

Universidad de la Ciencias Informáticas

Facultad 3

Centro de Informatización para la Gestión de Entidades (CEIGE)



Título: Subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en la Facultad 3.

Autor: Ana Beatriz González Rodríguez

Tutor: Ing. Zénel Reyes Pérez

La Habana, julio de 2012

AGRADECIMIENTOS

A todos los que colaboraron en la realización del presente trabajo...

A mi papá, por las alegrías y atenciones que me fueron ofrecidas, por su intransigencia y enseñanzas, por su amistad, pero sobre todo por la belleza de su persona...

A mi mamá por estar en todo momento, por ser mi mejor amiga, mi consejera, mi apoyo incondicional, mi todo...

A mis tíos y tía del alma, por ser conmigo excepcionales.

A mis hermanos con los cuales siento la responsabilidad de ser su ejemplo a seguir.

A mis queridos abuelos Adelaida e Hiram que han sido, son y serán el ejemplo de mis días...

A mi Abuela Lola por el cariño, dedicación y amor...

A mi amiga, prima y hermana Laura por existir en sus labios siempre una sonrisa...

A mis bisabuelos Valentina, Jorge, Justa y Gabino por formar parte importante y especial de mi razón de ser...

A mi prima Sarahí por todo su cariño.

A Alfredo, por su amor, ternura, cariño, pasión y sobre todo por ser quien es...

A todas las personas que durante estos 5 años aportaron a mi formación profesional y personal... A mi tutor por su ayuda y el tiempo dedicado...

A mis amigas (os) y compañeras (os), esos con los que compartí momentos inolvidables en la universidad, que estuvieron presente cuando más los necesitaba y que se ganaron mi cariño, en especial a Daymara, Yisel, Susell y Ernesto por estar siempre para todo. A mi familia, por el apoyo incondicional durante estos 5 años...

A Enrique que me ofreció su ayuda cuando más la necesitaba...

A Rainer por todo el apoyo brindado a lo largo de la carrera...

A mis profesoras Teresa Morales, Mónica Cortina, Yadira Calimano, Odette Fernández y Yaili Pérez por marcar mi carrera con sus buenas prácticas profesionales...

A mis compañeros del Comité Primario, en especial a Yadian, Lisset y Yoel.

A mis compañeros del Centro Cultural en especial a Malcom y a Luis...

DEDICATORIA

A mis padres

A mis abuelos

A mis bisabuelos

A mis hermanos

DECLARACIÓN JURADA DE AUTORÍA Y AGRADECIMIENTOS

Declaro que soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Ana Beatriz González Rodríguez

Ing. Zénel Reyes Pérez

RESUMEN

Con el objetivo de incrementar el impacto de La Universidad de las Ciencias Informáticas en la informatización del país, incrementar las exportaciones y fortalecer la formación pedagógica y científica de trabajadores y estudiantes tienen lugar en el centro un grupo de misiones de cooperación internacional.

En La Facultad 3 se realizan varios procesos, entre ellos se encuentra la gestión de Trámites para la salida y visita desde y hacia el exterior. Este proceso tiene como objetivo gestionar los trámites referentes a las solicitudes de misiones de Cooperación Internacional que se realizan desde las diferentes áreas de la facultad. Actualmente no se cuenta con una solución informática que permita realizar las actividades necesarias para la gestión de trámites de cooperación internacional. No se aprovechan al máximo las tecnologías existentes, afectándose la calidad de las mismas. De ahí la necesidad de emplear nuevas variantes que agilicen el flujo de actividades y que a su vez garanticen su calidad, eficiencia y rapidez.

Por tanto, el objetivo principal del presente trabajo se centra en la realización del Subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en La Facultad 3, que gestione adecuadamente las actividades necesarias para tramitar una solicitud de misión de Cooperación Internacional.

El presente documento recoge un estudio sobre otros sistemas de trámites existentes en el ámbito nacional y en La Universidad, además se describen las características de las herramientas usadas para el desarrollo de la propuesta de solución. Se especifican las funcionalidades que brinda el sistema a sus clientes, además el diseño, implementación y prueba de la aplicación web propuesta.

Palabras Claves

Trámites, misiones de Cooperación Internacional, procesos, gestión, toma de decisiones, aplicación web.

Índice

INTRODUCCIÓN	- 9 -
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	- 14 -
1.1 Génesis y desarrollo de la cooperación internacional en Cuba y su contextualización en la universidad de las ciencias informática	- 14 -
1.2 La informatización del proceso de gestión de trámites de las salidas y visitas hacia y desde el exterior	- 18 -
1.3 Metodología, Herramientas y tecnologías utilizadas	- 20 -
1.3.1 Aplicaciones Web	- 20 -
1.3.2 Metodología de desarrollo	- 21 -
1.3.3 Notación para el Modelado de Procesos del Negocio	- 23 -
1.3.4 Gestor de Base de Datos	- 24 -
1.3.5 IDE de desarrollo	- 25 -
1.3.6Tecnologías Web.	- 26 -
1.4 Conclusiones	- 30 -
CAPÍTULO 2: ANÁLISIS Y DISEÑO	- 31 -
2.1 Introducción	- 31 -
2.2 Modelado de los procesos de negocio	- 31 -
2.3Lista de reserva del producto	- 38 -
2.4 Historia de usuarios (UH)	- 40 -
2.5 Patrones usados en el diseño	- 41 -
2.6 Tarjetas CRC	- 44 -
2.7 Modelo de datos	- 46 -
2.8 Conclusiones	- 48 -

CAPÍTULO 3: VALIDACIÓN TÉCNICA DEL SISTEMA.	- 50 -
3.1 Introducción.....	- 50 -
3.2 Tareas de ingeniería.....	- 50 -
3.2 Diagrama de componentes.....	- 53 -
3.3 Diagrama de despliegue.....	- 54 -
3.4 Pruebas.....	- 54 -
3.5 Validación de la implementación.....	- 58 -
3.6 Conclusiones.....	- 64 -
CONCLUSIONES	- 65 -
BIBLIOGRAFÍA	- 67 -

INTRODUCCIÓN

En el mundo actual, caracterizado por la globalización neoliberal y la unipolaridad, es necesario que se desarrolle una universidad que busque lo óptimo, lo más perfecto posible dentro de las cosas humanas, lo más novedoso y creativo. El campo de las Tecnologías de la Información y las Comunicaciones (TIC) constituye una acertada vía para contribuir al logro de estos propósitos

Las TIC han alcanzado auge mundial sin precedentes, desempeñando un papel importante en las estrategias de negocios y en la forma en que las entidades realizan sus procesos. Hoy más que nunca las empresas requieren de herramientas que les proporcionen control y centralización de su información con el fin de tomar mejores decisiones.

En Cuba, se han confeccionado estrategias para lograr que los conocimientos y las tecnologías de la información estén en función de garantizar la cultura digital como elemento esencial e imprescindible para el desarrollo del hombre nuevo, así mismo la industria del software y los primeros programas de informatización representan una significativa fuente de ingresos que han posibilitado la inserción del país en el mercado mundial.

Este desarrollo de las tecnologías informáticas ha estado patrocinado por las universidades evidenciado en varios trabajos investigativos relacionados con las TIC así como en la producción de software para la industria nacional y la exportación.

La Universidad de las Ciencias Informáticas (UCI) lleva a cabo el desarrollo de Software y Servicios Informáticos que tienen en cuenta la integración de los procesos de formación, investigación y producción en torno a una temática para convertirla en una rama productiva. Pueden citarse entre otros proyectos los siguientes: Sistemas Legales, Salud, Software Libre, Realidad Virtual, Teleformación, entre otros.

El subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en La Facultad 3 que se propone, posibilita gestar fuerzas de trabajo altamente preparadas para el desarrollo de software e incita además la formación de

pregrado, de postgrado, el fomento de líneas de investigación, además del desarrollo y la colaboración nacional e internacional.

En la sede tienen lugar un conjunto de misiones de ayuda internacional con el propósito de hacer mayor el número de exportaciones y fortalecer la formación pedagógica y científica del grupo de egresados y docentes, trabajadores y estudiantes y además posibilitan comercializar, explorar, promover, comprar o vender productos y/o soluciones informáticas; intercambio con instituciones extranjeras; asistencia a seminarios, conferencias, congresos en el exterior o también para la realización de cursos, diplomados, maestrías, doctorados, entre otros.

Para efectuar la entrada y la salida desde y hacia el exterior se requiere de un conjunto de trámites y de actividades vinculadas a este proceso. En La Facultad 3 se gestionan de forma manual provocando congestión y poca seguridad en la documentación. En ocasiones no se cumple con el tiempo establecido para efectuar los pasos relacionados con los trámites a efectuar, pues la cantidad de individuos que intervienen en los debates para la aprobación de una misión, dilata el tiempo de duración del proceso ya que la información y documentación que se maneja es abundante. Unido a esto los solicitantes, en su mayoría, no llenan las planillas con todos los datos correspondientes por lo que se pierde tiempo completando las mismas. No se posee un registro minucioso de los resultados que aporta la cooperación internacional. Por lo que se precisa del empleo de novedosas variantes que permitan agilizar el proceso para tramitar la entrada y salida al exterior y que haga efectiva la calidad, eficiencia y rapidez de este proceso.

Dada la **situación problemática** existente se plantea el siguiente **problema a resolver**: El manejo ineficiente de la información en el proceso actual de gestión de los trámites de las salidas y visitas hacia y desde el exterior en la facultad 3 no permite a los directivos disponer de información íntegra y confiable, conduciendo una visión parcializada en el proceso de aprobación.

Será **objeto de estudio**: Proceso de gestión de la información sobre trámites migratorios.

Campo de acción: Sistemas de gestión de trámites de salidas y visitas, hacia y desde el exterior, en La Facultad 3.

Objetivo General: Desarrollar un subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en La Facultad 3, permitiendo a sus directivos disponer de información íntegra y confiable.

Objetivos específicos:

- Realizar el marco teórico de la investigación.
- Realizar el análisis, diseño e implementación de la solución.
- Validar y evaluar técnicamente la solución.

Para dar cumplimiento a los objetivos propuestos se trazaron las siguientes **tareas investigativas:**

- Análisis conceptual de Misión de Cooperación Internacional, proceso, cultura de gestión y toma de decisiones.
- Caracterización de la situación actual del objeto de estudio en La Universidad de las Ciencias Informáticas y en La Facultad 3.
- Estudio de los sistemas existentes para la gestión de procesos de cooperación internacional en el país a fin de identificar las principales características a tener en cuenta para el desarrollo de la solución.
- Justificación del uso de las tecnologías, lenguajes y herramientas propuestas para el desarrollo del sistema.
- Realización de la captura y descripción de los requerimientos asociados a La Gestión de Trámites de las salidas y visitas hacia y desde el exterior
- Realización de la implementación del sistema.
- Validación de los resultados obtenidos.

Posibles resultados:

- Solución informática que permitirá el manejo eficiente de los datos relacionados con los trámites de salida-entrada hacia y desde el exterior en La Facultad 3.

- Integración con la plataforma tecnológica de la Facultad 3.

Para el cumplimiento de estas tareas investigativas se emplearon los siguientes métodos de investigación:

Métodos Teóricos:

- **Histórico – Lógico:** Para conocer los antecedentes, el desarrollo y la tendencia actual sobre los sistemas referentes a la cooperación internacional, la metodología de desarrollo, herramientas y lenguajes a utilizar en la solución.
- **Analítico - Sintético:** Para el procesamiento de la información en la conformación del marco teórico de la investigación y en la búsqueda e interpretación de la diversidad de argumentos que se valoran en torno al tema objeto de estudio de manera que permita proponer alternativas diferentes a las actuales.
- **Modelación:** Para modelar el negocio y lograr un mejor entendimiento del problema que se está investigando y los procesos implicados en el mismo.
- **Inductivo deductivo:** Se puso en práctica para llegar a conclusiones y aplicar las concepciones al proceso de Gestión de Trámites de las salidas y visitas hacia y desde el exterior.

Métodos Empíricos:

- **Observación:** Para observar cómo se comporta el proceso de Gestión de Trámites de las salidas y visitas hacia y desde el exterior.
- **Análisis de documentos:** Se usó para determinar las orientaciones que precisan los trámites de las salidas y vistas hacia y desde el exterior.

Técnicas

- **Entrevistas a miembros del consejo de votación:** Permitió obtener información acerca de cómo se realiza el proceso de aprobación de Trámites de las salidas y visitas hacia y desde el exterior.
- **Encuesta a individuos involucrados en el proceso:** Para conocer sus criterio acerca de los Trámites de las salidas y visitas hacia y desde el exterior.

Para una mejor comprensión de la investigación, el trabajo se estructuró de la siguiente forma:

Capítulo 1: Fundamentación Teórica.

Se describen conceptos utilizados a lo largo del documento, facilitando así la comprensión de temas posteriores. Se realiza un breve estudio de otros sistemas existentes en Cuba similares al que se pretende modelar a fin de identificar las principales características y deficiencias de los mismos. Por último se realiza un análisis de las metodologías de desarrollo de software, herramientas y lenguajes a utilizar para modelar la solución.

Capítulo 2: Análisis y Diseño

En este capítulo se establece un estudio del negocio existente en la cooperación internacional de La Facultad 3, dando paso al análisis y diseño para la propuesta de solución al problema descrito anteriormente, presentando una visión general de lo que será el sistema.

