



Universidad de las Ciencias Informáticas  
Facultad 1

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

Sistema Integrado de Transportación (SIT).

Implementación de los Módulos

Administración - Configuración y Seguridad

**AUTORES:** José Alejandro Garcia Calderón  
Iran Roberto Rodríguez Moreno

**TUTORES:** Ing. Alexander Rodríguez Mompíe  
Ing. Yariel Ramos Negrín

**CO-TUTOR:** Ing. Aní Bermudez Peña

*Ciudad de la Habana, 2009*

*“Año del 50 Aniversario del Triunfo de la Revolución”*

## Dedicatoria

A la Revolución, a mi abuela y a mis padres.

***Iran***

A toda mi familia.

***José***

## **Agradecimientos.**

A nuestras abuelas por su comprensión, nuestros familiares para que no falten ninguno, que sepan que los tuvimos presentes y por habernos apoyado en todo el trabajo realizado.

A nuestros tutores por todo el apoyo ofrecido en la tutoría del documento de tesis.

A Lidiana, Dania, Yamila, Dalia, Laritza y Marianny por la ayuda en la revisión del documento de tesis.

A todas personas que de una forma u otra hayan colaborado en la elaboración del documento, de la implementación de la aplicación web, a los que aportaron su granito de arena para ellos gracias.

## Declaración de autoría

Por este medio declaramos que Iran Roberto Rodríguez Moreno y José Alejandro Garcia Calderón somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas (UCI) para que haga el uso que estime pertinente con este trabajo.

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

Iran Roberto Rodríguez Moreno

---

Firma del Autor

José Alejandro Garcia Calderón

---

Firma del Autor

Ing. Alexander Rodríguez Mompíe.

---

Firma del Tutor

Ing. Yariel Ramos Negrín

---

Firma del Tutor

Ing. Anié Bermudez Peña

---

Firma del Co-Tutor





*"El individuo ha luchado siempre para no ser absorbido por la tribu. Si lo intenta, a menudo estará solo, y a veces asustado. Pero ningún precio es demasiado alto por el privilegio de ser uno mismo."*

***Friedrich Nietzsche***

## **Resumen**

El presente trabajo pretende mejorar todo lo referente a la administración, configuración y seguridad de los diferentes Sistemas de Reservaciones de Transporte en la UCI. Teniendo en cuenta que la forma en que se realizan actualmente es algo difícil y de cierto modo ambiguo; además de tenerlas separadas, no brindan una serie de funcionalidades que, con el aumento del flujo de datos que dichas aplicaciones manejan se les hace necesaria tener. Es por ello, que se realizó la implementación de los módulos de Administración – Configuración y Seguridad del Sistema de Reservaciones de Transporte en la UCI para brindar mayor usabilidad y eficiencia a la hora de la gestión de la administración de dicha aplicación. Para su desarrollo se realizó un amplio estudio de los procesos implicados en el tema, con el objetivo de garantizar un mejor funcionamiento, identificar deficiencias, corregirlas e incluirle mejoras. Se tuvieron en cuenta las herramientas y tecnologías actuales a ser usadas; así como la generación de la documentación asociada (descripción detallada de las clases y operaciones utilizadas para el desarrollo del sistema) durante la implementación de dichos módulos. Finalmente se hicieron pruebas para verificar la calidad, seguridad, confiabilidad, y disponibilidad, con el objetivo de obtener la aceptación de los clientes y usuarios finales.

**Palabras Claves:** Implementación, Configuración, Seguridad, Transportación, Reservación.



## Tabla de contenido

Introducción.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA .....	7
1.1. Introducción.....	7
1.2. Procesos de Administración - Configuración y Seguridad.....	7
1.2.1Procesos de Reservasiones a Nivel Mundial .....	7
1.2.2 Procesos de Administración – Configuración y Seguridad en Transportación de la UCI .....	10
1.3 Sistema Integrado de Transportación .....	10
1.4 Sistemas automatizados existentes .....	11
1.4.1 Transportación de fin de semana .....	12
1.4.2 Transportación Nacional .....	12
1.5 Conceptos asociados al dominio del problema .....	12
1.5.1 Transportación Nacional o Masiva .....	12
1.5.2 Transportación semestral de trabajadores .....	13
1.5.3 Transportación de estudiantes de fin de semana .....	13
1.5.4 Transporte .....	13
1.5.5 Viaje .....	13
1.5.6 Configuración .....	13
1.5.7 Sistema Integrado de transportación.....	14
1.5.8 Programador.....	14
1.6 Tendencias y Tecnologías Actuales.....	14
1.6.1 Internet.....	15
1.6.2 Aplicaciones Web.....	15
1.6.3 Seguridad en Aplicaciones Web.....	15
1.6.4 Patrón del diseño Modelo Vista Controlador (MVC) .....	16
1.6.5 Lenguaje de Programación.....	16
1.6.5.1 PHP .....	16
1.6.6 Framework Symfony .....	17
1.6.7Tecnologías utilizadas en el proceso de desarrollo. ....	18

1.6.8 Gestor de Base de Datos .....	19
1.6.9 Servicios Web.....	19
1.6.9.1 Servicios Web usados en la Aplicación.....	20
1.6.10 Protocolo LDAP .....	20
1.7 Herramientas a Utilizar .....	21
1.7.1 Zend Studio 5.5 .....	21
1.7.2 PostgreSQL 8.2.....	21
1.7.3 Net Beans 6.5.....	21
1.8 Conclusiones .....	22
CAPÍTULO 2: DESCRIPCIÓN Y ANÁLISIS DE LA SOLUCIÓN PROPUESTA .....	23
2.1. Introducción.....	23
2.2. Descripción de la solución propuesta.....	23
2.2.1 Configuración .....	23
2.2.2 Autenticar Usuario .....	24
2.2.3 Gestionar Accesos .....	25
2.2.4 Bloquear Provincia .....	26
2.2.5 Gestionar Bloques .....	27
2.2.6 Gestionar Boletines .....	29
2.2.7 Gestionar Paradas Intermedias .....	29
2.2.8 Gestionar Puntos de Salida .....	31
2.2.9 Gestionar Puntos de Salida UCI.....	32
2.2.10 Gestionar Semestre .....	33
2.2.11 Gestionar Rutas .....	35
2.2.12 Gestionar Roles.....	36
2.2.13 Gestionar Transportes .....	38
2.2.14 Gestionar Usuarios.....	40
2.2.15 Gestionar Viajes .....	41
2.3. Valoración crítica del diseño propuesto por el analista.....	44
2.3.1 El patrón Modelo Vista Controlador (MVC) .....	44
2.4. Análisis de posibles implementaciones y componentes o módulos que son reutilizados .....	46

2.5.	Descripción de los algoritmos no triviales utilizados .....	46
2.6.	Descripción de las nuevas clases u operaciones necesarias.....	47
2.6.1	Clases Controladoras (CC).....	47
2.6.2	Clases Interfaz (CI) .....	58
2.6.3	Clases Entidades (CE).....	100
2.7.	Conclusiones.....	104
Capítulo 3: VALIDACIÓN DE LA SOLUCIÓN PROPUESTA .....		105
3.1.	Introducción.....	105
3.1.1	Objetivos del Flujo de Trabajo de Prueba .....	106
3.2.	Método de Prueba Adecuado al Nivel de Unidad .....	106
3.2.1	Métodos de Prueba basados en Caja Blanca .....	106
3.2.2	Propuesta de Prueba del Framework Symfony.....	107
3.2.3	Pruebas unitarias .....	108
3.2.4	Pruebas realizadas al sistema.....	110
3.2.4.1	Objetivo .....	110
3.2.4.2	Alcance.....	110
3.2.4.3	Descripción de los valores utilizados para los test.....	110
3.2.4.4	Evaluación de la ejecución del test y de los resultados obtenidos.....	111
3.3.	Conclusiones:.....	112
Conclusiones.....		114
Recomendaciones .....		115
Referencias Bibliográficas .....		116
Bibliografía .....		117
Anexos .....		118
Glosario de términos .....		119

## Índice de Tablas

TABLA 1 : Descripción de la CC autenticar .....	47
TABLA 2 : Descripción de la CC configuración.....	47
TABLA 3 : Descripción de la CC gest_accesos.....	47
TABLA 4 : Descripción de la CC gest_bloqueo_prov .....	48
TABLA 5 : Descripción de la CC gest_rols .....	49
TABLA 6 : Descripción de la CC gest_usuarios.....	50
TABLA 7 : Descripción de la CC gest_bloques.....	51
TABLA 8 : Descripción de la CC gest_paradas .....	51
TABLA 9 : Descripción de la CC gest_punto .....	52
TABLA 10 : Descripción de la CC gest_punto_salida_uci .....	53
TABLA 11 : Descripción de la CC gest_semestres .....	53
TABLA 12 : Descripción de la CC gest_rutas .....	54
TABLA 13 : Descripción de la CC gest_viajes .....	55
TABLA 14 : Descripción de la CC gest_boletines.....	56
TABLA 15 : Descripción de la CC gest_transportes .....	56
TABLA 16 : Descripción de la CI configurar .....	58
TABLA 17 : Descripción de la CI autenticar usuario.....	58
TABLA 18 : Descripción de la CI buscar del caso de uso gestionar accesos.....	59
TABLA 19 : Descripción de la CI mostrar del caso de uso gestionar accesos .....	59
TABLA 20 : Descripción de la CI listar del caso de uso gestionar accesos.....	60
TABLA 21 : Descripción de la CI editar del caso de uso gestionar accesos.....	60
TABLA 22 : Descripción de la CI mostrar detalle del caso de uso gestionar accesos .....	61
TABLA 23 : Descripción de la CI editar del caso de uso bloquear provincia .....	62
TABLA 24 : Descripción de la CI listar del caso de uso bloquear provincia.....	63
TABLA 25 : Descripción de la CI mostrar del caso de uso bloquear provincia .....	63
TABLA 26 : Descripción de la CI principal del caso de uso gestionar bloques.....	63
TABLA 27 : Descripción de la CI nueva regla del caso de uso gestionar bloques .....	64
TABLA 28 : Descripción de la CI reglas del caso de uso gestionar bloques .....	65
TABLA 29 : Descripción de la CI criterios del caso de uso gestionar bloques .....	65

TABLA 30 : Descripción de la CI crear del caso de uso gestionar bloques .....	66
TABLA 31 : Descripción de la CI editar del caso de uso gestionar paradas .....	66
TABLA 32 : Descripción de la CI listar del caso de uso gestionar paradas .....	67
TABLA 33 : Descripción de la CI mostrar del caso de uso gestionar paradas .....	68
TABLA 34 : Descripción de la CI editar del caso de uso gestionar punto .....	69
TABLA 35 : Descripción de la CI mostrar del caso de uso gestionar punto .....	69
TABLA 36 : Descripción de la CI listar del caso de uso gestionar punto .....	70
TABLA 37 : Descripción de la CI principal del caso de uso generar boletines .....	70
TABLA 38 : Descripción de la CI mostrar estudiantes del caso de uso generar boletines .....	71
TABLA 39 : Descripción de la CI mostrar grupos .....	71
TABLA 40 : Descripción de la CI principal del caso de uso gestionar roles .....	72
TABLA 41 : Descripción de la CI nuevo del caso de uso gestionar roles .....	72
TABLA 42 : Descripción de la CI mostrar funcionalidades del caso de uso gestionar roles .....	73
TABLA 43 : Descripción de la CI principal del caso de uso gestionar usuarios .....	74
TABLA 44 : Descripción de la CI mostrar usuario del caso de uso gestionar usuarios .....	74
TABLA 45 : Descripción de la CI nuevo usuario del caso de uso gestionar usuarios .....	75
TABLA 46 : Descripción de la CI adicionar rol del caso de uso gestionar usuarios .....	76
TABLA 47 : Descripción de la CI editar del caso de uso gestionar punto de salida UCI .....	76
TABLA 48 : Descripción de la CI mostrar del caso de uso gestionar punto de Salida UCI .....	77
TABLA 49 : Descripción de la CI listar del caso de uso gestionar punto de salida UCI .....	77
TABLA 50 : Descripción de la CI crear del caso de uso gestionar punto de salida UCI .....	78
TABLA 51 : Descripción de la CI Principal del caso de uso gestionar semestres .....	78
TABLA 52 : Descripción de la CI meses del caso de uso gestionar semestres .....	79
TABLA 53 : Descripción de la CI crear del caso de uso gestionar semestres .....	79
TABLA 54 : Descripción de la CI listar del caso de uso gestionar transporte .....	80
TABLA 55 : Descripción de la CI modificar transporte del caso de uso gestionar transporte. ....	81
TABLA 56 : Descripción de la CI actualizar error del caso de uso gestionar transporte .....	82
TABLA 57 : Descripción de la CI mostrar del caso de uso gestionar transporte .....	84
TABLA 58 : Descripción de la CI paradas del caso de uso gestionar transporte .....	85
TABLA 59 : Descripción de la CI municipios del caso de uso gestionar transporte .....	85
TABLA 60 : Descripción de la CI editar del caso de uso gestionar transporte .....	85

TABLA 61 : Descripción de la CI adicionar asientos del caso de uso gestionar transporte .....	88
TABLA 62 : Descripción de la CI detalles asientos del caso de uso gestionar transporte .....	88
TABLA 63 : Descripción de la CI adicionar coches del caso de uso gestionar transporte .....	89
TABLA 64 : Descripción de la CI nueva reglas del caso de uso gestionar viajes .....	90
TABLA 65 : Descripción de la CI reglas del caso de uso gestionar viajes .....	91
TABLA 66 : Descripción de la CI crear viaje del caso de uso gestionar viajes .....	91
TABLA 67 : Descripción de la CI mostrar datos del caso de uso gestionar viajes .....	92
TABLA 68 : Descripción de la CI mostrar detalles del caso de uso gestionar viajes .....	93
TABLA 69 : Descripción de la CI editar del caso de uso gestionar viajes .....	95
TABLA 70 : Descripción de la CI viajes correspondientes del caso de uso gestionar viajes .....	96
TABLA 71 : Descripción de la CI editar rutas del caso de uso gestionar rutas .....	97
TABLA 72 : Descripción de la CI listar del caso de uso gestionar ruta .....	99
TABLA 73 : Descripción de la CI mostrar municipios del caso de uso gestionar rutas .....	99
TABLA 74 : Descripción de la CI mostrar puntos del caso de uso gestionar rutas .....	99
TABLA 75 : Descripción de la CI mostrar detalles del caso de uso gestionar rutas .....	100
TABLA 76 : Descripción de la CEiajes .....	100
TABLA 77 : Descripción de la CE viajes aux .....	101
TABLA 78 : Descripción de la CE aplicación .....	101
TABLA 79 : Descripción de la CE transporte .....	101
TABLA 80 : Descripción de la CE viajes general .....	102
TABLA 81 : Descripción de la CE viajes tn .....	102
TABLA 82 : Descripción de la CE viajes te .....	102
TABLA 83 : Descripción de la CE transporte .....	103
TABLA 84 : Descripción de la CE bloque .....	103
TABLA 85 : Métodos utilizados en las pruebas .....	110
TABLA 86 : Clase ldap .....	111
TABLA 87 : Clase classws .....	111
TABLA 88 : Clase TbDviajeTnl .....	111
TABLA 89 : Clase TbDviajeTn .....	112
TABLA 90 : Clase MyUser .....	112
TABLA 91 : Operacionalización de las Variables .....	118

## Índice de Figuras

Fig 2.1	Enlace Configurar .....	24
Fig 2.2	Configuración general de la aplicación .....	24
Fig 2.3	Formulario para registrarse en la aplicación.....	25
Fig 2.4	Usuario bloqueado.....	25
Fig 2.5	Enlace Accesos .....	26
Fig 2.6	Listado de usuarios bloqueados.....	26
Fig 2.7	Listado de provincias bloqueadas .....	27
Fig 2.8	Enlace a Bloques .....	27
Fig 2.9	Bloques de transportación nacional .....	28
Fig 2.10	Crear bloque de transportación.....	28
Fig 2.11	Adicionar regla.....	28
Fig 2.12	Enlace Boletines .....	29
Fig 2.13	Generar boletines .....	29
Fig 2.14	Enlace a la funcionalidad paradas intermedias .....	30
Fig 2.15	Paradas intermedias.....	30
Fig 2.16	Crear parada intermedia.....	31
Fig 2.17	Enlace Punto.....	31
Fig 2.18	Crear o editar punto.....	32
Fig 2.19	Listado de puntos .....	32
Fig 2.20	Enlace Puntos de salida UCI.....	32
Fig 2.21	Nuevo punto de salida .....	33
Fig 2.22	Listado de puntos de salida UCI .....	33
Fig 2.23	Enlace a Semestres.....	33
Fig 2.24	Gestionar semestres.....	34
Fig 2.25	Adicionar o eliminar semestres.....	34
Fig 2.26	Enlace Rutas.....	35
Fig 2.27	Nueva ruta .....	36
Fig 2.28	Listado de rutas .....	36

Fig 2.29	Enlace Roles.....	37
Fig 2.30	Crear nuevo rol .....	37
Fig 2.31	Listado de roles y funcionalidades a las que se tiene acceso.....	38
Fig 2.32	Enlace Viajes .....	38
Fig 2.33	Listado de viajes donde se señala el ícono de transporte .....	39
Fig 2.34	Listado de transportes de un viaje .....	39
Fig 2.35	Formulario para la creación de un transporte .....	40
Fig 2.36	Enlace Usuarios.....	41
Fig 2.37	Formulario para buscar los datos de un usuario.....	41
Fig 2.38	Datos de un usuario buscado.....	41
Fig 2.39	Listado de usuarios con roles en el sistema.....	41
Fig 2.40	Listado de viajes por modalidad de reservación.....	42
Fig 2.41	Formulario para crear un viaje para la transportación estudiantil .....	43
Fig 2.42	Formulario para crear un viaje para la transportación nacional .....	43
Fig 2.43	Formulario para crear un viaje para la transportación nacional de trabajadores internos.....	44
Fig 2.44	El patrón MVC.....	45
Fig 3.1	Fases y flujos de trabajos de RUP .....	106
Fig 3.2	Método de prueba.....	106
Fig 3.3	Prueba unitaria en test/unit/ldapTest.php .....	109
Fig 3.4	Prueba unitaria en test/unit/ldapTest.php .....	109



## **Introducción**

Desde los inicios el hombre siempre ha buscado la forma de hacer la mayor cantidad de operaciones en el menor tiempo posible. Las eras han ido pasando y el hombre ha ido mejorando su forma de pensar y de protegerse, se ha visto en la necesidad de perfeccionar sus herramientas de trabajo, para no quedar fuera del desarrollo de las tecnologías de la informática y las comunicaciones.

Hoy en día, una de las herramientas más usadas son las computadoras, que sobrepasan el billón de operaciones por segundo, a través de las cuales existen muchas tecnologías que están en funcionamiento, como Internet. En la actualidad con la modernización del sistema de redes, se hace factible una serie de servicios mucho más fáciles de obtener y brindar, con esto los usuarios cada día tienen acceso a realizar muchas tareas con menos tramitación y mucha más rapidez de lo que normalmente se estaba acostumbrado cuando se realizaba manualmente.

Actualmente el mundo está avanzando a la era tecnológica, e incluso en el país se está efectuando todo un proceso de informatización. El Comandante en Jefe, uno de los mejores pensadores del siglo XX y del XXI, fue el creador de la Universidad de Ciencias Informáticas (UCI). En estos momentos en la UCI se está realizando la informatización y optimización de algunos sistemas usados con el fin de mejorar los servicios brindados a la comunidad universitaria, como lo es el Sistema de Reservación de Transporte. La Dirección de Informatización de la universidad se planteó la idea de la integración de todos los sistemas de reservación de Transporte, decidiendo centralizar todo lo referente a la configuración, administración y seguimientos de seguridad de dichos sistemas.

Tomando como punto de partida que la UCI es una universidad de excelencia, que tiene como eje central el estudio de la informática, debe garantizar un óptimo funcionamiento de los servicios del Sistema de Transporte; utilizándose herramientas y lenguajes de programación libres; a partir de los diferentes Sistemas Operativos que utiliza la universidad, diseñando una aplicación que sea multiplataforma.

Este sistema se ha diseñado tomando como referencia la procedencia habitacional de trabajadores que laboran en la institución y estudiantes que cursan sus años de estudio; por lo que se le garantizan tres modalidades de transportación, las cuales se presentan a continuación:

- ✓ Transportaciones masivas para estudiantes y trabajadores en fin de año y vacaciones .

- ✓ Transportaciones semestrales para los trabajadores internos.
- ✓ Transportaciones de fin de semana hacia todos los municipios de Ciudad de la Habana para los estudiantes.

Actualmente estas transportaciones ocurren de la siguiente forma:

### **Transportaciones masivas para estudiantes y trabajadores en fin de año y vacaciones**

En la UCI, la reservación de pase masivo es un proceso que se lleva a cabo a través de una aplicación, se planifica la salida masiva de los estudiantes y trabajadores internos, distribuidos por transportes y organizados por listados, asignando a un profesor que vaya para ese destino como responsable del transporte. Generalmente al realizar este proceso en la aplicación, no se ejecutaba óptimamente, e incluso, el sistema cometía errores y mantenía una baja seguridad a la hora del usuario proceder a la reservación; implicando esto, la pérdida de información, desorganización en el momento de salida, además de no tener en cuenta la reservación de familiares, y el proceso de distribución de las solicitudes sólo se realizaban por el orden de la reservación.

### **Transportaciones de fin de semana hacia todos los municipios de Ciudad de la Habana para los estudiantes**

Una posibilidad que brinda la universidad, es la salida de pase los fines de semana a los estudiantes para cualquier municipio de Ciudad Habana. Para este servicio hay creado una aplicación en la cual todos los estudiantes tienen acceso a reservar, siempre que tuviesen algún familiar en ese municipio o fueran pertenecientes al mismo. Esta aplicación necesita de nuevas funcionalidades para optimizar las ya existentes, centrándose en mejorar la configuración de los viajes; ya que sólo se le tenía permitido reservar para los sábados por la tarde hasta el domingo por la noche y la seguridad de la información del usuario a la hora de reservar era insuficiente.

### **Transportaciones semestrales para los trabajadores internos**

En cursos anteriores fue desarrollado un sistema, el cual no fue usado porque no llegaba a cumplir con todas la funcionalidades requeridas por la Dirección de Transporte. Mientras el proceso de reservación se efectuaba manualmente y por medio del correo electrónico, por lo que resultaba muy incómodo y

extenuante, para el encargado de realizar las distribuciones de transporte debido a la cantidad de trabajadores que actualmente solicitan el servicio que se incrementa cada año. Los listados eran creados a través de Excel y cuando se enviaban los correos, muchos trabajadores desconocían de los itinerarios o no confirmaban su reservación a tiempo.

Conociendo los problemas existentes y teniendo en cuenta que las dos primeras aplicaciones no estaban desarrolladas en un ambiente libre, el Departamento de Transportaciones Nacionales de la UCI decide agrupar todos estos sistemas en una sola Aplicación Web “Sistema Integrado de Transportación”. La informatización del Sistema de Transportación está dada por un conjunto de funcionalidades técnicas y actividades dirigidas al manejo de toda la información vinculada a la reservación del transporte en la UCI.

Por todas las insuficiencias planteadas se formula el siguiente **Problema a resolver**: necesidad de implementar los módulos de Administración - Configuración y Seguridad de un sistema que automatice los procesos de gestión y reservación del Departamento de Transportaciones Nacionales de la UCI.

El esbozo de todo el sistema fue basado principalmente en el **Objeto de estudio**: procesos de gestión y reservación del Departamento de Transportaciones Nacionales en la UCI.

