



Universidad de las Ciencias Informáticas

Facultad 4

Título: Sistema de Gestión de Calidad del Grupo de Calidad de FORTES. Módulo de Pruebas.

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

Autora: Elizabeth de los Milagros Ramírez Chacón.

Tutor: Ing. Yudislandry Aguila González

Ciudad de la Habana

Año 53 de la Revolución



Se ha de tener fe en lo mejor del hombre y desconfiar de lo peor de él. Hay que dar ocasión a lo mejor para que se revele y prevalezca sobre lo peor. Si no, lo peor prevalece.

José Martí

DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Elizabeth de los Milagros Ramírez Chacón

Firma del Autor.

Ing. Yudislandry Aguila González

Firma del Tutor.

Agradecimientos

A:

Mi mamá y papito por convertirme en esta súper personita.

Mi hermanita por enseñarme tantas cosas y por todo lo vivido.

Toda mi familia por ser la mejor de todas.

Mi maestro por permitirme ser su Padawan DLM.

Misael por demostrarme que los sueños se hacen realidad.

Adriana por su ácida dulzura.

Mi Panchop por la poesía, las conversaciones y por ayudar al crecimiento de las plantas.

Maite por los momentos geniales que pasamos juntas.

Yaulio por haberme ayudado y comprendido durante todos estos años incondicionalmente.

Yera por ponerme en sus agradecimientos y por darme su abrigo.

Felandruja por creer siempre en mí.

Yamila por confiar en mi miedo.

Wisel por mostrarme el camino del Drupal.

Eduardo por mostrarme que ir contra los principios no siempre es malo.

Todos los que han compartido mi día a día (en el apto, lab y aula) y han soportado mi perfección.

Mi por todo, y por este amor incondicional.

Dedicatoria

*A las 10 personas que más he amado en mi vida. Elsa,
Nestor, Evita, Luisa, Placede, Eva, Emeterio, Chela,
Elva y Yo.*

Resumen

La gran competencia que existe en la actualidad en el mercado del software obliga a los diferentes productores a lograr productos de calidad para mantener satisfechos a los clientes. Esto ha provocado la puesta en marcha en las diferentes empresas de procesos para la gestión y perfeccionamiento de la calidad, de esta manera se garantizará al cliente un producto final que responda completamente a sus necesidades.

El Grupo de Calidad perteneciente al Centro de Tecnologías para la Formación de la Universidad de las Ciencias Informáticas es el encargado de gestionar y asegurar la calidad de los productos que se desarrollan en este centro, para ello entre los servicios que ofrece está el proceso de pruebas que se le aplica a los productos para validar la calidad de los mismos. Actualmente no existe una herramienta que apoye este proceso, debido a esto el trabajo de diploma que lleva por título “Sistema de Gestión de Calidad del Grupo de Calidad de FORTES. Módulo de Pruebas” tiene como objetivo la implementación de una aplicación Web para la gestión de la información referente al proceso de pruebas que se realiza en el Grupo de Calidad.

El desarrollo de esta aplicación estuvo guiado por la metodología de desarrollo XP. Se presentó el estudio e investigación realizados sobre las diferentes tecnologías y herramientas que se utilizan para la creación de aplicaciones Web, además de las pruebas realizadas al sistema para garantizar su calidad.

Índice

| | |
|---|----------|
| Resumen | V |
| Introducción | 1 |
| Capítulo 1: Fundamentación Teórica | 6 |
| 1.1 Introducción | 6 |
| 1.2 Estado del arte | 6 |
| 1.2.1 Aplicaciones web..... | 6 |
| 1.2.1.1 QAction | 7 |
| 1.2.1.2 MPR Systema® | 8 |
| 1.2.1.3 TRAC..... | 8 |
| 1.2.1.4 REDMINE | 9 |
| 1.2.2 Los servidores web..... | 10 |
| 1.2.2.1 Sun Java™ System Web Server | 11 |
| 1.2.2.2 Apache | 11 |
| 1.2.3 Los CMS..... | 12 |
| 1.2.3.1 DRUPAL | 12 |
| 1.2.3.2 Joomla | 14 |
| 1.2.4 Lenguaje de Programación | 15 |
| 1.2.4.1 Python | 15 |
| 1.2.4.2 JavaScript | 16 |
| 1.2.4.3 PHP | 16 |
| 1.2.5 Los Sistemas Gestores de base de datos | 17 |
| 1.2.5.1 MySQL | 17 |
| 1.2.5.2 PostgreSQL | 18 |
| 1.2.6 Los entornos de desarrollo integrado o Integrated Development Environment (IDE) | 19 |
| 1.2.6.1 NetBeans..... | 19 |
| 1.2.6.2 Eclipse | 19 |
| 1.2.7 Metodologías | 20 |
| 1.2.7.1 Programación Extrema (XP) | 20 |
| 1.2.7.2 Rational Unified Process (RUP) | 23 |
| 1.2.8 Herramienta Case | 26 |
| 1.2.8.1 Rational Rose | 26 |
| 1.2.8.2 Visual Paradigm..... | 26 |
| 1.2.9 Conclusiones | 27 |

| | |
|---|-----------|
| Capítulo 2: Características del sistema..... | 28 |
| 2.1 Introducción..... | 28 |
| 2.2 Propuesta de sistema..... | 28 |
| 2.3 Objetivos del Módulo de Pruebas..... | 29 |
| 2.4 Especificación de los requisitos del software | 29 |
| 2.4.1 Requisitos funcionales | 29 |
| Historias de Usuario (HU)..... | 30 |
| Estimación de esfuerzos por historias de usuarios..... | 36 |
| Plan de duración de las Iteraciones | 37 |
| 2.4.2 Requisitos no funcionales | 38 |
| Software:..... | 38 |
| Hardware: | 39 |
| Apariencia: | 39 |
| Usabilidad: | 39 |
| Rendimiento:..... | 39 |
| Soporte:..... | 39 |
| Portabilidad:..... | 39 |
| Seguridad: | 39 |
| Confiabilidad: | 40 |
| Disponibilidad: | 40 |
| Interfaz:..... | 40 |
| 2.4.3 Diseño de la Base de Datos..... | 40 |
| 2.4.4 Diagrama de despliegue..... | 42 |
| 2.4.5 Interfaz de usuario (IU) | 43 |
| 2.4.6 Conclusiones: | 46 |
| Capítulo 3: Implementación y pruebas..... | 47 |
| 3.1 Introducción..... | 47 |
| 3.2 Roles..... | 47 |
| 3.3 Módulos de apoyo | 48 |
| 3.4 Casos de prueba (CP) | 49 |
| Caso de Prueba Historia de Usuario: Insertar producto | 49 |
| Caso de Prueba Historia de Usuario: Ver producto | 49 |
| Caso de Prueba Historia de Usuario: Eliminar producto | 50 |
| Caso de Prueba Historia de Usuario: Insertar tipo de prueba | 50 |

| | |
|---|-----------|
| Caso de Prueba Historia de Usuario: Ver tipo de prueba | 51 |
| Caso de Prueba Historia de Usuario: Solicitar prueba | 51 |
| Caso de Prueba Historia de Usuario: Ver solicitud de prueba | 52 |
| Caso de Prueba Historia de Usuario: Modificar solicitud de prueba | 53 |
| Caso de Prueba Historia de Usuario: Modificar prueba planificada | 54 |
| Caso de Prueba Historia de Usuario: Eliminar prueba planificada | 55 |
| Caso de Prueba Historia de Usuario: Reportar NC..... | 55 |
| Caso de Prueba Historia de Usuario: Ver NC | 56 |
| Caso de Prueba Historia de Usuario: Revisar NC | 57 |
| Caso de Prueba Historia de Usuario: Generar reportes..... | 58 |
| Caso de Prueba Historia de Usuario: Responder NC | 58 |
| Conclusiones sobre las pruebas | 59 |
| 3.4 Conclusiones: | 60 |
| Conclusiones..... | 61 |
| Recomendaciones..... | 62 |
| Referencias Bibliográficas | 63 |
| Bibliografía | 64 |
| Glosario de términos | 66 |
| Anexos | 68 |
| Anexo 1: Casos de Prueba..... | 68 |
| Caso de Prueba Historia de Usuario: Eliminar solicitud de prueba | 68 |
| Caso de Prueba Historia de Usuario: Planificar prueba | 68 |
| Caso de Prueba Historia de Usuario: Ver pruebas planificadas..... | 69 |
| Caso de Prueba Historia de Usuario: Insertar iteración..... | 70 |
| Caso de Prueba Historia de Usuario: Ver iteración..... | 70 |
| Caso de Prueba Historia de Usuario: Insertar proyecto | 71 |
| Caso de Prueba Historia de Usuario: Ver proyecto | 71 |
| Caso de Prueba Historia de Usuario: Eliminar proyecto..... | 72 |
| Anexo 2: Interfaz de usuario | 73 |

Índice de tablas

| | |
|--|----|
| Tabla 1: Diferencias entre metodologías ágiles y no ágiles | 20 |
| Tabla 2: HU - Insertar iteración..... | 30 |
| Tabla 3: HU - Ver iteración..... | 30 |
| Tabla 4: HU – Insertar proyecto..... | 30 |
| Tabla 5: HU – Ver proyecto..... | 30 |
| Tabla 6: HU – Eliminar proyecto | 31 |
| Tabla 7: HU – Insertar producto | 31 |
| Tabla 8: HU – Ver producto..... | 31 |
| Tabla 9: HU – Eliminar producto..... | 31 |
| Tabla 10: HU – Insertar tipo de prueba | 31 |
| Tabla 11: HU – Ver tipo de prueba..... | 32 |
| Tabla 12: HU – Solicitar prueba | 32 |
| Tabla 13: HU – Ver solicitud de prueba | 32 |
| Tabla 14: HU – Modificar solicitud de prueba | 33 |
| Tabla 15: HU – Eliminar solicitud de prueba..... | 33 |
| Tabla 16: HU – Planificar prueba | 33 |
| Tabla 17: HU – Ver prueba planificada | 33 |
| Tabla 18: HU – Modificar prueba planificada | 34 |
| Tabla 19: HU – Eliminar prueba planificada..... | 34 |
| Tabla 20: HU – Reportar NC..... | 34 |
| Tabla 21: HU – Ver NC..... | 35 |
| Tabla 22: HU – Revisar NC..... | 35 |
| Tabla 23: HU – Responder NC..... | 35 |
| Tabla 24: HU – Generar reportes..... | 35 |
| Tabla 25: Estimación de esfuerzo en días por historia de usuario. | 36 |
| Tabla 26: Estimación de esfuerzo en días por iteración. | 37 |
| Tabla 27: Datos que almacenan las tablas que se utilizan en el Módulo de Pruebas | 41 |
| Tabla 33: CP HU Insertar producto | 49 |
| Tabla 34: CP HU Ver producto | 49 |
| Tabla 35: CP HU Eliminar producto..... | 50 |
| Tabla 36: CP HU Insertar tipo de prueba | 51 |

| | |
|---|----|
| Tabla 37: CP HU Ver tipo de prueba | 51 |
| Tabla 38: CP HU Solicitar prueba | 52 |
| Tabla 39: CP HU Ver solicitud de prueba | 53 |
| Tabla 40: CP HU Modificar solicitud de prueba | 53 |
| Tabla 44: CP HU Modificar prueba planificada | 54 |
| Tabla 45: CP HU Eliminar prueba planificada | 55 |
| Tabla 46: CP HU Reportar NC..... | 56 |
| Tabla 47: CP HU Ver NC | 56 |
| Tabla 48: CP HU Revisar NC | 57 |
| Tabla 49: CP Generar reportes..... | 58 |
| Tabla 50: CP Responder NC..... | 59 |
| Tabla 51: No conformidades por iteración realizada..... | 59 |
| Tabla 41: CP HU Eliminar solicitud de prueba..... | 68 |
| Tabla 42: CP HU Planificar prueba | 68 |
| Tabla 43: CP HU Ver pruebas planificadas | 69 |
| Tabla 28: CP HU Insertar iteración..... | 70 |
| Tabla 29: CP HU Ver iteración..... | 71 |
| Tabla 30: CP HU Insertar proyecto..... | 71 |
| Tabla 31: CP HU Ver proyecto..... | 72 |
| Tabla 32: CP HU Eliminar proyecto | 72 |

Índice de Imágenes

| | |
|--|----|
| Ilustración 1: Fases de desarrollo de XP | 21 |
| Ilustración 2: Fases del ciclo de vida de RUP (4) | 24 |
| Ilustración 3: Interacción de tablas del Módulo de Pruebas y de Drupal | 41 |
| Ilustración 4: Diagrama de despliegue del “Módulo de Pruebas” | 43 |
| Ilustración 5: IU Insertar iteración | 44 |
| Ilustración 6: IU Ver iteraciones..... | 44 |
| Ilustración 7: IU Insertar proyecto | 44 |
| Ilustración 8: IU Ver proyectos | 45 |
| Ilustración 9: IU Insertar productos | 45 |
| Ilustración 10: IU Ver productos | 45 |
| Ilustración 11: Roles que intervienen en el Módulo de Pruebas..... | 47 |
| Ilustración 21: Utilización del Módulo DATE..... | 48 |
| Ilustración 11: IU Insertar tipo de prueba..... | 73 |
| Ilustración 12: IU Ver tipos de prueba | 73 |
| Ilustración 13: IU Consultar no conformidades | 74 |
| Ilustración 14: IU Responder no conformidades | 74 |
| Ilustración 15: IU Reportar no conformidad..... | 75 |
| Ilustración 16: IU Ver solicitudes de prueba..... | 75 |
| Ilustración 17: IU Insertar solicitud de prueba | 76 |
| Ilustración 18: IU Planificar prueba..... | 77 |
| Ilustración 19: Ver pruebas planificadas..... | 78 |

Introducción

Las tecnologías de la información y las comunicaciones son todas aquellas tecnologías que actualmente favorecen la comunicación y el intercambio de información. El desarrollo de la informática ha traído aparejado lo que se conoce actualmente como Informatización de la Sociedad para lo que ha sido necesario la creación de soluciones informáticas que satisfagan las necesidades en los más disímiles campos como la medicina, la agricultura y la educación, pero no basta con crear un software, es necesario hacerlo con calidad.

La creación de un sistema informático con calidad es una de las tareas más importantes y difíciles a la que se puede enfrentar un equipo de desarrollo, organización o empresa que se dedique a la producción de software.

Por iniciativa del Comandante en Jefe Fidel Castro Ruz con el objetivo de contribuir a la socialización del conocimiento e informatización de la sociedad cubana se dan notables avances hacia el desarrollo masivo de la informática en nuestro país concretándose estos en primer lugar el 8 de septiembre de 1987 con la creación de los Joven Club de Computación y Electrónica, y luego en el año 2002 con la creación de la Universidad de las Ciencias Informáticas (UCI).

La UCI, desde su creación hasta la fecha se ha convertido en uno de los principales productores de software del país, dicha universidad está inmersa en lograr productos con una calidad total, para ello consta en su organización con diferentes grupos que aseguran el control de la calidad. A mediados del año 2010 se decide implementar en la UCI un nuevo modelo formativo, vinculando la formación, producción e investigación, garantizando que los estudiantes, desde el segundo semestre de tercero, cuarto y quinto año aseguren la excelencia de su formación, lo que permite consolidar la eficiencia y productividad del proceso de desarrollo de software.

Esta nueva orientación de la formación introdujo la creación de los centros productivos asociados a cada una de las facultades que la UCI posee, centros que dirigen la producción y además se centran en la formación desde la producción. El centro creado en la facultad 4, el cual lleva como nombre: “Centro de Tecnologías para la Formación (FORTES)”, contiene un Grupo de Calidad, el cual está subordinado al

Centro de Calidad para Soluciones Tecnológicas (Calisoft), rector de todos los grupos que aseguran esta temática en la universidad.