Capítulo 3: Implementación y Prueba

Se realiza la implementación y validación del sistema a través de técnicas aportadas por la metodología a seguir. En el capítulo se espera reflejar la aceptación del subsistema por parte de los principales cliente, documentando cada uno de los pasos a seguir.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En el presente capítulo se brindan los fundamentos teóricos de la investigación, se refieren los conceptos más importantes utilizados en el documento para proporcionar mejor entendimiento de la investigación. Se muestran los resultados de un estudio realizado a Sistemas de Trámites Migratorios. Se lleva a cabo el análisis de diferentes lenguajes, tecnologías y metodologías de desarrollo de software y otros aspectos importantes a emplear en el desarrollo del trabajo.

1.1 Génesis y desarrollo de la cooperación internacional en Cuba y su contextualización en la universidad de las ciencias informática.

La cooperación internacional consiste en el trabajo en comunidad llevado a cabo por parte de uno o más grupos de personas. Sin embargo, en la actualidad, la definición de cooperación internacional es más amplia, porque responde al creciente desarrollo y complejidad de las relaciones internacionales. Es así que la cooperación internacional conjuga hoy en día la ayuda o apoyo al desarrollo con otros elementos como los de promoción comercial e intereses políticos y económicos. Esto permite que se entienda la cooperación internacional como el conjunto de acciones a través de las cuales se intenta coordinar políticas o unir esfuerzos para poder alcanzar objetivos en plano el internacional. (1)

La misma es entendida como la interacción creativa entre los estados, la promoción del diálogo y el acercamiento para resolver problemas comunes a partir del entendimiento y no del enfrentamiento. La realización de la cooperación internacional en la esfera política tiene como resultado el fortalecimiento de la paz y la seguridad internacional, así como el enfoque coordinado de problemas que afronta la comunidad internacional. (1)

Los trámites necesarios para efectuar la entrada y la salida desde y hacia el exterior en La Universidad de las Ciencias Informáticas poseen un conjunto de pasos para la aprobación de las mismas que incluyen la solicitud por el interesado en recibir al visitante o el interesado en viajar. Esta solicitud puede ser realizada por un profesor

investigador, especialista, directivo u otra persona a sus jefes inmediatos, la misma es presentada por los jefes de áreas para iniciar el trámite de aprobación de esta.

Otra actividad vinculada al proceso de cooperación internacional es el análisis de la solicitud por parte del área solicitante con los representantes políticos y de masas (PCC, UJC, FEU, CTC) y se someta a aprobación de la misión a nivel de facultad en el Consejo de Dirección. Posteriormente se procede al llenado de la planilla de entrada o salida al exterior y ciertos modelos establecidos para este fin. Las planillas anteriormente llenadas se someten a aprobación por la Rectoría: de ser aprobada, la Oficina de Trámites gestiona una serie de documentos imprescindible para la entrada y la salida al exterior de los solicitantes. La duración de esta gestión incluye el tiempo transcurrido desde que se hace la solicitud hasta que esta culmina.

Sistemas de Trámites Migratorios en Cuba

El **Ministerio de la Industria Sideromecánica (SIME)** cuenta con un Sistema de Atención a Misiones que se utiliza desde el año 2000, elaborado por la empresa de Servicios Informáticos de Pinar del Río. Se desarrolló en el plazo de 4 meses sin la utilización de ninguna metodología de desarrollo, basándose solo en la interacción programador cliente y las entrevistas realizadas para conocer los requisitos. El sistema es una aplicación de escritorio realizada en Delphi, representa grandes beneficios para esta empresa y le brinda a la misma algunos servicios como la búsqueda de información de los misioneros que han viajado y de las misiones realizadas, búsqueda sobre el estado de los pasaportes, generación de los modelos oficiales de Solicitud de Trámites, Pasaporte y Visa de la Dirección de Inmigración y Extranjería, además de brindar un sistema de reportes.

El uso de este sistema ahorra tiempo y fuerza de trabajo a los responsables de tramitar misioneros del SIME pero por otro lado, desde hace varios años la empresa de Servicios Informáticos de Pinar del Río no brinda soporte a este sistema por lo que en la actualidad las necesidades de la empresa no son cubiertas totalmente por la aplicación. Una de las necesidades de este ministerio es que las empresas y grupos empresariales adjuntos al mismo utilicen una aplicación con el objetivo de tener toda la información en la misma base de datos, esta pudiera ser satisfecha si el Sistema de Atención a Misiones fuera una aplicación web. En la actualidad esta empresa realiza

una serie de trámites a extranjeros que el Sistema de Atención a Misiones no cubre por completo ya en el momento en que fue desarrollado estas gestiones no se realizaban.

(2)

Por otra parte el **Ministerio de Educación Superior (MES)** cuenta con un sistema de gestión para la cooperación internacional que opera sobre web e incluye una potente herramienta para realizar cualquier tipo de reporte. Presenta el inconveniente de que por cada cliente que hace una petición al sistema se ejecuta una aplicación diferente en el servidor, lo cual no constituye un problema a una empresa o institución pequeña y con recursos limitados pero es un inconveniente para la UCI donde puede haber un gran número de usuarios conectados simultáneamente al sistema. Por otra parte la base de datos está diseñada en Microsoft Access, gestor que no ofrece mucha seguridad en la protección de la información.

El **Ministerio de la Informática y las Comunicaciones (MIC)** cuenta con un sistema web que incluye la gestión y aprobación de solicitudes a nivel de Ministerio y procesos homólogos a los desarrollados por el Grupo de Cooperación Internacional de la Dirección de Cooperación internacional (DCI) en la UCI.

La **Universidad de las Ciencias Informáticas** cuenta con un Sistema Informatizado de Relaciones Internacionales el cual surge con el objetivo de informatizar los procesos de la DCI de la UCI. En el año 2006, la aplicación web desarrollada con este propósito, tenía disponibles las funcionalidades dedicadas a soportar los procesos del Grupo de Trámites. Su implementación se realizó utilizando como plataforma de desarrollo Visual Studio .NET 2003 y Microsoft SQLServer 2000 como gestor de base de datos. Hasta esa fecha se había llegado hasta la fase de análisis de los procesos de los dos restantes grupos de la DCI. Como resultado de un cambio en las políticas de la UCI en cuanto a la selección de herramientas para la informatización del centro hacia la migración a tecnologías libres, fue necesario rediseñar la aplicación existente, adaptándola a la nueva arquitectura y se comenzó a redefinir el trabajo realizado como paradigma de la soberanía tecnológica.

El proceso de cooperación internacional

Considerando **proceso** como el “concepto que designa el devenir o cambio como elemento fundamental de la realidad, y se opone a todo concepto de ser estático o a una sustancia fija y determinada. A veces, se identifica con flujo, procesión, corriente.”

(3)

Entonces, podemos afirmar que la cooperación internacional se concreta en el **proceso**.

La **Dirección de Cooperación Internacional (DCI)** es la estructura organizativa que se encarga de la gestión de las actividades de colaboración internacional y relaciones públicas en la UCI. Pertenece a la Vicerrectoría de Investigaciones y Cooperación Internacional; la misma está compuesta por el Grupo de Cooperación Internacional, Grupo de Relaciones Públicas y Grupo de Trámites. El primero es el encargado de gestionar la estrategia de colaboración internacional de la universidad, coordinar la participación de la UCI en organizaciones internacionales de excelencia y contribuir a la identificación y gestión de programas de cooperación internacional. Por otra parte el Grupo de Relaciones Públicas se encarga de manejar todo lo referente a la imagen institucional de la universidad así como a la promoción y divulgación de actividades de interés del centro. Conjuntamente con el Grupo de Cooperación Internacional, dirige metodológica y funcionalmente el sistema de colaboración internacional de la UCI, gestiona y conforma la información de apoyo a los intercambios inter- institucionales además de garantizar las acciones necesarias para el éxito de las misiones de la Universidad en el exterior y la satisfacción de los visitantes en Cuba. (4)

Misión de Cooperación Internacional: Es una inversión a mediano o largo plazo para alcanzar objetivos estratégicos de la organización a través de la cooperación internacional. (2)

En la universidad se encuentran definidas de la siguiente forma:

Tipo I- Oficial: consiste en el viaje al exterior de algún directivo de la universidad para intercambiar con instituciones extranjeras.

Tipo II- Comercial: Son las misiones cuyo objetivo es comercializar, explorar, promover, comprar o vender productos y/o soluciones informáticas o firmar contratos con este fin, esto puede suceder en el exterior del país o en el centro.

Tipo III- Evento: Consisten en seminarios, conferencias, congresos que se realizan en el exterior o en el centro. También se consideran eventos los concursos en los que fundamentalmente participan los estudiantes.

Tipo IV- Proyecto: Misiones que generalmente son financiadas por el proyecto de desarrollo las cuales se realizan en función de los objetivos del mismo. Estos proyectos se clasifican en tres tipos: Productivos (Misión que financia o financiará el proyecto) y Colaboración (con otros países, organismos, entidades. Ejemplo: Discapacitados, Correos de Cuba, entre otros.)

Tipo V- Estancias: Viaje al exterior de profesores por un período determinado que puede extenderse desde una semana a 6 meses por motivos de: Curso, Diplomado, Maestría, Doctorado, Post doctorado, Intercambio académico, Investigación, Profesor invitado y Entrenamiento.

1.2 La informatización del proceso de gestión de trámites de las salidas y visitas hacia y desde el exterior.

Gestión

Manejo de los fines en lo cual se demanda por parte de los directivos, imaginación, capacidad, compromiso, así como conceptualizar los propósitos para generalizar la comprensión. Poner todos los medios para conseguir un fin. (3)

Cultura de gestión

Los directivos deben poseer cultura de gestión, que no es más que un sistema de conocimientos y desarrollo de hábitos, habilidades y capacidades, sobre un conjunto de principios, métodos, procedimientos, formas y técnicas de gestión que constituyen una necesidad insoslayable para que el dirigente pueda realizar las diligencias necesarias que le permitan ejercer con eficiencia su función, siendo capaz de lograr una cabal toma de decisiones. (5)

Toma de Decisiones

Dirigir científicamente el proceso de cooperación internacional es tomar decisiones acertadas a partir de situaciones concretas y condiciones objetivas que permiten

atender al hombre colocándolo como sujeto y objeto del proceso y atender el desarrollo corriente y perspectiva del mismo. (5)

La autora precisa que la decisión de cooperación internacional es un hecho con repercusión social, es un sistema de acciones que definen en sí tanto los objetivos a alcanzar como los medios para lograrlos y las coordinaciones necesarias para ejecutarlas, encierra un aspecto objetivo y otro subjetivo e influye tanto en el objeto como en el sujeto, basada en la mejor selección de variantes o alternativas de acuerdo a un criterio de valoración.

No se concibe el proceso de cooperación internacional sin estar presente una toma de decisiones acertada ya que están implícitas en todo el ciclo (desde la solicitud hasta la aprobación) de ahí que se exprese que las decisiones en este sentido son importantes porque:

- 1.- La selección adecuada del solicitante, es tomar decisiones.
- 2.- Contribuyen al éxito del Sistema de Cooperación Internacional.
- 3.- Garantiza la adecuada orientación y organización de los colectivos de trabajo.

Toda decisión en cuanto a cooperación internacional implica un riesgo que puede ser reducido si se tienen en cuenta las siguientes exigencias fundamentales:

- Las propuestas deben estar correctamente fundamentadas.
- Deben estar en correspondencia con la importancia, magnitud y necesidad real de la colaboración.
- Tener carácter legal.
- Deben ser un instrumento para llevar a la práctica la política del PCC y el Estado.
- Deben ser oportunas y ejecutables.
- Deben tener función movilizadora, organizadora, creadora y estimuladora.
- Ser flexibles (que permitan hacer variaciones y adaptaciones al ejecutarse).

La toma de decisiones de cooperación internacional presenta faltas al manifestarse las siguientes causas que constituyen fuentes de errores.

- La incertidumbre.- cuando se presentan situaciones nuevas, sin antecedentes.

- La información es incompleta.
- La emotividad.- Cuando se tiene la tendencia a decidir por motivo de ira o simpatía.
- Objetivo equivocado.- Se decide sobre un problema erróneo y no por el legítimo.
- Análisis no sistemático. No tomar en cuenta todas las condiciones y factores.
- El problema no se ha presentado plenamente.
- La causa real no se ha identificado.

Consecuencias:

- Aumenta el riesgo de errores.
- Aumenta la posibilidad de tomar decisiones contradictorias.

Se proponen como reglas que conducen a alcanzar una buena decisión en la cooperación internacional.

- Obtener, procesar, conservar y evaluar información.
- Visión y predicción de probables consecuencias.
- Identificar riesgos y saber valorarlos.
- Comprometer a los ejecutores desde patrones éticos para hacer más confiable la información.

1.3 Metodología, Herramientas y tecnologías utilizadas

1.3.1 Aplicaciones Web

Las aplicaciones web nos deleitan con su forma de instalación y distribución lo cual nos lleva a elegir por el desarrollo de una aplicación de este tipo para gestionar trámites para las salidas-entradas hacia y desde el Exterior en la facultad 3. Se fundamenta esta deliberación ya que facilita ser accedida por web ya sea internet o intranet y el acceso a ésta puede ser público o restringido dependiendo de quien haga la visita. Las actualizaciones y el mantenimiento de las aplicaciones son completamente transparente a los usuarios de la aplicación ya que no es necesario instalar ni actualizar software en sus computadoras. Una de sus mayores ventajas es que posee

la cualidad de ser multiplataforma. Esto garantiza que pueda ejecutarse en cualquier Sistema Operativo que posea un navegador web, además permite la interacción de varios clientes conectados a una única aplicación instalada en un servidor por lo cual sus clientes verán los resultados inmediatamente después de actualizar su sistema.

1.3.2 Metodología de desarrollo

Dentro de la ingeniería de software la metodología de desarrollo, se encarga de elaborar estrategias de desarrollo de software a fin de promover prácticas centradas en las personas o equipos. Su objetivo es convertir el desarrollo de software en un proceso formal, con resultados predecibles que permitan obtener un producto de alta calidad, que satisfaga las necesidades y expectativas del cliente. (6)

Existen las Metodologías Tradicionales y las Metodologías Ágiles. Las primeras intentan conseguir un objetivo común por medio de orden y documentación mientras que las llamadas Ágiles tratan de mejorar la calidad del software por medio de una comunicación directa e inmediata.

Haciendo un análisis de las metodologías existentes y más utilizadas a nivel mundial se define como Metodologías Ágiles: Extreme Programming (XP) y SCRUM. Entre las Metodologías Tradicionales: Rational Unified Process (RUP) y Microsoft Solution Framework (MSF). Seguidamente se caracteriza las metodologías apropiadas según el estudio realizado a fin de seleccionar la más adecuada que se ajuste a las necesidades existentes.

Rational Unified Process

RUP fue creada por el mismo grupo de expertos que crearon UML, Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1998. El objetivo que se perseguía con esta metodología era producir software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos (7), esta metodología concibió desde sus inicios el uso de UML como lenguaje de modelado.

Extreme Programming (XP)

Desarrollada por Kent Beck, está 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. XP está guiada por una rápida programación y se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, en la reutilización de código, en la realización de pruebas a los principales procesos con el objetivo de tratar de obtener los posibles errores futuros, conocido como pruebas unitarias, en la simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. (7)

SCRUM

SCRUM define un proceso empírico, iterativo e incremental de desarrollo que intenta obtener ventajas respecto a los procesos definidos (cascada, espiral, prototipos, etc.) mediante la aceptación de la naturaleza caótica del desarrollo de software, y la utilización de prácticas tendientes a manejar lo impredecible y el riesgo a niveles aceptables. (8)

Modelo de desarrollo adoptado

Después del estudio en profundidad de cada una de las metodologías se acierta en utilizar una metodología ágil dada por la unión de las dos metodologías anteriormente estudiadas. SXP proporciona todo lo necesario para la investigación, siendo una metodología compuesta por lo mejor de XP y SCRUM, esto es un método adaptativo de gestión de proyectos que se basa en los principios ágiles.

SXP está especialmente indicada para proyectos de pequeños equipos de trabajo, rápido cambio de requisitos o requisitos imprecisos, muy cambiantes, donde existe un alto riesgo técnico y se orienta a una entrega rápida de resultados y una alta flexibilidad.

Lenguaje Unificado de Modelado (UML).

UML (Unified Modeling Lenguaje) o Lenguaje Unificado de Modelado es un lenguaje basado en una notación gráfica la cual permite: especificar, construir, visualizar y

documentar los objetos de un sistema programado. UML se quiere convertir en un lenguaje estándar con el que sea posible modelar todos los componentes del proceso de desarrollo de aplicaciones, ha sido ampliamente aceptado debido al prestigio de sus creadores. (9)

De forma general las principales características son:

- Lenguaje unificado para la modelación de sistemas.
- Tecnología orientada a objetos.
- El cliente participa en todas las etapas del proyecto.

Herramientas de Modelado

Existen varias herramientas para el modelado las llamadas herramientas CASE (Computer Aided Software Engineering) o Ingeniería de Software Asistida por Computadora. Tienen como objetivo aumentar la productividad en el desarrollo de un software mitigando los costes en términos de tiempo.

Visual Paradigm for UML

Es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Visual Paradigm es una herramienta creada para soportar el ciclo de vida completo del proceso de desarrollo del software a través de la representación de todo tipo de diagramas entre otras opciones. Constituye una herramienta de software libre de gran utilidad para el analista. (10)

1.3.3 Notación para el Modelado de Procesos del Negocio

La Notación para el Modelado de Procesos del Negocio (BPMN: Business Process Modeling Notation) es un estándar de modelado de procesos de negocio donde se presentan gráficamente las diferentes etapas del proceso del mismo. La notación ha

sido diseñada específicamente para coordinar la secuencia de procesos y los mensajes que fluyen entre los diferentes procesos participantes. Es muy utilizado a nivel mundial y considerado un estándar avanzado. BPMN ayuda a definir, documentar y hacer mucho más eficientes los procesos del negocio. Su principal objetivo en la investigación es facilitar el entendimiento de las colaboraciones y transacciones del negocio de cooperación internacional de la facultad 3. (11)

1.3.4 Gestor de Base de Datos

Sistema Gestor o Manejador de Bases de Datos (SGBD) es un conjunto de programas que permite a los usuarios crear y mantener una BD, por lo tanto, el SGBD es un software de propósito general que facilita el proceso de definir, construir y manipular la BD para diversas aplicaciones. (12)

PostgreSQL 9.1

Es un sistema de gestión de bases de datos, objeto-relacional (ORDBMS), Está considerado como el sistema de base de datos de código abierto más avanzado del mundo, publicado bajo la licencia BSD. Brinda beneficios de código abierto y estabilidad. El sistema que se desea desarrollar requiere la gestión de altos volúmenes de información, por lo que se necesita un sistema que maneje eficientemente las transacciones y la persistencia de los datos de manera segura. (13)

A continuación se enumeran las principales características y por lo que fue elegido este gestor de bases de datos:

- Implementación del estándar SQL92/SQL99.
- Incorpora una estructura de datos de arreglos.
- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, etc.

PgAdmin III.

PgAdmin III es una aplicación gráfica para gestionar el gestor de bases de datos PostgreSQL, siendo la más completa y popular con licencia Open Source. Está escrita en C++ usando la librería gráfica multiplataforma wxWidgets, lo que permite que se pueda usar en Linux, FreeBSD, Solaris, Mac OS X y Windows. Es capaz de gestionar

versiones a partir de PostgreSQL 7.3 ejecutándose en cualquier plataforma. Incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor, un agente para lanzar scripts programados, soporte para el motor de replicación Slony-I y mucho más. (14)

1.3.5 IDE de desarrollo.

Un entorno de desarrollo integrado o IDE (acrónimo en inglés de integrated development environment), es un programa informático compuesto por un conjunto de herramientas de programación, puede dedicarse en exclusiva a un sólo lenguaje de programación o bien, poder utilizarse para varios. Entre las principales herramientas de trabajo se encuentran Eclipse y NetBeans esta última seleccionada para el desarrollo de la aplicación.

NetBeans.

NetBeans IDE es un entorno de desarrollo, una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java; pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el NetBeans IDE. NetBeans IDE es un producto libre y gratuito sin restricciones de uso. (15)

Características principales.

- El proyecto NetBeans ofrece una versión del IDE a la medida para el desarrollo en PHP de sitios web que abarcan una variedad de secuencias de comandos y lenguajes de marcado.
- El IDE NetBeans respalda plenamente el desarrollo iterativo, por lo que los proyectos desarrollados en PHP siguen los patrones clásicos conocidos para los desarrolladores web.
- El editor de PHP entiende espacios de nombres y definiciones de tipo variable en los comentarios que mejora la finalización de código y navegación hipervínculo.

1.3.6Tecnologías Web.

Las tecnologías web son un conjunto de herramientas que facilitan lograr mejores resultados a la hora del desarrollo de un sitio web.

JavaScript

Es un lenguaje de programación que permite a los desarrolladores crear acciones en sus páginas web, puede ser utilizado por profesionales y para quienes se inician en el desarrollo y diseño de sitios web. No requiere de compilación ya que el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos. (16)

Hojas de estilo en cascada (CSS).

Las hojas de estilo en cascada (Cascading Style Sheets), es la tecnología desarrollada por el World Wide Web Consortium (W3C) con el fin de separar la estructura de la presentación, permite crear páginas web de una manera más exacta, gracias a las CSS el desarrollador es mucho más dueño de los resultados finales de la página, pudiendo hacer muchas cosas que no se podía hacer utilizando solamente HTML, como fuentes, colores, márgenes, líneas, altura, anchura, imágenes de fondo, posicionamiento avanzado y muchos otros temas. (17)

Entre los beneficios concretos de CSS se encuentran:

- Control de la presentación de muchos documentos desde una única hoja de estilo.
- Control más preciso de la presentación.
- Numerosas técnicas avanzadas y sofisticadas.

Lenguaje de Marcas de Hipertexto (HTML).

El Lenguaje de Marcas de Hipertexto (Hyper Text Markup Language), no es más que un conjunto de etiquetas o comandos, complementados en la mayoría de los casos por extensiones que permiten dar formato a un archivo, con el objetivo básico de crear un documento que pueda ser visualizado en ambiente Internet en forma de Página Web y que esta, además, pueda, por medio de dichas etiquetas, tener la estructura o forma

deseada por quien la diseñó. Las etiquetas HTML tienen, por lo general, una etiqueta de apertura y una de cierre; aunque existen algunas excepciones en las que solo basta con colocar la de apertura. (17)

Frameworks

Un framework es una estructura conceptual y tecnológica de soporte definida, normalmente, con artefactos o módulos de software concretos, en base a la cual otro proyecto de software puede ser organizado y desarrollado. Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio. (18)

ExtJs Framework

Es una librería construida con JavaScript que proporciona una interfaz a las famosas librerías de Yahoo, jQuery, y Prototype + Scriptaculous, su potencia radica en la rica colección de componentes para el diseño de GUI's del lado del cliente haciendo uso extensivo de Ajax.

ExtJS es neutral al lenguaje que se use en el servidor. Siempre que el resultado se envíe a la página en el formato adecuado, ExtJS no se preocupará de lo que pase en el servidor. Proporciona un selector de nodos DOM extremadamente poderoso llamado DomQuery (puede usarse como una librería independiente, pero en el contexto de ExtJS se usará para seleccionar elementos para poder interactuar con ellos a través de la interfaz Element, contiene mucho de los métodos y propiedades de DOM que se necesitará proporcionando una interfaz conveniente, unificada y multinavegador). Entre los componentes que esta librería ofrece se encuentran cuadros de diálogo, menús, tablas editables, layouts, paneles, pestañas y todo lo necesario para construir atractivos desarrollos al estilo de Web 2.0. (19)

Ventajas:

- La orientación a objetos intensa hará modular todos los scripts.
- El diseño está completamente separado de la funcionalidad.

- Funciones comunes muy fáciles de implementar como validación, combo boxes editables, ventanas arrastrables (con minimizar y maximizar), grillas editables.

Symfony framework

El framework Symfony ha sido un proyecto de Código Abierto por más de tres años y se ha convertido en uno de los framework PHP más populares gracias a sus excelentes características y gran documentación. Esto ha sido una gran tradición desde sus inicios. Symfony es además el framework más documentado del mundo, ya que cuenta con miles de páginas de documentación distribuidas en varios libros gratuitos y decenas de tutoriales. La elección del framework es primordial para lograr una integración con el sistema SO3 (Plataforma Informativa de la facultad 3).

Ventajas

- Funciona con todas las bases de datos comunes (MySQL, PostgreSQL, SQLite, Oracle, MS SQL Server)
- Compatible solamente con PHP 5 desde hace años, para asegurar el mayor rendimiento y acceso a las características más avanzadas de PHP
- Basado en la premisa de "convenir en vez de configurar", en la que el desarrollador solo debe configurar aquello que no es convencional

Mapeado de objetos relacionales. (ORM)

Un ORM o (Object Relation Mapper) es una técnica de programación que permite convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional, es decir, las tablas de la base de datos pasan a ser clases y los registros objetos que se pueden manejar con facilidad. (20)

Doctrine

Doctrine es un mapeador de objetos relacionales (ORM) para PHP 5.2.3+ que se sienta encima de una potente capa de abstracción de bases de datos (dbal). Una de sus principales características es la opción de escribir las consultas de base de datos en un objeto de propiedad orientada dialecto SQL llamada Doctrine Query Language (DQL) que no es más que el lenguaje que utiliza Doctrine para ejecutar sus consultas.

Propel

Propel es un ORM para PHP (Pre-procesador Hipertexto) que facilita la labor de desarrollo de aplicaciones web, gracias a la capa que transforma el tratamiento de la BD mediante objetos, con la que se puede recuperar, insertar y modificar datos.

ORM adoptado

Después de analizar profundamente cada una de las características que ofrece propel y doctrine, se decide optar por doctrine ya que el mismo es el ORM por defecto de Symfony y posee todo el soporte de apoyo de Sensio Labs empresa desarrolladora del framework. Por otra parte soporta dos tipos de herencia: Simple, donde todas las clases tienen las mismas columnas, y la herencia de agregación, donde almacena un valor adicional en la tabla, permitiendo la instanciación automática del tipo de modelo correcto cuando se realiza una consulta.

Hipertext Preprocesor (PHP)

PHP es el acrónimo de Hipertext Preprocesor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Se escribe dentro del código HTML, lo que lo hace realmente fácil de utilizar. Es independiente de plataforma, puesto que existe un módulo de PHP para casi cualquier servidor web.

Esto hace que cualquier sistema pueda ser compatible con el lenguaje y es una ventaja primordial. Algunas de las más importantes capacidades de PHP son: compatibilidad con las bases de datos más comunes, como MySQL, mSQL, Oracle, Informix, y ODBC, por ejemplo. Incluye funciones para el envío de correo electrónico, upload de archivos, crear dinámicamente en el servidor imágenes en formato GIF, incluso animadas y una lista interminable de utilidades adicionales. (21)

Servidor Web

Un servidor web o servidor HTTP es un programa informático que procesa una aplicación del lado del servidor realizando conexiones bidireccionales y síncronas o asíncronas con el cliente generando o cediendo una respuesta en cualquier lenguaje o Aplicación del lado del cliente. Entre los más conocidos y utilizados se encuentran

Apache, Microsoft IIS y Google GFE. Siendo el apache uno de los más reconocidos mundialmente por seguridad y poderío.

Apache

Apache es el servidor web hecho por excelencia, su configuración, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. Entre sus principales características se encuentran: (20)

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Es una tecnología gratuita de código fuente abierta. Esto le da una transparencia al software de manera que si se desea ver lo que está instalando como servidor, se puede saber, sin ningún secreto o puertas traseras.
- Es un servidor altamente configurable de diseño modular.

1.4 Conclusiones

En el presente capítulo a través de la investigación del estado del arte se logra esclarecer aún más los objetivos de la investigación con el fin de darle respuesta a la situación problemática existente. Mediante el estudio realizado las tecnologías escogidas plantean el uso de herramientas libres que estén acordes con las políticas tecnológicas de la Universidad y del país. Una vez realizadas las investigaciones pertinentes, se concluye que los procesos de cooperación internacional pueden ser gestionables a través de una aplicación web brindando resultados satisfactorios y mejorando su accionar en la Facultad 3 de la Universidad de las Ciencias Informáticas.

CAPÍTULO 2: ANÁLISIS Y DISEÑO

2.1 Introducción

El presente capítulo tiene como objetivo el análisis y diseño de la solución informática del Subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en La Facultad 3. Se obtiene el modelado del negocio a fin de comprender la estructura y la dinámica de la organización donde se va a implantar el sistema a través de la descripción de los procesos de negocio identificados. Quedará definido el listado de requerimientos que será objeto de informatización, así como la descripción de los mismos y se muestran las historias de usuarios realizadas desde la perspectiva del cliente. Por último se modelan los artefactos que exige la metodología ágil SXP para la fase de diseño, la cual no requiere la presentación del sistema mediante diagramas de clases, en su lugar se usan otras técnicas como las tarjetas CRC que se exponen en este capítulo.

2.2 Modelado de los procesos de negocio.

La modelación de los procesos de negocio permite realizar una exploración del dominio del problema, con el fin de lograr comprensión por parte del equipo de desarrollo de los procesos que se realizan actualmente en la entidad y la relación que existe entre estos. De esta forma se van determinando necesidades operacionales, así como restricciones que presenta la entidad, obteniéndose finalmente un entendimiento del negocio para dar paso a la fase inicial del sistema, lográndose a través de los objetivos siguientes (22):

1. Realizar un estudio de los procesos existentes con el fin de contribuir con el principio de reutilización.
2. Detallar las características del negocio a través de la descripción de los procesos.
3. Verificar que se haya realizado un buen análisis del negocio.

Procesos de Negocio

Para comprender los procesos del negocio se realizó un previo estudio, así como entrevistas con los clientes para garantizar un mejor entendimiento con el equipo de desarrollo. Como resultado final fue elaborado los procesos de negocios del subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en la Facultad 3. El mismo se compone de los 3 subprocesos siguientes:

- Presentar propuesta de la solicitud.
- Crear planilla de la solicitud.
- Aprobar solicitud.

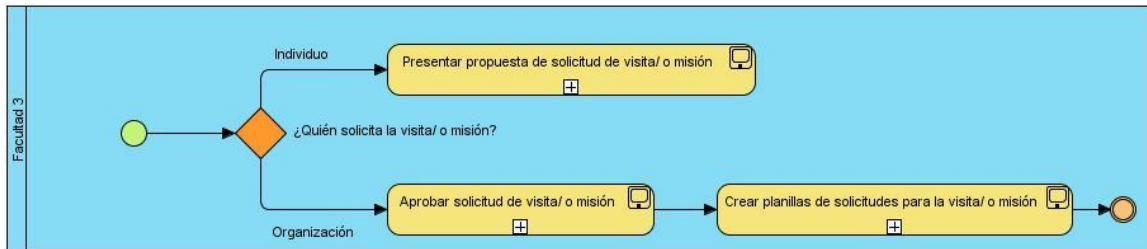


Ilustración 1 Diagrama del modelo de proceso

Descripción del proceso de negocio.

Seguidamente se muestra la descripción y diagrama de modelado de negocio del subproceso Presentar propuesta de solicitud, los dos restantes se pueden evidenciar en el expediente de proyecto.

Tabla 1. Descripción del subproceso Presentar propuesta de solicitud.

Objetivo	Presentar, validar y evaluar los datos de propuesta de misión.
Evento(s) que lo genera(n)	No aplica
Pre condiciones	No aplica
Clientes externos	No aplica

Entradas	1. Datos de consulta (datos personales del solicitante, tipo de misión que solicita y motivo por el cual realiza la solicitud).
Flujo de eventos	
Flujo básico Presentar propuesta de solicitud	
1.	Presentar datos de propuesta de solicitud: El solicitante presenta ante el Área de Administración a la cual pertenece los datos personales y otros relacionados con el tipo de misión que solicita para que sean consultados.
2.	Recepcionar datos de propuesta de solicitud: El Área de Administración recibe por parte del solicitante los datos referentes a la propuesta de solicitud.
3.	Validar datos de propuesta de solicitud: El Área de Administración a la cual pertenece el solicitante valida que la propuesta de misión presentada responde a los objetivos y metas del colectivo y cumple con los requisitos pertinentes.
4.	Presentar propuesta de solicitud: El Área de Administración, una vez realizada la validación positiva de la propuesta de misión presenta ante los Órganos de Consulta dicha propuesta para que sea evaluada.
5.	Evaluar propuesta de solicitud: Los Órganos de Consulta (Consejo de Investigación-Posgrado-Cooperación) evalúan la propuesta de misión presentada.
6.	Notificar propuesta de posible misión: Los Órganos de Consulta, tras evaluar positivamente la propuesta de misión presentada notifican al Asesor de Cooperación Internacional de la facultad sobre la solicitud de una posible misión.

7.	Revisar propuesta de misión: El Asesor de Cooperación Internacional revisa que los datos de la propuesta de solicitud estén correctos.
8.	Registrar datos sobre posible misión: Una vez realizada la revisión satisfactoria de los datos de la propuesta de misión el Asesor de Cooperación Internacional registra los mismos e incluye la propuesta realizada por el solicitante en una lista de posibles misiones para próxima consulta y aprobación.
9.	Notificar sobre posible misión: El Asesor de Cooperación Internacional notifica al Área de Administración que la propuesta presentada por el solicitante se considera una posible misión.
10.	Recibir notificación de posible misión: El Área de Administración recibe la notificación de que la propuesta presentada fue validada como posible misión.
11.	Informar sobre posible misión: El Área de Administración informa al solicitante que la propuesta presentada se consideró posible misión.
12.	Recibir notificación de posible misión: El solicitante recibe la notificación de que la propuesta presentada fue validada como posible misión.
Pos-condiciones	
1.	Es validada positivamente la propuesta presentada y pasa a ser considerada posible misión.
2.	Se incluye la propuesta en el listado de posibles misiones que realiza el Asesor de Cooperación Internacional para discutir su aprobación en el Consejo de Dirección de la facultad.
Salidas	

<ol style="list-style-type: none"> 1. Datos de consulta (actualizado) 2. Listado de posibles misiones.
<p>Flujos alternos</p>
<p>Flujo alternativo 3.a La propuesta de solicitud presentada no cumple con los requisitos establecidos.</p>
<ol style="list-style-type: none"> 1. Notificar resultado de la validación: El Área de Administración notifica al solicitante que la propuesta de solicitud presentada no responde a los objetivos y metas del colectivo.
<ol style="list-style-type: none"> 2. Recibir resultado de la validación de la propuesta de misión: El solicitante recibe por parte del Área de Administración a la cual pertenece la notificación de los resultados de la validación de la propuesta de solicitud presentada.
<p>Pos-condiciones</p>
<ol style="list-style-type: none"> 1. No es validada la propuesta de solicitud presentada por el solicitante ante el Área de Administración.
<ol style="list-style-type: none"> 2. No se registra la propuesta como posible misión.
<p>Salidas</p>
<p>No aplica</p>
<p>Flujo alternativo 5.a La propuesta de misión no es evaluada positivamente por los Órganos de Consulta.</p>
<ol style="list-style-type: none"> 1. Notificar resultado de la evaluación: Los Órganos de Consulta notifican al Área de Administración que la propuesta de solicitud no fue aceptada.
<ol style="list-style-type: none"> 2. Recibir resultado de la evaluación: El Área de Administración recibe el resultado de la evaluación realizada por parte de los Órganos de Consulta sobre la

propuesta de misión.
3. Notificar resultado de la evaluación: El Área de Administración informa al solicitante que la propuesta de misión presentada no fue evaluada positivamente por los Órganos de Consulta.
4. Recibir resultado de la validación de la propuesta de solicitud: El solicitante recibe por parte del Área de Administración a la cual pertenece la notificación de los resultados de la validación de la propuesta de solicitud presentada.
Pos-condiciones
1. No es validada la propuesta de solicitud presentada por el solicitante ante los Órganos de Consulta.
2. No se registra la propuesta como posible misión.
Salidas
No aplica
Flujo alternativo 7.a El Asesor de Cooperación Internacional encuentra datos incorrectos.
1. Notificar errores en los datos: El Asesor de Cooperación Internacional notifica al Área de Administración que existen errores que debe corregir en los datos enviados de la propuesta de solicitud.
2. Recibir notificación de errores en los datos: El Área de Administración recibe la notificación de que se encontraron errores en los datos de la propuesta de solicitud presentada.
3. Corregir datos erróneos: El Área de Administración corrige los errores encontrados por parte del Asesor de Cooperación Internacional en los datos de la propuesta de solicitud.

<p>4. Enviar datos de consulta: El Área de Administración luego de corregir los errores encontrados envía nuevamente los datos de la propuesta de misión para que sean revisados por el Asesor de Cooperación Internacional.</p>
<p>5. Continúa con el paso 7 del flujo básico.</p>
<p>Pos-condiciones</p>
<p>1. Se actualizan los datos erróneos encontrados.</p>
<p>Salidas</p>
<p>1. Datos de consulta (actualizado)</p>
<p>Asuntos pendientes</p>
<p>No aplica</p>

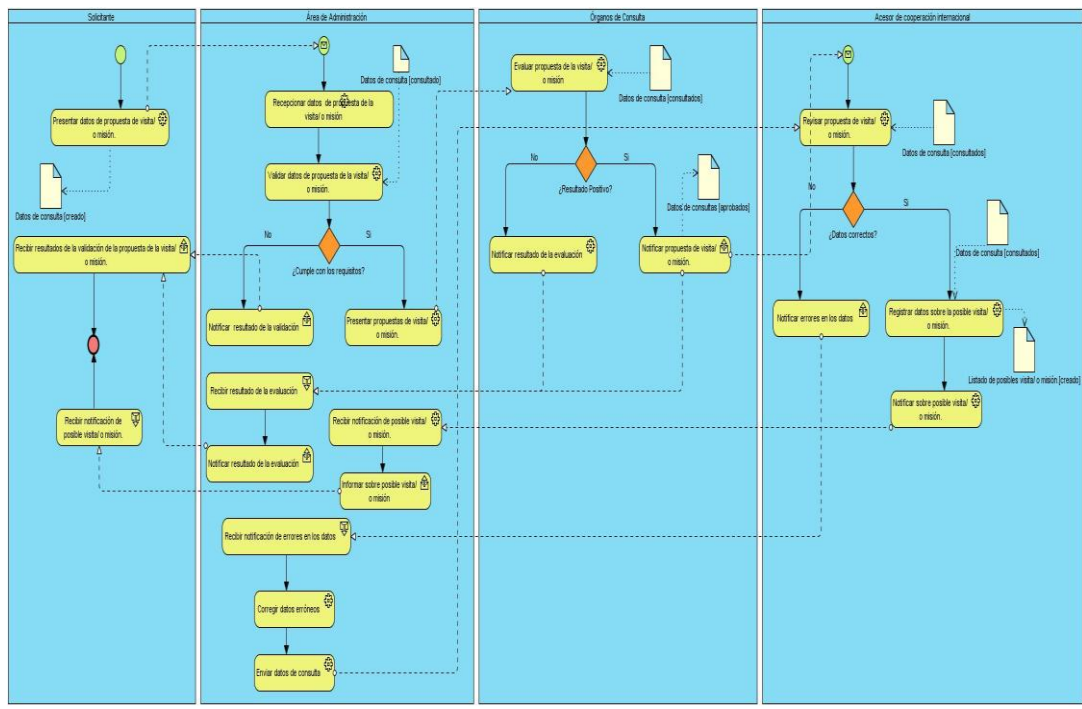


Ilustración 2 Diagrama del subproceso Presentar propuesta de solicitud

2.3 Lista de reserva del producto

El propósito de la definición de la lista de reserva de producto es especificar las condiciones o capacidades que el sistema debe cumplir y las restricciones bajo las cuales debe operar, logrando un entendimiento entre el equipo de desarrollo y el cliente, y especificando las necesidades reales de forma que cumpla sus expectativas.

Identificación y clasificación de la historia de usuarios

Requisitos funcionales.