**Campo de acción**: módulos Administración – Configuración y Seguridad.

Todo está basado en la **Hipótesis**: la implementación de los módulos de Administración - Configuración y Seguridad del Sistema Integrado de Reservación de Transporte en la UCI, incrementará la eficiencia en la configuración de transportaciones y la seguridad de la información manejada.

### **Variables Independientes**

- Módulos de Administración – Configuración y Seguridad.

### **Variables Dependientes**

- Eficiencia en la configuración de transportaciones.
- La seguridad de la información manejada.

La Operacionalización de las variables se muestra en el Anexo 1.

El **Objetivo general** es: implementar los módulos de Administración - Configuración y Seguridad para un sistema que gestione los procesos de reservaciones de transporte en la UCI. Del objetivo general se derivan los siguientes **Objetivos específicos**:

1. Implementar el acceso, control y asignación de roles a los diferentes usuarios del sistema.
2. Implementar los permisos de los roles a los módulos del sistema; así como el tipo de acciones que tendrán sobre los mismos.
3. Permitir la autenticación de los usuarios tanto con el Directorio Activo UCI, como, con la Base de Datos Local.
4. Controlar la seguridad en el sistema llevando a cabo un minucioso tratamiento de errores; asegurando la seguridad, integridad y disponibilidad de la información generada en el proceso de reservación del transporte en la UCI.

#### **Tareas de la investigación científica**

- ✓ Entrevistar a personas que estén inmersos en el proceso para identificar las necesidades del mismo.
- ✓ Estudiar la metodología RUP en los flujos de trabajo de implementación, prueba y despliegue.
- ✓ Estudiar las herramientas a utilizar en el desarrollo del proyecto.
- ✓ Realizar una selección de las tecnologías a utilizar para justificar el por qué se seleccionaron los mismos.
- ✓ Estudiar framework de desarrollo.
- ✓ Estudiar e investigar las funcionalidades e importancia de la arquitectura de MVC (Modelo Vista Controlador) que propone el framework Symfony.
- ✓ Implementar un algoritmo de búsqueda en el Directorio Activo UCI.
- ✓ Implementar un algoritmo de autenticación en el Directorio Activo UCI.

- ✓ Implementar una clase que facilite el trabajo con los Servicios Web de la UCI.
- ✓ Implementar una clase que interactúe con el Directorio Activo UCI.
- ✓ Elaborar la fundamentación teórica de todo el proceso.
- ✓ Realizar pruebas de validación del sistema.

**Posibles resultados:** módulos Configuración - Administración y Seguridad del Sistema de Reservas funcionales con su documentación asociada. Llámese documentación asociada a la descripción detallada de las clases utilizadas, así como las funcionalidades de las mismas.

Con estos módulos implementados se obtendría los siguientes **beneficios:** los procesos de Administración – Configuración y Seguridad que serán automatizados, favoreciendo el incremento de la eficiencia en los procesos que se desarrollan en el Departamento de Transportaciones Nacionales, disminuyendo los trámites e incrementando la ligereza en la administración-configuración de los procesos de reservación.

Para garantizar un desarrollo óptimo de la investigación es necesario el empleo de métodos científicos y técnicas:

#### **Métodos empíricos**

Entrevista, se realizó a directivos para obtener información relacionada con la aplicación lo que contribuyó al perfeccionamiento de la propuesta realizada.

#### **Métodos teóricos**

Análisis y síntesis, se utilizó en el estudio de la literatura especializada relacionada con el tema y la exploración de resultados de investigaciones afines, lo que permitió adoptar posiciones teóricas relacionadas con el objeto de investigación.

Histórico y lógico, se empleó para el estudio y profundización de la evolución de las aplicaciones lo que facilitó la indagación de soluciones al problema planteado.

Modelación, se utilizó durante la elaboración del Sistema Integrado de Transportación porque se hace necesario explicarle al cliente mediante modelos, cómo se tiene pensado que quede el sistema para saber si cumple con sus necesidades.

Enfoque de sistema, se aplicó para el estudio y profundización constante del tema de investigación, lo que propició la introducción de proyecciones concebidas en las diferentes tareas integradoras realizadas en función del objetivo propuesto.

### **Método Matemático**

Medición, para establecer comparaciones entre las aplicaciones y versiones anteriores con respecto a la presente, en la realización del cálculo de eficiencia y rapidez.

El trabajo de diploma está estructurado de la siguiente forma:

Capítulo 1 “Fundamentación Teórica”, abarca los principales conceptos manipulados en el transcurso de la investigación y una breve referencia al estado del arte de las herramientas utilizadas en el mundo para dar solución a problemas similares.

Capítulo 2 “Descripción y Análisis de la Solución Propuesta” se orienta a la implementación del sistema; así como componentes o módulos que puedan ser rehusados. Además de las descripciones de clases y algoritmos complejos desarrollados o utilizados en la investigación.

Capítulo 3 “Validación de la Solución Propuesta” donde se realizaron los diseños de las pruebas de unidad que permiten validar la solución propuesta, se describen los valores utilizados para los test y finalmente se realiza la evaluación de la ejecución del test y de los resultados obtenidos.

Además de Conclusiones, Recomendaciones, Referencias bibliográficas, Bibliografía, Glosario de términos y Anexos.



**CAPÍTULO 1**

## **FUNDAMENTACIÓN TEÓRICA**

### **1.1. Introducción**

El presente capítulo trata acerca de los principales conceptos manejados para la realización del Sistema Integrado de Transportación (SIT). Se describe como se implementaron los diferentes procesos que se llevan a cabo en el Departamento de Transportaciones Nacionales. Se tratan además las principales tecnologías aprovechadas, lenguajes utilizados; así como las herramientas y metodologías que serán usadas para desarrollar la aplicación dando una breve descripción de las mismas.

### **1.2. Procesos de Administración - Configuración y Seguridad**

#### **1.2.1. Procesos de Reservasiones a Nivel Mundial**

Internacionalmente, muchas empresas, hoteles, agencias de viajes, ofrecen servicios los cuales generan beneficios para todos los usuarios. Aprovechando las tecnologías, estos se han ido informatizando, y hoy en día se pueden acceder a través de Internet. La mayoría de ellos, para ser usados deben ser comunicados previamente, por lo cual se va haciendo necesario hacer uso de un proceso de reservación para las prestaciones. Muchas empresas se dedican a la implementación y confección de sistemas de reservación, los cuales se implementan para las necesidades del cliente, otorgando beneficios a los usuarios. Estos sistemas tienen muchas características en común como son:

- ✓ Formas especiales para el registro de reservasiones.
- ✓ Se adapta a diversos tipos de establecimientos (agencias, restaurantes, etc.).
- ✓ Las reservasiones se almacenan en la base de datos del sistema.
- ✓ Envío de notificaciones inmediatas al registrarse una reservación.

- ✓ Se conecta a sistemas ya existentes (off-line) de reservaciones.

En Internet se encuentran varios ejemplos de estos tipos de aplicaciones como: Iberworld, la cual representa uno de los grandes operadores turísticos de Europa, esta se dedica a la reservación de asientos en vuelos internacionales. Se encuentra además Alaska Airlines que es una compañía que lleva más de 75 años ofreciendo servicios, de los cuales el más característico es la reservación de viajes, al mismo tiempo de que es pionera en innovaciones de tecnologías para los clientes. Por último Radixx Solutions International, la cual fue precursora en el diseño del primer sistema de reservaciones cliente/servidor en Windows, tiene más de 10 años de experiencia brindando este tipo de servicios. Esta se especializa en la implementación de software para líneas aéreas que supera a los sistemas tradicionales en costo y rendimiento. Estos sistemas son altamente funcionales a la hora de cumplir sus objetivos específicos, pero al utilizarlos en la UCI no sería conveniente porque estos sistemas poseen funcionalidades que no se necesitan. Estos sistemas pueden ser visitados a través de estas direcciones:

- ✓ Iberworld: <http://www.iberworld.es/>
- ✓ Alaska Airline: <http://www.alaskaair.com/default.aspx>
- ✓ Radixx: <http://radixx.com/>

Al visitar las aplicaciones de este tipo se ha observado que se dividen en tres grupos principalmente, los cuales son, reservación hotelera y de turismo, reservación de restaurantes y reservaciones de viajes. En los dos primeros se puede observar que los sistemas son creados en una base estándar, para un servicio que ha de variar esporádicamente, y que se configura lo menos posible, ya que las prestaciones varían muy poco. En cambio en el último los servicios son más configurables porque se conforman los horarios de salida de los viajes y tienen en cuenta las inclemencias del tiempo, pero lo más estándar de todos se basan en las necesidades específicas del usuario; a través de información obtenida directamente del mismo; además de su procedencia y la forma de localizarlo para noticias de última hora.

En Cuba también se encuentran sistemas de reservación online, de ellos podemos citar Cubana de Aviación que es caracterizada como una de las primeras aerolíneas que abrieron la línea de vuelos comerciales, esta ofrece una buena gama de servicios siempre pensando en el cliente; existe además la empresa CUBATUR, agencia de viajes turísticos, que ofrece reservaciones de hoteles, excursiones



turísticas, renta de autos y es reconocida en Cuba por la calidad en sus procesos de reservaciones, estas pueden ser visitadas a través de las direcciones siguientes:

- ✓ Cubana de Aviación: <http://www.cubana.cu/html/espanol/index.asp>
- ✓ CUBATUR: <http://www.cubatur.cu/>

En estos sitios se observa que a pesar de que Cuba es un país pequeño, presenta desarrollo tecnológico en esta escala. En compañías de alto nivel se usa aún software propietario, por las ganancias que genera al país, permitiendo subsidiar los gastos generados en la elaboración de un sistema de este tipo. Estos ejemplos son claves en este tipo de sistema en Cuba.

En la UCI, también se cuenta con sistemas de reservación de servicios como pueden ser la solicitud del gas (<http://servicios.uci.cu/web/index.php>), reservación del pase estudiantil (<http://pase.uci.cu/>), reservación de transportación nacional (<http://smasiva.uci.cu/>), y reservación de alimentos (<http://alimentacion.uci.cu/reservacion/>). La UCI, una universidad de excelencia migrando al software libre, encuentra satisfactoria la alternativa de reutilización de código y de poder cambiar lo existente en nuevos software para el uso propio, sin tener presente que fuesen propietarios y la suma considerable que conlleva el utilizarlos.

Muchos de estos sistemas que han sido implementados para optimizar servicios, pueden ser vistos en sistemas de reservación online, estos podrían ser ejemplos claves para seguir los objetivos de implementar otro sistema de reservación pero están elaborados con herramientas y tecnologías propietarias lo que ofrece una desventaja, teniendo presente que el objetivo del país es migrar a software libre para evitar los grandes costos de los software propietarios; además es importante recalcar que los software libres después de utilizados pueden ser modificados e incluso reutilizados sin costo alguno, mientras que el software propietario por cada nuevo software se debe de pagar una considerable suma por reutilizarlos. Muchos de estos sistemas no cumplen con todas las funcionalidades requeridas por el cliente, por lo cual la necesidad de una nueva aplicación que cumpla todos estos requisitos se hace inminente.

En la utilización y observación realizada a sitios de este tipo se ha hecho presente que el software libre persigue más los estándares web utilizados en aplicaciones web. Estos muestran abiertamente como funciona el sistema de reservaciones, también se puede tomar como experiencia para obtener la mayor

ganancia y gastar la menor cantidad de recursos al país. El proceso migratorio permite que después de obtenidos los software con los cuales se piensa trabajar, estos pueden ser modificados a la forma de trabajo y necesidades de quien las trabaja y mucho más en apoyo al proceso migratorio de la UCI hacia el software libre.

### **1.2.2. Procesos de Administración – Configuración y Seguridad en Transportación Nacional de la UCI**

Transportación Nacional, surgida a raíz de la creación de la UCI, es un sistema implementado para automatizar los servicios de reservación de salida en fin de año y salida del fin de curso para los estudiantes y trabajadores de todas las provincias y municipios del país, el cual se configura para crear los bloques de salida y los viajes permitiendo así solicitar la reservación de la salida.

Cuando el sistema activa las reservaciones, por vía de los decanos o desde Intranet, se informa a todos los usuarios para que estos puedan hacer uso del sistema. La seguridad permanece presente mientras se autentica en el sistema, a través de roles, los cuales les da el acceso necesario para la disponibilidad, integridad y confidencialidad de la información que se maneja. Pero la aplicación ha tenido problemas o fallas en el sistema de seguridad, porque se puede acceder a través del URL (*Uniform Resource Locator*) a páginas que sólo el administrador debiera tener acceso; así como los accesos a la base de datos que actúan a través del servidor, de ahí es que se obtienen los datos para después ser mostrados en la aplicación, datos como las direcciones de los estudiantes. Si el estudiante desea reservar hacia otro lugar esto debe hacerse manualmente e incluso la reservación de familiares se realiza de esa manera.

La aplicación de transportación nacional tanto como la de transporte de fin de semana, aún tienen y ejecutan una buena configuración de sus viajes, pero en cuanto a la seguridad están propensas y son muy débiles a ataques de hacker, aunque en la UCI es muy poco probable, pero pudiese ocurrir. Estas vulnerabilidades podrían ser evitadas prohibiendo la autenticación y autorización insuficiente; así como las inyecciones SQL (*Structured Query Language*).

### **1.3. Sistema Integrado de Transportación**

En la UCI se efectúan o se brindan actualmente tres servicios de reservación, los cuales dan como resultado los diferentes procesos de reservaciones, ellos son:

- ✓ Proceso de reservación estudiantil para el pase de fin de semana.
- ✓ Proceso de reservación semestral de profesores y trabajadores internos.
- ✓ Proceso de reservación masiva para estudiantes, profesores y trabajadores internos.

Estas diferentes modalidades de reservaciones son administradas y configuradas por el Departamento de Transportaciones Nacionales de la UCI. La aplicación tendrá estos procesos inmersos dentro de ella a los cuales les dará solución de una forma eficiente e incrementará la seguridad de los mismos. Para esto se debe tener en cuenta los errores cometidos en las versiones anteriores que eran vulnerables a las entradas por URL, inyecciones SQL, y no realizan un estricto control de los errores con los cuales se pueden detectar fallas y ser más vulnerables, lo cual hacen inseguro el flujo de datos y a su vez la integridad de los mismos.

SIT será un sistema potencialmente navegable, que estará diseñado para que la aplicación pueda ser vista desde cualquier sistema operativo; así como sus navegadores correspondientes. Los procesos antes mencionados al estar relacionados a la aplicación reducirán las posibilidades de que no se ofrezca el servicio adecuadamente y automatizará los servicios de forma tal que el usuario sea capaz de utilizar los mismos desde cualquier computadora conectada a la red de la universidad.

El sistema se implementará de forma tal que pueda ser configurable para cualquier situación que pueda surgir de imprevisto. La seguridad será fomentada a los casos más actuales de ataques informáticos a sitios web y fundamentalmente cumpliendo con los códigos de ética internacionales y para evitar los errores cometidos en versiones anteriores.

#### **1.4. Sistemas automatizados existentes**

De los servicios que el sistema tendrá implementado ya existían anteriormente dos aplicaciones en la UCI, las cuales habían sido automatizados en otras ocasiones hasta que se decidió crear un sistema, que fuera capaz de administrar y tener implícito todos los sistemas que estuviesen relacionados con la reservación del transporte en la UCI, ya que los anteriores no contaban con algunas funcionalidades que se hacían necesarias debido al aumento de la cantidad de información manejada; además de tener Bases de Datos diferentes lo que hacía más difícil el envío y recepción de datos. Estos sistemas eran específicamente el de transportación de fin de semana y transportación nacional.

### **1.4.1. Transportación de fin de semana**

Este sistema tenía como funcionalidad que los estudiantes de Ciudad de la Habana y de otros lugares pudieran reservar transporte para salir el fin de semana hacia sus hogares o de familiares. El estudiante para reservar tenía como plazo en la semana de lunes a viernes, donde se registraba los datos del que realizó la reservación; además de los detalles de su paradero en la salida de la universidad, donde se le otorgaba al estudiante un boletín como confirmación de que había reservado, y poder efectuar así su salida desde los transportes de la UCI los sábados y regresar los domingos. Este sistema no permitía reservar para otro día salvo los fines de semana y específicamente para las horas predefinidas de 6:00 PM la salida de la UCI y la salida de la Habana a las 9:00 PM, sin tener en cuenta los casos imprevistos ni que pudiera ser utilizado en otros lugares.

### **1.4.2. Transportación Nacional**

Transportación Nacional fue creada principalmente para las salidas de fin de año y salidas de fin de curso, este proceso se realiza conjuntamente con el Ministerio de Transporte (MITRANS), donde la Vicerrectoría de formación entrega una planificación en la que el especialista de transporte elabora un plan para coordinar con el MITRANS, en cuanto a la cantidad de estudiantes por provincias y municipios del país. El MITRANS elabora las posibles salidas de transportes que puede ofrecer para después hacer entrega a la UCI de esta. Después de entregada la propuesta del MITRANS, los administradores del sistema se encargarían de configurar los bloques de viajes, posteriormente se distribuirían los viajes para así poder efectuar las salidas. Los estudiantes tenían derecho a reservar hacia su localidad, y no permitía que los mismos pudiesen reservar en casos excepcionales hacia otros lugares.

## **1.5. Conceptos asociados al dominio del problema**

### **1.5.1. Transportación Nacional o Masiva**

- ✓ Es aquella que se realiza como un todo, no separada, donde se circunscriben los medios terrestres y marítimos. El funcionamiento seguro de estos medios son los que van a garantizar el éxito de una transportación.
- ✓ En la Aplicación, es la posibilidad que se le brinda a los estudiantes y trabajadores internos de realizar su reservación, para la planificación de su traslado hacia sus lugares de procedencia al culminar el año, labores docentes o productivas.

### **1.5.2. Transportación semestral de trabajadores**

- ✓ Es la que se realiza en un período comprendido de seis meses o intrasemestral utilizando los medios de transportación masiva.
- ✓ En la Aplicación, es la que se le brinda a los trabajadores internos partiendo de transportes previamente creados con destino a las provincias centrales y orientales, dándole la posibilidad de reservar para los que estén disponibles o de anotarse en la lista de espera, en caso de que se hayan agotado las capacidades del mismo.

### **1.5.3. Transportación de estudiantes de fin de semana**

- ✓ Se realiza en un período comprendido de siete días; teniendo en cuenta los días de descanso establecidos, utilizando cualquier modalidad de la transportación masiva.
- ✓ En la Aplicación, es la posibilidad que se le brinda a los estudiantes, de reservar para los diferentes municipios de Ciudad de la Habana, para realizar la planificación de dicha transportación, en la fecha previamente definida por el administrador.

### **1.5.4. Transporte**

- ✓ Traslado de personas de un lugar a otro o bienes desde un lugar hasta otro. Estos pueden distinguirse según su uso y beneficio. Existen medios de transportes con disímiles características y propósitos, en dependencia de la necesidad de cada sujeto o sociedad.
- ✓ En la Aplicación, es la entidad que define en que se van a transportar los usuarios a su destino, siendo este de tres tipo: Ómnibus, Tren y Barco.

### **1.5.5. Viaje**

- ✓ Recorrido o itinerario que se realiza para ir de un lugar a otro utilizando cualquier medio de transporte.
- ✓ En la Aplicación, es la entidad donde se planifica la salida, se define una fecha y se determinan los usuarios que podrán disponer del mismo.

### **1.5.6. Configuración**

- ✓ Conózcase como disposición y forma de las partes que componen un todo. En informática es el conjunto de los elementos externos e internos que conforman un ordenador y hacen que tenga unas peculiaridades específicas.

- ✓ En la Aplicación, se tiene como el panel que determina características del sistema como: activación de las reservaciones, la dirección de los correos de quejas e inquietudes, si se tiene en cuenta o no los usuarios bloqueados; para cada modalidad de reservación en particular.

#### **1.5.7. Sistema Integrado de Transportación**

- ✓ Se aplica a una serie de medios de transporte que actúan simultáneamente para trasladar grandes cantidades de personas en lapsos cortos.
- ✓ En la Aplicación, es un sistema que integra tres modalidades de reservación de transporte, dígame Transportación Nacional o Masiva, Transportación Estudiantil de Fin de Semana y Transportación Nacional para Trabajadores Internos.

#### **1.5.8. Programador**

- ✓ En esta investigación se toma el concepto de programador o desarrollador como la persona que se dedica a implementar las funcionalidades requeridas por el cliente basado en la descripción de los requisitos funcionales, las funcionalidades para el desarrollo de la aplicación en todo su marco contextual.

### **1.6. Tendencias y Tecnologías Actuales**

Para la realización del trabajo, se ha realizado una investigación sobre las tendencias y las tecnologías más usadas actualmente a lo largo del mundo, que pudiesen ser útiles. Con este previo estudio se podrá comenzar un trabajo con el cual aprovechar al máximo la optimización del sistema, navegar lo más rápido posible o hacerle mucho más fácil la navegación al cliente, siendo necesarios como requerimientos mínimos el estar conectado a una red y estar trabajando en una computadora perteneciente a la misma.

El desarrollo de las TIC (Tecnologías de la Informática y las Comunicaciones) tiene un papel fundamental para el mejor funcionamiento de la institución, basado en el fundamento de que la UCI siendo una universidad que está completamente informatizada, no puede estar enajenada a estas tecnologías. Los procesos de reservación se han ido informatizando de tal manera que han permitido que el funcionamiento de los mismos haya incrementado en eficiencia a los servicios prestados. En este epígrafe se realiza un

análisis de las tecnologías y herramientas utilizadas en la elaboración del sistema que automatizará estos servicios de reservación.

### **1.6.1. Internet**

Internet, sus orígenes datan de la década del 1960 a 1970 específicamente en 1962, desde aquel momento ha sido un foco fundamental, para las redes intercomunicadas, donde se han generado diferentes servicios que actualmente se usan y tienen gran futuro en ese eje de la informática, en la actualidad todas las comunidades, compañías y empresas del mundo buscan como punto clave de desarrollo el acceso a Internet (2).

### **1.6.2. Aplicaciones Web**

Una de las tendencias que más llama la atención son las aplicaciones web que utilizan un servicio proveniente de Internet World Wide Web (*WWW*). Viendo la preferencia perseguida por la humanidad, se ha trazado como tarea fundamental el utilizar y observar que Internet está poblado de grandes aplicaciones web, tanto para ejercer, negocios electrónicos ya sea con bienes lucrativos o no; además para el uso y empleo de la información, la recreación y bienes de interés.

Al observar que los sistemas más empleados para el manejo de información son las aplicaciones web, se decidió entonces para la realización del sistema la elaboración de la misma en una aplicación web. Esta favorecería en creces la cantidad de usuarios que se podrían conectar a través de los Servicios Web (WS).

### **1.6.3. Seguridad en Aplicaciones Web**

La seguridad, se ha hecho importante desde los inicios de la computación, mucho más desde el surgimiento del Internet, el manejo de la información es cada vez mayor y mantener la misma al margen más que un derecho viene siendo un deber de cada aplicación web. Actualmente se ha incrementado el número de ataques en internet por año a aplicaciones web, no sólo a estas sino también a computadoras donde estén montados algún tipo de servidores, que tengan que ver de alguna forma con el sistema.

La mayoría de los ataques son a través de inyección SQL como es comúnmente llamado, pero la utilización del framework symfony del cual se hablará más adelante invalida la posibilidad de ataques de este tipo a sistemas implementados en este framework.

