

Universidad Central "Marta Abreu" de las Villas  
Facultad Matemática, Física y Computación  
Ingeniería Informática



Título: Sistema informático para la gestión de los  
procesos de la Dirección de Relaciones  
Internacionales en la Universidad Central "Marta  
Abreu" de Las Villas (SIRI-UCLV).

**Autores:** Alejandro Bermúdez Espinosa  
Raúl Armando Alfonso Camacho

**Tutores:**  
Dr. Romel Vázquez Rodríguez  
Dr. Alina Montero Torres

Santa Clara  
2016

## **Dedicatoria**

*A mis padres Belkis y Raúl y mi hermano Alex que siempre han estado a mi lado y me han apoyado en los momentos difíciles a lo largo de estos cinco años.*

*A mi abuelo Machado por haberme alentado a estudiar con la frase que me decía siempre "Méchate que todo lo que aprendas nadie te lo quita".*

*A mi novia Dianelys por estar a mi lado estos últimos años de la carrera y brindarme tanto amor.*

*A toda mi familia, los que están aquí y a los que se encuentran un poco más lejos.*

*Raúl Armando Alfonso Camacho*

## **Dedicatoria**

*A mis padres que siempre depositan su confianza en mí, me aconsejan y me apoyan en mis decisiones.*

*A todos mis amigos, a mi familia y vecinos que me apoyaron y ayudaron a lo largo de esta carrera y estuvieron siempre atentos.*

*Alejandro Bermúdez Espinosa*

# Agradecimientos

*A mis padres Belkis y Raúl por exhortarme a seguir adelante y ser un hombre de bien.*

*A mi hermano y a toda mi familia por estar ahí para mí.*

*A mi prima Jessica por siempre ayudarme cuando lo necesitaba.*

*A mi compañero de tesis Alejandro por haber desarrollado esta tesis conmigo.*

*A mis amigos Arocha y Dayron por estar ahí cuando necesitaba de su ayuda.*

*A mis tutores, los Dres. Romel Vázquez Rodríguez y Alina Montero Torres por su ayuda.*

*A Saydani y Annia del departamento de Trámites por su apoyo y ayuda desde el comienzo.*

*A la compañera Milagros del departamento de Colaboración Internacional por su ayuda.*

*A la profesora Yailén por su ayuda en la revisión de la tesis.*

*A mis compañeros de aula por haber pasado tantos momentos buenos juntos y compartir tantas experiencias.*

*Raúl Armando Alfonso Camacho*

## **Agradecimientos**

*A mis padres que son lo más grande que tengo en mi vida y a los que le debo todo.*

*A todos mis amigos y personas allegadas que me apoyaron y ayudaron a lo largo de esta carrera en especial a Daily, Leidy, David y Joel que fueron de vital apoyo.*

*A todos mis compañeros de grupo a Yasser, Rafael, Raúl, Ernesto, Yandy, Landin, Héctor, Eddy, Pablo.*

*A mi familia y vecinos que estuvieron siempre atentos.*

*A mis tutores, los Dres. Romel Vázquez Rodríguez y Alina Montero Torres por su ayuda.*

*A Saydani y Annia del departamento de trámites.*

*A la compañera Milagros del departamento de colaboración internacional por su ayuda.*

*Alejandro Bermúdez Espinosa*

## Resumen

El desarrollo de la humanidad es el resultado de su desarrollo tecnológico, por lo que no explotar la tecnología solo frenará el avance, ya sea desde un simple negocio hasta el mundo entero. Dentro de los tipos de tecnologías existentes se encuentran las Tecnologías de la Información y las Comunicaciones (TIC), el uso de éstas para aumentar la calidad en la ejecución de los procesos de internacionalización institucional de nuestra universidad se hacen de vital importancia. Por consiguiente, la presente investigación se realiza en la Dirección de Relaciones Internacionales de la Universidad Central “Marta Abreu” de Las Villas (DRI-UCLV). El propósito de ésta, es implementar un Sistema Informático con el fin de gestionar los datos que son manejados por dicha entidad, este sistema será capaz de registrar toda la información relevante, evitar errores en la inserción de datos, facilitar la búsqueda de estos y alcanzar un nivel de socialización de la información necesario para lograr un mejor funcionamiento de dicha institución, dicho sistema será implementado haciendo uso del *framework* Symfony y el gestor de bases de datos PostgreSQL.

## **Abstract**

The development of mankind is the result of technological development, so not exploiting technology would jeopardize the advance either of a simple business or of the world. Among the types of technologies we can find the Information and Communications Technologies (ICT). Their use to increase the quality in the execution of the processes of institutional internationalization of our university becomes vital. Therefore, this research is done in the International Relations Office of the Universidad Central “Marta Abreu” de Las Villas (DRI-UCLV). The purpose of this research is to implement a computer system in order to manage the data that are managed by that entity. This system will be able to record all relevant information, avoid errors in data entry, facilitate their search and achieve a level of socialization of the information that is necessary to attain a better functioning of the institution. The system will be implemented using the symphony framework and the database management system PostgreSQL.

## Tabla de contenido

<b>Introducción.....</b>	<b>1</b>
<b>Capítulo I. Fundamentación teórica. ....</b>	<b>5</b>
1.1    Dirección de relaciones internacionales .....	5
1.1.1    Objeto de estudio.....	5
1.1.2    Procesos de gestión de la Dirección de Relaciones Internacionales.....	8
1.2    Tendencias y tecnologías actuales .....	14
1.2.1    Software Libre .....	14
1.2.1.1    ¿Por qué utilizar Software Libre? .....	15
1.2.2    Framework .....	17
1.2.3    ¿Qué es PHP? .....	21
1.2.4    Sistemas Gestores de Bases de Datos .....	22
1.3    Conclusiones Parciales .....	24
<b>Capítulo II. Modelo del Negocio y Requisitos.....</b>	<b>26</b>
2.1    Modelo del negocio en la DRI-UCLV.....	26
2.1.1    Reglas del negocio a considerar .....	26
2.1.2    Actores del negocio .....	27
2.1.3    Diagrama de casos de uso del negocio .....	28
2.1.4    Trabajadores del negocio en la DRI-UCLV .....	30
2.1.5    Casos de uso del negocio .....	31
2.1.6    Actores del sistema a automatizar.....	32
2.1.7    Requisitos funcionales.....	33
2.1.8    Requisitos no funcionales. ....	34
2.1.9    Diagrama de Casos de Uso del Sistema. ....	38
2.1.10    Descripción de los casos de uso del Sistema (Significativos) .....	41
2.1.11    Paquetes del software.....	47
2.2    Conclusiones Parciales .....	48
<b>Capítulo III. Descripción de la propuesta de solución .....</b>	<b>49</b>
3.1    Arquitectura del Sistema.....	49
3.2    Diagrama de clases de diseño .....	50
3.3    Diagrama de secuencia .....	51
3.4    Tratamiento de errores .....	52
3.5    Diseño de la base de datos.....	55
3.5.1    Modelo conceptual de datos.....	55
3.5.2    Modelo físico de datos .....	56



3.6	Modelo de componentes .....	58
3.7	Diagrama de despliegue.....	59
3.8	Conclusiones Parciales .....	60
<b>Capítulo IV. Pruebas y análisis de factibilidad .....</b>		<b>61</b>
4.1	Planificación basada en uno de los métodos de estimación.....	61
4.1.1	Estimación basada en Casos de Uso.....	61
4.1.2	Conclusiones del Análisis de Factibilidad .....	68
4.2	Casos de Pruebas (caja negra).....	68
4.3	Conclusiones parciales .....	70
Conclusiones .....		71
Recomendaciones .....		72
Referencias bibliográficas .....		73
Anexos .....		74
Anexo 1.....		74
Anexo 2.....		74
Anexo 3.....		74
Anexo 4.....		74
Anexo 5.....		75
Anexo 6.....		75
Anexo 7.....		75
Anexo 8.....		76

## **Introducción**

El desarrollo de la humanidad es el resultado de su desarrollo tecnológico, por lo que no explotar la tecnología solo frenará el avance ya sea desde un simple negocio hasta el mundo entero. Dentro de los tipos de tecnologías existentes se encuentran las Tecnologías de la Información y las Comunicaciones (TIC), estas favorecen la gestión de la información sin importar el formato en que se encuentre.

El término tecnologías de la información se usa a menudo para referirse a cualquier forma de hacer cómputo. Como nombre de un programa de licenciatura, se refiere a la preparación que tienen estudiantes para satisfacer las necesidades de tecnologías en cómputo y comunicación de gobiernos, seguridad social, escuelas y cualquier tipo de organización (ACM).

Planificar y gestionar la infraestructura de las TIC de una organización es un trabajo difícil y complejo, que requiere una base muy sólida de la aplicación de los conceptos fundamentales de áreas como las ciencias de la computación, así como de gestión y habilidades del personal. Se requieren habilidades especiales en la comprensión, por ejemplo, de cómo se componen y se estructuran los sistemas en red, y cuáles son sus fortalezas y debilidades. En sistemas de información hay importantes preocupaciones de software como la fiabilidad, seguridad, facilidad de uso y la eficacia y eficiencia para los fines previstos, todas estas preocupaciones son vitales para cualquier tipo de organización (ACM).

Un sistema de información es un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio. Teniendo muy en cuenta el equipo computacional necesario para que el Sistema de Información pueda operar y el recurso humano que interactúa con el Sistema de Información, el cual está formado por las personas que utilizan el sistema (Peralta, 2008).

La presente investigación se desarrolla en la Dirección de Relaciones Internacionales en la Universidad Central "Marta Abreu" de Las Villas (DRI-UCLV), proponiendo el diseño de un sistema informático con el fin de informatizar los procesos llevados a cabo en dicho centro. Para ello se analizan los antecedentes con respecto a previos sistemas de información o aplicaciones presentes en la DRI.

## **Antecedentes**

La DRI-UCLV tiene como tarea fundamental el diseño, seguimiento y evaluación de políticas y estrategias para el proceso de internacionalización institucional y la prestación de asesoría metodológica y servicios de tramitación y gestión efectivos para la consecución de acciones internacionales.

Dicha institución cuenta con varios departamentos donde se encuentran repartidas las tareas a realizar por dicho centro.

Dentro de los departamentos de la entidad se encuentra el departamento de trámites, este es el encargado tramitar la documentación de todas las misiones en el exterior que ejecutan los profesores, investigadores, estudiantes cubanos y administrativos de la UCLV. El departamento de colaboración internacional orienta su labor en la búsqueda de oportunidades de colaboración con universidades e instituciones extranjeras a través de la gestión y seguimiento de convenios y protocolos de colaboración; coordinación de programas de intercambio de especialistas y estudiantes y la presentación de candidatos de la UCLV a las diversas convocatorias de becas, también atiende todo lo referente a la membresía de la UCLV en organizaciones internacionales. Por otra parte la oficina coordinadora de servicios académicos (OCSA) tiene como misión coordinar y dar seguimiento a los servicios académicos internacionales, sirviendo como gestor e interfaz entre los clientes extranjeros y la UCLV, además de manejar los contratos con los estudiantes extranjeros que deseen cursar alguna de las modalidades de tipos de estudios que oferta la UCLV. El departamento de atención a estudiantes extranjeros (DAEE) tiene como cometido recibir y atender a los estudiantes extranjeros de pre- y postgrado, contribuyendo a su formación integral y propiciando el desarrollo de un profundo sistema de relaciones de amistad y solidaridad internacional.

## **Planteamiento del Problema**

En la DRI-UCLV, la información en estos momentos es manejada mediante ficheros de datos creados en Excel, donde cada departamento lleva un registro independiente de la información, por consiguiente, no se cuenta con una herramienta capaz gestionar la información de dicho centro de forma tal que responda a las necesidades de información del área. No se registra toda la información relevante y existen errores al recopilarla, lo que trae consigo dificultades en la búsqueda y la toma de decisiones por parte de los directivos. Existe poco grado de socialización de la información en sus procesos. Es por ello que para el desarrollo de la presente investigación se ha trazado el siguiente objetivo general.

## **Objetivo General**

Desarrollar un sistema informático para el perfeccionamiento de la gestión de los procesos de la DRI-UCLV, dando así, paso a los siguientes objetivos específicos:

## **Objetivos Específicos**

- Captar las diferentes reglas del negocio (RN) involucradas en los diferentes procesos que se gestionan en la DRI-UCLV.
- Implementar un sistema informático para la DRI-UCLV utilizando tecnologías web.
- Evaluar la validez del sistema informático mediante los casos de uso de pruebas.

## **Justificación de la investigación**

Se pretende con esta investigación dar solución a los problemas existentes en la DRI-UCLV, para esto se procura implementar una herramienta capaz de gestionar la información de dicho centro respondiendo a sus necesidades. Facilitar la búsqueda de información, la generación de reportes y la eliminación de información repetida. Toda la información quedará recogida en una base de datos única y no dividida en ficheros Excel por departamentos como ocurre en la actualidad.

## **Estructura del documento**

El presente trabajo de diploma quedó estructurado de la siguiente manera: introducción, cuatro capítulos, conclusiones, recomendaciones, referencias bibliográficas y anexos.

En el **Capítulo I** se abordan de manera general los conceptos necesarios para comprender la problemática expuesta en este trabajo de diploma, se exponen las tecnologías a usar en la implementación del nuevo sistema y las características de las mismas.

En el **Capítulo II** se recoge todo lo relacionado a la implementación del sistema así como la presentación de los principales diagramas del negocio, modelos y requisitos funcionales y no funcionales, además se consideran aspectos fundamentales de la arquitectura del sistema.

En el **Capítulo III** se detalla la propuesta de solución computacional a la problemática planteada, describiendo los servicios web diseñados, la arquitectura del sistema y se muestran los diagramas de clase y de secuencia, entre otros elementos de apoyo a la comprensión de la solución.

En el **Capítulo IV** se recogen, además de los casos de prueba apoyados en caja negra, un análisis de factibilidad mediante la planificación basada en casos de uso como método de estimación a utilizar.

## Capítulo I. Fundamentación teórica.

En este capítulo se abordan de manera general los conceptos necesarios para comprender la problemática expuesta en este trabajo de diploma, se exponen las tecnologías a usar en la implementación del nuevo sistema y las características de las mismas.

### 1.1 Dirección de relaciones internacionales

La Dirección de Relaciones Internacionales de la Universidad Central “Marta Abreu” de Las Villas tiene como misión fundamental el diseño, seguimiento y evaluación de políticas y estrategias para el proceso de internacionalización institucional y la prestación de asesoría metodológica y servicios de tramitación y gestión efectivos para la consecución de acciones internacionales.

#### 1.1.1 Objeto de estudio

La DRI-UCLV tiene varias funciones fundamentales; entre ellas se pueden destacar: el diseño consensuado con las estructuras universitarias de estrategias y políticas, la asesoría metodológica para el desarrollo de los procesos de internacionalización, la puesta en práctica de instrumentos para la inserción de la dimensión internacional en los procesos sustantivos de la institución, el seguimiento de las acciones, convenios, y planes específicos, el control y evaluación del cumplimiento de la estrategia y en base a los resultados proyectar su futura consecución, la prestación de servicios de gestión y tramitación efectivos. En los departamentos del centro se encuentran repartidas las tareas fundamentales.

El departamento de trámites se encarga de gestionar las propuestas de salida de todas las personas con intención de realizar un viaje oficial mediante la UCLV. Esta propuesta se almacena en un documento denominado diáspora, donde además de recoger los datos entregados por la persona (nombre y apellidos, número del carné de identidad, foto, país e institución a la que pretende viajar, en caso de ser profesor su categoría científica, facultad y departamento al cual pertenece, de ser trabajador el área al cual pertenece, de ser estudiantes cuál carrera estudia, el tipo de misión que puede ser misión oficial, profesor invitado, evento, beca pre doctoral y postdoctoral, entre otros; la fecha de inicio y fecha de fin. Dependiendo del país al cual pretende viajar la persona, es necesario entregar una Ficha de Salida, en el caso de viajar hacia Nicaragua y Bolivia se necesita entregar la Ficha de Salida del ALBA,

si es hacia Colombia se necesita entregar la Ficha de Salida de Colombia, en el caso de que los que vayan a viajar ocupen los cargos de decano de facultad, vicerrector o el de rector de la universidad tienen que entregar la Ficha de Salida RT. También se necesita el número del pasaporte de la persona de tenerlo confeccionado, además de la fecha de emisión, fecha de vencimiento del mismo y la persona que ha actualizado la diáspora). Se agrega un campo estado, este estado se refiere al estado en que se encuentra la solicitud de salida que puede ser: pendiente, aprobado o no aprobado. Al pasar el campo estado ha aprobado, se añade la fecha de aprobación en rectoría, la fecha real de salida y fecha real de regreso (estos dos campos de fecha dependen de que exista una fecha de aprobación en rectoría).

Otra de las funcionalidades del departamento de trámites consiste en la confección de nuevo pasaporte, este puede ser por concepto de nueva confección. Para la confección de dicho pasaporte es necesario que la persona llene una Solicitud de Pasaporte en la cual plasme todos sus datos personales, ya sea nombre y apellidos, color de piel, foto, fecha de nacimiento, padre, madre, sexo, estatura, color de ojos, color de cabello, ciudadanía, dirección particular, entre otras.

El departamento de atención a estudiantes extranjeros (DAEE) se encarga de atender a todos los estudiantes extranjeros del centro, ya sean de intercambio becarios o los autofinanciados y financiados por el gobierno. De los estudiantes se les recogen datos como el número del contrato en el caso de los financiados de gobierno y los autofinanciados, nombre y apellidos, fecha de entrada, fecha de caducidad, tipo de estudio que van a realizar, carrera que van a cursar, facultad donde la van a cursar y el tiempo que la van a cursar.