No	Requisito	Descripción	Complejidad
1	Realizar votación	Requisito creado para realizar la votación sobre cada solicitud existente, tiene incluida la realización del veto sobre una solicitud.	Media
2	Mostrar estado de la votación	Muestra el estado de la votación por cada solicitud de cada una de las personas que tienen que votar.	Media
3	Gestionar entes que vetan	Se agrega y configura quienes son las personas encargadas de vetar en cada situación	Baja
4	Gestionar entes que votan	Se agrega y configura quienes son las personas encargadas de votar en cada situación	Baja
5	Gestionar solicitudes internas	Se gestiona la solicitud por parte de un individuo perteneciente a un área, se modifica, elimina o listan las solicitudes y sus estados.	Alta
6	Gestionar solicitudes	Se gestiona la solicitud por parte de un individuo perteneciente a un área, se modifica, elimina o listan	Alta

	externa	las solicitudes y sus estados.	
7	Revisar solicitud	La solicitud creada pasa por la revisión de las contra-partes (jefe de área, investigación, postgrado y cooperación), se proveen criterios a tener en cuenta para los votantes.	Media
8	Gestionar estado de la solicitud	Requisito creado para determinar el estado en que se encuentra una solicitud [cualquiera sea el tipo de esta] a partir del momento en que esta se encuentre.	Media
9	Gestionar contactos	Gestiona los datos de los contactos obtenidos tanto de la visita como de la salida, creando una especie de directorio internacional con datos adicionales para facilitar la gestión de clientes.	Media
10	Notificar decisiones	Se realiza la notificación a los solicitantes o jefes de área, según se configure, de la decisión tomada, respecto a la solicitud.	Baja
11	Gestionar servicios	Se realiza la gestión de los servicios a disposición de las personas que salgan o nos visiten, se realiza a través de planillas existentes por cada una de las áreas que brindan esos servicios.	Media
12	Definir reglas de votación	Se establecen y configuran las reglas por las que se efectuará y se determinará la aprobación o no de una solicitud.	Media
13	Gestionar de datos externos	Garantiza la interoperabilidad entre los sistemas de SO3 y el resto de los sistemas UCI [transportación, alimentos, nodo])	Alta
14	Gestionar	Se gestionan los datos de los presentes realizados a	Baja

	regalos	personas que nos visitan para tener constancias de los gastos y de futuros regalos.	
--	---------	---	--

Requisitos no funcionales.

Autenticación: Los usuarios se autenticarán haciendo uso del servicio LDAP de la universidad y la administración de los roles será por la aplicación correspondiente a la plataforma SO3.

Requisitos de Seguridad

Confiability: La información manejada por el sistema estará protegida de accesos no autorizados y divulgación.

Integridad: La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma será considerada igual a la fuente o autoridad de los datos.

Disponibilidad: Solo tendrán acceso al sistema el jefe cooperación internacional, el asesor principal o la persona facultada para hacer uso de sus funcionalidades.

Requisitos de Licencia: El sistema utiliza la política de software libre, todas las herramientas que se utilizan son software libre. Para la herramienta Visual Paradigm se utiliza la licencia que la Universidad posee.

2.4 Historia de usuarios (UH)

Las historias de usuario son la técnica utilizada por la metodología para especificar los requisitos del software, lo que equivaldría a los casos de uso en el proceso unificado. Son utilizadas como el único documento de requisitos que se genera en la metodología. Son escritas en lenguaje natural, sin un formato predeterminado. En el momento de implementar se deben detallar a través de la comunicación con el cliente. Son la base para las pruebas funcionales. (23)

A continuación se muestra la descripción de la historia de usuarios Realizar votación que se obtuvo como resultado del análisis del sistema. Las demás se pueden apreciar en el expediente de proyecto.

Tabla 2. Historia de usuario Realizar votación

Historia de Usuario	
Número: HU_1	Nombre Historia de Usuario: Realizar votación
Modificación de Historia de Usuario Número: Ninguna	
Usuario: votante	Iteración Asignada: 1
Prioridad en Negocio: Alta	Puntos Estimados: 3 días
Riesgo en Desarrollo: Media	Puntos Reales: 3 días
<p>Descripción: Los usuarios se autentican en el sistema después de recibir la notificación enviada por correo electrónico de que deben entrar al sistema a votar. El sistema valida el permiso de administración del usuario. Muestra la interfaz realizar votación, en la cual se ejerce el voto de manera electrónica e independiente. El usuario selecciona el solicitante o los solicitantes por los cual va a votar acepta o deniega la solicitud y concluye la acción.</p>	
<p>Observaciones: En el caso de que desee ejercer el voto en otro momento el usuario selecciona la opción cancelar</p>	

2.5 Patrones usados en el diseño

El diseño fue elaborado siguiendo patrones basados en la experiencia que de manera general constituyen soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objetos.

Symfony está basado en un patrón clásico del diseño web conocido como arquitectura MVC (Modelo, Vista, Controlador), que está formado por tres niveles:

- El Modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La Vista transforma el modelo en una página web que permite al usuario interactuar con ella.
- El Controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

El principio más importante de la arquitectura MVC es la separación del código del programa en tres capas, dependiendo de su naturaleza. La lógica relacionada con los datos se incluye en el modelo, el código de la presentación en la vista y la lógica de la aplicación en el controlador. La programación se puede simplificar si se utilizan otros patrones de diseño, de esta forma, las capas del modelo, la vista y el controlador se pueden subdividir en más capas. (24)

Para la implementación de Symfony se utilizan varios patrones, situándolos en las capas de Modelo y Control que plantea el patrón arquitectónico MVC.

Patrones GRASP

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos. Los patrones GRASP que se utilizaron son los siguientes:

Experto: Dicho patrón es evidenciado en la definición de las clases de acuerdo a las funcionalidades que deben realizar a partir de la información manejada dentro del componente, como por ejemplo las clases controladoras y las del modelo. La cual gestiona la configuración de las solicitudes, será la responsable de efectuar las operaciones que conciernen a las funciones: adicionar, modificar, eliminar y actualizar las solicitudes, asumiendo toda la lógica para cada una de ellas.

Creador: En la clase Acción de cada uno de los módulos se encuentran definidas las acciones del sistema y se ejecutan cada una de ellas. En las acciones se crean los objetos de las clases que representan las entidades, evidenciando de este modo que la clase Acción es "creador" de dichas entidades.

Alta cohesión: Este patrón fue utilizado en el diseño del componente de manera general; donde se agruparon las clases en dependencia de las historias de usuarios (Gestionar solicitud, Gestionar votación) a los que se les debía dar respuesta, según la premisa de que cada clase debe implementar las operaciones que estén sobre la misma área funcional.

Symfony permite asignar responsabilidades con una alta cohesión, por ejemplo la clase Acción tiene la responsabilidad de definir las acciones para las plantillas y colabora con otras para realizar diferentes operaciones, instanciar objetos y acceder a las propiedades, es decir, está formada por diferentes funcionalidades que se encuentran estrechamente relacionadas proporcionando que el software sea flexible frente a grandes cambios.

Controlador: Todas las peticiones son manejadas por un solo controlador frontal (sfAcción), que es el punto de entrada único de toda la aplicación en un entorno determinado. Cuando el controlador frontal recibe una petición, utiliza el sistema de enrutamiento para asociar el nombre de una acción y el nombre de un módulo con la URL entrada por el usuario.

Bajo Acoplamiento: Este patrón se manifiesta en cada uno de los módulos del sistema, la clase Acción hereda solamente de sfAcción para lograr un bajo acoplamiento de clases.

Patrones GOF.

Ya que se usan varios frameworks de desarrollo y estos a su vez implementan independientemente sus propios patrones de diseños, a continuación se presenta una breve descripción de los principales patrones empleados.

Singleton (Instancia única): Garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. En el controlador frontal hay una llamada a sfContext: getInstance (). En una acción, el método getContext (), un objeto muy útil que guarda una referencia a todos los objetos del núcleo de Symfony.

Decorator (Envoltorio): Añade funcionalidad a una clase dinámicamente. El archivo layout.php, que también se denomina plantilla global, almacena el código HTML que es

común a todas las páginas de la aplicación, para no tener que repetirlo en cada página. El contenido de la plantilla se integra en el layout, o si se mira desde el otro punto de vista, el layout decora la plantilla.

2.6 Tarjetas CRC

Las **tarjetas CRC** (clase, responsabilidad y colaboración) son una metodología para el diseño de software orientado por objetos creada por Kent Beck y Ward Cunningham. Se emplean como técnica para la representación del diseño de las clases. Las mismas representan en la metodología lo que es el diagrama de clases del diseño. Son un puente de comunicación entre los diferentes participantes. La lluvia de ideas es una buena práctica para sugerir cómo rellenar las tarjetas. Permite ver las clases como algo más que repositorio de datos, sino conocer el comportamiento de cada una en un alto nivel. Constan de tres secciones, nombre de la clase, responsabilidades y colaboradores. La sección responsabilidades se listan cada una de las funciones o tareas que debe ser capaz cumplir un objeto de dicha clase reflejando en si una descripción de ella, mientras que la sección colaboradora contiene otras clases del diseño con las que interactúan. A continuación se muestra la tarjeta CRC del modulo 1 Administrador en el expediente de proyecto se pueden apreciar los otros dos módulos.

Tabla 3. Tarjeta CRC modulo 1 Administración

Módulo 1 Administración		
Clases	Descripción	Clases relacionadas
entityFilterAction	La clase contiene el método executeRead que se ocupa de leer los datos necesarios de la BD	Nente
entityVoteFilterAction	La clase contiene el método executeRead	NreglaVotacion

	que se ocupa de leer los datos necesarios de la BD	
entityVoteWriteAction	La clase contiene el método executeWrite que se ocupa de actualizar crear y eliminar	NreglaVotacion
entityWriteAction	La clase contiene el método executeWrite que se ocupa de actualizar crear y eliminar	Nente
missionFilterAction	La clase contiene el método executeRead que se ocupa de leer los datos necesarios de la BD	NtipoMision
missionWriteAction	La clase contiene el método executeWrite que se ocupa de actualizar crear y eliminar	NtipoMision
serviceFilterAction	La clase contiene el método executeRead que se ocupa de leer los datos necesarios	NtipoServicio

	de la BD	
serviceWriteAction	La clase contiene el método executeWrite que se ocupa de actualizar crear y eliminar	NtipoServicio
solicityStateFilterAction	La clase contiene el método executeRead que se ocupa de leer los datos necesarios de la BD	NestadoSolicitud
solicityTypeFilterAction	La clase contiene el método executeRead que se ocupa de leer los datos necesarios de la BD	NtipoSolicitud
voteStateFilterAction	La clase contiene el método executeRead que se ocupa de leer los datos necesarios de la BD	NestadoVoto

2.7 Modelo de datos.

Un modelo de datos es un lenguaje orientado a describir una Base de Datos. Típicamente un modelo de datos permite describir las estructuras de datos de la base, el tipo de los datos que hay en la base y la forma en que se relacionan.

El modelo de datos propuesto en la solución cuenta con un total de 27 tablas. Para su construcción se tuvo en cuenta la reducción a la mínima expresión de los campos

nulos y la persistencia de campos resúmenes para agilizar recuperaciones frecuentes de algunos datos que son complejos de calcular.

Teniendo en cuenta los requisitos Gestionar Solicitud Interna y Gestionar Solicitud Externa se crea la tabla solicitud la cual es genérica con las tablas solicitud interna y solicitud externa, en ella se guardan los datos correspondientes de las solicitudes. Para los requisitos que gestionan todo lo referente a la votación se crean las tablas voto. Por último decir que en la tabla persona se almacenan los datos del solicitante se dirige genéricamente a la tabla persona interna o externa.

Del análisis anterior se obtuvo el siguiente modelo de datos en el cual se representan las tablas obtenidas del sistema de SO3 con el color verde, las tablas de los nomencladores del subsistema con el color amarillo y el resto simboliza las tablas responsables de las principales funcionalidades del subsistema :

