



Universidad de las Ciencias
Informáticas



SISTEMA DE GESTIÓN DE INFORMACIÓN PARA EL SINDICATO DE TRABAJADORES DE LA UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

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



AUTOR:

- SAMUEL RODRÍGUEZ CARRETERO

TUTORA:

- ING. YNET PÉREZ TERÁN PÉREZ

LA HABANA, 16 DE MARZO DE 2015
"AÑO 57 DE LA REVOLUCIÓN"



"El éxito no es definitivo, el fracaso no es fatídico: lo que cuenta es el valor para continuar". - Winston Churchill

Winston S. Churchill

Declaración de autoría

Declaro por este medio que yo Samuel Rodríguez Carretero, soy el autor principal de la tesis de pregrado “SISTEMA DE GESTIÓN DE INFORMACIÓN PARA EL SINDICATO DE TRABAJADORES DE LA UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS”, desarrollada como parte del curso para trabajadores de la facultad 6 y autorizo a la Universidad de las Ciencias Informáticas a hacer uso de la misma en su beneficio, así como los derechos patrimoniales con carácter exclusivo.

Y para que así conste, firmo la presente declaración jurada de autoría en La Habana a los _____ días del mes de _____ del año _____.

Samuel Rodríguez Carretero

Ing. Yinet Pérez-Terán Pérez
Tutora

Dedicatoria

A mi pequeña familia, mi madre y mi hermano, por apoyarme con tanto amor en esta emprendedora carrera.

A todos mis amigos, por su ayuda incondicional.

RESUMEN

En la Universidad de las Ciencias Informáticas sus trabajadores son afiliados al Sindicato Nacional de Trabajadores de la Educación, la Cultura y el Deporte. Para ejercer la labor sindical en la universidad, se dispone de una estructura organizativa compuesta por el Buró Sindical Universitario, al cual se subordinan noventa y seis secciones sindicales que agrupan a más de cinco mil miembros. Debido a la cifra tan elevada de afiliados, se deben controlar grandes volúmenes de información en el sindicato, por lo que surgen deficiencias en la gestión de los procesos claves de la organización tales como la divulgación de la información, entregas de actas de reunión, atención y respuestas a inquietudes, entre otras. La presente investigación tiene como objetivo el desarrollo de un sistema de gestión de información en la universidad, para contribuir con los procesos que se llevan a cabo en la organización sindical, apoyado en la implementación de un portal Web. Todo el proceso de desarrollo estuvo guiado por las especificaciones que propone la metodología ágil para el desarrollo de software *Extreme Programming (XP)*. Se realizó además un análisis de las herramientas y tecnologías que se utilizaron para el desarrollo de la solución y se muestran los resultados de las pruebas efectuadas para garantizar su aceptación por parte de los clientes. Con esta herramienta los afiliados de las secciones sindicales y los miembros del Buró Sindical, podrán acceder y gestionar la información derivada de los principales procesos que se desarrollan en la organización sindical de la universidad.

Palabras Clave: Sistema de gestión de información, Procesos de organizaciones sindicales.

Introducción	1
Capítulo 1: Fundamentación Teórica	4
1.1. Introducción.....	4
1.2. Conceptos asociados al dominio del problema.....	4
1.3. Objeto de estudio	5
1.3.1. Sistemas de gestión	5
1.3.2. Aplicación web	7
1.4. Análisis de soluciones existentes.....	7
1.4.1. Soluciones existentes a nivel internacional	7
1.4.2. Soluciones existentes a nivel nacional.....	8
1.4.3. Portales web y sistemas de gestión del sindicato realizados en la UCI ...	8
1.4.4. Análisis de los antecedentes	10
1.5. Metodología de desarrollo de software	11
1.6. Herramientas y tecnologías a utilizar	12
1.6.1. Lenguaje Unificado de Modelado (UML).....	13
1.6.2. Herramienta CASE (Computer Aided Software Engineering)	13
1.6.3. Lenguajes de programación de código abierto.....	14
1.6.4. Sistema Gestor de Contenido (CMS)	15
1.6.5. CMS Drupal	16
1.6.6. Entorno de Desarrollo Integrado(IDE).....	17
1.6.7. Servidor de aplicaciones	18
1.6.8. Sistema gestor de base de datos (SGBD).....	18
1.7. Conclusiones parciales.....	19
Capítulo 2. Análisis y diseño del sistema.....	20
2.1. Introducción.....	20
2.2. Flujo actual del proceso.....	20
2.3. Características del sistema.....	22
2.4. Fase de exploración.....	24

2.4.1.	Historias de usuario	24
2.4.2.	Personal relacionado con el sistema	27
2.5.	Fase de Planificación	29
2.5.1.	Prioridad de las Historias de Usuario	30
2.5.2.	Estimación del esfuerzo de las HU	30
2.5.3.	Cronograma de liberación	31
2.6.	Fase de Iteración	32
2.6.1.	Características no funcionales	33
2.6.2.	Utilización de Patrones	35
2.6.3.	Tarjetas CRC	39
2.6.4.	Diseño de la Base de Datos	41
2.7.	Conclusiones parciales	42
Capítulo 3.	Implementación y Validación del Sistema	44
3.1.	Introducción	44
3.2.	Implementación del sistema	44
3.3.	Estándares de codificación	47
3.3.1.	Nomenclatura de las clases	48
3.3.2.	Nomenclatura de variables	48
3.4.	Pruebas	49
3.4.1.	Pruebas unitarias	49
3.4.2.	Pruebas de aceptación	49
3.4.3.	Registro de no conformidades	52
3.5.	Conclusiones parciales	53
Conclusiones Generales	54
Recomendaciones	55
Referencias Bibliográficas	56
Anexo 1	59
Anexo 2	62
Anexo 3	63

Anexo 4.....	71
--------------	----

Índice de Figuras

Figura 1: Proceso de divulgación de la Información.....	22
Figura 2: Diagrama de clases patrón Experto.	36
Figura 3: Código donde se implementa el patrón Experto.	36
Figura 4: Diagrama de clases patrón Creador.....	37
Figura 5: Código que implementa el patrón Creador.....	38
Figura 6: Modelo de datos.....	42
Figura 7: Ubicación de los módulos en la estructuran de archivos de Drupal.	47
Figura 8. Nomenclatura de las clases.....	48
Figura 9. Nomenclatura de variables y funcionalidades.	48
Figura 10: Diagrama de clases persistentes.....	71

Índice de tablas

Tabla 1: Análisis de los antecedentes.....	10
Tabla 3: HU Autenticar usuario	26
Tabla 4: HU Gestionar afiliado	27
Tabla 5: HU Gestionar afiliados de una sección sindical.	27
Tabla 6: HU Gestionar acta.....	27
Tabla 2: Personal relacionado con el sistema.	28
Tabla 7: Prioridad de las HU.	30
Tabla 8: Estimación del esfuerzo para las HU.....	31
Tabla 9: Cronograma de liberación.	32
Tabla 10: Primera iteración.	33
Tabla 11: Segunda iteración.	33
Tabla 12: Tercera iteración.	33
Tabla 13: Tarjeta CRC: Clase Base.....	40
Tabla 14: Tarjeta CRC: Clase Actas.....	40
Tabla 15: Tarjeta CRC: Clase Ausente.....	40
Tabla 16: Tarjeta CRC: Clase Asistencia.....	41
Tabla 17: Tarjeta CRC: Clase Invitado.	41
Tabla 18: Tarjeta CRC: Clase Acuerdo.....	41
Tabla 19: Tarjeta CRC: Clase Intervención.	41
Tabla 20: Tarjeta CRC: Clase Tema.....	41
Tabla 21: Tarjeta CRC: Clase Inquietud.	41
Tabla 22: Tarea: Establecer arquitectura base.....	44
Tabla 23: Tarea: Diseñar la Base de datos.	45
Tabla 24: Tarea: Establecer Prototipo de Diseño.	45
Tabla 25: Tarea: Diseñar vista de autenticar.	45
Tabla 26: Tarea: Mapear Base de Datos.....	46
Tabla 27: Tarea: Conectar autenticación con Base de Datos.....	46
Tabla 28: Prueba de aceptación H.U1_P.A1.....	49
Tabla 29: Prueba de aceptación H.U4_P.A1.....	50
Tabla 30: Prueba de aceptación H.U6_P.A1.....	51
Tabla 31: Prueba de aceptación H.U7_P.A1.....	51

Índice de tablas

Tabla 32: Registro de no conformidades.	52
Tabla 33: HU 2: Gestionar Áreas de las SS.	59
Tabla 34: HU 3: Gestionar Secciones Sindicales.	59
Tabla 35: HU 5: Asignar roles a la cuenta de un Afiliado.	59
Tabla 36: HU 8: Gestionar las inquietudes de una sección sindical.	59
Tabla 37: HU 9: Gestionar las respuestas a inquietudes de una sección sindical.	59
Tabla 38: HU 10: Especificar la plantilla de la estructura organizativa de las secciones sindicales.	60
Tabla 39: HU 11: Modificar la plantilla de la estructura organizativa de las secciones sindicales.	60
Tabla 40: HU 12: Mostrar la estructura organizativa de una sección sindical.	60
Tabla 41: HU 13: Mostrar la estructura organizativa del buró sindical.	60
Tabla 42: HU 14: Mostrar reportes de las secciones sindicales.	61
Tabla 43: HU 15: Mostrar reportes de los afiliados de las secciones sindicales.	61
Tabla 44: HU 16: Mostrar reportes de las actas de reuniones de las secciones sindicales.	61
Tabla 45: HU 17: Mostrar reportes de las inquietudes y respuestas de las secciones sindicales.	61
Tabla 46: Tarjeta CRC: Clase Autenticar.	62
Tabla 47: Tarjeta CRC: Clase Afiliado.	62
Tabla 48: Tarjeta CRC: Clase Área.	62
Tabla 49: Tarjeta CRC: Clase SecciónSindical.	62
Tabla 50: Tarjeta CRC: Clase Estructura.	62
Tabla 51: Tarjeta CRC: Clase Inquietud.	63
Tabla 52: Tarjeta CRC: Clase Inquietud.	63
Tabla 53: Tarea: Diseñar vista de gestionar áreas de SS.	63
Tabla 54: Tarea: Conectar vista de gestionar áreas de SS con la BD.	63
Tabla 55: Tarea: Diseñar vista de gestionar SS.	64
Tabla 56: Tarea: Conectar vista de gestionar SS con la BD.	64
Tabla 57: Tarea: Diseñar vista de gestionar la cuenta de un afiliado.	64
Tabla 58: Tarea: Conectar la vista de gestionar la cuenta de un afiliado con la BD.	64
Tabla 59: Tarea: Diseñar vista de gestionar actas de reuniones de una SS.	65
Tabla 60: Tarea: Conectar la vista de gestionar actas de reuniones de una SS con la BD.	65
Tabla 61: Tarea: Diseñar vista de gestionar afiliados de una SS.	65
Tabla 62: Tarea: Conectar la vista de gestionar afiliados de una SS con la BD.	65

Índice de tablas

Tabla 63: Tarea: Diseñar vista de gestionar las inquietudes de una SS.	66
Tabla 64: Tarea: Conectar la vista de gestionar las inquietudes de una SS con la BD.	66
Tabla 65: Tarea: Diseñar vista de gestionar las respuestas a las inquietudes de una SS.....	66
Tabla 66: Tarea: Conectar vista de gestionar las respuestas a las inquietudes de una SS con la BD.	67
Tabla 67 Tarea: Diseñar vista de gestionar plantilla de estructura organizativa de las SS.....	67
Tabla 68 Tarea: Conectar vista de la plantilla de la estructura organizativa de las SS con la BD.....	67
Tabla 69: Tarea: Diseñar la vista de la estructura organizativa del buró sindical.	67
Tabla 70: Tarea: Conectar la vista de la estructura organizativa del buró sindical con la BD.....	68
Tabla 71: Tarea: Diseñar la vista de los reportes de las SS.	68
Tabla 72: Tarea: Conectar la vista para mostrar los reportes de las SS con la BD.	68
Tabla 73: Tarea: Diseñar la vista de los reportes de los afiliados de las SS.	68
Tabla 74: Tarea: Conectar la vista para mostrar los reportes de los afiliados de las SS con la BD.	69
Tabla 75: Tarea: Diseñar la vista de los reportes de las actas de reuniones de las SS.....	69
Tabla 76: Conectar la vista para mostrar los reportes de las actas de reuniones de las SS con la BD.	69
Tabla 77: Tarea: Diseñar la vista de los reportes de las inquietudes y respuestas de las SS.....	69
Tabla 78: Conectar la vista para mostrar los reportes de las inquietudes y respuestas de las SS con la BD.	70

Introducción

Durante muchos años las organizaciones sindicales a nivel mundial han sido sinónimo de lucha por defender los derechos y el bienestar de las masas trabajadoras. Ejemplo de esto es la Central de Trabajadores de Cuba (CTC), organización representativa de los trabajadores cubanos. Esta organización está integrada por varios sindicatos, que representan a los organismos y centros laborales del país, entre los cuales se encuentra el Sindicato Nacional de Trabajadores de la Educación, la Ciencia y el Deporte (SNTECD), que agrupa a los afiliados comprendidos en los sectores de la educación y el deporte, al cual pertenece la Universidad de las Ciencias Informáticas (UCI), como centro de educación superior.

Para ejercer la labor sindical en la universidad, se dispone de una estructura organizativa compuesta por el Buró Sindical Universitario (BUS) como máxima representación, al cual se subordinan 96 secciones sindicales y agrupa a más de 4500 trabajadores en toda la UCI. Durante la ejecución de la labor sindical se genera una gran cantidad de información en fichas técnicas, expedientes y actas de reuniones por cada sección sindical, que son revisadas y cotejadas por los miembros del BUS. Esto trae consigo que revisar al menos la mitad de la documentación que incluye las fichas, actas y expedientes de disímiles secciones tomaría al menos una semana, perdiendo la inmediatez de analizar los datos en los se plasman documentos como asistencia de los afiliados. Al no contar con estos y otros datos se dificulta la toma de decisiones en el buró y no se tiene conocimiento de las inquietudes que se generaron en las asambleas y por tanto la respuesta a ellas se ve demorada.

Además la gestión de los procesos claves de la organización, tales como: la divulgación de información, entrega de actas de reunión, atención y respuestas a inquietudes, entre otros, se realizan por vía correo electrónico y utilizando en todos los casos las aplicaciones del paquete de office (Word, Excel y PowerPoint). Por tanto, ante la situación anteriormente descrita y la disposición de mejorar cada día el funcionamiento sindical, la dirección del BUS identificó la necesidad de informatizar parte de los procesos para mitigar las actuales deficiencias generadas por los mecanismos tradicionales de los cuales se apoya para realizar su trabajo diario.

A raíz de la situación problemática anteriormente expuesta, se plantea como **problema de la investigación**: ¿Cómo contribuir con la gestión de la información correspondiente a los procesos que se llevan a cabo en el sindicato de la UCI?

En correspondencia con el problema planteado, el **objeto de estudio** lo constituye: Procesos de gestión de la información en organizaciones sindicales, delimitando como **campo de acción**: Procesos de gestión de la información en la organización sindical de la UCI.

Introducción

Para dar solución a dicho problema se plantea como **objetivo general**: Desarrollar un sistema de gestión de información para el sindicato de trabajadores de la UCI.

Del objetivo general formulado se derivan los siguientes **objetivos específicos**:

1. Elaborar la fundamentación teórica de la investigación para obtener las bases de una propuesta de solución.
2. Realizar el análisis y diseño de la propuesta de solución.
3. Implementar la solución informática.
4. Validar la solución mediante la realización de pruebas al sistema.

Para dar cumplimiento a los objetivos expuestos se definen las siguientes **tareas de investigación**:

1. Definición de los principales conceptos asociados al objeto de la investigación así como a su campo de acción.
2. Análisis de la información existente relacionada servicios web para la gestión de procesos.
3. Selección de las principales herramientas que se ajustan a las necesidades de la investigación para el desarrollo del sistema propuesto.
4. Identificación de las principales funcionalidades del sistema.
5. Realización del diseño del sistema propuesto.
6. Creación de los artefactos para la fase implementación.
7. Implementación de las funcionalidades identificadas
8. Validación de los resultados obtenidos a través de pruebas unitarias.
9. Validación de los resultados obtenidos a través de pruebas de aceptación.

Observando la situación actual del sindicato de la UCI, se define como **idea a defender**:

El desarrollo de una herramienta para la organización sindical de la UCI, contribuirá a mejorar la gestión de la información correspondiente a los procesos que se llevan a cabo en el sindicato.

Desde el punto de vista investigativo se emplearon los siguientes **métodos científicos**:

Métodos Teóricos:

1. **Analítico-sintético**: Se utilizó este método en el estudio de diversas bibliografías y documentos, asociadas al objeto de estudio, de las cuales se tomaron los elementos más significativos que ayudaron a obtener conclusiones que sustentaron la necesidad de la investigación.
2. **Histórico-lógico**: se utilizó para realizar un estudio de la existencia de sistemas similares empleados por otras instituciones del país y la universidad, además de las tendencias más recientes del desarrollo de portales web y basándose en estos datos, complementar las características necesarias para la solución propuesta.

Métodos Empíricos:

1. **Entrevista:** se realizaron entrevistas a los miembros del buró sindical y secciones sindicales de diferentes áreas, para constatar las necesidades de nuevas funcionalidades y mejoras de los procesos existentes.

La investigación ha sido estructurada de la siguiente manera:

Resumen, introducción, tres capítulos, conclusiones, recomendaciones, bibliografía y anexos. En los capítulos se describe todo lo relacionado con el trabajo investigativo, organizado de la siguiente manera:

Capítulo 1. Fundamentación Teórica.

Se detallan los conceptos asociados al dominio del problema, se realiza una descripción del objeto de estudio y del problema de la investigación para una mejor comprensión de los mismos, así como el análisis de los diferentes sistemas existentes en el ámbito internacional, nacional y en la UCI, además de las tendencias en cuanto a metodología de desarrollo, herramientas y tecnologías utilizadas, así como los lenguajes de modelado y programación utilizados durante el desarrollo del sistema.

Capítulo 2. Características del sistema. Análisis y Diseño.

En este capítulo se realiza una descripción del negocio donde se describen los procesos a informatizar. Además se obtienen y se describen los requisitos funcionales y no funcionales del sistema, así como la identificación y descripción de los casos de uso y la estructuración del Modelo de Casos de Uso del Sistema. Se describen los patrones arquitectónicos y de diseño empleados en el desarrollo de la aplicación.

Capítulo 3. Implementación y validación del sistema.

Se describe el proceso de implementación a través de artefactos. Por último, se realizan los casos de prueba al sistema implementado.

