonio Asilva, realizó labores de tutor(a) Académico de Pasantías del Br. Juan Florville, durante el año 2013, quién realizó el trabajo titulado: "Gestion de Pasantias".

Juan Florville realizó sus prácticas profesionales durante 240 horas, en la organización: Josi Developers Departamento Informatica.

Sin más a que hacer referencia,

Atentamente

Prof. Felix Montes

Telf.+58-212-3728406 / +58-aasdf

fmontes86@gmail.com



Figura 45 Constancia Individual por Tutor en formato PDF. Autores: Florville - Montes 2013

En la Figura 46, se puede observar un ejemplo de la generación de constancias generales por tutor en formato PDF, incluyendo semestres anteriores.

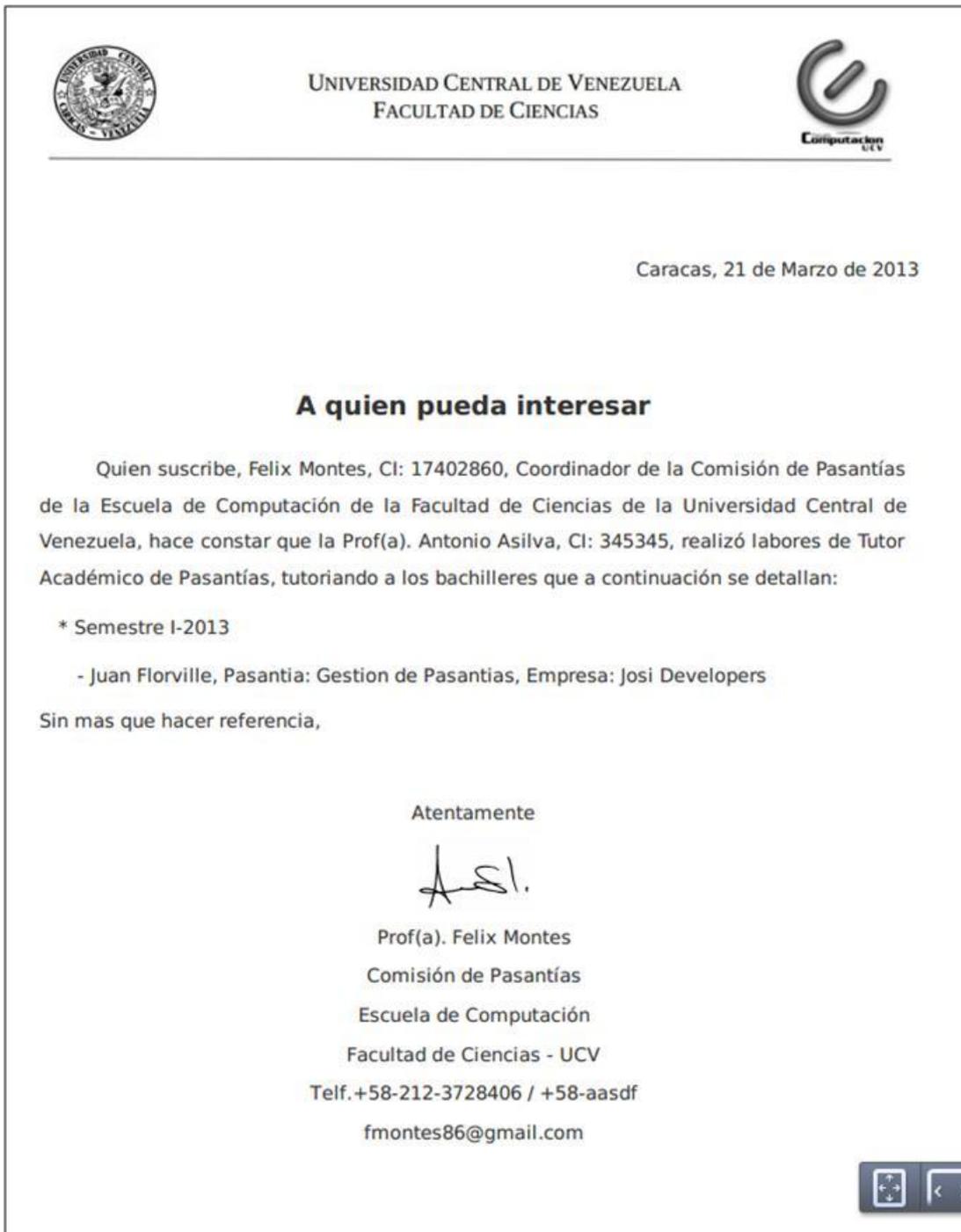


Figura 46 Constancia General por Tutor en formato PDF. Autores: Florville - Montes 2013

En la Figura 47, se puede observar la descarga en un archivo en formato “.ZIP” de todas las constancias individuales por cada tutor para cada pasante que le fue asignado.



