



Universidad de las Ciencias Informáticas
Facultad 4

Módulo Solicitud de servicios para el Sistema de Gestión de Calidad del centro FORTES

Trabajo de Diploma para optar por el título de Ingeniero en Ciencias
Informáticas

Autora

Annia Verdecia Boza

Tutor

Ing. Roberto Carlos Aballe Ochoa

Cotutora

Ing. Celia María Soularý Reyes

La Habana, Junio 2012

Declaración de autoría

Declaro ser la única autora del trabajo “Módulo Solicitud de servicios para el Sistema de Gestión de Calidad del centro FORTES.” y autorizo a la Facultad 4 de la Universidad de las Ciencias Informáticas a hacer uso del mismo, con carácter exclusivo.

Para que así conste firmo la presente a los ___ días del mes de _____ del año _____.

Autora:

Tutor:

Annia Verdecia Boza

Ing. Roberto Carlos Aballe Ochoa



“Quiero saberlo todo. Y siempre me encuentro como antes, triste como la vida y resignado como la sabiduría.”

Giovanni Papini.

Agradecimientos

A mi tío Nenene por ser la persona que siempre me guió en los estudios y en la vida profesional, y ser mi bastón en mis días de tristeza, y aunque el tiempo no le alcanzó para ver este momento, sé que hubiera estado muy orgulloso de verme graduada.

A mi mamá, Nidia por darme la vida, por su apoyo incondicional, por luchar tanto por mi hermano y por mí y por darnos todo ese amor tan maravilloso.

A mi novio Reynaldo por aguantarme y apoyarme en los momento difíciles.

A Greisy Gálvez, que fue en su momento el alma del Grupo de Calidad y dio siempre su apoyo incondicionalmente a sus estudiantes, y aunque el tiempo no le alcanzó para ver este momento y sé que hubiera estado muy orgullosa de nosotros.

A mi tutor y cotutora muchas gracias por toda su ayuda.

A todos los profesores que me dieron clases, gracias por todo lo que me enseñaron.

A Gerardo, Yamila, Sandy, Adrián y Adrián Borrego por su ayuda con Drupal.

A mis amigos por todos los momentos que hemos disfrutados.

Dedicatoria

A las personas más importantes de mi vida, a mi mamá, mi hermano, mi tío Nenene y a mí.

Resumen

Ante la necesidad que presenta el Grupo de Calidad del centro FORTES¹ de mejorar los procesos de revisión y obtención de reportes de las solicitudes de los servicios que ofrece. Surge el presente trabajo, que lleva por título “Módulo Solicitud de servicios para el Sistema de Gestión de Calidad del centro FORTES”, que tiene como objetivo principal la implementación de un módulo que dé solución de forma rápida al proceso de gestión de solicitud de servicios del centro. El análisis de soluciones homólogas, arrojó como resultado que las soluciones son privativas o no tienen las funcionalidades requeridas para llevar un control de las peticiones de los servicios que exige el cliente. En la confección del módulo se utilizó la metodología de desarrollo XP²; Visual Paradigm, como herramienta CASE³; Drupal como sistema de gestión de contenido; PostgreSQL como sistema gestor de base de datos; Apache HTTP Server como servidor web; PHP⁴, XHTML⁵ y CSS⁶ como lenguajes de programación y NetBeans como entorno de desarrollo. Se obtuvo como resultado un módulo que permite la informatización y un mayor control de las plantillas generadas en el proceso de gestión de solicitudes del Grupo de Calidad del centro FORTES.

Palabras Clave: gestión de solicitud, solicitud, solicitud de servicios.

¹ **FORTES:** Centro de Tecnologías para la Formación.

² **XP:** *Extreme Programming* o Programación Extrema.

³ **CASE:** *Computer Automated System Engineering*.

⁴ **PHP:** *Hypertext Pre-processor*.

⁵ **XHTML:** Lenguaje de Marcado de Hipertexto Extendido.

⁶ **CSS:** Hojas de Estilo en Cascada.

Índice

Introducción	10
Capítulo 1. Fundamentación teórica	15
1.1 Introducción.....	15
1.2 Solicitud.....	15
1.3 Análisis de la Plantilla de Solicitud del servicio de Prueba del Grupo de Calidad del centro FORTES	17
1.4 Gestión de solicitud	17
1.5 Gestión de solicitudes en el Grupo de Calidad del centro FORTES	19
1.6 Análisis de soluciones existentes	20
1.6.1 Soluciones a nivel internacional	21
1.6.2 Soluciones a nivel nacional	22
1.6.3 Análisis de las soluciones	23
1.7 Herramientas y tecnologías de desarrollo.....	23
1.7.1 Metodologías de desarrollo	23
1.7.2 Herramientas CASE	26
1.7.3 Sistema de Gestión de Contenido (CMS).....	28
1.7.4 Sistema Gestor de Base de Datos (SGBD)	30
1.7.5 Servidores web	30
1.7.6 Tecnologías del lado del servidor.....	31
1.7.7 Tecnologías del lado del cliente	31
1.7.8 Entornos de desarrollo integrado (IDE)	33
1.8 Conclusiones parciales.....	34
Capítulo 2. Características del sistema	35
2.1 Introducción.....	35
2.2 Descripción del sistema.....	35
2.3 Planificación	36
2.3.1 Historias de usuarios.....	37

2.3.2 Estimación de esfuerzos por historias de usuario	38
2.3.3 Plan de iteraciones.....	38
2.3.4 Plan de entregas.....	39
2.4 Diseño	40
2.4.1 Diagrama de estado del sistema.....	40
2.4.2 Tablas a incorporar en la base de datos de Drupal	42
2.4.3 Tarjetas CRC (Clases, Responsabilidades y Colaboraciones)	43
2.4.4 Arquitectura de Drupal	45
2.4.5 Patrón arquitectónico Modelo Vista Controlador (MVC) en Drupal	46
2.4.6 Patrones de diseño de Drupal.....	47
2.5 Conclusiones parciales.....	49
Capítulo 3. Codificación y prueba	50
3.1 Introducción.....	50
3.2 Codificación.....	50
3.2.1 Roles del sistema.....	50
3.2.2 Distribución física del sistema	51
3.2.4 Módulos de apoyo.....	52
3.2.5 Librerías de apoyo	53
3.2.6 Estándares de codificación	53
3.3 Prueba.....	56
3.3.1 Casos de prueba (CP)	57
3.3.2 Resultado de las pruebas.....	59
3.4 Conclusiones parciales.....	60
Conclusiones generales.....	61
Recomendaciones	62
Referencias bibliográfica.....	63
Glosario de términos.....	66
Anexos.....	68
Anexo 1 Plantilla de solicitud para el servicio de Prueba del Grupo de Calidad del centro FORTES.	68

Anexo 2 Historias de usuario.....	69
Anexo 3 Casos de prueba.	73
Anexo 4 Interfaces de usuario.	83

Introducción

Las Tecnologías de la Información y las Comunicaciones (TIC) favorecen la informatización de disímiles procesos, para lo cual ha sido necesaria la creación de soluciones informáticas que satisfagan las necesidades en los más diversos ámbitos del desarrollo de la sociedad, como la medicina, la agricultura y la educación. Pero no basta con crear un *software*, es necesario hacerlo con calidad. Uno de los pasos importantes que ha dado Cuba para involucrarse en las nuevas tendencias, ha sido la creación de la Universidad de las Ciencias Informáticas (UCI). La cual tiene la misión de formar profesionales calificados en la rama de la computación, además de producir aplicaciones y servicios informáticos a través del modelo de formación. Este modelo vincula el estudio, la investigación y la producción en el proceso de enseñanza y aprendizaje.

Con la nueva organización de la infraestructura productiva, se crean los centros de desarrollo de *software* asociado a cada una de las facultades. En la Facultad 4 radica el Centro de Tecnologías para la Formación (FORTES), su misión es desarrollar tecnologías que permitan ofrecer servicios y productos para la implementación de soluciones educativas, aplicando las TIC. (1)FORTES cuenta con un Grupo de Calidad, que tiene como responsabilidad el aseguramiento de la calidad de los proyectos del centro, además apoya en las pruebas para la liberación de productos. Entre los servicios que brinda se encuentran, las Pruebas de *software*, Auditorías, Revisiones y Asesoramiento.



Figura 1 Mapa de procesos de servicio del Grupo de Calidad de FORTES. Tomado de Grupo de Calidad del centro FORTES. Política Organizacional.

En la figura 1, se muestra el Mapa de procesos de servicios del Grupo de Calidad, el cual se explica a continuación.

Como proceso fundamental que conciernen a diferentes áreas del servicio y tienen impacto en el cliente, creando valor para este, se encuentra el servicio de Prueba de *software*, el cual se encarga de realizar diferentes tipos de pruebas a un producto mediante un equipo de revisores técnicos. En el proceso estratégico se incluyen los servicios de Revisión y Auditoría, que evalúan el proceso de desarrollo de un proyecto. Mientras que el servicio de Pruebas Exploratorias realiza un reporte de errores sobre un producto realizado por un revisor técnico en particular en un tiempo determinado y con una misión a cumplir. Este último aún no se aplica en FORTES. El proceso de soporte, incluye el servicio de Asesoramiento, el cual se encarga de apoyar la realización de los procesos fundamentales, en cuanto a elementos organizativos para el proceso de prueba, capacitación para la elaboración de documentos y utilización de herramientas de automatización y gestión de pruebas.

Para lograr una correcta ejecución de cualquiera de los servicios, se necesita, como primer paso, realizar la solicitud del servicio. La comprensión de lo que el proyecto solicitante desea facilita la correcta

elaboración de una solicitud y una buena planificación del servicio. Además, la comunicación entre las partes permite que el resultado beneficie y satisfaga las necesidades del cliente.

El procedimiento de solicitud de un servicio del Grupo de Calidad se realiza de la siguiente forma: un estudiante o profesor del centro se dirige a la dirección del Grupo de Calidad solicitando un servicio. Se le entrega una planilla en formato digital, la cual es enviada, después de ser llenada, mediante correo electrónico a la dirección del proyecto. El líder del proyecto es el encargado de recibir las solicitudes en su correo, las revisa y posteriormente envía al usuario solicitante la confirmación de aceptación o rechazo de su solicitud. Pero este proceso presenta diversas dificultades. Debido a que el uso del correo personal para el envío y recepción de solicitudes del líder del Grupo de Calidad provoca que: las solicitudes puedan sobrecargar el correo, y por ende, perderse, llegar con retraso o no llegar y en otro caso borrarse por error. El envío de respuesta a los clientes sobre el estado de su solicitud se vuelve engorroso cuando el número de peticiones recibidas es alto. La recopilación de datos de las plantillas de solicitud se vuelve complejo para la revisión, obtención de reportes y planificación de servicios, ya que es un trabajo manual que provoca lentitud en estas actividades cuando se tiene muchas solicitudes.

Por tanto, el **problema científico** queda formulado de la siguiente forma: ¿Cómo mejorar el proceso de solicitud de servicios del Grupo de Calidad del centro FORTES?

El **objeto de estudio** radica en la gestión de solicitud de servicios. Por lo que el presente trabajo centra su **campo de acción** en la gestión de solicitud de los servicios de calidad de *software*.

Se plantea como **objetivo general**, implementar un módulo que informatice el proceso de solicitud de servicios del Grupo de Calidad del centro FORTES.

Para cumplir con el objetivo general planteado, se han derivado los siguientes **objetivos específicos**:

- ✓ Analizar sistemas informáticos para la gestión de solicitud de servicios y sus conceptos asociados, así como las herramientas a emplear para su desarrollo.
- ✓ Describir la propuesta del sistema.
- ✓ Implementar y probar un módulo que sirva de apoyo a la gestión de solicitud de los servicios.

Para lo cual se plantea la siguiente **idea a defender**: el Módulo Solicitud de servicios del Sistema de Gestión de Calidad del centro FORTES, informatizará el proceso de solicitudes de servicios que ofrece el centro.

Se define las siguientes **tareas de investigación**:

- ✓ Revisión y análisis crítico de la documentación relacionada con la gestión de las solicitudes de los servicios.
- ✓ Análisis crítico de sistemas de gestión de solicitudes.
- ✓ Estudio y definición de las metodologías, tecnologías y herramientas para el desarrollo de la solución informática.
- ✓ Análisis y descripción de las funcionalidades y características que tendrá la aplicación.
- ✓ Realización del análisis y diseño utilizando la metodología definida.
- ✓ Realización de la implementación y prueba al producto para la validación de la solución.

Para el presente trabajo investigativo se tuvo en cuenta varios **métodos de investigación**. A continuación se mencionan los métodos utilizados y la forma en que se manifiestan.

Métodos teóricos

Análisis – síntesis: este método facilitó analizar las teorías, documentos y bibliografías en general, permitiendo la extracción de los elementos más importantes que se relacionan con la gestión de solicitud de servicios.

Genético: este método se puede presenciar mediante el análisis y estudio realizado sobre el comportamiento, estructuración e información sobre la gestión de solicitudes de servicios en el mundo mediante diferentes sistemas y aplicaciones.

Este documento consta de tres capítulos estructurados como se describe a continuación.

El **primer capítulo** aborda los fundamentos teóricos, conceptos y conocimientos necesarios para sustentar la investigación y la plena comprensión del resto del documento. Se realiza el estudio del estado

del arte de sistemas informáticos destinados a la gestión de solicitudes de servicios, además, de las herramientas y tecnologías seleccionadas para desarrollar la herramienta.

El **segundo capítulo** plantea la propuesta del sistema. Mediante la metodología XP se desarrollan las fases de Planeación y Diseño, las cuales permiten confeccionar las historias de usuario, la estimación de las historias, planificación de las iteraciones, diagrama de estado del sistema, las nuevas tablas incorporadas a la base de datos de Drupal, las tarjetas CRC, además se especifica los patrones de diseño y la arquitectura que utiliza Drupal.

En el **tercer capítulo** se refleja las fases de Codificación y Prueba. El capítulo define los roles del sistema, el modelo físico de la aplicación, los módulos y librerías de apoyo, el estándar de codificación, los casos de prueba y descripción del resultado de las pruebas.

Capítulo 1. Fundamentación teórica

1.1 Introducción

La palabra servicio define a la actividad y consecuencia de servir (verbo que se emplea para dar nombre a la condición de alguien que está a disposición de otro para hacer lo que este exige u ordena). (2) Stanton, Etzel y Walker definen los servicios como: actividades identificables e intangibles que son el objeto principal de una transacción ideada para brindar a los clientes satisfacción de deseos o necesidades. (3)

El ofrecimiento de servicios es la manera que proporciona una empresa para relacionarse con sus clientes. Es el conjunto de actividades interrelacionadas que un suministrador proporciona para que su consumidor obtenga el producto en el momento, lugar adecuado y se asegure de un uso correcto del mismo.

Lograr la satisfacción del cliente brindando un buen servicio no es difícil si el proveedor busca estrategias y mejora las actividades para alcanzar el bienestar del usuario. Por lo que es necesario que la solicitud de un servicio permita recoger la mayor cantidad de información para aplicar el servicio de manera rápida.

En el presente capítulo se realiza un estudio de los principales conceptos relacionados con la gestión de solicitudes y de sistemas informáticos destinados a este proceso. Además, se efectúa una selección de metodologías, tecnologías y herramientas para cumplir con el objetivo propuesto.

1.2 Solicitud

La palabra solicitud es una diligencia cuidadosa o un pedido. El verbo solicitar, por su parte, hace referencia a pedir, pretender o buscar algo. (4) Por lo general, una solicitud es un documento escrito que va dirigido a un organismo público o autoridad a los que se le pide algo o se plantea una reclamación con la exposición de los motivos en los que se basan. La mayor parte de los organismos públicos disponen de impresos destinados a este fin. (5)

Estructura de una solicitud

Muchos organismos disponen de plantillas de solicitud. Según el motivo o interés, son confeccionadas para lograr recopilar información con un fin. Una plantilla de solicitud formal debe contener la estructura siguiente.

Encabezamiento: consta de los datos personales de quién firma la solicitud (nombre y apellidos, lugar, fecha de nacimiento, domicilio, teléfono y datos opcionales como profesión y centro de trabajo). (5)

Exposición de motivos: se debe describir las circunstancias y causas que motivan la solicitud, de manera sistemática y esquemática, se deben exponer todos los datos de forma argumentada y citar las disposiciones legales en que esté basada la solicitud. Se inicia con la palabra “expongo”, seguida de dos puntos y normalmente se termina con la expresión “por ello”. Si es el caso, se hará referencia a la documentación aportada con fórmulas del tipo: “lo que acredito con la documentación anexa”, seguida de la relación de documentos incorporados a la solicitud. (5)

El tratamiento personal del emisor puede ser en primera persona (yo), y el del receptor, la segunda persona del plural (os) o bien una fórmula impersonal. En ocasiones, cuando la solicitud va dirigida a una autoridad se hace uso del tratamiento que corresponde a la misma. (5)

Petición o exposición de la solicitud: se comienza con la palabra “solicito”, seguida de dos puntos. Si esta palabra va seguida de una oración con un verbo conjugado, se debe introducir antes la conjunción “que”. Por ejemplo, SOLICITO: Que sea concedido..., o bien, SOLICITO: Que se acepte esta petición de... (5)

La despedida debe indicar (5):

- ✓ Fecha: lugar, día, mes y año.
- ✓ Firma de quien presenta la solicitud.
- ✓ Nombre del organismo, institución, unidad o autoridad a quien se dirige la instancia. Estos nombres deben ir escritos todos en mayúsculas.

1.3 Análisis de la Plantilla de Solicitud del servicio de Prueba del Grupo de Calidad del centro FORTES

La solicitud para un servicio de calidad de *software* es el documento, plantilla o formulario con espacios en blanco que han de ser llenados con la finalidad de recoger los datos del cliente y del servicio en cuestión. Con el objetivo que la entidad pueda saber quién solicita, qué servicio desea y cómo desea recibirlo. El Grupo de Calidad del centro FORTES, con el propósito de organizar las solicitudes e implementar correctamente los servicios, propone una plantilla para la solicitud del servicio de Prueba de *software*, la cual se estructura de la siguiente forma.

Encabezamiento: presenta el logo del Grupo de Calidad, el nombre del documento, una breve explicación de los objetivos de la solicitud y recomendaciones que debe seguir el cliente.