#### 1.6.4. Patrón del diseño Modelo Vista Controlador (MVC)

Este es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz del usuario, y la lógica de control en tres componentes diferentes. Este patrón está generalmente en aplicaciones web, donde la vista es la página definición de la sigla HTML (*HyperText Markup Language*) y el código que provee los datos dinámicos a la página, el modelo es el Sistema de Gestión de Base de Datos (SGBD) y el controlador es la lógica de negocio (6).

#### 1.6.5. Lenguaje de Programación

La programación web parte de las siglas WWW. Para confeccionar una página web se debe tener en cuenta tres conceptos fundamentales:

- ✓ **URL** (*Uniform Resource Locator*)
- ✓ **HTTP** (*Hypertext Transfer Protocol*)
- ✓ **HTML** (*HyperText Markup Language*)

##### 1.6.5.1. PHP

Es un lenguaje de programación utilizado para la creación de sitios web. PHP es un acrónimo recursivo que significa “PHP Hypertext Pre-processor”, (inicialmente se llamó Personal Home Page). Surgió en 1995, desarrollado por PHP Group (1).

PHP es un lenguaje de Script interpretado en el lado del servidor, utilizado para la generación de páginas web dinámicas y ejecutadas en un servidor. PHP no necesita ser compilado para ejecutarse y genera así una página HTML para ser mostrada al cliente. Para su funcionamiento necesita tener instalado Apache o IIS (*Internet Information Server*) con las librerías de PHP. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas.

Ventajas:

- ✓ Fácil de aprender.
- ✓ Se caracteriza por ser un lenguaje rápido.
- ✓ Soporta en cierta medida la orientación a objeto. Clases y herencia.
- ✓ Es un lenguaje multiplataforma: Linux, Windows, entre otros.



- ✓ Capacidad de conexión con la mayoría de los manejadores de BD: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras.
- ✓ Capacidad de expandir su potencial utilizando módulos.
- ✓ Posee documentación en su página oficial la cual incluye descripción y ejemplos de cada una de sus funciones.
- ✓ Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- ✓ Incluye gran cantidad de funciones.
- ✓ No requiere definición de tipos de variables ni manejo detallado de bajo nivel.

### **1.6.6. Framework Symfony**

Symfony es un framework que tiene como objetivo fundamental automatizar los patrones más utilizados en la elaboración de sistemas web; además de obligar a clarificar a los programadores el código, establece un estándar de código legible, encapsula operaciones complejas en simples líneas de código, ahorrando mucho más tiempo a la hora de mostrar datos directamente de la base de datos (6).

Este framework está realizado o implementado bajo el lenguaje PHP 5, ha sido utilizado en varias aplicaciones web obteniéndose de esta manera resultados satisfactorios. Es compatible con la mayoría de gestores de base datos existentes, se puede comentar sobre MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. También otorga facilidades en diferentes plataformas (6).

Está basado en el patrón de diseño web mencionado anteriormente en el epígrafe 1.6.4, del cual se puede agregar que en el modelo, symfony provee una abstracción de la base de datos, que permite una mayor facilidad de obtención de los datos haciendo a la vista y a las acciones independientes del gestor de base de datos, en la vista provee de helpers que facilitan el trabajo para los diseñadores en el código HTML de la aplicación, aunque es necesario crear una nueva vista, mantiene al modelo y al controlador original; se encarga de mantener alejados a la vista y al modelo de los detalles del protocolo utilizado (6).

#### **Características particulares de symfony**

- ✓ Es lo suficientemente flexible para amoldarse a cualquier proyecto.
- ✓ Extensible para utilizar librerías de terceros.
- ✓ Es independiente del gestor de base de datos.

- ✓ La vista, compuesta por plantillas y capas, puede ser fácilmente modificada por diseñadores sin conocimientos de este framework.
- ✓ La utilización de Helpers, ayuda a tener un código más minimizado en la presentación.
- ✓ Tiene un caché muy potente, que evita la carga del servidor y reduce el ancho de banda.
- ✓ Los generadores de ABMs (*Asynchronous Balanced Mode*) o CRUDs (*Create, Read, Update and Delete*) hacen fácil la creación y validación de formularios, en conjunto con listas de registros y búsquedas con filtros.
- ✓ Maneja un completo sistema de prueba para depurar en modo desarrollo.

### 1.6.7. Tecnologías utilizadas en el proceso de desarrollo

Para el transcurso de la elaboración del sistema desde el framework symfony se pueden encontrar diferentes tecnologías que favorecen y facilitan el uso del symfony en la elaboración de cualquier Aplicación Web. En entre ellas se puede observar:

**Prototype:** Framework escrito en Javascript que se orienta al sencillo y dinámico de aplicaciones web, amplía las posibilidades de programar en ese lenguaje, ofrece muy buenos mecanismos para la manipulación del DOM (*Document Object Model*) optimizando la obtención de objetos, clases CSS (*Cascade Style Sheet*) permitiendo un lenguaje tan conciso como el PHP.

**AJAX** (*Asynchronous Javascript and XML*): Un nuevo avance de la tecnología con el objetivo de incrementar el intercambio de datos con el servidor, con la ayuda de Prototype se hace de una forma más simple el poder actualizar datos de la aplicación sin necesidad de refrescarla y con pocas líneas de código se puede hacer esto posible.

**Script aculo:** Librería de efectos visuales que implementado en AJAX ofreciendo un ambiente visual más amigable con el usuario de forma que cada vez más se vincule a efectos visuales más complejos, y puede ser utilizado con pocas líneas de código.

**Propel:** Herramienta elaborada en software libre y utilizado para el mapeo de los objetos en la base de datos más conocido como ORM (*Object Relational Mapping*), ofrece una de las mejores abstracciones de objetos/relaciones. Permite que los datos del modelo se generen automáticamente, en casos de restricciones de llave foránea crea métodos o funcionalidades especiales para modificar o acceder a los

mismos, también una persistencia para los objetos y un servicio de consultas. Todas estas ventajas favorecen la vinculación de un programador a la hora de programar en symfony.

#### **1.6.8. Gestor de Base de Datos**

**PostgreSQL:** Se dice actualmente que es uno de los gestores de base de datos de código abierto más potente, soporta consultas complejas, integridad referencial, fue diseñado y creado para tener un mantenimiento y un ajuste menor que el de otros productos. Ninguna compañía ha reportado que haya tenido un percance en años de alta actividad, es multiplataforma extensible, diseñado además para ambientes de alto volumen de información con satisfactorios resultados, ofrece varios modos de bloqueos para controlar el acceso concurrente a los datos, múltiple tipos de datos redefinidos, y sobre todo lo más importante, ahorros considerables en costos de operación.

#### **1.6.9. Servicios Web**

La utilización de Servicios Web es importante para sistemas que necesitan tomar datos para usarlo sin la necesidad de la complejidad de consultas a una base datos y evitando la sobrecarga de la misma, es importante recalcar que opera asincrónicamente, es muy utilizado para el envío y recepción de información a través de Internet por su nivel de optimización, no requiere de conocimiento acerca de cómo están implementados, proporciona un nivel alto de interoperabilidad.

En la UCI la utilización de Servicios Web es imprescindible ya que la misma cuenta con varias aplicaciones, y la utilización de los mismos permite la integración de diferentes aplicaciones, tiene presente diferentes vías de transporte o los diferentes protocolos entiéndase HTTP, HTTPS, HTTP-R, BEEP, JABBER, IIOP, SMTP o FTP.

Los Servicios Web usan muchos tipos de tecnologías con las cuales se trabajó en el transcurso de la elaboración del sistema.

#### **Ellas son:**

- ✓ XML (*Extensible Markup Language*)
- ✓ SOAP (*Simple Object Access Protocol*)
- ✓ WSDL (*Web Service Definition Language*)
- ✓ UDDI (*Universal Description, Discovery and Integration*)

### 1.6.9.1 Servicios Web usados en la Aplicación

**Autenticación:** Servicio de donde se obtienen y verifican los datos del usuario que se autentica en la aplicación.

**Identificación:** Servicio de donde se obtienen detalles más generales e identificativos de los usuarios.

**Akademios:** Servicio de donde se obtienen detalles más específicos de los usuarios que son estudiantes.

**Assets:** Servicio de donde se obtienen detalles más específicos de los usuarios que son trabajadores directamente vinculados con la UCI.

**Personal:** Servicio de donde se obtienen detalles más específicos de los usuarios que son trabajadores de áreas tercerizadas.

### 1.6.10. Protocolo LDAP

La UCI un lugar donde residen alrededor de 20 000 personas en el cual cada vez se hace más difícil el encontrar una persona u obtener datos de la misma, para ello se hace un poco necesario el protocolo LDAP (*Lightweight Directory Access Protocol*) o sea Protocolo de Ligerero Acceso a Directorios, este permite ayuda en la autenticación de los usuarios y la verificación de lo que hace único a un usuario, su usuario y su contraseña, donde la contraseña verifica la autenticidad del usuario (4).

En otras palabras se puede decir que el LDAP es un protocolo a nivel de aplicación que permite el acceso a un directorio ordenado y distribuido para buscar diversa información en un entorno de red, muchas veces es encontrado o utilizado para la autenticación de usuarios dado que se utiliza mucho para almacenar los datos de usuario (usuario, contraseña), aunque también puede ser utilizado para guardar datos específicos de otro tipo relacionado con el usuario.

Presenta muchas ventajas, entre ellas podemos decir que su rapidez en la lectura de registros, la mayoría de las aplicaciones presentan soporte para LDAP, los servidores LDAP son fáciles de instalar, mantener y optimizar y además se podría agregar que permite múltiples directorios independientes.

## 1.7. Herramientas a Utilizar

### 1.7.1. Zend Studio 5.5

Es un editor PHP dirigido a la elaboración de Aplicaciones Web en este lenguaje, muy utilizado por la comunidad del PHP a nivel mundial, ofrece depuración de código, además soporta lenguajes utilizados en la web como son javascript, css, html, etc. Es un editor multiplataforma elaborado en software libre, el editor ofrece herramientas de programación, ofrece ayuda contextual para las sentencias del lenguaje y otras muchas ventajas.

### 1.7.2. PostgreSQL 8.2

El gestor PostgreSQL 8.2 es un gestor potente para la utilización del PostgreSQL, presenta una interfaz amigable, sobre todo elaborado en software libre y multiplataforma, con esta se obtiene 20% más de performance (*Rendimiento*), incluye ordenamientos más rápidos en disco y en memoria, los usuarios pueden ganar más eficiencia en sistemas de análisis de datos y sobre todo resulta menos costoso en relación con grandes gestores de base de datos.

### 1.7.3. Net Beans 6.5

Esta versión tiene nuevos soportes para lenguajes tanto web como desktop. Importante recalcar que es un software libre, multiplataforma, esta versión es más rápida a la hora de compilar, permite un completamiento de código más extenso al estar utilizando el symfony como framework, ofrece ventajas técnicas y económicas para todo aquel que desee usarlo, presenta un plugin para el desarrollo de la tecnología AJAX, presenta una interfaz amigable y sobre todo permite depurar el código realizado.

Se han seleccionado las siguientes herramientas porque se ha entendido que con su uso sería más factible, ágil y robusta la elaboración del SIT. Teniendo en cuenta algunas ventajas de las herramientas antes mencionadas las cuales no son todas las que presenta, se entendió que para el programador resultaría ventajoso el uso de estos IDE (*Integrated Development Enviroment*) y el gestor de base de datos, dado la integración favorable que presentan.

## **1.8. Conclusiones**

En la elaboración del capítulo se describe el uso de las tecnologías y herramientas usadas en la confección del trabajo y un marco para poder guiarse mientras se trabajaba en el presente proyecto. Es de mucha utilidad el conocer las ventajas que traería el uso de las antes mencionadas para la elaboración del trabajo; así como conocer las ventajas, cuáles son las sobresalientes y más utilizadas tecnologías en construcción de Aplicaciones Web, para eso se realizó una investigación llegando a fondo de sistemas elaborados con la misma concepción, para que sirviera de apoyo a los que fuesen a realizar un producto de este tipo. De esta forma se concluye con la fundamentación teórica y se da paso a la descripción y análisis de la solución propuesta.



## DESCRIPCIÓN Y ANÁLISIS DE LA SOLUCIÓN PROPUESTA

### 2.1. Introducción

Después de haberse realizado un estudio detallado de la bibliografía referente al framework de desarrollo que se utilizó para la implementación en symfony y todo lo concerniente a los procesos de reservación de transporte en la UCI, se estarán introduciendo en este capítulo los diferentes conceptos a tener en cuenta a la hora de realizar la implementación de los módulos de Administración – Configuración y Seguridad del Sistema Integrado de Transportación (SIT), quedando dividido en cuatro epígrafes principales los cuales son: análisis de las soluciones propuestas por los analistas, análisis de posibles implementaciones y componentes o módulos que son reutilizados, descripción de los algoritmos no triviales utilizados e información detallada de las clases realizadas.

### 2.2. Descripción de la solución propuesta

#### 2.2.1. Configuración

Cuando se selecciona en el menú del SIT el vínculo Configurar figura 2.1, se muestra el panel de configuración del sistema, donde se tiene un formulario con un campo de texto para definir el título que aparecerá en el sistema, seguido por otros tres campos de textos donde se podrán poner los correos de los responsables de atender los problemas planteados por los usuarios durante el uso de la aplicación. Se cuenta con un menú, el cual define la cantidad de días previos que se le concede a los trabajadores internos para confirmar su reservación, tiene un checkbox donde se activan las reservaciones de forma general, lo cual define que una vez que las reservaciones están activas no se pueden borrar los viajes ni los transportes por motivos de seguridad. Seguido a esto, tres checkbox los cuales definen el estado de las tres modalidades de reservación que conforman el SIT, los cuales especifican si estarán activas o no,

dándole la posibilidad al administrador de habilitar y deshabilitar las aplicaciones de forma independiente, como se muestra en la figura 2.2.



Fig 2.1 Enlace Configurar

**Título de la aplicación:** Servicios Comunitarios - Sistema de Reservasiones

**Enviar Quejas y Sugerencias a:** transportacion@uci.cu Transportación Nacional  
 transportacion@uci.cu Transportación de Trabajadores Internos  
 transportacion@uci.cu Transportación de Fin de Semana

**Días para confirmar la reservación:** 3

**Activar las reservaciones:**  *Una vez que se activan las reservaciones NO se pueden eliminar los viajes ni los transportes...*

**Activar/desactivar aplicaciones**

- Transportación Nacional
- Transportación de Trabajadores Internos
- Transportación de Fin de Semana

**Activar/desactivar el chequeo de personas bloqueadas**

- Activar/desactivar el chequeo de personas bloqueadas

Guardar cambios

Fig 2.2 Configuración general de la aplicación

### 2.2.2. Autenticar Usuario

El usuario ingresa sus credenciales en el formulario que se muestra en la figura 2.3. y selecciona la opción enviar, se verifica con el LDAP si los datos introducidos son correctos, se buscan los datos personales del usuario en los servicio web de la UCI, se le asignan las credenciales y las funcionalidades, asignándole así el nivel de acceso dentro del sistema, se filtran con los datos obtenidos previamente los bloques a los que el usuario tendrá acceso, filtrando también los viajes del bloque al cual accedió, esto si no ha sido



bloqueado por el administrador del sistema, de haber sido bloqueado se le mostrará en la página los detalles de su bloqueo y no tendrá acceso a las reservaciones hasta que no se haya cumplido el tiempo de bloqueo, como se muestra en la figura 2.4

Fig 2.3 Formulario para registrarse en la aplicación


Datos Del Usuario Bloqueado		
	<b>Nombre:</b> José Alejandro García Calderón	<b>Área:</b> Facultad 1
	<b>Municipio:</b> SAN LUIS PR	<b>CI:</b> 85112011627
	<b>Provincia:</b> PINAR DEL RIO	
Datos del Bloqueo		
<b>Fecha Inicio:</b> 2009-05-10	<b>Fecha Fin:</b> 2009-05-12	
<b>Observaciones:</b> Prueba de estudiante Bloqueado		

Fig 2.4 Usuario bloqueado

### 2.2.3. Gestionar Accesos

Cuando se selecciona en el menú del SIT el vínculo Accesos, aparece un listado con datos de las personas que han sido bloqueadas, en la parte inferior izquierda se visualiza un vínculo titulado Bloquear,

si es seleccionado se visualiza un formulario donde se selecciona el usuario al cual se le va a impedir el acceso a la aplicación, puesto que el administrador determinó que incurrió en alguna falta, por lo que se le aplica como medida impedirle viajar en la transportación brindada por la universidad. Se introduce en un campo de texto su Id de expediente, carné de identidad o usuario, se visualizan sus datos, como se muestra en la figura 2.5, se selecciona la fecha de inicio y fin del bloqueo, se registra en un textarea el motivo por el cual está siendo privado del servicio, y se guardan los datos, como se muestra en la figura 2.6.

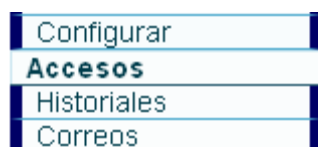


Fig 2.5 Enlace Accesos

Nombre y apellidos	Fecha de inicio	Fecha de fin	Observación	Tipo de transporte
José Alejandro García Calderón	2009-05-10	2009-05-12	Prueba de estudiante Bloqueado	TRANSPORTACION ESTUDIANTIL

Fig 2.6 Listado de usuarios bloqueados

#### 2.2.4. Bloquear Provincia

Cuando se selecciona en el menú del SIT el vínculo Accesos se visualiza un enlace titulado Bloqueo de provincias, como se muestra en la figura 2.6, se selecciona la fecha de inicio y fin del bloqueo, se registra en un textarea el motivo por el cual está siendo privado del servicio. Se tiene un menú con las provincias del país se selecciona la que se quiere bloquear, pues al administrador debido a alguna actividad de la universidad, se le orienta que el fin de semana no pueden salir de pase los estudiantes de dicha provincia, una vez seleccionada, se selecciona el tipo de viaje (Ida/Regreso) y se guardan los datos, como se muestra en la figura 2.7.

[Bloquear Provincias]  [Bloquear Usuario]

Provincia	Fecha de inicio	Fecha de fin	Descripción	Tipo de viaje
PINAR DEL RIO	2009-05-07 00:00:00	2009-05-15 00:00:00	Bloqueo de Prueba	IDA
LA HABANA	2009-05-20 00:00:00	2009-05-30 00:00:00	Bloqueo de prueba1	REGRESO

Fig 2.7 Listado de provincias bloqueadas

### 2.2.5. Gestionar Bloques

Cuando se selecciona en el menú del SIT el vínculo Bloques figura 2.8 se visualiza un formulario donde se tiene un campo de texto para introducir el nombre del bloque, se selecciona el tipo de reservación (si es estándar los estudiantes sólo podrán reservar para el municipio que tienen registrado en la BD UCI, si es personalizado el sistema brinda la posibilidad de reservar para cualquier municipio del país), como se muestra en la figura 2.10. Una vez creado el bloque, da la posibilidad de adicionarle reglas que son las que van definir quienes son los usuarios que tendrán acceso al mismo. Para adicionar reglas se selecciona el enlace Adicionar Regla donde se mostrará una interfaz con un menú que contiene los conceptos, estos pueden ser: municipio, tipo de persona, año, mundial y área. En dependencia del concepto seleccionado se genera otro menú con los criterios correspondientes al concepto seleccionado, se le define un orden en que se quiere verificar dicha regla y se guardan los datos seleccionados como se muestra en la figura 2.11. Un vez guardado se accede a una lista de bloques donde se pueden gestionar reglas, como se muestra en la figura 2.9

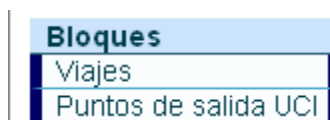


Fig 2.8 Enlace a Bloques

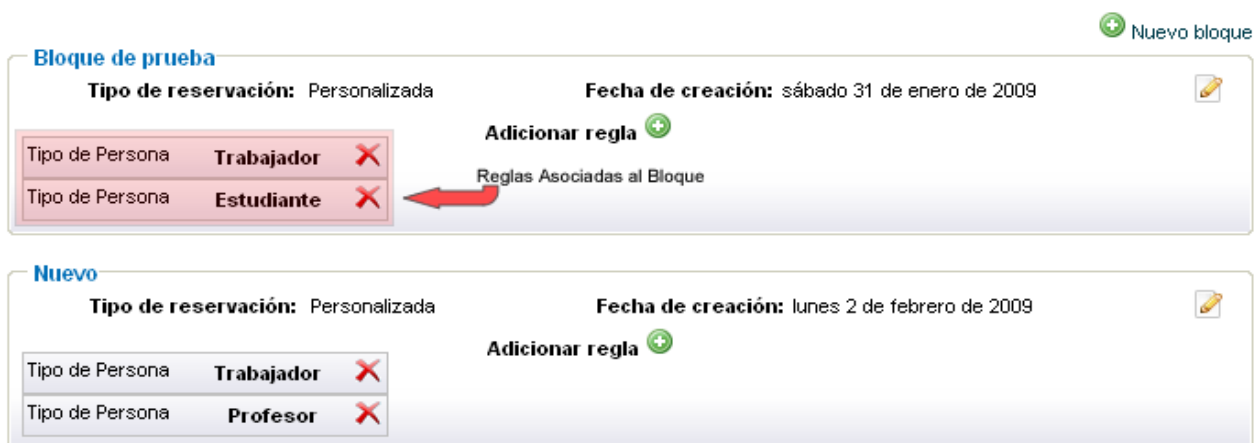


Fig 2.9 Bloques de transportación nacional

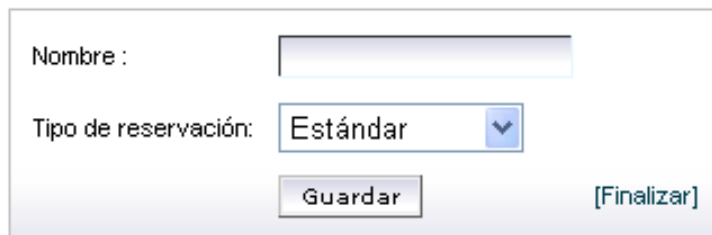


Fig 2.10 Crear bloque de transportación

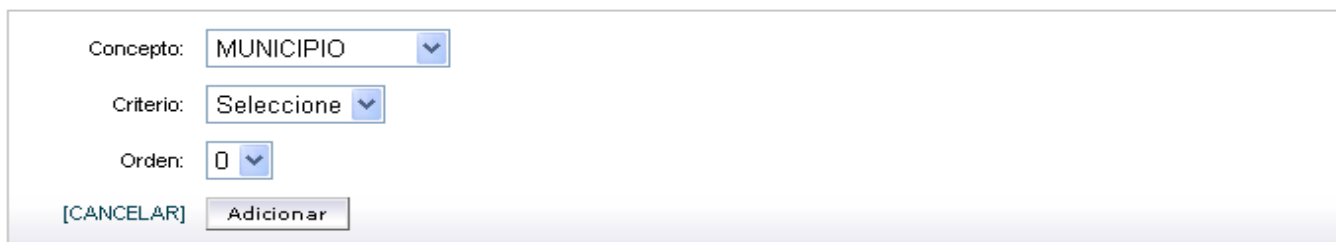


Fig 2.11 Adicionar regla

### 2.2.6. Gestionar Boletines

Cuando se selecciona en el menú del SIT el vínculo Boletines figura 2.12, se muestra un formulario donde se tiene una selección con el listado de las facultades, se elige la facultad a la cual se le va a generar los boletines, donde automáticamente se genera una selección con los grupos correspondientes a la misma, se selecciona uno y este a su vez genera la selección de los estudiantes reservados de ese grupo, teniendo la opción de generar los boletines asumiendo la selección del usuario con el debido acceso como se muestra en la figura 2.13.