Figura 47 Descarga en archivo .zip del lote de constancias individuales por tutor. Autores: Florville - Montes 2013

Se pudo demostrar a través del escenario de uso, las funcionalidades más importantes de las cuales se dota este nuevo sistema para administrar, controlar y supervisar la información a la comunidad de la Escuela de Computación, en su proceso de pasantías. Durante el desarrollo del sistema surgieron innumerables variables que llevaron a crear y reorganizar la estructura de la Base de Datos y sus relaciones entre la data, de tal manera, se obtuvo mayor confiabilidad de la información del pasante y tutor académico referente al semestre en curso, de igual manera, surgió la necesidad de la creación de una interfaz usable para cada uno de los perfiles, que diera a entender todas las funcionalidades y su ubicación.

Es importante destacar que cada uno de los requerimientos propuestos como objetivos específicos, como también, algunos otros propuestos por el personal administrativo de la Comisión de Pasantías, fueron desarrollados y aceptados con éxito por todo su personal, dándole una aceptación máxima para su liberación y puesta en producción.

Conclusiones

El objetivo general de este Trabajo Especial de Grado se cumplió, dado que se desarrolló un módulo administrativo que permita controlar, asesorar y evaluar al pasante, además de dirigir y supervisar todas las actividades referentes a las pasantías dentro de la Escuela de Computación de la Facultad de Ciencias para estudiantes y profesores. A su vez, se desarrollo un módulo pasante, el cual permite al estudiante poder formalizar su pasantía, cargar el informe final y cargar ambas planillas; Desempeño del Pasante y Control de Pasantía, para que el tutor académico asignado pueda descargar y visualizar dichos archivos y finalmente evaluar al pasante, por esta razón se desarrollo un módulo tutor, con el fin de generar facilidad al momento de revisión y evaluación al pasante, el cual, además tiene como función poder visualizar la formalización y generar reportes y constancias individuales por pasante como por semestre.

Para el desarrollo de este nuevo Sistema de Gestión de Pasantías (SIGEPAS) se utilizó el método de desarrollo *AgilUs*, el cual mantiene una estrecha relación con el usuario, haciendo uso de los artefactos propuestos en cada una de las etapas de la implementación. Para el desarrollo de SIGEPAS se utilizaron patrones de iteración, los cuales son muy importantes, ya que describen los pasos que se llevaron a cabo en su implementación para obtener un producto final, además, se utilizó el mecanismo de tormentas de ideas, prototipos en papel, entre otros, que ayudaron a concretar la estructura y planificación de todas las funcionalidades del sistema.

Las tecnologías estudiadas permitieron crear interfaces de usuario usables e intuitivas para todos los perfiles de usuarios, cumpliendo de esta manera con los objetivos del presente trabajo.

En cuanto al desarrollo del módulo administrativo se empleó la herramienta *Active_admin*, la cual es una librería de *Ruby* distribuida por *RubyGems*, que permite la generación de una interfaz anexa a la de la aplicación principal, teniendo como objetivo poder administrar todo el contenido y funcionalidades de manera más dinámica y eficaz, además, brindando apoyo del manejo de la información. Tiene como objetivo hacer que las tareas por parte del administrador sean fáciles y seguras de implementar. Hay que tomar en cuenta que la distribución del contenido dentro de la interfaz no es manejable, por lo que se decidió no

agregar más funcionalidades, evitando diferencias e inconsistencias significativas dentro de la aplicación. Es por esto que durante su desarrollo se encontraron limitantes en cuanto a usabilidad, ya que la herramienta no proporcionó mecanismos ágiles e intuitivos para la implementación de búsquedas dinámicas para las listas de pasantes y tutores, mejor distribución de los botones de acciones para generación de reportes y constancias en formato PDF, como también la implementación de vistas alternas que ayuden al manejo del contenido de los datos.

Mediante el estudio de diversas herramientas de aplicación web y tomando en cuenta todos los requerimientos funcionales y no funcionales del sistema, se aplicaron en conjunto para su implementación y desarrollo del módulo de pasante y de tutor, permitiendo así integrar las funcionalidades de evaluación, cargar archivos, descargar archivos y generación de cartas y constancias en formato PDF, sin alterar la integridad y persistencia de los datos. Esto trae como beneficio que el personal encargado de controlar, evaluar y administrar la información no requiera de tantas herramientas innecesarias para ejercer su labor.

Finalmente, esta investigación forma parte de un proyecto de gran envergadura, que no solamente provee a la comunidad de la Escuela de computación, sino también a la Universidad Central de Venezuela, ofreciendo múltiples funcionalidades que hacen sus tareas más sencillas, además, que brindan a los estudiantes y profesores la posibilidad de retribuirle a la Facultad de Ciencias lo que ésta le enseñó durante sus años de formación como Licenciado en Computación.