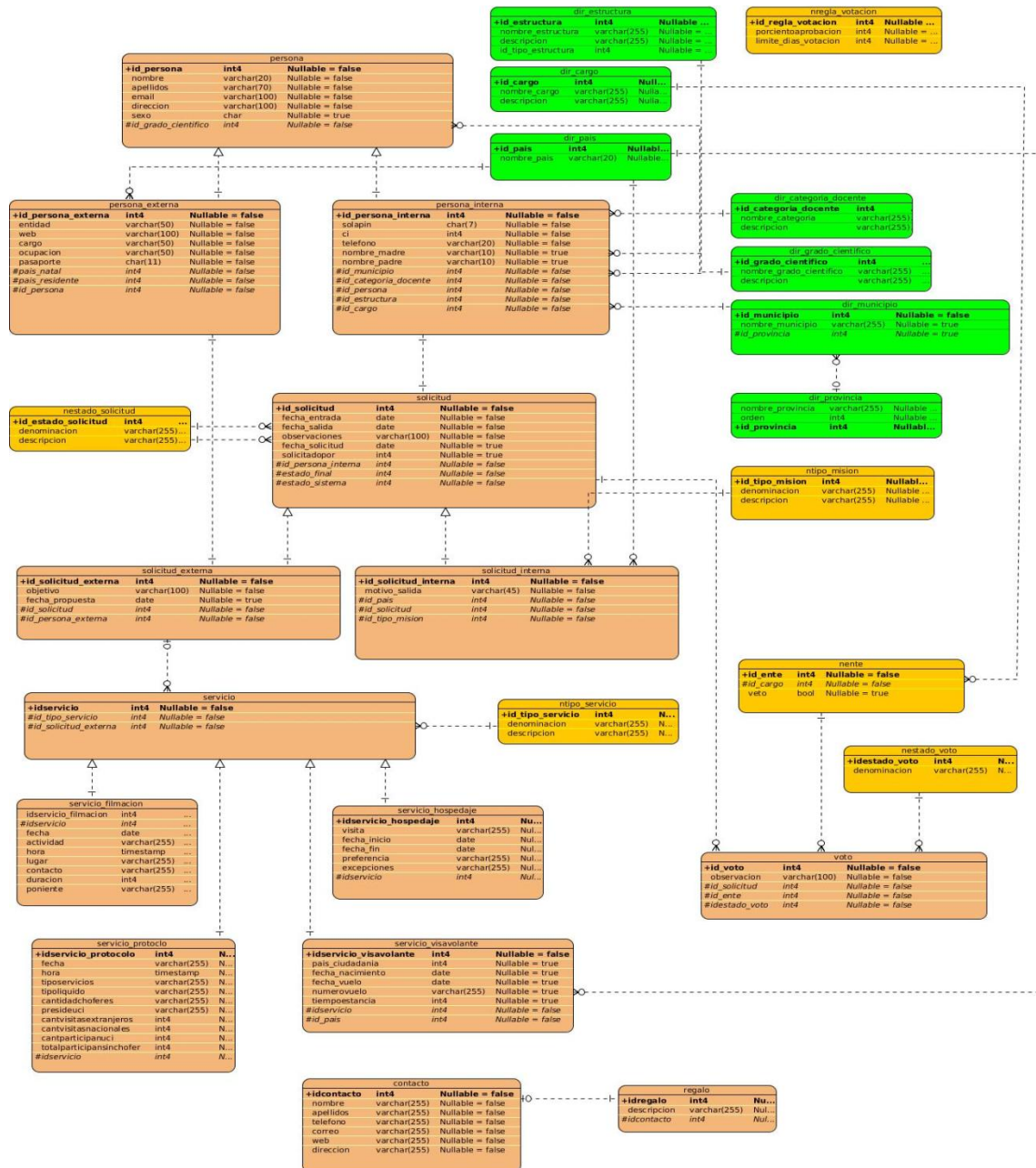


Ilustración 3. Modelo de datos

2.8 Conclusiones

En el desarrollo del capítulo se obtuvo el análisis y diseño de la solución informática para el mejor funcionamiento de la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en La Facultad 3. Logrando alcanzar una estructura

de la dinámica de organización de los procesos identificados del negocio con el que se alcanzan cada una de las fases de gestión de cooperación internacional. Con el apoyo de la metodología se establece la base para la implementación de la solución.

CAPÍTULO 3: VALIDACIÓN TÉCNICA DEL SISTEMA.

3.1 Introducción

A partir de los resultados obtenidos, se comenzará la implementación del sistema dando paso a las tareas de ingeniería, diagrama de componentes y despliegue. Para realizar la validación del diseño y la implementación se aplicarán uno de los pilares fundamentales de SXP que es el proceso de pruebas. Esta metodología divide las pruebas en dos grupos: pruebas unitarias, desarrolladas por los programadores, encargadas de verificar el código de forma automática y las pruebas de aceptación, destinadas a evaluar si al final de una iteración se obtuvo la funcionalidad requerida, además de comprobar que dicha funcionalidad sea la esperada por el cliente.

3.2 Tareas de ingeniería

Las tareas de ingeniería establecen una ficha que contiene el número identificador de la tarea, el identificador de la historia de usuario con la que está relacionada, el nombre de la tarea, el tipo (nuevo desarrollo, corrección, mejora), la fecha de inicio, su fecha de fin, el responsable y la descripción. Estas desglosan en pasos más detallados en contenido de trabajo de cada historia de usuario para facilitar el proceso.

A continuación se muestra la descripción de las tareas de ingeniería que se obtuvieron como resultado del análisis de las historias de usuario 1 y 2. Las demás se pueden apreciar en el expediente de proyecto.

Tabla 4. Tarea de ingeniería 1 de la historia de usuario 1

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 1
Nombre Tarea: Diseñar la interfaz para realizar votación	
Tipo de Tarea: Diseño	Puntos Estimados: 1 día

Fecha Inicio: 10/4/2012	Fecha Fin:10/4/2012
Programador Responsable :Ana Beatriz González Rodríguez	
Descripción: Diseñar una interfaz capaz de realizar todo el proceso de votación.	

Tabla 5. Tarea de ingeniería 2 de la historia de usuario 1

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 1
Nombre Tarea: desarrollar las funcionalidades para realizar el proceso de votación.	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1 día
Fecha Inicio: 11/4/2012	Fecha Fin:12/4/2012
Programador Responsable :Ana Beatriz González Rodríguez	
Descripción: Desarrollar las funcionalidades necesarias que sean capaz de llevar a cabo el proceso de votación.	

Tabla 6. Tarea de ingeniería 3 de la historia de usuario 1

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: 1
Nombre Tarea: Validar campos de la interfaz para realizar votación	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1 día
Fecha Inicio: 13/4/2012	Fecha Fin:14/4/2012

Programador Responsable :Ana Beatriz González Rodríguez
Descripción: Verificar que se introduzca la observación en el caso pertinente y que los datos que se inserten sean correctos.

Tabla 7. Tarea de ingeniería 1 de la historia de usuario 2

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 2
Nombre Tarea: Diseñar la interfaz para mostrar estado de la votación	
Tipo de Tarea: Diseño	Puntos Estimados: 1 día
Fecha Inicio: 12/4/2012	Fecha Fin:12/4/2012
Programador Responsable :Ana Beatriz González Rodríguez	
Descripción: Diseñar una interfaz capaz de mostrar el estado en que encuentra cada una de las solicitudes en proceso de voto.	

Tabla 8. Tarea de ingeniería 2 de la historia de usuario 2

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 2
Nombre Tarea: Validar la interfaz para mostrar estado de la votación	
Tipo de Tarea: Desarrollo	Puntos Estimados: 2 día
Fecha Inicio: 13/4/2012	Fecha Fin:14/4/2012
Programador Responsable :Ana Beatriz González Rodríguez	

Descripción: valida todo lo referente con la interfaz que muestra el estado de los votos por cada solicitud.

3.2 Diagrama de componentes.

Un diagrama de componentes representa los componentes, sus interfaces y las relaciones de los componentes con las interfaces que utilizan. A continuación se presenta el diagrama de componentes del Subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en La Facultad 3. (25)

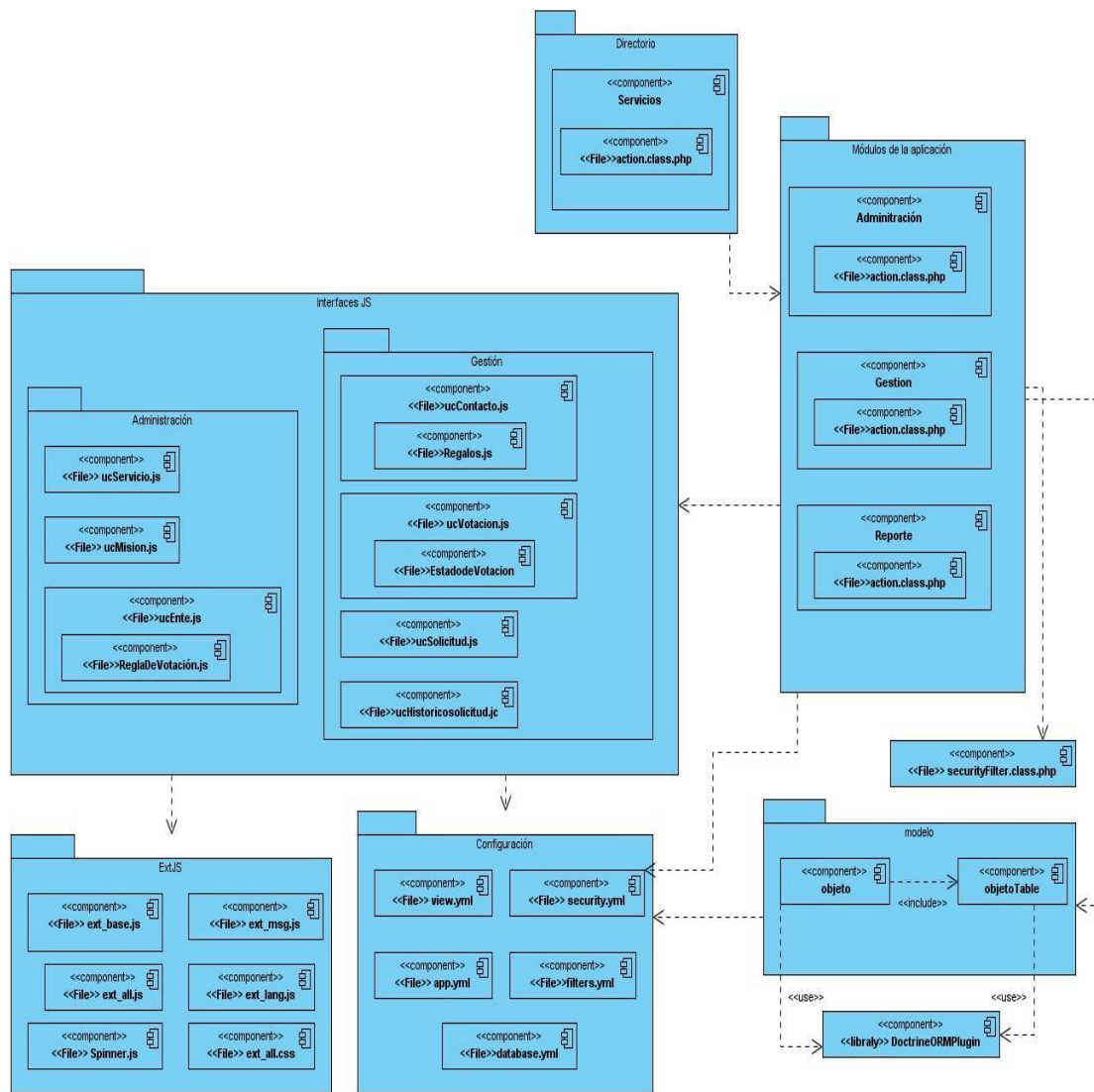


Ilustración 4 Diagrama de componentes.

3.3 Diagrama de despliegue.

Un Diagrama de Despliegue modela la arquitectura en tiempo de ejecución de un sistema. Esto muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se trazan en esos nodos. (26)

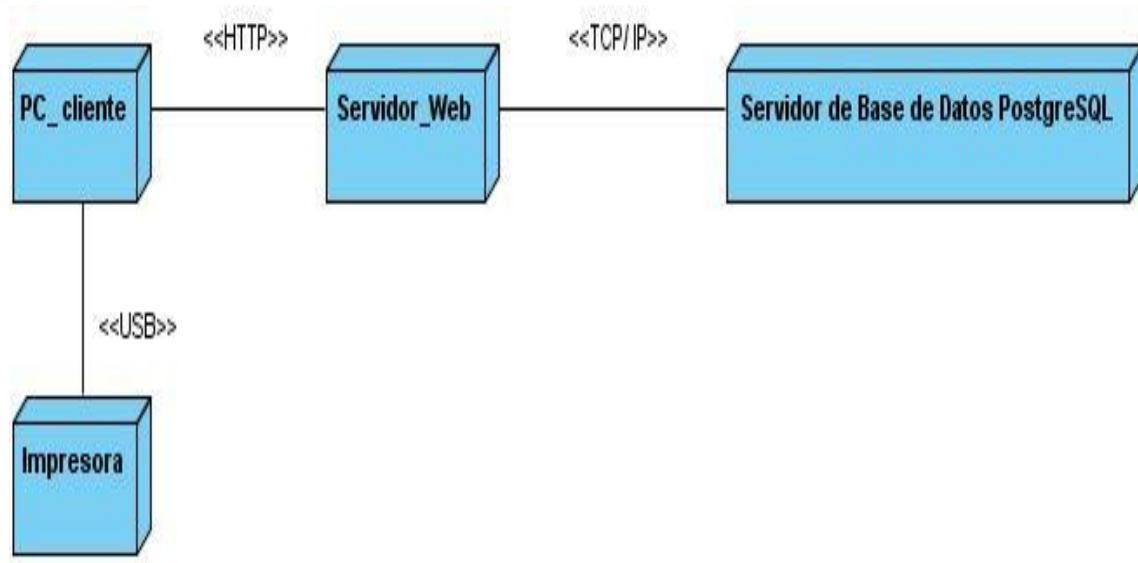


Ilustración 5. Diagrama de despliegue

3.4 Pruebas

Las pruebas constituyen una etapa imprescindible durante el proceso de desarrollo del software pues permiten detectar y corregir el máximo de errores posibles antes de la entrega al cliente del software desarrollado, por lo que el éxito de las mismas puede mejorar la percepción de calidad del usuario final y lograr su satisfacción. El objetivo principal de las pruebas es asegurar que el software cumpla con las especificaciones requeridas y eliminar los posibles defectos que este pudiera tener. Es importante considerar que las pruebas de software no garantizan que un sistema esté libre de errores, sino que se detecten la mayor cantidad de defectos posibles para su debida corrección. (27)

Principios de la prueba

Antes de la aplicación de métodos para el diseño de casos de prueba efectivos, se deberá entender ciertos principios básicos que guían las pruebas del software. (28)

- La prueba puede ser usada para mostrar la presencia de errores, pero nunca de su ausencia.
- La principal dificultad del proceso de prueba es decidir cuándo parar.
- Evitar casos de pruebas no planificados, no reusables y triviales a menos que el programa sea verdaderamente sencillo.
- Una parte necesaria de un caso de prueba es la definición del resultado esperado.
- Los casos de pruebas tienen que ser escritos no solo para condiciones de entrada válidas y esperadas sino también para condiciones no válidas e inesperadas.
- Los casos de pruebas tienen que ser escritos para generar las condiciones de salida deseadas.
- El número de errores sin descubrir es directamente proporcional al número de errores descubiertos.
- Las pruebas deberían empezar por lo pequeño y progresar hacia lo grande.
- Con la excepción de las pruebas de unidad e integración, un programa deberá ser probado por la persona u organización que lo desarrolló.
- Asigna el programador más creativo a la prueba.

