

**Universidad de las Ciencias Informáticas**

**Facultad 6**



**Título:** Portal de la Comunidad Técnica Cubana de PostgreSQL

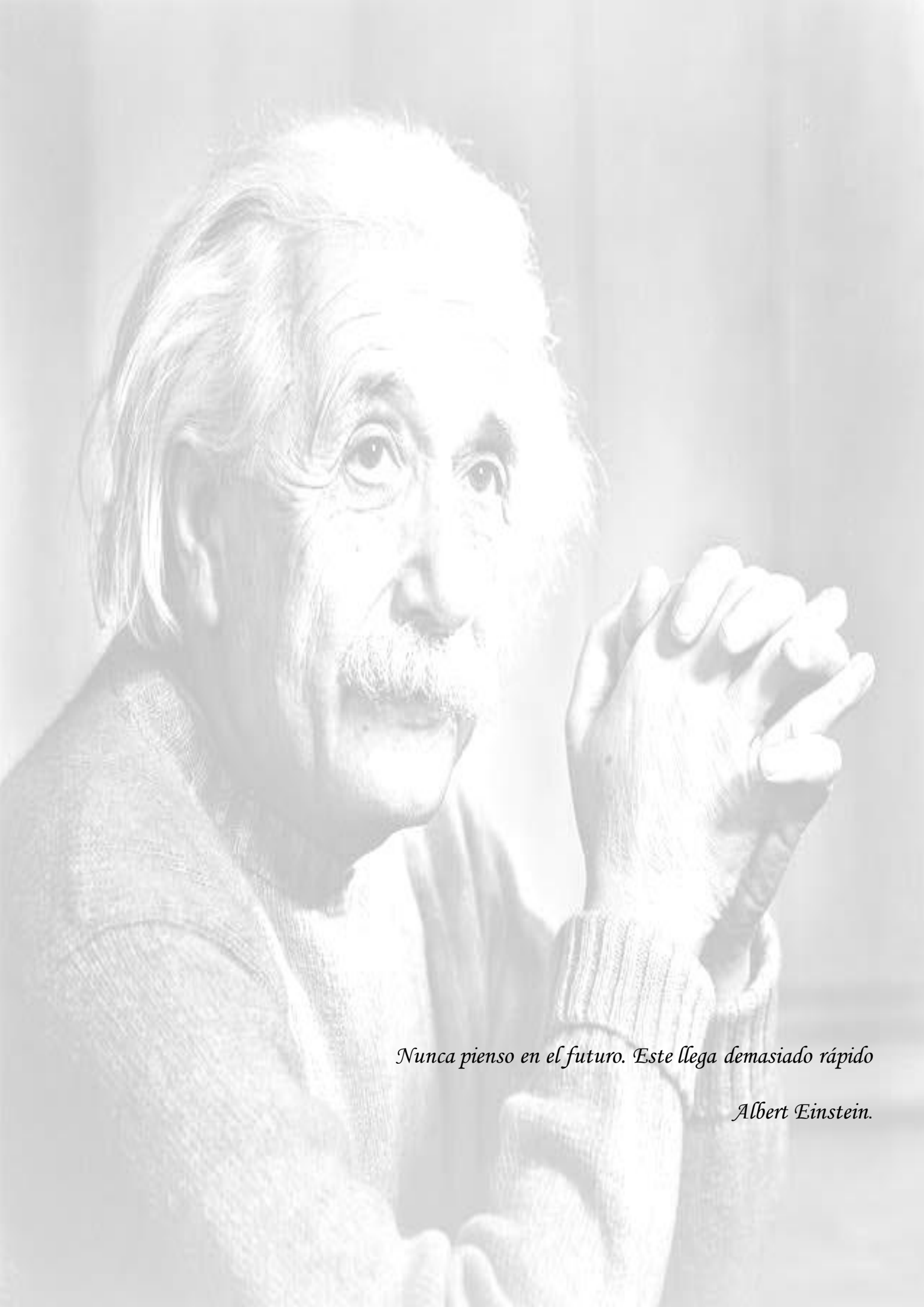
V1.0. Implementación y Prueba

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

**Autor:** Pedro Valentin Paneque Silva

**Tutor:** Ing. Yunior Mesa Reyes

**Ciudad de la Habana Junio 2011**



*Nunca pienso en el futuro. Este llega demasiado rápido*

*Albert Einstein.*

**DECLARACIÓN DE AUTORÍA**

Declaro ser el 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 \_\_\_\_\_.

Pedro Valentin Paneque Silva

Ing. Yunior Mesa Reyes

\_\_\_\_\_  
Firma del Autor

\_\_\_\_\_  
Firma del Tutor

## DATOS DE CONTACTO

**Tutor:** Ing. Yunior Mesa Reyes

**Email:** [ymreyes@uci.cu](mailto:ymreyes@uci.cu)

### **Experiencia profesional:**

- Ingeniero en Ciencias Informáticas, tercera graduación de la Universidad de las Ciencias Informáticas.
- Analista y Líder del Grupo de Trabajo "Comunidad" del Departamento de PostgreSQL Empresarial del Centro de Tecnologías y Gestión de Datos (DATEC).
- Webmaster del portal de la Comunidad Técnica Cubana de PostgreSQL.

### *Agradecimientos*

A Ileana V. Silva y a Pedro V. Paneque, los mejores padres del mundo, por su dedicación, cariño, respeto y apoyo, por hacer suyas cada una de las dificultades que encontré a lo largo de estos cinco años y su empeño en que este sueño se hiciera realidad, por siempre darme todo sin esperar nada a cambio, por ser mis ídolos, no tengo palabras para expresar mi agradecimiento por ustedes.

A mi pequeña hermana, única en su clase, siempre viva, inteligente y alegre, por estar cuando fue necesitada, por ser mi confidente, mi amiga, por esa sonrisa que siempre mostró, capaz de transformar los grandes problemas en detalles insignificantes.

A mis abuelos, siempre tan buenos, comprensivos y preocupados, que tanto me consintieron, de tan excelentes consejos y guías sin igual.

A Ruber Sánchez por su apoyo, su desinterés, sus consejos, su cariño, por ser como un padre más, por todas las cosas buenas que hizo por mí, así como por enseñarme otros caminos, los cuales pienso seguir algún día.

A Yamilka por todo el apoyo, amistad incondicional, y entre un millón de cosas por ser una madre más.

A Yanisel y a El FI por su amistad y sus locuras.

A todos los amigos de mis padres por su lealtad, a Jaguar, Linier, Bernardo y Benny Moré, por ser tan amigos de ellos como míos, por sus consejos y los momentos que pasamos juntos.

A Mario Raúl por ser un segundo tutor durante la realización de este trabajo, ayudando siempre con sus conocimientos de todas las ramas informáticas.

A Yunior Mesa por desarrollar un buen trabajo como tutor, trabajando siempre en todo momento.

A todas las personas y amigos que de una forma u otra han estado siempre conmigo, ayudando en todo lo que han podido o brindando algún otro tipo de apoyo.

Como parte final quiero agradecer a Ena Galindo, mi compañera, esa niña tan importante para mí, que cambió mi vida y que ha sabido estar conmigo en todo momento dando lo mejor de sí, soportando todos mis malos ratos, y brindando lo mejor de ella, su cariño.

A todas estas personas que están aquí:

Muchas Gracias.

### *Dedicatoria*

Quiero dedicar este trabajo de diploma a mi abuelo, ese segundo padre que siempre me dio lo mejor de sí, y que siempre quiso que esto se hiciera realidad, yo sé que donde quiera que esté, puede estar satisfecho.

### **RESUMEN**

Las aplicaciones informáticas, sean de tipo web o desktop, en su mayoría poseen como soporte un sistema de gestión de base de datos. En nuestro país se cuenta con una amplia gama de gestores tanto libres como privativos en los sistemas informáticos de las organizaciones e instituciones del mismo.

Un grupo de especialistas y operadores del gestor postgresQL crearon la Comunidad Técnica Cubana de PostgreSQL con los objetivos de contribuir al desarrollo de las tecnologías de bases de datos tomando como base el gestor postgresQL con desarrollos que aporten a la comunidad internacional, proveer soluciones integrales y consultorías relacionadas con la migración y la explotación de postgresQL, y al mismo tiempo lograr la formación de especialistas de alto nivel apoyando el desarrollo tecnológico cubano. La misma cuenta con un portal web para el intercambio de información entre los especialistas y seguidores del gestor, siendo este una fuente importante de unificación para dichos miembros. El portal presenta dificultades para su modificación lo que impide que se brinden los servicios deseados por los integrantes de la comunidad impidiendo así el logro de los objetivos para lo cual fue diseñada.

La presente investigación pretende darle solución al problema planteado mediante el desarrollo de un portal web que integre a todos los seguidores, especialistas e interesados en el uso y aprendizaje de postgresQL como gestor de base de datos; sustituyendo la aplicación existente para este fin, facilitando el intercambio de ideas y de información entre los miembros de la Comunidad.

### **Palabras Claves**

Comunidad Técnica Cubana de PostgreSQL, portal, postgresQL.

---

INTRODUCCIÓN .....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	4
1.1    Conceptos asociados al problema .....	4
1.2    Portal web de la Comunidad Técnica Cubana de PostgreSQL .....	4
1.3    Metodologías, tecnologías y herramientas basadas en la implementación .....	5
1.3.1    Metodología de Desarrollo.....	5
1.3.2    Sistema de Gestión de Contenido .....	6
1.3.3    Lenguajes de Programación.....	9
1.3.4    Herramienta de Desarrollo (IDE).....	12
1.3.5    Gestor de Base de Datos .....	14
1.3.6    Servidor Web .....	16
1.4    Tecnologías y herramientas para las pruebas de software. ....	17
1.4.1    Pruebas de Caja Blanca:.....	18
1.4.2    Pruebas de Caja Negra .....	18
1.4.3    Casos de Prueba .....	19
1.4.4    Pruebas de aceptación en XP .....	19
1.4.5    Pruebas Unitarias en XP .....	21
1.4.6    Pruebas de Carga y Estrés .....	21
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA .....	25
2.1    Propuesta del Sistema.....	25
2.2    Roles de Usuarios .....	26
2.3    Seguridad en la aplicación.....	26
2.3.1    Pautas a seguir para diferentes tipos de ataques: .....	27
2.3.2    Módulos adicionales para la seguridad .....	28
2.4    Historias de Usuario .....	28
2.5    Requisitos no Funcionales.....	31
2.6    Prototipo de interfaz de usuario .....	32
CAPÍTULO 3: IMPLEMENTACION.....	35
3.1    Plan de Release.....	35
3.2    Diseño de la base de datos.....	36
3.3    Estándar de código .....	37
3.4    Tarjetas CRC .....	38
3.5    Patrones de Diseño en Drupal .....	40
CAPÍTULO 4: PRUEBAS Y VALIDACION DE LA SOLUCION PROPUESTA .....	42
4.1    Prueba de caja blanca .....	42
4.1.1    Prueba del Camino básico.....	42
4.2    Prueba de caja negra.....	45
4.3    Pruebas de Carga y Estrés .....	48
CONCLUSIONES GENERALES .....	50
RECOMENDACIONES .....	51



REFERENCIAS BIBLIOGRAFICAS .....	52
ANEXOS .....	54
Anexo 1 Historias de usuario .....	54
Anexo 2 Casos de Uso de Pruebas.....	63
Anexo 3 Resultados arrojados por JMeter .....	71

## INTRODUCCIÓN

Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En la actualidad, y debido al desarrollo tecnológico en los campos de la informática y la electrónica, la mayoría de las bases de datos están en formato digital, ofreciendo un amplio rango de soluciones al problema de almacenar datos.

Los programas que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada son denominados Sistemas Gestores de Bases de Datos (SGBD).

PostgreSQL está considerado ampliamente como el sistema gestor de base de datos de código abierto más avanzado del mundo, clasificado así por diversos especialistas del tema en el sitio oficial del mismo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre; pero es su condición de software libre quien lo hace uno de los gestores más importantes para nuestro país, debido a nuestra condición de subdesarrollo así como el bloqueo económico al cual estamos sometidos; sustentado por la idea de que la utilización de otro de estos gestores trae consigo problemas monetarios y de permisos con sus patentes, debido a la restricción que tienen muchas de las empresas que dirigen estos gestores para comercializar con Cuba. Esto constituye un impedimento a la hora de exportar aplicaciones basadas en algún gestor de base datos propietarios a las cuales haya que brindarles mantenimiento y soporte posteriormente.

Debido al actual crecimiento de las Tecnologías de la Informática y las Comunicaciones, el flujo de información se hace cada día mayor y de una forma más eficiente utilizando diversos recursos para ello entre los que se destacan los portales web, que facilitan grandes cúmulos de información de diversos temas. En el mundo existen portales que permiten la comunicación entre usuarios interesados tanto por uno u otro gestor, posibilitando un aumento de conocimientos acerca del tema. En múltiples países se pueden encontrar este tipo de portales web que brindan información, datos de interés y eventos a desarrollarse de acuerdo al gestor de base de datos que representan.

En Cuba hasta este momento existía un portal web para mantener actualizado a los seguidores de PostgreSQL, en sus inicios fungió como puente unificador entre los miembros de la comunidad dispersos por todo el país, y como punto de partida para lograr acciones de conjunto en el desarrollo del gestor. Dicho portal presenta una estructura que no es la más adecuada para lograr los objetivos que se persiguen con este tipo de portales web; inicialmente fue desarrollado usando Redmine, que es

un sistema de gestión de proyectos; como su nombre lo indica es para gestionar proyectos, no para la gestión de portales web, además de ser poco flexible a cambios, por lo que se hace complejo llevar a cabo el aumento de servicios y prestaciones a los miembros de la Comunidad. La información que se muestra en la aplicación no es suficiente para satisfacer las necesidades que requieren los miembros de la comunidad, ejemplo: el intercambio entre los usuarios, el aumento de la información del tema tratado y un mayor nivel de servicios ya que las prestaciones que se ofrecen son insuficientes. Lo expuesto anteriormente da lugar a la existencia de problemas de comunicación entre los diferentes especialistas del tema, provocando un desconocimiento por parte de las demás personas acerca de lo que se realiza, perdiendo tiempo muchas veces cuando se trabaja sobre las bases de algún proyecto que ya se encuentra desarrollado.

Debido a la problemática antes planteada se identifica como **problema de la investigación**: ¿Cómo contribuir a elevar la calidad de los servicios y brindar mejores prestaciones a la Comunidad Técnica Cubana de PostgreSQL?

El **objeto de estudio** de esta investigación es el proceso de desarrollo de aplicaciones web para gestionar comunidades virtuales, enmarcándose en el **campo de acción** del proceso de implementación y pruebas de la aplicación web para la Comunidad Técnica Cubana de PostgreSQL.

Para dar solución al problema de la investigación se ha definido como **objetivo general** realizar la fase de desarrollo de la versión 1.0 del Portal de la Comunidad Técnica Cubana de PostgreSQL.

Para lograr el objetivo general se han trazado los siguientes **objetivos específicos**:

- Implementar los módulos del Portal Web.
- Aplicar las pruebas diseñadas al Portal Web.
- Evaluar los resultados obtenidos en las pruebas.

Para darle cumplimiento a los objetivos propuestos se plantean el siguiente conjunto de **tareas de la investigación**:

- Revisión del Análisis y Diseño realizado al Portal Web.
- Estudio de los lenguajes de programación web enfatizando en los relacionados con la Metodología Ágil XP.
- Estudio de las herramientas de pruebas, enfatizando en los relacionados con la Metodología Ágil XP.
- Refinamiento de los requisitos a implementar.

- Configuración de módulos de funcionalidad básica.
- Adición e implementación de módulos para otras funcionalidades.
- Diseño y aplicación de los Casos de Prueba.
- Diseño y aplicación de las pruebas de Cargas y Estrés.

## CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

### Introducción

En este capítulo se abordarán los principales temas relacionados con el desarrollo del portal web, la selección de las tecnologías y herramientas a utilizar así como los conceptos relacionados al proceso de pruebas de acuerdo a la metodología seleccionada. Se realizará una fundamentación teórica-conceptual de los principales temas que se abordarán a lo largo de la investigación.

### 1.1 Conceptos asociados al problema

- **Portal Web de Internet:** es un sitio cuya característica fundamental es la de servir de Puerta de Entrada para ofrecer al usuario, de forma fácil e integrada, el acceso a una serie de recursos y de servicios relacionados a un mismo tema. Incluye: enlaces, buscadores, foros, documentos, aplicaciones, compra electrónica, etc. Principalmente un portal en Internet está dirigido a resolver necesidades de información específica de un tema en particular.

Existen tres modalidades de portales:

1. Portales horizontales: también llamados portales masivos o de propósito general, se dirigen a una audiencia amplia, tratando de llegar a toda la gente con muchas cosas.
2. Portales verticales: se dirigen a usuarios para ofrecer contenido dentro de un tema específico.
3. Portales diagonales: se trata de una mezcla entre el portal horizontal y el vertical. Se trata de portales que utilizan redes sociales o aplicaciones generalistas.

Los portales normalmente tienen programación que requiere muchos recursos computacionales y por su alto tráfico generalmente se hospedan en servidores de Internet dedicados.

### 1.2 Portal web de la Comunidad Técnica Cubana de PostgreSQL

En Cuba también se utilizan de forma frecuente los portales web, en muchas ocasiones con el objetivo de divulgar las principales actividades que tienen lugar en la isla, así como dar a conocer verdades que intentan ocultarse por las grandes potencias mundiales. El país tiene una comunidad de usuarios y desarrolladores de PostgreSQL, los cuales tienen como objetivo principal el desarrollo de tecnologías de bases de datos tomando como base el gestor postgresQL. Estos se encargan de proveer soluciones integrales y consultorías relacionadas con la migración y la explotación de este gestor.

Con la implementación del portal de la Comunidad Técnica Cubana de PostgreSQL se pretende gestionar los proyectos y soluciones para el desarrollo del gestor como apoyo al trabajo colaborativo tanto nacional como internacional, lograr la publicación de cursos de capacitación y formación para usuarios interesados en el tema de postgresQL, poner a disposición herramientas de apoyo al uso y explotación del gestor, así como descripciones de los eventos realizados en el país relacionados con el tema, brindando la posibilidad de inscripción para los miembros y acceso a los documentos generados en estos eventos. Por otra parte contar con funcionalidades como gestión de encuestas, video tutoriales, galería de imágenes, servicio de foro, y reportes estadísticos proporcionarían al portal y a sus miembros un mayor auge profesional e informativo.

### 1.3 Metodologías, tecnologías y herramientas basadas en la implementación

#### 1.3.1 Metodología de Desarrollo

En una tesis previa de Planificación y Diseño del portal web de la Comunidad Cubana de PostgreSQL, se definió como metodología de desarrollo XP para dicho portal web. Es una metodología ágil utilizada en proyectos de corto plazo. Se basa en pruebas unitarias, que se realizan a los principales procesos y a las posibles fallas que pudieran ocurrir, también en la reutilización de código para lo cual se crean patrones o modelos estándares. Algunas de las particularidades más significativas de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto determinado en una misma estación de trabajo. El cliente forma parte del equipo de desarrollo, esto facilita grandemente la comunicación entre todos los miembros del equipo de trabajo, permitiendo que el mismo esté en todo momento presente y de acuerdo con todos los cambios por los que pase el sistema en desarrollo. (1)

XP es la metodología seleccionada para el desarrollo del proyecto. A continuación, se exponen algunas de las razones tomadas en cuenta al escoger esta metodología:

- **El proyecto es pequeño.** XP está concebida para ser utilizada dentro de proyectos pequeños y de desarrollo rápido.
- **Empieza en pequeño y añade funcionalidad con retroalimentación continua.** El desarrollo del sistema comienza a partir de los requerimientos básicos y a partir de ahí se van añadiendo funcionalidades que tanto el desarrollador como el cliente entiendan necesarias.
- **Pocos roles.** Esta metodología está dirigida a grupos de desarrollo pequeños y con pocos roles.

- **El manejo del cambio se convierte en parte sustantiva del proceso.** A medida que el proyecto avanza pueden surgir nuevas expectativas o ideas que pueden ser incorporadas fácilmente permitiéndole mayor adaptabilidad al producto.
- **El cliente o el usuario se convierte en miembro del equipo.** Con el uso de esta metodología y la importancia que esta le concede a la retroalimentación, el cliente es parte del equipo de desarrollo.
- **Propiedad colectiva del código.** XP plantea que todos los programadores pueden realizar cambios en cualquier parte del código en cualquier momento.

### 1.3.2 Sistema de Gestión de Contenido

CMS son las siglas de Content Management System, que se traduce al español como Sistema de Gestión de Contenidos, que como su nombre lo indica nos permite gestionar o administrar contenidos en un medio digital, en este caso administrar los contenidos plasmados en una aplicación web. Dicho de otra forma, un CMS es una herramienta que permite a un editor crear, clasificar y publicar cualquier tipo de información en una página web, generalmente llevan por detrás una base de datos en la cual se modifican o actualizan los cambios y configuraciones en los contenidos del sitio web.

#### 1.3.2.1 Joomla

Este es uno de los más usados y conocidos en el mundo del CMS opensource. Permite la creación de páginas web de una manera muy fácil y potente. Es un sistema de administración de contenidos de código abierto construido con PHP bajo una licencia pública general (en inglés: General Public License o GPL). Este administrador de contenidos se usa para publicar en Internet e Intranets utilizando una base de datos MySQL.

En Joomla se incluyen características como: hacer caché de páginas para mejorar el rendimiento, indexamiento web, feed o alimentación RSS, versiones imprimibles de páginas, flash con noticias, blogs, foros y chats, motores de comercio y venta electrónica, encuestas, calendarios, búsqueda en el sitio web, galerías de imágenes y multimedia, calendarios, servicios de suscripción e internacionalización del lenguaje.

Una de las ventajas por las cuales se destaca es que con este sólo preocupa la información que se desea publicar ya que este CMS gestionará todos los detalles técnicos y administrativos. Otra de las ventajas es que está preparado para organizar toda la información por secciones y categorías

facilitando la navegabilidad de los usuarios. Joomla ofrece la posibilidad de instalar, desinstalar y administrar componentes y módulos que agregarán servicios de valor a los visitantes del sitio.

No obstante, hay varios aspectos negativos relacionados a Joomla como es el caso de la usabilidad, tiene además el problema de ser muy rígido en su organización del contenido, con una estructura jerárquica en dos niveles, esto obliga a planificar muy bien qué contenido tendrá el sitio y cómo se va a estructurar, porque no es sencillo cambiarlo una vez se tenga mucho publicado. Tiene limitaciones en las opciones para personalizar rangos de usuarios, no sólo para personalizarlos, sino para crear nuevos rangos y permisos. Entre otras de sus limitantes más significativas resaltan su alta dependencia de javascript en su panel de administración y el cierto nivel de lentitud que presenta en su funcionamiento.

### **1.3.2.2 WordPress**

WordPress es un popular sistema de gestión de contenidos, más orientado a lo que son los blogs, o bitácoras en línea. Por medio de este software sus usuarios pueden crear sus propios blogs de una manera sencilla y personalizada. Su licencia GPL, su facilidad de uso, su enfoque hacia la elegancia y la estética, y todas sus atractivas características lo han convertido en una de las plataformas de publicación personal más populares del mundo.

Como sistema de gestión de contenidos, WordPress consiste en un programa que permite una estructura de soporte para que usuarios de Internet puedan crear y administrar contenidos. Un CMS controla bases de datos que permiten manejar el diseño y el contenido de una página web. Es por eso que con WordPress, los usuarios pueden cambiar, cuantas veces quieran, el diseño de su página o blog, sin necesidad de cambiar el formato. Además, permite la publicación de material de una manera sencilla para luego ser vista por otros usuarios.

Para usar WordPress el usuario sólo necesita de un correo electrónico como requisito y luego puede bajar el programa o "script", que es totalmente gratuito. La forma de operarlo está explicada por pasos muy sencillos. WordPress, también ofrece su propio hospedaje de blogs con un almacenaje de 3 GB, y este puede ser privado o público; esta última es la opción más fácil para comenzar, ya que no se necesita de servidores ni mayores conocimientos técnicos. Hay disponibles 60 plantillas, y se desarrollan más, periódicamente, con diseños y estilos diferentes, para dar el carácter que usuario desee en su blog, ya sea uno profesional, juvenil, deportivo, infantil, etc. También es muy popular su sistema de estadísticas, que permite al usuario saber en todo momento, la cantidad de personas que están visitando el blog, de dónde provienen, cuáles son los más populares, y cuáles términos de búsqueda está enviando la gente.



Con WordPress, se puede disponer de un corrector ortográfico para escribir en el blog o comentar en otras páginas. También se pueden subir fotos, insertar videos y todo de manera sencilla. Además, tiene un servicio de bloqueo de spam o correo basura que puede provenir de los comentarios de otros usuarios y también los que provienen de blogs que tratan de escabullirse; en WordPress es importante señalar que son frecuentes los ataques a estos sistemas masivos de blogs; los hackers permanentemente buscan nuevas formas de vulnerar estos sistemas. (2)

### 1.3.2.3 Drupal

Otro ejemplo de CMS muy utilizado es Drupal. Este es un sistema de gestión de contenido muy configurable. Es un programa de código abierto, con licencia GPL, escrito en PHP, desarrollado y mantenido por una activa comunidad de usuarios. Ofrece una gran consistencia de todo el sistema. Gracias a su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales de los que dispone, hace que sea adecuado para la realización de muchos sitios web. Permite publicar artículos, imágenes, otros archivos y servicios añadidos como foros, encuestas, votaciones, blogs, administración de usuarios y permisos. Es un sistema dinámico, en lugar de almacenar los contenidos en archivos estáticos en el sistema de ficheros del servidor de forma fija, el contenido textual de las páginas y otras configuraciones son almacenados en una base de datos y se editan utilizando un entorno Web incluido en el producto.

Dentro de sus características que lo hacen uno de los más utilizados en el desarrollo de aplicaciones se encuentran:

- **Ayuda on-line:** Un robusto sistema de ayuda online y páginas de ayuda para los módulos del núcleo, tanto para usuarios como para administradores.
- **Búsqueda:** Todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- **Código abierto:** El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario de otros sistemas de blogs o de gestión de contenido propietarios, es posible extender o adaptar Drupal según las necesidades.
- **Módulos:** La comunidad de Drupal ha contribuido con muchos módulos que proporcionan funcionalidades como página de categorías, autenticación mediante jabber, mensajes privados.
- **Personalización:** Un sólido entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo a las preferencias definidas por el usuario. (1)

Drupal cuenta además con innumerables ventajas:

- Configuración de Idiomas. Traducción al español casi completamente.
- Mucha información tanto en inglés como en español, gran comunidad de usuarios aportando información.
- Fácil configuración de temas y una gran cantidad de plantillas.
- Gran cantidad de opciones de configuración con una interfaz amigable para el administrador.
- Fácilmente extensible a través de módulos que permiten prototipar o adaptar rápidamente la aplicación a necesidades específicas. Más de 540 módulos para extender el sitio. (3)

Después de un detallado estudio de los principales CMS de código abierto se decide emplear Drupal en su versión 6.13 por todas las características que el mismo presenta, así como las ventajas vistas anteriormente.

### **1.3.3 Lenguajes de Programación**

Un lenguaje de programación es un idioma artificial diseñado para expresar computaciones que son interpretadas por las computadoras, diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Por lo tanto, es un modo práctico para que los seres humanos puedan dar instrucciones a una computadora. También la palabra programación se define como el proceso de creación de un programa de computadora, mediante la aplicación de procedimientos lógicos. (4)

#### **1.3.3.1 Hypertext Pre-processor (PHP)**

Es un lenguaje interpretado de alto nivel embebido en páginas HTML (Lenguaje de Mercado de Hipertexto, su significado en inglés: Hypertext Markup Language), y ejecutado en el servidor. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas. Con PHP se puede hacer cualquier cosa que se pueda realizar con un script, como el procesamiento de información en formularios, foros de discusión, manipulación de cookies y páginas dinámicas. También ofrece la integración con varias bibliotecas externas, que permiten que el desarrollador tenga mayores opciones, desde generar documentos en PDF hasta analizar código XML (Lenguaje de marcas extensible, siglas en inglés: Extensible Markup Language).

PHP ofrece una solución simple y universal para las paginaciones dinámicas de la web de fácil programación. Su diseño elegante lo hace perceptiblemente, más fácil de mantener y ponerse al día en comparación con el código de otros lenguajes. Debido a su amplia distribución está perfectamente

soportado por una gran comunidad de desarrolladores. Como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y se reparen rápidamente. El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP. Este lenguaje es completamente expandible. Está compuesto de un sistema principal, un conjunto de módulos y una variedad de extensiones de código. Posee interfaces distintas para cada tipo de servidor. Es de código abierto, el usuario no depende de una compañía específica para arreglar elementos que no funcionan y no tiene que pagar actualizaciones anuales para tener una versión que funcione.

En cuanto al tema de la seguridad PHP es un poderoso lenguaje e intérprete, es capaz de acceder a archivos, ejecutar comandos y abrir conexiones de red en el servidor. PHP está diseñado específicamente para ser un lenguaje más seguro para escribir programas, y con la selección correcta de opciones de configuración en tiempos de compilación y ejecución, y siguiendo algunas prácticas correctas de programación.

PHP cuenta con un gran número de ventajas que lo hacen ser uno de los lenguajes de programación más utilizados hoy en día para el desarrollo de páginas web, entre estas características se encuentran:

- Muy fácil de aprender.
- Soporta en cierta medida la orientación a objeto. Clases y herencia.
- Es un lenguaje multiplataforma: se puede ejecutar en Linux, Windows, entre otros.
- Capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras.
- Capacidad de expandir su potencial utilizando módulos.
- Posee documentación en su página oficial la cual incluye descripción y ejemplos de cada una de sus funciones.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Incluye gran cantidad de funciones.
- No requiere definición de tipos de variables ni manejo detallado del bajo nivel (5)

### 1.3.3.2 Hyper Text Markup Language (HTML)

Es un lenguaje de programación muy sencillo que se utiliza para crear los textos y las páginas web. Si se hace la traducción de su nombre del inglés al español, sería “Lenguaje de Marca de Hipertextos”, ya que es justamente un lenguaje que se basa en las marcas para crear los hipertextos.

Esta definición se debe a que está compuesto por etiquetas que definen la estructura y el formato del documento que verá el usuario a través de la web. Esas etiquetas son leídas por el navegador o visualizador, es decir el programa que se utiliza para navegar, y que es el que ejecuta las funciones creadas en HTML permitiendo que puedan ser visibles en nuestra máquina.

Por otra parte, el HTML permite incluir scripts, códigos que pueden modificar el comportamiento de los navegadores web y de otros procesadores de HTML.

Dentro de las ventajas que presenta se pueden mencionar:

- Texto presentado de forma estructurada y agradable.
- No necesita de grandes conocimientos cuando se cuenta con un editor de páginas web.
- Archivos pequeños.
- Despliegue rápido.
- Lenguaje de fácil aprendizaje.
- Es interpretado por todos los exploradores o navegadores web.

### 1.3.3.3 Java

Es un lenguaje de programación sencillo, orientado a objetos, de propósito general e independiente de la plataforma de desarrollo, permite crear programas que funcionan en la mayoría de los ordenadores y sistemas operativos. Se usa Java para crear programas especiales denominados applets, que pueden ser incorporados en páginas web para hacerlas interactivas.

Dentro de sus características más conocidas se relacionan las siguientes:

- **Es robusto:** Realiza verificaciones en busca de problemas tanto en tiempo de compilación como en tiempo de ejecución. La comprobación de tipos ayuda a detectar errores, lo antes posible, en el ciclo de desarrollo.
- **Es de arquitectura neutral:** Para establecer Java como parte integral de la red, el compilador compila su código a un fichero objeto de formato independiente de la arquitectura de la máquina en que se ejecutará.
- **Es dinámico:** Se beneficia todo lo posible de la tecnología orientada a objetos. No intenta conectar todos los módulos que comprenden una aplicación hasta el tiempo de ejecución.

- **Es seguro:** El sistema de Java tiene ciertas políticas que evitan que se puedan codificar virus con este lenguaje.
- **Multithreaded:** Un lenguaje que soporta múltiples threads es un lenguaje que puede ejecutar diferentes líneas de código al mismo tiempo. (6)

Este lenguaje de programación posee un conjunto de ventajas que durante la investigación realizada sobresalen en la mayor parte de las bibliografías consultadas, entre las que se encuentran:

- Es una fuente abierta por lo que los usuarios no tienen que luchar con los impuestos sobre patente cada año.
- Independiente de la plataforma.
- Java realiza la colección de basura, así que la gerencia de memoria es automática.
- Java asigna siempre objetos en el apilado.
- Java abrazó el concepto de especificaciones de la excepción.
- Usando JAVA podemos desarrollar aplicaciones web dinámicas.
- Permite que usted cree programas modulares y códigos reutilizables.

Por las ventajas que posee y las facilidades que proporciona relacionadas con el desarrollo y rendimiento para sistemas con características específicas al de este trabajo, en adición a que los módulos que sustentan el CMS escogido, Drupal, se implementan usando este lenguaje, se decide emplear PHP en su versión 5.3, como lenguaje de programación para el desarrollo de los módulos del portal web, así como también se escoge HTML para la disposición de los elementos dentro de las páginas de contenido.

### 1.3.4 Herramienta de Desarrollo (IDE)

#### 1.3.4.1 PhpDesigner

Es un completo entorno de desarrollo especialmente diseñado para la programación php como su nombre lo indica, también es una completa herramienta de HTML, CSS y editor de Javascript potenciado con características ejemplares. Para los principiantes y desarrolladores profesionales PhpDesigner le ayuda con todo, desde editar, analizar, y hasta depurar a los sitios web de publicación alimentados por PHP, HTML y CSS.

PhpDesigner posee características que desde el punto de vista de múltiples desarrolladores constituyen ventajas en su uso.

- Resaltado de sintaxis sin tenue y débil.
- Autocompletado de código para PHP, HTML, CSS y Javascript y jQuery, incluidos otros marcos.
- Inspector de código HTML para el modo de navegación intuitivo a través del código incluyendo apoyo a los objetos anidados PHP.
- Apoyo avanzado para trabajar con clases PHP.
- Explorador del código de PHP (con filtro), CSS y Javascript.
- Fácil de aprender la interfaz.

### 1.3.4.2 Zend Studio

Es un completo entorno de desarrollo integrado para el lenguaje de programación PHP. Está escrito en Java, y está disponible para las plataformas Microsoft Windows, Mac OS X y GNU/Linux.

#### Características

- No requiere la instalación previa de PHP ni del entorno de ejecución de Java.
- Soporte para PHP 4 y PHP 5.
- Resaltado de sintaxis, autocompletado de código, ayuda de código y lista de parámetros de funciones y métodos de clase.
- Plegado de código (comentarios, bloques de phpDoc, cuerpo de funciones y métodos e implementación de clases).
- Inserción automática de paréntesis y corchetes de cierre.
- Sangrado automático y otras ayudas de formato de código.
- Emparejamiento (matching) de paréntesis y corchetes (si se sitúa el cursor sobre un paréntesis (corchete) de apertura (cierre), Zend Studio localiza el correspondiente paréntesis (corchete) de cierre (apertura)).
- Detección de errores de sintaxis en tiempo real.
- Funciones de depuración: Botón de ejecución y traza, marcadores, puntos de parada (breakpoints), seguimiento de variables y mensajes de error del intérprete de PHP. Permite también la depuración en servidores remotos (requiere Zend Platform).
- Instalación de barras de herramientas para Internet Explorer y Mozilla Firefox (opcional).
- Soporte para gestión de grandes proyectos de desarrollo.
- Manual de PHP integrado.
- Soporte para control de versiones usando CVS o Subversion (a elección del desarrollador).
- Cliente FTP integrado.
- Soporte para navegación en bases de datos y ejecución de consultas SQL. (18)

Zend Studio fue diseñado para usarse con el lenguaje PHP; sin embargo ofrece soporte básico para otros lenguajes Web, como HTML, JavaScript y XML. Por las características descritas anteriormente se decide entonces como entorno de desarrollo seleccionar Zend Studio.

### 1.3.5 Gestor de Base de Datos

#### 1.3.5.1 MySQL

Es un sistema de gestión de base de datos relacional, multihilo y multiusuario. Este gestor de bases de datos es, probablemente, uno de los gestores mas usados, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración.

#### **Características:**

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
- Gran portabilidad entre sistemas.
- Tablas hash en memoria temporales.
- Completo soporte para operadores y funciones en cláusulas select y where. 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.
- Soporta gran cantidad de datos.
- MySQL Server tiene bases de datos de hasta 50 millones de registros.

#### 1.3.5.2 PostgreSQL

PostgreSQL es un avanzado sistema de bases de datos relacionales basado en Open Source. Esto quiere decir que el código fuente del programa está disponible a cualquier persona libre de cargos directos, permitiendo a cualquiera colaborar con el desarrollo del proyecto o modificar el sistema para ajustarlo a sus necesidades. Un sistema de base de datos relacionales es un sistema que permite la manipulación de acuerdo con las reglas del algebra relacional. Los datos se almacenan en tablas de columnas y renglones. Con el uso de llaves, esas tablas se pueden relacionar unas con otras.

#### **Características:**

- Está bajo licencia BSD (Berkeley Software Distribution). Esta licencia tiene menos restricciones en comparación con otras como la GPL estando muy cercana al dominio público. La licencia BSD al contrario que la GPL permite el uso del código fuente en software no libre.
- Atomicidad (Indivisible) es la propiedad que asegura que la operación se ha realizado o no, y por lo tanto ante un fallo del sistema no puede quedar a medias.
- Consistencia es la propiedad que asegura que sólo se empieza aquello que se puede acabar. Por lo tanto se ejecutan aquellas operaciones que no van a romper la reglas y directrices de integridad de la base de datos.
- Aislamiento es la propiedad que asegura que una operación no puede afectar a otras. Esto asegura que dos transacciones sobre la misma información nunca generarán ningún tipo de error.
- Durabilidad es la propiedad que asegura que una vez realizada la operación, ésta persistirá y no se podrá deshacer aunque falle el sistema.
- Corre en casi todos los principales sistemas operativos: Linux, Unix, BSDs, Mac OS, Beos, Windows, etc.
- Documentación muy bien organizada, pública y libre, con comentarios de los propios usuarios.
- Comunidades muy activas, varias comunidades en castellano.
- Soporte nativo para los lenguajes más populares del medio: PHP, C, C++, Perl, Python, etc.
- Soporte de todas las características de una base de datos profesional (triggers, store procedures, funciones, secuencias, relaciones, reglas, tipos de datos definidos por usuarios, vistas, vistas materializadas, etc.)
- Altamente adaptable a las necesidades del cliente

Como sistema gestor de base de datos se empleará PostgreSQL. Es el sistema de gestión de bases de datos de código abierto más avanzado del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales. Este gestor utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando.

Sus características técnicas lo hacen uno de los gestores de base de datos más potentes y robustos. PostgreSQL funciona muy bien con grandes cantidades de datos y una alta concurrencia de usuarios accediendo a la vez al sistema. Por todas estas características y por ser el principal objetivo de la Comunidad es que se selecciona el mismo para el desarrollo de la base de datos del portal web.



### 1.3.6 Servidor Web

#### 1.3.6.1 Apache

Su configuración, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. La licencia Apache es una descendiente de la licencias BSD, no es GPL. Esta licencia te permite hacer lo que quieras con el código fuente siempre que les reconozcas su trabajo. Este está continuamente actualizado y evoluciona a mayor velocidad. Además, puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona. También incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para la solución de los mismos.

#### Características:

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Apache es una tecnología gratuita de código fuente abierto. El hecho de ser gratuita es importante pero no tanto como que se trate de código fuente abierto. Esto le da una transparencia a este software de manera que si queremos ver que es lo que estamos instalando como servidor, lo podemos saber, sin ningún secreto.
- Apache es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor Web Apache. Actualmente existen muchos módulos para Apache que son adaptables a este, y están ahí para que los instalemos cuando los necesitemos. Otra cosa importante es que cualquiera que posea una experiencia decente en la programación de C o Perl puede escribir un modulo para realizar una función determinada.
- Apache trabaja con gran cantidad de Perl, PHP y otros lenguajes de script. Perl destaca en el mundo del script.
- Apache te permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.
- Tiene una alta configurabilidad en la creación y gestión de logs. Apache permite la creación de ficheros de logs a medida del administrador, de este modo puedes tener un mayor control sobre lo que sucede en tu servidor. (8)

#### 1.3.6.2 Lighttpd

Lighttpd es un servidor web diseñado para ser rápido, seguro, flexible, y fiel a los estándares. Está optimizado para entornos donde la velocidad es muy importante, y por eso consume menos CPU y

memoria RAM que otros servidores. Por todo lo que ofrece, lighttpd es apropiado para cualquier servidor que tenga problemas de carga. Lighttpd es software libre y se distribuye bajo la licencia BSD. Funciona en GNU/Linux y UNIX de forma oficial. Para Microsoft Windows actualmente hay una distribución conocida como Lighttpd For Windows mantenida por Kevin Worthington.

### **Características:**

- Virtual hosting (alojar varios dominios en la misma IP).
- Soporte para PHP, Ruby, y otros.
- Compresión (gzip, bzip2.).
- Autenticación (LDAP, htpasswd, otros).
- Consumo de memoria constante.
- Puede enviar partes de un fichero (rangos).
- También permite otros sistemas de notificación de eventos como kqueue y epoll. Muestra un listado de ficheros cuando se entra a un directorio sin index.html. Redirección condicional.
- Permite módulos externos.
- Cache Meta Language. (7)

Por las características que presenta y ventajas como que es un software libre, existe amplia disposición de Documentos e información en general para el trabajo con este software en la universidad, tiene amplia aceptación en la red, es multiplataforma, a su vez extensible, entre otras se decide emplear Apache como servidor.

### **1.4 Tecnologías y herramientas para las pruebas de software.**

Uno de los pilares de la Extreme Programming es el proceso de pruebas. XP anima a probar constantemente tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. También permite aumentar la seguridad de evitar efectos colaterales no deseados a la hora de realizar modificaciones y refactorizaciones. XP divide las pruebas del sistema en dos grupos: pruebas unitarias, encargadas de verificar el código y diseñada por los programadores, y pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñadas por el cliente final. (9)

### 1.4.1 Pruebas de Caja Blanca:

Se basan en el minucioso examen de los detalles procedimentales. Se comprueban los caminos lógicos del software proponiendo casos de prueba que examinen que están correctas todas las condiciones y/o bucles para determinar si el estado real coincide con el esperado o afirmado. Esto genera gran cantidad de caminos posibles por lo que hay que dedicar esfuerzos a la determinación de las condiciones de prueba que se van a verificar. En estas pruebas estamos siempre observando el código, que las pruebas se dedican a ejecutar con ánimo de "probarlo todo".

Cuando se implementa un software se pretende garantizar la obtención de un producto con calidad, para lo que resulta recomendable comprobar que el código que se ha escrito funciona correctamente. Con este propósito se implementan pruebas que verifican que el programa genera los resultados que realmente esperamos. Las pruebas realizadas al código son las llamadas pruebas de caja blanca, y se basan en un examen minucioso de los detalles procedimentales, logrando examinar el estado del programa en varios puntos para determinar si el estado real coincide con el esperado. Estas no se deben confundir con las pruebas informales que realiza el programador mientras está desarrollando el módulo.

Las pruebas de caja blanca han tomado un lugar muy importante en el desarrollo de sistemas de cualquier tipo, estas pruebas constituyen un método mediante el cual el ingeniero del software puede obtener casos de prueba que garanticen:

- Ejercitar por lo menos una vez todos los caminos independientes de cada módulo.
- Ejercitar todas las decisiones lógicas en sus vertientes verdadera y falsa.
- Comprobar los ciclos en sus límites y con sus límites operacionales.
- Ejercitar las estructuras de datos internas para asegurar su validez. (10)

### 1.4.2 Pruebas de Caja Negra

Las pruebas de caja negra se centran en los requisitos funcionales, permitiendo al ingeniero del software derivar conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. Estas pruebas se llevan a cabo sobre la interfaz del software, y es completamente indiferente al comportamiento interno y la estructura del programa. Los casos de prueba de caja negra pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada, que se produce una salida correcta, y que se mantiene la integridad de la información externa.

Estas pruebas tienen también como meta determinar la eficiencia del programa desde el desempeño en el equipo, el tiempo de retardo de las salidas hasta el nivel de recuperación del sistema luego de fallas o caídas, sean estas producidas por manejo incorrecto de datos, equipo, o producidas externamente.

Los casos de prueba de caja negra pretende demostrar que:

- Las funciones del software son operativas.
- La entrada se acepta de forma adecuada.
- Se produce una salida correcta, y
- La integridad de la información externa se mantiene. (10)

### **1.4.3 Casos de Prueba**

Un caso de prueba muestra una forma de probar el sistema, incluyendo la entrada o resultado con la que se ha de probar y las condiciones bajo las que ha de probarse. Con el propósito de comprobar que todos los requisitos de una aplicación son revisados, debe haber al menos un caso de prueba para cada uno de ellos.

Los casos de prueba según la metodología XP deben cumplir las siguientes características:

- Los casos de prueba deben escribirse para realizar el testing desde el punto de vista del usuario.
- Brindar una retroalimentación rápida y concreta de cómo se está desarrollando la iteración y el proyecto.
- Los casos de prueba exitosos de una iteración deben repetirse con éxito en las siguientes iteraciones.
- Un error aunque sea en un sólo paso de un caso hace que se considere que falló el caso entero. (11)

### **1.4.4 Pruebas de aceptación en XP**

Las pruebas de aceptación son parte integral del creciente desarrollo experimentado por XP. Todas las historias de usuario están apoyadas por pruebas de aceptación que son definidas por el propio cliente. Las pruebas de aceptación son más importantes que las pruebas unitarias dado que significan la satisfacción del cliente con el producto desarrollado y el final de una iteración y el comienzo de la

siguiente. Estas pruebas llevan al cliente a precisar lo que la aplicación debe hacer en determinadas circunstancias por esto, el cliente es la persona adecuada para diseñar las pruebas de aceptación.

Sin embargo esto supone el grave problema de que el cliente no tiene que tener, y en general no tiene, la formación adecuada para desarrollar buenas pruebas de aceptación. Por ejemplo, el cliente, en la mayoría de los casos sabe que es lo que quiere que la aplicación haga correctamente, pero puede no ser capaz de desarrollar un conjunto de pruebas que garantice la total cobertura de la funcionalidad especificada en la historia de uso, limitándose a probar que el sistema hace lo que debe sin verificar todas las variantes que pueden aparecer. El objetivo de las pruebas del sistema es verificar los requisitos. Por este motivo, los propios requisitos del sistema son la principal fuente de información a la hora de construir pruebas del sistema. Un desarrollo mediante programación extrema está compuesto por una serie de iteraciones cortas. Cada iteración concluye ejecutando un conjunto de pruebas de aceptación que permitan al cliente comprobar si está satisfecho con el resultado.

Una propuesta metodológica para guiar la generación de pruebas de aceptación por parte del cliente debe ser lo suficientemente sencilla para que cualquier persona sin experiencia en ingeniería del software y en pruebas pueda ponerla en práctica. Tampoco debe estar atada a ninguna notación o herramienta específica, aunque existen propuestas interesantes para esto aunque es recomendable no atarse de antemano a ninguna herramienta.

### **Características:**

- Definidas por el cliente: es el cliente quien juega el papel fundamental
- Pruebas de caja negra
- Asegurar que las funcionalidades del sistema cumplen con lo que se espera de ellas.
- Marcan el camino a seguir indicándole al equipo de desarrollo las funcionalidades más relevantes
- Permiten que el cliente sepa cuando el sistema está funcionando.
- Permiten que los programadores conozcan qué es lo que resta por hacer.
- Tienen una importancia crítica para el éxito de una iteración.
- Deben estar prontas lo antes posible a partir del comienzo de la iteración.
- El tester debe reunirse con el cliente para interpretar sus ideas y escribir los casos de prueba.
- A diferencia de las pruebas unitarias no se exige un 100% de efectividad
- Cuanto más alto es el % de efectividad exigido, más largo va a ser el tiempo estimado para la iteración.
- Incluir pruebas no críticas que si fallan se repiten a la siguiente iteración. (11)

### 1.4.5 Pruebas Unitarias en XP

Una prueba unitaria es la verificación de un módulo (unidad de código) determinado dentro de un sistema. Son llevadas a cabo por los programadores encargados de cada módulo; aseguran que un determinado módulo cumpla con un comportamiento esperado en forma aislada antes de ser integrado al sistema. Se usan cuando la interfaz de un método no es clara, la implementación es complicada y para testear entradas y condiciones inusuales.

Tienen una gran importancia pues en XP los cambios realizados son integrados en lapsos de tiempo muy breves, brinda una visión de lo que se quiere realizar, demuestra que lo implementado es lo que se pensaba al principio y es más sencillo y rápido programar teniendo los casos de prueba escritos.

Aportan beneficios como brindar al programador una inmediata retroalimentación de cómo está realizando su trabajo, además el programador puede realizar cambios de forma segura respaldado por efectivos casos de prueba y permite saber si una determinada funcionalidad se puede agregar al sistema existente sin alterar el funcionamiento actual del mismo. (11)

#### Características:

- Las pruebas unitarias se tienen que poder ejecutar sin necesidad de intervención manual. Esta característica posibilita que podamos automatizar su ejecución.
- Las pruebas unitarias tienen que poder repetirse tantas veces como uno quiera. Por este motivo, la rapidez de las pruebas tiene un factor clave. Si pasar las pruebas es un proceso lento no se pasarán de forma habitual, por lo que se perderán los beneficios que éstas nos ofrecen.
- Las pruebas unitarias deben poder cubrir casi la totalidad del código de nuestra aplicación. Una prueba unitaria será tan buena como su cobertura de código. La cobertura de código marca la cantidad de código de la aplicación que está sometido a una prueba. Por tanto, si la cobertura es baja, significará que gran parte de nuestro código está sin probar.
- Las pruebas unitarias tienen que poder ejecutarse independientemente del estado del entorno. Las pruebas tienen que pasar en cualquier ordenador del equipo de desarrollo.

### 1.4.6 Pruebas de Carga y Estrés

#### 1.4.6.1 Prueba de Carga

Este tipo de prueba se enfoca en validar y evaluar aceptabilidad de un elemento de un sistema sobre diferentes cargas de trabajo mientras el sistema permanece constante. Generalmente se incluye

simulación de cargas de trabajo promedio que puedan ocurrir dentro de la tolerancia operacional normal. (10)

### 1.4.6.2 Pruebas de Estrés

Este tipo de prueba se enfoca a evaluar el comportamiento del sistema basado condiciones anormales. Stress del sistema se refiere a extrema carga, memoria insuficiente, no disponibilidad de servicios y hardware o recursos compartidos limitados. Este tipo de prueba permite comprender mejor cómo y qué áreas del sistema colapsarán, de este modo es posible planificar contingencias y actualizar el mantenimiento y planear y asignar recursos de antemano.

### 1.4.6.3 Herramientas para pruebas de carga y estrés

- **LoadRunner**

LoadRunner es una herramienta para realizar pruebas de carga de Mercury Interactive que permite pre-visualizar el comportamiento y el rendimiento del sistema. Además, permite poner a prueba toda la infraestructura corporativa para identificar y aislar los posibles problemas mediante la simulación de la actividad de miles de usuarios. Realiza pruebas en toda la infraestructura corporativa, que comprende las soluciones e-business y las aplicaciones personalizadas, simulando la actividad de miles de usuarios, con lo que los equipos de desarrollo de aplicaciones y sitios Web pueden mejorar el rendimiento de las aplicaciones.

Es la herramienta de pruebas de carga más escalable que permite simular la actividad de miles de usuarios con los mínimos recursos de hardware. Se integra a la perfección con las herramientas de gestión del rendimiento de Mercury Interactive. Los mismos scripts creados durante las pruebas pueden volverse a utilizar para monitorizar la aplicación una vez terminada su implantación. Presenta una interfaz API abierta, con la que los usuarios y otros fabricantes pueden integrar LoadRunner en sus propios entornos. (10)

- **Quality Center**

Permite automatizar los procesos de calidad, uniendo todos los componentes con las aplicaciones correctas para acelerar los tiempos de depuración. El resultado es una mejora impresionante en la calidad y en la consistencia de la aplicación. Ayuda a manejar y controlar el riesgo, mientras se desarrolla y prueba la aplicación. En todos los momentos del proceso, se tiene visibilidad de en dónde se encuentra el proyecto con respecto a calidad (requerimientos probados y satisfechos, pruebas ejecutadas, defectos y tendencias encontradas, etc.). Maneja y automatiza el proceso de entrega, con indicadores claves de rendimiento, a tiempo real, proporciona aplicaciones que automatizan todas las

actividades claves en los procesos de calidad, y apoya a todas las personas y roles que necesitan involucrarse en los procesos de entrega. Esta herramienta no es una herramienta de pruebas de sistemas pero se ha decidido involucrarla en la investigación ya que la utilización de la misma en el manejo de la calidad del proceso de pruebas influye en la garantía de los resultados y del proceso en general, permitiendo una mayor organización del proceso e influye directamente en el rendimiento del equipo de trabajo. (10)

- **JMeter**

Es una herramienta Java desarrollada dentro del proyecto Jakarta, que permite realizar Pruebas de Rendimiento y Pruebas Funcionales sobre Aplicaciones Web. JMeter permite realizar pruebas Web clásicas así como test. No se necesita la utilización de una licencia ya que se permite bajar de manera gratuita el software desde la Web que representa al proyecto Jakarta. Se puede decir además que existe una amplia documentación sobre la utilización el modo de empleo de la herramienta y los requerimientos no funcionales que ella precisa. Así mismo es válido referir que esta herramienta fue realizada en el marco del software libre.

Dentro de las funcionalidades que brinda la herramienta se encuentran:

- Permite realizar test de FTP, JDBC, JNDI, LDAP, SOAP/XML-RPC, y WebServices (en Beta).
- Ofrece la posibilidad que el propio usuario desarrolle en Java un “Controller” a medida, cumpliendo una interfaz Java, y depositando el .jar correspondiente al desarrollo en el directorio “lib” de JMeter lo que permite reducir el consumo de rendimiento del CPU.
- Permite realizar pruebas distribuidas en distintos ordenadores que actuarán como clientes simulando varios hilos que harán función de usuarios.
- Permite generar un caso de prueba es a través de una navegación de usuario.

Tipos de informes que puede generar:

- Assertion Results: Muestra la URL de cada petición e indica los errores que se produzcan (Assertions que no se han cumplido) en el test.
- Graph Full Results: Simplemente muestra el tiempo.
- Graph Results: Muestra un gráfico con los tiempos medio, desviación, throughput, etc. de la ejecución del plan de prueba.
- Mailer Visualizar: Permite enviar un e-mail si el plan de pruebas falla o no, o supera un determinado valor de fallos o éxitos.
- Simple Data Writer: Vuelca los resultados a un fichero.



- Spline Visualizer: Gráfico de tiempos como spline.
- Aggregate Report: Muestra una tabla con una fila por cada URL solicitada, indicando el tiempo min, máx, medio, etc. Es una tabla que totaliza por URL.
- View Results in Table: Muestra una tabla con todas las respuestas, la URL, tiempo y resultado de ejecución de cada una de ellas. (10)

Se decide emplear la herramienta JMeter debido a las características mencionadas que presenta así como ventajas tales como que es la más completa y útil de las herramientas gratis, es una herramienta que sirve para realizar pruebas funcionales, pero también sirve para realizar pruebas de regresión en aplicaciones web, algo que a veces es verdaderamente complicado, según la aplicación, pero que es casi imprescindible en el mantenimiento y evolución de las aplicaciones si se quiere asegurar un nivel de capacidad adecuado en la entrega del producto. Además tiene una estructura en árbol que le da potencia, permitiendo que sea la imaginación de quien la use la que ponga los límites a la hora de diseñar el plan de prueba y brinda mayor cantidad de variantes para recoger los resultados obtenidos, que el resto de las herramientas gratis, lo que permiten hacer un análisis exhaustivo de las pruebas realizadas.

### **Conclusiones**

En este capítulo, se analizaron algunos conceptos importantes relacionados con el tema del desarrollo del portal, así como un análisis de las funcionalidades que con la implementación del portal web se quieren lograr.

Se estudiaron diferentes metodologías de desarrollo, tecnologías y herramientas para la implementación, seleccionando la Metodología XP para el desarrollo, empleando Drupal como sistema gestor de contenido, utilizando lenguajes de programación tales como PHP y HTML, utilizando como entorno de desarrollo ZendStudio por las características que el mismo presenta. Como sistema gestor de base de datos PostgreSQL es el candidato elegido, y Apache Web Server es la elección para las funciones de servidor.

Para la fase de pruebas según la metodología de desarrollo XP se definen pruebas de Aceptación de Software por parte del cliente utilizando técnicas de caja negra, y pruebas Unitarias auxiliándose de técnicas de caja blanca para la verificación de los módulos. Como herramienta para pruebas de Carga y Estrés, JMeter por sus cualidades es la selección empleada.

### CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

#### Introducción

Este capítulo contiene el análisis de la información vinculada con el objeto de estudio, y características asociadas al dominio del sistema; contiene a su vez la propuesta de solución del sistema a desarrollar, los roles de usuarios definidos para el acceso a la información y la seguridad en la aplicación, los requisitos funcionales denominados historias de usuario por la metodología de desarrollo seleccionada, y los requisitos no funcionales, así como la propuesta de interfaz de usuario diseñada.

#### 2.1 Propuesta del Sistema

La realización del siguiente trabajo se enfoca en la implementación y pruebas de un sistema que permita la comunicación entre los diferentes seguidores del gestor de base de datos PostgreSQL. Para ello se lleva a cabo la gestión de la información relacionada con los temas de interés para la Comunidad Técnica Cubana de PostgreSQL donde se podrá obtener conocimientos actualizados relacionados con el contenido especificado así como otras prestaciones. El portal estará constituido por una serie de secciones con el objetivo de brindar a todos los usuarios la siguiente información:

- Publicación de eventos.
- Amplia y detallada información sobre todo el desarrollo relacionado con PostgreSQL en Cuba.
- Noticias recientes sobre el tema.
- Publicación de servicios.
- Intercambio de ideas mediante el uso del foro.
- Gestión de usuarios.
- Gestión de información.
- Servicio de Galerías de imágenes y videos.

Parte de la información que brindará el portal podrá verse por cualquier tipo de usuario que tenga interés en interactuar con el sistema, aunque otro por ciento de la información podrá manipularse por usuarios autenticados, ya que presentarán un alto nivel de privilegios con respecto a los usuarios simples. El administrador será el que contará con todos los privilegios sobre el sistema en cuanto a los cambios a los cuales se someterá la aplicación teniendo un manejo total sobre ella.

### 2.2 Roles de Usuarios

Una de las premisas fundamentales a tener en cuenta cuando se comienza el desarrollo de cualquier sistema informático, la constituye el delimitar la audiencia a la cual va dirigido el mismo, teniendo en cuenta que esta puede estar dividida a su vez en grupos atendiendo a sus competencias. Dentro de la audiencia antes mencionada se incluyen, como personal relacionado al sistema, al administrador que será el encargado de gestionar toda la información, el usuario general y el usuario autenticado. Los usuarios del portal serán aquellos que obtengan un resultado de la ejecución de uno o varios procesos del sistema. A continuación se muestra un listado con los roles de usuarios definidos para esta aplicación:

Roles	Descripción
Administrador	Encargado de gestionar toda la información tanto para los usuarios autenticados como para los invitados. Establece los permisos pertinentes para los distintos tipos de usuarios
Usuario Autenticado	Tiene acceso a todo lo publicado en el sitio de forma pública definido por el administrador del sistema. Tendrá privilegios extras en dependencia de los permisos que le sean asignados por el administrador.
Usuario Anónimo	Cualquier otra persona que acceda a la aplicación. Su acceso a la misma es definido por el administrador funcional, así como su nivel de visualización de la información.

### 2.3 Seguridad en la aplicación

La seguridad en el CMS seleccionado es un tema que siempre está en constante cambio, Drupal es un sistema gestor de contenido de software libre, existe un equipo de desarrollo que esta en constante búsqueda de posibles fallas de seguridad, entonces cada persona que utilice Drupal debería saber que tiene la responsabilidad de actualizar sus instalaciones cada vez que alguien descubre un nuevo agujero de seguridad.

Por eso, la primera y principal tarea a desarrollar en este tema referente a la seguridad es no desestimar las advertencias de seguridad de Drupal, porque estas son actualizaciones muchas veces de nuevas vulnerabilidades descubiertas por parte del equipo de desarrollo encargado de la seguridad de Drupal. Sin embargo, aunque se mantenga su sitio perfectamente actualizado, probablemente esto no es suficiente para garantizar la seguridad del sitio.

La API (Application Program Interface) de Drupal y la configuración predeterminada están diseñados para ser seguros cuando se utiliza en los modos predeterminados. Temas como la inyección, ataques del tipo XSS (Cross Site Scripting), gestión de sesiones, y otras falsificaciones, tienen soluciones estándares de la API de Drupal.

### 2.3.1 Pautas a seguir para diferentes tipos de ataques:

#### Código en php, un camino muy simple al desastre:

Nunca darle permiso de ejecución de código php a alguien en quien no se confíe, y de hecho no se debe confiar en nadie para esta tarea. Este detalle es tan importante que desde Drupal 6, la posibilidad de ejecutar código en PHP debe activarse desde un módulo aparte que viene desactivado por defecto como se muestra en la siguiente figura:

<input type="checkbox"/>	<b>Path</b>	6.10	Permite a los usuarios cambiar el nombre a URL. Requerido por: Pathauto ( <b>desactivado</b> )
<input checked="" type="checkbox"/>	<b>PHP filter</b>	6.10	Permite la evaluación de fragmentos de código PHP.
<input type="checkbox"/>	<b>Ping</b>	6.10	Alerta a otros sitios cuando el de usted se actualiza.

Figura 1. Ejemplo de Módulo Desactivado

#### XSS (Ataques del tipo Cross Site Scripting):

El Cross-Site-Scripting es una vulnerabilidad que aprovecha la falta de mecanismos de filtrado en los campos de entrada y permiten el ingreso y envío de datos sin validación alguna, aceptando el envío de scripts completos, pudiendo generar secuencias de comandos maliciosas que impacten directamente en el sitio o en el equipo de un usuario.

Según estadísticas más del 40% de los ataques que sufren los sitios de Drupal son del tipo XSS. Alguien, de forma mal intencionada coloca un trozo de código en javascript en algún nodo, comentario, o lo que sea que permita el sitio. Y luego, otro usuario (generalmente un administrador), visualizando el contenido, le permite realizar tareas un tanto peligrosas.

Afortunadamente es bastante sencillo evitar este tipo de ataques, por lo menos a nivel básico:

**Primero:** Revisar los filtros de entrada, nunca permitir HTML sin filtrar en los contenidos. Solo permitir etiquetas confiables.

**Segundo:** Actualizar regularmente los módulos si presentan fallas de seguridad.

**Tercero:** Si se programan módulos propios, como es el caso seguir las normas de seguridad de la implementación de los mismos.

### **Inyecciones SQL:**

Inyección SQL es una vulnerabilidad informática en el nivel de la validación de las entradas a la base de datos de una aplicación. El origen es el filtrado incorrecto de las variables utilizadas en las partes del programa con código SQL. Es, de hecho, un error de una clase más general de vulnerabilidades que puede ocurrir en cualquier lenguaje de programación o de script que esté incrustado dentro de otro.

Una inyección SQL sucede cuando se inserta o "inyecta" un código SQL "invasor" dentro de otro código SQL para alterar su funcionamiento normal, y hacer que se ejecute maliciosamente el código "invasor" en la base de datos.

Para prevenir este tipo de ataques se recomienda utilizar la capa de abstracción de base de datos para evitar este tipo de ataques que posee Drupal, de esta manera el código será más robusto.

### **2.3.2 Módulos adicionales para la seguridad**

Como parte de la seguridad contamos con la inclusión de otro módulo que gestiona otros puntos de la seguridad no menos importantes a su vez, a continuación se relacionan:

**Security Review:** Realiza una revisión sobre el sitio de puntos vulnerables y muestra en un listado los problemas de seguridad existentes.

Con todas estas medidas y líneas de seguridad a seguir la aplicación estará menos vulnerable a todos estos ataques que se han descrito.

## **2.4 Historias de Usuario**

Uno de los artefactos más importantes que genera la metodología XP son las Historias de Usuario. Éstas tienen el mismo propósito que los casos de uso y son confeccionadas por el cliente. Las mismas expresan su punto de vista en cuanto a las necesidades del sistema. Son descripciones cortas y escritas en el lenguaje del usuario sin terminología técnica. Otra de sus características es que solamente proporcionan los detalles sobre la estimación del riesgo y cuánto tiempo conllevará su implementación. Su nivel de detalle debe ser el mínimo posible, de manera que permita hacerse una ligera idea de cuánto costará implementar el sistema.

A continuación una descripción de algunas historias de usuario definidas, las restantes se encuentran en el Anexo 1.

Historia de Usuario	
<b>Número :</b> 1	<b>Nombre:</b> Autenticar usuario
<b>Usuario:</b> Usuario general	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alto
<b>Puntos Estimados:</b> 0.2	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> El administrador o el usuario autenticado podrán autenticarse para acceder a algunas opciones del sistema. Datos para autenticarse: . Nombre del Usuario (Obligatorio). . Contraseña (Obligatorio).	
<b>Observaciones:</b> Para que un usuario pueda autenticarse en el sistema debe primero registrarse en el sistema.	

**Tabla 1. Historia de usuario 1**

Historia de Usuario	
<b>Número :</b> 2	<b>Nombre:</b> Gestionar Usuario
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alto
<b>Puntos Estimados:</b> 0.6	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> El administrador podrá adicionar, eliminar, mostrar un determinado usuario y modificar sus datos. Datos de usuario: <ol style="list-style-type: none"> <li>1 Usuario (Obligado)</li> <li>3 Dirección de correo electrónico (Obligado)</li> <li>4 Contraseña (Obligado)</li> <li>5 Confirmar Contraseña (Obligado)</li> <li>6 Estado de la cuenta</li> <li>7 Idioma</li> </ol>	
<b>Observaciones:</b> Se deben llenar el todos los campos que sean obligatorios para que se añada el usuario.	

**Tabla 2. Historia de usuario 2**

Historia de Usuario	
<b>Número :</b> 3	<b>Nombre:</b> Descargar información de los eventos

<b>Usuario:</b> Administrador, Usuario autenticado, Usuario general	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Media
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite descargar informaciones sobre un evento determinado.	
<b>Observaciones:</b> En dependencia del rol que desempeña el usuario autenticado en el sistema, podrá acceder a informaciones confidenciales y no confidenciales.	

**Tabla 3. Historia de usuario 3**

Historia de Usuario	
<b>Número :</b> 4	<b>Nombre:</b> Inscribir participación en eventos.
<b>Usuario:</b> Administrador, Usuario autenticado, Usuario general	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alto
<b>Puntos Estimados:</b> 0.4	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite a los usuarios inscribirse en un evento determinado mediante un formulario con sus datos .El usuario debe llenar los datos: <ul style="list-style-type: none"> <li>. Nombre (Obligatorio)</li> <li>. Apellidos(Obligatorio)</li> <li>. Carnet de identidad (Obligatorio)</li> <li>. Correo electrónico (Obligatorio)</li> <li>. Empresa o entidad (Obligatorio)</li> <li>. Curso (Obligatorio)</li> </ul>	
<b>Observaciones:</b> Todos los campos son de carácter obligatorio, en caso de que el usuario introduzca algunos de los campos incorrectos o los pueda dejar vacíos se muestra un mensaje de error.	

**Tabla 4. Historia de usuario 4**

Historia de Usuario	
<b>Número :</b> 5	<b>Nombre:</b> Enviar trabajos para el evento.
<b>Usuario:</b> Administrador, Usuario autenticado, Usuario general	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alto
<b>Puntos Estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al usuario enviar o subir un archivo al sitio para un determinado evento.	
<b>Observaciones:</b> Los documentos deben estar en formato PDF y deben ser de tamaño menor de 1 MB, en caso de que el usuario no introduzca un archivo del formato requerido o del tamaño especificado se muestra un mensaje de error.	

**Tabla 5. Historia de usuario 5**

Historia de Usuario	
<b>Número :</b> 6	<b>Nombre:</b> Insertar comentarios en el foro.
<b>Usuario:</b> Administrador, Usuario autenticado	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alto
<b>Puntos Estimados:</b> 0.2	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al usuario realizar comentarios en el foro, así como crear un nuevo tema de discusión.	
<b>Observaciones:</b> El usuario deberá autenticarse en el foro. Para esto previamente deberá registrarse en el foro.	

Tabla 6. Historia de usuario 6

## 2.5 Requisitos no Funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Estas propiedades reflejan las características que hacen al producto atractivo, usable, rápido y confiable.

### 1. Requerimientos de Software

Se necesitan como requerimientos mínimos:

- **Para el cliente:** Sistema Operativo Windows XP o superior y Linux, Navegador web Mozilla Firefox 3.0 o Internet Explorer 5 o versiones superiores.
- **Para el servidor:** Sistema Operativo Windows Server, Linux (cualquier distribución), Servidor web: Apache 2, Lenguaje de Programación: PHP 5.3.
- **Para la Base de Datos:** PostgreSQL 8.3 o superior.

### 2. Requerimientos de Hardware

Se necesitan como requerimientos mínimos:

- **Para el cliente:** Procesador Pentium III, 128 MB de RAM, 100 MB de disco duro.
- **Para el servidor:** Procesador Pentium IV a 3.0 GHz, 1 GB de RAM, 10 GB de disco duro.

### 3. Requerimientos de apariencia o interfaz externa

La aplicación contará con una interfaz amigable, con páginas no cargadas de mucha información, colores suaves, lo más cercano a una aplicación de escritorio.

### 4. Requerimientos de Seguridad

- **Confidencialidad:** Existencia de distintos roles que establezcan que la información sólo sea vista por aquellos usuarios que posean los privilegios suficientes; restringir la ejecución de acciones a usuarios sin credenciales que intenten acceder a las mismas.



- **Integridad:** Validación de los datos en el servidor para evitar estados inconsistentes. La información manejada por el sistema estará protegida del acceso y divulgación no autorizada. Se debe realizar la confirmación sobre acciones irreversibles como eliminaciones.
- **Disponibilidad:** El sistema estará disponible las 24 horas del día a los usuarios autorizados, garantizando el acceso a la información en cualquier momento. Los mecanismos utilizados para lograr la seguridad no obstruyen el acceso a la información.
- **Requerimientos de Usabilidad:** La aplicación podrá ser utilizada por cualquier personal interesado en saber sobre el tema, pero con restricciones previas.
- **Requerimientos de Rendimiento:** Reducción de los tiempos de respuestas y alta velocidad de procesamiento de la información a través de peticiones asincrónicas al servidor. Los tiempos de respuestas deben ser los más cortos posibles al igual que la velocidad de procesamiento de la información.

### 5. Restricciones del diseño y la implementación

Para el diseño y documentación de la aplicación se utiliza la metodología XP, usando el lenguaje de modelado UML 6.4, utilizando para el desarrollo Visual Paradigm 3.4, CMS Drupal. El lenguaje empleado será PHP.

### 6. Requerimientos de soporte

Garantía de instalación y prueba del sistema, además de un breve entrenamiento a los futuros usuarios, se contará con una extensa documentación sobre el tema. Se le dará asistencia técnica en un período de 6 meses.

### 7. Portabilidad

La aplicación estará montada sobre el sistema operativo GNU Linux.

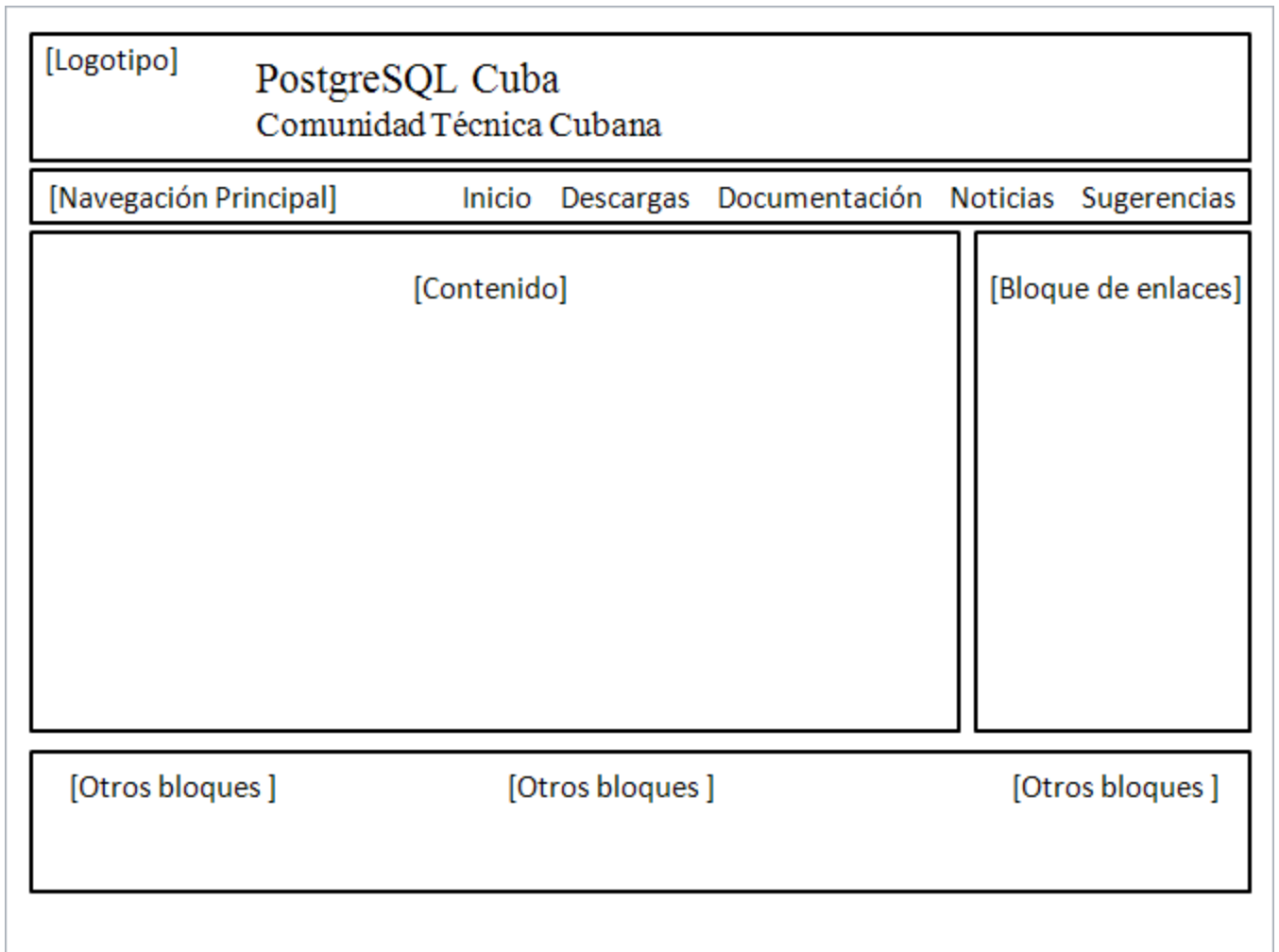
## 2.6 Prototipo de interfaz de usuario

Un prototipo es una visión preliminar del sistema futuro, es un modelo operable, fácilmente ampliable y modificable, que tiene todas las características que hasta el momento debe tener el sistema.

Las ventajas principales de los prototipos son:

- Posibilidad de cambiar el modelo.
- Oportunidad para suspender el desarrollo del modelo si no es funcional.
- Posibilidad de crear un nuevo modelo que se ajuste mejor a las necesidades y expectativas del usuario.

Tras el diseño realizado se propone el siguiente prototipo de interfaz de usuario, el mismo constituye un primer punto de partida dirigida a la interfaz inicial que presentará el desarrollo del portal para la Comunidad Técnica Cubana de PostgreSQL, lo cual significa que está sujeto a posibles cambios y que no será definitivo sino que se modificará en la medida que el desarrollo de la aplicación así lo requiera.



**Figura 2. Prototipo de Interfaz de Usuario**

El prototipo de interfaz de usuario quedó estructurado de la siguiente forma:

**Logotipo:** Se encuentra en la parte superior del portal. Identifica al logotipo de la Comunidad Técnica Cubana PostgreSQL. Esta sección es muy importante porque se representa la validez legal de la aplicación además de que aporta un elemento muy particular con la que se identifica el sistema.

**Menú horizontal:** Está ubicado en la parte superior debajo del logotipo del portal, como buena práctica de diseño, ya que el usuario acostumbra a verlo en esta parte de su navegador. Mostrará las funcionalidades como eventos, gestión, servicios y noticias.

**Autenticación del usuario:** Hacia la derecha del sitio se encuentra el área de autenticación del usuario, permite crear una cuenta nueva y solicitar una nueva contraseña para poder acceder a otras funcionalidades.

**Lugares:** En la parte lateral derecha del portal se pretende visualizar esta área donde se permitirá acceder a otras funcionalidades del portal como lo son foro, blog, galería de imágenes, video y encuestas.

**Buscador:** Se encuentra en la parte inferior derecha. Permite hacer una búsqueda en el sitio.

**Área de Trabajo:** El Área de Trabajo ocupa la mayor parte del espacio de la aplicación Muestra todo el contenido según la opción indicada. Es en esta área es donde se mostrarán todas y cada una de las Vistas del sistema, siendo algunas de estas simples, pero muchas otras de complejidad alta, se debe lograr una correcta distribución de los elementos que permita aprovechar el espacio.

Así queda estructurado el prototipo de interfaz de usuario propuesto y como se mencionó anteriormente sujeto a cambios.

### Conclusiones

En el presente capítulo se ha realizado un análisis a grandes rasgos de la propuesta del sistema a implementar, así como una clasificación de los roles y una descripción de los mismos, los cuales son una parte importante en el desarrollo de la aplicación en lo que respecta a los accesos de información que requieren cada uno, se analizó también el tema referente a las líneas a seguir para que el sistema sea seguro, se han analizado las principales historias de usuario identificadas con sus respectivas descripciones, de igual forma se analizaron los requisitos no funcionales que no dejan de poseer menor importancia y el prototipo de interfaz de usuario .

## CAPÍTULO 3: IMPLEMENTACION

### Introducción

Este capítulo contiene información relacionada con la implementación del sistema, a continuación se abordan tópicos como el plan de release, diseño de la base de datos, las pautas a seguir en el estándar de codificación utilizado, las tarjetas CRC (Clase- Colaboración -Responsabilidad) así como los patrones de diseño a utilizar.

### 3.1 Plan de Release

El plan de release es un artefacto generado durante el comienzo de la fase de desarrollo donde se describen las iteraciones a realizar con una descripción de las mismas así como las historias de usuario que se desarrollarán y su duración total. A continuación se representa:

Release	Descripción de la iteración	Orden de la HU a implementar	Duración total
Iteración 1	En la primera iteración se implementan las historias de usuario iniciales, obteniendo al final de la misma una primera versión de prueba y dando al sistema las funcionalidades básicas.	1, 2, 3, 4, 5, 6, 7,8,9	3 semanas
Iteración 2	En la segunda iteración se realiza la implementación de las historias de usuario con prioridad de negocio alta. Además, se corregirán errores o disconformidades del usuario con las historias de usuario implementadas en la iteración anterior. De esta forma se obtiene la segunda versión de pruebas del software. Esta segunda versión será mostrada a los clientes con el único objetivo de realizar cambios en base a la aceptación del mismo.	10 ,13, 12, 11, 14	3 semanas
Iteración 3	En la tercera iteración, ya implementadas las funcionalidades básicas, se realiza el desarrollo de las historias de usuario con prioridad media. Se corrigen errores de iteraciones anteriores, y se crean las bases para la implementación de las últimas historias de usuario que a su vez tienen una complejidad superior.	17, 15 ,16, 18, 20, 19	3.5 semanas
Iteración 4	En la cuarta iteración se implementarán los módulos que agregan funcionalidades adicionales a la aplicación, que no por eso dejan de ser menos importantes, y a su vez son las más complejas.	21,22, 23, 24, 25, 27, 26,28,29	4 semanas

**Tabla 7. Plan de release**

### 3.2 Diseño de la base de datos

El diseño de la base de datos es de gran importancia para el almacenamiento de los datos y para permitir a los usuarios recuperarla y actualizarla en base a sus peticiones posteriormente.

El modelo de datos describe la representación lógica y física de los datos persistentes en el sistema, es frecuentemente necesitado cuando se tiene un modelo de objetos y el mecanismo de almacenamiento se basa en una base de datos relacional. El modelo de datos del portal tiene baja complejidad, debido a que está desarrollado en el CMS Drupal con un alto número de entidades manejadas por el sistema, por lo que no se contemplarán todas en el modelo.

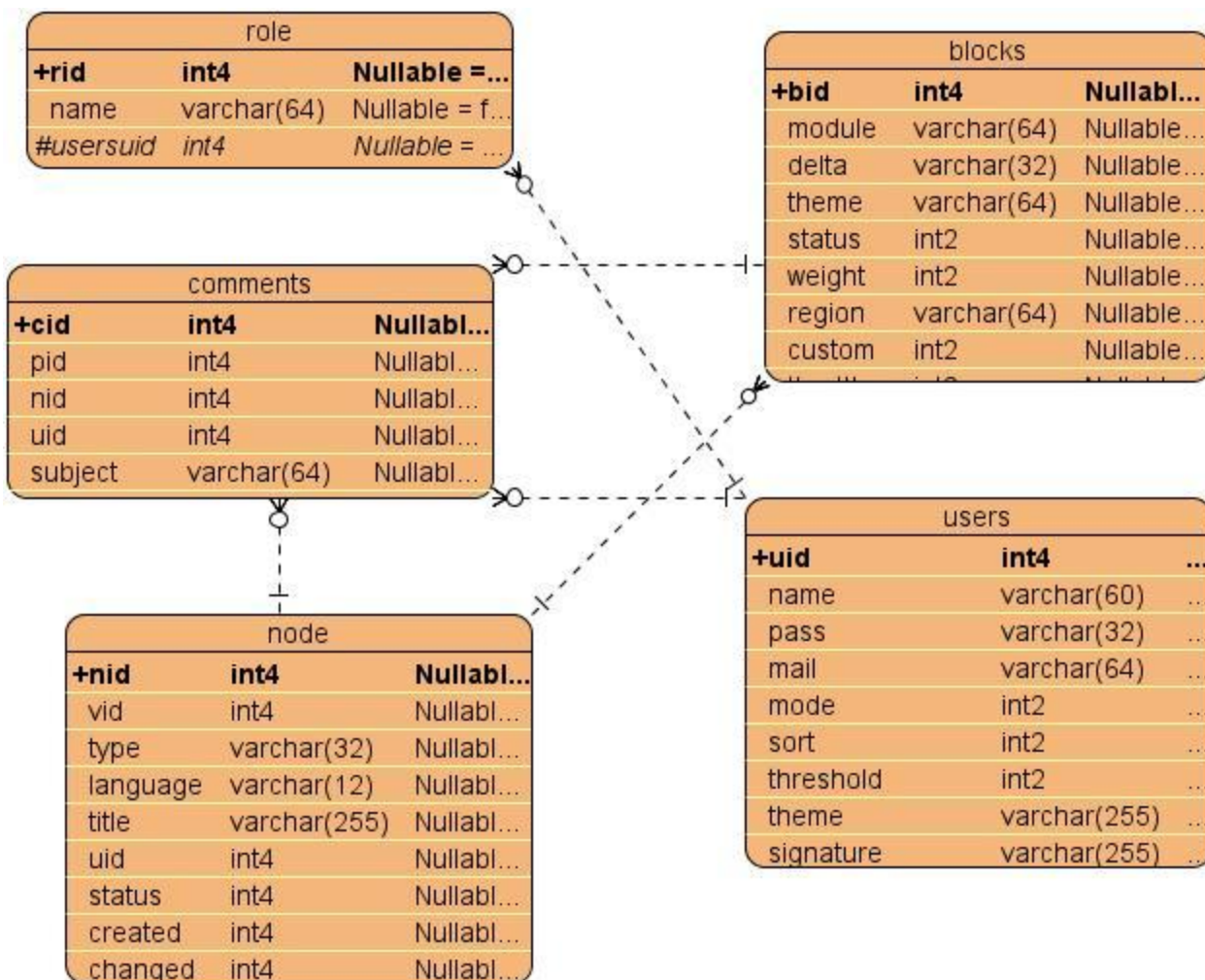


Figura 3. Diseño de la base de datos

### 3.3 Estándar de código

Los estándares permiten que el código que desarrollamos sea de una mejor calidad, que el mantenimiento de nuestros programas se vuelva menos complejo, y provoca que baje la tasa errores ingenuos. A continuación un resumen del estándar de codificación:

- **Indentación**

La indentación debe ser a cuatro espacios sin caracteres de tabulación. Esto es debido a que ciertos IDE's de desarrollo introducen caracteres de tabulación cuando indentan un texto automáticamente.

- **Estructuras de control**

Las estructuras de control deben tener un espacio entre el keyword de la estructura y el signo de apertura de paréntesis para distinguir entre las llamadas de las funciones y el signo de llaves debe estar sobre la línea de la estructura.

- **Llamadas de funciones**

Las funciones deben ser llamadas sin espacios entre el nombre de la función, el signo de paréntesis y el primer parámetro, espacios entre cada coma por parámetro y sin espacios entre el ultimo paréntesis, el signo de paréntesis cerrado y el signo de punto y coma (;).

- **Estilos de comentarios**

El estilo de los comentarios debe ser como el estilo de comentarios para C (`/* */` ó `//`), no debe de utilizarse el estilo de comentarios de Perl (`#`).

- **Inclusión de archivos**

Cuando se incluya un archivo de dependencia incondicionalmente se debe utilizar `require_once` y cuando sea condicionalmente, utilice `include_once`.

- **Bloques de código**

Siempre se debe utilizar las etiquetas `<?php?>` para abrir un bloque de código. No utilizar el método de etiquetas cortas, por que esto depende de las directivas de configuración en el archivo `PHP.INI` y hace que el script no sea tan portable.

- **Nombres**

Los nombres de las clases deben de iniciar con letra mayúscula. Los nombres de las variables y de las funciones pueden iniciar con letra minúscula, pero si estas tienen más de una palabra, cada nueva palabra debe iniciar con letra mayúscula (el nombre puede escribirse separado por signos de guión mayor). Si una función, en una clase, es privada; deberá comenzar con el signo de guión mayor para una fácil identificación. Las constantes deben de escribirse siempre en mayúsculas y tanto estas como las variables globales deben de tener como prefijo el nombre de la clase a la que pertenecen.

- **Constantes**

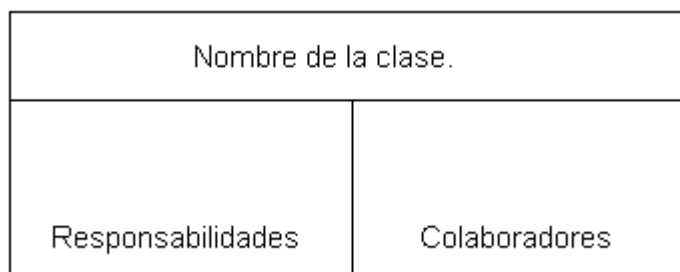
Los nombres de las constantes pueden contener caracteres alfanuméricos, guiones bajos (\_) y números. Todas las letras deben estar en mayúscula. Las constantes deben ser definidas como miembros de una clase usando el modificador "const". No se recomienda definir constantes con alcance global (utilizando la función "define").

Los estándares de código resultan importantes en cualquier proyecto de desarrollo, ayudan a asegurar que el código tenga una alta calidad, menos errores, y pueda ser mantenido fácilmente.

### 3.4 Tarjetas CRC

Para poder diseñar el sistema como un equipo se debe cumplir con tres principios: Cargo o Clase, Responsabilidad y Colaboración (CRC). Las tarjetas CRC permiten desprenderse del método de trabajo basado en procedimientos y trabajar con una metodología basada en objetos.

Las tarjetas CRC permiten que el equipo completo contribuya en la tarea del diseño. Una tarjeta CRC representa un objeto. El nombre de la clase se coloca a modo de título en la tarjeta, las responsabilidades se colocan a la izquierda, y las clases que se implican en cada responsabilidad a la derecha, en la misma línea que su requerimiento correspondiente, tal y como muestra la figura.



**Figura 4. Estructura de las tarjetas CRC**

**Clase:** es cualquier persona, cosa, evento, concepto, pantalla o reporte.

**Responsabilidades:** las responsabilidades de una clase son las cosas que conoce y las que realizan, sus atributos y métodos.

**Colaboradores:** los colaboradores de una clase son las demás clases con las que trabaja en conjunto para llevar a cabo sus responsabilidades. A continuación se muestran algunas tarjetas CRC consideradas las más importantes. El resto aparece en el anexo 2.

Tarjeta CRC	
Clase: Proyectos	
Responsabilidades	Colaboraciones
<ul style="list-style-type: none"> <li>✓ Listar proyectos en ejecución</li> <li>✓ Insertar vínculo al sistema de gestión de proyectos</li> <li>✓ Consultar información de los proyectos</li> <li>✓ Insertar vínculos de otras comunidades</li> <li>✓ Listar vínculos de otras comunidades</li> </ul>	Usuario

Figura 5. Tarjeta CRC 1

Tarjeta CRC	
Clase: Servicios	
Responsabilidades	Colaboraciones
<ul style="list-style-type: none"> <li>✓ Mostrar servicios</li> <li>✓ Inscripción de Usuarios (Cursos de Verano)</li> <li>✓ Consultar noticias</li> </ul>	Eventos Usuario

Figura 6. Tarjeta CRC 2



Tarjeta CRC	
Clase: Foro	
Responsabilidades	Colaboraciones
<ul style="list-style-type: none"> <li>✓ Gestión foro</li> <li>✓ Insertar comentarios en el foro</li> <li>✓ Participar foro</li> </ul>	<ul style="list-style-type: none"> <li>Usuario</li> <li>Servicios</li> </ul>

Figura 7. Tarjeta CRC 3

### 3.5 Patrones de Diseño en Drupal

Los patrones de diseño son los que expresan esquemas para definir estructuras de diseño con las que se pueden construir sistemas de software. Drupal hace uso de algunos patrones de diseño que son propiamente de sistemas orientados a objetos. Estos son:

- **Bridge:**

Los módulos deben ser escritos independientemente del sistema de base de datos que se está utilizando, y la capa de abstracción está proporcionada para ello. Se pueden definir nuevas capas de base de datos de acuerdo a la *API* definida, añadiendo soporte para otros sistemas de bases de datos sin necesidad de modificar el código del módulo. Desacoplando de esta manera la abstracción de su implementación.

- **Observer:**

Cuando se hace una modificación a un vocabulario del sistema de taxonomía de Drupal, el gancho taxonomía se llama en todos los módulos que lo implementan. Por la implementación del gancho, estos módulos quedan como observadores del objeto vocabulario; cualquier cambio que sufra este objeto, los módulos dependientes del objeto, actúan tanto como sea apropiado.

- **Chain of Responsibility**

En cada solicitud de la página, el menú del sistema determina si hay un módulo para gestionar la solicitud, si el usuario tiene acceso a los recursos solicitados, y que función se llama para hacer el trabajo. Para ello, el mensaje se pasa a la opción del menú correspondiente a la vía de la solicitud. Si el elemento de menú no puede manejar la petición, se pasa a otro eslabón de la cadena. Esto continúa hasta que un módulo se encarga de la petición, un módulo niega el acceso para el usuario, o la cadena se ha agotado.

- ***Reflection Pattern***

Describe el comportamiento de todo sitio basado en Drupal. Gracias a este patrón el sistema se convierte en una aplicación muy adaptable a diferentes entornos puesto que permite prácticamente la modificación de todo su comportamiento a través de los módulos instalables, sin la necesidad de modificar el núcleo.

### **Conclusiones**

Durante la realización de este capítulo se analizó el plan de release, en el cual se definen las historias de usuarios o requisitos funcionales a implementar por cada iteración y la duración de esta en semanas, se analizó a su vez el diseño de la base de datos , o sea el diagrama de clases persistentes, las cuales son las encargadas del almacenamiento de los datos , las buenas prácticas de programación a ejecutar por parte de los estándares de códigos definidos por algunos lenguajes de programación y otros definidos por parte del implementador así como lo relacionado a las tarjetas CRC y los patrones de diseño a utilizar.

### **CAPÍTULO 4: PRUEBAS Y VALIDACION DE LA SOLUCION PROPUESTA**

#### **Introducción**

Uno de los pilares de la metodología XP es el proceso de pruebas que lleva a los programadores a probar tanto como sea posible permitiendo disminuir el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección.

XP divide las pruebas del sistema en dos grupos: pruebas unitarias, encargadas de verificar el código mediante la técnica de caja blanca y diseñadas por los programadores, y pruebas de aceptación o pruebas funcionales mediante la técnica de caja negra destinadas a evaluar si al finalizar una iteración se lograron las funcionalidades propuestas.

Otro grupo de pruebas a realizar son las pruebas de carga y estrés que permiten evaluar el comportamiento del sistema ante situaciones determinadas de baja disponibilidad de recursos o en condiciones normales, y mediante las cuales se detectan vulnerabilidades y se pueden preparar planes de contingencia así como fortalecer las partes más débiles de nuestra aplicación.

#### **4.1 Prueba de caja blanca**

Dentro del método de prueba de caja blanca se encuentra la técnica del camino básico. La prueba del camino básico permite al diseñador de casos de prueba obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución. El resultado final ,denominado complejidad ciclomatica , es el número de caminos básicos independientes por donde el flujo puede circular, y da el número mínimo de casos de prueba a ejecutar para el código.

##### **4.1.1 Prueba del Camino básico**

A continuación las sentencias de un fragmento de código enumeradas mostradas en la figura:

```

<?
function inscripcion_page_form_validate ( $form , & $form_state ) {
    if(!($form_state [ 'values' ][ 'name' ])) { 1
        form_set_error ( '' , t ( 'Por favor complete todos los campos ' )); 2
    }
    if(!ereg("^[a-z A-Z á-ú]+$",$form_state [ 'values' ][ 'name' ])) 3
    {
        form_set_error ( '' , t ( 'El campo Nombre debe contener letras ' )); 4
    }
    if(!check_email_address($form_state [ 'values' ][ 'mail' ])) 5
    {
        form_set_error ( '' , t ( 'Email incorrecto ' )); 6
    }
    form_set_message ( '' , t ( 'Validación correcta ' )); 7
}
?>

```

Figura 8. Enumeración del código

Después de enumerar las sentencias se procede a realizar el grafo como se muestra en la siguiente figura:

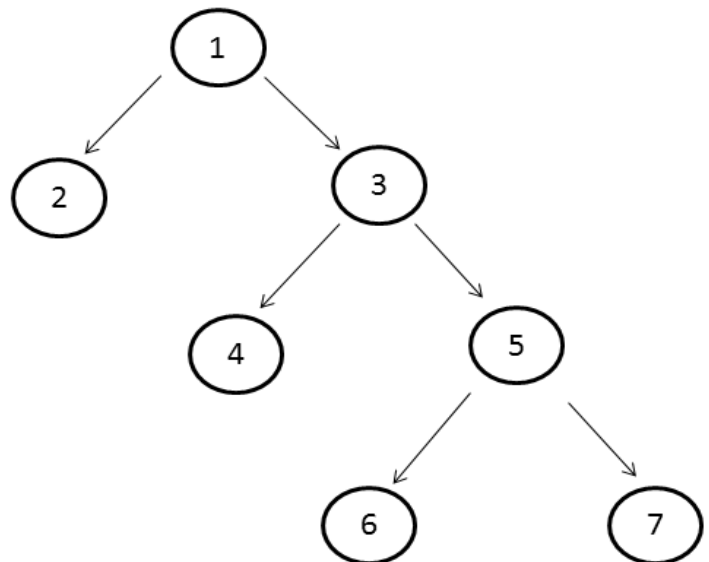


Figura 9. Gráfico de flujo asociado al código

Ahora se procede a calcular la complejidad ciclomática mediante la fórmula  $V(G) = P + 1$ , siendo **P** el número de nodos predicado, un nodo predicado es aquel que contiene una condición, se caracteriza porque dos o más bifurcaciones emergen de él, entonces  $V(G)=3+1=4$ . Esto significa que existen 4 posibles flujos que se pueden ejecutar y a la vez es el número mínimo de casos de pruebas a efectuar para el procedimiento en cuestión.

Es necesario representar los caminos básicos que puede recorrer el flujo, seguidamente se muestran en la siguiente tabla:

Camino básico #1	1 – 2
Camino básico #2	1 – 3 – 4
Camino básico #3	1 – 3 – 5 – 6
Camino básico #4	1 – 3 – 5 – 7

**Tabla 8. Caminos básicos del flujo del procedimiento**

Después de haber realizado los caminos básicos, es necesario realizar los casos de prueba, al menos uno por cada camino básico. Para la confección de los casos de prueba se requieren los siguientes parámetros:

**Descripción:** Se hace la entrada de datos necesaria, rectificando que ningún parámetro obligatorio esté vacío al entrar al procedimiento y que no se introduzcan datos erróneos.

**Condición de ejecución:** Se especifica cada parámetro para que cumpla una condición deseada para ver el funcionamiento del procedimiento.

**Entrada:** Se muestran los parámetros que entran al procedimiento

**Resultados Esperados:** Se expone el resultado que se espera que devuelva el procedimiento.

Descripción:	Se debe introducir un formulario y la variable \$form_state ['values'] ['name'] perteneciente a este formulario debe estar vacía
Condición de ejecución:	Cuando la variable \$form_state ['values'] ['name'] esta vacía.
Entrada:	\$form, \$form_state
Resultados esperados:	Se espera que el sistema muestre un mensaje de error indicando al usuario que dejó un campo vacío.

**Tabla 9. Caso de prueba para el camino básico 1**

Después de realizar los casos de prueba se verifica que el algoritmo esté cumpliendo con las condiciones necesarias que se plantean. De esta manera se realiza al menos un caso de prueba para cada camino básico, demostrando si el procedimiento ejecuta correctamente la función para la cual fue diseñado.

## 4.2 Prueba de caja negra

La prueba de caja negra son las pruebas que se llevan a cabo sobre la interfaz del software. O sea, los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa se mantiene. Las pruebas de caja negra, también denominada prueba de comportamiento, se centran en los requisitos funcionales del software (Pressman, 2002).

La prueba de caja negra intenta encontrar errores de las siguientes categorías.

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a bases de datos externas.
- Errores de rendimiento.
- Errores de inicialización y de terminación.

A continuación se muestran algunos de los casos de prueba que se refieren a las funcionalidades de Autenticar usuario, Gestionar Usuario e Inscribir participante en evento, las restantes se encuentran en el Anexo 2.

Escenario	Descripción	V 1	V2	Respuesta del sistema	Flujo central
EC 1.1 Autenticar Usuario correctamente	Prueba para la funcionalidad de Autenticar Usuario	V	V	El usuario registrado ingresa en el sitio.	1- El usuario hace clic en el link Inicio de Sesión. 2- El usuario introduce el usuario y la contraseña. 3- El usuario da clic en el botón Iniciar Sesión
EC 1.2 Autenticar Usuario. Flujo Alternativo	Prueba para la funcionalidad de Autenticar Usuario cuando el usuario escribe sus datos de	V	/	El sistema muestra un mensaje de error.	El usuario registrado debe dar clic en "Inicio de sesión", luego introduce sus datos de acceso con errores y hace clic en el botón "Iniciar sesión". El sistema muestra un mensaje de error. El usuario registrado debe introducir

	acceso incorrectos.				nuevamente sus datos de acceso y hacer clic en el botón "Iniciar sesión".
--	---------------------	--	--	--	---

**Tabla 10. Caso de prueba para Autenticar usuario**

Escenario	Descripción	Variable1	V2	V3	V4	V5	V6	V7	V8	V9	V10	Respuesta del sistema	Flujo central
EC 1.0 Modificar Usuario correctamente	Prueba para la función de Gestionar Usuario (Modificar datos de Usuario).	V	V	V	V	V	V	V	V	V	N/A	Se registra el usuario correctamente	1- El usuario hace clic en el link Administrar Usuarios. 2-El usuario selecciona el usuario que desea modificar y da clic en editar. 3-El usuario modifica los campos. 4-El usuario hace clic en el botón Guardar.
EC 2.0 Eliminar Usuario correctamente	Prueba para la función de Gestionar Usuario (Eliminar Usuario).	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Se elimina correctamente el usuario	1- El usuario hace clic en el link Administrar/ Usuarios. 2-El usuario para seleccionar el usuario que desea eliminar da clic en editar. 3-El usuario da clic en el botón Eliminar.
EC 3.0 Insertar Usuario Correctamente	Prueba para la funcionalidad de Gestionar Usuario (Insertar Usuario).	V	V	V	V	V	V	V	V	V	V	Se realizó la inserción correctamente	1- El usuario hace clic en el link Administrar/ Usuarios. 2-El usuario selecciona la pestaña

																		Agregar Usuario. 3-El usuario introduce los datos. 4-El usuario hace clic en el botón Crear nueva cuenta.	
EC 4.0 Listar usuario	Prueba para la función de Gestionar Usuario (Listar Usuarios).	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Se realizó la opción seleccionada correctamente .	1- El usuario hace clic en el link Administrar/ Usuarios. 2-El sistema muestra el listado de todos los usuarios.

Tabla 11. Caso de prueba para Gestionar usuario

Escenario	Descripción	Variable1	V2	V3	V4	V5	V6	Respuesta del sistema	Flujo central
EC 1.1 Inscribir en eventos	El usuario autenticado desea inscribirse para los eventos ofertados por la comunidad.	V	V	V	V	V	V	El sistema inscribe el curso deseado al usuario autenticado.	El usuario da clic en la opción de los eventos en el módulo derecho. Consulta el evento deseado y da clic en el vínculo a la inscripción del evento.  El sistema muestra el formulario de inscripción.  El usuario introduce los datos requeridos y da clic en el botón "Inscribir"  El sistema inscribe al usuario.
EC 1.2 Inscribir en eventos.  Flujo alterno.	El usuario autenticado desea inscribirse para los eventos ofertados por la comunidad dejando los datos incompletos.	I	V	V	V	V	V	El sistema no inscribe en el curso deseado al usuario autenticado.	El usuario da clic en la opción de los eventos en el módulo derecho. Consulta el evento deseado y da clic en el vínculo a la inscripción del evento.  El sistema muestra el formulario de inscripción.  El usuario introduce los datos requeridos dejando campos obligatorios en



									blanco o introduciendo datos erróneos y da clic en el botón "Inscribir"  El sistema emite un mensaje de error y no inscribe al usuario.
--	--	--	--	--	--	--	--	--	---

Tabla 12. Caso de prueba para Inscribir en eventos

### 4.3 Pruebas de Carga y Estrés

Las pruebas de carga y estrés se ejecutaron utilizando una PC con sistema operativo Linux (Ubuntu), con 1 GB de memoria RAM y microprocesador Pentium IV, instalando la máquina virtual de Java y ejecutando la herramienta JMeter donde se probaron las funcionalidades significativas.

Los resultados arrojan una serie de variables obtenidas de la ejecución de estas pruebas a las funcionalidades antes mencionadas y al sistema como un todo.

Se realizaron pruebas con la simulación de 200 usuarios interactuando con el sistema desde la misma URL, con un período de subida de 1 segundo y 100 hilos en 2 bucles, o sea 100 conexiones en un primer momento y luego 100 conexiones más.

Los resultados totales de las pruebas se puede observar en las imágenes que exporta la herramienta con todos los datos de interés. Ver Anexo 3.

En la siguiente tabla se relacionan algunos parámetros válidos de forma general.

Número	Variable	(100 hilos en dos bucles)
1	Muestras	14400
2	Media	799
3	Mediana	14
4	Línea del 90%	981
5	KB/sec	304.2
6	Rendimiento	56.7/sec

**Tabla 13. Variables del resultado de la herramienta JMeter**

A continuación las variables descritas en la tabla:

- **Muestras:** Número de muestras para cada URL.
- **Mediana:** Tiempo promedio que han tardado en cargarse las páginas.
- **Media:** Media del tiempo total que demoraron las peticiones en cargarse.
- **Línea 90%:** Tiempo máximo en que corrieron el 90 % de las peticiones reales.
- **Kb/Sec:** Velocidad de carga de las páginas.
- **Rendimiento:** Representa el número de muestras por unidad de tiempo.

Según se refleja en la tabla anterior para 14400 peticiones o muestras la velocidad de carga de las páginas es de 304.2 KB/sec y el rendimiento de la aplicación es de 56.7/sec o sea que es capaz de atender 56.7 peticiones por segundo, lo que demuestra que el sistema soporta una gran carga y estrés bajo condiciones de concurrencia de usuarios y de flujo de datos, manteniendo los parámetros relativamente estables en los casos simulados con una velocidad de carga aceptable y un rendimiento estable.

### Conclusiones

Luego de haber realizado el proceso de pruebas se obtuvieron una serie de resultados concretos. Con las pruebas de caja negra realizadas, a través de la técnica de partición de equivalencia y para la primera iteración se detectaron un total de 10 No Conformidades en la aplicación, las cuales fueron solucionadas. Además se aplicaron las Pruebas de Carga y Estrés con la herramienta Apache Jmeter arrojando los resultados de que la aplicación soporta situaciones de concurrencia de usuarios y de flujo de datos bajo condiciones anormales tal y como se esperaba manteniendo los parámetros de funcionamiento estables.

### **CONCLUSIONES GENERALES**

Con el propósito de darle cumplimiento a los objetivos y a la problemática planteada, se han llevado a cabo satisfactoriamente cada una de las tareas que fueron trazadas al comienzo del trabajo.

Durante el desarrollo del presenta trabajo de diploma se estudiaron las técnicas, metodologías, lenguajes de programación y herramientas necesarias para la construcción de la aplicación seleccionándose las más adecuadas para cada caso y cada propósito. Se implementaron los módulos necesarios para lograr el cumplimiento de los requisitos inicialmente definidos, se realizaron las pruebas correspondientes y se analizaron sus resultados contribuyendo a la validación del producto final, obteniéndose un portal web que permite la publicación y gestión de toda la información que necesitan los miembros de la Comunidad Cubana de PostgreSQL.

### **RECOMENDACIONES**

Como parte del proceso constante de investigación y desarrollo de la aplicación, surgen ideas que sería recomendable tener en cuenta para el futuro mejoramiento del sistema, como lo es la agregación de nuevas funcionalidades de acuerdo con las nuevas expectativas que se puedan crear dentro de la comunidad.

### REFERENCIAS BIBLIOGRAFICAS

1. ArPUG. Grupo de usuarios PostgreSQL de Argentina. *ArPUG. Grupo de usuarios PostgreSQL de Argentina*. [Online] [Citado: Mayo 24, 2010.] Disponible en: <http://www.arpug.com.ar/>
2. PostgreSQL-es.org. Portal en español sobre PostgreSQL. *PostgreSQL-es.org. Portal en español sobre PostgreSQL*. [Online] [Citado: Mayo 25, 2010.] Disponible en: <http://www.postgresql-es.org/>
3. PostgreSQL Chile. La base de datos opensource más avanzada. *PostgreSQL Chile. La base de datos opensource más avanzada*. [Online] Noviembre 13, 2009. [Citado: Mayo 25, 2010.] Disponible en: <http://www.postgresql.cl/>
4. PostgreSQL.it. *PostgreSQL.it*. [Online] Diciembre 1, 2009. [Citado: Mayo 27, 2010.] Disponible en: <http://www.psql.it/>
5. Drupal Hispano. Comunidad de usuarios de Drupal. *Drupal Hispano. Comunidad de usuarios de Drupal*. [Online] Abril 11, 2005. [Citado: Mayo 26, 2010.] Disponible en: <http://www.drupal.org/es/>
6. Jacobson, Ivar. *El proceso unificado de desarrollo de Software*. España: Addison-Wesley Iberoamericana España, 2000.
7. Free Download Manager. *Free Download Manager*. [Online] [Citado: Mayo 27, 2010.] Disponible en: [http://www.freedownloadmanager.org/es/downloads/Paradigma\\_Visual\\_para\\_UML\\_\(M%C3%8D\)\\_14720\\_p/](http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_(M%C3%8D)_14720_p/)
8. Hernández Orallo, Enrique. *Disca*. [Online] 2009. [Citado: Mayo 26, 2010.] Disponible en: <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>
9. White, Steve. *Ayuda de Joomla*. [Online] [Citado: Mayo 25, 2010.] Disponible en: <http://ayuda.joomlaspanish.org/ayuda-joomla/>
10. Joomlaos.net. *Joomlaos.net*. [Online] 2009. [Citado: Mayo 25, 2010.] Disponible en: <http://www.joomlaos.net/caracteristicas-de-joomla.php>

11. Rodríguez Doforno, Daniel o. CMS. *CMS*. [Online] [Citado: Mayo 25, 2010.] Disponible en: <http://danderesi.files.wordpress.com/2007/10/cms-daniel-rodriguez-doforno.pdf>
  
12. WordPress. Español. *WordPress. Español*. [Online] [Citado: Mayo 26, 2010.] Disponible en: <http://es.wordpress.org/>
  
13. Drupal Hispano. Comunidad de usuarios de Drupal. *Drupal Hispano. Comunidad de usuarios de Drupal*. [Online] Abril 11, 2005. [Citado: Mayo 26, 2010.] Disponible en: <http://www.drupal.org/es/>
  
14. Lenguajes de programación. *Lenguajes de programación*. [Online] 2009. [Citado: Mayo 26, 2010.] Disponible en: <http://www.lenguajes-de-programacion.com/lenguajes-de-programacion.shtml>
  
15. PostgreSQL-es.org. *PostgreSQL-es.org*. [Online] Marzo 22, 2009. [Citado: Mayo 26, 2010.] Disponible en: [http://www.postgresql-es.org/sobre\\_postgresql](http://www.postgresql-es.org/sobre_postgresql)
  
16. Barrapunto.org. *Barrapunto.org*. [Online] [Citado: Mayo 26, 2010.] Disponible en: <http://acs.barrapunto.org/articulos/trunk/LinuxActual/Apache/apache.pdf>
  
17. Taringa.net. *Taringa.net* [Online] [Citado: Noviembre 23,2010.] Disponible en : [http://www.taringa.net/posts/downloads/5432111/PHP\\_Designer\\_v7\\_2\\_1\\_15\\_WinAll\\_Incl\\_Keygen-CRD.html](http://www.taringa.net/posts/downloads/5432111/PHP_Designer_v7_2_1_15_WinAll_Incl_Keygen-CRD.html)
  
18. Zend.com *Zend.com* [Online] [Citado: Noviembre 26,2010.] Disponible en: <http://www.zend.com/en/products/studio/>

## ANEXOS

## Anexo 1 Historias de usuario

Historia de Usuario	
<b>Número :</b> 7	<b>Nombre:</b> Acceder a información de la comunidad.
<b>Usuario:</b> Administrador, Usuario autenticado, Usuario general	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Medio
<b>Puntos Estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al usuario acceder a información de la Comunidad como es la misión, visión, objetivos y su historia o surgimiento.	
<b>Observaciones:</b>	

Tabla 1. Historia de usuario 7

Historia de Usuario	
<b>Número :</b> 8	<b>Nombre:</b> Gestionar noticias
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Medio
<b>Puntos Estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al administrador insertar, modificar, visualizar y eliminar noticias.	
<b>Observaciones:</b>	

Tabla 2. Historia de usuario 8

Historia de Usuario	
<b>Número :</b> 9	<b>Nombre:</b> Gestionar roles
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Medio
<b>Puntos Estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al administrador del sistema añadir, modificar o eliminar los roles de usuarios existentes en la aplicación. Para añadir un rol el administrador introduce los datos: Nombre(Obligatorio) Para editar un rol el administrador modifica uno de los datos Nombre(Obligatorio) o algún permiso de este rol	

Para eliminar el rol el administrador selecciona el rol y lo efectúa la operación.
<b>Observaciones:</b>

Tabla 3. Historia de usuario 9

Historia de Usuario	
<b>Número :</b> 10	<b>Nombre:</b> Consultar eventos
<b>Usuario:</b> Usuario general	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Medio
<b>Puntos Estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> El usuario puede consultar la información disponible de le eventos en cuestión a efectuarse o ya efectuados.	
<b>Observaciones:</b>	

Tabla 4. Historia de usuario 10

Historia de Usuario	
<b>Número :</b> 11	<b>Nombre:</b> Descargar información de los eventos
<b>Usuario:</b> Administrador, Usuario autenticado, Usuario general	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Media
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite descargar informaciones sobre un evento determinado.	
<b>Observaciones:</b> En dependencia del rol que desempeña el usuario autenticado en el sistema, podrá acceder a informaciones confidenciales y no confidenciales.	

Tabla 5. Historia de usuario 11

Historia de Usuario	
<b>Número :</b> 12	<b>Nombre:</b> Listar vínculos a otras comunidades
<b>Usuario:</b> Administrador, Usuario autenticado, Usuario general	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Media
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Muestra un listado de todos los link a otras comunidades.	
<b>Observaciones:</b>	

Tabla 6. Historia de usuario 12



Historia de Usuario	
<b>Número :</b> 13	<b>Nombre:</b> Gestionar foro
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Media
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> El administrador podrá añadir, modificar o eliminar toda la información del foro.	
<b>Observaciones:</b>	

Tabla 7. Historia de usuario 13

Historia de Usuario	
<b>Número :</b> 14	<b>Nombre:</b> Gestionar productos
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alta
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al administrador publicar un nuevo producto (Soluciones desarrolladas para el trabajo con PostgreSQL) así como modificar información sobre estos y eliminar la publicación.	
<b>Observaciones:</b>	

Tabla 8. Historia de usuario 14

Historia de Usuario	
<b>Número :</b> 15	<b>Nombre:</b> Descargar productos
<b>Usuario:</b> Administrador, Usuario autenticado ,Usuario general	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Media
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> El usuario tiene la posibilidad de descargar productos como son las herramientas o los instaladores publicados en el sitio.	
<b>Observaciones:</b>	

Tabla 9. Historia de usuario 15

Historia de Usuario
---------------------

<b>Número</b> : 16	<b>Nombre</b> : Consultar noticias
<b>Usuario</b> : Usuario general	
<b>Prioridad en negocio</b> : Alta	<b>Riesgo de desarrollo</b> : Medio
<b>Puntos Estimados</b> : 1	<b>Iteración asignada</b> :4
<b>Programador responsable</b> : Pedro Valentín Paneque Silva	
<b>Descripción</b> : Permite consultar las noticias disponibles sobre la Comunidad y el Gestor.	
<b>Observaciones</b> :	

Tabla 10. Historia de usuario 16

Historia de Usuario	
<b>Número</b> : 17	<b>Nombre</b> : Consultar comentarios del foro.
<b>Usuario</b> : Administrador, Usuario autenticado, Usuario general	
<b>Prioridad en negocio</b> : Alta	<b>Riesgo de desarrollo</b> : Alto
<b>Puntos Estimados</b> : 0.2	<b>Iteración asignada</b> : 2
<b>Programador responsable</b> : Pedro Valentín Paneque Silva	
<b>Descripción</b> : Permite al usuario consultar comentarios posteados en el foro.	
<b>Observaciones</b> :	

Tabla 11. Historia de usuario 17

Historia de Usuario	
<b>Número</b> : 18	<b>Nombre</b> : Buscar información existente en el sistema
<b>Usuario</b> : Administrador, Usuario autenticado, Usuario general	
<b>Prioridad en negocio</b> : Alta	<b>Riesgo de desarrollo</b> : Medio
<b>Puntos Estimados</b> : 0.2	<b>Iteración asignada</b> : 3
<b>Programador responsable</b> : Pedro Valentín Paneque Silva	
<b>Descripción</b> : Permite al usuario realizar búsquedas sobre la información contenida en el sitio.	
<b>Observaciones</b> : El usuario debe introducir un criterio para la búsqueda.	

Tabla 12. Historia de usuario 18

Historia de Usuario	
<b>Número</b> : 19	<b>Nombre</b> : Consultar inscripciones de eventos
<b>Usuario</b> : Administrador	
<b>Prioridad en negocio</b> : Alta	<b>Riesgo de desarrollo</b> : Alto
<b>Puntos Estimados</b> : 0.2	<b>Iteración asignada</b> : 4
<b>Programador responsable</b> : Pedro Valentín Paneque Silva	

<b>Descripción:</b> Permite al administrador obtener un listado con todas las inscripciones efectuadas por los usuarios para los eventos publicados.
<b>Observaciones:</b>

Tabla 13. Historia de usuario 19

Historia de Usuario	
<b>Número :</b> 20	<b>Nombre:</b> Gestionar blog
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alta
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<p><b>Descripción:</b> El administrador podrá añadir, modificar o eliminar las páginas de blog publicadas en el sitio.</p> <p>Para añadir un blog el administrador llena los campos</p> <p>Título(Obligatorio) Cuerpo</p> <p>Para modificar un blog el administrador modifica alguno de los siguientes datos</p> <p>Título(Obligatorio) Cuerpo</p> <p>Para eliminar un blog el administrador selecciona el blog y efectúa la operación.</p>	
<b>Observaciones:</b>	

Tabla 14. Historia de usuario 20

Historia de Usuario	
<b>Número :</b> 21	<b>Nombre:</b> Escribir blog.
<b>Usuario:</b> Administrador, Usuario autenticado	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Medio
<b>Puntos Estimados:</b> 0.2	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al usuario crear una nueva bitácora o blog.	
<b>Observaciones:</b> El usuario deberá tener permisos para realizar estas acciones.(rol blog)	

Tabla 15. Historia de usuario 21

Historia de Usuario
---------------------

<b>Número :</b> 22	<b>Nombre:</b> Participar en encuestas
<b>Usuario:</b> Usuario autenticado	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alto
<b>Puntos Estimados:</b> 0.2	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al usuario realizar una encuesta determinada.	
<b>Observaciones:</b> El usuario deberá estar autenticado en el sistema.	

Tabla 16. Historia de usuario 22

Historia de Usuario	
<b>Número :</b> 23	<b>Nombre:</b> Realizar sugerencias
<b>Usuario:</b> Usuario autenticado	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alto
<b>Puntos Estimados:</b> 0.2	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al usuario realizar sugerencias a los desarrolladores del sitio. Para ello el usuario deberá introducir los siguientes datos: <ul style="list-style-type: none"> <li>• Nombre.(Obligatorio)</li> <li>• Tema</li> <li>• Sugerencia(Obligatorio)</li> </ul>	
<b>Observaciones:</b>	

Tabla 17. Historia de usuario 23

Historia de Usuario	
<b>Número :</b> 24	<b>Nombre:</b> Gestionar Encuestas
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alta
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al administrador insertar, eliminar, mostrar y actualizar toda la información de las encuestas del sitio.	
<b>Observaciones:</b>	

Tabla 18. Historia de usuario 24

Historia de Usuario
---------------------

<b>Número :</b> 25	<b>Nombre:</b> Gestionar chat
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alta
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al Administrador insertar o eliminar un chat.	
<b>Observaciones:</b>	

Tabla 19. Historia de usuario 25

Historia de Usuario	
<b>Número :</b> 26	<b>Nombre:</b> Participar chat
<b>Usuario:</b> Usuario autenticado	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alta
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al usuario interactuar con otros usuarios del sitio.	
<b>Observaciones:</b> El usuario debe estar autenticado en el sistema.	

Tabla 20. Historia de usuario 26

Historia de Usuario	
<b>Número :</b> 27	<b>Nombre:</b> Visualizar reportes estadísticos
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alta
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<b>Descripción:</b> Permite al usuario obtener datos estadísticos sobre el comportamiento del sistema. Los reportes pueden ser: <ul style="list-style-type: none"> <li>• Actividad de usuario(Ver reporte de actividad de usuario (visitas, nodos, comentarios)</li> <li>• Días del mes (Ver páginas por días del mes)</li> <li>• Días de la semana (Ver páginas por días de la semana)</li> </ul>	

<ul style="list-style-type: none"> <li>• Historia mensual (Ver contador de páginas por mes.)</li> <li>• Horas(Ver páginas por hora)</li> <li>• Host(Ver páginas por host)</li> <li>• Referidos(Lista de referidos)</li> <li>• Visitas recientes(Ver páginas que han sido recientemente visitadas)</li> <li>• Páginas top(Páginas visitadas frecuentemente)</li> </ul>
<b>Observaciones:</b>

Tabla 21. Historia de usuario 27

Historia de Usuario	
<b>Número :</b> 28	<b>Nombre:</b> Gestionar galería de imágenes
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alta
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	
<p><b>Descripción:</b> El administrador podrá añadir, modificar o eliminar las galerías de imágenes publicadas en el sitio.</p> <p>Para añadir una galería el administrador llena los campos</p> <p>Título(Obligatorio) Cuerpo</p> <p>Para modificar una galería de imágenes el administrador modifica alguno de los siguientes datos</p> <p>Título(Obligatorio) Cuerpo</p> <p>Para eliminar una galería de imágenes el administrador selecciona la galería y efectúa la operación.</p>	
<b>Observaciones:</b>	

Tabla 22. Historia de usuario 28

Historia de Usuario	
<b>Número :</b> 29	<b>Nombre:</b> Gestionar videos
<b>Usuario:</b> Administrador	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo de desarrollo:</b> Alta
<b>Puntos Estimados:</b> 0.5	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Pedro Valentín Paneque Silva	

---

<b>Descripción:</b> El administrador podrá añadir, modificar o eliminar los videos publicadas en el sitio.
<b>Observaciones:</b>

Tabla 23. Historia de usuario 29

**Anexo 2 Casos de Uso de Pruebas**

Escenario	Descripción	V 1	V 2	V 3	V 4	V 5	V 6	V 7	V 8	V 9	V 10	V 11	V1 2	V 13	V 14	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Escribir en Blog.	Prueba para la funcionalidad de escribir en un Blog.	V	V	V	V	V	V	V	V	V	V	V	V	V	V	Se realizó la operación seleccionada correctamente.	El usuario con r en Crear Conte Entrada de Bitá
<b>EC 1.2</b> Escribir en Blog Flujo Alterno	Prueba para la funcionalidad de escribir en un Blog de forma incorrecta.	I	V	V	V	V	V	V	V	V	V	V	V	V	V	La operación seleccionada no se ejecutó correctamente.	El usuario con r en Crear Conte Entrada de Bitá dejando campo introduciendo d un mensaje de

**Tabla 24. Caso de prueba para Escribir blog.**

Escenario	Descripción	V 1	V 2	V 3	V 4	V 5	V 6	V 7	V 8	V 9	V 10	V 11	V 12	V 13	V 14	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Modificar entrada de Blog	Prueba para la funcionalidad de Gestionar Blog (Modificar).	V	V	V	V	V	V	V	V	V	V	V	V	V	V	Se realizó la operación seleccionada correctamente.	El usuario, en e la opción de “B El sistema mue existentes hasta en caso de que El administrador un clic encima El sistema mue El administrador encuentra en la El sistema mue del mismo. El usuario modi clic en el botón



<p><b>EC 1.2</b> Eliminar entrada de Blog</p>	<p>Prueba para la funcionalidad de Gestionar Blog (Eliminar)</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>N/A</p>	<p>Se realizó la operación seleccionada correctamente.</p>	<p>El usuario, en e la opción de “B El sistema mue existentes hasta El usuario adm eliminar, da un El sistema mue El Administrado después da clic El sistema mue El usuario da c ejecuta la acció</p>
<p><b>EC 1.3</b> Modificar entrada de Blog(Flujo alternativo)</p>	<p>Prueba para la funcionalidad de Gestionar Blog (Modificar incorrectamente)</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>V</p>	<p>El sistema muestra mensaje de error.</p>	<p>El usuario, en e la opción de “B El sistema mue existentes hasta El administrador un clic encima El sistema mue El usuario da c encuentra en la El sistema mue del mismo. El usuario modi dejando en blan el botón “Guard realizados. El sistema emit</p>

Tabla 25. Caso de prueba para Gestionar blog.

Escenario	Descripción	Variable 1	Variable 2	Respuesta del sistema	Flujo central
-----------	-------------	------------	------------	-----------------------	---------------

<b>EC 1.1</b> Descargar productos correctamente.	Prueba para la funcionalidad de Descargar Productos	v1	v2	Se realizó la opción seleccionada correctamente	El usuario autenticado o el u Descargas en el menú prima herramientas o soluciones d
---	---	----	----	---	--

**Tabla 26. Caso de prueba para Descargar productos.**

Escenario	Descripción	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Consultar eventos	Prueba para la funcionalidad de consultar eventos	El sistema muestra la página del evento escogido por el usuario.	El usuario escoge la funcionalidad evento y le aparecerán todos l el sitio, donde el escogerá cual es el que desea consultar y enton esa página apareciendo la información de los mismos.

**Tabla 27. Caso de prueba para Consultar eventos.**

Escenario	Descripción	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Vínculo a otras Comunidades.	Prueba para la funcionalidad de efectuar vínculo a otras Comunidades.	El sistema ejecuta el enlace de los vínculos a otras comunidades.	El usuario va al módulo inferior e encuentran listados todos los enl identificadas.  El usuario da clic encima del vínc  El sistema re-direcciona al usuari seleccionada.

**Tabla 28. Caso de prueba para Listar Vínculo a otras comunidades.**

Escenario	Descripción	Variable 1	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Realizar Búsqueda	El usuario introduce texto en el campo para realizar una búsqueda.	V	Se muestra una página con los elementos relacionados con la búsqueda realizada.	1- El Usuario introduce texto en el cuadro superior del sitio y presiona el botón de búsqueda y se muestra una página para  2- El sistema encuentra y muestra una p relacionados con la palabra que se intro
<b>EC 1.2</b> Realizar Búsqueda introduciendo texto incorrecto.	El usuario introduce texto incorrecto en el campo para realizar una búsqueda.	I	Se muestra una página en la que muestra posibles errores en la búsqueda realizada y se muestran otras opciones de búsqueda.	1- El Usuario introduce texto incorrecto parte superior del sitio y presiona el bot  2- El sistema no encuentra resultados y de texto, otras opciones de búsqueda y antes realizada.
<b>EC 1.2</b> Realizar Búsqueda sin introducir criterio de búsqueda.	El usuario no introduce texto en el cuadro de texto y presiona el botón Buscar.	I	El sistema no muestra resultado alguno.	1- El Usuario no introduce texto en el ca búsqueda.  2- El sistema muestra un mensaje de er búsqueda.

**Tabla 29. Caso de prueba para Realizar búsquedas de información.**

Escenario	Descripción	Variable 1	Variable 2	Respuesta del sistema	Flujo central
<b>EC 1.1.</b> Descargar información de los eventos satisfactoriamente	Prueba para la Descarga de información de los eventos	v1	v2	El sistema muestra la página del evento escogido por el usuario y permite al mismo realizar la descarga que desee.	El usuario escoge la función de los eventos que están publicados y desea realizar la descarga de información de estos eventos. El sistema le direccionará hacia esa página de información de estos eventos.

**Tabla 30. Caso de prueba para Descargar información de los eventos**

Escenario	Descripción	Variable 1	Respuesta del sistema	Flujo central
<b>EC 1.1.</b> Consultar Noticias	El usuario desea consultar las noticias publicadas en el portal.	--	El sistema muestra al usuario las noticias publicadas.	1- El usuario accede mediante el link a la parte superior derecha del portal. 2- El sistema le muestra una página con los títulos de las noticias. 3- El usuario hace clic en el titular de la noticia que desea leer. 4- El sistema le muestra la noticia seleccionada en una nueva página.

**Tabla 31. Caso de prueba para Consultar noticias**

Escenario	Descripción	Variable 1	Variable 2	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Acceder a Información GUP	Prueba para la funcionalidad de Acceder a Información Grupo de Usuarios PostgreSQL.	V	N/A	Se realizó la operación seleccionada correctamente.	El usuario selecciona la información para visualizar su contenido. El sistema muestra la interfaz de la Comunidad o en blanco en caso de error.
<b>EC 1.2</b> Acceder a Información (Misión-Visión Contacto)	Prueba para la funcionalidad de Acceder a Información	N/A	V	Se realizó la operación seleccionada correctamente.	El usuario selecciona la información para visualizar su contenido. El sistema muestra la interfaz de la Comunidad o en blanco en caso de error.

**Tabla 32. Caso de prueba para Acceder información de la comunidad**

Escenario	Descripción	V 1	V 2	V 3	V 4	V 5	V 6	V 7	V 8	V 9	V 10	V 11	V 12	V 13	V 14	Respuesta del sistema	Flujo central
-----------	-------------	-----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	-----------------------	---------------

<p><b>EC 1.1</b> Insertar Encuesta</p>	<p>Prueba para la funcionalidad de Gestionar Encuesta (Insertar Encuesta)</p>	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	<p><b>EC 1.2</b> Modificar Encuesta</p>	<p>Prueba para la funcionalidad de Gestionar Encuesta (Modificar Encuesta)</p>	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	I	<p><b>EC 1.3</b> Eliminar Encuesta</p>	<p>Prueba para la funcionalidad de Gestionar Encuesta (Eliminar Encuesta)</p>	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	<p><b>EC 1.4</b> Insertar Encuesta (Flujo Alternativo)</p>	<p>Prueba para la funcionalidad de Gestionar Encuesta (Insertar Encuesta de forma incorrecta.)</p>	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	<p><b>EC 1.5</b> Modificar Encuesta (Flujo Alternativo)</p>	<p>Gestionar Encuesta (Modificar Encuesta de forma incorrecta.)</p>	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
--	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	--	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	--	---	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	--	--	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	---	---	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

sistema muestra r

**Tabla 33. Caso de prueba para Gestionar Encuestas**

Escenario	Descripción	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Participar en el foro	Cualquier usuario puede participar en el foro	El sistema le da acceso al usuario a participar en el foro.	1- El Usuario no autenticado accede al foro mediante en la esquina superior derecha del sitio 2- El sistema lo re-direcciona a la página principal del 3- El usuario no autenticado puede ver todos los post comentario.

**Tabla 34. Caso de prueba para Participar en el foro**

Escenario	Descripción	Variable 1	Variable 2	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Realizar comentario	El Usuario se Autentica para poder insertar un comentario en el foro	V	V	El sistema permite al usuario autenticado insertar comentarios en el foro	1- El Usuario Autenticado puede hacer clic en la opción de tema o puede abrir un nuevo tema. 2- El sistema muestra una p el Usuario Autenticado puede o el tema con su respectivo : 3- Si se introdujo texto el sistema insertar el comentario o el te
<b>EC 1.2</b> Realizar comentario. Flujo Alterno.	El Usuario no Autenticado no se le permite insertar comentarios en el foro	I	V	El sistema no permite al usuario insertar comentarios en el foro hasta tanto no se identifique en mismo.	1- El Usuario que no está au foro no puede insertar un co 2- El sistema mostrará una p Autenticado Identificarse o R
		V	I		

**Tabla 35. Caso de prueba para Insertar comentarios en el foro**

Escenario	Descripción	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	Respuesta del sistema	Flujo cent
<b>EC 1.1</b> Gestionar Noticias (Insertar)	Prueba para la funcionalidad de Gestionar Noticias (Insertar).	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	Se realizó la operación seleccionada correctamente.	El adminis noticias, y puede inse formato rec
<b>EC 1.2</b> Gestionar Noticias (Eliminar)	Prueba para la funcionalidad	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	Se realizó la operación	El adminis noticias y

	de Gestionar Noticias (Eliminar)																	seleccionada correctamente.	puede elim
<b>EC 1.3</b> Gestionar Noticias (Modificar)	Prueba para la funcionalidad de Gestionar Noticias (Modificar)	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	Se realizó la operación seleccionada correctamente.	El administ noticias y h puede mod

**Tabla 37. Caso de prueba para Gestionar Noticias**

Escenario	Descripción	V 1	V 2	V 3	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Realizar Sugerencia	El usuario debe autenticarse en el sistema para realizar una sugerencia	V	V	V	La sugerencia es enviada y el usuario es trasladado por el sistema a la página principal del portal	1- El usuario autenticado hace clic en el superior del Portal 2- El sistema muestra en el área de trabajo insertar un formulario. 3- El usuario introduce los valores en los campos y hace clic en Enviar 4- El sistema acepta la sugerencia y traslada al usuario a la página principal del sitio
<b>EC 1.2</b> Realizar Sugerencia(Flujo Alternativo)	El usuario debe autenticarse en el sistema para realizar una sugerencia	I	I	I	El sistema no envía la sugerencia y muestra mensajes de error.	1- El usuario autenticado hace clic en el superior del Portal 2- El sistema muestra en el área de trabajo insertar un formulario. 3- El usuario no introduce los valores en los campos y hace clic en Enviar 4- El sistema muestra en la página principal del sitio mensajes de error de la sugerencia.

**Tabla 41. Caso de prueba para Realizar sugerencias**

Escenario	Descripción	Respuesta del sistema	Flujo central
<b>EC 1.1</b> Participar Encuestas	Caso de Prueba para el escenario Participar en encuestas satisfactoriamente.	El Sistema le muestra al usuario todas las encuestas disponibles.	1- El usuario autenticado accede al sitio mediante el menú en el área derecha del sitio. 2- El sistema le muestra las encuestas disponibles en el área izquierda del sitio. 3- El usuario accede a la encuesta seleccionada y se encuentra referenciado con su nombre. 4-El sistema guarda los datos introducidos.
<b>EC 1.2</b> Participar Encuestas(Flujo Alternativo)	Caso de Prueba para el escenario Participar en encuestas introduciendo datos erróneos.	El sistema muestra mensaje de error.	1- El usuario autenticado accede al sitio mediante el menú en el área derecha del sitio. 2- El sistema le muestra las encuestas disponibles en el área izquierda del sitio. 3- El usuario accede a la encuesta seleccionada y se encuentra referenciado con su nombre. 4-El sistema emite un mensaje de error al introducir datos erróneos o dejando campos vacíos.

Tabla 42. Caso de prueba para Participar en encuestas

Escenario	Descripción	Respuesta del sistema	Flujo central
<i>EC1.1.</i> Enviar trabajos para el evento de forma correcta	Prueba para la funcionalidad de enviar trabajos para el evento.	El sistema muestra el formulario de envío escogido por el usuario y realiza el mismo.	El usuario autenticado o anónimo hace trabajos en la página del evento en c aparece adjunta el archivo a subir.
<i>EC1.2.</i> Enviar trabajos (Flujo alterno)	Prueba para enviar trabajos para el evento de forma incorrecta	El sistema muestra mensaje de error.	El usuario autenticado o anónimo hace trabajos en la página del evento en c aparece adjunta un archivo que no cu en formato PDF o del tamaño requerido

Tabla 43. Caso de prueba para Participar en Enviar trabajos para Eventos.

Escenario	Descripción	Variable 1	Variable 2	Respuesta del sistema	Flujo central
<i>EC1.1.</i> Participar en el chat de forma satisfactoria.	Prueba para la funcionalidad de Participar en el Chat	v1	v2	Se realizó la opción seleccionada correctamente	El usuario autenticado hace de chat, se lleva hacia una p entonces mediante un vínculo autenticado puede ir a las sa el que quiere participar.

Tabla 44. Caso de prueba para Participar chat

Escenario	Descripción	Respuesta del sistema	Flujo central
<i>EC 1.1</i> Consultar Inscripciones	El Administrador desea ver las inscripciones realizadas hasta el momento para el evento o curso.	El sistema muestra el listado de los usuarios inscritos en el evento o curso seleccionado.	El Administrador da clic en configuración del sitio-ins con las inscripciones hasta

Tabla 46. Caso de prueba para Consultar inscripciones.

## Anexo 3 Resultados arrojados por JMeter

Label	# Muestras	Media	Mediana	Linea de 90%	Rendimiento
/	600	434	386	1045	2,6/sec
/modules/user/...	6400	51	12	32	27,5/sec
/niftycube.js	800	401	289	991	3,5/sec
/node/54	200	941	665	2114	1,0/sec
/node/niftycub...	2200	366	24	978	9,7/sec
/safebrowsing/...	400	21008	20999	21003	1,6/sec
/node/46	200	729	591	1727	1,0/sec
/node/78	200	543	467	1342	1,0/sec
/foro	200	86	23	240	1,0/sec
/foro/	200	85	16	229	1,0/sec
/foro/style.php	200	30	14	78	1,0/sec
/foro/cron.php	200	355	17	1134	1,0/sec
/favicon.ico	200	7	5	17	1,0/sec
/node/135	200	445	14	1181	1,0/sec
/node/43	200	615	15	1643	1,0/sec
/blog	200	67	11	1095	1,0/sec
/sites/all/them...	200	16	11	35	1,0/sec
/node/149	200	348	12	1050	1,0/sec
/node/161	200	393	13	1092	1,0/sec
/node/66	200	365	9	1072	59,9/min
/node/56	200	403	16	1118	59,7/min
/node/65	200	404	17	1087	59,5/min
/node/63	200	416	18	1159	59,4/min
/imagenes/11.j...	200	7	5	15	59,4/min
/node/64	200	389	17	1075	59,2/min
<b>TOTAL</b>	<b>14400</b>	<b>799</b>	<b>14</b>	<b>981</b>	<b>56,7/sec</b>

Tabla 48. Resultado arrojado por JMeter