El Grupo de Calidad tiene como principal misión el aseguramiento de la calidad de los diferentes productos que se desarrollan en el centro, para ello realiza disímiles tareas, entre las que se destacan los servicios que presta a los diferentes proyectos, como son: revisiones y auditorías, pruebas de software en las diferentes etapas de desarrollo de los mismos, así como también brinda apoyo a los proyectos, guiándolos, orientándolos y monitoreando la calidad, contribuyendo además al proceso de liberación de estos una vez concluidos los mismos. Como principal eslabón en la cadena del aseguramiento de la calidad se encuentra el proceso de pruebas de software que valida la calidad del mismo.

Las pruebas de software se definen como: una actividad en la cual un sistema o componente es ejecutado bajo condiciones o requerimientos específicos, los resultados son observados y registrados.

Es un proceso de ejecución de un programa con la intención de descubrir errores. Una buena prueba es aquella que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces. (1)

Las mismas son de vital importancia, pues permiten verificar y revelar la calidad de un producto. El proceso de prueba es clave a la hora de detectar errores o fallas, ya que es utilizado para identificar posibles fallos de implementación, calidad, o usabilidad de un software.

El proceso del servicio de Pruebas en el Grupo de Calidad de FORTES consta de seis tareas fundamentales:

1. Atención a las solicitudes de Pruebas.
2. Planificación de Pruebas.
3. Capacitación para la ejecución del servicio.
4. Ejecución de las Pruebas.
5. Evaluación de las Pruebas.
6. Cierre de las Pruebas.

Una vez terminada la ejecución de las pruebas es necesario generar informes donde se refleje el comportamiento de las no conformidades¹ (NC) a través de diferentes criterios como pueden ser:

¹ No conformidad (NC): “Un no cumplimiento a un requisito” de acuerdo a la definición en la norma ISO 9000: 2000

categorías, probadores, proyectos o productos, estos criterios son importantes pues mediante ellos se centra la información en los reportes, los cuales se utilizarán en las reuniones de conciliación así como para manejar la información histórica de las NC por proyectos para hacer análisis que reporten beneficios al servicio de pruebas que ofrece el Grupo de Calidad de FORTES, además estos reportes son necesarios para emitir los partes diarios una vez terminadas las diferentes secciones de trabajo.

Actualmente en la Universidad para realizar el proceso de Gestión de las NC se utiliza la herramienta REDMINE, pero esta herramienta no permite la personalización de los campos a utilizar para realizar el reporte de las NC lo que provoca que el equipo de probadores pierda tiempo escogiendo los campos que se necesitan realmente, provocando que en ocasiones se envíen las NC con errores. Además el REDMINE no posibilita generar los informes necesarios, lo que se traduce en una gran pérdida de tiempo pues tienen que ser generados manualmente.

Por las razones expuestas anteriormente se identifica la necesidad de implementar una herramienta que contribuya a optimizar el servicio de pruebas que brinda el Grupo de Calidad de FORTES.

De lo anterior puede plantearse el **Problema a resolver**:

¿Cómo optimizar el proceso de pruebas del Grupo de Calidad de FORTES?

Aportes prácticos esperados del trabajo.

A partir del problema planteado se espera que con la implementación y puesta en marcha del sistema se logre obtener una solución informática que garantice, facilite y agilice la gestión de las pruebas de software que se realizan en el Grupo de Calidad de FORTES.

El **objeto de estudio** lo conforman los procesos de pruebas, definiéndose como **campo de acción**, aquellas actividades automatizables dentro del proceso de pruebas que realiza el Grupo de Calidad del Centro FORTES. Como **idea a defender** se parte de, que si se realiza el análisis, diseño e implementación del Módulo de Pruebas, se podrá contar con un sistema que entre sus funcionalidades incluya la gestión de los procesos de pruebas en el Grupo de Calidad del Centro FORTES.

Para resolver el problema se trazó como **objetivo general**: Realizar el análisis, diseño e implementación del Módulo de Pruebas para el Sistema de Gestión de la Calidad del Grupo de Calidad del Centro FORTES de la facultad 4.

Los siguientes **objetivos específicos** conducen el desarrollo de la presente investigación:

1. Investigar sobre los procesos de pruebas de manera general y del Grupo de Calidad del Centro FORTES de la facultad 4.
2. Realizar la ingeniería de requerimientos, el análisis y diseño del Módulo de Pruebas.
3. Realizar la implementación y validación del Módulo de Pruebas.

Para el cumplimiento de los objetivos se definieron las siguientes **tareas**:

- Análisis del proceso de prueba que se realiza en el Grupo de Calidad del Centro FORTES de la facultad 4.
- Estudio y definición de la metodología para elaborar la documentación del software.
- Identificación de las actividades automatizables.
- Definición de las herramientas y tecnologías informáticas a utilizar para desarrollar el sistema.
- Realización del análisis y diseño de la aplicación utilizando la metodología definida.
- Implementación y validación de la aplicación.

La investigación está distribuida en tres capítulos, en los cuales se tratan los siguientes aspectos:

Capítulo I: Fundamentación teórica.

En este capítulo se exponen los principales conceptos asociados al dominio del problema, así como las principales herramientas y metodologías que serán usadas en el desarrollo de la aplicación.

Capítulo II: Características del sistema.

En este capítulo se describe la propuesta del sistema, identificando los procesos del negocio y las reglas que lo rigen. Se detallan las funcionalidades que el sistema debe ser capaz de cumplir y los prototipos de interfaz de usuario.

Capítulo III: Implementación y pruebas:

En este capítulo se abordan los aspectos relacionados con la construcción de la solución propuesta además se definen los casos de pruebas para poder realizar las pruebas al software, para así verificar la integridad del mismo y el cumplimiento de los requisitos definidos por el cliente.

Capítulo 1: Fundamentación Teórica

1.1 Introducción

En este capítulo se expondrán los diferentes conceptos que se manejarán durante la investigación, además se construirá un estado del arte donde se expondrán diferentes portales que gestionan las NC en la actualidad, para así determinar las características fundamentales que debe tener la solución del problema descrito, se estudiarán además las diferentes tecnologías, herramientas y lenguajes de programación que existen para poder determinar las que sean más óptimas para la creación del sistema que se ha propuesto como solución al problema planteado.

1.2 Estado del arte

La informatización de la sociedad y el desarrollo de las tecnologías han supuesto la creación de software o aplicaciones que satisfagan las más disímiles necesidades, existen en la actualidad aplicaciones tan sencillas como pueden ser las que realizan una operación matemática hasta aplicaciones sumamente complejas que guían la navegación de los satélites. Lo importante de una aplicación no es la complejidad que posee sino la capacidad de satisfacer las necesidades de sus usuarios o no, una manera de comprobar que una aplicación cumpla con todas las exigencias de sus clientes antes de ser entregadas es realizando las pruebas de software, las cuales juegan un papel fundamental en la creación de cualquier aplicación, ya que validan la calidad real de los mismos, una vez ejecutadas las pruebas de software es necesario realizar la gestión de las NC encontradas, tarea fundamental para garantizar un software con calidad total.

A nivel mundial la gestión de estas NC se realiza en su mayoría mediante aplicaciones web.

1.2.1 Aplicaciones web

No son más que aplicaciones que se pueden utilizar accediendo a un servidor web a través de Internet o de una Intranet mediante un navegador web, estas aplicaciones son muy populares debido a la independencia del sistema operativo, así como la facilidad para actualizar y mantener sin la necesidad de distribuir e instalar software a los usuarios potenciales, solo se necesita un navegador web que cumpla

con los estándares web publicados por el World Wide Web Consortium².

Para garantizar la eficiencia de una aplicación web se necesita además una conexión garantizada al servidor web donde está almacenada la aplicación, al estar la aplicación almacenada en un servidor y no necesariamente en la estación de trabajo en la que se está utilizando ofrece además la ventaja de no necesitar espacio libre en el disco duro para almacenar la misma.

Algunas de las aplicaciones web que se utilizan para la gestión de las NC son QAction, MPR Systema, TRAC y REDMINE, a continuación se detallan algunas de sus características fundamentales:

1.2.1.1 QAction

QAction es un software pensado en el usuario. Lleva un control completo de todas las auditorías, quejas de clientes, reportes de productos o servicios no conformes y de las acciones correctivas y preventivas. (8) QAction permite la notificación mediante correo electrónico. Reduce considerablemente el manejo de información impresa, ya que la mayor parte de la evidencia la registra de forma electrónica.

Características principales de QAction:

Es un software propietario, posee funcionalidades para enviar comunicados a través del correo electrónico a la persona responsable cuando se detecta una no conformidad, una vez que se ha dado solución a la no conformidad se envía al emisor de la misma un mensaje notificándolo, incluye además un reporte de atrasos que se verifica diariamente automáticamente, posee un potente generador de consultas que permite obtener prácticamente cualquier reporte de los datos guardados en las bases de datos. QActionWeb permite registrar auditorías, quejas de clientes, productos no conforme, inspecciones, NC, acciones correctivas y acciones preventivas por Intranet/ Internet usando el explorador de Internet, QActionWeb utiliza la base de datos de QAction, de manera que las dos aplicaciones comparten la misma información. Con esta aplicación pueden existir gran cantidad de usuarios conectados desde cualquier computadora e interactuar con QAction. Esto le permite integrar a los clientes y a los proveedores al sistema de calidad de la empresa.

² **World Wide Web Consortium** es un consorcio internacional que produce estándares para la World Wide Web.

1.2.1.2 MPR Systema®

MPR Systema es una poderosa herramienta propietaria que integra todo el software de calidad en UNO. (9) Disminuye la cantidad de tiempo que las organizaciones desperdician en actividades no productivas, genera reportes automáticos, posee acceso instantáneo a la información del sistema, facilita conocer y evaluar los resultados de manera que se puedan aplicar medidas correctivas cuando no se ajusten a las previsiones. La gestión de toda la información de relevancia se realiza desde un único soporte y una única base de datos, potenciando el flujo interno de comunicación a todos los niveles y el compromiso de todo el personal.

Características principales de MPR Systema:

Permite al usuario conocer el estado de las actividades que han sido creadas, así como ordenarlas utilizando diferentes criterios, permite el registro y trazabilidad de actividades detalladas en planes de acción, llamado «historial de la actividad»; puede personalizar y modificar el formato base de reporte de NC internas, a partir de los principales ítems de análisis (sección, cliente, operario, máquina). El software trabaja a partir de listas predefinidas y modificables para evitar errores de digitación. El análisis gráfico es sencillo y automático, puede reportar sus conclusiones y solicitar las acciones correctivas respectivas.

1.2.1.3 TRAC

Es un sistema de seguimiento para los proyectos de desarrollo de software. Proporciona una interfaz para Subversion (u otros sistemas de control de versiones), un Wiki integrado y cómodas instalaciones de informes, permite la creación de enlaces, tareas, archivos y páginas wiki.

Trac es una herramienta para la gestión de proyectos y el seguimiento de errores. Está desarrollado en el lenguaje de programación Python y actualmente está mantenido por Edgewall Software, es software libre y de código abierto.

1.2.1.4 REDMINE

Es una flexible aplicación web para la administración de proyectos, es una aplicación de código abierto y fue realizada bajo los términos de la licencia GNU General Public License (GPL).³

Características principales de REDMINE:

Consta de un flexible control de acceso basado en roles, permite la publicación de noticias, documentos e información de importancia para la administración de la empresa, permite además el envío de correos y notificaciones a los diferentes miembros de la entidad, permite la autenticación de usuarios mediante los *Lightweight Directory Access Protocol* o *Protocolo Ligero de Acceso a Directorios* (LDAP), soporta diferentes lenguajes y bases de datos, permite generar peticiones a los diferentes miembros del grupo las cuales son notificadas mediante correo electrónico, el estado de estas peticiones se puede ir actualizando y una vez cumplida esta petición es cerrada, permite la generación de reportes mediante diferentes parámetros.

Realizado el estudio de algunas de las aplicaciones web que se utilizan actualmente para la gestión de las NC se ha determinado que QAction y MPR Systema® no son soluciones factibles debido a que son herramientas propietarias, y se aboga por la utilización de soluciones libres. La aplicación web TRAC es óptima para la gestión de las NC, pero la misma no brinda soporte para todo el proceso de pruebas del Grupo de Calidad de FORTES y la aplicación web REDMINE que es la que se utiliza actualmente, posee deficiencias las cuales han sido explicadas con anterioridad.

Una vez analizadas las herramientas utilizadas para la gestión de las no conformidades, se definieron las características y funcionalidades de las que no puede prescindir el Módulo de Pruebas del Grupo de Calidad de FORTES estas son:

- Permitir la autenticación de usuarios mediante el protocolo LDAP.

³ **GNU General Public License** o simplemente **GNU GPL**, es una licencia creada por la Free Software Foundation en 1989 y está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan estas libertades a los usuarios.

- Permitir acceso a las diferentes funcionalidades del módulo según el rol que desempeñe el usuario que se ha autenticado.
- Permitir el reporte de las NC, ajustando las características de este reporte a las necesidades del grupo.
- Permitir generar informes desde diferentes criterios.
- Permitir exportar o imprimir los diferentes informes generados.
- Permitir el envío de notificaciones.
- Permitir el seguimiento de las diferentes NC.
- Permitir la publicación de información relevante referente a las pruebas.

Para el desarrollo de la aplicación web que se propone es necesario tener en cuenta algunas herramientas y tecnologías como: el servidor web que se utilizará para almacenar la misma, los lenguajes de programación necesarios para llevar a cabo su desarrollo, el sistema gestor de base de datos, la metodología de desarrollo, el entorno de desarrollo integrado o IDE en que editará y depurará el código, todos estos elementos serán explicados a continuación:

1.2.2 Los servidores web

Son programas que implementan el protocolo **HyperText Transfer Protocol** o Protocolo de Transferencia de Hipertexto (HTTP), manteniéndose a la espera de peticiones que se generan mediante este protocolo por parte de un cliente al que se le conoce como navegador web, una vez que el servidor web recibe esta petición la procesa y envía mediante este mismo protocolo una respuesta con el contenido que se solicitó en la petición. Dentro de los servidores más utilizados se encuentran Apache, Microsoft, Sun Java™ System Web Server, Google y NCSA.

1.2.2.1 Sun Java™ System Web Server

Es un servidor de alto rendimiento, seguro, y de una escalabilidad masiva. Es una aplicación de subprocesos y procesos múltiples, permite hasta 100 000 conexiones simultáneas, permite la fácil administración y publicación de contenido en el Web Server mediante herramientas estándares del sector, la compresión de los datos antes de su envío a los usuario, permite que los administradores conserven el ancho de banda y mejoren el rendimiento de los sitios, tiene una protección integrada contra los ataques comunes, incluso el uso de secuencias de comandos a través de sitios, los ataques de solicitudes distribuidas en contra de URI y el exceso de solicitudes de las direcciones IP.

Al estar la licencia de **Sun Java** controlada por las leyes de control de exportación de los Estados Unidos, queda terminantemente prohibida la exportación o reexportación a países sujetos al Bloqueo o a entidades identificadas en las listas de exclusión de exportación de los Estados Unidos, este servidor web queda excluido de su uso en Cuba por lo tanto no se puede utilizar como servidor web para apoyar la solución propuesta.

1.2.2.2 Apache

Su configurabilidad, robustez y estabilidad hacen de Apache un servidor web de excelencia, gratuito y de código abierto, es multiplataforma, debido a su diseño modular es altamente configurable, trabaja con gran cantidad de los lenguajes script que se utilizan en la actualidad incluyendo PHP, permite la configuración de las respuestas ante los posibles errores que puedan ocurrir en el servidor.

Unas de las ventajas que poseen estos servidores web es que permiten mostrar una página web sin tener que albergarla en un web hosting o alojamiento web, y permiten además, probar el avance de la misma vía local y acceder a la misma desde equipos remotos.

Es necesario que el servidor web que alojará el Módulo de Pruebas sea un servidor potente y además sea una herramienta libre, una vez analizadas las características que posee cada uno de los servidores descritos anteriormente, se escoge como servidor web a utilizar en la solución propuesta Apache, por todo lo explicado con anterioridad sobre este servidor.