Niveles de Prueba

La Prueba es aplicada para diferentes tipos de objetivos, en diferentes escenarios o niveles de trabajo. Se distinguen los siguientes niveles de pruebas (29):

- Prueba de desarrollador
- Prueba independiente
- Prueba de Unidad
- Prueba de Integración
- Prueba de sistema
- Prueba de aceptación

Prueba de Desarrollador. Es la prueba diseñada e implementada por el equipo de desarrollo. Tradicionalmente estas pruebas han sido consideradas solo para la prueba

de unidad, aunque en la actualidad en algunos casos pueden ejecutar pruebas de integración. Se recomienda que estas pruebas cubran más que las pruebas de unidad.

Prueba independiente: Es la prueba que es diseñada e implementada por alguien independiente del grupo de desarrolladores. El objetivo de estas pruebas es proporcionar una perspectiva deferente y en un ambiente más rico que los desarrolladores. Una vista de la prueba independiente es la prueba independiente de los stakeholder, que son pruebas basadas en las necesidades y preocupaciones de los stakeholders.

Prueba de unidad: Es la prueba enfocada a los elementos testeables más pequeño del software. Es aplicable a componentes representados en el modelo de implementación para verificar que los flujos de control y de datos están cubiertos, y que ellos funcionen como se espera. La prueba de unidad siempre está orientada a caja blanca. Antes de iniciar cualquier otra prueba es preciso probar el flujo de datos de la interfaz del componente. Si los datos no entran correctamente, todas las demás pruebas no tienen sentido. El diseño de casos de prueba de una unidad comienza una vez que se ha desarrollado, revisado y verificado en su sintaxis el código a nivel fuente.

Prueba de integración: Es ejecutada para asegurar que los componentes en el modelo de implementación operen correctamente cuando son combinados para ejecutar un caso de uso. Se prueba un paquete o un conjunto de paquetes del modelo de implementación. Estas pruebas descubren errores o incompletitud en las especificaciones de las interfaces de los paquetes. Esta prueba debe ser responsabilidad de desarrolladores y de independientes, sin solaparse las pruebas.

Es el proceso de combinar y probar múltiples componentes juntos. El objetivo es tomar los componentes probados en unidad y construir una estructura de programa que esté de acuerdo con lo que dicta el diseño.

Prueba de Sistema: Son las pruebas que se hacen cuando el software está funcionando como un todo. Es la actividad de prueba dirigida a verificar el programa final, después que todos los componentes de software y hardware han sido integrados.

En un ciclo iterativo estas pruebas ocurren más temprano, tan pronto como subconjuntos bien formados de comportamiento de caso de uso son implementados.

Prueba de aceptación: Prueba de aceptación del usuario es la prueba final antes del despliegue del sistema. Su objetivo es verificar que el software está listo y que puede ser usado por usuarios finales para ejecutar aquellas funciones y tareas para las cuales el software fue construido. Un comentario sobre la secuencia de los niveles de prueba. Las pruebas de unidad son implementadas en la iteración más temprana como el primer nivel de prueba.

Pero en un proceso iterativo ejecutar todas las pruebas de unidad antes de pasar a niveles siguientes de prueba como regla es inapropiada. Una mejor estrategia es identificar las pruebas de unidad, integración y sistema que ofrecen mayor potencial para encontrar errores y entonces implementarlas y ejecutarlas.

Tipos de prueba de Caja Blanca y Negra

Cualquier producto de ingeniería puede ser probado de una de estas formas:

1. Conociendo la funcionalidad específica para la cual fue diseñado el producto, se pueden llevar a cabo pruebas que demuestren que cada función es completamente operativa.
2. Conociendo el funcionamiento del producto se pueden desarrollar pruebas que aseguren que “todas las piezas encajen”, o sea, que la operación interna se ajusta a las especificaciones y que todos los componentes internos se han comprobado de forma adecuada.

El 1er enfoque se denomina Prueba de Caja Negra y el 2do Prueba de Caja Blanca.

Prueba de caja negra: Se refiere a las pruebas que se llevan a cabo sobre la interfaz del software, por lo que los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce una salida correcta, así como que la integridad de la información externa se mantiene. Esta prueba examina algunos aspectos del modelo fundamentalmente del sistema sin tener mucho en cuenta la estructura interna del software.

Prueba de caja blanca: Se basa en el minucioso examen de los detalles procedimentales. Se comprueban los caminos lógicos del software proponiendo casos de prueba que examinen que están correctas todas las condiciones y/o bucles para

determinar si el estado real coincide con el esperado o afirmado. Esto genera gran cantidad de caminos posibles por lo que hay que dedicar esfuerzos a la determinación de las condiciones de prueba que se van a verificar.

3.5 Validación de la implementación.

Partiendo del estudio de la metodología aplicada se seleccionan las pruebas unitarias y de aceptación a fin de obtener la validación del subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en la Facultad 3.

Evaluación de la ejecución de las pruebas Unitarias.

A continuación se describe el caso de prueba unitaria aplicando caja blanca para el método `ChequearEstadoVotacion`.

Prueba del camino básico.

La prueba del camino básico es una técnica de prueba de la Caja Blanca propuesta por Tom McCabe. Esta técnica permite obtener una medida de la complejidad lógica de un diseño y usar esta medida como guía para la definición de un conjunto básico. La idea es derivar casos de prueba a partir de un conjunto dado de caminos independientes por los cuales puede circular el flujo de control. Para obtener dicho conjunto de caminos independientes se construye el Grafo de Flujo asociado y se calcula su complejidad ciclomática. Los pasos que se siguen para aplicar esta técnica son:

- A partir del diseño o del código fuente, se dibuja el grafo de flujo asociado.
- Se calcula la complejidad ciclomática del grafo.
- Se determina un conjunto básico de caminos independientes.
- Se preparan los casos de prueba que obliguen a la ejecución de cada camino del conjunto básico.

Los casos de prueba derivados del conjunto básico garantizan que durante la prueba se ejecuta por lo menos una vez cada sentencia del programa. (28)

Notación de Grafo de Flujo.

Para aplicar la técnica del camino básico se debe introducir una sencilla notación para la representación del flujo de control, el cual puede representarse por un Grafo de

Flujo. Cada nodo del grafo corresponde a una o más sentencias de código fuente. Todo segmento de código de cualquier programa se puede traducir a un Grafo de Flujo. A continuación se muestra la sentencia de código del método ChequearEstadoVotacion enumerada y posteriormente el Grafo de Flujo correspondiente a este método.

```

public function chequearEstadoVotacion($idsolicitud) {
    1   $vetan = NenteTable::obtenerVotosVetan($idsolicitud)->count();
    $aceptados = NenteTable::obtenerVotosAceptados($idsolicitud)->count()
    $rechazados = NenteTable::obtenerVotosRechazados($idsolicitud)->count();
    $total = NenteTable::obtenerTotalVotan();
    $totalvotos = $aceptados + $rechazados;
    $reglaVotacion = NreglaVotacionTable::obtenerReglaVotacion();
    $estadorechazado = array("id_estado_sistema" => EEstado_Solicitud::$Rechazada);
    2   if($vetan > 0)
    3       $obj = GenericTable::updateRecord("Solicitud", $idsolicitud, array($estadorechazado));
    4   if($totalvotos == $total) {
    5       $porcentaje = $aceptados * 100 / $total;
    $estado = array("id_estado_sistema" => EEstado_Solicitud::$Aceptada); 7
    6   if($porcentaje < $reglaVotacion->porcentajeaprobacion || $vetan > 0)
    8       $estado["id_estado_sistema"] = EEstado_Solicitud::$Rechazada;
    9       $estado["id_estado_final"] = $estado["id_estado_sistema"];
    GenericTable::updateRecord("Solicitud", $idsolicitud, $estado);
    10  }
}

```

Ilustración 6. Prueba de caja blanca del método ChequearEstadoVotacion

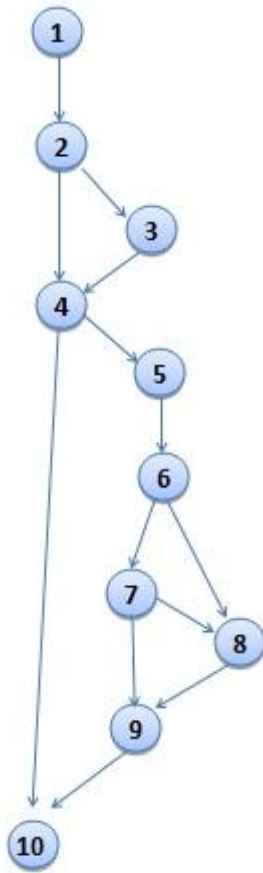


Ilustración 7. Grafo de Flujo

Complejidad Ciclomática

La complejidad ciclomática es una métrica de software extremadamente útil pues proporciona una medición cuantitativa de la complejidad lógica de un programa. El valor calculado como complejidad ciclomática define el número de caminos independientes del conjunto básico de un programa y nos da un límite superior para el número de pruebas que se deben realizar para asegurar que se ejecute cada sentencia al menos una vez. (28)

El número de regiones del grafo de flujo coincide con la complejidad ciclomática. La complejidad ciclomática, $V(G)$, se define como:

$$V(G) = A - N + 2$$

Donde: A es el número de aristas del grafo y N es el número de nodos.

$$V(G) = A - N + 2$$

$$V(G) = 13 - 10 + 2$$

$$V(G) = 5$$

Camino Independiente

Un camino independiente es cualquier camino del programa que introduce por lo menos un nuevo conjunto de sentencias de procesamiento o una nueva condición. El camino independiente se debe mover por lo menos por una arista que no haya sido recorrida anteriormente.

Como resultado final del análisis del Grafo de Flujo generado por el método ChequearEstadoVotacion se obtiene los siguientes caminos independientes, los cuales son recorridos para cualquier x arrojando siempre resultados positivos:

Camino1: 1-2-4-10

Camino2: 1-2-3-4-10

Camino3: 1-2-3-4-5-6-7-9-10

Camino4: 1-2-3-4-5-6-7-8-9-10

Camino5: 1-2-3-4-5-6-8-9-10

Evaluación de la ejecución de las pruebas de aceptación.

Estas pruebas se crean a partir de las historias de usuario. Una HU puede tener todas las pruebas de aceptación que necesite para asegurar su correcto funcionamiento. El objetivo final de estas es garantizar que los requerimientos han sido cumplidos y que el sistema es aceptable.

Se efectuaron pruebas de caja negra según el plan de iteración en tres ocasiones arrojando resultados de un 85% de aceptación en la primera iteración, en la segunda de un 95% y en la última a un 100%. En gráfica se establece una comparación de los resultados obtenidos por iteraciones.

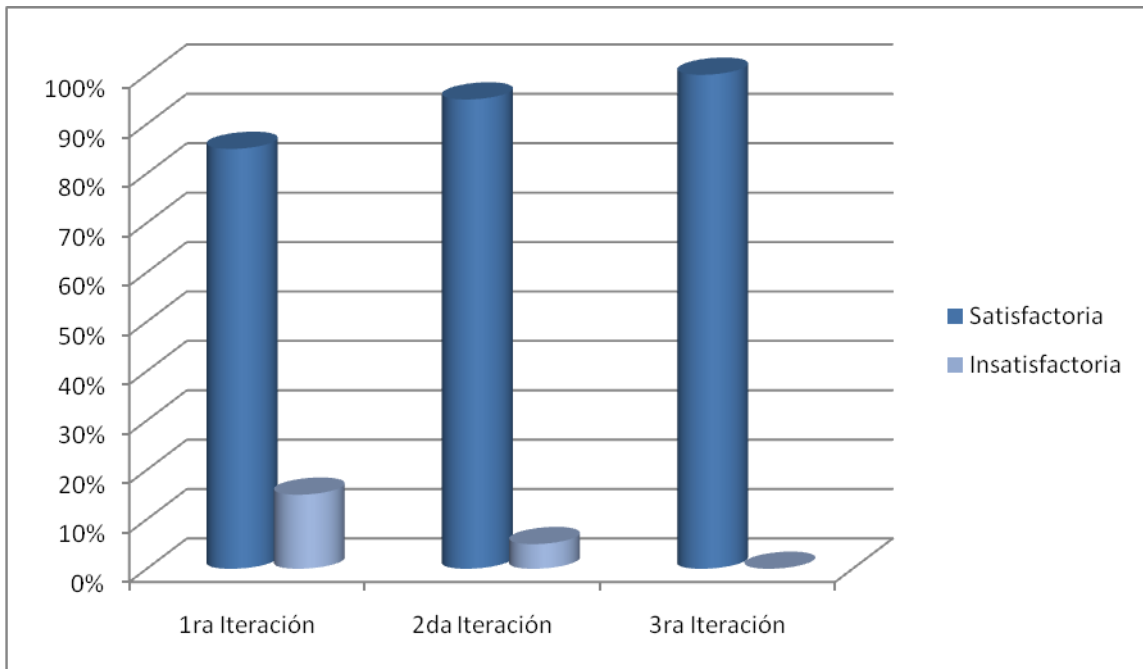


Tabla 9. Gráfica de resultados por iteraciones

A continuación se muestra las dos primeras pruebas de aceptación efectuadas con resultados satisfactorios las demás se pueden apreciar en el expediente de proyecto.

Tabla 10. Caso de prueba 1

Caso de Prueba de Aceptación	
Código Caso de Prueba: SIGECI11	Nombre Historia de Usuario: Realizar votación.
Nombre de la persona que realiza la prueba: Raynel Batistas Tellez	
Descripción de la Prueba: La página inicial del subsistema contiene a la izquierda una columna con los módulos gestión y administración, dentro del módulo gestión se puede observar el link a la interfaz de realizar votación, donde se encuentran todas las solicitudes que han sido revisadas por lo que su estado está pendiente a votación. Se da clic en el botón realizar votación y se presencia como el estado	

cambia a aceptado o rechazado en dependencia de la decisión del votante.
Condiciones de Ejecución: El usuario debe tener identificado su rol de votante y al menos estar una solicitud pendiente a votación.
Entrada / Pasos de ejecución: El votante escoge la opción realizar votación, señala la solicitud por la cual desea votar, ejecuta un clic en el botón realizar votación, escoge si va a ser rechazada o aceptada y da clic en el botón aceptar.
Resultado Esperado: La votación se realizo correctamente obteniendo los resultados esperados según los requisitos previstos.
Evaluación de la Prueba: Satisfactoria

Tabla 11. Caso de prueba 2

Caso de Prueba de Aceptación	
Código Caso de Prueba: SIGECI22	Nombre Historia de Usuario: Mostrar estado de la votación.
Nombre de la persona que realiza la prueba: Raynel Batistas Tellez	
Descripción de la Prueba: La página inicial del subsistema contiene a la izquierda una columna con los módulos gestión y administración, dentro del módulo gestión se puede observar el link a la interfaz de realizar votación, donde se encuentran todas las solicitudes que han sido revisadas por lo que su estado está pendiente a votación .Se selecciona la solicitud de la que se desea conocer su estado de la votación. Se da clic en el botón Estado de la votación y se muestra en una interfaz el estado en que se encuentra la votación.	
Condiciones de Ejecución: El usuario debe tener identificado su rol de votante y al menos estar una solicitud pendiente a votación.	
Entrada / Pasos de ejecución: El votante escoge la opción realizar votación,	

señala la solicitud a la cual desea verle el estado de la votación, ejecuta un clic en el botón Estado de la votación.

Resultado Esperado: Se realizó correctamente el objetivo planteado para la prueba y se cumplieron los resultados esperados según los requisitos previstos.

Evaluación de la Prueba: Satisfactoria

Acta de Aceptación

A modo de concluir satisfactoriamente el trabajo se confeccionaron 2 actas de aceptación una dirigida al asesor de cooperación internacional de la facultad 3 y otra al jefe del Grupo de Investigación y Desarrollo de la Plataforma SO3. En la misma consta la aceptación total por parte de los principales clientes del Subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en la Facultad 3.

3.6 Conclusiones

Con el presente capítulo se obtuvo los resultados del diseño y las pruebas propuestas por la metodología. Arroja el cumplimiento satisfactorio de la realización de las pruebas de unitarias y de aceptación. Mediante el transcurso de construcción, se encontraron errores que fueron rectificados para el uso correcto del producto y la satisfacción del cliente. De esta manera logra el software la calidad y la confiabilidad adecuada para la utilización del producto.

CONCLUSIONES

En el diagnóstico inicial se puede constatar que la muestra seleccionada valora la necesidad e importancia de la realización del proceso de negocio de trámites de cooperación internacional sin embargo presentan insuficiencia en la determinación de las variables que necesitan ser guardadas y manipuladas en determinado momento por los directivos y por tanto es necesario y novedoso el aporte brindado por el presente trabajo de diploma.

El Subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en La Facultad 3 que se presenta esta caracterizado por su disponibilidad e integridad a través de una aplicación web

La solución informática implementada, demostró el manejo eficiente de los datos relacionados con los trámites de salidas y visitas, hacia y desde el exterior, en La Facultad 3 y la integración con la plataforma tecnológica de esta, aspecto este medular en la novedad e importancia del resultado científico alcanzado.

RECOMENDACIONES

Después de haber sido realizadas las conclusiones del presente trabajo se recomienda:

- Implantar La Plataforma Informativa SO3 y a su vez el Subsistema para la gestión de los procesos de trámites de salidas y visitas, hacia y desde el exterior, en la Facultad 3.
- Estudiar los procesos asociados a la gestión de trámites de cooperación internacional con el fin de identificar nuevos procesos a integrar al subsistema
- Aplicar la solución en las diferentes facultades de la Universidad.

•

BIBLIOGRAFÍA

1. **Bravo, Alfredo Perez.** Cooperacion Tecnica Internacional. 2007.
2. **Ayala, Reisser Mejias y Ruiz, Keytia Quintero.** *Sistema Informatizado de Cooperación Internacional. Análisis y Diseño del Subsistema de Trámites.* La Habana : s.n., 2008.
3. **Academia, Real.** *diccionario de la lengua española.* 2007.
4. **Rojas, Suset Fernández y Fonseca, Eduardo Luis Piña.** *Análisis y Diseño del Sistema Informatizado de Cooperación Internacional.Subsistema de Cooperación.* La Habana : s.n., 2008.
5. **Guzmán., Dra.C. Esther María Pino.** *La Excelencia en la Dirección Educacional Latinoamericana.* 2007.
6. **ERP-Cuba, Proyecto.** *Modelo de Desarrollo Orientado a Componentes del proyecto ERP-Cuba.* 2008.
7. **International, V.P. 2006.** Introducción a los sistemas y herramientas CASE. 2006.
8. **GARCIA, X.C and J. M. ALFONSO.** Introducción a los Sistemas de Gestión de Contenidos. 2004.
9. **Romero, Oscar Casasola.** Programacion en Castellano. [En línea] 27 de agosto de 2010. [Citado el: 24 de enero de 2011.] http://www.programacion.com/articulo/introduccion_a_uml_181.
10. **Visual Paradim.** freedownloadmanager.org. [En línea] freedownloadmanager.org, 2008. [Citado el: 24 de enero de 2011.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.
11. **Stephen A. White, IBM Corporation.** Introduction to BPMN.
12. Conferencia 1: Introducción a la asignatura de sistemas de bases de datos. *Entorno Virtual de Aprendizaje.* [En línea] 2010. [Citado el: 28 de Marzo de 2011.]

13. **Toledo, Álvaro.** UPTODOWN. [En línea] 2010. [Citado el: 24 de 1 de 2011.] <http://postgresql.uptodown.com/>.
14. **Guía Ubuntu.** Guía Ubuntu. [En línea] 10 de marzo de 2008. [Citado el: 24 de enero de 2011.] http://www.guia-ubuntu.org/index.php?title=PgAdmin_III.
15. **Alvarez, Miguel angel.** desarrolloweb. [En línea] 2009.
16. **Maestros del web.** Maestros del web. [En línea] 2010. [Citado el: 8 de febrero de 2011.] <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>.
17. **Grupo de autores.** HTML.net. [En línea] 2010. [Citado el: 20 de enero de 2011.] <http://es.html.net/tutorials/css/lesson1.php>.
18. **Patxi.** Eslomas.com. *Frameworks de Zend para el desarrollo de aplicaciones PHP*. [En línea] 3 de julio de 2007. [Citado el: 8 de febrero de 2011.] <http://www.eslomas.com/index.php/archives/2007/07/03/framework-de-zend-para-el-desarrollo-de-aplicaciones-php/>.
19. **Vargas.** Wordpress.com. *Librería ExtJs*. [En línea] 6 de agosto de 2007. [Citado el: 8 de febrero de 2011.]
20. **Ciberaula.** Ciberaula.com. *Una Introducción a APACHE*. [En línea] 2010. [Citado el: 8 de febrero de 2011.] http://linux.ciberaula.com/articulo/linux_apache_intro/.
21. **The PHP Group.** PHP: Hipertext Preprocesor. [En línea] php.net, 4 de 2 de 2011. [Citado el: 6 de febrero de 2011.] <http://php.net/manual/es/intro-what-is.php>.
22. **BETSIME.** Revista BETSIME. *La revista del empresario cubano*. [En línea] 2011. [Citado el: 20 de enero de 2011.] http://www.betsime.disaic.cu/secciones/eco_enemar_07.htm#2.
23. **Peñalver Romero, Gladys Marsi, y otros.** *Metodología ágil para proyectos de software libre*. Ciudad de la habana : s.n., 2009.
24. **Potencier, Fabien.** *Symfony la guía definitiva* . 2008.
25. **Fernández González, Mairelys y Zorrilla Rivera, Osley.** Diseño e implementación del componente Ajuste al Costo del Subsistema Costos y Procesos del Sistema Integral de Gestión

de Entidades CEDRUX. *Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas*. Ciudad de la Habana : UCI, 2010.

26. Sparx Systems. [En línea] Sparx Systems Pty Ltd, 2007. [Citado el: 5 de mayo de 2011.] http://www.sparxsystems.com.ar/resources/tutorial/uml2_deploymentdiagram.html.

27. **Unidad de Compatibilización Integración y Desarrollo**. *Proceso de Desarrollo y Gestión de Proyectos de Software*. Ciudad de la Habana : s.n., 2009.

28. Entorno virtual de aprendizaje. [En línea] 2012. http://eva.uci.cu/file.php/160/Curso_2010-2011/Semana_9/Conferencia_7/Materiales_Basicos/Documentacion_sobre_Pruebas.pdf.

29. **Pressman, Roger S**. Ingeniería del Software. Un enfoque práctico. 2004.

30. **Mozilla Firefox**. GetFirefox. [En línea] 2009. [Citado el: 6 de febrero de 2011.] <http://www.getfirefox.es/firefox-features>.

31. **Dirección de Economía del MINFAR**. *Manual del Sistema de Contabilidad Financiera para la Actividad Presupuestada en las FAR*. Plaza de la Revolución, Ciudad de la Habana : Centro de Información para la Defensa del MINFAR, 2003.

32. **Microsoft**. Microsoft Dynamics|Soluciones empresariales ERP y CRM. [En línea] Microsoft Dynamics, 2011. [Citado el: 20 de enero de 2011.] <http://www.microsoft.com/dynamics/es/es/default.aspx>.

33. **Openbravo Wiki**. Openbravo Wiki. [En línea] 26 de 6 de 2009. [Citado el: 20 de enero de 2011.] http://wiki.openbravo.com/wiki/User_Manual_2.40/es.

34. **CENDITEL**. Plataforma de Desarrollo de Software Libre (PDSL). *Manual del Usuario del Sistema de Control de Versiones (SVN)*. [En línea] CENDITEL, 2011. [Citado el: 20 de enero de 2011.] <http://plataforma.cenditel.gob.ve/wiki/ManualUsuarioSvn>.

35. **CITMATEL**. Rodas XXI Sistema Integral Económico Administrativo. [En línea] 2011. [Citado el: 20 de enero de 2011.] <http://www.rodasxxi.cu/contabilidad.php>.

36. **Tobías, María**. Lainformacion.com. [En línea] 8 de 12 de 2008. [Citado el: 20 de enero de 2011.] <http://tobiassig08.bitacoras.com/archivos/2008/12/08/software-empresarial-y-nuevas-tecnologias-erp2>.

37. **Scribd.** Scribd. [En línea] 2010. [Citado el: 22 de enero de 2011.] <http://www.scribd.com/doc/3062020/Capitulo-I-HERRAMIENTAS-CASE>.
38. **Colectivo de Json.** JSON. [En línea] JSON, 2011. [Citado el: 22 de enero de 2011.] <http://www.json.org/json-es.html>.
39. **Grupo editorial.** Entrebits.CL Comunidad Informática. [En línea] 10 de abril de 2008. [Citado el: 8 de febrero de 2011.] <http://www.entrebits.cl/foros/herramientas-de-programacion/11472-zend-studio-eclipse.html>.
40. **ProgramacionWeb.net.** ProgramacionWeb.net. [En línea] ProgramacionWeb.net, 2009. [Citado el: 8 de febrero de 2011.] <http://www.programacionweb.net/cursos/curso.php?num=2>.
41. **Colectivo editorial.** W3C Consortium. [En línea] 9 de enero de 2008. [Citado el: 8 de febrero de 2011.] <http://www.w3c.es/divulgacion/guiasbreves/tecnologiasxml>.
42. **machado, Daymel.** *Implementación del módulo Distribución de productos*. Ciudad de la Habana : s.n., 2010.
43. **Doctrine-Project.org.** Doctrine. [En línea] 2010. [Citado el: 8 de febrero de 2011.] <http://www.doctrine-project.org>.
44. **Colectivo de editores.** Programación I y Estructura de Datos. [En línea] 2007-2008. [Citado el: 6 de febrero de 2011.] <http://upsg01.foroactivo.com/f10-tema-2-estructura-de-datos-paralelo-22>.
45. **Hernandez, Juan Darien Macias.** Dragones, Facultad 3. [En línea] Proyecto PCI, Facultad 3., 15 de marzo de 2011. [Citado el: 20 de abril de 2011.] http://facultad3.uci.cu/index.php?option=com_content&view=article&id=331&Itemid=146.
46. **Pérez, Ing. Joisel Pérez.** *Línea base de la arquitectura de software Contabilidad*. s.l. : ERP Cuba, 2010.
47. **ERP Cuba.** *Vista de datos de la arquitectura* . 2010.
48. **Martinez, Rafael.** PostgreSQL-es. [En línea] PostgreSQL-es, 6 de junio de 2009. [Citado el: 24 de mayo de 2011.] <http://www.postgresql.org.es/node/297>.