Fig 2.12 Enlace Boletines

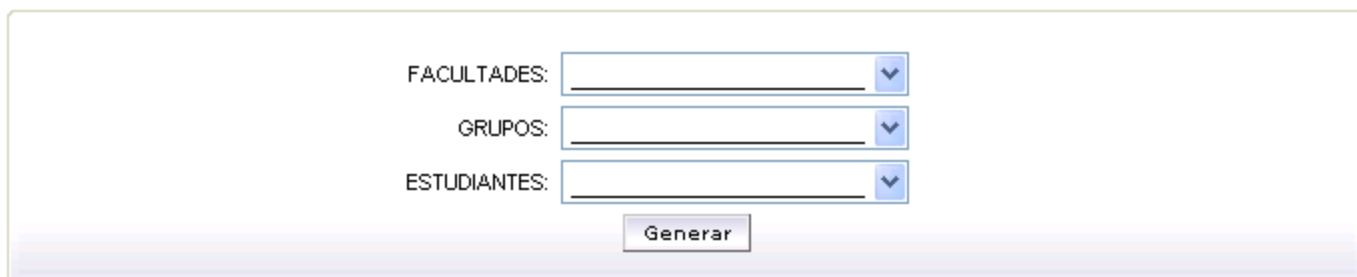
A form with three dropdown menus. The first is labeled 'FACULTADES:', the second 'GRUPOS:', and the third 'ESTUDIANTES:'. Each dropdown menu has a blue arrow pointing down. Below the dropdowns is a button labeled 'Generar'.

Fig 2.13 Generar boletines

### 2.2.7. Gestionar Paradas Intermedias

Cuando se selecciona en el menú del SIT el vínculo Paradas figura 2.14 se tiene una lista con las paradas intermedias, que para las Transportaciones Nacionales son los municipios por donde deberá pasar el transporte para llegar a su destino final, dando la posibilidad de eliminar las que el administrador seleccione, esto da la posibilidad de filtrar por aplicación dichas paradas, como se muestra en la figura 2.15. Se tiene en la parte superior derecha el enlace Nueva, que permite adicionar una nueva parada intermedia, se selecciona de un menú la modalidad de reservación a la cual se quiere adicionar, generándose un menú con las posibles paradas intermedias teniendo en cuenta la modalidad

seleccionada. Una vez que el administrador del SIT haya realizado su selección se guardan los datos como se muestra en la figura 2.16.



Fig 2.14 Enlace a la funcionalidad paradas intermedias

## Paradas intermedias

**Municipios a utilizar como parada intermedia para los transportes.** + [Nueva]

TRANSPORTACION NACIONAL ▼ Filtrar

⏪ ◀ 1 - 2 - 3 - 4 - 5 ▶ ⏩ Página: 1 de un total de 23

Aplicación	Parada	X	☐
TRANSPORTACION NACIONAL	MANTUA-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	MANTUA-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	MINAS DE MATAHAMBRE-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	MINAS DE MATAHAMBRE-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	VIÑALES-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	VIÑALES-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	LA PALMA-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	LA PALMA-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	BAHIA HONDA-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	BAHIA HONDA-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	CANDELARIA-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	CANDELARIA-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	SAN CRISTOBAL-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	SAN CRISTOBAL-PINAR DEL RIO	X	☐
TRANSPORTACION NACIONAL	LOS PALACIOS-PINAR DEL RIO	X	☐

Fig 2.15 Paradas intermedias

The screenshot shows a web-based form with two main sections. On the left, there is a label 'Aplicación\*' followed by a dropdown menu. On the right, there is a label 'Parada\*' followed by a scrollable list of municipalities. Below the form, there are two buttons: 'Guardar' and 'Cancelar'.

Aplicación*	Parada*
	CANDELARIA
	SAN CRISTOBAL
	LOS PALACIOS
	CONSOLACION DEL SUR
	PINAR DEL RIO
	SAN LUIS
	SAN JUAN Y MARTINEZ
	GUANE
	MARIEL
	GUANAJAY
	CAIMITO
	BAUTA
	SAN ANTONIO DE LOS BAÑOS
	BEJUCAL
	SAN JOSE DE LAS LAJAS
	JARUCO
	SANTA CRUZ DEL NORTE
	MADRUGA
	NUEVA PAZ
	SAN NICOLAS
	GUINES

Fig 2.16 Crear parada intermedia

### 2.2.8. Gestionar Puntos de Salida

Cuando se selecciona en el menú del SIT el vínculo Punto figura 2.17 se visualiza un formulario donde se introduce en un campo de texto el nombre del nuevo punto de salida, el cual se usará para la transportación estudiantil de fin de semana y se guardan los datos introducidos, como se muestra en la figura 2.18. Una vez guardados se muestra un listado con los puntos que hayan sido salvados, brindando la posibilidad de modificarlos o eliminarlos, como se muestra en la figura 2.19.



Fig 2.17 Enlace Punto

Punto:

**Fig 2.18** Crear o editar punto

### Transportación estudiantil de fin de semana

Punto
100 y 31 (Obelisco)
100 y Boyeros
Carlos III
Ciudad Deportiva
Novia del Medio Día

**Fig 2.19** Listado de puntos

#### 2.2.9. Gestionar Puntos de Salida UCI

Cuando se selecciona en el menú del SIT el vínculo Punto de salida UCI figura 2.20, se visualiza un formulario donde se introduce en un campo de texto el nombre del nuevo punto de salida, el cual se usará para la transportación nacional, como se muestra en la figura 2.21 y se guardan los datos introducidos. Una vez guardados se muestra un listado con los puntos de salida UCI que ya hayan sido guardados, brindando la posibilidad de modificarlos o eliminarlos, como se muestra en la figura 2.22.



**Fig 2.20** Enlace Puntos de salida UCI



Fig 2.21 Nuevo punto de salida

Puntos de Salida		
Plaza de las Banderas		
Parque Doc. 1		
Canchas		
Explanada Nodo Central		
100 y 31 (Obelisco)		

[Nuevo Punto](#)

Fig 2.22 Listado de puntos de salida UCI

### 2.2.10. Gestionar Semestre

Cuando se selecciona en el menú del SIT el vínculo Semestres figura 2.23 se visualiza un formulario donde se tiene un menú con los semestres, se selecciona el semestre y automáticamente se muestra los meses que este tiene definido, como se muestra en la figura 2.24, brindando la posibilidad de asignarle o eliminarle meses al semestre seleccionado, como se muestra en la figura 2.25, esto sucede solamente si las reservaciones están desactivadas.

Fig 2.23 Enlace a Semestres

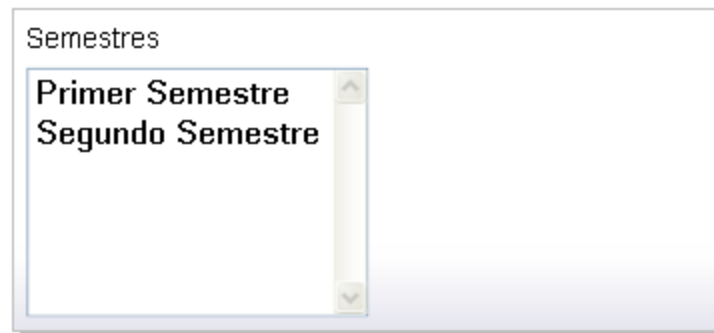


Fig 2.24 Gestionar semestres

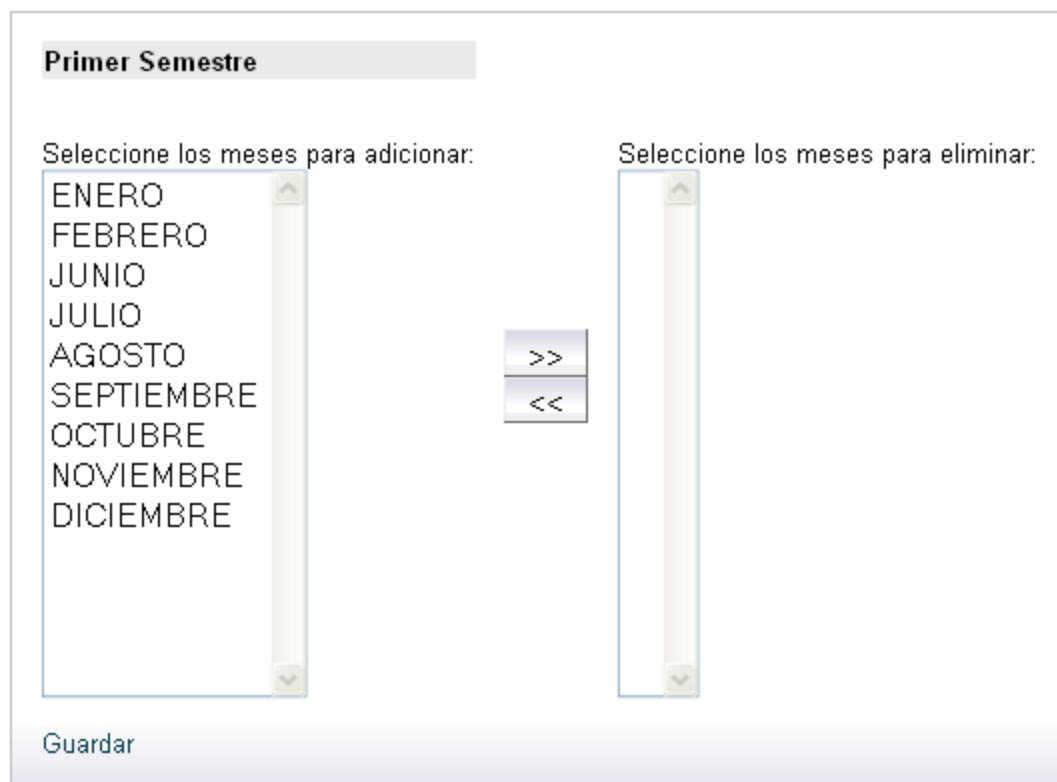
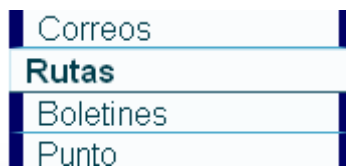


Fig 2.25 Adicionar o eliminar semestres

### 2.2.11. Gestionar Rutas

Cuando se selecciona en el menú del SIT el vínculo Rutas figura 2.26 se visualiza un formulario donde se tiene inicialmente cuatro campos de texto para introducir el itinerario de ida y regreso, el número de la ruta y el nombre respectivamente, además se tienen cuatro menús donde, se selecciona la aplicación a la que va a pertenecer la ruta, el punto de salida UCI definido previamente, se selecciona una de las dos provincias que se le brinda el servicio de salida el fin de semana (La Habana o Ciudad Habana), se genera un menú con los municipios de dichas provincias, se selecciona el municipio que será el destino u origen de la ruta, como se muestra en la figura 2.27 y se guardan los datos, brindado después el listado de las rutas definidas, como se muestra en la figura 2.28 posibilitando la modificación o eliminación de las mismas.



**Fig 2.26** Enlace Rutas

Fig 2.27 Nueva ruta

	Número ruta	Ruta
	1	100 y 31

Fig 2.28 Listado de rutas

### 2.2.12. Gestionar Roles

Cuando se selecciona en el menú del SIT el vínculo Roles figura 2.29 se visualiza un formulario donde se introduce en un campo de texto el nombre del nuevo rol, luego se procede a seleccionar la modalidad de reservación a la cual pertenecerá, después se seleccionan las funcionalidades a la que dicho rol tendrá acceso, como se muestra en la figura 2.30 y se guardan los datos. Una vez guardados se muestra un

listado con los roles que hayan sido guardados, como se muestra en la figura 2.31, brindando la posibilidad de modificarlos o eliminarlos.

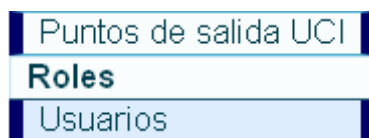
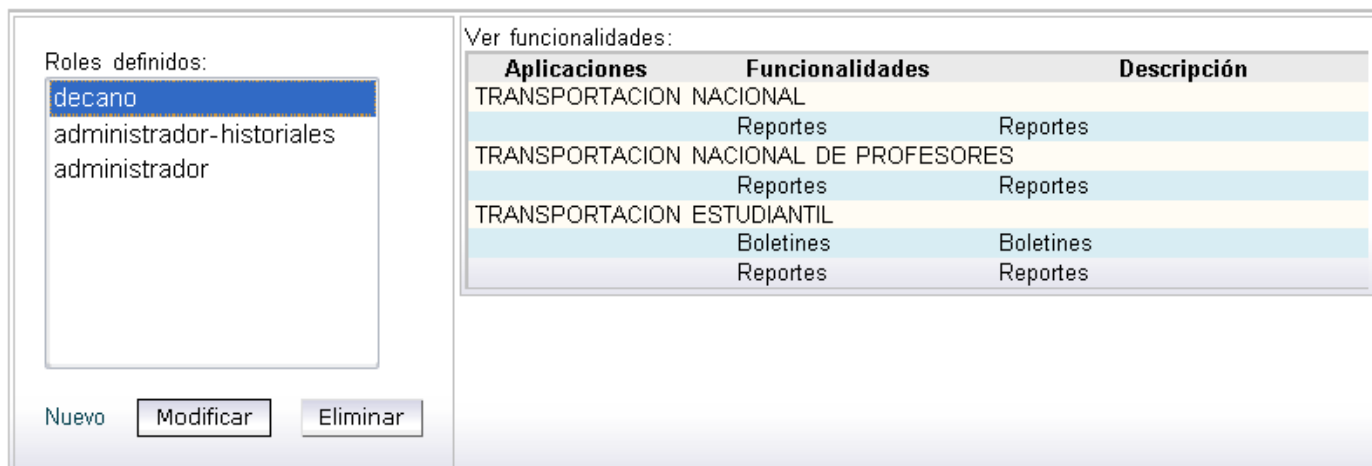


Fig 2.29 Enlace Roles

A web form for creating a new role. It includes a 'Nombre:' text input field, an 'Aplicación:' dropdown menu currently showing 'TRANSPORTACION NACIONAL', and two columns for selecting permissions. The left column, 'Seleccione permisos para adicionar:', contains a list of permissions: Bloques, Viajes, Accesos, Cuotas, Roles, Usuarios, Semestres, Rutas, Boletines, and Listados. The right column, 'Seleccione permisos para eliminar:', is currently empty. Between the columns are two buttons: '>>' and '<<'. At the bottom left of the form is a 'Finalizar' button.

Fig 2.30 Crear nuevo rol



**Fig 2.31** Listado de roles y funcionalidades a las que se tiene acceso

### 2.2.13. Gestionar Transportes

Cuando se selecciona en el menú del SIT el vínculo Viajes figura 2.32, se visualiza una lista con los viajes divididos por modalidades de reservaciones, como se muestra en la figura 2.33, se selecciona el ícono del transporte en los viajes a los que se les va a crear, se muestra una lista con todos los transportes que se tengan creados, como se muestra en la figura 2.34. Para crearle un nuevo transporte al viaje se selecciona el vínculo Crear Transporte, luego se muestra un formulario para introducir los datos del transporte, una vez introducidos los datos teniendo en cuenta el tipo de transporte que se desea crear (Ómnibus, Barco o Tren). Se visualiza un formulario correspondiente al tipo seleccionado donde se introducirán las particularidades de cada uno, como se muestra en la figura 2.35.



**Fig 2.32** Enlace Viajes

[Crear viaje]

Transportación Nacional						
Tipo	Fecha	Inicio período reservación	Fin período reservación	Reglas aplicadas	Acciones	
✉	Ida	26/02/2009 09:44	26/02/2009 09:44	26/02/2009 09:44	+ No hay reglas	✖ Icono de Transporte → Distribuciones
✉	Regreso	19/01/2009 09:57	26/02/2009 09:57	27/02/2009 09:57	+ Provincia <b>CIENFUEGOS</b> ✖	✖ Distribuciones
✉	Regreso	27/03/2009 04:56	22/02/2009 04:56	03/03/2010 04:56	+ Provincia <b>CAMAGUEY</b> ✖ Provincia <b>PIJAR DEL RIO</b> ✖	✖ Distribuciones
✉	Ida	12/03/2009 04:48	01/02/2008 04:49	31/03/2009 04:49	+ Provincia <b>CIENFUEGOS</b> ✖ Tipo de Persona <b>Estudiante</b> ✖	✖ Distribuciones
✉	Ida	01/02/2009 04:56	01/02/2009 04:56	31/03/2009 04:56	+ Tipo de Persona <b>Trabajador</b> ✖ Tipo de Persona <b>Estudiante</b> ✖	✖ Distribuciones
+ Transportación de Trabajadores Internos						
+ Transportación de Fin de Semana						

Fig 2.33 Listado de viajes donde se señala el ícono de transporte

Nombre	Capacidad	Fecha hora s	Fecha hora II	Punto salida	Punto llegada	Municipio	Tipo
Prueba	45	19/05/2009 3:06 pm	25/05/2009 3:40 pm	UCI	UCI	ISLA DE LA JUVENTUD	OMNIBUS 

Fig 2.34 Listado de transportes de un viaje

<b>Capacidad*:</b>	45
<b>Nombre transporte*:</b>	<input type="text"/>
<b>Fecha hora s*:</b>	2009-05-19 15:06 ...
<b>Fecha hora II*:</b>	<input type="text"/> ...
<b>Punto llegada*:</b>	<input type="text"/>
<b>Profesor responsable*:</b>	<input type="text"/>
<b>Municipio:</b>	PINAR DEL RIO ▼ <input type="text"/>
<b>Punto de salida UCI*:</b>	Plaza de las Banderas ▼
<b>Tipo transporte*:</b>	OMNIBUS ▼
<b>Asientos disponibles:</b>	desde-- 1 --hasta-- 40

Fig 2.35 Formulario para la creación de un transporte

#### 2.2.14. Gestionar Usuarios

Cuando se selecciona en el menú del SIT el vínculo Usuarios figura 2.36 se visualiza un formulario donde se introduce en un campo de texto el Id de expediente, carné de identidad o usuario, que se desea asignar permisos al sistema, como se muestra en la figura 2.37, se visualizan sus datos, como se muestra en la figura 2.38 y se procede a seleccionar el rol que dicho usuario va a tener en la aplicación, luego se guardan los datos seleccionados, una vez guardados se muestra un listado con los usuarios que tienen acceso a la aplicación con un rol diferente al de usuario, como se muestra en la figura 2.39, brindando la posibilidad de modificarles el rol o quitar los permisos que tenían previamente.

Roles
<b>Usuarios</b>
Viajero



Fig 2.36 Enlace Usuarios

Formulario con un campo de texto etiquetado "Usuario o Solapín" y dos botones: "Buscar" y "Listado".

Fig 2.37 Formulario para buscar los datos de un usuario

Formulario de detalles de usuario. Incluye una foto, nombre, CI, Solapín, roles (administrador-historiales) y un botón "Guardar".

Aplicaciones	Funcionalidades	Descripción
TRANSPORTACION NACIONAL		
	Historiales	Traza y reservaciones
TRANSPORTACION NACIONAL DE PROFESORES		
TRANSPORTACION ESTUDIANTIL		

Fig 2.38 Datos de un usuario buscado

Lista de usuarios con roles en el sistema. Cada entrada muestra una foto, nombre, CI, Solapín y un enlace "ver".

	<b>Nombre:</b> Iran Roberto Rodríguez Moreno	<b>CI:</b> 85120601622
	<b>Nombre:</b> José Alejandro García Calderón	<b>CI:</b> 85112011627

Fig 2.39 Listado de usuarios con roles en el sistema

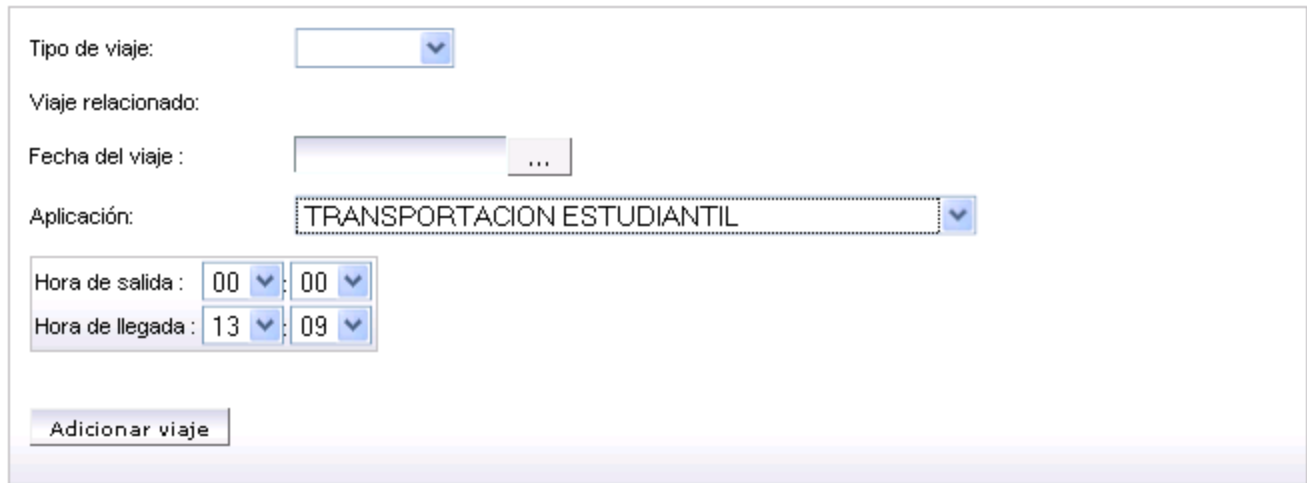
### 2.2.15. Gestionar Viajes

Cuando se selecciona en el menú del SIT el vínculo Viajes figura 2.32, inicialmente se visualiza una lista con los viajes divididos por aplicaciones, como se muestra en la figura 2.40. Para crear un nuevo viaje se selecciona la opción Crear Viaje que se encuentra en la parte superior derecha, luego aparece un

formulario donde se introducirán los siguientes datos: una selección con el tipo de viaje (ida / regreso), seguidamente se genera una selección donde aparecen los viajes que se pueden relacionar con el viaje que se esté creando. En caso de no seleccionar ninguno, se selecciona la fecha en la que se realizará de un campo de tipo fecha y en dependencia de la modalidad seleccionada, se procederá a llenar los datos particulares de cada una. Para Transportación Estudiantil se seleccionarán la hora de salida y llegada, como se muestra en la figura 2.41. Para Transportación Nacional se seleccionará el bloque al que pertenecerá el viaje, así como la fecha de inicio y fin del período de reservación, como se muestra en la figura 2.42. Para Transportación Nacional de Profesores se selecciona el tipo de reservación que se desea brindar, estándar o personalizada, estándar si sólo se va a dar la posibilidad de reservar a los profesores que aparecen registrados en la base de datos, y personalizada cuando se le permite reservar para sus destinos siempre que haya transporte, como se muestra en la figura 2.43.