Existen en la actualidad aplicaciones web que facilitan a su vez la creación de otras aplicaciones web estos son los conocidos como **Content Management Systems** o **Sistemas de Gestión de Contenidos (CMS)**

1.2.3 Los CMS

Se utilizan además para facilitar el mantenimiento a una web de forma fácil, rápida y segura, proporcionando un entorno que posibilita la actualización y ampliación de la web con la colaboración de múltiples usuarios. Varios son los CMS que se utilizan, entre los que se destacan Drupal y Joomla, estos CMS son de acceso libre con licencias de código abierto.

1.2.3.1 DRUPAL

Es un CMS de código abierto, con licencia GNU/GPL, escrito en PHP, para su desarrollo y mantenimiento consta de una gran y activa comunidad de usuarios. Es un sistema multiusuario, multiplataforma, multilinguaje y extensible. Drupal consta de muchos módulos que ayudan a construir una aplicación web que se ajuste a las necesidades de los usuarios, esta estructura modular permite ser extendida y configurada por el administrador para distintas funciones por medio de módulos de fácil instalación.

Algunas de las características generales de Drupal son:

- **Multiplataforma:** Puede funcionar con Apache o Microsoft IIS como servidor Web y en sistemas como Linux, BSD, Solaris, Windows y Mac OS X.
- **Existe amplia documentación** sobre este CMS, además hay disponible zonas de descarga de las que se pueden obtener módulos, temas, traducciones así como diferentes versiones del CMS.
- **Portabilidad:** Al estar implementado en PHP, las aplicaciones web que se realicen con este CMS son totalmente portables.
- **Búsqueda:** Todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.

- Código abierto: El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL por lo que es posible extender o adaptar Drupal según las necesidades.
- Personalización: Tanto el contenido como la presentación pueden ser individualizados de acuerdo a las preferencias definidas por el usuario.
- Autenticación de usuarios: El registro o inicio de sesión de los usuarios se puede realizar de manera local o utilizando un sistema de autenticación externo. Para su uso en una Intranet, Drupal se puede integrar con un servidor LDAP.
- Permisos basados en roles: La gran facilidad que brinda los permisos basados en roles es que los administradores no deben otorgar permisos a cada usuario, pueden asignar permisos a un rol y agrupar a los diferentes usuarios por roles.
- Independencia de la base de datos: Aunque la mayor parte de las instalaciones de Drupal utilizan MySQL, existen otras opciones. Drupal incorpora una capa de abstracción de base de datos que actualmente está implementada y mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos.
- Múltiples idiomas y Localización: Drupal está pensado para una audiencia internacional por lo que proporciona opciones para crear un portal multilingüe.
- Administración vía Web: La administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional.
- Control de congestión: Drupal 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 Caché: El mecanismo de caché elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor.

1.2.3.2 Joomla

Joomla es un CMS, que permite crear sitios web y aplicaciones en línea de gran alcance. Muchos aspectos, incluyendo su facilidad de uso y extensibilidad, han hecho de Joomla uno de los software para la creación de sitios Web más populares. Es una solución de código abierto que está disponible gratuitamente para todos.

Joomla está diseñado para ser fácil de instalar y configurar, incluso si no se es un usuario avanzado. Con Joomla se puede crear un nuevo sitio en marcha y funcionando en pocos minutos.

Algunas de las características generales de Joomla son:

- Organización del sitio web: Está preparado para organizar eficientemente los contenidos de su sitio en secciones y categorías.
- Publicación de Contenidos: Se pueden crear páginas ilimitadas y editarlas desde un sencillo editor que permite dar formato a los textos con los estilos e imágenes deseados. Los contenidos son totalmente editables y modificables.
- Escalabilidad e implementación de nuevas funcionalidades: Ofrece la posibilidad de instalar, desinstalar y administrar componentes y módulos con gran facilidad y rapidez.
- Administración de usuarios: Permite almacenar datos de usuarios registrados y también la posibilidad de enviar E-mails masivos a todos los usuarios. La administración de usuarios es jerárquica, y los distintos grupos de usuarios poseen diferentes niveles de permisos dentro de la gestión y administración del sitio.
- Diseño y aspecto estético del sitio: Es muy fácil cambiar el aspecto debido al sistema de plantillas que utiliza.
- Navegación y menú: Totalmente editables desde el panel administrador de Joomla.
- Encuestas: Posee un sistema de votaciones y encuestas dinámicas con resultados en barras porcentuales.

Al no existir un CMS que sea absolutamente mejor en todos los aspectos que otro y ajustándose a las necesidades de la investigación se escoge Drupal como el CMS a aplicar porque es multiplataforma, posee amplia documentación, es muy potente y flexible en cuanto a la construcción del sitio. El sistema de taxonomía de Drupal es muy bueno, tiene un alto rendimiento y es un CMS muy profesional.

1.2.4 Lenguaje de Programación

Escoger el o los lenguajes de programación que más se ajusten a las necesidades de la aplicación web que se desarrollará es de vital importancia ya que así se logrará una aplicación que cumpla con todos los requisitos exigidos por el cliente y esta poseerá todas las funcionalidades necesarias para tener un desempeño óptimo.

Los lenguajes de programación son los principales elementos a definir antes de comenzar el desarrollo de una aplicación web, el cual no es más que aquel elemento dentro de la informática que permite crear programas mediante un conjunto de instrucciones, operadores y reglas de sintaxis; que pone a disposición del programador para que este pueda comunicarse con los dispositivos hardware y software existentes (2). En el desarrollo de aplicaciones web se utilizan lenguajes de interpretación del lado del servidor y lenguajes de interpretación del lado del cliente, los lenguajes del lado del servidor son aquellos que procesan los datos una vez recibidos en el servidor web mediante el protocolo HTTP, y los lenguajes del lado del cliente son aquellos que se procesan antes de enviar la solicitud mediante el protocolo HTTP hacia el servidor web. Existen numerosos lenguajes de programación en la actualidad para el desarrollo de aplicaciones web entre los que se destacan PHP, Javascript, Python y Ruby.

1.2.4.1 Python

Es un lenguaje interpretado del lado del cliente que permite trabajar con gran rapidez e integrar sus sistemas con eficacia. Es un lenguaje multiplataforma, es administrado por la Python Software Foundation, posee una licencia de código abierto, es multiparadigma, esto significa que más que forzar a los programadores a adoptar un estilo particular de programación, permite varios estilos: programación orientada a objetos, programación imperativa y programación funcional. Otros paradigmas están soportados mediante el uso de extensiones.

1.2.4.2 JavaScript

Es un lenguaje de tipo script compacto, interpretado del lado del cliente, basado en objetos y guiado por eventos, diseñado específicamente para el desarrollo de aplicaciones cliente-servidor dentro del ámbito de Internet. Además es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos, se encargan de realizar acciones en el cliente, como pueden ser solicitar datos, confirmaciones, mostrar mensajes, crear animaciones, comprobar campos.

Ventajas:

- Los programas realizados en este lenguaje no requieren de mucha memoria ni tiempo adicional de transmisión, por ser pequeños y compactos.
- No requiere un tiempo de compilación; ya que los scripts se pueden desarrollar en un periodo de tiempo relativamente corto.
- Es independiente de la plataforma hardware o sistema operativo, y funciona correctamente siempre y cuando exista un navegador que lo soporte.

1.2.4.3 PHP

Siglas de *Hypertext Pre-processor* es un lenguaje de programación utilizado principalmente para la interpretación del lado del servidor que fue diseñado especialmente para el desarrollo de aplicaciones web y puede ser incrustado dentro de código HTML. PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix (incluyendo HP-UX, Solaris y OpenBSD), Microsoft Windows, Mac OS X, RISC OS. PHP soporta la mayoría de servidores web de hoy en día, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape e iPlanet, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd y muchos otros. Su curva de aprendizaje es muy corta, lo mejor de usar PHP es que es extremadamente simple para el principiante, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales. PHP también cuenta con soporte para comunicarse con otros servicios usando protocolos tales como LDAP, IMAP, SNMP, NNTP, POP3, HTTP.

Luego de analizados los lenguajes de programación descritos anteriormente se decide utilizar PHP, debido a que es un lenguaje que ofrece muchas funciones que facilitan el desarrollo de una aplicación,

además es un lenguaje libre, multiplataforma y el CMS escogido DRUPAL fue desarrollado específicamente en este lenguaje, y requiere que los módulos que se le instalen estén programados en este lenguaje, no se necesita además ningún lenguaje del lado del cliente ya que las validaciones necesarias para el funcionamiento óptimo del Módulo de Pruebas se pueden realizar mediante funciones que el CMS Drupal posee.

Otro aspecto fundamental para el desarrollo de una aplicación web son las bases de datos las cuales no son más que un conjunto de datos almacenados para su posterior uso, para poder gestionar estos datos es necesario el uso de los Sistemas Gestores de Base de Datos (SGBD).

1.2.5 Los Sistemas Gestores de base de datos

Son un conjunto de programas que se utilizan para crear y dar mantenimiento a una base de datos. Estos poseen opciones para asegurar su integridad, confidencialidad y seguridad. Dentro de los gestores de base de datos más conocidos mundialmente se encuentran MySQL y PostgreSQL.

1.2.5.1 MySQL

La base de datos MySQL asegura un alto rendimiento, alta fiabilidad y facilidad de uso. MySQL es multiplataforma. Lo mismo para usuarios sin experiencia en tecnología de base de datos o desarrolladores experimentados, MySQL ofrece una amplia gama de herramientas de base de datos, asistencia técnica, capacitación y servicios de consultoría que aseguran el éxito.

MySQL por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso.

MySQL es un sistema gestor de base de datos multihilo y multiusuario. Ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.

Es un potente gestor de base de datos pero al ser un software que necesita ser comprado no es óptimo para utilizarse en la creación de la solución informática propuesta. Se muestra a continuación una variante

libre PostgreSQL que es además un gestor muy poderoso y por lo tanto cumple con las condiciones necesarias para utilizarse, las cuales se detallan a continuación:

1.2.5.2 PostgreSQL

Es un potente sistema de base de datos objeto-relacional de código abierto. Se ejecuta en todos los principales sistemas operativos. Tiene soporte completo para claves foráneas, uniones, vistas, disparadores y procedimientos almacenados (en varios idiomas). Incluye la mayoría de los tipos de datos incluyendo INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, y TIMESTAMP. También soporta almacenamiento de objetos binarios grandes, como imágenes, sonidos o vídeo. Cuenta además con una documentación excepcional.

PostgreSQL, cuenta con características avanzadas tales como: punto de recuperación en el tiempo, replicación asincrónica, transacciones anidadas (puntos de retorno), copias de seguridad en línea, un planeador de consultas sofisticadas. Hay sistemas activos de PostgreSQL en entornos de producción que manejan más de 4 terabytes de datos.

La infraestructura GiST (*Generalized Search Tree o árbol generalizado de búsqueda*) proporciona a PostgreSQL una interfaz que permite tanto la creación de tipos de datos personalizados, así como los métodos de consulta extensible con la que buscarlos. Por lo tanto, GiST ofrece la flexibilidad para especificar lo que se almacena y cómo almacenarlo.

PostgreSQL cuenta con una comunidad de desarrolladores y organizaciones comerciales que trabajan en su desarrollo.

El seguimiento y mantenimiento de las bases de datos que se manejan en una aplicación son de vital importancia ya que estas son las encargadas de manejar una de las partes más sensibles de toda aplicación que son sus datos, los cuales deben ser protegidos de ataques y deben estar almacenados en lugares seguros y de los cuales debe asegurarse su existencia e integridad.

Una vez abordados los servidores web más utilizados, los diferentes lenguajes de programación existentes, los SGBD y los CMS trataremos entonces herramientas y metodologías que facilitan el desarrollo de las aplicaciones web.

1.2.6 Los entornos de desarrollo integrado o Integrated Development Environment (IDE)

No son más que editores de código, compiladores, depuradores y constructores de interfaz gráfica, es posible que un mismo IDE pueda funcionar con varios lenguajes de programación. Los IDE sirven de gran ayuda a los desarrolladores de las aplicaciones ya que permiten que se optimice el tiempo en que se desarrolla y también permiten un manejo y comprensión de los errores con más facilidad. Dentro de los IDEs más utilizados en el mundo se encuentran NetBeans y Eclipse.

1.2.6.1 NetBeans

Es un entorno de desarrollo integrado exitoso de código abierto, posee una comunidad en constante crecimiento. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio del 2000 y continúa siendo el patrocinador principal de los proyectos. NetBeans IDE 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 el NetBeans IDE.

1.2.6.2 Eclipse

Es un entorno de desarrollo integrado de código abierto, multiplataforma, mediante plugins se pueden incluir soporte para los diferentes lenguajes de programación, la compilación que realiza Eclipse es en tiempo real, posee en su sitio oficial una amplia documentación, posee una gran comunidad de desarrollo. Este IDE ofrece el control del editor de código, del compilador y del depurador desde una única interfaz de usuario. Su misión consiste en evitar tareas repetitivas, facilitar la escritura correcta del código, disminuir el tiempo de depuración e incrementar la productividad del desarrollador.

Una vez analizadas ambos IDEs se escoge Eclipse para facilitar la escritura del código necesario para la implementación del Módulo de Pruebas.

Antes de comenzar el desarrollo de una aplicación es necesario definir la metodología que más se ajusta a sus exigencias, pues así se logrará optimizar el proceso y lograr un producto final con calidad y documentado según las exigencias y necesidades del cliente.

1.2.7 Metodologías

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software. (3)

Considerando la filosofía de desarrollo de las metodologías que tengan un mayor énfasis en la planificación y control del proyecto, en especificación precisa de requisitos y modelado, reciben el apelativo de Metodologías Tradicionales. Otras, denominadas Metodologías Ágiles, están más orientadas a la generación de código con ciclos muy cortos de desarrollo, se dirigen a equipos de desarrollo pequeños, hacen especial hincapié en aspectos humanos asociados al trabajo en equipo e involucran activamente al cliente en el proceso. A continuación se muestra una tabla en la cual se realiza una comparación entre estos dos tipos de metodologías tomando en cuenta diferentes aspectos.

Tabla 1: Diferencias entre metodologías ágiles y no ágiles

| Características | Metodologías Ágiles | Metodologías Tradicionales |
|--|---|---|
| Facilidad de cambios en el software | Especialmente preparados para cambios durante el Proyecto. | Cierta resistencia a los cambios. |
| Restricciones en el proceso | Proceso menos controlado, con pocos principios. | Proceso mucho más controlado, con numerosas políticas. |
| Contratos | No existe contrato tradicional o al menos es bastante flexible | Existe un contrato prefijado. |
| Papel del cliente en el equipo de desarrollo | El cliente es parte del equipo de desarrollo. | El cliente interactúa con el equipo de desarrollo mediante reuniones. |
| Cantidad de integrantes del equipo de trabajo y su localización | Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio. | Grupos grandes y posiblemente distribuidos. |
| Cantidad de Artefactos | Pocos artefactos. | Más artefactos. |
| Cantidad de roles | Pocos roles. | Más roles. |
| Importancia de la arquitectura del software | Menos énfasis en la arquitectura del software. | La arquitectura del software es esencial y se expresa mediante modelos. |

1.2.7.1 Programación Extrema (XP)

XP es una metodología ágil, diseñada para grupos pequeños de desarrollo, potencia las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo. XP se basa en retroalimentación continua entre el cliente y el equipo de desarrollo por lo que es estrictamente necesario que el cliente forme parte del equipo de trabajo. XP permite la flexibilidad ante el cambio por lo

que si en cualquier etapa del desarrollo se necesita hacer algún cambio en el software el costo será el mínimo.