Cuerpo del documento: consta de varias secciones que se encargan de recoger información útil para el servicio que se va a ofrecer, tales como: datos del proyecto, con los campos nombre del proyecto, jefe del proyecto, de la persona que solicita el servicio y el usuario del dominio UCI de estas personas; información del producto, que abarca el nombre del producto, tipo de prueba y dirección URL (Localizador de Recurso Uniforme) de la documentación y la aplicación; información de la prueba, donde se exige el artefacto, tipo de artefacto, versión, cantidad de páginas, cantidad de módulos y objetivos de la prueba; documentación que se va a utilizar para el servicio; el estado que toma la solicitud después que es revisada y fundamentación del estado de la solicitud. La plantilla del Grupo de Calidad carece de datos como la fecha en que se realizó la solicitud. La misma puede ser consultada en el [Anexo 1](#).

1.4 Gestión de solicitud

Como consecuencia de la informatización de la sociedad y la necesidad de optimizar los procesos para brindar servicios con mayor calidad y lograr una superior planificación de los mismos, se precisa mejorar la gestión de las solicitudes de los servicios, con el objetivo de satisfacer al cliente.

La gestión de solicitudes no es más que controlar la información que genera una entidad. La información, según el Diccionario de la Real Academia Española (DRAE): es comunicación o adquisición de

conocimientos que permiten ampliar o precisar los que se poseen sobre una materia determinada. Conocimientos comunicados o adquiridos. (6)

Según Idalberto Chiavenato, información, es un conjunto de datos con un significado, o sea, que reduce la incertidumbre o que aumenta el conocimiento de algo. En verdad, la información es un mensaje con significado en un determinado contexto, disponible para uso inmediato, que proporciona orientación a las acciones por el hecho de reducir el margen de incertidumbre con respecto a nuestras decisiones. (7)

Se hace necesario que la información transite por varios procesos que cumplirán un fin para una entidad o investigación y a este conjunto de actividades se le denomina gestión de la información. Gestionar la información equivale a los procesos de analizar y utilizar los datos que se ha obtenido y registrado para permitir a los especialistas de todos los niveles tomar decisiones documentadas. (8) Por lo tanto, la gestión de la información implica (8):

- ✓ Determinar la información que se precisa.
- ✓ Recoger y analizar la información.
- ✓ Registrar y recuperar la información cuando sea necesario.
- ✓ Utilizar la información.
- ✓ Divulgar la información.

Teniendo en cuenta los conceptos anteriores, se concluye que la gestión de información es un conjunto de procesos (extracción, combinación, depuración y distribución), por los cuales se controla el ciclo de vida de la información con el propósito de garantizar la integridad, disponibilidad y confidencialidad de la misma. Por tanto, la gestión de solicitudes son los procesos de recopilación, almacenamiento, procesamiento, análisis, conservación y acceso a las solicitudes con la intención de mejorar las estrategias de administración y de esta manera elevar la eficiencia y la eficacia en las empresas.

1.5 Gestión de solicitudes en el Grupo de Calidad del centro FORTES

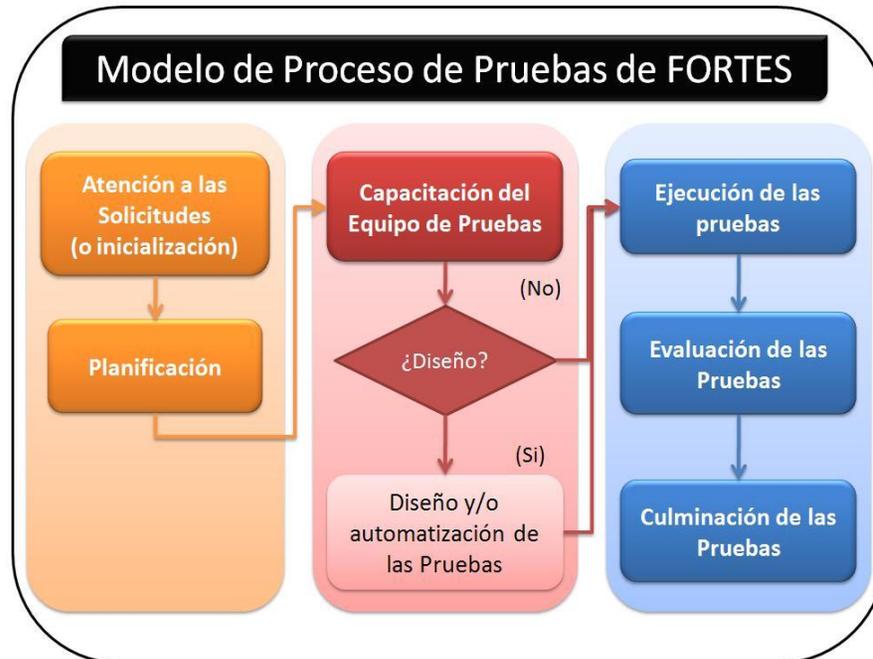


Figura 2 Modelo de proceso para la realización de Pruebas Grupo de Calidad del centro FORTES.
Tomado de Grupo de Calidad del centro FORTES. Procesos fundamentales.

El Grupo de Calidad cuenta con un Modelo de proceso para el servicio de Prueba de *software*, representado en la figura 2, donde se describen las actividades y pasos para ejecutar este servicio. El presente trabajo solo va hacer referencia dentro de este proceso a la actividad de Atención a las solicitudes o inicialización.

Atención a las solicitudes de Pruebas (o Inicialización)

La atención a las solicitudes o inicialización es la primera actividad dentro del proceso de realización de Pruebas, y tiene como objetivo establecer las comunicaciones previas a la coordinación del servicio y crear las condiciones necesarias para su preparación.

Para solicitar el servicio de Prueba de *software* al Grupo de Calidad del centro FORTES, las solicitudes deben ser comunicadas al jefe del equipo de prueba a través de la dirección del Grupo de Calidad. Los interesados llenarán una Plantilla de Solicitud de Pruebas que deben enviar, y junto a ella, proveerán al

jefe del equipo de prueba toda la documentación y artefactos necesarios para la realización del servicio. Toda la información se evalúa teniendo en cuenta su correcta elaboración, integridad, actualización y como resultado, las solicitudes pueden ser rechazadas o aceptadas. Para el último caso se asignarán prioridades, teniendo en cuenta la complejidad y la disponibilidad del equipo de prueba. La actividad termina con la comunicación a los clientes del estado de las solicitudes realizadas (aceptadas o rechazadas). Para el caso de las rechazadas, deben volver a realizar la solicitud y para el caso de que esta sea aceptada, se convocará a la reunión de inicio del servicio.

Actualmente este proceso afronta algunas dificultades. Debido a que el líder del proyecto es el encargado de recibir las solicitudes en su correo, revisarlas y posteriormente enviar la respuesta de aceptación o rechazo. Esto puede provocar que las solicitudes sobrecarguen el correo, y por ende, perderse, llegar con retraso o no llegar y en otro caso borrarse por error. El envío de respuesta a los clientes sobre el estado de su solicitud se vuelve engorroso cuando el número de peticiones recibidas es alto. La recopilación de datos de las plantillas de solicitud se vuelve complejo para la revisión, obtención de reportes y planificación de servicios, ya que es un trabajo manual que provoca lentitud en estas actividades cuando se tiene muchas solicitudes.

1.6 Análisis de soluciones existentes

La informatización de la sociedad y el desarrollo de las tecnologías han supuesto la creación de programas o aplicaciones relacionados con la solicitud de servicios de una entidad que satisfagan las más disímiles necesidades. Lo importante de una aplicación no es la complejidad que posee sino la capacidad de satisfacer a sus usuarios o no. La gestión de solicitudes juega un papel importante para la recopilación de información de los servicios que se desean recibir y para una correcta planificación del trabajo. Por lo que se hace inevitable el estudio y análisis de soluciones que existen a nivel nacional e internacional que realicen funcionalidades similares a lo que se propone y cumplan con las funcionalidades para la gestión de solicitudes.

1.6.1 Soluciones a nivel internacional

Las soluciones estudiadas a nivel internacional presentan funcionalidades que automatizan actividades dentro del proceso de la gestión de las solicitudes. En su mayoría están relacionadas con servicios universitarios o empresas dedicadas a la consultoría.

Sistema de gestión de solicitud de la Universidad ICESI

El sistema de gestión de solicitud de la Universidad ICESI Colombia, permite a los estudiantes reportar casos (solicitudes) e incidencias de la universidad. La aplicación puede ser consultada en la dirección http://www.icesi.edu.co/solicitud_servicios/login_page.php. Lleva un control completo de todos los casos reportados por el usuario, registrando la fecha y hora para cada caso y el estado en dependencia de las acciones que se realizan al caso. (9)

Es un *software* propietario, que posee funcionalidades para adicionar casos, listar los casos según el usuario registrado y brinda la opción de ver los diferentes estados que tienen los casos del usuario: no asignados, se listan los casos que ha reportado el usuario, pero aún no han sido asignados a un analista; confirmados, lista los casos que han sido confirmados por los analistas junto con el usuario que los reportó; reportados por mí, lista los casos que han sido reportados por el usuario; modificados recientemente, lista los casos a los que se les ha realizado alguna modificación recientemente, las modificaciones incluyen: asignación a un analista, adición de notas, actualización del campo fecha y hora acordada, cambio de estado; fecha y hora acordada, campo que permite visualizar si el caso reportado ya tiene fecha y hora para su solución. Además, cuenta por un tránsito de varios estados según la actividad que se encuentre realizando al caso: nuevo, no aprobado, aprobado, asignado, resuelto, confirmado y cerrado. (9)

El sistema de gestión de solicitudes de la universidad de ICESI presenta estados que dependen de las acciones que se acometen. Estos son de utilidad para la solución que se desea implementar. Pero como es un *software* privativo se debe prescindir de su uso.

BITEC Soluciones C.A

La empresa BITEC Soluciones cuenta con una herramienta de solicitud para sus servicios de soporte, consultoría, sistema y venta de equipo dentro de su portal web, a la cual se puede acceder a través de la

dirección: <http://www.bitec.com.ve/solicitud.php?go=1>. Mediante un formulario recoge los datos de una solicitud para cualquiera de los servicios que esta entidad ofrece. BITEC en su herramienta para la solicitud de servicios no facilita la actualización o eliminación de los datos por parte del cliente, no presenta generación de reportes, envío de notificación del estado de la solicitud al cliente ni funcionalidades que permitan la aceptación o rechazo de las solicitudes.

1.6.2 Soluciones a nivel nacional

Las soluciones estudiadas a nivel nacional presentan funcionalidades que automatizan actividades dentro del proceso de gestión de las solicitudes. En su mayoría están relacionadas con servicios migratorios o empresas dedicadas a la calidad de *software*.

Sistema de Solicitud y Emisión de Movimientos Migratorios (SAIME)

El Sistema de Solicitud y Emisión de Movimientos Migratorios es un trabajo de diploma realizado en la Facultad 1 de la Universidad de las Ciencias Informáticas e implementado para el Sistema Autónomo de Identificación, Migración y Extranjería (SAIME) de Venezuela. Entre las partes que constituyen este sistema, se encuentra el Sistema de Solicitud, el cual es el encargado de registrar solicitudes de movimientos migratorios, permitiendo captar los datos de entidades (públicas o naturales) y hacer efectiva la solicitud; buscar una solicitud, según la letra y número de cédula de una persona, número de pasaporte, o por código del país o nacionalidad; seleccionar de un listado de solicitudes de movimientos migratorios listas para impresión y mandar a imprimir de forma manual; comprobar que los datos de la solicitud sean los correctos verificando los reportes impresos con el listado de solicitudes; enviar la solicitud realizada de movimientos migratorios a la Oficina Regional; entregar solicitud de movimientos migratorios, se muestra el estado en que se encuentra la solicitud de movimientos migratorios realizada (solicitado, impreso, supervisado, enviado a oficina) y entrega el oficio correspondiente. No se pudo acceder al sistema vía Internet, debido que se necesita ser ciudadano venezolano y la Facultad 1 no permitió consultar la información por ser un proyecto del Ministerio del Interior (MININT).

CALISOFT

CALISOFT recientemente ha incorporado dentro de su portal el servicio de solicitud de evaluación, a la cual se puede acceder mediante la dirección <http://calisoft.uci.cu/index.php/component/solicitudes/>.

Presenta como principal característica el llenado de varios campos para la recogida de información del proyecto y del tipo de evaluación que el cliente solicita. La herramienta carece de funcionalidades para la modificación o eliminación de solicitudes por parte del usuario, no genera reportes, no envía notificación del estado de la solicitud al cliente, ni facilita la aceptación o rechazo de una solicitud.

1.6.3 Análisis de las soluciones

El análisis y estudio de sistemas de gestión de solicitudes arrojó como resultado que no se puede acceder a muchas de las herramientas. En el caso del Sistema de gestión de solicitud de la Universidad de ICESI por ser privativa, mientras que al Sistema de Solicitud y Emisión de Movimientos Migratorios por pertenecer al MININT. En general, las soluciones que se encuentran en sitios de entidades como CALISOFT y BITEC, carecen de funcionalidades para la gestión de las solicitudes como generación de reportes, eliminación y modificación de las solicitudes creadas o la notificación al cliente del estado de su solicitud después de revisada. Sin embargo, el estudio sirvió para tomar varias ideas para la implementación del sistema, tales como: el tránsito de estados de una solicitud.

1.7 Herramientas y tecnologías de desarrollo

El Sistema de Gestión de Calidad lo conforman un conjunto de módulos para garantizar la calidad de los servicios que ofrece el Grupo de Calidad del centro FORTES. Entre los módulos con que debe contar el sistema se tiene el Módulo de Prueba que ha sido diseñado e implementado. Para el cual se realizó un estudio y análisis de diferentes herramientas necesarias en su desarrollo, por ende, estas aplicaciones deben utilizarse en la presente solución. Partiendo que el módulo a desarrollar debe ser integrado al Sistema de Gestión de Calidad. Por tanto se abordarán las herramientas ya utilizadas, fundamentando en sus principales características, como es el caso de la metodología de desarrollo, el sistema de gestión de contenido, sistema gestor de base de datos, el servidor web y los lenguajes de programación. Mientras que a las restantes herramientas se realizará un estudio y comparación, con el fin de seleccionar las tecnologías que más se ajusten a las necesidades del programa a implementar.

1.7.1 Metodologías de desarrollo

Las metodologías de desarrollo de *software* son un conjunto de pasos y procedimientos que deben seguirse para llevar a cabo el desarrollo de programas con calidad. Estas brindan un conjunto de detalles

organizativos, añadiendo rigurosidad y normas, permitiendo que los integrantes de un equipo de desarrollo puedan seguir un criterio común a la hora de realizar las tareas. Una metodología debe ser lo suficientemente adaptable como para poder aplicarse en distintos proyectos, y lo suficientemente sencilla para que no resulte muy pesada su utilización, pero a la vez suficientemente completa como para que su utilización por parte de un equipo sea provechosa.

Programación Extrema (XP)

La metodología ágil XP se basa en la retroalimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los involucrados, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define, especialmente, para proyectos con requisitos imprecisos y muy cambiantes, donde existe un alto riesgo técnico, el equipo de trabajo sea pequeño y cuente con pocos desarrolladores, además que se requiera de poco tiempo para la implementación del *software*. El ciclo de vida ideal de XP consiste en cuatro fases y un conjunto de prácticas fundamentadas en un grupo de valores.

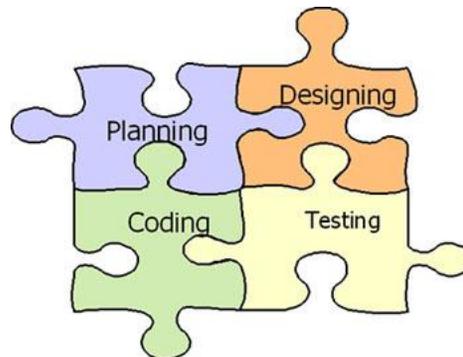


Figura 3 Fases de desarrollo de XP.

Planificación

Historias de usuario (HU): el primer paso de cualquier proyecto que siga la metodología XP es definir las historias de usuario con el cliente. Tienen la misma finalidad que los casos de uso, pero con la diferencia que son escritas por el cliente, en un lenguaje no técnico sin hacer muchos detalles. No se debe hablar ni de posibles algoritmos para su implementación ni de diseños de base de datos adecuados. (10)

Release planning o Planificación de entrega: un plan de entrega es una planificación donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario, la prioridad y las HU que serán implementadas en cada versión del programa. (10)

Iteraciones: todo proyecto que siga la metodología XP se ha de dividir en iteraciones de aproximadamente tres semanas de duración. Al comienzo de cada iteración los clientes deben seleccionar las historias de usuario definidas en el *Release planning* que serán implementadas. También se seleccionan las que no pasaron las pruebas de aceptación que se realizó al terminar la iteración anterior. Estas historias de usuario son divididas en tareas de entre uno y tres días de duración que se asignarán a los programadores. (10)

Velocidad del proyecto: la velocidad del proyecto es una medida que representa la rapidez con la que se desarrolla el proyecto. Basta con contar el número de historias de usuario que se pueden implementar en una iteración, de esta forma se sabrá la totalidad que se pueden desarrollar en las distintas iteraciones. (10)

Programación en pareja: el trabajo en pareja involucra a dos programadores trabajando en el mismo equipo. Mientras uno codifica haciendo hincapié en la calidad de la función o método que está implementando, el otro analiza si ese método o función es adecuado y está bien diseñado. (10)

Reuniones diarias: es necesario que los desarrolladores se reúnan diariamente y expongan sus problemas, soluciones e ideas de forma conjunta. (10)

Diseño

Diseños simples: la metodología XP sugiere la implementación de diseños simples y sencillos, que costarán menos tiempo y esfuerzo para desarrollar. (10)

Glosarios de términos: usar glosarios de términos, además de una correcta especificación de los nombres de métodos y clases, ayudará a comprender el diseño y facilitará posteriores ampliaciones y la reutilización del código. (10)

Riesgos: Si surgen problemas potenciales durante el diseño, XP sugiere utilizar una pareja de desarrolladores para que investiguen y reduzcan al máximo el riesgo que supone ese problema. (10)

Funcionalidades extra: nunca se debe añadir funcionalidades extra al programa, aunque se piense que en un futuro será utilizada. (10)

Refactorizar: supone revisar de nuevo los códigos para procurar optimizar su funcionamiento. Es muy común rehusar códigos ya creados que contienen funcionalidades que no serán usadas y diseños obsoletos. (10)

Tarjetas CRC: El uso de las tarjetas CRC (Clases, Responsabilidades y Colaboraciones) permite al programador centrarse y apreciar el desarrollo orientado a objetos, olvidándose de los malos hábitos de la programación procedural clásica. (10)

Codificación

La codificación debe hacerse contando con el cliente como parte del equipo de desarrollo, atendiendo a estándares de codificación ya creados. Facilita la comprensión y escalabilidad creando pruebas que validen el funcionamiento de los distintos códigos implementados. La programación debe ser en pareja, pues permite obtener códigos más eficiente y de calidad. (10)

Pruebas

Test de aceptación o Pruebas de aceptación: tienen el propósito de asegurar el funcionamiento final de una determinada historia de usuario. (10)

1.7.2 Herramientas CASE

Las herramientas *Computer Automated System Engeneering* (CASE) son un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros de *software* y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un programa. Entre los beneficios que ofrecen se encuentra la verificación del uso de todos los elementos en el sistema diseñado, automatiza el dibujo de diagramas, ayuda en la documentación del sistema, ayuda en la creación de bases de datos relacionales y genera estructuras de código. (11)

Entre las herramientas CASE más utilizadas se encuentra **Visual Paradigm**, tecnología que proporciona un conjunto de ayuda para el desarrollo de *software* de aplicación. Soporta estándares de la industria,

como lo es el Lenguaje de Modelado Unificado (UML). Ofrece un completo conjunto de herramientas para los equipos de desarrollo necesario para la captura de requisitos, la planificación de programas, modelado de clases y modelado de datos. (12) Presenta una licencia dual, una gratuita que puede ser utilizada con fines educativos como tiene nuestra universidad y otra comercial para aquellas empresas que comercializan programas y utilicen el Visual Paradigm para desarrollar sus aplicaciones.