**Fig 2.40** Listado de viajes por modalidad de reservación



Tipo de viaje:  ▼

Viaje relacionado:

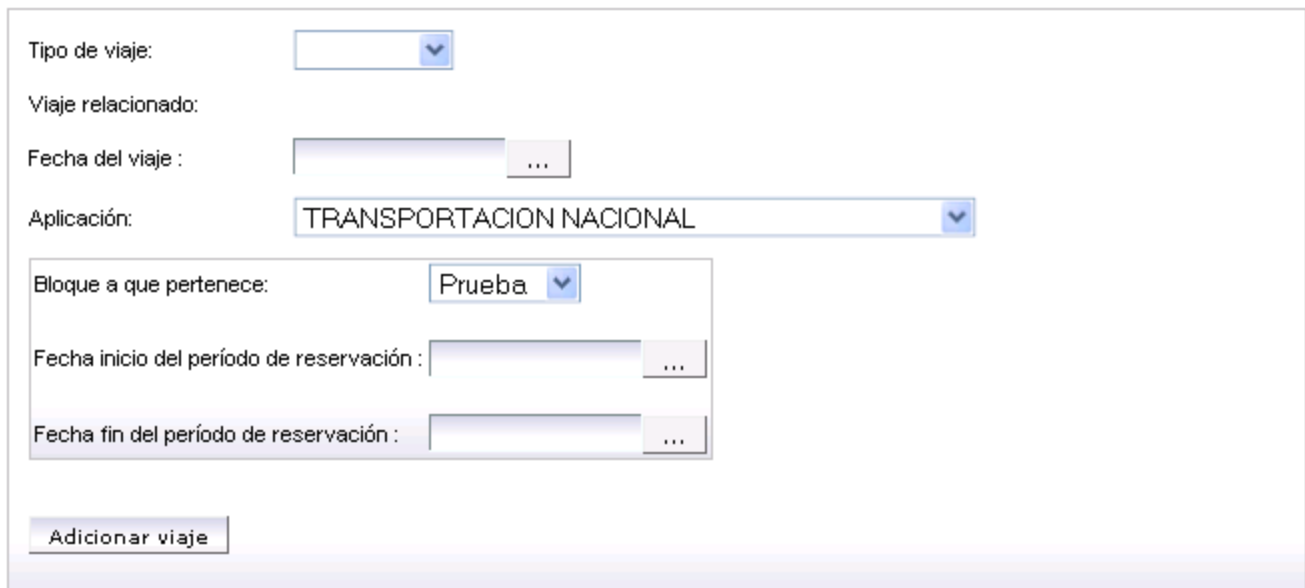
Fecha del viaje :  ...

Aplicación:  ▼

Hora de salida :  ▼ :  ▼

Hora de llegada :  ▼ :  ▼

**Fig 2.41** Formulario para crear un viaje para la transportación estudiantil



Tipo de viaje:  ▼

Viaje relacionado:

Fecha del viaje :  ...

Aplicación:  ▼

Bloque a que pertenece:  ▼

Fecha inicio del período de reservación :  ...

Fecha fin del período de reservación :  ...

**Fig 2.42** Formulario para crear un viaje para la transportación nacional

Tipo de viaje:  ▼

Viaje relacionado:

Fecha del viaje :  ...

Aplicación:  ▼

Tipo de reservación:  ▼

Fecha inicio del período de reservación :  ...

Fecha fin del período de reservación :  ...

**Fig 2.43** Formulario para crear un viaje para la transportación nacional de trabajadores internos

### 2.3. Valoración crítica del diseño propuesto por el analista

Los analistas y diseñadores del sistema han realizado una propuesta de acuerdo con el desarrollo que se quiere llevar a cabo. Primero en el momento de implementar el sistema y luego a la hora de ponerlo en funcionamiento.

En el diseño propuesto se definen las clases necesarias para un correcto funcionamiento del sistema así como los atributos y métodos que deben tener las mismas, mostrando una idea clara de lo que se debe implementar. Juntamente con una serie de diagramas de interacción, que explican la relación entre las clases y como son llamados los métodos y sentencias dentro de cada una de ellas, generando así toda la información necesaria para conocer el orden de implementación de las acciones.

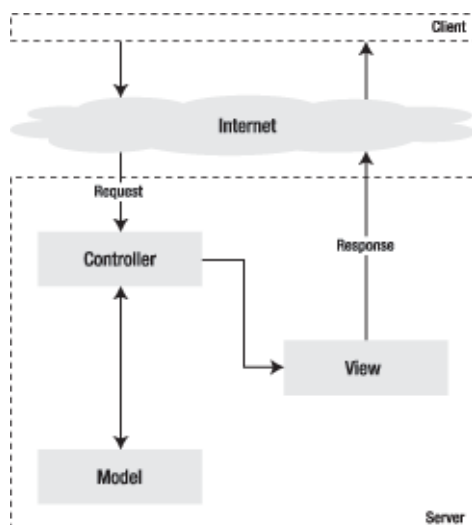
#### 2.3.1. El patrón Modelo Vista Controlador (MVC)

Symfony está basado en un patrón clásico del diseño web conocido como arquitectura MVC, que está formado por 3 niveles:

- El modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La vista transforma el modelo en una página web que permite al usuario interactuar con ella.

- El controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

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. Por ejemplo, una misma aplicación debe ejecutarse tanto en un navegador estándar como en 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.



**Fig 2.44** El patrón MVC

## **2.4. Análisis de posibles implementaciones y componentes o módulos que son reutilizados**

Los plugins permiten agrupar todo el código diseminado por diferentes archivos y reutilizar este código en otros proyectos. Además permiten encapsular clases, filtros, mixins, helpers, archivos de configuración, tareas, módulos, esquemas y extensiones para el modelo, fixtures, archivos estáticos, etc.

Dentro de los Plugins utilizados se cuenta con: sfDateTimePlugin: añade una biblioteca de manipulación de fechas a symfony, brindando una serie de funciones para el trabajo con fechas, dichas funciones pueden ser complejas como sencillas, permitiendo legibilidad al código. SfPropelParanoidBehaviorPlugin: deshabilita el borrado de los objetos y lo reemplaza por la actualización de una columna llamada fecha\_desactivacion. Permitiendo de esta forma mantener datos históricos referentes a la información generada en todo el proceso de administración.

Teniendo en cuenta la necesidad del utilizar de servicios web y la optimización de su uso se implementó una clase llamada classws, la cual se encarga de toda la interacción con dichos servicios de una forma eficiente y óptima, siendo necesaria su uso en casi todos los módulos de la aplicación.

Se hace necesario llevar un estricto control de todas las operaciones realizadas por los usuarios en el sistema, es necesaria además la implementación de una clase scTraza que es la que se encarga de dicho control siendo utilizada en todos los módulos de la aplicación.

## **2.5. Descripción de los algoritmos no triviales utilizados**

El algoritmo Autenticar, una vez que el usuario introduce sus datos, se verifican con el LDAP, de ser correctos se procede a la asignación de las credenciales las cuales en dependencia del rol que tenga en la Base de Datos serán las funcionalidades que este posea en el sistema; así como el nivel de acceso. Se verifica que el usuario sea interno en la UCI de serlo se procede a obtener sus datos identificativos de los servicios web, se verifica que no esté bloqueado, o que su provincia tampoco lo esté para entonces, filtrar los bloques al que dicho usuario tiene acceso, luego se procede a filtrar los viajes de los bloques antes permitidos.

## 2.6. Descripción de las nuevas clases u operaciones necesarias

### 2.6.1. Clases Controladoras (CC)

TABLA 1 : Descripción de la CC autenticar

Nombre: autenticarActions	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que muestra el formulario de autenticar si no estás logueado.
Nombre:	executeAutenticar
Descripción:	Función que verifica usuario y contraseña.
Nombre:	executeLogout
Descripción:	Función que destruye la sesión del que está logueado.
Nombre:	executeIntrusos
Descripción:	Función para visualizar intentos de entradas no permitidas.

TABLA 2 : Descripción de la CC configuración

Nombre: configuracionActions	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite cargar los datos de la configuración.
Nombre:	executeGuardar
Descripción:	Función que permite guardar los cambios realizados.

TABLA 3 : Descripción de la CC gest\_accesos

Nombre: gest_accesosActions
-----------------------------

Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite visualizar los usuarios bloqueados.
Nombre:	executeList
Descripción:	Función que busca y lista los usuarios bloqueados.
Nombre:	executeShow
Descripción:	Función que visualiza los detalles del usuario bloqueado.
Nombre:	executeCreate
Descripción:	Función que permite obtener y preparar los detalles del usuario bloqueado a modificar.
Nombre:	executeEdit
Descripción:	Función que permite visualizar un formulario con los detalles del usuario bloqueado a modificar.
Nombre:	executeUpdate
Descripción:	Función que permite modificar los datos del usuario bloqueado.
Nombre:	executeDelete
Descripción:	Función que desbloquea al usuario bloqueado.
Nombre:	executeBuscar
Descripción:	Función que visualiza formulario de búsqueda para usuarios a bloquear.
Nombre:	executeMostrar
Descripción:	Función que permite visualizar los detalles del usuario a bloquear.
Nombre:	esUsuarioBloqueado
Descripción:	Función que permite saber si es un usuario bloqueado o no.
Nombre:	desbloquearUsuario
Descripción:	Función que modifica los datos del usuario bloqueado y lo desbloquea.

TABLA 4 : Descripción de la CC gest\_bloqueo\_prov

Nombre: gest_bloqueo_provActions	
Tipo de clase: controladora	
Atributo	Tipo



Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite visualizar las provincias bloqueadas.
Nombre:	executeList
Descripción:	Función que busca y lista las provincias bloqueadas.
Nombre:	executeShow
Descripción:	Función que visualiza los detalles de la provincia bloqueada a modificar.
Nombre:	executeCreate
Descripción:	Función que permite obtener y preparar los detalles de la provincia bloqueada a modificar.
Nombre:	executeEdit
Descripción:	Función que permite visualizar un formulario con los detalles de la provincia bloqueada a modificar.
Nombre:	executeUpdate
Descripción:	Función que permite modificar los datos de la provincia bloqueada.
Nombre:	executeDelete
Descripción:	Función que desbloquea al usuario bloqueado.
Nombre:	executeBuscar
Descripción:	Función que visualiza un formulario de búsqueda para provincias a bloquear.
Nombre:	executeMostrar
Descripción:	Función que permite visualizar los detalles de la provincia a bloquear.
Nombre:	esProvinciaBloqueada
Descripción:	Función que permite saber si es una provincia bloqueada o no.
Nombre:	desbloquearProvincia
Descripción:	Función que modifica los datos de la provincia bloqueada y la desbloquea.

TABLA 5 : Descripción de la CC gest\_roles

Nombre: gest_rolesActions	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	

Nombre:	executeIndex
Descripción:	Función que permite obtener todos los roles existentes en el sistema.
Nombre:	executeUpdate
Descripción:	Función que determina la acción a realizar con los roles.
Nombre:	executeDatos
Descripción:	Función que permite obtener todos los roles existentes en el sistema
Nombre:	executeNuevo
Descripción:	Función que permite crear un nuevo rol.
Nombre:	executeEliminar
Descripción:	Función que elimina un rol.
Nombre:	executeShowfuncionalidades
Descripción:	Función que permite obtener las funcionalidades de un rol

TABLA 6 : Descripción de la CC gest\_usuarios

Nombre: gest_usuariosActions	
Tipo de clase controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite mostrar los usuarios del sistema.
Nombre:	executeNuevousuario
Descripción:	Función que permite buscar los datos del nuevo usuario.
Nombre:	executeAddroll
Descripción:	Función que adiciona el rol al usuario seleccionado.
Nombre:	executeCreate
Descripción:	Función que permite almacenar los datos del nuevo usuario.
Nombre:	executeMostrarusuario
Descripción:	Función que permite visualizar los datos de un usuario del sistema.
Nombre:	executeDelete
Descripción:	Función que permite eliminar un usuario del sistema.
Nombre:	executeMostrarrolles

Descripción:	Función que permite mostrar los roles de un usuario del sistema.
--------------	--

TABLA 7 : Descripción de la CC gest\_bloques

Nombre: gest_bloquesActions	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite visualizar y obtener los bloques habilitados.
Nombre:	executeCrear
Descripción:	Función que permite obtener y visualizar el bloque a editar o adicionar uno nuevo.
Nombre:	executeAdicionar
Descripción:	Función que permite adicionar un bloque nuevo.
Nombre:	executeEliminar
Descripción:	Función que permite cambiar el estado de un bloque.
Nombre:	executeReglas
Descripción:	Función que permite visualizar las reglas que tiene un bloque.
Nombre:	executeNuevaregla
Descripción:	Función que permite preparar los datos de la regla a adicionar.
Nombre:	executeCriterios
Descripción:	Función que permite obtener los valores posibles del criterio seleccionado.
Nombre:	executeInsertarregla
Descripción:	Función que permite insertar una regla a un bloque existente.
Nombre:	handleErrorAdicionar
Descripción:	Función que permite controlar errores al adicionar.

TABLA 8 : Descripción de la CC gest\_paradas

Nombre: gest_paradasActions	
Tipo de clase: controladora	
Atributo	Tipo

Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite visualizar las paradas.
Nombre:	executeList
Descripción:	Función que busca y lista las paradas.
Nombre:	executeShow
Descripción:	Función que visualiza los detalles de la parada a modificar.
Nombre:	executeCreate
Descripción:	Función que permite obtener y preparar los detalles de la parada a modificar.
Nombre:	executeEdit
Descripción:	Función que permite visualizar un formulario con los detalles de la parada a modificar.
Nombre:	executeUpdate
Descripción:	Función que permite modificar los datos de la parada.
Nombre:	executeDelete
Descripción:	Función que elimina la parada.

TABLA 9 : Descripción de la CC gest\_punto

Nombre: gest_puntoActions	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite visualizar los puntos.
Nombre:	executeList
Descripción:	Función que busca y lista los puntos.
Nombre:	executeShow
Descripción:	Función que visualiza los detalles del punto a modificar.
Nombre:	executeCreate
Descripción:	Función que permite obtener y preparar los detalles del punto a modificar.
Nombre:	executeEdit
Descripción:	Función que permite visualizar un formulario con los detalles del punto a modificar.

Nombre:	executeUpdate
Descripción:	Función que permite modificar los datos del punto.
Nombre:	executeDelete
Descripción:	Función que elimina el punto.

TABLA 10 : Descripción de la CC gest\_punto\_salida\_uci

Nombre: gest_punto_salida_uciActions	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite visualizar los puntos de salida en la UCI.
Nombre:	executeList
Descripción:	Función que busca y lista los puntos de salida en la UCI.
Nombre:	executeShow
Descripción:	Función que visualiza los detalles del punto de salida en la UCI a modificar.
Nombre:	executeCreate
Descripción:	Función que permite obtener y preparar los detalles del punto de salida en la UCI a modificar.
Nombre:	executeInsertar
Descripción:	Función que permite adicionar un nuevo punto de salida en la UCI.
Nombre:	executeEdit
Descripción:	Función que permite visualizar un formulario con los detalles del punto de salida en la UCI a modificar.
Nombre:	executeUpdate
Descripción:	Función que permite modificar los datos del punto de salida en la UCI.
Nombre:	executeDelete
Descripción:	Función que elimina el punto de salida en la UCI.

TABLA 11 : Descripción de la CC gest\_semestres

Nombre: gest_semestresActions
-------------------------------

Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite mostrar los semestres existentes.
Nombre:	executeGuardar
Descripción:	Función que permite guardar un semestre con sus meses.
Nombre:	executeCrear
Descripción:	Función que permite obtener los meses de un semestre.
Nombre:	executeMeses
Descripción:	Función que permite obtener los meses de un semestre por una petición ajax.
Nombre:	executeRedefinir
Descripción:	Función que elimina los semestres.

TABLA 12 : Descripción de la CC gest\_rutas

Nombre: gest_rutasActions	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite visualizar las rutas.
Nombre:	executeList
Descripción:	Función que busca y lista las rutas.
Nombre:	executeShow
Descripción:	Función que visualiza los detalles de la ruta a modificar.
Nombre:	executeCreate
Descripción:	Función que permite obtener y preparar los detalles de la ruta a modificar.
Nombre:	executeEdit
Descripción:	Función que permite visualizar un formulario con los detalles de la ruta a modificar.

Nombre:	executeUpdate
Descripción:	Función que permite modificar los datos de la ruta.
Nombre:	executeDelete
Descripción:	Función que elimina la ruta.
Nombre:	executeShowpuntos
Descripción:	Función que permite visualizar los puntos de una ruta.
Nombre:	executeShowmunicipios
Descripción:	Función que permite obtener los municipios de una provincia.

TABLA 13 : Descripción de la CC gest\_viajes

Nombre: gest_viajesActions	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite visualizar los viajes creados.
Nombre:	executeMostrar
Descripción:	Función que permite obtener los datos de los viajes y los muestra.
Nombre:	executeCrear
Descripción:	Función que visualiza el formulario para crear un nuevo viaje.
Nombre:	executeCorrespondiente
Descripción:	Función que permite hacer corresponder el viaje de ida con su viaje de regreso.
Nombre:	executeDatos
Descripción:	Función que permite obtener los meses para la aplicación de profesores, y los bloques para transportación nacional.
Nombre:	executeEditar
Descripción:	Función que permite modificar un viaje.
Nombre:	executeNuevaregla
Descripción:	Función que visualiza el formulario para adicionar una regla al viaje.
Nombre:	executeReglas
Descripción:	Función que permite mostrar las reglas de un viaje y eliminarlas.

Nombre:	executeInsertarregla
Descripción:	Función que permite agregarle nuevas reglas al viaje.
Nombre:	executeEliminar
Descripción:	Función que permite deshabilitar un viaje.
Nombre:	executeDeshabilitarVarios
Descripción:	Función que permite deshabilitar varios viajes.
Nombre:	handleErrorGuardar
Descripción:	Función que avisa si hay error al guardar un viaje.
Nombre:	handleErrorEditar
Descripción:	Función que avisa si hay error al editar un viaje.

TABLA 14 : Descripción de la CC gest\_boletines

Nombre: gest_boletinesActions	
Tipo de clase: controladora	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite visualizar el formulario con las áreas a realizar el boletín.
Nombre:	executeShowgrupos
Descripción:	Función para visualizar los grupos de la facultad seleccionada.
Nombre:	executeShowestudiantes
Descripción:	Función que permite visualizar los estudiantes del grupo seleccionado.
Nombre:	executeBoletines
Descripción:	Función que genera los boletines según el criterio seleccionado.
Nombre:	executeBoletines_estudiantes
Descripción:	Función que permite generar los boletines de todos los estudiantes que hayan reservado.

TABLA 15 : Descripción de la CC gest\_transportes

Nombre: gest_transportesActions	
Tipo de clase: controladora	
Atributo	Tipo



Para cada responsabilidad:	
Nombre:	executeIndex
Descripción:	Función que permite visualizar los transportes.
Nombre:	executeList
Descripción:	Función que busca y lista los transportes.
Nombre:	executeShow
Descripción:	Función que visualiza los detalles del transporte a modificar.
Nombre:	executeCreate
Descripción:	Función que permite obtener y preparar los detalles del transporte a modificar.
Nombre:	executeUpdate
Descripción:	Función que permite adicionar un nuevo transporte.
Nombre:	executeEdit
Descripción:	Función que permite visualizar un formulario con los detalles del transporte a modificar.
Nombre:	executeModificarTransporte
Descripción:	Función que permite modificar los datos del transporte.
Nombre:	executeDelete
Descripción:	Función que elimina el transporte.
Nombre:	executeCoches
Descripción:	Función que permite visualizar o adicionar los coches a un transporte de tipo tren.
Nombre:	executeShowmunicipios
Descripción:	Función que permite obtener los municipios de una provincia.
Nombre:	executeAsientos
Descripción:	Función que permite la gestión de los asientos en los transportes.
Nombre:	executeParadas
Descripción:	Función que permite la gestión de las paradas de los transportes.
Nombre:	handleErrorUpdate
Descripción:	Función que avisa si hay error al guardar un transporte.
Nombre:	handleErrorModificarTransporte
Descripción:	Función que avisa si hay error al modificar un transporte.

## 2.6.2. Clases Interfaz (CI)

TABLA 16 : Descripción de la CI configurar

Nombre: indexSuccess	
Tipo de clase interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	image_tag('ruta', array)
Descripción:	Función que permite visualizar una imagen.
Nombre:	form_tag('url')
Descripción:	Función que permite crear un formulario.
Nombre:	input_tag('title')
Descripción:	Función que visualiza un campo de texto.
Nombre:	input_tag('correo')
Descripción:	Función que visualiza un campo de texto.
Nombre:	select_tag ('confirmar')
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	checkbox_tag('reservar')
Descripción:	Función que visualiza un checkbox
Nombre:	checkbox_tag('tn')
Descripción:	Función que visualiza un checkbox
Nombre:	checkbox_tag('tnp')
Descripción:	Función que visualiza un checkbox
Nombre:	checkbox_tag('te')
Descripción:	Función que visualiza un checkbox
Nombre:	submit_tag('Guardar cambios')
Descripción:	Función que visualiza un botón que envía el formulario.

TABLA 17 : Descripción de la CI autenticar usuario

Nombre: indexSuccess
Tipo de clase: interfaz

Atributo	Tipo
Para cada responsabilidad:	
Nombre:	form_tag('url')
Descripción:	Función que permite crear un formulario.
Nombre:	input_tag('usuario')
Descripción:	Función que visualiza un campo de texto.
Nombre:	input_tag('pass')
Descripción:	Función que visualiza un campo de texto.
Nombre:	submit_tag('Entrar',array())
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	Include_partial('nombre')
Descripción:	Función que permite visualizar el contenido xhtml que contenga donde es llamada.

TABLA 18 : Descripción de la CI buscar del caso de uso gestionar accesos

Nombre: buscarSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_remote_tag(array("update"=>"mostrar",'url'=>'gest_accesos/mostrar'))
Descripción:	Función que permite crear un formulario de tipo ajax.
Nombre:	input_tag('parámetro')
Descripción:	Función que visualiza un campo de texto.
Nombre:	submit_tag('Buscar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('Cancelar', ruta)
Descripción:	Función que permite visualizar un vínculo.

TABLA 19 : Descripción de la CI mostrar del caso de uso gestionar accesos

Nombre: mostrarSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	image_tag('ruta', array)
Descripción:	Función que permite visualizar una imagen.
Nombre:	link_to('Siguiete', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	button_to('Modificar', 'ruta')
Descripción:	Función que permite visualizar un botón.
Nombre:	button_to('Listado', 'ruta')
Descripción:	Función que permite visualizar un botón.

TABLA 20 : Descripción de la CI listar del caso de uso gestionar accesos

Nombre: listSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to('[Bloqueo de provincias]', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('nombre', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Bloquear', 'ruta')
Descripción:	Función que permite visualizar un vínculo.

TABLA 21 : Descripción de la CI editar del caso de uso gestionar accesos

Nombre: editSuccess	
Tipo de clase: interfaz	
Atributo	Tipo

Para cada responsabilidad:	
Nombre:	use_helper('Object', 'Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_accesos/update')
Descripción:	Función que permite crear un formulario.
Nombre:	object_input_hidden_tag(\$tb_dusuario_bloqueado,'getIdBloqueoUsuario')
Descripción:	Función que permite crear un campo oculto que llama a un método de un objeto.
Nombre:	input_hidden_tag('id_expediente', \$id_expediente)
Descripción:	Función que permite crear un campo oculto.
Nombre:	object_input_date_tag(\$tb_dusuario_bloqueado, 'getFechaInicio')
Descripción:	Función que permite crear un campo fecha que llama a un método de un objeto.
Nombre:	object_input_date_tag(\$tb_dusuario_bloqueado, 'getFechaFin')
Descripción:	Función que permite crear un campo fecha que llama a un método de un objeto.
Nombre:	object_textarea_tag(\$tb_dusuario_bloqueado, 'getObservacion')
Descripción:	Función que permite crear un texarea que llama a un método de un objeto.
Nombre:	select_tag('id_aplicacion','array')
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('Aceptar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('Desbloquear','ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Cancelar','ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Atrás', 'ruta')
Descripción:	Función que permite visualizar un vínculo.