La fase de desarrollo de XP se divide en cuatro etapas:

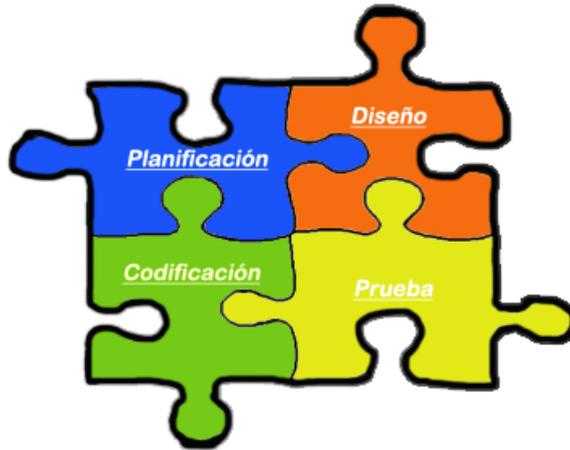


Ilustración 1: Fases de desarrollo de XP

- **Planificación**

Las historias de usuario: La creación de las historias de usuario es el primer paso en cualquier proyecto que siga esta metodología, en XP son de vital importancia ya que mediante estas se determinarán los requisitos del software que se implementará, estas historias de usuarios se descomponen en tareas de programación las que serán entregadas a los desarrolladores, las mismas deben concebirse de manera que el programador pueda darles solución en el período aproximado de una iteración.

Release planning o Planificación de publicaciones: En esta se indican las historias de usuario que se crearán para cada versión del programa, se definen además los tiempos de implementación ideales de las historias de usuario y la prioridad.

Iteraciones: Las iteraciones en XP deben ser de alrededor de tres semanas cada una, al iniciarse cada iteración se definirán las historias de usuarios que se implementarán, además se seleccionan las historias de usuario que no pasaron las pruebas de aceptación de la iteración anterior.

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 para saber las historias que se pueden desarrollar en las distintas iteraciones. Usando la velocidad del proyecto se controlará que todas las tareas se puedan desarrollar en el tiempo del que dispone la iteración.

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

- **Diseño**

Diseños simples: La metodología XP sugiere que hay que conseguir diseños simples y sencillos, para que puedan ser entendidos e implementados con facilidad, lo cual se traducirá en ahorro de tiempo.

Glosarios de términos: Usar glosarios de términos y una correcta especificación de los nombres de métodos y clases ayudará a comprender el diseño y facilitará sus posteriores ampliaciones y la reutilización del código.

Riesgos: Si se encuentra algún error es necesario que los desarrolladores investiguen con rapidez para así reducir al mínimo el riesgo que provoca este error.

Funcionalidad extra: Añadir funcionalidades extras es una pérdida de tiempo para el equipo de desarrollo.

Refactorizar: Es mejorar y modificar la estructura y codificación de códigos ya creados sin alterar su funcionalidad si van a ser reutilizados.

Tarjetas C.R.C: El uso de las tarjetas C.R.C (**C**lass, **R**esponsabilities and **C**ollaboration) permiten al programador desprenderse del método de trabajo basado en procedimientos y trabajar con una metodología basada en objetos.

- **Codificación**

La codificación debe hacerse atendiendo a estándares de codificación ya creados. Programar bajo estándares mantiene el código consistente y facilita su comprensión.

- **Prueba**

Las pruebas de aceptación son creadas y usadas por los clientes para comprobar que las distintas historias de usuario cumplen su cometido.

1.2.7.2 Rational Unified Process (RUP)

Es una metodología tradicional propiedad de IBM, es una de las metodologías de desarrollo más general de las que existen en la actualidad, requiere un equipo de desarrollo grande, los procesos son iterativos e incrementales, están centrado en la arquitectura y son guiado por los casos de uso, el ciclo de vida de RUP se organiza en fases e iteraciones, cada fase finaliza con un hito, en cada una de sus fases genera una serie de artefactos los cuales sirven para comprender mejor el sistema.

El ciclo de vida de RUP se divide en cuatro fases y nueve flujos de trabajo:

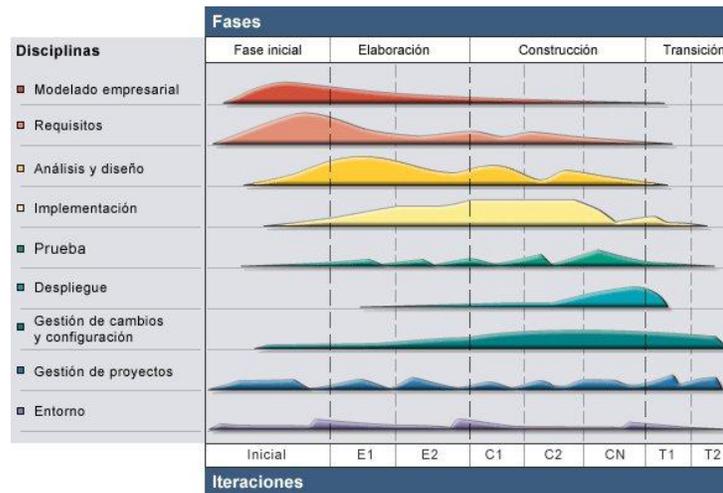


Ilustración 2: Fases del ciclo de vida de RUP (4)

Fases:

- **Inicio:**

En la fase de inicio se realiza una identificación de riesgos y un análisis inicial del negocio para saber si es más factible construir el software o comprarlo y para definir si se sigue adelante con el proyecto o se abandona.

- **Elaboración:**

En la fase de elaboración se desarrolla la línea base de la arquitectura, y se realiza la mitigación de los mayores riesgos que fueron identificados en la fase de inicio.

- **Construcción:**

En la fase de construcción se lleva a cabo la implementación de los diferentes casos de uso y se desarrollan los casos de prueba que servirán como guía para la realización de las pruebas de software.

- **Transición:**

En la fase de transición se finalizan los artefactos y se prepara una versión beta, la finalidad de la fase de transición es poner el producto en manos de los usuarios finales.

Flujos de trabajo:

Los flujos de trabajo de RUP se dividen en flujos de ingeniería y apoyo. Estos flujos de trabajo se recorren una y otra vez durante las diferentes fases.

Flujos de trabajo de ingeniería:

- Modelado del negocio
- Requisitos
- Análisis y diseño
- Implementación
- Pruebas
- Despliegue

Los flujos de trabajo de apoyo son:

- Administración del proyecto
- Configuración y control de cambios
- Entorno

Estudiadas las metodologías de desarrollo descritas anteriormente y analizando las características que posee la solución propuesta se decide, utilizar como metodología para guiar todo el proceso del Módulo de Pruebas, la metodología XP pues es una metodología ágil, libre y el cliente forma parte del equipo de trabajo, se ajusta completamente a las especificidades de la solución informática que se desea implementar, debido a que las metodologías ágiles están concebidas para grupos de desarrollo pequeños además se proponen no generar más documentación de la que es realmente importante para lograr un software de calidad, además asegura que la flexibilidad de un proyecto es fundamental para su éxito.

Las herramientas CASE permiten que el proceso de modelado de un software se haga de manera rápida y sencilla lo que permite la optimización del tiempo que se emplea en esta tarea, además permiten que este proceso de modelado se haga mediante una interfaz agradable.

1.2.8 Herramienta Case

Las herramientas CASE (*Computer Aided Software Engineering* o Ingeniería de Software Asistida por Computadora) no son más que herramientas que permiten y facilitan el modelado de los software, permiten reducir el tiempo, coste de desarrollo y mantenimiento de los sistemas informáticos.

1.2.8.1 Rational Rose

Es una herramienta de diseño que soporta la generación de código a partir de modelos en 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.

1.2.8.2 Visual Paradigm

Es una herramienta CASE que utiliza "UML": como lenguaje de modelado. Está diseñada para una amplia gama de usuarios, ofrece amplias características de modelado de casos de uso incluyendo la función completa de UML, proporciona a los desarrolladores una plataforma que les permite diseñar un producto con calidad de forma rápida. Facilita la interoperabilidad con otras herramientas CASE como Rational Rose. Es una de las más potentes herramientas de diseño que existen, es multiplataforma, en ella se pueden construir diferentes tipos de diagramas y además permite la generación de código a partir de un diseño determinado, lo que le confiere un alto grado de usabilidad.

Se definió Visual Paradigm como herramienta de modelado por ser una herramienta libre, fácil de utilizar, con interfaces muy amigables y curvas de aprendizajes muy cortas.

1.2.9 Conclusiones

En este capítulo se realizó una investigación sobre los procesos de pruebas de software y su puesta en práctica en el Grupo de Calidad de FORTES, arrojando estas cuales son las actividades automatizables dentro del proceso de pruebas que se realiza en el Grupo de Calidad de FORTES. Se realizó además un estudio del estado del arte donde se analizaron las características de las diferentes aplicaciones que soportan hoy en día las pruebas de software permitiendo estas la definición de las características que debe poseer la solución propuesta, además se definieron las tecnologías y las herramientas que se utilizarán para el desarrollo del Módulo de Pruebas del Sistema de Gestión de Calidad del Grupo de Calidad de FORTES.

Capítulo 2: Características del sistema

2.1 Introducción

En este capítulo se expondrá el funcionamiento del Módulo de Pruebas del Sistema de Gestión de Calidad del Grupo de Calidad de FORTES, así como las historias de usuario que definirán los requisitos funcionales que debe cumplir el módulo que se implementará y se especificarán los requisitos no funcionales necesarios para que la aplicación pueda satisfacer al cliente, además se expondrá la interacción de las tablas que necesita el Módulo de Pruebas con las creadas por Drupal y se presentarán las principales interfaces de usuario del Módulo de Pruebas.

2.2 Propuesta de sistema

El Sistema de Gestión de la Calidad del Grupo de Calidad de FORTES, es una aplicación web que se realizará para dar solución a un grupo de deficiencias que dificultan los procesos que brinda el Grupo de Calidad de FORTES, para ejercer de forma eficiente los diferentes servicios que este presta.

Para la creación de este sistema se define la creación de cinco módulos que gestionen todos los servicios que se prestan, estos módulos son:

Módulo de Servicios Generales: gestionará toda la información referente a las solicitudes de servicios que se generen por los diferentes proyectos de centro.

Módulo de Pruebas: gestionará todo el proceso de pruebas que se realiza en el Grupo de Calidad de FORTES.

Módulos de Auditoría: gestionará todo el proceso de auditorías que se realiza en el Grupo de Calidad de FORTES.

Módulo de Revisiones: gestionará todo el proceso de revisiones que se realiza en el Grupo de Calidad de FORTES.

Módulo de Capacitación: contribuye a la superación del personal. También controla la participación de los

miembros del Grupo de Calidad de FORTES y mejora el control de la evaluación de los mismos.

Todos estos módulos se integrarán en una aplicación web que además, publicará información de interés sobre el Grupo de Calidad de FORTES y los servicios que presta.

El alcance de esta investigación se centra en la realización de una herramienta web que soporte el proceso de pruebas que se realiza en el Grupo de Calidad de FORTES, así como la gestión de las NC, permitiendo el reporte de estas, ajustándose completamente a las necesidades del Grupo de Calidad de FORTES, debe garantizar además el almacenamiento de estas NC, una vez confeccionadas para su posterior utilización. La herramienta debe permitir también la generación de informes atendiendo a diferentes criterios definidos por el usuario, estos criterios deben ser flexibles, los informes que se generen deben poderse imprimir y exportarlos a formato PDF.

2.3 Objetivos del Módulo de Pruebas

La automatización del proceso de pruebas que se realiza en el Grupo de Calidad de FORTES es de vital importancia ya que permitirá elevar la eficiencia y lograr realizar este proceso de una manera más fácil y rápida, por lo tanto los objetivos que se persiguen con la implementación del Módulo de Pruebas son:

1. Automatizar las actividades que se desarrollan a lo largo del proceso de pruebas, dentro de las que se encuentran: la solicitud de las pruebas, la planificación de las mismas y la gestión de las NC.
2. Generar reportes en formato pdf, atendiendo a diferentes criterios como son: NC encontradas a los diferentes proyectos, productos, NC detectadas por cada uno de los probadores, y NC detectadas en cada una de las pruebas realizadas.
3. Evaluar el trabajo realizado por los probadores.

2.4 Especificación de los requisitos del software

2.4.1 Requisitos funcionales

Los requisitos funcionales de un sistema describen lo que el sistema debe hacer. Estos requisitos funcionales definen los recursos específicos que el sistema debe proporcionar. En algunos casos, los

requisitos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer (5). Las Historias de Usuario son las que definen en la metodología XP los requisitos funcionales que debe cumplir el software que se desarrolla, a continuación se muestran las que guiarán el desarrollo del Módulo de Pruebas.

Historias de Usuario (HU)

Tabla 2: HU - Insertar iteración

| HISTORIA DE USUARIO | |
|--|--|
| Número: 1. | Nombre Historia de Usuario: Insertar iteración. |
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se insertará el nombre de una iteración. | |

Tabla 3: HU - Ver iteración

| HISTORIA DE USUARIO | |
|---|---|
| Número: 2. | Nombre Historia de Usuario: Ver iteración. |
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se consultará el nombre de las iteraciones que se hayan insertado con anterioridad. | |

Tabla 4: HU – Insertar proyecto

| HISTORIA DE USUARIO | |
|--|---|
| Número: 3. | Nombre Historia de Usuario: Insertar proyecto. |
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se insertará el nombre de un proyecto. | |

Tabla 5: HU – Ver proyecto

| HISTORIA DE USUARIO | |
|-------------------------------|--|
| Número: 4. | Nombre Historia de Usuario: Ver proyecto. |
| Usuario: Planificador. | Iteración Asignada: 1. |

| | |
|---|---------------------------|
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se consultará el nombre de los proyectos que se hayan insertado con anterioridad. | |

Tabla 6: HU – Eliminar proyecto

| HISTORIA DE USUARIO | |
|--|---|
| Número: 5. | Nombre Historia de Usuario: Eliminar proyecto. |
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se eliminará un proyecto que se haya insertado con anterioridad. | |

Tabla 7: HU – Insertar producto

| HISTORIA DE USUARIO | |
|--|---|
| Número: 6. | Nombre Historia de Usuario: Insertar producto. |
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se insertará el nombre de un producto. | |

Tabla 8: HU – Ver producto

| HISTORIA DE USUARIO | |
|---|--|
| Número: 7. | Nombre Historia de Usuario: Ver producto. |
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se consultará el nombre de los productos que se hayan insertado con anterioridad. | |

Tabla 9: HU – Eliminar producto

| HISTORIA DE USUARIO | |
|--|---|
| Número: 8. | Nombre Historia de Usuario: Eliminar producto. |
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se eliminará un producto que se haya insertado con anterioridad. | |

Tabla 10: HU – Insertar tipo de prueba

| HISTORIA DE USUARIO | |
|---------------------|---|
| Número: 9. | Nombre Historia de Usuario: Insertar tipo de prueba. |

| | |
|--|-------------------------------|
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se insertará el nombre de un tipo de prueba. | |

Tabla 11: HU – Ver tipo de prueba

| HISTORIA DE USUARIO | |
|--|--|
| Número: 10. | Nombre Historia de Usuario: Ver tipo de prueba. |
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se consultará el nombre de los tipos de pruebas que se hayan insertado con anterioridad. | |

Tabla 12: HU – Solicitar prueba

| HISTORIA DE USUARIO | |
|---|--|
| Número: 11. | Nombre Historia de Usuario: Solicitar prueba. |
| Usuario: Cliente. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 4 días. |
| Descripción: Se solicitará una prueba en la cual se deben registrar los datos: <ul style="list-style-type: none"> • Nombre de la prueba • Tipo de prueba • Nombre de proyecto • Nombre del producto • Fecha • Objetivo | |

Tabla 13: HU – Ver solicitud de prueba

| HISTORIA DE USUARIO | |
|---|---|
| Número: 12. | Nombre Historia de Usuario: Ver solicitud de prueba. |
| Usuario: Usuario autenticado. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se consultarán las solicitudes de prueba que se hayan insertado con anterioridad. | |

Tabla 14: HU – Modificar solicitud de prueba

| HISTORIA DE USUARIO | | |
|---|---|--|
| Número: 13. | Nombre Historia de Usuario: Modificar solicitud de prueba. | |
| Usuario: Planificador. | Iteración Asignada: 1. | |
| Prioridad en Negocio: Alta. | Estimación: 1 día. | |
| Descripción: Se modificará una solicitud de prueba que se haya insertado con anterioridad. | | |

Tabla 15: HU – Eliminar solicitud de prueba

| HISTORIA DE USUARIO | | |
|--|--|--|
| Número: 14. | Nombre Historia de Usuario: Eliminar solicitud de prueba. | |
| Usuario: Planificador. | Iteración Asignada: 1. | |
| Prioridad en Negocio: Alta. | Estimación: 1 día. | |
| Descripción: Se eliminará una solicitud de prueba que se haya insertado con anterioridad. | | |

Tabla 16: HU – Planificar prueba

| HISTORIA DE USUARIO | | |
|---|---|--|
| Número: 15. | Nombre Historia de Usuario: Planificar prueba. | |
| Usuario: Planificador. | Iteración Asignada: 1. | |
| Prioridad en Negocio: Alta. | Estimación: 4 días. | |
| Descripción: Se planificará una prueba en la cual se deben registrar los datos: <ul style="list-style-type: none"> • Nombre de la prueba • Nombre del profesor responsable del proyecto en prueba • Nombre del profesor responsable del Grupo de Calidad • Nombre de los probadores que intervendrán en la prueba • Fecha de inicio de la prueba • Fecha de fin de la prueba • Archivos necesarios para ejecutar la prueba • Observaciones | | |

Tabla 17: HU – Ver prueba planificada

| HISTORIA DE USUARIO | | |
|--------------------------------------|--|--|
| Número: 16. | Nombre Historia de Usuario: Ver prueba planificada. | |
| Usuario: Usuario autenticado. | Iteración Asignada: 1. | |
| Prioridad en Negocio: Alta. | Estimación: 1 día. | |

Descripción: Se consultarán las pruebas que se hayan planificado con anterioridad.