Visual Paradigm es una potente herramienta empleada para visualizar y diseñar elementos de *software*, proporciona a los desarrolladores una plataforma para diseñar un producto con calidad de forma rápida. Se integra con diversos IDEs como: NetBeans y Eclipse. Genera código y realiza ingeniería inversa para diferentes lenguajes de programación como: Java, C++, PHP, entre otros. Posibilita la representación gráfica de los diagramas permitiendo ver el sistema desde diferentes perspectivas, como el diagrama de componentes, despliegue, secuencia, casos de uso, clase, actividad, estado, entre otros. Además, identifica requisitos y comunica información, se centra en cómo los componentes del sistema interactúan entre ellos, sin entrar en detalles excesivos, permitiendo ver las relaciones entre los componentes del diseño y mejora la comunicación entre los miembros del equipo usando un lenguaje gráfico. (13)

Sin embargo, **Rational Rose** es una herramienta con licencia privativa, pero la mejor elección para el ambiente de modelado que soporte la generación de código a partir de modelos en Ada, ANSI C++, C++, CORBA, Java/J2EE, Visual C++ y Visual Basic. Como todos los demás productos Rational Rose, proporciona un lenguaje común de modelado para el equipo que facilita la creación de *software* de calidad más rápidamente. (14)

Rational Rose se caracteriza por mantener la consistencia de los modelos del sistema de *software*, chequea la sintaxis UML, la generación de la documentación es automática, la generación de código es a partir de modelos y permite la ingeniería inversa. (15)

Selección de la herramienta CASE

Se propone la utilización de Visual Paradigm debido a que la universidad cuenta con una licencia para este programa, con fines educativos. Ofrece un entorno de creación de diagramas para UML, presenta disponibilidad de integrarse con los principales IDEs como NetBeans y Eclipse, además, puede utilizarse en plataformas Linux y Windows. Se plantea la utilización de Visual Paradigm Suite 3.4 para UML 6.4, con el objetivo de modelar el diagrama de estado o máquina de estado del sistema.

1.7.3 Sistema de Gestión de Contenido (CMS)

Los Sistemas de Gestión de Contenidos (CMS) son programas desarrollados para crear, editar, manejar y publicar contenidos. Las ventajas más comunes son: completo control sobre el contenido, se puede actualizar el contenido dinámicamente sin conocimientos de programación, fácil integración de nuevas funcionalidades y completo control de usuarios. (16)

Drupal es un sistema de gestión de contenido de código abierto que sirve para administrar recursos web. Sistema dinámico que en lugar de almacenar sus contenidos en archivos estáticos en el sistema de ficheros del servidor, el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno web incluido en el producto. Sistema multiusuario, multiplataforma, multilinguaje, extensible y modular. (17) Se utilizará Drupal 6.20 debido a que el Sistema de Gestión de Calidad se encuentra soportado en esta versión. A continuación se describen varias de sus características (18):

- ✓ Ayuda on-line: presenta un robusto sistema de ayuda en línea y páginas de ayuda para los módulos del núcleo, tanto para usuarios como para administradores.
- ✓ Búsqueda: todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- ✓ Módulos: la comunidad de Drupal ha contribuido con infinidad de módulos que proporcionan funcionalidades como página de categorías, autenticación mediante chat, mensajes privados, *bookmarks*⁷, entre otros.
- ✓ Personalización: es un robusto entorno de personalización que se encuentra implementado en el núcleo de Drupal, tanto el contenido como la presentación pueden ser individualizados de acuerdo con las preferencias definidas por el usuario.

⁷ **Bookmark (marcadores)**: conocidos también como Favoritos, son aquellos enlaces a páginas web que hemos almacenado en nuestro navegador web por su interés o para su posterior visualización.

Gestión de usuario

- ✓ Autenticación de usuarios: los usuarios se pueden registrar e iniciar sesión de forma local o utilizando un sistema de autenticación externo como Jabber⁸, Blogger⁹, LiveJournal¹⁰ u otro sitio de Drupal. Para su uso en una intranet, Drupal se puede integrar con un servidor LDAP (Protocolo Ligerero de Acceso a Directorios).
- ✓ Permisos basados en roles: los administradores no tienen que establecer permisos para cada usuario. En lugar de eso, pueden asignar permisos a un rol y agrupar los usuarios por roles.

Gestión de contenido

- ✓ Objetos de contenido (Nodos): el contenido creado en Drupal es funcionalmente un objeto (nodo). Esto permite un tratamiento uniforme de la información como una misma cola de moderación para envíos de diferentes tipos. Permite promocionar cualquiera de estos objetos a la página principal o permitir comentarios o no sobre cada objeto.
- ✓ Multiplataforma: Drupal ha sido diseñado desde el principio para ser multiplataforma. Puede funcionar con Apache o Microsoft IIS como servidor web y en sistemas Linux, BSD, Solaris, Windows y Mac OS X. Por otro lado, al estar implementado en PHP, es totalmente portable.

Rendimiento y escalabilidad

- ✓ Control de congestión: incorpora un mecanismo de control de congestión que permite habilitar y deshabilitar determinados módulos o bloques dependiendo de la carga del servidor. Este mecanismo es totalmente configurable y ajustable.
- ✓ Sistema de Cache: el mecanismo de cache elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor.

⁸ **Jabber**: es un protocolo libre de mensajería instantánea, baso en un estándar robusto como XML (*Extensible Markup Language*) y gestionado por XMPP Standards Foundation.

⁹ **Blogger**: es un servicio creado por Pyra Labs y adquirido por Google en el año 2003, que permite crear y publicar una bitácora en línea. Para publicar contenidos, el usuario no tiene que escribir ningún código o instalar programas de servidor o de scripting.

¹⁰ **LiveJournal**: es el nombre de un sitio de weblog que permite a los internautas mantener un periódico o diario en línea.

1.7.4 Sistema Gestor de Base de Datos (SGBD)

Con el propósito de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante para una organización, están los lenguajes de base de datos y los Sistemas Gestores de Base de Datos (SGBD), estos últimos son un tipo de *software* que sirve de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

Entre los más reconocidos se encuentra **PostgreSQL**, es un SGBD relacional. Basado en el proyecto POSTGRES, de la universidad de Berkeley, el cual es una derivación libre de este proyecto que utiliza el lenguaje SQL92/SQL99. Incluye características de la Programación Orientación a Objetos (POO), como son la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. (19) Cuenta con una arquitectura que ha ganado gran reputación por su confiabilidad, estabilidad y mantenimiento de la integridad de los datos. Contiene extensiones de orientación a objetos, permitiendo definir un nuevo tipo de tabla a partir de una previamente definida. (20) Se usará para el desarrollo del módulo PostgreSQL en su versión 8.4, ya que debe ser compatible con la base de datos que tiene el Sistema de Gestión de Calidad cuando se integre.

1.7.5 Servidores web

Para procesar las aplicaciones del lado del servidor realizando conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente, existen los programas llamados servidores web. Entre los más utilizados en el mundo se encuentra **Apache HTTP Server**, *software* de tecnología libre, para plataformas Unix (BSD, GNU/Linux), Windows y otras.

Apache está estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías: módulos base, módulo con las funciones básicas y módulos multiproceso. Son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones y los módulos adicionales, cualquier otro módulo que le añada una funcionalidad al servidor. (21) También, Apache trabaja con Perl, PHP y otros lenguajes de script y permite personalizar la respuesta ante los posibles errores. Como requisito mínimo para la

instalación de Drupal 6 se requiere de Apache HTTP Server, en este caso en su versión 2.2, el cual será utilizado para montar el servidor web donde se encontrará el Sistema de Gestión de Calidad.

1.7.6 Tecnologías del lado del servidor

Para el desarrollo de aplicaciones web existen numerosos lenguajes de programación de lado del servidor, entre los que destacan PHP, Java, Perl, Ruby, Python, entre otros. Uno de los lenguajes de programación más popular, que se caracteriza por la gran fluidez y rapidez de sus script es **Hypertext Pre-processor (PHP)**. Lenguaje interpretado, diseñado originalmente para la creación de páginas web dinámicas. Gratuito e independiente de plataforma, con una gran librería de funciones y numerosa documentación. (22)

PHP se escribe dentro del código HTML, lo que lo hace realmente fácil de utilizar. Es independiente de plataforma, puesto que existe un módulo de PHP para casi cualquier servidor web. Esto hace que cualquier sistema pueda ser compatible con el lenguaje y portar el sitio desarrollado de un sistema a otro sin prácticamente ningún trabajo. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Su programación es segura y confiable, compatible con las bases de datos más comunes tales como: MySQL y PostgreSQL. Permite aplicar técnicas de programación orientada a objetos. Este lenguaje de programación se encuentra preparado para realizar muchos tipos de aplicaciones web gracias a la extensa librería de funciones con la que está dotado. (22)Debido a que Drupal se encuentra escrito en PHP y se necesita que el módulo a implementar no presente inconvenientes cuando se integre al Sistema de Gestión de Calidad, se propone utilizar PHP 5.2.

1.7.7 Tecnologías del lado del cliente

En cuanto a los lenguajes de programación de lado del cliente más utilizados están JavaScript, CSS, XHTML, entre otros. Estos tipos de lenguajes permiten que las aplicaciones, portales y sitios web sean accesibles desde cualquier navegador. Facilitan la interacción entre el usuario y el diseño visual para que sea atractivo. El CMS Drupal emplea estos lenguajes para el tratamiento de sus *themes* o temas, con el objetivo de establecer organización y el diseño de sus páginas. A continuación se aborda el lenguaje XHTML y CSS que se utilizará en la implementación del módulo.

Lenguaje de Marcado de Hipertexto Extendido (XHTML) se utiliza para generar documentos y contenidos de hipertexto generalmente publicados en la web, es una reformulación del lenguaje HTML y XML. Un lenguaje de marcado nos permite dejar indicaciones o marcas en un documento que sirven para diferenciar distintos tipos de contenidos, estructuras o secciones. Por ejemplo hay marcas (elementos o etiquetas) que pueden marcar qué es un título y su lugar en una jerarquía de importancia o advertir que una palabra debe estar enfatizada, o bien que una oración es además un enlace hacia otro recurso. Todo esto es posible de marcar sin ambigüedades con el correcto uso de XHTML. Otra de sus virtudes es que está ideado para mantener una saludable separación entre el contenido y el diseño, es decir, que uno no afecte al otro y se puedan modificar independientemente. El tema de Drupal a utilizar presenta la versión de XHTML 1.0. (23)

Otro muy empleado, es las **Hojas de Estilo en Cascada (CSS)**, lenguaje usado para definir la presentación de un documento estructurado escrito en XHTML. La World Wide Web Consortium (W3C) es la encargada de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores. Con el objetivo de detallar los elementos del tema de Drupal y los del módulo a implementar se utilizará la versión de CSS 2. Las ventajas de su uso son (24):

- ✓ Control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.
- ✓ Los navegadores permiten a los usuarios especificar su propia hoja de estilo local que será aplicada a un sitio web, con lo que aumenta considerablemente la accesibilidad. Por ejemplo, personas con deficiencias visuales pueden configurar su propia hoja de estilo para aumentar el tamaño del texto o remarcar más los enlaces.
- ✓ Una página puede disponer de diferentes hojas de estilo según el dispositivo que la muestre o incluso a elección del usuario. Por ejemplo, para ser impresa, mostrada en un dispositivo móvil, o ser leída por un sintetizador de voz.
- ✓ El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño siempre y cuando no se utilice estilo en línea.

1.7.8 Entornos de desarrollo integrado (IDE)

Un Entorno de Desarrollo Integrado (IDE) es un entorno de programación que ha sido empaquetado como un programa de aplicación, o sea, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Los IDEs pueden ser aplicaciones por sí solos o pueden ser parte de aplicaciones existentes.

NetBeans es un proyecto exitoso de código abierto con una gran base de usuarios y una comunidad en constante crecimiento. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio 2000 y continúa siendo el patrocinador principal de los proyectos. Es un entorno de desarrollo y una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender la herramienta. (25) Se caracteriza por presentar mejoras en el editor de código, su instalación y actualización es más simple que otros IDE, presenta características visuales para el desarrollo web, mejoras para SOA y UML y soporte para PHP. (26)

Eclipse es principalmente una plataforma de programación, usada para crear entornos integrados de desarrollo. Creado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge. Eclipse es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto. (27) Este entorno integrado de desarrollo emplea módulos para proporcionar toda su funcionalidad al frente de la plataforma de cliente rico, a diferencia de otros entornos monolíticos donde las funcionalidades están todas incluidas, las necesite el usuario o no. Este mecanismo de módulos es una plataforma ligera para componentes de *software*. (28)

Selección del entorno de desarrollo

Estos IDE's tienen funcionalidades similares y los dos son una excelente selección, pero se propone como entorno de desarrollo NetBeans. Debido que se va por la simplicidad y peso ligero, todo funciona sin necesidad de configuración y se puede centrar en la codificación en lugar de tratar con el IDE, todo lo contrario de Eclipse. Agregar que es una herramienta de código abierto, presenta soporte para PHP y sus funcionalidades son extensibles mediante la instalación de paquetes. Tiene un potente motor para la conexión a bases de datos, fácil de instalar y se ejecuta en varias plataformas como Windows y Linux. Es más intuitivo, teniendo memoria puede funcionar relativamente bien. NetBeans es perfecto y cómodo para

los programadores, debido que tiene un excelente balance entre una interfaz con múltiples opciones y un aceptable completamiento de código. Se utilizará en la versión 7.1, producto al uso del PHP como lenguaje del lado del servidor, es ideal para la integración con el Drupal.

1.8 Conclusiones parciales

- ✓ La gestión de las solicitudes son un conjunto de actividades destinadas a recopilar, analizar, evaluar, verificar y conservar la información para permitir a las empresas u organizaciones mejorar en eficacia, rentabilidad, desempeño y seguridad.
- ✓ Las aplicaciones estudiadas tanto en el ámbito internacional como nacional son privativas o no cuentan con las funcionalidades requeridas por el cliente para la gestión de solicitudes en el centro FORTES.
- ✓ Para el desarrollo de la solución se seleccionaron varias herramientas, como metodología de desarrollo se determinó XP, Visual Paradigm Suite 3.4 como herramienta de modelado UML, como sistema de gestión de contenido Drupal 6.20, como sistema de gestor de base de datos PostgreSQL 8.4, como servidor web Apache HTTP Server 2.2, como lenguaje de programación PHP 5.2, CSS 2 y XHTML 1.0, como entorno de desarrollo NetBeans 7.1.

Capítulo 2. Características del sistema

2.1 Introducción

XP es la metodología seleccionada para el proceso de desarrollo del *software*. Promueve el aumento de las relaciones interpersonales, se preocupa por la superación de sus desarrolladores, incita al trabajo en equipo y brinda un buen ambiente. La base de XP es la comunicación continua entre el cliente y el equipo de desarrollo, existiendo fluidez entre todos los involucrados.

El presente capítulo hace alusión a las fases de Planificación y Diseño, las primeras de la metodología de desarrollo XP. El objetivo principal de ellas es conocer el alcance del producto a desarrollar y estimar los tiempos de entrega de cada versión. Además, se presenta las Historias de Usuarios (HU), las nuevas tablas de la base de datos que necesita Drupal para gestionar la información del sistema, las tarjetas CRC y la descripción de la arquitectura, patrones de diseño y como se manifiesta el Modelo Vista Controlador en Drupal.

2.2 Descripción del sistema

El Sistema de Gestión de Calidad del centro FORTES es una herramienta que permite optimizar e informatizar los procesos de los servicios que brinda el Grupo de Calidad. De los módulos que componen el sistema se define el Módulo Solicitud, el cual facilita la gestión de las solicitudes de los servicios del proyecto.

El Grupo de Calidad del centro FORTES cuenta con cuatro servicios: el servicio de Prueba de *software*, Auditoría y Revisión (el sistema resume estos dos últimos servicios en Evaluación) y Asesoramiento. La solución propuesta consta de dos secciones, la primera es gestionar solicitud. Esta sección permite que cualquier estudiante o profesor del centro se autentique e inserte los datos de una solicitud para un servicio, la solicitud inmediatamente toma el estado Pendiente. Además, el usuario puede modificar o eliminar la solicitud que él agregó, si dicha solicitud se mantiene en estado Pendiente, el sistema le permite verificar los datos mediante un listado de todas las solicitudes y la seleccionada se exporta a PDF.