TABLA 22 : Descripción de la CI mostrar detalle del caso de uso gestionar accesos

Nombre: showSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	

Nombre:	button_to('Modificar', 'ruta')
Descripción:	Función que permite visualizar un botón.
Nombre:	button_to('Listar', 'ruta')
Descripción:	Función que permite visualizar un botón.

TABLA 23 : Descripción de la CI editar del caso de uso bloquear provincia

Nombre: editSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Objeto')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_bloqueo_prov/update')
Descripción:	Función que permite crear un formulario.
Nombre:	object_input_hidden_tag(\$tb_dprovincia_bloqueada, 'getIdBloqueo')
Descripción:	Función que permite crear un campo oculto que llama a un método de un objeto.
Nombre:	object_input_date_tag(\$tb_dprovincia_bloqueada, getFechaInicio)
Descripción:	Función que permite crear un campo fecha que llama a un método de un objeto.
Nombre:	object_input_date_tag(\$tb_dprovincia_bloqueada, 'getFechaFin')
Descripción:	Función que permite crear un campo fecha que llama a un método de un objeto.
Nombre:	object_textarea_tag(\$tb_dprovincia_bloqueada, 'getObservacion')
Descripción:	Función que permite crear un texarea que llama a un método de un objeto.
Nombre:	select_tag('id_provincia', 'array')
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('id_tipo_viaje', 'array')
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('Aceptar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('Desbloquear', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Cancelar', 'ruta')

Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Atras','ruta')
Descripción:	Función que permite visualizar un vínculo.

TABLA 24 : Descripción de la CI listar del caso de uso bloquear provincia

Nombre: listSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to('[Bloqueo de Usuarios]','ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('provincia', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Bloquear', 'ruta')
Descripción:	Función que permite visualizar un vínculo.

TABLA 25 : Descripción de la CI mostrar del caso de uso bloquear provincia

Nombre: showSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	button_to('Modificar', 'ruta')
Descripción:	Función que permite visualizar un botón.
Nombre:	button_to('Listar', 'ruta')
Descripción:	Función que permite visualizar un botón.

TABLA 26 : Descripción de la CI principal del caso de uso gestionar bloques

Nombre: indexSuccess	
Tipo de clase: interfaz	
Atributo	Tipo

Para cada responsabilidad:	
Nombre:	image_tag('/images/fondos/lassists.png',array)
Descripción:	Función que permite visualizar una imagen.
Nombre:	link_to ('nuevo bloque', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	image_tag('/images/acciones/add.png',array)
Descripción:	Función que permite visualizar una imagen.
Nombre:	format_datetime(\$bloque->getFechaCreacion(),'P', 'es')
Descripción:	Función que permite dar formato a una fecha.
Nombre:	link_to ('Editar', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
	link_to ('Eliminar', 'ruta')
	Función que permite visualizar un vínculo.
Nombre:	image_tag('/images/indicadores/indicator_snake.gif')
Descripción:	Función que permite visualizar una imagen.
Nombre:	javascript_tag()
Descripción:	Función que permite usar código javascript donde se llame.
Nombre:	remote_function()
Descripción:	Función que permite llamar una función mediante ajax.
Nombre:	link_to ('Adicionar Regla','ruta')
Descripción:	Función que permite visualizar un vínculo.

TABLA 27 : Descripción de la CI nueva regla del caso de uso gestionar bloques

Nombre: nuevareglaSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_bloques/insertarregla')



Descripción:	Función que permite visualizar un formulario
Nombre:	input_hidden_tag('id_bloque', \$id_bloque)
Descripción:	Función que permite crear un campo oculto.
Nombre:	observe_field('concepto', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	select_tag('concepto', 'array')
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('aux', 'array')
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('orden', 'array')
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('Adicionar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('Cancelar', 'ruta')
Descripción:	Función que permite visualizar un vínculo.

TABLA 28 : Descripción de la CI reglas del caso de uso gestionar bloques

Nombre: reglasSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to('Eliminar', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	image_tag('/images/acciones/eliminar.gif', array)
Descripción:	Función que permite visualizar una imagen.

TABLA 29 : Descripción de la CI criterios del caso de uso gestionar bloques

Nombre: criteriosSuccess	
Tipo de clase: interfaz	
Atributo	Tipo

Para cada responsabilidad:	
Nombre:	select_tag('criterio', 'array')
Descripción:	Función que permite visualizar un menú desplegable.

TABLA 30 : Descripción de la CI crear del caso de uso gestionar bloques

Nombre: crearSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper(' Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_bloques/adicionar')
Descripción:	Función que permite crear un formulario.
Nombre:	input_hidden_tag('id_bloque', \$id_bloque)
Descripción:	Función que permite crear un campo oculto.
Nombre:	input_tag('nombre')
Descripción:	Función que visualiza un campo de texto.
Nombre:	select_tag('Tipo', 'array')
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('Guardar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('Finalizar', 'ruta')
Descripción:	Función que permite visualizar un vínculo.

TABLA 31 : Descripción de la CI editar del caso de uso gestionar paradas

Nombre: editSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	

Nombre:	use_helper('Objeto')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_paradas/update')
Descripción:	Función que permite crear un formulario.
Nombre:	object_input_hidden_tag(\$tb_raplicacion_parada_intermedia, 'getIdAplicacionParada')
Descripción:	Función que permite crear un campo oculto que llama a un método de un objeto.
Nombre:	select_tag('id_aplicacion','array')
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('id_parada[]','array')
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('Guardar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('Eliminar', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Cancelar','ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Cancelar','ruta1')
Descripción:	Función que permite visualizar un vínculo.

TABLA 32 : Descripción de la CI listar del caso de uso gestionar paradas

Nombre: listSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	form_tag('gest_paradas/list')
Descripción:	Función que permite crear un formulario.
Nombre:	select_tag('id_aplicacion',options_for_select(\$apps,\$id_aplicacion))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('Filtrar')
Descripción:	Función que visualiza un botón que envía el formulario.

Nombre:	image_tag('/images/acciones/add.png')
Descripción:	Función que permite visualizar una imagen.
Nombre:	link_to(['Nueva'], 'Ruta')
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to(image_tag('/images/acciones/first.png'),'ruta')
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to(image_tag('/images/acciones/previous.png'),'ruta')
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to_unless(\$page == \$pager->getPage(), \$page,'ruta')
Descripción:	Función permite visualizar un vínculo si se cumple la condición pasada como parámetro.
Nombre:	link_to(image_tag('/images/acciones/next.png'),'ruta')
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to(image_tag('/images/acciones/last.png'),'ruta')
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	form_tag('gest_paradas/delete')
Descripción:	Función que permite crear un formulario.
Nombre:	checkbox_tag('batch[]', 'value')
Descripción:	Función que visualiza un checkbox
Nombre:	link_to(image_tag('/images/acciones/eliminar.gif'),'ruta')
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	submit_tag('Eliminar seleccionadas')
Descripción:	Función que visualiza un botón que envía el formulario.

TABLA 33 : Descripción de la CI mostrar del caso de uso gestionar paradas

Nombre: showSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to('Edit', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('List', 'ruta')

Descripción:	Función que permite visualizar un vínculo.
--------------	--

TABLA 34 : Descripción de la CI editar del caso de uso gestionar punto

Nombre: editSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Objeto')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_punto/update')
Descripción:	Función que permite crear un formulario.
Nombre:	object_input_hidden_tag(\$tb_dpunto, 'getIdPunto')
Descripción:	Función que permite crear un campo oculto que llama a un método de un objeto.
Nombre:	object_input_tag(\$tb_dpunto, 'getPunto')
Descripción:	Función que permite crear un campo oculto que llama a un método de un objeto.
Nombre:	submit_tag('Guardar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	button_to('Eliminar', ruta)
Descripción:	Función que permite visualizar un botón.
Nombre:	button_to('Cancelar', ruta)
Descripción:	Función que permite visualizar un botón.

TABLA 35 : Descripción de la CI mostrar del caso de uso gestionar punto

Nombre: showSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to('Edit', 'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('List', 'ruta')

Descripción:	Función que permite visualizar un vínculo.
--------------	--

TABLA 36 : Descripción de la CI listar del caso de uso gestionar punto

Nombre: listSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to(\$tb_dpunto->getPunto(),'ruta')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	button_to ('Crear', 'gest_punto/create')
Descripción:	Función que permite visualizar un botón.

TABLA 37 : Descripción de la CI principal del caso de uso generar boletines

Nombre: indexSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_boletines/boletines_estudiantes')
Descripción:	Función que permite crear un formulario.
Nombre:	image_tag('astro.jpg', 'width=60')
Descripción:	Función que permite visualizar una imagen.
Nombre:	select_tag('id_facultad',\$facultades)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('id_facultad', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	select_tag('id_grupo', array)
Descripción:	Función que permite visualizar un menú desplegable.

Nombre:	image_tag('/images/indicadores/indicator_snake.gif')
Descripción:	Función que permite visualizar una imagen.
Nombre:	select_tag('id_estudiante', array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('Generar')
Descripción:	Función que visualiza un botón que envía el formulario.

TABLA 38 : Descripción de la CI mostrar estudiantes del caso de uso generar boletines

Nombre: showestudiantesSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	select_tag('id_estudiante', \$estudiantes)
Descripción:	Función que permite visualizar un menú desplegable.

TABLA 39 : Descripción de la CI mostrar grupos

Nombre: showgruposSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	select_tag('id_grupo', \$grupos)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('id_grupo', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	visual_effect('appear', 'divindicador')
Descripción:	Función que permite visualizar un efecto ajax.
Nombre:	visual_effect('fade', 'divindicador')

Descripción:	Función que permite visualizar un efecto ajax.
--------------	--

TABLA 40 : Descripción de la CI principal del caso de uso gestionar roles

Nombre: indexSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_roles/update')
Descripción:	Función que permite crear un formulario.
Nombre:	include_partial('info')
Descripción:	Función que permite visualizar el contenido xhtml que contenga donde es llamada.
Nombre:	select_tag('id_rol',options_for_select(\$roles),array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('id_rol', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	link_to('Nuevo', 'gest_roles/nuevo')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	submit_tag('Modificar',array('name'=>'modificar'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	submit_tag('Eliminar',array('name'=>'eliminar'))
Descripción:	Función que visualiza un botón que envía el formulario.

TABLA 41 : Descripción de la CI nuevo del caso de uso gestionar roles

Nombre: nuevoSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	



Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	include_partial('info')
Descripción:	Función que permite visualizar el contenido xhtml que contenga donde es llamada.
Nombre:	form_tag('gest_rol/nuevo')
Descripción:	Función que permite crear un formulario.
Nombre:	input_hidden_tag('id_rol', \$id_rol)
Descripción:	Función que permite crear un campo oculto.
Nombre:	input_tag('nombre_rol', \$nombre)
Descripción:	Función que visualiza un campo de texto.
Nombre:	select_tag('id_aplicacion', options_for_select(\$aplicaciones), array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('id_aplicacion', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	select_tag('id_funcionalidad', \$funcionalidades, array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('>>', array('name' => 'add'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	submit_tag('<<', array('name' => 'del'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	select_tag('id_funcion_rol', \$funcion_rol, array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	link_to('Finalizar', 'gest_rol/index')
Descripción:	Función que permite visualizar un vínculo.

TABLA 42 : Descripción de la CI mostrar funcionalidades del caso de uso gestionar roles

Nombre: showfuncionalidadesSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	

Nombre:	select_tag('id_funcionalidad', \$funcionalidades, array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('>>',array('name'=>'add'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	submit_tag('<<',array('name'=>'del'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	select_tag('id_funcion_rol', \$funcion_rol, array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	link_to('Finalizar', 'gest_rols/index')
Descripción:	Función que permite visualizar un vínculo.

TABLA 43 : Descripción de la CI principal del caso de uso gestionar usuarios

Nombre: indexSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to(image_tag('/images/acciones/kuser.png', array)
Descripción:	Función que permite visualizar un vínculo.
Nombre:	image_tag(\$user['foto'],array('size'=>'70x70'))
Descripción:	Función que permite visualizar una imagen.
Nombre:	link_to('ver','gest_usuarios/mostrarusuario?id_expediente='.\$user['solapin'])
Descripción:	Función que permite visualizar un vínculo.

TABLA 44 : Descripción de la CI mostrar usuario del caso de uso gestionar usuarios

Nombre: mostrarusuariosSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	image_tag(\$usuario['foto'],array('size'=>'70x70'))
Descripción:	Función que permite visualizar una imagen.

Nombre:	link_to('Cambiar Rol', 'gest_usuarios/addroll')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Eliminar', 'gest_usuarios/delete')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Cancelar', 'gest_usuarios/index')
Descripción:	Función que permite visualizar un vínculo.

TABLA 45 : Descripción de la CI nuevo usuario del caso de uso gestionar usuarios

Nombre: nuevousuarioSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_usuarios/nuevousuario')
Descripción:	Función que permite crear un formulario.
Nombre:	submit_tag('Buscar', array('name'=>'buscar'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	input_tag('parámetro')
Descripción:	Función que visualiza un campo de texto.
Nombre:	link_to('Listado', 'gest_usuarios/index')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	image_tag(\$usuario['foto'],array('size'=>'70x70'))
Descripción:	Función que permite visualizar una imagen.
Nombre:	input_hidden_tag('id_expediente', \$usuario['solapin'])
Descripción:	Función que permite crear un campo oculto.
Nombre:	submit_tag('Seleccionar', array('name'=>'seleccionar'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('Cancelar', 'gest_usuarios/index')
Descripción:	Función que permite visualizar un vínculo.

TABLA 46 : Descripción de la CI adicionar rol del caso de uso gestionar usuarios

Nombre: addrolSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_usuarios/create')
Descripción:	Función que permite crear un formulario.
Nombre:	input_hidden_tag('id_usuario', \$id_propiedad_usuario)
Descripción:	Función que permite crear un campo oculto.
Nombre:	image_tag(\$usuario['foto'],array('size'=>'70x70'))
Descripción:	Función que permite visualizar una imagen.
Nombre:	input_hidden_tag('id_expediente', \$usuario['solapin'])
Descripción:	Función que permite crear un campo oculto.
Nombre:	select_tag('id_rol', \$roles, array('id'=>'id_rol'))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('id_rol', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	submit_tag('Guardar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('Cancelar', 'gest_usuarios/index')
Descripción:	Función que permite visualizar un vínculo.

TABLA 47 : Descripción de la CI editar del caso de uso gestionar punto de salida UCI

Nombre: editSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	

Nombre:	use_helper('Objeto')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_punto_salida_uci/update')
Descripción:	Función que permite crear un formulario.
Nombre:	object_input_hidden_tag(\$tb_dpunto_salida_uci, 'getIdPsuci')
Descripción:	Función que permite crear un campo oculto que llama a un método de un objeto.
Nombre:	object_input_tag(\$tb_dpunto_salida_uci, 'getPsuci', array)
Descripción:	Función que permite crear un campo de texto que llama a un método de un objeto.
Nombre:	link_to(image_tag('/images/acciones/eliminar.gif', array)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	submit_tag('Modificar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('Atr&aacute;s', 'gest_punto_salida_uci/list')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Atr&aacute;s', 'gest_punto_salida_uci/list')
Descripción:	Función que permite visualizar un vínculo.

TABLA 48 : Descripción de la CI mostrar del caso de uso gestionar punto de Salida UCI

Nombre: showSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to('Modificar/Eliminar', 'gest_punto_salida_uci/edit')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Listado', 'gest_punto_salida_uci/list')
Descripción:	Función que permite visualizar un vínculo.

TABLA 49 : Descripción de la CI listar del caso de uso gestionar punto de salida UCI

Nombre: listSuccess	
Tipo de clase: interfaz	
Atributo	Tipo

Para cada responsabilidad:	
Nombre:	link_to(image_tag('/images/acciones/eliminar.gif',array))
Descripción:	Función que permite visualizar un vínculo en una imagen.
Nombre:	link_to(image_tag('/images/acciones/edit.png', array))
Descripción:	Función que permite visualizar un vínculo en una imagen.
Nombre:	button_to ('Nuevo Punto', 'gest_punto_salida_uci/create')
Descripción:	Función que permite visualizar un botón.

TABLA 50 : Descripción de la CI crear del caso de uso gestionar punto de salida UCI

Nombre: createSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	form_tag('gest_punto_salida_uci/insertar')
Descripción:	Función que permite crear un formulario.
Nombre:	input_tag('Psuci')
Descripción:	Función que visualiza un campo de texto.
Nombre:	submit_tag('Crear')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('Listado', 'gest_punto_salida_uci/list')
Descripción:	Función que permite visualizar un vínculo.

TABLA 51 : Descripción de la CI Principal del caso de uso gestionar semestres

Nombre: indexSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	image_tag('reservacion.png', array)
Descripción:	Función que permite visualizar una imagen.

Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_semestres/crear')
Descripción:	Función que permite crear un formulario.
Nombre:	image_tag('/images/indicadores/indicator_snake.gif')
Descripción:	Función que permite visualizar una imagen.
Nombre:	select_tag('id_semestre', \$semestres, array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('id_semestre', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	visual_effect('appear', 'indicador')
Descripción:	Función que permite visualizar un efecto ajax.
Nombre:	visual_effect('fade', 'indicador')
Descripción:	Función que permite visualizar un efecto ajax.
Nombre:	submit_tag('Definir')
Descripción:	Función que visualiza un botón que envía el formulario.

TABLA 52 : Descripción de la CI meses del caso de uso gestionar semestres

Nombre: mesesSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	select_tag('meses', \$meses, array)
Descripción:	Función que permite visualizar un menú desplegable.

TABLA 53 : Descripción de la CI crear del caso de uso gestionar semestres

Nombre: crearSuccess	
Tipo de clase: interfaz	
Atributo	Tipo

Para cada responsabilidad:	
Nombre:	form_tag('gest_semestres/guardar')
Descripción:	Función que permite crear un formulario.
Nombre:	input_hidden_tag('id_semestre', \$semestre['id_semestre'])
Descripción:	Función que permite crear un campo oculto.
Nombre:	select_tag('meses', \$meses, array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('>>', array('name'=>'add'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	submit_tag('<<', array('name'=>'del'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	select_tag('semestre_meses', \$semestre_meses, array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	link_to('Guardar', 'gest_semestres/index')
Descripción:	Función que permite visualizar un vínculo.

TABLA 54 : Descripción de la CI listar del caso de uso gestionar transporte

Nombre: listSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	image_tag('/images/yutong.jpg', array('width'=>100))
Descripción:	Función que permite visualizar una imagen.
Nombre:	link_to ('[Crear transporte]', 'gest_transportes/create')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('nombre transporte', 'gest_transportes/show')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to(image_tag('/images/acciones/edit.png', array)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to(image_tag('/images/acciones/first.png'), ruta)
Descripción:	Función que Visualiza un vínculo en una imagen.



Nombre:	link_to(image_tag('/images/acciones/previous.png'), ruta)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to_unless(\$page == \$pager->getPage(), \$page, ruta)
Descripción:	Función permite visualizar un vínculo si se cumple la condición pasada como parámetro.
Nombre:	link_to(image_tag('/images/acciones/next.png'), ruta)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to(image_tag('/images/acciones/last.png'), ruta)
Descripción:	Función que Visualiza un vínculo en una imagen.

TABLA 55 : Descripción de la CI modificar transporte del caso de uso gestionar transporte.

Nombre: modificartransporteSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Validation')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_transportes/modificarTransporte')
Descripción:	Función que permite crear un formulario.
Nombre:	input_hidden_tag('id_transporte', \$tb_dtransporte)
Descripción:	Función que permite crear un campo oculto.
Nombre:	input_hidden_tag('id_viaje', \$id_viaje)
Descripción:	Función que permite crear un campo oculto.
Nombre:	input_tag('nombre_transporte', \$tb_dtransporte)
Descripción:	Función que visualiza un campo de texto.
Nombre:	input_date_tag('fecha_hora_s', \$tb_dtransporte)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	input_date_tag('fecha_hora_ll', \$tb_dtransporte)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	input_tag('punto_salida', \$tb_dtransporte)
Descripción:	Función que visualiza un campo de texto.
Nombre:	input_tag('punto_llegada', \$tb_dtransporte)

Descripción:	Función que visualiza un campo de texto.
Nombre:	input_tag('profesor_responsable', \$tb_dtransporte)
Descripción:	Función que visualiza un campo de texto.
Nombre:	submit_tag('Modificar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	button_to('cancelar','gest_transportes/index?id_viaje='.\$id_viaje)
Descripción:	Función que permite visualizar un botón.

TABLA 56 : Descripción de la CI actualizar error del caso de uso gestionar transporte

Nombre: updateErrorSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Object')
Descripción:	Función que permite incluir clases a usar.
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	use_helper('Validation')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_transportes/update')
Descripción:	Función que permite crear un formulario.
Nombre:	object_input_hidden_tag(\$tb_dtransporte, 'getIdTransporte')
Descripción:	Función que permite crear un campo oculto que llama a un método de un objeto.
Nombre:	input_hidden_tag('id_viaje', \$id_viaje)
Descripción:	Función que permite crear un campo oculto.
Nombre:	form_error('capacidad', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_tag(\$tb_dtransporte, 'getCapacidad', array)
Descripción:	Función que permite crear un campo de texto que llama a un método de un objeto.
Nombre:	form_error('nombre_transporte', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.

Nombre:	object_input_tag(\$tb_dtransporte, 'getNombreTransporte')
Descripción:	Función que permite crear un campo de texto que llama a un método de un objeto.
Nombre:	form_error('fecha_hora_s', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	input_date_tag('fecha_hora_s', \$viaje, array)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	form_error('fecha_hora_s', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_date_tag(\$tb_dtransporte, 'getFechaHoraS', array)
Descripción:	Función que permite crear un campo fecha que llama a un método de un objeto.
Nombre:	form_error('fecha_hora_ll', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_date_tag(\$tb_dtransporte, 'getFechaHoraLI', array)
Descripción:	Función que permite crear un campo fecha que llama a un método de un objeto.
Nombre:	form_error('punto_salida', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_tag(\$tb_dtransporte, 'getPuntoSalida')
Descripción:	Función que permite crear un campo de texto que llama a un método de un objeto.
Nombre:	form_error('punto_llegada', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_tag(\$tb_dtransporte, 'getPuntoLlegada')
Descripción:	Función que permite crear un campo de texto que llama a un método de un objeto.
Nombre:	select_tag('provincia', options_for_select(\$provincias))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('provincia', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	form_error('municipio', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	select_tag('id_psuci', options_for_select(\$puntos_salidas))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('id_tipo_transporte', options_for_select(\$tipos_transportes))

Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('id_tipo_transporte', options_for_select(array))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	include_partial('asientos')
Descripción:	Función que permite visualizar el contenido xhtml que contenga donde es llamada.
Nombre:	link_to('Eliminar', 'gest_transportes/delete')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Cancelar', 'gest_transportes/show')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Cancelar', 'gest_transportes/list')
Descripción:	Función que permite visualizar un vínculo.

TABLA 57 : Descripción de la CI mostrar del caso de uso gestionar transporte

Nombre: showSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Javascript', 'Date')
Descripción:	Función que permite incluir clases a usar.
Nombre:	image_tag('/images/20080415105638-trenes-tren.jpg',array)
Descripción:	Función que permite visualizar una imagen.
Nombre:	image_tag('/images/20080415105638-trenes-tren.jpg',array)
Descripción:	Función que permite visualizar una imagen.
Nombre:	link_to('[Editar]', 'gest_transportes/modificarTransporte')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('[Listado]', 'gest_transportes/list')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('[Eliminar transporte]', 'gest_transportes/delete')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('[Coches]', 'gest_transportes/coches')
Descripción:	Función que permite visualizar un vínculo.