Tabla 18: HU – Modificar prueba planificada

| HISTORIA DE USUARIO | |
|--|--|
| Número: 17. | Nombre Historia de Usuario: Modificar prueba planificada. |
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se modificará una prueba que se haya planificado con anterioridad. | |

Tabla 19: HU – Eliminar prueba planificada

| HISTORIA DE USUARIO | |
|---|---|
| Número: 18. | Nombre Historia de Usuario: Eliminar prueba planificada. |
| Usuario: Planificador. | Iteración Asignada: 1. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se eliminará una prueba que se haya planificado con anterioridad. | |

Tabla 20: HU – Reportar NC

| HISTORIA DE USUARIO | |
|--|---|
| Número: 19. | Nombre Historia de Usuario: Reportar NC. |
| Usuario: Ingeniero en Pruebas de Software. | Iteración Asignada: 2. |
| Prioridad en Negocio: Alta. | Estimación: 4 días. |
| <p>Descripción: Se documentará cada NC encontrada, se deben registrar los datos:</p> <ul style="list-style-type: none"> • Nombre de la prueba • Título de la NC • Prioridad • Estado • Iteración • Descripción • Imagen • Fecha (se insertará automáticamente de la PC) • Nombre del probador (se insertará según el usuario que está registrado) <p>Una vez creada esta NC se enviará una notificación al profesor responsable de las pruebas en el Grupo de Calidad.</p> | |

Tabla 21: HU – Ver NC

| HISTORIA DE USUARIO | |
|--|--|
| Número: 20. | Nombre Historia de Usuario: Ver NC. |
| Usuario: Usuario autenticado. | Iteración Asignada: 2. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se consultarán las NC que se hayan reportado con anterioridad. | |

Tabla 22: HU – Revisar NC

| HISTORIA DE USUARIO | |
|---|--|
| Número: 21. | Nombre Historia de Usuario: Revisar NC. |
| Usuario: Profesor responsable de las pruebas (SET). | Iteración Asignada: 2. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: El profesor al frente de las pruebas por el Grupo de Calidad podrá modificar una NC que se haya reportado con anterioridad. Una vez revisada esta NC se enviará una notificación al profesor responsable del proyecto que se encuentra en prueba. | |

Tabla 23: HU – Responder NC

| HISTORIA DE USUARIO | |
|---|--|
| Número: 22. | Nombre Historia de Usuario: Responder NC. |
| Usuario: Cliente. | Iteración Asignada: 2. |
| Prioridad en Negocio: Alta. | Estimación: 1 día. |
| Descripción: Se mostrará la NC revisada por el profesor a cargo de las pruebas. Se deben llenar los datos: <ul style="list-style-type: none"> • Estado (procede o no procede) • Descripción | |

Tabla 24: HU – Generar reportes

| HISTORIA DE USUARIO | |
|---|--|
| Número: 23. | Nombre Historia de Usuario: Generar reportes. |
| Usuario: Usuario autenticado. | Iteración Asignada: 3. |
| Prioridad en Negocio: Alta. | Estimación: 3 días. |
| Descripción: Se generarán reportes atendiendo a los siguientes criterios: <ul style="list-style-type: none"> • Nombre de proyecto • Nombre de producto | |

- Nombre del probador
- Fecha de inicio
- Fecha de fin
- Prioridad

Estos reportes una vez generados deben permitir al usuario imprimir o exportar a formato pdf.

Estimación de esfuerzos por historias de usuarios

Definidas las HU es necesario que los desarrolladores planifiquen el tiempo que necesitarán para la implementación de cada una de ellas, para así determinar el tiempo total que consumirá la implementación de la aplicación que se está desarrollando.

Tabla 25: Estimación de esfuerzo en días por historia de usuario.

| HISTORIA DE USUARIO | ESTIMACIÓN (DÍAS) |
|-------------------------------|-------------------|
| Insertar iteración | 1 |
| Ver iteración | 1 |
| Insertar proyecto | 1 |
| Ver proyecto | 1 |
| Eliminar proyecto | 1 |
| Insertar producto | 1 |
| Ver producto | 1 |
| Eliminar producto | 1 |
| Insertar tipo de prueba | 1 |
| Ver tipo de prueba | 1 |
| Solicitar prueba | 4 |
| Ver solicitud de prueba | 1 |
| Modificar solicitud de prueba | 1 |
| Eliminar solicitud de prueba | 1 |
| Planificar prueba | 4 |
| Ver prueba planificada | 1 |
| Modificar prueba planificada | 1 |
| Eliminar prueba planificada | 1 |
| Reportar NC | 4 |
| Ver NC | 1 |
| Revisar NC | 1 |

| | |
|------------------|-----------|
| Responder NC | 1 |
| Generar reportes | 3 |
| TOTAL | 34 |

Después de analizado el tiempo que se necesitará para implementar cada una de las HU descritas, se determinó que es necesario un total de 34 días laborales para implementar el Módulo de Pruebas.

Plan de duración de las Iteraciones

Una vez determinado el tiempo que consumirá desarrollar cada una de las HU, es necesario ubicar las mismas en iteraciones, dependiendo de la importancia que tengan para el desarrollo del negocio cada una de estas.

Para la creación del Módulo de Pruebas se decidió dividir la implementación en tres iteraciones, en la primera iteración se desarrollarán todas aquellas funcionalidades que son necesarias para poder realizar una solicitud de prueba, y para poder realizar después la planificación de las mismas. En la segunda iteración se implementarán todas aquellas funcionalidades necesarias para realizar el reporte de las NC y posteriormente su gestión. En la tercera iteración se implementarán las funcionalidades imprescindibles para generar automáticamente los reportes.

Tabla 26: Estimación de esfuerzo en días por iteración.

| ITERACIÓN | HISTORIA DE USUARIO | ESTIMACIÓN (DÍAS) |
|-----------|-------------------------|-------------------|
| 1 | Insertar iteración | 1 |
| | Ver iteración | 1 |
| | Insertar proyecto | 1 |
| | Ver proyecto | 1 |
| | Eliminar proyecto | 1 |
| | Insertar producto | 1 |
| | Ver producto | 1 |
| | Eliminar producto | 1 |
| | Insertar tipo de prueba | 1 |
| | Ver tipo de prueba | 1 |

| | | |
|----------|----------------------------------|-----------|
| | Solicitar prueba | 4 |
| | Ver solicitud de prueba | 1 |
| | Modificar solicitud de prueba | 1 |
| | Eliminar solicitud de prueba | 1 |
| | Planificar prueba | 4 |
| | Ver prueba planificada | 1 |
| | Modificar prueba planificada | 1 |
| | Eliminar prueba planificada | 1 |
| | Total de días iteración 1 | 24 |
| | Reportar NC | 4 |
| | Ver NC | 1 |
| 2 | Revisar NC | 1 |
| | Responder NC | 1 |
| | Total de días iteración 2 | 7 |
| 3 | Generar reportes | 3 |
| | Total de días iteración 3 | 3 |
| | TOTAL | 34 |

2.4.2 Requisitos no funcionales

Los requisitos no funcionales, como su nombre sugiere, son aquellos requisitos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de este como la fiabilidad, el tiempo de respuesta entre otros. (5)

Software:

- En el cliente
 - Sistema operativo Linux o Windows (98 o superior).
 - Navegador web Mozilla Firefox 3.0 o superior.
- En el servidor
 - Servidor Web Apache.
 - PHP5 o superior.
 - PostgreSQL 8.4 o superior.

Hardware:

Las PC (*Personal Computer* o Computadora Personal) servidora debe poseer las siguientes características:

- Microprocesador Intel Pentium III o superior.
- Disco duro con capacidad de 15GB como mínimo.
- Tarjeta de red cableada, inalámbrica o módem.

Apariencia:

- La herramienta debe ser sencilla, fácil de emplear y muy amigable para el usuario sin saturación de colores ni imágenes.

Usabilidad:

- La herramienta está orientada todos los miembros del Grupo de Calidad de FORTES y al profesor responsable del proyecto que se encuentra en prueba.
- La herramienta será desplegada en un entorno web.

Rendimiento:

- Debe estar disponible las 24 horas del día.

Soporte:

- Documentación necesaria para el uso o desarrollo de las funcionalidades de la herramienta.

Portabilidad:

- El sistema debe ejecutarse en Windows XP, 2000 u otro superior y en cualquier distribución GNU/Linux, es decir el sistema debe ser multiplataforma.

Seguridad:

- El sistema debe proteger la integridad de la información que se maneja.

- Los servidores deben permanecer con un cuidado extremo en los laboratorios de producción.
- Se permitirá la autenticación, la cual será una contraseña de acceso para los usuarios autorizados por el administrador. El sistema informático debe garantizar que la información sea vista únicamente por quien tenga permiso para esto. El sistema debe permitir solo el acceso a cada uno de sus servicios al personal requerido para esto.
- El sistema informará quien es el usuario correspondiente con la sección que esté activa.

Confiabilidad:

- Garantía de un tratamiento adecuado de las excepciones y validación de las entradas del usuario.

Disponibilidad:

- El sistema debe permitirle al usuario conectarse desde cualquier PC de la red.

Interfaz:

- Las interfaces deben ser sencillas y de fácil entendimiento por los usuarios.
- Ninguna interfaz puede ir cargada de muchos colores evitando la desconcentración del usuario.

2.4.3 Diseño de la Base de Datos

El diseño de base de datos que se expone a continuación muestra la interacción de las tablas del Módulo de Pruebas y las tablas que crea por defecto todo sitio realizado con Drupal. Las tablas que están resaltadas en rojo son las que pertenecen a las tablas que crea Drupal cuando se crea un sitio, estas tablas son las que permiten manejar los usuarios y los roles que interactúan en el sitio, siendo estas de vital importancia para definir y manejar los permisos que tendrán cada uno de los usuarios que interactuarán con el Módulo de Pruebas.

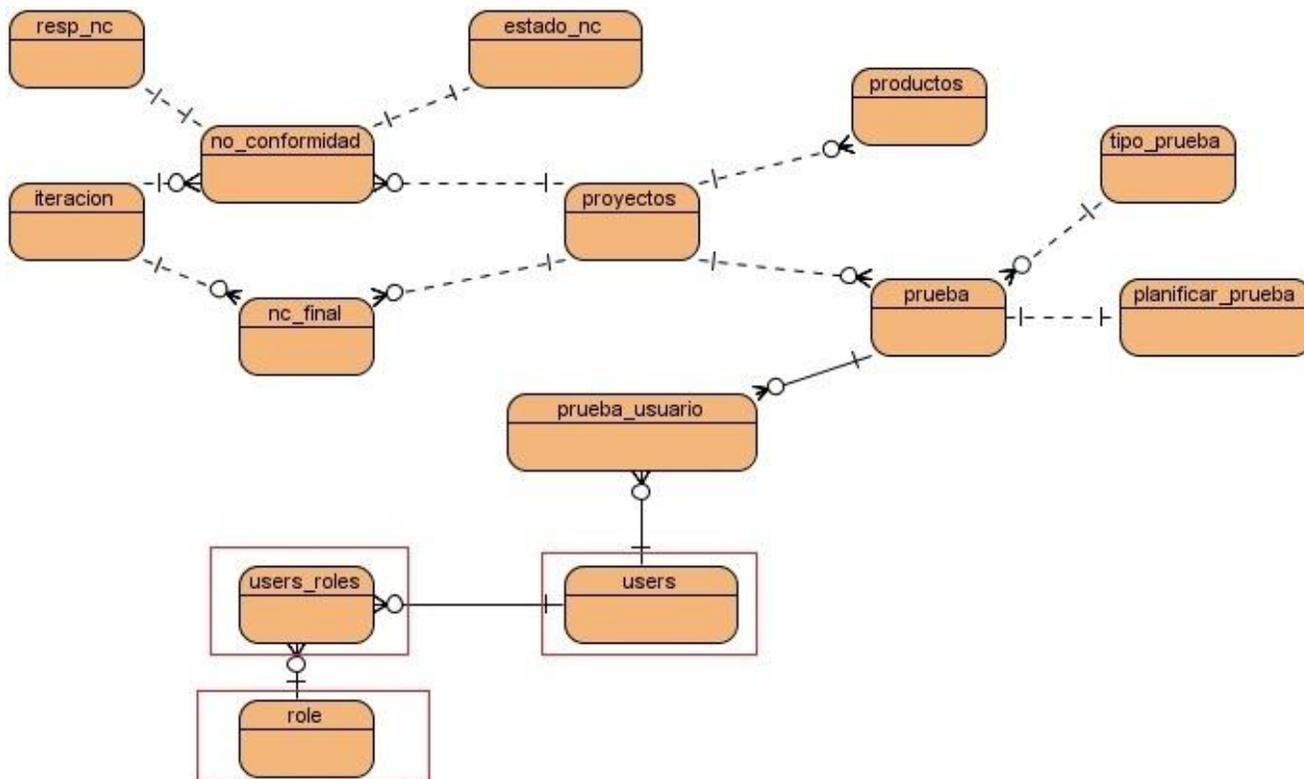


Ilustración 3: Interacción de tablas del Módulo de Pruebas y de Drupal

A continuación se muestra una tabla en la que se describirán la información que se almacena en cada una de las tablas que se utilizan en el Módulo de Pruebas.

Tabla 27: Datos que almacenan las tablas que se utilizan en el Módulo de Pruebas

| NOMBRE DE LA TABLA | INFORMACIÓN QUE ALMACENA |
|--------------------|---|
| iteracion | Se almacenan los números de iteración que inserta el Planificador mediante la IU "Insertar iteración". |
| proyectos | Se almacenan los nombres de los proyectos que inserta el Planificador mediante la IU "Insertar proyecto". |
| productos | Se almacenan los nombres de los productos que inserta el Planificador mediante la IU "Insertar producto". |
| tipo_prueba | Se almacenan los tipos de pruebas, que se realizan en el Grupo de Calidad de FORTES, estos se insertan mediante la IU "Insertar tipos de prueba". |
| prueba | Se almacenan todos los datos necesarios para crear una |

| | |
|--------------------------|---|
| | solicitud de prueba, estos se insertan mediante la IU "Solicitud de prueba". |
| planificar_prueba | Se almacenan todos los datos necesarios para planificar una prueba que ha sido solicitada con anterioridad, estos se insertan mediante la IU "Planificar prueba". |
| prueba_usuario | Se almacenan los usuarios que participarán en una prueba determinada, estos se insertan mediante la IU "Planificar prueba". |
| estado_nc | Se almacenan los diferentes estados por los que transita una no conformidad (creada, revisada, aceptada, rechazada). |
| no_conformidad | Se almacenan todos los datos necesarios para reportar una no conformidad, estos se insertan mediante la IU "Reportar no conformidad". |
| nc_final | Se almacenan los datos de una no conformidad una vez que la misma ha sido revisada, estos se insertan mediante la IU "Evaluar no conformidad". |
| resp_nc | Se almacena la respuesta que le otorga el Cliente a la no conformidad, estos se insertan mediante la IU "Responder no conformidad" |
| users | Se almacenan los diferentes usuarios que interactúan con el Módulo de Pruebas. |
| roles | Se almacenan los diferentes roles que se han definido por el administrador del Módulo de Pruebas. |
| users_roles | Se almacenan los roles que desempeñan cada uno de los usuarios que interactúan con el Módulo de Pruebas. |

2.4.4 Diagrama de despliegue

Un diagrama de despliegue permite apreciar de forma visual cómo interactúan físicamente los componentes de una aplicación. Para el caso específico del "Módulo de Pruebas", el mismo estará alojado en una PC servidor, la cual tendrá contenido un servidor web Apache, que estará conectado a una base de datos PostgreSQL (que se encuentra almacenada en la propia PC servidor), necesaria para acceder a los diferentes datos que esta almacena, los cuales son de vital importancia para responder las peticiones que se realizan, la PC servidor le brinda servicios a las estaciones de trabajo o PC clientes que no son más que todas aquellas PC que realicen peticiones mediante el protocolo HTTP, además las estaciones de trabajo se podrán conectar mediante el protocolo TCP/IP a las diferentes impresoras que se

encuentran compartidas en la red de la UCI para imprimir los informes que se generen.

En el Grupo de Calidad de FORTES la PC servidor será la PC que tiene destinada el mismo como servidor para almacenar toda la información sensible, las estaciones de trabajo se encuentran en el laboratorio donde radica el Grupo de Calidad, o dentro de la red de la UCI.

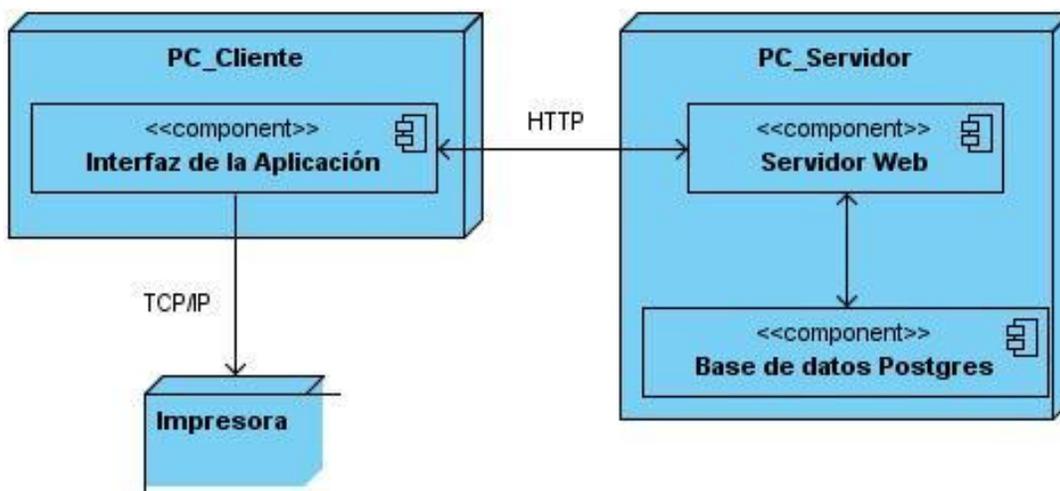


Ilustración 4: Diagrama de despliegue del "Módulo de Pruebas"

2.4.5 Interfaz de usuario (IU)

A continuación se muestran algunas de las interfaces de usuario que están presente en el Módulo de Pruebas.

Insertar iteración

Insertar Iteración: *

Introduzca el nombre de la Iteración

Insertar

Ilustración 5: IU Insertar iteración

Ver iteraciones

Iteraciones

| # | Nombre de la Iteración |
|---|------------------------|
| 1 | 1 |
| 2 | 2.1 |
| 3 | 2 |

Ilustración 6: IU Ver iteraciones

Insertar proyecto

Insertar Proyecto: *

Alfaomega

Introduzca el nombre del proyecto

Insertar

Ilustración 7: IU Insertar proyecto



Ilustración 8: IU Ver proyectos



Ilustración 9: IU Insertar productos



Ilustración 10: IU Ver productos

2.4.6 Conclusiones:

Durante este capítulo se expusieron las principales características que debe cumplir el Módulo de Pruebas según las especificaciones del cliente, así como la estimación de días que llevará implementar las funcionalidades que permitirán el funcionamiento óptimo de este módulo, además se mostró la interacción entre las tablas creadas por Drupal para poder gestionar los sitios y las tablas que necesita el Módulo de Pruebas para poder funcionar y se presentaron algunas de las interfaces de usuario que posee el Módulo de Pruebas.

Capítulo 3: Implementación y pruebas

3.1 Introducción

En este capítulo se expondrán aspectos relacionados con la construcción del Módulo de Pruebas, se definirán los roles que intervendrán en el Módulo de Pruebas, los cuales son de gran importancia para poder controlar la seguridad de las acciones que se realizan, además se definirán los casos de pruebas necesarios para poder ejecutar las pruebas al software para así validar la calidad del mismo, asegurándose el cumplimiento de todos los requisitos.

3.2 Roles

Los roles en Drupal ofrecen la posibilidad de asignar permisos específicos a un grupo de usuarios, permiten ajustar la seguridad, uso y administración de Drupal. A los usuarios asignados a un rol determinado se les conceden los permisos asignados a este rol. Este método para brindar permisos a los usuarios ofrece la gran ventaja de que no es necesario, establecer permiso a los usuarios uno por uno, lo que optimiza este proceso y disminuye la posibilidad de errores en este proceso. Es de vital importancia definir los roles que intervendrán en el Módulo de Prueba para así poder otorgar los permisos necesarios y asegurar de esta manera la seguridad.

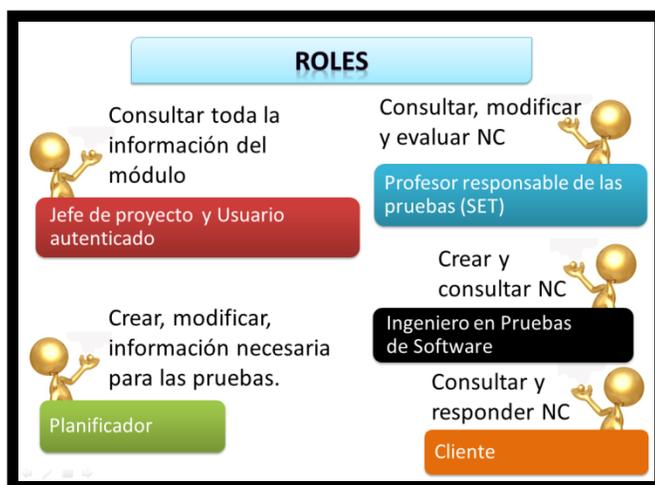


Ilustración 11: Roles que intervienen en el Módulo de Pruebas

3.3 Módulos de apoyo

Para poder desarrollar una solución que cumpliera con todos los requisitos especificados por el cliente, se necesitó el apoyo de cuatro módulos, desarrollados para Drupal con anterioridad por el amplio grupo de desarrolladores que crean soluciones para compartir con la comunidad que utiliza este CMS para el desarrollo de aplicaciones web. Los módulos de apoyo utilizados fueron:

1. SMTP: Este módulo sirve de soporte al protocolo que lleva el mismo nombre, el cual se utiliza para el envío de notificaciones por correo electrónico.
2. 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 la aplicación web.
3. TCPDF: Este módulo se utiliza para la creación de documentos en formato pdf, este módulo se utilizó específicamente en la creación de los reportes.
4. DATE: Este módulo se utiliza como soporte para facilitar la inserción de fechas en las aplicaciones web.



Ilustración 12: Utilización del Módulo DATE

3.4 Casos de prueba (CP)

Caso de Prueba Historia de Usuario: Insertar producto

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Insertar producto. En esta historia de usuario se inserta el nombre de un producto en la base de datos.

Tabla 28: CP HU Insertar producto

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|--|
| Número de caso de prueba: 6. | Número de Historia de Usuario: 6. |
| Descripción de la Prueba: El Planificador selecciona la opción "Insertar producto", se muestra la IU Insertar producto, debe permitir escoger el nombre del proyecto al que pertenece e insertar el nombre de un producto. Casos en que se muestran errores: Al introducirse caracteres que no sean alfanuméricos. Al introducirse un nombre de producto que ha sido insertado con anterioridad. Al dejar vacío alguno de los campos obligatorios. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. | |
| Entrada / Pasos de ejecución: Se inserta el nombre del producto. | |
| Resultado Esperado: Se guarda el nombre del producto en la base de datos y se muestra el mensaje de confirmación: "Producto insertado correctamente". | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Ver producto

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Ver producto. En esta historia de usuario se muestran los nombres de los productos y los nombres de los proyectos a los que están asociados existentes en la base de datos.

Tabla 29: CP HU Ver producto

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|--|
| Número de caso de prueba: 7. | Número de Historia de Usuario: 7. |
| Descripción de la Prueba: El Planificador selecciona la opción "Ver productos", se muestran los nombres de los productos existentes en la base de datos así como los nombres de los proyectos a los que están asociados, permite la opción de eliminar los productos que se muestran (ver: Caso de Prueba Historia de Usuario: Eliminar producto). | |

| |
|---|
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. |
| Entrada / Pasos de ejecución: Se muestran los nombres de productos existentes en la base de datos. |
| Resultado Esperado: Se muestran los nombres de proyectos existentes en la base de datos. |
| Evaluación de la Prueba: Prueba satisfactoria. |

Caso de Prueba Historia de Usuario: Eliminar producto

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Eliminar producto. En esta historia de usuario se elimina el nombre de la base de datos del producto que se ha seleccionado.

Tabla 30: CP HU Eliminar producto

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|-----------------------------------|
| Número de caso de prueba: 8. | Número de Historia de Usuario: 8. |
| Descripción de la Prueba: El Planificador selecciona la opción "Ver productos" y se muestran todos los nombres de los productos existentes en la base de datos así como el nombre del proyecto al que está asociado, el planificador selecciona la opción "Eliminar", se muestra una alerta donde se especifica que la acción no se puede deshacer y se pide la confirmación de la acción. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. | |
| Entrada / Pasos de ejecución: Se elimina el nombre del producto seleccionado, así como toda la información asociada a este que exista en la base de datos. | |
| Resultado Esperado: Se elimina el nombre del producto seleccionado de la base de datos, se eliminarán además las solicitudes de prueba, las pruebas planificadas y las no conformidades asociadas al producto eliminado. | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Insertar tipo de prueba

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Insertar tipo de prueba. En esta historia de usuario se inserta el nombre de un tipo de prueba en la base de datos.

Tabla 31: CP HU Insertar tipo de prueba

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|-----------------------------------|
| Número de caso de prueba: 9. | Número de Historia de Usuario: 9. |
| Descripción de la Prueba: El Planificador selecciona la opción "Insertar tipo de prueba", se muestra la IU Insertar tipo de prueba, debe permitir insertar el nombre de un tipo de prueba. Casos en que se muestran errores: Al introducirse caracteres que no sean letras. Al introducirse un nombre de un tipo de prueba que ha sido insertado con anterioridad. Al dejar vacío el campo obligatorio. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. | |
| Entrada / Pasos de ejecución: Se inserta el nombre del tipo de prueba. | |
| Resultado Esperado: Se guarda el nombre del tipo de prueba en la base de datos y se muestra el mensaje de confirmación: " Tipo de prueba insertado correctamente ". | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Ver tipo de prueba

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Ver tipo de prueba. En esta historia de usuario se muestran los nombres de los tipos de prueba existentes en la base de datos.

Tabla 32: CP HU Ver tipo de prueba

| CASO DE PRUEBA DE ACEPTACIÓN | |
|--|------------------------------------|
| Número de caso de prueba: 10. | Número de Historia de Usuario: 10. |
| Descripción de la Prueba: El Planificador selecciona la opción "Ver tipo de prueba", se muestran los nombres de los tipos de prueba existentes en la base de datos. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. | |
| Entrada / Pasos de ejecución: Se muestran los nombres de tipos de prueba existentes en la base de datos. | |
| Resultado Esperado: Se muestran los nombres de tipos de prueba existentes en la base de datos. | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Solicitar prueba

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Solicitar

prueba. En esta historia de usuario se insertan todos los datos necesarios para crear una solicitud de prueba.

Tabla 33: CP HU Solicitar prueba

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Número de caso de prueba: 11. | Número de Historia de Usuario: 11. |
| <p>Descripción de la Prueba: El Cliente selecciona la opción "Solicitud de prueba", se muestra la IU Solicitud de prueba, permite insertar los siguientes datos:</p> <ul style="list-style-type: none"> • Nombre de la prueba que se realizará • Tipo de prueba que solicita • Nombre del proyecto • Nombre del producto • La fecha en que desea que se realicen las pruebas • Objetivo que persigue con las pruebas <p>Casos en que se muestran errores: Al dejar vacío alguno de los campos obligatorios. Al escoger una fecha que sea anterior a la del día en que se realiza la solicitud.</p> | |
| <p>Condiciones de Ejecución: Que la sesión activa tenga permisos de Cliente.</p> | |
| <p>Entrada / Pasos de ejecución: Se completan los datos que se solicitan en el formulario y una vez enviado el formulario se insertan en la base de datos y se envía una notificación al planificador.</p> | |
| <p>Resultado Esperado: Se inserta una solicitud de prueba en la base de datos y se envía una notificación al planificador.</p> | |
| <p>Evaluación de la Prueba: Prueba satisfactoria.</p> | |

Caso de Prueba Historia de Usuario: Ver solicitud de prueba

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Ver solicitud de prueba. En esta historia de usuario se muestran las solicitudes de prueba existentes en la base de datos.

Tabla 34: CP HU Ver solicitud de prueba

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Número de caso de prueba: 12. | Número de Historia de Usuario: 12. |
| Descripción de la Prueba: El Usuario autenticado selecciona la opción " Ver solicitud de prueba", se muestran las solicitudes de prueba existentes en la base de datos. Permite las opciones "Modificar" y "Eliminar" (ver: Caso de Prueba Historia de Usuario: Modificar solicitud de prueba y Caso de Prueba Historia de Usuario: Eliminar solicitud de prueba). | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Usuario autenticado. | |
| Entrada / Pasos de ejecución: Se muestran los nombres de las solicitudes de prueba existentes en la base de datos. | |
| Resultado Esperado: Se muestran los nombres de las solicitudes de prueba existentes en la base de datos. | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Modificar solicitud de prueba

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Modificar solicitud de prueba. En esta historia de usuario se podrán modificar todos los datos de una solicitud de prueba.

Tabla 35: CP HU Modificar solicitud de prueba

| CASO DE PRUEBA DE ACEPTACIÓN | |
|--|---|
| Número de caso de prueba: 13. | Número de Historia de Usuario: 13. |
| Descripción de la Prueba: El Planificador selecciona la opción "Ver solicitudes de prueba", luego escoge la opción "Modificar" se muestra la IU Modificar solicitud de prueba, se muestra un formulario que tiene cargados los datos que posee la solicitud de prueba que se desea modificar, permite modificar los siguientes datos: <ul style="list-style-type: none"> • Nombre de la prueba que se realizará • Tipo de prueba que se ejecutará • Nombre del proyecto • Nombre del producto • La fecha en que desea que se realicen las pruebas • Objetivo que persigue con las pruebas | |
| Casos en que se muestran errores: Al dejar vacío alguno de los campos obligatorios. | |

Al escoger una fecha que sea anterior a la del día en que se realiza la solicitud.

Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador y que se haya insertado el proyecto, producto, tipo de prueba y una solicitud de prueba con anterioridad en la base de datos.