La segunda sección es administrar solicitudes y solo tiene permiso el profesor planificador del proyecto. Con el objetivo de realizar un proceso de revisión de las últimas solicitudes, el planificador puede aceptar, rechazar o cerrar una solicitud, con estas acciones cambia el estado a Aceptada, Rechaza o Cerrada, respectivamente. Cuando se acepta una solicitud el sistema envía una notificación por correo al solicitante y al responsable asignado al servicio. En caso de rechazar la solicitud el sistema envía por correo al solicitante la justificación del rechazo. Solo se pasa de estado Aceptada o Rechazada si la solicitud se encuentra en Pendiente y a Cerrada si tiene el estado de Aceptada o Rechazada. Si el planificador desea obtener estadísticas de las solicitudes realizadas en el centro, el sistema permite realizar reporte de filtro a las solicitudes por el tipo de servicio, el estado, el nombre del proyecto y la fecha de la solicitud. Las peticiones filtradas son exportables a PDF.

La aplicación es fácil de usar, presenta una interfaz amigable, con colores suaves y pocos elementos visuales, lo que facilita que el usuario centre su atención en los elementos propios del trabajo con el *software* y no distraiga su atención. Puede utilizarse tanto en sistemas operativo Linux como en Windows, con navegador web Mozilla Firefox 3.0 o superior e Internet Explorer 8 o superior. Para instalar y ejecutar Drupal 6.20 en el proveedor web debe cumplir con ciertos requisitos mínimos: en cuanto al espacio en disco, la instalación mínima de la base requiere de al menos 15 MB de espacio en disco, pero no son las necesidades reales, en este caso son 100MB, esto depende de en gran medida de los módulos que se adhieran al CMS; para el servidor web se requiere Apache 2.2; para el servidor de base de datos se recomienda PostgreSQL 8.4 y como lenguaje de programación PHP se recomienda la versión 5.2. Las interfaces de usuarios pueden ser consultadas en el [Anexo 4](#).

2.3 Planificación

Planificación es la primera fase de la metodología XP donde el cliente establece la prioridad de cada historia de usuario (HU), y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente.

2.3.1 Historias de usuarios

Las historias de usuarios (HU) son escritas por el cliente con ayuda de los diseñadores, en un lenguaje común y sin detalles de elementos de programación. Además, son los clientes que establecen la prioridad de las HU.

Las tablas están compuesta por los siguientes campos: No., contiene el identificador de la HU; Nombre, contiene el nombre que identifica a la HU; Usuario, roles que pueden acceder a la HU; Modificación de la HU, se especifica el número de veces que se ha modificado la HU; Iteración asignada, número de la iteración en que se va a implementar la HU; Prioridad, tipo de prioridad de la HU (Alta, Media o Baja); Puntos estimados, estimación hecha por el equipo de desarrollo del tiempo de duración de la HU; Riesgo en desarrollo, nivel de riesgo que posee la HU en caso de no realizarse (Alto, Medio o Bajo); Puntos reales, tiempo real en el que se realizó la HU; Descripción, breve descripción del proceso que define la HU y Observaciones, comentarios aclaratorios relacionados con la HU.

En la tabla 1, se muestra la HU Crear solicitud de servicio, con prioridad alta, es decir, arquitectónicamente significativa. El resto de las HU se incluyen en el [Anexo 2](#).

Tabla 1 HU Crear solicitud de servicio.

No.: 1	Nombre: Crear solicitud de servicio
Usuario: Usuario autenticado, planificador y administrador	
Modificación de HU: 0	Iteración asignada: 1
Prioridad en negocio: Alta	Puntos estimados: 4 días
Riesgo en desarrollo: Alto	Punto reales: 4 días
Descripción:	<p>El usuario ingresa los datos necesarios para crear una nueva solicitud de un servicio y selecciona la opción de Guardar. El sistema valida y guarda la información en la base de datos, al finalizar muestra un mensaje de confirmación. En caso que exista datos incorrectos, se muestra un mensaje de error.</p> <p>Si selecciona la opción de Cancelar, el sistema borra los datos introducidos y re-direcciona a la misma página.</p>
Observaciones:	<ul style="list-style-type: none"> - El usuario debe estar autenticado. - Apenas que se cree una nueva solicitud, esta toma el estado Pendiente.

2.3.2 Estimación de esfuerzos por historias de usuario

La estimación de esfuerzos por historias de usuario la establecen los programadores. Las historias de usuarios deben ser programadas en un tiempo estimado entre una y tres semanas. Si la estimación supera las tres semanas la HU deberá ser dividida hasta que pueda ser desarrollada en un tiempo factible. En caso de que el esfuerzo sea menor de una semana la HU será combinada con otra historia. En este caso la estimación se efectuará en días debido a la poca complejidad de las HU.

Tabla 2 Estimación de esfuerzos por historias de usuario.

Historias de usuario	Estimación (días)
Crear solicitud de servicio	4
Modificar solicitud de servicio	3
Eliminar solicitud de servicio	3
Ver solicitud de servicio	1
Listar solicitudes	1
Aceptar solicitud del servicio	2
Rechazar solicitud del servicio	2
Cerrar solicitud del servicio	2
Realizar reportes a las solicitudes de servicios	3
Exportar a PDF	2

2.3.3 Plan de iteraciones

Todo proyecto que siga la metodología XP se ha de dividir en iteraciones de aproximadamente tres semanas de duración. Al comienzo de cada iteración los clientes deben seleccionar las historias de usuario definidas, que serán implementadas. También se seleccionan las historias de usuario que no pasaron las pruebas de aceptación que se realizó al terminar la iteración anterior. (10)

Iteración 1: esta iteración tiene como objetivo la codificación de las historias de usuario de alta prioridad, es decir, las que se caracterizan porque el cliente introduzca los datos necesarios para crear, modificar, eliminar y ver una solicitud de un determinado servicio.

- ✓ HU1 Crear una solicitud de servicios.
- ✓ HU2 Modificar solicitud de servicios.

- ✓ HU3 Eliminar solicitud de servicios.
- ✓ HU4 Ver solicitud de servicios.
- ✓ HU5 Listar solicitudes.

Iteración 2: el objetivo de esta iteración es la codificación de las funcionalidades con prioridad media y baja, que permiten llevar a cabo el proceso de revisión de solicitudes de servicios, a medida que cambia de estado y realizar los reportes de las solicitudes.

- ✓ HU6 Aceptar solicitud del servicio.
- ✓ HU7 Rechazar solicitud del servicio.
- ✓ HU8 Cerrar la solicitud del servicio.
- ✓ HU9 Realizar reportes a las solicitudes de servicios.
- ✓ HU10 Exportar a PDF.

Plan de duración de iteraciones

Tabla 3 Plan de duración de las iteraciones.

Iteraciones	Historias de usuario	Duración (días)
Iteración1	Crear solicitud de servicio	14 días
	Modificar solicitud de servicio	
	Eliminar solicitud de servicio	
	Ver solicitud de servicio	
	Listar solicitudes	
Iteración 2	Aceptar solicitud del servicio	14 días
	Rechazar solicitud del servicio	
	Cerrar solicitud del servicio	
	Realizar reportes a las solicitudes de servicios	
	Exportar a PDF	

2.3.4 Plan de entregas

En el plan de entregas se establecen el orden y las historias de usuario que serán agrupadas para conformar una entrega. Este plan tiene como finalidad reflejar la duración de cada iteración, lo que ayuda

a obtener una idea aproximada del tiempo que durará la confección del sistema en su totalidad. Este cronograma será el resultado de una reunión con todo el equipo de desarrollo incluyendo al cliente, permitiendo alcanzar un mayor entendimiento en la implementación del sistema.

Tabla 4 Plan de entregas.

Historias de usuario	Fin 1ra Iteración	Fin 2ra Iteración
Crear solicitud de servicio	V 1.0	Finalizado
Modificar solicitud de servicio	V 1.0	Finalizado
Eliminar solicitud de servicio	V 1.0	Finalizado
Ver solicitud de servicio	V 1.0	Finalizado
Listar solicitudes	V 1.0	Finalizado
Aceptar solicitud del servicio	-	V 1.0
Rechazar solicitud del servicio	-	V 1.0
Cerrar solicitud del servicio	-	V 1.0
Realizar reportes a las solicitudes de servicios	-	V 1.0
Exportar a PDF	-	V 1.0

2.4 Diseño

En esta fase la metodología XP sugiere que hay que conseguir diseños simples y sencillos para lograr diseño fácilmente entendible que a la larga costará menos tiempo y esfuerzo desarrollar. Nunca se debe añadir funcionalidad extra al programa aunque se piense que en un futuro será utilizada.

2.4.1 Diagrama de estado del sistema

La máquina de estado o diagrama de estado no es un artefacto que genere la metodología XP, pero se aborda en el presente trabajo para mayor entendimiento del tránsito de una solicitud en el sistema. Los diagramas de estados son una técnica conocida para describir el comportamiento de un objeto en un sistema. Describen todos los estados posibles en los que puede entrar un objeto particular y la manera en que cambia el estado del objeto, como resultado de los eventos que llegan a él. (29)

Estados: un estado puede considerarse una descripción abreviada de la situación actual del sistema. Un ejemplo para este sistema, las solicitudes pueden pasar por el estado Pendiente, Aceptada, Rechazada y Cerrada. (30)

Eventos: un evento es un mensaje de entrada a la máquina de estados. El evento puede provocar que un objeto cambie de estado. Ejemplo, cuando un planificador desea aceptar una solicitud, la presión del botón Aceptar, es un evento que permitirá el cambio de una solicitud de Pendiente para Aceptada. (30)

Transiciones: una transición es una ruta dirigida de un estado a otro. La transición se designa generalmente con un nombre de evento, indicando así que la transición tendrá lugar si el evento especificado sucede. El resultado final es el cambio de estado del objeto buscado por la transición. Para incrementar el poder expresivo de la notación de la máquina de estados, una transición puede protegerse mediante condiciones de vigilancia que deben cumplirse para llevarse a cabo. (30)

Acciones: una acción es una actividad a realizar por la máquina de estados en un punto determinado en el tiempo. Generalmente, una acción realiza alguna tarea en el entorno y pueden asociarse a transiciones, o bien pueden especificarse como acciones de entrada y salida en estados específicos. (30)

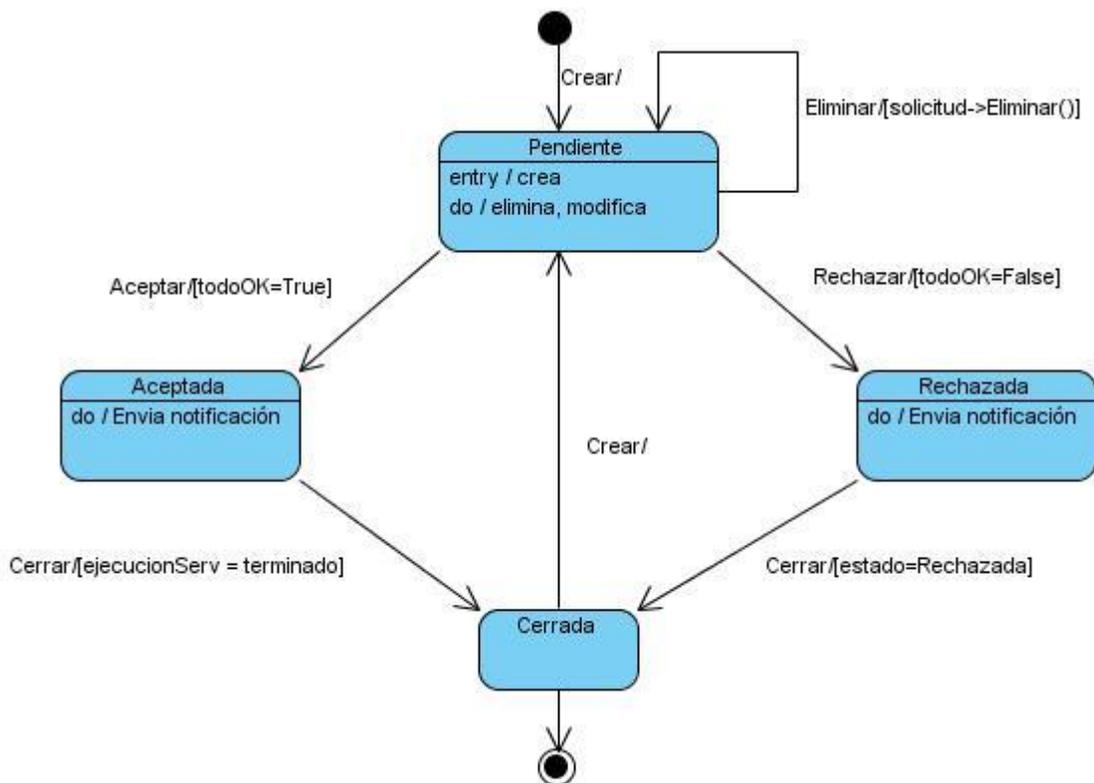


Figura 4 Diagrama de estado de una solicitud de servicio.

2.4.2 Tablas a incorporar en la base de datos de Drupal

La metodología XP no propone un modelo de datos, pero es importante el almacenamiento de información debido que es una actividad fundamental para el funcionamiento de los módulos en Drupal. Esto permite almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Drupal no utiliza un modelo de datos relacional, ya que los módulos se encargan de realizar las acciones pertinentes para relacionar las entidades, mientras que el sistema gestor de base de datos solo almacena los datos. Por lo que se decide realizar un diseño de las nuevas tablas que se incorporan en la base de datos de Drupal, donde se estructuran las tablas con sus claves primarias y sin relaciones. Las entidades permiten el almacenamiento de los datos del proyecto, de la persona solicitante, la solicitud, del servicio solicitado y del estado que toma la solicitud.

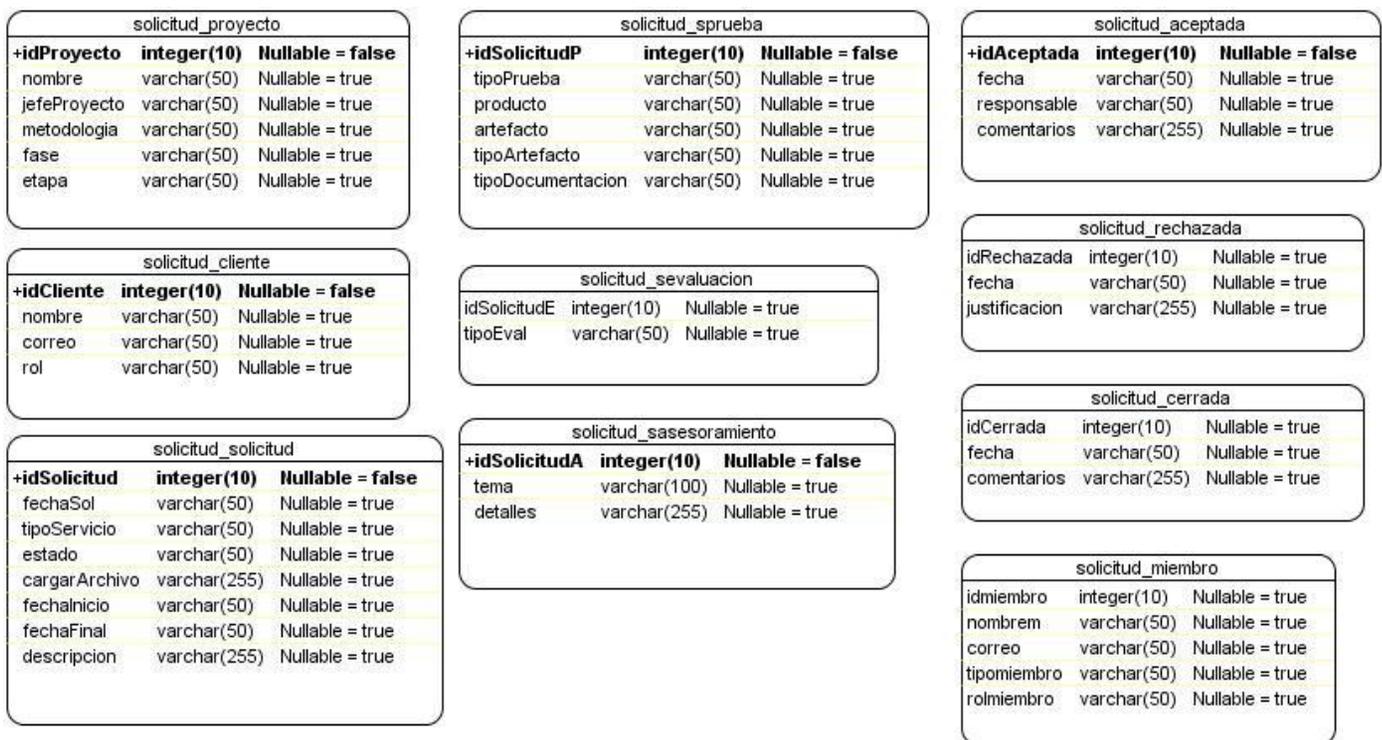


Figura 5 Tablas a incorporar en la base de datos de Drupal.

2.4.3 Tarjetas CRC (Clases, Responsabilidades y Colaboraciones)

Las tarjetas Clases, Responsabilidades y Colaboraciones (CRC) son una técnica de modelado orientado a objetos que permite identificar las clases, sus responsabilidades y colaboraciones, además ayudan al refinamiento de clases. Drupal no utiliza la programación orientada a objetos, en cuanto a la falta de clases explícitamente declaradas, pero en su diseño se reflejan muchos paradigmas de la POO como objetos, abstracción, encapsulamiento, polimorfismo y herencia. Sin embargo las tarjetas CRC también son muy útiles para mejorar la idea de la arquitectura de Drupal, permitiendo conocer las responsabilidades y la distribución de los módulos que serán desarrollados. Para adecuar las tarjetas al módulo de esta trabajo se adopta lo siguiente: cada tarjeta será un módulo, las responsabilidades serán las funciones que realiza y sus colaboradores los métodos propios del módulo.

Tabla 5 Tarjeta CRC. Módulo Solicitud.