Capítulo 1. Fundamentación Teórica

Capítulo 1: Fundamentación Teórica

1.1. Introducción

En el presente capítulo se hace necesario conocer los conceptos relacionados al dominio del problema, la descripción del objeto de estudio, así como el análisis de los sistemas de gestión y los portales web similares al desarrollado. Además se hace referencia a las tecnologías y herramientas, que son objeto de análisis y estudio con el fin de fundamentar su uso en la solución propuesta.

1.2. Conceptos asociados al dominio del problema

Para un mejor entendimiento de lo que representa la gestión de información y sus funcionalidades, se describen a continuación los siguientes conceptos directamente relacionados con el objeto de estudio.

Gestión

La gestión consiste en un conjunto de trámites que se llevan a cabo para resolver un asunto. Dirección, administración de una empresa, negocio, entre otros (1).

Además la gestión se considera como un proceso en el cual pueden ser reconocidos ciertas etapas. La primera de ellas es la planificación, en esta etapa se fijarán los objetivos a corto y largo plazo y el modo en que serán alcanzados. Es a partir de esta organización donde se determinarán el resto de las etapas. Luego puede ser mencionada la organización, en este momento los gestores determinan detalladamente el procedimiento para alcanzar los objetivos formulados anteriormente.

Por último debe ser mencionado el control, en este caso el o los gestores examinan si la planificación es respetada y los objetivos son cumplidos. Para ello deben ser capaces de realizar ciertas correcciones si las normas no son acatadas (2).

Información

Según la Real Academia Española, la información se define como: “Comunicación o adquisición de conocimientos que permiten ampliar o precisar los que se poseen sobre una materia determinada” (3).

Es el conocimiento adquirido por causa del procesamiento de datos. En contradicción a esta diferencia se puede afirmar que en cadenas de servicios, lo que es dato para un cliente es información para el proveedor-procesador.

Capítulo 1. Fundamentación Teórica

Gestión de información

Comprende las actividades relacionadas con la obtención de la información adecuada, a un precio adecuado, en el tiempo y lugar adecuado, para tomar la decisión adecuada” (5).

La gestión de la información es un proceso que incluye operaciones como extracción, manipulación, tratamiento, depuración, conservación, acceso y/o colaboración de la información adquirida por una organización a través de diferentes fuentes y que gestiona el acceso y los derechos de los usuarios sobre la misma” (6).

La gestión de la información se establece como una disciplina transversal que aparece entrelazada en todas las diferentes capas de una organización, en todos los conceptos de administración como recursos humanos, marketing, finanzas, estrategia, operaciones, entre otros (6).

Los sistemas de gestión de información permiten la comprensión de cada organización desde un enfoque analítico, evaluador y creativo que permite develar las oportunidades que merezcan ser explotadas y contrarrestar las amenazas, además de establecer los factores que resulten críticos y buscar nuevas oportunidades (7).

1.3. Objeto de estudio

A continuación se profundizará en el objeto de estudio y también se abordarán los aspectos fundamentales de las aplicaciones web.

1.3.1. Sistemas de gestión

Un sistema de gestión es un conjunto de etapas unidas en un proceso continuo, que permite trabajar ordenadamente una idea hasta lograr mejoras y su continuidad. Se establecen cuatro etapas en este proceso, que hacen de este sistema un proceso circular, pues en la medida que el ciclo se repita recurrente y recursivamente, se logrará en cada ciclo obtener una mejora (8).

Las cuatro etapas de un sistema de gestión son las siguientes:

Etapas de Ideación: el objetivo de esta etapa es trabajar en la idea que guiará los primeros pasos del proceso de creación que se logra con el sistema de gestión propuesto. Existen varias metodologías para lograr refinar la idea. Sin embargo, se recomienda una práctica muy conocida como la tormenta de ideas o brainstorming

Etapas de Planeación: se definen las estrategias que se utilizarán, la estructura organizacional, el personal que se necesita, la tecnología a utilizar, los recursos necesarios y la clase de controles que se aplican en

Capítulo 1. Fundamentación Teórica

todo el proceso. El proceso de planificación depende de las características particulares de cada organización, sin embargo dentro de cualquier proceso formal de planificación, existen tres perspectivas básicas comunes: la estrategia corporativa, de negocios y funcional. A nivel de negocios y funcional, los resultados de esta etapa se enmarcan en propuestas de programas estratégicos de acción y programación de presupuestos. Estas propuestas son finalmente evaluadas y consolidadas a nivel corporativo (8).

Etapa de Implementación: en su significado más general, se entiende por gestión, la acción y efecto de administrar. Pero en un contexto empresarial, esto se refiere a la dirección que toman las decisiones y las acciones para alcanzar los objetivos trazados. Es importante destacar que las decisiones y acciones que se toman para llevar adelante un propósito, se sustentan en los mecanismos o instrumentos administrativos (estrategias, tácticas, procedimientos, presupuestos, entre otros.)

Etapa de Control: función administrativa esencialmente reguladora, que permite verificar si el elemento seleccionado (actividad, proceso, unidad, sistema, etc.), está cumpliendo sus objetivos o alcanzando los resultados que se esperan (8). En un sistema de gestión esta etapa ayuda a lograr los objetivos de una organización mediante una serie de estrategias que incluyen la optimización de procesos, el enfoque centrado en la gestión de la información, entre otros (9).

Las mejores empresas funcionan como unidades completas con una visión compartida. Ello engloba la información compartida, evaluaciones comparativas, trabajo en equipo y un funcionamiento acorde con los más rigurosos principios de calidad y del medioambiente (10).

La implementación de un sistema de gestión tiene las siguientes ventajas:

- Control efectivo de las actividades de la organización.
- Integración de nuevas tecnologías y herramientas.
- Disponibilidad de mayor y mejor información para los usuarios en tiempo real.
- Elimina la barrera de la distancia trabajando con un mismo sistema en puntos distantes.
- Disminuye errores, tiempo y recursos.

Entre las desventajas que presentan estos sistemas se encuentran:

- El tiempo que pueda tomar su implementación.
- La resistencia al cambio de los usuarios.
- Los problemas técnicos como fallas de hardware y/o software.

Capítulo 1. Fundamentación Teórica

- La inadecuada implementación de las funcionalidades para el apoyo de actividades de la organización.

1.3.2. Aplicación web

Consiste en un documento electrónico que contiene información, cuyo formato se adapta para estar insertado en la World Wide Web, de manera que los usuarios puedan entrar a la misma por medio del uso de un navegador, visualizándola con un dispositivo móvil como un smartphone o el monitor de computadora (11). Es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador (12).

Una aplicación web es más que un conjunto de páginas entrelazadas entre sí, supone la existencia de código ejecutable. Las aplicaciones web están formadas por diferentes módulos, independientes entre sí, que se pueden ejecutar en el servidor o en el cliente y pueden estar implementados utilizando lenguajes o tecnologías diferentes. Para organizar la información se suelen utilizar bases de datos (13).

1.4. Análisis de soluciones existentes

La necesidad de hacer más factible las vías de comunicaciones entre las diferentes organizaciones sindicales a nivel mundial, han dado origen al desarrollo de diversos portales web que brindan la posibilidad de realizar una gran cantidad de procesos y acciones que se gestionan en el sindicato. Por lo que se hace necesario hacer un análisis de las soluciones homólogas existentes no solo a nivel nacional, sino también internacional.

1.4.1. Soluciones existentes a nivel internacional

UGT (Unión General de Trabajadores de España)

Portal web diseñado para fomentar la actividad sindical en España, en el cual se pueden observar varias funcionalidades que facilitan a los usuarios el intercambio de opiniones mediante una sala de prensa, el acceso a revistas y videos relacionados con la actividad sindical. Se puede conocer la historia de la organización y estar informados de las noticias nacionales e internacionales en el ámbito sindical (14).

CUT (Central Unitaria de Trabajadores de Chile)

Ofrece variados servicios entre los que se pueden encontrar: el correo electrónico, servicios de búsqueda interna para permitir al usuario acceder a la información almacenada, enlaces a otras páginas y portales relacionados con organizaciones de trabajadores tanto en el país como en el mundo (15).

Capítulo 1. Fundamentación Teórica

1.4.2. Soluciones existentes a nivel nacional

El desarrollo de disímiles portales web sindicales, constituye uno de los factores prominentes para fomentar los lazos de comunicación entre los sindicatos, tanto en el ámbito internacional como nacional.

Cuba Sindical

Ofrece una amplia información relacionada con el funcionamiento a nivel nacional de la CTC, como son: los miembros del secretariado nacional, sindicatos nacionales, congresos, CTC provinciales, derechos laborales, noticias del acontecer sindical en el país y otros servicios. Cuenta además con un boletín informativo, enlaces a otras páginas y un blog que permite opinar sobre diversos temas de la actualidad (16).

SNTECD (Sindicato Nacional de trabajadores de la educación, el deporte y la salud)

Es el sindicato que agrupa a los trabajadores cubanos de la educación y el deporte que por su propia voluntad desean ingresar en esta organización. El portal cuenta con una gran cantidad de documentos relacionados con su historia, objetivos, metas a cumplir, eventos que se realizan, sedes provinciales y municipales. Se permite conocer la estructura sindical del Secretariado Nacional del SNTECD, Comités Provinciales y Municipales, así como noticias nacionales e internacionales y enlaces a otros sitios.

Ofrece una revista que lleva por nombre "Con Luz Propia", la cual contiene anuncios de instituciones afines a la educación y al deporte. Posee una sección denominada Pedagogos, donde se divulgan elementos de la vida y obra de educadores cubanos distinguidos (17).

1.4.3. Portales web y sistemas de gestión del sindicato realizados en la UCI

SigestCTC (Sistema de Gestión de la Central de Trabajadores de Cuba):

Es una aplicación para gestionar los procesos del Sindicato Nacional de Trabajadores de la Administración Pública (SNTAP). Este sindicato tiene representación en cada una de las provincias y municipios del país, lo que en su conjunto conforma la estructura organizativa.

Los procesos que se realizan en esta aplicación están vinculados al modeloA1, donde se controla la cantidad de organizaciones de masas y el total de dirigentes, delegados y secciones sindicales, así como la gestión de los datos de las provincias y los municipios. Para el desarrollo de la misma se utilizó la metodología RUP, como lenguaje de programación PHP y como gestor de base de datos MySQL (18).

Sistema de Gestión para la sección sindical Vicerrectoría Primera en la UCI

Capítulo 1. Fundamentación Teórica

Se analizan los procesos de gestión de información del sindicato en la Vicerrectoría de la UCI. Entre estos procesos se encuentra el pago de las finanzas, registro de opinión de cada afiliado, gestión de los comités sindicales y las propuestas de actividades, de los cuales no se desarrolla el proceso de divulgación de información.

Este sistema fue desarrollado a través de RUP como metodología de desarrollo, Java como lenguaje de programación y PostgreSQL como el sistema gestor de base de datos (19).

Sistema de Gestión de Información para la UJC y el sindicato de la Facultad 8

Aplicación Web encaminada a mejorar el control y gestión del sindicato de trabajadores y la UJC de la Facultad # 8. Los principales procesos que se gestionan en el sindicato están vinculados al control de la asistencia a reuniones, emulación y control de los trabajadores y la conformación de las actas de asambleas. No se realizan los procesos de divulgación de información y gestión de inquietudes, los cuales se proponen en el presente trabajo. En la construcción del sistema se utilizó la metodología RUP, PHP como lenguaje de programación y Apache como servidor de aplicaciones (20).

Portal web para la sección sindical de la Facultad 8

Se describen los procesos de divulgación de información, gestionar emulación y gestionar finanzas. En el proceso de divulgación de información se encuentran una serie de subprocesos como: mostrar historia, estructura, opiniones, entre otros.

Para la construcción y desarrollo de este sistema se seleccionó: Drupal como CMS, como lenguaje de programación PHP, PostgreSQL como servidor de base de datos y el uso del servidor de aplicaciones Apache (21).

Sistema de Gestión y Control de Información para la sección sindical de la Facultad 1

Los procesos que se gestionan en este sistema son: control de inquietudes, control del desempeño de los afiliados, gestionar modelos de actas de reunión y publicación de información. Aunque en el proceso de publicación de información no se tratan algunos objetivos como: historia, reglamentos, estructura sindical, entre otras informaciones. Se empleó como metodología de desarrollo RUP, como CMS Drupal, MySQL como sistema gestor de base de datos y como servidor de aplicaciones Apache (22).

Portal Web para la CTC de la Facultad 6

Capítulo 1. Fundamentación Teórica

Se gestionan los procesos vinculados a la gestión financiera, inquietudes, emulación, información de actividades recreativas y secciones sindicales. No se realiza el proceso de divulgación de información. Se desarrolló utilizando la metodología XP, PHP como lenguaje de programación del lado del servidor; como sistema gestor de base de datos PostgreSQL y como CMS Drupal (23).

1.4.4. Análisis de los antecedentes

Teniendo en cuenta las características y los diversos procesos que se informatizan en los sistemas internacionales y nacionales estudiados, se puede observar que su principal objetivo está encaminado a la divulgación de información. Por lo que no se realizan los demás procesos que intervienen en el funcionamiento interno de la organización, mediante el cual se controla la actividad realizada por cada uno de sus afiliados.

En el análisis de los portales web y sistemas de gestión desarrollados en la UCI para el sindicato, se pudieron identificar las diferentes características de cada uno. A continuación se muestran los principales procesos de gestión de información que en ellos se realizan.

Tabla 1: Análisis de los antecedentes.

Trabajos realizados en la UCI	Gestión de actas de reuniones	Gestión de inquietudes	Gestión de divulgación de información	Gestión de secciones sindicales
Sistema de Gestión para la Sección Sindical Vicerrectoría Primera en la UCI.	✓	✓	✗	✗
Sistema de gestión de información para la UJC y el sindicato de la Fac. 8.	✓	✗	✗	✗
Portal web para la sección sindical de la Fac. 8.	✗	✗	✓	✗
Sistema de Gestión y Control de Información para la Sección Sindical de la Fac. 1.	✓	✓	✓	✗
Portal Web para la CTC de la Fac. 6.	✗	✓	✓	✓

Partiendo de este análisis se puede concluir que ninguno de los portales web y sistemas de gestión analizados cuentan con todos los procesos que se desean desarrollar en el presente trabajo. Es

Capítulo 1. Fundamentación Teórica

necesario destacar que los mismos han sido creados para un área en específico, por lo que solucionan parcialmente las necesidades requeridas por el BUS. Dando como resultado que ninguno de ellos se puedan considerar como una solución integral al problema de la investigación.

1.5. Metodología de desarrollo de software

Las metodologías de desarrollo de software surgen ante la necesidad de utilizar una serie de procedimientos, técnicas, herramientas y soporte documental a la hora de desarrollar un producto software. Dichas metodologías pretenden guiar a los desarrolladores a crear un nuevo software, pero los requisitos de uno u otro son tan variados y cambiantes que ha dado lugar a que exista una gran variedad de metodologías para la creación del software (24).

Dichas metodologías encuentran divididas en dos grupos fundamentales:

Las metodologías robustas: dentro de las que se encuentran: *Rational Unified Process (RUP* por sus siglas en inglés), *Software Capability Maturity Model (SW –CMM* por sus siglas en inglés), *Microsoft Solution Framework (MSF* por sus siglas en inglés).

Las metodologías ágiles: dentro de las que se encuentran: *Extreme Programming (XP* por sus siglas en inglés), *SCRUM*, *Feature Driven Development (FDD* por sus siglas en inglés).

Para el desarrollo del presente trabajo se optó por emplear la metodología XP pues teniendo en cuenta las características de la solución se consideró la mejor opción a tomar en cuenta. Se basa en la realimentación continua entre el cliente y el equipo de desarrollo, por lo que existe una comunicación fluida entre todos los participantes y además aporta simplicidad en las soluciones implementadas.

XP (Extreme Programming)

La metodología *Extreme Programming (XP)* o Programación Extrema es una metodología centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. Además se basa en la realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y audacia para enfrentar los cambios. Se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico (25).

Capítulo 1. Fundamentación Teórica

La metodología se basa en:

- **Pruebas Unitarias:** se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándose en algo hacia el futuro, se podrá hacer pruebas de las fallas que pudieran ocurrir. Es como si se adelantara a obtener los posibles errores.
- **Refabricación:** se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- **Programación en pares:** una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto: mientras uno conduce, el otro consulta el mapa.

El ciclo de vida ideal consta de 6 fases:

- **Exploración:** los clientes plantean a grandes rasgos las Historias de usuario que son de interés para la primera entrega del producto.
- **Planificación de entregas:** se establece la prioridad de cada Historia de usuario y los programadores realizan una estimación del esfuerzo necesario de cada una de ellas.
- **Iteraciones:** incluye varias iteraciones sobre el sistema antes de ser entregado. El plan de entrega está compuesto por iteraciones de no más de tres semanas.
- **Producción:** requiere de pruebas adicionales y revisiones del rendimiento antes de que el sistema sea trasladado al entorno del cliente.
- **Mantenimiento:** mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones.
- **Muerte del proyecto:** un proyecto entrará en esta fase cuando el cliente no tenga más historias de usuario para ser incluidas en el sistema (25).

Por todo lo mencionado anteriormente se ha decidido utilizar la metodología ágil XP, teniendo en cuenta su sencillez en cuanto a su aprendizaje, como en su aplicación. A demás, el equipo de desarrollo está integrado por una persona y el cliente se encuentra disponible para las consultas requeridas durante todo el proceso de desarrollo.

1.6. Herramientas y tecnologías a utilizar

Capítulo 1. Fundamentación Teórica

Uno de los aspectos fundamentales que se tienen en cuenta en el desarrollo de las aplicaciones web, está determinada por la selección adecuada de las tecnologías y herramientas en su implementación. Es por eso que se debe hacer un análisis de las mismas para seleccionar las que faciliten en mayor medida el trabajo a realizar.

1.6.1. Lenguaje Unificado de Modelado (UML)

El lenguaje de modelado de objetos es un conjunto estandarizado de símbolos y de modos de disponerlos para modelar un diseño de software orientado a objetos. Algunas organizaciones los usan extensivamente en combinación con una metodología de desarrollo de software para avanzar de una especificación inicial a un plan de implementación y para comunicar dicho plan a todo un equipo de desarrolladores (26).

UML es un lenguaje muy conocido y utilizado en el modelado de sistemas de software, permite visualizar, especificar, construir y documentar el sistema a desarrollar. Ofrece un estándar para detallar un plano del sistema (modelo), el cual incluye características conceptuales como procesos de negocio, funciones del sistema, aspectos concretos como expresiones de lenguaje de programación, esquemas de bases de datos, entre otros (27).

A través de este lenguaje los desarrolladores de fases posteriores tendrán mayor dominio y comprensión sobre qué es lo que se debe implementar, permitiendo tanto al cliente como al equipo de trabajo tener una representación real del alcance que puede llegar a tener el producto.

1.6.2. Herramienta CASE (Computer Aided Software Engineering)

Las herramientas CASE brindan soporte para desarrollar y mantener software. Además, se consideran de gran ayuda al desarrollador de software o administrador de proyecto, durante una o más fases del desarrollo de software (26).

Para el desarrollo del presente trabajo se selecciona la herramienta CASE *Visual Paradigm*.

Visual Paradigm.

Es una herramienta que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, pruebas y despliegue. La herramienta CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas y proyectos UML (27).

Capítulo 1. Fundamentación Teórica

Características:

- Soporte de UML versión 2.1.
- Diagramas de Procesos de Negocio.
- Ingeniería inversa.
- Modelo a código, diagrama a código.
- Diagramas de flujo de datos.
- Soporte ORM.
- Generación de bases de datos.
- Transformación de diagramas de Entidad-Relación en tablas de base de datos.
- Distribución automática de diagramas.

1.6.3. Lenguajes de programación de código abierto

Entre los lenguajes de programación existentes para desarrollar aplicaciones orientadas a la web se encuentran dos grupos fundamentales de acuerdo con la arquitectura Cliente/Servidor: programación del lado del cliente y la programación del lado del servidor.

La programación del lado del cliente incluye aquellos lenguajes que son únicamente interpretados por una aplicación cliente como el navegador web, entre estos lenguajes se encuentran HTML, Java Script y CSS. Los lenguajes de programación del lado servidor son reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible para él. Entre ellos se encuentran: PHP, Java, entre otros.

HTML (*Hyper Text Markup Language*)

Lenguaje de Marcado de Hipertexto (*HTML*, en inglés *HyperText Markup Language*), es mantenido por la *World Wide Web Consortium (W3C)*, es utilizado para la creación de páginas web estáticas. Esta versión incorpora algunas etiquetas nuevas pensadas para hacer que la estructura de la página sea más lógica y funcional.

El enfoque general ha cambiado bastante respecto a versiones anteriores de *HTML*, añadiendo semántica y accesibilidad implícitas, especificando cada detalle y borrando cualquier ambigüedad. También se tiene en cuenta que muchas páginas actuales son dinámicas, pareciéndose más a aplicaciones que a documentos. Algo básico es que *HTML5* está definido en base al *DOM* (la representación interna de una web con la que trabaja un navegador), dejando de lado la representación real, definiendo a la vez un estándar *HTML* y *XHTML* (28).

Capítulo 1. Fundamentación Teórica

CSS 3

Las Hojas de Estilos en Cascada (CSS, por sus siglas en inglés) es un lenguaje creado para controlar el aspecto o 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 crear páginas web con diseño complejo. Separar la definición de los contenidos y la definición de su aspecto presenta numerosas ventajas, ya que obliga a crear documentos *HTML/XHTML* bien formados y con significado completo (también llamados documentos semánticos). Además, mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes (29).

Una vez creados los contenidos de una página se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, además de otras funcionalidades (29).

PHP (*Hyper Text Preprocessor*)

Lenguaje de programación gratuito, usado normalmente para la creación de páginas web dinámicas. Es un lenguaje rápido, con una gran librería de funciones y una amplia documentación.

Además es de código abierto, especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. Entre sus principales características cabe destacar su potencia, alto rendimiento, facilidad de aprendizaje, escasez de consumo de recursos y que es extremadamente simple para el principiante, pero a su vez, ofrece muchas características avanzadas para los programadores profesionales (30).

Algunas de sus ventajas son: es un lenguaje multiplataforma, con capacidad de conexión con la mayoría de los gestores de base de datos que se utilizan en la actualidad, capacidad de expandir su potencial al utilizar una enorme cantidad de módulos (llamados extensiones), posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda. Al ser de código abierto se presenta como una alternativa de fácil acceso para todos, permite las técnicas de Programación Orientada a Objetos, biblioteca nativa de funciones sumamente amplia, no requiere definición de tipos de variables y tiene manejo de excepciones (31).

1.6.4. Sistema Gestor de Contenido (CMS)

Capítulo 1. Fundamentación Teórica

La evolución de las herramientas para la creación de páginas web ha permitido que se construyan sitios web con mayor funcionalidad y presentación visual entre las que se encuentran los Sistemas de Gestión de Contenidos, del inglés *Content Management System* (CMS).

Un CMS, es una aplicación web que permite la creación de software desde un navegador de Internet y sin necesidad de tener grandes conocimientos informáticos. Según Fran Gil Rodríguez un CMS: "...es un software que permite crear una estructura base para la creación y administración de contenidos, principalmente de páginas web. Generalmente un CMS 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" (32).

Los CMS brindan ventajas tales como (33):

- **Inclusión de nuevas funcionalidades en la web:** mediante la inclusión de módulos realizados por terceros pueden solucionarse necesidades concretas, por lo que el sistema puede crecer y adaptarse a las necesidades requeridas.
- **Reutilización de objetos o componentes:** un CMS permite la recuperación y reutilización de páginas, documentos y de cualquier otro objeto publicado o almacenado.
- **Cambios del aspecto de la web:** si no hay una correcta separación entre el contenido y la presentación, un cambio en el diseño puede conllevar a la revisión de muchas páginas para su adaptación. Los CMS facilitan los cambios con la utilización, por ejemplo de estándar CSS (*Cascading Style Sheets* u hojas de estilo en cascada).
- **Control de acceso:** permite gestionar los diferentes permisos a cada área de un sitio web, aplicados a grupos o individuos.

Para el desarrollo de la solución propuesta, se selecciona el CMS Drupal en su versión 7.23, pues además de ser un software completamente libre, cuenta con una amplia comunidad de soporte y abundante bibliografía. Las características fundamentales y principales ventajas de dicho CMS se exponen seguidamente en el epígrafe 1.6.5 del presente documento.

1.6.5. CMS Drupal

El CMS Drupal, es una solución Open-Source distribuida bajo la licencia *General Public License* (GPL8), escrito en PHP y desarrollado y mantenido por una activa comunidad de usuarios. Reconocido por la

Capítulo 1. Fundamentación Teórica

calidad de su código y de las páginas generadas, así como el énfasis especial en la usabilidad y consistencia de todo el sistema (34).

Entre las características que destacan a este CMS se identifican las siguientes (35):

- Gestionar las taxonomías y la estructuración de contenidos de forma personalizable, algo indispensable para sitios de complejidad media-alta.
- Aunque la mayor parte de las instalaciones de Drupal utilizan *MySQL*, existen otras opciones. Drupal incorpora una “capa de abstracción de base de datos” que actualmente está implementada y mantenida para *MySQL* y *PostgreSQL*.
- Drupal ha sido diseñado desde el principio para ser multiplataforma. Puede funcionar con Apache o Microsoft IIS como servidor web y en sistemas como Linux, Solaris, Windows y Mac OS X.
- La administración y configuración del sistema se puede realizar enteramente con un navegador y no precisa de ningún software adicional.
- Drupal puede mostrar en las páginas web de administración, informes sobre popularidad del contenido, o de cómo los usuarios navegan por el sitio.

1.6.6. Entorno de Desarrollo Integrado(IDE)

El entorno de desarrollo es imprescindible en la producción de un software. Un Entorno de Desarrollo Integrado (IDE, en inglés *Integrated Development Environment*) es un conjunto de herramientas integradas para el desarrollo de software. Las herramientas se ejecutan a través de una interfaz gráfica (GUI) y constan de un compilador, un editor y un depurador, entre otros (35). Algunos de los IDE más utilizados son: Netbeans, Eclipse, Aptana, entre otros.

NetBeans

Es una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero se utiliza para desarrollar en varios lenguajes de programación (36).

Características:

- IDE de código abierto y gratuito.
- Es multiplataforma, haciendo posible su uso en diversos sistemas operativos.
- Brinda las herramientas necesarias para el desarrollo en lenguajes como Java, PHP, HTML.
- Presenta aplicaciones para la detección y tratamiento de errores.
- Tiene una amplia comunidad de desarrollo y soporte.

Capítulo 1. Fundamentación Teórica

Para el desarrollo del presente trabajo se selecciona el IDE NetBeans en su versión 7.3, por presentar continuas mejoras en la creación de aplicaciones web con PHP 5, lo cual permite que se puedan reconocer los diferentes formatos de archivos (info, install, module), que integran los módulos de Drupal.

1.6.7. Servidor de aplicaciones

Existen numerosos servidores web dentro de los que se encuentran: Tomcat, Cherokee, Sun Java System Web Server, Samba Server, Apache, entre otros, este último se seleccionó para el desarrollo del sistema propuesto por las características descritas a continuación:

Apache server es software libre y uno de los más difundidos. Algunas investigaciones realizadas han arrojado que más del 70% de los sitios web en internet están manejados por Apache, haciéndolo más extensamente usado que otros servidores web (37).

Entre las características más relevantes de Apache se encuentran:

- Flexible, rápido y eficiente, continuamente actualizado y adaptado a los nuevos protocolos HTTP.
- Multiplataforma.
- Modular: que puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos.
- Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.

1.6.8. Sistema gestor de base de datos (SGBD)

Un Sistema Gestor de Base de Datos (SGBD) o DBMS (*Data Base Management System*) es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Está compuesto por un lenguaje de definición y manipulación de datos y un lenguaje de consulta. Un SGBD permite definir los datos a distintos niveles de abstracción y manipular dichos datos, además de garantizar la seguridad e integridad de los mismos (38).

PostgreSQL

PostgreSQL es un SGBD relacional, orientado a objetos y libre, que tiene prestaciones y funcionalidades equivalentes a muchos gestores de bases de datos comerciales.

Dentro de sus principales ventajas se encuentran:

- Soporta distintos tipos de datos.

Capítulo 1. Fundamentación Teórica

- Posee una gran escalabilidad, haciéndolo idóneo para su uso en sitios web que atienden un gran número de solicitudes.
- Puede ser instalado un número ilimitado de veces sin temor de sobrepasar la licencia.
- Posee estabilidad y confiabilidad legendaria.
- Es extensible a través del código fuente, disponible sin costos adicionales.
- Soporte nativo para los lenguajes más populares del medio: PHP, C++, Perl, Python, entre otros.
- Extensiones para alta disponibilidad, nuevos tipos de índices, datos especiales, minería de datos, entre otros.
- Es multiplataforma, disponible en Linux, Unix, Mac Os X y Windows, entre otros (39).

Se seleccionó PostgreSQL v9.2, debido a que es una herramienta libre, multiplataforma, confiable, estable, con gran escalabilidad, control de concurrencia y funcionalidades que lo destacan como uno de los SGBD más potentes en la actualidad.

1.7. Conclusiones parciales

En este capítulo se analizaron los elementos concernientes a la fundamentación teórica de la investigación, como fueron los diferentes conceptos relacionados con el tema en cuestión para lograr un mayor entendimiento del mismo. Se realizó la investigación de los sistemas similares existentes, a raíz de la cual se pudo concluir que ninguno de los portales web y sistemas cubren los procesos de gestión de información del Buró sindical UCI. Luego de realizar el análisis del proceso de desarrollo de software, tecnologías y herramientas utilizadas para desarrollar sistemas web, se decidió utilizar como Metodología XP con la cual se garantizó el desarrollo de un sistema basado en la rapidez de construcción y se asistirá del lenguaje de modelado UML 2.0 y la herramienta CASE Visual Paradigm 8.0.

Para la implementación del sistema se utilizó el CMS Drupal 7.22 debido a su flexibilidad, su elevado nivel de control en los detalles, y la facilidad de uso. Entre los lenguajes de programación, HTML5 Y CCS3 del lado del cliente y PHP5 del lado del servidor, utilizando como IDE el NETBEANS 7.3, PostgreSQL v9.2 como SGBD por ser una herramienta libre, multiplataforma, confiable, y de gran escalabilidad y como Servidor Web Apache 2.2.

Capítulo 2. Análisis y diseño del sistema

2.1. Introducción

El presente capítulo está dedicado a las características del sistema. Se abordan temas relacionados con la propuesta de solución para responder a la situación problemática en cuestión, además quedarán definidas las principales funcionalidades y los requisitos no funcionales del sistema. Las funcionalidades se describen mediante las historias de usuario (HU). Se lleva a cabo la realización del plan de entrega, donde se indican las historias de usuario que se crearán para cada versión de la aplicación propuesta, así como las fechas en las que se publicarán dichas versiones. Se realiza el plan de iteraciones donde se muestran las HU a realizar en cada iteración según su prioridad en el negocio.

Se describirán los patrones a utilizar en el diseño del sistema y se hace una propuesta de la arquitectura base que brindará soporte a la aplicación.

De igual forma se estructuran las tarjetas CRC para de una manera global, tener idea de la responsabilidad a cumplir por cada clase, así como sus relaciones con el resto del sistema. Por último se presenta el diagrama entidad – relación de la base de datos que sustentará la información persistente sobre los reportes almacenados

2.2. Flujo actual del proceso

Con el objetivo de lograr un mayor conocimiento sobre cómo se desarrollan los procesos en el negocio actualmente, se realizó un estudio que aportó la información necesaria para la modelación del negocio en su estado actual. En cada una de las secciones sindicales (SS) que integran el Buró Sindical de la Universidad (BUS) se realiza mensualmente una reunión en la que intervienen todos los afiliados. Esta reunión está presidida por el Secretario General de la Sección Sindical (SG de la SS) y el organizador en caso que el secretario no se encuentre presente. En cada reunión realizada, se redacta un acta en la cual queda constancia sobre los puntos tratados, las inquietudes planteadas, las respuestas brindadas y los distintos acuerdos tomados durante la misma.

El acta de reunión es un documento oficial del sindicato, en el cual se toman ciertos datos que son indispensables, como son:

Logotipos.

- Título
- Fecha
- Nombre de la Sección Sindical

Capítulo 2. Análisis y Diseño

- Número de acta
- Asistencia
- Invitados

Orden del día.

1. Chequeo de acuerdos.
2. Informe de la Administración.
3. Proceso por una educación de calidad.
4. Informe Sindical.
 - a. Guardia obrera.
 - b. Finanzas.
 - c. Próximas actividades.
5. Asuntos Generales.
 - a. Relación de Acuerdos.
 - b. Principales Inquietudes.

En cada uno de los aspectos que conforman el orden del día, puede haber por parte de los afiliados presentes intervenciones, acuerdos e inquietudes, que se tendrían en cuenta dentro del informe. De acuerdo al cumplimiento de dichos aspectos, se determina si se tienen en cuenta para la próxima reunión, como pendientes, sin cumplir o parcialmente cumplidos. El acta es nuevamente redactada en un modelo digital (previamente realizado), y se cuenta con tres días hábiles para ser enviadas. Finalmente las actas son guardadas por el SG o el organizador del BUS, conformándose de esta manera los reportes de las actas para posteriores consultas.

Las inquietudes planteadas por los afiliados en una reunión pueden ser respondidas por el SG de la SS, quien a su vez consulta previamente con el dirigente responsable de dar la respuesta. En caso de ser una inquietud general, el SG de la SS la envía por correo al SG o al Organizador del Buró, quien consulta la inquietud planteada con el responsable de darle solución a la misma, y a su vez enviará la respuesta a la sección donde fue planteada.

Para la divulgación de la información que se realiza en los diferentes niveles de la organización, se emplean diversos medios de comunicación, tales como: el mural del sindicato, la intranet, correo

Capítulo 2. Análisis y Diseño

electrónico, verbalmente en los matutinos y reuniones realizadas en cada SS. Como resultado de los procesos de divulgación se obtienen: avisos, noticias nacionales e internacionales, efemérides, documentos legales, estructura del Buró y SS, estado financiero de los afiliados, trabajos voluntarios, realización de la guardia obrera, emulación, entre otros. Cuando el proceso de divulgación es efectivo, se logra un adecuado funcionamiento de la organización a todos los niveles, así como mantener a cada afiliado informado en cuanto a los diferentes temas de interés en el ámbito sindical.

A continuación se muestra un diagrama del funcionamiento del proceso de divulgación de la información en la organización sindical de la UCI:

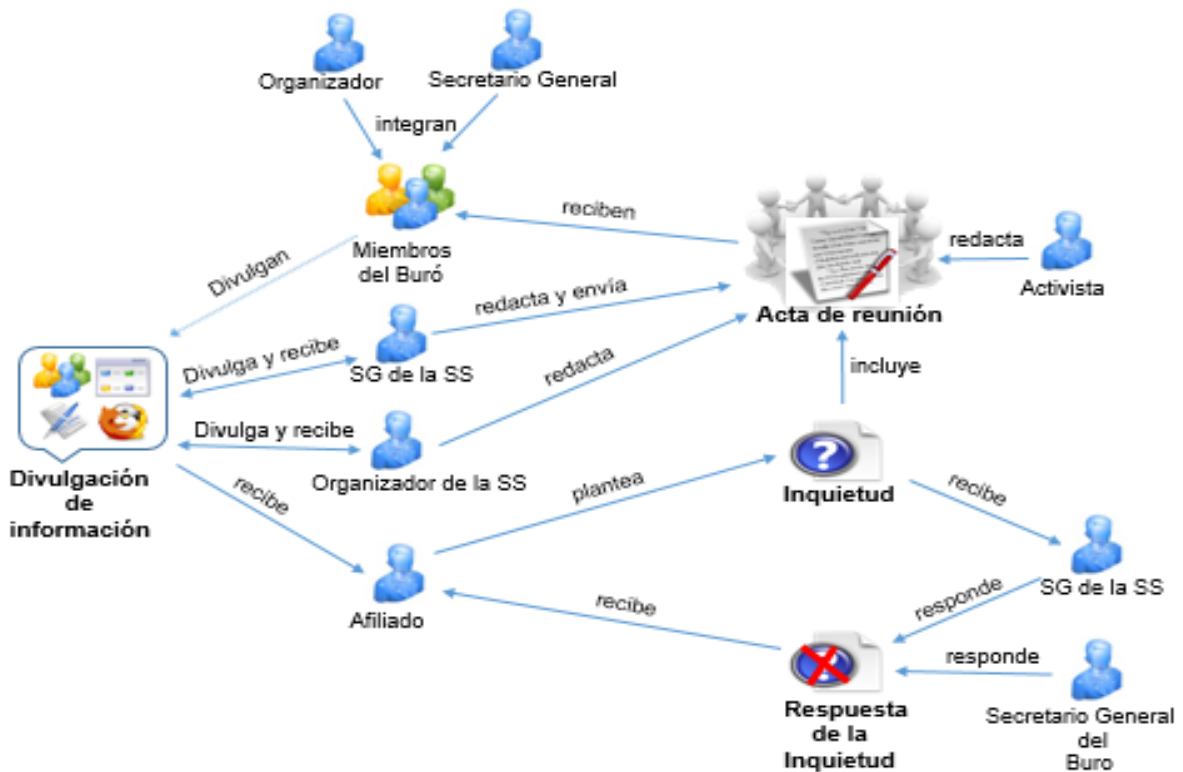


Figura 1: Proceso de divulgación de la Información.

2.3. Características del sistema

Para solucionar el problema identificado se propone la construcción de un sistema de gestión de información que permita tramitar los procesos que se realizan sistemáticamente en el sindicato de la UCI, permitiendo al BUS controlar las SS y estas a su vez a los afiliados que la integran. Además de la divulgación de información, se deben brindar una serie de servicios como: enlaces a otras páginas, descargas de materiales y servicios que se ofertan en la universidad.

Capítulo 2. Análisis y Diseño

El sistema contará con varios módulos a través de los cuales los usuarios podrán acceder, y si su rol lo permite, modificar las siguientes informaciones:

- 1- **Acta:** Se encarga de la gestión de las actas de reuniones que se generan. Se utiliza para crear las actas por SS.
- 2- **Usuario del dominio:** Se utiliza para la sincronización de los usuarios del dominio, con su cuenta relacionada de afiliado en la web, es decir, cada afiliado puede tener una cuenta de usuario asociada, entonces este módulo permite crear la sincronización de los afiliados con sus cuentas de usuario del dominio.
- 3- **Afiliado:** Se emplea para la gestión de los afiliados de las secciones sindicales. Es utilizado por el módulo Sección sindical, el cual permite asignar afiliados a una sección sindical en específico.
- 4- **Área:** Se utiliza para la gestión de las áreas a las que pertenecen las secciones sindicales. Está relacionado con el módulo Sección Sindical utiliza para asignar a las diferentes áreas que se crean las SS correspondientes a dichas áreas.
- 5- **Sección sindical:** Se encarga de la gestión de las secciones sindicales del sitio. Utiliza el modulo Afiliado, para poder asignar afiliados a una sección sindical.
- 6- **Estructura:** Se utiliza para la gestión de las estructuras que se aplican a las SS. Una estructura no es más que la organización que posee una sección sindical por cargos.
- 7- **Inquietud:** Empleado en la gestión de las inquietudes que se realizan en las reuniones por SS. Las inquietudes se obtienen a partir de las actas de reuniones, luego los afiliados con los permisos correspondientes pueden realizar cualquier operación de gestión sobre esas inquietudes, así como asignarles una respuesta. Cuando un afiliado no puede responder una inquietud, entonces puede elevar dicha inquietud al BUS para que se le conceda una respuesta.
- 8- **Reporte:** Utilizado para la generación de reportes por los diferentes módulos que se gestionan en el sistema. Los reportes permiten realizar filtros para luego poder exportarlos al formato PDF y descargarlos.
- 9- **Sindicato:** Constituye la base de toda la aplicación. Implementa la gestión de los cargos y permisos, así como las configuraciones generales que se utilizan en la aplicación.
- 10- **Capital humano:** Se obtienen diferentes datos utilizando servicios web de la universidad. Por ejemplo, los trabajadores que pertenecen a un área específica. Implementa la lógica de conexión a los servicios web y la obtención de datos de forma abstracta.

Capítulo 2. Análisis y Diseño

2.4. Fase de exploración

Esta es la primera fase de la metodología *XP*, en la cual los clientes plantean a grandes rasgos las funcionalidades que son de interés para la elaboración del producto, transformándose las mismas en historias de usuario. Partiendo de la información obtenida el equipo de desarrollo evaluará de forma general el tiempo de codificación, se familiarizará con las herramientas, tecnologías y prácticas que serán utilizadas en el proyecto, además se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo para ello.

La fase de exploración toma de unas semanas a pocos meses, dependiendo de la habilidad que tengan los programadores con las tecnologías seleccionadas. Es válido aclarar que las estimaciones realizadas en esta etapa son primarias, pues estas se basan en datos de alto nivel los cuales pueden variar a medida que se analicen con mayor cuidado y detalle en las siguientes fases.

2.4.1. Historias de usuario

Es una simple descripción de una funcionalidad del software por la que el cliente va a pagar. Los detalles extra son agregados cuando la historia de usuario es implementada, siendo estos procesos simples y sencillos. En cualquier momento estas historias pueden romperse, remplazarse, unirse o dividirse. Entre las principales características que debe tener una HU se encuentran: debe ser entendida por el cliente (representa un concepto y no una especificación detallada), debe devolver algún valor para el cliente, el tamaño debe ser tal que se puedan construir varias de ellas en una iteración (duración aproximada entre 1-3 semanas ideales), deben ser independientes unas de otras y cada historia debe ser probada (40).

Para la información contenida en la HU existen varias plantillas sugeridas pero no existe un consenso al respecto. En muchos casos sólo se propone utilizar un nombre y una descripción, además de una estimación de esfuerzo en días ideales. Otras de las actividades a tener en cuenta es la granularidad de las historias, lo cual es muy variable en dependencia de la complejidad del sistema, debido a esto, cada característica importante debe establecerse como una HU.

Al comienzo de cada iteración estarán registrados los cambios en las HU y es a partir de ello que se planificará la siguiente. Además pueden ser descompuestas en tareas de programación y asignadas a los programadores para ser implementadas durante una iteración.

Capítulo 2. Análisis y Diseño

En el análisis para el desarrollo del sistema propuesto, fueron obtenidas cuarenta (40) funcionalidades. A continuación se plantean las funcionalidades de la aplicación:

RF1 Autenticar un usuario afiliado

RF2 Gestionar afiliado

RF2.1 Adicionar afiliado

RF2.2 Modificar afiliado

RF2.3 Buscar afiliado

RF2.4 Mostrar datos de un afiliado

RF2.5 Eliminar afiliado

RF3 Asignar roles a un afiliado

RF4 Gestionar área de secciones sindicales

RF4.1 Adicionar un área

RF4.2 Modificar área

RF4.3 Listar áreas

RF4.4 Eliminar área

RF5 Gestionar sección sindical

RF5.1 Adicionar sección sindical

RF5.2 Modificar sección sindical

RF5.3 Listar secciones sindicales

RF5.4 Buscar sección sindical

RF5.5 Eliminar sección sindical

RF6 Gestionar afiliado de una sección sindical

RF6.1 Adicionar afiliado a sección sindical

RF6.2 Eliminar afiliado de sección sindical

RF6.3 Listar afiliados

RF7 Gestionar acta

RF7.1 Crear acta

RF7.2 Modificar datos de un acta

RF7.3 Mostrar datos de un acta

RF7.4 Exportar a PDF un acta

RF7.5 Listar actas

Capítulo 2. Análisis y Diseño

RF7.6 Eliminar acta

RF8 Gestionar inquietud

RF8.1 Listar inquietudes

RF8.2 Mostrar inquietud

RF8.3 Eliminar inquietud

RF8.4 Escalar inquietud al buró

RF9 Gestionar respuesta

RF9.1 Asignar respuesta

RF9.2 Modificar respuesta

RF9.3 Eliminar respuesta.

RF10 Especificar la plantilla de la estructura organizativa de las secciones sindicales.

RF11 Modificar la plantilla de la estructura organizativa de las secciones sindicales.

RF12 Mostrar la estructura organizativa de una sección sindical.

RF13 Mostrar la estructura organizativa del buró sindical.

RF14 Mostrar reportes de las secciones sindicales.

RF15 Mostrar reportes de los afiliados de las secciones sindicales.

RF16 Mostrar reportes de las actas de reuniones de las secciones sindicales.

RF17 Mostrar reportes de las inquietudes y respuestas de las secciones sindicales.

Una vez identificadas las funcionalidades del sistema se procede a realizar las historias de usuario, a continuación se muestra como quedan conformadas las siguientes HU: Autenticar usuario, Gestionar afiliado, Gestionar afiliados de una sección sindical y Gestionar acta. El resto de las HU se encuentran en los anexos del presente documento (Ver Anexo 1).

Tabla 2: HU Autenticar usuario

Historia de Usuario		
ID: 1	Nombre: Autenticar usuario	
Descripción: El usuario que va a interactuar con el sistema se debe identificar insertando su usuario y contraseña del dominio.		
Estimación aproximada: 7	Prioridad: Alta	H.U original: 1
Observaciones:		

Capítulo 2. Análisis y Diseño

Tabla 3: HU Gestionar afiliado

Historia de Usuario		
ID: 2	Nombre: Gestionar afiliado	
Descripción: Permite insertar, modificar, buscar y eliminar la información relacionada con la cuenta asignada a un afiliado.		
Estimación aproximada: 15	Prioridad: Alta	H.U original: 2
Observaciones:		

Tabla 4: HU Gestionar afiliados de una sección sindical.

Historia de Usuario		
ID: 3	Nombre: Gestionar afiliados de una sección sindical	
Descripción: Permite asignar o eliminar un afiliado de una SS. Además muestra el listado de afiliados de una SS.		
Estimación aproximada: 15	Prioridad: Media	H.U original: 6
Observaciones:		

Tabla 5: HU Gestionar acta

Historia de Usuario		
ID: 4	Nombre: Gestionar acta	
Descripción: Permite crear, modificar, mostrar los datos y eliminar un acta de reunión, además permite listar todas las actas de reuniones y exportar a PDF un acta de reunión.		
Estimación aproximada: 30	Prioridad: Media	H.U original: 7
Observaciones:		

2.4.2. Personal relacionado con el sistema

Uno de los principales factores que se deben tener en cuenta cuando se comienza el desarrollo de un sistema informático es la delimitación de la audiencia a la cual va dirigido el mismo, lo cual en algunas ocasiones puede referirse a un personal relacionado con el sistema o alguien completamente ajeno a él. Se debe especificar que esta audiencia a su vez puede ser dividida en grupos atendiendo a sus necesidades.

Capítulo 2. Análisis y Diseño

Se define como persona relacionada con el sistema, a toda aquella que de una manera u otra interactúe con este, obteniendo un resultado de uno o varios procesos que se ejecutan en el mismo. También son considerados como personas relacionadas con el sistema aquellas que se encuentran involucradas en dichos procesos, participan en ellos pero no obtienen ningún resultado de valor. La siguiente tabla muestra las personas que interactúan con el sistema y las acciones que los mismos pueden realizar en él.

Tabla 6: Personal relacionado con el sistema.

Personal	Justificación
Usuario sin autenticar	<ul style="list-style-type: none"> - Interactúa solamente con los eventos disponibles en el portal principal. El menú para la gestión sindical no está incluido
Usuario autenticado	<ul style="list-style-type: none"> - Ver acta de reunión (de su SS) - Ver estructura organizativa (de su SS) - Ver inquietudes y Respuestas
Organizador	<ul style="list-style-type: none"> - Ver acta de reunión (de su SS). - Editar acta de reunión (de su SS). - Modificar Acta de reunión (de su SS). - Ver estructura organizativa (de su SS). - Ver inquietudes y Respuestas.
Secretario General de la SS.	<ul style="list-style-type: none"> - Ver acta de reunión (de su SS). - Editar acta de reunión (de su SS). - Modificar Acta de reunión (de su SS). - Ver estructura organizativa (de su SS). - Modificar estructura organizativa (de su SS). - Gestiona los roles y permisos de los trabajadores (de su SS). - Gestiona las respuestas a las inquietudes realizadas por los trabajadores (de su SS). - Generar reporte de la SS. - Gestionar Avisos (para su SS). - Ver inquietudes y Respuestas (de su SS).
Miembro BUS	<ul style="list-style-type: none"> - Gestionar actas de reuniones (de todas las SS). - Generar reporte de Actas de reunión (de todas las SS). - Gestiona los roles y permisos de los trabajadores (Todos).

Capítulo 2. Análisis y Diseño

	<ul style="list-style-type: none">- Gestiona las respuestas a las inquietudes realizadas por los trabajadores (de todas las SS).- Generar reporte de las SS (todas).- Generar reporte de Inquietudes y respuesta (todas las SS).- Ver inquietudes y Respuestas (de Todas las SS).- Gestionar las áreas.- Gestionar las Secciones Sindicales.- Gestionar Historia en el portal.- Gestionar la Estructura Sindica presente en el portal.- Gestionar Aspectos Legales presentes en el portal.- Gestionar Avisos y Noticias del portal.
--	--

2.5. Fase de Planificación

La fase de planificación tiene como propósito general que los clientes y desarrolladores lleguen a un acuerdo sobre cuáles HU deben estar listas para la primera iteración. De forma particular es en esta fase donde los programadores obtienen la estimación del esfuerzo necesario para la elaboración de las HU. Es necesario aclarar que el tiempo ideal de desarrollo es aquel en el cual se escribe el código sin distracciones, logrando dedicarse a tiempo completo a esta actividad. Dicha estimación incluye todo el esfuerzo asociado a la implementación de la HU, como son: las pruebas unitarias, integración, refactorización del código, la preparación y ejecución de las pruebas de aceptación, entre otras actividades.

La fase de planificación es de muy corta duración, pero de relevante importancia para el avance de la solución a desarrollar. Esta se puede realizar basándose en el tiempo o el alcance. La velocidad del proyecto es utilizada para establecer cuántas historias se pueden implementar antes de una fecha determinada o cuánto tiempo tomará implementar un conjunto de HU. Al planificar por tiempo, se multiplica el número de iteraciones por la velocidad del proyecto, determinándose cuántos puntos se pueden completar. Al planificar según el alcance del sistema, se divide la suma de puntos de las historias de usuario seleccionadas entre la velocidad del proyecto, obteniendo el número de iteraciones necesarias para su implementación (41).

Capítulo 2. Análisis y Diseño

2.5.1. Prioridad de las Historias de Usuario

La asignación de la prioridad a las HU es una actividad llevada a cabo en conjunto con el cliente, en la misma se decide el orden en el cual se implementarán las HU, siempre y cuando estén de acuerdo usuario y desarrollador. Las HU con menor número de prioridad son las más importantes quedando conformadas de la siguiente manera:

Tabla 7: Prioridad de las HU.

Historia de Usuario	Prioridad
Autenticar usuario	1
Gestionar área	1
Gestionar sección sindical	1
Gestionar afiliado	1
Asignar roles a un afiliado	2
Gestionar afiliados de una sección sindical	2
Gestionar acta	2
Gestionar inquietud	2
Gestionar respuesta	2
Especificar la plantilla de la estructura organizativa de las SS	3
Modificar la plantilla de la estructura organizativa de las SS	3
Mostrar la estructura organizativa de una SS	3
Mostrar la estructura organizativa del buró sindical	3
Mostrar reportes de las SS	3
Mostrar reportes de los afiliados de las S	3
Mostrar reportes de las actas de reuniones de las SS	3
Mostrar reportes de las inquietudes y respuestas de las SS	3

2.5.2. Estimación del esfuerzo de las HU

Una vez asignadas las prioridades de las HU, se prosigue a estimar el esfuerzo necesario para la elaboración de cada una de ellas por parte del desarrollador. Esta estimación se basa principalmente en la velocidad del equipo de desarrollo y en la semejanza con historias de usuario desarrolladas con anterioridad. Las HU de la presente investigación tienen un valor entre 4-20 puntos (días ideales), 1-3 puntos (medida utilizada normalmente, en semanas ideales).

Capítulo 2. Análisis y Diseño

Los puntos estimados son expresados en días ideales, ha de tenerse en cuenta que pocas veces el cronograma se lleva a cabo exactamente como se planifica. El esfuerzo necesario para construir las HU está basado en la técnica de estimación para el desarrollo ágil, quedando conformada de la siguiente manera:

Tabla 8: Estimación del esfuerzo para las HU.

Historia de Usuario	Esfuerzo estimado (Puntos estimados)
Autenticar usuario	4
Gestionar área de sección sindical	10
Gestionar sección sindical	10
Gestionar afiliado	10
Asignar roles a un afiliado	5
Gestionar afiliado de una sección sindical	10
Gestionar acta	20
Gestionar inquietud	10
Gestionar respuesta	8
Especificar la plantilla de la estructura organizativa de las SS	15
Modificar la plantilla de la estructura organizativa de las SS	12
Mostrar la estructura organizativa de una SS	10
Mostrar la estructura organizativa del buró sindical	10
Mostrar reportes de los datos de las SS	8
Mostrar reportes de los afiliados de las SS	8
Mostrar reportes de las actas de las SS	8
Mostrar reportes de las inquietudes y respuestas	8
Total:	166 días

2.5.3. Cronograma de liberación

En el momento de planificar la liberación de un producto de software se debe llevar un balance de esta, pues si se realiza muy pronto no se tendrán suficientes funcionalidades que ameriten dicha liberación, por otro lado esperar mucho tiempo conlleva a que el software desarrollado quede atrás respecto a la competencia.

Capítulo 2. Análisis y Diseño

Entre los elementos a tener en cuenta en el cronograma de liberación es fundamental saber que rara vez un plan marcha respecto a lo acordado, por esa razón el plan de liberación está en constante cambio. Un ejemplo de ello se tiene cada vez que el usuario cambia la prioridad de una historia, cuando el desarrollador divide o une HU, cuando aparecen imprevistos en las tecnologías a utilizar, entre otras acciones. Es por esto que los cambios deben ser aceptados. El cronograma de liberación de la presente investigación queda conformado de la siguiente manera:

Tabla 9: Cronograma de liberación.

Iteración	Fecha de liberación
Primera iteración	1ra semana Octubre 2014
Segunda iteración	4ta semana Noviembre 2014
Tercera iteración	1ra semana Enero 2015

2.6. Fase de Iteración

Una vez identificadas HU y estimado el esfuerzo necesario para el desarrollo de cada una de ellas, se procede a realizar la planificación de las etapas de implementación del sistema. Con este fin se define un plan que contiene las iteraciones a realizar y la relación de las HU que serán implementadas y en qué orden para cada iteración, teniéndose en cuenta la duración de las mismas. La cantidad de iteraciones necesarias (IN) para realizar todas las HU identificadas, es obtenida sumando todos los puntos de esfuerzo (PE) de las HU y dividiéndolos entre la velocidad de iteración del equipo (VIE).

$$IN = PE / VIE$$

$$PE = 166 \text{ días (23.7 semanas)}$$

$$IN = 23.7 / 7.5$$

$$IN = 3.16$$

Para calcular la velocidad de iteración del equipo (VIE) se divide la cantidad de desarrolladores (CD) entre el factor de dedicación (FD) al proyecto (para la presente investigación es de 2 [50%]) y se multiplica por el tiempo de duración máximo de una iteración (DMI) (en el caso de la presente investigación el tiempo es de 15 días máximo).

$$VIE = (CD / FD) * DMI$$

$$CD = 1, FD = 50\% (2), DMI = 15$$

$$VIE = (1 / 2) * 15$$

$$VIE = 7.5$$

Capítulo 2. Análisis y Diseño

Del cálculo anterior se obtiene que el valor de la velocidad de equipo es aproximadamente 8, y las iteraciones necesarias para desarrollar las historias de usuario 3. De manera que el tiempo de desarrollo de las iteraciones esté balanceado se tomaron todas aquellas HU de la más alta prioridad (1, 2) y se promediaron sus puntos de esfuerzo (9.8), quedando las iteraciones conformadas de la siguiente manera:

Tabla 10: Primera iteración.

Iteración		
Número: 1	HU por orden: 1, 2, 3, 4, 5, y 7.	Duración total: 59
Descripción: Se desarrollan las primeras HU por las que poseen una mayor prioridad, conformando así un producto que aún sin terminar puedan mostrar las funcionalidades del sistema. Además dará una primera vista al cliente que suministrará su criterio para la incorporación de nuevos elementos y modificación de los existentes. Al finalizar dicho proceso se contará con las funcionalidades descritas en las historias de usuarios 1, 2, 3, 4, 5 y 7.		

Tabla 11: Segunda iteración.

Iteración		
Número: 2	HU por orden: 6, 8, 9, 10 y 11	Duración total: 55
Descripción: Se continúan desarrollando historias de usuario de la más alta prioridad que por cuestión de tiempo no pudieron añadirse a la primera iteración. Se llevan a cabo los señalamientos hechos por el cliente sobre la primera iteración.		

Tabla 12: Tercera iteración.

Iteración		
Número: 3	HU por orden: 12, 13, 14, 15, 16 y 17	Duración total: 52
Descripción: Se desarrollan las historias de usuario de prioridad más baja. Al finalizar dicho proceso se contará con las funcionalidades descritas en las historias de usuarios 12, 13, 14, 15, 16 y 17 obteniéndose el producto final.		

2.6.1. Características no funcionales

Al concebir las historias de usuario no se detallan ciertos aspectos que se deben de precisar en la elaboración de una aplicación. Estos aspectos influyen en gran medida en el funcionamiento del producto

Capítulo 2. Análisis y Diseño

final. La metodología *XP* orienta que los detalles de implementación de la historia de usuario se tienen en cuenta en el mismo momento de la concepción, por estas razones se describen a continuación de forma general, algunas de las características no funcionales que debe tener el sistema.

Los requerimientos no funcionales de un sistema son “propiedades, cualidades o restricciones que el producto debe cumplir, tales como restricciones de tiempo, estándares, de desarrollo, entre otros con el objetivo de lograr un producto confiable, atractivo y seguro” (42).

RNF 1 Usabilidad

El sistema podrá ser usado por personas con conocimientos básicos en el manejo de computadoras. Deberá contar con una interfaz amigable para facilitar la interacción.

RNF 2 Confidencialidad

Para que el usuario pueda realizar funcionalidades avanzadas en el sistema debe estar autenticado y de acuerdo con el rol que le sea asignado tendrá los permisos asociados. Esto garantizará el acceso controlado a la información, donde se mostrarán las interfaces a los usuarios, en dependencia de los niveles de acceso que posea.

RNF 3 Rendimiento

El sistema deberá ser capaz de gestionar toda la información y dar respuesta a las solicitudes en el menor tiempo posible.

RNF 4 Portabilidad

El sistema debe ser multiplataforma, por lo que debe utilizarse tanto sistema operativo Windows como en GNU/Linux.

RNF 5 Requisitos de software

Para las computadoras clientes:

- *Sistema Operativo:* Windows o GNU/Linux.
- *Navegador Web:* Internet Explorer 9.0 o mayor, Mozilla Firefox 25.0 o mayor.
- *Conectividad:* Tener conexión a la red de la universidad.

Para los servidores de aplicaciones y de base de datos:

- *Servidor de aplicaciones:* Apache 2.0, PHP 5.4 o mayor.

Capítulo 2. Análisis y Diseño

- *Servidor de Base Datos*: PostgreSQL 9.0.

RNF 6 Requisitos de hardware

Para las computadoras clientes:

- Al menos 256 MB de memoria RAM.
- Procesador Pentium 4 o superior.

Para los servidores:

- Se requiere tarjeta de red.
- Al menos 4GB de RAM.
- Al menos 4GB de espacio libre en el disco duro.
- Procesador Pentium 4 o superior.

2.6.2. Utilización de Patrones

Los patrones de diseño son descripciones de clases y objetos relacionados que están particularizados para resolver un problema de diseño general en un determinado contexto (44). Entre los más conocidos se encuentran los patrones de Asignación de Responsabilidades y los patrones de la Banda de los Cuatro.

Patrones de asignación de responsabilidades (GRASP)

Describen los principios fundamentales de la asignación de responsabilidades a objetos, de forma tal que se pueda diseñar software orientado a objetos (45). Los patrones *GRASP* son los siguientes: Experto, Creador, Bajo Acoplamiento, Alta Cohesión, Controlador, entre otros. Muchos de estos principios básicos se tuvieron presente durante la planificación del diseño de los componentes de la aplicación propuesta.

Experto en información

Es el principio básico de asignación de responsabilidades. Indica que la responsabilidad de la creación de un objeto o la implementación de un método, debe recaer sobre la clase que conoce toda la información necesaria para crearlo. De este modo se obtiene un diseño con mayor cohesión y así la información se mantiene encapsulada (disminución del acoplamiento).

Este patrón es utilizado en toda la implementación de los módulos que se utilizan en el sistema, un ejemplo se puede apreciar en el módulo Acta que es el encargado de la gestión de las actas de reuniones de las secciones sindicales, donde se utiliza la clase Acta. Un acta como entidad, está compuesta por

Capítulo 2. Análisis y Diseño

intervenciones, acuerdos, inquietudes, temas, etc. Cada una de estas propiedades, se corresponde con una clase, ya que el acta en sí, no debe manejar la información relacionada con un acuerdo o una inquietud, por lo que aplicando este patrón, quedaría de la siguiente forma:

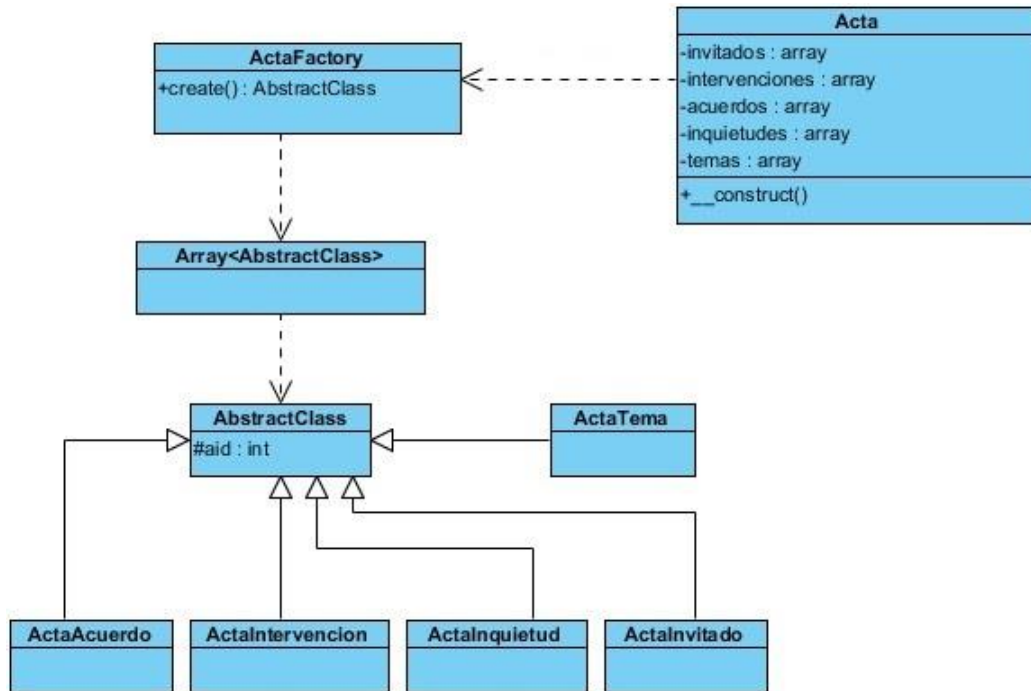


Figura 2: Diagrama de clases patrón Experto.

```
2
3 /**
4  * Ejemplo de la utilización del patrón 'Experto en Información'
5  * que utiliza la clase Acta del módulo 'Acta'
6  */
7 class Acta extends Entity {
8
9     public $invitados;
10    public $intervenciones;
11    public $acuerdos;
12    public $inquietudes;
13    public $temas;
14
15    public function __construct() {
16        $propiedades = array('invitados', 'intervenciones', 'acuerdos', 'inquietudes', 'temas');
17        foreach ($propiedades as $propiedad) {
18            // Instanciamos cada propiedad utilizando el patrón Factory
19            // que nos permite crear una instancia de cualquier clase
20            // registrada en el registro de clases de la fábrica
21            $this->$propiedad = ActaFactory::create($propiedad);
22        }
23    }
24
25 }
```

Figura 3: Código donde se implementa el patrón Experto.

Capítulo 2. Análisis y Diseño

Creador

El patrón creador permite identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases.

La nueva instancia deberá ser creada por la clase que tiene la información necesaria para realizar la creación del objeto, o usa directamente las instancias creadas del objeto, o almacena o maneja varias instancias de la clase y contiene o agrega la clase.

Una de las consecuencias de usar este patrón es la visibilidad entre la clase creada y la clase creador. Una ventaja es el bajo acoplamiento, lo cual supone facilidad de mantenimiento y alta reutilización. La creación de instancias, es una de las actividades más comunes en un sistema orientado a objetos. En consecuencia es útil contar con un principio general para la asignación de las responsabilidades de creación. Si se asignan bien, el diseño puede soportar un bajo acoplamiento, mayor claridad, encapsulación y reutilización.

La utilización de este patrón se evidencia en el módulo Acta, con la implementación de la clase Acta, específicamente en la función que permite agregar un acuerdo. Es decir, este patrón permitió identificar que la clase Acta es quien tiene o posee la información que requiere un acuerdo, ya que se comporta como un controlador, entonces es ella quien debe crear en todo momento, la creación o instanciación de un nuevo acuerdo para poder agregarlo en el acta. Se crea la instancia de la clase Acuerdo, ya que esta es la clase que se especializa en dicha información, pero la información la posee el Acta, por lo tanto se le asigna la responsabilidad de crear las instancias de los acuerdos.

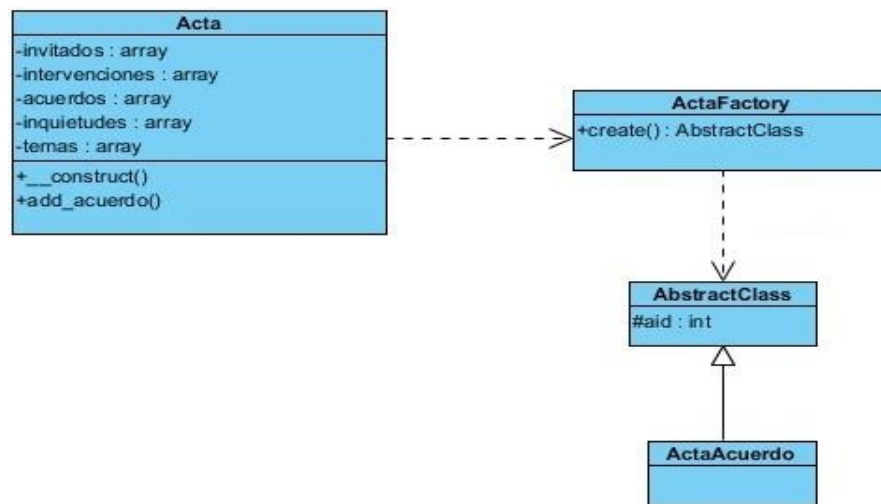


Figura 4: Diagrama de clases patrón Creador.

Capítulo 2. Análisis y Diseño

```
3 /**
4  * Agrega un acuerdo a la lista de acuerdos que se
5  * definen en esta acta de reunión
6  */
7 public function add_acuerdo($values = array(), $group = ACTA_GRUPO_INFORME_ADMINISTRACION, $db = FALSE) {
8     if (empty($db)) {
9         // Procedemos a asignar los valores por defecto
10        // que no se definieron en la lista de valores
11        .
12        .
13        .
14        // Primero creamos una instancia de la clase Acuerdo
15        // para utilizar las ventajas de la misma
16        $acuerdo = new Acuerdo($values);
17    }
18    else {
19        $acuerdo = new Acuerdo();
20        $acuerdo->load_row($values);
21    }
22
23    // Agregamos el acuerdo creado al grupo correspondiente
24    // y actualizamos la referencia al último número de acuerdos
25    .
26    .
27    .
28 }
```

Figura 5: Código que implementa el patrón Creador.

Alta cohesión y bajo acoplamiento

Los conceptos de cohesión y acoplamiento no están íntimamente relacionados, sin embargo se recomienda tener un mayor grado de cohesión con un menor grado de acoplamiento. De esta forma se tiene menor dependencia y se especifican los propósitos de cada objeto en el sistema.

Alta cohesión

Este patrón plantea que la información que almacena una clase debe de ser coherente y debe estar (en la medida de lo posible) relacionada con la clase.

La alta cohesión se utiliza, en la construcción de cada módulo que se propone para el sistema. Es decir, en dependencia de la arquitectura de Drupal, si se enfoca cada módulo como una clase en sí, cada módulo entonces representa la información correspondiente con cada funcionalidad presente en el sistema. Específicamente, se lleva a cabo con la utilización de la Cohesión Lógica, donde cada módulo realiza múltiples tareas relacionadas, pero en tiempo de ejecución sólo una de ellas será llevada a cabo.

Un ejemplo se aprecia en el módulo Sección Sindical, el cual permite la gestión de las SS y la información que poseen en el sistema, o sea permite crear, editar o eliminar una sección sindical entre otras tareas. Asimismo durante la ejecución de la edición o eliminación de una SS, todas ellas son tareas relacionadas pero que se ejecutan únicamente durante un tiempo de ejecución, en dependencia de la tarea que se esté realizando.

Capítulo 2. Análisis y Diseño

Bajo acoplamiento

Aporta la idea de tener una baja relación entre las clases, de forma tal que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización, y aumentando la facilidad para encontrar errores.

El bajo acoplamiento se identifica con la especialización de cada módulo, lo que ha permitido obtener módulos que puedan ser totalmente reutilizables para otros sistemas. Un ejemplo lo constituye el hecho de que cada SS posee una estructura, por lo que se implementa un módulo que se encarga de generar la estructura, así como gestionarla y configurarla como punto único de gestión. Como dicho módulo no depende de la SS, puede ser llevado a otro sistema con similares características e indicando las entidades a las que se desea aplicar una estructura determinada.

Otro ejemplo que se pone en evidencia este patrón es en el módulo Capital Humano, el cual se especializa en la utilización y obtención de servicios web ofrecidos por la universidad para obtener datos de trabajadores, áreas, cargos, etc. que ya están definidos por esos servicios. El módulo actúa como una capa de conexión con dichos servicios y puede ser utilizado por cualquiera del resto de módulos, como el módulo Afiliado, lo que permite que dicho módulo Capital Humano, sea independiente y reutilizable en toda su totalidad para otros sistemas similares que se desee acceder al mismo tipo de información.

2.6.3. Tarjetas CRC

Cuando se lleva a cabo el diseño de un sistema XP propone la idea de llevar a cabo las tareas de la manera más simple posible, aplicando prácticas especializadas que la metodología propone. Dichas prácticas inciden directamente en la realización y elaboración del diseño de un software pues, cuando se tiene un diseño simple se presta mayor atención a codificar, tarea sobre la cual hace más énfasis esta metodología.

La metodología XP no recomienda la utilización de diagramas de clases, debido a la gran cantidad de documentación e inconvenientes que ello requiere, aunque tampoco excluye del todo su elaboración, en su lugar orienta la realización de Tarjetas de Contenido, Responsabilidad, Colaboración, (*CRC*, por sus siglas en inglés). Dichas tarjetas en cada una de las iteraciones van siendo definidas como herramientas de reflexión en el diseño de software orientado a objetos. Estas tarjetas brindan un acercamiento a la identificación de las clases y la información referente a los objetos desarrollados en el proyecto. Cada tarjeta identifica una clase, sus responsabilidades y relaciones con otras clases.

Capítulo 2. Análisis y Diseño

Cada tarjeta CRC contiene el nombre de la clase que representa, además existe una descripción de las responsabilidades (métodos) asociados con la clase, así como una lista de otras clases relacionadas mediante el envío de mensajes. El sistema CRC se usa principalmente como herramienta didáctica y como metodología para estudiar la conducta de los diseñadores orientados a objetos, son también un recordatorio y ayudan a los programadores experimentados y principiantes a comunicarse entre sí acerca de la modelación del entorno con objetos (46).

A continuación se muestra como quedaron conformadas las tarjetas CRC para las clases que pertenecen al módulo Actas, el resto de dichas tarjetas se encuentran en los anexos del presente documento (Ver Anexo 2).

Tabla 13: Tarjeta CRC: Clase Base.

Clase: Base	
Responsabilidades:	Colaboraciones:
<ol style="list-style-type: none">1. Es la clase base del Módulo.2. Maneja la información de: invitados, asistencia, acuerdos, intervención, tema y ausentes.	<ul style="list-style-type: none">- Acta_DB_Objects.

Tabla 14: Tarjeta CRC: Clase Actas.

Clase: Actas	
Responsabilidades:	Colaboraciones:
<ol style="list-style-type: none">1. Crea un acta de reunión de una SS.	<ul style="list-style-type: none">- <i>Entity</i> (Clase nativa del API de Drupal).- Inquietud.- Tema.- Intervención.- Acuerdo.- Invitado.- Asistencia.- Ausente.

Tabla 15: Tarjeta CRC: Clase Ausente.

Clase: Ausente	
Responsabilidades:	Colaboraciones:
<ol style="list-style-type: none">1. Crear un ausente.	<ul style="list-style-type: none">- Base.

Capítulo 2. Análisis y Diseño

Tabla 16: Tarjeta CRC: Clase Asistencia.

Clase: Asistencia	
Responsabilidades:	Colaboraciones:
1. Crear registro de asistencia.	- Base.

Tabla 17: Tarjeta CRC: Clase Invitado.

Clase: Invitado	
Responsabilidades:	Colaboraciones:
1. Crear un invitado.	- Base.

Tabla 18: Tarjeta CRC: Clase Acuerdo.

Clase: Acuerdo	
Responsabilidades:	Colaboraciones:
1. Crear un acuerdo.	- Base.

Tabla 19: Tarjeta CRC: Clase Intervención.

Clase: Intervención	
Responsabilidades:	Colaboraciones:
1. Crear el registro de una intervención.	- Base.

Tabla 20: Tarjeta CRC: Clase Tema.

Clase: Tema	
Responsabilidades:	Colaboraciones:
1. Crear el registro de los temas tratados en la reunión.	- Base.

Tabla 21: Tarjeta CRC: Clase Inquietud.

Clase: Inquietud	
Responsabilidades:	Colaboraciones:
1. Crear una inquietud.	- Base.

2.6.4. Diseño de la Base de Datos

En cualquier aplicación en la cual se gestione información la base de datos desempeña un papel fundamental. El diseño de la misma es algo esencial para el desarrollo de sistemas de este tipo, ya que

Capítulo 2. Análisis y Diseño

esta permite el almacenamiento de la información de forma coherente y organizada, para evitar pérdidas e inconsistencias, de esta manera la recuperación de la información es de forma rápida y flexible. Por tal motivo enfocarse en su desarrollo es una tarea vital. Para lograr diseñar la base de datos se parte del diagrama de clases persistentes (Ver Anexo 4), una vez realizado dicho diagrama, se procede a realizar el diagrama modelo de datos mostrado en la siguiente figura:

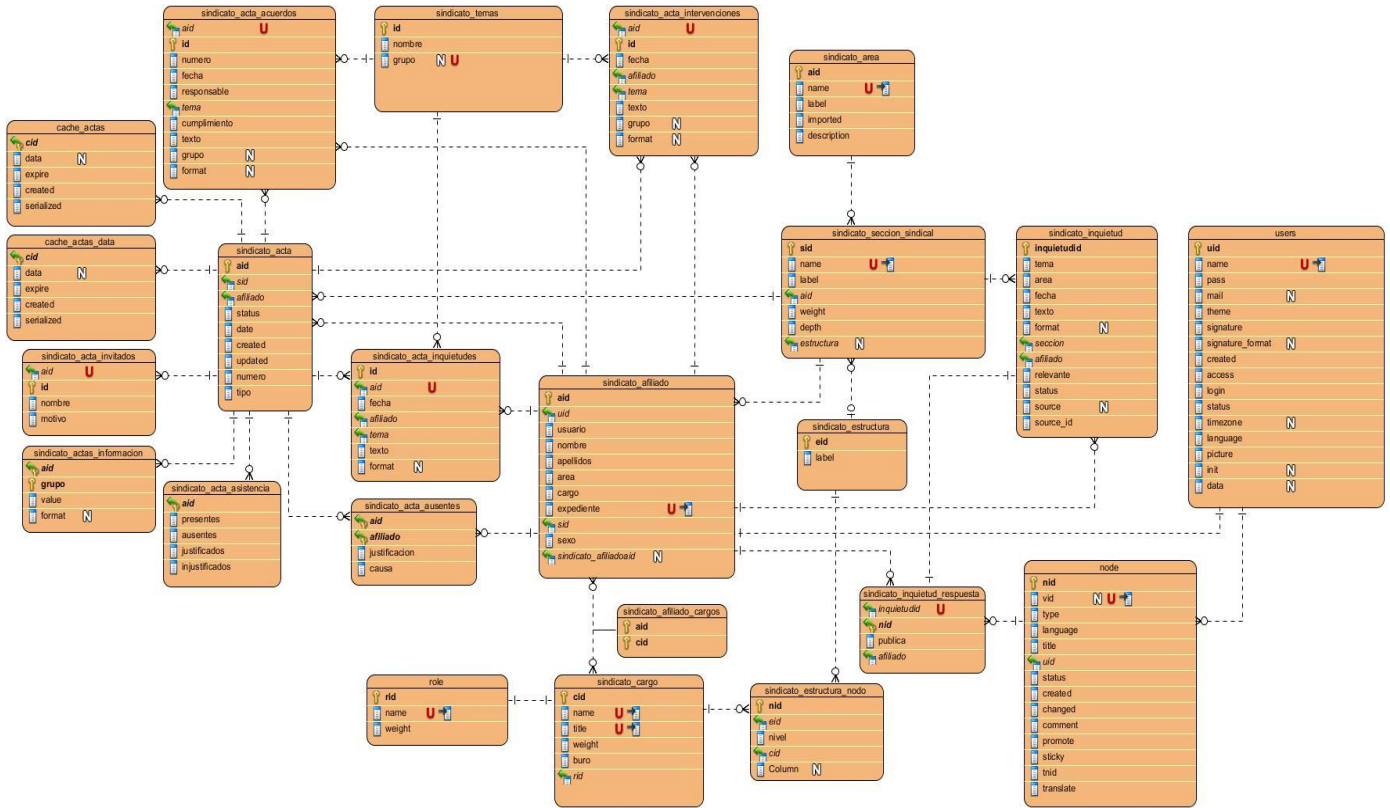


Figura 6: Modelo de datos.

2.7. Conclusiones parciales

La solución propuesta constituye una importante herramienta cuyas funcionalidades van a dar respuesta a una necesidad existente para el buró sindical de la UCI.

Se identificaron los requisitos a implementar en la solución, obteniéndose un total de 40. Las funcionalidades que componen la propuesta de solución fueron recogidas en las 17 HU diseñadas, quedando documentados de esta manera los requisitos del sistema así como sus funcionalidades.

Capítulo 2. Análisis y Diseño

Además se determinaron los patrones de asignación de responsabilidades que le proveen al diseño realizado de un bajo acoplamiento, baja complejidad de mantenimiento, baja cantidad de pruebas y una alta reutilización.

A partir de los artefactos obtenidos la solución podrá ser entendida por desarrolladores y clientes con vistas a futuras mejoras y adaptaciones a las condiciones de diversos proyectos.

La planificación del trabajo y el desarrollo por iteraciones aportaron limpieza y orden al proceso de implementación del sistema que continúa en el ciclo de vida, ahorrando en recursos y tiempo.

Capítulo 3. Implementación y Validación

Capítulo 3. Implementación y Validación del Sistema

3.1. Introducción

El presente capítulo está dedicado a la obtención de las tareas de las HU, las cuales no tendrán que se obligatoriamente entendidas por el usuario. Además se define el estándar de codificación a tener en cuenta cuando se comience con la implementación (código fuente) para lograr un mayor entendimiento técnico y facilidad al momento de identificar una clase, un atributo o un método.

Otro de los elementos abordados será la realización de las pruebas unitarias al código y la obtención de la aceptación por parte del cliente tanto de las funcionalidades del sistema como de los artefactos generados.

3.2. Implementación del sistema

Para describir las tareas llevadas a cabo en la fase de implementación se emplea un lenguaje técnico el cual no necesariamente debe ser entendible por el cliente. Dichas tareas son asignadas al equipo o programador responsable, normalmente la codificación se lleva a cabo por una pareja de programadores. Esta labor realiza con el objetivo de detallar mejor las HU, lo cual facilita el entendimiento en el proceso de implementación. Cada HU puede contener una o más tareas de ingeniería, explicando de forma general las acciones que se realizan en la misma. El resto de las tareas se encuentran en los anexos del presente documento (Ver anexo 3).

3.2.1. Tareas de las historias de usuario

Tabla 22: Tarea: Establecer arquitectura base.

Tareas de Historia de Usuario		
ID: 1	H.U: Autenticar usuario.	Iteración:1
Nombre: Establecer arquitectura base.		
Tipo: Desarrollo	Puntos estimados: 1	
Fecha inicio: 7/8/14	Fecha Fin: 8/8/14	

Capítulo 3. Implementación y Validación

Responsable: Samuel Rodríguez Carretero
Descripción: Establecer la arquitectura y diseño de clases que serán utilizadas como base en el desarrollo de la aplicación.

Tabla 23: Tarea: Diseñar la Base de datos.

Tareas de Historia de Usuario		
ID: 2	H.U: Autenticar usuario.	Iteración: 1
Nombre: Diseñar Base de Datos.		
Tipo: Desarrollo	Puntos estimados: 1	
Fecha inicio: 8/8/14	Fecha Fin: 9/8/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Establecer el diseño de la base de datos a utilizar por la aplicación.		

Tabla 24: Tarea: Establecer Prototipo de Diseño.

Tareas de Historia de Usuario		
ID: 3	H.U: Autenticar usuario.	Iteración: 1
Nombre: Establecer prototipo de diseño.		
Tipo: Desarrollo	Puntos estimados: 0.5	
Fecha inicio: 9/8/14	Fecha Fin: 9/8/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Establecer el prototipo de interfaz de usuario a utilizar en la aplicación.		

Tabla 25: Tarea: Diseñar vista de autenticar.

Tareas de Historia de Usuario		
ID: 4	H.U: Autenticar usuario.	Iteración: 1
Nombre: Diseñar vista de autenticar.		
Tipo: Desarrollo	Puntos estimados: 3	
Fecha inicio: 9/8/14	Fecha Fin: 12/8/14	

Capítulo 3. Implementación y Validación

Responsable: Samuel Rodríguez Carretero
Descripción: Desarrollar la interfaz de usuario de la primera vista de la aplicación, centrándose en el formulario de autenticación.

Tabla 26: Tarea: Mapear Base de Datos.

Tareas de Historia de Usuario		
ID: 5	H.U: Autenticar usuario.	Iteración: 1
Nombre: Mapear Base de Datos.		
Tipo: Desarrollo	Puntos estimados: 2	
Fecha inicio: 12/8/14	Fecha Fin: 14/8/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Crear las clases de persistencia que servirán para el intercambio de información con la Base de Datos.		

Tabla 27: Tarea: Conectar autenticación con Base de Datos.

Tareas de Historia de Usuario		
ID: 6	H.U: Autenticar usuario.	Iteración: 1
Nombre: Conectar autenticación con Base de Datos.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 14/8/14	Fecha Fin: 18/8/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista de autenticación con la Base de Datos y validar los datos del formulario.		

Capítulo 3. Implementación y Validación

Para obtener una mayor organización del sistema, se realiza una estructura en carpetas de cada uno de los módulos desarrollados. La ubicación de estos se encuentra en el directorio sites/all/modules/custom (ver Figura 7).

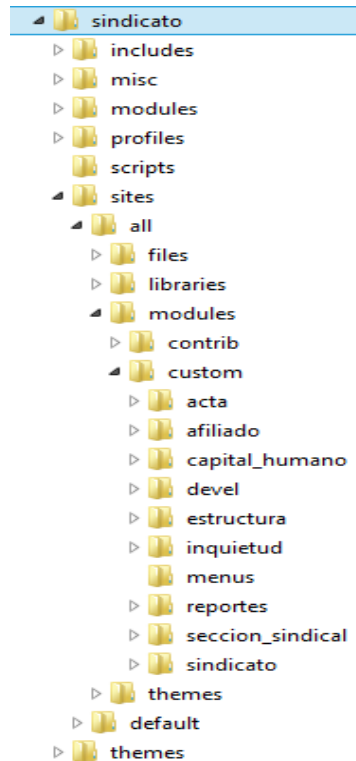


Figura 7: Ubicación de los módulos en la estructura de archivos de Drupal.

3.3. Estándares de codificación

Los estándares de codificación establecen las pautas de programación enfocadas en la estructura y apariencia física del código para facilitar la legibilidad, comprensión y mantenimiento del mismo. Dichos estándares definen la nomenclatura de las variables, objetos, métodos y funciones y su principal objetivo es llevar a cabo la realización de buenas prácticas y recomendaciones de diseño para lograr mayores niveles de calidad en los productos de software.

A continuación se describe la nomenclatura para las clases, objetos, funciones y variables utilizadas durante la implementación del sistema.

Capítulo 3. Implementación y Validación

3.3.1. Nomenclatura de las clases

Los nombres de las clases comienzan con la primera letra en mayúsculas, si el nombre es compuesto se separan las palabras con el carácter guión bajo “_” y la primera letra de la palabra siguiente se escribe en minúsculas.

```
2
3 /**
4  * Ejemplo de la utilización del patrón 'Experto en Información'
5  * que utiliza la clase Acta del módulo 'Acta'
6  */
7 class Acta extends Entity {
8
9     public $invitados;
10    public $intervenciones;
11    public $acuerdos;
12    public $inquietudes;
```

Figura 8. Nomenclatura de las clases.

3.3.2. Nomenclatura de variables

Los nombres de las variables y los objetos se escriben con la primera letra en minúsculas y si el nombre es compuesto se separa cada palabra utilizando el guión bajo. De igual forma las funcionalidades la primera letra se escribe en minúsculas y si el nombre es compuesto se separan las palabras con el guión bajo.

```
7 function acta_ajax_form($js, $form_id, $acta, $seccion_sindical) {
8     // Primero, obtenemos la configuración definida
9     // para el tipo de formulario que se desea crear
10    $form = acta_ajax_forms($form_id);
11    if (empty($form)) {
12        // Si no se encontró un formulario válido dado
13        // por el identificador $form_id entonces no
14        // se puede continuar con el algoritmo
15        return MENU_NOT_FOUND;
16    }
17    // Procedemos a obtener los parámetros dinámicamente
18    // que se deben pasar al formulario que se está creando
19    // y los guardamos en la variable $args
20    ...
21    // Construimos el estado del formulario que se está creando
22    // y que es necesario para poder mostrarlo a través de Drupal
23    $form_state = acta_build_form_state($js, $form_id, $acta, $args);
```

Figura 9. Nomenclatura de variables y funcionalidades.

Capítulo 3. Implementación y Validación

3.4. Pruebas

Dentro de las buenas prácticas de la metodología XP, se encuentra la de llevar a cabo un Desarrollo Guiado por Pruebas (TDD por su siglas en inglés), pues de esta manera se reduce el número de errores no detectados, así como el tiempo entre la introducción de estos en el sistema y su descubrimiento. Es por esta razón que el sistema en desarrollo está siendo probado constantemente. La metodología XP divide las pruebas en 2 grupos fundamentales, las pruebas unitarias y las pruebas de aceptación.

3.4.1. Pruebas unitarias

Las pruebas unitarias o test unitarios son los más importantes para el desarrollador. Cada prueba es un paso de avance en el camino de la implementación del software y cada una de ellas debe ser: atómica, independiente, inocua y rápida. Para conseguir cumplir estos requisitos, una prueba unitaria aísla la parte del programa que necesita ejecutar, de tal manera que el resto está inactivo durante la ejecución (47). Es válido señalar que la refactorización es una de las actividades que conforma la fase de pruebas en XP. Dicha actividad tiene como objetivo remover la duplicidad, mejorar la legibilidad, así como simplificar y hacer más flexible la codificación, en aras de facilitar los cambios futuros.

3.4.2. Pruebas de aceptación

Las pruebas de aceptación o test de aceptación permiten comprobar que el software cumple con un requisito. Una funcionalidad escrita con el lenguaje del cliente pero que puede ser ejecutada por la máquina. Estas pruebas son creadas a partir de las HU. Las mismas son el punto de partida del desarrollo en cada iteración (47). Las pruebas de aceptación permiten además comprobar que la funcionalidad que se está probando sea la esperada por el cliente. Este tipo de pruebas funcionan como una caja negra pues cada una de ellas representa una salida esperada del sistema, donde es responsabilidad del cliente verificar la corrección de las pruebas y tomar decisiones acerca de las mismas. A continuación se detallan algunas de las pruebas realizadas a la aplicación:

Tabla 28: Prueba de aceptación H.U1_P.A1.

Prueba de aceptación	
Código: H.U1_P.A1	H.U: Autenticar usuario

Capítulo 3. Implementación y Validación

Nombre: Identificar usuario autenticado		
Descripción: Según los datos entrados por el usuario obtener el nombre de usuario procesado		
Condiciones de Ejecución: El sistema almacena durante el proceso de entrega o recibo, los datos del usuario autenticado.		
Entrada / Pasos de ejecución: El usuario llena los datos para autenticarse, para de esta manera tener acceso al sistema, los datos entrados por el mismo son procesados y almacenados durante todo el proceso.		
Resultados esperados:	Escenario positivo: Datos correctos	Escenario negativo: Datos incorrectos
	Es procesado y almacenado correctamente el usuario del técnico.	No puede ser almacenado el usuario del técnico pues los datos fueron llenados incorrectamente, señalando el sistema los campos con borde rojo.
Evaluación: Satisfactoria		

Tabla 29: Prueba de aceptación H.U4_P.A1.

Prueba de aceptación		
Código: H.U4_P.A1	H.U: Gestionar afiliado	
Nombre: Gestionar afiliado		
Descripción: El usuario que interactúa con el sistema registra, modifica, busca o elimina los datos de un afiliado.		
Condiciones de Ejecución: A medida que va siendo modificada la información del afiliado, esta se va modificando en la base de datos, para no tener que esperar al final del proceso.		
Entrada / Pasos de ejecución: El usuario va insertando o modificando los datos que conforman la información de un afiliado, el usuario actualiza la información de un afiliado, el usuario elimina la información de un afiliado.		
Resultados	Escenario positivo: Datos correctos	Escenario negativo: Datos incorrectos

Capítulo 3. Implementación y Validación

esperados:	Es procesado y almacenado correctamente la información entrada por el usuario y se termina la operación.	La información no puede ser almacenada porque contiene datos erróneos y se muestra un mensaje de error.
Evaluación: Satisfactoria		

Tabla 30: Prueba de aceptación H.U6_P.A1.

Prueba de aceptación		
Código: H.U6_P.A1	H.U: Gestionar afiliados de una SS.	
Nombre: Gestionar Afiliado.		
Descripción: El usuario que interactúa con el sistema asigna afiliados a una SS, obtiene el listado de los afiliados de una SS en particular, modifica los datos, muestra los datos y elimina los datos de un afiliado de una SS en particular.		
Condiciones de Ejecución: A medida que va siendo modificada la información del afiliado, esta se va modificando en la base de datos.		
Entrada / Pasos de ejecución: El usuario va asignando afiliados a una SS. Además puede modificar, mostrar o eliminar los datos de un afiliado, también se puede eliminar un afiliado de una SS.		
Resultados esperados:	Escenario positivo: Datos correctos	Escenario negativo: Datos incorrectos
	Es procesado y almacenado correctamente la información entrada por el usuario y se termina la operación.	La información no puede ser almacenada porque contiene datos erróneos y se muestra un mensaje de error.
Evaluación: Satisfactoria		

Tabla 31: Prueba de aceptación H.U7_P.A1.

Prueba de aceptación	
Código: H.U7_P.A1	H.U: Gestionar acta
Nombre: Gestionar acta	

Capítulo 3. Implementación y Validación

Descripción: El usuario que interactúa con el sistema puede crear acta de reunión, modificar los datos de un acta de reunión, mostrar los datos de un acta de reunión, exportar a PDF un acta de reunión. Listar actas de reuniones y eliminar un acta de reunión.		
Condiciones de Ejecución: A medida que la información de las actas de reunión va siendo modificada, esta se va modificando en la base de datos.		
Entrada / Pasos de ejecución: El usuario va entrando en el sistema los datos asociados a las actas de reunión de una SS en particular, esta información puede ser modificada, listada o eliminada, además las actas pueden ser exportadas a formato PDF para su posterior impresión.		
Resultados esperados:	Escenario positivo: Datos correctos	Escenario negativo: Datos incorrectos
	Es procesado y almacenado correctamente la información entrada por el usuario y se termina la operación.	La información no puede ser almacenada porque contiene datos erróneos y se muestra un mensaje de error.
Evaluación: Satisfactoria		

3.4.3. Registro de no conformidades

Uno de los detalles a no pasar por alto al momento de realizar las pruebas son las no conformidades, las cuales se traducen en los errores encontrados detectados en cada iteración de pruebas. Al final de cada iteración se le muestra al cliente una versión funcional del software, de forma tal que pueda detectar aquellas no conformidades que serán corregidas al inicio de la siguiente iteración. La presente investigación está dividida en 4 iteraciones, a continuación se listan las no conformidades encontradas en cada una de ellas:

Tabla 32: Registro de no conformidades.

No conformidades	
Iteración:	
Primera	<ol style="list-style-type: none"> 1. Errores de falta de ortografía en los mensajes de la pantalla de autenticación. 2. Se detiene el indicador de carga. 3. No se especifica en el mensaje mostrado al usuario los errores que tiene el formulario al autenticarse.

Capítulo 3. Implementación y Validación

	4. El nombre del usuario acepta caracteres numéricos.
Segunda	1. Retardo en la carga de las pestañas. 2. Pérdida de datos. 3. Mezcla de datos. 4. Inconsistencia en mensajes de error.
Tercera	1. Error en la sincronización de las áreas. 2. Al ocurrir un error en la generación de un acta en formato PDF, el sistema no muestra mensaje. 3. Errores de concordancia en los mensajes.
Cuarta	1. No se detecta ninguna no conformidad.

Como parte de la metodología XP, las no conformidades encontradas en cada iteración son las primeras tareas a resolver de la iteración siguiente, siendo el cliente el encargado de ordenarlas por prioridad. Algunas de ellas al no ser críticas son arrastradas a la siguiente iteración. Llevando a cabo este proceso, se logran minimizar los niveles de aceptación de errores. De esta manera quedaron resueltas las no conformidades detectadas en la aplicación desarrollada.

3.5. Conclusiones parciales

En el presente capítulo se han obtenido los artefactos que propone la metodología usada en la investigación. Entre ellos se encuentran las 32 tareas de HU o tareas de implementación que ayudaron a detallar más ampliamente las HU creadas y además para facilitar el entendimiento de los desarrolladores. Se obtuvo además un código fuente legible y estandarizado por el estándar definido, lo que logró un mayor entendimiento técnico y facilidad al momento de identificar una clase, un atributo o un método.

Las tareas de implementación llevadas a cabo sirvieron como soporte organizativo en desarrollo de la codificación del software propuesto.

Las pruebas realizadas permitieron determinar en tiempo los errores introducidos en la aplicación. Una de las ventajas que provee XP al realizar las pruebas, es la posibilidad de que una vez mostrado el avance de la aplicación al cliente, este puede identificar las inconformidades existentes las cuales el equipo de desarrollo corrige de inmediato.

Conclusiones Generales

Una vez concluido el desarrollo del presente trabajo y analizados los resultados obtenidos, se arriban a las siguientes conclusiones:

- ❖ Se logró desarrollar un sistema de gestión de información para el sindicato de trabajadores de la UCI.
- ❖ Se logró elaborar la fundamentación teórica de la investigación, a través de la cual se demostró que portales y sistemas analizados no cumplían con todos los requisitos funcionales necesarios para el sistema.
- ❖ Se logró realizar el análisis y diseño de la propuesta de solución, donde se generaron los artefactos siguientes: listado de requisitos funcionales, historias de usuario, tarjetas CRC y modelo de datos.
- ❖ Se logró implementar la solución informática teniendo en cuenta las tareas de implementación, así como validar la solución mediante la realización de pruebas unitarias y de aceptación.

Recomendaciones

Como resultado del proceso de investigación y realización de la presente investigación se sugieren aspectos que serían importantes a tener en cuenta para el futuro perfeccionamiento del sistema:

- ❖ Implementar un nuevo módulo para gestionar las finanzas y la emulación entre SS de la Universidad de las Ciencias Informáticas.
- ❖ Mantener una constante interacción con los usuarios de la aplicación que propicie una retroalimentación, dando margen a la corrección de posibles errores que no hayan sido detectados.
- ❖ Extender el uso del sistema a otras entidades donde funcione el órgano sindical.

Referencias Bibliográficas

Referencias Bibliográficas

1. **WordReference.** Gestión. [En línea] [Citado el: 12 de Febrero de 2014.] <http://www.wordreference.com/definicion/gesti%C3%B3n>.
2. **Concepto de.** Concepto de gestión. [En línea] [Citado el: 10 de Febrero de 2014.] <http://concepto.de/concepto-de-gestion/>.
3. **Real Academia.** Real Academia Española. [En línea] 2014. <http://lema.rae.es/drae/?val=informaci%C3%B3n>.
4. **SISTEMAS DE INFORMACIÓN Y TOMA DE DECISIONES.** [En línea] <http://www.isecuniv.edu.mx/documentos/CLASE%203%20SI.pdf>.
5. *Gestión de información, gestión del conocimiento y gestión de la calidad en las organizaciones.* **Quiroga, Lourdes Aja.** 5, Ciudad de La Habana : Ciencias Médicas, octubre de 2002, SciELO, Vol. 10.
6. **Dans, Enrique.** Information Management. *Reflexiones sobre las tecnologías de la información.* [En línea] <http://informationmanagement.wordpress.com/category/gestion/gestion-de-la-informacion/>.
7. **Díaz Pérez, Maidelyn, Contreras, Yimian de Liz y Rivero Amador, Soleidys.** *Características de los sistemas de información que permiten la gestión oportuna de la información y el conocimiento institucional.* Ciudad Habana : ACIMED v.20, 2009. SciELO. ISSN 1561-2880.
8. **VERGARA, Gonzalo.** Mejora tu Gestión. [En línea] 2009. [Citado el: 12 de Febrero de 2014.] <http://mejoratugestion.com/mejora-tu-gestion/que-es-un-sistema-de-gestion/>.
9. **The.British.Standards.Institution.** BSI. [En línea] [Citado el: 10 de Febrero de 2014.] <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion>.
10. **Integra-Visión.** La importancia de los sistemas de gestión. [En línea] [Citado el: 20 de Enero de 2014.] <http://veloz-informacioncontable.com/blog-3/la-importancia-de-los-sistemas-de-gestion.html>.
11. **MasterMagazine.** Definición de Web. [En línea] [Citado el: 25 de Enero de 2014.] <http://www.mastermagazine.info/termino/7216.php>.
12. **BuenasTareas.** Definición y tipos de aplicaciones web. [En línea] [Citado el: 9 de Diciembre de 2013.] <http://www.buenastareas.com/ensayos/Definici%C3%B3n-y-Tipos-De-Aplicaciones-Web/317130.html>.
13. **AddinTech.** Soluciones creativas y personalizadas, funcionales y de entretenimiento. [En línea] [Citado el: 19 de Diciembre de 2013.] <http://www.addintech.com/index.php?producto=99>.
14. **Unión General de Trabajadores.** [En línea] www.ugt.es..
15. **CUT.** [En línea] <http://www.cutchile.cl/>.
16. **Cuba Sindical.** [En línea] <http://www.cubasindical.cu/news.php>.

Referencias Bibliográficas

17. **SNTECD**. [En línea] <http://www.sindicatoeducacion.cu/>.
18. **Meriño Menadier, Dennis**. *"Sistema de Gestión del Sindicato Nacional de Administración Pública"*. s.l. : Universidad de las Ciencias Informáticas, 2007.
19. **Medina Soca, Yoalys Miriam y Muñoz Delgado, Yamisleidys**. *"Análisis yDiseño del Proceso de Funcionamiento para la Sección Sindical Vicerrectoría Primera en la UCI"*. s.l. : Universidad de las Ciencias Informáticas, 2009.
20. **Sotolongo Arguelles, Yuniór**. *Sistema de gestión de Información Facultad 8. Módulo para la Gestión de la UJC y el Sindicato*. s.l. : Universidad de las Ciencias Informáticas, 2007.
21. **GutiérrezDíaz, Ayerim y Hernández moreno, jeorquis**. *Análisis, diseño e implementación del sitio web "Sección Sindical Facultad No. 8"*. s.l. : Universidad de las Ciencias Informáticas , 2008.
22. **Carralero Peña, María Elena y García Santana, Daylenis**. *Sistema de Gestión y Control de Información para la Sección Sindical de la facultad 1*. s.l. : Universidad de las Ciencias Informáticas , 2009.
23. **Moya Alcalá, Daymaris**. *Portal Web para la CTC de la Facultad 6*. s.l. : Universidad de las Ciencias Informáticas, 2011.
24. **Carrillo P., Isaías, Pérez G., Rodrigo y Rodríguez M., Aureliano D**. Metodología de Desarrollo del Software. [En línea] [Citado el: 10 de Febrero de 2014.] <http://tecnicaytecnologiasc.wikispaces.com/file/view/Metodologias%20de%20desarrollo.pdf/409623082/Metodologias%20de%20desarrollo.pdf>.
25. **Canós, José Hilario y Penadés, M. Carmen**. Metodologías ágiles en el desarrollo del software. [En línea] [Citado el: 10 de Febrero de 2014.] <http://issi.dsic.upv.es/archives/f-1069167248521/actas.pdf>.
26. **EUI-FI**. Introducción a Herramientas CASE y System Architect. [En línea] [Citado el: 10 de Febrero de 2014.] http://users.dsic.upv.es/asignaturas/eui/mtp/doc-practicas/intro_case_SA.pdf.
27. **FreeDownloadManager**. Visual Paradigm for UML. [En línea] [Citado el: 10 de Febrero de 2014.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.
28. **Nieto, Tito Fernando Ale**. Introducción al HTML5 2010. [En línea] [Citado el: 15 de Octubre de 2014.] <http://www.monografias.com/trabajos89/introduccion-al-html-5/introduccion-al-html-5.shtml>.
29. **Pérez, Javier Eguíluz**. LibrosWeb: . *Introducción a CSS 2007*. [En línea] http://www.librosweb.es/css/capitulo_1/breve_historia_de_css.html.
30. **PHP**. Manual de PHP. [En línea] [Citado el: 5 de Febrero de 2014.] <http://www.php.net/manual/es/intro-whatish.php>.

Referencias Bibliográficas

31. **ProgramaciónWeb.Net.** ¿Que se puede decir del PHP? [En línea] [Citado el: 8 de Febrero de 2014.] <http://www.programacionweb.net/articulos/articulo/que-se-puede-decir-php/>.
32. **Gil Rodríguez, F.** *Experto en Drupal 7. Nivel Avanzado.* Forcontu : s.n., 2011. ISBN: 978-84-939410-2-4.
33. **Alfonso, X.** Introducción a los Sistemas de Gestión de Contenido. 2008. [En línea]
34. **Tomlinson, Todd y Vandyk, John K.** *Pro Drupal 7 Development, 2010.*
35. **Centro de apoyo tecnológico a emprendedores.** *Estudio de los sistemas gestión de contenido web, 2012.*
36. **NetBeans.** Bienvenido a NetBeans. [En línea] [Citado el: 10 de Febrero de 2014.] https://netbeans.org/index_es.html.
37. **OpenSuse.** Apache. [En línea] [Citado el: 16 de Febrero de 2014.] <http://es.opensuse.org/Apache>.
38. **Cavsi.** Computer Audio Video Systems Integrator. [En línea] [Citado el: 10 de Febrero de 2014.] <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.
39. **Scribid.** Manual del usuario de PostgreSQL. [En línea] [Citado el: 8 de Febrero de 2014.] <http://es.scribd.com/doc/5703210/Manual-del-usuario-de-PostgreSQL>.
40. **Kent Beck, Martin Fowler.** *Planificando la programación extrema.* Traducido de: Planning Extreme Programming.
41. **Penadés, Patricio Letelier y M^a Carmen.** *Métodologías ágiles para el desarrollo de software: Extreme Programming.* Valencia : Universidad Politécnica de Valencia, 2013.
42. **Michael Dobson, Joe Luttrell.** LESSONS FROM HISTORY. *Functional versus NonFunctional Requirements and Testing.* [En línea] Febrero de 2013. <http://www.lessons-from-history.com/node/83>.
43. **Fowler, Martin.** Ha muerto el diseño? 2000. *Traducido de: Is design dead?* [En línea] <http://www.martinfowler.com/articles/designDead.html>.
44. **Erick Gamma, Richard Helm, Ralph Jonhson, Jonh Vlissides.** *Patrones de Diseño: elementos de software orientados a objetos reutilizables.* [trad.] Acebal, C. F. Addison Wesley, 313 p. Traducido de: Design Patterns: Elements of Reusable Object-Oriented Software.
45. **Grosso, Andrés.** Prácticas de Software: Patrones Grasp 2011. *Última actualización: [21 de Marzo de 2012].* [En línea] 2012. <http://www.practicadesoftware.com.ar/2011/03/patrones-grasp/>.
46. **Prezi.** Tarjetas CRC. [En línea] 2013. [Citado el: 2 de Marzo de 2014.] <http://prezi.com/9hc9opeav71b/tarjetas-crc/>.
47. **Jurado, Carlos Blé.** *Diseño ágil con TDD.* 2010. 298 p. ISBN 978-1445264714.

Anexo 1

Tabla 33: HU 2: Gestionar Áreas de las SS.

Historia de Usuario		
ID: 5	Nombre: Gestionar Areas de SS.	
Descripción: Permite insertar, listar, modificar y eliminar la información relacionada las áreas de las SS.		
Estimación aproximada: 10	Prioridad: Alta	H.U original: 2
Observaciones:		

Tabla 34: HU 3: Gestionar Secciones Sindicales.

Historia de Usuario		
ID: 6	Nombre: Gestionar Secciones Sindicales.	
Descripción: Permite insertar, listar, modificar, buscar y eliminar la información relacionada con la SS en la UCI, además permite definir el buró sindical partiendo de una SS.		
Estimación aproximada: 10	Prioridad: Alta	H.U original: 3
Observaciones:		

Tabla 35: HU 5: Asignar roles a la cuenta de un Afiliado.

Historia de Usuario		
ID: 7	Nombre: Asignar roles a la cuenta de un Afiliado.	
Descripción: Permite asignar roles y permisos a la cuenta de un afiliado, estableciendo mediante estos el nivel de acceso y las acciones que este podrá realizar en el sistema.		
Estimación aproximada: 5	Prioridad: Media	H.U original: 5
Observaciones:		

Tabla 36: HU 8: Gestionar las inquietudes de una sección sindical.

Historia de Usuario		
ID: 8	Nombre: Gestionar las inquietudes de una sección sindical.	
Descripción: Permite insertar, listar y eliminar las inquietudes obtenidas de las actas de reuniones de una sección sindical, así como elevar una inquietud al buró sindical.		
Estimación aproximada: 10	Prioridad: Media	H.U original: 8
Observaciones:		

Tabla 37: HU 9: Gestionar las respuestas a inquietudes de una sección sindical.

Historia de Usuario

ID: 9	Nombre: Gestionar las respuestas a inquietudes de una sección sindical.	
Descripción: Permite asignar respuesta a una inquietud, indicar si una inquietud ha sido respondida, editar los datos de la respuesta a una inquietud y eliminar una respuesta.		
Estimación aproximada: 8	Prioridad: Media	H.U original: 9
Observaciones:		

Tabla 38: HU 10: Especificar la plantilla de la estructura organizativa de las secciones sindicales.

Historia de Usuario		
ID: 10	Nombre: Especificar la plantilla de la estructura organizativa de las secciones sindicales.	
Descripción: Permite establecer y especificar la plantilla de la estructura organizativa de las SS en la UCI.		
Estimación aproximada: 15	Prioridad: Baja	H.U original: 10
Observaciones:		

Tabla 39: HU 11: Modificar la plantilla de la estructura organizativa de las secciones sindicales.

Historia de Usuario		
ID: 11	Nombre: Modificar la plantilla de la estructura organizativa de las secciones sindicales.	
Descripción: Permite modificar la plantilla de la estructura organizativa de las SS previamente establecida.		
Estimación aproximada: 12	Prioridad: Baja	H.U original: 11
Observaciones:		

Tabla 40: HU 12: Mostrar la estructura organizativa de una sección sindical.

Historia de Usuario		
ID: 12	Nombre: Mostrar la estructura organizativa de una sección sindical.	
Descripción: Muestra la estructura organizativa de la SS seleccionada por el usuario que interactúa con el sistema.		
Estimación aproximada: 10	Prioridad: Baja	H.U original: 12
Observaciones:		

Tabla 41: HU 13: Mostrar la estructura organizativa del buró sindical.

Historia de Usuario

ID: 13	Nombre: Mostrar la estructura organizativa del buró sindical.	
Descripción: Muestra la estructura organizativa del buró sindical.		
Estimación aproximada: 10	Prioridad: Baja	H.U original: 13
Observaciones:		

Tabla 42: HU 14: Mostrar reportes de las secciones sindicales.

Historia de Usuario		
ID: 14	Nombre: Mostrar reportes de las secciones sindicales.	
Descripción: Muestra los diferentes reportes generados de las distintas SS de la Universidad.		
Estimación aproximada: 8	Prioridad: Baja	H.U original: 14
Observaciones:		

Tabla 43: HU 15: Mostrar reportes de los afiliados de las secciones sindicales.

Historia de Usuario		
ID: 15	Nombre: Mostrar reportes de los afiliados de las secciones sindicales.	
Descripción: Muestra los diferentes reportes realizados por los afiliados de las distintas SS de la Universidad.		
Estimación aproximada: 8	Prioridad: Baja	H.U original: 15
Observaciones:		

Tabla 44: HU 16: Mostrar reportes de las actas de reuniones de las secciones sindicales.

Historia de Usuario		
ID: 16	Nombre: Mostrar reportes de las actas de reuniones de las secciones sindicales.	
Descripción: Muestra los diferentes reportes recogidos en las actas de las reuniones de las SS.		
Estimación aproximada: 8	Prioridad: Baja	H.U original: 16
Observaciones:		

Tabla 45: HU 17: Mostrar reportes de las inquietudes y respuestas de las secciones sindicales.

Historia de Usuario		
ID: 17	Nombre: Mostrar reportes de las inquietudes y respuestas de las secciones sindicales.	
Descripción: Muestra los reportes con las inquietudes y las respuestas recogidas en cada SS.		
Estimación aproximada: 8	Prioridad: Baja	H.U original: 17

Observaciones:

Anexo 2

Tabla 46: Tarjeta CRC: Clase Autenticar.

Clase: Autenticar	
Responsabilidades:	Colaboraciones:
1. Sincronización de los usuarios del dominio, con su cuenta relacionada de afiliado en el sitio.	<ul style="list-style-type: none"> - Módulo base de Drupal (User). - Módulo base de Drupal (LDAP).

Tabla 47: Tarjeta CRC: Clase Afiliado.

