

**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
FACULTAD 4**



**SISTEMA INFORMÁTICO DE GESTIÓN DEL FÓRUM DE CIENCIA Y TÉCNICA
DE LA FACULTAD 4**

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

Autor: José Antonio Sardina López

Tutores: MSc. Arlenys Palmero Ortega
Ing. Yisel Quiñones Guerrero

La Habana, junio de 2015

“Año 57 de la Revolución”

Declaración de Autoría

Declaro que soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a que haga el uso que estime pertinente con el mismo.

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

Autor:

José Antonio Sardina López

Tutores:

Ing. Yisel Quiñones Guerrero

MSc. Arlenys Palmero Ortega

Dedicatoria

*Dedico mi tesis a mi familia por el apoyo incondicional en todos estos años de la carrera.
A mi mamá por ser una luchadora incansable y haberme dado todo su amor incondicional todo este tiempo.*

A mi papá por haberme enseñado tantas cosas de la vida y forjarme como un hombre de principios.

A mi hermano niñito por ser mi ejemplo a seguir y por estar siempre a mi lado en los momentos malos y buenos.

A mi hermano Euker por ser hermano y amigo, por estar siempre a mi lado.

A mi hermano kiki por siempre estar al tanto de mí y ser el más pequeño del equipo.

A mi tía Mami y mi tío Ché por todo su cariño y amor incondicional.

A mi abuela Fidelia por siempre estar al tanto de mí.

A mi abuelo Chelo, que aunque ya no esté físicamente le agradezco todo sus consejos.

Agradecimientos

Agradezco a Dios por haberme dado las fuerzas que necesitaba para cumplir con mi meta. A mi familia por encima de todas las cosas, por siempre estar a mi lado y haberse sacrificado tanto por mí. A todas mis amistades de la UCI por estar siempre en los momentos buenos y malos para darme su apoyo. A mis hermanos Poll, Andrés y Maikel por haber sido más que amigos en esta universidad. Agradezco a mi novia Yuliet por estar siempre a mi lado cuando más lo necesitaba. A todo el que de una manera u otra siempre estuvo pendiente de cómo estaban mis estudios.

Resumen

La Universidad de las Ciencias Informáticas, como parte de la estrategia de informatización de la gestión de eventos científicos, emplea diferentes herramientas para automatizar los principales procesos de estos eventos. La presente investigación surge al existir dificultades en como se realizan los procesos de convocatoria al evento, inscripción, banco de problemas, envío de ponencias y dictamen proporcionado por las comisiones del evento en la Facultad 4. El objetivo de la investigación fue desarrollar un sistema que contribuya a la gestión del Fórum de Ciencia y Técnica de la Facultad 4. Para llevarlo a cabo, se realizó un estudio del estado del arte sobre sistemas similares, tanto en el ámbito nacional como internacional. Se desarrolló una herramienta para la Gestión del Fórum de Ciencia y Técnica de la Facultad 4, fue guiada por la metodología de desarrollo Programación Extrema. Como lenguajes de programación del lado del cliente se utilizaron HTML, JavaScript y CSS. Como lenguaje de programación se utilizó PHP, a través del CMS Drupal 7. Se utilizó el NetBeans IDE como Entorno de Desarrollo Integrado, Apache como servidor web y PostgreSQL como gestor de bases de datos. Se determinaron e implementaron todas las funcionalidades que debía tener el sistema, se realizaron las pruebas unitarias y de aceptación, donde se obtuvieron resultados favorables. El sistema informático creado es de gran importancia para la Facultad 4, pues realiza la automatización del proceso de gestión del Fórum de Ciencia y Técnica de la Facultad 4 garantizando que el evento de desarrolle con mayor calidad y eficiencia.

Palabras clave: eventos, gestión, procesos.

Índice

Introducción..... 1

Capítulo 1 Fundamentación Teórica.....5

1.1. Conceptos asociados al dominio de la investigación.5

1.2. Análisis de soluciones existentes 6

1.3. Metodología de desarrollo.8

1.4. Aplicaciones Web12

1.5. Frameworks de desarrollo web14

1.6. Sistemas de Gestión de Contenidos15

1.7. Lenguaje de programación17

 1.7.1 Lenguaje del lado del servidor18

 1.7.2 Lenguaje del lado del cliente.....19

1.8 Entorno de desarrollo integrado IDE20

1.9 Herramienta CASE de modelado con UML21

1.10 Servidor de Base de Datos23

1.11 Servidor Web25

Conclusiones del capítulo.....25

Capítulo 2: Descripción de la propuesta de solución.....27

2.1 Objeto de informatización27

2.2 Diagrama conceptual del negocio28

2.3 Definición de los roles en el sistema29

2.4 Exploración.....32

 2.4.1 Historia de usuario32

 2.4.2 Historias de usuario de la propuesta de solución33

2.5 Estimación de esfuerzos por HU35

2.6 Planificación de las iteraciones.....35

 2.6.1 Iteración 1.36

2.6.2 Iteración 2.	36
2.6.3 Iteración 3.	36
2.7 Plan de duración de las iteraciones	36
2.8 Plan de entregas	37
2.9 Tarjetas Clase-Responsabilidades-Colaboradores	38
Conclusiones del capítulo.....	39
Capítulo 3: Implementación y Pruebas.....	40
3.1 Patrón arquitectónico de Drupal	40
3.2 Estructura de Drupal.....	41
3.2.1 Tema.....	43
3.2.2 Usuarios y permisos	44
3.2.3 Bloques y menús	45
3.2.4 Módulos	46
3.3 Tareas de Ingeniería.....	49
3.4 Pruebas.....	52
3.4.1 Pruebas unitarias	53
3.4.2 Pruebas de aceptación	55
Conclusiones del capítulo.....	61
Conclusiones generales.....	62
Recomendaciones.....	63
Referencias bibliográficas.....	64
Anexos.....	69
Anexo 1 Historias de usuarios	69
Anexo 2 Tareas de ingeniería para la iteración 1	74
Anexo 3 Tareas de ingeniería para la iteración 2.....	81
Anexo 4 Tareas de ingeniería para la iteración 3.....	88
Anexo 5 Casos de prueba de aceptación para la primera entrega.	96
Anexo 6 Casos de prueba de aceptación para la segunda entrega.....	107

Anexo 7 Casos de prueba de aceptación para la tercera entrega.118
Anexo 8 Entrevista utilizada.....128

Introducción

La sociedad actual impulsada por el avance científico tecnológico y apoyado en el uso generalizado de las tecnologías de la información y las comunicaciones (TIC), ha experimentado cambios que alcanzan diferentes esferas. El hombre se ha visto en la obligación de crear de nuevos instrumentos, herramientas y métodos novedosos como apoyo a sus actividades fundamentales. La necesidad de buscar nuevas vías de desarrollo y soluciones a los problemas se ha convertido en un medio indispensable para las instituciones, permitiendo ayudar, ahorrar trabajo y agilizar los procesos sustantivos. Es por esto que en Cuba se aplican estrategias que tienen como fin encontrar soluciones prácticas a problemas cotidianos y se cuenta en este sentido con un grupo de eventos y espacios que promueven el desarrollo científico técnico, como es el caso del Fórum de Ciencia y Técnica (FCT). Este evento se desarrolla en todas las instituciones del país, tiene como objetivo fundamental estimular la generación de soluciones e innovaciones que respondan a las carencias de recursos y a las limitaciones para mantener la producción y los servicios. Además tiene gran importancia para la gestión de información y persigue evitar la duplicidad de esfuerzos, la sustitución de importaciones y soportar las investigaciones. (CIGET, 2006)

El FCT surge a finales de 1980, debido a la fuerte crítica realizada por el Comandante en Jefe Fidel Castro Ruz por la poca atención brindada en el quinquenio 76-80 a las piezas de repuesto y los enormes gastos en divisas convertibles en que incurrió el país por las importaciones de piezas. Desde ese momento Cuba impulsa la participación masiva en la búsqueda de soluciones que resuelvan los problemas (CIGET-VC & IDICT, 2015). Mediante la realización del FCT se fomenta el desarrollo tecnológico y el uso de las TIC para incrementar la producción y la investigación científica, es por ello que cuenta con el apoyo de instituciones como la Universidad de las Ciencias Informáticas (UCI).

Una de las tareas fundamentales que ha llevado a cabo la Universidad consiste en la creación de espacios donde estudiantes, trabajadores y profesores, puedan exponer el resultado de su trabajo, la producción y la investigación. En la UCI se desarrollan varios eventos que promueven la actividad científica productiva, como son la Jornada Científica Estudiantil, Mi Web por Cuba, Fórum de Historia, Seminario Juvenil Martiano, entre otros, pero sin duda uno de los más importantes es el Fórum de Ciencia y Técnica. El FCT se realiza en varios niveles, primero a nivel de Facultad, luego a nivel de Universidad, municipal, provincial y por último a nivel nacional, con el objetivo de resolver los problemas desde la base.

En la Universidad, específicamente en la Facultad 4, se han desarrollado varias ediciones del Fórum de Ciencia y Técnica. Desde la primera edición hasta la fecha se han presentado una gran cantidad de trabajos, que reflejan la labor científica y la productividad de la Facultad y del Centro de Tecnologías para la Formación FORTES.

El FCT tiene una comisión científica y un comité organizador, que se encargan de la ejecución del evento. Publicar la convocatoria, circular la documentación, recepcionar los trabajos, crear las comisiones y mantener a los participantes informados del cronograma, son actividades fundamentales del fórum. Todos los trabajos presentados deben dar solución a los problemas del Banco de problemas de la Facultad y el Centro, pero a pesar de los esfuerzos realizados estas actividades muchas veces no se realizan con la mejor calidad. Debido a las experiencias alcanzadas luego de la realización de diferentes ciclos del evento, se han podido detectar algunas deficiencias durante el proceso de concepción y ejecución. Hoy no se cuenta con el registro de trabajos presentados ni con resultados de ediciones anteriores. La convocatoria, la documentación y la recepción de los trabajos se hacen vía correo electrónico, que no es de la manera más eficiente, esto provoca en ocasiones pérdida de la información y afecta la disponibilidad.

La gestión del evento puede verse afectada en cierta medida por errores humanos como son, no publicar en tiempo el cronograma y las comisiones a las que pertenecen los trabajos, esto provoca la desorientación en los participantes. La inscripción y la recepción de los trabajos son unos de los puntos críticos del problema que se plantea, pues en ocasiones la información de la persona que recepciona los trabajos no llega a todos y complejiza la participación en el evento. Otro factor que se pone de manifiesto es que no existe el escenario para tener el banco de problemas disponible para la consulta de los interesados, lo que trae consigo que no se cumpla el principal objetivo del fórum, que es dar soluciones a problemas reales presentes en la Facultad. Los resultados no se publican en un lugar específico para el conocimiento de todos, esto impide el reconocimiento del trabajo de las personas.

Por lo antes expuesto se propone el siguiente **problema investigativo**: ¿Cómo contribuir a la gestión del Fórum de Ciencia y Técnica de la Facultad 4?

De lo planteado anteriormente se define como **objeto de estudio** el proceso de gestión del Fórum de Ciencia y Técnica.

Determinando para la investigación como **campo de acción** el proceso de gestión del Fórum de Ciencia y Técnica de la Facultad 4.

El **objetivo general** de esta investigación es: desarrollar un sistema informático que contribuya a la gestión del Fórum de Ciencia y Técnica de la Facultad 4.

Del objetivo general se definen los siguientes **objetivos específicos**:

- ✓ Elaborar el marco teórico-conceptual para fundamentar la investigación.
- ✓ Realizar el estudio del proceso de gestión del Fórum de Ciencia y Técnica de la Facultad 4.
- ✓ Diseñar la propuesta de solución que contribuya a la gestión del Fórum de Ciencia y Técnica de la Facultad 4.
- ✓ Desarrollar un sistema de gestión del Fórum de Ciencia y Técnica en la Facultad 4.
- ✓ Validar el funcionamiento del sistema mediante la aplicación de pruebas.

La **Hipótesis** que se desea demostrar es: El desarrollo de un sistema informático para gestionar el Fórum de Ciencia y Técnica de la Facultad 4 contribuirá a una mejor gestión del proceso del evento en el área.

Las **tareas de investigación** planteadas para dar cumplimiento a los objetivos específicos son:

- ✓ Realización de un estudio acerca del funcionamiento actual del proceso de gestión del Fórum de Ciencia y Técnica de la Facultad 4.
- ✓ Realización de un estudio de sistemas similares que incorporan el proceso de gestión de eventos.
- ✓ Selección de las tecnologías, herramientas y la metodología a utilizar para el desarrollo de la solución.
- ✓ Definición de los requisitos funcionales y no funcionales del sistema a desarrollar.
- ✓ Desarrollo de las funcionalidades mediante la metodología de desarrollo seleccionada.
- ✓ Implementación de las funcionalidades de la solución propuesta.
- ✓ Realización de pruebas al sistema desarrollado.

Los **métodos científicos** empleados para el desarrollo de la investigación permitieron un mejor entendimiento del proceso de gestión del FCT en la Facultad 4.

Entre los métodos teóricos que se utilizaron están el **analítico-sintético**, que permitió la extracción de los elementos más importantes que se relacionan con el proceso de gestión del FCT, determinar las características principales de la solución propuesta, así como las tecnologías y herramientas más adecuadas para su desarrollo. La utilización del método **histórico-lógico**, fue orientado a investigar cómo funciona en la actualidad la gestión del FCT en la Facultad 4. Se aplicó para conocer los antecedentes y el estado actual de los sistemas informáticos que se relacionan con la investigación para el análisis del código fuente y funcionalidades.

Se empleó como método empírico **la observación**, para describir y explicar el comportamiento del proceso de gestión del Fórum de Ciencia y Técnica en la facultad 4. Este método se empleó además con el objetivo de determinar las funcionalidades que debe tener la solución propuesta. Otro método empírico empleado fue la **Entrevista** (No estructurada): en este caso se utilizó este método para tener un mejor entendimiento de la situación real del problema y extraer los principales requerimientos que dieron lugar al problema científico en cuestión.

Capítulo 1 Fundamentación Teórica

En el presente capítulo se abordan los elementos que fundamentan la base teórica de la investigación, como los diferentes conceptos, entre los que se encuentran, proceso, gestión, fórum de ciencia y técnica, evento, ciencia y banco de problemas, con el objetivo de comprender el campo de acción. Se hace un estudio de herramientas existentes que se utilizan a nivel internacional, en Cuba y en la UCI para la gestión de eventos. Se analizan y seleccionan las tecnologías, herramientas y metodología que se utilizarán en el desarrollo del sistema propuesto.

1.1. Conceptos asociados al dominio de la investigación.

Para un mayor entendimiento de los conceptos que se utilizan en la investigación es necesario sentar las bases para un mejor dominio del problema, a continuación se mencionan dichos conceptos.

Ciencia: La palabra ciencia es definida como una actividad que produce resultados que se expresan en conocimientos y debe ser concebida como una práctica social que está dirigida a la producción, difusión y aplicación de conocimientos (León, 2002). En el contexto del problema analizado, este concepto se encuentra presente desde el momento que un participante en el evento comienza el estudio de las herramientas que necesitará para darle solución a un problema dado, pasando por una investigación que es corroborada con un resultado.

Fórum de Ciencia y Técnica: Uno de los conceptos fundamentales asociados al objeto y campo de acción de esta investigación es el de Fórum de Ciencia y Técnica, el mismo se definió como *“un Movimiento Político Ideológico con conceptos y principios definidos para integrar a las masas en la misión de buscar soluciones útiles que resuelvan los problemas cotidianos, que impiden cumplir con los Objetivos y Prioridades de las entidades, mediante la aplicación adecuada de la Ciencia y la Técnica”*. (CIGET-VC & IDICT, 2015)

Evento científico: Durante la investigación se utiliza en varias ocasiones el término evento científico que se define en (Real Academia Española, 2015) como una eventualidad, hecho imprevisto o que puede acaecer. También es conocido como un suceso importante y programado, de índole social o académica.

En el contexto del problema investigativo queda definido como parte de un proceso que se inicia en la organización de las investigaciones, pasa por una producción científica y continúa con un proceso de divulgación como una vía para introducir los resultados en la práctica social.

Problema: Un problema es un determinado asunto o una cuestión que requiere de una solución. A nivel social, se trata de alguna situación en concreto que en el momento en que se logra solucionar, aporta beneficios a la sociedad (Definición.de, 2015). En el presente trabajo se define como cada una de los problemas encontrados en los departamentos de la Facultad 4 y en el centro FORTES, que no han sido publicados en un lugar específico.

Proceso: Un proceso es una colección de actividades que son realizadas de forma coordinada en un ambiente técnico y organizacional. La conjunción de estas actividades logra un objetivo específico del negocio (Weske, 2007). En el contexto del problema en cuestión se enmarca en las actividades creadas y organizadas por la Facultad 4 en aras de que la realización del FCT se realice con la mejor calidad posible.

Gestión: Significa ejecutar determinadas acciones con el objetivo de lograr un éxito con medios adecuados. Es nombrado como la acción y efecto de realizar tareas con cuidado, esfuerzo y eficacia que conduzcan a una finalidad (eumendnet, 2015). De la investigación se refiere al proceso de llevar a cabo diligencias que hacen posible la realización de los FCT, lograda con el manejo de los medios que dispone la Facultad 4.

1.2. Análisis de soluciones existentes

Debido a la importancia que se le confiere a los eventos científicos, hoy existen varias herramientas creadas con el objetivo de apoyar el proceso de gestión y control de los mismos. Es por ello que se decide hacer un análisis de varios sistemas desarrollados a nivel nacional e internacional con el objetivo de determinar las principales características y funcionalidades presentes en estos sistemas similares. Así como las causas que originaron la creación de los mismos, sus ventajas y desventajas que puedan favorecer el desarrollo del sistema. Las soluciones varían desde las más generales hasta las más específicas, a continuación se muestran algunas de ellas:

Sitio de Investigaciones de la UCI

El sitio de Investigaciones de la UCI tiene como objetivo fundamental realizar la gestión de varios de los eventos científicos realizados en la Universidad ya sea FCT, Jornada Científica Estudiantil, entre otros. El sitio posibilita llevar una gestión de eventos de manera general. No gestiona los eventos científicos a nivel de Facultad, por lo que no posee un banco de problemas actualizados por departamentos de cada Facultad. No existe una manera de acceder a los trabajos realizados en ediciones anteriores de los eventos. No es posible realizar el desarrollo del FCT de la Facultad 4 pues el sistema no brinda la posibilidad de dar permisos a terceros para realizar operaciones de gestión de eventos. Todo el trabajo del sitio de Investigaciones de la UCI tiene como prioridad organizar las investigaciones en la Universidad de manera general y brindar informaciones de interés a los investigadores. El sitio se encuentra disponible aunque los últimos servicios relacionados con el FCT no se actualizan desde el año 2014, lo que provoca desorientación por parte de la comunidad universitaria. (Dirección de Investigaciones, 2015)

Portal de Congresos de la UNLP

El portal de Congresos de la UNLP es un servicio que brinda la Universidad de la Plata en Argentina para la organización y planificación de cualquier tipo de reunión científica-académica. Se engloba al término Congreso a cualquier evento científico dígame Conferencia, Mesa de Debate, Simposio. La aplicación brinda la posibilidad de realizar la inscripción en el evento, posee un calendario del evento a desarrollar y un rango de espacio de tiempo en el que se deben enviar los trabajos para su revisión. El servicio es exclusivo para las personas que son miembros de la UNLP, no es posible que otras personas no miembros de la UNLP puedan favorecerse de sus servicios principales. (Plata, 2015)

CENCOMED

El Centro Virtual de Convenciones de Salud (CENCOMED) es una aplicación informática cuya función principal es brindar servicios que faciliten la promoción y gestión de eventos científicos del sistema de salud cubano. Brinda información actualizada al personal de salud, acerca de los eventos, congresos y otras actividades científico-académicas relacionadas con las ciencias de la salud. Facilita un espacio para la promoción, a través de la red, a los organizadores de eventos. Aunque el sitio es actualizado de forma gradual, no brinda la posibilidad de dar permisos a otras personas para que realicen la administración de un determinado evento en el mismo. El sitio está programado para la realización de eventos nacionales e internacionales pertenecientes al campo de la salud. No posee un banco de problemas disponible, ni

cuenta con la posibilidad de agregar nuevos problemas para su posible solución (Infomed, 2015). Por todo lo anterior planteado no se considera la aplicación del CENCOMED como una solución viable para dar respuesta al problema investigativo.