Módulo: Solicitud	
Responsabilidades	Colaboraciones
Crear solicitud	solicitud_crearprueba_form
Modificar solicitud	solicitud_crearprueba_form_validate
Eliminar solicitud	solicitud_crearprueba_form_submit
Ver solicitud	solicitud_creareva_form
Listar solicitudes	solicitud_creareva_form_validate
Aceptar solicitud	solicitud_creareva_form_submit
Rechazar solicitud	solicitud_crearase_form
Cerrar solicitud	solicitud_crearase_form_validate
Filtrar solicitudes	solicitud_crearase_form_submit
Exportar a PDF	solicitud_modificar_form
	solicitud_modificar_form_validate
	solicitud_modificar_form_submit
	solicitud_eliminar_form
	solicitud_eliminar_form_submit
	solicitud_ver_form
	solicitud_veruna_form
	solicitud_aceptada_form
	solicitud_aceptada_form_validate
	solicitud_aceptada_form_submit
	solicitud_rechazada_form
	solicitud_rechazada_form_submit
	solicitud_cerrada_form
	solicitud_cerrada_form_submit
	solicitud_reportes
	solicitud_reportes_form
	solicitud_reportes_form_submit
	solicitud_pdf_form

2.4.4 Arquitectura de Drupal

El secreto de Drupal para conseguir su reconocida flexibilidad y facilidad en la creación de sitios web es la abstracción y organización en capas que aplica en el tratamiento de los contenidos. En lugar de considerar el sitio web como un conjunto de páginas interrelacionadas, Drupal estructura los contenidos en una serie de elementos básicos. Estos son los nodos, módulos, bloques, menús, permisos de usuario y plantillas. (31)

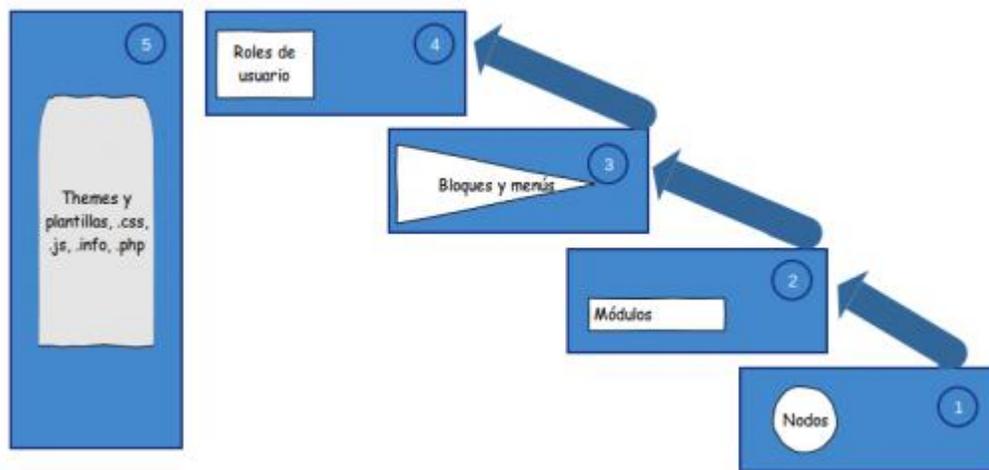


Figura 6 Arquitectura de Drupal.

- ✓ **Nodo:** es una unidad básica de información para los contenidos. Los nodos no dejan de ser una abstracción sobre una serie de tablas de la base de datos que Drupal utiliza para la estructura del sitio.
- ✓ **Módulos:** son los elementos que operan sobre los nodos y otorgan funcionalidad a Drupal, permitiendo incrementar sus capacidades o adaptarlas a las necesidades de cada sitio web.
- ✓ **Bloques y menús:** permiten estructurar y organizar los contenidos del sitio web. Es decir, que son los elementos que albergan y permiten acceder al usuario a la salida generada y procesada por los módulos a partir de la información almacenada en los nodos.

- ✓ **Permisos de usuarios:** con esta finalidad Drupal dispone de un registro de usuarios y de roles que permiten especificar que tareas pueden realizar y a que contenidos puede acceder cada tipo de usuario.
- ✓ **Plantilla:** establece la apariencia gráfica o estilo de la información que se le muestra al usuario. La plantilla es un conjunto de ficheros de configuración para crear la vista del contenido y su publicación.

2.4.5 Patrón arquitectónico Modelo Vista Controlador (MVC) en Drupal

El patrón de arquitectura Modelo Vista Controlador (MVC) permite realizar la programación multicapa, separando en tres componentes distintos los datos de una aplicación, la interfaz del usuario y la lógica de control. A continuación se detallan los tres componentes en que se divide (32):

- ✓ **Modelo:** representa la información con la que trabaja la aplicación, o sea, su lógica de negocio.
- ✓ **Vista:** convierte el modelo en una página web que facilita al usuario interactuar con ella.
- ✓ **Controlador:** es el encargado de procesar las interacciones del usuario y ejecuta los cambios adecuados en el modelo o en la vista.

Drupal concibe en su esencia una fuerte separación entre los progresos de la interacción directa con la base de datos (modelo), la interfaz gráfica (vista) y la interacción entre administración de los datos y su representación (controlador). Un desarrollador debe saber que cuando desarrolla para Drupal ya está desarrollando en el MVC. Aunque no esté haciendo la separación de sus archivos modelo, vista o controlador en diferentes carpetas, se están programando por separado estos aspectos. (33) Debido a que cuando se va a implementar un módulo, el modelo se desarrolla en el archivo `.install`, la vista en los archivos `.tpl` y el controlador mediante los archivos `.module` y `.inc`. El archivo `.install` facilita la instalación, actualización y desinstalación de las tablas de Drupal en la base de datos. Los archivos `.tpl` proporcionan el diseño de una interfaz gráfica para la aplicación web. Mientras que el archivo `.module` junto con los `.inc` son los encargados de la lógica del sistema, realizando peticiones a la base de datos y asignando el diseño a las interfaces.

2.4.6 Patrones de diseño de Drupal

Los patrones de diseño representan soluciones a problemas que surgen cuando se desarrolla un programa en un contexto particular. Capturan la estructura estática y dinámica, y la colaboración entre los participantes claves del diseño del *software*, además, facilitan la reutilización de arquitecturas y diseños de programas exitosos. (34)

Patrones de diseño GRASP

Existen los patrones GRASP que representan los principios básicos de la asignación de responsabilidades a objetos, expresados en forma de patrones. GRASP es el acrónimo para *General Responsibility Assignment Software Patterns* (Patrones Generales de *Software* para Asignar Responsabilidades). (32)

Patrones de diseño GoF

Mientras que los patrones GoF (*Gang of Four*, en español Pandilla de los Cuatro, formada por Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides) se clasifican en 3 categorías basadas en su propósito: creacionales, estructurales y de comportamiento. (32)

Muchos de los patrones GoF que se detallan en el libro *Design Patterns: Elements of Reusable Object-Oriented Software*, se pueden observar en Drupal. A continuación se explican algunos patrones utilizados en la implementación del módulo (35):

Singleton

El patrón Singleton garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. Si se piensa en los módulos y los temas como objetos, estos no encapsulan los datos, lo que separa a un módulo a otro es el conjunto de funciones que contiene, por lo que debe ser considerado como una clase con una instancia.

Decorador

Drupal hace un uso extensivo del patrón decorador. Más interesante es el uso de `hook_nodeapi()`, que permite a los módulos extender arbitrariamente el comportamiento de todos los nodos. Esta característica

permite una amplia variedad de comportamientos que se añaden a los nodos sin la necesidad de subclases.

Observador

El patrón Observer es un fenómeno generalizado en toda Drupal, ya que muchos de los ganchos, esencialmente permiten que los módulos puedan registrarse como observadores de los objetos de Drupal. Por ejemplo, cuando se realiza una modificación a un vocabulario en el sistema de taxonomía, un gancho de la taxonomía como `hook_taxonomy_vocabulary_update()` se llama en todos los módulos que lo implementan. Al implementar el gancho, los módulos se han inscrito como observadores del objeto de vocabulario, cualquier cambio a esto puede entonces actuar como es apropiado.

Puente

La base de datos de Drupal, capa de abstracción se lleva a cabo de una manera similar a la del patrón de diseño Puente. Los módulos necesitan ser escritos en una forma que es independiente del sistema que se está utilizando en la base de datos, y proporciona la capa de abstracción para ello. La nueva capa de base de datos se puede escribir que conforme la API definida por el puente, añadiendo soporte para más sistemas de bases de datos sin la necesidad de modificar el código del módulo.

Cadena de Responsabilidad

El sistema de menús de Drupal sigue el patrón Cadena de responsabilidades. En cada solicitud de página, el sistema de menú determina si hay un módulo para atender la solicitud, si el usuario tiene acceso al recurso solicitado, y qué función se llama para hacer el trabajo. Para hacer esto, un mensaje se transmite al elemento de menú correspondiente a la trayectoria de la solicitud. Si el elemento de menú no puede atender la solicitud, que se transmite por la cadena. Esto continúa hasta que un módulo se encarga de la solicitud, un módulo deniega el acceso al usuario, o la cadena se ha agotado.

Comando

Muchos de los ganchos de Drupal utilizan el patrón Comando para reducir el número de funciones que son necesarias para la aplicación, pasando por la operación como un parámetro, junto con los

argumentos. De hecho, el propio sistema de gancho utiliza este modelo, a fin de que los módulos no tienen que definir cada gancho, sino sólo los que vaya aplicar.

2.5 Conclusiones parciales

- ✓ Se describió la solución que se provee con el Módulo Solicitud de servicios del Sistema de Gestión de la Calidad del centro FORTES.
- ✓ Las fases de Planificación y Diseño posibilitaron la confección de las historias de usuarios, la planificación y estimación de estas, diseño de las tablas para la base de datos y tarjetas CRC.
- ✓ La metodología XP permitió minimizar el tiempo de trabajo, en cuanto a reducir la cantidad y complejidad de los artefactos a modelar.

Capítulo 3. Codificación y prueba

3.1 Introducción

La descripción del sistema, los artefactos generados y presentados en el capítulo anterior, ayudan al programador a entender las funcionalidades que necesita el cliente, y por ende, a ser capaz de implementar el sistema.

La metodología XP gestiona las fases de Codificación y Prueba, que fomentan prácticas como: el cliente es una parte más del equipo de desarrollo, la codificación debe hacerse atendiendo a estándares de codificación ya creados, la programación en pareja y crear pruebas que comprueben el funcionamiento de los distintos códigos implementados.

3.2 Codificación

Codificación es la fase de la metodología XP donde el cliente es una parte más del equipo de desarrollo. A la hora de codificar una historia de usuario su presencia es aún más necesaria debido a que los clientes son los que crean las HU y negocian los tiempos en los que serán implementadas. Además, la fase propone un modelo de desarrollo colectivo en el que todos los programadores están implicados en todas las tareas. (10)

3.2.1 Roles del sistema

Un punto crítico en los gestores de contenidos son los permisos y es importante disponer de algún mecanismo que permita establecer que operaciones podrá realizar cada usuario. El objetivo es evitar que, por accidente o de forma intencionada, un usuario pueda realizar operaciones que comprometan la integridad y funcionamiento de la herramienta web. Para este fin, Drupal utiliza un sistema de roles, permisos y usuarios. (36) Por tanto, con el objetivo de mantener la seguridad del módulo implementado se define los siguientes roles.

Tabla 6 Roles del sistema.

Roles	Justificación
Administrador	El administrador es el usuario encargado de realizar las funciones administrativas del sistema y tiene permisos ilimitados.
Planificador	Son los usuarios que el administrador asigna como planificador con permisos especiales en el sistema, puede gestionar una solicitud (crear, modificar, eliminar y ver), además, de administrarla (aceptar, rechazar, cerrar y realizar reportes).
Usuario autenticado	Es la persona que se encuentra autenticada en la aplicación web, que tendrá acceso dentro del módulo a crear, modificar, eliminar y ver las solicitudes.

3.2.2 Distribución física del sistema

La metodología XP no propone la generación del artefacto Modelo de Despliegue. Pero el presente trabajo considera necesario ilustrar la forma en que quedará distribuido el módulo para su despliegue. Además, es conveniente realizar algunas aclaraciones con el objetivo de mejorar su entendimiento. En la figura 6 se muestra la disposición física que tendrá el módulo una vez que se ponga en explotación.

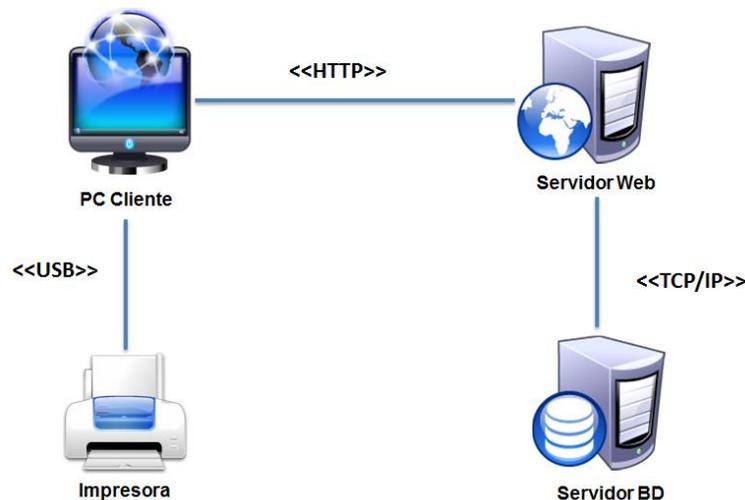


Figura 7 Distribución física del sistema.

Descripción de la capacidad que el dispositivo provee al sistema

Impresora: el sistema utiliza una impresora como dispositivo externo para su completo funcionamiento, que se conecta a la PC cliente. Este aporta la funcionalidad de impresión de los informes asociados al

reporte de las solicitudes. No se necesita de una impresora en específico, solo requiere estar activa y lista para imprimir cuando el usuario esté trabajando con el *software*.

Descripción de la funcionalidad y capacidad del nodo

PC Cliente: ordenador que se conecta a través de un navegador web al servidor central donde reside la aplicación. No se necesita tener instalado el producto localmente.

Servidor Central: se encuentra compuesto por un servidor web que tendrá instalado Apache 2.2 y es donde radica la aplicación web soportada por Drupal 6.20. Además, de un servidor de base de datos con PostgreSQL 8.4 y todos los componentes necesarios para el funcionamiento del producto. En la imagen anterior se muestran ambos servidores conectados a través del protocolo TCP/IP. La memoria RAM debe ser de al menos 512 MEGABYTES (MB).

Descripción de elementos e interfaces de comunicación

<<HTTP>>: representa la conexión entre el navegador del usuario y el servidor del sistema.

<<USB>>: conexión que existe entre la impresora y la PC Cliente del usuario.

3.2.4 Módulos de apoyo

Drupal está formado por un conjunto de módulos, que no son más que una colección de funciones que dotan de cierta funcionalidad al sistema. Los módulos se pueden clasificar en tres categorías:

- ✓ **Módulos del núcleo (Core modules):** son los módulos provistos por Drupal al instalarse, algunos de ellos fueron contribuciones de la comunidad de Drupal que se incorporaron. (37)
- ✓ **Módulos contribuidos (Contributed modules)** son los módulos que son compartidos para la comunidad de Drupal, están bajo GNU de Licencia Pública (GPL). (37)
- ✓ **Módulos personalizados (Custom modules):** son los módulos creados por el desarrollador del sitio. (37) En este caso se ejemplifica el Módulo Solicitud de servicio, el cual se implementa en el presente trabajo.

Módulos de apoyo

Para que la solución propuesta cumpliera con todos los requisitos previamente definidos por el cliente, se necesitó del soporte de algunos módulos de Drupal que se detallan a continuación.

Módulo LDAP: este módulo sirve de soporte al protocolo que lleva el mismo nombre, el cual se utiliza para garantizar la autenticación de los usuarios en el sistema mediante el dominio UCI, permitiendo obtener datos de forma confiable, que son necesarios para el correcto funcionamiento del módulo.

Módulo Date: el módulo Date se utiliza para facilitar la inserción de fechas con formatos predeterminados en la aplicación, permitiendo validarlas y reducir el número de errores cometidos por el usuario durante la inserción de datos.

3.2.5 Librerías de apoyo

Las librerías son útiles debido a que permiten agrupar varias funciones y variables en un mismo fichero, de manera que se puede incluir en distintas páginas y disponer de esas funciones fácilmente. Son un conjunto de subprogramas utilizados para desarrollar aplicaciones. Contienen código y datos, que proporcionan servicios a programas independientes. (38)El sistema utiliza las siguientes librerías de PHP.

Librería FPDF: es una clase desarrollada en PHP para poder realizar documentos en PDF. Permite modificar la unidad de medida, el formato de la página, los márgenes, las cabeceras y los pies de página, los saltos de línea, las imágenes, colores y enlaces. Además facilita establecer el orden de la información a mostrar, así como el formato que tendrá el documento. Es una librería gratuita para cualquier uso, tanto comercial como personal. (39)

Librería Mail: esta librería sirve de soporte al protocolo SMTP, el cual se utiliza para el envío de notificaciones por correo electrónico.

3.2.6 Estándares de codificación

Un estándar de codificación o estilo de programación es el conjunto de reglas o normas usadas para escribir código e incluye una gama de aspectos dentro del proceso de codificación. El uso de estilos de

programación facilita que los programas sean más fácil de leer y de comprender por otros programadores, además, se reduce el número de errores que se presentan durante el desarrollo del programa. (40)

Las reglas de codificación de Drupal se aplican al código dentro de Drupal y sus módulos de terceros, las cuales se basan libremente en las normas de codificación de PHP *Extension and Application Repository* (PEAR). El Módulo Solicitud de servicios aplica varias de las normas, que se explican a continuación (41):

Operadores

- ✓ Todos los operadores binarios, que se encuentran entre dos valores como +, -, =, ==, > deben tener un espacio antes y después del operador, para facilitar la lectura. Por ejemplo, \$foo = \$bar.
- ✓ Los operadores unarios, que operan en un solo valor, tales como ++, no debería tener un espacio entre el operador y el número variable. Ejemplo, \$a++.