Nombre:	link_to('[Gestionar asientos para los coches] ',' ruta)
Descripción:	Función que permite visualizar un vínculo.
Nombre:	image_tag('/images/indicadores/indicator_snake.gif)
Descripción:	Función que permite visualizar una imagen.
Nombre:	format_datetime(\$tb_dtransporte->getFechaHoraS(),'f', 'es')
Descripción:	Función que permite dar formato a una fecha.
Nombre:	format_datetime(\$tb_dtransporte->getFechaHoraLI(),'f', 'es')
Descripción:	Función que permite dar formato a una fecha.
Nombre:	include_partial('municipio', array)
Descripción:	Función que permite visualizar el contenido xhtml que contenga donde es llamada.

TABLA 58 : Descripción de la CI paradas del caso de uso gestionar transporte

Nombre: paradasSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	link_to(image_tag('/images/acciones/eliminar.gif'),'ruta')
Descripción:	Función que Visualiza un vínculo en una imagen.

TABLA 59 : Descripción de la CI municipios del caso de uso gestionar transporte

Nombre: municipiosSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	select_tag('municipio',options_for_select(\$municipios,'SANDINO'))
Descripción:	Función que permite visualizar un menú desplegable.

TABLA 60 : Descripción de la CI editar del caso de uso gestionar transporte

Nombre: editSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Object')
Descripción:	Función que permite incluir clases a usar.
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	use_helper('Validation')
Descripción:	Función que permite incluir clases a usar.
Nombre:	use_helper('Date')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_transportes/update')
Descripción:	Función que permite crear un formulario.
Nombre:	object_input_hidden_tag(\$tb_dtransporte, 'getIdTransporte')
Descripción:	Función que permite crear un campo oculto que llama a un método de un objeto.
Nombre:	input_hidden_tag('id_viaje', \$id_viaje)
Descripción:	Función que permite crear un campo oculto.
Nombre:	form_error('capacidad', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_tag(\$tb_dtransporte, 'getCapacidad', array)
Descripción:	Función que permite crear un campo de texto que llama a un método de un objeto.
Nombre:	form_error('nombre_transporte', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_tag(\$tb_dtransporte, 'getNombreTransporte')
Descripción:	Función que permite crear un campo de texto que llama a un método de un objeto.
Nombre:	form_error('fecha_hora_s', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	input_date_tag('fecha_hora_s', \$viaje, array)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	form_error('fecha_hora_s', array)

Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_date_tag(\$tb_dtransporte, 'getFechaHoraS', array)
Descripción:	Función que permite crear un campo fecha que llama a un método de un objeto.
Nombre:	form_error('fecha_hora_ll', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_date_tag(\$tb_dtransporte, 'getFechaHoraLI', array)
Descripción:	Función que permite crear un campo fecha que llama a un método de un objeto.
Nombre:	form_error('punto_salida', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_tag(\$tb_dtransporte, 'getPuntoSalida')
Descripción:	Función que permite crear un campo de texto que llama a un método de un objeto.
Nombre:	form_error('punto_llegada', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	object_input_tag(\$tb_dtransporte, 'getPuntoLlegada')
Descripción:	Función que permite crear un campo de texto que llama a un método de un objeto.
Nombre:	select_tag('provincia', options_for_select(\$provincias))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('provincia', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	form_error('municipio', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	select_tag('id_psuci', options_for_select(\$puntos_salidas))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('id_tipo_transporte', options_for_select(\$tipos_transportes))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('id_tipo_transporte', options_for_select(array))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	include_partial('asientos')
Descripción:	Función que permite visualizar el contenido xhtml que contenga donde es llamada.
Nombre:	link_to('Eliminar', 'gest_transportes/delete')
Descripción:	Función que permite visualizar un vínculo.

Nombre:	button_to('Cancelar', 'gest_transportes/show)
Descripción:	Función que permite visualizar un botón.
Nombre:	button_to('Cancelar', 'gest_transportes/list)
Descripción:	Función que permite visualizar un botón.

TABLA 61 : Descripción de la CI adicionar asientos del caso de uso gestionar transporte

Nombre: asientosSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to('[Mostrar transporte]', 'gest_transportes/show)
Descripción:	Función que permite visualizar un vínculo.
Nombre:	form_tag('gest_transportes/asientos', array)
Descripción:	Función que permite crear un formulario.
Nombre:	input_hidden_tag('id_transporte', \$id_transporte)
Descripción:	Función que permite crear un campo oculto.
Nombre:	input_hidden_tag('id_viaje', \$id_viaje)
Descripción:	Función que permite crear un campo oculto.
Nombre:	input_hidden_tag('id_coche', \$coche->getIdCoche())
Descripción:	Función que permite crear un campo oculto.
Nombre:	include_partial('asientos', array)
Descripción:	Función que permite visualizar el contenido xhtml que contenga donde es llamada.
Nombre:	submit_tag('+')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('[Mostrar transporte]', 'gest_transportes/show)
Descripción:	Función que permite visualizar un vínculo.

TABLA 62 : Descripción de la CI detalles asientos del caso de uso gestionar transporte

Nombre: _asientos	
Tipo de clase interfaz	
Atributo	Tipo



Para cada responsabilidad:	
Nombre:	use_helper('Validation')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_error('asiento1',array
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	input_tag('asiento1',\$rango1,array)
Descripción:	Función que visualiza un campo de texto.
Nombre:	input_tag('asiento1','1',array
Descripción:	Función que visualiza un campo de texto.
Nombre:	form_error('asiento2',array
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	input_tag('asiento2',\$rango2,array)
Descripción:	Función que visualiza un campo de texto.
Nombre:	input_tag('asiento2','40',array)
Descripción:	Función que visualiza un campo de texto.

TABLA 63 : Descripción de la CI adicionar coches del caso de uso gestionar transporte

Nombre: cochesSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to('[Eliminar]','gest_transportes/coches)
Descripción:	Función que permite visualizar un vínculo.
Nombre:	form_tag('gest_transportes/coches')
Descripción:	Función que permite crear un formulario.
Nombre:	input_hidden_tag('id_transporte', \$id_transporte)
Descripción:	Función que permite crear un campo oculto.
Nombre:	input_hidden_tag('id_viaje', \$id_viaje)
Descripción:	Función que permite crear un campo oculto.
Nombre:	select_tag('numero_coche', options_for_select(array)

Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	input_tag('capacidad_coche','80',array
Descripción:	Función que visualiza un campo de texto.
Nombre:	input_tag('responsable_coche')
Descripción:	Función que visualiza un campo de texto.
Nombre:	submit_tag('Guardar/Adicionar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to('[Mostrar transporte]', 'gest_transportes/show)
Descripción:	Función que permite visualizar un vínculo.

TABLA 64 : Descripción de la CI nueva reglas del caso de uso gestionar viajes

Nombre: nuevaReglaSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper(' Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_viajes/insertarregla')
Descripción:	Función que permite crear un formulario.
Nombre:	input_hidden_tag('id_viaje', \$id_viaje)
Descripción:	Función que permite crear un campo oculto.
Nombre:	select_tag('concepto', options_for_select(\$conceptos_a_mostrar),array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('concepto', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	select_tag('aux', options_for_select(array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('orden', options_for_select(array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	link_to('[CANCELAR]', 'gest_viajes/index')

Descripción:	Función que permite visualizar un vínculo.
--------------	--

TABLA 65 : Descripción de la CI reglas del caso de uso gestionar viajes

Nombre: reglasSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to(image_tag('/images/acciones/eliminar.gif',array)
Descripción:	Función que Visualiza un vinculo en una imagen.

TABLA 66 : Descripción de la CI crear viaje del caso de uso gestionar viajes

Nombre: crearSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Validation')
Descripción:	Función que permite incluir clases a usar.
Nombre:	link_to(['Listado de viajes'],'gest_viajes/index')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_viajes/guardar')
Descripción:	Función que permite crear un formulario.
Nombre:	select_tag('tipo_viaje', array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('tipo_viaje', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	link_to(['Finalizar'],'gest_viajes/index')
Descripción:	Función que permite visualizar un vínculo.

Nombre:	input_date_tag('fecha_viaje', "", array)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	select_tag('aplicacion', options_for_select(\$aplicaciones),array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('aplicacion', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	image_tag('/images/indicadores/indicator_snake.gif')
Descripción:	Función que permite visualizar una imagen.
Nombre:	submit_tag('Agregar viaje')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	form_error('fecha_viaje', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.

TABLA 67 : Descripción de la CI mostrar datos del caso de uso gestionar viajes

Nombre: datosSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	use_helper('DateForm')
Descripción:	Función que permite incluir clases a usar.
Nombre:	input_date_tag('fecha_inicio', "", array)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	input_date_tag('fecha_fin', "", array)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	select_tag('tipo_res', options_for_select(array))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	input_date_tag('fecha_inicio', "", array)
Descripción:	Función que permite visualizar un campo de tipo fecha

Nombre:	input_date_tag('fecha_fin', ", array)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	input_hidden_tag('aplicacion_trabajadores', \$aplicacion)
Descripción:	Función que permite crear un campo oculto.
Nombre:	select_time_tag('hora_salida', ", array
Descripción:	Función que permite visualizar un campo para seleccionar el tiempo, horas-minutos-segundos.
Nombre:	select_time_tag('hora_llegada')
Descripción:	Función que permite visualizar un campo para seleccionar el tiempo, horas-minutos-segundos.
Nombre:	select_tag('hora_trompo', options_for_select(\$hora_trompo))
Descripción:	Función que permite visualizar un menú desplegable.

TABLA 68 : Descripción de la CI mostrar detalles del caso de uso gestionar viajes

Nombre: mostrarSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Date', 'Fechas')
Descripción:	Función que permite incluir clases a usar.
Nombre:	image_tag('reservacion.png', array)
Descripción:	Función que permite visualizar una imagen.
Nombre:	link_to(['Crear viaje'], 'gest_viajes/crear')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	form_tag('gest_viajes/deshabilitarVarios')
Descripción:	Función que permite crear un formulario.
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	visual_effect('fade', 'vtncontainer')
Descripción:	Función que permite visualizar un efecto ajax.
Nombre:	visual_effect('appear', 'vtncontainer')

Descripción:	Función que permite visualizar un efecto ajax.
Nombre:	link_to(image_tag('/images/acciones/e-mail.png', array
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to(\$viaje_tn->getTbDviajeTnl()->getFechaInicioPr('d/m/Y h:i'),'gest_viajes/editar)
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to(\$viaje_tn->getTbDviajeTnl()->getFechaFinPr('d/m/Y h:i'),'gest_viajes/editar)
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to(image_tag('/images/acciones/add.png',array)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	image_tag('/images/indicadores/indicator_snake.gif')
Descripción:	Función que permite visualizar una imagen.
Nombre:	javascript_tag( remote_function( array)
Descripción:	Función que permite usar código javascript donde se llame.
Nombre:	link_to(image_tag('/images/acciones/eliminar.gif,array)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to(image_tag('/images/bus1.jpg',array)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to(image_tag('/images/reporte.gif', array)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to('Distribuciones', 'distribuir/index)
Descripción:	Función que permite visualizar un vínculo.
Nombre:	image_tag('/images/indicadores/indicator_snake.gif')
Descripción:	Función que permite visualizar una imagen.
Nombre:	visual_effect('fade', 'vtnpcontainer')
Descripción:	Función que permite visualizar un efecto ajax.
Nombre:	visual_effect('appear', 'vtnpcontainer')
Descripción:	Función que permite visualizar un efecto ajax.
Nombre:	form_tag('gest_viajes/index')
Descripción:	Función que permite crear un formulario.
Nombre:	select_tag('pagina', options_for_select(\$meses))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('Filtrar')

Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to(image_tag('/images/lists.png', array)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	format_date(\$viaje_tnp->getTbDviajeTnl()->getTbDviaje()->getFechaViaje(),'f','es')
Descripción:	Función que permite darle formato a una fecha
Nombre:	format_date(\$viaje_tnp->getTbDviajeTnl()->getFechaInicioPr(),'f','es')
Descripción:	Función que permite darle formato a una fecha
Nombre:	format_date(\$viaje_tnp->getTbDviajeTnl()->getFechaFinPr(),'f','es')
Descripción:	Función que permite darle formato a una fecha
Nombre:	link_to(image_tag('/images/bus1.jpg',array)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to(image_tag('/images/reporte.gif', array)
Descripción:	Función que Visualiza un vínculo en una imagen.
Nombre:	link_to_remote(image_tag('/images/acciones/e-mail.png', array)
Descripción:	Función que permite visualizar un vínculo de tipo ajax en una imagen
Nombre:	link_to_remote(image_tag('/images/acciones/error.png', array)
Descripción:	Función que permite visualizar un vínculo de tipo ajax en una imagen
Nombre:	form_tag('gest_viajes/index')
Descripción:	Función que permite crear un formulario.
Nombre:	select_tag('pagina_te', options_for_select(\$meses))
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('Filtrar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	link_to(image_tag('/images/reporte.gif', array)
Descripción:	Función que Visualiza un vínculo en una imagen.

TABLA 69 : Descripción de la CI editar del caso de uso gestionar viajes

Nombre: editSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	

Nombre:	use_helper('Validation')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_viajes/editar')
Descripción:	Función que permite crear un formulario.
Nombre:	input_hidden_tag('aplicacion', \$aplicacion)
Descripción:	Función que permite crear un campo oculto.
Nombre:	input_hidden_tag('id_viaje', \$viaje_a_editar)
Descripción:	Función que permite crear un campo oculto.
Nombre:	input_date_tag('fecha_viaje', \$viaje_a_editar)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	form_error('fecha_viaje', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	input_date_tag('fecha_inicio_pr', \$viaje_a_editar)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	form_error('fecha_inicio_pr', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	input_date_tag('fecha_fin_pr', \$viaje_a_editar)
Descripción:	Función que permite visualizar un campo de tipo fecha
Nombre:	form_error('fecha_fin_pr', array)
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	submit_tag('Guardar')
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	button_to('Atr&aacute;s', 'gest_viajes/index')
Descripción:	Función que permite visualizar un botón.

TABLA 70 : Descripción de la CI viajes correspondientes del caso de uso gestionar viajes

Nombre: correspondienteSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Javascript', 'Date')



Descripción:	Función que permite incluir clases a usar.
Nombre:	format_datetime(\$viaje[0]->getFechaViaje(),'P', 'es')
Descripción:	Función que permite dar formato a una fecha.
Nombre:	select_tag('id_viaje_vinculado', options_for_select(\$filtrados))
Descripción:	Función que permite visualizar un menú desplegable.

TABLA 71 : Descripción de la CI editar rutas del caso de uso gestionar rutas

Nombre: editSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	use_helper('Validation')
Descripción:	Función que permite incluir clases a usar.
Nombre:	use_helper('Javascript')
Descripción:	Función que permite incluir clases a usar.
Nombre:	form_tag('gest_rutas/update')
Descripción:	Función que permite crear un formulario.
Nombre:	input_hidden_tag('id_ruta', \$tb_druta->getIdRuta())
Descripción:	Función que permite crear un campo oculto.
Nombre:	form_error('itinerario_i')
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	input_tag('itinerario_i', \$tb_druta->getItinerarioI() , array)
Descripción:	Función que visualiza un campo de texto.
Nombre:	form_error('itinerario_r')
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	input_tag('itinerario_r', \$tb_druta->getItinerarioR() , array)
Descripción:	Función que visualiza un campo de texto.
Nombre:	form_error('numero_ruta')
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	input_tag('numero_ruta', \$tb_druta->getNumeroRuta() , array)
Descripción:	Función que visualiza un campo de texto.

Nombre:	form_error('ruta')
Descripción:	Función que permite visualizar el error cometido en el campo asociado de un formulario.
Nombre:	input_tag('ruta', \$tb_druta->getRuta() , array)
Descripción:	Función que visualiza un campo de texto.
Nombre:	select_tag('id_aplicacion', \$aplicaciones, array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('id_aplicacion', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	select_tag('id_psuci', \$puntos)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('id_psuci', array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('id_provincia', \$provincias, array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('id_municipio', \$municipios)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	select_tag('id_municipio', array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	observe_field('id_provincia', array)
Descripción:	Función que permite ejecutar una acción cuando ocurre algún cambio en el elemento que tiene asignado.
Nombre:	select_tag('id_puntosintermedios', \$parada_intermedia , array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('>>',array('name'=>'add'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	submit_tag('<<',array('name'=>'del'))
Descripción:	Función que visualiza un botón que envía el formulario.
Nombre:	select_tag('paradas_ruta', \$paradas_ruta , array)
Descripción:	Función que permite visualizar un menú desplegable.
Nombre:	submit_tag('Guardar', array('name'=>'sav'))
Descripción:	Función que visualiza un botón que envía el formulario.

Nombre:	link_to('Eliminar', 'gest_rutas/delete')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Cancelar', 'gest_rutas/list')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Cancelar', 'gest_rutas/list')
Descripción:	Función que permite visualizar un vínculo.

TABLA 72 : Descripción de la CI listar del caso de uso gestionar ruta

Nombre: listSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to(image_tag('/images/lists.png', array)
Descripción:	Función que permite visualizar un vínculo.
Nombre:	button_to ('Nueva Ruta', 'gest_rutas/create')
Descripción:	Función que permite visualizar un botón.

TABLA 73 : Descripción de la CI mostrar municipios del caso de uso gestionar rutas

Nombre: showMunicipiosSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	select_tag('id_municipio', \$municipios)
Descripción:	Función que permite visualizar un menú desplegable.

TABLA 74 : Descripción de la CI mostrar puntos del caso de uso gestionar rutas

Nombre: showPuntosSuccess	
Tipo de clase: interfaz	
Atributo	Tipo

Para cada responsabilidad:	
Nombre:	select_tag('id_psuci', \$puntos)
Descripción:	Función que permite visualizar un menú desplegable.

TABLA 75 : Descripción de la CI mostrar detalles del caso de uso gestionar rutas

Nombre: showSuccess	
Tipo de clase: interfaz	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	link_to('Modificar', 'gest_rutas/edit')
Descripción:	Función que permite visualizar un vínculo.
Nombre:	link_to('Listado', 'gest_rutas/list')
Descripción:	Función que permite visualizar un vínculo.

### 2.6.3. Clases Entidades (CE)

TABLA 76 : Descripción de la CE viaje

Nombre: TbDviaje	
Tipo de clase: entidad	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	DistribuirMunicipioEn(\$id_municipio,\$id_transporte)
Descripción:	Función que permite distribuir todas las personas que reservaron para un municipio.
Nombre:	DistribuirReservacionEn(\$id_reservacion, \$id_transporte)
Descripción:	Función que permite Distribuir una persona en un transporte dada el id de la reservación y el id del transporte.
Nombre:	Desactivar()
Descripción:	Función que permite cambiar el estado de un viaje así como las reservaciones y transporte.
Nombre:	CancelarDistribucionAutomatica(\$ids_municipio)

Descripción:	Función que permite cancelar la distribución automática de una provincia.
Nombre:	CancelarDistribucionMunicipio(\$id_municipio)
Descripción:	Función que permite cancelar la distribución de un municipio.
Nombre:	CancelarPuntual(\$id_reservacion, \$id_transporte)
Descripción:	Función que permite cancelar la distribución de una reservación para un transporte.
Nombre:	EstaReservada(\$id_persona)
Descripción:	Función que permite verificar si una persona está reservada.

TABLA 77 : Descripción de la CE viajes aux

Nombre: TbDviajePeer	
Tipo de clase: entidad	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	CancelarReservacion(\$id_reservacion)
Descripción:	Función que permite cancelar una reservación.
Nombre:	DeshabilitarVarios(\$array_ids)
Descripción:	Función que permite deshabilitar un grupo de viajes.

TABLA 78 : Descripción de la CE aplicación

Nombre: TbNaplicacionPeer	
Tipo de clase: entidad	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	doSelectAsArray(\$criteria)
Descripción:	Función que permite devolver una consulta en un arreglo.

TABLA 79 : Descripción de la CE transporte

Nombre: TbDtransportePeer	
Tipo de clase: entidad	
Atributo	Tipo

Para cada responsabilidad:	
Nombre:	ListadoTransporte(\$id_transporte)
Descripción:	Función que permite obtener las personas de un transporte.

TABLA 80 : Descripción de la CE viajes general

Nombre: TbDviajesTnl	
Tipo de clase: entidad	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	ListadoTransporte(\$id_transporte)
Descripción:	Función que permite obtener las personas que están reservadas para un transporte.
Nombre:	EstaEnPeriodoReservacion()
Descripción:	Función que permite saber si un viaje tnl está en período de reservación.

TABLA 81 : Descripción de la CE viajes tn

Nombre: TbDviajesTn	
Tipo de clase: entidad	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Desactivar()
Descripción:	Función que permite cambiar el estado de un viaje.
Nombre:	ProvinciasViaje()
Descripción:	Función que permite obtener las provincias que están en las reglas del viaje.
Nombre:	Boletines(\$id_transporte)
Descripción:	Función que permite obtener todas las distribuciones dado un transporte.

TABLA 82 : Descripción de la CE viajes te

Nombre: TbDviajeTe	
Tipo de clase: entidad	

Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Desactivar()
Descripción:	Función que permite cambiar el estado de un viaje.

TABLA 83 : Descripción de la CE transporte

Nombre: TbDtransporte	
Tipo de clase: entidad	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Capacidad()
Descripción:	Función que permite obtener la capacidad de un transporte.
Nombre:	Disponibles()
Descripción:	Función que permite obtener la cantidad de asientos desocupados de un transporte.
Nombre:	VerificarAplicacion()
Descripción:	Función que verifica a que aplicación pertenece el transporte.
Nombre:	ParadasIntermedias()
Descripción:	Función que permite obtener las paradas intermedias de un transporte.
Nombre:	Desactivar()
Descripción:	Función que permite cambiar el estado de un transporte.

TABLA 84 : Descripción de la CE bloque

Nombre: TbDbloque	
Tipo de clase: entidad	
Atributo	Tipo
Para cada responsabilidad:	
Nombre:	Desactivar()
Descripción:	Función que permite cambiar el estado de un bloque.

## **2.7. Conclusiones**

Al finalizar este capítulo queda implementado el sistema y descritas todas las clases utilizadas, cumpliendo con los objetivos específicos planteados y gran parte de las tareas propuestas para la elaboración del sistema, a partir de una breve descripción de como tener acceso a las funcionalidades implementadas, y el sistema al estar completamente funcional, se concluye expresando que se debería pasar a la etapa de pruebas y realizar las ensayos necesarios para asegurar que el sistema esté libre de no conformidades y con la debida calidad para ser entregada al cliente.