Entrada / Pasos de ejecución: Se completan los datos que se solicitan en el formulario y una vez enviado el formulario se insertan en la base de datos y se envía una notificación al planificador.

Resultado Esperado: Se modifica la solicitud de prueba seleccionada en la base de datos y se envía una notificación al planificador.

Evaluación de la Prueba: Prueba satisfactoria.

Caso de Prueba Historia de Usuario: Modificar prueba planificada

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Modificar prueba planificada. En esta historia de usuario se podrán modificar todos los datos de una prueba que se ha planificado con anterioridad.

Tabla 36: CP HU Modificar prueba planificada

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Número de caso de prueba: 17. | Número de Historia de Usuario: 17. |
| Descripción de la Prueba: El Planificador selecciona la opción "Ver pruebas planificadas", luego escoge la opción "Modificar" se muestra la IU Modificar pruebas planificadas, se muestra un formulario que tiene cargados los datos que posee la solicitud de prueba que se desea modificar, permite modificar los siguientes datos: | |
| <ul style="list-style-type: none">• Nombre de la prueba que se planificará• Nombre del profesor responsable del proyecto que se probará• Nombre del profesor responsable por el proyecto de Calidad• Nombre de los probadores que intervendrán en la prueba• Fecha de inicio de la prueba• Fecha de fin de la prueba• Archivos necesarios para realizar la prueba• Observaciones | |
| Casos en que se muestran errores: | |
| Al dejar vacío alguno de los campos obligatorios. | |
| Al escoger una fecha de inicio que sea anterior a la del día en que se realiza la planificación. | |
| Al escoger una fecha de fin que sea menor que la fecha de inicio. | |

| |
|---|
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador y que se hayan planificado una prueba con anterioridad. |
| Entrada / Pasos de ejecución: Se completan los datos que se solicitan en el formulario y una vez enviado el formulario se insertan en la base de datos y se envía una notificación a todos los usuarios que intervendrán en la prueba. |
| Resultado Esperado: Se modifica la solicitud de prueba seleccionada en la base de datos y se envía una notificación a todos los usuarios que intervendrán en la prueba. |
| Evaluación de la Prueba: Prueba satisfactoria. |

Caso de Prueba Historia de Usuario: Eliminar prueba planificada

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Eliminar solicitud de prueba. En esta historia de usuario se elimina la solicitud de prueba escogida de la base de datos del producto que se ha seleccionado.

Tabla 37: CP HU Eliminar prueba planificada

| CASO DE PRUEBA DE ACEPTACIÓN | |
|--|------------------------------------|
| Número de caso de prueba: 18. | Número de Historia de Usuario: 18. |
| Descripción de la Prueba: El Planificador selecciona la opción "Ver pruebas planificadas", luego escoge la opción "Eliminar" se muestra una alerta donde se especifica que la acción no se puede deshacer y se pide la confirmación de la acción. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador y que se hayan planificado pruebas con anterioridad. | |
| Entrada / Pasos de ejecución: Se elimina la prueba planificada seleccionada y se muestra el mensaje de confirmación "La prueba planificada ha sido eliminada". | |
| Resultado Esperado: Se elimina la prueba planificada seleccionada de la base de datos, así las no conformidades que se hayan generado. | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Reportar NC

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Reportar NC. En esta historia de usuario se insertan todos los datos necesarios para reportar una NC.

Tabla 38: CP HU Reportar NC

| CASO DE PRUEBA DE ACEPTACIÓN | |
|--|---|
| Número de caso de prueba: 19. | Número de Historia de Usuario: 19. |
| <p>Descripción de la Prueba: El Ingeniero en Pruebas de Software selecciona la opción "Reportar no conformidad", se muestra la IU Reportar no conformidad, permite insertar los siguientes datos:</p> <ul style="list-style-type: none"> • Nombre de la prueba que se está realizando • Título de la NC • Prioridad • Iteración • Descripción de la NC encontrada <p>Casos en que se muestran errores: Al dejar vacío alguno de los campos obligatorios.</p> | |
| <p>Condiciones de Ejecución: Que la sesión activa tenga permisos de Ingeniero en Pruebas de Software y que se haya planificado una prueba con anterioridad.</p> | |
| <p>Entrada / Pasos de ejecución: Se completan los datos que se solicitan en el formulario y una vez enviado el formulario se insertan en la base de datos y se envía una notificación al profesor responsable de las pruebas.</p> | |
| <p>Resultado Esperado: Se insertan los datos necesarios para crear la planificación de una prueba en la base de datos y se envía una notificación al profesor responsable de las pruebas.</p> | |
| <p>Evaluación de la Prueba: Prueba satisfactoria.</p> | |

Caso de Prueba Historia de Usuario: Ver NC

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Ver NC. En esta historia de usuario se muestran los datos referentes a las NC existentes en la base de datos.

Tabla 39: CP HU Ver NC

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Número de caso de prueba: 20. | Número de Historia de Usuario: 20. |
| <p>Descripción de la Prueba: El Usuario autenticado selecciona la opción "Consultar no conformidades", se muestran los datos referentes a las NC existentes en la base de datos.</p> | |

| |
|---|
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Usuario autenticado y que se hayan insertado NC con anterioridad en la base de datos. |
| Entrada / Pasos de ejecución: Se muestran los datos referentes a las NC que existen en la base de datos. |
| Resultado Esperado: Se muestran los datos referentes a las NC existentes en la base de datos. |
| Evaluación de la Prueba: Prueba satisfactoria. |

Caso de Prueba Historia de Usuario: Revisar NC

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Revisar NC. En esta historia de usuario se modifican todos los datos necesarios para revisar una NC.

Tabla 40: CP HU Revisar NC

| CASO DE PRUEBA DE ACEPTACIÓN | |
|--|------------------------------------|
| Número de caso de prueba: 21. | Número de Historia de Usuario: 21. |
| <p>Descripción de la Prueba: El Profesor responsable de las pruebas (SET) selecciona la opción " Evaluar no conformidades", se muestra la IU Evaluar no conformidades, permite modificar los siguientes datos:</p> <ul style="list-style-type: none"> • Nombre de la prueba que se está realizando • Título de la NC • Prioridad • Iteración • Descripción de la NC encontrada • Archivo necesario para validar la NC encontrada • Evaluación de la NC. <p>Casos en que se muestran errores: Al dejar vacío alguno de los campos obligatorios.</p> | |
| <p>Condiciones de Ejecución: Que la sesión activa tenga permisos de Profesor responsable de las pruebas (SET) y que se haya creado una NC con anterioridad.</p> | |
| <p>Entrada / Pasos de ejecución: Se completan los datos que se solicitan en el formulario y una vez enviado el formulario se modifican en la base de datos y se envía una notificación al profesor responsable del proyecto en prueba.</p> | |
| <p>Resultado Esperado: Se insertan los datos necesarios en la base de datos y se envía una</p> | |

notificación al profesor responsable del proyecto en prueba.

Evaluación de la Prueba: Prueba satisfactoria.

Caso de Prueba Historia de Usuario: Generar reportes

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Generar reportes. En esta historia de usuario genera reportes atendiendo a diferentes criterios o filtros.

Tabla 41: CP Generar reportes

| CASO DE PRUEBA DE ACEPTACIÓN | |
|--|---|
| Número de caso de prueba: 22. | Número de Historia de Usuario: 22. |
| Descripción de la Prueba: El Usuario autenticado selecciona la opción "Generar reportes", se muestra la IU Generar reportes, permite escoger varios criterios de búsqueda como son: <ul style="list-style-type: none">• Generar reporte general• Generar reporte por prueba• Generar reporte por proyecto• Generar reporte por producto• Generar reporte por probador | |
| Se muestran las NC que cumplen con el criterio de búsqueda en formato pdf. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Usuario autenticado y que se haya evaluado una NC con anterioridad. | |
| Entrada / Pasos de ejecución: Se muestran las NC encontradas según el criterio de búsqueda en formato pdf. | |
| Resultado Esperado: Se muestran las NC encontradas según el criterio de búsqueda en formato pdf. | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Responder NC

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Responder NC. En esta historia de usuario se consulta una NC y se responde.

Tabla 42: CP Responder NC

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|--|
| Número de caso de prueba: 23 | Número de Historia de Usuario: 23 |
| <p>Descripción de la Prueba: El Cliente selecciona la opción "Responder no conformidades", se muestra la IU Responder no conformidades, permite ver las NC que han sido evaluadas con anterioridad, se selecciona la opción "Responder" permite darle respuesta a la NC seleccionada se gestionan los siguientes datos:</p> <ul style="list-style-type: none"> • Estado • Observaciones <p>Casos en que se muestran errores: Al dejar alguno de los campos vacíos.</p> | |
| <p>Condiciones de Ejecución: Que la sesión activa tenga permisos de Cliente y que se haya evaluado una NC con anterioridad.</p> | |
| <p>Entrada / Pasos de ejecución: Se completan los datos que se solicitan en el formulario y una vez enviado el formulario se modifican en la base de datos y se envía una notificación al profesor responsable del proyecto Calidad de FORTES.</p> | |
| <p>Resultado Esperado: Se insertan los datos necesarios en la base de datos y se envía una notificación al profesor responsable del proyecto Calidad de FORTES.</p> | |
| <p>Evaluación de la Prueba: Prueba satisfactoria.</p> | |

Conclusiones sobre las pruebas

Se ejecutaron dos iteraciones de pruebas de funcionalidad a nivel de sistema para validar la calidad del software, basándose en los Casos de Prueba descritos anteriormente, en la primera iteración se detectaron trece no conformidades las cuales fueron respondidas para ejecutar una segunda iteración, donde se detectaron tres no conformidades las cuales fueron respondidas.

Tabla 43: No conformidades por iteración realizada

| ITERACIÓN | CANTIDAD DE NC DETECTADAS | PRIORIDAD |
|-----------|---------------------------|-----------------------------|
| 1 | 13 | 5 Alta 2 Media 6 Baja |

| | | |
|---|---|------------------|
| 2 | 3 | 1 Alta 2 Baja |
|---|---|------------------|

La tabla anterior muestra en detalle el comportamiento de cada una de las iteraciones realizadas, como se puede observar en la segunda iteración solo se detectaron tres no conformidades, de las cuales solo una alcanzo la prioridad alta, estas no conformidades detectadas fueron respondidas de forma satisfactoria. Teniéndose en cuenta la cantidad de no conformidades detectadas y el nivel de respuesta de las mismas, se decide que el Módulo de Pruebas satisface los requisitos especificados por el cliente y está listo para su utilización en el Grupo de Calidad de FORTES.

3.4 Conclusiones:

En este capítulo se evidenció los diferentes roles que intervienen en el Módulo de Pruebas, una vez definidos estos y determinada cada una de las acciones que pueden realizarse, se pudo gestionar la seguridad de las acciones que se ejecutan en el Módulo de Pruebas, se definieron además los casos de pruebas necesarios para ejecutar la validación de la calidad del mismo, asegurándose el cumplimiento de todos los requisitos.

Conclusiones

Con la realización del presente trabajo de diploma que lleva como título: "Sistema de Gestión de Calidad del Grupo de Calidad de FORTES. Módulo de Pruebas" y la investigación realizada se concluye que:

- Se realizó un estudio del Proceso de Pruebas del Grupo de Calidad de FORTES identificándose las actividades automatizables constituyendo estas la base de la aplicación web.
- Se indagó acerca de los software más utilizados para la gestión de las no conformidades, arrojando esto las características imprescindibles de la aplicación web resultante.
- La herramienta desarrollada permitirá elevar la eficiencia en el Proceso de Pruebas del Grupo de Calidad de FORTES, brindando la posibilidad de dar respuesta a las solicitudes de prueba de los clientes en el menor tiempo posible además se obtendrá como resultado la centralización y el control de todas las No Conformidades encontradas en las pruebas realizadas a un software en el Grupo de Calidad del Centro FORTES.

Recomendaciones

Luego de la investigación realizada, de los objetivos propuestos y de la obtención de la herramienta “Módulo de Pruebas” se recomienda que:

Realizar la integración de la herramienta obtenida con los restantes módulos desarrollados.

Comenzar a emplear para la gestión de las no conformidades en el Grupo de Calidad del Centro FORTES.

Proponer la herramienta a otros Centros Productivos, con el objetivo de garantizar una calidad total en todos los productos y proyectos de la Universidad de las Ciencias Informáticas.

Referencias Bibliográficas

1. **DEPARTAMENTO DE INGENIERÍA DE SOFTWARE, UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS.** Conferencia 1: Proceso de Pruebas.
2. MiTecnológico. [En línea] <http://www.mitecnologico.com/Main/DefinicionDeLenguajeDeProgramacion>.
3. Universidad de Murcia. [En línea] <http://www.um.es/docencia/barzana/IAGP/lagp2.html> .
4. Banny's WebBlog. [En línea] <http://bannysolano.wordpress.com/2010/02/09/rup-en-espanol/>.
5. **Sommerville, Ian.** *Ingeniería del software*. 2005.
8. MPR Systema®. [En línea] <http://www.mprconsulting.net/mprsystema/index.html>.
9. REDMINE. [En línea] www.redmine.org/.

Bibliografía

1. **DEPARTAMENTO DE INGENIERÍA DE SOFTWARE, UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS.** Conferencia 1: Proceso de Pruebas.
2. MiTecnológico. [En línea] <http://www.mitecnologico.com/Main/DefinicionDeLenguajeDeProgramacion>.
3. Universidad de Murcia. [En línea] <http://www.um.es/docencia/barzana/IAGP/lagp2.html> .
4. Banny's WebBlog. [En línea] <http://bannysolano.wordpress.com/2010/02/09/rup-en-espanol/>.
5. **Sommerville, Ian.** *Ingeniería del software*. 2005.
6. Joven Club de Computación y Electrónica. [En línea] <http://www.jovenclub.cu/index.php/es/quienes-somo>.
7. **Wilsoft.** Software para el control de la calidad. [En línea] <http://www.wilsoft-la.com/QAction.htm>.
8. MPR Systema®. [En línea] <http://www.mprconsulting.net/mprsystema/index.html>.
9. REDMINE. [En línea] www.redmine.org/.
10. Drupal. [En línea] <https://www.drupal.org/>.
11. Joomla. [En línea] <http://www.joomla.org/about-joomla.html>.
12. PHP. [En línea] <http://www.php.net/manual/es/intro-what-is.php> .
13. **José H. Canós, Patricio Letelier y M^a Carmen Penadés.** *Métodologías Ágiles en el Desarrollo de Software*. s.l. : Universidad Politécnica de Valencia.
14. **IBM.** [En línea] <http://www-01.ibm.com/software/awdtools/rup/>.
15. MySQL. [En línea] <http://www.mysql.com/>.
16. PostgreSQL . [En línea] <http://www.postgresql.org/about/>.
17. Sun Java. [En línea] http://www.library.tuc.gr/index_es.html.
18. The Apache Software Foundation. [En línea] <http://www.apache.org/>.
19. Eclipse. [En línea] <http://www.eclipse.org/org/>.
20. **OMG.** [En línea] http://www.omg.org/gettingstarted/what_is_uml.htm .

21. *Visual Paradigm*. [En línea] <http://www.visual-paradigm.com/product/vpuml/> .
22. Edgewall Software. [En línea] <http://trac.edgewall.org>.
23. *GNU Operating System*. [En línea] <http://www.gnu.org/licenses/gpl.html>.
24. Python. [En línea] <http://www.python.org/>.
25. Netbeans. [En línea] <http://netbeans.org/>.
26. IBM. [En línea] <http://www-01.ibm.com/software/awdtools/developer/rose/>.

Glosario de términos

Artefactos: un artefacto es un producto tangible resultante del proceso de desarrollo de software. Algunos artefactos como los casos de uso, diagrama de clases u otros modelos UML ayudan a la descripción de la función, la arquitectura o el diseño del software. Otros se enfocan en el proceso de desarrollo en sí mismo, como planes de proyecto, casos de negocios o enfoque de riesgos.

Web Hosting o alojamiento web: es el servicio que provee a los usuarios de Internet un sistema para poder almacenar información, imágenes, vídeo, o cualquier contenido accesible vía web.