Estructuras de control

- ✓ Las sentencias de control debe tener un espacio entre la palabra de control y paréntesis de apertura para distinguirlas de las llamadas a funciones.
- ✓ Siempre se utilizan las llaves, incluso en situaciones en las que son técnicamente opcionales. Aumentan la legibilidad y disminuye la probabilidad de errores lógicos que se introducen cuando las líneas se añaden nuevos elementos.
- ✓ No use else if utilice siempre elseif.

```
If (condition1 || condition2) {  
    action1;  
}  
elseif (condition3 && condition4) {  
    action2;  
}  
else {  
    defaultaction;  
}
```

Longitud de la línea y el ajuste

- ✓ En general, todas las líneas de código no deben ser superiores a 80 caracteres.

Matrices o arreglos

- ✓ Las matrices deben tener el formato con un espacio separando cada elemento después de la coma, y los espacios alrededor del operador => clave de la asociación.

```
$some_array = array('hello', 'world', 'foo' => 'bar');
```

- ✓ Tenga en cuenta que si en la línea se declara una matriz que se extiende por más de 80 caracteres, cada elemento debe ser dividida en su propia línea y sangría.

```
$form['title'] = array(  
  '#type' => 'textfield',  
  '#title' => t('Title'),  
  '#size' => 60,  
  '#maxlength' => 128,  
  '#description' => t('The title of your node.'),  
);
```

- ✓ Tenga en cuenta la coma al final del último elemento del *array*, no es un error. Esto ayuda a evitar errores de análisis, si otro elemento se coloca al final de la lista más adelante.

Comillas

Drupal no tiene un duro estándar para el uso de comillas simples contra comillas dobles. Siempre que sea posible, mantener la coherencia dentro de cada módulo, y respetar el estilo personal de otros desarrolladores. Con esta advertencia en mente, las cadenas de comillas simples son conocidas por ser más rápidas debido a que el analizador no tiene que buscar en línea las variables. Su uso se recomienda excepto en dos casos:

- ✓ En la línea de uso variable, por ejemplo, "<h2> \$header </ h2>".

- ✓ Cadenas traducidas, donde se puede evitar el escape de comillas simples encerrando la cadena entre comillas dobles. Una cadena, sería "Él es una buena persona."

Concatenaciones de cadenas

- ✓ Utilice siempre un espacio entre el punto y las partes concatenadas para mejorar la legibilidad.

```
$string = 'Foo' . $bar;
```

```
$string = $bar . 'foo';
```

```
$string = 'foo' . 'bar';
```

- ✓ Al concatenar variables simples puede utilizar comillas dobles y añadir la variable dentro, de lo contrario, utilice comillas simples.

```
$string = "Foo $bar";
```

- ✓ Cuando se utiliza el operador de asignación de concatenación (.=), utilice un espacio a cada lado como con el operador de asignación:

```
$string .= 'Foo';
```

```
$string .= $bar;
```

PHP etiquetas de código

- ✓ A partir de Drupal 4.7, la etiqueta (?>) al final de los archivos de código, está omitido. Esto se incluye para módulos y los archivos de inclusión.

3.3 Prueba

Uno de los pilares de la metodología XP es el uso de pruebas para comprobar el funcionamiento de los códigos que se está implementando. Para asegurar el funcionamiento final de una determinada historia de usuario se deben crear pruebas de aceptación. Estas pruebas son creadas y usadas por los clientes para comprobar que las distintas historias de usuario cumplen su cometido.

3.3.1 Casos de prueba (CP)

En las tablas de las Pruebas de Aceptación, se incluyen los siguientes campos: número de caso de prueba (CP), que contiene el identificador de caso de prueba; nombre y número de la historia de usuario (HU); nombre del CP; descripción de la prueba, que contiene una breve descripción de la prueba realizada; condiciones de ejecución, se incluyen las condiciones necesarias para que se pueda realizar la prueba; entrada o pasos de ejecución, contiene una serie de pasos enumerados para lograr realizar la prueba de esta HU; resultado esperado, contiene la descripción de lo que se espera luego de realizar la prueba; y evaluación de la prueba, muestra si la prueba fue satisfactoria o insatisfactoria.

A continuación se incluyen la tabla 7, que contienen uno de los casos de pruebas de aceptación de la HU Crear solicitud de servicio, clasificada con prioridad alta, o sea, arquitectónicamente significativa para el sistema. El resto se detallan en el [Anexo 3](#).

Tabla 7 CP Crear solicitud de Prueba.

Caso de Prueba	
Número de CP: 1	HU 1: Crear solicitud de servicio
Nombre: Crear solicitud de prueba.	
Descripción: Se insertan los datos de un servicio de prueba, inicialmente se insertarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> El usuario debe estar autenticado o tener el rol de Planificador. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> Para crear una solicitud de prueba se debe seleccionar del bloque Solicitud la opción: <u>Crear solicitud de Prueba/Guardar</u>, o la opción <u>Gestionar solicitud/Crear solicitud de Prueba/Guardar</u>. Dejar los campos en blanco. Insertar en el campo nombre del proyecto números y comenzar con letra minúscula. Insertar en el campo jefe de proyecto números, comenzar con letra minúscula y solo ponga el nombre y el primer apellido. No seleccione ninguna metodología. Insertar en el campo fase números y comenzar con letra minúscula. Insertar en el campo etapa números y comenzar con letra minúscula. Insertar en el campo rol números y comenzar con letra minúscula. No seleccione ningún tipo de prueba. Insertar en el campo nombre del producto números y comenzar con letra minúscula. Insertar en el campo artefacto números y comenzar con letra minúscula. Insertar en el campo tipo de artefacto números y comenzar con letra minúscula. No seleccione ningún tipo de documentación. Insertar los datos correctos. Verificar que la nueva solicitud fue almacenada. Ir al <u>CP 7: Listar solicitudes</u>. 	
Resultado esperado:	
<ul style="list-style-type: none"> El sistema debe alertar al usuario cuando se deje campos en blancos, se comience a escribir con minúscula o se escriban números. Cuando se inserten los datos correctamente, el sistema debe almacenarlos en la base de datos y mostrarlos en el listado <u>CP 7: Listar solicitudes</u>. 	
Evaluación de prueba: Satisfactoria.	

3.3.2 Resultado de las pruebas

Pruebas de Aceptación

Las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada. Las pruebas de aceptación son consideradas como pruebas de caja negra. Los clientes son responsables de verificar que los resultados de estas pruebas sean correctos. Asimismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución. Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. (42)

Con el objetivo de asegurar el funcionamiento final de las historias de usuario, se realiza las pruebas de aceptación por parte del cliente. El cual evaluó de forma satisfactoria el 100% de los casos de prueba.

Pruebas de Funcionalidad

Ante la necesidad de validar el módulo implementado y obtener un sistema funcional desplegable, en el Grupo de Calidad, se propuso por parte del cliente, que al producto se le realizara pruebas de funcionalidad con la técnica de caja negra con el equipo de prueba del Grupo de Calidad. Durante la primera iteración del proceso se encontraron cuatro no conformidades, tres de prioridad media y una recomendación, las cuales fueron resueltas. En la segunda iteración se detectaron dos no conformidad, una de prioridad alta y otra de prioridad media, también resueltas. En la tercera iteración no se detectaron no conformidades. Se obtuvo como resultado, la evaluación de todos los casos de pruebas, libres de errores.

Tabla 8 Resumen de las no conformidades.

Iteración	Ortográficas	Validación	Funcional	Prioridad
1				Alta
	1	2		Media
				Baja
		1		Recomendación
Total	1	3	0	4
2			1	Alta
	1			Media
				Baja
				Recomendación
Total	1	0	1	2
3				Alta
				Media
				Baja
				Recomendación
Total	0	0	0	0

3.4 Conclusiones parciales

- ✓ Se incluyeron y explicaron los artefactos generados por la metodología XP para las fases de Codificación y Prueba, aplicados al Módulo Solicitud.
- ✓ Se validó el sistema propuesto mediante la realización de pruebas de aceptación a las historias de usuarios, a través de dos iteraciones, donde las pruebas fueron evaluadas de satisfactoria por parte del cliente.

Conclusiones generales

Después de finalizar la investigación y constatar los resultados, se concluye que:

- ✓ El análisis de las características de los procesos de gestión de solicitudes, teniendo en cuenta el procedimiento definido en el Grupo de Calidad del centro FORTES, posibilitó el diseño de una solución que se adecua a las necesidades del cliente.
- ✓ El análisis de sistemas informáticos demostró, que algunas herramientas son privativas y el resto no cumple con los requerimientos expresados por el cliente en las historias de usuario.
- ✓ Mediante la aplicación de la metodología ágil XP se logró efectividad y rapidez en el desarrollo de *software*.
- ✓ Se obtuvo un producto funcional, probado y validado durante su proceso de desarrollo utilizando pruebas de aceptación, donde las pruebas fueron evaluadas de satisfactoria por parte del cliente.

Recomendaciones

- ✓ Agregar nuevas funcionalidades que permitan realizar reportes que tributen a mejorar el proceso de planificación del Grupo de Calidad del centro FORTES.
- ✓ Proponer la herramienta a otros centros productivos, con el objetivo de garantizar una mejor recepción y planificación de los servicios en los Grupos de Calidad de la Universidad de las Ciencias Informáticas.

Referencias bibliográfica

1. Bienvenidos al Redmine del Centro FORTES. [En línea] UCI, 2010-2012. [Citado el: 15 de Mayo de 2012.] <http://portal.fortes.prod.uci.cu>.
2. Definición de servicio. [En línea] [Citado el: 15 de Mayo de 2012.] <http://definicion.de/servicio>.
3. **T., I.** Definición de Servicios. [En línea] Agosto de 2006. [Citado el: 15 de Mayo de 2012.] <http://www.promonegocios.net/mercadotecnia-servicios/definicion-servicios.html>.
4. Definición de solicitud. [En línea] [Citado el: 15 de Mayo de 2012.] <http://definicion.de/solicitud>.
5. **T., G.** La Solicitud (Concepto, Características, Estructura, Ejemplo). [En línea] 2009. [Citado el: 15 de Mayo de 2012.] <http://nuestraenciclopedia.blogspot.com/2009/10/la-solicitud-concepto-caracteristicas.html>.
6. Diccionario de la Real Academia Española. Información. [En línea] 29 de octubre de 2008. [Citado el: 17 de Mayo de 2012.] http://buscon.rae.es/draeI/SrvltConsulta?TIPO_BUS=3&LEMA=informaci%C3%B3n.
7. **T.I.** Definición de información. [En línea] Octubre de 2008. [Citado el: 17 de Mayo de 2012.] <http://www.promonegocios.net/mercadotecnia/definicion-informacion.html>.
8. **Bartle, Phil.** Información para la gestión y gestión de la información. [En línea] CEC Community Empowerment Collective, 10 de Enero de 2011. [Citado el: 17 de Mayo de 2012.] <http://www.scn.org/mpfc/modules/mon-miss.htm>.
9. http://www.icesi.edu.co/wiki/index.php/Manual_solicitud_servicios_apoyo. [En línea] 9 de Noviembre de 2010. [Citado el: 13 de Junio de 2012.] http://www.icesi.edu.co/wiki/index.php/Manual_solicitud_servicios_apoyo.
10. **Castillo Oswaldo, Figueroa Daniel, Sevilla Hector.** Fases de la Programación Extrema. [En línea] [Citado el: 20 de Abril de 2012.] <http://programacionextrema.tripod.com/fases.htm>.
11. **INFORMATICA, COLECCION CULTURA.** Herramientas Case - Inei. [En línea] Noviembre de 1999. [Citado el: 6 de Junio de 2012.] <http://www.inei.gob.pe/biblioineipub/bancopub/Inf/Lib5103/Libro.pdf>.
12. Visual Paradigm for UML - Herramienta de UML para el desarrollo de software de aplicación. [En línea] 2010. [Citado el: 15 de Enero de 2011.] <http://www.visual-paradigm.com/product/vpuml>.
13. Visual Paradigm. [En línea] [Citado el: 20 de Abril de 2012.] http://ucipedia.uci.cu/index.php/Visual_Paradigm.

14. Rational Rose Enterprise. [En línea] [Citado el: 15 de Enero de 2011.]
<http://www.rational.com.ar/herramientas/roseenterprise.html>.
15. Introducción a Rational Rose. [En línea] [Citado el: 15 de Enero de 2011.]
<http://www.google.com.cu/url?sa=t&source=web&cd=5&ved=0CDAQFjAE&url=http%3A%2F%2Fwww.essi.upc.edu%2F~es-e%2Fweb%2Fdocuments%2Ftab%2F0304Q2%2Flessons%2Flese-2%2FLESE-2%2520-%2520Introduccion%2520a%2520Rational%2520Rose.ppt>.
16. **Merca-Tech®**. Ventajas de Usar un CMS. [En línea] [Citado el: 13 de Junio de 2012.]
<http://www.merca-tech.com.mx/articulos/47-ventajas-de-usar-un-cms.html>.
17. Drupal. [En línea] [Citado el: 15 de enero de 2011.] <http://ucipedia.uci.cu/index.php/Drupal>.
18. 27 Características de DRUPAL. [En línea] 2009. [Citado el: 15 de Enero de 2011.]
<http://www.pilos.com.co/drupal/27-caracteristicas-de-drupal>.
19. PostgreSQL. [En línea] http://danielpecos.com/docs/mysql_postgres/x15.html.
20. PostgreSQL Global Development Group. PostgreSQL: The world's most advanced open source database. [En línea] [Citado el: 15 de Enero de 2012.] <http://www.postgresql.org>.
21. Arquitectura del servidor Apache. [En línea] [Citado el: 15 de Enero de 2012.]
<http://www.desarrolloweb.com/articulos/1112.php>.
22. Qué es PHP. [En línea] 2001. [Citado el: 13 de Junio de 2012.]
<http://www.desarrolloweb.com/articulos/392.php>.
23. Manual XHTML. [En línea] [Citado el: 10 de Junio de 2012.] <http://manual-xhtml.blogspot.com/2006/05/primer-documento-xhtml.html#L3060>.
24. ECURED. CSS. [En línea] [Citado el: 10 de Junio de 2012.] <http://www.ecured.cu/index.php/CSS>.
25. Bienvenido a NetBeans y www.netbeans.org. [En línea] [Citado el: 15 de Enero de 2011.]
http://netbeans.org/index_es.html.
26. **Cerda, Felipe**. NetBeans. El único IDE que necesitas. [En línea] [Citado el: 15 de Enero de 2011.]
<http://www.aeap.es/ficheros/be6241eaf9a61dcf6f169d49c79808e4.pdf>.
27. Eclipse. [En línea] [Citado el: 15 de Enero de 2011.] <http://plataformaclipse.com>.
28. Eclipse. [En línea] [Citado el: 15 de Enero de 2011.] <http://ucipedia.uci.cu/index.php/Eclipse>.
29. UML - Diagramas de estados. [En línea] [Citado el: 10 de Mayo de 2012.]
<http://jms32.eresmas.net/tacticos/UML/UML08/UML0801.html>.

30. **Holmberg, Anders y IAR Systems.** Diseño de una máquina de estados para resolver un problema. [En línea] [Citado el: 10 de Mayo de 2012.] <http://www.convertronic.net/Diseno/disenode-una-maquina-de-estados-para-resolver-un-problema.html>.
31. Tutorial de Drupal. [En línea] [Citado el: 5 de Junio de 2012.] <http://www.cursosdrupal.com/content/arquitectura>.
32. ECURED. Patrones de diseño y arquitectura. [En línea] [Citado el: 14 de Junio de 2012.] http://www.ecured.cu/index.php/Patrones_de_dise%C3%B1o_y_arquitectura.
33. KOALASOFT. [En línea] [Citado el: 15 de Junio de 2012.] <http://www.koala-soft.com/drupal>.
34. **Martínez, Prof. Ivette C.** CLASE 9: DISEÑO CON PATRONES. [En línea] [Citado el: 14 de Junio de 2012.] http://ldc.usb.ve/~martinez/cursos/ci3715/clase9_AJ2010.pdf.
35. DrupalTM. [En línea] 5 de Febrero de 2012. [Citado el: 14 de Junio de 2012.] <http://drupal.org/node/547518>.
36. Tutorial de Drupal. [En línea] [Citado el: 13 de Junio de 2012.] <http://www.cursosdrupal.com/content/usuarios-roles-y-permisos>.
37. ECURED. [En línea] [Citado el: 5 de Junio de 2012.] <http://www.ecured.cu/index.php/Drupal>.
38. OpenAlfa Soluciones Open Source. [En línea] [Citado el: 13 de Junio de 2012.] <http://www.openalfa.com/index.php/php/funciones-librerias>.
39. FPDF Library PDF Generator. [En línea] [Citado el: 14 de Junio de 2012.] <http://www.fpdf.org/>.
40. **Cornejo, José Enrique González.** ACERCA DEL ESTILO EN PROGRAMACION. [En línea] 18 de Abril de 2009. [Citado el: 14 de Junio de 2012.] http://www.docirs.cl/acerca_del_estilo_programacion.htm.
41. DrupalTM. [En línea] 23 de Abril de 2012. [Citado el: 14 de Junio de 2012.] <http://drupal.org/coding-standards>.
42. **Joskowicz, Ing. José.** *Reglas y Prácticas en eXtreme Programming*. [Documento PDF] 2008.

Glosario de términos

CASE (Computer Automated System Engineering): conjunto de métodos, utilidades y técnicas que facilitan la automatización del ciclo de vida del desarrollo de sistemas de información, completamente o en alguna de sus fases.

Flexibilidad: es la capacidad que tiene un objeto o cosa de adaptar a nueva situación. Dentro del mundo del deporte, entendemos por flexibilidad la cualidad que tiene los músculos de estirarse a fin de adaptarse a un nuevo rango de amplitud de movimientos.

GNU/GPL (Licencia Pública General): es una licencia creada por la *Free Software Foundation* en 1989 y está orientada principalmente a proteger la libre distribución, modificación y uso de programas. Su propósito es declarar que el software cubierto por esta licencia es libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

Interoperabilidad: capacidad de los sistemas de tecnologías de la información y las comunicaciones (TIC), y de los procesos empresariales a los que apoyan, de intercambiar datos y posibilitar la puesta en común de información y conocimientos.

LDAP (Protocolo Ligero de Acceso a Directorios): es un protocolo de tipo cliente-servidor para acceder a un servicio de directorio. Se usó inicialmente como un Front-end o interfaz final para x.500, pero también puede usarse con servidores de directorio únicos y con otros tipos de servidores de directorio.