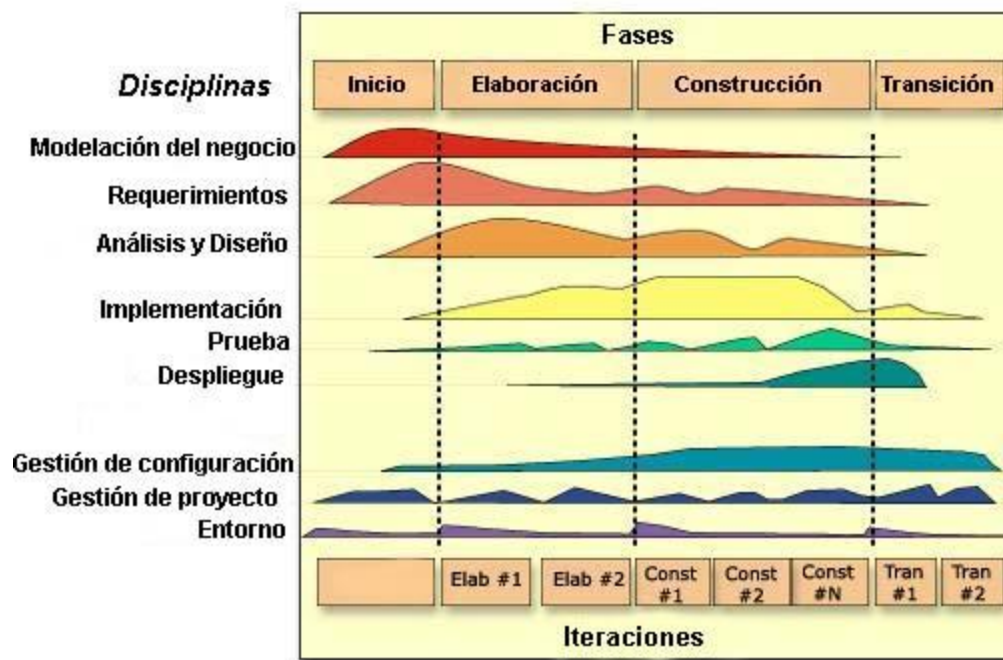
# 3

## CAPÍTULO 3

### VALIDACIÓN DE LA SOLUCIÓN PROPUESTA

#### 3.1. Introducción

En el presente capítulo se realizará un análisis de los diferentes tipos de pruebas donde se profundizará en las pruebas de unidades que son las que se usarán para validar la solución propuesta, específicamente la prueba de caja blanca. Se introducirán conceptos como: tipos de pruebas, pruebas de unidad y ventajas de las mismas y prueba de Caja Blanca. La realización de prueba de Caja Blanca a las principales funcionalidades de los módulos de Administración – Configuración y Seguridad del SIT. Teniendo como base los objetivos que propone RUP en el flujo de trabajo prueba, como se muestra en la figura 3.1.

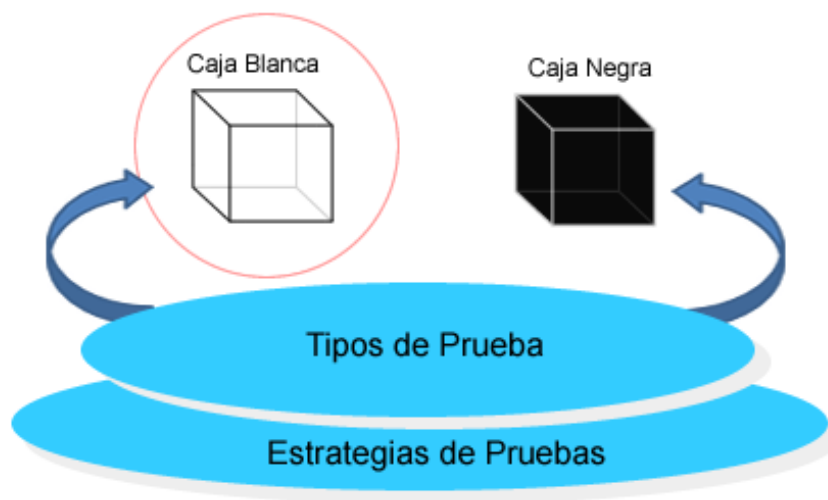


**Fig 3.1** Fases y flujos de trabajos de RUP

### 3.1.1. Objetivos del flujo de trabajo de prueba

- Encontrar y documentar los defectos que puedan afectar la calidad del software.
- Verificar que el software trabaje como fue diseñado.
- Validar y probar los requisitos que debe cumplir el software.
- Validar que los requisitos fueron implementados correctamente.

### 3.2. Método de prueba adecuado al nivel de unidad



**Fig 3.2** Método de prueba

#### 3.2.1. Métodos de prueba basados en caja blanca

- ✓ La prueba del camino básico

Esta prueba permite al diseñador de casos de prueba obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución.

- ✓ La prueba de condición

Es un método de diseño de casos de prueba que ejercita las condiciones lógicas contenidas en el módulo de un programa.

- ✓ La prueba de flujo de datos

Se seleccionan caminos de prueba de un programa de acuerdo con la ubicación de las definiciones y los usos de las variables del programa.

- ✓ La prueba de bucles

Una técnica de prueba de caja blanca que se centra exclusivamente en la validez de las construcciones de bucles.

### 3.2.2. Propuesta de prueba del framework Symfony

En el ámbito de PHP existen muchos frameworks para crear pruebas unitarias, siendo los más conocidos PHPUnit y SimpleTest. Symfony incluye su propio framework llamado Lime. Se basa en la librería Test::More de Perl y es compatible con TAP, lo que significa que los resultados de las pruebas se muestran con el formato definido en el *“Test Anything Protocol”*, creado para facilitar la lectura de los resultados de las pruebas. Lime proporciona el soporte para las pruebas unitarias, es más eficiente que otros frameworks de pruebas de PHP y tiene las siguientes ventajas (6):

- Ejecuta los archivos de prueba en un entorno independiente para evitar interferencias entre las diferentes pruebas. No todos los frameworks de pruebas garantizan un entorno de ejecución “limpio” para cada prueba.
- Las pruebas de Lime son fáciles de leer y sus resultados también lo son. En los sistemas operativos que lo soportan, los resultados de Lime utilizan diferentes colores para mostrar de forma clara la información más importante. Symfony utiliza Lime para sus propias pruebas y su

“regression testing”, por lo que el código fuente de Symfony incluye muchos ejemplos reales de pruebas unitarias y funcionales.

- El núcleo de Lime se valida mediante pruebas unitarias.
- Está escrito con PHP, es muy rápido y está bien diseñado internamente. Consta únicamente de un archivo, llamado lime.php, y no tiene ninguna dependencia. Las pruebas que se muestran en las secciones siguientes utilizan la sintaxis de Lime, por lo que funcionan directamente en cualquier instalación de symfony.

### **3.2.3. Pruebas unitarias**

Las pruebas unitarias de Symfony son archivos PHP normales cuyo nombre termina en Test.php y que se encuentran en el directorio test/unit/ de la aplicación. Su sintaxis es sencilla y fácil de leer.

La figura 3.3 muestra un conjunto típico de pruebas unitarias para la función Login (). En primer lugar, se instancia el objeto lime\_test. Cada prueba unitaria consiste en una llamada a un método de la instancia de lime\_test. El último parámetro de estos métodos siempre es una cadena de texto opcional que se utiliza como resultado del método.

```

1  <?php
2
3  //*****inicializaciones*****
4  define('SF_ROOT_DIR',    dirname(__FILE__).'/../..');
5  define('SF_APP',        'uci_transporte');
6  define('SF_ENVIRONMENT', 'test');
7  define('SF_DEBUG',      true);
8  include(dirname(__FILE__).'/../bootstrap/unit.php');
9  require_once
10     SF_ROOT_DIR.DIRECTORY_SEPARATOR.'apps'.DIRECTORY_SEPARATOR.SF_APP.
11     DIRECTORY_SEPARATOR.'config'.DIRECTORY_SEPARATOR.'config.php';
12
13  //*****
14
15
16  $t = new lime_test(3, new lime_output_color());
17
18  $ldap = new ldap();
19
20  $t->diag('-----Probando classws.php class Ldap-----');
21  $t->diag(' ');
22  $t->isa_ok($ldap->Login('jacalderon','*****'),'boolean','Verificar Tipo de Dato esperado');
23  $t->ok($ldap->Login('jacalderon','jawedeberit'),'Prueba del metodo Login para autenticar');
24  $t->is($ldap->Login('jacalderon','jawedeberitty'),'false','Verificar Valores Esperados');
25  $t->diag(' ');
26  $t->diag('----- FIN Probando classws.php class Ldap-----');

```

Fig 3.3 Prueba unitaria en test/unit/ldapTest.php

Para ejecutar el conjunto de pruebas, se utiliza la tarea test-unit desde la línea de comandos. El resultado de esta tarea en la línea de comandos es muy explícito, lo que permite localizar fácilmente las pruebas que han fallado y las que se han ejecutado correctamente. La figura 3.4 Muestra el resultado del ejemplo anterior.

```

Z:\>D:\wamp\bin\php\php5.2.5\php.exe symfony test-unit Ldap
1..3
# -----Probando classws.php class Ldap-----
#
ok 1 - Verificar Tipo de Dato esperado
ok 2 - Prueba del metodo Login para autenticar
not ok 3 - Verificar Valores Esperados
#   Failed test (.lib\symfony\vendor\lime\lime.php at line 103)
#       got: false
#       expected: 'false'
#
# ----- FIN Probando classws.php class Ldap-----
Looks like you failed 1 tests of 3.

```

Fig 3.4 Prueba unitaria en test/unit/ldapTest.php

### 3.2.4. Pruebas realizadas al sistema

#### 3.2.4.1. Objetivo

El objetivo de realizar este tipo de prueba al sistema es garantizar que se ejerciten por lo menos una vez todos los caminos independientes de cada método, todas las llamadas a los Servicios Web con gran cantidad de datos y su respuesta en sus límites operacionales; así como las estructuras internas de datos para asegurar su validez y el tipo de dato de retorno.

#### 3.2.4.2. Alcance

Con este tipo de prueba se pretende centralizarse en las funcionalidades de más peso en la aplicación, para asegurar, que estas estén libres de posibles errores que puedan ocurrir en la ejecución del mismo y comprobar la eficiencia del sistema implementado.

#### 3.2.4.3. Descripción de los valores utilizados para los test

Estas pruebas son completamente internas, ejecutadas en el código del programa, analizando la estructura interna de la aplicación, con las posibles alternativas o diferentes caminos que puede acceder el código con disímiles variables que se le puede pasar a la funcionalidad para su funcionamiento, para este tipo de prueba symfony trae su propio framework llamado Lime, de este framework, se utilizará tres pruebas fundamentales las cuales ayudarán a demostrar, la calidad de las funcionalidades, su buen funcionamiento y escasez de errores.

TABLA 85 : Métodos utilizados en las pruebas

Método	Descripción
diag(\$mensaje)	Muestra un comentario, pero no ejecuta ninguna prueba.
ok(\$prueba, \$mensaje)	Si la condición que se indica es true, la prueba tiene éxito.
is(\$valor1, \$valor2, \$mensaje)	Compara 2 valores y la prueba pasa si los 2 son iguales (==).
isa_ok(\$objeto, \$clase, \$mensaje)	Comprueba si el objeto que se le pasa es de la clase que se indica.

Este tipo de pruebas se utiliza para el conocimiento y ayuda de los programadores para confirmar que las funcionalidades devuelvan un resultado esperado, tipo de variable esperada, verifica si los datos recibidos son variables válidas o inválidas.

### 3.2.4.4. Evaluación de la ejecución del test y de los resultados obtenidos

Para la evaluación y ejecución de las pruebas se tomaron las funcionalidades más importantes vinculadas a los módulos asociados, para ello se prueban las clases ldap, classws, TbDviajeTnl, TbDviajeTn y myuser.

TABLA 86 : Clase ldap

#	Funcionalidad	Tipo de Prueba	Recibe	Esperado	Resultado
1	Login()	isa_ok	'jacalderon','*****'	ok	ok
2	Login()	ok	'jacalderon','jawedeberit'	ok	ok
3	Login()	is	'jacalderon','jawedeberitty'	Not ok	Not ok
4	SearchUser()	isa_ok	'jacalderon','*****', 'irrodriguez'	ok	ok
5	SearchUser()	ok	'jacalderon','jawedeberit', 'irrodriguez'	ok	ok
6	SearchUser()	is	'jacalderon','jawedeberit', 'irrodriguez'	ok	ok

TABLA 87 : Clase classws

#	Funcionalidad	Tipo de Prueba	Recibe	Esperado	Resultado
1	ScPersonaDadoLoginTested("jacalderon")	ok	"jacalderon"	ok	ok
2	ScPersonaDadoLoginTested("jacalderon")	is	"jacalderon"	ok	ok
3	ScPersonaDadoLoginTested("jacalderon")	isa_ok	"jacalderon"	ok	ok
4	ObtenerEstudianteDadoidGrupo()	is	"000104"	ok	ok
5	ObtenerEstudianteDadoidGrupo()	isa_ok	"000104"	ok	ok
8	ObtenerProvinciaDadoid()	is	"0100"	ok	ok
9	ObtenerProvinciaDadoid()	isa_ok	"0100"	ok	ok
10	ObtenerMunicipiosXidProv()	is	'valido'	ok	ok
11	ObtenerMunicipiosXidProv()	isa_ok	'valido'	ok	ok

TABLA 88 : Clase TbDviajeTnl

#	Funcionalidad	Tipo de Prueba	Recibe	Esperado	Resultado
1	ObtenerTransportes	is	"0100"	ok	ok
2	ObtenerTransportes	isa_ok	"0100"	ok	ok

TABLA 89 : Clase TbDviajeTn

#	Funcionalidad	Tipo de Prueba	Recibe	Esperado	Resultado
1	Boletines	is	"620"	ok	ok
2	Boletines	isa_ok	"620"	ok	ok

TABLA 90 : Clase MyUser

#	Funcionalidad	Tipo de Prueba	Recibe	Esperado	Resultado
1	Funcionalidades Usuario	is	"50024"	ok	ok
2	Funcionalidades Usuario	isa_ok	"50024"	ok	ok
3	ProvinciaBloqueada	ok	"0100","112"	Parámetros Válidos	Parámetros Válidos
4	ProvinciaBloqueada	is	"0100","112"	Not ok	Not ok
5	ProvinciaBloqueada	isa_ok	"0100","112"	ok	ok
6	UsuarioBloqueado	ok	"50008","4"	Parámetros Inválidos	Parámetros Inválidos
7	UsuarioBloqueado	ok	"50008","0"	Parámetros Válidos	Parámetros Válidos
8	UsuarioBloqueado	is	"50008","0"	Not ok	Not ok
9	UsuarioBloqueado	isa_ok	"50008","0"	ok	ok
10	BloquesReglas	is	array( "0" => "L0000", "1" => "5", "2" => "0111", "3" => "0100", "4" => "Estudiante")	ok	ok
11	BloquesReglas	isa_ok	array( "0" => "L0000", "1" => "5", "2" => "0111", "3" => "0100", "4" => "Estudiante")	ok	ok
12	ViajesReglas	is	array( "0" => "L0000", "1" => "5", "2" => "0111", "3" => "0100", "4" => "Estudiante")	ok	ok
13	ViajesReglas	isa_ok	array( "0" => "L0000", "1" => "5", "2" => "0111", "3" => "0100", "4" => "Estudiante")	ok	ok

### 3.3. Conclusiones

Es importante y relevante recalcar que todas las pruebas que se le pueda realizar a un sistema no lo hace perfecto, ni mucho menos exento de errores; pero con las pruebas realizadas, se garantiza que los



módulos Administración - Configuración y Seguridad de SIT quede limpio de errores en el código, muestra de ello son las pruebas realizadas y sus resultados. En todos los casos las pruebas devolvieron el resultado esperado, asegurando la fiabilidad y eficiencia del sistema para el desarrollo de un sistema con alta y potente calidad.

## Conclusiones

El trabajo siempre ha sido el forjador del hombre en la elaboración de conocimientos y de su experiencia propia. Muchos trabajos de este tipo han ayudado a la formación profesional de las personas y si hay algo que se ha aprendido en esta labor es escalar en el mundo de la informática, por lo tanto con la elaboración de este trabajo se logró:

- ✓ La implementación de los módulos Administración - Configuración y Seguridad con su documentación asociada.
- ✓ Dar cumplimiento al objetivo general y a los específicos planteados para la implementación de SIT.
- ✓ La realización de un sistema potencialmente navegable, confiable y eficiente.
- ✓ Realizar pruebas unitarias al sistema.

Además se realizó un estudio de las herramientas usadas para la elaboración de la solución propuesta posibilitando estos logros antes mencionados.

## Recomendaciones

Una vez vencidos los objetivos de esta investigación, teniendo en cuenta las experiencias obtenidas a lo largo de su desarrollo, se recomienda:

- ✓ Seguir trabajando en el desarrollo del sistema si son añadidos a éste otros requerimientos funcionales, pues se cuenta con las especificaciones necesarias para comenzar a materializar la implementación del mismo.
- ✓ Implementar servicios web para las aplicaciones de Residencia y Control de Acceso.
- ✓ Implantar el SIT en las Facultades Regionales.

Con la culminación de todas las etapas de desarrollo de software se demuestra que el software se está convirtiendo en un producto muy importante en una sociedad con una población altamente entrenada, educada y capacitada.

## Referencias Bibliográficas

1. (27 de marzo 2009). Retrieved 28 de marzo, 2009, from <http://mx.php.net/manual/es/history.php.php>.
2. (2008). "Internet History " Retrieved 10 de Febrero, 2009, from [http://www.computerhistory.org/internet\\_history/](http://www.computerhistory.org/internet_history/).
3. (W3C), W. W. W. C. (27 de marzo del 2009). Retrieved 27 de marzo, 2009, from <http://www.w3.org/>.
4. Casamor, A. S. (2003). Los protocolos en las redes de ordenadores: 165.
5. Cerami, E. (2002). Web Services Essentials: 288.
6. Fabien Potencier, F. Z. (2008). Symfony la guía definitiva
7. (Group, et al.)
8. (IBERWORLD AIRLINES)
9. (Radixx Solutions International, 1993)

## Bibliografía

**Alaska Air Group, Alaska Airlines and Horizon Air Industries.** Alaska Airlines | Horizon Air. [En línea] <http://www.alaskaair.com>.

Cubana de Aviación. [En línea] <http://www.cubana.cu>.

**Fabien Potencier . 2005.** Symfony , Web PHP framework. Symfony , Web PHP framework. [En línea] Sensio-Labs, 2005. [www.symfony-project.org](http://www.symfony-project.org).

**IBERWORLD AIRLINES, S.A. IBERWORLD.** [En línea] <http://www.iberworld.es>.

**Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000.** *El proceso unificado de desarrollo de software*. 2000.

**Jim Conallen, Building Web Applications with UML, Object Technology Series by Addison Wesley Longman, 1999**

**Jim Conallen, Modeling Web Application Architectures with UML, Rational Software, June 1999**  
[http://www.rational.com/media/uml/resources/documentation/27662\\_webapps.pdf](http://www.rational.com/media/uml/resources/documentation/27662_webapps.pdf)

**Jim Conallen, UML Extension for Web Applications 0.91, 22-Mar-1999.**

**Pressman, Roger S. 2001.** Ingeniería del Software. Un enfoque práctico.(traducido). 2001.

**Radixx Solutions International, Inc. 1993.** Radixx International. [En línea] <http://radixx.com>.

**Roberto Hernández Sampieri, Carlos Fernández Collado, Pilar Baptista Lucio. 1998.** Metodología de la Investigación. México D.F. : McGRAW-HILL INTERAMERICANA EDITORES, S. A. de C. V., 1998. ISBN 970-10-1899-0.

**Stan Ward, Per Kroll, Building Web Solutions with the Rational Unified Process: Unifying the Creative Design Process and the Software Engineering Process.**

## Anexos

### Anexo I

TABLA 91 : Operacionalización de las Variables

VARIABLES	DIMENSIONES	INDICADORES	ÍNDICE DE LOS INDICADORES
Módulos de Administración – Configuración y Seguridad.	Administración	Ordenamiento	Simple
			Complejo
		Funcionamiento	Mínimo
			Medio
			Alto
		Seguridad Informática	Confidencialidad
	Media		
	Baja		
	Integridad		Alta
			Media
			Baja
	Disponibilidad	Alta	
Media			
Baja			
Eficiencia en la configuración de transportaciones.	Eficiente	Rapidez	Alta
			Media
			Baja
		Corrección de errores	Alta
			Media
			Baja
		Organización del trabajo	Bueno
			Malo
		La Seguridad de la información manejada	Seguridad Informática
No			

## Glosario de términos

- ✓ **AJAX** (*Asynchronous Javascript and XML*): Javascript asíncrono y XML es un nuevo avance de la tecnología con el objetivo de incrementar el intercambio de datos con el servidor.
- ✓ **CSS**: Hojas en estilo de cascada aplicables a documentos HTML que contienen diferentes estilos y son fáciles de cambiar y diseñar.
- ✓ **DHTML** (*Dynamic HTML*): Idioma de hipertexto dinámico, compuesto de opciones agregadas en los documentos HTML.
- ✓ **DOM** (*Document Object Model*): Una especificación W3C para interfaces de programa de aplicación para el acceso al contenido de los documentos HTML y XML.
- ✓ **Framework**: Una estructura para elaboración de proyectos donde se parte de un esqueleto de apoyo utilizado como base para algo que se está construyendo.
- ✓ **FTP** (*File Transfer Protocol*): Protocolo de transferencia de archivos
- ✓ **Helpers**: Los Helpers son clases a modo de componentes para la capa de presentación de la aplicación que contienen lógica de presentación que puede ser compartida por muchas vistas, elementos y layouts.
- ✓ **HTML** (*HyperText Markup Language*): Es el lenguaje para representar la información que contiene la página web, es decir es el lenguaje para codificar y construir una página web (3).
- ✓ **HTTP** (*Hypertext Transfer Protocol*): Protocolo de transmisión del hipertexto basados fundamentalmente en el lenguaje HTML (3).
- ✓ **IDE** (*Integrated Development Enviroment*): Entorno Integrado de Desarrollo.
- ✓ **Internet**: Red de computadoras alrededor de todo el mundo que comparten información unas con otras por medio de páginas o sitios.
- ✓ **IIS** (*Internet Information Server*): Es una serie de servicios para los ordenadores que funcionan con Windows.

- ✓ ORM (*Object-Relational Mapping*): El mapeo objeto-relacional es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional.
- ✓ PHP (*Hypertext Pre-processor*): Es un lenguaje de programación utilizado para la creación de sitios web. PHP es un acrónimo recursivo que significa “PHP Hypertext Pre-processor”, (inicialmente se llamó Personal Home Page).
- ✓ Prototype: Framework escrito en Javascript que se orienta al sencillo y dinámico de aplicaciones web, amplía las posibilidades de programar en ese lenguaje, ofrece muy buenos mecanismos para la manipulación del DOM optimizando la obtención de objetos, clases CSS permitiendo un lenguaje tan conciso como el PHP.
- ✓ SOAP (*Simple Object Access Protocol*): es un protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. Es uno de los protocolos utilizados en los servicios Web.
- ✓ SQL (*Structured Query Language*): Lenguaje de Consulta Estructurado, es un lenguaje declarativo de acceso a bases de datos relacionales.
- ✓ SSL (*Secure Sockets Layer*): Protocolo diseñado por la empresa Netscape para proveer comunicaciones encriptadas.
- ✓ Servicio Web: Es una colección de protocolos y estándares que sirven para intercambiar datos entre aplicaciones.
- ✓ UDDI (*Universal Description, Discovery and Integration*): Descripción, Descubrimiento e Investigación Universal es un elemento básico sobre el que se asientan los Servicios Web, hace posible que empresas pueden tanto publicar como encontrar servicios web (5).
- ✓ URL (*Uniform Resource Locator*): Localizador uniforme de recursos. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.



- ✓ WSDL (*Web Service Definition Language*): El lenguaje de descripción de servicios web permite definir lo que hace un Servicio Web según la funcionalidad que ofrece. Mediante este lenguaje se representa la interfaz de uso del servicio, lo que tendrán que tener en cuenta otros servicios a la hora de acceder a su funcionalidad (5).
- ✓ WWW (*World Wide Web*): Red Global Mundial es básicamente un medio de comunicación de texto, gráficos y otros objetos multimedia a través de Internet, es decir, la web es un sistema de hipertexto que utiliza Internet como su mecanismo de transporte.
- ✓ XML (*Extensible Markup Language*): un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium.