



FACULTAD # 8

**ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DEL PORTAL WEB PARA LA ESCUELA
CUBANA DE BOXEO**

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO EN
CIENCIAS INFORMÁTICAS**

AUTORES:

Pedro Castillo Urquiza

Mario Enrique González Benavides

TUTORES:

Ing. Yasmani Ceballos Izquierdo

Ing. Karenia Donatien Goliath

Ciudad de La Habana, junio de 2010

“Año 52 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaración de autoría

Declaramos que somos los únicos autores de este trabajo que lleva por título “Análisis, diseño e implementación del portal Web para la Escuela Cubana de Boxeo”, y autorizamos a la Universidad de las Ciencias Informáticas (UCI) a que haga el uso que estime pertinente con este trabajo.

Para que así conste firmamos la presente a los __ días del mes de junio de 2010.

Autores:

Pedro Castillo Urquiza

Mario Enrique González Benavides

Tutores:

Ing. Karenia Donatien Goliath

Ing. Yasmani Ceballos Izquierdo



“Tu tiempo está limitado, así que no lo desaproveches viviendo la vida de algún otro. No te dejes arrastrar por los dogmas, que es lo mismo que vivir con los resultados del pensamiento de otras personas. No dejes que el ruido de las opiniones de otros ahoguen completamente tu voz interior. Y más importante, ten el valor de seguir a tu corazón y a tu intuición. Ellos, de algún modo, ya saben en lo que verdaderamente te quieres convertir. Todo lo demás es secundario.”

Steve Jobs

Dedicatoria

A mis padres por todo el apoyo y el amor incondicional que siempre me han dado. Sin su ejemplo no hubiera podido lograr lo que he sido hasta hoy, he aquí la retribución por toda su confianza, ternura, dedicación e incondicionalidad.

Pedro

Dedico este trabajo y mi título a toda mi familia por darme apoyo e inspiración en todo momento y a todos mis compañeros de grupo y de la escuela en general y a todo aquel que puso su granito de arena durante estos cinco inolvidables años de estudio.

Mario

Agradecimientos

A mis padres por ser mi ejemplo profesional, por haberme guiado por el buen camino del estudio, la dedicación y la responsabilidad. A mi hermana, mis familiares y amistades por estar pendientes de mí todos estos años. A todos mis compañeros por permitirme el privilegio de tenerlos y hacer de estos cinco años juntos lo mejor que ha podido pasarme.

A todos los que hicieron posible la realización de este trabajo.

Pedro

Mis agradecimientos a todas aquellas personas que de una forma u otra influyeron en la realización de este trabajo. A mi novia por su apoyo durante todo este tiempo y a mis compañeros de aula por ser tan especiales y estar siempre disponibles cuando se necesitan.

Las amistades van cambiando y se renuevan al pasar del tiempo, pero cada una deja una huella que es lo que hace que se recuerden siempre. Un abrazo para esas amistades que no influyeron directamente pero que siempre estuvieron presentes y aportaron su granito de arena.

Mario

Resumen

Cuba ostenta un gran desarrollo en el deporte, pero en la actualidad no se cuenta con muchos espacios oficiales para dar cobertura a la información de los deportistas, incluso a equipos cubanos de gran prestigio. Hoy en día, los progresos en las denominadas tecnologías de la información tienen un gran impacto y están creadas las condiciones para centralizar y dar a conocer toda esta información.

El presente trabajo tiene como objetivo el desarrollo de un portal Web para la Escuela Cubana de Boxeo. Todo el proceso de desarrollo estuvo guiado por las especificaciones que propone la metodología XP, elaborando los artefactos propuestos en cada iteración. También se presenta un estudio del estado del arte donde se hizo una investigación sobre portales dedicados al deporte, las tecnologías y las tendencias actuales en el desarrollo de aplicaciones Web. Se presentan las pruebas realizadas al sistema para garantizar su aceptación por parte del cliente y la calidad final del producto. Con el *software* se facilita el trabajo de las personas encargadas de divulgar esta información, realizando una eficiente automatización de los procesos de gestión de la información deportiva del boxeo existente en la Escuela Cubana de Boxeo. La aplicación proveerá a los aficionados de un sitio oficial, al cual podrán acceder en cualquier momento para encontrar la información que deseen referente al equipo nacional de boxeo y otros temas de interés sobre este deporte.

Índice

| | |
|--|----|
| Introducción | 1 |
| 1. FUNDAMENTACIÓN TEÓRICA | 4 |
| 1.1 Introducción | 4 |
| 1.2 Conceptos asociados al dominio del problema | 4 |
| 1.3 Portales Web dedicados al deporte | 8 |
| 1.4 Aportes del estudio de los portales a la propuesta de solución | 10 |
| 1.5 Tendencias y tecnologías actuales | 11 |
| 1.5.1 Metodologías de desarrollo de <i>software</i> | 11 |
| 1.5.1.1 La Familia de Crystal | 12 |
| 1.5.1.2 SCRUM..... | 13 |
| 1.5.1.3 Desarrollo de <i>Software</i> Adaptable (ASD) | 13 |
| 1.5.1.4 Desarrollo Manejado por Rasgos (FDD) | 14 |
| 1.5.1.5 Programación Extrema | 15 |
| 1.5.1.6 Selección de la metodología | 17 |
| 1.5.2 Herramientas de modelado | 18 |
| 1.5.2.1 Rational Rose | 18 |
| 1.5.2.2 Visual Paradigm..... | 19 |
| 1.5.2.3 Lenguaje Unificado de Modelado (UML) | 20 |
| 1.5.2.4 Selección de la herramienta de modelado | 21 |
| 1.5.3 Principales lenguajes de programación | 21 |
| 1.5.3.1 Active Server Pages | 21 |
| 1.5.3.2 Python | 22 |
| 1.5.3.3 PHP | 24 |
| 1.5.3.4 Selección del lenguaje de programación..... | 25 |
| 1.5.4 Entorno de Desarrollo Integrado..... | 26 |
| 1.5.4.1 PHP Designer | 26 |

| | |
|---|-----------|
| 1.5.4.2 Zend Studio | 26 |
| 1.5.4.3 Adobe Dreamweaver | 27 |
| 1.5.4.4 Selección del Entorno de Desarrollo Integrado | 28 |
| 1.5.5 Sistema Gestor de Bases de Datos (SGBD) | 28 |
| 1.5.5.1 MySQL..... | 29 |
| 1.5.5.2 Oracle | 30 |
| 1.5.5.3 PostgreSQL | 31 |
| 1.5.5.4 Selección del Sistema Gestor de Bases de Datos | 31 |
| 1.5.6 Sistemas de Gestión de Contenidos..... | 32 |
| 1.5.6.1 Drupal | 33 |
| 1.5.6.2 Joomla | 34 |
| 1.5.6.3 Mambo..... | 36 |
| 1.5.6.4 Selección del CMS a utilizar..... | 36 |
| 1.6 Conclusiones | 37 |
| 2. DESARROLLO DE LA SOLUCIÓN | 38 |
| 2.1 Introducción | 38 |
| 2.2 Flujo actual de actividades en la Escuela Cubana de Boxeo | 38 |
| 2.3 Propuesta del Sistema..... | 39 |
| 2.4 Exploración, Planificación e Iteraciones | 40 |
| 2.5 Personas relacionadas con el sistema | 42 |
| 2.6 Historias de Usuario..... | 42 |
| Para ver las restantes HU consultar anexo 7. | 46 |
| 2.7 Iteraciones | 46 |
| 2.8 Prototipo de Interfaz de Usuario..... | 48 |
| 2.9 Estimación del esfuerzo por historia de usuario | 48 |
| 2.10 Plan de Entregas | 49 |
| 2.11 Conclusiones | 50 |

| | |
|---|----|
| 3. IMPLEMENTACIÓN Y PRUEBAS | 51 |
| 3.1 Introducción | 51 |
| 3.2 Diseño del sistema..... | 51 |
| 3.2.1 Tarjetas CRC..... | 52 |
| 3.2.2 Visión general de la arquitectura de Drupal | 53 |
| 3.3 Fase de implementación | 55 |
| 3.3.1 Iteración 1 | 55 |
| 3.3.1.1 Historias abordadas en la primera iteración | 55 |
| 3.3.1.2 Tareas de las HU abordadas en la primera iteración..... | 55 |
| 3.3.2 Iteración 2 | 60 |
| 3.3.2.1 Historias abordadas en la segunda iteración..... | 60 |
| 3.3.3 Iteración 3 | 60 |
| 3.3.3.1 Historias abordadas en la tercera iteración | 60 |
| 3.3.4 Iteración 4 | 61 |
| 3.3.4.1 Historias abordadas en la 4ta iteración | 61 |
| 3.3.5 Diagrama de Despliegue | 61 |
| 3.4 Pruebas | 62 |
| 3.4.1 Pruebas Unitarias..... | 63 |
| 3.4.2 Pruebas de Aceptación | 63 |
| 3.4.2.1 Iteración 1 | 64 |
| 3.5 Conclusiones | 74 |
| Conclusiones | 75 |
| Recomendaciones | 76 |
| Referencias bibliográficas | 77 |
| Bibliografía | 80 |
| Glosario de términos | 82 |
| Anexos | 84 |

Introducción

La Escuela Cubana de Boxeo es producto de los esfuerzos del Instituto Nacional de Deporte, Educación Física y Recreación (INDER). Desde sus inicios un grupo de promotores de esta escuela desarrolló prácticas importantes, de las cuales surgieron boxeadores que fueron convocados por el INDER para llevar a cabo numerosas tareas deportivas de carácter popular. Estos compañeros se convirtieron en entrenadores que contaron con recursos e impartieron sus conocimientos, permitiendo reconstruir las teorías y las prácticas del boxeo.

Todo esto permitió formar los nuevos estilos y patrones técnicos que corresponden con la idiosincrasia del boxeador cubano que se venía formando. Muchos han sido los resultados obtenidos por la Escuela Cubana de Boxeo, que ha sabido ubicarse entre los primeros lugares en la mayoría de los eventos en que ha participado, especialmente Campeonatos Mundiales y Juegos Olímpicos. Ha sido un factor decisivo en la ubicación de Cuba en la tabla general por países. Sus significativos resultados a lo largo de la historia hacen posible que el boxeo cubano sea mundialmente reconocido, lo que enorgullece a los seguidores del deporte de los puños en la isla. (1)

Actualmente, dentro de esta organización se identifican problemas con respecto a la divulgación de la información referente a este deporte, la cual es de difícil acceso y no existen mecanismos encargados a los que se pueda acudir para consultar sobre la historia y los logros del boxeo en el país. Estos datos normalmente se archivan en hojas de papel y se corre el riesgo de que se deterioren, ocurriendo pérdidas parciales o totales de información. Generalmente, el seguimiento a los boxeadores y a los eventos importantes puede hacerse mediante programas televisivos o radiales, siguiendo los criterios de los comentaristas y especialistas, pero los aficionados no tienen acceso para consultar los datos y las noticias que aquí se manejan. Han existido grandes boxeadores a través de la historia, pero muchas personas desconocen el balance de sus peleas, sus contrincantes, eventos en que participó, títulos alcanzados y otros datos de interés que permiten conocer a fondo la labor de los púgiles cubanos.

Es evidente la carencia de un mecanismo que dé seguimiento a toda la labor desempeñada por el boxeo cubano. Al ser planteadas estas necesidades, surge el siguiente **problema científico**: ¿Cómo lograr de forma efectiva brindar la información deportiva del boxeo referente a la Escuela Cubana de Boxeo?

De esta forma, el **objetivo general** es realizar un portal Web que gestione la información deportiva del boxeo existente en la Escuela Cubana de Boxeo.

Por tanto, el **objeto de estudio** de la investigación está dirigido al proceso de gestión de la información deportiva.

Teniendo en cuenta el objetivo general es necesario plantearse los siguientes **objetivos específicos**:

1. Definir el lenguaje, la metodología y las herramientas a utilizar.
2. Diseñar e implementar el portal Web para la Escuela Cubana de Boxeo.
3. Elaborar la documentación de los diferentes artefactos según los flujos de trabajo que proponga la metodología.

El **campo de acción** está dirigido al proceso de gestión de la información deportiva del boxeo en la Escuela Cubana de Boxeo.

La idea a defender es: si se realiza un portal Web referente al desarrollo del boxeo en el país se logrará brindar la información deportiva de este deporte existente en la Escuela Cubana de Boxeo.

Para dar cumplimiento a los objetivos trazados se proponen un conjunto de **tareas** que ayudan a que la investigación se haga de forma eficiente, las mismas son las que se exponen a continuación:

- Revisión bibliográfica del tema y estudio del mismo para definir el estado del arte.
- Estudio sobre portales Web dedicados al deporte.
- Estudio de las metodologías de desarrollo de *software*.
- Estudio de los lenguajes de programación para el desarrollo de aplicaciones Web.
- Estudio sobre las herramientas más utilizadas en el desarrollo de aplicaciones Web.
- Realización de la captura de requerimientos funcionales y no funcionales del portal.
- Implementación del portal Web del boxeo.
- Realización de pruebas funcionales a la primera versión.

Para recopilar lo abordado anteriormente, se utilizaron **métodos de investigación científica** teóricos y empíricos. Entre los **teóricos** se encuentra el Analítico-Sintético, que facilitó realizar una búsqueda bibliográfica y conformar el estado del arte, posibilitando sintetizar las características esenciales para arribar a conclusiones de investigación. Además, se utilizó el Análisis Histórico – Lógico que permitió estudiar la evolución de los portales Web dedicados al deporte en Cuba y en el resto del mundo, así como los diferentes tipos de portales Web.

Dentro de los **empíricos** se empleó la entrevista para obtener información sobre las necesidades de la Escuela Cubana de Boxeo, con el propósito de conocer las funcionalidades que debe tener el sistema a desarrollar. Ésta se emplea al inicio y durante todo el desarrollo del trabajo, pues a medida que se desarrolla el sistema se analiza si cumple las expectativas requeridas.

1. FUNDAMENTACIÓN TEÓRICA

1.1 Introducción

Las Tecnologías de la Información han sido conceptualizadas como la integración y convergencia de la computación, las telecomunicaciones y la técnica para el procesamiento de datos, sus principales componentes son: el factor humano, los contenidos de la información, las herramientas, la infraestructura, el *software* y los mecanismos de intercambio. Para la elaboración de un sistema de gestión de información es necesario estudiar su situación actual en el mundo y las tecnologías adecuadas para su desarrollo. A través de la historia de la humanidad, el hombre ha utilizado diferentes formas de comunicarse, desde la comunicación con señas, hasta la comunicación a distancia por medio de dispositivos tecnológicos avanzados. Los medios de comunicación y divulgación, se han perfeccionado alejándose un poco de los medios tradicionales. Así aparecieron los portales Web y se fueron convirtiendo en una de las alternativas más exploradas por los usuarios de internet.

1.2 Conceptos asociados al dominio del problema

El fenómeno de internet ha revolucionado el modo en que los seres humanos se relacionan y se comunican. Internet es el resultado de la interconexión de miles de computadoras de todo el mundo: *World Wide Web* (Tela de araña Mundial o Red de redes) o *www* como simplemente se le conoce, es toda la información accesible por medio de internet que está representada mediante páginas. Las principales ventajas de la *www* son: permite la combinación de diferentes tipos de representación de la información (ejemplos: textos, audio y video entre otros), se puede acceder a páginas de cualquier otro servidor conectado a internet sin importar su localización geográfica y la creación de páginas Web es bastante sencilla. La importancia de *www* es la difusión de la información puesto a que millones de personas tienen acceso a ella. Para realizar la exploración se hace mediante un *software* especializado denominado navegador (*browser* en inglés). Es uno de los medios de comunicación más importantes de la actualidad debido a que antes no había sido posible tener acceso a tantas personas con culturas y antecedentes tan diferentes.(2)

Página Web

La página Web es considerada por muchos la unidad básica de *www*. Una página Web es un documento adaptado para la Web que contiene información específica de uno o varios temas y que es almacenado en un servidor de internet, de tal forma que este documento pueda ser consultado en cualquier momento por toda persona que se

conecte a la red. Su principal característica es los hipervínculos, que pueden conducir a otra sección del documento o hacia otras páginas, siguiendo los enlaces se puede navegar por la red visitando páginas ubicadas en distintos lugares como si formaran una unidad. Es el medio o soporte de internet para la publicación de información, generalmente en formato HTML. Se clasifican de acuerdo con su contenido: si éste es predeterminado se conoce como página Web estática, pero si se genera en el momento de visualizar la página o de hacer alguna petición al servidor se denomina página Web dinámica.

Sitio Web

Un sitio Web (*website* en inglés) es un *software*, que contiene un conjunto de páginas Web que están relacionadas entre sí, dotadas de una dirección Web única. Su principal función es facilitar la comunicación entre gobierno, instituciones educativas, empresas, asociaciones y personas físicas, con el propósito de estrechar aún más los vínculos entre ellos. Por lo general se construye alrededor de una página de bienvenida que ofrece vínculos a otras páginas.

La diferencia principal entre un sitio Web y los medios tradicionales de comunicación, es que está en una red mundial de computadoras y está codificado de manera que los usuarios interactúen hasta cierto punto. De un sitio Web no se espera interacción con el usuario mayor que la localización de la información que él busca.

A veces se utiliza erróneamente el término página Web para referirse a sitio Web. Se debe tener en cuenta para no confundir los conceptos, que una página Web es un único documento, con un único nombre de archivo asignado, mientras que un sitio Web es un conjunto de archivos llamados páginas Web.

Portal Web

Un portal Web también es un sitio, pero está orientado a lograr la participación del usuario o lector, con el fin de obtener algo de éste. Es mucho más que un sitio porque es completamente dinámico, éste no sólo brinda información, sino que la recopila y lo que se espera es que el usuario interactúe con la empresa u organización pertinente.

El portal Web constituye un punto de entrada a internet donde se organizan sus contenidos, ayudando al usuario, concentrando servicios y recursos de forma que pueda encontrar allí todo lo que utiliza a diario sin necesidad de salir de dicho sitio. Su objetivo final es lograr fortalecer la relación del usuario y que se establezca entre éste y el portal un vínculo casi personal. En estos sitios se suelen encontrar herramientas

para compra electrónica, programas, documentos de toda clase, foros de usuarios y buscadores entre otros.

A continuación se menciona un grupo de características que permitirán reconocer un portal Web ante otros tipos de páginas o sitios:

- Personalización para usuarios finales.
- Recursos informativos organizados y divididos.
- Trayectoria o seguimiento a las actividades de los usuarios.
- Acceso a base de datos.
- Localización de personas o cosas importantes.

Debe quedar claro que un portal es, en todos los casos, un sitio Web, una página, pero no viceversa, de igual forma ni cualquier página ni cualquier sitio Web sería un portal.

Los gestores del portal podrían utilizar la información derivada de la identificación del usuario para dos objetivos básicos. Desde el punto de vista del servicio prestado, puede reconocerlo tanto a él como sus preferencias y a partir del seguimiento de sus actividades, anticipar lo que necesitará para tratar de adelantarse en lo posible a ello.

Las instituciones, ya sean privadas o públicas, suelen necesitar crear portales Web para dar acceso a sus contenidos informáticos, dado que una página muy sencilla en general no sería suficiente. El portal es considerado un intermediario de información que tiene como fuente de ingreso la publicidad de las entidades que ahí se anuncian. Se considera como la portada para un grupo de internautas que están buscando información precisa sobre algún tema.

Clasificación de los portales

Debido a la proliferación del número de portales y de usuarios, así como al vertiginoso desarrollo de internet, se hace necesario clasificarlos en función del público al que van dirigidos, es decir, la línea de contenidos que pretenden cubrir.

Entre las clasificaciones de los portales se pueden encontrar:

Generales (mega portales o portales horizontales): están orientados a todo tipo de público y ofrecen contenidos muy amplios, siendo su pretensión cubrir los temas más demandados. Generalmente incorporan servicios de valor añadido que buscan su seguimiento en torno a comunidades virtuales, tales como espacio Web gratuito, información de diversos tipos, personalización de la información, chat, correo gratuito, mensajes a teléfonos móviles, *software* de libre distribución, grupos de discusión, comercio electrónico o buscador entre otros. Este modelo tiende a resultar obsoleto,

pues se estima que sólo un número muy reducido puede ser capaz de ofrecer el adecuado nivel de servicios y contenidos a un público tan diverso. Ello hace que resulte insuficiente para los usuarios más expertos y profesionales, ya que exigen mayor especialización y profundidad, tanto en los servicios como en los contenidos, por eso están orientados fundamentalmente hacia los más inexpertos y menos exigentes.

Especializados: están orientados a cubrir la demanda de usuarios que por su experiencia o por sus necesidades profesionales no son satisfechos por los portales generales, esto se debe a que sus contenidos son muy globales. Con éstos se intentará cubrir un tema concreto o las necesidades de las personas relacionadas con una corporación específica, de aquí que estos se clasifiquen en verticales o corporativos.

1. Los corporativos tienden a ser una prolongación de las intranets corporativas, en las que se organiza la información y la navegación potenciando el acceso a datos de la propia institución, la edición de material de trabajo propio, el contacto con clientes y proveedores entre otros. En ellos se distingue la parte del cortafuegos dependiendo de que el destinatario de esa información sea miembro de la institución o un elemento ajeno a ésta.
2. Los verticales son portales Web que proveen información y servicios a un sector o industria en particular. Deben saber cómo atraer parte de los usuarios que los generales ya no son capaces de atender. Su potencial para ello radica en la posibilidad de profundizar en los contenidos que ofrecen y en su oferta de servicios personalizados. Éstos a su vez se clasifican en: geográficos y temáticos.

Las tendencias de desarrollo de portales Web apuntan a que en el futuro se reduzca drásticamente el número de portales generales por ser poco competitivos, o buscarán un momento idóneo para convertirse en verticales, para los que se vislumbran mejores expectativas de éxito. (3)

De acuerdo con las pretensiones del equipo se desarrollará un portal Web, ya que lo que se busca con el *software* es fomentar el boxeo cubano poniendo a disposición del usuario un determinado grupo de recursos y servicios. No sólo se quiere brindar información al usuario, sino que éste interactúe con el sitio, que pueda personalizar la interfaz de acuerdo con sus prioridades y recibir boletines informativos sobre temas específicos del boxeo. Además intercambiar opiniones con otros usuarios mediante foros de debates y podrá conocer más sobre los boxeadores cubanos. También se

dispondrá de espacios de descargas, encuestas, galerías de imágenes, videos, concursos, promociones y otras secciones con informaciones novedosas.

Una vez analizados los conceptos relacionados con los portales expuestos anteriormente y los objetivos que pretende el equipo de trabajo, se descarta el desarrollo de un portal general debido a que su futuro es incierto por las exigencias cada vez mayores de los usuarios. Se pretende realizar un portal especializado vertical ya que estará dirigido a un sector específico (los seguidores del boxeo) y bien definido, con contenidos concretos (información sobre el boxeo cubano). Dentro de los portales verticales entraría en los temáticos, pues éste se especializará en brindar servicios e informaciones que sean de interés para la Escuela Cubana de Boxeo.

1.3 Portales Web dedicados al deporte

Portal de la Federación Cubana de Béisbol Aficionado

Es un portal Web dedicado al acontecer actual del béisbol cubano en el ámbito nacional e internacional. El portal cuenta con una sección dedicada a la serie nacional en curso, donde brinda un calendario con los juegos que se efectuarán en cada jornada, resúmenes de los juegos, posiciones de los equipos, líderes individuales y colectivos de la serie. Además se puede acceder a las estadísticas de las series anteriores y otros eventos relacionados con el pasatiempo nacional. Posee información sobre las reglas del béisbol y el reglamento de la serie nacional en curso. Existe un espacio titulado Centro de Prensa donde se publican noticias, entrevistas y archivos que permiten acceder a noticias anteriormente publicadas.

Portal Web Boxeo.ws

Es un portal Web dedicado al boxeo profesional donde se aborda toda la información actualizada y posee diferentes secciones donde se destacan las siguientes:

- Video: muestra videos de peleas completas, están disponibles solamente a miembros de Boxeo.ws.
- Boxeadores: aparecen todos los boxeadores profesionales y el usuario puede acceder a la ficha técnica de cada uno de ellos.
- Artículos: están desglosados artículos y noticias referentes al boxeo profesional. El usuario tiene la posibilidad de hacer un comentario a los artículos que se publican en el sitio.
- Fotos: se publican fotos de los boxeadores profesionales, organizadas por fecha de publicación, hora y galería a que pertenece.

- Foros: un grupo de personas intercambia en forma online información, opiniones, preguntas y respuestas, archivos y todo tipo de material sobre diversos temas.
- Encuestas: se realizan encuestas relacionadas con deportistas, sobre noticias de última hora y sobre peleas que se aproximan.

ESPN Deportes

Portal Web dedicado al deporte, ofrece al usuario de forma sencilla e integrada el acceso a una serie de recursos y servicios referentes al deporte en general. Este portal Web incluye varias secciones, destacándose las siguientes:

- Buscador: encargado de realizar cualquier tipo de búsqueda en el sitio. Además incluye la sección refinar búsqueda, que ayuda al usuario a buscar por tipo la información, ya sea foto, video, audio o mercancía. La búsqueda se puede realizar por fecha y por deportes.
- Noticias: se dedica a las noticias referentes al deporte. Expone el apartado mis noticias donde el usuario registrado recibirá noticias de sus deportes, equipos y jugadores favoritos.
- Encuestas: se realizan encuestas relacionadas con deportistas destacados o sobre noticias relevantes.
- Fantasía & Juegos: se podrá disfrutar de juegos virtuales relacionados con los diferentes deportes.
- Foros: existen temas publicados relacionados con diferentes deportes y están organizados por categorías.

Mediante éste portal se puede acceder a **ESPN DEPORTES Boxeo** que se dedica al boxeo profesional, es similar a Boxeo.ws, pero ofreciendo más funcionalidades que facilitan la comprensión y seguimiento del boxeo, dentro de ellas se destaca:

- Ránkings divisionales: ofrece información relacionada sobre el ránking por cada división.
- Campeones: brinda información sobre los mejores de cada división.
- Credencial de pelea: ofrece una ficha técnica de próximas. La ficha contiene el nombre de los peleadores, fecha, lugar, hora y división. También se ofrece un pronóstico (organizado por asaltos) elaborado por un colaborador sobre cómo quedará la pelea al finalizar. Incluye fotos de los peleadores.

- Entrevistas a expertos: se muestran las entrevistas realizadas a los expertos en el boxeo.

Federación Argentina de box

Sitio Web Oficial de la Federación Argentina de Boxeo (FAB) que ofrece datos sobre la historia del boxeo argentino. Ofrece el medallero del boxeo argentino en los Juegos Olímpicos y Panamericanos. Brinda información detallada de los campeones mundiales argentinos y de los campeones en las diferentes divisiones del país. Posee vínculos a las páginas de diferentes organizaciones que rigen el boxeo a nivel mundial como:

- Asociación Internacional de Boxeo Amateur (AIBA).
- Asociación Mundial de Boxeo.
- Consejo Mundial de Boxeo (WBC).
- Federación Internacional de Boxeo (IBF).
- Organización Mundial de Boxeo (WBO).

1.4 Aportes del estudio de los portales a la propuesta de solución

Después de haber analizado una serie de portales deportivos publicados en la Web (dentro de ellos los mencionados con anterioridad), resulta de utilidad para la elaboración de la propuesta de solución auxiliarse en las facilidades de muchos de los servicios y funcionalidades que ofrecen. Se puede resaltar la sección de Noticias que informa al usuario de la actualidad del deporte a nivel nacional e internacional, la publicación del calendario competitivo facilita la planificación de los atletas y permite que el usuario esté actualizado de la fecha en que se efectuará algún evento competitivo. La información del ranking es algo interesante para los seguidores del deporte.

La publicación de entrevistas realizadas a atletas destacados puede ser una sección de mucho interés para los usuarios, así como la publicación del reglamento por el que se rige el deporte. Contar con enlaces a los sitios de diferentes organizaciones que rigen el boxeo a nivel mundial contribuye a elevar el nivel cultural del usuario. Las fichas sobre las peleas motivan a los usuarios a hacer pronósticos.

Los foros y las encuestas permiten interactuar con el usuario y sus comentarios enriquecen las técnicas y métodos para llevar adelante el deporte. La publicación de artículos permite a los usuarios seguir temas novedosos.

También es una excelente propuesta en el momento de diseñar el portal, tener en cuenta los colores rojo y azul que son los que más se manifiestan en el boxeo y representan a Cuba.

1.5 Tendencias y tecnologías actuales

1.5.1 Metodologías de desarrollo de *software*

Las metodologías de desarrollo de *software* son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos de *software*(4). Una metodología está compuesta por:

- ¿Cómo dividir un proyecto en etapas?
- ¿Qué tareas se llevan a cabo en cada etapa?
- ¿Qué restricciones deben aplicarse?
- ¿Qué técnicas y herramientas se emplean?
- ¿Cómo se controla y gestiona un proyecto?

Indican paso a paso todas las actividades a realizar para lograr el producto informático deseado, indicando además las personas que deben participar en el desarrollo de las actividades y su papel. También detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

El desarrollo de *software* no es una tarea fácil ya que existen numerosas propuestas metodológicas que inciden en distintas dimensiones del proceso de desarrollo. Por una parte están aquellas propuestas más tradicionales que se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, las herramientas y notaciones que se usarán. Estas propuestas han demostrado ser efectivas y necesarias en un gran número de proyectos, pero han presentado problemas en muchos otros. Una posible mejora es incluir en los procesos de desarrollo más actividades, más artefactos y más restricciones, basándose en los puntos débiles detectados. Sin embargo, el resultado final sería un proceso de desarrollo muy complejo que puede incluso limitar la habilidad del equipo para llevar a cabo el proyecto.

Una aproximación es centrarse en otras dimensiones, como por ejemplo el factor humano o el producto de *software*. Ésta es la filosofía de las metodologías ágiles, las cuales dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del *software* con iteraciones muy cortas. Este enfoque está mostrando su

efectividad en proyectos con requisitos muy cambiantes y cuando se exige reducir drásticamente los tiempos de desarrollo, pero manteniendo una alta calidad. Las metodologías ágiles están revolucionando la manera de producir *software* y generan un amplio debate entre sus seguidores y quienes no las ven como alternativa para las metodologías tradicionales.(5)

1.5.1.1 La Familia de Crystal

Consiste en una familia de metodologías con un código genético común. Cada proyecto y organización usará este código para generar su propia metodología. El nombre Crystal deriva de la caracterización de los proyectos según 2 dimensiones, tamaño y complejidad (como en los minerales, color y dureza). Por ejemplo, *Crystal Clear* para equipos de hasta 8 personas, *Crystal Yellow* para equipos entre 10 y 20 miembros y *Crystal Orange* para equipos de 20 a 25 personas, entre otras. La idea es poder armar distintas metodologías para distintos tipos de proyectos. Los roles definidos por Crystal son los siguientes:

- Patrocinador ejecutivo.
- Jefe de proyecto.
- Experto en el dominio.
- Experto de uso.
- Programador diseñador.
- Diseñador de interfaz de usuario.
- Realizador de pruebas.
- Programador técnico.

La guía de trabajo que presenta *Crystal Clear* es altamente recomendable para equipos pequeños. Da flexibilidad y prioriza la parte humana (como todas las metodologías ágiles), apuntando a lograr eficiencia, habitabilidad y confianza en los miembros del equipo.

Presta especial importancia a la ubicación física del grupo, donde la comunicación cumple el principal rol. La entrega frecuente de código confiable y funcionando mantiene el foco y evita distracciones en el personal de trabajo. El equipo es el que elige las técnicas que se van a aplicar según lo que consideren apropiado en cada proyecto. (6)

Crystal da vital importancia a las personas que componen el equipo de un proyecto y sus puntos de estudio son:

- Aspecto humano del equipo.
- Tamaño de un equipo (número de componentes).
- Comunicación entre los componentes.
- Distintas políticas a seguir.
- Espacio físico de trabajo.(5)

1.5.1.2 SCRUM

SCRUM es una metodología ágil para la gestión de proyectos. Es un proceso ágil y liviano que sirve para administrar y controlar el desarrollo de *software*. El desarrollo se realiza en forma iterativa e incremental (una iteración es un ciclo corto de construcción repetitivo). Cada ciclo o iteración termina con una pieza de *software* ejecutable que incorpora nueva funcionalidad. Las iteraciones en general tienen una duración entre 2 y 4 semanas. Se utiliza como marco para otras prácticas de ingeniería de *software* como el Proceso Unificado de Desarrollo (RUP de *Rational Unified Process* en inglés) o Programación Extrema (XP por sus siglas en inglés).

Potencia la formación de equipos de trabajos autosuficientes y multidisciplinarios, reduciendo la carga de gestión y proporcionando a los miembros del equipo un entorno amigable y productivo para desarrollar sus habilidades al máximo. Este entorno proporciona además mayor calidad de vida a los trabajadores y mejora drásticamente la moral en las organizaciones.(5)

1.5.1.3 Desarrollo de *Software* Adaptable (ASD)

Esta metodología se adapta al cambio en lugar de luchar contra el. Se basa en la adaptación continua a circunstancias cambiantes. En ella no hay un ciclo de planificación-diseño-construcción del *software*, sino un ciclo especular-colaborar-aprender. Al igual que otras metodologías ágiles, su funcionamiento es cíclico y reconoce que en cada iteración se producirán cambios e incluso errores. ASD hace énfasis en aplicar las ideas que se originaron en el mundo de los sistemas complejos, adaptación continua del proceso al trabajo.

Las principales características de ASD son:

- Iterativo.

- Orientado a los componentes de *software* (la funcionalidad que el producto va a tener, características y otras) más que a las tareas con las que se va a alcanzar dicho objetivo.
- Tolerante a los cambios.
- Guiado por los riesgos.
- La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo.

Utiliza un cambio orientado hacia el ciclo de vida, que tiene tres componentes:

- Especulación: es donde se inicia y se planifican las características del *software*.
- Colaboración: se desarrollan las características del *software*.
- Aprendizaje: se revisa la calidad, si no se tienen errores se entrega al cliente.

Las principales ventajas de ASD son las siguientes:

- La tercera fase del ciclo de vida, revisión de los componentes, sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo.
- Apunta hacia el Desarrollo Rápido de Aplicaciones (*Rapid Application Development* en inglés, por sus siglas RAD), el cual enfatiza velocidad de desarrollo para crear un producto de alta calidad, bajo mantenimiento e involucrando al usuario lo más posible.
- Utiliza información disponible de los cambios para mejorar el comportamiento del *software*.
- Promueve la colaboración de personas.
- Anticipa cambios y los trata automáticamente .

Se puede usar en las empresas cambiantes como las minoristas, donde cada día cambian sus necesidades de acuerdo con la oferta y demanda, en este tipo de desarrollo es probable que el cliente esté pidiendo adecuaciones continuamente, el ciclo de vida de esta metodología es dirigible y fácil de implementar.(7)

1.5.1.4 Desarrollo Manejado por Rasgos (FDD)

FDD (por sus siglas en inglés de *Feature-Driven Development*) se estructura alrededor de la definición de características, que representan la funcionalidad que debe contener el sistema y tiene un alcance lo suficientemente corto como para ser implementado en un par de semanas.(6)

El seguimiento del proceso se realiza mediante un examen de pequeñas funcionalidades descompuestas y funciones valoradas por el cliente. Un rasgo es una función pequeña expresada en la forma <acción> <resultado> <por | para | de | a> <objeto> con los operadores adecuados entre los términos.

Algunas de las características de FDD son:

- No cubre todo el ciclo de vida, sino diseño y construcción.
- Se considera apto para proyectos mayores o de misión crítica.
- Numerosos artefactos para el control del proceso.

En síntesis, FDD es un método de desarrollo de ciclos cortos que se concentra en la fase de diseño y construcción. Durante la primera fase, el modelo global de dominio es elaborado por expertos del dominio y desarrolladores. El modelo de dominio consiste en diagramas de clases con clases, relaciones, métodos y atributos. Los métodos no reflejan conveniencias de programación sino rasgos funcionales.

Se encuentra en FDD un proceso ágil orientado a la funcionalidad del *software* por encima de las tareas, de las cuales ofrece guías mínimas. El proceso sugiere organizar bloques de características para construirlos en forma incremental mediante iteraciones de dos semanas, provee estrategias de planeamiento para el líder de proyecto y fomenta la colaboración mediante la creación de equipos dirigidos por un programador jefe.(8)

1.5.1.5 Programación Extrema

El objetivo de XP es muy simple: la satisfacción del cliente. Esta metodología trata de dar al cliente el *software* que él necesita y en el momento que lo necesita. Por tanto, se debe responder muy rápido a las necesidades del cliente, incluso cuando los cambios sean al final de ciclo de la programación. XP se encarga de potenciar al máximo el trabajo en grupo. Tanto los jefes de proyecto, los clientes y los desarrolladores, son parte del equipo y están involucrados en el desarrollo de *software*.

Se basa en retroalimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. Está definida especialmente para proyectos con requisitos imprecisos y muy cambiantes, donde existe un alto riesgo técnico. Define cuatro variables para proyectos de *software*: coste, tiempo, calidad y ámbito.(5)

XP supone:

- Las personas son clave en los procesos de desarrollo.
- Los programadores son profesionales que no necesitan supervisión.
- Los procesos se aceptan y se acuerdan, no se imponen.
- Desarrolladores y gerentes comparten el liderazgo del proyecto.
- El trabajo de los desarrolladores con las personas que conocen el negocio es regular, no puntual.

Las características esenciales de XP son las siguientes: historias de usuario(HU), roles, proceso y prácticas.

HU: es la técnica utilizada para especificar las necesidades del *software*. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean funcionales o no funcionales. El tratamiento de las HU es muy dinámico y flexible. Cada HU es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

Roles:

- Programador: el programador escribe las pruebas unitarias y produce el código del sistema.
- Cliente: escribe las HU y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las HU y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio.
- Encargado de pruebas: ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.
- Encargado de seguimiento: proporciona realimentación al equipo. Verifica el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, con el objetivo de mejorar futuras estimaciones. Realiza el seguimiento del progreso de cada iteración.
- Entrenador: es responsable del proceso global. Debe proveer guías al equipo de forma que se apliquen las prácticas de XP y se siga el proceso correctamente.
- Consultor: es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto.

- Gestor: es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es la coordinación.

Proceso: el ciclo de desarrollo consiste (a grandes rasgos) en los siguientes pasos:

- El cliente define el valor de negocio a implementar.
- El programador estima el esfuerzo necesario para su implementación.
- El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
- El programador construye ese valor de negocio.
- Vuelve al principio.

El ciclo de vida ideal de XP consiste de seis fases:

- Exploración.
- Planificación de entrega (*Release*).
- Iteraciones.
- Producción.
- Mantenimiento.
- Muerte del Proyecto.

Prácticas: la principal suposición que se realiza en XP es la posibilidad de disminuir la mítica curva exponencial del costo del cambio a lo largo del proyecto, lo suficiente para que el diseño evolutivo funcione.

La mayoría de las prácticas propuestas por XP no son novedosas, sino que en alguna forma ya habían sido propuestas en ingeniería de *software* e incluso demostradas en la práctica. El mérito de XP es integrarlas de una forma efectiva y complementarlas con otras ideas desde la perspectiva del negocio, los valores humanos y el trabajo en equipo.(9)

1.5.1.6 Selección de la metodología

Después de haber estudiado las metodologías de desarrollo de *software* anteriormente expuestas, se llega a la conclusión de que las metodologías ágiles son poco orientadas al documento, exigiendo una cantidad pequeña de documentación para una

tarea dada. De muchas maneras son más bien orientadas al código, siguiendo un camino que dice que la parte importante de la documentación es el código fuente.

La tendencia, de cualquier modo, es hacer cada iteración tan corta como se pueda manejar. Esto proporciona una retroalimentación más frecuente, por lo que se sabe más a menudo en qué punto se encuentra. Se decide seleccionar XP como metodología de desarrollo ya que el plazo de entrega del proyecto es más corto, además las necesidades actuales pueden estar sujetas a futuros cambios y éste es flexible a los mismos administrando las modificaciones de una forma óptima.

Se escoge en gran medida XP porque se diferencia de las demás metodologías que forman este grupo en un aspecto particular: el alto nivel de disciplina de las personas que participan en el proyecto. Propone que la comunicación y la satisfacción del cliente es lo principal y no se hace mucho énfasis en la documentación, es más importante definir las necesidades y las pruebas de calidad. Está diseñada para el trabajo de equipos pequeños y se apoya en las pruebas. Aunque todos los procesos estudiados mencionan la comprobación, la mayoría no lo hace con el rigor requerido. Sin embargo, XP pone la comprobación como el fundamento del desarrollo, con cada programador escribiendo pruebas cuando escriben su código de producción. Las pruebas se integran en el proceso de integración continua y construcción, lo que rinde una plataforma altamente estable para el desarrollo futuro.

1.5.2 Herramientas de modelado

Las herramientas CASE (*Computer Aided Software Engineering* en inglés) nacen para auxiliar a los desarrolladores de sistemas, lo que permite apoyo computarizado en todo o en parte del ciclo de vida de desarrollo de un sistema de *software*, tal es el caso de las herramientas CASE con soporte UML.(10)

1.5.2.1 Rational Rose

Rational Rose Data Modeler es una herramienta de modelado visual que posibilita que los diseñadores de bases de datos, analistas, arquitectos, desarrolladores y todos los demás miembros del equipo de desarrollo trabajen juntos, capturando y compartiendo los requerimientos de negocio y dándole seguimiento a medida que cambian a través del proceso. Proporciona una realización de la metodología ER (Entidad relación) usando la notación UML para unificar a los diseñadores de bases de datos con el equipo de desarrollo de *software*. Con UML, el diseñador de bases de datos puede capturar información tal como restricciones, *triggers* e índices directamente en el diagrama en lugar de tener que representarlos con propiedades ocultas por fuera.

Otorga la libertad de pasar del objeto al modelo de datos y obtener ventaja de los tipos de transformaciones básicas como las relaciones muchos-a-muchos. Esta herramienta brinda una forma intuitiva de visualizar la arquitectura de las bases de datos y la forma en que se vinculan con la aplicación.(11)

Propone la utilización de cuatro tipos de modelo para realizar el diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico. Permite crear y refinar estas vistas creando un modelo completo que representa el dominio del problema y el sistema de *software*.(12)

Alguna de las ventajas que brinda son las siguientes:

- Trabajo en grupo.
- Generación de código.
- Generación de documentos automáticamente.
- Ingeniería inversa.

1.5.2.2 Visual Paradigm

Visual Paradigm es una herramienta CASE profesional que soporta el ciclo de vida completo de desarrollo de *software*: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Esta herramienta tiene unas características gráficas muy cómodas que facilitan la realización de los diagramas de modelado que sigue el estándar de UML, tales como diagramas de clase, casos de uso, comunicación, secuencia, estado, actividad y componentes entre otros.(13)

Algunas de sus principales características son:

- Producto de calidad.
- Soporta aplicaciones Web.
- Varios idiomas.
- Generación de código para Java y exportación como HTML.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo, lo que facilita la comunicación.
- Capacidades de ingeniería directa (versión profesional) e inversa.

- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- Disponibilidad de múltiples versiones.
- Disponibilidad de integrarse con los principales IDEs.
- Disponibilidad de múltiples plataformas.
- Ofrece una serie de facilidades para generar informes que permiten documentar el proyecto.(14)

1.5.2.3 Lenguaje Unificado de Modelado (UML)

Es un lenguaje para especificar, construir, visualizar y documentar los artefactos de un sistema de *software* orientado a objetos. No pretende definir un modelo estándar de desarrollo, sino únicamente un lenguaje de modelado. Los procesos son diferentes según los distintos dominios de trabajo, no puede ser el mismo proceso para crear una aplicación en tiempo real que el proceso de desarrollo de una aplicación orientada a gestión, por poner un ejemplo. Con la creación de UML se persigue obtener un lenguaje que sea capaz de abstraer cualquier tipo de sistema mediante los diagramas, que son representaciones gráficas que contienen la información relevante del sistema.(15)

Los bloques básicos de construcción de UML son tres, los elementos, las relaciones y los diagramas.

- Los elementos son abstracciones que actúan como unidades básicas de construcción. Hay cuatro tipos, los estructurales (clase, clase activa, interfaz, colaboración, caso de uso, componente y nodo), los de comportamiento (interacción, máquinas de estado), los de agrupación y los de notación. En cuanto a los elementos estructurales, son las partes estáticas de los modelos y representan aspectos conceptuales o materiales. Otro caso es el de los elementos de comportamiento, éstos constituyen las partes dinámicas de los modelos y representan comportamientos en el tiempo y el espacio. Las divisiones en que se puede fraccionar un modelo, la establecen los elementos de agrupación, ellos son la parte organizativa de UML. Sólo hay un elemento de agrupación (el paquete) y se emplea para organizar otros elementos en grupos. La parte explicativa de UML la constituyen los elementos de notación, ellos son comentarios que pueden describir textualmente cualquier aspecto de un modelo. Hay un sólo elemento de notación principal, la nota.

- Las relaciones son abstracciones que actúan como unión entre los distintos elementos. Hay cuatro tipos: la dependencia, la asociación, la generalización y la realización.
- Los diagramas son la disposición de un conjunto de elementos, que representan el sistema modelado desde diferentes perspectivas. UML tiene nueve diagramas fundamentales, agrupados en dos grandes grupos, uno para modelar la estructura estática del sistema y otro para modelar el comportamiento dinámico.(16)

1.5.2.4 Selección de la herramienta de modelado

Después de haber analizado las características de las herramientas de modelado expuestas anteriormente, se considera que ambas facilitan el trabajo durante el desarrollo de un producto de *software*, garantizando su calidad final. Por tanto, ambas son herramientas potentes y fáciles de usar. Para llevar a cabo el desarrollo del portal Web para la Escuela Cubana de Boxeo se utilizará Visual Paradigm 6.4, el cual soporta aplicaciones Web y posee características gráficas muy cómodas.

1.5.3 Principales lenguajes de programación

Un lenguaje de programación es un conjunto de símbolos junto a un conjunto de reglas para combinar dichos símbolos, se usa para expresar programas. Consta de un léxico (conjunto de símbolos permitidos o vocabulario), una sintaxis (reglas que indican cómo realizar las construcciones del lenguaje) y una semántica (reglas que permiten determinar el significado de cualquier construcción del lenguaje). Es una técnica de comunicación que se utiliza para controlar el comportamiento de una máquina (usualmente un computador).(17)

Actualmente existen diferentes lenguajes de programación para desarrollar en la Web, estos han ido surgiendo debido a las tendencias y necesidades de las plataformas. A medida que pasa el tiempo, las tecnologías han ido desarrollándose y surgen nuevos problemas a los que dar solución. Esto dió lugar al desarrollo de lenguajes de programación dinámicos para la Web que permiten interactuar con los usuarios.

1.5.3.1 Active Server Pages

Active Server Pages (ASP por sus siglas) es un lenguaje de programación de *Microsoft* del lado del servidor utilizado para la generación de páginas Web dinámicas. La programación de sitios Web mediante el lenguaje ASP se ve facilitada gracias a diferentes objetos incorporados, correspondiéndose cada uno

de ellos con un grupo de funcionalidades habitualmente utilizadas en la construcción de sitios Web dinámicos, combinando HTML con un lenguaje de secuencia de comandos o lenguaje *script*.

Otra característica de ASP es que permite integrar componentes *ActiveX* en las páginas, para de esta forma acceder a recursos residentes en el propio servidor. Componentes que, por ejemplo, pueden utilizarse para la comunicación con una base de datos, aportando acceso a más alto nivel.

El lenguaje de programación ASP, ofrece las siguientes ventajas:

- Separar el código HTML del ASP.
- Mayor facilidad para realizar cambios.
- Fácil instalación.
- Funcionamiento y mayor protección del código.(18)

1.5.3.2 Python

Python es un lenguaje de *scripting* independiente de plataforma y orientado a objetos, preparado para realizar cualquier tipo de programa, desde aplicaciones *Windows* a servidores de red o incluso, páginas Web. Es un lenguaje interpretado, lo que significa que no se necesita compilar el código fuente para poder ejecutarlo, ofreciendo ventajas como la rapidez de desarrollo e inconvenientes como una menor velocidad.

En los últimos años el lenguaje se ha hecho muy popular, gracias a varias razones como:

- La cantidad de librerías que contiene, tipos de datos y funciones incorporadas en el propio lenguaje que ayudan a realizar muchas tareas habituales sin necesidad de tener que programarlas desde cero.
- La sencillez y velocidad con la que se crean los programas. Un programa en Python puede tener de 3 a 5 líneas de código menos que su equivalente en Java o C.
- La cantidad de plataformas en las que se puede desarrollar, como Unix, Windows, OS/2, Mac, Amiga y otras.
- Es gratuito, incluso para propósitos empresariales.

Características del lenguaje:

- Propósito general: se puede crear todo tipo de programas. No es un lenguaje creado específicamente para la Web, aunque entre sus posibilidades sí se encuentra el desarrollo de páginas.
- Multiplataforma: hay versiones disponibles de Python en muchos sistemas informáticos distintos. Originalmente se desarrolló para Unix, aunque cualquier sistema es compatible con el lenguaje siempre y cuando exista un intérprete programado para él.
- Interpretado: quiere decir que no se debe compilar el código antes de su ejecución. En realidad sí que se realiza una compilación, pero se realiza de manera transparente para el programador. Hay casos en los que cuando se ejecuta por primera vez un código, se producen unos *bytecodes* que se guardan en el sistema y sirven para acelerar la compilación implícita que realiza el intérprete cada vez que se ejecuta el mismo código.
- Interactivo: dispone de un intérprete por línea de comandos en el que se pueden introducir sentencias. Cada sentencia se ejecuta y produce un resultado visible, que puede ayudar a entender mejor el lenguaje y probar los resultados de la ejecución de segmentos de código rápidamente.
- Es un lenguaje de programación multiparadigma: lo cual fuerza a que los programadores adopten un estilo de programación particular, ya sea orientada a objetos, estructurada, funcional u orientada a aspectos.
- Funciones y librerías: dispone de muchas funciones incorporadas en el propio lenguaje para el tratamiento de cadenas, números, archivos y otros. Existen muchas librerías que se pueden importar en los programas para tratar temas específicos como la programación de ventanas, sistemas en red o cosas tan interesantes como crear archivos comprimidos.
- Sintaxis clara: tiene una sintaxis muy visual, debido a que presenta una notación indentada (con márgenes) de obligado cumplimiento. En muchos lenguajes, para separar segmentos de código, se utilizan elementos como las llaves o las palabras clave *begin* y *end*. Para separar las porciones de código en Python se debe tabular hacia dentro, colocando un margen al código que irá dentro de una función o un bucle. Esto ayuda a que todos los programadores adopten una misma notación y que los programas de cualquier persona tengan un aspecto similar.(19)

1.5.3.3 PHP

Es uno de los lenguajes del lado del servidor más extendidos en la Web. PHP es un acrónimo recursivo que significa *Hypertext Pre-processor*, es interpretado, de propósito general y ampliamente usado, está diseñado especialmente para desarrollo Web y puede ser insertado dentro de código HTML. Puede ser desplegado en la mayoría de los servidores Web y en casi todos los sistemas operativos y plataformas sin costo alguno. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas.

Ventajas de PHP:

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- Lee y manipula datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
- Capacidad de expandir su potencial utilizando una enorme cantidad de módulos (llamados extensiones).
- Posee una amplia documentación en su página oficial, donde destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos (POO).
- Permite crear formularios para la Web.
- Presenta una biblioteca nativa de funciones sumamente amplia.
- No requiere definición de tipos de variables ni manejo detallado del bajo nivel. (20)

La mayor fuerza de PHP es que está preparado para soportar accesos a muchos tipos de bases de datos como:

- Adabas D.
- dBase.
- Empress.

- FiclePro.
- informix.
- InterBase.
- Solid.
- Sybase.
- Velocis.
- Unix dbm.
- mSQL.
- MySQL.
- Oracle.
- PostgreSQL.(21)

1.5.3.4 Selección del lenguaje de programación

Analizando las características y ventajas de los lenguajes de programación Web antes expuestos se llega a la conclusión de que estos lenguajes pueden ser usados para desarrollar portales Web de óptima calidad. En el caso de ASP a pesar de que cuenta con códigos prediseñados que ofrecen mayor facilidad a la hora de diseñar una página Web, sólo funciona en *Windows* y es de licencia propietaria, por lo que su costo es muy elevado.

A diferencia de ASP, Python es multiplataforma por lo que se puede desarrollar en diferentes sistemas operativos, además es gratuito y de libre distribución. Python también cuenta con una sintaxis clara lo que ayuda a que todos los programadores adopten una misma notación y que los programas de cualquier persona tengan un aspecto similar, es sencillo y fácil de programar. Al ser un lenguaje interpretado es extremadamente lento por lo que no es el más viable para el desarrollo del portal.

Por lo anteriormente analizado se elige al lenguaje PHP 5.0 para el desarrollo del portal Web para la Escuela Cubana de Boxeo ya que es muy fácil de aprender, se caracteriza por ser un lenguaje de programación muy rápido, es multiplataforma y tiene capacidad de conexión con la mayoría de los manejadores de bases de datos incluyendo el PostgreSQL. También es libre, lo que representa una alternativa de fácil acceso para todos. La aplicación que se va a desarrollar no es grande y PHP es adecuado para pequeñas aplicaciones Web dinámicas.

1.5.4 Entorno de Desarrollo Integrado

Un Entorno de Desarrollo Integrado (IDE, por sus siglas en inglés, *Integrated Development Environment*) permite editar, compilar, ejecutar y depurar programas de una forma cómoda y ágil.(22)

Los IDEs incluyen un número enorme de ventajas y herramientas como el coloreado de la sintaxis y el autocompletado de código que ayudan al desarrollador a programar rápidamente. Otro objetivo de éstos, es indicar errores tipográficos de forma rápida resaltando los errores de sintaxis y haciendo verificaciones en el código justo cuando se introduce. Además, un buen IDE ayuda en la organización de los ficheros del código fuente y efectúa un seguimiento de los símbolos y funciones que están definidos en ellos.(23)

1.5.4.1 PHP Designer

Es una herramienta útil y gratis para codificar PHP, tanto para principiantes como profesionales. Está pensado para acelerar y optimizar el proceso de edición, eliminación de errores, análisis y publicación de aplicaciones, además de sitios Web proporcionados por PHP y otros lenguajes Web.

Ofrece toda una serie de asistentes y diálogos integrados que facilitan en todo momento las tareas, además tiene acceso directo a librerías de código o *scripts* de uso habitual, utilidades diversas y herramientas, también cuenta con una interfaz de diseño sencilla y elegante. Cuenta con cliente de FTP y navegador de ficheros integrado, herramientas de corrección y autocompletado, búsqueda integrada en Google, soporte para proyectos y usa un práctico esquema de color para la sintaxis del código fuente facilitando la programación.

PHP Designer soporta: PHP, HTML, XHTML, CSS, Java, Perl, JavaScript, VB, C# y SQL.

1.5.4.2 Zend Studio

Es un IDE propietario, compatible con las plataformas Linux, MAC y *Windows*. Incluye todos los componentes necesarios durante el ciclo de vida de una aplicación en PHP. Cubre todas las típicas características de un IDE más un conjunto de características especiales para el entorno PHP, con el objetivo de ayudar a la productividad del desarrollador de PHP. Posee soporte para Web Services, PHP4, PHP5 y SQL. Zend Studio permite agilizar el desarrollo Web y simplificar proyectos complejos.

Algunas de sus características especiales son:

- SOAP: generación del WSDL a partir de la definición de clases PHP (incluyendo la evaluación de comentarios PHPDoc).
- SOAP: autocompletado de código para clientes SOAP basado en un archivo WSDL (local o remoto).
- SCM: soporte integrado para Subversion y CVS.
- DB: visor de la estructura de base de datos y de contenidos, cliente SQL para diferentes tipos de bases de datos.
- Editor: coloreo de la sintaxis. Coloreo de errores.
- Editor: autocompletado de código para funciones PHP y código propio basado en comentarios PHPDoc.
- Editor: analizador de código para encontrar problemas típicos en código PHP.
- Editor: plantillas de código PHP.
- Depurador: depuración local o remota.
- Depurador: *profiling* local o remoto.
- Depurador: barra del navegador que permite empezar el depurado de errores en cualquier lugar de su aplicación.(24)

1.5.4.3 Adobe Dreamweaver

Dreamweaver es una herramienta propietaria utilizada para la creación de sitios Web. Fue desarrollada para los grupos de usuarios principiantes y avanzados.

Características: plataformas *Windows* y *MAC*, disponible en varios idiomas, completamiento, coloreado en las sintaxis del código, lenguajes: *HTML*, *Coldfusion*, *PHP*, *ASP VBScript*, *ASP.NET C#*, *Javascript*, *XML*, *XSLT*, *CSS*, *ActionScript*, *JSP*, entre otros. Incluye *Framework* para *AJAX*, administrador *CSS*, compatibilidad con dispositivos móviles, integración con *Adobe Photoshop* y *Adobe Fireworks*.

Su principal ventaja es que genera código organizado, permite la creación de páginas en diferentes lenguajes, creación de plantillas y definición de sitio. Requiere de conexión a Internet para registrar el producto y costo de la licencia.(23)

1.5.4.4 Selección del Entorno de Desarrollo Integrado

Es evidente que los IDEs estudiados cumplen con las necesidades que requiere un usuario, ofrecen ayudas y diálogos que permiten desarrollar el proyecto con mucha eficiencia, además presentan una interfaz de diseño sencilla, facilidades de corrección de código, así como buen completamiento de código. Se considera como una desventaja, en el caso del PHP Designer, el cambio de ventana para el modo de edición y el modo de diseño. El Zend Studio ofrece más facilidades que el Dreamweaver para trabajar con el código, ya que permite realizar *debug* para encontrar dónde están los errores.

Por todo lo anteriormente mencionado se escoge el Zend Studio 7.1.0 como entorno de desarrollo para desarrollar el portal Web de la Escuela Cubana de Boxeo, aunque es un *software* propietario, de los dos IDEs estudiados es uno de los que con más experiencia de trabajo se cuenta en la Universidad y es la herramienta establecida por el proyecto. Además, posee los componentes necesarios durante el ciclo de vida de una aplicación PHP.

1.5.5 Sistema Gestor de Bases de Datos (SGBD)

Los sistemas de gestión de bases de datos, son aplicaciones que permiten a los usuarios definir, crear y mantener la base de datos, proporcionando un acceso controlado a la misma. Un SGBD es la aplicación que interactúa con los usuarios de programas de aplicación y la base de datos.

Un SGBD debe permitir:

- Definir una base de datos: especificar tipos, estructuras y restricciones de datos.
- Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD.
- Manipular la base de datos: realizar consultas, actualizarla y generar informes.(25)

Los componentes de un SGBD son:

Componentes de procesamiento de sentencias DML/DDL:

- Compilador DML: traduce las sentencias DML en instrucciones a bajo nivel para ser ejecutadas por el motor de ejecución del gestor. Suelen usar algoritmos para optimizar las sentencias DML y hacerlas más eficientes.

- Precompilador DML: traduce las sentencias DML insertadas en cierto programa en sentencias propias de su lenguaje de programación.
- Intérprete DDL: interpreta las instrucciones DDL y genera los metadatos necesarios en el diccionario de datos.
- Motor de ejecución: encargado de que se ejecuten las sentencias ya compiladas, utiliza los componentes de gestión de almacenamiento.

Componentes de gestión de almacenamiento: se encargan del acceso directo a los datos solicitados por el motor de ejecución.

- Gestor de autorización e integridad: comprueba que el usuario tiene los permisos pertinentes y que no se violan las restricciones de integridad.
- Gestor de transacciones: asegura un estado consistente de la base de datos, aunque se produzcan fallos del sistema o existan transacciones concurrentes.
- Gestor de archivos: gestiona el espacio en disco a través de archivos.
- Gestor de memoria intermedia: gestiona los datos que deben traerse o llevarse de disco a memoria principal.

Otros componentes almacenados en disco: archivos de datos, diccionario de datos, índices, datos estadísticos y otros.(24)

1.5.5.1 MySQL

Es un sistema de gestión de bases de datos relacional, fue creado por la empresa sueca MySQL AB, la cual tiene el *copyright* del código fuente del servidor SQL, así como la marca.

MySQL es un *software* de código abierto, licenciado bajo la GPL de GNU, aunque MySQL AB distribuye una versión comercial, en lo único que se diferencia de la versión libre es en el soporte técnico que se ofrece y la posibilidad de integrar este gestor en un *software* propietario, ya que de otra manera, se violaría la licencia GPL.

El lenguaje de programación que utiliza es *Structured Query Language* (SQL) que fue desarrollado por IBM en 1981 y desde entonces es utilizado de forma generalizada en las bases de datos relacionales.

En las últimas versiones se pueden destacar las siguientes características:

- El principal objetivo de MySQL es velocidad y robustez.

- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Cada base de datos cuenta con 3 archivos: uno de estructura, uno de datos y uno de índice, soportando hasta 32 índices por tabla.
- Aprovecha la potencia de sistemas multiproceso gracias a su implementación multihilo.
- Flexible sistema de contraseñas (*passwords*) y gestión de usuarios, con un alto nivel de seguridad en los datos.
- El servidor soporta mensajes de error en distintos lenguajes.

Ventajas:

- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Facilidad de configuración e instalación.
- Soporta gran variedad de sistemas operativos.
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Conectividad y seguridad.(26)

1.5.5.2 Oracle

Oracle es básicamente una herramienta cliente/servidor para la gestión de base de datos, es un producto muy vendido a nivel mundial, aunque por la gran potencia que tiene y su elevado precio conlleva a que generalmente solo se vea en empresas muy grandes y multinacionales. Corre en computadoras personales (PC), microcomputadoras, *mainframes* y computadoras con procesamiento paralelo masivo. Soporta unos 17 idiomas, corre automáticamente en más de 80 arquitecturas de *hardware* y *software* distintas sin tener la necesidad de cambiar ninguna línea de código. Esto es posible porque más del 80% de los códigos

internos de Oracle son iguales a los establecidos en todas las plataformas de sistemas operativos (*Windows, Unix, Macintosh y Mainframes*).

Algunas de sus mejoras son que soporta bases de datos de todos los tamaños, desde severas cantidades de *bytes* y *gigabytes*, provee salvar con seguridad de error lo visto en el monitor y la información de acceso y uso.

El soporte mundial y la experiencia de sus clientes hacen que sea una sólida elección para aquellas organizaciones que buscan el acceso a una base amplia de experiencia.(27)

1.5.5.3 PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. Fue el pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual, incluidos más tarde en otros sistemas de gestión comercial. A pesar de esto, no es un sistema de gestión de bases de datos puramente orientado a objetos.

Es un sistema de gestión de bases de datos distribuido bajo licencia BSD (*Berkeley Software Distribution*) y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones es tan potente como otras bases de datos comerciales. 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.

Es un sistema estable, de alto rendimiento y gran flexibilidad. Disponible para Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows. Presenta APIs para programar en C/C++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, PHP, Lisp, Scheme, Qt y muchos otros. Soporta consultas complejas, incluyendo subselects, integridad referencial (*Foreign Keys*), *triggers*, vistas (*Views*), integridad transaccional (ACID) y control de versionado concurrente (MVCC).(28)

1.5.5.4 Selección del Sistema Gestor de Bases de Datos

Después de analizar los SGBD anteriormente expuestos se llega a la conclusión de que Oracle a pesar de su gran potencia y de ser multiplataforma no es la opción más viable para el proyecto en cuestión, debido a que en contratos y precios sus

consumidores continúan presentando problemas por los elevados precios que posee. Otro problema a la hora de trabajar con este gestor es saber qué funcionalidades se cobran como parte del SGBD y cuales como partes opcionales.

PostgreSQL y MySQL son los SGBD de mayor representación dentro del mundo de *software* libre, principalmente por su alta integración con herramientas de programación Web como PHP o JAVA. Tienen muy buen soporte Web, actualmente es para lo que más se les utiliza. Si se piensa desarrollar una aplicación Web, donde el factor principal es la velocidad, se puede utilizar MySQL, ya que éste es más rápido debido a que no hace ningún tipo de control o auditoría. De acuerdo a los objetivos de desarrollo del equipo se desea realizar una aplicación a la medida donde predomine la estabilidad, integridad y seguridad del sistema, por lo que la orientación es PostgreSQL teniendo en cuenta que está orientado al manejo de grandes volúmenes de información.

Por estas razones se escoge PostgreSQL 8.2 como Sistema Gestor de Base de Datos para el desarrollo del portal de la Escuela Cubana de Boxeo, además porque el costo es su principal ventaja y permite observar el código fuente para entender qué está sucediendo.

1.5.6 Sistemas de Gestión de Contenidos

Un Sistema de Gestión de Contenido (CMS siglas por el nombre en inglés *Content Management System*) permite la creación y administración de contenidos principalmente en páginas Web. Consiste en una interfaz que permite manejar de manera independiente el contenido y el diseño. Así, es posible darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido nuevamente, además de permitir fácil y controladamente la publicación en el sitio por varios editores.

Un CMS permite que usuarios sin conocimientos técnicos ni de diseño de páginas Web puedan actualizar sus sitios, añadiendo secciones, noticias, páginas o productos con relativa facilidad.(29)

Existen multitud de CMS comerciales y de código libre, entre los que se destacan algunos como Drupal y Joomla, que son de los más usados tanto en la Universidad de las Ciencias Informáticas como en el resto del mundo.

1.5.6.1 Drupal

Es un sistema de gestión de contenido modular y configurable. Permite publicar artículos, imágenes u otros archivos y servicios añadidos como foros, encuestas, votaciones, *blogs*, administración de usuarios y permisos.

Desarrolla y mantiene una cantidad activa de usuarios. Destaca por la calidad de su código y de las páginas que de él pueden ser generadas, el respeto de los estándares de la Web y un énfasis especial en funcionalidad y consistencia de todo el sistema. Es multiplataforma, multilenguaje, modular, extensible y flexible.

Drupal es un *software* libre utilizado para la administración de contenido dentro de un portal Web. Entre los tipos de proyectos para los que Drupal es usado se encuentran: portales Web de la comunidad, sitios de discusión, sitios Web corporativos, aplicaciones de intranet, sitios Web personales o *blogs*, sitios aficionados, aplicaciones de comercio electrónico, directorio de recursos y sitios de redes sociales.

En cuanto a base de datos, este posee independencia de la misma, es decir los módulos son escritos en una forma que es independiente del sistema de base de datos en uso. Aunque la mayor parte de las instalaciones de Drupal utilizan MySQL, existen otras opciones. Drupal incorpora una capa de abstracción de base de datos que actualmente está implementada y mantenida para MySQL y PostgreSQL, aunque permite incorporar fácilmente soporte para otras bases de datos.

Algunas de sus características son:

- Posee una ayuda en línea y páginas de ayuda para los módulos del núcleo, tanto para usuarios como para administradores.
- Todo el contenido es indexado en tiempo real y se puede consultar en cualquier momento.
- Contiene un código abierto que está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario de otros sistemas de *blogs*, es posible extender o adaptar Drupal según las necesidades.
- La comunidad brinda muchos módulos que proporcionan diferentes funciones como página de categorías, autenticación mediante *jabber*, mensajes privados, *bookmarks* y otros.

- Tiene a su favor un entorno de personalización que está implementado en el núcleo. Tanto el contenido como la presentación pueden ser individualizados de acuerdo las preferencias del usuario.
- Posee urls amigables donde usa el *mod_rewrite* de Apache para crear urls que son manejables por los usuarios y los motores de búsquedas.
- El sistema de control de versiones de Drupal permite seguir totalmente las sucesivas actualizaciones del contenido que se ha cambiado, la hora, la fecha y quién lo ha cambiado. También permite mantener comentarios sobre los sucesivos cambios o deshacer los cambios recuperando una versión anterior.
- Ha sido diseñado desde el principio para ser multiplataforma. Puede funcionar con Apache o Microsoft IIS como servidor Web y en sistemas como Linux, BSD, Solaris, *Windows* y Mac OS X. Por otro lado, al estar implementado en PHP, es totalmente portable.
- Está pensado para una audiencia internacional y proporciona opciones para crear un portal multilingüe. Todo el texto puede ser fácilmente traducido utilizando una interfaz Web, importando traducciones existentes o integrando otras herramientas de traducción como GNU *gettext*.
- Proporciona un potente modelo de comentarios enlazados que posibilita seguir y participar fácilmente en la discusión sobre el comentario publicado. Los comentarios son jerárquicos, como en un grupo de noticias o un foro. También posee encuestas, foros de discusión y un libro colaborativo.
- Incorpora un mecanismo de control de congestión que permite habilitar y deshabilitar determinados módulos o bloques dependiendo de la carga del servidor. Este mecanismo es totalmente configurable y ajustable. También posee un mecanismo de caché que elimina consultas a la base de datos incrementando el rendimiento y reduciendo la carga del servidor.(30)

1.5.6.2 Joomla

Joomla es uno de los gestores de contenido más populares y versátiles que existe. Una ventaja muy importante es que tiene código abierto, además es gratis. Muchas compañías de alojamiento de páginas Web lo tienen integrado en su panel de control, lo que facilita mucho su instalación, incluso para personas con poco conocimiento en este tema. Es un sistema de administración de contenidos de código abierto construido con PHP bajo una licencia GPL.

Sus características principales son las siguientes:

- Organización del sitio Web: está preparado para organizar eficientemente los contenidos de un sitio en secciones y categorías, lo que facilita la navegabilidad para los usuarios y permite crear una estructura sólida, ordenada y sencilla para los administradores. Desde el panel administrador de Joomla se podrá crear, editar y borrar las secciones y categorías de un sitio de la manera más conveniente.
- Publicación de contenidos: se podrán crear páginas ilimitadas y editarlas desde un sencillo editor que permite formatear los textos con los estilos e imágenes deseados. Los contenidos son totalmente editables y modificables.
- Escalabilidad e implementación de nuevas funcionalidades: ofrece la posibilidad de instalar, desinstalar y administrar componentes y módulos, los que agregarán servicios de valor a los visitantes del sitio Web, por ejemplo: galerías de imágenes, foros, *newsletters*, clasificados y otros.
- Administración de usuarios: permite almacenar datos de usuarios registrados y también enviar *E-mails* masivos a todos los usuarios. La administración de usuarios es jerárquica y los distintos grupos de usuarios poseen diferentes niveles de permisos dentro de la gestión y administración del sitio.
- Diseño y aspecto estético del sitio: es posible cambiar todo el aspecto del sitio Web tan sólo con un par de *clicks*, gracias al sistema de *templates* (plantillas) que utiliza Joomla.
- Navegación y menú: totalmente editables desde el panel administrador de Joomla
- Administrador de imágenes: posee una herramienta para subir imágenes al servidor y usarlas en todo el sitio.
- Disposición de módulos modificables: la posición de los módulos puede acomodarse como se prefiera.
- Encuestas: posee un sistema de votaciones y encuestas dinámicas con resultados en barras porcentuales.(31)

1.5.6.3 Mambo

Mambo es un sistema gestor de contenidos Web totalmente libre que posee un manejo sencillo y fácil. Trabaja con un programado en lenguaje PHP y base de datos MySQL. Posee múltiples funcionalidades y es totalmente interactivo. Tan eficiente es que bastaría de unas horas para construir un completo sitio Web interactivo, sin necesidad de tener conocimientos avanzados en el tema de implementación y diseño Web. Está orientado a brindar un sistema sencillo de trabajo y existe una enorme comunidad que trabaja en pro del proyecto, que aportan constantemente nuevas herramientas y utilidades.

Sus principales características son:

- Base de datos para los estándares PHP/MySQL. Estos estándares están establecidos por defecto en las versiones GNU/Linux, así que trabajar en Mambo en uno de estos sistemas operativos no será mayor problema; en el caso de Windows se deberán instalar los programas relacionados a cada uno de los estándares.
- Módulo de seguridad multinivel para usuarios y administradores. Uno de los pilares es la correcta administración de la seguridad y los privilegios.
- Noticias, productos o secciones totalmente editables y configurables. Se puede colocar contenido interactivo para que el visitante del sitio se sienta más atraído con la Web.
- Plantillas totalmente configurables, incluyendo menú central y bloques a izquierda y derecha. La comunidad de Mambo, constantemente está desarrollando plantillas, cuyo diseño puede ser seleccionado sin restricción alguna.
- Foros dinámicos y encuestas con vista de resultados, todo sitio Web interactivo debe contar con alguno ellos.
- Soporta Linux, FreeBSD, MacOSX *server*, Solaris, AIX, SCO, WinNT, Win2K. Una gran cantidad de sistemas operativos es compatible totalmente con Mambo, esto hace del programa una alternativa multiplataforma.(32)

1.5.6.4 Selección del CMS a utilizar

Cualquiera de los CMS estudiados serviría para el desarrollo de la aplicación, pues todos ofrecen una serie de funcionalidades que facilitarían el trabajo con el portal. Joomla es la evolución de Mambo pero con un nombre distintivo dado por sus

desarrolladores. Se usa Joomla si se quiere tener un sitio agradable de forma rápida y se puede afrontar tener un sistema más lento, una categorización de contenido rígida y unas opciones de configuración y de diseño limitadas. Mambo y Joomla están concebidos para utilizar como SGBD a MySQL. Por tanto, se decide utilizar el CMS Drupal 6.15 para el desarrollo del portal Web de la Escuela Cubana de Boxeo ya que se necesita alto rendimiento, escalabilidad, buen gestor de contenido y una flexibilidad de diseño significativa. Drupal tiene gran cantidad de módulos y ayudas para éstos, están disponibles en el sitio de la comunidad, además es independiente de la base de datos.

1.6 Conclusiones

Se hizo un estudio de sistemas similares que brindan información deportiva, obteniendo propuestas de funcionalidades para la solución. Una vez estudiadas las tendencias de las tecnologías y las herramientas más usadas en el campo de la informática, se decidió desarrollar el portal Web para la Escuela Cubana de Boxeo guiado por la metodología XP, usando como lenguaje de programación PHP 5.0 y como IDE el Zend Studio 7.1.0, utilizando como SGBD el PostgreSQL 8.2 y como CMS el Drupal 6.15 para facilitar el trabajo con la Web.

2. DESARROLLO DE LA SOLUCIÓN

2.1 Introducción

El proceso de desarrollo de *software* lleva asociado un marcado énfasis en el control, mediante una rigurosa definición de roles, actividades y artefactos, incluyendo modelado y documentación detallada. Por estar especialmente orientadas para proyectos pequeños, las metodologías ágiles constituyen una solución para ese entorno, aportando una elevada simplificación que a pesar de ello no renuncia a las prácticas esenciales que aseguran la calidad del producto. Las tres primeras fases definidas en la metodología XP son: exploración, planificación e iteraciones. En éstas se abordan temas relacionados con las necesidades del *software* y la planificación del proyecto. También se hace la descripción del flujo actual de las actividades para determinar los objetos que serán automatizados, conformando así una propuesta del sistema. Estas etapas definen en que momento se dará solución a cada una de las necesidades del cliente, además se constituye el Plan de Entrega.

2.2 Flujo actual de actividades en la Escuela Cubana de Boxeo

Actualmente la gestión de la información deportiva del boxeo en la Escuela Cubana de Boxeo es muy tediosa y engorrosa, ya que la misma se realiza de forma manual. No existen mecanismos para divulgar la información referente al boxeo cubano, por lo que muchas veces se desconoce sobre los grandes logros obtenidos por el mismo. De igual manera no existen medios para recopilar la opinión de los seguidores de este deporte en Cuba ni espacios donde se pueda debatir sobre temas del mismo, además no se puede consultar el ranking de los boxeadores y los resultados alcanzados en las competencias. Lo anteriormente planteado constituye un impedimento para la rapidez y eficiencia en la manipulación de los datos referentes a este deporte.

El desarrollo de las actividades se hace de la siguiente manera:

Si algún interesado o aficionado desea conocer sobre la historia del boxeo cubano o los resultados alcanzados por el país en eventos competitivos, debe acudir a varias bibliografías (periódicos, revistas y libros) que pueden deteriorarse y ocurrir pérdidas de datos. También puede dirigirse a instituciones que manejen esta información (dentro de ellas la Escuela Cubana de Boxeo) para encontrar los datos que desee, pero este proceso se torna tedioso para el aficionado ya que en muchas ocasiones no se encuentra la persona capacitada para ofrecer la información solicitada. El mecanismo es similar si se decide investigar sobre la trayectoria de los boxeadores o integrantes del colectivo técnico. Todo esto puede resultar un poco engorroso y se

corre el riesgo de que los datos consultados estén desactualizados y no se pueda completar la investigación.

Uno de los medios utilizados para anunciar y divulgar las investigaciones realizadas por los especialistas del deporte es mediante conferencias impartidas, lo cual no garantiza que la información esté disponible para todos los que deseen utilizarla.

Cuba no cuenta con un espacio que pueda ser consultado en cualquier momento para publicar las fechas en que se realizarán los eventos nacionales e internacionales. El calendario de los eventos competitivos que serán efectuados durante el año es conocido por los entrenadores a través de los organismos superiores (Comisión Nacional), por otra parte los eventos internacionales son dados a conocer por la Asociación Internacional de Boxeo Amateur (AIBA) a través de los medios de comunicación masivos. Debido a estas razones muchas veces existe desconocimiento por parte de los aficionados de las fechas estipuladas para los eventos importantes.

Las entrevistas realizadas a los atletas, las imágenes y los videos obtenidos durante las competencias o entrenamientos son divulgados por los medios de difusión (radio, televisión y prensa) en ocasiones especiales, esto obstaculiza la idea de que puedan ser utilizados como material de consulta por el personal que lo requiera. Cuando son transmitidas por la televisión los aficionados de otros países no pueden acceder al canal Tele Rebelde que es por donde normalmente se transmiten los eventos deportivos. Otro de los medios empleados es la prensa, pero se publican las noticias resumidas, sin entrar en muchos detalles y muchas veces no se satisfacen los intereses de los aficionados.

2.3 Propuesta del Sistema

El desarrollo del presente trabajo se proyecta a la implementación de un sistema que permita manejar información sobre el boxeo cubano. Entre la información que se quiere manejar se encuentran videos, imágenes, historia y desempeño de este deporte desde sus inicios. Además se quiere mostrar también la ficha técnica de cada uno de los atletas e integrantes del colectivo técnico, entrevistas realizadas a las atletas de alto rendimiento e investigaciones realizadas por los especialistas.

Se publicarán las reglas del boxeo amateur así como los estatutos por los que se rige la Escuela Cubana de Boxeo, además se mostrará el ranking de los boxeadores organizados por división, el código de ética y los 10 mandamientos del boxeador. También se mostrarán los resultados alcanzados en los diferentes eventos en los que ha participado el boxeo cubano a lo largo de la historia, ya sean nacionales (Juegos

Escolares, Campeonato Nacional Juvenil, Campeonato Playa Girón y Campeonato Giraldo Córdova Cardín) o internacionales (Copa del Mundo, Juegos del ALBA, Juegos Panamericanos, Juegos Centroamericanos, Juegos Olímpicos y Campeonatos Mundiales).

Este sistema brindará las noticias más relevantes del boxeo, tanto nacionales como internacionales, publicándolas en forma de resumen y ofreciendo la posibilidad de mostrarlas en forma ampliada.

Estará habilitado un foro de debate que permitirá el intercambio de opiniones de los aficionados que consulten este sitio. Los temas del mismo serán propuestos por los propios usuarios y aprobados por el administrador.

Existirá un espacio donde se publicarán pronósticos de las peleas próximas a efectuarse, estos pronósticos son realizados por especialistas y una vez realizados los combates se darán a conocer los resultados reales. Se mostrará un calendario, donde se resaltarán los días en que se efectuará alguna competencia y serán promocionados concursos y encuestas para lograr una mayor participación de los usuarios del sistema. Además se dispondrá de enlaces con sitios dedicados al boxeo que puedan ser útiles para los aficionados. Podrán publicarse artículos relacionados con temas novedosos del boxeo.

La sección Contáctenos permitirá a los interesados contactar con el administrador para hacer llegar sus sugerencias e inconformidades sobre el portal y de ésta manera ayudar al desarrollo y la mejora del mismo.

Dentro del sitio se podrán realizar búsquedas relacionadas con los temas abordados en cualquiera de sus secciones, facilitando al usuario encontrar los contenidos de su interés.

El administrador es el encargado de gestionar (insertar, modificar y eliminar) toda la información que se publicará en el sitio y dar soporte a todas las funcionalidades y necesidades del sistema. En caso de que sea necesario también podrá otorgar ciertos privilegios a determinados usuarios.

2.4 Exploración, Planificación e Iteraciones

Fase I: exploración

En esta etapa se definen las historias de usuario (HU), las mismas constan de 3 ó 4 líneas escritas por el cliente en un lenguaje no técnico y describen algo que el sistema debe hacer. Además se produce el contacto con las herramientas y tecnologías que se emplearán para construir el sistema y algunas ideas experimentales en cuanto a su

arquitectura son consideradas. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo. Esta fase dura típicamente par de semanas y el resultado es una visión general del sistema.

Fase II: planificación y entrega

En esta fase el cliente establece la prioridad de cada HU y a su vez los programadores realizan una estimación del esfuerzo necesario para la realización de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días.

Las estimaciones de esfuerzo asociadas a la implementación de las HU las establecen los programadores utilizando como medida el punto (semana ideal de programación).

Esta planificación se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuántas HU se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementarlas. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuántos puntos se pueden completar. Cuando se hace según el alcance del sistema, se divide la suma de puntos de las HU seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación.

Fase III: iteraciones

Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El Plan de Entrega está compuesto por iteraciones de no más de tres semanas. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Lo anterior se logra escogiendo las HU que fueren a la creación de esta arquitectura, sin embargo, no siempre es posible ya que el cliente decide qué HU se implementará en cada iteración (para maximizar el valor de negocio). Al final de la última iteración el sistema estará listo para entrar en producción.

Los elementos que deben tomarse en cuenta durante la elaboración del Plan de Iteración son: HU no abordadas, velocidad del proyecto, pruebas de aceptación no superadas en la iteración anterior y las tareas no terminadas en la iteración anterior.

2.5 Personas relacionadas con el sistema

Las personas relacionadas con el sistema son todas aquellas que de una forma u otra van a interactuar con la aplicación, incluyendo a los que mantienen el sistema actualizado en correcto funcionamiento y los que utilizan sus servicios.

Tabla 1: Modelo de HU

| Personas relacionadas con el sistema | Descripción |
|--------------------------------------|---|
| Invitado | Es la persona que navega por el sistema sin haberse creado una cuenta y lo hace sin privilegio alguno. Tiene la posibilidad de acceder a los distintos contenidos publicados. También puede acceder al foro para opinar, leer respuestas a los temas o proponer temas nuevos. |
| Administrador | Es la persona que realiza la gestión de todo el contenido del sistema. Es el encargado de administrar las diferentes cuentas de los usuarios autenticados en la aplicación. Necesariamente debe autenticarse de lo contrario navega como Invitado. |

2.6 Historias de Usuario

La HU es la técnica utilizada en XP para especificar las características del *software*. Son escritas por el propio cliente y describen brevemente las cualidades que el sistema debe poseer, sean funcionales o no funcionales.

No hay que preocuparse si en un principio no se identifican todas las HU. Al comienzo de cada iteración estarán registrados los cambios en las HU y según eso se planificará la siguiente iteración. A continuación se muestran las HU que tienen prioridad alta en el negocio.

Tabla 2: Representación de la HU #1

| Historia de Usuario | |
|---------------------|--------------------------------|
| Número: 1 | Nombre: Noticias en portada |

| | |
|---|---------------------------------|
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 1 |
| Prioridad en Negocio: alta (Alta / Media / Baja) | Puntos Estimados: 0.4 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: Se publican noticias relacionadas con el acontecer del boxeo cubano. El administrador podrá insertar, modificar y eliminar las noticias. El invitado puede consultar las noticias publicadas. | |
| Observaciones: la noticia podrá ser mostrada en forma de resumen, o todo su contenido. Se mostrarán las 3 últimas noticias publicadas, aunque se podrá acceder a todas las publicadas anteriormente. | |

Tabla 3: Representación de la HU #3

| Historia de Usuario | |
|--|---------------------------------|
| Número: 3 | Nombre: Ficha técnica |
| Usuario: administrador , invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 1 |
| Prioridad en Negocio: alta (Alta / Media / Baja) | Puntos Estimados: 1.0 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |

CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

| |
|--|
| <p>Descripción: el administrador tiene la posibilidad de crear una ficha técnica del atleta solicitando nombre, apellidos, edad, división, nivel escolar, desempeño deportivo, fotografía y otros datos de interés. Cada ficha técnica puede ser modificada, eliminada y mostrada. El invitado podrá acceder a la información de las fichas técnicas.</p> |
| <p>Observaciones:</p> |

Tabla 4: Representación de la HU #15

| Historia de Usuario | |
|--|---------------------------------|
| Número: 15 | Nombre: Reglas |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 1 |
| Prioridad en Negocio: alta (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se expondrán las reglas del boxeo amateur. El administrador podrá insertar, modificar y mostrar las reglas. El invitado podrá acceder a las reglas publicadas. | |
| Observaciones: | |

Tabla 5: Representación de la HU #16

| Historia de Usuario | |
|--|-----------------------------|
| Número: 16 | Nombre: Estatutos |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: |

| | |
|--|---------------------------------|
| | 1 |
| Prioridad en Negocio: alta (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se expondrán los estatutos por los que se rige la Escuela Cubana de Boxeo. El administrador podrá insertar, modificar y mostrar los estatutos. El invitado podrá acceder a los estatutos publicados. | |
| Observaciones: En esta opción también se desplegará una opción para acceder a los 10 mandamientos del boxeador y al código de ética. | |

Tabla 6: Representación de la HU #17

| Historia de Usuario | |
|--|---------------------------------|
| Número: 17 | Nombre: Historia |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 1 |
| Prioridad en Negocio: alta (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se dará a conocer la historia de la Escuela Cubana de Boxeo desde sus inicios. El administrador podrá insertar, modificar y mostrar contenidos que abarcan la historia del boxeo cubano. El invitado podrá acceder a la información publicada. | |
| Observaciones: | |

Tabla 7: Representación de la HU #22

| Historia de Usuario | |
|--|----------------------------------|
| Número: 22 | Nombre: Equipo técnico |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 1 |
| Prioridad en Negocio: alta (Alta / Media / Baja) | Puntos Estimados: 0.4 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se publicará información de los integrantes del equipo técnico. El administrador podrá insertar, modificar y eliminar información de los integrantes del equipo técnico como nombre, apellidos, rol desempeñado, resultados alcanzados y otros datos de interés. El invitado podrá acceder a los datos publicados. | |
| Observaciones: | |

Para ver las restantes HU consultar anexo 7.

2.7 Iteraciones

Iteración 1

En esta iteración se dará cumplimiento a las HU 1, 3, 15, 16, 17 y 22. Las HU mencionadas anteriormente son las que cubren las funcionalidades básicas del sistema como las noticias, reglas del boxeo amateur, estatutos de la Escuela Cubana de Boxeo e historia del deporte en el país. Además se ofrecen las fichas técnicas de los púgiles cubanos y datos de los integrantes del equipo técnico. Todas éstas son las que brindan la mayor parte de la información que se quiere publicar.

Iteración 2

Esta iteración se centra en una parte de las necesidades de prioridad media, las cuales no presentan riesgos elevados para los programadores en su mayoría. Está conformada por las HU 2, 4, 7, 12, 13, 18 y 19. Aquí se realizan las HU relacionadas con las entrevistas, el ranking, el calendario competitivo y los resultados obtenidos por la Escuela Cubana de Boxeo a lo largo de la historia en los diferentes eventos efectuados. La presente iteración además está dirigida a la gestión de las imágenes y videos relacionados con el deporte que sean de interés. Además se encarga de la parte relacionada con administrar usuarios y roles que puede ser de utilidad para dar cumplimiento a alguna funcionalidad del sistema.

Iteración 3

El objetivo de esta iteración es dar seguimiento a los objetivos de la anterior, se enmarca en las necesidades de prioridad media que no han sido abordadas, por lo que abarca las HU 8, 9, 10, 14, 23 y 24. En ella se gestiona y administra todo lo relacionado con el foro de discusión, se publican archivos que puedan ser de interés para los usuarios (como investigaciones realizadas por especialistas u otros documentos), los que podrán ser descargados. Se enmarca en la publicación de pronósticos hechos por los especialistas sobre los venideros torneos, así como los resultados reales de los mismos una vez efectuados. También recoge la publicación de información sobre el presidente de la Escuela Cubana de Boxeo. Además esta iteración se enmarca en la publicación de artículos sobre temas relacionados con el deporte en cuestión.

Iteración 4

Agrupar las funcionalidades que poseen baja prioridad en el negocio según la clasificación del cliente. La iteración se conforma por las HU 5, 6, 11, 20 y 21. En ellas se recogen los procesos dirigidos a la autenticación de usuarios, la realización de búsquedas en el portal, la publicación de concursos y encuestas relacionadas con el deporte en cuestión. Además abarca enlaces con otros sitios de interés relacionados con el boxeo.

2.8 Prototipo de Interfaz de Usuario



Figura 1: Prototipo de interfaz de usuario

2.9 Estimación del esfuerzo por historia de usuario

Tabla 8: Estimación de esfuerzo por HU

| Historias de Usuarios | Tiempo Estimado (semana ideal de trabajo) | Iteración Asignada | Tiempo Real |
|---|---|--------------------|-------------|
| Noticias en portada Ficha técnica Reglas Estatutos Historia Equipo técnico | 2.4 | 1 | 2.6 |
| Entrevistas Calendario competitivo | 4.2 | 2 | 3.4 |

CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN

| | | | |
|------------------------------|-----|---|-----|
| Ranking | | | |
| Administrar usuarios y roles | | | |
| Resultados | | | |
| Imágenes | | | |
| Videos | | | |
| Material de archivo | 1.4 | 3 | 1.6 |
| Administrar foro | | | |
| Foro | | | |
| Artículos | | | |
| Combates del día | | | |
| Presidencia | | | |
| Autenticar | 1.2 | 4 | 1.2 |
| Otros sitios | | | |
| Realizar búsquedas | | | |
| Concursos | | | |
| Encuestas | | | |

2.10 Plan de Entregas

Tabla 9: Plan de entregas

| Historia de Usuarios | Iteración 1 Marzo 12 | Iteración 2 Abril 8 | Iteración 3 Abril 19 | Iteración 4 Abril 28 |
|----------------------|-------------------------|------------------------|-------------------------|-------------------------|
| Noticias en Portada | 1.0 | Terminado | Terminado | Terminado |
| Ficha Técnica | | | | |
| Reglas | | | | |
| Estatutos | | | | |
| Historia | | | | |
| Equipo Técnico | | | | |

| | | | | |
|------------------------------|-----|-----|-----------|-----------|
| Entrevistas | 0.1 | 1.0 | Terminado | Terminado |
| Calendario competitivo | | | | |
| Ranking | | | | |
| Administrar usuarios y roles | | | | |
| Resultados | | | | |
| Imágenes | | | | |
| Videos | | | | |
| Material de archivo | - | 0.1 | 1.0 | Terminado |
| Administrar foro | | | | |
| Foro | | | | |
| Artículos | | | | |
| Combates del día | | | | |
| Presidencia | 0.1 | 0.1 | 0.1 | 1.0 |
| Autentificar | | | | |
| Otros sitios | | | | |
| Realizar búsquedas | | | | |
| Concursos | | | | |
| Encuestas | | | | |

2.11 Conclusiones

Una vez estudiada la forma en que se realiza la gestión de la información en la Escuela Cubana de Boxeo, se obtuvo la propuesta de las funcionalidades del sistema, las cuales serán implementadas en 4 iteraciones. Se constituyó el Plan de Entrega del proyecto determinando un cronograma en conjunto con el cliente que especifica en que momento debe realizarse cada una de las entregas.