Trabajos Futuros

Se recomienda, para enriquecer la aplicación y aumentar sus funcionalidades, los siguientes aspectos:

- Se recomienda crear un modulo adicional dentro del sistema de gestión de pasantías que permita a las empresas u organismos hacer uso de la metodología aplicada, para publicar información sobre la captación de estudiantes en carácter de pasante.
- La metodología descrita en esta investigación puede ser aplicada en forma segura por toda la comunidad estudiantil de las diferentes facultades que conforma la UCV.
- Se sugiere a futuros investigadores de la escuela de computación de la implementación de un formato de edición de texto para la creación de plantilla de correo, el cual pueda ofrecer múltiples opciones.
- Se recomienda la búsqueda de un mejor formato para la creación de las plantillas de correo, donde se especifique al momento de crear la plantilla, el nombre del pasante su cedula, en que organización presta sus servicios de pasantes, nombre del informe final, entre otros. De igual manera para la creación de plantillas de correo para los tutores.

Bibliografía

- Acosta, A. E. (2010). *AgilUs: Construcción ágil de la Usabilidad*. Caracas.
- Acuña, K. B. (2009). *Selección de metodologías de desarrollo web en la facultad de informática en la Universidad de Cienfuegos*. CUBA.
- Alegsá. (26 de Julio de 1998). *Alegsá*. Obtenido de <http://www.alegsa.com.ar/Dic/modelo%20de%20datos%20de%20base%20de%20datos.php>
- Alvarez, M. A. (29 de Junio de 2008). *Desarrollo WEB*. Obtenido de <http://www.desarrolloweb.com/articulos/introduccion-css3.html>
- Alvarez, M. A. (19 de Junio de 2009). *Desarrollo web*. Obtenido de <http://www.desarrolloweb.com/articulos/introduccion-jquery.html>
- Alvarez, M. A. (2009). *Desarrollo Web*. Obtenido de <http://www.desarrolloweb.com/articulos/25.php>
- Andina, F. U. (s.f.). *Fundación Universitaria del Área Andina* . Obtenido de http://www.areandina.edu.co/portal/index.php?option=com_content&view=article&id=486&Itemid=356&lang=es
- Arnedo, L. (13 de Julio de 2006). *Proyecto Agiles*. Obtenido de <http://www.proyectosagiles.org/que-es-scrum>
- Bautista, J. M. (2008). *Programación extrema*.
- BAUTISTA, J. M. (2008). *Programación extrema*.
- Cesar. (28 de Enero de 2010). *Tictech Consulting*. Obtenido de <http://www.tictech.es/es/blog/2010/01/28/scrum-metodologia-%E2%80%9Cagil%E2%80%9D-para-tus-proyectos/>
- Chacón, J. C. (2006). *APLICACIÓN DE LA METODOLOGÍA RUP PARA EL*.

-
- Consejo Universitario. (1973). Normas para la aplicación del sistema de créditos en la UCV. Caracas.
 - Figuerola, N. (28 de Enero de 2008). *Scribd.com*. Obtenido de <http://www.tictech.es/es/blog/2010/01/28/scrum-metodologia-%E2%80%9Cagil%E2%80%9D-para-tus-proyectos/>
 - Fundación Universitaria del Área Andina. (2002). *Fundación Universitaria del Área Andina*. Obtenido de http://www.areandina.edu.co/portal/index.php?option=com_content&view=article&id=486&Itemid=356&lang=es
 - Karen. (1 de Noviembre de 2008). Obtenido de <http://kasyles.blogspot.com/>
 - Mazt. (2003). *An Interview with the Creator of Ruby*.
 - More, R. (Agosto de 2005). Obtenido de <http://sistemas-site.blogspot.com/>
 - Nerfites. (19 de Octubre de 2009). *Autorneto*. Obtenido de <http://autorneto.com/tecnologia/programacion/javascript-asincrono/>
 - Oracle. (2000). *MYSQL*. Obtenido de <http://dev.mysql.com/doc/refman/5.0/es/introduction.html>
 - PCMAG. (2003). Obtenido de http://www.pcmag.com/encyclopedia_term/0,2542,t=Web+client&i=54284,00.asp
 - PCMAG. (s.f.). *PCMAG*. Obtenido de http://www.pcmag.com/encyclopedia_term/0,2542,t=Web+client&i=54284,00.asp
 - Piñol, C. M. (2003). *Desarrollo de aplicaciones web*. UOC.
 - Rafael, J. (22 de Junio de 2008). *Neleste 2.0*. Obtenido de <http://www.neleste.com/acerca-de/>

- Rodríguez, H. A. (24 de febrero de 2006). *MailxMail*. Obtenido de <http://www.mailxmail.com/curso-procesamiento-datos-oracle/sistema-manejador-base-datos>
- Santos, R. D. (13 de Octubre de 2006). *Maestro del Web*. Obtenido de <http://www.maestrosdelweb.com/editorial/rubyonrails/>
- Systems, A. (2010). *Flex*. Obtenido de <http://www.flex.org>
- Tanenbaum, A. (2002). *Distributed Systems*. Prentice Hall.
- Valdés, D. P. (27 de Octubre de 2007). *Maestro del Web*. Obtenido de <http://www.maestrosdelweb.com/principiantes/%C2%BFque-son-las-bases-de-datos/>
- Valencia. (10 de Abril de 2008). *Hooping*. Obtenido de <http://www.hooping.net/glossary/aplicaciones-web-146.aspx>