PEAR (PHP Extension and Application Repository): es un entorno de desarrollo y sistema de distribución para componentes de código PHP. Guía de estilo dirigido a programadores que implementan en PHP.

Plataforma de cliente rico: es un entorno de administración del ciclo de vida de una aplicación, la cual constituye una base para el desarrollo de aplicaciones de escritorio.

SOA (Arquitectura Orientada a Servicios): es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requisitos del negocio. Permite la creación de sistemas altamente escalables que reflejan el negocio de la organización, a su vez brinda una forma bien definida de exposición e invocación de servicios (comúnmente pero no exclusivamente servicios web), lo cual facilita la

interacción entre diferentes sistemas propios o de terceros.

SQL92/SQL99 (lenguaje de consulta estructurado): es un lenguaje declarativo de acceso a base de datos relacionales que permite especificar diversos tipos de operaciones en éstas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo efectuar consultas con el fin de recuperar de una forma sencilla información de interés de una base de datos, así como también hacer cambios sobre ella.

TIC (Tecnologías de la Información y las Comunicaciones): agrupan los elementos y las técnicas utilizadas en el tratamiento y la transmisión de las informaciones, principalmente de informática, internet y telecomunicaciones.

UML (Lenguaje Unificado de Modelado): es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

Web 2.0: es la transición que se ha dado de aplicaciones tradicionales hacia aplicaciones que funcionan a través de las web enfocadas al usuario final. Se trata de aplicaciones que generen colaboración y de servicios que reemplacen las aplicaciones de escritorio.

World Wide Web (o Web): es básicamente un medio de comunicación de texto, gráficos y otros objetos multimedia a través de Internet, es decir, la web es un sistema de hipertexto que utiliza Internet como su mecanismo de transporte o desde otro punto de vista, una forma gráfica de explorar Internet.

W3C (World Wide Web Consortium): un consorcio fundado en 1994 para dirigir a la Web hacia su pleno potencial mediante el desarrollo de protocolos comunes que promuevan su evolución y aseguren su interoperabilidad. Está dirigida por Tim Berners-Lee, el creador original de URL (*Uniform Resource Locator*, Localizador Uniforme de Recursos), HTTP (*HyperText Transfer Protocol*, Protocolo de Transferencia de HiperTexto) y HTML (Lenguaje de Marcado de HiperTexto) que son las principales tecnologías sobre las que se basa la Web.

Anexos

Anexo 1 Plantilla de solicitud para el servicio de Prueba del Grupo de Calidad del centro FORTES.



Plantilla de Solicitud del Servicio de Pruebas de Software

Con el objetivo de ordenar las solicitudes que gestiona el Grupo de Calidad De FORTES e implementar con objetividad y eficiencia el servicio de pruebas definido para el servicio se propone la siguiente plantilla.
Se recomienda llenar todos los campos. Marque con una cruz solo cuando sea necesario y en caso de que deba responder ponga, sí o no.

Datos del Proyecto:				
Nombre del Proyecto:				
Nombre del Jefe de Proyecto:			Usuario:	
Nombre de quién solicita el servicio:			Usuario:	
Responsable por el Equipo de Proyecto:			Usuario:	
Información del Producto:				
Nombre del Producto:				
Tipo de Prueba que Solicita:				
Dirección URL de la Información:				
Dirección URL de la Aplicación:				
Artefacto	Tipo de Artefacto	Versión	Cant. de Páginas	Cant. de Módulos
Información de la Prueba				
TIPO DE PRUEBA:				
OBJETIVO DE LA PRUEBA:				
Documentación				
NOTA: Para la aprobación o no de la presente solicitud, el Equipo de Desarrollo debe proveer la documentación correspondiente para su revisión, garantizando siempre el acceso a la misma así como también su corrección, completitud y actualización. De la siguiente columna, seleccione con una cruz el tipo de documentación que será utilizada durante la prueba.			Lista de Chequeo:	
			Casos de Prueba:	
			Casos de Uso:	
			Guión técnico:	
Estado	Aprobada:	Rechazada:		
Fundamentación:				
NOTA: En caso de que no haya documentación o la documentación entregada desde el inicio sea obsoleta, el GCF no se responsabiliza por las NC que se informen, si estas no fueron debidamente aclaradas por parte del Equipo de Desarrollo durante la realización del servicio, así mismo se aborta la prueba por la no presencia de los desarrolladores.				

Figura 8 Plantilla de solicitud para el servicio de prueba del Grupo de Calidad del centro FORTES.

Anexo 2 Historias de usuario.

No.: 1		Nombre: Crear solicitud de servicio	
Usuario: Usuario autenticado, planificador y administrador			
Modificación de historia número: 0		Iteración asignada: 1	
Prioridad en negocio: Alta		Puntos estimados: 4 días	
Riesgo en desarrollo: Alto		Punto reales: 4 días	
Descripción:	<p>El usuario ingresa los datos necesarios para crear una nueva solicitud de un servicio y selecciona la opción de Guardar. El sistema valida y guarda la información en la base de datos, al finalizar muestra un mensaje de confirmación. En caso que exista datos incorrectos, se muestra un mensaje de error.</p> <p>Si selecciona la opción de Cancelar, el sistema borra los datos introducidos y re-direcciona a la misma página.</p>		
Observaciones:	<ul style="list-style-type: none"> - El usuario debe estar autenticado. - Apenas que se cree una nueva solicitud, esta toma el estado Pendiente. 		

No.: 2		Nombre: Modificar solicitud de servicio	
Usuario: Usuario autenticado, planificador y administrador			
Modificación de historia número: 0		Iteración asignada: 1	
Prioridad en negocio: Alta		Puntos estimados: 3 días	
Riesgo en desarrollo: Alto		Punto reales: 3 días	
Descripción:	<p>El usuario selecciona la opción de modificar una solicitud de una lista. Introduce los datos que desea modificar y selecciona la opción de Actualizar. El sistema valida y guarda la información en la base de datos, muestra un mensaje de confirmación. En caso que exista datos incorrectos, se muestra un mensaje de error.</p> <p>Si selecciona la opción de Cancelar, el sistema mantiene los datos que se encuentra almacenados en el sistema de la solicitud y re-direcciona a la misma página.</p>		
Observaciones:	<ul style="list-style-type: none"> - El usuario debe estar autenticado. - El usuario debe ser el autor de la solicitud a modificar. - La solicitud debe tener el estado Pendiente para ser modificada. 		

No.: 3		Nombre: Eliminar solicitud de servicio	
Usuario: Usuario autenticado, planificador y administrador			
Modificación de historia número: 0		Iteración asignada: 1	
Prioridad en negocio: Alta		Puntos estimados: 3 días	
Riesgo en desarrollo: Alto		Punto reales: 3 días	
Descripción:	<p>El usuario selecciona la opción de eliminar una solicitud de una lista. El sistema permite confirmar la acción, si el usuario selecciona Eliminar, los datos se eliminan de la base de datos y se muestra un mensaje de información.</p> <p>Si el usuario selecciona Cancelar, se mantiene la información y el sistema lleva al usuario a la página anterior (HU 5: Listar solicitudes).</p>		
Observaciones:	<ul style="list-style-type: none"> - El usuario debe estar autenticado. - El usuario debe ser el autor de la solicitud a eliminar. - La solicitud debe tener el estado Pendiente para ser eliminada. 		

No.: 4		Nombre: Ver solicitud de servicio	
Usuario: Usuario autenticado, planificador y administrador			
Modificación de historia número: 0		Iteración asignada: 1	
Prioridad en negocio: Alta		Puntos estimados: 1 día	
Riesgo en desarrollo: Bajo		Punto reales: 1 día	
Descripción:	<p>El usuario selecciona de una lista la opción Ver una determinada solicitud. El sistema muestra los datos y le permite al usuario modificar, eliminar o exportar a PDF la información de dicha solicitud y regresar atrás.</p>		
Observaciones:	<ul style="list-style-type: none"> - El usuario debe estar autenticado. 		

No.: 5		Nombre: Listar solicitudes	
Usuario: Usuario autenticado, planificador y administrador			
Modificación de historia número: 0		Iteración asignada: 1	
Prioridad en negocio: Alta		Puntos estimados: 1 día	
Riesgo en desarrollo: Bajo		Punto reales: 1 día	
Descripción:	<p>El sistema muestra una lista de todas las solicitudes guardadas en el sistema. Además, se pueden realizar las acciones de modificar, eliminar, ver y exportar a PDF, cuando se localice en la sección Gestionar solicitud; si es un planificador en la sección de Administrar solicitud, el usuario podrá aceptar, rechazar, cerrar, ver y exportar a PDF una solicitud.</p> <p>Se tiene la opción atrás que permite ir a la página Gestionar solicitud o Administrar solicitud.</p>		
Observaciones:	<ul style="list-style-type: none"> - El usuario debe estar autenticado. 		

No.: 6 Nombre: Aceptar solicitud del servicio	
Usuario: Planificador y administrador	
Modificación de historia número: 0	Iteración asignada: 2
Prioridad en negocio: Media	Puntos estimados: 2 días
Riesgo en desarrollo: Alto	Punto reales: 2 días
Descripción:	El usuario selecciona de una lista la opción Ver una determinada solicitud. El sistema le permite seleccionar la opción Aceptar solicitud y asigna un responsable para realizar el servicio. El sistema envía una notificación por correo para informar a la persona de que su solicitud fue aceptada y planificada y al responsable que se encuentra a cargo de ejecutar el servicio. La solicitud toma el estado de Aceptada.
Observaciones:	-El usuario debe tener el rol de planificador o administrador. -Sólo puede aceptar una solicitud que se encuentre en estado Pendiente.

No.: 7 Nombre: Rechazar solicitud del servicio	
Usuario: Planificador y administrador	
Modificación de historia número: 0	Iteración asignada: 2
Prioridad en negocio: Media	Puntos estimados: 2 días
Riesgo en desarrollo: Alto	Punto reales: 2 días
Descripción:	El usuario selecciona de una lista la opción Ver una determinada solicitud. El sistema le permite seleccionar la opción Rechazar la solicitud y justificar la razón del estado seleccionado. El sistema envía una notificación por correo para informar a la persona de que su solicitud fue rechazada. La solicitud toma el estado de Rechazada.
Observaciones:	-El usuario debe tener el rol de planificador o administrador. -Sólo puede aceptar una solicitud que se encuentre en estado Pendiente.

No.: 8 Nombre: Cerrar solicitud del servicio	
Usuario: Planificador y administrador	
Modificación de historia número: 0	Iteración asignada: 2
Prioridad en negocio: Media	Puntos estimados: 2 días
Riesgo en desarrollo: Alto	Punto reales: 2 días
Descripción:	Después que el equipo de trabajo termine de ejecutar el servicio, el planificador a través del listar solicitud selecciona la acción Cerrar solicitud, si se desea se comenta el cambio de estado. La solicitud toma el estado de Cerrada.
Observaciones:	-El usuario debe tener el rol de planificador o administrador. -Sólo puede cerrar una solicitud que se encuentre en estado Aceptada o Rechazada.

No.: 9		Nombre: Realizar reportes a las solicitudes de servicios	
Usuario: Planificador y administrador			
Modificación de historia número: 0		Iteración asignada: 2	
Prioridad en negocio: Media		Puntos estimados: 2 días	
Riesgo en desarrollo: Alto		Punto reales: 3 días	
Descripción:	El sistema muestra todas las solicitudes existentes y permite al usuario realizar reportes a dichas solicitudes, filtrando por varios campos (proyecto, tipo de servicio, estado y fecha), exportar los resultados a formato PDF.		
Observaciones:	-El usuario debe tener el rol de planificador o administrador.		

No.: 10		Nombre: Exportar a PDF	
Usuario: Usuario autenticado, planificador y administrador			
Modificación de historia número: 0		Iteración asignada: 2	
Prioridad en negocio: Media		Puntos estimados: 2 días	
Riesgo en desarrollo: Bajo		Punto reales: 2 días	
Descripción:	El sistema permite exportar a PDF los resultados filtrados en forma de lista.		
Observaciones:	-El usuario debe estar autenticado.		

Anexo 3 Casos de prueba.

Caso de Prueba	
Número de CP: 1	HU 1: Crear solicitud de servicio
Nombre: Crear solicitud de prueba.	
Descripción: Se insertan los datos de un servicio de prueba, inicialmente se insertarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> El usuario debe estar autenticado o tener el rol de Planificador. 	
Entrada/Pasos de ejecución:	
<p>16. Para crear una solicitud de prueba se debe seleccionar del bloque Solicitud la opción: <u>Crear solicitud de Prueba/Guardar</u>, o la opción <u>Gestionar solicitud/Crear solicitud de Prueba/Guardar</u>.</p> <p>17. Dejar los campos en blanco.</p> <p>18. Insertar en el campo nombre del proyecto números y comenzar con letra minúscula.</p> <p>19. Insertar en el campo jefe de proyecto números, comenzar con letra minúscula y solo ponga el nombre y el primer apellido.</p> <p>20. No seleccione ninguna metodología.</p> <p>21. Insertar en el campo fase números y comenzar con letra minúscula.</p> <p>22. Insertar en el campo etapa números y comenzar con letra minúscula.</p> <p>23. Insertar en el campo rol números y comenzar con letra minúscula.</p> <p>24. No seleccione ningún tipo de prueba.</p> <p>25. Insertar en el campo nombre del producto números y comenzar con letra minúscula.</p> <p>26. Insertar en el campo artefacto números y comenzar con letra minúscula.</p> <p>27. Insertar en el campo tipo de artefacto números y comenzar con letra minúscula.</p> <p>28. No seleccione ningún tipo de documentación.</p> <p>29. Insertar los datos correctos.</p> <p>30. Verificar que la nueva solicitud fue almacenada. Ir al <u>CP 7: Listar solicitudes</u>.</p>	
Resultado esperado:	
<ul style="list-style-type: none"> El sistema debe alertar al usuario cuando se deje campos en blancos, se comience a escribir con minúscula o se escriban números. Cuando se inserten los datos correctamente, el sistema debe almacenarlos en la base de datos y mostrarlos en el listado <u>CP 7: Listar solicitudes</u>. 	
Evaluación de prueba: Satisfactoria.	

Caso de Prueba	
Número de CP: 2	HU 1: Crear solicitud de servicio
Nombre: Crear solicitud de evaluación.	
Descripción: Se insertan los datos de un servicio de prueba, inicialmente se insertarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> El usuario debe estar autenticado o tener el rol de Planificador. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> Para crear una solicitud de evaluación se debe seleccionar del bloque Solicitud la opción: <u>Crear solicitud de Evaluación/Guardar</u>, o la opción <u>Gestionar solicitud/Crear solicitud de Evaluación/Guardar</u>. Dejar los campos en blanco. Insertar en el campo nombre del proyecto números y comenzar con letra minúscula. Insertar en el campo jefe de proyecto números, comenzar con letra minúscula y solo ponga el nombre y el primer apellido. No seleccione ninguna metodología. Insertar en el campo fase números y comenzar con letra minúscula. Insertar en el campo etapa números y comenzar con letra minúscula. Insertar en el campo rol números y comenzar con letra minúscula. No seleccione ningún tipo de evaluación. Insertar los datos correctos. Verificar que la nueva solicitud fue almacenada. Ir al <u>CP 7 Listar solicitudes</u>. 	
Resultado esperado:	
<ul style="list-style-type: none"> El sistema debe alertar al usuario cuando se deje campos en blancos, se comience a escribir con minúscula o se escriban números. Cuando se inserten los datos correctamente, el sistema debe almacenarlos en la base de datos y mostrarlos en el listado CP 7: Listar solicitudes. 	
Evaluación de prueba: Satisfactoria.	

Caso de Prueba	
Número de CP: 3	HU 1: Crear solicitud de servicio
Nombre: Crear solicitud de asesoramiento.	
Descripción: Se insertan los datos de un servicio de prueba, inicialmente se insertarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar autenticado o tener el rol de Planificador. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Para crear una solicitud de evaluación se debe seleccionar del bloque Solicitud la opción: <u>Crear solicitud de Asesoramiento/Guardar</u>, o la opción <u>Gestionar solicitud/Crear solicitud de Asesoramiento/Guardar</u>. 2. Dejar los campos en blanco. 3. Insertar en el campo nombre del proyecto números y comenzar con letra minúscula. 4. Insertar en el campo jefe de proyecto números, comenzar con letra minúscula y solo ponga el nombre y el primer apellido. 5. No seleccione ninguna metodología. 6. Insertar en el campo fase números y comenzar con letra minúscula. 7. Insertar en el campo etapa números y comenzar con letra minúscula. 8. Insertar en el campo rol números y comenzar con letra minúscula. 9. Insertar en el campo tema números y comenzar con letra minúscula. 10. Insertar los datos correctos. 11. Verificar que la nueva solicitud fue almacenada. Ir al <u>CP 7 Listar solicitudes</u>. 	
Resultado esperado:	
<ul style="list-style-type: none"> • El sistema debe alertar al usuario cuando se deje campos en blancos, se comience a escribir con minúscula o se escriban números. • Cuando se inserten los datos correctamente, el sistema debe almacenarlos en la base de datos y mostrarlos en el listado <u>CP 7: Listar solicitudes</u>. 	
Evaluación de prueba: Satisfactoria.	