Clase: Afiliado	
Responsabilidades:	Colaboraciones:
1. Maneja la información asociada a un afiliado, así como los permisos y roles del mismo.	<ul style="list-style-type: none"> - <i>Entity</i> (Clase nativa del API de Drupal). - <i>EntityController</i> (Clase nativa del API de Drupal).

Tabla 48: Tarjeta CRC: Clase Área.

Clase: Área	
Responsabilidades:	Colaboraciones:
1. Crear un área.	<ul style="list-style-type: none"> - <i>Entity</i> (Clase nativa del API de Drupal). - <i>EntityController</i> (Clase nativa del API de Drupal).

Tabla 49: Tarjeta CRC: Clase SecciónSindical.

Clase: SecciónSindical	
Responsabilidades:	Colaboraciones:
1. Crear una sección sindical.	<ul style="list-style-type: none"> - <i>Entity</i> (Clase nativa del API de Drupal). - <i>EntityController</i> (Clase nativa del API de Drupal).

Tabla 50: Tarjeta CRC: Clase Estructura.

Clase: Estructura	
Responsabilidades:	Colaboraciones:
1. Crear la estructura de una SS por cargos.	<ul style="list-style-type: none"> - <i>Entity</i> (Clase nativa del API de Drupal).

	- <i>EntityController</i> (Clase nativa del API de Drupal).
--	---

Tabla 51: Tarjeta CRC: Clase Inquietud.

Clase: Inquietud	
Responsabilidades:	Colaboraciones:
1. Crear una inquietud a partir de las actas de reunión.	- <i>Entity</i> (Clase nativa del API de Drupal). - <i>EntityController</i> (Clase nativa del API de Drupal).

Tabla 52: Tarjeta CRC: Clase Inquietud.

Clase: Respuesta	
Responsabilidades:	Colaboraciones:
1. Crear la respuesta a una inquietud recogida.	- <i>Node</i> (Clase nativa del API de Drupal).

Anexo 3

Tabla 53: Tarea: Diseñar vista de gestionar áreas de SS.

Tareas de Historia de Usuario		
ID: 7	H.U: Gestionar áreas de SS.	Iteración: 1
Nombre: Diseñar la vista de gestionar áreas de SS.		
Tipo: Desarrollo	Puntos estimados: 3	
Fecha inicio: 19/8/14	Fecha Fin: 22/8/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Desarrollar la interfaz de usuario de la primera vista de la estructura de las áreas de las SS, centrándose en el formulario de gestión de áreas de SS de la aplicación.		

Tabla 54: Tarea: Conectar vista de gestionar áreas de SS con la BD.

Tareas de Historia de Usuario		
ID: 8	H.U: Gestionar áreas de SS.	Iteración: 1
Nombre: Conectar la vista de gestionar áreas de SS con la Base de Datos (BD).		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 22/8/14	Fecha Fin: 26/8/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista de gestionar áreas de las SS con la Base de Datos y validar los datos del formulario.		

Tabla 55: Tarea: Diseñar vista de gestionar SS.

Tareas de Historia de Usuario		
ID: 9	H.U: Gestionar SS.	Iteración: 1
Nombre: Diseñar la vista de gestionar SS.		
Tipo: Desarrollo	Puntos estimados: 2	
Fecha inicio: 28/8/14	Fecha Fin: 30/8/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Desarrollar la interfaz de usuario de la primera vista de la estructura de las SS, centrándose en el formulario de gestión de las SS de la aplicación.		

Tabla 56: Tarea: Conectar vista de gestionar SS con la BD.

Tareas de Historia de Usuario		
ID: 10	H.U: Gestionar SS.	Iteración: 1
Nombre: Conectar la vista de gestionar SS con la Base de Datos (BD).		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 31/8/14	Fecha Fin: 4/9/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista de gestionar las SS con la Base de Datos y validar los datos del formulario.		

Tabla 57: Tarea: Diseñar vista de gestionar la cuenta de un afiliado.

Tareas de Historia de Usuario		
ID: 11	H.U: Gestionar la cuenta de un afiliado.	Iteración: 1
Nombre: Diseñar la vista de gestionar cuentas de afiliados.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 4/9/14	Fecha Fin: 8/9/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Desarrollar la interfaz de usuario de la primera vista de la gestión de afiliados, centrándose en el formulario de gestión de la cuenta de un afiliado de la aplicación.		

Tabla 58: Tarea: Conectar la vista de gestionar la cuenta de un afiliado con la BD.

Tareas de Historia de Usuario		
ID: 12	H.U: Gestionar la cuenta de un afiliado.	Iteración: 1
Nombre: Conectar la vista de gestionar la cuenta de un afiliado con la Base de Datos.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 8/9/14	Fecha Fin: 12/9/14	

Responsable: Samuel Rodríguez Carretero
Descripción: Conectar la vista de gestionar cuenta de afiliados con la Base de Datos y validar los datos del formulario.

Tabla 59: Tarea: Diseñar vista de gestionar actas de reuniones de una SS.

Tareas de Historia de Usuario		
ID: 13	H.U: Gestionar actas de reuniones de una SS.	Iteración: 1
Nombre: Diseñar la vista para gestionar actas de reuniones de una SS.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 12/9/14	Fecha Fin: 16/9/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Desarrollar la interfaz de usuario de la primera vista de para la gestión de las actas generadas en las reuniones de una SS.		

Tabla 60: Tarea: Conectar la vista de gestionar actas de reuniones de una SS con la BD.

Tareas de Historia de Usuario		
ID: 14	H.U: Gestionar actas de reuniones de una SS.	Iteración: 1
Nombre: Conectar la vista para gestionar actas de reuniones de una SS con la BD.		
Tipo: Desarrollo	Puntos estimados: 10	
Fecha inicio: 16/9/14	Fecha Fin: 26/9/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista de gestionar actas de reuniones de una SS con la Base de Datos y validar los datos del formulario.		

Tabla 61: Tarea: Diseñar vista de gestionar afiliados de una SS.

Tareas de Historia de Usuario		
ID: 15	H.U: Gestionar afiliados de una SS.	Iteración: 2
Nombre: Diseñar la vista para gestionar los afiliados de una SS en particular.		
Tipo: Desarrollo	Puntos estimados: 2	
Fecha inicio: 26/9/14	Fecha Fin: 28/9/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Desarrollar la interfaz de usuario de la primera vista de para la gestión de los afiliados a una SS en particular.		

Tabla 62: Tarea: Conectar la vista de gestionar afiliados de una SS con la BD.

Tareas de Historia de Usuario

ID: 16	H.U: Gestionar afiliados de una SS.	Iteración: 2
Nombre: Conectar la vista para gestionar los afiliados de una SS con la BD.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 28/9/14	Fecha Fin: 2/10/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista para la gestión de los afiliados a una SS en particular con la Base de Datos y validar los datos del formulario.		

Tabla 63: Tarea: Diseñar vista de gestionar las inquietudes de una SS.

Tareas de Historia de Usuario		
ID: 17	H.U: Gestionar las inquietudes de una SS.	Iteración: 2
Nombre: Diseñar la vista para gestionar las inquietudes planteadas en una SS en particular.		
Tipo: Desarrollo	Puntos estimados: 3	
Fecha inicio: 2/10/14	Fecha Fin: 5/10/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Desarrollar la interfaz de usuario de la primera vista de para la gestión de las inquietudes planteadas en una SS en particular.		

Tabla 64: Tarea: Conectar la vista de gestionar las inquietudes de una SS con la BD.

Tareas de Historia de Usuario		
ID: 18	H.U: Gestionar las inquietudes de una SS.	Iteración: 2
Nombre: Conectar la vista para gestionar las inquietudes planteadas en una SS con la BD.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 5/10/14	Fecha Fin: 9/10/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista para la gestión de las inquietudes planteadas en una SS con la Base de Datos y validar los datos del formulario.		

Tabla 65: Tarea: Diseñar vista de gestionar las respuestas a las inquietudes de una SS.

Tareas de Historia de Usuario		
ID: 19	H.U: Gestionar las respuestas a inquietudes de una SS.	Iteración: 2
Nombre: Diseñar la vista para gestionar las respuestas a las inquietudes de una SS.		
Tipo: Desarrollo	Puntos estimados: 3	
Fecha inicio: 9/10/14	Fecha Fin: 12/10/14	

Responsable: Samuel Rodríguez Carretero
Descripción: Desarrollar la interfaz de usuario de la primera vista de para la gestión de las respuestas a las inquietudes planteadas en una SS en particular.

Tabla 66: Tarea: Conectar vista de gestionar las respuestas a las inquietudes de una SS con la BD.

Tareas de Historia de Usuario		
ID: 20	H.U: Gestionar las respuestas a inquietudes de una SS.	Iteración: 2
Nombre: Conectar la vista para gestionar las respuestas planteadas en una SS con la BD.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 12/10/14	Fecha Fin: 16/10/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista para la gestión de las respuestas a las inquietudes planteadas en una SS con la Base de Datos y validar los datos del formulario.		

Tabla 67 Tarea: Diseñar vista de gestionar plantilla de estructura organizativa de las SS.

Tareas de Historia de Usuario		
ID: 21	H.U: Gestionar la plantilla de la estructura organizativa de las SS.	Iteración: 2
Nombre: Diseñar la vista de la plantilla para gestionar la estructura organizativa de las SS.		
Tipo: Desarrollo	Puntos estimados: 3	
Fecha inicio: 16/10/14	Fecha Fin: 19/10/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Desarrollar la interfaz para gestionar la plantilla de la estructura organizativa de las SS.		

Tabla 68 Tarea: Conectar vista de la plantilla de la estructura organizativa de las SS con la BD.

Tareas de Historia de Usuario		
ID: 22	H.U: Gestionar la plantilla de la estructura organizativa de las SS.	Iteración: 2
Nombre: Conectar la vista de la estructura organizativa de una SS con la BD.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 19/10/14	Fecha Fin: 23/10/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista de gestionar la pantilla estructura organizativa de una SS con la Base de Datos.		

Tabla 69: Tarea: Diseñar la vista de la estructura organizativa del buró sindical.

Tareas de Historia de Usuario		
ID: 23	H.U: Mostrar la estructura organizativa del buró sindical.	Iteración: 3

Nombre: Diseñar la vista con la plantilla con la estructura organizativa del buró sindical.	
Tipo: Desarrollo	Puntos estimados: 3
Fecha inicio: 23/10/14	Fecha Fin: 26/10/14
Responsable: Samuel Rodríguez Carretero	
Descripción: Desarrollar la interfaz de la plantilla de la estructura organizativa del buró sindical.	

Tabla 70: Tarea: Conectar la vista de la estructura organizativa del buró sindical con la BD.

Tareas de Historia de Usuario		
ID: 24	H.U: Mostrar la estructura organizativa del buró sindical.	Iteración: 3
Nombre: Conectar la vista de la estructura organizativa del buró sindical con la BD.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 26/10/14	Fecha Fin: 30/10/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista de la estructura organizativa del buró sindical con la Base de Datos.		

Tabla 71: Tarea: Diseñar la vista de los reportes de las SS.

Tareas de Historia de Usuario		
ID: 25	H.U: Mostrar reportes de las SS.	Iteración: 3
Nombre: Diseñar la vista de los reportes de las SS.		
Tipo: Desarrollo	Puntos estimados: 3	
Fecha inicio: 30/10/14	Fecha Fin: 2/11/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Desarrollar la interfaz de la primera vista de los reportes de las SS.		

Tabla 72: Tarea: Conectar la vista para mostrar los reportes de las SS con la BD.

Tareas de Historia de Usuario		
ID: 26	H.U: Mostrar reportes de las SS.	Iteración: 3
Nombre: Mostrar reportes de las SS.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 26/10/14	Fecha Fin: 30/10/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista de reportes de las SS con la Base de Datos.		

Tabla 73: Tarea: Diseñar la vista de los reportes de los afiliados de las SS.

Tareas de Historia de Usuario		
ID: 27	H.U: Mostrar reportes de los afiliados de las SS.	Iteración: 3
Nombre: Diseñar la vista de los reportes de los afiliados de las SS.		

Tipo: Desarrollo	Puntos estimados: 3
Fecha inicio: 30/10/14	Fecha Fin: 2/11/14
Responsable: Samuel Rodríguez Carretero	
Descripción: Desarrollar la interfaz de la primera vista de los reportes los afiliados de las SS.	

Tabla 74: Tarea: Conectar la vista para mostrar los reportes de los afiliados de las SS con la BD.

Tareas de Historia de Usuario		
ID: 28	H.U: Mostrar reportes de los afiliados de las SS.	Iteración: 3
Nombre: Mostrar reportes de los afiliados de las SS.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 2/11/14	Fecha Fin: 6/11/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista de reportes de los afiliados de las SS con la Base de Datos.		

Tabla 75: Tarea: Diseñar la vista de los reportes de las actas de reuniones de las SS.

Tareas de Historia de Usuario		
ID: 29	H.U: Mostrar reportes de las actas de reuniones de las SS.	Iteración: 3
Nombre: Diseñar la vista de los reportes de las actas de reuniones de las SS.		
Tipo: Desarrollo	Puntos estimados: 3	
Fecha inicio: 6/11/14	Fecha Fin: 9/11/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Desarrollar la interfaz de la primera vista de los reportes de las actas de reuniones de las SS.		

Tabla 76: Conectar la vista para mostrar los reportes de las actas de reuniones de las SS con la BD.

Tareas de Historia de Usuario		
ID: 30	H.U: Mostrar reportes de las actas de reuniones de las SS.	Iteración: 3
Nombre: Mostrar reportes de las actas de reuniones de las SS.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 9/11/14	Fecha Fin: 13/11/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista de reportes de las actas de reuniones de las SS con la Base de Datos.		

Tabla 77: Tarea: Diseñar la vista de los reportes de las inquietudes y respuestas de las SS.

Tareas de Historia de Usuario		
ID: 31	H.U: Mostrar reportes de las inquietudes y respuestas de las SS.	Iteración: 3

Nombre: Diseñar la vista de los reportes Mostrar reportes de las inquietudes y respuestas de las SS.	
Tipo: Desarrollo	Puntos estimados: 3
Fecha inicio: 13/11/14	Fecha Fin: 16/11/14
Responsable: Samuel Rodríguez Carretero	
Descripción: Desarrollar la interfaz de la primera vista de los reportes Mostrar reportes de las inquietudes y respuestas de las SS.	

Tabla 78: Conectar la vista para mostrar los reportes de las inquietudes y respuestas de las SS con la BD.

Tareas de Historia de Usuario		
ID: 32	H.U: Mostrar reportes Mostrar reportes de las inquietudes y respuestas de las SS.	Iteración: 3
Nombre: Mostrar reportes Mostrar reportes de las inquietudes y respuestas de las SS.		
Tipo: Desarrollo	Puntos estimados: 4	
Fecha inicio: 16/11/14	Fecha Fin: 20/11/14	
Responsable: Samuel Rodríguez Carretero		
Descripción: Conectar la vista de reportes reportes de las inquietudes y respuestas de las SS con la Base de Datos.		

Anexo 4

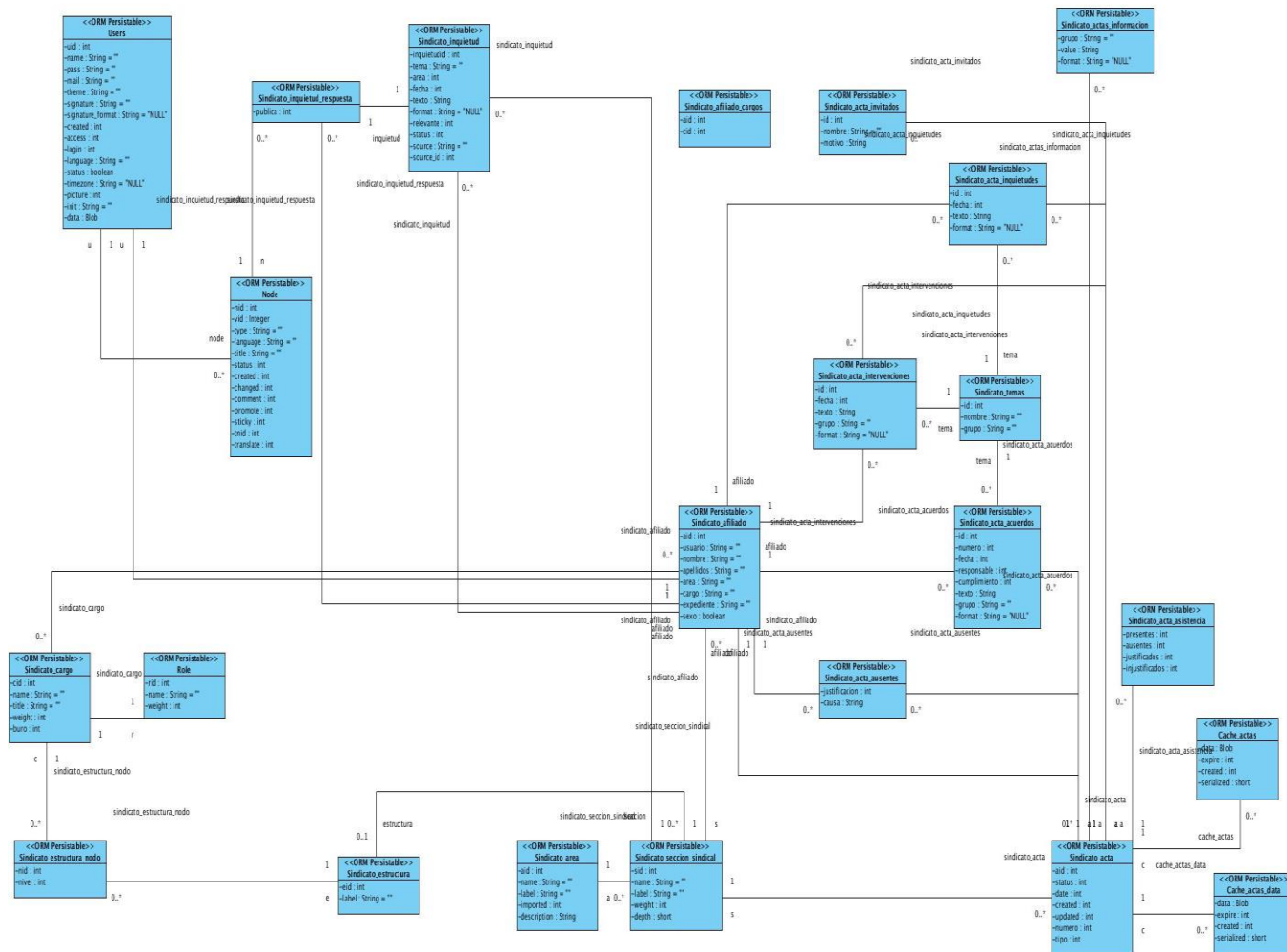


Figura 10: Diagrama de clases persistentes.