La oficina coordinadora de servicios académicos (OCSA) se encarga de varios procesos, dentro de ellos se encuentra el de la contratación de los estudiantes extranjeros financiados por el gobierno y autofinanciados, este contrato se confecciona en el momento de la llegada por primera vez del estudiante al centro. En él son recogidos los datos tales como el número del contrato, el nombre del estudiante, el monto total a remesar, el país de donde proviene, el programa que va a cursar (pregrado o postgrado), el tipo de estudio que pretende realizar (carrera completa, pasantía, curso corto, preparatoria, doctorado, maestría, curso, curso corto, especialización, entrenamiento), la facultad en la cual pretende realizar los estudios, el nombre del decano de la facultad y la fecha de inicio y fecha de cierre del contrato. Además, el estudiante debe realizar los pagos de acuerdo con el contrato firmado, estos pagos se harán siempre antes de empezar los diferentes semestres hasta alcanzar el monto total a pagar

especificado en el contrato. Luego de realizado el contrato, se le envía al DAEE un documento con los datos de todos los estudiantes que hayan realizado contratos y sus datos respectivos.

La OCSA también se encarga de atender los profesores (Profesores Invitados) con misiones aprobadas de asistencia técnica (PI) confeccionándole el expediente necesario para presentar en CUBATECNICA, además se registran los datos correspondientes al profesor, tales como: carné , nombre y apellidos, país, institución y facultad a la cual pretende viajar, fecha de salida y fecha de regreso, salario mensual, salario total, monto a remesar y monto remesado, fecha de entrega del expediente a la OCSA, fecha de entrega del expediente al MES y observaciones.

También se manejan los datos de los profesores que van a viajar por la empresa exportadora ANTEX hacia Angola, estos datos son: nombre y apellidos, la facultad en la cual trabaja, la fecha de aprobación de la rectoría, fecha de entrega de expediente a OCSA, la fecha de entrega de expediente al MES, observaciones, aprobación por ANTEX, esta aprobación puede tener tres estados: si, no y pendiente, de ser aprobado por ANTEX se necesita una fecha de salida por ANTEX y una fecha de regreso por ANTEX.

Otra de las tareas a realizar por la OCSA es el paquete de agencia, por esta vía se manejan las visitas de personas extranjeras al centro, las cuales pueden visitarlo con diferentes propósitos, de estas visitas se recogen los siguientes datos: nombre de la agencia (pueden ser varias agencias), fecha de solicitud, fecha de aceptación y fecha de visita, número de pax (cantidad de personas), tipo del servicio (pueden ser recorridos, conferencias especializadas, entre otros), el monto, la fecha de solicitud de factura al departamento de finanzas, fecha de entrega de factura a la DRI, fecha de entrega de factura a la agencia, fecha de pago y el país del cual provienen los visitantes.

El departamento de colaboración internacional es el encargado de varias tareas, dentro de estas se encuentra la de atención de visitas. De estas visitas se registran los datos de los visitantes tales como nombre y apellidos, ciudadanía, objetivo de la visita, actividad a realizar, fecha de entrada, fecha de salida y observaciones.

También el departamento se encarga de la recepción de estudiantes de intercambio (becarios). A partir de las solicitudes de recepción de estudiantes extranjeros que desean desarrollar pasantías por la modalidad de intercambio académico, los datos a recopilar son: nombre y



apellidos, su número de pasaporte, universidad de la que provienen, carrera que pretenden estudiar, fecha de comienzo de los estudios y fecha de fin de estudios, observaciones y el responsable que los va a atender; con el fin de llevar un registro exhaustivo del estudiante.

El departamento también maneja las redes y asociaciones a las que pertenece la UCLV, de estas se recopilan su nombre, año de afiliación, su representante en la UCLV, país a la que pertenece, cuota de afiliación, monto de pago de la afiliación, la fecha de pago en el año, área (Latino América, Europa, etc.), financiamiento y observaciones.

### 1.1.2 Procesos de gestión de la Dirección de Relaciones Internacionales

La DRI-UCLV maneja todo el proceso de la Estrategia de Internacionalización Institucional de la Universidad. Algunos de los principales procesos manejados por la DRI-UCLV serán explicados a continuación.

#### **Análisis crítico de la ejecución de los procesos y procesos objetos de automatización.**

La investigación realizada arrojó como resultado que de forma general en los departamentos encargados del manejo de la información de los procesos de la DRI-UCLV, se realiza el proceso de gestión y control, apoyándose en herramientas de trabajo como Microsoft Excel, además de tener la mayor parte de la información almacenada en formato duro por lo que el manejo de esta información resulta muy complicado y engorroso a la hora de realizar los reportes y brindar información real.

Como resultado de esta investigación se pretende informatizar varios procesos críticos existentes actualmente en la DRI-UCLV tales como la emisión de reportes, eliminar la duplicación de la información recopilada por los diversos departamentos, gestionar la información de manera correcta. Para esto se hace necesario tener en cuenta los siguientes procesos:

- **Propuesta de salida.**

En la UCLV, todas las personas, ya sean trabajadores, profesores o estudiantes, tienen la posibilidad de realizar viajes hacia otra institución universitaria en el exterior. El proceso comienza cuando la persona que pretende realizar un viaje hace entrega al especialista del departamento de trámites de la propuesta de salida, este revisa la propuesta de salida, esta puede ser aceptada o rechazada ya que puede presentar datos incorrectos o estar incompleta, en el caso de ser rechazada, se le pide a la persona que realice la solicitud nuevamente, en caso de ser aceptada, se registran en la diáspora los datos de la propuesta.

Luego de registrar los datos en la diáspora, el especialista formula un excel de la rectoría con todas las propuestas de salida que todavía no han sido discutidas o quedaron pendientes de la reunión anterior. El especialista envía a la rectoría el excel de la rectoría con todas las propuestas de salida, las cuales son analizadas y posteriormente se envía al departamento de trámites respuesta sobre que solicitudes de salida fueron aprobadas, cuales no y cuales quedaron pendientes para la próxima discusión. Posteriormente el especialista actualiza el campo “Estado de las solicitudes” en la diáspora. El especialista envía vía correo respuesta a la persona diciendo si fue aprobada o no.

En caso de ser aprobada, el especialista actualiza la diáspora agregando la fecha de aprobación de la rectoría, fecha real de salida y fecha real de regreso. El especialista revisa si la persona que fue aprobada tiene pasaporte. Si tiene pasaporte y no esta vencido, se revisa si al país al cual va a viajar necesita visa. Si necesita visa se le solicitan todos los datos referentes al país. La persona entrega los datos referentes al país. El especialista recoge los datos. En el caso de no tener pasaporte el especialista entrega a la persona una solicitud de pasaporte, la persona llena la solicitud y la entrega. El especialista revisa la solicitud de pasaporte, de estar correcta envía la solicitud a Inmigración. Luego Inmigración confecciona el pasaporte y lo envía al especialista y este a su vez entrega el pasaporte a la persona solicitante del trámite.

- **Proceso de gestión de la tarjeta de datos.**

A cada persona que haya sido aprobada para viajar mediante la UCLV, se le hace necesario registrar sus datos en el departamento de trámites con el fin de llevar el control de sus salidas, para esto el especialista del departamento chequea que a la persona en cuestión se le haya aprobado en rectoría el viaje, para comprobar esto revisa si existe la fecha de aprobación en

rectoría en la diáspora. Si la persona ha sido aprobada, se chequea si viaja por primera vez, de ser así se le confecciona una nueva tarjeta de datos. Si la persona ya ha viajado, se revisa si existe su tarjeta de datos, si existe, se le actualizan los datos, sino se le confecciona una nueva tarjeta.

- **Proceso de registro de estudiantes extranjeros en la UCLV de manera autofinanciada o financiada por el gobierno.**

Cuando un estudiante extranjero hace su entrada en la universidad pueden ser por concepto de estudiantes por convenio, autofinanciados o financiados por el gobierno. Los autofinanciados y financiados por el gobierno necesitan realizar un contrato económico, para esto, los estudiantes entregan sus datos al especialista de la OCSA con los cuales se les confecciona el contrato. Posteriormente el especialista de la OCSA envía una lista al especialista del DAEE con los datos de los estudiantes autofinanciados y financiados de gobierno que han realizado el contrato. El especialista del DAEE revisa que los estudiantes financiados y autofinanciados tengan su contrato confeccionado previamente.

- **Proceso de pago de contrato por parte de los estudiantes autofinanciados.**

Luego de haber sido confeccionado el contrato del estudiante este debe realizar los pagos correspondientes con dicho contrato. Para esto primeramente el estudiante realiza el pago en el departamento de finanzas de la UCLV. Consecutivamente el estudiante presenta el comprobante en la OCSA. El especialista de la OCSA revisa el comprobante de pago del estudiante. El especialista de la OCSA actualiza los pagos del estudiante en la base de datos, agregando el monto pagado y la fecha del pago.

- **Proceso de pago de contrato por parte de los estudiantes financiados por el gobierno.**

Los pagos de los estudiantes financiados por el gobierno se realizan mediante el MES, con este fin, se realiza una transferencia al departamento de contabilidad y finanzas de la UCLV mediante el MES. El departamento de contabilidad y finanzas de la UCLV envía información sobre la transferencia realizada por el MES al especialista de la OCSA. El especialista de la OCSA chequea la información. El especialista de la OCSA actualiza los pagos del estudiante en la base de datos, agregando el monto pagado y la fecha del pago.

- **Proceso de viajar por convenio con ANTEX (profesores y trabajadores).**

Una de las vías para realizar un viaje al exterior desde la universidad es mediante ANTEX, para esto, el especialista de la OCSA chequea que los profesores que van a viajar hayan sido aprobados en rectoría. De no estar aprobados en rectoría no puede viajar. De estarlo el especialista solicita a los profesores los documentos para confeccionar el expediente. El profesor envía los documentos al especialista. El especialista de la OCSA revisa si están completos los documentos. De no estarlo le solicita al profesor entregarlos nuevamente; si están correctos el especialista confecciona el expediente. El especialista de la OCSA envía a ANTEX la solicitud de la persona a viajar hacia Angola mediante ANTEX y a su vez el especialista entrega al MES el expediente de ANTEX. Luego ANTEX envía la respuesta al especialista de si aprobó o no la salida de la persona. De ser aprobado por ANTEX, este envía la fecha de salida y la fecha de regreso por ANTEX. El especialista de la OCSA adiciona la fecha de salida y de regreso por ANTEX en su base de datos. El especialista informa al profesor las fechas de salida y de regreso por ANTEX.

- **Proceso de viajar por concepto asistencia técnica exportada (CUBATECNICA)**

Otra de las vías para realizar un viaje es mediante CUBATECNICA, para esto el especialista de la OCSA chequea que la persona que pretende viajar haya sido aprobada en rectoría. De no estar aprobados en rectoría no puede viajar. De estarlo el especialista solicita a los profesores los documentos para confeccionar el expediente de CUBATECNICA. El profesor envía los documentos al especialista. El especialista de la OCSA revisa si están completos los documentos. De no estarlo le solicita a la persona entregarlos nuevamente; si están correctos el especialista confecciona el expediente. El especialista de la OCSA envía el expediente confeccionado para el Departamento de Exportación del Ministerio de Educación Superior (MES). El MES envía el expediente de la persona para CUBATECNICA, luego CUBATECNICA recibe el expediente y lo revisa. Posteriormente se firma el contrato con CUBATECNICA por parte del profesor.

- **Proceso de recibir visitas por paquete de agencia.**

La UCLV en ocasiones recibe visitas desde el exterior, todas estas visitas se gestionan mediante las agencias de turismo, para esto la agencia envía una solicitud de visita al especialista de la OCSA. El especialista de la OCSA revisa la solicitud, este acepta o rechaza la solicitud de visita mediante el paquete de agencia. De aceptarla el especialista de la OCSA

inserta en la base de datos la fecha de aceptación, fecha de visita y el monto a pagar. Acto seguido la agencia realiza la visita a la UCLV. Luego de producirse la visita, el especialista de la OCSA le solicita al departamento de contabilidad y finanzas de la UCLV la factura. El especialista registra en la base de datos la fecha de la solicitud de la factura. El departamento de contabilidad y finanzas de la UCLV entrega al especialista de la OCSA la factura, el especialista actualiza la base de datos insertando la fecha de entrega de la factura. El especialista de la OCSA envía la factura a la agencia y actualiza la base de datos insertando la fecha de entrega de la factura a la agencia. La agencia recibe la factura y realiza el pago a la DRI. El especialista de la OCSA actualiza la base de datos insertando la fecha de pago de la agencia.

- **Proceso de asociarse a una nueva red o asociación**

Como parte del proceso de internacionalización llevado a cabo por la UCLV, se crean nuevas asociaciones y con redes y asociaciones, para esto el especialista del departamento solicita a una red o asociación ser miembro de esta, respondiendo a las solicitudes y necesidades de los profesores y las facultades de la UCLV. La red o asociación solicita documentos a la UCLV. Luego el especialista del departamento envía documentación requerida. La red o asociación revisa la documentación, de estar incompleta o incorrecta se solicita nuevamente la información, de estar correcta, posteriormente la red o asociación gestiona la membresía de la universidad. El especialista del departamento registra la nueva red o asociación en la base de datos.

- **Proceso de recepción de visitas**

La facultad o centro de la UCLV envía carta a la DRI-UCLV informando sobre la visita a llevarse a cabo. La DRI chequea los datos correspondientes a la solicitud de visita, de no poder ser aceptada la visita, se informa a la facultad o centro el rechazo de la visita; de ser aceptada, el especialista del departamento confecciona una carta con el fin de permitir el acceso de los visitantes a la UCLV y la entrega en el punto de control de la puerta de la UCLV y se registra en la base de datos los datos correspondiente a la visita.

- **Proceso de recepción de estudiantes de intercambio (becarios) por convenio**

Las solicitudes de recepción de estudiantes extranjeros que desean desarrollar pasantías por la modalidad de intercambio académico (exentas de pago de colegiatura) deberán ser

presentadas a la DRI-UCLV. Luego el especialista del departamento de colaboración internacional verifica que existan convenios específicos rubricados para la realización de intercambio académico con la universidad que lo solicita (se tiene conocimiento de antemano de los convenios existentes). La universidad solicitante envía la información con los candidatos seleccionados y en periodo y la carrera que pretenden cursar. De existir el convenio con la institución, el especialista revisa si existe disponibilidad en las facultades para aceptar a la cantidad de estudiantes que solicitan para cada carrera. De existir disponibilidad en las facultades se aceptan los estudiantes, de no poder aceptar la cantidad de estudiantes solicitados para cada facultad, se les envía a la universidad solicitante un documento diciendo la cantidad de estudiantes que la UCLV puede aceptar por carrera.

Luego de ser enviado el documento con la cantidad de estudiantes que pueden ser aceptados, la universidad solicitante realiza una selección de estudiantes de acuerdo a los que puede aceptar la UCLV y envía la solicitud nuevamente. Se aprueba por parte del especialista de la UCLV el convenio de movilidad y se registra por parte de del especialista toda la información necesaria para procesar la estadía del estudiante.

#### • **Proceso de Realizar convenio**

Se recibe una solicitud de firma de convenio por parte de las facultades de la UCLV o de universidad extranjera a través de una carta de intención dirigida a la directora. La directora aprueba la solicitud. A partir de la aprobación de la solicitud empiezan las negociaciones, luego se intercambian las proformas de convenio entre nuestra universidad y la universidad extranjera. El departamento jurídico del rectorado revisa y dictamina el documento a firmar. El documento es firmado por ambos rectores. Se registra el convenio en la base de datos.

Teniendo en cuenta la forma en que se ejecutan los procesos y la problemática que presenta la DRI-UCLV, se hace necesario escoger las tecnologías que se adecúen correctamente para dar tratamiento a la problemática en cuestión. Para esto explicaremos algunas de las tendencias y tecnologías actuales ajustables para alcanzar una solución óptima a nuestro problema.

## 1.2 Tendencias y tecnologías actuales

La web de hoy en día ha conferido un flujo de comunicación a escala global sin precursores en la historia de la humanidad. Gracias a su representación virtual, la información en la web puede ser buscada de manera más fácil y eficientemente que en cualquier medio físico, siendo mucho más rápido de lo que una persona podría recabar por sí misma a través de otro medio de comunicación.

Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. La Web 2.0, caracterizada por la interoperabilidad, la colaboración y el diseño centrado en el usuario, se ha hecho la palabra de moda para describir entornos disponibles en la Internet, como blogs y sitios de redes sociales. En el reconocimiento de estas tendencias, incluso los gobiernos de países ya comienzan a mirar muy de cerca a estas tecnologías y a las comunidades en línea para utilizarlas en el diseño de servicios de calidad (Aikins, 2012 ).

Expuesto lo anterior se vuelve imprescindible elaborar un entorno web que contribuya a resolver el problema planteado y que aproveche los beneficios que aportan las tecnologías web.

### 1.2.1 Software Libre

Software libre (en inglés *free software*) es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El usuario tiene la libertad de usarlo para cualquier propósito, de estudiar su funcionamiento, distribuir copias; hacerle los cambios que estime necesarios ya sea para su beneficio propio o para una comunidad y hacer públicos esos cambios.