Caso de Prueba	
Número de CP: 4	HU 2: Modificar solicitud de servicio
Nombre: Modificar solicitud.	
Descripción: Se modifican los datos de una solicitud de un servicio, inicialmente se modificarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta y se muestra los datos actualizados.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario debe estar autenticado o tener el rol de Planificador. • Los usuarios solo pueden modificar la solicitud que creó, excepto los que tienen el rol de Planificador. • Solo pueden ser modificadas las solicitudes con estado Pendiente. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Para modificar una solicitud de servicio se debe seleccionar del bloque Solicitud la opción: <u>Mostrar solicitudes/Modificar/Actualizar</u>, o la opción <u>Gestionar solicitud/Mostrar solicitudes/Modificar/Actualizar</u>. 2. Dejar los campos en blanco. 3. Insertar en el campo nombre del proyecto números y comenzar con letra minúscula. 4. Insertar en el campo jefe de proyecto números, comenzar con letra minúscula y solo ponga el nombre y el primer apellido. 5. No seleccione ninguna metodología. 6. Insertar en el campo fase números y comenzar con letra minúscula. 7. Insertar en el campo etapa números y comenzar con letra minúscula. 8. Insertar en el campo rol números y comenzar con letra minúscula. <p>En dependencia de la solicitud seleccionada será el tipo de servicio, si es una solicitud de Prueba:</p> <ol style="list-style-type: none"> 9. No seleccione ningún tipo de prueba. 10. Insertar en el campo nombre del producto números y comenzar con letra minúscula. 11. Insertar en el campo artefacto números y comenzar con letra minúscula. 12. Insertar en el campo tipo de artefacto números y comenzar con letra minúscula. 13. No seleccione ningún tipo de documentación. <p>Si es una solicitud de Evaluación:</p> <ol style="list-style-type: none"> 14. Insertar en el campo tipo de evaluación números y comenzar con letra minúscula <p>Si es una solicitud de Asesoramiento:</p> <ol style="list-style-type: none"> 15. Insertar en el campo tema números y comenzar con letra minúscula. 16. Insertar los datos correctos. 	

<p>Resultado esperado:</p> <ul style="list-style-type: none"> El sistema debe alertar al usuario cuando se deje campos en blancos, se comience a escribir con minúscula o se escriban números. El sistema debe alertar al usuario cuando no tiene permisos para eliminar la solicitud o cuando la solicitud no tenga el estado Pendiente. Cuando se actualicen los datos correctamente, el sistema debe almacenarlos en la base de datos y mostrarlos inmediatamente que se presione el botón Actualizar <u>CP 6: Ver solicitud de servicio</u> y en el listado <u>CP 7: Listar solicitudes</u>.
<p>Evaluación de prueba: Satisfactoria.</p>

Caso de Prueba	
Número de CP: 5	HU 3: Eliminar solicitud de servicio
Nombre: Eliminar solicitud.	
Descripción: Se elimina una solicitud de un servicio.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> El usuario debe estar autenticado o tener el rol de Planificador. Los usuarios solo pueden modificar la solicitud que creó, excepto los que tienen el rol de Planificador. Solo pueden ser modificadas las solicitudes con estado Pendiente. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> Para eliminar una solicitud de servicio se debe seleccionar del bloque Solicitud la opción: <u>Mostrar solicitudes/Eliminar/Eliminar</u>, o la opción <u>Gestionar solicitud/Mostrar solicitudes/Eliminar/Eliminar</u>. Verificar que la solicitud eliminada no aparece en el listado <u>CP 7: Listar solicitudes</u>. 	
Resultado esperado:	
<ul style="list-style-type: none"> El sistema debe alertar al usuario cuando no tiene permisos para eliminar la solicitud o cuando la solicitud no tenga el estado Pendiente. Cuando se elimine la solicitud, el sistema debe eliminar la información de la base de datos y quitar la solicitud del listado <u>CP 7: Listar solicitudes</u>. 	
Evaluación de prueba: Satisfactoria.	

Caso de Prueba	
Número de CP: 6	HU 4: Ver solicitud de servicio
Nombre: Ver solicitud.	
Descripción: Se muestra los datos de una solicitud.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> El usuario debe estar autenticado. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> Para ver una solicitud de servicio se debe seleccionar del bloque Solicitud la opción: <u>Mostrar solicitudes/Ver</u>, o la opción <u>Gestionar solicitud/Mostrar solicitudes/ver</u>. El sistema permite las opciones de modificar <u>CP 4: Modificar solicitud</u>, eliminar <u>CP 5: Eliminar solicitud</u>, exportar a PDF <u>CP 11: Exportar a PDF</u>. 	
Resultado esperado:	
<ul style="list-style-type: none"> Ver los datos de la solicitud seleccionada. 	
Evaluación de prueba: Satisfactoria.	

Caso de Prueba	
Número de CP: 7	HU 5: Listar solicitudes
Nombre: Listar solicitudes.	
Descripción: Se muestra todas las solicitudes almacenadas en el sistema mediante un paginado, por página se tendrá 10 solicitudes ordenadas de forma descendente.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> El usuario debe estar autenticado o tener el rol de Planificador. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> Para listar las solicitudes de servicios se debe seleccionar del bloque Solicitud la opción: <u>Mostrar solicitudes</u>, o la opción <u>Gestionar solicitud/Mostrar solicitudes</u>. <p>Para la sección de Administrar solicitud, donde solo puede acceder el rol de Planificador:</p> <ol style="list-style-type: none"> Para listar las solicitudes de servicios se debe seleccionar del bloque Solicitud la opción: <u>Cambiar estado de una solicitud</u>, o la opción <u>Administrar solicitud/Cambiar estado de una solicitud</u>. 	
Resultado esperado:	
<ul style="list-style-type: none"> El sistema debe mostrar mensaje de alerta en caso de no tener permisos. Se muestra todas las solicitudes almacenadas en el sistema. 	
Evaluación de prueba: Satisfactoria.	

Caso de Prueba	
Número de CP: 8	HU 6: Aceptar solicitud de servicio
Nombre: Aceptar solicitud.	
Descripción: Se insertan los datos para aceptar una solicitud de un servicio, inicialmente se insertarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario autenticado debe tener el rol de Planificador. • La solicitud de tener estado Pendiente. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Para aceptar una solicitud de servicio se debe seleccionar del bloque Solicitud la opción: <u>Cambiar estado de una solicitud/Aceptar/Aceptar</u>, o la opción <u>Administrar solicitud/Cambiar estado de una solicitud/Aceptar/Aceptar</u>. 2. Dejar los campos en blanco. 3. No seleccione ningún responsable. 4. Los restantes campos no son obligatorios. 5. Insertar los datos correctos. 	
Resultado esperado:	
<ul style="list-style-type: none"> • El sistema debe alertar al usuario cuando se deje campos en blancos. • El sistema debe alertar al usuario cuando no tiene permisos para aceptar la solicitud o cuando la solicitud no tenga el estado Pendiente. • Cuando se acepta una solicitud correctamente, el sistema actualiza el estado de la solicitud a Aceptada y se envía una notificación por correo a la persona creadora de la solicitud y al responsable seleccionado. Ir a <u>CP 7: Listar solicitudes</u> para verificar el cambio de estado. 	
Evaluación de prueba: Satisfactoria.	

Caso de Prueba	
Número de CP: 9	HU 7: Rechazar solicitud de servicio
Nombre: Rechazar solicitud.	
Descripción: Se insertan los datos para rechazar una solicitud de un servicio, inicialmente se insertarán incorrectamente para verificar las validaciones del sistema, luego de forma correcta para comprobar que los datos sean almacenados.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario autenticado debe tener el rol de Planificador. • La solicitud de tener estado Pendiente. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Para rechazar una solicitud de servicio se debe seleccionar del bloque Solicitud la opción: <u>Cambiar estado de una solicitud/Rechazar/Rechazar</u>, o la opción <u>Administrar solicitud/Cambiar estado de una solicitud/Rechazar/Rechazar</u>. 2. Dejar los campos en blanco. 3. Insertar los datos correctos. 	
Resultado esperado:	
<ul style="list-style-type: none"> • El sistema debe alertar al usuario cuando se deje campos en blancos. • El sistema debe alertar al usuario cuando no tiene permisos para rechazar la solicitud o cuando la solicitud no tenga el estado Pendiente. • Cuando se rechaza una solicitud correctamente, el sistema actualiza el estado de la solicitud a Rechazada y se envía una notificación por correo a la persona creadora de la solicitud. Ir a <u>CP 7: Listar solicitudes</u> para verificar el cambio de estado. 	
Evaluación de prueba: Satisfactoria.	

Caso de Prueba	
Número de CP: 10	HU 8: Cerrar solicitud de servicio
Nombre: Cerrar solicitud.	
Descripción: Se insertan los datos para cerrar una solicitud de un servicio, luego se comprueba que los datos sean almacenados.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario autenticado debe tener el rol de Planificador. • La solicitud de tener estado Aceptada o rechazada. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Para cerrar una solicitud de servicio se debe seleccionar del bloque Solicitud la opción: <u>Cambiar estado de una solicitud/Cerrar/Cerrar</u>, o la opción <u>Administrar solicitud/Cambiar estado de una solicitud/Cerrar/Cerrar</u>. 2. Los campos no son obligatorios. 	
Resultado esperado:	
<ul style="list-style-type: none"> • El sistema debe alertar al usuario cuando no tiene permisos para cerrar la solicitud o cuando la solicitud no tenga el estado Aceptada o Rechazada. • Cuando se cierra una solicitud correctamente, el sistema actualiza el estado de la solicitud a Cerrada. Ir a <u>CP 7: Listar solicitudes</u> para verificar el cambio de estado. 	
Evaluación de prueba: Satisfactoria.	

Caso de Prueba	
Número de CP: 11	HU 8: Realizar reportes a las solicitudes de servicios
Nombre: Reportes de solicitudes.	
Descripción: Se selecciona según el usuario desee los criterios de búsqueda para filtrar las solicitudes.	
Condiciones de ejecución:	
<ul style="list-style-type: none"> • El usuario autenticado debe tener el rol de Planificador. 	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Para realizar reportes a las solicitudes se debe seleccionar del bloque Solicitud la opción: <u>Reportes de solicitudes/Buscar</u>, o la opción <u>Administrar solicitud/Reportes de solicitudes/Buscar</u>. 2. Seleccionar uno o varios criterios. 	
Resultado esperado:	
<ul style="list-style-type: none"> • El sistema debe mostrar una tabla con las solicitudes según los filtros que realice el usuario. • El sistema debe permitir exportar a PDF los reportes realizados. Ir a <u>CP 12: Exportar a PDF</u>. 	
Evaluación de prueba: Satisfactoria.	

Caso de Prueba	
Número de CP: 12	HU 8: Exportar a PDF
Nombre: Exportar a PDF.	
Descripción: El sistema muestra en un documento PDF los datos seleccionados.	
Condiciones de ejecución:	
Entrada/Pasos de ejecución:	
<ol style="list-style-type: none"> 1. Para exportar a PDF se debe seleccionar del bloque Solicitud la opción: <u>Reportes de solicitudes/Buscar/Exportar a PDF</u>, o la opción <u>Administrar solicitud/Reportes de solicitudes/Buscar/Exportar a PDF</u>. 2. Para exportar a PDF se debe seleccionar del bloque Solicitud la opción: <u>Mostrar solicitudes/PDF</u>, <u>Mostrar solicitudes/Ver/PDF</u> o la opción <u>Gestionar solicitud/Mostrar solicitudes/PDF</u>, <u>Gestionar solicitud/Mostrar solicitudes/Ver/PDF</u>. 3. Para exportar a PDF se debe seleccionar del bloque Solicitud la opción: <u>Cambiar estado de una solicitud/PDF</u>, <u>Cambiar estado de una solicitud/Ver/PDF</u> o la opción <u>Administrar solicitud/Cambiar estado de una solicitud/PDF</u>, <u>Administrar solicitud/Cambiar estado de una solicitud/Ver/PDF</u>. 	
Resultado esperado:	
<ul style="list-style-type: none"> • El sistema debe permitir mostrar en documento PDF los datos seleccionados. 	
Evaluación de prueba: Satisfactoria.	

Anexo 4 Interfaces de usuario.

Crear solicitud de Prueba

▼ Datos del solicitante

Nombre del proyecto: *	Jefe de proyecto: *	Metodología de desarrollo: *
<input type="text"/>	<input type="text"/>	Seleccione ▼
Fase de desarrollo: *	Etapa de la fase: *	Rol en su proyecto: *
<input type="text"/>	<input type="text"/>	<input type="text"/>

▼ Datos de la Solicitud

Tipo de prueba: *	Nombre del producto: *	Artefacto a probar: *
Seleccione ▼	<input type="text"/>	<input type="text"/>
Tipo de artefacto a probar: *	Tipo de documentación: *	Subir archivo:
<input type="text"/>	Seleccione ▼	<input type="text"/> <input type="button" value="Examinar..."/>

Suba un archivo (.rar).

Para proponer las fechas debe tener en cuenta:

- La fecha de inicio debe ser 3 días después de la fecha actual (24/05/2012).
- La fecha final no puede ser menor o igual que la fecha de inicio.

Fecha de inicio:	Fecha final:
<input type="text"/>	<input type="text"/>
Formato: 24-05-2012	Formato: 24-05-2012

Descripción:

Escriba como quiere que se realice su solicitud.

Figura 9 Crear solicitud de Prueba.

Crear solicitud de Evaluación

▼ Datos del solicitante

Nombre del proyecto: *	Jefe de proyecto: *	Metodología de desarrollo: *
<input type="text"/>	<input type="text"/>	Seleccione ▼
Fase de desarrollo: *	Etapas de la fase: *	Rol en su proyecto: *
<input type="text"/>	<input type="text"/>	<input type="text"/>

▼ Datos de la Solicitud

Tipo de evaluación: * ▼

Subir archivo:

Suba un archivo (.rar).

Para proponer las fechas debe tener en cuenta:

- La fecha de inicio debe ser 3 días después de la fecha actual (24/05/2012).
- La fecha final no puede ser menor o igual que la fecha de inicio.

Fecha de inicio:	Fecha final:
<input type="text"/>	<input type="text"/>
Formato: 24-05-2012	Formato: 24-05-2012

Descripción:

Escriba como quiere que se realice su solicitud.

Figura 10 Crear solicitud de Evaluación.

Crear solicitud de Asesoramiento

▼ Datos del solicitante

Nombre del proyecto: *	Jefe de proyecto: *	Metodología de desarrollo: *
<input type="text"/>	<input type="text"/>	Seleccione ▼
Fase de desarrollo: *	Etapas de la fase: *	Rol en su proyecto: *
<input type="text"/>	<input type="text"/>	<input type="text"/>

▼ Datos de la Solicitud

Tema: *

Subir archivo:

Suba un archivo (.rar).

Detalles: *

Indique los detalles del tema solicitado.

Para proponer las fechas debe tener en cuenta:

- La fecha de inicio debe ser 3 días después de la fecha actual (24/05/2012).
- La fecha final no puede ser menor o igual que la fecha de inicio.

Fecha de inicio:	Fecha final:
<input type="text"/>	<input type="text"/>
Formato: 24-05-2012	Formato: 24-05-2012

Descripción:

Escriba como quiere que se realice su solicitud.

Figura 11 Crear solicitud de Asesoramiento.

Aceptar solicitud

▼ Aceptar solicitud

Responsable de la solicitud: *
 Seleccione
 Persona encargada de ejecutar el servicio.

Comentarios:

Escriba cualquier aclaracion que desee hacer.

Aceptar Cancelar

Figura 12 Aceptar solicitud.

Reportes de solicitudes

▼ Criterios de búsqueda

Tipo de servicio: Seleccione Estado de la solicitud: Seleccione Proyecto: Seleccione Fecha de solicitud:
 Formato: 31-05-2012

Buscar Cancelar Eportar a PDF

▼ Solicitudes

Proyecto	Jefe de proyecto	Metodología	Solicitante	Correo	Fecha de solicitud	Tipo servicio	Estado
Alfaomega	Yoandy Perez Caceres	RUP	nadie	nadie@uci.cu	24-05-2012	Prueba	Pendiente
Alfaomega	Yoandy Perez Caceres	RUP	admin2	aboza@estudiantes.uci.cu	19-05-2012	Prueba	Aceptada
Recursos didacticos	Ronaldo Valdez Fernandez	SCRUM	admin2	aboza@estudiantes.uci.cu	19-05-2012	Prueba	Pendiente
Calidad	Aimet Bajuelo Tejido	XP	ambajuelo	ambajuelo@estudiantes.uci.cu	18-05-2012	Asesoramiento	Pendiente
Calidad	Katherin Borrego Carrillo	XP	kcarrillo	kcarrillo@estudiantes.uci.cu	18-05-2012	Prueba	Pendiente
Calidad	Aimet Bajuelo Tejido	SCRUM	ambajuelo	ambajuelo@estudiantes.uci.cu	18-05-2012	Evaluación	Pendiente
Calidad	Yusdel Merinno Alquizar	RUP	apahumier	apahumier@estudiantes.uci.cu	18-05-2012	Prueba	Pendiente
Calidad	Aimet Bajuelo Tejido	RUP	ambajuelo	ambajuelo@estudiantes.uci.cu	18-05-2012	Prueba	Pendiente
Recursos didacticos	Lisandra Guibert Estrada	SCRUM	admin2	aboza@estudiantes.uci.cu	14-05-2012	Prueba	Pendiente
Calidad	Yoandy Perez Caceres	SCRUM	rey	ralpajon@estudiantes.uci.cu	14-05-2012	Prueba	Cerrada
Multisaber	Abel Rodriguez Baez	XP	rey	ralpajon@estudiantes.uci.cu	14-05-2012	Evaluación	Cerrada
Alfaomega	Yoandy Perez Caceres	RUP	rey	ralpajon@estudiantes.uci.cu	14-05-2012	Asesoramiento	Cerrada
Recursos didacticos	Lisandra Guibert Estrada	XP	rey	ralpajon@estudiantes.uci.cu	14-05-2012	Asesoramiento	Rechazada
Recursos didacticos	Lisandra Guibert Estrada	XP	admin2	aboza@estudiantes.uci.cu	14-05-2012	Evaluación	Rechazada
Alfaomega	Ronaldo Valdez Molina	SCRUM	rey	ralpajon@estudiantes.uci.cu	14-05-2012	Evaluación	Aceptada
Calidad	Yusdel Meriño Almaguer	Otra	rey	ralpajon@estudiantes.uci.cu	14-05-2012	Evaluación	Aceptada
Calidad	Yusdel Meriño Almaguer	Otra	rey	ralpajon@estudiantes.uci.cu	14-05-2012	Asesoramiento	Aceptada
Alfaomega	Yoandy Perez Caceres	RUP	rey	ralpajon@estudiantes.uci.cu	13-05-2012	Asesoramiento	Aceptada
Recursos didacticos	Arlan Bomez Benitez	XP	admin2	aboza@estudiantes.uci.cu	10-05-2012	Evaluación	Aceptada
El Navegante	Abel Rodriguez Baez	XP	admin2	aboza@estudiantes.uci.cu	10-05-2012	Prueba	Aceptada

Figura 13 Reportes de solicitudes.