HyperText Transfer Protocol o protocolo de transferencia de hipertexto (HTTP): es el protocolo usado en cada transacción de la World Wide Web, es un protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor.

Internet Message Access Protocol o Protocolo de acceso a mensajes de Internet (IMAP): es un protocolo de red de acceso a mensajes electrónicos almacenados en un servidor. Mediante IMAP se puede tener acceso al correo electrónico desde cualquier equipo que tenga una conexión a Internet.

Libre o Software Libre: se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.

Lightweight Directory Access Protocol o Protocolo Ligero de Acceso a Directorios (LDAP): hace referencia a un protocolo a nivel de aplicación el cual permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red. LDAP también es considerado una base de datos (aunque su sistema de almacenamiento puede ser diferente) a la que pueden realizarse consultas.

Multihilo: es una característica que permite a una aplicación realizar varias tareas a la vez (concurrentemente).

Multiplataforma: un software multiplataforma es aquel que puede ser ejecutado en diferentes sistemas operativos.

Multiusuario: es la característica de un sistema operativo o programa que permite proveer servicio y procesamiento a múltiples usuarios simultáneamente.

Simple Mail Transfer Protocol o Protocolo Simple de Transferencia de Correo (SMTP): es un protocolo de red basado en texto utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos

Post Office Protocol o Protocolo de la oficina de correo (POP3): es un protocolo que se utiliza en clientes locales de correo para obtener los mensajes de correo electrónico almacenados en un servidor remoto.

Portabilidad: es una aplicación informática que puede ser utilizada en cualquier ordenador que posea el sistema operativo para el que fue programada sin instalación previa.

Privativo: el software no libre (también llamado software propietario, software privativo, software privado, software con propietario o software de propiedad) se refiere a cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo (con o sin modificaciones), o cuyo código fuente no está disponible o el acceso a éste se encuentra restringido.

Plugins o complemento: es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.

Simple Network Management Protocol o Protocolo Simple de Administración de Red (SNMP): es un protocolo de la capa de aplicación que facilita el intercambio de información de administración entre dispositivos de red. Es parte de la familia de protocolos TCP/IP. SNMP permite a los administradores supervisar el funcionamiento de la red, buscar y resolver sus problemas, y planear su crecimiento.

Anexos

Anexo 1: Casos de Prueba

Caso de Prueba Historia de Usuario: Eliminar solicitud de prueba

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Eliminar solicitud de prueba. En esta historia de usuario se elimina la solicitud de prueba escogida de la base de datos del producto que se ha seleccionado.

Tabla 44: CP HU Eliminar solicitud de prueba

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Número de caso de prueba: 14. | Número de Historia de Usuario: 14. |
| Descripción de la Prueba: El Planificador selecciona la opción "Ver solicitudes de prueba", luego escoge la opción "Eliminar" se muestra una alerta donde se especifica que la acción no se puede deshacer y se pide la confirmación de la acción. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. | |
| Entrada / Pasos de ejecución: Se elimina la solicitud de prueba seleccionada y se muestra el mensaje de confirmación "La solicitud de prueba ha sido eliminada". | |
| Resultado Esperado: Se elimina la solicitud de prueba seleccionado de la base de datos, así como las pruebas que se hayan planificado asociadas a esta solicitud y las no conformidades que se hayan generado. | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Planificar prueba

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Planificar prueba. En esta historia de usuario se insertan todos los datos necesarios para planificar una prueba.

Tabla 45: CP HU Planificar prueba

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Número de caso de prueba: 15. | Número de Historia de Usuario: 15. |
| Descripción de la Prueba: El Planificador selecciona la opción "Planificar prueba", se | |

muestra la IU Planificar prueba, permite insertar los siguientes datos:

- Nombre de la prueba que se planificará
- Nombre del profesor responsable del proyecto que se probará
- Nombre del profesor responsable por el proyecto de Calidad
- Nombre de los probadores que intervendrán en la prueba
- Fecha de inicio de la prueba
- Fecha de fin de la prueba
- Archivos necesarios para realizar la prueba
- Observaciones

Casos en que se muestran errores:

Al dejar vacío alguno de los campos obligatorios.

Al escoger una fecha de inicio que sea anterior a la del día en que se realiza la planificación.

Al escoger una fecha de fin que sea menor que la fecha de inicio.

Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador y que se haya insertado una solicitud de prueba con anterioridad en la base de datos.

Entrada / Pasos de ejecución: Se completan los datos que se solicitan en el formulario y una vez enviado el formulario se insertan en la base de datos y se envía una notificación a los usuarios que intervendrán.

Resultado Esperado: Se insertan los datos necesarios para crear la planificación de una prueba en la base de datos y se envía una notificación a los usuarios que intervendrán.

Evaluación de la Prueba: Prueba satisfactoria.

Caso de Prueba Historia de Usuario: Ver pruebas planificadas

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Ver pruebas planificadas. En esta historia de usuario se muestran los datos referentes a las pruebas planificadas existentes en la base de datos.

Tabla 46: CP HU Ver pruebas planificadas

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|---|
| Número de caso de prueba: 16. | Número de Historia de Usuario: 16. |
| Descripción de la Prueba: El Planificador selecciona la opción " Ver pruebas planificadas", se muestran los datos referentes a las pruebas planificadas existentes en la base de datos. Permite las opciones "Modificar" y "Eliminar" (ver: Caso de Prueba Historia de Usuario: Modificar prueba planificada y Caso de Prueba Historia de Usuario: Eliminar prueba planificada). | |

| |
|---|
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. |
| Entrada / Pasos de ejecución: Se muestran los datos referentes a las pruebas planificadas que existen en la base de datos. |
| Resultado Esperado: Se muestran los datos referentes a las pruebas planificadas existentes en la base de datos. |
| Evaluación de la Prueba: Prueba satisfactoria. |

Caso de Prueba Historia de Usuario: Insertar iteración

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Insertar iteración. En esta historia de usuario se inserta el número de una iteración en la base de datos.

Tabla 47: CP HU Insertar iteración

| CASO DE PRUEBA DE ACEPTACIÓN | |
|--|-----------------------------------|
| Número de caso de prueba: 1. | Número de Historia de Usuario: 1. |
| Descripción de la Prueba: El Planificador selecciona la opción "Insertar iteración", se muestra la IU Insertar Iteración, debe permitir insertar el número de una iteración. Casos en que se muestran errores: Al introducirse caracteres no numéricos. Al introducirse un número de iteración que ha sido insertado con anterioridad. Al dejar vacío el campo obligatorio. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. | |
| Entrada / Pasos de ejecución: Se inserta el número de una iteración. | |
| Resultado Esperado: Se guarda el número de la iteración en la base de datos y se muestra el mensaje de confirmación: "Iteración insertada correctamente". | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Ver iteración

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Ver iteración. En esta historia de usuario se muestran los números de las iteraciones que existen en la base de datos.

Tabla 48: CP HU Ver iteración

| CASO DE PRUEBA DE ACEPTACIÓN | |
|--|--|
| Número de caso de prueba: 2. | Número de Historia de Usuario: 2. |
| Descripción de la Prueba: El Planificador selecciona la opción "Ver Iteración", se muestra los números de las iteraciones existentes en la base de datos. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. | |
| Entrada / Pasos de ejecución: Se muestran los números de iteraciones existentes en la base de datos. | |
| Resultado Esperado: Se muestran los números de iteraciones existentes en la base de datos. | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Insertar proyecto

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Insertar proyecto. En esta historia de usuario se inserta el nombre de un proyecto en la base de datos.

Tabla 49: CP HU Insertar proyecto

| CASO DE PRUEBA DE ACEPTACIÓN | |
|--|--|
| Número de caso de prueba: 3. | Número de Historia de Usuario: 3. |
| Descripción de la Prueba: El Planificador selecciona la opción "Insertar proyecto", se muestra la IU Insertar proyecto, debe permitir insertar el nombre de un proyecto. Casos en que se muestran errores: Al introducirse caracteres que no sean alfanuméricos. Al introducirse un nombre de proyecto que ha sido insertado con anterioridad. Al dejar vacío el campo obligatorio. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. | |
| Entrada / Pasos de ejecución: Se inserta el nombre del proyecto. | |
| Resultado Esperado: Se guarda el nombre del proyecto en la base de datos y se muestra el mensaje de confirmación: "Proyecto insertado correctamente". | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Ver proyecto

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Ver proyecto. En esta historia de usuario se muestran los nombres de los proyectos que existen en la base de

datos.

Tabla 50: CP HU Ver proyecto

| CASO DE PRUEBA DE ACEPTACIÓN | |
|---|--|
| Número de caso de prueba: 4. | Número de Historia de Usuario: 4. |
| Descripción de la Prueba: El Planificador selecciona la opción "Ver proyectos", se muestran los nombres de los proyectos existentes en la base de datos, permite la opción de eliminar los proyectos que se muestran (ver: Caso de Prueba Historia de Usuario: Eliminar proyecto). | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador. | |
| Entrada / Pasos de ejecución: Se muestran los nombres de proyectos existentes en la base de datos. | |
| Resultado Esperado: Se muestran los nombres de proyectos existentes en la base de datos. | |
| Evaluación de la Prueba: Prueba satisfactoria. | |

Caso de Prueba Historia de Usuario: Eliminar proyecto

Esta sección cubre el conjunto de pruebas funcionales relacionadas con la historia de usuario: Eliminar proyecto. En esta historia de usuario se elimina el nombre de la base de datos del proyecto que se ha seleccionado.

Tabla 51: CP HU Eliminar proyecto

| CASO DE PRUEBA DE ACEPTACIÓN | |
|--|--|
| Número de caso de prueba: 5. | Número de Historia de Usuario: 5. |
| Descripción de la Prueba: El Planificador selecciona la opción "Ver proyectos" y se muestran todos los nombre de los proyectos existentes en la base de datos, el Planificador selecciona la opción "Eliminar proyecto", se muestra una alerta donde se especifica que la acción no se puede deshacer y se pide la confirmación de la acción. | |
| Condiciones de Ejecución: Que la sesión activa tenga permisos de Planificador y que se hayan insertado proyectos con anterioridad en la base de datos. | |
| Entrada / Pasos de ejecución: Se elimina el nombre del proyecto seleccionado de la base de datos, además se eliminarán los productos asociados al proyecto, se eliminarán también | |

las solicitudes de prueba, las pruebas planificadas y las no conformidades asociadas al proyecto eliminado.

Resultado Esperado: Se elimina el nombre del proyecto y toda la información asociada a este que exista en la base de datos y se muestra el mensaje de confirmación: "El proyecto ha sido eliminado".

Evaluación de la Prueba: Prueba satisfactoria.

Anexo 2: Interfaz de usuario



Insertar tipo de prueba

Nombre de la prueba: *

Documentación

Escriba el nombre del tipo de prueba

Insertar

Ilustración 13: IU Insertar tipo de prueba



Ver tipos de prueba

▼ Tipo de prueba

| # | Nombre del tipo de prueba | |
|---|---------------------------|----------|
| 1 | documentacion | Eliminar |
| 2 | doc | Eliminar |

Ilustración 14: IU Ver tipos de prueba



Ilustración 15: IU Consultar no conformidades

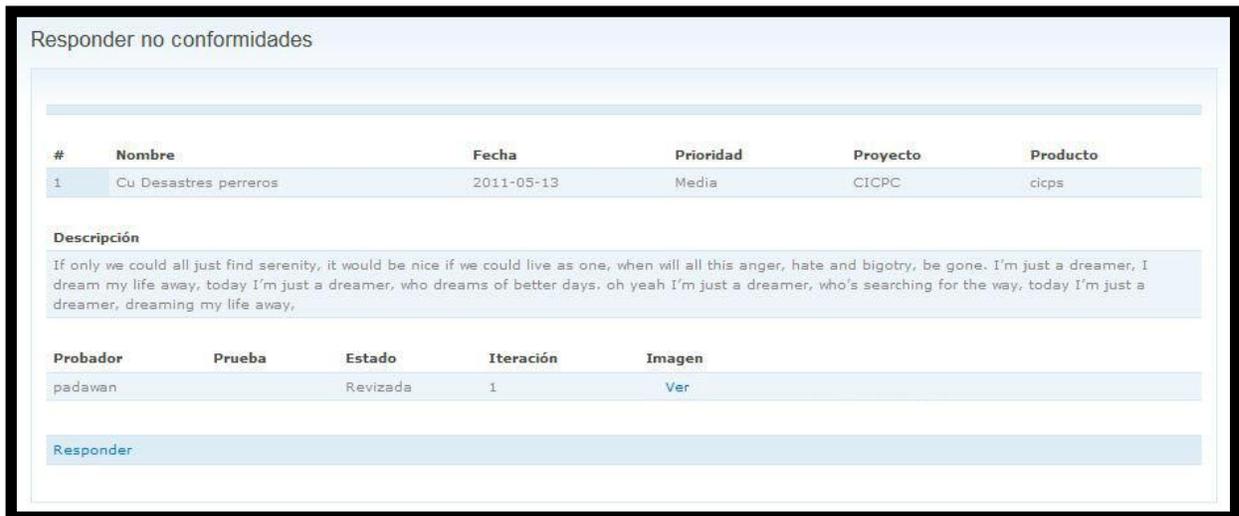


Ilustración 16: IU Responder no conformidades

Reportar no conformidad

Nombre de la prueba: *
 Pruebas a producto FM
Seleccione el nombre de la prueba que se está ejecutando.

Título: *
 Falta de concordancia
Introduzca el título de la no conformidad.

Prioridad: *
 Alta
Escoja la prioridad de la no conformidad.

Iteración: *
 1
Seleccione la iteración de la prueba.

Descripción: *
 Introduzca la descripción de la no conformidad, recuerde: debe ser en un lenguaje natural y error encontrado.
 Introduzca la descripción de la no conformidad, recuerde: debe ser en un lenguaje natural y dar los mayores detalles posibles d

Imagen:
 Q:\Pics_2\3.jpg
Introduzca una imagen que valide la no conformidad detectada.

Ilustración 17: IU Reportar no conformidad

Ver solicitudes de prueba

▼ Solicitudes de prueba existentes

| # | Nombre | Fecha | Proyecto | Producto | Tipo de prueba |
|---|----------------------------|------------|----------|----------|----------------|
| 1 | Pruba de liberacion secada | 2011-05-14 | CICPC | cicps | documentacion |

Objetivo
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam tincidunt imperdiet mi, sed hendrerit est accumsan quis. Morbi quis magna urna, vel gravida ipsum.

Ilustración 18: IU Ver solicitudes de prueba

Solicitud de prueba

Nombre:

Escriba el nombre de la prueba

Tipo de Prueba:

Seleccione el tipo de prueba que se realizará

Proyecto:

Seleccione el nombre del proyecto

Producto:

Seleccione el producto al que se le realizará la prueba

Fecha: *

Formato: 13-05-11
Escoja la fecha en que se realizará la prueba.

Objetivo:

Introduzca el objetivo de la prueba

Ilustración 19: IU Insertar solicitud de prueba

Planificar prueba

Seleccione la prueba:
Prueba de liberacion secada ▾
Seleccione el tipo prueba que se realizará

Profesor responsable del proyecto en prueba:
 emramirez
 emrys
 padawan
Seleccione el profesor responsable de la prueba por el proyecto que será probado.

Profesor Responsable de Calidad:
manolo ▾
Seleccione el profesor responsable.

Probadores:
 yamila
 emramirez
 padawan
Escoja los probadores que trabajarán el la prueba.

Fecha de inicio: *
13-05-11
Formato: 13-05-11
Escoja la fecha de inicio de la prueba.

Fecha de fin: *
28-05-11
Formato: 13-05-11
Escoja la fecha de fin de la prueba.

Archivos necesarios:
Q:\Expediente.rar
Suba los archivos necesarios para realizar la prueba deben estar comprimidos

Observaciones:
orem ipsum dolor sit amet, consectetur adipiscing elit. Etiam tincidunt imperdiet mi, magna urna, vel gravida ipsum.

Información de interés

Ilustración 20: IU Planificar prueba