El estudio realizado sobre algunos de los sistemas que son utilizados en el proceso de gestión de eventos tanto a nivel nacional como internacional, evidenció que dichos *softwares* fueron contruidos a la medida del problema que resuelven, con poca posibilidad de poder ajustarlos a las necesidades de la problemática planteada. El estudio de estos sistemas arrojó un conjunto de características que pueden proveer a la investigación un valor agregado, entre las que se encuentran:

- ✓ La publicación de la convocatoria y del cronograma del evento.
- ✓ La inscripción en el evento en un rango determinado de fecha.
- ✓ Para que el usuario pueda acceder a todas las funcionalidades debe estar autenticado.
- ✓ La posibilidad de agregar un evento en el sistema.
- ✓ Listar los eventos por fecha.
- ✓ Listar los eventos por categoría.
- ✓ La creación de comisiones.

1.3. Metodología de desarrollo.

Antes de comenzar a desarrollar el sistema es necesario sentar las bases sobre lo que es una metodología de desarrollo y sus aportes al desarrollo de la propuesta de solución. Según expresa (León, 2002) la metodología de la investigación organiza y dirige el proceso de investigación, ayuda a seleccionar los métodos, los procedimientos y las técnicas a emplear en correspondencia con las características de la investigación que se va a realizar.

Existen dos tipos de metodologías, las tradicionales y la ágiles. Las metodologías tradicionales van centradas por lo general en la planificación y el control del proceso. Establece de manera estricta los artefactos a producir, las herramientas y notaciones que se van a utilizar. Por otra parte se encuentran 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. (Letelier y Penadés, 2006)

Capítulo 1: Fundamentación Teórica

La tabla 1 muestra las principales diferencias entre las metodologías ágiles y las tradicionales. (Letelier y Penadés, 2006)

Tabla 1 Principales diferencias entre las metodologías ágiles y las tradicionales.

Metodologías Ágiles	Metodologías Tradicionales
Preparadas para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el equipo)	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas normas.
No existe contrato tradicional, o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (10 integrantes) y trabajando en el mismo sitio	Grupos grandes y con posibilidad de estar distribuidos
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menos énfasis en la arquitectura del <i>software</i>	La arquitectura del <i>software</i> es esencial y se expresa mediante modelos

Después de la comparación establecida, se seleccionan las metodologías ágiles para el desarrollo de la propuesta. La selección se fundamenta con los siguientes parámetros:

- ✓ El equipo de desarrollo es pequeño, en este caso de una sola persona.
- ✓ El período de tiempo de confección del *software* es corto.

Capítulo 1: Fundamentación Teórica

- ✓ El cliente formará parte del equipo de desarrollo.

Entre las principales metodologías ágiles se encuentran XP¹, Scrum, Crystal Methods, ASD². Estas metodologías ponen de relevancia que la capacidad de respuesta a un cambio es más importante que el seguimiento estricto de un plan.

La siguiente tabla compara las distintas aproximaciones ágiles en base a tres parámetros: vista del sistema como algo cambiante, tener en cuenta la colaboración entre los miembros del equipo y características más específicas de la propia metodología como son simplicidad, excelencia técnica, resultados, adaptabilidad. También incorpora como referencia no ágil el CMM³. (Highsmith, 2002)

Tabla 2 Ranking de “agilidad” (Los valores más altos representan una mayor agilidad).

	CMM	ASD	Crystal	DSDM	FDD	LD	Scrum	XP
Sistema como algo cambiante	1	5	4	3	3	4	5	5
Colaboración	2	5	5	4	4	4	5	5
Características Metodologías(CM)								
Resultados	2	5	5	4	4	4	5	5
Simplicidad	1	4	4	3	5	3	5	5
Adaptabilidad	2	5	5	3	3	4	4	3
Excelencia Técnica	4	3	3	4	4	4	3	4
Prácticas de colaboración	2	5	5	4	3	3	4	5
Media CM	2.2	4.4	4.4	3.6	3.8	3.6	4.2	4.4
Media Total	1.7	4.8	4.5	3.6	3.6	3.9	4.7	4.8

¹ Del inglés eXtremeProgramming, en español Programación Extrema.

² Del inglés Adaptive Software Development.

³ Del inglés Capability Maturity Model.

Como se observa en la Tabla 2, todas las metodologías ágiles tienen una significativa diferencia del índice de agilidad respecto a CMM y entre ellas destacan ASD, Scrum y XP como las más ágiles.

SCRUM

Está indicada de forma especial para proyectos con un rápido cambio de requisitos. Sus principales características se pueden resumir en dos. El desarrollo de *software* se realiza mediante iteraciones, denominadas *sprints*, con una duración de 30 días. El resultado de cada *sprint* es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo del proyecto, entre ellas destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración. (Letelier y Penadés, 2006)

ASD

Sus principales características son: iterativo, orientado a los componentes de *software* más que a las tareas y tolerante a los cambios. El ciclo de vida que propone tiene tres fases esenciales: especulación, colaboración y aprendizaje. En la primera de ellas se inicia el proyecto y se planifican las características del *software*, en la segunda se desarrollan las características y por último en la tercera se revisa su calidad y se entrega al cliente. La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo. (Letelier y Penadés, 2006)

Programación extrema

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de *software*, preocupándose por el aprendizaje de los desarrolladores y capaz de propiciar un buen clima de trabajo. Se basa en la 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. XP es adecuada para proyectos con requisitos imprecisos y muy cambiantes.

Un proyecto que utiliza XP tiene éxito cuando el cliente selecciona el valor de negocio a implementar basado en la habilidad del equipo para medir la funcionalidad que puede entregar a través del tiempo. El ciclo de desarrollo consiste (a grandes rasgos) en los siguientes pasos:

- ✓ Paso 1: El cliente define el valor de negocio a implementar.
- ✓ Paso 2: El programador estima el esfuerzo necesario para su implementación.
- ✓ Paso 3: El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
- ✓ Paso 4: El programador construye ese valor de negocio.
- ✓ Paso 5: Vuelve al Paso 1.

Posibles inconvenientes de XP

- ✓ Está dirigida a equipos pequeños o medianos, por lo general menores de 10 personas.
- ✓ El entorno físico debe ser un ambiente que permita la comunicación y colaboración entre todos los miembros del equipo durante todo el tiempo, cualquier resistencia del cliente o del equipo de desarrollo hacia las prácticas y principios puede llevar al proceso al fracaso.
- ✓ El uso de tecnologías que no tengan un ciclo rápido de retroalimentación o que no soporten de manera fácil el cambio puede impedir el éxito. (Letelier y Penadés, 2006)

El autor de la presente investigación determina utilizar XP como metodología para guiar el proceso de desarrollo de la propuesta de solución, pues propone entregas más rápidas. Además XP es flexible a cambios en cualquier momento, de manera que las funcionalidades que se terminan no se apartan, sino que están en todo tiempo expuestas a modificaciones en caso necesario.

1.4. Aplicaciones Web

Con el transcurso del tiempo ha alcanzado un gran desarrollo y se han abierto las puertas al acceso de la información desde cualquier parte del mundo. Por lo tanto las aplicaciones web se han convertido en un desafío para sus desarrolladores, debido a que el modo de generar las páginas ha evolucionado, conjuntamente con la demanda de aplicaciones más rápidas, ligeras y robustas. Ahora surgen aplicaciones dinámicas, basadas en la constante actualización que permiten un alto nivel de personalización, de manera que puedan ser adaptadas a diferentes tipos de usuario. Esto ha encontrado su medio ideal en la web, con el fin de no hacer costoso su mantenimiento y modificación. (Sanz, 2015)

Capítulo 1: Fundamentación Teórica

Sitio Web: Un sitio web es un conjunto de páginas web que están relacionadas entre sí, por lo general porque se ingresan desde un mismo dominio o porque mantienen constante la raíz de la dirección URL. Todos los sitios web accesibles de manera pública constituyen una gigantesca "World Wide Web" de información donde una página web es un documento electrónico que contiene información específica de un tema en particular creado en formato HTML. El lenguaje HTML permite que cualquier persona pueda crear su propia página web sin tener grandes conocimientos de informática. (ALEGSA, 2015)

Muchas son las ventajas que permiten el uso de aplicaciones web, dentro de las que se encuentran: (RAMA, 2015)

No requiere instalar *software* especial (en los clientes): En esencia, para acceder a un *software* web solo se necesita disponer de un navegador de páginas web. Debido a la arquitectura de las aplicaciones web, el navegador suele quedar relegado a mostrar la interfaz de usuario (menús, opciones, formularios) mientras que toda la compleja lógica de negocio se lleva en el lado del servidor.

Bajo coste en actualizar los equipos con una nueva versión: Los navegadores web visualizan las páginas web que son servidas por el servidor web de forma dinámica. En ese sentido, es el servidor quien ejecuta la mayor parte del código de la aplicación y suministra de forma centralizada las vistas (las páginas) a los navegadores conectados. En consecuencia, no hay que instalar nada en los puestos de trabajo, pues la actualización se realiza en el servidor y es vista de manera automática por todos los usuarios. De esta manera se evita que pueda existir algún equipo que ejecute una versión diferente y desactualizada.

Información centralizada: En una aplicación web, la lógica de negocio no es la única que está centralizada en el servidor, sino también los datos que se ubican en una base de datos centralizada en un servidor determinado.

Seguridad y copias de seguridad: Como se dispone de datos centralizados es más fácil establecer y llevar el control de una política de copias de seguridad centralizada. Es más, al no ubicarse los datos en el puesto de trabajo, en caso de robo o incendio, la empresa no ha perdido información y puede desplegar de carácter rápido un nuevo puesto de trabajo.

Movilidad: Es un concepto relativo y dependiente de la implantación concreta. Si el *software* está ubicado en un servidor web en internet o bien disponemos de una intranet externalizada (*extranet*), cualquier usuario con un portátil y una conexión a internet móvil podría acceder a la aplicación.

Reducción de costes en los puestos cliente (mayor longevidad): Debido a que las páginas se ofrecen desde el servidor web el equipo cliente queda relegado a mostrar los resultados y formularios, para lo cual no es necesario un hardware potente en los puestos de trabajo, lo que se traduce en reducción de costes.

El autor de la presente investigación decide utilizar para el desarrollo de la propuesta de solución tecnología de tipo web. La selección se fundamenta en las ventajas y oportunidades mencionadas anteriormente.

1.5. Frameworks de desarrollo web

Un *framework* simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes. Además proporciona estructura al código fuente, obliga al desarrollador a crear códigos más legibles y fáciles de mantener. Un *framework* facilita la programación de aplicaciones debido a que encapsula operaciones complejas en instrucciones sencillas. Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Brinda una estructura metodológica de trabajo la cual extiende o utiliza las aplicaciones del dominio. (Potencier, 2008)

Symfony: *Symfony* es un completo *framework* diseñado para optimizar el desarrollo de las aplicaciones web. Separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permite al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. *Symfony* está caracterizado por:

- ✓ Fácil de instalar y configurar en sistemas Windows, Linux y Mac.
- ✓ Independiente del sistema gestor de bases de datos.
- ✓ Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.

- ✓ Basado en la premisa de "convenir en vez de configurar", en la que el desarrollador solo debe configurar aquello que no es convencional.
- ✓ Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
- ✓ Código fácil de leer que incluye comentarios de y que permite un mantenimiento sencillo.
- ✓ Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros. (Potencier, 2008)

Dada la complejidad del sistema a desarrollar y que la curva de aprendizaje de *Symfony* es alta y que las opciones de administración, asignación de roles y permisos y gestión dinámica de los contenidos tendrían que ser implementadas desde cero, esto incurre en una pérdida considerable de tiempo, cuestiones que con el uso de otra herramienta sería realizado de manera más rápida y eficaz. Por lo anterior planteado no se considera *Symfony* como una opción favorable para el desarrollo de la propuesta de solución.

1.6. Sistemas de Gestión de Contenidos

Un sistema de gestión de contenidos (CMS) *“es un software que permite crear una estructura base para la creación y administración de contenidos, principalmente de páginas web. Generalmente un CMS⁴ es una aplicación con una base de datos asociada en la que se almacenan los contenidos, separados de los estilos o diseño. El CMS controla también quién puede editar y visualizar los contenidos, convirtiéndose en una herramienta de gestión integral para la publicación de sitios web.”* (Gil, 2012)

De los CMS genéricos, algunos de los más utilizados en la actualidad son Joomla y Drupal. Ambos tienen en común que están desarrollados en PHP y MySQL y que se distribuyen como software libre. Cada uno de ellos cuenta con su propia comunidad de usuarios y desarrolladores que contribuyen al desarrollo del proyecto, ya sea mediante el trabajo en la mejora del software o en la creación de nuevos módulos para incrementar o mejorar sus funcionalidades (Gil, 2012). En la presente investigación se aborda sobre ambos CMS.

⁴ Del inglés Content Management Systems.

Joomla

Joomla es un sistema de gestión de contenidos que permite la creación de sitios web dinámicos, es de código abierto y es posible obtenerlo de manera gratuita. Es un CMS muy poderoso, resulta fácil de instalar. Garantiza la posibilidad de poder personalizarse y expandirse según la necesidad del usuario. Muestra una interfaz amigable, lo que favorece el trabajo al desarrollador y al usuario final. Joomla se encuentra disponible bajo la licencia GPL en todas sus versiones. Utiliza PHP como lenguaje de programación, MySQL como sistema gestor de base de datos y Apache como servidor web. Cabe mencionar que gran cantidad de los mejores módulos de Joomla no se encuentran de manera gratuita en la red, existen versiones que no cuentan con gran estabilidad. (Webempresa, 2015)

CMS Drupal

Drupal es un sistema de gestión de contenidos que se distribuye como *software* libre bajo licencia GNU GPL⁵ versión 2 o superior. Drupal puede ser modificado y distribuido de carácter libre, pero siempre se debe hacer bajo una única licencia. Esto quiere decir que sí, por ejemplo, se desarrolla un módulo específico, se debe distribuir con todos sus archivos fuente, de forma que cualquier otra persona pueda a su vez modificarlo y distribuirlo. Es posible compartir un paquete, un módulo o un tema de Drupal desarrollado, sólo se tiene que distribuir bajo la misma licencia, de forma que también puedan ser modificados y redistribuidos por otras personas. El *software* está desarrollado con el lenguaje de programación PHP⁶ y utiliza una base de datos MySQL o PostgreSQL. Está maquetado con hojas de estilo CSS⁷, con lo que es posible construir sitios web totalmente accesibles. Hoy en día casi cualquier proveedor de alojamiento (*hosting*) dispone de las características mínimas requeridas por Drupal para su instalación y correcto funcionamiento. (Ramírez, 2012)

Pueden identificarse otras características que hacen del CMS Drupal una buena selección para la propuesta de solución del presente trabajo.

⁵ Del inglés General Public License.

⁶ Acrónimo recursivo de "PHP: Hypertext Preprocessor".

⁷ Del inglés Cascading Stylesheets.

- ✓ *Ayuda on-line*: Es un robusto sistema de ayuda *on-line* y páginas de ayuda para los módulos del núcleo, tanto para usuarios como para administradores.
- ✓ *Búsqueda*: Todo el contenido en Drupal es indexado en tiempo real y se puede consultar en cualquier momento.
- ✓ *Código abierto*: El código fuente de Drupal está disponible de forma libre bajo los términos de la licencia GNU/GPL. Al contrario de otros sistemas de gestión de contenido propietarios, es posible extender o adaptar Drupal según las necesidades.
- ✓ *Módulos*: La comunidad de Drupal ha contribuido a muchos módulos que proporcionan funcionalidades como 'página de categorías', autenticación mediante jabber, mensajes privados.
- ✓ *Personalización*: Un entorno robusto de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo las preferencias definidas por el usuario.
- ✓ *URLs amigables*: Drupal usa el *mod_rewrite* de Apache para crear URLs que son manejables por los usuarios y los motores de búsqueda (Martin y otros, 2015).

La selección se basa principalmente en los conocimientos que tiene el desarrollador sobre este CMS y en las facilidades que brinda Drupal, lo que favorecería el ahorro de tiempo y recursos en aprender otra tecnología. Cabe mencionar que la comunidad de personas que se dedican al desarrollo de aplicaciones en Drupal en la UCI es grande, lo cual crea un gran cúmulo de conocimientos en la comunidad universitaria alrededor del mismo, que puede ser útil en el desarrollo del *software*. El CMS seleccionado para el desarrollo es Drupal en su versión 7.34.

1.7. Lenguaje de programación

Un lenguaje de programación puede ser usado para controlar el comportamiento de una máquina o una computadora. Se componen de un conjunto de reglas sintácticas y semánticas que permiten expresar instrucciones que luego serán interpretadas (ALEGSA, 2015). A continuación se profundiza en el estudio de los lenguajes de programación a utilizar para la creación del sitio.

1.7.1 Lenguaje del lado del servidor

Los lenguajes que se ejecutan del lado del servidor son aquellos que el propio servidor es capaz de interpretar y ejecutar. Son muchos los lenguajes desarrollados para cumplir con esta función, dentro de los más conocidos se encuentra PHP, es en dicho lenguaje en el que se encuentra implementado el CMS Drupal. Por lo tanto es el utilizado en el desarrollo de las funcionalidades del sistema.

PHP: Es un lenguaje creado de manera especial para el desarrollo de páginas web dinámicas. Puede ser incluido con facilidad dentro del código HTML y permite una serie de funcionalidades tan extraordinarias que se ha convertido en el favorito de millones de programadores en todo el mundo a continuación se exponen algunas características.

- ✓ Velocidad: Posee buena velocidad de ejecución y no requiere de grandes recursos de sistema. PHP se integra muy bien junto a otro *software*, cuando se configura como módulo de Apache, está listo para ser ejecutado.
- ✓ Estabilidad: PHP utiliza su propio sistema de administración de recursos y dispone de un sofisticado método de manejo de variables, conforma un sistema robusto y estable.
- ✓ Seguridad: PHP provee diferentes niveles de seguridad que pueden ser configurados desde el archivo **.ini**.
- ✓ Simplicidad: Se les debe permitir a los programadores generar código funcional en el menor tiempo posible.
- ✓ Es un lenguaje multiplataforma.
- ✓ Sencilla integración con disímiles gestores de base de datos.

PHP es de código abierto, lo cual significa que el usuario no depende de una compañía específica para arreglar cosas que no funcionan, además no está forzado a pagar actualizaciones anuales para tener una versión que funcione (Mariño, 2008).

1.7.2 Lenguaje del lado del cliente

Los lenguajes de programación del lado del cliente son aquellos que pueden ser ejecutados directamente en el navegador, por lo general se encargan de organizar la estructura de los sitios web, su contenido y compartir información. (Gauchat, 2015)

Existen diferentes lenguajes del lado del cliente, entre los más utilizados se encuentran: JavaScript, CSS, HTML, DHTML, aunque en la presente investigación solo se hará alusión a los tres primeros mencionados pues son los utilizados en Drupal.

JavaScript: JavaScript es un lenguaje de programación que se utiliza por lo general para crear páginas web dinámicas. Los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. Los programas JavaScript van incrustados en los documentos HTML y se encargan de realizar acciones en el cliente, como pueden ser pedir datos, confirmaciones, mostrar mensajes, crear animaciones, comprobar campos. (Luque, 2001)

Otras de las características que posee JavaScript es que se interpreta por el ordenador que recibe el programa, no se compila. Tiene una programación orientada a objetos, el código de los objetos está predefinido y es expandible. Los navegadores más modernos en la actualidad utilizan soporte para JavaScript como es el caso de Firefox, Opera y Safari. (Eguíluz, 2008)

HTML: El lenguaje HTML⁸ es un estándar reconocido en todo el mundo cuyas normas define un organismo conocido como W3C⁹. Una misma página HTML se visualiza de forma muy similar en cualquier navegador de cualquier sistema operativo. HTML es el lenguaje que se emplea para el desarrollo de páginas de internet. Está compuesto por una serie de etiquetas que el navegador interpreta y da forma en la pantalla. La más reciente de las versiones es HTML5, ha sido desarrollada con el objetivo de perfeccionar el lenguaje para hacerlo más óptimo, corrige sobre todo los principales problemas que se presentaban en versiones anteriores, además de agregarle nuevas funcionalidades (Eguiluz, 2015).

⁸ Del inglés HyperText Markup Language.

⁹ Del inglés World Wide Web Consortium.

Principales novedades de HTML5: (Lancker, 2013)

- ✓ Un nuevo *doctype*, simplificado y unificado.
- ✓ Supresión de etiquetas y atributos de presentación.
- ✓ Nuevas etiquetas semánticas o de organización.
- ✓ Nuevas etiquetas de audio y de video que ya no necesitan más *plug-ins* dedicados.
- ✓ Profusión de formularios innovadores como, por ejemplo, cursores o calendarios y la integración de forma nativa en los navegadores de la validación de datos.