El software libre suele estar disponible gratuitamente o a precio de coste de distribución, pero no hay que asociarlo necesariamente a software gratuito (en inglés *freeware*), pues puede ser vendido comercialmente y aun conservar su carácter de libre. Análogamente, el *freeware* incluye en algunas ocasiones el código fuente; no obstante, este tipo de software no es libre en el mismo sentido que el software libre, a menos que se garanticen los derechos de modificación y redistribución de las versiones modificadas del programa.

El adjetivo «libre» en el software libre hace referencia a la libertad: libertad del usuario para ejecutar, modificar y redistribuir software. El software libre contribuye al saber humano, al contrario que el software propietario. Por este motivo, las universidades deberían fomentar el software libre, para hacer una aportación al progreso del conocimiento humano, del mismo modo que deben animar a científicos y académicos a publicar sus obras (Stallman, 2004).

#### 1.2.1.1 ¿Por qué utilizar Software Libre?

Desde hace varias décadas los países poderosos han tratado de monopolizar y privatizar las tecnologías de punta restringiéndoles a los pobres el acceso y el beneficio que representa su uso. Como consecuencia, el desarrollo tecnológico de los de abajo, ya de por sí disminuido, encuentra una nueva traba poco menos que insalvable en el pago de costosísimas licencias. El software libre presenta numerosas características ventajosas que son, en gran medida, consecuencia de las libertades que ofrece y esquivan, entre otras, esa cota injusta. Algunas de ellas son:

- Libre: Existe libertad para usar, modificar, regalar o vender los programas de software libre.
- La copia es legal: Es legal repartir software libre a otras personas lo que evita en gran medida los problemas de la piratería. Si lo natural es compartir los programas con otras personas, con software libre es además legal.
- Abierto: se puede usar el código de los programas y modificarlo.
- Colaborativo: el modelo de desarrollo de software libre es colaborativo y participativo. Todo se puede modificar o criticar.
- Ayuda: existen innumerables grupos de usuarios que se ayudan entre sí a través de Internet.
- Auditable: el software libre se puede inspeccionar al disponer de su código fuente.

Corrección más rápida y eficiente de fallos: El funcionamiento e interés conjunto de la comunidad permite solucionar más rápidamente los fallos de seguridad en el software libre, algo que desgraciadamente en el software propietario es más difícil y costoso. Cuando se notifica a las empresas propietarias del software, éstas niegan inicialmente la existencia de dichos fallos por cuestiones de imagen, y cuando finalmente admiten la existencia de esos errores tardan meses hasta proporcionar los parches de seguridad.

- Economía: El bajo o nulo costo de los productos libres permite a las instituciones ampliar sus infraestructuras sin que se vean mermados sus intentos de crecimiento por no poder hacer frente al pago de las licencias.



- Libertad de uso y redistribución: Las licencias de software libre actuales permiten la instalación del software tantas veces y en tantas máquinas como el usuario desee.
- Independencia tecnológica: El código fuente libre permite el desarrollo de nuevos productos sin la necesidad de desarrollar todo el proceso partiendo de cero.
- Fomento de la libre competencia al basarse en servicios y no licencias: Uno de los modelos de negocio que genera el software libre es la contratación de servicios de atención al cliente. Este sistema permite que las compañías que den el servicio compitan en igualdad de condiciones al no poseer la propiedad del producto del cual dan el servicio, lo que provoca un cambio vertical positivo en cuanto a atención al cliente y contratación de empleados, en contraposición a sistemas mayoritariamente sostenidos por la venta de licencias y desatención.
- Soporte y compatibilidad a largo plazo: Más que una ventaja del software libre es una desventaja del software propietario. Al vendedor, una vez que ha alcanzado el máximo de ventas que puede realizar de un producto, no le interesa que sus clientes continúen con él. La opción es sacar un nuevo producto, producir software que emplee nuevas tecnologías solo para éste y no dar soporte para la resolución de fallos al anterior tratando de hacerlo obsoleto por todos los medios, pese a que el primero pudiera cubrir perfectamente las necesidades de muchos de los usuarios. La elección del software libre evade el problema.
- Formatos estándar: los formatos estándar permiten una interoperabilidad más alta entre sistemas, evitando incompatibilidades.
- Sistemas sin puertas traseras y más seguros: El acceso al código fuente permite a las empresas de seguridad de todo el mundo puedan auditar los programas, por lo que la existencia de puertas traseras es ilógica ya que se pondría en evidencia y contraviene el interés de la comunidad que es la que lo genera.
- Sistema en expansión: Las ventajas especialmente económicas que aportan las soluciones libres a muchas empresas y las aportaciones de la comunidad, han permitido un constante crecimiento del software libre hasta superar, en ocasiones (por ejemplo, servidores Web) al mercado propietario. El software libre ya no es una promesa, es una realidad y se utiliza en sistemas de producción por algunas de las empresas tecnológicas más importantes como IBM, SUN Microsystems, Google y Hewlett-Packard.

La alternativa del software libre evita la dependencia tecnológica y ha venido a elevar los niveles de desarrollo de una comunidad ya grande que se reproduce en su seno. Cada día es usada con mayor frecuencia y amplía sus aristas sociales hacia ramas insospechadas. Sin la obligación de pagar las costosas licencias impuestas por los monopolios y trabajando por un interés común, la comunidad de quienes la asumen como paradigma consigue consolidarse y extender sus raíces.

El software libre es una alternativa para la excesiva globalización corporativa que tiene la potencialidad de acabar con la monopolización de la tecnología para siempre y puede llevar a un nuevo modelo de economía en este sentido. Nuestro país desde hace tiempo comparte esta visión.

### 1.2.2 Framework

Según (Riehle, 2000), la palabra inglesa "*framework*" (marco de trabajo) define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

En el desarrollo de software, un *framework* o infraestructura digital, es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software. Típicamente, puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio, y provee una estructura y una especial metodología de trabajo, la cual extiende o utiliza las aplicaciones del dominio.

En general, con el término *framework*, nos estamos refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un *framework* se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta (Gutiérrez, 2014).

### 1.2.2.1 Frameworks para aplicaciones Web

Estos *framework* están diseñados para apoyar el desarrollo de sitios web dinámicos, aplicaciones web y servicios web. Este tipo de *framework* intenta aliviar el exceso de carga asociado con actividades comunes usadas en desarrollos web. Por ejemplo, muchos *framework* proporcionan bibliotecas para acceder a bases de datos, estructuras para plantillas y gestión de sesiones, y con frecuencia facilitan la reutilización de código. Algunos de los *framework* más conocidos y usados son:

- Symfony: Symfony es un completo *framework* diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web (Fabien Potencier).
- Laravel: Es un *framework* para la programación en PHP. Mientras PHP se conoce por tener menos de una sintaxis deseable, Laravel es fácil de usar, fácil de desplegar, y puede encontrarse impulsando muchos de los modernos sitios web que se usan día a día. Laravel no sólo proporciona atajos útiles, herramientas, y componentes que ayudan a lograr el éxito en todos sus proyectos web bases, pero también apunta para arreglar alguna de las fallas de PHP. Laravel tiene una bonita semántica, y una sintaxis creativa que le permite mantenerse de pie entre un gran número de *frameworks* disponible para el lenguaje. Esto hace fácil de usar el PHP sin sacrificar el poder y la eficacia. Laravel es una gran opción para los proyectos amateurs y soluciones empresariales (Rees, 2012).
- Zend: Este *framework* es 100% orientado a objetos y sus componentes tienen un bajo acoplamiento por lo que se pueden usar en forma independiente. Gran rendimiento y robusta implementación en el Modelo Vista Controlador (MVC), da al usuario una fácil manipulación del código creado para reutilizarlo, código abierto, presenta un gran rendimiento. No es fácil de aprender, necesita de mucha configuración antes de usarse, presenta una estructura de archivos no definida (Sierra et al.).

### 1.2.2.2 Symfony como framework de trabajo

Symfony es un completo *framework* diseñado para optimizar el desarrollo de las aplicaciones web basado en el patrón Modelo Vista Controlador. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web (EGUILUZ, 2013).

Symfony está desarrollado completamente en PHP 5.3. Ha sido probado en numerosos proyectos reales y se utiliza en sitios web de comercio electrónico de primer nivel. Es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y Microsoft SQL Server. Se puede ejecutar tanto en plataformas \*nix (Unix, Linux, etc.) como en plataformas Windows (Fabien Potencier).

Fue diseñado para ajustarse a los siguientes requisitos:

- Fácil de instalar y configurar en la mayoría de plataformas.
- Independiente del sistema gestor de bases de datos. Su capa de abstracción y el uso de Propel, permiten cambiar con facilidad de SGBD en cualquier fase del proyecto.
- Utiliza programación orientada a objetos, de ahí que sea imprescindible PHP 5.
- Sencillo de usar, aunque es preferible para el desarrollo de grandes aplicaciones Web que para pequeños proyectos.
- Aunque utiliza MVC (Modelo Vista Controlador), tiene su propia forma de trabajo en este punto, con variantes del MVC clásico como la capa de abstracción de base de datos, el controlador frontal y las acciones.
- Basado en la premisa de “convenir en vez de configurar”, en la que el desarrollador sólo debe configurar aquello que no es convencional.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
- Preparado para aplicaciones empresariales y adaptables a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.

- Fácil de extender, lo que permite su integración con las bibliotecas de otros fabricantes.
- Una potente línea de comandos que facilitan generación de código, lo cual contribuye a ahorrar tiempo de trabajo.

**Las características más comunes para el desarrollo de proyectos web en Symfony según (Potencier and Zaninotto, 2007) son:**

- Permite la internacionalización para la traducción del texto de la interfaz, los datos y el contenido de localización.
- La presentación usa *templates* y *layouts* que pueden ser construidos por diseñadores de HTML que no posean conocimientos del *framework*.
- Los formularios soportan la validación automática, lo cual asegura mejor calidad de los datos en las bases de datos y una mejor experiencia para el usuario.
- El manejo de caché reduce el uso de banda ancha y la carga del servidor.
- La facilidad de soportar autenticación y credenciales facilita la creación de áreas restringidas y manejo de seguridad de los usuarios.
- El enrutamiento y las URLs inteligentes hacen amigables las direcciones de las páginas de la aplicación.
- Las listas son más amigables, ya que permiten la paginación, clasificación y filtraje automáticos.
- Los *plugins* proveen un alto nivel de extensibilidad.
- La interacción con AJAX es mucho más sencilla.

### 1.2.2.3 Symfony 2.6 como *framework* de trabajo a utilizar

Se va a hacer uso del *framework* de trabajo Symfony, esta vez en su versión 2.6, la cual proporciona algunas mejoras sobre versiones anteriores, esto permitirá un mejor desempeño del sistema a implementar.

Algunas de las características del *framework* de trabajo Symfony son las siguientes:

- Escalable: Symfony es infinitamente escalable si se disponen de los recursos necesarios. Yahoo! utiliza Symfony para programar aplicaciones con 200 millones de usuarios.
- Probado: Symfony ha sido probado con éxito durante años en varias aplicaciones gigantescas (*Yahoo! Answers*, *Dailymotion*, *delicious*) y en otros miles de sitios pequeños y medianos.

- Soporte: Symfony sigue una política de tipo LTS (*long term support*), por la que las versiones estables se mantienen durante 3 años sin cambios pero con una continua corrección de errores.
- Licencia: Symfony se publica bajo licencia MIT, con la que puedes desarrollar aplicaciones web comerciales, gratuitas y/o de software libre.
- Seguro: Symfony permite controlar hasta el último acceso a la información e incluye por defecto protección contra ataques XSS (en inglés *Cross-Site Scripting* o Secuencias de comandos en sitios cruzados) y CSRF (en inglés *Cross-Site Request Forgery* o Falsificación de Petición en Sitios Cruzados).
- Código: Desde su primera versión Symfony ha sido creado sólo para PHP 5, para obtener el máximo rendimiento de PHP y aprovechar todas sus características.
- Compromiso: Los creadores de Symfony no reciben las mayores ganancias del *framework*, sino de las aplicaciones que se desarrollan con él, por lo que les interesa aspectos como el rendimiento, la buena documentación y el soporte muy largo.
- Documentado: Symfony es el *framework* mejor documentado, ya que ha publicado cinco libros gratuitos de calidad y siempre actualizados. Además, toda la documentación está traducida al español.
- Calidad: Su código fuente incluye más de 9.000 pruebas unitarias y funcionales.
- Internacionalización: Symfony está traducido a más de 40 idiomas e incluye todas las herramientas necesarias para traducir fácilmente cualquier aplicación.

### 1.2.3 ¿Qué es PHP?

PHP es un lenguaje de script que se ejecuta del lado del servidor, se incluye en una página HTML normal. Por lo tanto, se puede comparar con otros lenguajes de script que se ejecutan según el mismo principio: ASP (*Active Server Pages*), JSP (*Java Server Pages*) o PL/SQL *Server Pages* (PSP) (HEURTEL, 2011).

A diferencia de un lenguaje como JavaScript, donde el código se ejecuta del lado del cliente (en el explorador), el código PHP se ejecuta del lado del servidor. El resultado de esta ejecución se incrusta en la página HTML, que se envía al navegador. Este último no tiene conocimiento de la existencia del procesamiento que se ha llevado a cabo en el servidor.

#### 1.2.3.1 ¿Por qué utilizar PHP?

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como “C” y “Perl”, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones. Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, DB2, Microsoft SQL Server, Firebird y SQLite. PHP tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, y puede interactuar con los servidores web más populares ya que existe en versión CGI, módulo para Apache e ISAPI.

#### 1.2.3.2 Ventajas y Desventajas de PHP

Algunas de las ventajas y desventajas que nos brinda el lenguaje PHP:

- Es un lenguaje multiplataforma, completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.
- El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite aplicar técnicas de programación orientada a objetos.
- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Tiene manejo de excepciones (desde PHP5).

Una de las desventajas fundamentales es que al ser un lenguaje interpretado se necesita tener el intérprete de PHP, y esto lleva al alto consumo de las aplicaciones, ya que son imposibles de compilar.

#### 1.2.4 Sistemas Gestores de Bases de Datos

Los sistemas gestores de bases de datos (SGBD), son un conjunto de programas que permiten crear y mantener una base de datos, asegurando su integridad, confidencialidad y seguridad (MySQL, 2008).

Definimos un Sistema Gestor de Bases de Datos o SGBD, también llamado DBMS (*Data Base Management System*) como una colección de datos relacionados entre sí, estructurados y organizados, y un conjunto de programas que acceden y gestionan esos datos. La colección de esos datos se denomina Base de Datos o BD, (*DB Data Base*).

#### 1.2.4.1 ¿Qué es PostgreSQL?

PostgreSQL es un sistema para el manejo de bases de datos relacionales basado en POSTGRES, Versión 4.21, desarrollada en la Universidad de California en el departamento de Informática Berkeley. POSTGRES abrió el camino a muchos conceptos que sólo se volvieron disponibles en algunos sistemas de base de datos comerciales tiempo después. PostgreSQL es un código fuente abierto descendiente del código original de Berkeley (Group, 2013).

Soporta en gran parte el estándar SQL y ofrece muchas características modernas como:

- Consultas complejas
- Llaves foráneas
- *Triggers*
- *Updates*
- Integridad transaccional

También, PostgreSQL puede extenderse por el usuario de muchas maneras, por ejemplo, agregando nuevos:

- Tipos de datos
- Funciones
- Operadores
- Agregar funciones
- Idiomas procedurales



Debido a la licencia libre, PostgreSQL puede ser usado, modificado, distribuido gratuitamente por cualquiera para cualquier propósito, sea privado, comercial, o académico.

#### 1.2.4.2 Principales características del PostgreSQL

A continuación se hace referencia a las principales características que nos brinda PostgreSQL:

- **Alta concurrencia**

Mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo *commit*. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

- **Amplia variedad de tipos nativos**

PostgreSQL provee nativamente soporte para:

- Números de precisión arbitraria.
- Texto de largo ilimitado.
- Figuras geométricas (con una variedad de funciones asociadas).
- Direcciones IP (IPv4 e IPv6).
- Bloques de direcciones estilo CIDR.
- Direcciones MAC.
- *Arrays*.

### 1.3 Conclusiones Parciales

En este capítulo se describe el estudio realizado sobre los procesos que ocurren en los diferentes departamentos de la DRI-UCLV con sus limitaciones, donde existe la necesidad de un sistema que permita una mejor gestión de la información y una mejor organización de la misma para la elaboración de reportes y la realización de búsquedas y filtrados. En función de la implementación de una aplicación web que respondiera a las necesidades de gestión de la información, se analizaron diferentes herramientas y métodos, se optó por utilizar el marco de trabajo Symfony acompañado del gestor de bases de datos PostgreSQL. Esta decisión se

basó en la robustez y flexibilidad que brindan estos sistemas además de ser software libre. Con este estudio se abre paso a los siguientes capítulos donde se abordan los procesos del negocio y del sistema.

## Capítulo II. Modelo del Negocio y Requisitos

En este capítulo se recoge todo lo relacionado con la implementación del negocio, la presentación de los principales diagramas, modelos y requisitos funcionales y no funcionales del negocio, además se consideran aspectos fundamentales de la arquitectura del sistema.

### 2.1 Modelo del negocio en la DRI-UCLV

Según (IVAR JACOBSON, 2000), un modelo de caso de uso del negocio describe los procesos de una empresa en términos de casos de uso del negocio y actores del mismo que se corresponden con los procesos del negocio y los clientes, respectivamente. El modelo de casos de uso del negocio presenta un sistema, desde la perspectiva de su uso, y esquematiza como proporciona valor a sus usuarios.

Para el sistema informático que se pretende implementar deben considerarse un conjunto de reglas y políticas que se establecen en el negocio.

En el capítulo anterior se hace referencia al modelo del negocio mediante una explicación detallada del funcionamiento de los procesos llevados a cabo en la DRI-UCLV. Partiendo de la explicación anteriormente expuesta se da paso a las reglas a tener en cuenta en dicho negocio.

#### 2.1.1 Reglas del negocio a considerar

A continuación se enumeran y se describen las reglas del negocio existentes en la Dirección de Relaciones Internacionales en la Universidad Central "Marta Abreu" de Las Villas.

- **RN1: Pasaporte**

El pasaporte se confecciona por primera vez siempre y cuando sea aprobada en rectoría la solicitud de salida de la persona.

- **RN2: Pagos de estudiantes autofinanciados**

Cada estudiante autofinanciado debe realizar los pagos de las cuotas correspondientes a su contrato. Estas cuotas siempre se pagarán antes de comenzar cada semestre (dando hasta una semana como máximo después de comenzado el semestre para realizar el pago) dependiendo del tipo de estudio ya sea un curso completo, una pasantía, etc.

- **RN3: Viajar por ANTEX**

Los profesores y trabajadores son los únicos que pueden viajar por ANTEX hacia Angola.

- **RN4: Validez de los Convenios**

En los convenios, a partir de la fecha de firma del convenio, este tendrá validez por cinco años, pasados estos 5 años el convenio caduca, a no ser que se tenga interés de continuarlo y ambas instituciones soliciten una prórroga.

- **RN5: Convenio Activo**

Un convenio se encuentra activo con un país o institución si se ha efectuado intercambio entre ambas partes en un periodo igual o menor de dos años.

- **RN6: Visa académica para estudiantes**

Cuando el estudiante viene por más de 30 días es necesario procesarle una visa académica para que pueda permanecer en el centro.

- **RN7: Acuerdos de Contratación.**

Los acuerdos de contratación de personal requieren un convenio con la institución solicitante de la contratación de dicho personal.

- **RN8: Pagos de visitas por paquete de agencia.**

Los pagos de las visitas realizadas por el paquete de agencia se llevan a cabo luego de concluida la visita al centro.

### 2.1.2 Actores del negocio

Los actores representan terceros fuera del sistema que colaboran con él. Los actores del sistema suelen corresponderse con los trabajadores del negocio y si algún actor del negocio va a interactuar con el sistema pasa también a ser actor del mismo. En la Tabla 1 se hace referencia a los actores del negocio con una respectiva descripción de las acciones que realizan sobre este.

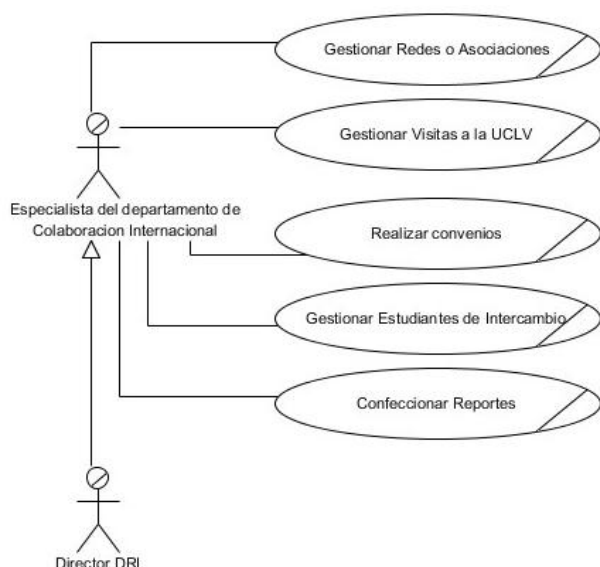
*Tabla 1. Actores del negocio*

<i>Actor del negocio</i>	<i>Descripción</i>
--------------------------	--------------------

Personal de la UCLV (estudiante, profesor, investigador o trabajador de la UCLV)	Persona que solicita realizar un viaje mediante la UCLV hacia diferentes países e instituciones.
Estudiante extranjero (becario por concepto de intercambio bilateral, auto financiado y financiado por el gobierno)	Persona que solicita comenzar estudios en la UCLV en alguna de sus modalidades, ya sea un curso corto, una pasantía, curso completo, entre otras.

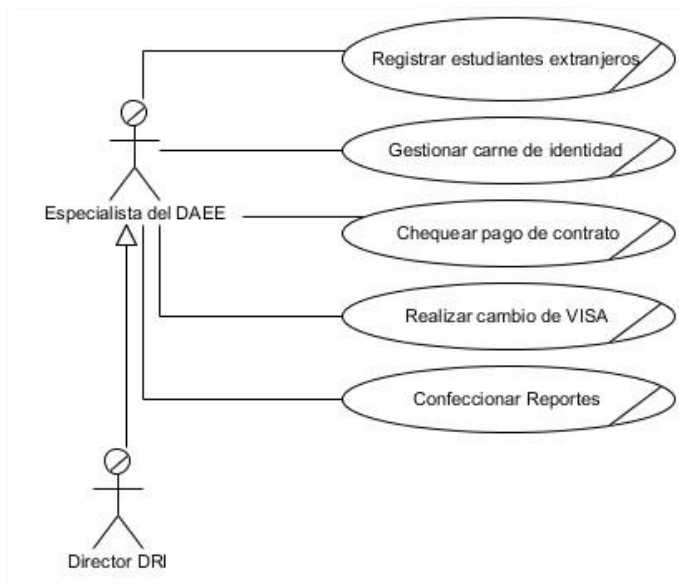
### 2.1.3 Diagrama de casos de uso del negocio

En esta sección se muestran los principales diagramas de casos de uso del negocio. En la Figura 1 se presentan los casos de uso correspondientes al especialista del departamento de colaboración, dentro de ellos se encuentra el caso de uso gestionar estudiantes de intercambio, en este caso uno de los más importantes para este actor.



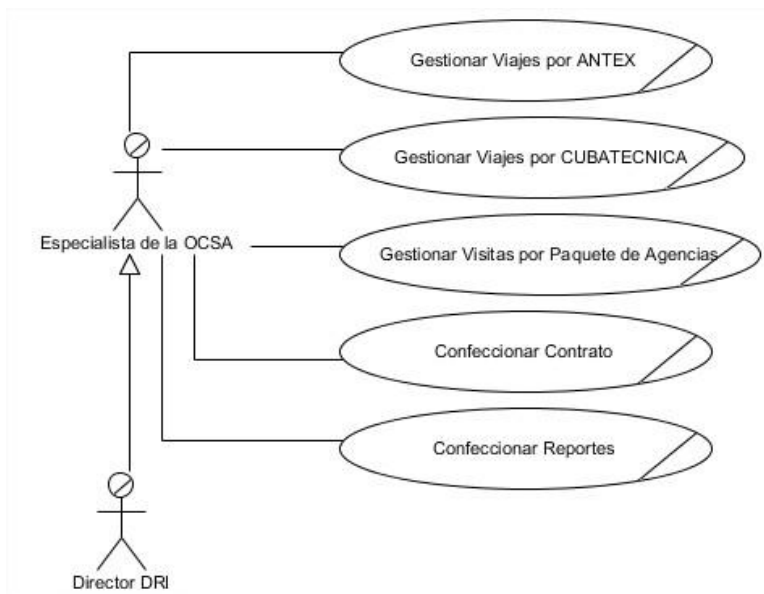
*Figura 1 Caso de uso del negocio correspondiente al especialista del departamento del Colaboración Internacional*

En la Figura 2 se presenta los casos de uso tales como registrar un estudiante extranjero, entre otros, correspondiente al especialista del departamento de estudiantes extranjeros.



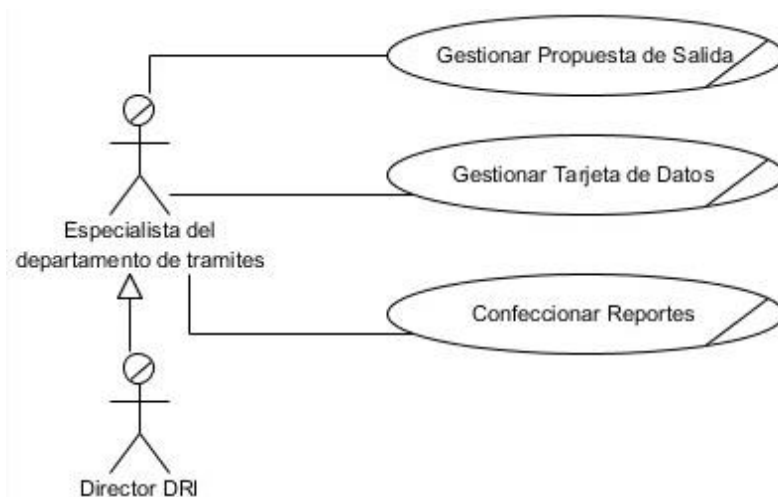
*Figura 2 Caso de uso del negocio correspondiente al especialista del DAEE*

En la Figura 3 se presentan los casos de uso correspondientes al especialista del departamento de estudiantes extranjeros mostrando los casos de uso asociados a dicho actor.



*Figura 3 Caso de uso del negocio correspondiente al especialista de la OCSA*

En la Figura 4 se presentan los casos de uso correspondientes al especialista del departamento de trámites, siendo el CU gestionar propuesta de salida el más fundamental en lo que respecta a dicho actor.



*Figura 4 Caso de uso del negocio correspondiente al especialista del departamento de trámites.*

En la Tabla 3 se muestra una detallada reseña de algunos de los casos de uso del negocio considerado más importantes.

#### 2.1.4 Trabajadores del negocio en la DRI-UCLV

En esta sección se hace referencia a los trabajadores del negocio ofreciendo una breve descripción de las funcionalidades correspondientes a cada uno, todo esto queda expresado en la Tabla 2 en la cual se muestran los datos antes mencionados.

*Tabla 2. Relación de los trabajadores del negocio.*

Trabajador del negocio	Descripción
Especialista del departamento de trámites	Persona que se encarga de gestionar las propuestas de salida del personal de la UCLV, procesar las solicitudes de pasaporte y de confeccionar las tarjetas de datos.
Especialista del DAEE	Persona que se encarga de registrar la entrada a todos los estudiantes extranjeros al centro, gestionar el carné de identidad, también realizan el cambio de VISA, chequean también que el pago del contrato de los estudiantes autofinanciados y financiados de gobierno sea llevado a cabo en el tiempo correspondiente.

Especialista de la OCSA	Persona que se encarga de procesar los viajes por ANTEX y CUBATECNICA, recibir visitas por paquete de agencia y confeccionar el contrato a los estudiantes que solicitan realizar estudios en la modalidad de estudiantes autofinanciados en la UCLV.
Especialista del departamento de Colaboración Internacional.	Persona que se encarga de gestionar los procesos a realizar para asociarse con nuevas redes o asociaciones, también se encargan de la recepción de visitas al centro, realizar convenios con otras instituciones y la admisión de estudiantes de intercambio por convenio.

### 2.1.5 Casos de uso del negocio

En la Tabla 3 se explican algunos de los casos de uso del negocio fundamentales.

*Tabla 3 Casos de uso del negocio fundamentales*

Caso de uso del negocio	Descripción
Gestionar Propuestas de Salidas	Estas propuestas de Salidas contienen toda la información necesaria para realizar un viaje fuera del país, dicha propuesta pueden ser entregadas al especialista del Departamento de tramites ya sea por un profesor, facultad o área.
Gestionar tarjeta de datos	Cuando una persona es aprobada en rectoría para realizar un viaje se le confecciona una tarjeta de datos donde se recogen además de sus datos personales, el país al cual va a viajar, la institución y el tiempo de estancia y el año en que se realizó el viaje.
Gestionar Estudiantes de Intercambio	Cuando se desean realizar intercambios estudiantiles ya sea nuestra institución o alguna institución extranjera la que lo solicite, se comienza la gestión de este proceso registrando todos los datos del estudiante a intercambiar.
Confeccionar Reportes	Cada departamento de la DRI-UCLV debe hacer entrega de informes sobre informaciones específicas, ya sea mostrar que Propuestas de



	Salida estén pendientes o que estudiantes extranjeros están en la institución por concepto de intercambio, entre otras.
Registrar estudiantes extranjeros	El Departamento de Atención a estudiantes Extranjeros es el encargado de llevar el control de los estudiantes que ingresan a la universidad ya sea por concepto de estudiantes por convenio, autofinanciados o financiados por el gobierno.
Confeccionar contrato	Cuando un estudiante extranjero, en este caso auto financiado o financiado por el gobierno solicita estudiar en la UCLV se le formaliza un contrato de pago, requisito indispensable para poder ejercer sus estudios en la UCLV.

#### 2.1.6 Actores del sistema a automatizar

En la Tabla 4 se describen cuáles son los actores del sistema y las funciones que realizan sobre el mismo.

*Tabla 4 Actores del sistema a automatizar*

Actores del Sistema	Descripción
Especialista del departamento de trámites	Este actor atiende las propuestas de salida del personal de la UCLV, procesar las Solicitudes de Pasaporte, confeccionar las tarjetas de datos, autenticarse y gestionar reportes.
Especialista del DAEE	Este actor se encarga de registrar la entrada a todos los estudiantes extranjeros al centro, chequear el pago del contrato de los estudiantes autofinanciados y financiados de gobierno, autenticarse y gestionar reportes.
Especialista de la OCSA	Este actor se encarga de procesar los viajes por ANTEX y CUBATECNICA, recibir visitas por paquete de agencia y confeccionar el contrato a los estudiantes que solicitan realizar estudios en la modalidad de estudiantes autofinanciados en la UCLV, autenticarse y gestionar reportes.

Especialista del departamento de Colaboración Internacional.	Este actor se encarga de gestionar los procesos a realizar para asociarse con nuevas redes o asociaciones, también se encargan de la recepción de visitas al centro, realizar convenios con otras instituciones y la admisión de estudiantes de intercambio por convenio, autenticarse y gestionar reportes.
Director de la DRI	Este actor hereda todas las funcionalidades de todos los especialistas de los diferentes departamentos (Colaboración Internacional, Trámites, OCSA, DAEE).
Administrador	Este actor es el encargado de gestionar los usuarios que van a interactuar con el sistema, para realizar esa acción necesita estar previamente autenticado.

### 2.1.7 Requisitos funcionales.

Según (SOMMERVILLE, 2002), los requisitos funcionales son declaraciones de los servicios que proveerá el sistema, de la manera en que éste reaccionará a entradas particulares y de cómo se comportará en situaciones particulares. En algunos casos, los requerimientos funcionales de los sistemas también declaran explícitamente lo que el sistema no debe hacer. Para la implementación de nuestro sistema se han tenido en cuenta los siguientes requisitos funcionales:

- **RF1: Gestionar usuario**

RF1.1: Insertar usuario.

RF1.2: Modificar usuario.

RF1.3: Eliminar usuario.

- **RF2: Autenticar usuario**

- **RF3: Gestionar visitas por paquete de agencias.**

RF3.1: Insertar visita por agencias.

RF3.2: Modificar visita por agencias.

- **RF4: Gestionar propuesta de Salida.**

RF4.1: Insertar Propuesta de Salida.

RF4.2: Modificar Propuesta de Salida.

- **RF5: Gestionar Pasaporte.**

RF5.1: Insertar Datos de Pasaporte.

RF5.2: Modificar Datos de Pasaporte.

- **RF6: Gestionar Tarjeta de Datos.**

RF6.1: Crear tarjeta de datos.

RF6.2: Modificar tarjeta de datos.

- **RF7: Confeccionar Reportes.**

- **RF8: Registrar estudiantes extranjeros.**

RF8.1: Insertar datos del estudiante extranjero.

RF8.2: Modificar datos del estudiante extranjero.

- **RF9: Gestionar Viajes por ANTEX.**

RF9.1: Insertar datos del profesor.

RF9.2: Modificar datos del profesor.

- **RF10: Gestionar Viajes por CUBATECNICA.**

RF10.1: Insertar datos de la persona.

RF10.2: Modificar datos de la persona.

- **RF11: Confeccionar Contrato.**

RF11.1: Insertar datos del contrato.

RF11.2: Modificar datos del contrato.

- **RF13: Gestionar Redes y Asociaciones.**

RF13.1: Insertar datos de redes o asociaciones.

RF13.2: Modificar datos de redes o asociaciones.

- **RF14: Gestionar Visitas a la UCLV.**

RF14.1: Insertar datos de la visita.

RF14.2: Modificar datos de la visita.

#### 2.1.8 Requisitos no funcionales.

Los requisitos no funcionales, según (SOMMERVILLE, 2002) “son restricciones de los servicios o funciones ofrecidas por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo, estándares, etcétera”.

- **RNF1: Usabilidad.**

Diseño: Las interfaces serán diseñadas de manera tal que le facilite el acceso al usuario permitiéndole entender con mayor claridad el contenido que se desea dar a conocer.

- **RNF2: Rendimiento.**

Tiempo de respuesta: El tiempo de respuesta no debe exceder los 5 segundos ante las solicitudes del usuario.

- **RNF3: Restricciones de diseño.**

Sistema de Gestión de Bases de Datos: Se utilizará como gestor de base de datos PostgreSQL.

Lenguaje de Programación: Se necesita PHP en su versión 5.5.12 o superior.

- **RNF4: Ayuda y Documentación.**

Manuales de usuario y Capacitaciones: *Se brindará un manual de ayuda que documente cómo trabajar de forma adecuada con el software.*

- **RNF5: Interfaz**

**RNF5.1:** Elementos gráficos distintivos: La interfaz contará con elementos gráficos representativos y menús desplegables que faciliten su utilización.

**RNF5.2:** Logo de la empresa: La interfaz debe tener incluido el logo de la empresa como un término de identificación.

- **RNF6: Requisitos de Portabilidad y Software**

Compatibilidad con todos los navegadores: El sistema deberá ser capaz de ejecutarse de forma correcta en todos los navegadores.

- **RNF6: Seguridad**

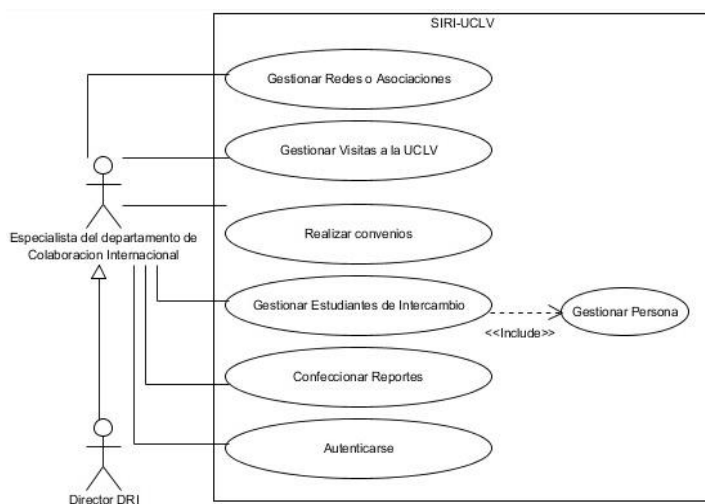
Seguridad: Se debe garantizar un control estricto sobre la seguridad de la información teniendo en cuenta el establecimiento de niveles de acceso. No se deben permitir accesos sin autorización al sistema. Además se debe definir una política de usuarios con roles y privilegios diferentes que garantice que la información pueda ser consultada de acuerdo al nivel de privilegios que puedan tener determinados grupos de usuarios.

Es de suma importancia garantizar la integridad de los datos que se almacenen en el servidor. La información almacenada deberá ser consistente y se utilizarán validaciones que limiten la entrada de datos irreales y mecanismos de vuelta atrás en procesos críticos que terminen abruptamente y produzcan estados inconsistentes de la información. Esta deberá estar disponible a los usuarios en todo momento, limitada solamente por las restricciones que estos tengan de acuerdo a la política de seguridad del sistema.

#### 2.1.9 Diagrama de Casos de Uso del Sistema.

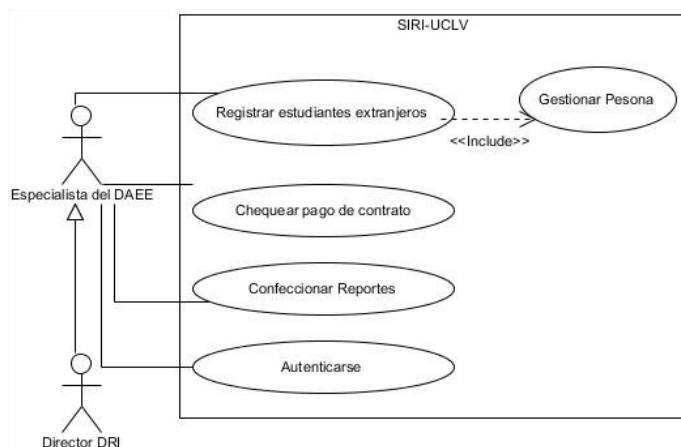
En las siguientes figuras se muestran los diagramas de actores y casos de uso para cada uno de los actores del sistema.

En la Figura 5 se muestran los casos de uso correspondientes al especialista del departamento de colaboración internacional, siendo el caso de uso gestionar estudiantes de intercambio uno de los más destacados ya que mediante esta vía se les brinda a los estudiantes de la UCLV la oportunidad de cursar estudios en universidades extranjeras.



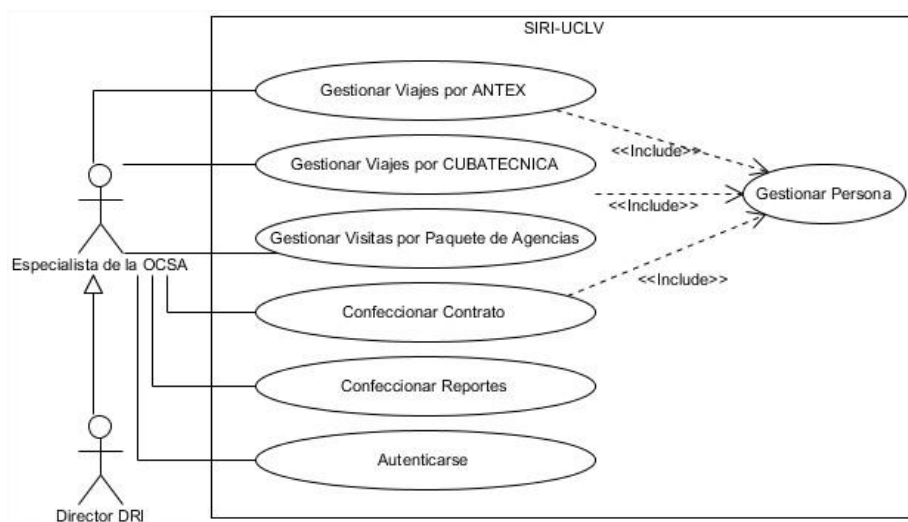
*Figura 5 Caso de uso del sistema correspondiente al especialista del departamento de Colaboración Internacional*

En la Figura 6 se muestran los casos de uso correspondientes al especialista del DAEE, teniendo como uno de los casos de uso más destacados el de registrar a los estudiantes extranjeros que comienzan en la UCLV en alguna de las modalidades de estudios ofertadas.



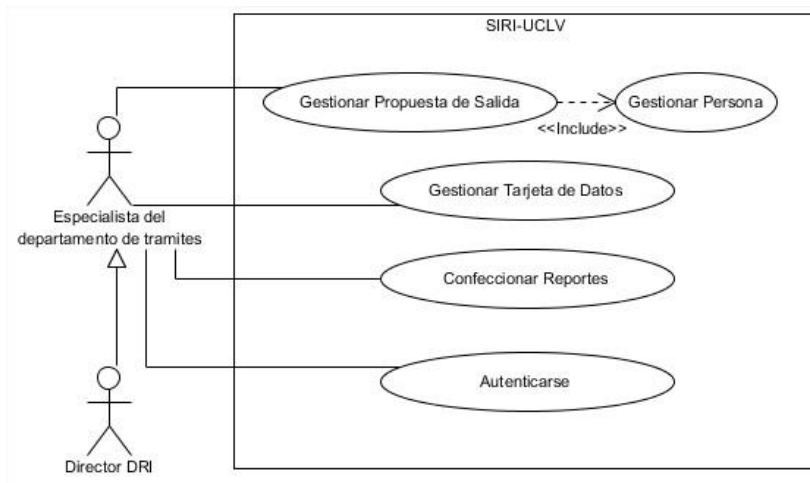
*Figura 6 Caso de uso del sistema correspondiente al especialista del DAEE*

En la Figura 7 se brindan los casos de uso correspondientes al especialista de la OCSA, teniendo dentro de sus casos de uso el de confeccionar contrato a los estudiantes que soliciten ejercer estudios en la UCLV, ya sean autofinanciados o financiados por el gobierno.



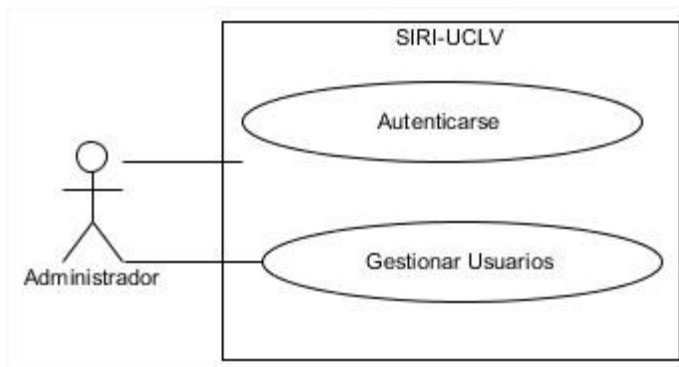
*Figura 7 Caso de uso del sistema correspondiente al especialista del departamento de la OCSA*

En la Figura 8 se brindan los casos de uso correspondientes al especialista de la oficina de trámites, el cual tiene dentro de sus responsabilidades gestionar todas las propuestas de salida presentadas por profesores, estudiantes y trabajadores del centro de estudios superiores.



*Figura 8 Caso de uso del sistema correspondiente al especialista del departamento de trámites*

En la Figura 9 se brindan los casos de uso correspondientes al administrador del sitio, dicho administrador va a tener dentro de sus funcionalidades la de manejar toda la gestión de los usuarios que van a hacer uso del sistema.



*Figura 9 Caso de uso del sistema correspondiente al administrador*

## 2.1.10 Descripción de los casos de uso del Sistema (Significativos)

A continuación se muestra una descripción de los casos de uso del sistema más significativos, además de los actores correspondientes a cada caso de uso. Esta descripción nos ayuda a entender cómo se realizan estas actividades a la hora de operar con el software. En la Tabla 5 y Tabla 6 se describen los casos de usos gestionar propuesta de salida y gestionar persona respectivamente.

*Tabla 5 Descripción del CUS Gestionar propuesta de salida*

Caso de Uso del Sistema	Gestionar Propuesta de Salida
Actores	Especialista del Departamento de Trámites, Director de la DRI
Propósito	Insertar una propuesta de salida
Resumen	El caso de uso del sistema se inicia cuando el Especialista de Trámites adiciona una nueva propuesta de salida en la Diáspora.
Precondiciones	El Especialista de Trámites debe estar autenticado en el sistema.
Descripción	



Flujo normal de los eventos	
Acción del Actor	Respuesta del sistema
<p>1. El especialista de Trámites elige la opción Crear proceso de viaje (A).</p> <p>3. El especialista de Trámites inserta los datos en el formulario (todos los valores del 1 al 15), los campos encerrados en los recuadros azules no son campos obligatorios.</p> <p>4. El especialista de Trámites escoge la opción Adicionar (17).</p>	<p>2. El sistema muestra la ventana para la inserción de los datos (B).</p> <p>5. El sistema revisa que no existan campos obligatorios (1 al 15) vacíos y que los datos insertados sean correctos (B).</p> <p>6. El sistema guarda los datos en la base de datos.</p> <p>7. El sistema muestra la vista de los datos del proceso de viaje que acaba de ser insertado (C).</p>
Otra Sección	
Presionar Botón Cancelar	
<p>4. El especialista de Trámites escoge la opción Cancelar (16).</p>	<p>5. El sistema regresa a la pantalla inicial (A).</p>
Flujo Alternativo de los Eventos	
Mensajes de error por presentar campos vacíos o datos incorrectos	
<p>4. El especialista de Trámites escoge la opción Adicionar (17).</p>	<p>5. El sistema revisa que no existan campos obligatorios vacíos y que los datos insertados sean correctos.</p> <p>6. El sistema muestra mensaje de error por la existencia de campos vacíos o con datos incorrectos.</p> <p>7. El sistema regresa a la pantalla inicial (A).</p>

*Tabla 6 Descripción del CUS Gestionar persona*

Caso de Uso del Sistema	Gestionar Persona
-------------------------	-------------------

Actores	Especialista del DAEE, Especialista del Departamento de Trámites, Especialista de la OCSA, Especialista del Departamento de Colaboración Internacional y Director del DRI.
Propósito	Almacenar datos de las personas, dichos datos luego serán usados en las diferentes acciones a ejecutar por el sistema.
Resumen	El caso de uso del sistema se inicia cuando alguno de los especialistas de los departamentos (DAEE, OCSA, Trámites, Colaboración Internacional) o el director de la DRI insertan los datos correspondientes a una persona.
Precondiciones	Se debe estar autenticado en el sistema para realizar cualquier acción.
Descripción	

Lista de Personas 1 **Crear persona**

Lista de Personas 2

100 ▼

3	4	5	6	7	8	9
CI	Nombre	Apellidos	Nacimiento	Sexo	Tipo	Acciones
90232323245	Pedro	Perez Perez	1995-12-31	Hombre	Estudiante	Mostrar Tarjeta de datos
91082735349	Alejandro	Bermudez Espinosa	1991-08-27	Hombre	Estudiante	Mostrar Tarjeta de datos
99999999999	Gertrudis	Albert Quintana	1983-01-30	Mujer	Profesor	Mostrar Tarjeta de datos

Mostrando 1 a 3 de 3 entradas

10 **1** Next →

Escoja el tipo de persona B

Tipo

1

2 **Crear**

Profesor  
Profesor  
Estudiante



Frank Pedro Gutierrez Quintana

Tarjeta de datos

Eliminar

Datos básicos

Nombre: Frank Pedro Gutierrez Quintana

Ci: 91998137859

Fecha de nacimiento: 1991-09-01

Ciudadania: Cubana

Estado Civil: Soltero

Telefono: 206896

Telefono del trabajo: 202025

Celular: 53805543

Email: alfonso@uclv.cu

Sexo: Masculino

Padre: Raul

Madre: Belkis

Estatuta: 1.76

Color de Ojos: Pardo

Color de Piel: Blanca

Color de Cabello: Negro

3

Dirección

D

2

1

Pais: Cuba

Provincia:

Municipio:

Calle: Ruperto Perez

Entre Calle 1: Marti

Entre Calle 2: Pedro Pedrasa

Numero de Casa: 23

Carretera:

Km:

Apartamento:

Cpa:

Finca:

Localidad:

Circunscripcion:

### Flujo Normal de los procesos

Acción del Actor	Respuesta del sistema
<p>1. El especialista de Trámites elige la opción Crear Persona (A opción 1).</p> <p>3. El especialista de Trámites escoge el tipo de persona a crear (B) (1) y oprime el botón Crear (B) (2).</p> <p>5. El especialista de Trámites inserta los datos en el formulario (C) (todos los valores del 1 al 42).</p> <p>6. El especialista de Trámites escoge la opción Adicionar (C) (44).</p>	<p>2. El sistema muestra la ventana para escoger el tipo de persona que se va a crear (B).</p> <p>4.El sistema muestra la ventana donde se van a insertar los datos(C)</p> <p>7.El sistema revisa que no existan campos obligatorios vacios y que los datos insertados sean correctos (C).</p> <p>8.El sistema guarda los datos en la base de datos.</p> <p>9.El sistema muestra la vista de los datos de la nueva persona que acaba de ser insertada (D).</p>
Otra Sección	
Presionar Botón Cancelar	
6. El especialista de Trámites escoge la opción Cancelar (C) (43).	7. El sistema regresa a la pantalla inicial (A).

46

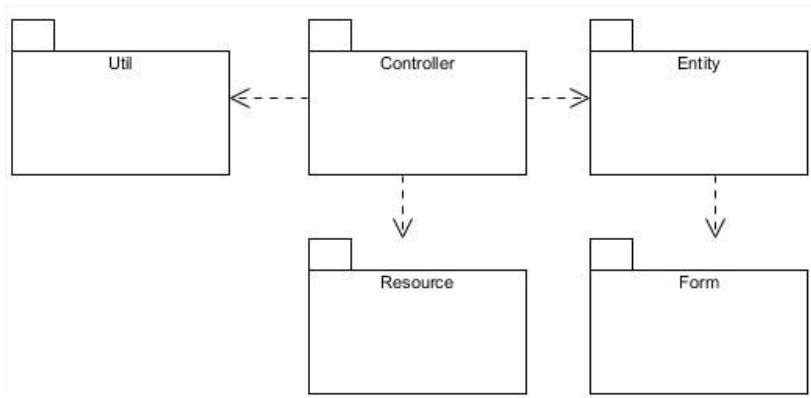
Flujo Alternativo de los Eventos	
Mensajes de error por presentar campos vacíos o datos incorrectos	
6. El especialista de Trámites escoge la opción Adicionar (C) (44).	<p>7.El sistema revisa que no existan campos obligatorios vacíos y que los datos insertados sean correctos.</p> <p>8.El sistema muestra mensaje de error por la existencia de campos vacíos o con datos incorrectos.</p> <p>9.El sistema muestra nuevamente la pantalla de inserción de datos mostrando un mensaje en el lugar donde se encuentra el error (C).</p>

#### 2.1.11 Paquetes del software

Los diagramas de paquetes son un tipo de modelado que muestra cómo se divide en agrupaciones lógicas un sistema, de acuerdo a las relaciones de interdependencia que presentan dichas agrupaciones. Por tanto, a través de los diagramas de paquetes se puede descomponer, para la fácil comprensión de la estructura de una aplicación, la jerarquía lógica que presenta el sistema. Las agrupaciones lógicas que presenta la aplicación desarrollada en este trabajo son las siguientes:

- **Controller:** Es el encargado de gestionar toda la lógica del software para su correcto funcionamiento.
- **Entity:** Son las clases del sistema que se transformarán en las tablas de la base de datos.
- **Resources:** En este paquete se encuentra todo lo relacionado con las vistas y las rutas de la aplicación.
- **Form:** Contiene todos los formularios a utilizar en el sistema.
- **Util:** Almacena aquellos códigos que presentan un alto grado de reutilización.

En la siguiente Figura 12 se muestra el diagrama de paquetes de software correspondiente al sitio web.



*Figura 10 Diagrama de paquetes*

## 2.2 Conclusiones Parciales

En este capítulo se abordó todo lo referente al modelado del negocio, dígame diagramas de casos de uso, actores, reglas y requisitos tanto funcionales como no funcionales; sirviéndonos para sentar las bases y el punto de partida para la implementación del sistema; el cual solo fue mencionado brevemente en este capítulo mediante algún que otro diagrama puesto que en el próximo capítulo será abordado en su totalidad.

## Capítulo III. Descripción de la propuesta de solución

En este capítulo se razona la propuesta de solución computacional a la problemática planteada, describiendo los servicios web diseñados, la arquitectura del sistema y se muestran los diagramas de clase y de secuencia, entre otros elementos de apoyo a la comprensión de la solución.

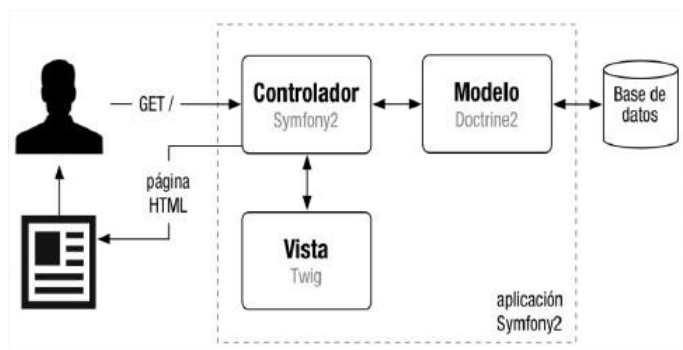
### 3.1 Arquitectura del Sistema

El sistema se encuentra implementado en Symfony en su versión 2.6, este presenta una arquitectura Modelo Vista Controlador (MVC):

Symfony está basado en un patrón clásico del diseño web conocido como arquitectura MVC, que está formado por tres niveles (SENSIOLABS, 2010):

- **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.

En la siguiente Figura 11 se muestra como Symfony 2 aplica los principios fundamentales del MVC según (Potencier, 2011).



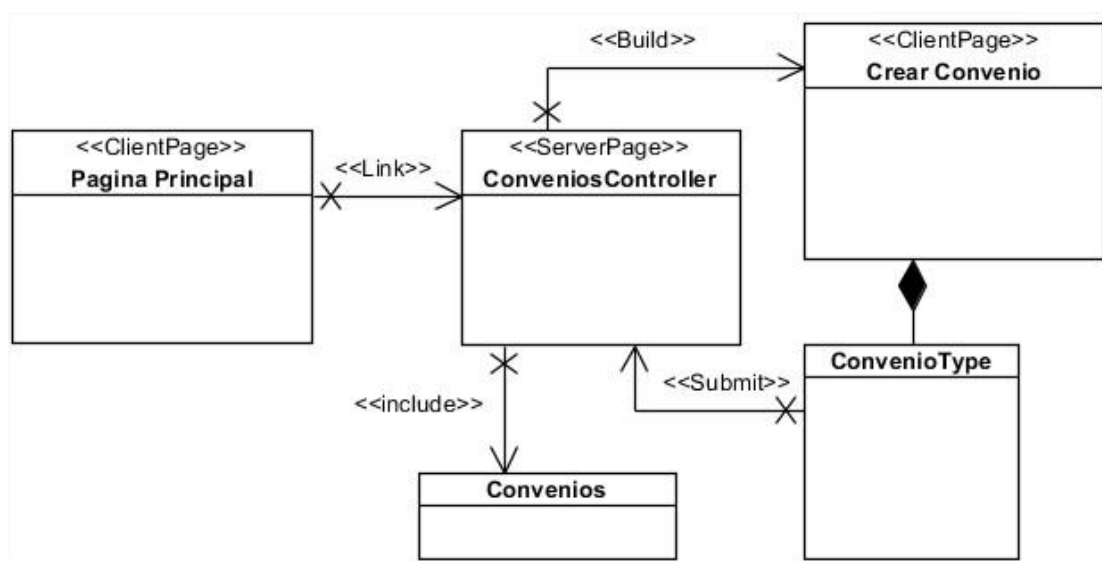
*Figura 11 Diagrama de la arquitectura MVC en Symfony*



La arquitectura MVC separa la lógica de negocio (el modelo) y la presentación (la vista) por lo que se consigue un mantenimiento más sencillo de las aplicaciones. Si por ejemplo una misma aplicación debe ejecutarse tanto en un navegador estándar como un navegador de un dispositivo móvil, solamente es necesario crear una vista nueva para cada dispositivo; manteniendo el controlador y el modelo original. El controlador se encarga de aislar al modelo y a la vista de los detalles del protocolo utilizado para las peticiones (HTTP, consola de comandos, email, etc.). El modelo se encarga de la abstracción de la lógica relacionada con los datos, haciendo que la vista y las acciones sean independientes de, por ejemplo, el tipo de gestor de bases de datos utilizado por la aplicación.

### 3.2 Diagrama de clases de diseño

A continuación en la Figura 12 se hace referencia al diagrama de clases del diseño correspondiente al especialista de Colaboración internacional y al caso de uso “Crear convenio”.



*Figura 12 Diagrama de clases de diseño para el especialista de Colaboración Internacional*

Partiendo de la clase principal, el especialista de colaboración internacional tiene la opción de acceder a varios enlaces (Convenio, Redes y Asociación, Cursos Auspiciados, Becas al extranjero e Intercambio Bilateral). Al acceder al enlace Convenios se muestra la opción de Crear un nuevo convenio, al escoger dicha opción, el sistema a través de la clase controladora ConveniosController muestra una página al cliente compuesta por un formulario ConvenioType en el cual se deben insertar los datos siempre que se pretenda realizar un

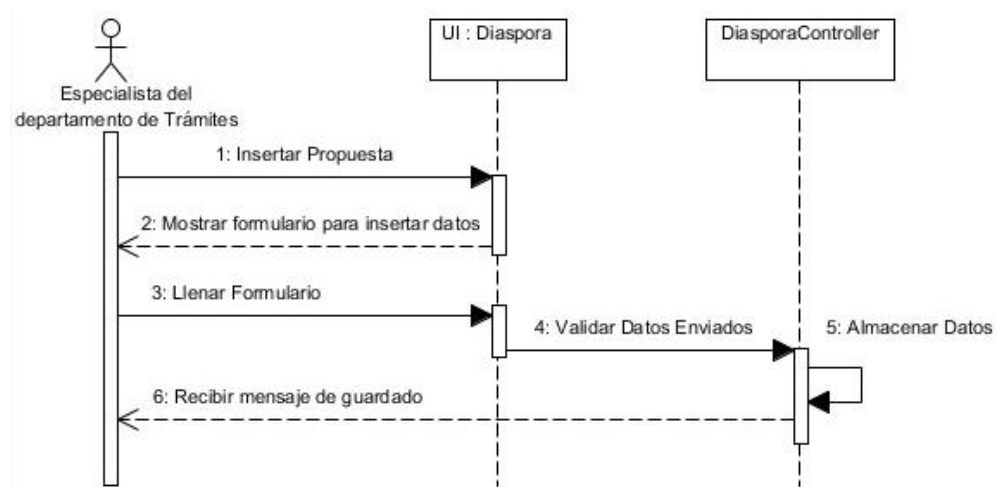
nuevo convenio, luego de insertados los datos, estos son enviados a la clase controladora ConveniosController la cual es la encargada de guardar los datos en la clase modelo Convenios.

### 3.3 Diagrama de secuencia

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. Mientras que el diagrama de casos de uso permite el modelado de una vista del escenario del negocio, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes intercambiados entre los objetos.

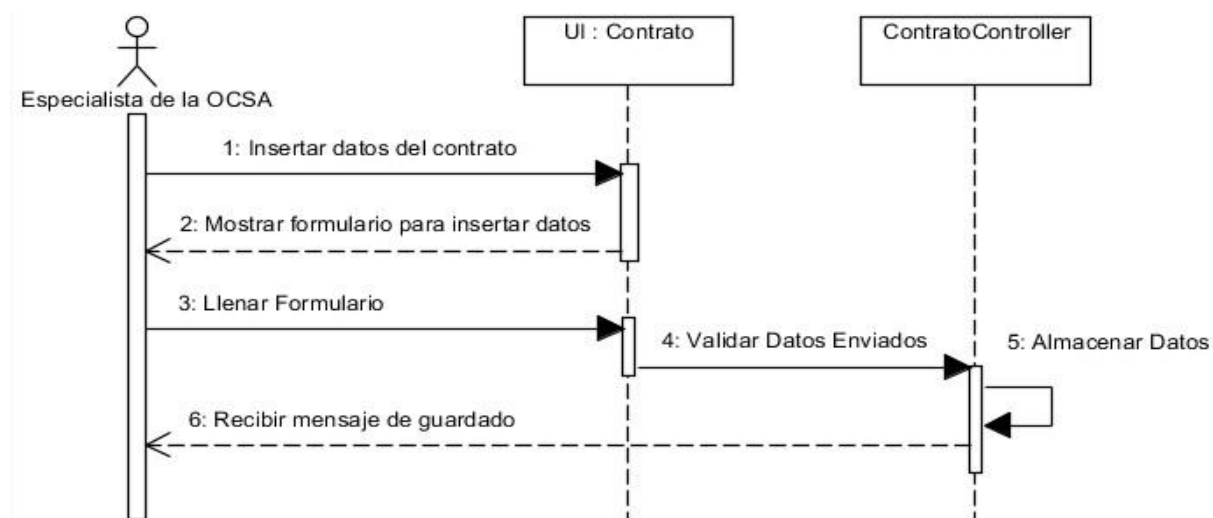
Típicamente se examina la descripción de un caso de uso para determinar qué objetos son necesarios para la implementación del escenario. Si se dispone de la descripción de cada caso de uso como una secuencia de varios pasos, entonces se puede "caminar sobre" esos pasos para descubrir qué objetos son necesarios para que se puedan seguir los pasos. Un diagrama de secuencia muestra los objetos que intervienen en el escenario con líneas discontinuas verticales, y los mensajes pasados entre los objetos como flechas horizontales.

En la Figura 13 se presenta la secuencia que persigue el sistema en el caso de uso relacionado con Gestionar la propuesta de salida del personal de la UCLV.



*Figura 13 Diagrama de secuencia para el caso de uso Gestionar propuesta de salida*

En la Figura 14 se exhibe el diagrama de secuencia correspondiente con el caso de uso Confeccionar contrato llevado a cabo por el especialista de la OCSA siendo esta una de sus funcionalidades principales.



*Figura 14 Diagrama de secuencia para el caso de uso Confeccionar contrato*

### 3.4 Tratamiento de errores

El tratamiento de los errores es fundamental para el buen funcionamiento de cualquier sistema, particularmente en el sistema SIRI, donde se validaran todos los formularios de entrada de datos, para garantizar así que toda la información que se almacena en la base de datos sea correcta, por ejemplo si se dejan campos en blancos se indicara al usuario lo que está ocurriendo, si se introducen información que ya se encuentran en la base de datos, recibirá un mensaje indicándole que dicha información ya está almacenada, además cuando sea posible se utilizarán campos de selección de información para evitar así que el usuario entre información errónea y si el usuario olvida seleccionar la información, recibirá un mensaje indicándole su olvido.

Para la validación de los campos de los formularios se usa la clase *validation* que trae consigo el Symfony, este archivo se encuentra en la siguiente dirección (C:\xampp\htdocs\2.6\src\Siri\MainBundle\Resources\config\validation.yml), poniendo en este todas las reglas de los formularios como propiedades a los campos correspondientes.

El siguiente ejemplo muestra varias de las validaciones utilizadas en los formularios del sitio:

Se muestra la ruta de la entidad a la cual se le van a realizar los cambios (Siri\MainBundle\Entity\PersonaProfesor:) y a continuación se usa la etiqueta *properties* para insertar las validaciones siendo pasadas a manera de propiedades como se muestra a continuación:

- Validación del campo de monto a reducir en el formulario de Contrato, donde el valor insertado debe ser un número en el rango de 0 a 100.  
montoreduccion:
  - Regex: {pattern:'/^[-+]?[d\*\.]?[d+\$/]',message: Entre un número válido.}
  - Range: {min:0,max:100,minMessage: Debe ser un valor entre 0 y 100,maxMessage: Debe ser un valor entre 0 y 100}
- Validar que el campo carné de identidad no este vacío, contenga solamente números y sea de 11 cifras; esto es tenido en cuenta para los cubanos pues los extranjeros si pueden contener letras en el número de su documento identificativo. Ver imagen en [Anexo 1](#).  
ci:
  - NotBlank: ~
  - Regex: {pattern:'/^([+]?[0-9]{11})\*\$/',message: El CI debe tener 11 cifras válidas.}
- Validación de los campos tipo nombre y apellidos, este no va a estar vacío, debe contener letras solamente y la primera letra insertada debe ser en mayúscula. Ver imagen en Anexo 2.  
primerNombre:
  - NotBlank: ~
  - Regex: {pattern:' /^[A-Z]+([a-z A-ZáéíóúÁÉÍÓÚñÑ-])\*\$/',message: La primera letra es en mayúscula. }
- Validación del campo de email, este no debe estar vacío y debe tener una estructura valida como la siguiente (usuario@subdominio.dominio). Ver imagen en [Anexo 3](#).  
email:
  - NotBlank: ~
  - Regex: {pattern:' /^[a-z0-9.\_%-]+@[a-z0-9.-]+\.[a-z]{2,4}\$/i',message: Entre una dirección de email válida. }

**Además de las validaciones manejadas en Symfony previamente mencionadas, fueron usadas validaciones en JavaScript, en este caso para la validación del identidad.** Ver imagen en [¡Error! No se encuentra el origen de la referencia.](#),

[Anexo 4](#)

Anexo 5, [Anexo 6](#).

- Validación para chequear que un carné de identidad no se repita.

```

var $valido=false;
    var $test=false;
    var $ci = $('#siri_mainbundle_personaprofesor_ci');

    $ci.change(function () {

        $('#ci-img').attr("src", function () {
            return "{{ asset('img/spinner.gif') }}";
        });
        $('#ci-help').text("Verificando que no exista en la base de datos");
        var $form = $(this).closest('form');
        var data = {ci: $(this).val()};
        $test=true;
        $.ajax({
            url: '{{ path("select_ci") }}',
            type: $form.attr('method'),
            data: data,
            success: function (data) {
                $valido=false;
                for (var i = 0, total = data.length; i < total; i++) {

                    if ($ci.val()===data[i].ci) {
                        $valido=true;
                        $('#ci-img').attr("src", function () {
                            return "{{ asset('img/no.png') }}";
                        });
                        $('#ci-help').text("El  carne  ya  existe  le  pertenece  a
"+data[i].primerNombre+" "+data[i].primerApellido+" "+data[i].segundoApellidos);
                    };
                }
            }
        });
    });

```

```

    }
    if (!$valido ) {
        $test=false;
        $('#ci-img').attr("src", function () {
            return "{{ asset('img/yes.png') }}";
        });
        $('#ci-help').text("Verificado");

    };
    }
    });
});

function verificar(){
    if( $valido || $test) {
        alert("Verifique el carnet");
        return false;
    }
    else{
        return true;}
    }
}

```

### 3.5 Diseño de la base de datos

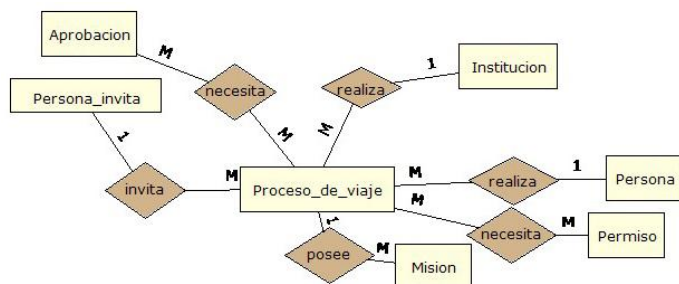
El modelado de la base de datos es de gran importancia, ya que este nos muestra cómo están encauzadas las disímiles entidades que conforman la base de datos, además se muestran un conjunto de condiciones que deben cumplir los datos que van a ser insertados. Dicho modelado de la base de datos, sirve como respaldo para un mejor entendimiento de los tipos de datos que conforman la base de datos, además del tipo de relación que existe entre estos.

#### 3.5.1 Modelo conceptual de datos

En el modelado conceptual de los datos es donde los elementos y las relaciones entre ellos quedan correctamente representadas. A continuación se muestran algunas de las tablas más relevantes de la base de datos.

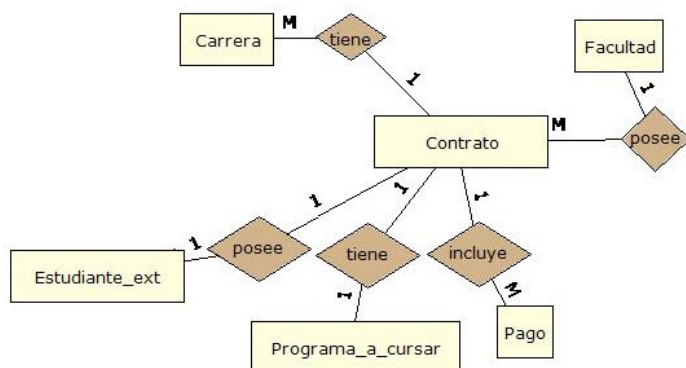
En la Figura 15 se muestra el diagrama correspondiente al proceso de gestionar propuesta de salida, en esta se representan las tablas más importantes, tal es el caso de la tabla **Proceso\_de\_viaje** en la cual se almacenan todas las fechas y datos correspondientes a los

procesos de viajes y la tabla **Persona** donde se almacena toda la información referente a una persona.



*Figura 15 Tablas asociadas al proceso de gestionar una propuesta de salida*

En la Figura 16 se muestra el diagrama correspondiente al proceso de gestionar contrato, en esta se representan las tablas más importantes, tal es el caso de la tabla **Estudiante\_ext** en la cual se almacenan los datos de los estudiantes extranjeros, también está la tabla **Contrato**, la cual recoge los datos correspondientes a un contrato, esta tabla se encuentra en estrecha relación con la tabla **Pago** pues un contrato se encuentra sujeto a diversos pagos según el contrato que se haya realizado.



*Figura 16 Tablas asociadas al proceso de gestionar contrato*

Para ver el modelo conceptual de los datos en toda su integridad remitirse al **¡Error! No se encuentra el origen de la referencia..**

### 3.5.2 Modelo físico de datos

El modelo físico de datos muestra cómo queda distribuida la base de datos finalmente, luego de realizarle un análisis a los requisitos existentes en la aplicación. A continuación se muestra algunas de las tablas relacionadas con el proceso de gestionar contratos, y gestionar propuesta de salida en la Figura 17 y la Figura 18 respectivamente.

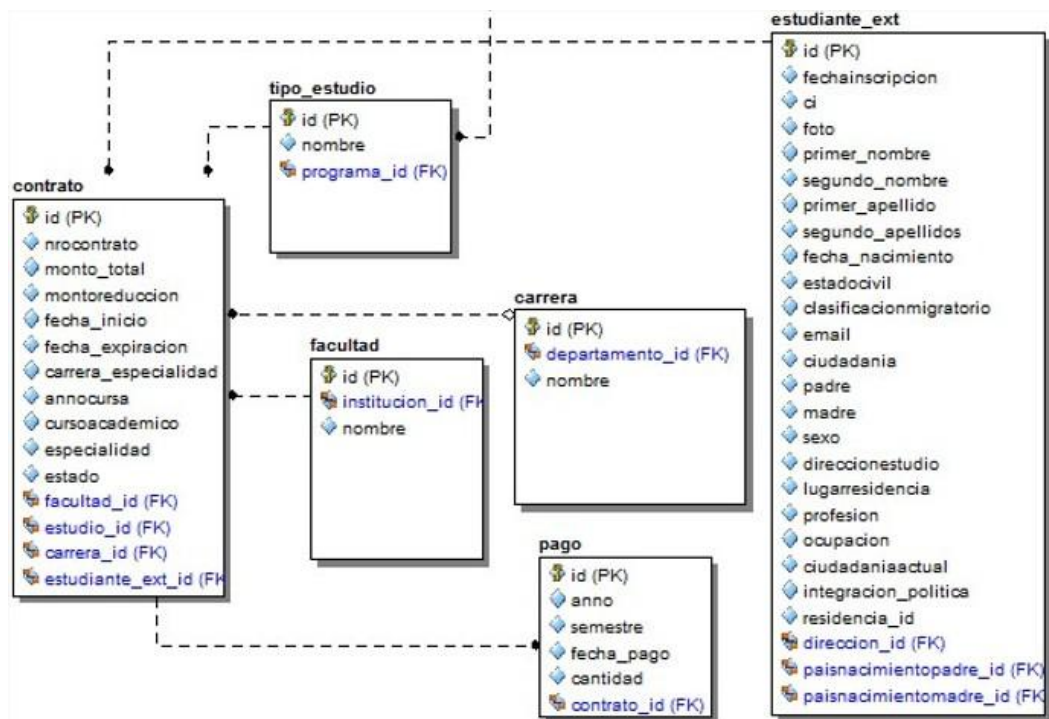


Figura 17 Tablas asociadas al proceso de gestionar contrato

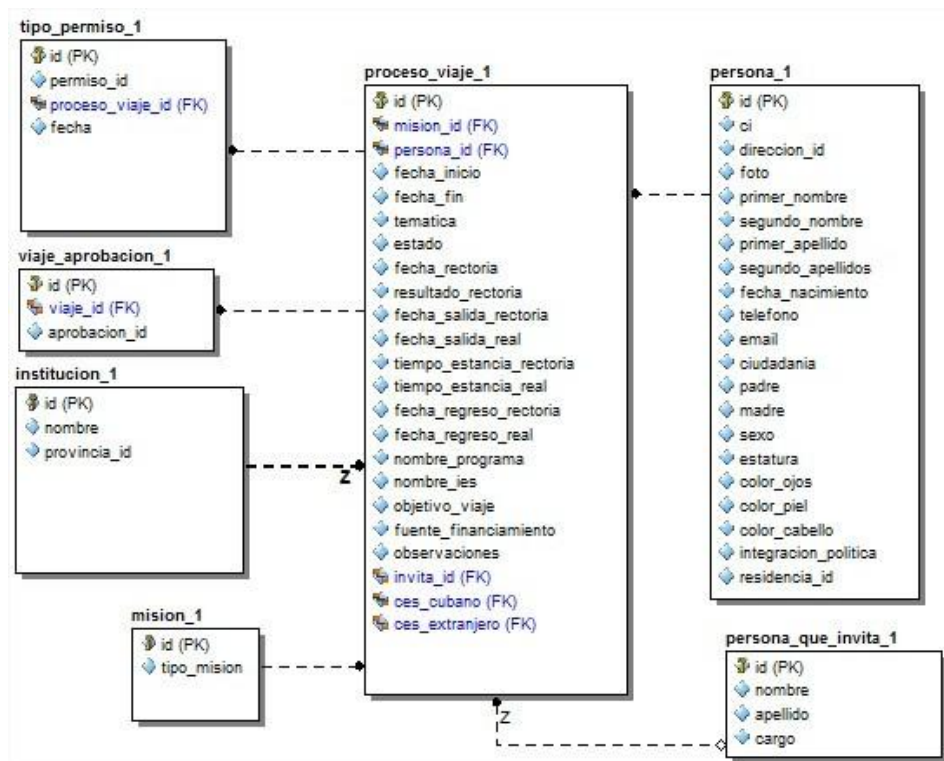


Figura 18 Tablas asociadas al proceso de gestionar propuesta de salida

Para ver el modelo físico de datos en su totalidad remitirse al



Anexo 8.

### 3.6 Modelo de componentes

Los diagramas de componentes son utilizados para modelar una vista estática del sistema. Muestra la organización y las dependencias lógicas entre un conjunto de componentes software como son: componentes de código fuente, librerías o componentes de código ejecutable.

En la Figura 19 se muestra el diagrama de componentes correspondiente a nuestro sistema, mostrando un componente de tipo *file* llamado principal, este nos conduce a la vista, mediante la vista se accede al controlador el cuál es el encargado de gestionar toda la lógica del software para su correcto funcionamiento, además del controlador la vista está relacionada con varias bibliotecas, tales como la biblioteca *Boostraps*, la cual actúa como complementario de la vista, la biblioteca *jQuery*, brindado todos los métodos *JavaScripts* necesarios para el funcionamiento correcto de la vista, también se encuentra la *DataTable* la cual posee las funciones correspondientes para el filtrado de los datos en las tablas y por último se encuentra la biblioteca *FormValidation*, encargada de proporcionar las reglas para la validación de la entrada de los datos suministrados por los usuarios que van a interactuar con el sistema.

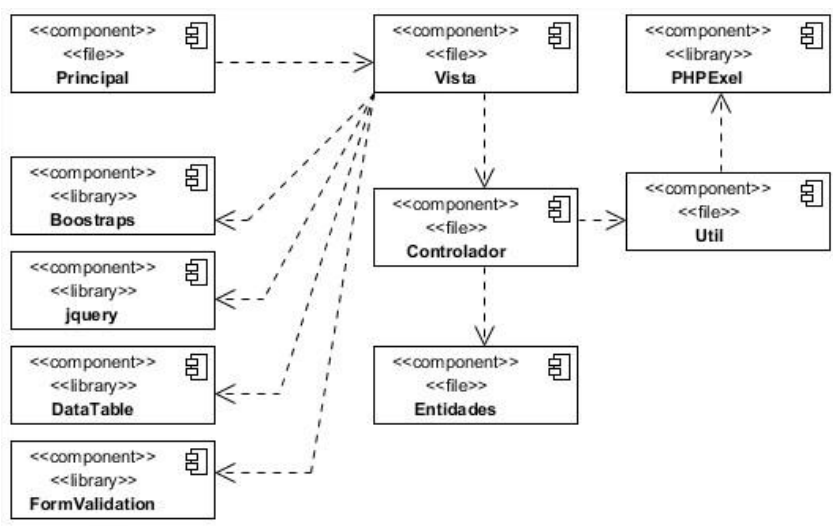


Figura 19 Modelo de componentes

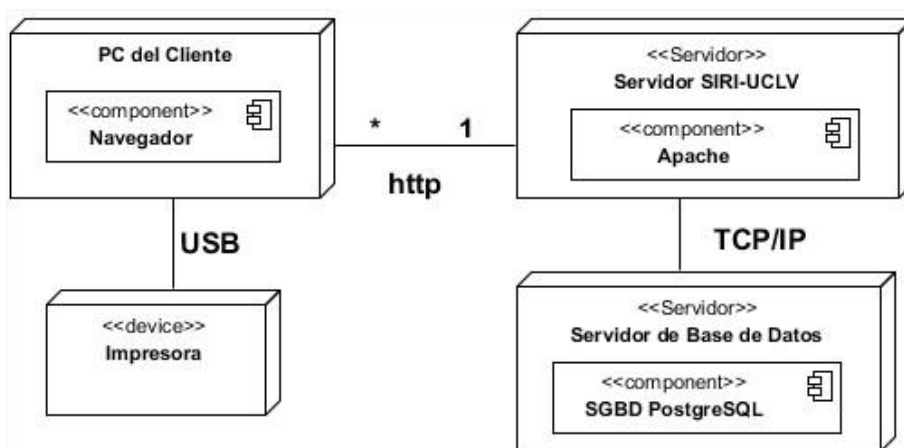
### 3.7 Diagrama de despliegue

Un diagrama de despliegue muestra un conjunto de nodos y sus relaciones. Los diagramas de despliegue se utilizan para describir la vista de despliegue estática de una arquitectura.

Dicho diagrama está conformado por distintos componentes, según (2014) “Estos componentes son los necesarios y suficientes para formar un sistema ejecutable, tales como las bibliotecas dinámicas (DLL, Dynamic Link Libraries) y los ejecutables (EXE). Esta definición de UML de componente es lo bastante amplia para cubrir modelos de objetos clásicos, tales como COM+, CORBA y Enterprise Java Beans, así como modelos de objetos alternativos, que quizá impliquen páginas web dinámicas, tablas de bases de datos y ejecutables que utilicen mecanismos de comunicación propietarios”.

La aplicación necesitará de al menos una computadora cliente la cual se conectará a la computadora servidor mediante el protocolo HTTP. A continuación se presenta el diagrama propuesto.

En la siguiente Figura 20 se muestra el diagrama de despliegue de cómo se encuentra distribuida la aplicación.



*Figura 20 Diagrama de despliegue*

EL nodo Servidor SIRI-UCLV, provee los servicios del sistema a los nodos clientes, almacena todos los archivos que conforman el entorno y procesa las solicitudes de los clientes en la red, este a su vez se conecta vía TCP/IP al nodo servidor de base de datos en el cual se encuentra montado el servidor de PostgreSQL.

El nodo pc del cliente, accede a los servicios del sistema en el servidor de forma remota a través de la red, a su vez tiene conectado una impresora vía USB.

### 3.8 Conclusiones Parciales

En este capítulo se obtuvo el modelado del sistema, partiendo de la arquitectura del sistema hasta llegar al despliegue, no sin antes tocar el tratamiento de errores tenido en cuenta para la implementación de este sistema, en dicho modelado se encuentra implícita la descripción de la propuesta de solución de acuerdo a las necesidades del cliente.

## Capítulo IV. Pruebas y análisis de factibilidad

En el presente capítulo se desarrolla además de los casos de prueba apoyados en caja negra, el análisis de factibilidad utilizando la planificación basada en uno de los métodos de estimación, siendo el de casos de uso el seleccionado para esta ocasión.

### 4.1 Planificación basada en uno de los métodos de estimación

La planificación es un proceso de toma de decisiones para alcanzar un futuro deseado, teniendo en cuenta la situación actual y los factores internos y externos que pueden influir en el logro de los objetivos.

#### 4.1.1 Estimación basada en Casos de Uso

La estimación mediante el análisis de puntos de casos de uso es un método propuesto originalmente por Gustav Karner de Objectory AB, y posteriormente refinado por muchos otros autores. Se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores (Josue Carralero Iznaga, 2006).

##### 4.1.1.1 Estimación del Esfuerzo Basada en Casos de Uso.

Cálculo de Puntos de Casos de Uso sin Ajustar (UUCP).

Este valor se calcula a partir de la siguiente ecuación:

$UUCP = UAW + UUCW$  donde,

UUCP: Puntos de Casos de uso sin ajustar

UAW: Factor de peso de los actores sin ajustar

UUCW: Factor de peso de los casos de uso sin ajustar

#### **Determinación del factor de peso de los actores sin ajustar (UAW).**

Este valor se calcula mediante un análisis de la cantidad de Actores presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los actores se establece, teniendo en cuenta en primer lugar, si se trata de una persona o de otro sistema, y en segundo lugar, la

forma en que el actor interactúa con el sistema como aparece a continuación en la Tabla 7 correspondiente a los factores de peso de los actores:

Tabla 7 Factores de peso de los actores.

Tipo de actor	Descripción	Factor de peso	Número de actores	Resultado
Simple	Otro sistema que interactúa con el sistema a desarrollar mediante una interfaz de programación(API, Application Programming Interface)	1	0	0
Promedio	Otro sistema que interactúa con el sistema a desarrollar mediante un protocolo o una interfaz basada en texto.	2	0	0
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica.	3	6	18
Total				18

De manera que  $UAW = 18$

#### **Determinación del factor de peso en los casos de uso sin ajustar (UUCW).**

Este valor se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los casos de uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómicas, es decir, se efectúa la secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia.

#### **Determinación del factor de peso en los casos de uso sin ajustar (UUCW).**

Este valor se calcula mediante un análisis de la cantidad de Casos de Uso presentes en el sistema y la complejidad de cada uno de ellos. La complejidad de los casos de uso se establece teniendo en cuenta la cantidad de transacciones efectuadas en el mismo, donde una transacción se entiende como una secuencia de actividades atómicas, es decir, se efectúa la

secuencia de actividades completa, o no se efectúa ninguna de las actividades de la secuencia. En la Tabla 8 se muestran los factores de peso correspondiente a los casos de uso.

*Tabla 8 Factores de peso de los casos de uso.*

Tipo de caso de uso	Descripción	Factor de peso	Número de Casos de Uso	<i>Resultado</i>
Simple	1-3 Transacciones	5	2	10
Promedio	4-7 Transacciones	10	13	130
Complejo	Mayor de 8 Transacciones.	15	1	15
		<i>Total</i>		145

$$\text{UUCW} = 145$$

Calculando

$$\text{UUCP} = \text{UAW} + \text{UUCW}$$

$$\text{UUCP} = 18 + 145$$

$$\text{UUCP} = 163$$

#### **Cálculo de Puntos de Casos de Uso ajustados.**

Seguidamente de calcular los Puntos de Casos de Uso sin ajustar, se debe ajustar este valor mediante la siguiente ecuación:

$$\text{UCP} = \text{UUCP} \times \text{TCF} \times \text{EF} \text{ donde,}$$

**UCP:** Puntos de Casos de Uso ajustados

**UUCP:** Puntos de Casos de Uso sin ajustar

**TCF:** Factor de complejidad técnica

**EF:** Factor de ambiente

#### **Determinación del factor de complejidad técnica (TCF).**

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante y 5 un aporte muy importante. En la siguiente Tabla 9 se muestran los factores de complejidad técnica a tener en cuenta.

Tabla 9 Factores de complejidad técnica.

Número de factor	Descripción	Peso	Valor	Factor	Comentario
T1	Sistema Distribuido	2	4	8	<i>El sistema es Web, por lo que posee cierto nivel de distribución</i>
T2	Tiempo de respuesta	1	3	3	<i>El tiempo de respuesta respalda los objetivos que se persiguen con el proyecto realizado, por lo que es el adecuado.</i>
T3	Eficiencia por el usuario	1	3	3	<i>El usuario no tiene que ser eficiente necesariamente.</i>
T4	Proceso interno complejo	1	2	2	<i>El sistema no posee cálculos complejos, aunque proporciona una serie de datos lógicos que necesitan un nivel medio de conocimiento para lograr su correcta comprensión.</i>
T5	Reusabilidad	1	5	5	<i>Se desea que el código sea lo más reutilizable posible por las magnitudes que puede alcanzar el software</i>
T6	Facilidad de instalación	0.5	1	0.5	<i>Por ser un sistema Web la complejidad de instalación es mínima.</i>
T7	Facilidad de uso	0.5	5	2.5	<i>El software debe ser muy fácil de usar por cuanto los clientes no siempre tienen dominio sobre el trabajo con sistemas informáticos</i>
T8	Portabilidad	2	2	4	<i>Los usuarios del sistema no pretenden hacer cambios en el SO.</i>
T9	Facilidad de cambio	1	5	5	<i>El sistema encuentra estructurada para que los cambios realizados afecten lo menos posible las funcionalidades del sistema.</i>
T10	Concurrencia	1	5	5	<i>La concurrencia es tratada con suma importancia pues pueden existir varios clientes conectados a la vez en el mismo instante.</i>
T11	Objetivos especiales de seguridad	1	5	5	<i>La seguridad del sistema es un tema bastante controlado, ya que el sistema sólo permite que un usuario realice las funcionalidades</i>



					<i>correspondientes a su rol dentro del sitio.</i>
T12	Acceso directo a terceras partes	1	2	2	<i>La aplicación es accesible a cualquier usuario.</i>
T13	Facilidades especiales de entrenamiento a usuarios finales	1	1	1	<i>No se hace necesario el entrenamiento de los usuarios finales, debido a la facilidad de uso que presenta el sistema, pero se debe incluir un manual de usuario para garantizar la correcta usabilidad de dicho sistema.</i>
<b>Total Factor</b>				<b>46</b>	

El Factor de complejidad técnica se calcula mediante la siguiente ecuación:

$$\text{TCF} = 0.6 + 0.01 * \Sigma(\text{Peso} \times \text{Valor asignado})$$

$$\text{TCF} = 0.6 + 0.01 * 46$$

$$\text{TCF} = 1.06$$

#### **Determinación del factor ambiente (EF).**

Las habilidades y el entrenamiento del grupo involucrado en el desarrollo tienen un gran impacto en las estimaciones de tiempo. Estos factores son los que se contemplan en el cálculo del factor de ambiente mostrado en la Tabla 10.

*Tabla 10 Factores de ambiente.*

Número del factor	Descripción	Peso	Valor	Factor	Comentario
E1	Familiaridad con el modelo del proyecto usado.	1.5	5	7.5	<i>Se está familiarizado con el modelo del proyecto, pero la experiencia en el modelado es media.</i>
E2	Experiencia en la aplicación	0.5	5	2.5	<i>Con el estudio profundo a cerca del lenguaje a utilizar se ha adquirido suficiente experiencia durante el período de trabajo.</i>
E3	Experiencia OO.	1	5	5	<i>Se considera cierto grado de experiencia en la programación orientada a objetos (OO), debido a que esta es la que se ha estudiado en cursos anteriores.</i>

E4	Capacidad del analista líder.	0.5	5	2.5	No existe analista líder, los analistas que integran el equipo de trabajo poseen capacidad media.
E5	Motivación.	1	5	5	Existe alta motivación ya que con este proyecto se van a agilizar la ejecución de todos los procesos que se llevan a cabo en la DRI-UCLV.
E6	Estabilidad de los requerimientos.	2	5	10	Aunque el sistema se encuentra sujeto a cambios, el mismo brinda las funcionalidades esenciales que dan cumplimiento a los objetivos que iniciaron su realización.
E7	Personal media jornada.	-1	0	0	Se trabajará a tiempo completo.
E8	Dificultad en lenguaje de programación.	-1	0	0	Como el lenguaje empleado fue C# y este ofrece grandes facilidades y ventajas, se considera una dificultad media su empleo.
Total				32.5	

El factor de ambiente se calcula mediante la siguiente ecuación:

$$EF = 1.4 - 0.03 * \sum (\text{Peso} \times \text{Valor asignado})$$

$$EF = 1.4 - 0.03 * 32.5$$

$$EF = 0.42$$

#### Cálculo de los Puntos de Casos de Uso Ajustados:

$$UCP = UUCP * TCF * EF$$

$$UCP = 163 * 1.06 * 0.42$$

$$UCP = 72.57$$

#### Cálculo del esfuerzo.

El esfuerzo en horas-hombre viene dado por:

$$E = UCP * CF \text{ donde:}$$

E: esfuerzo estimado en horas-hombre.

UCP: Puntos de casos de uso ajustados.

CF: Factor de conversión (20 horas-hombre por defecto).

$$E = 72.57 * 20$$

$$E = 1451.4 \text{ Horas-Hombre}$$

Para la obtención de una estimación más exacta de la duración del proyecto, se hace necesario agregar a la estimación del esfuerzo obtenida por los puntos de casos de uso, las estimaciones de esfuerzo de las restantes actividades que se llevaron a cabo durante el desarrollo del

software; así la distribución del esfuerzo entre dichas actividades está dada por la siguiente aproximación mostrada en la Tabla 11.

*Tabla 11 Distribución genérica del esfuerzo*

Actividad	Porcentaje
Análisis	10.00%
Diseño	20.00%
Programación	40.00%
Pruebas	15.00%
Sobrecarga(otras actividades)	15.00%

Con este criterio y tomando como entrada la estimación de tiempo calculada a partir de los puntos de casos de uso, se pueden calcular las demás estimaciones para obtener la duración total del proyecto. En la Tabla 12 se muestra la distribución real del esfuerzo.

*Tabla 12 Distribución real del esfuerzo.*

Actividad	Porcentaje
Análisis	362.85
Diseño	725.70
Programación	1451.4
Pruebas	544.27
Sobrecarga(otras actividades)	544.27
<b>Total</b>	<b>3628.5</b>

**Cálculo del esfuerzo total:**

$E_{Total} = 3628.5 \text{ horas / hombre}$

**Cálculo del tiempo de desarrollo:**

$T_{Desarrollo} = E_{Total} / CHT_{Total}$  CHTTotal: Cantidad de hombres =2

$T_{Desarrollo} = 3628.5 / 2 \text{ horas}$

$T_{Desarrollo} = 1814.25 \text{ horas}$

Considerando que se trabajan 8 horas diarias:

$T_{Desarrollo} = T_{Desarrollo} / 8 \text{ horas/día}$

$T_{Desarrollo} = 1814.25 \text{ horas} / 8 \text{ horas/día}$

$T_{Desarrollo} = 227 \text{ días aproximadamente}$

### Cálculo del costo:

$$\text{Costo Total} = \text{ETotal} * 2 * \text{TH}$$

TH: El salario promedio de 1 desarrollador es de \$400 y por tanto la **TH** =  $400 / 160 = 2.50$

$$\text{Costo Total} = 1814.25 * 2 * 2.5$$

$$\text{Costo Total} = \$9071.25$$

#### 4.1.2 Conclusiones del Análisis de Factibilidad

Teniendo en cuenta los resultados arrojados respecto a la factibilidad del software después del estudio perpetrado en este capítulo, se puede expresar que se brinda suficientes beneficios para cubrir sus costos, o sea, que se apoya la realización del sistema la para la DRI-UCLV, ya que es factible y económico; el mismo implicará un esfuerzo total *de* 3628.5 horas /hombre, para un tiempo de desarrollo de 227 días aproximadamente, se contarán con 2 hombres para su realización, lo que implica un costo de \$9071.25 para una tarifa horaria de \$2.50.

#### 4.2 Casos de Pruebas (caja negra)

Las pruebas de caja negra son las llevadas a cabo sobre la interfaz del software. O sea, los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, así como que la integridad de la información externa (por ejemplo, archivos de datos) se mantiene. Una prueba de caja negra examina algunos aspectos del modelo fundamental del sistema sin tener mucho en cuenta la estructura lógica interna del software.

Según (PRESSMAN, 2002), las pruebas de caja negra, también denominada prueba de comportamiento, se centran en los requisitos funcionales del software. Estas pruebas permiten obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. La prueba de caja negra no es una alternativa a las técnicas de prueba de caja blanca. Más bien se trata de un enfoque complementario que intenta descubrir diferentes tipos de errores que los métodos de caja blanca.

A continuación, en la siguiente Tabla 13 y Tabla 14 se muestran las pruebas de caja negra correspondientes a los casos de uso Gestionar persona y Gestionar propuesta de salida respectivamente.

Tabla 13 Escenarios a probar en el caso de uso “Gestionar propuesta de salida”.

<b>Id del escenario</b>	<b>Escenario</b>	<b>Var. 1 Fecha</b>	<b>Var. 2 Nombre</b>	<b>Var.3 Cargo</b>	<b>Respuesta del Sistema</b>	<b>Resultado de la Prueba</b>
EC 1	Gestionar Propuesta de Salida	C	C	C	El sistema envía los datos insertados por el usuario a la base de datos.	Los datos son guardados.
EC 2		C	I	C	Inserte solo texto poniendo la primera letra en mayúsculas.	Los datos no son guardados.
		C	C	I	Inserte solo texto poniendo la primera letra en mayúsculas.	
		I	C	C	El campo fecha no puede estar vacío.	

Tabla 14 Escenarios a probar en el caso de uso “Gestionar persona”.

<b>Id del escenario</b>	<b>Escenario</b>	<b>Var. 1 CI</b>	<b>Var. 2 Nombre</b>	<b>Var.3 Correo</b>	<b>Var.4 Teléfono</b>	<b>Var.5 Estatura</b>	<b>Var.6 Fecha Nacimiento</b>	<b>Respuesta del Sistema</b>	<b>Resultado de la Prueba</b>
EC 1	Gestionar Persona	C	C	C	C	C	C	El sistema envía los datos insertados por el usuario a la base de datos.	Se guardan los datos
EC 2		C	I	C	C	C	C	Inserte solo texto poniendo la primera letra en mayúsculas.	Los datos no son guardados
		C	C	I	C	C	C	Entre una dirección de email válida.	
		C	C	C	I	C	C	Entre un número válido.	

		<i>C</i>	<i>C</i>	<i>C</i>	<i>C</i>	<i>I</i>	<i>C</i>	<i>Entre un número válido.</i>	
		<i>C</i>	<i>C</i>	<i>C</i>	<i>C</i>	<i>C</i>	<i>I</i>	<i>Entre un número válido.</i>	

### 4.3 Conclusiones parciales

En este capítulo, se efectuó un análisis de factibilidad utilizando el método de estimación basada en casos de uso lo que permitió obtener un estimado del esfuerzo, el tiempo y el costo que figura llevar a cabo el desarrollo de un proyecto; además se efectuaron pruebas de caja negra efectuadas sobre la interfaz del software con el fin de demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto.

## Conclusiones

Como resultado de esta investigación, se obtuvo un sistema informático para la gestión de los procesos de la DRI\_UCLV que corresponde con las necesidades y requerimientos de este, siendo el sistema antes mencionado de gran utilidad para los trabajadores del centro, cumpliéndose de esta forma los objetivos planteados, ya que:

1. Se identificaron las reglas del negocio vigentes en los diferentes procesos que se gestionan en la DRI-UCLV.
2. Se implementó el sistema informático mediante el uso de las tecnologías web que facilitan el trabajo de la DRI-UCLV.
3. Se evaluó la validez del sistema informático mediante las pruebas realizadas a este.

## Recomendaciones

Luego de concluido el trabajo quedan propuestas algunas recomendaciones pensando en el futuro desarrollo y perfeccionamiento del sistema informático, entre las que se destacan:

1. Integrar la autenticación de los usuarios al sistema a través del directorio activo LDAP.
2. Generar gráficos como apoyo a los reportes.
3. Implementar la notificación por correo sobre la aprobación en rectoría de la propuesta de salida entregada por cierta persona al especialista del departamento de trámites.
4. Implementar el llenado de la propuesta de salida vía online por parte de la persona interesada sin tener que entregar directamente un documento en la oficina del trámites.



## Referencias bibliográficas

Sistemas gestores de bases de datos 1.

2014. Conferencia 4. Diagramas de componentes y de despliegue.

ACM Computing Carrers and Degrees.

ACM Computing Degrees and Jobs.

AIKINS, R. 2012

EGUILUZ, J. 2013. Desarrollo web ágil con Symfony.

FABIEN POTENCIER, F. Z. Symfony la guía definitiva.

FABIEN POTENCIER, R. W. Symfony 2.4, el libro oficial.

GROUP, T. P. G. D. 2013. PostgreSQL 9.3.0 Documentation.

GUTIÉRREZ, J. J. 2014. ¿ Qué es un framework Web?

HEURTEL, O. 2011. PHP 5.3 Desarrollar un sitio web dinamico e interactivo.

IVAR JACOBSON, G. B., JAMES RUMBAUGH 2000. El Proceso Unificado de Desarrollo de SW.

JOSUE CARRALERO IZNAGA, J. F. T. A., JULIO MARTÍNEZ PRIETO, JOSÉ A. FRANCO NAVARRO, SAYLIS

CABRERA SIERRA, PAULA ARDANZA MENÉNDEZ, NORAILIS BRITO HERNÁNDEZ 2006.

“Hotel X” Un ejemplo práctico para la asignatura de Ingeniería de Sosftware 1.

MYSQL 2008. Panorámica del sistema de gestión de base de datos MySQL

PERALTA, M. 2008. Sistema de Información.

POTENCIER, F. 2011. What is Symfony2?

POTENCIER, F. & ZANINOTTO, F. 2007. *The Definitive Guide to symfony (Definitive Guide)*, Apress.

PRESSMAN, R. 2002. INGENIERIA DEL SOFTWARE UN ENFOQUE PRACTICO.

REES, D. 2012. Laravel: Code Bright.

RIEHLE, D. 2000. Framework Design: A Role Modeling Approach.

SENSIOLABS 2010. A Gentle Introduction to symfony. Build professional websites faster with PHP and symfony.

SIERRA, F., ACOSTA, J., ARIZA, J. & SALAS, M. Estudio y análisis de los framework en php basados en el modelo vista controlador para el desarrollo de software orientado a la web.

SOMMERVILLE, I. 2002. Ingeniería de Software.

STALLMAN, R. 2004. *Software libre para una sociedad libre*, Madrid: Traficantes de Sueños, 2004.

## Anexos

### Anexo 1

**Datos básicos**

---

**Foto**  No file selected.

**CI**

• El CI debe tener 11 cifras válidas.

### Anexo 2

**Primer Apellido**

• Inserte solo texto poniendo la primera letra en mayúsculas.

**Segundo Apellido**

• Inserte solo texto poniendo la primera letra en mayúsculas.

### Anexo 3

**Email**

• Entre una dirección de email válida

### Anexo 4

**Datos básicos**

---

**Foto**  No file selected.

**CI**

🌀 Verificando que no exista en la base de datos

## Anexo 5

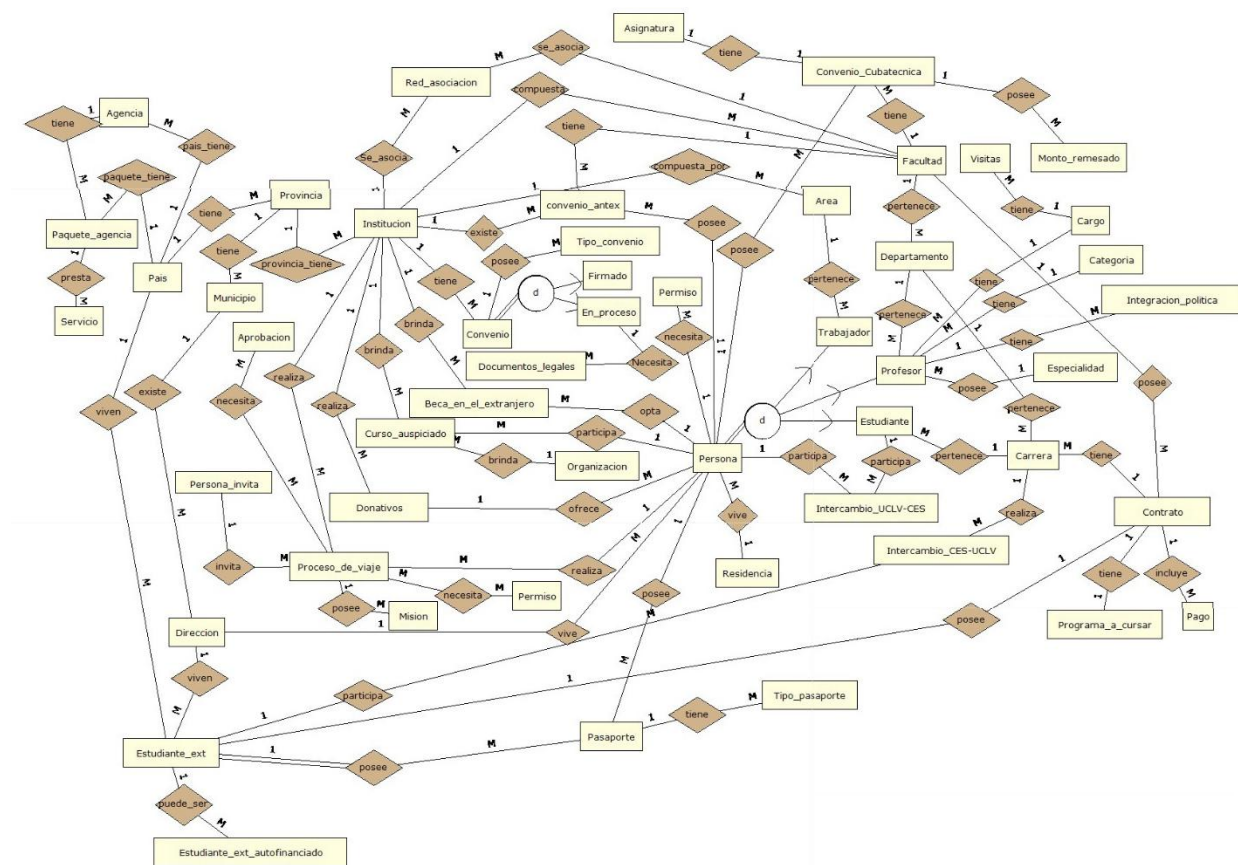
CI 91090135327  
✓ Verificado

## Anexo 6

CI 91090135325

✖ El carnet ya existe le pertenece a Jhony Deep Deep

## Anexo 7



## Anexo 8