3. IMPLEMENTACIÓN Y PRUEBAS

3.1 Introducción

La metodología XP plantea que la implementación debe realizarse de forma iterativa, obteniéndose después de cada iteración un producto que debe ser probado y mostrado al cliente, lo que permite una constante retroalimentación entre los desarrolladores y clientes haciendo posible que los primeros puedan ampliar su visión del producto apoyados en la visión de los segundos. Las iteraciones especifican en que momento se dará cumplimiento a las HU según la prioridad establecida por el cliente y las tareas generadas ayudan a dar solución a las historias. Todas las funcionalidades del *software* deben ser probadas para que satisfagan las necesidades del cliente y mejorar la calidad del código.

3.2 Diseño del sistema

La metodología XP propone entre sus prácticas la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto, la complejidad innecesaria y el código extra deben ser removidos inmediatamente. El diseño adecuado para el *software* es aquel que supera con éxito todas las pruebas, no tiene lógica duplicada, refleja claramente la intención de implementación de los programadores y tiene el menor número posible de clases y métodos. Si alguna parte de la implementación resulta especialmente compleja, se replantea. De esta manera, cualquier cambio o modificación será mucho más sencillo, en eso se basa la simplicidad del diseño.

Para el diseño de las aplicaciones, la metodología XP no requiere la representación del sistema mediante diagramas de clases utilizando notación UML, en su lugar se usan las tarjetas CRC (Contenido, Responsabilidad y Colaboración), se trata de simples tarjetas de papel para fichado que reemplazan a los diagramas en la representación de modelos. El proceso se realiza creando las tarjetas (al inicio sólo se escribe el nombre en ellas, el resto ya se irá completando) y situándolas próximas a las que comparten interfaces o llamadas. Las tarjetas que corresponden a objetos que heredan o son interfaces de otros pueden situarse encima o debajo.(33)

Este mecanismo ayuda a que todas las personas participen y aporten sus ideas en el diseño moviendo las tarjetas encima de la mesa según éste va progresando. Si se llega a un punto muerto y se decide volver a empezar, será tan simple como limpiar la mesa y volver a ir situando las tarjetas. No se tendrán que borrar diagramas ni hacer trabajo extra diseñando alternativas. Cuando se tenga el diseño definitivo, será más

fácil de mantener en la memoria por parte de los participantes y es entonces cuando se entra en detalle en cada objeto y se hacen los diagramas necesarios.(34)

A continuación se brindan los módulos desarrollados mediante sus respectivas tarjetas CRC.

3.2.1 Tarjetas CRC

Tabla 10: Tarjeta CRC módulo Noticias

| Módulo Noticias | |
|------------------------|---------------|
| Insertar noticia | Bloque |
| Modificar noticia | Cargar |
| Eliminar noticia | Autenticación |

Tabla 11: Tarjeta CRC módulo Fichas

| Módulo Fichas | |
|----------------------|---------------|
| Insertar ficha | Bloque |
| Modificar ficha | Cargar |
| Eliminar ficha | Autenticación |
| | Entrevistas |

Tabla 12: Tarjeta CRC módulo Equipo

| Módulo Equipo | |
|----------------------|---------------|
| Insertar integrante | Bloque |
| Modificar integrante | Cargar |
| Eliminar integrante | Autenticación |

Tabla 13: Tarjeta CRC módulo Entrevistas

| Módulo Entrevistas | |
|---------------------------|---------------|
| Insertar entrevista | Bloque |
| Modificar entrevista | Cargar |
| Eliminar entrevista | Autenticación |
| | Fichas |

Tabla 14: Tarjeta CRC módulo Artículos

| Módulo Artículos | |
|-------------------------|---------------|
| Insertar artículo | Menú |
| Modificar artículo | Cargar |
| Eliminar artículo | Autenticación |

Tabla 15: Tarjeta CRC módulo Vínculos

| Módulo Vínculos | |
|------------------------|---------------|
| Insertar vínculo | Bloque |
| Modificar vínculo | Cargar |
| Eliminar vínculo | Autenticación |

Tabla 16: Tarjeta CRC módulo Concursos

| Módulo Concursos | |
|-------------------------|---------------|
| Insertar concurso | Menú |
| Modificar concurso | Cargar |
| Eliminar concurso | Autenticación |

3.2.2 Visión general de la arquitectura de Drupal

El secreto de Drupal para conseguir su reconocida flexibilidad y facilidad en la creación de sitios Web es la abstracción y organización en capas que aplica en el tratamiento de los contenidos. En lugar de considerar el sitio Web como un conjunto de páginas interrelacionadas, estructura los contenidos en una serie de elementos básicos. Éstos son los nodos (*nodes*), módulos (*modules*), bloques y menús (*blocks & menus*), permisos de usuarios y plantillas (*templates*).

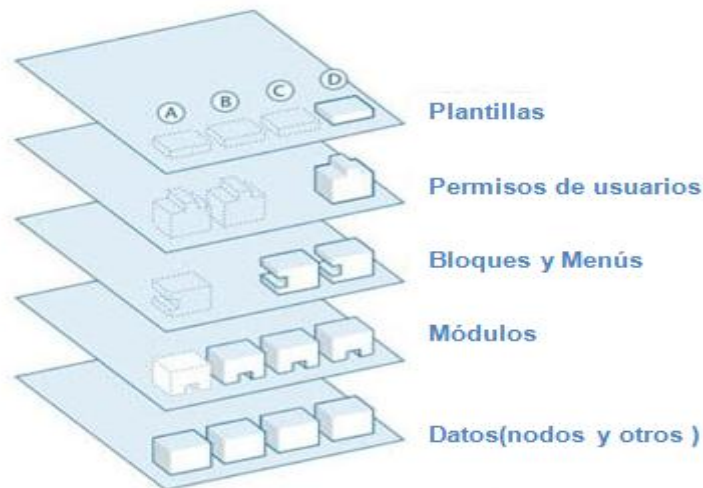


Figura 2: Arquitectura del CMS Drupal(35)

Los nodos son los elementos básicos en que se almacena la información y los contenidos. Así a medida que el sitio Web crece, lo va haciendo el número de nodos formando un depósito de nodos. Se puede decir que la primera capa de la estructura la forma este depósito de nodos.

Los elementos que operan sobre los nodos son los módulos y otorgan funcionalidad a Drupal permitiendo incrementar su capacidad o adaptar las necesidades de cada sitio Web. Estos módulos son como *Plug-Ins* que se instalan proporcionando nuevas funcionalidades.

La siguiente capa la constituyen los bloques y menús. Éstos permiten estructurar y organizar los contenidos en la página Web. Es decir, son los elementos que albergan y permiten acceder al usuario a la salida generada y procesada por los módulos a partir de la información almacenada en los nodos.

Le sigue la capa de control de usuarios y permisos. Actualmente, la mayor parte de los sitios Web son multiusuario, por lo que la seguridad y control de los usuarios es un punto clave para garantizar la integridad de la información almacenada. Con esta finalidad se dispone de un registro de usuarios y de roles que permiten especificar las tareas que se pueden realizar y a qué contenidos puede acceder cada tipo de usuario. Es decir que las operaciones que se pueden realizar sobre los elementos provenientes de las capas inferiores (lectura, modificación o creación) se encuentran limitadas por la capa de control de usuarios.

En el caso de la apariencia gráfica o estilo de la información que se muestra al usuario, la establece la capa de temas. Esta separación entre información y aspecto gráfico permite cambiar el diseño del sitio Web sin necesidad de modificar los

contenidos, lo que es muy práctico si lo único que se quiere es renovar la apariencia del sitio.(35)

3.3 Fase de implementación

En el comienzo de esta fase se hace un chequeo de cada una de las HU junto con el plan de iteraciones y se modifica en caso de ser necesario. Como parte de este plan se crean tareas para ayudar a organizar la implementación exitosa de las HU, asignando a un grupo de desarrollo (que puede ser una persona), responsable de su implementación. Como estas tareas son para el uso estricto de los programadores pueden escribirse en un lenguaje técnico, al contrario de las HU que son escritas en el lenguaje del cliente.

Teniendo en cuenta la planificación realizada anteriormente, se desarrollaron 4 iteraciones sobre el sistema, obteniéndose un producto con las características deseadas por el cliente al terminarse la última iteración. A continuación se detalla cada una de las iteraciones.

3.3.1 Iteración 1

En esta iteración se dará cumplimiento a las HU que cubren las funcionalidades básicas del sistema, ya que las mismas son las que brindan la mayor parte de la información que se quiere publicar y son las de máximo interés del cliente.

3.3.1.1 Historias abordadas en la primera iteración

Las HU abordadas en la primera iteración son las siguientes:

Tabla 17: HU abordadas en la 1ra iteración

| Historia de Usuario | Estimación | Real |
|---------------------|------------|------|
| Noticias en portada | 0.4 | 0.6 |
| Ficha técnica | 1.0 | 1.0 |
| Reglas | 0.2 | 0.2 |
| Estatutos | 0.2 | 0.2 |
| Historia | 0.2 | 0.2 |
| Equipo técnico | 0.4 | 0.4 |

3.1.1.2 Tareas de las HU abordadas en la primera iteración.

Las tareas para dar cumplimiento a las HU de la primera iteración son las siguientes:

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Tabla 18: Tarea #1 de la HU noticias en portada

| Tarea | |
|---|--|
| Número de tarea: 1 | HU: 1 |
| Nombre de la tarea: crear la tabla perteneciente al módulo noticias. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 23 de febrero del 2010 | Fecha fin: 24 de febrero del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se establecerá un archivo de tipo <i>.install</i> donde se creará y destruirá la tabla <i>tb_noticias</i> para permitir la persistencia de datos. | |

Tabla 19: Tarea #2 de la HU noticias en portada

| Tarea | |
|--|--|
| Número de tarea: 2 | HU: 1 |
| Nombre de la tarea: crear las funcionalidades del módulo noticias | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 24 de febrero del 2010 | Fecha fin: 25 de febrero del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: implementar funcionalidades para la gestión de las noticias. Se implementará un archivo <i>.module</i> con la ayuda de diversos <i>hooks</i> . | |

Tabla 20: Tarea #3 de la HU noticias en portada

| Tarea | |
|---|--|
| Número de tarea: 3 | HU: 1 |
| Nombre de la tarea: Implementar la interfaz de noticias. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 25 de febrero del 2010 | Fecha fin: 26 de febrero del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean bloques y formularios para mostrar las noticias. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Tabla 21: Tarea #1 de la HU ficha técnica

| Tarea | |
|---|---------------------------------------|
| Número de tarea: 1 | HU: 3 |
| Nombre de la tarea: crear la tabla perteneciente al módulo fichas. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 26 de febrero del 2010 | Fecha fin: 2 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se establecerá un archivo de tipo <i>.install</i> donde se creará y destruirá la tabla <i>tb_fichas</i> para permitir la persistencia de datos. | |

Tabla 22: Tarea #2 de la HU ficha técnica

| Tarea | |
|---|---------------------------------------|
| Número de tarea: 2 | HU: 3 |
| Nombre de la tarea: crear las funcionalidades del módulo fichas. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.4 |
| Fecha de inicio: 2 de marzo del 2010 | Fecha fin: 3 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: implementar funcionalidades para la gestión de las fichas técnicas. Se implementará un archivo <i>.module</i> con la ayuda de diversos <i>hooks</i> . | |

Tabla 23: Tarea #3 de la HU ficha técnica

| Tarea | |
|--|---------------------------------------|
| Número de tarea: 3 | HU: 1 |
| Nombre de la tarea: implementar la interfaz de fichas. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.4 |
| Fecha de inicio: 3 de marzo del 2010 | Fecha fin: 5 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean bloques y formularios para mostrar las fichas técnicas. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Tabla 24: Tarea #1 de la HU reglas

| Tarea | |
|---|---------------------------------------|
| Número de tarea: 1 | HU: 15 |
| Nombre de la tarea: implementar la interfaz de reglas. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 5 de marzo del 2010 | Fecha fin: 5 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean las páginas para mostrar las reglas. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

Tabla 25: Tarea #1 de la HU estatutos

| Tarea | |
|--|---------------------------------------|
| Número de tarea: 1 | HU: 16 |
| Nombre de la tarea: implementar la interfaz de estatutos. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 8 de marzo del 2010 | Fecha fin: 8 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean las páginas para mostrar los estatutos. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

Tabla 26: Tarea #1 de la HU historia

| Tarea | |
|--|---------------------------------------|
| Número de tarea: 1 | HU: 17 |
| Nombre de la tarea: implementar la interfaz de historia. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 9 de marzo del 2010 | Fecha fin: 9 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean las páginas para mostrar la historia. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

Tabla 27: Tarea #1 de la HU equipo técnico

| Tarea | |
|---|--|
| Número de tarea: 1 | HU: 22 |
| Nombre de la tarea: crear la tabla perteneciente al módulo equipo. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 9 de marzo del 2010 | Fecha fin: 10 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se establecerá un archivo de tipo <i>.install</i> donde se creará y destruirá la tabla <i>tb_equipo</i> para permitir la persistencia de datos. | |

Tabla 28: Tarea #2 de la HU equipo técnico

| Tarea | |
|---|--|
| Número de tarea: 2 | HU: 22 |
| Nombre de la tarea: crear las funcionalidades del módulo equipo. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 10 de marzo del 2010 | Fecha fin: 11 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: implementar funcionalidades para la gestión de la información de los integrantes de equipo técnico. Se implementará un archivo <i>.module</i> con la ayuda de diversos <i>hooks</i> . | |

Tabla 29: Tarea #3 de la HU equipo técnico

| Tarea | |
|--|--|
| Número de tarea: 3 | HU: 22 |
| Nombre de la tarea: implementar la interfaz de equipo. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 11 de marzo del 2010 | Fecha fin: 11 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean bloques y formularios para mostrar la información de los integrantes de equipo técnico. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

3.3.2 Iteración 2

Esta iteración se centra en una parte de las HU de prioridad media, los cuales no presentan riesgos elevados para los programadores en su mayoría.

3.3.2.1 Historias abordadas en la segunda iteración

Las HU abordadas en la segunda iteración son las siguientes:

Tabla 30: HU abordadas en la 2da iteración

| Historia de Usuario | Estimación | Real |
|------------------------------|------------|------|
| Entrevistas | 0.4 | 0.4 |
| Calendario competitivo | 1.0 | 0.6 |
| Ranking | 0.6 | 0.6 |
| Administrar usuarios y roles | 0.4 | 0.4 |
| Resultados | 1.0 | 0.4 |
| Imágenes | 0.4 | 0.4 |
| Videos | 0.4 | 0.4 |

Para ver las tareas de las HU abordadas en la segunda iteración consultar anexo 1.

3.3.3 Iteración 3

El objetivo de esta iteración es dar seguimiento a los objetivos de la anterior, se enmarca en las necesidades de prioridad media que no han sido abordados en las iteraciones anteriores.

3.3.3.1 Historias abordadas en la tercera iteración

Las HU abordadas en la tercera iteración son las siguientes:

Tabla 31: HU abordadas en la 3ra iteración

| Historia de Usuario | Estimación | Real |
|---------------------|------------|------|
| Material de archivo | 0.2 | 0.2 |
| Administrar foro | 0.2 | 0.2 |
| Foro | 0.2 | 0.2 |
| Artículos | 0.4 | 0.6 |
| Combates del día | 0.2 | 0.2 |

| | | |
|-------------|-----|-----|
| Presidencia | 0.2 | 0.2 |
|-------------|-----|-----|

Para ver las tareas de las HU abordadas en la tercera iteración consultar anexo 2.

3.3.4 Iteración 4

Esta iteración agrupa las HU que poseen baja prioridad en el negocio según la clasificación del cliente.

3.3.4.1 Historias abordadas en la 4ta iteración

Las HU abordadas en la cuarta iteración son las siguientes:

Tabla 32: HU abordadas en la 4ta iteración

| Historia de Usuario | Estimación | Real |
|---------------------|------------|------|
| Autenticar | 0.2 | 0.2 |
| Otros sitios | 0.4 | 0.4 |
| Realizar búsquedas | 0.2 | 0.2 |
| Concursos | 0.4 | 0.4 |

Para ver las tareas de las HU abordadas en la cuarta iteración consultar anexo 3.

3.3.5 Diagrama de Despliegue

El diagrama de despliegue se utiliza para modelar el *hardware* utilizado en la implementación de sistemas y la relación entre sus componentes. Describe la topología del sistema, la estructura de los elementos de *hardware* y el *software* que ejecuta cada uno de ellos. Las máquinas físicas y los procesadores se representan como nodos. Los nodos son conectados por asociaciones de comunicación tales como enlaces de red, conexiones TCP/IP y otras(36). En este caso en particular la aplicación se encuentra hospedada en un servidor Web Apache y se comunica con una base de datos montada con el SGBD PostgreSQL.

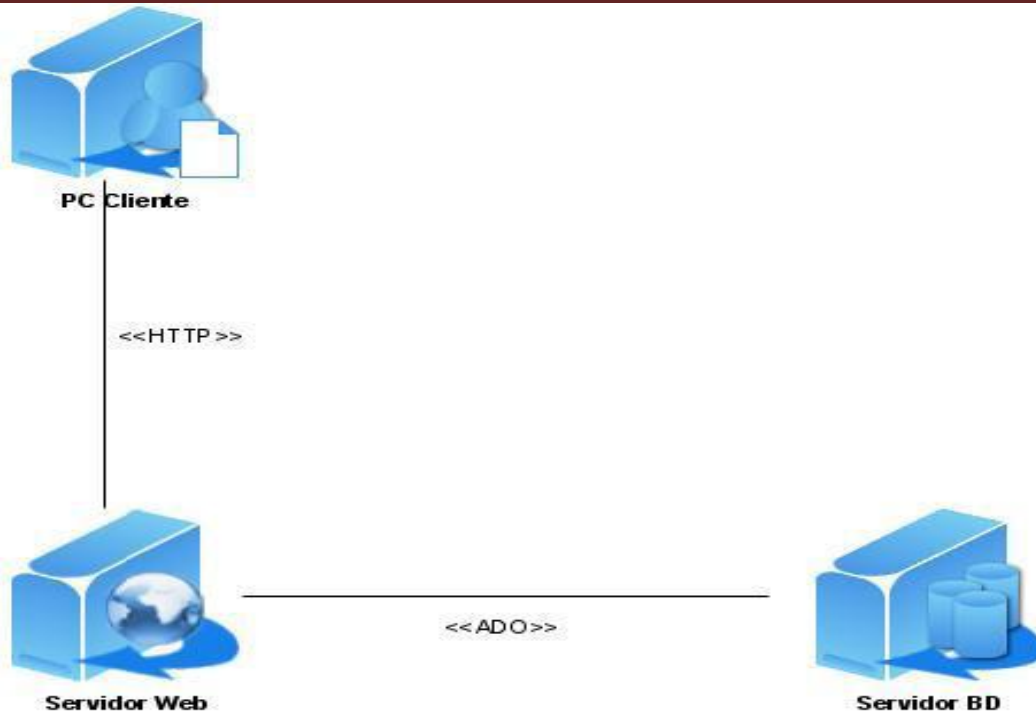


Figura 3: Diagrama de despliegue

3.4 Pruebas

Una de las características principales de XP es su fuerte énfasis en las pruebas, poniendo la comprobación como el fundamento del desarrollo. Las pruebas se integran en el proceso de integración continua y construcción, lo que rinde una plataforma altamente estable para el desarrollo futuro.

Las pruebas constituyen unos de los pilares básicos de XP. No debe existir ninguna característica en el programa que no pueda ser probada, los programadores escriben pruebas para chequear el correcto funcionamiento del programa, los clientes realizan pruebas funcionales. El resultado es un programa más seguro que con el paso del tiempo sea capaz de aceptar nuevos cambios.(37)

Estas pruebas se dividen en dos grupos: pruebas unitarias y pruebas de aceptación. En el caso de las unitarias son desarrolladas por los programadores y se realizan a los principales procesos, de tal manera que adelantándose en algo hacia el futuro, se puedan hacer pruebas de las fallas que pudieran ocurrir, es como adelantarse a obtener los posibles errores(38). Por su parte las pruebas de aceptación están destinadas a evaluar si al final de una iteración se obtuvo la funcionalidad requerida, además comprobar que dicha funcionalidad sea la esperada por el cliente.

3.4.1 Pruebas Unitarias

La prueba unitaria es una práctica que consiste en tomar una pequeña porción de código y someterlo a pruebas de programación para verificar su corrección. Por lo general se someten a prueba funciones o pequeñas funcionalidades específicas con un conjunto de parámetros controlados que servirán para hacer las comprobaciones necesarias en función de los valores de retorno de la funcionalidad probada.

A continuación se exponen algunas utilidades de este tipo de pruebas:

- ✓ Ayudan a producir un código de mayor calidad.
- ✓ Detección rápida de errores cuando se programan nuevas funcionalidades o se realizan cambios en el código.
- ✓ Sirven como pequeña fuente de documentación sobre qué es lo que se espera que haga el código.
- ✓ Realizar pruebas unitarias obliga al programador a escribir el código en pequeñas porciones con el fin de que puedan ser probadas independientemente.

En el caso de Drupal, la herramienta principal que usa es el módulo SimpleTest, que es una adaptación a Drupal del proyecto SimpleTest de PHP. Este módulo añade gran cantidad de herramientas para la realización de las pruebas como por ejemplo crear nuevos usuarios, cambiar variables de configuración, establecer permisos a usuarios y otros.(39)

Se le hizo pruebas unitarias a todo el código de los nuevos módulos implementados y se aseguró su correcto funcionamiento.