CSS: Es un lenguaje de hojas de estilos creado para controlar la presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para la creación de páginas web complejas. La separación de los contenidos y su presentación presenta numerosas ventajas, pues obliga a crear documentos HTML o XHTML bien definidos y con significado completo. Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes. (Pérez, 2007)

CSS3 no solo cubre diseño y estilos web si no también forma y movimiento. La especificación de CSS3 es presentada en módulos que permiten a la tecnología proveer una especificación estándar por cada aspecto involucrado en la presentación visual del documento. Desde esquinas redondeadas y sombras, hasta transformaciones y reposicionamiento de los elementos ya presentados en pantalla. Todos estos cambios convierten CSS3 en una tecnología novedosa comparada con versiones anteriores. (Gauchat, 2012)

1.8 Entorno de desarrollo integrado IDE

Un IDE es un programa informático compuesto por un conjunto de herramientas de programación. Consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación (IDE, 2009). Dentro de los IDE más populares a nivel mundial se encuentran NetBeans y Eclipse, a continuación se describen ambos entornos.

Eclipse: Eclipse es un IDE de código abierto y multiplataforma. Proporciona herramientas para gestionar espacios de trabajo, construir, lanzar y depurar aplicaciones y compartir objetos con el equipo de desarrollo. Está construido sobre un mecanismo para descubrir, integrar y ejecutar módulos. Este mecanismo de módulos es una plataforma ligera para componentes de software, que adiciona al Eclipse la capacidad de extenderse mediante el uso de otros lenguajes de programación como C/C++, Python y Java. (*Eclipse Foundation, 2004*)

NetBeans IDE: Es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. La plataforma NetBeans es un IDE de código abierto escrito en Java. Soporta el desarrollo de todos los tipos de aplicación Java (J2SE, Web y aplicaciones móviles). También tiene soporte para varios lenguajes como PHP, JavaScript, HTML, CSS. Es un producto libre y gratuito sin restricciones de uso. Esta plataforma permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Además, es una base modular y extensible usada como una estructura de integración para crear aplicaciones de escritorio grandes, permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación. (NetBeans, 2015)

El autor de la presente investigación decide utilizar NetBeans como IDE de programación por ser de código abierto, multiplataforma (Windows, Linux, MacOS) y gratuito. Además permite la utilización de varios lenguajes, incluido PHP, que es en el que está programado Drupal. Se utiliza NetBeans en su versión 8.0.

1.9 Herramienta CASE de modelado con UML

Las herramientas CASE proporcionan un conjunto de herramientas semiautomatizadas y automatizadas que brindan ayuda y dan asistencia a los analistas. Favorecen también a los ingenieros de software y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un producto determinado. (Nobrega, 2005)

Rational Rose: Es una potente herramienta de modelado visual para ayudar en el análisis y diseño de *software* orientado a objetos. Como todos los demás productos Rational Rose, proporciona un lenguaje común de modelado para el equipo que facilita la creación de software de calidad, posee características adicionales incluidas (IBM, 2010):

Capítulo 1: Fundamentación Teórica

- ✓ Basado en modelos que se integran con las bases de datos y los IDE de las principales plataformas del sector.
- ✓ Posee características de control por separado de componentes, modelo que permite una administración más granular y el uso de modelos.
- ✓ Tiene soporte de ingeniería *forward* o reversa para algunos de los conceptos más comunes de Java.
- ✓ Tiene integración con los sistemas de control de versiones
- ✓ Es capaz de crear definiciones de tipo de documento XML para el uso en la aplicación.

Visual Paradigm: Visual Paradigm es una herramienta CASE que soporta el modelado mediante UML y proporciona asistencia a los analistas, ingenieros de software y desarrolladores, durante todas las fases del ciclo de vida de un producto.

Las ventajas que proporciona Visual Paradigm son (Universidad Carlos III de Madrid, 2015):

- ✓ Facilita el modelado de UML, pues proporciona herramientas específicas para ello. Esto también permite la estandarización de la documentación, pues se ajusta al estándar soportado por la herramienta.
- ✓ Corrección sintáctica. Controla que el modelado con UML sea correcto.
- ✓ Coherencia entre diagramas. Al disponer de un repositorio común, es posible visualizar el mismo elemento en varios diagramas, esto evita duplicidades.
- ✓ Integración con otras aplicaciones. Permite integrarse con otras aplicaciones, como herramientas ofimáticas, lo cual aumenta la productividad.
- ✓ Trabajo multiusuario. Permite el trabajo en grupo, proporciona herramientas de compartición de trabajo.
- ✓ Facilita la reutilización, pues se dispone de una herramienta centralizada donde se encuentran los modelos utilizados para otros proyectos.
- ✓ Permite generar código de forma automática, reduce los tiempos de desarrollo para evitar errores en la codificación del software.
- ✓ Permite generar diversos informes a partir de la información introducida en la herramienta.

El autor de la presente investigación decide utilizar Visual Paradigm para UML en su versión 8.0 para modelar el sistema debido a las ventajas que brinda la herramienta en el proceso de desarrollo de software.

1.10 Servidor de Base de Datos

Una base de datos es una colección de archivos relacionados. La mayor parte de las base de datos son de tipo relacional. Se denominan así porque utilizan tablas de datos relacionadas por un campo en común. (Gilfillan, 2015)

MySQL: MySQL es un gestor de bases de datos muy utilizado a la hora de realizar páginas web dinámicas y sitios de comercio electrónico. Se suele trabajar en combinación con PHP debido a que comparten una serie de características que lo favorecen para su elección. Entre ellas (Vázquez, 2003):

- ✓ La mayoría de sus versiones son gratuitas.
- ✓ Posee una fuerte documentación en internet.
- ✓ Gran velocidad de ejecución.
- ✓ Es un gestor de base de datos multiplataforma.
- ✓ No requiere de demasiado tiempo de estudio para su entendimiento.
- ✓ Permite crear bases de datos, así como agregar, manipular y recuperar datos en función de criterios específicos.

Aunque existen versiones libres, desde hace algún tiempo han sido colocadas de manera privativa las últimas versiones desarrolladas, lo que impide que sea seleccionado para ser utilizado en el desarrollo de la propuesta de solución.

PostgreSQL: PostgreSQL es un sistema gestor de bases de datos (SGBD) orientado a objetos muy conocido y usado en entornos de *software* libre porque cumple los estándares SQL92 y SQL99, también por el conjunto de funcionalidades avanzadas que soporta, lo que lo sitúa al mismo o a un mejor nivel que muchos SGBD comerciales. PostgreSQL puede funcionar en múltiples plataformas (en general, en todas las modernas basadas en Unix) y también en Windows de forma nativa. (Ginestá& Pérez Mora, 2015)

Capítulo 1: Fundamentación Teórica

PostgreSQL se ha convertido en una de las bases de datos de elección en infinidad de proyectos, proporciona al usuario algunas prestaciones del mismo nivel que las que ofrecen sistemas gestores de bases de datos comerciales como Informix y Oracle. Las características más destacadas de PostgreSQL son: (Mateu, 2004)

- ✓ Soporte para transacciones.
- ✓ Subconsultas.
- ✓ Soporte de vistas.
- ✓ Integridad referencial.
- ✓ Herencia de tablas.
- ✓ Tipos definidos por el usuario.
- ✓ Columnas como vectores que pueden almacenar más de un valor.
- ✓ Añadir campos a tablas en tiempo de ejecución.
- ✓ *Triggers*, comandos SQL que deben ejecutarse al actuar sobre una tabla.
- ✓ Tablas de sistema que se pueden consultar para obtener información de tablas, de la base de datos y del motor de bases de datos.
- ✓ Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.

Se decide utilizar PostgreSQL en su versión 9.1 como gestor de base de datos por ser una herramienta capaz de brindar las ventajas necesarias para el desarrollo del sistema deseado. Además ha sido diseñado y creado para tener un mantenimiento y ajuste mucho menor que otros productos, conserva todas las características, estabilidad y rendimiento. Es capaz de ajustarse al número de unidades de procesamiento y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta. Todo esto lo hace ser un fuerte exponente a la hora de realizar la elección.

1.11 Servidor Web

Un servidor web es un programa que sirve para atender y responder a las diferentes peticiones de los navegadores, proporciona los recursos que soliciten mediante el protocolo HTTP o el protocolo HTTPS. (cibernetia, 2015)

Apache: Apache es un servidor web de código libre robusto cuya implementación se realiza de forma colaborativa, con prestaciones y funcionalidades equivalentes a las de los servidores comerciales. Apache presenta muchas características, entre ellas un elaborado índice de directorios, un directorio de alias, negociación de contenidos, informe de errores HTTP configurable, gestión de recursos para procesos hijos, integración de imágenes del lado del servidor, reescritura de las URL, comprobación de la ortografía de las URL y manuales online. Se pueden mencionar otras características importantes de Apache como son: (Kabir, 2015)

- ✓ Soporte del último protocolo HTTP.
- ✓ Sencillo, con la configuración basada en un poderoso archivo.
- ✓ Soporte de *host* virtuales.
- ✓ Soporte de autenticación HTTP.
- ✓ Soporte de *scripts* PHP.
- ✓ Servidor proxy integrado.
- ✓ Estado del servidor y adaptación de registros.

El autor de la presente investigación decide seleccionar Apache en su versión 2.2.22 como servidor web, teniendo en cuenta que es gratuito, multiplataforma, eficiente y seguro. Además su selección también viene respaldada por la posibilidad de descarga a través de internet, lo que posibilita la actualización del mismo.

Conclusiones del capítulo

En el capítulo se caracterizaron los conceptos relacionados con la gestión de eventos científicos desarrollados en la UCI, específicamente lo referente al Fórum de Ciencia y Técnica. Los métodos científicos empleados en la investigación permitieron elaborar los conceptos que ayudan a comprender

Capítulo 1: Fundamentación Teórica

mejor el problema de investigación. Se realizó un estudio de soluciones existentes a nivel internacional y nacional. No se identificó un sistema que cumpla con el proceso de gestión del FCT de la Facultad 4, pero si se obtuvieron características importantes para el desarrollo del sistema. Para llevar a cabo el desarrollo de la propuesta de solución se definió XP como metodología de desarrollo pues presenta prácticas muy afines para darle solución al problema planteado. Drupal en su versión 7.34 como CMS a usar para la propuesta de solución. Visual Paradigm en su versión 8.0 para realizar los diagramas necesarios, PostgreSQL en su versión 9.1 como SGBD, NetBeans en su versión 8.0 como IDE. Los lenguajes de programación definidos son PHP en su versión 5.3.2 como lenguaje del lado del servidor, JavaScript, HTML como lenguajes del lado del cliente y CSS3 para dar estilos a las páginas del sitio y como servidor web Apache en su versión 2.4.7.

Capítulo 2: Descripción de la propuesta de solución

Luego de haberse escogido la metodología, herramientas y tecnologías para el desarrollo de la propuesta de solución, se definen los requisitos para el funcionamiento del sistema. El desarrollo será guiado por la metodología XP en las dos primeras fases en las que se divide: Exploración y Planificación. Se definirán las funcionalidades y generarán los artefactos propuestos por la metodología seleccionada, concretando la cantidad de entregas y tiempos de ejecución. Se crearán las historias de usuario (HU) definidas para cada requisito funcional, se definirán los prototipos de interfaz de usuario y se elaborarán las tarjetas CRC (Clase-Responsabilidad-Colaboración) identificando las responsabilidades.

2.1 Objeto de informatización

Para llevar a cabo una correcta gestión del FCT en la Facultad 4 es necesario la informatización de los siguientes procesos, los cuales se mencionan a continuación:

- ✓ Publicar la convocatoria: El sistema permitirá realizar la publicación de la convocatoria del FCT para todas sus ediciones.
- ✓ Inscripción en el evento: El sistema brindará a los usuarios que se hayan autenticado en el sistema la posibilidad de inscribirse en el evento. El proceso de inscripción contará de un formulario donde se le especificará a cada usuario los campos que son necesarios para realizar dicho proceso de forma satisfactoria.
- ✓ Gestión del banco de problemas: El sistema permitirá realizar la gestión del banco problemas que estará disponible para la selección de uno a varios problemas para darle solución. El proceso de creación del banco de problemas dependerá de la creación de los problemas creados de forma individual.
- ✓ Creación de las comisiones El sistema brindará la posibilidad de crear comisiones, identificando el presidente de la misma y sus miembros. Permitirá la asignación de ponencias a estas comisiones para su revisión. El proceso de creación de las comisiones contará con un formulario donde se le especificará al usuario los campos que son necesarios para la creación de las mismas.
- ✓ Obtención del dictamen de participación: El sistema permitirá la creación del dictamen de participación otorgado por las comisiones a sus trabajos asignados. Durante la creación del dictamen se le especificara

Capítulo 2: Descripción de la propuesta de solución

a cada trabajo presentado la categoría alcanzada en el evento, las cuales serán 'Mención', 'Relevante' o 'Destacado'.

2.2 Diagrama conceptual del negocio

Un diagrama conceptual del negocio es un artefacto construido bajo las reglas de UML durante la concepción de un proyecto informático. Es el mapa conceptual que detalla la estructura de la entidad, es utilizado para capturar y expresar el entendimiento ganado sobre el negocio. (CEPYME, 2015)

El autor consideró necesario generar dicho artefacto para un mejor entendimiento de la propuesta de solución a desarrollar, a partir de identificar los conceptos y objetos relacionados con la gestión y control del FCT de la Facultad 4. A continuación se muestra el diagrama de modelo de negocio propuesto para el desarrollo del sitio.

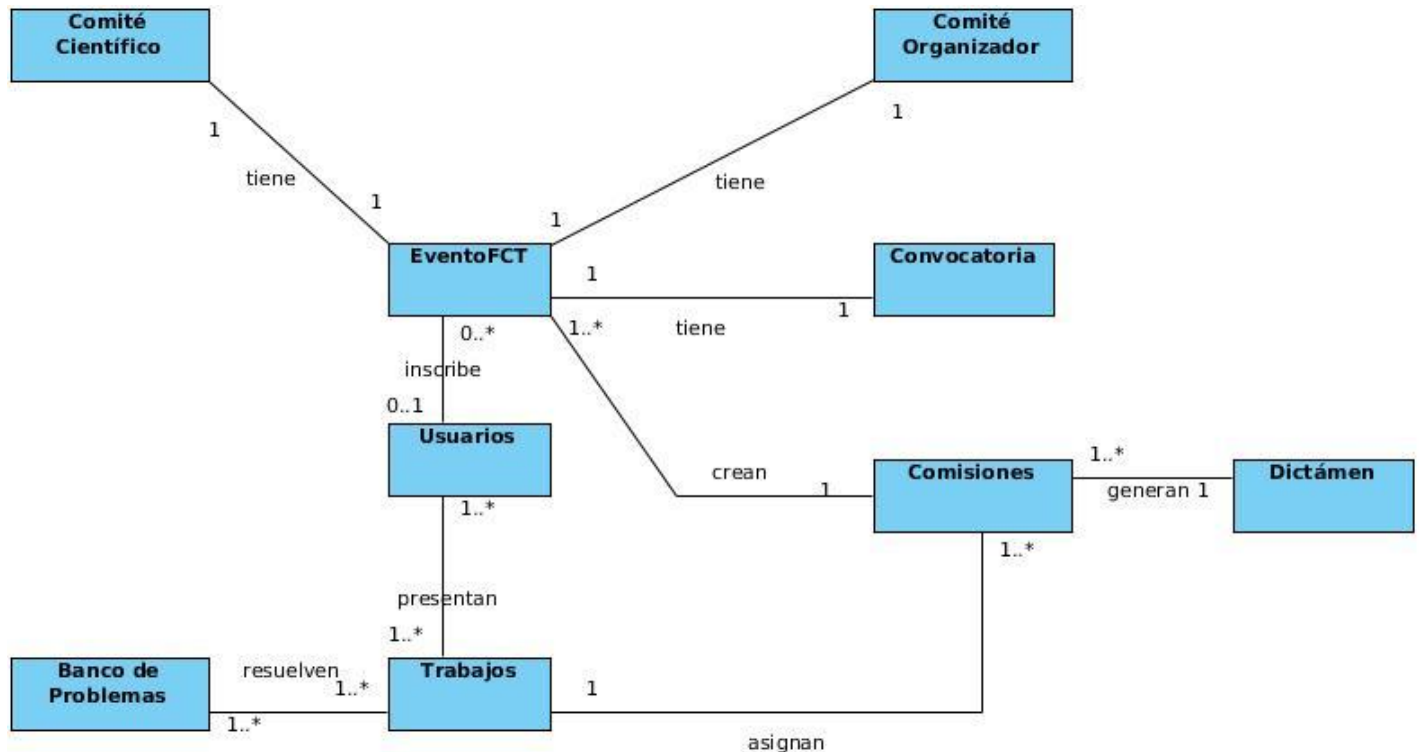


Figura 1 Diagrama conceptual del negocio.

2.3 Definición de los roles en el sistema

Se definieron los roles que tendrá cada usuario dentro del sistema. A continuación se definen los roles y sus respectivas responsabilidades dentro del sistema.

Tabla 3: Roles de los usuarios relacionados con el sistema.

Usuarios	Responsabilidades
Administrador	Es el encargado de administrar toda la información almacenada en el sistema, tiene la responsabilidad de dar soporte y mantenimiento al mismo, así como la asignación de permisos a los usuarios para acceder a las informaciones a las cuales puedan tener acceso.
Organizador	Tiene acceso al sistema, se le asigna la responsabilidad de gestionar el banco de problemas, gestionar las comisiones, gestionar la convocatoria, gestionar la inscripción y gestionar los certificados. El Organizador tiene acceso a toda la información almacenada en el sitio.
Integrante de la Comisión	Tiene acceso al sistema, se encarga de la descarga de las propuestas de solución, revisión de las mismas y otorgar una evaluación.
Usuario Autenticado	Podrá ser capaz de subir al sistema su trabajo para inscribirse en el fórum. Es válido aclarar que todos los privilegios del Usuario Inscrito son solamente sobre la información almacenada por el mismo.
Usuario	Es cualquier persona que navega por el sitio de forma anónima.

Capítulo 2: Descripción de la propuesta de solución

Requisitos funcionales

Los requisitos funcionales son declaraciones que identifican atributos, capacidades y características que necesita cumplir un sistema para que tenga valor y utilidad para el usuario. En otras palabras, los requisitos muestran qué elementos y funciones son necesarias para un proyecto. (ALEGSA, 2015)

Para darle cumplimiento a los servicios y contenidos trazados, se plantean los requisitos funcionales que deberá cumplir el sistema, a continuación se definen cada uno de ellos.

- ✓ Gestionar evento: El sistema debe permitir al Organizador del FCT la posibilidad de gestionar el evento, debe facilitar crear el evento, visualizar el evento, editar y eliminar el evento.
- ✓ Gestionar convocatoria: El sistema debe permitir al Organizador del FCT gestionar la convocatoria del evento, debe brindar la posibilidad de crear la convocatoria, visualizar la convocatoria, editar y eliminar la convocatoria.
- ✓ Gestionar comité científico: El sistema debe permitir al Organizador del FCT gestionar el comité científico, debe facilitar crear el comité científico, visualizar el comité científico, editar y eliminar el comité científico.
- ✓ Gestionar comité organizador: El sistema debe permitir al Organizador del FCT gestionar el comité científico, debe facilitar crear el comité organizador, visualizar el comité organizador, editar y eliminar el comité organizador.
- ✓ Gestionar banco de problemas: El sistema debe proveer al Organizador del FCT la posibilidad de gestionar el banco de problemas, debe facilitar crear el banco de problemas, visualizar el banco de problemas, editar y eliminar un determinado problema.
- ✓ Gestionar inscripción: El sistema debe permitir gestionar la inscripción en el evento, visualizar inscripción, editar y eliminar la inscripción.
- ✓ Gestionar comisión: El sistema debe permitir al Integrante crear la comisión, visualizar la comisión, editar y eliminar la comisión.
- ✓ Gestionar notificaciones: El sistema debe permitir gestionar las notificaciones, debe propiciar la posibilidad de crear la notificación, visualizar la notificación, editar y eliminar la notificación.

Capítulo 2: Descripción de la propuesta de solución

- ✓ Gestionar documentación del FCT: El sistema debe permitir gestionar la documentación del FCT, debe brindar la posibilidad de crear la documentación, visualizar la documentación, editar y eliminar la documentación.
- ✓ Gestionar roles y permisos: El sistema debe permitir gestionar los roles y permisos, debe propiciar la posibilidad de asignar roles y permisos a un determinado usuario dentro del sistema, visualizar los roles y permisos, editar y eliminar roles y permisos.
- ✓ Gestionar programa: El sistema debe permitir gestionar el programa del evento, debe garantizar la posibilidad de crear el programa, visualizar el programa, editar y eliminar el programa.
- ✓ Gestionar dictamen: El sistema debe brindar la posibilidad de gestionar el dictamen, debe garantizar la posibilidad de crear el programa, visualizar el programa, editar y eliminar programa.
- ✓ Autenticar usuario: El sistema debe brindar la posibilidad de que los usuarios se autenticuen en el sistema.

Requisitos no funcionales

Los requisitos no funcionales son aquellas propiedades que se refieren a las características que de una u otra forma pueden limitar el sistema. Estas propiedades garantizan que el producto sea atractivo, rápido y confiable. Sin embargo el incumplimiento de un requisito no funcional puede significar que el sistema entero sea inutilizable. (Scribd Inc, 2015)

Interfaz:

- ✓ No debe existir exceso de imágenes u objetos que distraigan al cliente del objetivo de su empleo.
- ✓ La información y los textos han de ser concisos y legibles.
- ✓ La interfaz será amigable e intuitiva.

Usabilidad: La aplicación debe ser diseñada de forma que los usuarios que hagan uso de ella obtengan los conocimientos necesarios en el menor tiempo posible. El menú debe estar presente en todo el sistema. El usuario debe tener claro en cada momento en que parte del sistema se encuentra y como progresar hacia otras acciones.

Capítulo 2: Descripción de la propuesta de solución

Rendimiento: Debe existir buena rapidez y eficiencia tanto en los tiempos de respuesta como en la velocidad de procesamiento. Garantizar velocidad estable de navegación a nivel de aplicación. Proporcionar tiempos de respuesta de 5 segundos mínimos para los procesos del sistema.

Seguridad: Los usuarios podrán acceder a la información autorizada de acuerdo al rol atribuido en el sistema. Evitar que la información en el sistema sea modificada por usuarios anónimos o que no tengan los permisos para esta acción. Se definen los roles de cada usuario, lo que permite separar las funcionalidades que podrá realizar cada uno. La carpeta donde se encuentre el sitio solo tendrá permiso de lectura.

Fiabilidad:

- ✓ El sistema debe registrar cualquier fallo en su funcionamiento para que los administradores puedan resolverlo.
- ✓ Debe mostrar al usuario un mensaje indicándole que ha ocurrido un fallo en la operación ejecutada.
- ✓ Los mensajes que se muestren a los usuarios deben mostrar los errores sin dar detalles de información, que puedan comprometer la seguridad e integridad del mismo.

2.4 Exploración

En esta fase se describen las principales características que tendrá el sistema, se especifican cuáles son las más importantes para la primera entrega del producto. Se definen las dificultades técnicas que presentan cada una de las características planteadas. Durante esta fase los programadores se familiarizan con las herramientas, tecnologías y prácticas que serán usadas en el desarrollo del proyecto, se exploran todas las posibilidades que puede tener la arquitectura del sistema. (Calabria y Píriz, 2003)

En general se realizan tres operaciones fundamentales durante la exploración: escribir las historias de usuario, estimarlas y dividir en unas más pequeñas o agrupar en otras más generales aquellas que se considere necesario.

2.4.1 Historia de usuario

Capítulo 2: Descripción de la propuesta de solución

Las historias de usuario (HU) son conocidas como *“la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas”*. (Torres, 2003)

Se utilizan las HU que plantea la metodología XP para representar los requisitos funcionales del sistema. Las HU pueden ser modificadas o reemplazadas por otras más generales o específicas, o incluso añadir nuevas. No lograr determinar todas las HU desde un inicio no es un problema, puesto que al inicio de cada iteración se registran los cambios y a partir de ahí se planifica la próxima iteración. Las HU se descomponen en tareas de ingeniería con una duración de uno a tres días, las cuales serán asignadas a los programadores para implementarlas durante una iteración. (Calabria y Píriz, 2003)

2.4.2 Historias de usuario de la propuesta de solución

La metodología XP tiene definida una plantilla para el desarrollo de las HU, dicha plantilla define los siguientes parámetros:

- ✓ Nombre.
- ✓ Número de la HU.
- ✓ Iteración a la que pertenece.
- ✓ Complejidad: nivel de complejidad (Alto/Medio/Bajo).
- ✓ Usuario que realiza la acción.
- ✓ Estimación del tiempo de desarrollo.
- ✓ Nivel de prioridad (Alta/Media/Baja).
- ✓ Descripción.
- ✓ Observaciones.

Capítulo 2: Descripción de la propuesta de solución

Tabla 5 HU Autenticar usuario.

Historia de Usuario	
Nombre: Autenticar usuario	Número: 1
Usuario: Usuario	
Complejidad: Alta	Prioridad: Alta
Iteración: 1	Estimación: 2 días
Descripción: Permite a los usuarios autenticarse en el sistema.	
Observación:	

Tabla 6 HU Gestionar inscripción.

Historia de Usuario	
Nombre: Gestionar inscripción	Número: 4
Usuario: Organizador	
Complejidad: Alta	Prioridad: Alta
Iteración: 1	Estimación: 1
Descripción: El sistema permitirá a los usuarios que hayan sido autenticados, la inscripción en el evento. Una vez inscritos en el mismo, el Organizador podrá ver los datos de los usuarios desde la aplicación.	
Observación: El usuario primero deberá estar autenticado en el sitio y además el sistema le indicará los campos que son obligatorios para la inscripción.	

El resto de las HU se encuentran en el [Anexo 1](#) de la investigación.

2.5 Estimación de esfuerzos por HU

La estimación de esfuerzo es establecida por los programadores que serán los encargados de llevar a cabo la implementación del sistema a desarrollar. Se utiliza como medida la semana. En caso de que una HU determinada supere las tres semanas deberá ser dividida en otras HU que puedan ser desarrolladas en un espacio de tiempo factible para el programador.

Tabla 7 Estimación de esfuerzos por HU

No	Historias de usuario	Puntos de estimación (en semanas)
1	Autenticar usuario	0.4
2	Gestionar roles y permisos	1
3	Gestionar convocatoria	0.6
4	Gestionar inscripción	1
5	Gestionar banco de problemas	1
6	Gestionar comisión	1
7	Gestionar programa	0.4
8	Gestionar comité científico	0.6
9	Gestionar evento	1
10	Gestionar dictamen	1
11	Gestionar comité organizador	0.6
12	Gestionar notificaciones	1
13	Gestionar documentación del FCT	1
	Duración total de la estimación	11

2.6 Planificación de las iteraciones

Esta es la fase principal en el ciclo de desarrollo de XP, donde son desarrolladas las funcionalidades, se genera al final de cada una un entregable funcional que implementa las historias de usuario asignadas a la iteración. Como las historias de usuario no tienen suficiente detalle como para permitir su análisis y

Capítulo 2: Descripción de la propuesta de solución

desarrollo, al principio de cada iteración se realizan las tareas necesarias de análisis y se obtienen con el cliente todos los datos que sean necesarios. Las iteraciones son también utilizadas para medir el progreso del proyecto. Una iteración terminada sin errores es una medida clara de avance. (Joskowicz, 2008)

2.6.1 Iteración 1.

En la primera iteración se implementan las HU de mayor complejidad e importancia del negocio. Con la realización de estas HU se dotaría al sistema de sus funcionalidades básicas.

2.6.2 Iteración 2.

En la segunda iteración se implementan las HU que sean de complejidad media en el negocio. Se realiza la corrección de posibles errores encontrados en la implementación de las HU implementadas con anterioridad.

2.6.3 Iteración 3.

En la última iteración se realiza la implementación de las HU de complejidad baja para el negocio. Se corrigen posibles errores encontrados en iteraciones anteriores.

2.7 Plan de duración de las iteraciones

Tabla 8 Plan de duración de las iteraciones

Iteraciones	Orden de las HU	Duración total
1	Autenticar usuario Gestionar banco de problemas Gestionar convocatoria Gestionar evento Gestionar inscripción	4 semanas
2	Gestionar comisión Gestionar notificaciones Gestionar roles y permisos Gestionar dictamen	4 semanas

Capítulo 2: Descripción de la propuesta de solución

3	Gestionar comité científico Gestionar comité organizador Gestionar documentación del FCT Gestionar programa	3 semanas
---	--	-----------

2.8 Plan de entregas

Las historias de usuarios seleccionadas para cada entrega son desarrolladas y probadas en un ciclo de iteración, de acuerdo al orden preestablecido. Al comienzo de cada ciclo, se realiza una reunión de planificación de la iteración. Cada historia de usuario se traduce en tareas específicas de programación. (Joskowicz, 2008)

Para cada historia de usuario se establecen las pruebas de aceptación. Dichas pruebas se realizan al final del ciclo en el que se desarrollan, pero también al final de cada uno de los ciclos siguientes, para verificar qué subsiguientes iteraciones no han afectado a las anteriores. Las pruebas de aceptación que hayan fallado en el ciclo anterior son analizadas para evaluar su corrección, así como para prever que no vuelvan a ocurrir.

Tabla 9 Plan de entregas

Historia de usuario	Primera iteración	Segunda iteración	Tercera iteración
Autenticar usuario	V1.0	Finalizado	Finalizado
Gestionar inscripción	V1.0	Finalizado	Finalizado
Gestionar banco de problemas	V1.0	Finalizado	Finalizado
Gestionar convocatoria	V1.0	Finalizado	Finalizado
Gestionar evento	V1.0	Finalizado	Finalizado
Gestionar notificaciones	-	V1.0	Finalizado
Gestionar comisión	-	V1.0	Finalizado
Gestionar roles y permisos	-	V1.0	Finalizado
Gestionar dictamen	-	V1.0	Finalizado
Gestionar comité científico	-	-	V1.0
Gestionar comité organizador	-	-	V1.0

Capítulo 2: Descripción de la propuesta de solución

Gestionar documentación del FCT	-	-	V1.0
Gestionar programa	-	-	V1.0

2.9 Tarjetas Clase-Responsabilidades-Colaboradores

“Una de las partes más importantes de la filosofía XP es la simplicidad en todos los aspectos. Se considera que un diseño sencillo se logra más rápido y se implementa en menos tiempo, por lo cual esto es lo que se busca. La idea es que se haga el diseño más sencillo que cumpla con los requerimientos de las historias de usuario” (TOBÓN, 2007).

La principal funcionalidad que tienen las tarjetas Clase-Responsabilidades-Colaboradores (CRC) es ayudar a dejar el pensamiento procedimental para incorporarse al enfoque orientado a objetos. Cada tarjeta representa una clase con su nombre en la parte superior, en la sección inferior izquierda están descritas las responsabilidades y a la derecha las clases que le sirven de soporte (TOBÓN, 2007).

En la metodología XP las tarjetas CRC se utilizan para diseñar el sistema. Utilizan la Programación Orientada a Objetos (POO¹⁰) y se componen de tres elementos:

- ✓ Clase: se nombra cualquier persona, evento, concepto, pantalla o reporte.
- ✓ Responsabilidades: son los métodos relacionados con la clase y sus atributos.
- ✓ Colaboradores: son las otras clases con las que trabaja en conjunto para llevar a cabo sus responsabilidades.

Debido a que Drupal utiliza la POO y por tanto define conceptos o técnicas tales como objetos, abstracción, polimorfismo, encapsulación y herencia. Se adaptan las técnicas de las tarjetas CRC a estas características, cada tarjeta CRC representa a un módulo, las responsabilidades quedan definidas como las funciones que realiza y los colaboradores serían los métodos del módulo. A continuación se muestra un ejemplo de una de ellas:

¹⁰ Del inglés Object Oriented Programming

Capítulo 2: Descripción de la propuesta de solución

Tabla 10 tarjeta CRC Gestionar evento.

Responsabilidades	Colaboradores
Insertar contenido de tipo carpeta	m_archivos_menu() m_archivos_node_insert() m_archivos_form_carpeta_archivos_node_for m_alter()
Mostrar carpeta	load_data_file_show_callback() m_archivos_node_view() m_archivos_theme() printer_folder() m_determina_sub_directorio()
Editar carpeta	m_archivos_node_delete()

Conclusiones del capítulo

Luego de haber realizado la identificación de los procesos se definieron las principales características que debe tener la propuesta de solución. Se obtuvieron los artefactos generados por las dos primeras fases de la metodología XP, asegurando una implementación por etapas correctamente descritas y detalladas. Se elaboraron 13 HU, luego se agruparon en 3 iteraciones, que serán entregadas en un plazo de 11 semanas. Se delimitó que existirán 5 tipos de usuarios que interactúan con el sistema con diferentes permisos en el mismo. Se crearon las tarjetas CRC necesarias para tener un enfoque orientado a objetos.

Capítulo 3: Implementación y Pruebas

En el presente capítulo se toma como base el proceso de exploración y planificación descrito en el capítulo anterior dando paso al desarrollo de entregas funcionales, las cuales dan cumplimiento a los objetivos definidos. El centro del capítulo está en las fases Producción y Mantenimiento de la metodología seleccionada, se describe el tema usado y los módulos configurados, junto con los desarrollados para dar cumplimiento a los requerimientos propuestos por el cliente. Se realizarán las pruebas unitarias al código y pruebas de aceptación a las funcionalidades, documentando todos los resultados arrojados.

3.1 Patrón arquitectónico de Drupal

Durante el desarrollo de la propuesta de solución del sitio fue necesario definir la arquitectura que tendrá el mismo, la arquitectura seleccionada es el patrón modelo-vista-controlador pues garantiza organización y seguridad en el acceso a datos.

“El patrón modelo-vista-controlador (MVC) es una arquitectura de diseño software, para separar los componentes de aplicación en tres niveles, interfaz de usuario, lógica de control y lógica de negocio. Es una especialización de un modelo de capas, con la diferencia que se usa para entornos web como patrón por excelencia” (Gómez, 2011)

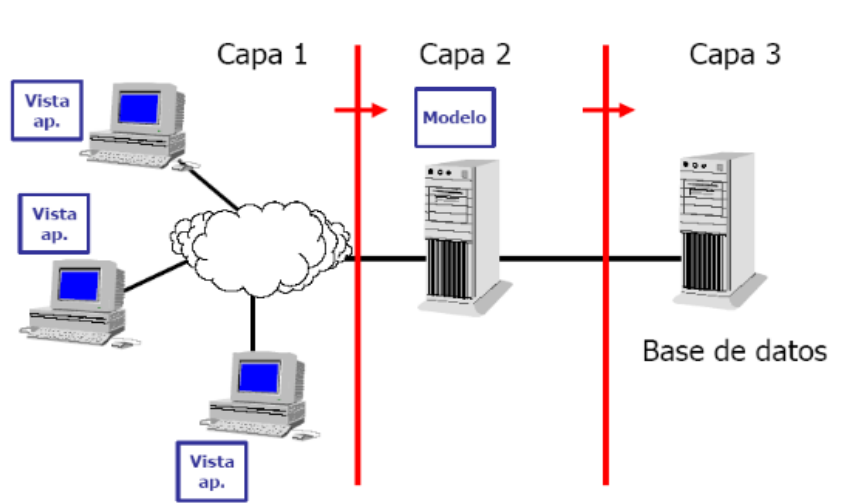


Figura 2 Patrón MVC (Gómez, 2011).

El modelo es la capa encargada de encapsular toda la lógica de negocio de la aplicación. Contiene clases para constituir lo referente a la capa de aplicación, es la encargada de atender a las peticiones de los controladores y así dar una respuesta acorde con lo recibido. Solo se comunica con la lógica de negocio. El controlador es el eje central de nuestra arquitectura, encargada de gestionar todas las peticiones y dirigir cualquier petición de cualquier tipo. Solo se comunica con el modelo y responde a través de vistas. La vista es la respuesta de cada controlador y lo que se le presenta al usuario final se puede comunicar con el controlador.

Drupal implementa la arquitectura MVC de la siguiente manera:



Figura 3 MVC en Drupal

3.2 Estructura de Drupal

Debido a que la estructura de Drupal está definida de manera automática en la instalación, no se considera una buena práctica tratar de cambiar la estructura. Durante el desarrollo del sistema se mantiene la estructura definida por Drupal en todos sus niveles.

El secreto de Drupal para conseguir su flexibilidad y facilidad en la creación de sitios web es la abstracción y organización en capas que aplica en el tratamiento de los contenidos. En lugar de considerar el sitio web como un conjunto de páginas interrelacionadas, Drupal estructura los contenidos en una serie de

elementos básicos. Estos son los nodos, módulos, bloques y menús, permisos de usuario y plantillas. (GitHub, 2015)

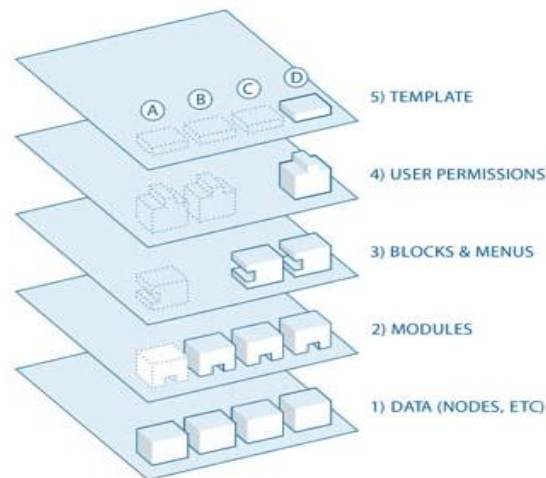


Figura 4 Estructura de Drupal (GitHub, 2015).

Los nodos son los elementos básicos en que Drupal almacena la información. Así a medida que el sitio web crece, lo hace también el número de nodos, los cuales forman un “depósito de nodos” cada vez mayor. Se puede decir que la primera capa de la estructura de Drupal la forma el “depósito” de nodos.

Los módulos son los elementos que operan sobre los nodos y otorgan funcionalidad a Drupal, permiten incrementar sus capacidades o adaptarlas a las necesidades de cada sitio web. Son como *plug-ins* que se instalan en el sitio web proporcionándole nuevas funcionalidades.

Los bloques y menús permiten estructurar y organizar los contenidos en la página web. Es decir que son los elementos que albergan y permiten acceder al usuario a la salida generada y procesada por los módulos a partir de la información almacenada en los nodos.

La siguiente capa importante en Drupal es la de control de usuarios y permisos. Drupal dispone de un registro de usuarios y de roles que permiten especificar que tareas pueden realizar y a que contenidos puede acceder cada tipo de usuario. Es decir que las operaciones que se pueden realizar sobre los elementos provenientes de las capas inferiores se encuentran limitadas por la capa de control de usuarios y permisos de Drupal.

La última capa, es la capa de Temas y es la que establece la apariencia gráfica o estilo de la información que se le muestra al usuario. La separación entre información y aspecto gráfico permite cambiar el diseño o apariencia del sitio web sin necesidad de modificar los contenidos, lo que es muy práctico si lo único que se quiere es renovar la apariencia de un sitio web. (Comunic-art programming and web design, 2015)

3.2.1 Tema

Durante el desarrollo del sistema no fue necesario crear un tema desde cero, pues Drupal brinda la posibilidad de utilizar uno o varios de sus temas que vienen con la instalación. El tema seleccionado brinda las características básicas para el desarrollo del sitio.

En Drupal un tema es un conjunto de archivos que permite cambiar el aspecto gráfico del sitio. Se puede instalar un único tema para todo el sitio o varios temas, si se quiere que los usuarios puedan seleccionar uno de entre los disponibles en el sitio. (Gil, 2012)

La elección del tema dependerá de la estructura que se quiera para el sitio que se ha diseñado, por eso es importante pensar primero en las secciones que tendrá y en su estructura (menús, bloques, columnas). Cuanto más se adapte el tema a la estructura final deseada, menos modificaciones habrá que realizarle al tema. (Gil, 2012)

El tema seleccionado para la propuesta de solución es *Bartik*, el mismo brinda la posibilidad de la creación de diferentes bloques y menú en el sitio. Fue necesario hacerle algunas modificaciones para ajustarlo a las verdaderas necesidades del desarrollador. En la ilustración queda reflejado el diseño de la interfaz del tema en la creación del nodo Eventos.

The screenshot displays the SGCFCT website interface. At the top, there is a header with the logo 'FORUM CIENCIA Y TÉCNICA' and 'SGCFCT Sistema de Gestión y Control Forum de Ciencia y Técnica'. Navigation links include 'Inicio', 'Evento', 'Banco de problemas', 'Comité Científico', 'Comité Organizador', 'Convocatoria', 'Temáticas', 'Trabajos-Inscritos', and 'Contacto'. A search bar is located in the top right corner. The main content area features a central announcement for the 'Fórum de Ciencia y Técnica' 2015, including the date 'Lunes, Junio 1, 2015 to Sábado, Junio 20, 2015' and a 'Leer más' link. The left sidebar contains 'Vista Comisiones' with a link to 'TrabajosXComisiones' and 'Gestionar evento' with links for 'Crear Comisión', 'Crear Dictamen', 'Crear Problema', 'Listar Problemas', and 'Problemas Propuestos'. The right sidebar includes 'Operaciones' with links for 'Crear Problema' and 'Inscribirse', and 'Informaciones' with a link for 'Historial de Trabajos'.

Figura 5 Tema del sistema.

3.2.2 Usuarios y permisos

En el sitio cada usuario podrá tener asignado uno o más roles. Por defecto todo usuario que se registre e identifique en el sistema tendrá asignado de carácter automático el rol de usuario autenticado. El usuario con rol Administrador tendrá asignado todos los permisos entre los disponibles en el sitio. En resumen, los permisos hacen referencia a determinadas funcionalidades de los módulos, que son los encargados de generarlos y se vinculan a los roles. Los usuarios tendrán o no permisos para las funcionalidades en función de los roles que se les haya asignado. (Gil, 2012)

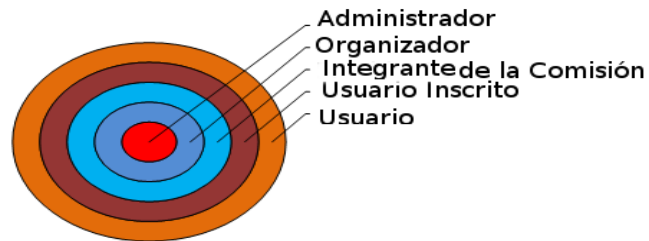


Figura 6 Jerarquía de Roles del sistema.

3.2.3 Bloques y menús

Para garantizar una mejor organización y estética en el sitio fue necesario crear varios bloques y menús. Los bloques y menú se realizaron con el objetivo de hacer el trabajo de los usuarios más placenteros y a su vez garantizar los principales servicios del sitio.

Los bloques son segmentos de código o contenido con distintas funcionalidades que se colocan en los laterales del sitio, aunque en realidad se pueden colocar en cualquiera de las regiones definidas del tema. También se pueden crear nuevos bloques de código PHP o contenido HTML. Para que un bloque se muestre en un sitio, tiene que estar asignado a una región determinada, de lo contrario no se podrá visualizar en el sistema. En el área de administración, los bloques aparecen agrupados según la región donde están ubicados. (Gil, 2012)

“Drupal cuenta con un potente gestor de menús que permite disponer de múltiples y variados menús en un mismo portal. Los menús se pueden colocar en distintas áreas o regiones de un tema y se adaptan perfectamente al diseño gráfico del sitio”. (Gil, 2012)

Drupal por defecto tiene bloques y menús predefinidos de los cuales en el sistema se utilizan algunos de ellos: menú principal, bloque de autenticación, bloque de búsqueda y el de usuarios en línea. A continuación se muestran ejemplos de bloque y menús creados en la propuesta de solución.



Figura 7 Distribución de bloques y menús en el sistema

3.2.4 Módulos

Drupal es un CMS que brinda muchas funcionalidades mediante la creación de contenidos que pueden resolver varios aspectos del problema en cuestión, pero para la creación de funcionalidades específicas en el sistema fue necesario incorporar nuevas funciones adicionales que no están disponibles en el núcleo de Drupal. Las funcionalidades son permitidas mediante el agregado de módulos desarrollados.

Los módulos permiten incrementar y mejorar las funcionalidades de un sitio, de una forma rápida y sencilla. Son un conjunto de funcionalidades con la capacidad de integrarse en una instalación de Drupal, añaden y mejoran el funcionamiento del sitio en multitud de áreas: administración, gestión de contenidos, gestión de usuarios. (Gil, 2012)

Los módulos se clasifican en 3 grandes tipos: los módulos del núcleo que son los que vienen en la instalación por defecto de Drupal, los módulos contribuidos que son los desarrollados por la comunidad que da soporte a Drupal y los personalizados que son los creados por los desarrolladores del sitio.

Dentro de los módulos contribuidos se utilizó:

- ✓ SMTP: sirve de soporte al protocolo que lleva el mismo nombre, el cual se utiliza para el envío de notificaciones por correo electrónico.

Capítulo 3: Implementación y pruebas

- ✓ LDAP: módulo que sirve de soporte al protocolo que lleva el mismo nombre, el cual se utiliza para garantizar la autenticación de los usuarios en la aplicación.
- ✓ DATE: se utiliza como soporte para facilitar la inserción de fechas en la aplicación
- ✓ Módulo ckeditor: editor de texto HTML para Drupal.
- ✓ Módulo features: permite la captura y gestión de los elementos en Drupal.
- ✓ Módulo views: diseñado para crear, editar y administrar vistas en la interfaz de usuario de Drupal.

Módulos incluidos en el núcleo de Drupal:

- ✓ Módulo Filter (Filtro): permite la entrada de contenido para el sitio y la configuración del mismo.
- ✓ Módulo Node (Nodo): todo el contenido en un sitio web de Drupal se almacena y se trata como “un nodo”. Un nodo es por ejemplo una página, una encuesta, una historia, un texto del foro, o una entrada del blog. Este módulo posibilita que se enumere, clasifique y maneje todo el contenido en el sitio.
- ✓ Módulo Block (Bloque): controla el montaje visual de los bloques con los que se construye una página.
- ✓ Módulo User (Usuario): administra el registro de usuarios y el sistema de inicio de sesión.
- ✓ Módulo System (Sistema): gestiona la configuración general del sitio por administradores.

Módulos opcionales incluidos en el núcleo de Drupal:

- ✓ Módulo Path (Camino): permite indicar alias a los URLs de Drupal, ayuda a mejorar la legibilidad. Una página puede tener más de un alias.
- ✓ Módulo Menu (Menú): permite a los administradores personalizar el menú de navegación del sitio mediante una interfaz amigable.
- ✓ Módulo Upload (Subir archivos): permite que los usuarios carguen y adjunten archivos.
- ✓ Módulo Taxonomy (Taxonomía): permite la categorización del contenido.
- ✓ Módulo Search (Búsqueda): permite la realización de búsquedas en todo el sitio creado.
- ✓ Módulo Rules (Reglas): utilizado para notificar al usuario los diferentes eventos en la aplicación con un sistema de correo propio.

Módulos creados para el funcionamiento del sitio:

- ✓ Módulo evento: Permite almacenar todos los archivos del evento en una carpeta específica.

En la implementación de los módulos están reflejados los siguientes patrones de diseño: **Instancia única, Decorador, Observer, Cadena de responsabilidades**, esta es la manera particular en que Drupal establece la escritura de sus módulos, lo que además hace que el código sea más legible, se fortalece la práctica de XP, que propone la utilización de estándares de codificación. A continuación se describen ejemplos de patrones GOF que son evidenciados en Drupal.

Instancia única (*Singleton*): consiste en garantizar que una clase sólo tenga una instancia y proporcionar un punto de acceso global a ella. Dentro del núcleo de Drupal se utiliza en diversas tareas, como la gestión del *pool* de conexiones con la base de datos, toma los módulos y temas como objetos para llevar a cabo la gestión de dichos elementos.

Decorador (*Decorator*): la evidencia del patrón decorador se encuentra en la definición de *hooks* por parte de los módulos del núcleo de Drupal y otros módulos contribuidos, que posibilitan que otros puedan extender el comportamiento de dichas funciones. Son funciones que permiten la comunicación con todo el núcleo de Drupal, pueden ser eventos u operaciones con campos. De esta forma se brinda la flexibilidad de que nuevos módulos puedan modificar el comportamiento del núcleo en cuanto al tratamiento de los datos y en cada uno de los eventos del sistema.

Cadena de responsabilidades (*Chain of responsibility*): el sistema de menús de Drupal es la evidencia del patrón *Chain of responsibility*. En cada petición de una página, el sistema de menús de Drupal determina si hay algún módulo para responder la petición, si el usuario tiene acceso al recurso solicitado y que función se debe llamar para procesar la petición. Durante el proceso se trasmite el mensaje de la petición por cada uno de los componentes que se encuentran inmersos. La cadena se continúa hasta que un módulo atienda la petición, hasta que un módulo deniegue el acceso o hasta que la cadena se agote.

3.3 Tareas de Ingeniería

En la metodología XP los programadores dividen las HU en tareas más pequeñas, denominadas tareas de ingeniería (TI). Luego, cada programador elige las tareas que desea implementar, las analiza en mayor detalle y realiza una estimación de su tiempo de desarrollo. (Piwen, 2001)

Durante el proceso de desarrollo del software se plantea por el equipo de desarrollo que 1 semana equivale a 5 días, es decir, se trabajará de lunes a viernes. Fueron detectadas un total de 13 HU, las cuales se dividieron en 49 TI, aquí solo se muestran las primeras TI de cada iteración, las restantes se encuentran en los Anexos 2, 3 y 4 de la investigación.

Para la **primera iteración** se definieron las siguientes TI:

HU: 1 Autenticar usuario.

- ✓ TI: 1 Autenticar usuario.

HU: 11 Gestionar evento

- ✓ TI: 2 Crear evento.
- ✓ TI: 3 Visualizar evento.
- ✓ TI: 4 Modificar evento.
- ✓ TI: 5 Eliminar evento.

HU: 3 Gestionar convocatoria

- ✓ TI: 6 Crear convocatoria
- ✓ TI: 7 Visualizar convocatoria.
- ✓ TI: 8 Modificar convocatoria.
- ✓ TI: 9 Eliminar convocatoria.

HU: 4 Gestionar inscripción.

- ✓ TI: 10 Crear inscripción.
- ✓ TI: 11 Visualizar inscripción.
- ✓ TI: 12 Modificar inscripción.
- ✓ TI: 13 Eliminar inscripción.

HU: 5 Gestionar banco de problemas.

- ✓ TI: 14 Crear banco de problemas.
- ✓ TI: 15 Visualizar banco de problemas.

- ✓ TI: 16 Modificar banco de problemas.
- ✓ TI: 17 Eliminar banco de problemas.

Tabla 11 TI Autenticar usuario.

Tarea	
Número de tarea: 1	Número de HU: 1
Nombre de la tarea: Autenticar usuario.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 1 de enero de 2015	Fecha fin: 2 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: Permite al usuario autenticarse en el sistema.	

Las restantes TI de la iteración 1 se encuentran en el [Anexo 2](#).

Para la **segunda iteración** se definieron las siguientes TI:

HU: 6 Gestionar comisión.

- ✓ TI: 18 Crear comisión.
- ✓ TI: 19 Visualizar comisión.
- ✓ TI: 20 Modificar comisión.
- ✓ TI: 21 Eliminar comisión.

HU: 12 Gestionar notificaciones.

- ✓ TI: 22 Crear notificaciones.
- ✓ TI: 23 Visualizar notificaciones.
- ✓ TI: 24 Modificar notificaciones.
- ✓ TI: 25 Eliminar notificaciones.

HU: 2 Gestionar roles y permisos.

- ✓ TI: 26 Crear roles y permisos.
- ✓ TI: 27 Visualizar roles y permisos.
- ✓ TI: 28 Modificar roles y permisos.
- ✓ TI: 29 Eliminar roles y permisos.

HU: 10 Gestionar dictamen.

- ✓ TI: 30 Crear dictamen.
- ✓ TI: 31 Visualizar dictamen.
- ✓ TI: 31 Modificar dictamen.
- ✓ TI: 33 Eliminar dictamen.

Tabla 12 TI Crear comisión.

Tarea	
Número de tarea: 18	Número de HU: 6
Nombre de la tarea: Crear comisión.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Fecha inicio: 3 de enero de 2015	Fecha fin: 5 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: Permite al Organizador crear una determinada comisión.	

Las restantes TI de la iteración 2 se encuentran en el [Anexo 3](#).

Para la **tercera iteración** se definieron las siguientes TI:

HU: 8 Gestionar comité científico.

- ✓ TI: 34 Crear comité científico.
- ✓ TI: 35 Visualizar comité científico.
- ✓ TI: 36 Modificar comité científico.
- ✓ TI: 37 Eliminar comité científico.

HU: 9 Gestionar comité organizador.

- ✓ TI: 38 Crear comité organizador.
- ✓ TI: 39 Visualizar comité organizador.
- ✓ TI: 40 Modificar comité organizador.
- ✓ TI: 41 Eliminar comité organizador.

HU: 13 Gestionar documentación del FCT.

- ✓ TI: 42 Crear documentación del FCT.

- ✓ TI: 43 Visualizar documentación del FCT.
- ✓ TI: 44 Modificar documentación del FCT.
- ✓ TI: 45 Eliminar documentación del FCT.
- HU: 7 Gestionar programa.
- ✓ TI: 46 Crear programa.
- ✓ TI: 47 Visualizar programa.
- ✓ TI: 48 Modificar programa.
- ✓ TI: 49 Eliminar programa.

Tabla 13 TI Crear comité científico.

Tarea	
Número de tarea: 30	Número de HU: 8
Nombre de la tarea: Crear comisión.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 días
Fecha inicio: 4 de enero de 2015	Fecha fin: 5 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: Permite al Organizador crear el comité científico del evento.	

Las restantes TI de la iteración 3 se encuentran en el [Anexo 4](#).

3.4 Pruebas

Según expresa (Tobón, 2007), la metodología XP enfatiza mucho en los aspectos relacionados con las pruebas, clasificándolas en diferentes tipos y funcionalidades específicas, indica quién, cuándo y cómo deben ser implementadas y ejecutadas. Según XP se debe ser muy estricto con las pruebas. Sólo se deberá liberar una nueva versión si ha pasado por completo el proceso de pruebas. En caso contrario se empleará el resultado para identificar el error y solucionarlo con mecanismos ya definidos.

Las pruebas en XP están divididas en pruebas unitarias y pruebas de aceptación, es de suma importancia resaltar en qué se diferencian las mismas en cuanto al papel del usuario se refiere. Mientras que en las pruebas de aceptación el usuario juega un papel fundamental al seleccionar los casos de prueba para

cada HU e identificando los resultados esperados, en las unitarias no tiene ninguna intervención, son creadas por los programadores y verifican el código de manera automática. (Tobón, 2007)

3.4.1 Pruebas unitarias

Las pruebas unitarias *“constituyen el primer paso para detectar errores en el código, pues se centran en la menor unidad de diseño del software: el módulo. El objetivo principal de las pruebas unitarias es detectar errores en cada uno de los módulos del software, al ser ejecutados independientemente del resto de los componentes”*. (Román, 2007)

Estas pruebas son el arte y la práctica de tomar una pequeña porción de código y someterlo a pruebas de programación para probar su corrección. Las unidades más pequeñas en el código PHP son normalmente las funciones. En las pruebas unitarias se ejecutan las funciones con conjuntos de parámetros controlados y luego se hacen afirmaciones acerca de los valores de retorno de la función, o sobre el estado de la aplicación antes y después de ejecutada la función. Por ejemplo, si la función está diseñada para validar direcciones de correo electrónico, en las pruebas unitarias se le pasarían direcciones de correo electrónico válidas conocidas para afirmar que se validan de carácter correcto. También se pasarían una serie de datos incorrectos y así afirmar que no validan. (Douglass, 2007)

Importancia de las pruebas unitarias.

Las pruebas unitarias revisten gran importancia, entre los elementos que así lo demuestran se pueden mencionar:

- ✓ Escribir pruebas unitarias ayuda a producir código de mayor calidad en muchos niveles.
- ✓ Las pruebas también sirven como fuente de documentación sobre lo que se espera que haga el código creado.
- ✓ El acto de escribir las pruebas desafía al programador para considerar posibles casos extremos y sus consecuencias.
- ✓ Por último, pero no menos importante, el acto de pruebas de unitarias anima al programador escribir código en pequeños trozos que pueden ser probados de forma independiente. (Douglass, 2007)

Simple Test

Simple Test es un marco de pruebas unitarias de código abierto para el lenguaje de programación PHP y fue creado por Marcus Panadero. La estructura de la prueba es similar a JUnit / PHPUnit. Soporta objetos simulados y se puede utilizar para automatizar las pruebas de regresión de aplicaciones web con scripts HTTP de cliente que puede analizar las páginas HTML y simular cosas como hacer clic en enlaces y la presentación de los formularios.

Simple Test proporciona algunas pruebas básicas por defecto, además de que se pueden escribir pruebas para módulos personalizados o formularios que se creen. Las pruebas utilizan archivos de prueba y se deben colocar en el directorio del módulo que se está probando. (Khachtourians, 2015)

Resultado de las pruebas con Simple Test

Se realizaron pruebas unitarias a la aplicación mediante el módulo Simple Test sobre las funcionalidades críticas que más se utilizan y otras que se encuentran en los módulos generados para ciertas especificaciones.

Realización de las pruebas

Para ello se accede al menú de acceso de la interfaz de prueba en el panel de administración del CMS como se muestra a continuación.

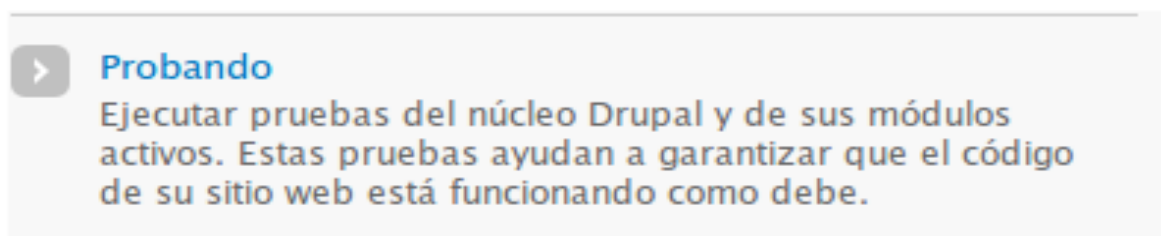


Figura 8 Menú de acceso a la interfaz de prueba

Una vez seleccionadas las pruebas del módulo propuesto se ejecutan las mismas. Debe tenerse en cuenta que el proceso se puede demorar en dependencia de la cantidad de pruebas a realizar. En la siguiente imagen se muestran las pruebas realizadas al módulo Menú.

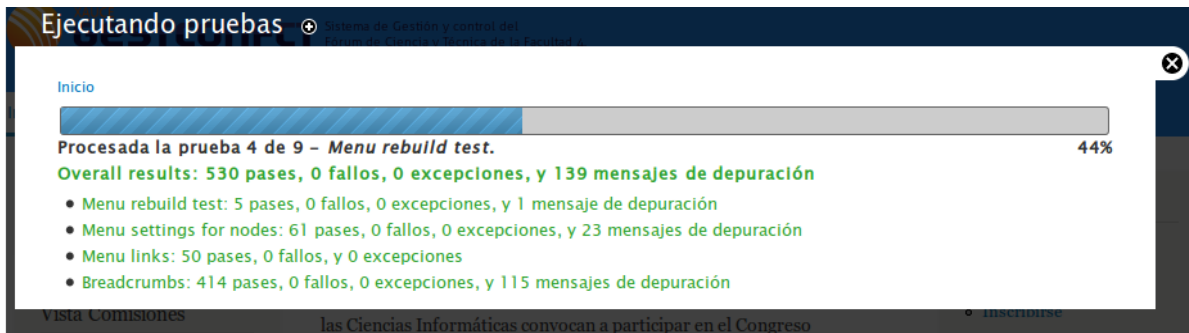


Figura 9 Proceso de ejecución de la prueba.

Una vez terminadas las pruebas el sistema arroja los resultados obtenidos: 1521 passes, 0 fallo, 0 excepciones y 359 mensajes de depuración.



Figura 10 Resultado final de la prueba

3.4.2 Pruebas de aceptación

Durante el proceso de desarrollo del sistema es necesario hacer una serie de pruebas de aceptación, las mismas se realizan con el objetivo de garantizar que las funcionalidades creadas cumplan con las HU planteadas anteriormente.

Capítulo 3: Implementación y pruebas

Según plantea (Joskowicz, 2008), las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada. Las mismas son consideradas como pruebas de caja negra. Los clientes son responsables de verificar que los resultados de las pruebas sean correctos. Asimismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución. Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. Dado que la responsabilidad es colectiva, es recomendable publicar los resultados de las pruebas de aceptación, de manera que el equipo esté al tanto de toda la información.

Durante el proceso de desarrollo del sitio fueron realizados 49 casos de pruebas de aceptación (CPA), en el presente epígrafe solo se muestra el primero de cada entrega, los restantes se encuentran en los anexos 5, 6 y 7.

Para la **primera entrega** quedaron definidos los siguientes CPA:

HU: 1 Autenticar usuario.

- ✓ CPA.1 Autenticar usuario.

HU: 11 Gestionar evento

- ✓ CPA.2 Crear evento.
- ✓ CPA.3 Visualizar evento.
- ✓ CPA.4 Modificar evento.
- ✓ T CPA.5 Eliminar evento.

HU: 3 Gestionar convocatoria

- ✓ CPA.6 Crear convocatoria
- ✓ CPA.7 Visualizar convocatoria.
- ✓ CPA.8 Modificar convocatoria.
- ✓ CPA.9 Eliminar convocatoria.

HU: 4 Gestionar inscripción.

- ✓ CPA.10 Crear inscripción.
- ✓ CPA.11 Visualizar inscripción.
- ✓ CPA.12 Modificar inscripción.

- ✓ CPA.13 Eliminar inscripción.
HU: 5 Gestionar banco de problemas.
- ✓ CPA.14 Crear banco de problemas.
- ✓ CPA.15 Visualizar banco de problemas.
- ✓ CPA.16 Modificar banco de problemas.
- ✓ CPA.17 Eliminar banco de problemas.

Tabla 14 CPA Autenticar usuario

Caso de prueba de aceptación	
Código: HU1_CPA1	Historia de usuario: 1
Nombre: Autenticar usuario.	
Descripción: Prueba para la funcionalidad que permite autenticar un usuario.	
Condiciones de ejecución: El usuario debe estar autenticado en el sitio bajo el rol de Usuario.	
Pasos de ejecución/Entrada:	
Entrada: La entrada consiste en los datos del usuario, serían los del dominio UCI.	
Pasos de ejecución: Para que un usuario se autentique en el sistema debe ir al menú e ingresar los campos de “Usuario” y “Contraseña”.	
Resultado esperado: El usuario se ha autenticado correctamente.	
Evaluación de la prueba: Satisfactoria.	

Los restantes CPA de la **primera entrega** se encuentran en el Anexo 5.

Para la **segunda entrega** quedaron definidos los siguientes CPA:

HU: 6 Gestionar comisión.

- ✓ CPA.18 Crear comisión.
- ✓ CPA.19 Visualizar comisión.
- ✓ CPA.20 Modificar comisión.

Capítulo 3: Implementación y pruebas

- ✓ CPA.21 Eliminar comisión.
HU: 12 Gestionar notificaciones.
- ✓ CPA.22 Crear notificaciones.
- ✓ CPA.23 Visualizar notificaciones.
- ✓ CPA.24 Modificar notificaciones.
- ✓ CPA.25 Eliminar notificaciones.
HU: 2 Gestionar roles y permisos.
- ✓ CPA.26 Crear roles y permisos.
- ✓ CPA.27 Visualizar roles y permisos.
- ✓ CPA.28 Modificar roles y permisos.
- ✓ CPA.29 Eliminar roles y permisos.
HU: 10 Gestionar dictamen.
- ✓ CPA.30 Crear dictamen.
- ✓ CPA.31 Visualizar dictamen.
- ✓ CPA.31 Modificar dictamen.
- ✓ CPA.33 Eliminar dictamen.

Tabla 15 CPA Crear comisión.

Caso de prueba de aceptación	
Código: HU1_CPA18	Historia de usuario: 6
Nombre: Crear comisión.	
Descripción: Prueba para la funcionalidad que permite crear una comisión en el sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador. Debe haberse cerrado el proceso de inscripción en el evento.	

Pasos de ejecución/Entrada:

Entrada: Son los datos que introduzca el usuario para que el sistema cree la comisión deseada.

Pasos de ejecución: Luego de que el usuario con el rol de Organizador se halla autenticado en el sistema, selecciona la opción de crear comisión. Se le mostrará en una vista los datos que son necesarios para crear la comisión, como son, “Nombre_Comisión”, “Integrante_Comisión” y “Trabajos_Asignados” a la comisión. Luego de guardar los datos, el sistema le mostrará la comisión creada.

Resultado esperado: La comisión ha sido creada satisfactoriamente.

Evaluación de la prueba: Satisfactoria.

Los restantes CPA de la **segunda entrega** se encuentran en el Anexo 6.

Para la **tercera entrega** quedaron definidos los siguientes CPA:

HU: 8 Gestionar comité científico.

- ✓ CPA.34 Crear comité científico.
- ✓ CPA.35 Visualizar comité científico.
- ✓ CPA.36 Modificar comité científico.
- ✓ CPA.37 Eliminar comité científico.

HU: 9 Gestionar comité organizador.

- ✓ CPA.38 Crear comité organizador.
- ✓ CPA.39 Visualizar comité organizador.
- ✓ CPA.40 Modificar comité organizador.
- ✓ CPA.41 Eliminar comité organizador.

HU: 13 Gestionar documentación del FCT.

- ✓ CPA.42 Crear documentación del FCT.
- ✓ CPA.43 Visualizar documentación del FCT.
- ✓ CPA.44 Modificar documentación del FCT.
- ✓ CPA.45 Eliminar documentación del FCT.

HU: 7 Gestionar programa.

- ✓ CPA.46 Crear programa.
- ✓ CPA.47 Visualizar programa.
- ✓ CPA.48 Modificar programa.
- ✓ CPA.49 Eliminar programa.

Tabla 16 CPA Crear comité científico.

Caso de prueba de aceptación	
Código: HU1_CPA34	Historia de usuario: 8
Nombre: Crear comité científico.	
Descripción: Prueba para la funcionalidad que permite crear un comité científico en el sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador.	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los datos que introduzca el usuario para que el sistema cree comité científico deseado.</p> <p>Pasos de ejecución: Luego de que el usuario con el rol de Organizador se halla autenticado en el sistema, selecciona la opción de crear comité científico. Se le mostrará en una vista los datos que son necesarios para crear el mismo, como son, “Presidente” y “Miembros” del comité científico. Luego de guardar los datos, el sistema le mostrará el comité científico creado.</p>	
Resultado esperado: El comité científico ha sido creado satisfactoriamente.	
Evaluación de la prueba: Satisfactoria.	

Los restantes CPA de la **tercera entrega** se encuentran en el Anexo 7.

Ejecución de las pruebas funcionales en el sistema

Las pruebas a las funcionalidades se realizaron en 3 iteraciones obteniendo para la primera iteración 8 no conformidades significativas, 6 no significativas y 2 recomendaciones. Para la segunda iteración, 5 no conformidades significativas, 4 no significativas y 2 recomendaciones y para la tercera iteración, 2 no

conformidades significativas, 1 no significativa, sin recomendaciones. Las no conformidades no significativas, se centraron en errores ortográficos como: omisiones de tildes, paréntesis y cambio de mayúscula por minúscula y las significativas, en errores de validación de campos como: validación del correo, validación de datos de entrada por el usuario.

Conclusiones del capítulo

La selección del CMS Drupal utilizado en el desarrollo de la propuesta de solución permitió cumplir con los objetivos definidos en la investigación. Se definió el patrón arquitectónico de la propuesta de solución, identificando MVC como patrón que usa por defecto Drupal 7. Se documentó todo el proceso de confección de la solución a través de los siguientes elementos: tema, usuarios, permisos, módulos, bloques y menús. La validación del sistema se realizó mediante las pruebas unitarias con el módulo Simple Test y las pruebas de aceptación realizadas a los requisitos funcionales. El análisis de estos datos arrojó que la solución desarrollada se encuentra en condiciones para ser implantada.

Conclusiones generales

- ✓ El estudio realizado sobre los sistemas relacionados con el proceso de gestión del Fórum de Ciencia y Técnica posibilitó fundamentar las bases teóricas de la solución propuesta.
- ✓ El análisis del estado del arte realizado permitió la selección de la metodología de desarrollo, las herramientas, lenguajes de programación y tecnologías más apropiadas para dar cumplimiento al objetivo general.
- ✓ La solución implementada permite la gestión del Fórum de Ciencia y Técnica de la Facultad 4.
- ✓ Las pruebas unitarias y de aceptación realizadas permitieron verificar el correcto funcionamiento de las funcionalidades definidas durante la investigación.

Recomendaciones

Como parte del proceso de investigación y desarrollo de la aplicación, surgieron ideas que son recomendables tener en cuenta para el futuro mejoramiento de la solución propuesta. Entre ellas se señalan:

- ✓ Agregar la funcionalidad de generar una estadística de participación en el evento, la cual pueda generar el porcentaje de trabajos por Categoría otorgada.
- ✓ Generalizar el presente trabajo en el resto de las facultades de la UCI.
- ✓ Implementar funcionalidades que permitan un proceso de revisión automática y por pares de acorde con las características de un sistema de gestión de eventos.

Referencias bibliográficas

1. **Calabria, Luis y Píriz, Pablo. Metodología XP. Universidad ORT Uruguay : Cátedra de**
2. **Ingeniería de Software. 2003.**
3. **Corporation, O.** (2012). Página Principal NetBeans. [Accedido 20 Enero 2015]. Obtenido de: http://netbeans.org/index_es.html.
4. **Cuerda, X. G.** (2004). CMS en el desarrollo web.
5. **Díaz-Balart, Fidel Castro. 2002.**Cuba. Amanecer del Tercer Milenio .
6. **Eclipse Foundation. 2004.** [En línea] 2004. http://wiki.eclipse.org/FAQ_What_is_Eclipse%3f.
7. **Eguíluz, J.** (2008). Introducción a JavaScript.
8. **Gauchat, Juan Diego. 2015.**El gran libro de HTML5, CSS3 y JavaScript. 2015.
9. **Gilfillan, Ian. 2015.**LA BIBLIA MySQL. 2015.
10. **Ginestá, M. G., & Pérez Mora, O.** (2015). Bases de Datos en PostgreSQL.
11. **Highsmith, J.** (2002). Agile Software Development Ecosystems. Addison-Wesley.
12. **Hispano.** Conceptos Básicos. [Accedido 19 Enero 2015]. Obtenido de: http://drupal.org/es/manuales/guia_del_usuario/conceptos_basicos.
13. **IDE. 2009.** Slideshare. Entornos-de-Desarrollos-Integrados. [En línea] 2009. <http://es.slideshare.net/GhaBiithahh/entornos-de-desarrollo-integrados>
14. **Joskowicz, J.** (2008). Reglas y Prácticas en eXtreme Programming.
15. **Kabir, M. J.** (2015). Servidor Apache 2.
16. **LETELIER, Patricio and M^a CARMEN PENADÉS.** Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). www.cyta.com.ar/ta0502/v5n2a1.htm [En línea]. 15 Abril 2006. [Accedido 18 Noviembre 2014]. Obtenido de: http://www.cyta.com.ar/ta0502/b_v5n2a1.htm
17. **Luque, L. C.** (2001). Curso de JavaScript.
18. **Mariño, C. V.** (2008). Programación en PHP 5.

19. **Mateu, C.** (2004). Desarrollo de aplicaciones web.
20. **RA-MA.** (2015). Conceptos generales de la arquitectura de aplicaciones web. [Accedido 30 Enero 2015].
21. **Pérez, J. E.** (2007). Introducción a CSS.
22. **Ramírez, L. J.** (2012). Experto en Drupal. En L. J. Ramírez.
23. **Sanz, L. F.** (2015). Tecnología, ingeniería del software. "El futuro de la tecnología de software". [Accedido 20 Enero 2015]. Obtenido de: <http://www.ati.es>.
24. **Vázquez, José Antonio Gallego.** 2003. Desarrollo Web con PHP y MySQL. 2003.
25. **Plata, Universidad Nacional de La.** 2015. Portal de Congresos UNPL Sitio de Gestión para la Organización de Congresos. [En línea] 2015. <http://congresos.unlp.edu.ar/>
26. **Infomed.** 2015. CENCOMED Centro Virtual de Convenciones de Salud. [En línea] 2015. <http://www.cencomed.sld.cu/>.
27. **©UNESCO-UIS .** 2013. UNESCO Institute for Statistics. [En línea] 2013. <http://www.uis.unesco.org>.
28. **ALEGSA.** 2015. [ALEGSA.COM.AR.](http://www.alegsa.com.ar/) [En línea] 2015. <http://www.alegsa.com.ar/Dic/lenguaje%20de%20programacion.php>.
29. **CIGET-VC & IDICT.** 2015. Portal Nacional del Fórum de Ciencia y Técnica. [En línea] 2015. <http://www.forumcyt.cu/forum/historia>.
30. **ALEGSA. (2015).** *ALEGSA.com.ar*. Obtenido de <http://www.alegsa.com.ar/Dic/sitio%20web.php>
31. **CEPYME.** (2015). *Vía Emprende.* Obtenido de <http://www.viaemprende.es/sites/default/files/manuales/ModeloNegocio.pdf>
32. **Gil, F. (2012).** *Experto en Drupal 7.*
33. **GitHub. (2015).** *GitHub.* Obtenido de <https://github.com/geneos/drupal-doc/wiki/Arquitectura-Drupal>
34. **Scribd Inc. (2015).** *Scribd.* Obtenido de <http://es.scribd.com/doc/37187866/Requerimientos-funcionales-y-no-funcionales#scribd>

35. **Torres, P. L. (2003).** *Metodologías Ágiles en el Desarrollo de Software.*
36. **Haimann, Theo. 2011.** Zona Economica. [En línea] 2011. <http://www.zonaeconomica.com/control>.
37. **Library, Simon Fraser University. 2014.** Open Conference Systems. [En línea] 2014. <https://pkp.sfu.ca/ocs/>.
38. **Webempresa. 2015.**Joomla! Spanish Documents. [En línea] 2015. <http://ayuda.joomlaspanish.org/-preguntas-frecuentes-enlaces-directos-82/388-caracteristicas-principales>.
39. **CIGET-VC & IDICT. 2015.** Portal Nacional Fórum de Ciencia y Técnica. [En línea] 2015. <http://www.forumcyt.cu/forum/conceptos>.
40. **Dirección de Investigaciones . 2015.** [En línea] 2015. <https://investigaciones.uci.cu>.
41. **Gil, Fran. 2012.** Experto en Drupal 7 Nivel Inicial. 2012.
42. **Potencier, Fabien. 2008.** Symfony la guía definitiva. 2008.
43. **CIGET. 2006.** [En línea] 2006. [Citado el: 10 de Mayo de 2015.] <http://www.forum.villaclara.cu/>.
44. **Gauchat, Juan Diego. 2012.**El gran libro de HTML 5, CSS3 y JavaScript. 2012.
45. **Khachtourians, Ariane. 2015.** Affinitybridge. Drupal SimpleTest Module Abridged. . [En línea] 2015. <http://affinitybridge.com/blog/drupal-simpletest-module-abridged>.
46. NetBeans. 2015. netbeans.org. [En línea] 2015. <http://netbeans.org>.
47. **CEPYME. (2015).** *Vía Emprende.* Obtenido de <http://www.viaemprende.es/sites/default/files/manuales/ModeloNegocio.pdf>
48. **GitHub. (2015).** *GitHub.* Obtenido de <https://github.com/geneos/drupal-doc/wiki/Arquitectura-Drupal>
49. **Scribd Inc. (2015).** *Scribd.* Obtenido de <http://es.scribd.com/doc/37187866/Requerimientos-funcionales-y-no-funcionales#scribd>
50. **Torres, P. L. (2003).** *Metodologías Ágiles en el Desarrollo de Software.*

Bibliografía consultada

- ✓ **CIGET-VC & IDICT. 2015.** Portal Nacional Fórum de Ciencia y Técnica. [En línea] 2015. <http://www.forumcyt.cu/forum/conceptos>.
- ✓ **Dirección de Investigaciones . 2015.** [En línea] 2015. <https://investigaciones.uci.cu>.
- ✓ **Gil, Fran. 2012.***Experto en Drupal 7 Nivel Inicial*. 2012.
- ✓ **IBM. 2010.**Rational Rose Enterprise. [En línea] 2010. [Citado el: 20 de abril de 2015.] [http://www-142.ibm.com/software/products/es/es/enterprise/..](http://www-142.ibm.com/software/products/es/es/enterprise/)
- ✓ **Monografías . 2015.** Monografías.com. [En línea] 2015. [Citado el: 5 de Mayo de 2015.] [ww.monografias.com/trabajos-pdf/direccion-integrada-de-proyectos/direccion-integrada-de-proyectos.shtml](http://www.monografias.com/trabajos-pdf/direccion-integrada-de-proyectos/direccion-integrada-de-proyectos.shtml).
- ✓ **Nobrega, María de. 2005.** Curso Sistemas de Información II. [En línea] 2005. [Citado el: 5 de Mayo de 2015.] http://curso_sin2.blogia.com/2005/060401-herramientas-case-rational-rose.-por-maria-de-nobrega.php.
- ✓ **Potencier, Fabien. 2008.***Symfony la guía definitiva*. 2008.
- ✓ **Tobón, Luis Miguel Hecheverry. 2007.***Caso Práctico de la metodología ágil XP al desarrollo de software*. 2007.
- ✓ **Universidad Carlos III de Madrid. 2015.** KNOWLEDGE Reuse Group. [En línea] 2015. [Citado el: 5 de Mayo de 2015.] <http://www.ie.inf.uc3m.es/grupo/docencia/reglada/ls1y2/PracticaVP.pdf>.
- ✓ **Beck, Kent. 2004.** *Extreme Programming Explained*. 2004.
- ✓ **cibernetia. 2015.**cibernetia. [En línea] 2015. [Citado el: 25 de Mayo de 2015.] http://www.cibernetia.com/manuales/instalacion_servidor_web/1_conceptos_basicos.php.
- ✓ **Joskowicz, José. 2008.***Reglas y Prácticas en eXtreme Programming*. 2008.
- ✓ **León, Rolando Alfredo Hernández. 2002.** *EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTÍFICA*. 2002.
- ✓ **Molina, Jesús García. 2015.** Ciencia y Técnica Administrativa. [En línea] 2015. [Citado el: 21 de Mayo de 2015.] <http://www.cyta.com.ar/ta0604/v6n4a1.htm>.

- ✓ **Piwen, Ariel Erlijman. 2001.***PROBLEMAS Y SOLUCIONES EN LA IMPLEMENTACIÓN DE EXTREME PROGRAMMING*. 2001.
- ✓ **Real Academia Española. 2015.** Real Academia Española. [En línea] 2015. [Citado el: 25 de Mayo de 2015.] <http://lema.rae.es/drae/?val=evento>.

Anexos

Anexo 1 Historias de usuarios

Tabla 7 HU Gestionar convocatoria

Historia de Usuario	
Nombre: Gestionar convocatoria	Número: 3
Usuario: Organizador	
Complejidad: Alta	Prioridad: Alta
Iteración: 1	Estimación: 3 días
Descripción: El sistema permitirá crear la convocatoria, modificar convocatoria y eliminar convocatoria, esto solo lo puede hacer el Organizador. Una vez creada, el Organizador podrá ver los datos de la convocatoria desde la aplicación. Se podrá generar la convocatoria en formato PDF.	
Observación: El usuario deberá estar autenticado con el rol Organizador y además el sistema le indicará los campos que son obligatorios para gestionar una convocatoria determinada.	

Tabla 8 HU Gestionar roles y permisos

Historia de Usuario	
Nombre: Gestionar roles y permisos.	Número: 2
Usuario: Administrador	
Complejidad: Media	Prioridad: Alta
Iteración: 2	Estimación: 1
Descripción: El sistema permitirá dar los permisos y quitar un rol a un determinado usuario dentro del sistema, esto solo lo puede hacer el Administrador.	

Observación: El usuario deberá estar autenticado con el rol Administrador y además el sistema le indicará los campos que son obligatorios para gestionar los roles y permisos de un usuario determinado.

Tabla 9 HU Gestionar banco de problemas

Historia de Usuario	
Nombre: Gestionar banco de problemas	Número: 5
Usuario: Organizador	
Complejidad: Alta	Prioridad: Alta
Iteración: 1	Estimación: 1
Descripción: El sistema permitirá crear el banco de problemas, modificar y eliminar problemas, esto solo lo puede hacer el Organizador. Una vez creado el mismo, el Organizador podrá ver los datos de los problemas desde la aplicación. Se podrá generar cada problema en formato PDF.	
Observación: El usuario deberá estar autenticado con el rol Organizador y además el sistema le indicará los campos que son obligatorios para gestionar un problema determinado.	

Tabla 10 HU Gestionar comisión

Historia de usuario	
Nombre: Gestionar comisión	Número: 6
Usuario: Organizador	
Complejidad: Media	Prioridad: Media
Iteración: 2	Estimación: 1
Descripción: El sistema permitirá crear las comisiones, editar las comisiones y eliminar las comisiones. Esto solo lo puede hacer el Organizador. Una vez creada las	

mismas el Organizador podrá ver los datos de las comisiones desde la aplicación.
Observación: El usuario deberá estar autenticado con el rol Organizador y además el sistema le indicará los campos que son obligatorios para gestionar una comisión determinada.

Tabla 11 HU Gestionar programa

Historia de Usuario	
Nombre: Gestionar programa	Número: 7
Usuario: Organizador	
Complejidad: Baja	Prioridad: Baja
Iteración: 3	Estimación: 2 días
Descripción: El sistema permitirá adicionar programa, editar y eliminar programa, esto solo lo puede hacer el Organizador. Una vez creado el mismo, el Organizador podrá ver los datos del programa desde la aplicación.	
Observación: El usuario deberá estar autenticado con el rol Organizador y además el sistema le indicará los campos para gestionar un programa determinado.	

Tabla 12 HU Gestionar comité científico

Historia de Usuario	
Nombre: Gestionar comité científico	Número: 8
Usuario: Organizador	
Complejidad: Baja	Prioridad: Baja
Iteración: 3	Estimación: 3 días
Descripción: El sistema permitirá crear el comité científico, modificar y eliminar el comité científico, esto solo lo puede hacer el Organizador. Una vez creado el mismo, el Organizador podrá ver los datos del comité científico desde la aplicación. Se podrá generar el comité científico en formato PDF.	

Observación: El usuario deberá estar autenticado con el rol Organizador y además el sistema le indicará los campos que son obligatorios para gestionar comité científico.

Tabla 13 HU Gestionar comité organizador

Historia de Usuario	
Nombre: Gestionar comité organizador	Número: 9
Usuario: Organizador	
Complejidad: Baja	Prioridad: Baja
Iteración: 3	Estimación: 3 días
Descripción: El sistema permitirá crear el comité organizador, modificar y eliminar el comité organizador, esto solo lo puede hacer el Organizador. Una vez creado el mismo, el Organizador podrá ver los datos del comité organizador desde la aplicación. Se podrá generar el comité organizador en formato PDF.	
Observación: El usuario deberá estar autenticado con el rol Organizador y además el sistema le indicará los campos que son obligatorios para gestionar comité organizador.	

Tabla 13 HU Gestionar dictamen

Historia de Usuario	
Nombre: Gestionar dictamen	Número: 10
Usuario: Organizador	
Complejidad: Media	Prioridad: Media
Iteración: 2	Estimación: 1
Descripción: El sistema permitirá crear el dictamen de evaluación de los participantes en el evento, modificar el dictamen y eliminar dictamen, el dictamen es otorgado por los presidentes de las comisiones, pero es el Organizador el encargado de su publicación final. Una vez creado el mismo, el Organizador podrá ver los datos del dictamen otorgado desde la aplicación. Se podrá generar el dictamen en formato	

PDF.
Observación: El usuario deberá estar autenticado con el rol Organizador y además el sistema le indicará los campos que son obligatorios para gestionar el dictamen de participación en el evento.

Tabla 14 HU Gestionar evento

Historia de Usuario	
Nombre: Gestionar evento	Número: 11
Usuario: Organizador	
Complejidad: Alta	Prioridad: Media
Iteración: 1	Estimación: 1
Descripción: El sistema permitirá crear el evento, modificar el evento y eliminar el evento. Se podrá generar un evento en formato PDF.	
Observación: El usuario deberá estar autenticado con el rol Organizador y además el sistema le indicará los campos que son obligatorios para gestionar el evento.	

Tabla 15 HU Gestionar notificaciones

Historia de Usuario	
Nombre: Gestionar notificaciones	Número: 12
Usuario: Organizador	
Complejidad: Media	Prioridad: Media
Iteración: 2	Estimación: 1
Descripción: El sistema permitirá crear las notificaciones, modificar y eliminar notificaciones.	
Observación: El usuario deberá estar autenticado con el rol Organizador y además el sistema le indicará los campos que son obligatorios para gestionar las notificaciones.	

Tabla 16 HU Gestionar documentación del FCT

Historia de Usuario	
Nombre: Gestionar documentación del FCT	Número: 13
Usuario: Organizador	
Complejidad: Baja	Prioridad: Media
Iteración: 3	Estimación: 1
Descripción: El sistema permitirá crear la documentación del FCT, modificar y eliminar documentación del FCT.	
Observación: El usuario deberá estar autenticado con el rol Organizador y además el sistema le indicará los campos que son obligatorios para gestionar la documentación del FCT.	

Anexo 2 Tareas de ingeniería para la iteración 1

Tarea	
Número de tarea: 2	Número de HU: 11
Nombre de la tarea: Crear evento.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Fecha inicio: 5 de enero de 2015	Fecha fin: 6 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema creará el evento.	

Tarea

Número de tarea: 3	Número de HU: 11
Nombre de la tarea: Visualizar evento.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 7 de enero de 2015	Fecha fin: 7 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema mostrará el evento creado.	

Tarea	
Número de tarea: 4	Número de HU: 11
Nombre de la tarea: Modificar evento.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 8 de enero de 2015	Fecha fin: 8 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar el evento creado.	

Tarea	
Número de tarea: 5	Número de HU: 11

Nombre de la tarea: Eliminar evento.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 9 de enero de 2015	Fecha fin: 9 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permite eliminar el evento creado.	

Tarea	
Número de tarea: 6	Número de HU: 3
Nombre de la tarea: Crear convocatoria.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 12 de enero de 2015	Fecha fin: 12 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear la convocatoria.	

Tarea	
Número de tarea: 7	Número de HU: 3
Nombre de la tarea: Visualizar convocatoria.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día

Fecha inicio: 12 de enero de 2015	Fecha fin: 12 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá visualizar la convocatoria previamente creada.	

Tarea	
Número de tarea: 8	Número de HU: 3
Nombre de la tarea: Modificar convocatoria.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 13 de enero de 2015	Fecha fin: 13 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar la convocatoria.	

Tarea	
Número de tarea: 9	Número de HU: 3
Nombre de la tarea: Eliminar convocatoria.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 13 de enero de 2015	Fecha fin: 13 de enero de 2015
Programador responsable: José Antonio Sardina López	

Descripción: El sistema permitirá eliminar la convocatoria.

Tarea	
Número de tarea: 10	Número de HU: 4
Nombre de la tarea: Crear inscripción.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Fecha inicio: 14 de enero de 2015	Fecha fin: 15 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear la inscripción.	

Tarea	
Número de tarea: 11	Número de HU: 4
Nombre de la tarea: Visualizar inscripción.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 16 de enero de 2015	Fecha fin: 16 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá visualizar la inscripción.	

Tarea	
Número de tarea: 12	Número de HU: 4
Nombre de la tarea: Modificar inscripción.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 19 de enero de 2015	Fecha fin: 19 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar la inscripción.	

Tarea	
Número de tarea: 13	Número de HU: 4
Nombre de la tarea: Eliminar inscripción.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 20 de enero de 2015	Fecha fin: 20 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá eliminar la inscripción.	

Tarea	
Número de tarea: 14	Número de HU: 5

Nombre de la tarea: Crear banco de problemas.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Fecha inicio: 21 de enero de 2015	Fecha fin: 22 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear el banco de problemas.	

Tarea	
Número de tarea: 15	Número de HU: 5
Nombre de la tarea: Visualizar banco de problemas.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 23 de enero de 2015	Fecha fin: 23 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá visualizar el banco de problemas.	

Tarea	
Número de tarea: 16	Número de HU: 5
Nombre de la tarea: Modificar banco de problemas.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día

Fecha inicio: 26 de enero de 2015	Fecha fin: 26 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar el banco de problemas.	

Tarea	
Número de tarea: 17	Número de HU: 5
Nombre de la tarea: Eliminar banco de problemas.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 27 de enero de 2015	Fecha fin: 27 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá eliminar el banco de problemas.	

Anexo 3 Tareas de ingeniería para la iteración 2

Tarea	
Número de tarea: 18	Número de HU: 6
Nombre de la tarea: Crear comisión.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Fecha inicio: 28 de enero de 2015	Fecha fin: 29 de enero de 2015

Programador responsable: José Antonio Sardina López
Descripción: El sistema permitirá crear una determinada comisión.

Tarea	
Número de tarea: 19	Número de HU: 6
Nombre de la tarea: Visualizar comisión.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 30 de enero de 2015	Fecha fin: 30 de enero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá visualizar una determinada comisión.	

Tarea	
Número de tarea: 20	Número de HU: 6
Nombre de la tarea: Modificar comisión.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 2 de febrero de 2015	Fecha fin: 2 de febrero de 2015
Programador responsable: José Antonio Sardina López	

Descripción: El sistema permitirá modificar una determinada comisión.

Tarea	
Número de tarea: 21	Número de HU: 6
Nombre de la tarea: Eliminar comisión.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 3 de febrero de 2015	Fecha fin: 3 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá eliminar una determinada comisión.	

Tarea	
Número de tarea: 22	Número de HU: 12
Nombre de la tarea: Crear notificaciones.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Fecha inicio: 4 de febrero de 2015	Fecha fin: 5 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear las notificaciones.	

Tarea	
Número de tarea: 23	Número de HU: 12
Nombre de la tarea: Visualizar notificaciones.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 6 de febrero de 2015	Fecha fin: 6 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá visualizar las notificaciones.	

Tarea	
Número de tarea: 24	Número de HU: 12
Nombre de la tarea: Modificar notificaciones.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 9 de febrero de 2015	Fecha fin: 9 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar las notificaciones.	

Tarea	
Número de tarea: 25	Número de HU: 12

Nombre de la tarea: Eliminar notificaciones	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 10 de febrero de 2015	Fecha fin: 10 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá eliminar notificaciones.	

Tarea	
Número de tarea: 26	Número de HU: 2
Nombre de la tarea: Crear roles y permisos.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Fecha inicio: 11 de febrero de 2015	Fecha fin: 12 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear roles y permisos.	

Tarea	
Número de tarea: 27	Número de HU: 2
Nombre de la tarea: Visualizar roles y permisos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día

Fecha inicio: 13 de febrero de 2015	Fecha fin: 13 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá visualizar los roles y permisos.	

Tarea	
Número de tarea: 28	Número de HU: 2
Nombre de la tarea: Modificar roles y permisos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 16 de febrero de 2015	Fecha fin: 16 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar los roles y permisos.	

Tarea	
Número de tarea: 29	Número de HU: 2
Nombre de la tarea: Eliminar roles y permisos.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 17 de febrero de 2015	Fecha fin: 17 de febrero de 2015
Programador responsable: José Antonio Sardina López	

Descripción: El sistema permitirá eliminar roles y permisos.

Tarea	
Número de tarea: 30	Número de HU: 10
Nombre de la tarea: Crear dictamen.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Fecha inicio: 18 de febrero de 2015	Fecha fin: 19 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear el dictamen.	

Tarea	
Número de tarea: 31	Número de HU: 10
Nombre de la tarea: Visualizar dictamen.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 20 de febrero de 2015	Fecha fin: 20 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá visualizar el dictamen.	

Tarea	
Número de tarea: 32	Número de HU: 10
Nombre de la tarea: Modificar dictamen.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 23 de febrero de 2015	Fecha fin: 23 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar el dictamen.	

Tarea	
Número de tarea: 33	Número de HU: 10
Nombre de la tarea: Eliminar dictamen.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 24 de febrero de 2015	Fecha fin: 24 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá eliminar dictamen.	

Anexo 4 Tareas de ingeniería para la iteración 3

Tarea

Número de tarea: 34	Número de HU: 8
Nombre de la tarea: Crear comité científico.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 25 de febrero de 2015	Fecha fin: 25 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear un comité científico.	

Tarea	
Número de tarea: 35	Número de HU: 8
Nombre de la tarea: Crear comité científico.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.1
Fecha inicio: 26 de febrero de 2015	Fecha fin: 26 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear un comité científico.	

Tarea	
Número de tarea: 36	Número de HU: 8

Nombre de la tarea: Visualizar comité científico.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.1
Fecha inicio: 27 de febrero de 2015	Fecha fin: 27 de febrero de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá visualizar un comité científico.	
Tarea	
Número de tarea: 36	Número de HU: 8
Nombre de la tarea: Modificar comité científico.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 2 de marzo de 2015	Fecha fin: 2 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar un comité científico.	

Tarea	
Número de tarea: 37	Número de HU: 8
Nombre de la tarea: Eliminar comité científico.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día

Fecha inicio: 3 de marzo de 2015	Fecha fin: 3 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá eliminar un comité científico.	

Tarea	
Número de tarea: 38	Número de HU: 9
Nombre de la tarea: Crear comité organizador.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.1
Fecha inicio: 4 de marzo de 2015	Fecha fin: 4 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear un comité organizador.	

Tarea	
Número de tarea: 39	Número de HU: 9
Nombre de la tarea: Visualizar comité organizador.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.1
Fecha inicio: 5 de marzo de 2015	Fecha fin: 5 de marzo de 2015
Programador responsable: José Antonio Sardina López	

Descripción: El sistema permitirá visualizar comité organizador.

Tarea	
Número de tarea: 40	Número de HU: 9
Nombre de la tarea: Modificar comité organizador.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 6 de marzo de 2015	Fecha fin: 6 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar un comité organizador.	

Tarea	
Número de tarea: 41	Número de HU: 9
Nombre de la tarea: Eliminar comité organizador.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 9 de marzo de 2015	Fecha fin: 9 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá eliminar un comité organizador.	

Tarea	
Número de tarea: 42	Número de HU: 13
Nombre de la tarea: Crear documentación del FCT.	
Tipo de tarea: Desarrollo	Puntos estimados: 2 días
Fecha inicio: 10 de marzo de 2015	Fecha fin: 11 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear la documentación del FCT.	

Tarea	
Número de tarea: 43	Número de HU: 13
Nombre de la tarea: Visualizar documentación del FCT.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 12 de marzo de 2015	Fecha fin: 12 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá visualizar la documentación del FCT.	

Tarea	
Número de tarea: 44	Número de HU: 13

Nombre de la tarea: Modificar documentación del FCT.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 13 de marzo de 2015	Fecha fin: 13 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar la documentación del FCT.	

Tarea	
Número de tarea: 45	Número de HU: 13
Nombre de la tarea: Eliminar documentación del FCT.	
Tipo de tarea: Desarrollo	Puntos estimados: 1 día
Fecha inicio: 16 de marzo de 2015	Fecha fin: 16 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar la documentación del FCT.	

Tarea	
Número de tarea: 46	Número de HU: 7
Nombre de la tarea: Crear programa.	

Tipo de tarea: Desarrollo	Puntos estimados: 0.1
Fecha inicio: 17 de marzo de 2015	Fecha fin: 17 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá crear el programa.	

Tarea	
Número de tarea: 47	Número de HU: 7
Nombre de la tarea: Visualizar programa.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.1
Fecha inicio: 18 de marzo de 2015	Fecha fin: 18 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá visualizar el programa.	

Tarea	
Número de tarea: 48	Número de HU: 7
Nombre de la tarea: Modificar programa.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.1

Fecha inicio: 19 de marzo de 2015	Fecha fin: 19 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá modificar el programa.	

Tarea	
Número de tarea: 49	Número de HU: 7
Nombre de la tarea: Eliminar programa.	
Tipo de tarea: Desarrollo	Puntos estimados: 0.1
Fecha inicio: 20 de marzo de 2015	Fecha fin: 20 de marzo de 2015
Programador responsable: José Antonio Sardina López	
Descripción: El sistema permitirá eliminar el programa.	

Anexo 5 Casos de prueba de aceptación para la primera entrega.

Caso de prueba de aceptación	
Código: HU11_CPA2	Historia de usuario: 11
Nombre: Crear evento.	
Descripción: Prueba para la funcionalidad que permite crear un evento en el sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador.	

<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los datos que introduzca el usuario para que el sistema cree el evento deseado.</p> <p>Pasos de ejecución: Luego de que el usuario con el rol de Organizador se halla autenticado en el sistema, selecciona la opción de crear evento. Se le mostrará en una vista los datos que son necesarios para crear el mismo, como son, "Nombre del Evento", "Año", "Fecha de inicio" y "Fecha fin".</p>
<p>Resultado esperado: El evento ha sido creado satisfactoriamente.</p>
<p>Evaluación de la prueba: Satisfactoria.</p>

Caso de prueba de aceptación	
Código: HU11_CPA3	Historia de usuario: 11
Nombre: Visualizar evento.	
Descripción: Prueba para la funcionalidad que permite visualizar un evento en el sistema.	
<p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado con el rol de Organizador.</p> <p>Debe haber al menos un evento creado.</p>	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los datos que introduzca el usuario para que el sistema muestre el evento deseado.</p> <p>Pasos de ejecución: Luego de que el usuario con el rol de Organizador se halla autenticado en el sistema, puede seleccionar la forma en que se visualizará el evento anteriormente creado, así como los campos que quiere que se le muestren a los usuarios.</p>	

Resultado esperado: El evento se visualiza correctamente.
Evaluación de la prueba: Satisfactoria.

Caso de prueba de aceptación	
Código: HU11_CPA4	Historia de usuario: 11
Nombre: Modificar evento.	
Descripción: Prueba para la funcionalidad que permite modificar un evento en el sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador. Debe haber al menos un evento creado.	
Pasos de ejecución/Entrada: Entrada: Son los datos que introduzca el usuario para que el sistema modifique el evento deseado. Pasos de ejecución: Luego de que el usuario con el rol de Organizador se halla autenticado en el sistema, puede seleccionar la opción de modificar el evento deseado. El sistema le permitirá modificar los valores del mismo a su preferencia.	
Resultado esperado: El evento se modifica correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU11_CPA5	Historia de usuario: 11

Nombre: Eliminar evento.
Descripción: Prueba para la funcionalidad que permite eliminar un evento en el sistema.
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador. Debe haber al menos un evento creado.
Pasos de ejecución/Entrada: Entrada: Son los datos que introduzca el usuario para que el sistema elimine el evento deseado. Pasos de ejecución: Luego de que el usuario con el rol de Organizador se halla autenticado en el sistema, puede seleccionar la opción de eliminar el evento deseado. El sistema le permitirá eliminar el evento de manera definitiva del sistema.
Resultado esperado: El evento se eliminó correctamente.
Evaluación de la prueba: Satisfactoria.

Caso de prueba de aceptación	
Código: HU3_CPA6	Historia de usuario: 3
Nombre: Crear convocatoria.	
Descripción: Prueba para la funcionalidad que permite crear la convocatoria en el sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador.	
Pasos de ejecución/Entrada: Entrada: Son los datos que introduzca el usuario para que el sistema cree la	

convocatoria.
Pasos de ejecución: Luego de que el usuario con el rol de Organizador se halla autenticado en el sistema, puede seleccionar la opción crear la convocatoria del evento. El sistema le mostrará los campos que son necesarios para crear la convocatoria.
Resultado esperado: La convocatoria se creo correctamente.
Evaluación de la prueba: Satisfactoria.

Caso de prueba de aceptación	
Código: HU3_CPA7	Historia de usuario: 3
Nombre: Visualizar convocatoria.	
Descripción: Prueba para la funcionalidad que permite visualizar la convocatoria en el sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador. Debe haber al menos una convocatoria creada.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema le muestre la convocatoria. Pasos de ejecución: Luego de que cualquier usuario entre al sistema y se halla creado al menos una convocatoria, el sistema le brindará la posibilidad de visualizarla en el sistema.	
Resultado esperado: La convocatoria se visualizó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU3_CPA8	Historia de usuario: 3
Nombre: Modificar convocatoria.	
Descripción: Prueba para la funcionalidad que permite modificar la convocatoria en el sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador. Debe haber al menos una convocatoria creada.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificar la convocatoria. Pasos de ejecución: Luego de que el usuario con rol Organizador entre al sistema y se halla creado al menos una convocatoria, el sistema le brindará la posibilidad de modificar dicha convocatoria en el sistema.	
Resultado esperado: La convocatoria se modificó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU3_CPA9	Historia de usuario: 3
Nombre: Eliminar convocatoria.	
Descripción: Prueba para la funcionalidad que permite eliminar una convocatoria en el sistema.	
Condiciones de ejecución:	

El usuario debe estar autenticado con el rol de Organizador.
Debe haber al menos una convocatoria creada.
Pasos de ejecución/Entrada:
Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminar la convocatoria.
Pasos de ejecución: Luego de que el usuario con rol Organizador entre al sistema y se halla creado al menos una convocatoria, el sistema le brindará la posibilidad de eliminar una convocatoria en el sistema.
Resultado esperado: La convocatoria se eliminó correctamente.
Evaluación de la prueba: Satisfactoria.

Caso de prueba de aceptación	
Código: HU3_CPA10	Historia de usuario: 4
Nombre: Crear inscripción.	
Descripción: Prueba para la funcionalidad que permite crear la inscripción en el evento.	
Condiciones de ejecución:	
El usuario debe estar autenticado en el sistema.	
Debe haber al menos un evento creado.	
Pasos de ejecución/Entrada:	
Entrada: Son los pasos que ejecuta el usuario para que el sistema permita crear la inscripción.	
Pasos de ejecución: Luego de que el usuario se halla autenticado en el sistema, el sistema le brindará la posibilidad de crear la inscripción en el evento.	
Resultado esperado: La inscripción se creó correctamente.	

Evaluación de la prueba: Satisfactoria.
--

Caso de prueba de aceptación	
Código: HU3_CPA11	Historia de usuario: 4
Nombre: Visualizar inscripción.	
Descripción: Prueba para la funcionalidad que permite visualizar la inscripción en el evento.	
Condiciones de ejecución: El usuario debe estar autenticado en el sistema. Debe haber al menos una inscripción creada por ese usuario en el evento.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita visualizar la inscripción. Pasos de ejecución: Luego de que el usuario se halla autenticado en el sistema, el sistema le brindará la posibilidad de visualizar la inscripción anteriormente creada por el mismo en el evento.	
Resultado esperado: La inscripción se visualizó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU3_CPA12	Historia de usuario: 4
Nombre: Modificar inscripción.	
Descripción: Prueba para la funcionalidad que permite modificar la inscripción en el evento.	

<p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado en el sistema.</p> <p>Debe haber al menos una inscripción creada por ese usuario en el evento.</p> <p>El proceso de inscripción no halla serrado.</p>
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificar la inscripción.</p> <p>Pasos de ejecución: Luego de que el usuario se halla autenticado en el sistema, el sistema le brindará la posibilidad de modificar la inscripción anteriormente creada por el mismo en el evento.</p>
<p>Resultado esperado: La inscripción se modificó correctamente.</p>
<p>Evaluación de la prueba: Satisfactoria.</p>

Caso de prueba de aceptación	
Código: HU3_CPA13	Historia de usuario: 4
Nombre: Eliminar inscripción.	
Descripción: Prueba para la funcionalidad que permite eliminar la inscripción en el evento.	
<p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado en el sistema.</p> <p>Debe haber al menos una inscripción creada por ese usuario en el evento.</p> <p>El proceso de inscripción no halla serrado.</p>	

<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminar la inscripción.</p> <p>Pasos de ejecución: Luego de que el usuario se halla autenticado en el sistema, el sistema le brindará la posibilidad de eliminar la inscripción anteriormente creada por el mismo en el evento.</p>
<p>Resultado esperado: La inscripción se eliminó correctamente.</p>
<p>Evaluación de la prueba: Satisfactoria.</p>

Caso de prueba de aceptación	
Código: HU3_CPA14	Historia de usuario: 5
Nombre: Crear banco de problemas.	
Descripción: Prueba para la funcionalidad que permite crear el banco de problemas.	
Condiciones de ejecución:	
El usuario debe estar autenticado con el rol de Organizador.	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita crear el banco de problemas.</p> <p>Pasos de ejecución: Luego de que el usuario se halla autenticado con el rol de Organizador en el sistema, el sistema le brindará la posibilidad de crear el banco de problemas.</p>	
Resultado esperado: El banco de problemas se creó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación

Código: HU3_CPA15	Historia de usuario: 5
Nombre: Visualizar banco de problemas.	
Descripción: Prueba para la funcionalidad que permite visualizar el banco de problemas.	
Condiciones de ejecución: Debe haber al menos un problema creado.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita visualizar el banco de problemas. Pasos de ejecución: Luego de que el usuario acceda al sistema se le brindará la posibilidad de visualizar el banco de problemas.	
Resultado esperado: El banco de problemas se visualizó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación

Código: HU3_CPA16	Historia de usuario: 5
Nombre: Modificar banco de problemas.	
Descripción: Prueba para la funcionalidad que permite modificar el banco de problemas.	
Condiciones de ejecución: Debe haber al menos un problema creado. El usuario debe estar autenticado con el rol de Organizador.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificar el	

banco de problemas.
Pasos de ejecución: Luego de que el usuario con el rol de Organizador acceda al sistema, se le brindará la posibilidad de modificar el banco de problemas.
Resultado esperado: El banco de problemas se modificó correctamente.
Evaluación de la prueba: Satisfactoria.

Caso de prueba de aceptación	
Código: HU3_CPA17	Historia de usuario: 5
Nombre: Eliminar banco de problemas.	
Descripción: Prueba para la funcionalidad que permite eliminar el banco de problemas.	
Condiciones de ejecución: Debe haber al menos un problema creado. El usuario debe estar autenticado con el rol de Organizador.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminar el banco de problemas. Pasos de ejecución: Luego de que el usuario con el rol de Organizador acceda al sistema, se le brindará la posibilidad de eliminar el banco de problemas.	
Resultado esperado: El banco de problemas se eliminó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Anexo 6 Casos de prueba de aceptación para la segunda entrega.

Caso de prueba de aceptación

Código: HU3_CPA19	Historia de usuario: 6
Nombre: Visualizar comisión.	
Descripción: Prueba para la funcionalidad que permite visualizar las comisiones creadas.	
Condiciones de ejecución: Debe haber al menos una comisión creada.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita visualizar las comisiones. Pasos de ejecución: Luego de que el usuario acceda al sistema, se le brindará la posibilidad de visualizar las comisiones creadas.	
Resultado esperado: Las comisiones se mostraron correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación

Código: HU3_CPA20	Historia de usuario: 6
Nombre: Modificar comisión.	
Descripción: Prueba para la funcionalidad que permite modificar las comisiones creadas.	
Condiciones de ejecución: Debe haber al menos una comisión creada. El usuario debe estar autenticado con el rol de Organizador.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificar las	

comisiones.
Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad de modificar las comisiones creadas.
Resultado esperado: Las comisiones se modificaron correctamente.
Evaluación de la prueba: Satisfactoria.

Caso de prueba de aceptación	
Código: HU3_CPA21	Historia de usuario: 6
Nombre: Eliminar comisión.	
Descripción: Prueba para la funcionalidad que permite eliminar las comisiones creadas.	
Condiciones de ejecución: Debe haber al menos una comisión creada. El usuario debe estar autenticado con el rol de Organizador.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminar las comisiones. Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad de eliminar las comisiones creadas.	
Resultado esperado: Las comisiones se eliminaron correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación

Código: HU3_CPA22	Historia de usuario: 12
Nombre: Crear notificaciones.	
Descripción: Prueba para la funcionalidad que permite crear las notificaciones del sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita crear las notificaciones del sistema. Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad crear las notificaciones del sistema.	
Resultado esperado: Las notificaciones se crearon correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación

Código: HU3_CPA23	Historia de usuario: 12
Nombre: Visualizar notificaciones.	
Descripción: Prueba para la funcionalidad que permite visualizar las notificaciones del sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador. Debe haber al menos una notificación creada.	

<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita visualizar las notificaciones del sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad visualizar las notificaciones del sistema.</p>
<p>Resultado esperado: Las notificaciones se visualizaron correctamente.</p>
<p>Evaluación de la prueba: Satisfactoria.</p>

Caso de prueba de aceptación	
Código: HU3_CPA24	Historia de usuario: 12
Nombre: Modificar notificaciones.	
Descripción: Prueba para la funcionalidad que permite modificar las notificaciones del sistema.	
<p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado con el rol de Organizador.</p> <p>Debe haber al menos una notificación creada.</p>	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificar las notificaciones del sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad modificar las notificaciones del sistema.</p>	
Resultado esperado: Las notificaciones se modificaron correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU3_CPA25	Historia de usuario: 12
Nombre: Eliminar notificaciones.	
Descripción: Prueba para la funcionalidad que permite eliminar las notificaciones del sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador. Debe haber al menos una notificación creada.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminar las notificaciones del sistema. Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad eliminar las notificaciones del sistema.	
Resultado esperado: Las notificaciones se eliminaron correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU3_CPA26	Historia de usuario: 2
Nombre: Crear roles y permisos.	
Descripción: Prueba para la funcionalidad que permite crear los roles y permisos del sistema.	
Condiciones de ejecución:	

El usuario debe estar autenticado con el rol de Administrador.
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita crear los roles y permisos del sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Administrador, se le brindará la posibilidad de crear los roles y permisos del sistema.</p>
Resultado esperado: Los roles y permisos se crearon correctamente.
Evaluación de la prueba: Satisfactoria.

Caso de prueba de aceptación	
Código: HU3_CPA27	Historia de usuario: 2
Nombre: Visualizar roles y permisos.	
Descripción: Prueba para la funcionalidad que permite visualizar los roles y permisos del sistema.	
<p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado con el rol de Administrador.</p>	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita visualizar los roles y permisos del sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Administrador, se le brindará la posibilidad de visualizar los roles y permisos del sistema.</p>	
Resultado esperado: Los roles y permisos se visualizaron correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU3_CPA28	Historia de usuario: 2
Nombre: Modificar roles y permisos.	
Descripción: Prueba para la funcionalidad que permite modificar los roles y permisos del sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Administrador.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificar los roles y permisos del sistema. Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Administrador, se le brindará la posibilidad de modificar los roles y permisos del sistema.	
Resultado esperado: Los roles y permisos se modificaron correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU3_CPA29	Historia de usuario: 2
Nombre: Eliminar roles y permisos	
Descripción: Prueba para la funcionalidad que permite eliminar los roles y permisos del sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Administrador.	

<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminar los roles y permisos del sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Administrador, se le brindará la posibilidad de eliminar los roles y permisos del sistema.</p>
<p>Resultado esperado: Los roles y permisos se eliminaron correctamente.</p>
<p>Evaluación de la prueba: Satisfactoria.</p>

Caso de prueba de aceptación	
Código: HU3_CPA30	Historia de usuario: 10
Nombre: Crear dictamen.	
Descripción: Prueba para la funcionalidad que permite crear el dictamen en el sistema.	
<p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado con el rol de Integrante.</p> <p>Debe haberse realizado por lo menos una inscripción en el evento.</p>	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita crear el dictamen en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Integrante, se le brindará la posibilidad de crear el dictamen en el sistema.</p>	
Resultado esperado: El dictamen se creó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación

Código: HU3_CPA31	Historia de usuario: 10
Nombre: Visualizar dictamen.	
Descripción: Prueba para la funcionalidad que permite visualizar el dictamen en el sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Integrante. Debe haberse creado al menos un dictamen.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita visualizar el dictamen en el sistema. Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Integrante, se le brindará la posibilidad de visualizar el dictamen anteriormente creado en el sistema.	
Resultado esperado: El dictamen se visualizó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación

Código: HU3_CPA32	Historia de usuario: 10
Nombre: Modificar dictamen.	
Descripción: Prueba para la funcionalidad que permite modificar el dictamen en el sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Integrante. Debe haberse creado al menos un dictamen.	

<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificar el dictamen en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Integrante, se le brindará la posibilidad de modificar el dictamen anteriormente creado en el sistema.</p>
<p>Resultado esperado: El dictamen se modificó correctamente.</p>
<p>Evaluación de la prueba: Satisfactoria.</p>

Caso de prueba de aceptación	
Código: HU3_CPA33	Historia de usuario: 10
Nombre: Eliminar dictamen.	
Descripción: Prueba para la funcionalidad que permite eliminar el dictamen en el sistema.	
<p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado con el rol de Integrante.</p> <p>Debe haberse creado al menos un dictamen.</p>	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminar el dictamen en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Integrante, se le brindará la posibilidad de eliminar el dictamen anteriormente creado en el sistema.</p>	
Resultado esperado: El dictamen se eliminó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Anexo 7 Casos de prueba de aceptación para la tercera entrega.

Caso de prueba de aceptación	
Código: HU8_CPA35	Historia de usuario: 8
Nombre: Visualizar comité científico.	
Descripción: Prueba para la funcionalidad que permite visualizar el comité científico en el sistema.	
Condiciones de ejecución: Debe haberse creado al menos un comité científico.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita visualizar el comité científico en el sistema. Pasos de ejecución: Luego de que el usuario acceda al sistema, se le brindará la posibilidad de visualizar el comité científico anteriormente creado en el sistema.	
Resultado esperado: El comité científico se visualizó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU8_CPA36	Historia de usuario: 8
Nombre: Modificar comité científico.	
Descripción: Prueba para la funcionalidad que permite modificar el comité científico en el sistema.	
Condiciones de ejecución:	

<p>El usuario debe estar autenticado con el rol de Organizador en el sistema.</p> <p>Debe haberse creado al menos un comité científico.</p>
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificar el comité científico en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad de modificar el comité científico anteriormente creado en el sistema.</p>
<p>Resultado esperado: El comité científico se modificó correctamente.</p>
<p>Evaluación de la prueba: Satisfactoria.</p>

Caso de prueba de aceptación	
Código: HU8_GPA37	Historia de usuario: 8
Nombre: Eliminar comité científico.	
Descripción: Prueba para la funcionalidad que permite eliminar comité científico en el sistema.	
<p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado con el rol de Organizador en el sistema.</p> <p>Debe haberse creado al menos un comité científico.</p>	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminar el comité científico en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad de eliminar el comité científico anteriormente creado en el sistema.</p>	

Resultado esperado: El comité científico se eliminó correctamente.
Evaluación de la prueba: Satisfactoria.

Caso de prueba de aceptación	
Código: HU9_CPA38	Historia de usuario: 9
Nombre: Crear comité organizador.	
Descripción: Prueba para la funcionalidad que permite crear el comité organizador en el sistema.	
Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador en el sistema.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita crear el comité organizador en el sistema. Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad de crear el comité organizador en el sistema.	
Resultado esperado: El comité organizador se creó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU9_CPA39	Historia de usuario: 9
Nombre: Visualizar comité organizador.	
Descripción: Prueba para la funcionalidad que permite visualizar el comité organizador en el sistema.	
Condiciones de ejecución:	

<p>El usuario debe estar autenticado con el rol de Organizador en el sistema.</p> <p>Debe haber al menos un comité organizador creado.</p>
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita visualizar comité organizador en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema, se le brindará la posibilidad de visualizar el comité organizador en el sistema.</p>
<p>Resultado esperado: El comité organizador se visualizó correctamente.</p>
<p>Evaluación de la prueba: Satisfactoria.</p>

Caso de prueba de aceptación	
Código: HU9_CPA40	Historia de usuario: 9
Nombre: Modificar comité organizador.	
Descripción: Prueba para la funcionalidad que permite modificar el comité organizador en el sistema.	
<p>Condiciones de ejecución:</p> <p>El usuario debe estar autenticado con el rol de Organizador en el sistema.</p> <p>Debe haber al menos un comité organizador creado.</p>	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificar el comité organizador en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol Organizador, se le brindará la posibilidad de modificar el comité organizador en el sistema.</p>	
Resultado esperado: El comité organizador se modificó correctamente.	

Evaluación de la prueba: Satisfactoria.
--

Caso de prueba de aceptación	
------------------------------	--

Código: HU9_CPA41	Historia de usuario: 9
--------------------------	-------------------------------

Nombre: Eliminar comité organizador.

Descripción: Prueba para la funcionalidad que permite eliminar el comité organizador en el sistema.
--

Condiciones de ejecución: El usuario debe estar autenticado con el rol de Organizador en el sistema. Debe haber al menos un comité organizador creado.

Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminar el comité organizador en el sistema. Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol Organizador, se le brindará la posibilidad de eliminar el comité organizador en el sistema.

Resultado esperado: El comité organizador se eliminó correctamente.
--

Evaluación de la prueba: Satisfactoria.
--

Caso de prueba de aceptación	
------------------------------	--

Código: HU9_CPA42	Historia de usuario: 13
--------------------------	--------------------------------

Nombre: Crear documentación del FCT.

Descripción: Prueba para la funcionalidad que permite crear la documentación del FCT en el sistema.
--

Condiciones de ejecución:

El usuario debe estar autenticado con el rol de Organizador en el sistema.
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita crear el la documentación del FCT en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol Organizador, se le brindará la posibilidad de crear la documentación del FCT en el sistema.</p>
Resultado esperado: La documentación del FCT se creó correctamente.
Evaluación de la prueba: Satisfactoria.

Caso de prueba de aceptación	
Código: HU9_CPA43	Historia de usuario: 13
Nombre: Visualizar documentación del FCT.	
Descripción: Prueba para la funcionalidad que permite visualizar la documentación del FCT en el sistema.	
Condiciones de ejecución: Debe haber al menos un documento del FCT creado.	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita visualizar la documentación del FCT en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema, se le brindará la posibilidad de visualizar la documentación del FCT en el sistema.</p>	
Resultado esperado: La documentación del FCT se visualizó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU9_CPA44	Historia de usuario: 13
Nombre: Modificar documentación del FCT.	
Descripción: Prueba para la funcionalidad que permite modificar la documentación del FCT en el sistema.	
Condiciones de ejecución: El usuario de be estar autenticado en el sistema con el rol de Organizador. Debe haber al menos un documento del FCT creado.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificarla documentación del FCT en el sistema. Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad de modificar la documentación del FCT en el sistema.	
Resultado esperado: La documentación del FCT se modificó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU9_CPA45	Historia de usuario: 13
Nombre: Eliminar documentación del FCT.	
Descripción: Prueba para la funcionalidad que permite eliminar la documentación del FCT en el sistema.	
Condiciones de ejecución: El usuario de be estar autenticado en el sistema con el rol de Organizador.	

Debe haber al menos un documento del FCT creado.
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminarla documentación del FCT en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad de eliminar la documentación del FCT en el sistema.</p>
Resultado esperado: La documentación del FCT se eliminó correctamente.
Evaluación de la prueba: Satisfactoria.

Caso de prueba de aceptación	
Código: HU9_CPA46	Historia de usuario: 7
Nombre: Crear programa.	
Descripción: Prueba para la funcionalidad que permite crear el programa en el sistema.	
Condiciones de ejecución:	
El usuario debe estar autenticado en el sistema con el rol de Organizador.	
Pasos de ejecución/Entrada:	
Entrada: Son los pasos que ejecuta el usuario para que el sistema permita crear el programa en el sistema.	
Pasos de ejecución: Luego de que el usuario acceda al sistema con el rol de Organizador, se le brindará la posibilidad de crear el programa en el sistema.	
Resultado esperado: El programa se creó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU9_CPA47	Historia de usuario: 7
Nombre: Visualizar programa.	
Descripción: Prueba para la funcionalidad que permite visualizar el programa en el sistema.	
Condiciones de ejecución: Debe haber al menos un programa creado.	
Pasos de ejecución/Entrada: Entrada: Son los pasos que ejecuta el usuario para que el sistema permita visualizar el programa en el sistema. Pasos de ejecución: Luego de que el usuario acceda al sistema, se le brindará la posibilidad de visualizar el programa en el sistema.	
Resultado esperado: El programa se visualizó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Caso de prueba de aceptación	
Código: HU9_CPA48	Historia de usuario: 7
Nombre: Modificar programa.	
Descripción: Prueba para la funcionalidad que permite modificar el programa en el sistema.	
Condiciones de ejecución: Debe haber al menos un programa creado. El usuario debe estar autenticado con el rol Organizador.	

<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita modificar el programa en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema, se le brindará la posibilidad de modificar el programa en el sistema.</p>
<p>Resultado esperado: El programa modificó correctamente.</p>
<p>Evaluación de la prueba: Satisfactoria.</p>

Caso de prueba de aceptación	
Código: HU9_CPA49	Historia de usuario: 7
Nombre: Eliminar programa.	
Descripción: Prueba para la funcionalidad que permite eliminar el programa en el sistema.	
<p>Condiciones de ejecución:</p> <p>Debe haber al menos un programa creado.</p> <p>El usuario debe estar autenticado con el rol Organizador.</p>	
<p>Pasos de ejecución/Entrada:</p> <p>Entrada: Son los pasos que ejecuta el usuario para que el sistema permita eliminar el programa en el sistema.</p> <p>Pasos de ejecución: Luego de que el usuario acceda al sistema, se le brindará la posibilidad de eliminar el programa en el sistema.</p>	
Resultado esperado: El programa eliminó correctamente.	
Evaluación de la prueba: Satisfactoria.	

Anexo 8 Entrevista utilizada

Principales preguntas planteadas durante la entrevista con el cliente.

¿Cómo funciona actualmente la gestión del Fórum de Ciencia y Técnica de la Facultad 4?

¿Cómo se realiza el proceso de creación y publicación de la convocatoria?

¿Cómo se realiza el proceso de inscripción en el evento y la recepción de los trabajos?

¿Cómo se realiza el proceso de creación de las comisiones y asignación de trabajos a estas?

¿El banco de problemas se encuentra disponible para su consulta?

¿Cómo se realiza el proceso de creación del banco de problemas?

¿Cómo se realiza la creación del dictamen?

¿Cuántas personas intervienen durante todo este proceso?