3.4.2 Pruebas de Aceptación

Cada HU lleva asociada criterios de aceptación y para cada criterio de aceptación se definen casos de prueba. Las pruebas de aceptación se escriben en las fases tempranas del desarrollo y antes de que el sistema se implemente para validar las necesidades de los usuarios y dirigir la implementación. Se puede considerar que las HU y por extensión, las pruebas asociadas, juegan el papel de especificaciones del sistema.

Las pruebas de aceptación son creadas a partir de las HU. Durante una iteración la HU seleccionada en la planificación de iteraciones se convertirá en una prueba de aceptación. El cliente o usuario especifica los aspectos a probar cuando una HU ha sido correctamente implementada. Una HU puede tener más de una prueba de

aceptación, tantas como sean necesarias para garantizar su correcto funcionamiento. Cada prueba de aceptación representa una salida esperada del sistema. Es responsabilidad del cliente verificar la corrección de las pruebas de aceptación y tomar decisiones acerca de las mismas. Ninguna HU se considera completa hasta que no supera sus pruebas de aceptación.(40)

3.4.2.1 Iteración 1

Tabla 33: Prueba a la funcionalidad publicar noticias en portada

| Caso de prueba de Aceptación | |
|--|-------------------------------|
| Código: HU1_P1 | Historia de Usuario: 1 |
| Nombre: Publicar noticias en el sistema. | |
| Descripción: Prueba para la funcionalidad de publicar noticias en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para publicar noticias. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: Se intenta publicar noticias por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Las noticias son publicadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 34: Prueba a la funcionalidad modificar noticias en portada

| Caso de prueba de Aceptación | |
|---|-------------------------------|
| Código: HU1_P2 | Historia de Usuario: 1 |
| Nombre: Modificar noticias en el sistema. | |
| Descripción: Prueba para la funcionalidad de modificar noticias en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para modificar las noticias. Se utilizará un usuario con datos válidos. Deben existir noticias publicadas en el sistema. | |
| Entradas/Pasos de ejecución: | |

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

| |
|--|
| Se intenta modificar noticias por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. |
| Resultado esperado: Las noticias son modificadas sin generar error. |
| Evaluación de la prueba: Prueba satisfactoria. |

Tabla 35: Prueba a la funcionalidad eliminar noticias en portada

| Caso de prueba de Aceptación | |
|--|-------------------------------|
| Código: HU1_P3 | Historia de Usuario: 1 |
| Nombre: Eliminar noticias en el sistema. | |
| Descripción: Prueba para la funcionalidad de eliminar noticias en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para eliminar las noticias. Se utilizará un usuario con datos válidos. Deben existir noticias publicadas en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta eliminar noticias por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Las noticias son eliminadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 36: Prueba a la funcionalidad crear ficha técnica

| Caso de prueba de Aceptación | |
|--|-------------------------------|
| Código: HU3_P1 | Historia de Usuario: 3 |
| Nombre: Crear ficha técnica de los atletas. | |
| Descripción: Prueba para la funcionalidad de crear ficha técnica de los atletas. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para crear las fichas técnicas. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: Se intenta crear fichas técnicas de los atletas por un usuario que contiene sus datos | |

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

| |
|--|
| válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. |
| Resultado esperado: Las fichas técnicas de los atletas son creadas sin generar error. |
| Evaluación de la prueba: Prueba satisfactoria. |

Tabla 37: Prueba la funcionalidad modificar ficha técnica

| Caso de prueba de Aceptación | |
|---|-------------------------------|
| Código: HU3_P2 | Historia de Usuario: 3 |
| Nombre: Modificar ficha técnica de los atletas. | |
| Descripción: Prueba para la funcionalidad de modificar ficha técnica de los atletas. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para modificar las fichas técnicas. Se utilizará un usuario con datos válidos. Deben existir fichas técnicas publicadas en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta modificar las fichas técnicas de los atletas por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Las fichas técnicas de los atletas son modificadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 38: Prueba la funcionalidad eliminar ficha técnica

| Caso de prueba de Aceptación | |
|---|-------------------------------|
| Código: HU3_P3 | Historia de Usuario: 3 |
| Nombre: Eliminar ficha técnica de los atletas. | |
| Descripción: Prueba para la funcionalidad de eliminar ficha técnica de los atletas. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para eliminar las fichas técnicas. Se utilizará un usuario con datos válidos. | |

| |
|---|
| Deben existir fichas técnicas publicadas en el sistema. |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta eliminar las fichas técnicas de los atletas por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> |
| <p>Resultado esperado: Las fichas técnicas de los atletas son eliminadas sin generar error.</p> |
| <p>Evaluación de la prueba: Prueba satisfactoria.</p> |

Tabla 39: Prueba a la funcionalidad publicar reglas

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU15_P1 | Historia de Usuario: 15 |
| Nombre: Publicar reglas del boxeo amateur. | |
| Descripción: Prueba para la funcionalidad de publicar reglas del boxeo amateur. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para publicar las reglas.</p> <p>Se utilizará un usuario con datos válidos.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta publicar las reglas del boxeo amateur por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: Las reglas del boxeo amateur son publicadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 40: Prueba a la funcionalidad modificar reglas

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU15_P2 | Historia de Usuario: 15 |
| Nombre: Modificar reglas del boxeo amateur. | |
| Descripción: Prueba para la funcionalidad de modificar reglas del boxeo amateur. | |
| Condiciones de ejecución: | |

| |
|--|
| <p>El usuario debe estar autenticado y tener permisos para modificar las reglas.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Deben existir reglas publicadas en el sistema.</p> |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta modificar las reglas del boxeo amateur por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> |
| <p>Resultado esperado: Las reglas del boxeo amateur son modificadas sin generar error.</p> |
| <p>Evaluación de la prueba: Prueba satisfactoria.</p> |

Tabla 41: Prueba a la funcionalidad eliminar reglas

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU15_P3 | Historia de Usuario: 15 |
| Nombre: Eliminar reglas del boxeo amateur. | |
| Descripción: Prueba para la funcionalidad de eliminar reglas del boxeo amateur. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para eliminar las reglas.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Deben existir reglas publicadas en el sistema.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta eliminar las reglas del boxeo amateur por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: Las reglas del boxeo amateur son eliminadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 42: Prueba a la funcionalidad publicar estatutos

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU16_P1 | Historia de Usuario: 16 |
| Nombre: Publicar los estatutos de la Escuela Cubana de Boxeo. | |

| |
|--|
| <p>Descripción: Prueba para la funcionalidad de publicar los estatutos de la Escuela Cubana de Boxeo.</p> |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para publicar los estatutos.</p> <p>Se utilizará un usuario con datos válidos.</p> |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta publicar los estatutos de la Escuela Cubana de Boxeo por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> |
| <p>Resultado esperado: Los estatutos de la Escuela Cubana de Boxeo son publicados sin generar error.</p> |
| <p>Evaluación de la prueba: Prueba satisfactoria.</p> |

Tabla 43: Prueba a la funcionalidad modificar estatutos

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU16_P2 | Historia de Usuario: 16 |
| Nombre: Modificar los estatutos de la Escuela Cubana de Boxeo. | |
| Descripción: Prueba para la funcionalidad de modificar los estatutos de la Escuela Cubana de Boxeo. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para modificar los estatutos.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Deben existir estatutos de la Escuela Cubana de Boxeo publicados en el sistema.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta modificar los estatutos de la Escuela Cubana de Boxeo por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: Los estatutos de la Escuela Cubana de Boxeo son modificados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 44: Prueba a la funcionalidad eliminar estatutos

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU16_P3 | Historia de Usuario: 16 |
| Nombre: Eliminar los estatutos de la Escuela Cubana de Boxeo. | |
| Descripción: Prueba para la funcionalidad de eliminar los estatutos de la Escuela Cubana de Boxeo. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para eliminar los estatutos. Se utilizará un usuario con datos válidos. Deben existir estatutos de la Escuela Cubana de Boxeo publicados en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta eliminar los estatutos de la Escuela Cubana de Boxeo por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los estatutos de la Escuela Cubana de Boxeo son eliminados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 45: Prueba a la funcionalidad publicar historia

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU17_P1 | Historia de Usuario: 17 |
| Nombre: Publicar la historia de la Escuela Cubana de Boxeo. | |
| Descripción: Prueba para la funcionalidad de publicar la historia de la Escuela Cubana de Boxeo. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para publicar la historia de la Escuela Cubana de Boxeo. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: Se intenta publicar la historia de la Escuela Cubana de Boxeo por un usuario que | |

CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

| |
|---|
| contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. |
| Resultado esperado: La historia de la Escuela Cubana de Boxeo es publicada sin generar error. |
| Evaluación de la prueba: Prueba satisfactoria. |

Tabla 46: Prueba a la funcionalidad modificar historia

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU17_P2 | Historia de Usuario: 17 |
| Nombre: Modificar la historia de la Escuela Cubana de Boxeo. | |
| Descripción: Prueba para la funcionalidad de modificar la historia de la Escuela Cubana de Boxeo. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para modificar la historia de la Escuela Cubana de Boxeo. Se utilizará un usuario con datos válidos. Debe existir historia de la Escuela Cubana de Boxeo publicada en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta modificar la historia de la Escuela Cubana de Boxeo por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: La historia de la Escuela Cubana de Boxeo es modificada sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 47: Prueba a la funcionalidad eliminar historia

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU17_P3 | Historia de Usuario: 17 |
| Nombre: Eliminar la historia de la Escuela Cubana de Boxeo. | |
| Descripción: Prueba para la funcionalidad de eliminar la historia de la Escuela Cubana de Boxeo. | |

| |
|--|
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para eliminar la historia de la Escuela Cubana de Boxeo.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Debe existir historia de la Escuela Cubana de Boxeo publicada en el sistema.</p> |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta eliminar la historia de la Escuela Cubana de Boxeo por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> |
| <p>Resultado esperado: La historia de la Escuela Cubana de Boxeo es eliminada sin generar error.</p> |
| <p>Evaluación de la prueba: Prueba satisfactoria.</p> |

Tabla 48: Prueba a la funcionalidad publicar equipo técnico

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU22_P1 | Historia de Usuario: 22 |
| Nombre: Publicar información de los integrantes del equipo técnico. | |
| Descripción: Prueba para la funcionalidad de publicar información de los integrantes del equipo técnico. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para publicar información sobre los integrantes del equipo técnico.</p> <p>Se utilizará un usuario con datos válidos.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta publicar información de los integrantes del equipo técnico por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: La información de los integrantes del equipo técnico es publicada sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 49: Prueba a la funcionalidad modificar equipo técnico

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU22_P2 | Historia de Usuario: 22 |
| Nombre: Modificar información de los integrantes del equipo técnico. | |
| Descripción: Prueba para la funcionalidad de modificar información de los integrantes del equipo técnico. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para modificar la información sobre los integrantes del equipo técnico. Se utilizará un usuario con datos válidos. Debe existir información de los integrantes del equipo técnico publicada en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta modificar información de los integrantes del equipo técnico por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: La información de los integrantes del equipo técnico es modificada sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 50: Prueba a la funcionalidad eliminar equipo técnico

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU22_P3 | Historia de Usuario: 22 |
| Nombre: Eliminar información de los integrantes del equipo técnico. | |
| Descripción: Prueba para la funcionalidad de eliminar información de los integrantes del equipo técnico. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para eliminar información sobre los integrantes del equipo técnico. Se utilizará un usuario con datos válidos. Debe existir información de los integrantes del equipo técnico publicada en el sistema. | |

Entradas/Pasos de ejecución:

Se intenta eliminar información de los integrantes del equipo técnico por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.

Resultado esperado: La información de los integrantes del equipo técnico es eliminada sin generar error.

Evaluación de la prueba: Prueba satisfactoria.

Para ver las pruebas de aceptación realizadas en las restantes iteraciones consultar anexo 4, 5 y 6 respectivamente.

3.5 Conclusiones

Se realizaron las tareas asociadas a las HU del sistema, lo que permitió ayudar a los programadores durante la implementación del mismo. Fueron elaboradas las tarjetas CRC de los módulos desarrollados, lo que permitió representar el diseño de los mismos. El sistema fue validado permitiendo un código de alta calidad y su aceptación por parte del cliente.

Conclusiones

Se automatizó el proceso de gestión de la información deportiva del boxeo existente en la Escuela Cubana de Boxeo, lo que permitió lograr una mayor seguridad, facilidad y rapidez en la manipulación de la misma. Con el desarrollo de esta investigación se realizó un estudio de las principales tendencias para el desarrollo de sistemas que permiten gestionar información deportiva, esto posibilitó seleccionar las herramientas adecuadas para la implementación exitosa del portal. Como producto de este trabajo se obtuvo una aplicación Web altamente configurable que permite la publicación de información del boxeo. La versión del sistema obtenido se validó, demostrando su utilidad y factibilidad, lo que constituye una base para futuras versiones del sistema.

Recomendaciones

Como parte del proceso de investigación y desarrollo de la aplicación, surgen ideas que sería recomendable tener en cuenta para el futuro mejoramiento del sistema, a continuación se listan las mismas:

- ✓ Agregar nuevas funcionalidades al sistema de acuerdo con las expectativas que surjan en la Escuela Cubana de Boxeo.
- ✓ Realizar el manual de usuario de la aplicación.

Referencias bibliográficas

1. SUAREZ, R. *Escuela Cubana de Boxeo* [Consultado el: 5 de diciembre de 2009]. Disponible en: <http://cubaenfotos.blogcip.cu/2008/10/16/escuela-cubana-de-boxeo-boxeo-cuba/>.
2. GUZMÁN, C. L. *El World Wide Web* [Consultado el: 7 de diciembre de 2009]. Disponible en: http://www.bibliodgsca.unam.mx/tesis/tes7cllg/sec_34.htm.
3. GÓMEZ, J. C. G. *Portales de internet: concepto, tipología básica y desarrollo* [Consultado el: 16 de diciembre de 2009]. Disponible en: <http://aprendeenlinea.udea.edu.co/lms/moodle/file.php/87/documentos/TipologiasPortales.pdf>.
4. BARZANALLANA, R. *Capítulo 2. IAGP 2005/06. Metodologías de desarrollo de software* [Consultado el: 20 de enero de 2010]. Disponible en: <http://www.um.es/docencia/barzana/IAGP/Iagp2.html>.
5. CANÓS, J. H. *Metodologías Ágiles en el Desarrollo de Software* [Consultado el: 20 de enero de 2010]. Disponible en: <http://www.willydev.netdescargasprevTodoAgil.pdf>.
6. MONTEJANO, G. *Metodologías Ágiles* [Consultado el: 20 de enero de 2010]. Disponible en: <http://sel.unsl.edu.ar/ApuntesMaes/2004/Metodologias%20Agiles.doc>.
7. HUARACHI, M. *Método ágil ASD* [Consultado el: 20 de enero de 2010]. Disponible en: <http://www.mex.tl/images/18149/informe.doc>.
8. HERNÁN, S. M. *Diseño de una metodología ágil de desarrollo de software* [Consultado el: 20 de enero de 2010]. Disponible en: <http://materias.fi.uba.ar/7500/scheone-tesisdegradoingenieriainformatica.pdf>.
9. SOLÍS, M. C. *Una explicación de la programación extrema (XP)* [Consultado el: 20 de enero de 2010]. Disponible en: <http://www.willydev.net/descargas/prev/ExplicaXp.pdf>.
10. MATÍAS, I. A. C. *UMLGEC ++: Una Herramienta CASE para la Generación de Código a partir de Diagramas de Clase UML* [Consultado el: 21 de enero de 2010]. Disponible en: http://www.utm.mx/~caff/doc/CASE_UML_ANIEI2003.pdf.
11. GSINNOVA, G. D. S. *Rational Rose Data Modeler* [Consultado el: 21 de enero de 2010]. Disponible en: <http://www.rational.com.ar/herramientas/rosedatamodeler.html>.
12. CARVAJAL, A. A. *Lenguaje Unificado de Modelado* [Consultado el: 21 de enero de 2010]. Disponible en: <http://www.google.com.cu/url?sa=t&source=web&ct=res&cd=4&ved=0CBMQFjAD&url=http%3A%2F%2Fsyacomputadores.googlepages.com%2FLenguajeUnificadodeModeladoUML.doc&rct=j&q=rational+rose+herramienta+de+modelado+caracteristicas&ei=-4dpS7aZFsue8AaYvfSrBw&usq=AFQjCNG71V1H8qOzlxVxWm1kgFt2MCmlg>.
13. ALTAMIRANO, A. V. *Comparativo de Entornos de Desarrollo Integrados* [Consultado el: 21 de enero de 2010]. Disponible en: <http://ubicuos.com/wp-content/uploads/2009/05/comparatvoides.pdf>.
14. GIRALDO, L. *HERRAMIENTAS DE DESARROLLO DE INGENIERIA DE SW PARA LINUX* [Consultado el: 21 de enero de 2010]. Disponible en:

-
- http://hugolopez.phi.com.co/docs/download/file=Giraldo-Zapata-Herramientas%20de%20ISW.pdf_id=17.
15. LEYTON, E. *Ingeniería de Software con UML* [Consultado el: 21 de enero de 2010]. Disponible en: <http://www.eduardoleyton.com/apuntes/Uml.pdf>.
 16. ARREGUI, M. *Tutorial de UML* [Consultado el: 21 de enero de 2010]. Disponible en: http://www.google.com/cu/url?sa=t&source=web&ct=res&cd=1&ved=0CAcQFjAA&url=http%3A%2F%2Fwww.seis.es%2Fseis%2Finforsalud04%2F2004_Inforsalud_TutorialUML-UP.doc&rct=j&q=vocabulario+de+UML&ei=16BpS-r5K9WzIAeKysmiCA&usq=AFQjCNHcMw2fJS1YgYynQDpOmXLsr1hvJQ.
 17. LOBOS, M. E. D. *Aprende a programar* [Consultado el: 25 de enero de 2010]. Disponible en: <http://www.mailxmail.com/curso-aprende-programar/concepto-lenguaje-programacion>.
 18. GONZÁLEZ, F. *Nuevos lenguajes para la red* [Consultado el: 25 de enero de 2010]. Disponible en: <http://trevinca.ei.uvigo.es/~pcuesta/publicaciones/asp.pdf>.
 19. ALVAREZ, M. A. *Qué es Python* [Consultado el: 25 de enero de 2010]. Disponible en: <http://www.desarrolloweb.com/articulos/1325.php>.
 20. VALDIVIESO, P. A. C. *Tutoriales elaborados para cursos* [Consultado el: 25 de enero de 2010]. Disponible en: <http://atc.ugr.es/pedro/tutoriales/php/>.
 21. K, I. *Base de datos y PHP* [Consultado el: 25 de enero de 2010]. Disponible en: <http://www.tuxteno.com/archivo/2003/06/base-de-datos-y-php/>.
 22. DIZ, A. J. B. *Entorno de Desarrollo Integrado SUN FORTE for JAVA 3.0* [Consultado el: 29 de enero de 2010]. Disponible en: <http://www-gsi.dec.usc.es/~alberto/fdp/practicas/SunForte/ForteSun.pdf>.
 23. VALDÉS, D. P. *Editores web que facilitan tu trabajo* [Consultado el: 29 de enero de 2010]. Disponible en: <http://www.radiocaribe.co.cu/Secundaria/Informatica/342.htm>.
 24. GÓMEZ, J. G. *Introducción y Conceptos Básicos para la Administración y de Bases de Datos Relacionales* [Consultado el: 6 de febrero de 2010]. Disponible en: http://www.lcc.uma.es/~bds/adminbd/apuntes/ABD1_Intro.pdf.
 25. GIL, F. *SISTEMAS DE GESTIÓN DE BASE DE DATOS SGBD / DBMS* [Consultado el: 6 de febrero de 2010]. Disponible en: <http://alfa.facyt.uc.edu.ve/computacion/pensum/cs0347/download/exposiciones/1/SGBD.pdf>.
 26. ALMA, E. T. *MySQL* [Consultado el: 6 de febrero de 2010]. Disponible en: <http://www.uaem.mx/posgrado/mcruz/cursos/miic/MySQL.pdf>.
 27. MACHACA, R. *Introducción al ORACLE* [Consultado el: 6 de febrero de 2010]. Disponible en: <http://www.angelfire.com/ult/lupa/revista/001/oracle1.html>.
 28. MARTINEZ, R. *Sobre PostgreSQL* [Consultado el: 6 de febrero de 2010]. Disponible en: http://www.postgresql-es.org/sobre_postgresql.
 29. IBERSOLUCIONES, G. *¿Qué es un sistema de gestión de contenidos?* [Consultado el: 10 de febrero de 2010]. Disponible en: <http://www.ibersoluciones.com/que-es-un-sistema-de-gesti-n-de-contenidos.html>.

-
30. GERSBACH, M. *Principales características de Drupal* [Consultado el: 10 de febrero de 2010]. Disponible en: <http://www.factoriadigital.com/ir.php/a/aplicacionesweb/drupal>.
31. RIOSEAS, M. A. *Gestor de Contenidos. Con Joomla CMS sólo debe ocuparse de la información que desee publicar, ya que el sistema gestionará todos los demás detalles técnicos y administrativos* [Consultado el: 10 de febrero de 2010]. Disponible en: http://sedeck.org/web/index.php?option=com_content&task=view&id=35&Itemid=43
32. ALEJANDRO, J. *Crea sitios web de manera sencilla con Mambo CMS* [Consultado el: 3 de marzo de 2010]. Disponible en: <http://www.acercadeinternet.com/crea-sitios-web-de-manera-sencilla-con-mambo-cms/>.
33. REYNOSO, C. *De los métodos heterodoxos en la construcción de software* [Consultado el: 30 de marzo de 2010]. Disponible en: <http://www.willydev.net/Descargas/Heterodoxia.pdf>.
34. PÉREZ, D. A. *Software Libre* [Consultado el: 14 de abril de 2010]. Disponible en: http://ocw.uoc.edu/computer-science-technology-and-multimedia/software-engineering-in-free-software-environments/software-engineering-in-free-software-environments/XP06_M2112_01486.pdf.
35. GRACIA, B. D. *Arquitectura de Drupal* [Consultado el: 20 de abril de 2010]. Disponible en: <http://www.comunic-art.com/portal/index.php/tutorial-drupal/arquitectura-de-drupal.html>.
36. HUALLPARA, M. *Análisis y diseño de sistemas II* [Consultado el: 10 de abril de 2010 de 2010]. Disponible en: http://www.google.com/cu/url?sa=t&source=web&ct=res&cd=6&ved=0CBMQFjAF&url=http%3A%2F%2Fvirtual.usalesiana.edu.bo%2Fweb%2Fpractica%2Farchiv%2Fdespliegue.doc&rct=j&q=+diagrama+de+despliegue&ei=aITWS-XpLYbq9QTjv_G_Dw&usq=AFQjCNGu7s_JDdYCEwEmbKynxa2XI2Q2Q.
37. ESCRIBANO, G. F. *Introducción a Extreme Programming* [Consultado el: 15 de abril de 2010]. Disponible en: <http://www.info-ab.uclm.es/asignaturas/42551/trabajosAnteriores/Trabajo-XP.pdf>.
38. SANCHEZ, M. A. M. *Metodologías De Desarrollo De Software* [Consultado el: 15 de abril de 2010]. Disponible en: <http://www.willydev.net/Descargas/cualmetodologia.pdf>.
39. TOLEILLO. *Pruebas Unitarias en Drupal* [Consultado el: 15 de mayo de 2010]. Disponible en: <http://www.tratonera.com/?q=blog/pruebas-unitarias-en-drupal-unit-testing>.
40. RIVA, C. D. L. *Actas de los Talleres de Ingeniería del Software y Bases de Datos* [Consultado el: 17 de abril de 2010]. Disponible en: <http://www.sistedes.es/TJISBD/Vol-3/No-4/PRIS09.pdf>.

Bibliografía

ÁLVAREZ, I. F. Lenguajes y paradigmas de programación, 2006. [2010]. Disponible en:

http://www.google.com/cu/url?sa=t&source=web&ct=res&cd=20&ved=0CCUQFjAJOAo&url=http%3A%2F%2Fwww.dis.eafit.edu.co%2Fdepto%2Fcolegios%2Factas%2FLeng_Progr.ppt&rct=j&q=lenguajes+de+programacion+caracteristicas&ei=kN1qS9_eKomflAfqiaDZBA&usq=AFQjCNG7zj8c-BsVq88hN0qMA-Jkv1yMzw

ALVAREZ, R. Introducción a la programación en PHP, 2003. [2010]. Disponible en:

<http://www.desarrolloweb.com/articulos/303.php>

AULKE, G. Zend Studio y Zend Platform en un entorno de equipo, 2008. [2010].

Disponible en:

http://www.google.com/cu/url?sa=t&source=web&ct=res&cd=49&ved=0CBsQFjAIOCg&url=http%3A%2F%2Fwww.concatel.com%2Fpfw_files%2Fcma%2Fweb%2Fdocs%2Fdocumentation%2Fdatasheet_zendplatform.pdf&rct=j&q=Zend+Studio+caracteristicas&ei=TipwS9KfGcjL8QaTkaSoAQ&usq=AFQjCNE5_NJ2fEC_zQEIzihRuhzY0M0JmQ

BAÑULS, E. C. Migración Oracle/Informix, 2009. [2010]. Disponible en:

<http://summit.solidq.com/Diapositivas/Rel%20009%20Migraci%C3%B3n%20OracleInformix.pdf>

BLANCO, R. G. LESE-2 Introducción a Rational Rose, 2000. [2010]. Disponible en:

<http://www.google.com/cu/url?sa=t&source=web&ct=res&cd=3&ved=0CA4QFjAC&url=http%3A%2F%2Fwww.essi.upc.edu%2F~ese%2Fweb%2Fdocuments%2Flab%2F0304Q2%2Flessons%2Flese-2%2FLESE-2%2520-%2520Introduccion%2520a%2520Rational%2520Rose.ppt&rct=j&q=Rational+rose+caracteristicas&ei=V4ZpS4zHDY2wIAfdieCNCA&usq=AFQjCNEz64ZXV5AtmvkH1QHjN5pJT-mhdA>

CARLOS. Drupal vs. Joomla: una comparativa sincera de un consultor de IBM, 2009.

[2010]. Disponible en: <http://carlos.rinconsanchez.com/drupal-vs-joomla-una-comparativa-sincera-de-un-consultor-de-ibm>

CASTILLO, C. C. Python, 2008. [2010]. Disponible en:

<http://www.it.uc3m.es/spickin/docencia/comsoft/presentations/spanish/doc/Python.pdf>

-
- CORNEJO, J. E. G. ¿Qué es UML?, 2003. [2010]. Disponible en:
<http://www.docirs.cl/uml.htm>
- DENZER, P. PostgreSQL, 2002. [2010]. Disponible en:
<http://profesores.elo.utfsm.cl/~agv/elo330/2s02/projects/denzer/informe.pdf>
- EXPÓSITO, F. B. Introducción a Python, 2005. [2010]. Disponible en:
<http://lsi.ugr.es/~pdo/Seminarios/Python2005.pdf>
- GERSBACH, M. Principales características de Drupal, 2005. [2010]. Disponible en:
<http://www.factoriadigital.com/ir.php/a/aplicacionesweb/drupal>
- GIRALDO, L. HERRAMIENTAS DE DESARROLLO DE INGENIERIA DE SW PARA LINUX, 2005. [2010]. Disponible en:
http://hugolopez.phicom.co/docs/download/file=Giraldo-Zapata-Herramientas%20de%20ISW.pdf,_id=17
- LOBOS, M. E. D. Capítulo 4: Concepto de lenguaje de programación, 2005. [2010]. Disponible en: <http://www.mailxmail.com/curso-aprende-programar/concepto-lenguaje-programacion>
- SOLARTE, D. S. M. Bases de datos con software libre, 2005. [2010]. Disponible en:
<http://www.umanizales.edu.co/programs/ingenieria/Ventana/ventana12/articulo25.pdf>
- TOUDERT, D. La accesibilidad a la World Wide Web: Hacia un enfoque de evaluación integral, 2002. [2010]. Disponible en:
<http://www.ceaqi.org/toudert/Varios/cibersociedad.pdf>

Glosario de términos

GPL: Es una licencia pública general de GNU o más conocida por su nombre en inglés *General Public License*. Está orientada principalmente a proteger la libre distribución, modificación y uso de *software*. Su propósito es declarar que el *software* bajo esta licencia es libre, protegiéndolo de intentos de apropiación que limiten esas libertades a los usuarios.

Ajax: Acrónimo de “*Asynchronous JavaScript + XML*” (en español JavaScript asíncrono y XML), no es una tecnología. Es realmente una técnica de desarrollo Web para crear aplicaciones interactivas o RIA (*Rich Internet Applications*). Éstas se ejecutan en el cliente, es decir, en el navegador de los usuarios y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la interactividad, velocidad y usabilidad en la misma.

API: Del inglés *Application Programming Interface* - Interfaz de Programación de Aplicaciones es el conjunto de funciones y procedimientos (o métodos si se refiere a programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro *software* como una capa de abstracción.

Hooks: Del inglés gancho, palabra usada en la comunidad de Drupal para nombrar los métodos de la interfaz que deben implementar los módulos desarrollados para este CMS.

XML: Acrónimo de “*Extensible Markup Language*” (en español Lenguaje de marcas extensible), es un metalenguaje extensible de etiquetas desarrollado por el *World Wide Web Consortium* (W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML). XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades. Algunos de los lenguajes que usan XML para su definición son XHTML, SVG y MathML.

HTML: *HyperText Transmission Protocol*, lenguaje (protocolo) que se utiliza para especificar la estructura de documentos que se visualizan a través de internet usando programas de despliegue.

Blogs: Sitio Web periódicamente actualizado que recopila cronológicamente textos o artículos de uno o varios autores, apareciendo primero el más reciente, donde el autor conserva siempre la libertad de dejar publicado lo que crea pertinente.

Código abierto: Es una tendencia internacional del desarrollo de *software* que proclama la distribución del código junto a las aplicaciones, se rige por licencias tales como GNU/GPL.

Plug-in: Son programas que se agregan a otros ya existentes para ofrecer una nueva funcionalidad. Estos programas no funcionarían de forma independiente, necesitan que ya esté instalado otro programa concreto en el ordenador.

Metodología: Hace referencia al conjunto de procedimientos basados en principios lógicos, utilizados para alcanzar una gama de objetivos que rigen en una investigación científica.

Artefacto: Un artefacto es un producto tangible resultante del proceso de desarrollo de *software*. Algunos artefactos ayudan a la descripción de la función, la arquitectura o el diseño del *software*. Otros se enfocan en el proceso de desarrollo, como planes de proyecto, casos de negocios o enfoque de riesgos.

Internautas: Personas que navegan por internet.

Software libre: *Software* que puede ser distribuido, modificado, redistribuido, copiado y usado libremente. Se basa en cuatro libertades: libertad para usarlo con cualquier propósito, libertad para modificarlo a las necesidades, libertad para distribuir copias y mejorarlo. Que un *software* sea libre no quiere decir que sea gratuito, error que viene de la traducción *Free Software*.

RUP: El Proceso Unificado de Desarrollo (*Rational Unified Process* en inglés, habitualmente resumido como RUP) es un proceso de desarrollo de *software*.

Herramientas CASE: Las herramientas CASE (*Computer Aided Software Engineering*, Ingeniería de *Software* Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de *software* reduciendo el coste en términos de tiempo y dinero.

Multiplataforma: Es un término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de *software*, que puedan funcionar en diversas plataformas.

Trigger: Un *trigger* (o disparador) en una base de datos, es un procedimiento que se ejecuta cuando se cumple una condición establecida al realizar una operación de inserción (*INSERT*), actualización (*UPDATE*) o borrado (*DELETE*).

Anexos

Anexo 1

Tareas abordadas para dar solución a las HU de la 2da iteración.

Tabla 51: Tarea #1 de la HU entrevistas

| Tarea | |
|--|--|
| Número de tarea: 1 | HU: 2 |
| Nombre de la tarea: crear la tabla perteneciente al módulo entrevistas. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 15 de marzo del 2010 | Fecha fin: 15 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se establecerá un archivo de tipo <i>.install</i> donde se creará y destruirá la tabla <i>tb_entrevistas</i> para permitir la persistencia de datos. | |

Tabla 52: Tarea #2 de la HU entrevistas

| Tarea | |
|---|--|
| Número de tarea: 2 | HU: 2 |
| Nombre de la tarea: crear las funcionalidades del módulo entrevistas. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 15 de marzo del 2010 | Fecha fin: 16 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: implementar funcionalidades para la gestión de las entrevistas. Se implementará un archivo <i>.module</i> con la ayuda de diversos <i>hooks</i> . | |

Tabla 53: Tarea #3 de la HU entrevistas

| Tarea | |
|---|--|
| Número de tarea: 3 | HU: 2 |
| Nombre de la tarea: implementar la interfaz de entrevistas. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 16 de marzo del 2010 | Fecha fin: 16 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |

Descripción: se crean bloques y formularios para mostrar la información de las entrevistas. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla.

Tabla 54: Tarea #1 de la HU calendario competitivo

| Tarea | |
|--|--|
| Número de tarea: 1 | HU: 4 |
| Nombre de la tarea: implementar la interfaz de calendario competitivo. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 1.0 |
| Fecha de inicio: 17 de marzo del 2010 | Fecha fin: 19 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean bloques para mostrar la información del calendario. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

Tabla 55: Tarea #1 de la HU Ranking

| Tarea | |
|--|--|
| Número de tarea: 1 | HU: 7 |
| Nombre de la tarea: implementar la interfaz de ranking. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.6 |
| Fecha de inicio: 22 de marzo del 2010 | Fecha fin: 24 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crea el menú y los formularios para mostrar la información del ranking. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

Tabla 56: Tarea #1 de la HU resultados

| Tarea | |
|---|--|
| Número de tarea: 1 | HU: 13 |
| Nombre de la tarea: implementar la interfaz de resultados. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 1.0 |
| Fecha de inicio: 24 de marzo del 2010 | Fecha fin: 25 de marzo del 2010 |

| |
|---|
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides |
| Descripción: se crean los bloques y los formularios para mostrar la información de los resultados. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. |

Tabla 57: Tarea #1 de la HU imágenes

| Tarea | |
|---|--|
| Número de tarea: 1 | HU: 18 |
| Nombre de la tarea: implementar la interfaz de imágenes. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.4 |
| Fecha de inicio: 26 de marzo del 2010 | Fecha fin: 29 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean los bloques y los formularios para mostrar las imágenes. Se configuran los permisos para que los usuarios invitados puedan ver las imágenes. | |

Tabla 58: Tarea #1 de la HU videos

| Tarea | |
|---|--|
| Número de tarea: 1 | HU: 19 |
| Nombre de la tarea: implementar la interfaz de videos. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.4 |
| Fecha de inicio: 30 de marzo del 2010 | Fecha fin: 31 de marzo del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean los bloques y los formularios para mostrar los videos. Se configuran los permisos para que los usuarios invitados puedan ver los videos. | |

Anexo 2.**Tareas abordadas para dar solución a las HU de la 3ra iteración.**

Tabla 59: Tarea #1 de la HU material de archivo

| Tarea | |
|--|--------------|
| Número de tarea: 1 | HU: 8 |
| Nombre de la tarea: implementar la interfaz de material de archivo. | |

| | |
|--|--|
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 12 de abril del 2010 | Fecha fin: 12 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean los bloques y los formularios para mostrar los materiales de descarga. Se configuran los permisos para que los usuarios invitados puedan ver los materiales publicados. | |

Tabla 60: Tarea #1 de la HU artículos

| Tarea | |
|--|--|
| Número de tarea: 1 | HU: 14 |
| Nombre de la tarea: crear la tabla perteneciente al módulo artículos. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 13 de abril del 2010 | Fecha fin: 14 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se establecerá un archivo de tipo <i>.install</i> donde se creará y destruirá la tabla <i>tb_articulos</i> para permitir la persistencia de datos. | |

Tabla 61: Tarea #2 de la HU artículos

| Tarea | |
|---|--|
| Número de tarea: 2 | HU: 14 |
| Nombre de la tarea: crear las funcionalidades del módulo artículos. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 14 de abril del 2010 | Fecha fin: 15 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: implementar funcionalidades para la gestión de los artículos. Se implementará un archivo <i>.module</i> con la ayuda de diversos <i>hooks</i> . | |

Tabla 62: Tarea #3 de la HU artículos

| Tarea | |
|--|---------------|
| Número de tarea: 3 | HU: 14 |
| Nombre de la tarea: implementar la interfaz de artículos. | |

| | |
|---|--|
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 15 de abril del 2010 | Fecha fin: 15 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crea el menú y los formularios para mostrar la información de los artículos. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

Tabla 63: Tarea #1 de la HU combates del día

| Tarea | |
|---|--|
| Número de tarea: 1 | HU: 23 |
| Nombre de la tarea: implementar la interfaz de combates del día. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.6 |
| Fecha de inicio: 16 de abril del 2010 | Fecha fin: 16 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean los bloques y los formularios para mostrar la información de los combates. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

Tabla 64: Tarea #1 de la HU presidencia

| Tarea | |
|---|--|
| Número de tarea: 1 | HU: 24 |
| Nombre de la tarea: implementar la interfaz de presidencia. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 19 de abril del 2010 | Fecha fin: 19 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean los bloques y los formularios para mostrar la información de la presidencia. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

Anexo 3.**Tareas abordadas para dar solución a las HU de la 4ta iteración.**

Tabla 65: Tarea #1 de la HU otros sitios

| Tarea | |
|--|--|
| Número de tarea: 1 | HU: 6 |
| Nombre de la tarea: crear la tabla perteneciente al módulo vínculos. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 19 de abril del 2010 | Fecha fin: 19 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se establecerá un archivo de tipo <i>.install</i> donde se creará y destruirá la tabla <i>tb_vinculo</i> para permitir la persistencia de datos. | |

Tabla 66: Tarea #2 de la HU otros sitios

| Tarea | |
|--|--|
| Número de tarea: 2 | HU: 6 |
| Nombre de la tarea: crear las funcionalidades del módulo vínculos. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 19 de abril del 2010 | Fecha fin: 20 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: Implementar funcionalidades para la gestión de los sitios. Se implementará un archivo <i>.module</i> con la ayuda de diversos <i>hooks</i> . | |

Tabla 67: Tarea #3 de la HU otros sitios

| Tarea | |
|---|--|
| Número de tarea: 3 | HU: 6 |
| Nombre de la tarea: implementar la interfaz de vínculos. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 20 de abril del 2010 | Fecha fin: 20 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean los bloques y los formularios para mostrar el nombre y la url de los sitios. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

Tabla 68: Tarea #1 de la HU concursos

| Tarea | |
|--|--|
| Número de tarea: 1 | HU: 20 |
| Nombre de la tarea: crear la tabla perteneciente al módulo concursos. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 21 de abril del 2010 | Fecha fin: 21 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se establecerá un archivo de tipo <i>.install</i> donde se creará y destruirá la tabla <i>tb_concursos</i> para permitir la persistencia de datos. | |

Tabla 69: Tarea #2 de la HU concursos

| Tarea | |
|---|--|
| Número de tarea: 2 | HU: 20 |
| Nombre de la tarea: crear las funcionalidades del módulo concursos. | |
| Tipo de tarea: desarrollo | Puntos de estimación: 0.2 |
| Fecha de inicio: 21 de abril del 2010 | Fecha fin: 22 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: implementar funcionalidades para la gestión de los concursos. Se implementará un archivo <i>.module</i> con la ayuda de diversos <i>hooks</i> . | |

Tabla 70: Tarea #3 de la HU concursos

| Tarea | |
|--|--|
| Número de tarea: 3 | HU: 20 |
| Nombre de la tarea: implementar la interfaz de concursos. | |
| Tipo de tarea: configuración-desarrollo | Puntos de estimación: 0.1 |
| Fecha de inicio: 22 de abril del 2010 | Fecha fin: 22 de abril del 2010 |
| Programador responsable: Pedro Castillo Urquiza-Mario González Benavides | |
| Descripción: se crean los bloques y los formularios para mostrar la información de los concursos. Se configuran los permisos para que los usuarios invitados puedan ver la información pero no modificarla. | |

Anexo 4.

Pruebas de aceptación realizadas en la 2da iteración.

Tabla 71: Prueba a la funcionalidad publicar entrevistas

| Caso de prueba de Aceptación | |
|---|-------------------------------|
| Código: HU2_P1 | Historia de Usuario: 2 |
| Nombre: Publicar entrevistas en el sistema. | |
| Descripción: Prueba para la funcionalidad de publicar entrevistas en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para publicar entrevistas en el sistema. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: Se intenta publicar entrevistas por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Las entrevistas son publicadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 72: Prueba a la funcionalidad modificar entrevistas

| Caso de prueba de Aceptación | |
|---|-------------------------------|
| Código: HU2_P2 | Historia de Usuario: 2 |
| Nombre: Modificar entrevistas en el sistema. | |
| Descripción: Prueba para la funcionalidad de modificar entrevistas en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para modificar entrevistas en el sistema. Se utilizará un usuario con datos válidos. Deben existir entrevistas publicadas en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta modificar entrevistas por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Las entrevistas son modificadas sin generar error. | |

Evaluación de la prueba: Prueba satisfactoria.

Tabla 73: Prueba a la funcionalidad eliminar entrevistas

| Caso de prueba de Aceptación | |
|---|-------------------------------|
| Código: HU2_P3 | Historia de Usuario: 2 |
| Nombre: Eliminar entrevistas en el sistema. | |
| Descripción: Prueba para la funcionalidad de eliminar entrevistas en el sistema. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para eliminar entrevistas en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Deben existir entrevistas publicadas en el sistema.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta eliminar entrevistas por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: Las entrevistas son eliminadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 74: Prueba a la funcionalidad manejar calendario competitivo

| Caso de prueba de Aceptación | |
|--|-------------------------------|
| Código: HU4_P1 | Historia de Usuario: 4 |
| Nombre: Manejar calendario de eventos. | |
| Descripción: Prueba a la funcionalidad manejar calendario de eventos. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para manejar el calendario.</p> <p>Se utilizará un usuario con datos válidos.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta manejar el calendario de eventos por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |

Resultado esperado: El calendario de eventos es manejado sin generar error.

Evaluación de la prueba: Prueba satisfactoria.

Tabla 75: Prueba a la funcionalidad publicar ranking

| Caso de prueba de Aceptación | |
|---|-------------------------------|
| Código: HU7_P1 | Historia de Usuario: 7 |
| Nombre: Publicar ranking. | |
| Descripción: Prueba a la funcionalidad publicar ranking. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para publicar el ranking en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta publicar el ranking por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: El ranking es publicado sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 76: Prueba a la funcionalidad modificar ranking

| Caso de prueba de Aceptación | |
|---|-------------------------------|
| Código: HU7_P2 | Historia de Usuario: 7 |
| Nombre: Modificar ranking. | |
| Descripción: Prueba a la funcionalidad modificar ranking. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para modificar el ranking en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Debe existir algún ranking publicado.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta modificar el ranking por un usuario que contiene sus datos válidos, es decir,</p> | |

| |
|---|
| el administrador u otro rol con privilegios definidos por el administrador. |
| Resultado esperado: El ranking es modificado sin generar error. |
| Evaluación de la prueba: Prueba satisfactoria. |

Tabla 77: Prueba a la funcionalidad crear roles

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU12_P1 | Historia de Usuario: 12 |
| Nombre: Crear roles. | |
| Descripción: Prueba a la funcionalidad crear roles. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para crear roles. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: Se intenta crear roles por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los roles son creados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 78: Prueba a la funcionalidad crear cuenta de usuario

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU12_P2 | Historia de Usuario: 12 |
| Nombre: Crear cuentas de usuarios. | |
| Descripción: Prueba a la funcionalidad crear cuentas de usuarios. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para crear cuentas de usuarios. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: Se intenta crear cuentas de usuarios por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Las cuentas de usuarios son creadas sin generar error. | |

Evaluación de la prueba: Prueba satisfactoria.

Tabla 79: Prueba a la funcionalidad eliminar cuenta de usuario

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU12_P3 | Historia de Usuario: 12 |
| Nombre: Eliminar cuentas de usuarios. | |
| Descripción: Prueba a la funcionalidad eliminar cuentas de usuarios. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para eliminar cuentas de usuarios. Se utilizará un usuario con datos válidos. Deben de existir cuentas de usuarios en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta eliminar cuentas de usuarios por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Las cuentas de usuarios son eliminadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 80: Prueba a la funcionalidad listar usuarios registrados

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU12_P4 | Historia de Usuario: 12 |
| Nombre: Listar usuarios registrados. | |
| Descripción: Prueba a la funcionalidad listar usuarios registrados. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para listar los usuarios registrados Se utilizará un usuario con datos válidos. Deben de existir cuentas de usuarios en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta listar los usuarios registrados por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |

| |
|--|
| Resultado esperado: El listado de usuarios registrados es mostrado sin generar error. |
| Evaluación de la prueba: Prueba satisfactoria. |

Tabla 81: Prueba a la funcionalidad modificar cuenta de usuario

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU12_P5 | Historia de Usuario: 12 |
| Nombre: Modificar cuentas de usuarios. | |
| Descripción: Prueba a la funcionalidad modificar cuentas de usuarios. | |
| Condiciones de ejecución: El usuario debe estar autenticado. El usuario debe estar autenticado y tener permisos para modificar las cuentas de usuarios. Tienen que existir cuentas de usuarios en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta modificar las cuentas de usuarios por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Las cuentas de usuarios son modificadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 82: Prueba a la funcionalidad publicar resultados

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU13_P1 | Historia de Usuario: 13 |
| Nombre: Publicar resultados de una competencia. | |
| Descripción: Prueba a la funcionalidad publicar resultados de una competencia. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para publicar los resultados. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: | |

| |
|--|
| Se intenta publicar los resultados de una competencia por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. |
| Resultado esperado: Los resultados son publicados sin generar error. |
| Evaluación de la prueba: Prueba satisfactoria. |

Tabla 83: Prueba a la funcionalidad modificar resultados

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU13_P2 | Historia de Usuario: 13 |
| Nombre: Modificar resultados de una competencia. | |
| Descripción: Prueba a la funcionalidad modificar resultados de una competencia. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para modificar los resultados. Se utilizará un usuario con datos válidos. Deben existir resultados de competencias publicados en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta modificar los resultados de una competencia por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los resultados son modificados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 84: Prueba a la funcionalidad eliminar resultados

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU13_P3 | Historia de Usuario: 13 |
| Nombre: Eliminar resultados de una competencia. | |
| Descripción: Prueba a la funcionalidad eliminar resultados de una competencia. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para eliminar los resultados. Se utilizará un usuario con datos válidos. | |

| |
|--|
| Deben existir resultados de competencias publicados en el sistema. |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta eliminar los resultados de una competencia por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> |
| <p>Resultado esperado: Los resultados son eliminados sin generar error.</p> |
| <p>Evaluación de la prueba: Prueba satisfactoria.</p> |

Tabla 85: Prueba a la funcionalidad publicar imágenes

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU18_P1 | Historia de Usuario: 18 |
| Nombre: Publicar imágenes en el sistema. | |
| Descripción: Prueba a la funcionalidad publicar imágenes en el sistema. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para publicar imágenes en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta publicar imágenes en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: Las imágenes son publicadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 86: Prueba a la funcionalidad eliminar imágenes

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU18_P2 | Historia de Usuario: 18 |
| Nombre: Eliminar imágenes en el sistema. | |
| Descripción: Prueba a la funcionalidad eliminar imágenes en el sistema. | |
| Condiciones de ejecución: | |

| |
|---|
| <p>El usuario debe estar autenticado y tener permisos para eliminar imágenes en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Deben existir imágenes publicadas en el sistema.</p> |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta eliminar imágenes en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> |
| <p>Resultado esperado: Las imágenes son eliminadas sin generar error.</p> |
| <p>Evaluación de la prueba: Prueba satisfactoria.</p> |

Tabla 87: Prueba a la funcionalidad publicar videos

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU19_P1 | Historia de Usuario: 19 |
| Nombre: Publicar videos en el sistema. | |
| Descripción: Prueba a la funcionalidad publicar videos en el sistema. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para publicar videos en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta publicar videos en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: Los videos son publicados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 88: Prueba a la funcionalidad eliminar videos

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU19_P2 | Historia de Usuario: 19 |
| Nombre: Eliminar videos en el sistema. | |
| Descripción: Prueba a la funcionalidad eliminar videos en el sistema. | |

| |
|---|
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para eliminar videos en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Deben existir videos publicados en el sistema.</p> |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta eliminar videos en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> |
| <p>Resultado esperado: Los videos son eliminados sin generar error.</p> |
| <p>Evaluación de la prueba: Prueba satisfactoria.</p> |

Anexo 5.

Pruebas de aceptación realizadas en la 3ra iteración.

Tabla 89: Prueba a la funcionalidad publicar material de archivo

| Caso de prueba de Aceptación | |
|--|-------------------------------|
| Código: H8_P1 | Historia de Usuario: 8 |
| Nombre: Publicar material de archivo en el sistema. | |
| Descripción: Prueba a la funcionalidad publicar material de archivo en el sistema. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para publicar material de archivo.</p> <p>Se utilizará un usuario con datos válidos.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta publicar material de archivo en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: Los materiales de archivos son publicados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 90: Prueba a la funcionalidad eliminar material de archivo

| Caso de prueba de Aceptación | |
|------------------------------|-------------------------------|
| Código: H8_P2 | Historia de Usuario: 8 |

| |
|---|
| Nombre: Eliminar material de archivo del sistema. |
| Descripción: Prueba a la funcionalidad eliminar material de archivo en el sistema. |
| Condiciones de ejecución: <p>El usuario debe estar autenticado y tener permisos para eliminar material de archivo.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Deben existir materiales de archivo publicados.</p> |
| Entradas/Pasos de ejecución: <p>Se intenta eliminar material de archivo del sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> |
| Resultado esperado: Los materiales de archivos son eliminados sin generar error. |
| Evaluación de la prueba: Prueba satisfactoria. |

Tabla 91: Prueba a la funcionalidad publicar artículos

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: H14_P1 | Historia de Usuario: 14 |
| Nombre: Publicar artículos en el sistema. | |
| Descripción: Prueba a la funcionalidad publicar artículos en el sistema. | |
| Condiciones de ejecución: <p>El usuario debe estar autenticado y tener permisos para publicar artículos en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> | |
| Entradas/Pasos de ejecución: <p>Se intenta publicar artículos en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: Los artículos son publicados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 92: Prueba a la funcionalidad modificar artículos

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: H14_P2 | Historia de Usuario: 14 |
| Nombre: Modificar artículos en el sistema. | |
| Descripción: Prueba a la funcionalidad modificar artículos en el sistema. | |
| Condiciones de ejecución: <p>El usuario debe estar autenticado y tener permisos para modificar artículos en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Deben existir artículos publicados en el sistema.</p> | |
| Entradas/Pasos de ejecución: <p>Se intenta modificar los artículos en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: Los artículos son modificados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 93: Prueba a la funcionalidad eliminar artículos

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: H14_P3 | Historia de Usuario: 14 |
| Nombre: Eliminar artículos en el sistema. | |
| Descripción: Prueba a la funcionalidad eliminar artículos en el sistema. | |
| Condiciones de ejecución: <p>El usuario debe estar autenticado y tener permisos para eliminar artículos en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Deben existir artículos publicados en el sistema.</p> | |
| Entradas/Pasos de ejecución: <p>Se intenta eliminar artículos en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el</p> | |

| |
|--|
| administrador. |
| Resultado esperado: Los artículos son eliminados sin generar error. |
| Evaluación de la prueba: Prueba satisfactoria. |

Tabla 94: Prueba a la funcionalidad publicar pronósticos

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: H23_P1 | Historia de Usuario: 23 |
| Nombre: Publicar pronósticos de los eventos en el sistema. | |
| Descripción: Prueba a la funcionalidad publicar pronósticos de los eventos en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para publicar pronósticos en el sistema. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: Se intenta publicar pronósticos de los eventos en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los pronósticos son publicados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 95: Prueba a la funcionalidad modificar pronósticos

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: H23_P2 | Historia de Usuario: 23 |
| Nombre: Modificar pronósticos de los eventos en el sistema. | |
| Descripción: Prueba a la funcionalidad modificar pronósticos de los eventos en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para modificar pronósticos en el sistema. Se utilizará un usuario con datos válidos. | |

| |
|---|
| Deben existir pronósticos publicados en el sistema. |
| Entradas/Pasos de ejecución: Se intenta modificar los pronósticos de los eventos en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. |
| Resultado esperado: Los pronósticos son modificados sin generar error. |
| Evaluación de la prueba: Prueba satisfactoria. |

Tabla 96: Prueba a la funcionalidad eliminar pronósticos

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: H23_P3 | Historia de Usuario: 23 |
| Nombre: Eliminar pronósticos de los eventos en el sistema. | |
| Descripción: Prueba a la funcionalidad eliminar pronósticos de los eventos en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado y tener permisos para eliminar pronósticos en el sistema. Se utilizará un usuario con datos válidos. Deben existir pronósticos publicados en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta eliminar pronósticos de los eventos en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los pronósticos son eliminados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 97: Prueba a la funcionalidad publicar presidencia

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: H24_P1 | Historia de Usuario: 24 |
| Nombre: Publicar presidencia en el sistema. | |
| Descripción: Prueba a la funcionalidad publicar presidencia en el sistema. | |

| |
|---|
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para publicar la presidencia en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta publicar la presidencia en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> |
| <p>Resultado esperado: La presidencia es publicada sin generar error.</p> |
| <p>Evaluación de la prueba: Prueba satisfactoria.</p> |

Tabla 98: Prueba a la funcionalidad modificar presidencia

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: H24_P2 | Historia de Usuario: 24 |
| Nombre: Modificar presidencia en el sistema. | |
| Descripción: Prueba a la funcionalidad modificar presidencia en el sistema. | |
| <p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado y tener permisos para modificar la presidencia en el sistema.</p> <p>Se utilizará un usuario con datos válidos.</p> <p>Debe estar publicada la presidencia en el sistema.</p> | |
| <p>Entradas/Pasos de ejecución:</p> <p>Se intenta modificar la presidencia en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador.</p> | |
| Resultado esperado: La presidencia es modificada sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Anexo 6.

Pruebas de aceptación realizadas en la 4ta iteración.

Tabla 99: Prueba a la funcionalidad publicar enlaces con otros sitios

| Caso de prueba de Aceptación | |
|--|-------------------------------|
| Código: HU6_P1 | Historia de Usuario: 6 |
| Nombre: Publicar enlaces con otros sitios en el sistema. | |
| Descripción: Prueba a la funcionalidad modificar enlaces con otros sitios en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: Se intenta publicar enlaces con otros sitios en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los enlaces con otros sitios son publicados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 100: Prueba a la funcionalidad modificar enlaces con otros sitios

| Caso de prueba de Aceptación | |
|---|-------------------------------|
| Código: HU6_P2 | Historia de Usuario: 6 |
| Nombre: Modificar enlaces con otros sitios en el sistema. | |
| Descripción: Prueba a la funcionalidad modificar enlaces con otros sitios en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado. Se utilizará un usuario con datos válidos. Deben existir enlaces con otros sitios publicados en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta modificar enlaces con otros sitios en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los enlaces con otros sitios son modificados sin generar error. | |

Evaluación de la prueba: Prueba satisfactoria.

Tabla 101: Prueba a la funcionalidad eliminar enlaces con otros sitios

| Caso de prueba de Aceptación | |
|--|-------------------------------|
| Código: HU6_P3 | Historia de Usuario: 6 |
| Nombre: Eliminar enlaces con otros sitios del sistema. | |
| Descripción: Prueba a la funcionalidad eliminar enlaces con otros sitios en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado. Se utilizará un usuario con datos válidos. Deben existir enlaces con otros sitios publicados en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta eliminar enlaces con otros sitios en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los enlaces con otros sitios son eliminados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 102: Prueba a la funcionalidad realizar búsquedas

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU11_P1 | Historia de Usuario: 11 |
| Nombre: Realizar búsquedas en el sistema. | |
| Descripción: Prueba a la funcionalidad realizar búsquedas en el sistema. | |
| Condiciones de ejecución: | |
| Entradas/Pasos de ejecución: Se intenta realizar una búsqueda por parte de un usuario. | |
| Resultado esperado: Si lo que se busca es encontrado, se devuelve, de lo contrario se hace saber al usuario que la información deseada no fue encontrada. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 103: Prueba a la funcionalidad publicar concursos

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU20_P1 | Historia de Usuario: 20 |
| Nombre: Publicar concursos en el sistema. | |
| Descripción: Prueba a la funcionalidad publicar concursos en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: Se intenta publicar concursos en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los concursos son publicados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 104: Prueba a la funcionalidad modificar concursos

| Caso de prueba de Aceptación | |
|--|--------------------------------|
| Código: HU20_P2 | Historia de Usuario: 20 |
| Nombre: Modificar concursos en el sistema. | |
| Descripción: Prueba a la funcionalidad modificar concursos en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado. Se utilizará un usuario con datos válidos. Deben existir concursos publicados en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta modificar concursos en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los concursos son modificados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 105: Prueba a la funcionalidad eliminar concursos

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU20_P3 | Historia de Usuario: 20 |
| Nombre: Eliminar concursos del sistema. | |
| Descripción: Prueba a la funcionalidad eliminar concursos del sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado. Se utilizará un usuario con datos válidos. Deben existir concursos publicados en el sistema. | |
| Entradas/Pasos de ejecución: Se intenta eliminar concursos del sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Los concursos son eliminados sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Tabla 106: Prueba a la funcionalidad publicar encuestas

| Caso de prueba de Aceptación | |
|---|--------------------------------|
| Código: HU21_P1 | Historia de Usuario: 31 |
| Nombre: Publicar encuestas en el sistema. | |
| Descripción: Prueba a la funcionalidad publicar encuestas en el sistema. | |
| Condiciones de ejecución: El usuario debe estar autenticado. Se utilizará un usuario con datos válidos. | |
| Entradas/Pasos de ejecución: Se intenta publicar encuestas en el sistema por un usuario que contiene sus datos válidos, es decir, el administrador u otro rol con privilegios definidos por el administrador. | |
| Resultado esperado: Las encuestas son publicadas sin generar error. | |
| Evaluación de la prueba: Prueba satisfactoria. | |

Anexo 7
Historias de Usuario.

Tabla 107: Representación de la HU #2

| Historia de Usuario | |
|--|---------------------------------|
| Número: 2 | Nombre: Entrevistas |
| Usuario: administrador , invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 2 |
| Prioridad en Negocio: media (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se publican entrevistas relacionadas con los boxeadores cubanos en entrenamientos y eventos competitivos. El administrador podrá insertar, modificar y eliminar las entrevistas. El invitado podrá acceder a las entrevistas publicadas. | |
| Observaciones: | |

Tabla 108: Representación de la HU #4

| Historia de Usuario | |
|--|--|
| Número: 4 | Nombre: Calendario competitivo |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 2 |

| | |
|--|---------------------------------|
| Prioridad en Negocio: Media (Alta / Media / Baja) | Puntos Estimados: 1.0 |
| Riegos en Desarrollo: Bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se mostrará el calendario del mes actual, donde se resaltan los días en que se desarrollarán los eventos competitivos, al dar clic en uno de esos días se mostrará el nombre del evento que se desarrollará, el país sede y una breve descripción del mismo. El administrador podrá insertar, modificar y mostrar los eventos competitivos. El invitado podrá acceder al calendario competitivo. | |
| Observaciones: | |

Tabla 109: Representación de la HU #5

| Historia de Usuario | |
|--|---------------------------------|
| Número: 5 | Nombre: Autenticar |
| Usuario: administrador | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 4 |
| Prioridad en Negocio: baja (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: el administrador debe autenticarse para realizar cambios en la aplicación. Una vez autenticado tiene la opción de cambiar su contraseña en caso de que lo desee. | |
| Observaciones: si es necesario se le puede dar permisos a un usuario para que tenga privilegios sobre determinada información. | |

Tabla 110: Representación de la HU #6

| Historia de Usuario | |
|--|-------------------------------------|
| Número: 6 | Nombre: Otros sitios |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 4 |
| Prioridad en Negocio: baja (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: permite mostrar enlaces con otros sitios de interés relacionados con el boxeo. El administrador publicará enlaces con otros sitios dedicados al boxeo. El invitado podrá acceder a los enlaces publicados. | |
| Observaciones: pueden ser sitios nacionales e internacionales. | |

Tabla 111: Representación de la HU #7

| Historia de Usuario | |
|---|----------------------------------|
| Número: 7 | Nombre: Ranking |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 2 |
| Prioridad en Negocio: media (Alta / Media / Baja) | Puntos Estimados: 0.6 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: el administrador podrá insertar, modificar y mostrar el ranking | |

| |
|---|
| ofreciendo el lugar que ocupa, país, nombre del atleta y división en la que se compite. El invitado podrá ver el ranking. |
| Observaciones: el ranking puede ser a nivel nacional o internacional. |

Tabla 112: Representación de la HU #8

| Historia de Usuario | |
|--|---------------------------------------|
| Número: 8 | Nombre: Material de archivo |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 3 |
| Prioridad en Negocio: media (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: permitirá la publicación de investigaciones realizadas por los integrantes del colectivo técnico y de cualquier otro material que pueda interesar al invitado para ser descargado. Estos podrán ser insertados, mostrados y eliminados por el administrador. | |
| Observaciones: los materiales se renovarían periódicamente. | |

Tabla 113: Representación de la HU #8

| Historia de Usuario | |
|--|------------------------------------|
| Número: 9 | Nombre: Administrar foro |
| Usuario: administrador | |
| Modificación de la Historia de Usuario: | Iteración |

| | |
|--|---------------------------------|
| | Asignada: 3 |
| Prioridad en Negocio: media (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se administrará el foro de modo que el administrador tiene la posibilidad de adicionar categorías al foro, editarlas y eliminarlas. Además puede adicionar un tema de discusión, editarlo y eliminarlo, así como adicionar las respuestas a los temas publicados, modificarlas y eliminarlas. El administrador podrá editar el foro y eliminar esta sección. | |
| Observaciones: | |

Tabla 114: Representación de la HU #10

| Historia de Usuario | |
|---|---------------------------------|
| Número: 10 | Nombre: Foro |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 3 |
| Prioridad en Negocio: media (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: le brinda la posibilidad al portal Web de contar con una sección de foro debate, donde se mostrará una lista de los temas activos, los invitados podrán enviar opiniones de un tema, permitirá mostrar el foro con o sin nuevos envíos desde la última visita a decisión del usuario. | |

Observaciones: el administrador revisa y acepta los envíos de los usuarios antes de ser publicados.

Tabla 115: Representación de la HU # 11

| Historia de Usuario | |
|---|--------------------------------------|
| Número: 11 | Nombre: Realizar búsquedas |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 4 |
| Prioridad en Negocio: baja (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: brinda la posibilidad de realizar búsquedas dentro del sitio. | |
| Observaciones: se podrá refinar la búsqueda especificando categorías, palabras y frases. | |

Tabla 116: Representación de la HU #12

| Historia de Usuario | |
|--|--|
| Número: 12 | Nombre: Administrar usuarios y roles |
| Usuario: administrador | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 2 |
| Prioridad en Negocio: media | Puntos Estimados: |

| | |
|--|-----------------------|
| (Alta / Media / Baja) | 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: el administrador puede crear roles y cuentas de usuario para así brindar una serie de privilegios a las cuentas en dependencia del rol asignado a las mismas. Además éste puede modificar los datos de las cuentas de usuarios, así como eliminarlas y mostrar un listado de todos los usuarios registrados. | |
| Observaciones: | |

Tabla 117: Representación de la HU #13

| Historia de Usuario | |
|---|---------------------------------|
| Número: 13 | Nombre: Resultados |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 2 |
| Prioridad en Negocio: media (Alta / Media / Baja) | Puntos Estimados: 1.0 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se mostrarán los resultados competitivos alcanzados por la escuadra cubana en los diferentes torneos nacionales e internacionales. También se hará una breve reseña del torneo, resaltando datos significativos del mismo. El administrador podrá insertar, modificar y mostrar los resultados competitivos. El invitado podrá acceder a los resultados publicados. | |
| Observaciones: los contenidos se clasificarán en resultados nacionales e internacionales. | |

Tabla 118: Representación de la HU #14

| Historia de Usuario | |
|--|---------------------------------|
| Número: 14 | Nombre: Artículos |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 3 |
| Prioridad en Negocio: alta (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se publicarán artículos sobre temas del boxeo. El administrador podrá insertar, modificar y eliminar los mismos. El invitado podrá acceder a los artículos publicados. | |
| Observaciones: | |

Tabla 119: Representación de la HU #18

| Historia de Usuario | |
|---|---------------------------------|
| Número: 18 | Nombre: Imágenes |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 2 |
| Prioridad en Negocio: media (Alta / Media / Baja) | Puntos Estimados: 0.4 |
| Riegos en Desarrollo: bajo | Puntos |

| | |
|---|----------------|
| (Alto / Medio / Bajo) | Reales: |
| Descripción: se mostrarán fotos tomadas en los diferentes eventos competitivos y entrenamientos. El administrador podrá insertar, modificar y mostrar las fotos. El invitado podrá acceder a las fotos publicadas y enviar comentarios sobre las mismas. | |
| Observaciones: Las fotos se mostrarán en miniatura y se podrán ver en su tamaño normal. | |

Tabla 120: Representación de la HU #19

| Historia de Usuario | |
|---|---------------------------------|
| Número: 19 | Nombre: Videos |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 2 |
| Prioridad en Negocio: media (Alta / Media / Baja) | Puntos Estimados: 0.4 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se publicarán videos tomados en eventos competitivos y en los entrenamientos. El administrador podrá insertar, modificar y mostrar los videos. El invitado podrá acceder a los videos publicados. | |
| Observaciones: | |

Tabla 121: Representación de la HU #20

| Historia de Usuario | |
|--|-----------------------------|
| Número: 20 | Nombre: Concursos |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración |

| | |
|---|---------------------------------|
| | Asignada: 4 |
| Prioridad en Negocio: baja (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se publicarán concursos sobre temas del boxeo cubano. El administrador escoge los temas de los concursos y establece la fecha de entrega. El invitado podrá ver los concursos publicados y una vez realizado el concurso lo mandará por correo electrónico. | |
| Observaciones: Los archivos podrán ser con extensiones .doc y .pdf. | |

Tabla 122: Representación de la HU #21

| Historia de Usuario | |
|---|---------------------------------|
| Número: 21 | Nombre: Encuestas |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 4 |
| Prioridad en Negocio: baja (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se publicarán encuestas sobre temas del boxeo cubano. El administrador escoge los temas de las encuestas. El invitado podrá ver las encuestas publicadas y votar según su criterio. | |
| Observaciones: se publicarán 3 encuestas, las cuales serán renovadas | |

mensualmente.

Tabla 123: Representación de la HU #23

| Historia de Usuario | |
|--|------------------------------------|
| Número: 23 | Nombre: Combates del día |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 3 |
| Prioridad en Negocio: media (Alta / Media / Baja) | Puntos Estimados: 0.6 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se publicarán los pronósticos de los diferentes torneos según especialistas, así como los resultados del evento una vez desarrollado. El administrador podrá insertar, modificar y mostrar información de los diferentes torneos. El invitado podrá acceder a los pronósticos y resultados publicados. | |
| Observaciones: los pronósticos serán organizados por divisiones, especificando el pronóstico mediante una imagen que recoge los datos. | |

Tabla 124: Representación de la HU #24

| Historia de Usuario | |
|--|---------------------------------|
| Número: 24 | Nombre: Presidencia |
| Usuario: administrador, invitado | |
| Modificación de la Historia de Usuario: | Iteración Asignada: 3 |

| | |
|--|---------------------------------|
| Prioridad en Negocio: media (Alta / Media / Baja) | Puntos Estimados: 0.2 |
| Riegos en Desarrollo: bajo (Alto / Medio / Bajo) | Puntos Reales: |
| Descripción: se publicará la foto del presidente de la Escuela Cubana de Boxeo. El administrador podrá insertar, modificar y mostrar información sobre el presidente de la Escuela Cubana de Boxeo. El invitado podrá acceder a la información publicada. | |
| Observaciones: | |