

**Universidad de las Ciencias Informáticas**

**Facultad 1**



**“Diseño e implementación del Módulo Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales”**

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

Autores:

Maite Correa Varona  
Alexander Escalona Fernández

Tutores:

Ing. Yanet Casado Maceo  
Ing. Yurisleis Alonso Beatón

Ciudad de la Habana, Junio 2010.  
“Año 52 de la Revolución”



*"El conocimiento nos hace responsables."*

*Ernesto Che Guevara*

*Maite*

*A:*

*Mi abuelo Cosita, por dejarme todo su amor y el mejor recuerdo del mundo.*

*Alexander*

*A:*

*Mi familia y amigos por el apoyo incondicional que siempre me han dado.*

*Maite*

*A mis padres por apoyarme en todo y darme siempre el mejor ejemplo. Mamita gracias por el amor que siempre me has dado, no sé qué sería de mí sin tus mimos y tus cariños, eres la mejor mamá del mundo, Pipo hoy espero que estés muy orgulloso de mí, esa siempre ha sido mi mayor meta. Los amo mucho.*

*A mis hermanas por tanto cariño, son las personas más importantes de mi vida y mi mayor orgullo.*

*A Maiquel Luis por ser mi sobrino favorito.*

*A Jose, por su extraordinaria forma de ser, que se sobreentiende, porque él no es de este planeta.*

*A mis cuñados Nodier y Arnaldo por ser como hermanos para mí.*

*A Liki por ser como mi segunda mamá, gracias por todo.*

*A Tati por su amistad incondicional desde el primer día, te quiero mucho.*

*A Yailín y Adriana por compartir conmigo buenos y malos momentos durante todo este tiempo.*

*A mi compañero de tesis Alexander por ser tan talentoso y trabajador.*

*A mis tutores Yuri y Yanet por su importante ayuda en la investigación.*

*Alexander*

*Le agradezco ante todo a mi familia, especialmente a mi madre, sin su educación, paciencia y apoyo en todo momento no fuese lo que soy, ni hubiese llegado nunca a donde estoy. A mis amistades, tantos los que se encuentran aquí conmigo en la UCI, como a los que están a kilómetros de distancia, gracias a todos por compartir conmigo los buenos y malos momentos. A mi compañera de tesis, sin la cual nunca hubiese terminado la tesis en tiempo y forma. A todos mis amigos programadores, sin olvidarme del programador anónimo que muchos llaman Google, gracias por compartir líneas de código y conocimientos que fueron de gran ayuda en el desarrollo de la tesis. En fin, gracias a todos los que de una forma u otra hicieron posible la realización de este trabajo.*

**Declaración de autoría**

Declaramos que somos los únicos autores del trabajo titulado: “Diseño e implementación del Módulo Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales”, y otorgamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Maite Correa Varona

Alexander Escalona Fernández

\_\_\_\_\_

\_\_\_\_\_

Firma del Autor

Firma del Autor

Ing. Yurisleis Alonso Beatón

Ing. Yanet Casado Maceo

\_\_\_\_\_

\_\_\_\_\_

Firma del Tutor

Firma del Tutor



República de Cuba

MINISTERIO DE ECONOMÍA  
Y PLANIFICACIÓN

1960-2010  
Aniversario Cincuenta  
DE LA PLANIFICACIÓN

*Dirección Desarrollo Social*

La Habana, 18 de mayo de 2010  
"Año 52 de la Revolución"

## Opinión del usuario del Trabajo de Diploma

El Sistema Unificado de Gestión de Fuerza de Trabajo Calificada (GeForza), fue desarrollado para el Departamento Fuerza de Trabajo Calificada del Ministerio de Economía y Planificación. Esta entidad considera que, en correspondencia con los objetivos trazados, el trabajo realizado hasta la fecha, satisfará los procesos de gestión de la información y realización del Plan de Ingreso a la Educación Superior del Curso Regular Diurno, el Plan de Continuidad de Estudio de los estudiantes de noveno grado, el Plan de Ingreso a los Centros Universitarios Municipales y el Plan de Distribución de graduados de la Educación Superior formados en el Curso Regular Diurno.

El producto Sistema Unificado de Gestión de Fuerza de Trabajo Calificada (GeForza) será de gran utilidad para el Departamento Fuerza de Trabajo Calificada, para la organización y realización de sus actividades. Logrará que el trabajo de las personas que laboran de este departamento sea más fácil, así como, conseguirá homogeneizar los nomencladores, establecer y centralizar la seguridad de la información, agilizar el proceso de gestión de la información necesaria para la realización de los planes pertinentes al departamento y la elaboración de los mismos.

Los diplomantes del trabajo han dado muestra de preparación y creatividad, lo que está permitiendo dar solución a los problemas que presentan las aplicaciones actuales.

Y para que así conste, se firma la presente a los 18 días del mes de mayo del año 2010.

Lic. Virginia Martín Martínez  
J' Dpto. Fuerza de Trabajo Calificada

cc Manuel Puerta La Rosa



**Resumen:**

Actualmente se desarrollan en Cuba métodos para conocer y atender las solicitudes de matrícula de los trabajadores que las entidades o programas priorizados de la Revolución, proponen para que continúen estudios en las carreras que se forman en los Centros Universitarios Municipales.

El presente trabajo tiene como objetivo realizar un sistema para gestionar los procesos del plan de ingreso en los Centros Universitarios Municipales del Departamento Fuerza de trabajo Calificada (FTC) del Ministerio de Economía y Planificación (MEP). En esta tarea se maneja gran cantidad de información y es de suma importancia para el país que se realice en el menor tiempo posible, pero, en la actualidad no se garantiza seguridad en la información ni organización en la gestión y control del proceso. Es por ello la necesidad de desarrollar una aplicación que permita gestionar los procesos que se llevan a cabo para la elaboración del plan de ingreso en los Centros Universitarios Municipales. Para su desarrollo se realizó un amplio estudio de los procesos implicados en el tema, con el objetivo de garantizar un mejor funcionamiento. 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). Finalmente, se hicieron pruebas para validar el sistema propuesto, con el objetivo de lograr una aplicación capaz de satisfacer todas las necesidades del cliente.

El uso del sistema facilitará las tareas organizativas asociadas al proceso del plan de ingreso a los Centros Universitarios Municipales, disminuirá el tiempo y esfuerzo que se dedica a las mismas, garantizará fácil procesamiento de la información y obtención de reportes que permitirán la toma de decisiones oportunas y las acciones correctivas necesarias.



## Índice

<b>INTRODUCCIÓN</b> .....	1
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA</b> .....	5
1.1 Introducción.....	5
1.2 Estado del arte .....	5
1.2.1 Conceptos básicos .....	5
1.2.2 Los sistemas de gestión de información en el mundo .....	6
1.2.3 Los sistemas de gestión de información en Cuba.....	8
1.2.4 Ideas a utilizar .....	9
1.3 Tendencias actuales en el desarrollo de software .....	10
1.3.1 Lenguaje de modelado.....	11
1.3.2 Herramienta CASE para el modelado .....	12
1.3.3 Marcos de trabajo.....	14
1.3.4 Lenguajes para el desarrollo Web .....	15
1.3.5 Servidor Web .....	18
1.3.6 Sistemas Gestores de base de datos .....	19
1.3.7 Entorno Integrado de Desarrollo (IDE) .....	19
1.4 Conclusiones.....	20
<b>CAPÍTULO 2: DESCRIPCIÓN DE LA PROPUESTA DE SOLUCIÓN</b> .....	22
2.1 Introducción.....	22
2.2 Modelo de negocio .....	22
2.3 Descripción del negocio .....	22
2.4 Reglas del negocio.....	23
2.5 Actores del negocio.....	24
2.6 Trabajadores del negocio .....	24
2.7 Diagrama de casos de uso del negocio.....	25

---

2.8	Diagramas de Actividades de los casos de uso del negocio.....	25
2.9	Descripción de los casos de uso del negocio .....	26
2.10	Modelo de objetos.....	27
2.11	Actores del sistema.....	27
2.12	Especificación de los Requerimientos de Software .....	28
2.12.1	Requerimientos Funcionales .....	28
2.12.2	Requerimientos no Funcionales .....	46
2.13	Definición de los casos de uso del sistema .....	49
2.13.1	Listado de los casos de uso del sistema .....	49
2.13.2	Diagrama de casos de uso del sistema.....	50
2.13.3	Descripción textual de los casos de uso del sistema.....	51
2.14	Conclusiones:.....	61
<b>CAPÍTULO 3: CONSTRUCCIÓN DE LA SOLUCIÓN PROPUESTA .....</b>		<b>62</b>
3.1	Introducción.....	62
3.2	Análisis.....	62
3.2.1	Diagramas de clases del análisis .....	62
3.2.2	Diagramas de Colaboración del análisis.....	63
3.3	Diseño.....	64
3.3.1	Arquitectura y patrones de diseño utilizado .....	64
3.3.2	Modelo de diseño .....	67
3.3.3	Diagramas de secuencias .....	71
3.4	Diseño de la base de datos .....	75
3.5	Conclusiones.....	76
<b>CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA.....</b>		<b>77</b>
4.1	Introducción.....	77
4.2	Modelo de implementación.....	77

4.2.1	Modelo despliegue .....	77
4.2.2	Diagrama de componentes .....	78
4.3	Estándares de implementación .....	80
4.4	Pruebas.....	81
4.4.1	Modelos de pruebas .....	81
4.5	Conclusiones.....	82
<b>CONCLUSIONES .....</b>		<b>83</b>
<b>REFERENCIA BIBLIOGRÁFICA .....</b>		<b>85</b>
<b>GLOSARIO DE TÉRMINOS .....</b>		<b>88</b>
<b>ANEXOS .....</b>		<b>ERROR! BOOKMARK NOT DEFINED.</b>
Anexo # 1: Operacionalización de las variables: .....		<b>Error! Bookmark not defined.</b>
Anexo # 2: Modelo de objetos.....		<b>Error! Bookmark not defined.</b>
Anexo # 3: Diagramas de Actividades.....		<b>Error! Bookmark not defined.</b>
Anexo # 4: Descripciones de los caso de uso del negocio .....		<b>Error! Bookmark not defined.</b>
Anexo # 5: Descripciones de los caso de uso del sistema. ....		<b>Error! Bookmark not defined.</b>
Anexo # 6: Diagramas de clases del análisis: .....		<b>Error! Bookmark not defined.</b>
Anexo # 7: Diagramas de colaboración del análisis. ....		<b>Error! Bookmark not defined.</b>
Anexo # 8: Diagramas de clases del diseño.....		<b>Error! Bookmark not defined.</b>
Anexo # 9: Diagramas de secuencia del diseño.....		<b>Error! Bookmark not defined.</b>
Anexo # 10: Diagramas de componentes.....		<b>Error! Bookmark not defined.</b>
Anexo # 11: Modelo físico.....		<b>Error! Bookmark not defined.</b>
Anexo # 12: Secciones de Pruebas. ....		<b>Error! Bookmark not defined.</b>

## Introducción

El Ministerio de Economía y Planificación (MEP) es el organismo encargado de dirigir, ejecutar y controlar la aplicación de la política del Estado y el Gobierno en materia de economía, planificación, estadística y diseño. Dentro del MEP se encuentra el Departamento Fuerza de Trabajo Calificada (FTC), que surge con el objetivo de lograr el máximo equilibrio entre la formación y el empleo de los estudiantes de la Educación Superior, Técnica y Profesional, acorde a las necesidades reales de los territorios, el país y atendiendo las indicaciones del nivel superior en relación con esta temática. En él se lleva a cabo un grupo de tareas, entre las que está la elaboración del plan de ingreso a la Educación Superior en los Centros Universitarios Municipales. (1)

Para la elaboración del plan de ingreso a la Educación Superior en los Centros Universitarios Municipales las entidades y los programas priorizados de la Revolución, deben llenar una solicitud de matrícula que será enviada a las comisiones municipales donde se evaluará y se hará una propuesta del plan de ingreso a la Educación Superior en los Centros Universitarios Municipales a nivel municipal. Luego esta propuesta es elevada a la comisión provincial donde se analiza y finalmente el Ministerio de Economía y Planificación, con la propuesta de las comisiones provinciales, realiza el plan de ingreso a la Educación Superior a nivel nacional.

Estas actividades actualmente se realizan de forma manual, por lo que se plantea la siguiente **situación problemática**: el incremento gradual del volumen de información, según va transitando por los diferentes niveles, provoca que el proceso sea lento y engorroso. Además, se dificulta la elaboración de informes con datos estadísticos que permitan llevar un control sobre la información y realizar estudios, y ocurren demoras en la obtención de la información referente a las solicitudes.

No existe un alto nivel de seguridad, pues no se controla qué personas tienen acceso a la información, y el no tener registrado en bases de datos las solicitudes de matrícula y las asignaciones de carreras realizadas, ha causado pérdidas y duplicidad de la información.

Teniendo en cuenta lo anteriormente expuesto, el **problema científico** queda formulado de la siguiente manera: ¿Cómo mejorar los procesos de gestión de información vinculada a la fuerza de trabajo calificada en el Ministerio de Economía y Planificación para la elaboración del Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales?

El **objeto de estudio** de la presente investigación lo constituyen los procesos de gestión de la información, vinculados con la fuerza de trabajo calificada. Derivándose como **campo de acción** los procesos de gestión de la información en el Departamento de Fuerza de Trabajo Calificada del Ministerio de Economía y Planificación para la elaboración de los planes de ingreso a la Educación Superior en los Centros Universitarios Municipales

Se define como **objetivo** de la investigación: Realizar el análisis, diseño e implementación del Módulo Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales. De ahí se derivan los siguientes **objetivos específicos**:

- ✚ Desarrollar el marco teórico de la investigación.
- ✚ Definir el análisis y diseño del Módulo Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales.
- ✚ Elaborar el Módulo Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales para el sistema GeForza.
- ✚ Validar los resultados obtenidos.

Para guiar la investigación se plantea la siguiente **hipótesis**: Si se desarrolla el Módulo Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales se mejorarán los procesos de gestión de información vinculada a la fuerza de trabajo calificada en el Ministerio de Economía y Planificación, en cuanto a rapidez y organización.

**Variable independiente:** Módulo Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales.

**Variable dependiente:** Proceso de gestión de información vinculada a la fuerza de trabajo calificada.

Para lograr el cumplimiento de los objetivos se proponen las siguientes **tareas investigativas**:

- ✚ Estudio de los procesos de ingreso a los Centros Universitarios Municipales en Cuba.
- ✚ Estudio crítico de sistemas de planificación y gestión de la información, existentes en Cuba y en el mundo.

- ✚ Fundamentación del uso de metodologías, tecnologías, lenguajes y herramientas definidas para el desarrollo del módulo.
- ✚ Identificación de las funcionalidades que tendrá el módulo.
- ✚ Obtención de los artefactos que define la metodología de desarrollo de software seleccionada.
- ✚ Implementación de la solución propuesta.
- ✚ Selección de las pruebas a realizar.
- ✚ Evaluación de los resultados obtenidos.

### **Métodos de investigación utilizados:**

**Histórico-lógico:** En esta investigación dicho método se usó con el objetivo de estudiar todo lo referente a los procesos de planificación y gestión de información en el ámbito nacional e internacional.

**Análítico-sintético:** Este método se usó para organizar y sintetizar toda la información obtenida del estudio del estado del arte, las tecnologías, metodologías, lenguajes y herramientas propuestas para la solución e identificar los principales problemas de los sistemas actuales. La utilización de este método permitió comprender mejor toda la información anterior.

**Método de la modelación:** Se realizaron los modelos correspondientes al ciclo de vida del software, que ayudan a dar cumplimiento a las tareas de diseño de los procesos involucrados en la solución.

**Entrevista:** Se entrevistó a los compañeros del MEP obteniéndose la mayor cantidad de información sobre los problemas que presenta el proceso de solicitud de ingreso en los Centros Universitarios Municipales, las posibles soluciones a estos problemas y los procesos a automatizar por el Módulo Plan de Ingreso a la Educación Superior en los CUM.

### **El documento está estructurado en 3 capítulos:**

**Capítulo 1 Fundamentación teórica:** se muestra el resultado de la investigación bibliográfica sobre el objeto de estudio, así como las tendencias que adquieren actualmente los procesos de gestión de información. Se detallan las tecnologías, metodologías, lenguajes y herramientas utilizadas, analizando sus características, desventajas y ventajas. Se proponen las herramientas con las que se va a desarrollar

el software, para esto se realizará un estudio de las tendencias actuales de tecnologías web, luego se concluirá el capítulo, justificando las herramientas seleccionadas para la implementación de la aplicación.

**Capítulo 2 Descripción de la propuesta de solución:** se definen los procesos del negocio, se describen los requisitos funcionales y no funcionales que deberá cumplir el sistema, y a partir de la descripción de los requisitos funcionales se realiza el modelo de casos de uso del sistema, así como la descripción de los mismos.

**Capítulo 3 Construcción de la solución propuesta:** se modelan los diagramas de clases del análisis y del diseño. Se hace referencia a los patrones utilizados para la construcción del diseño. Se muestran los modelos lógico y físico de datos.

**Capítulo 4 Implementación y prueba:** Se presenta todo lo relacionado con la implementación del sistema, modelo de despliegue, modelo de implementación y los resultados de las pruebas realizadas. Además se realiza la validación de la solución propuesta.

## Capítulo 1: Fundamentación teórica

### 1.1 Introducción

En este capítulo se muestra el resultado de la investigación bibliográfica sobre el objeto de estudio, así como las tendencias que adquieren actualmente los procesos de gestión de información. Se detallan las tecnologías, metodologías, lenguajes y herramientas utilizadas, analizando sus características, desventajas y ventajas.

### 1.2 Estado del arte

El desarrollo de los sistemas gestores de información tiene gran auge a nivel mundial producto de la disponibilidad actual de las computadoras que ha generado un incremento de la información en la sociedad ya que, la información que se genera por computadoras difiere en diversas formas de la que se obtiene manualmente. A menudo, se tiene una mayor cantidad de información si esta se genera utilizando sistemas computacionales, debido a que la información generada por las computadoras puede llegar a multiplicarse a velocidades impresionantes.

#### 1.2.1 Conceptos básicos

**Gestión:** Es el proceso mediante el cual se obtiene, despliega o utiliza una variedad de recursos básicos para apoyar los objetivos de la información. (2)

**Información:** Elemento fundamental que manejan los ordenadores en forma de datos binarios. Factor cualitativo que designa la posición de un sistema, y que eventualmente es transmitido por este sistema a otro. (3)

**Sistema de información:** Es un conjunto integrado de procesos, principalmente formales, desarrollados en un entorno usuario-ordenador que, operando sobre un conjunto de datos estructurales (bases de datos) de una organización, recopilan, procesan y distribuyen selectivamente la información necesaria para la operatividad habitual de la organización y las actividades propias de la dirección de la misma. (4)



## 1.2.2 Los sistemas de gestión de información en el mundo

### Ecuador

#### Sistema de administración académica SAAC

Es una aplicación que automatiza todo el proceso de gestión académica de un centro educacional, mediante ágiles opciones que permiten registrar y compartir información de estudiantes, profesores, cursos, aulas y la planificación de cada periodo en ejecución.

SAAC permite descentralizar la gestión académica por unidades de servicio. Es decir, cada unidad académica puede asumir sus propios procesos de planificación académica, registros de estudiantes e ingreso de resultados del período (calificaciones, asistencia, convalidaciones, etc.). (5)

#### Características generales:

- ✚ **SAAC** es una moderna herramienta de información, diseñada con tecnología de orientación de objetos para ambientes cliente/servidor.
- ✚ Posee el manejador de la base de datos “DB2 versión 8.0”, que garantiza seguridad y consistencia de la información.
- ✚ Permite la definición de información personalizada, para habilitar las opciones del sistema de acuerdo con los procedimientos propios de cada unidad; por ejemplo: tipos de cobros, tipos de evaluaciones, etc.
- ✚ Posibilita la definición de parámetros generales de la institución.
- ✚ Permite la inclusión de políticas internas de acuerdo la reglamentación vigente en materia de sistema de evaluación.
- ✚ Posee un módulo de control de acceso a las aplicaciones, que permite definir usuarios y roles, con funciones y autorizaciones específicas.

#### Características técnicas:

- ✚ El sistema académico está conformado por una aplicación cliente/servidor de escritorio y una aplicación web.

- ✚ La aplicación de escritorio está desarrollada con "VisualAge for SmallTalk" de IBM, herramienta de desarrollo orientada a objetos, actualmente se utiliza la versión 5.0.
- ✚ La aplicación de escritorio está dirigida para cliente en los siguientes sistemas operativos: Windows 98, Windows 2000 Profesional y Windows Xp con plataforma i386.
- ✚ Requiere un servidor de base de datos, instalado con "DB2", versión 8.0 que tiene todas las ventajas de las base de datos más modernas.
- ✚ La aplicación web está desarrollada en la plataforma .NET de Microsoft utilizando el Visual Studio.NET.

### **Sistema Integrado de Planificación Estadística (SINPLE)**

Su funcionalidad es mejorar la coordinación del Sistema estadístico de Andalucía. Es un sistema basado en Workflow que permite agilizar los trámites de aprobación y seguimiento de los programas estadísticos; de hecho, la aplicación contempla el proceso completo de declaración de una actividad estadística a incluir en un Programa estadístico, con todos los actores implicados, y permite añadir detalles del seguimiento de su ejecución. Además, la utilización del SINPLE no queda acotada al período de vigencia de un Plan estadístico concreto, sino que permite la incorporación de un nuevo plan con sus diferentes estructuras y poder así implantar este sistema durante un período largo de tiempo.

Es un software orientado a procesos, cuenta con cuatro subsistemas en su desarrollo y usa Oracle como sistema gestor de base de datos. (6)

Entre las utilidades del sistema se puede resaltar las siguientes:

- ✚ La interconexión y coordinación entre el Instituto de Estadística de Andalucía y las Unidades Estadísticas de las Consejerías u Organismos Autónomos pertenecientes al Sistema Estadístico de Andalucía.
- ✚ La definición de nuevas actividades estadísticas y la elaboración de los planes y programas estadísticos anuales.
- ✚ El seguimiento de la ejecución de toda actividad estadística programada o planificada por parte de todo el Sistema Estadístico.

- ✚ La normalización de la información contenida en el Inventario de actividades del Sistema Estadístico de Andalucía.

## 1.2.2 Los sistemas de gestión de información en Cuba

### Akademos v1.0

Sistema de Gestión Académica que se utiliza en la Universidad de las Ciencias Informáticas. Cuenta con siete módulos:

- ✚ Matrícula: permite la obtención de estudiantes y estructura.
- ✚ Estudiante: permite a los estudiantes estar informados sobre su desempeño docente.
- ✚ Registro: permite llevar el control del desarrollo docente, notas y asistencia.
- ✚ Profesores: permite gestionar todo lo referente a los profesores del claustro.
- ✚ Plan de Estudio: permite la definición de los planes de estudio, asignatura, disciplina y perfiles.
- ✚ Reporte: permite el diseño y generación de reportes referentes a la información contenida en las hojas de matrícula y prematrícula.
- ✚ Seguridad: todo lo referente a la asignación de permisos.

Permite exportar los reportes a diferentes formatos e imprimirlos, permitiendo su configuración. Contiene un asistente para realizar reportes de evaluaciones ya sean de tipo resumen o listado. Permite enviar reportes por correo electrónico, así como la configuración de estos para que se envíen un día determinado. El administrador del módulo puede configurar permisos según el rol que tenga el usuario.

### FTCINGRE

Es el sistema de gestión de información del plan de ingreso a la Educación Media y Superior que se encuentra en explotación actualmente por el Departamento de Fuerza de Trabajo Calificada del Ministerio de Economía y Planificación. Fue desarrollado en Visual FoxPro 5.0, lenguaje de poco uso y que no ofrece facilidades de reutilización de código. Cuenta con módulos de reportes y de gestión de ingreso al curso regular diurno y al curso para trabajadores. Se encuentra prácticamente obsoleto ante toda esta problemática surgida con las novedades en la Educación media y Superior y en consecuencia en el plan de ingreso a la misma, con el desarrollo actual de las tecnologías de la informática y comunicaciones esta

aplicación no cumple con las políticas de seguridad mínimas para un sistema de esta magnitud. Todo el intercambio de información entre sistemas homólogos se realiza mediante dispositivos de almacenamiento, el más frecuente, el disco de 3½, esto conlleva a que la información sea muy vulnerable a ser atacada. Es poco manuable y difícil de operar. El sistema de reportes que ofrece no es lo suficientemente genérico y no satisface las necesidades del cliente. El software presenta diversos errores en su ejecución. (1)

## **CEDRUX**

CEDRUX es una solución de software creada en la Universidad de las Ciencias Informáticas para la completa gestión empresarial en cualquier organismo o industria del país. Fue desarrollado usando las últimas tecnologías para el desarrollo web: PHP, Zend Framework, Ext JS, Doctrine y Postgree SQL como gestor de base de datos. Es una herramienta que permite fácilmente ser configurada para cualquier empresa.

### **1.2.3 Ideas a utilizar**

Después de realizar un estudio de los diferentes sistemas de gestión de información utilizados en el mundo y en el país, se pudo apreciar que a pesar de las amplias posibilidades que ofrecen, en general fueron diseñados según las particularidades de los centros donde son aplicados y no se ajustan a lo que realmente se necesita para la gestión de información del plan de ingreso a la Educación Superior en los CUM, no obstante, brindan algunas ideas y datos para el desarrollo del mismo.

Por ejemplo: el sistema FTCINGRE a pesar de ser un sistema casi obsoleto, ofrece datos y funcionalidades del Plan de Ingreso a la Educación Media y Superior en Cuba, que son comunes para la elaboración del Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales. Estos datos son:

- Organismos.
- Grupos de carrera.
- Carreras.
- Provincias.

Además de la funcionalidad de crear los modelos de solicitud, lo cual proporciona ideas para elaborar los modelos del Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales.

Mientras que los sistemas SAAC y SINPLE aportan ideas en cuanto a la planificación y el control de la información, ya que SAAC es una moderna herramienta de información y SINPLE define nuevas actividades estadísticas y elabora los planes y programas estadísticos anuales.

También se puede utilizar ideas brindadas por el sistema Akademos, que en el Módulo Matrícula permite el control de los datos de los estudiantes que van a comenzar a ser parte de la Universidad, además del Módulo Profesor que planifica la carga docente de los profesores, este sistema también ofrece la posibilidad de imprimir, exportar a formato .xls, .pdf, .doc los datos del reporte generado y se desea que el sistema también le brinde al usuario la opción de imprimir, exportar a formato .xls y a .pdf el reporte obtenido.

### **1.3 Tendencias actuales en el desarrollo de software**

Software libre es la denominación del software que respeta la libertad de los usuarios sobre su producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, cambiado y redistribuido libremente; de modo más preciso, se refiere a cuatro libertades de los usuarios del software: la libertad de usar el programa, con cualquier propósito; de estudiar el funcionamiento del programa, y adaptarlo a las necesidades; de distribuir copias, con lo cual se puede ayudar a otros, y de mejorar el programa y hacer públicas las mejoras, de modo que toda la comunidad se beneficie (para la segunda y última libertad mencionadas, el acceso al código fuente es un requisito previo). (7)

Para escoger las herramientas, tecnologías y lenguajes a utilizar para el desarrollo del Módulo Plan de Ingreso al Educación Superior a los CUM, se tuvo en cuenta que el mismo pertenece al proyecto FTC que se rige por los lineamientos generales que son especificados por la Dirección de Informatización, la cual se encarga entre otras cosas, de estandarizar el uso de las tecnologías en los sistemas que son desarrollados. Por lo que a decisión de la dirección del proyecto, las herramientas, tecnologías y lenguajes que se utilizarán son las mencionadas a continuación en los siguientes epígrafes.

#### **Metodología de desarrollo de software**

Una metodología de desarrollo de software tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos del cliente. Las tendencias presentes, luego del

perfeccionamiento de los procesos del software durante años, han llevado a cabo dos corrientes significativas: los llamados métodos ligeros y métodos pesados. Aunque ambos están enfocados a beneficiar la labor de aquellas personas que intervienen en el proceso de desarrollo.

### **RUP (Proceso unificado de desarrollo de software)**

RUP (Proceso unificado de desarrollo de software) es una metodología pesada, es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

Esta metodología unifica los mejores elementos de las anteriores, por lo que se encuentra preparada para el desarrollo de grandes y complejos proyectos. RUP, representa un ideal de referencia para todo el equipo de desarrollo. Es además, un proceso de desarrollo de software que contiene un conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de software, se puede especializar para gran variedad de sistemas de software, distintas áreas de aplicación, tipos de organizaciones, niveles de actitud y tamaños de proyecto.

Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso).

Las fases de RUP concluyen con un hito bien definido, en cada uno de estos se deben tomar acuerdos y decisiones, garantizando el cumplimiento de los objetivos y metas antes de la transición a la nueva fase.

#### **1.3.1 Lenguaje de modelado**

El constante avance en el nivel de complejidad de las soluciones informáticas ha hecho de la utilización de los modelos, un mecanismo para facilitar la comprensión de estos. “Los modelos proporcionan un mayor nivel de abstracción, permitiendo trabajar con sistemas mayores y más complejos, y facilitando el proceso de codificación e implementación del sistema de forma distribuida y en distintas plataformas.” (8)

“Un modelo es una descripción de (parte de) un sistema, descrito en un lenguaje bien definido.” (8)

### **UML (Lenguaje Unificado de Modelado)**

UML “es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software”. (9) No define un proceso de desarrollo específico, tan solo se trata de una notación. “Permite modelar sistemas de información, y su objetivo es lograr modelos que, además de describir con cierto grado de formalismo tales sistemas, puedan ser entendidos por los clientes o usuarios de aquello que se modela”. (9) Uno de los fines principales de la creación de UML fue posibilitar el intercambio de modelos entre las distintas herramientas CASE orientadas a objetos del mercado, para ello era necesario definir una notación y semántica común, es decir, un estándar. Se logró con este lenguaje “un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto”. (9)

Algunas de las propiedades de UML como lenguaje de modelado son:

- ✚ Concurrencia, es un lenguaje distribuido y adecuado a las necesidades de conectividad actuales y futuras.
- ✚ Reemplaza a decenas de notaciones empleadas con otros lenguajes.
- ✚ Modela estructuras complejas.
- ✚ Las estructuras más importantes que soportan tienen su fundamento en las tecnologías orientadas a objetos, tales como objetos, clase, componentes y nodos.
- ✚ Emplea operaciones abstractas como guía para variaciones futuras, añadiendo variables si es necesario.
- ✚ Comportamiento del sistema: casos de uso, diagramas de secuencia y de colaboraciones, que sirven para evaluar el estado de las máquinas.

### 1.3.2 Herramienta CASE para el modelado

Las herramientas CASE para el modelado son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas sirven de apoyo ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes,

implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras. (10)

CASE proporciona al ingeniero la posibilidad de automatizar actividades manuales y de mejorar su visión general de la ingeniería [...] Las herramientas CASE ayudan a garantizar que la calidad se diseñe antes de llegar a construir el producto". (11)

### **Visual Paradigm**

Es un software privativo, soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación.

Ventajas de Visual Paradigm:

- ✚ Entorno de creación de diagramas para UML 2.0.
- ✚ Diseño centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad.
- ✚ Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- ✚ Capacidades de ingeniería directa e inversa.
- ✚ Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- ✚ Disponibilidad de múltiples versiones, para cada necesidad.
- ✚ Disponibilidad en múltiples plataformas.
- ✚ Producto de calidad.
- ✚ Soporta aplicaciones Web.
- ✚ Varios idiomas.
- ✚ Fácil de instalar y actualizar.
- ✚ Compatibilidad entre ediciones.

Además Visual Paradigm produce documentación del sistema en formato PDF, HTML y Word formatos de MS. template designer. Los desarrolladores pueden diseñar con documentación sobre el sistema y los



analistas de sistemas pueden estimar las consecuencias de los cambios con los diagramas de análisis de impacto, tales como la matriz y el diagrama de análisis.

### 1.3.3 Marcos de trabajo

En el desarrollo de software, un marco de trabajo es una estructura conceptual y tecnológica de soporte definida, normalmente con artefactos de software concretos, mediante la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros programas para ayudar a desarrollar y unir los diferentes componentes de un proyecto. Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo que extiende o utiliza las aplicaciones del dominio. (14)

#### Zend framework

Es un marco de trabajo para desarrollo de aplicaciones Web y servicios Web con PHP, brinda soluciones para construir sitios web modernos, robustos y seguros. Además, es Open Source y trabaja con PHP 5 a diferencia de CakePHP que trabaja con PHP 4 y PHP 5.

Está implementado utilizando código totalmente orientado a objetos. Su estructura de componentes es única; cada componente se diseña con pocas dependencias en otros componentes. Esta arquitectura débilmente acoplada permite que los desarrolladores utilicen componentes individualmente los cuales forman una potente y extensible aplicación web al ser combinados, todas estas ventajas que brinda y las demás que a continuación se mencionan hacen que dicho *framework* sea el más adecuado para programar la capa de negocio del sistema.

Ventajas:

- ✚ Acoplable a variadas tecnologías de desarrollo (adición de múltiples *plugins* externos).
- ✚ Cuenta con módulos para manejar archivos PDF, canales RSS, Servicios Web.
- ✚ Completa documentación y pruebas de alta calidad.
- ✚ Un buscador compatible con Lucene.
- ✚ Robustas clases para autenticación y filtrado de entrada.
- ✚ Clientes para servicios web, incluidos Google Data APIs y Strikelron.
- ✚ Marco de trabajo oficial de la comunidad PHP.

✚ Orientado a componentes.

## **EXTJS**

Es un *framework* para JavaScript muy utilizado en el desarrollo de aplicaciones Web con AJAX. Tiene una librería inmensa que permite configurar las interfaces Web de manera semejante a aplicaciones *desktop*. Tiene incluidos la mayoría de los controles de los formularios Web incluyendo Grids para mostrar datos y elementos semejantes a la programación *desktop* como los formularios, paneles, barras de herramientas, menús y muchos otros. Dentro de su librería de componentes incluye componentes para el manejo de datos, lectura de XML, lectura de datos JSON e implementaciones basadas en AJAX. Presenta el uso de JavaScript con una programación orientada a objetos. (15)

## **Doctrine**

Doctrine es un potente y completo sistema Object Relational Mapper (ORM) para PHP 5.2 o superior, con un Database Abstraction layer (DBAL) incorporado. Tiene la posibilidad de exportar una base de datos existente a sus clases correspondientes y también a la inversa, es decir, convertir clases (convenientemente creadas siguiendo las pautas del ORM) a tablas de una base de datos. Su principal ventaja radica en poder acceder a la base de datos utilizando la Programación Orientada a Objetos (POO) debido a que Doctrine utiliza el patrón Active Record para manejar la base de datos, tiene su propio lenguaje de consultas y trabaja de manera rápida y eficiente. Es fácilmente integrable a los principales marcos de trabajo de desarrollos utilizados actualmente. (15)

### **1.3.4 Lenguajes para el desarrollo Web**

#### **PHP(Procesador de hipertexto)**

PHP es un lenguaje de programación interpretado, el cual fue diseñado originalmente para la creación de páginas web dinámicas aunque también ha intervenido en la creación de aplicaciones con interfaces gráficas usando bibliotecas específicas. Es capaz de combinarse con servidores de bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de aplicaciones Web robustas. La ejecución e interpretación de PHP es completamente en el servidor Web: en este se encuentra almacenado el script por lo que el cliente sólo puede recibir el resultado de la ejecución. PHP puede ser ejecutado en la mayoría de los sistemas operativos, tales como

UNIX, Linux, Mac OS X y Windows, a la vez es capaz de interactuar con servidores Web muy populares como ISAPI y Apache por existir también en versión CGI. (16)

El uso de PHP presenta ventajas como:

- ✚ Es capaz de leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
- ✚ Cuenta con una biblioteca sumamente amplia por defecto de funciones.
- ✚ No requiere definición de tipos de variables ni manejo detallado del bajo nivel.
- ✚ Tiene capacidad de conexión con la mayoría de los manejadores de base de datos utilizados en la actualidad.
- ✚ Posee una amplia documentación en su Web oficial de Internet. En la misma se encuentran muy bien explicadas todas las funciones del sistema, contando con ejemplos detallados.
- ✚ Pertenece a la alternativa de código abierto (Open Source), por lo que se presenta como una elección de fácil acceso para todos.
- ✚ Permite las técnicas de Programación Orientada a Objetos.
- ✚ Permite crear formularios para la Web.

## PHP 5

Con las primeras 2 versiones de PHP, PHP 3 y PHP 4, se había conseguido una plataforma potente y estable para la programación de páginas del lado del servidor. Estas versiones han servido de mucha ayuda para la comunidad de desarrolladores, haciendo posible que PHP sea el lenguaje más utilizado en la web para la realización de páginas avanzadas.

El 13 de julio de 2004, fue lanzado PHP 5, utilizando el motor **Zend Engine II** (o Zend Engine 2). La versión más reciente de PHP es la 5.2.6 (1 de mayo de 2008), que incluye todas las ventajas que provee el nuevo **Zend Engine 2** como:

Mejor soporte para la Programación orientada a objetos, que en versiones anteriores era extremadamente rudimentario, con PHP Data Objects.

- ✚ Mejoras de rendimiento.

- ✚ Mejor soporte para MySQL con extensión completamente reescrita.
- ✚ Mejor soporte a XML (XPath, DOM, etc.).
- ✚ Soporte nativo para SQLite.
- ✚ Soporte integrado para SOAP.
- ✚ Integradores de datos.

## HTML

El HTML, *Hyper Text Markup Language* (Lenguaje de marcación de hipertexto) es el lenguaje de marcas de texto utilizado normalmente en la *www* (*World Wide Web*). Fue creado en 1986 por el físico nuclear Tim Berners-Lee; el cual tomó dos herramientas preexistentes: el concepto de Hipertexto (Conocido también como *link* o ancla) el cual permite conectar dos elementos entre sí y el SGML (Lenguaje estándar de marcación general) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse. HTML no es propiamente un lenguaje de programación como C++, Visual Basic, etc., sino un sistema de etiquetas. HTML no presenta ningún compilador, por lo tanto, algún error de sintaxis que se presente este no lo detectará y se visualizará en la forma como este lo entienda. El entorno para trabajar HTML es simplemente un procesador de texto, como el que ofrecen los sistemas operativos Windows (Bloc de notas), UNIX o el que ofrece MS Office (Word). El conjunto de etiquetas que se creen, se deben guardar con la extensión *.htm*.

Estos documentos pueden ser mostrados por los visores o "*browsers*" de páginas web en Internet, como Netscape Navigator, Mosaic, Opera y Microsoft Internet Explorer.

También existe el HTML Dinámico (DHTML), que es una mejora de Microsoft de la versión 4.0 de HTML que le permite crear efectos especiales como, por ejemplo, texto que vuela desde la página palabra por palabra o efectos de transición al estilo de anuncio publicitario giratorio entre página y página. (17)

## JavaScript

JavaScript es un lenguaje de *scripting* basado en objetos, utilizado para acceder a objetos en aplicaciones, se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que

soporta la carga de procesamiento. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado.

Con Javascript podemos crear efectos especiales en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones Javascript y ejecutarlas para realizar estos efectos e interactividades, de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador. (18)

### 1.3.5 Servidor Web

El Servidor Web es el programa encargado de gestionar las peticiones de los usuarios que visitan su página web. Es el programa encargado de mostrar sus páginas cuando un visitante realiza una petición en su dominio.

#### Servidor Web Apache

Apache es considerado el Servidor Web por excelencia, no sólo por su gran aceptación, puesto que casi el 70% de los servidores Internet confía en él, sino porque desde su nacimiento ha demostrado con creces su estabilidad, solidez y su mayor rendimiento ante sus competidores.

Principales ventajas:

- ✚ **Fiabilidad:** Alrededor del 90% de los servidores con más alta disponibilidad funcionan bajo servidores Apache.
- ✚ **Software Libre:** Apache es gratuito, y se distribuye bajo la licencia *Apache Software License*, que permite la modificación del código.
- ✚ **Extensibilidad:** se pueden añadir módulos para ampliar las capacidades de Apache. También dispone de una amplia variedad de módulos, que permiten desde generar contenido dinámico (con PHP, Java, Perl) monitorizar el rendimiento del servidor, crear servidores virtuales por IP o por nombre (varias direcciones web son manejadas en un mismo servidor) y limitar el ancho de banda para cada uno de ellos. Dichos módulos están disponibles junto con su código fuente, por lo cual pueden ser incluso modificados por cualquier persona con conocimientos de programación

### 1.3.6 Sistemas Gestores de base de datos

Un Sistema Gestor de base de datos (SGBD) es un conjunto de programas que permiten crear y mantener una Base de datos, asegurando su integridad, confidencialidad y seguridad. Por tanto, debe permitir:

- ✚ Definir una base de datos: especificar tipos, estructuras y restricciones de datos.
- ✚ Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD
- ✚ Manipular la base de datos: realizar consultas, actualizarla, generar informes.(19)

### PostgreSQL

PostgreSQL es un sistema de gestión de base de datos relacional orientada a objetos y libre, publicado bajo la licencia BSD. Posee las características de los más potentes sistemas comerciales como Oracle o SQL Server, con la ventaja de que su licencia es gratuita. Como muchos otros proyectos open *source*, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales. Dicha comunidad es denominada *PostgreSQL Global Development Group* (PGDG).

Ventajas:

- ✚ Alta concurrencia.
- ✚ Amplia variedad de tipos nativos.
- ✚ Multiplataforma.
- ✚ Extensible.
- ✚ Estabilidad y Confiabilidad.
- ✚ Instalación ilimitada.
- ✚ Diseñado para ambientes de alto volumen.

### 1.3.7 Entorno Integrado de Desarrollo (IDE)

Un Entorno de Integrado Desarrollo (IDE de su significado en inglés *Integrated Development Environment*), es un conjunto de herramientas que ha sido creado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Un

IDE provee un marco de trabajo amigable para la mayoría de los lenguajes de programación y en algunos casos puede funcionar como un sistema en tiempo de ejecución en donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto. (20)

### **NetBeans IDE**

Es una herramienta para programadores de código abierto escrito en Java usando la plataforma NetBeans. Soporta el desarrollo de todos los tipos de aplicación Java, tales como: JEE (*Java Platform, Enterprise Edition*), Web, EJB (*Enterprise JavaBeans*) y aplicaciones móviles. Las funciones de este IDE son provistas por módulos donde cada uno provee una función bien definida: soporte de Java, edición o soporte para el sistema de control de versiones. NetBeans contiene todos los módulos necesarios para el desarrollo de aplicaciones Java en una sola descarga, facilitándole al usuario comenzar a trabajar inmediatamente. La plataforma NetBeans es una base modular y extensible usada como una estructura de integración para crear aplicaciones de escritorio grandes. Ofrece servicios comunes permitiéndole al desarrollador enfocarse en la lógica específica de su aplicación. Entre las características de la plataforma se encuentran:

- ✚ Administración de las interfaces de usuario.
- ✚ Administración de las configuraciones del usuario.
- ✚ Administración de almacenamiento.
- ✚ Administración de ventanas.
- ✚ Marco de trabajo basado en asistentes.

### **1.4 Conclusiones**

En este capítulo se realizó una amplia investigación sobre el estado del arte de los sistemas de gestión de información, tanto en el ámbito nacional como internacional, llegando a la conclusión de que aunque los sistemas estudiados no se pueden utilizar para resolver el problema por no adaptarse a los requerimientos del mismo, sirven para aportar ideas de funcionalidades para el nuevo sistema a implementar. Se realizó un análisis de las potencialidades de las tecnologías, lenguajes, metodologías y herramientas definidas por la dirección del proyecto para ser utilizadas en el desarrollo de la aplicación, definiendo: PHP; como lenguaje de modelado: UML; como sistema gestor de bases de datos: PostgreSQL; como metodología de

desarrollo: RUP; como herramienta de modelado: Visual Paradigm; como marco de trabajo para el desarrollo: Ext JS y Zend framework y como ORM: Doctrine.



## **Capítulo 2: Descripción de la propuesta de solución**

### **2.1 Introducción**

En el presente capítulo se describen los procesos del negocio que tiene que ver con el objeto de estudio, se modela el negocio propuesto, se identifican los actores, trabajadores, los casos de uso correspondientes y la descripción de los mismos. Se especifican las reglas del negocio. También se enumeran los requisitos; tanto funcionales como no funcionales que debe cumplir el sistema que se propone.

### **2.2 Modelo de negocio**

Para un mayor y mejor entendimiento de cómo se lleva actualmente a cabo la gestión del plan de ingreso en los Centros Universitarios Municipales, es mejor hacer uso de las técnicas de modelado que propone UML. El paso más importante dentro de dicho modelado es la identificación de los procesos del negocio. Dada la estructuración del negocio que se está estudiando se propone un Modelo del Negocio ya que este permite de forma visual mostrar a los usuarios los principales conceptos que se manejan. Esto se hace con el objetivo de ayudar a los usuarios, clientes, desarrolladores y otros a tener un mejor entendimiento de los procesos y, a utilizar un vocabulario común entre todos para un mejor entendimiento del contexto en que se pone el sistema.

### **2.3 Descripción del negocio**

Para realizar el plan de ingreso en los Centros Universitarios Municipales, las fuentes de ingreso, que son las entidades y los programas de la Revolución, solicitan mediante el modelo de solicitud MODELO 1. FTC-CUM, las carreras y las plazas que desean. Esta solicitud es enviada a la comisión municipal para realizar la propuesta de plan de carreras de un municipio. Para esto el municipio en cuestión realiza una reunión, la cual tiene como objetivo elevar al nivel provincial una solicitud en la cual se especifican las carreras que proponen que brinde el Centro Universitario Municipal de su municipio así como la cantidad que proponen por cada carrera. Como las carreras y la cantidad propuesta deben estar basadas en la demanda de las mismas, para esta reunión se utilizan las “Solicitudes de carreras de las entidades y programas de la Revolución” del municipio. Estas solicitudes brindan información sobre la demanda del municipio, y basado en la misma, se eleva la propuesta a la provincia, aunque el reporte de demanda no es absoluto, ya que la solicitud elevada a la provincia puede tener carreras no propuestas por las

solicitudes realizadas por las entidades, así como no necesariamente deben aparecer las carreras demandadas, ya que la solicitud elevada debe estar basada tanto en la demanda con en la necesidad del municipio en cuestión. Después de que ocurra la reunión municipal y con el resultado de la misma, se crea la “Propuesta de plan de carreras de un municipio”, que muestra los datos que fueron aprobados como resultado de la reunión celebrada en el municipio.

Después se hace la propuesta de carreras a brindar en la provincia, distribuidas por sus respectivos Centros Universitarios Municipales. Para esto la provincia realiza una reunión, la cual tiene como objetivo elevar al nivel nacional una solicitud en la cual se especifican las carreras que proponen que brinden los Centros Universitarios Municipales de la provincia, así como la cantidad que proponen por cada carrera. Como las carreras y la cantidad propuesta deben estar basadas en la demanda de las mismas, para esta reunión se utiliza la “Propuesta de plan de carreras de todos los municipios” de la provincia. Por cada reporte municipal propondrán las carreras que deberá proveer los Centros Universitarios Municipales de cada municipio, al igual que en la reunión municipal, a cada Centro Universitario Municipal se le podrá agregar carreras que no habían sido solicitadas, así como rechazar algunas de las propuestas y especificar el número exacto de cada una de estas por cada Centro Universitario Municipal. Luego de que ocurra la reunión provincial se crea la “Propuesta de plan de carreras de una provincia”, que muestra los datos que fueron aprobados como resultado de la reunión celebrada en la provincia. Luego se pasa hacer la propuesta del plan de carreras de los Centros Universitarios Municipales del país: Para esto el Departamento Fuerza de Trabajo Calificada se reúne y utiliza la “Propuesta de plan de carreras de cada provincia”.

## **2.4 Reglas del negocio**

Para garantizar las restricciones que existen en el negocio se debe tener en cuenta, a la hora de automatizar los procesos del plan de ingreso a los CUM las siguientes reglas:

- Para realizar la propuesta del plan de ingreso a nivel municipal, las entidades y los programas de la Revolución deben haber llenado y enviado a la comisión municipal los modelos de solicitudes.
- Para realizar la propuesta del plan de ingreso a nivel provincial, se debe de haber creado la propuesta del plan de ingreso a nivel municipal.

- Para realizar el plan de ingreso a nivel nacional, se debe de haber creado la propuesta del plan de ingreso a nivel provincial.

## 2.5 Actores del negocio

Actor	Descripción
Entidad	Representa la entidad o el programa de la Revolución que solicita las carreras.
Comisión Municipal	Representa la comisión municipal encargada de revisar las solicitudes de plazas de las entidades o programas de la Revolución y crear la propuesta del plan de ingreso municipal.
Comisión Provincial	Representa la comisión provincial encargada de revisar las propuestas del plan de ingreso municipal y crear la propuesta del plan de ingreso provincial.

Tabla # 1: Actores del Negocio.

## 2.6 Trabajadores del negocio

Trabajador	Descripción
Administrador de solicitudes	Es el encargado de llenar los datos de las solicitudes de matrícula.
Funcionario de la comisión municipal	Es el encargado de archivar la propuesta del plan de ingreso municipal.
Funcionario de la comisión provincial	Es el encargado de archivar la propuesta del plan de ingreso provincial.
Departamento de Fuerza de Trabajo Calificada (FTC)	Es el encargado de archivar el plan de ingreso nacional.

Tabla # 2: Trabajadores del negocio.

## 2.7 Diagrama de casos de uso del negocio

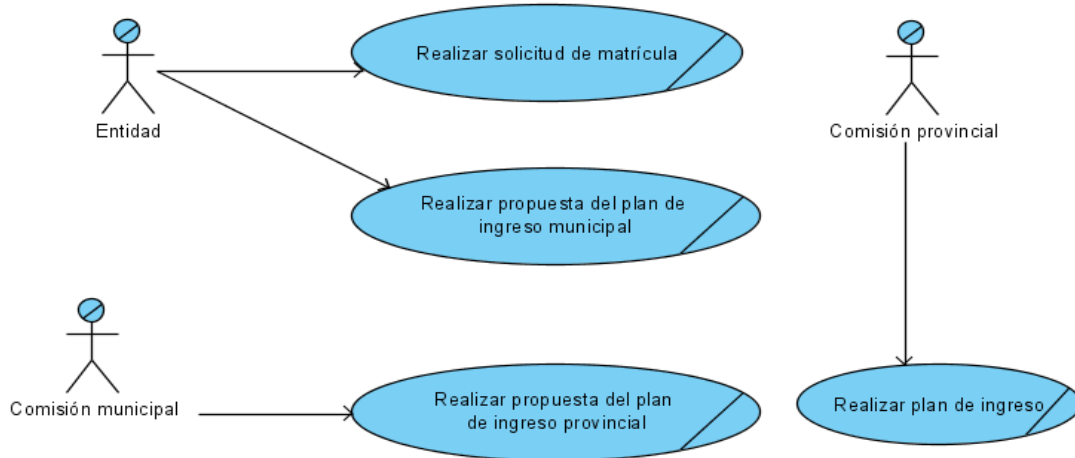


Figura # 1: Diagrama de casos de uso del negocio.

## 2.8 Diagramas de Actividades de los casos de uso del negocio

A continuación se muestra el Diagrama de Actividades del caso de uso Realizar plan de ingreso. En el Anexo # 3 se pueden ver los Diagramas de Actividades de los restantes casos de uso.

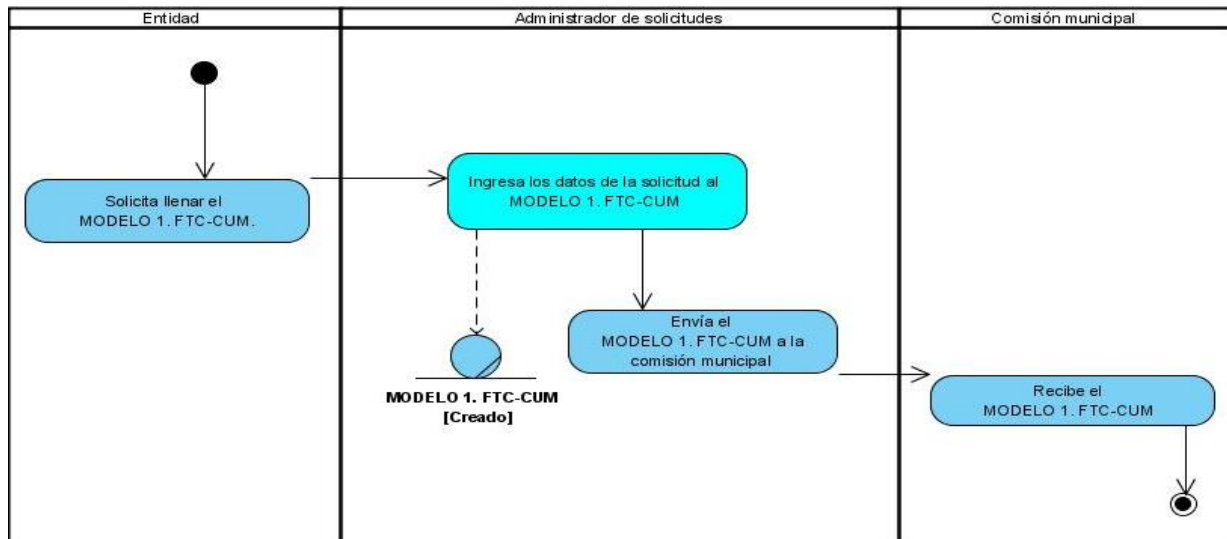


Figura # 2: Diagrama de Actividades del negocio del caso de uso Realizar solicitud de matrícula.

## 2.9 Descripción de los casos de uso del negocio

CUN: Realizar solicitud de matrícula

<b>Caso de Uso:</b>	Realizar solicitud de matrícula
<b>Actores:</b>	Entidad
<b>Trabajador:</b>	Administrador de solicitudes
<b>Resumen:</b>	El caso de uso se inicia cuando la entidad solicita la matrícula para estudiar en los CUM, se dirige al encargado de realizar las solicitudes de matrícula y este llena el Modelo VIII-1 FTC- SUM para solicitar las carreras. Luego le hace llegar estas solicitudes a la comisión municipal, terminado así el caso de uso.
<b>Precondiciones:</b>	

### Flujo Normal de Eventos

Acción del Actor	Respuesta del Negocio
1. La Entidad solicita llenar el MODELO 1. FTC-CUM.	2. El Administrador de solicitudes ingresa los siguientes datos de la solicitud al MODELO 1. FTC-CUM: <ul style="list-style-type: none"> <li>• Entidad o programa.</li> <li>• Organismo al que pertenece.</li> <li>• Provincia en la que se encuentra la entidad o el programa.</li> <li>• Municipio en la que se encuentra la entidad o el programa.</li> <li>• CUM en la cual desea que estudien los solicitantes de su entidad o programa de la Revolución</li> <li>• Carreras.</li> <li>• Cantidad</li> <li>• Fecha de creación de la solicitud.</li> </ul> 3. Envía el MODELO 1. FTC-CUM a la comisión municipal .
Flujo Alternativo 1	
Acción del Actor	Respuesta del Negocio

<b>Pos condiciones:</b>	Queda creado el Modelo VIII-1 FTC- CUM.
-------------------------	---

Tabla # 3: Descripción del CUN Realizar solicitud.

Las demás descripciones se pueden encontrar en el Anexo # 4.

## 2.10 Modelo de objetos

Son un caso especial de un diagrama de clases en el que se muestran instancias específicas de clases (objetos) en un momento particular del sistema. Los diagramas de objetos no muestran la multiplicidad ni los roles, aunque su notación es similar a los diagramas de clase.

En el Anexo # 2 se puede encontrar el diagrama de objetos del negocio.

## 2.11 Actores del sistema

Se le llama actor a toda entidad externa al sistema que guarda una relación con este y que le demanda una funcionalidad. Esto incluye a los operadores humanos pero también incluye a todos los sistemas externos así como a entidades abstractas. En el caso de los seres humanos se pueden ver a los actores como definiciones de rol, por lo que un mismo individuo puede corresponder a uno o más actores.

Actor	Descripción
Administrador de solicitudes	Representa al usuario responsable de gestionar la información del MODELO 1.FTC-CUM.
Funcionario municipal	Representa al usuario responsable de gestionar la información de la propuesta de plan de ingreso municipal.
Funcionario provincial	Representa al usuario responsable de gestionar la información de la propuesta del plan de ingreso provincial.
Funcionario del Dpto. FTC	Representa al usuario responsable de gestionar la información del plan de ingreso.
Funcionario	Representa una generalización de los

	funcionarios provinciales y municipales y del Dpt. FTC .
--	--

Tabla # 4: Actores del sistema.

## 2.12 Especificación de los Requerimientos de Software

El análisis de requerimientos permite al ingeniero refinar la asignación de software y representa el dominio de la información que será tratada por el programa. El análisis de requerimientos brinda al diseñador la representación de la información y las funciones que pueden ser traducidas en datos, arquitectura y diseño procedimental.

### 2.12.1 Requerimientos Funcionales

**Para el Administrador de solicitudes.**

#### **RF1. Gestionar MODELO 1.FTC-CUM:**

1.1 Mostrar todos los modelos creados hasta el momento con sus datos:

- Organismo.
- Entidad
- Programa.
- Provincia de la entidad o programa.
- Municipio de la entidad o programa.
- CUM donde estudiará el trabajador.

1.2 Mostrar las opciones:

- Adicionar.
- Modificar.
- Eliminar.
- Búsqueda avanzada.
- Limpiar búsqueda.
- Exportar.

1.3 Adicionar MODELO 1.FTC-CUM.

1.3.1 Introducir los siguientes datos:

- DPA de la entidad o programa.

- Municipio de la entidad o programa.
- Organismo.
- Entidad o programa.
- Municipio de la CUM.
- Código.
- Carrera.
- Cantidad de plazas.

1.3.1.1 Mostrar las opciones de:

- Agregar fila.
- Eliminar fila (sólo se muestra si existe alguna fila).
- Cancelar.
- Aceptar.
- Aplicar.

1.4 Modificar MODELO 1.FTC-CUM.

1.4.1 Seleccionar un modelo.

1.4.2 Modificar los siguientes datos del modelo seleccionado:

- Carrera.
- Cantidad de plazas.

1.4.2.1 Mostrar las opciones de:

- Agregar fila.
- Eliminar fila (sólo se muestra si existe alguna fila).
- Cancelar.
- Aceptar.
- Aplicar.

1.5 Eliminar MODELO 1.FTC-CUM.

1.5.1 Seleccionar un modelo.

1.5.2 Mostrar un mensaje de confirmación preguntando si está seguro de eliminar el modelo que está seleccionado en ese momento, además de las opciones “Aceptar” y “Cancelar”.

1.5.3 Eliminar el modelo.

1.6 Buscar modelo.



1.6.1 Introducir los datos:

- DPA de la entidad o programa.
- Municipio de la entidad o programa.
- Organismo Superior.

1.6.1.1 Mostrar las opciones para seleccionar:

- Aceptar.
- Cancelar.

1.7 Limpiar búsqueda.

1.7.1 Limpiar todos los campos.

1.8 Exportar.

1.8.1 Seleccionar modelo.

1.8.2 Seleccionar el formato en que se va a exportar:

- Excel.
- PDF.

**Para el Funcionario municipal.**

**RF2. Gestionar propuesta del plan de ingreso municipal.**

2.1 Mostrar todas las propuestas creadas según la:

- Provincia.
- CUM.

2.2 Mostar las opciones:

- Adicionar.
- Modificar.
- Eliminar.
- Búsqueda avanzada.
- Limpiar búsqueda.
- Exportar.

2.3 Adicionar propuesta del plan de ingreso municipal.

2.3.1 Introducir los siguientes datos:

- Provincia.
- CUM.

- Carrera.
- Solicitud de plazas.
- Tipo.

2.3.1.1 Mostrar las opciones de:

- Cargar demanda.
- Agregar fila.
- Eliminar fila (sólo se muestra si existe alguna fila).
- Cancelar.
- Aceptar.
- Aplicar.

2.4 Modificar propuesta del plan de ingreso municipal.

2.4.1 Seleccionar la propuesta del plan de ingreso municipal.

2.4.2 Modificar los siguientes datos:

- Carrera.
- Código.
- Solicitud de plazas.
- Tipo.

2.4.2.1 Mostrar las opciones de:

- Agregar fila.
- Eliminar fila (sólo se muestra si existe alguna fila).
- Cancelar.
- Aceptar.
- Aplicar.

2.5 Eliminar propuesta del plan de ingreso municipal.

2.5.1 Seleccionar la propuesta del plan de ingreso municipal.

2.5.2 Mostrar un mensaje de confirmación preguntando si está seguro de que desea eliminar la propuesta que está seleccionada en ese momento, además de las opciones “Aceptar” y “Cancelar”.

2.5.3 Eliminar la propuesta del plan de ingreso municipal.

2.6 Buscar propuesta municipal.

2.6.1 Introducir los datos:

- Provincia.
- CUM.

2.6.1.1 Mostrar las opciones para seleccionar:

- Aceptar.
- Cancelar.

2.7 Limpiar búsqueda.

2.7.1 Limpiar todos los campos.

2.8 Exportar.

2.8.1 Seleccionar modelo.

2.8.2 Seleccionar el formato en que se va a exportar:

- Excel.
- PDF.

**Para el Funcionario provincial.**

**RF3. Gestionar propuesta del plan de ingreso provincial.**

3.1 Mostrar todas las propuestas creadas según la:

- Provincia.
- CUM.

3.2 Mostar las opciones:

- Adicionar.
- Modificar.
- Eliminar.
- Búsqueda avanzada.
- Limpiar búsqueda.
- Exportar.

3.3 Adicionar propuesta del plan de ingreso provincial.

3.3.1 Introducir los siguientes datos:

- Provincia.
- CUM.
- Carrera.

- Solicitud de plazas.
- Tipo.

3.3.1.1 Mostrar las opciones de:

- Cargar plan municipal.
- Agregar fila.
- Eliminar fila (sólo se muestra si existe alguna fila).
- Cancelar.
- Aceptar.
- Aplicar.

3.4 Modificar propuesta del plan de ingreso provincial.

3.4.1 Seleccionar la propuesta del plan de ingreso provincial.

3.4.2 Modificar los siguientes datos:

- Carrera.
- Código.
- Solicitud de plazas.
- Tipo.

3.4.2.1 Mostrar las opciones de:

- Agregar fila.
- Eliminar fila (sólo se muestra si existe alguna fila).
- Cancelar.
- Aceptar.
- Aplicar.

3.5 Eliminar propuesta del plan de ingreso provincial.

3.5.1 Seleccionar la propuesta del plan de ingreso provincial.

3.5.2 Mostrar un mensaje de confirmación preguntando si está seguro de que desea eliminar la propuesta que está seleccionada en ese momento, además de las opciones “Aceptar” y “Cancelar”.

3.5.3 Eliminar la propuesta del plan de ingreso provincial.

3.6 Buscar propuesta provincial.

3.6.1 Introducir los datos:

- Provincia.
- CUM.

3.6.1.1 Mostrar las opciones para seleccionar:

- Aceptar.
- Cancelar.

3.7 Limpiar búsqueda.

3.7.1 Limpiar todos los campos.

3.8 Exportar.

3.8.1 Seleccionar modelo.

3.8.2 Seleccionar el formato en que se va a exportar:

- Excel.
- PDF.

**Para el funcionario del Dpto. FTC.**

**RF4. Gestionar plan de ingreso.**

4.1 Mostrar todas las propuestas creadas según la:

- Provincia.
- CUM.

4.2 Mostar las opciones:

- Adicionar.
- Modificar.
- Eliminar.
- Búsqueda avanzada.
- Limpiar búsqueda.
- Exportar.

4.3 Adicionar plan de ingreso.

4.3.1 Introducir los siguientes datos:

- Provincia.
- CUM.
- Carrera.
- Solicitud de plazas.

- Tipo.

#### 4.3.1.1 Mostrar las opciones de:

- Cargar plan provincial.
- Agregar fila.
- Eliminar fila (sólo se muestra si existe alguna fila).
- Cancelar.
- Aceptar.
- Aplicar.

#### 4.4 Modificar plan de ingreso.

##### 4.4.1 Seleccionar plan de ingreso.

##### 4.4.2 Modificar los siguientes datos:

- Carrera.
- Código.
- Solicitud de plazas.
- Tipo.

##### 4.4.2.1 Mostrar las opciones de:

- Agregar fila.
- Eliminar fila (sólo se muestra si existe alguna fila).
- Cancelar.
- Aceptar.
- Aplicar.

#### 4.5 Eliminar plan de ingreso.

##### 4.5.1 Seleccionar plan de ingreso.

4.5.2 Mostrar un mensaje de confirmación preguntando si está seguro de que desea eliminar el plan de ingreso que está seleccionado en ese momento, además de las opciones “Aceptar” y “Cancelar”.

##### 4.5.3 Eliminar el plan de ingreso.

#### 4.6 Buscar plan de ingreso.

##### 4.6.1 Introducir los datos:

- Provincia.

- CUM.

4.6.1.1 Mostrar las opciones para seleccionar:

- Aceptar.
- Cancelar.

4.7 Limpiar búsqueda.

4.7.1 Limpiar todos los campos.

4.8 Exportar.

4.8.1 Seleccionar modelo.

4.8.2 Seleccionar el formato en que se va a exportar:

- Excel.
- PDF.

**Para el funcionario de las comisiones municipales, provinciales y el Dpto. FTC.**

**RF5. Generar reporte “Listado de entidades que solicitan CUM”.**

5.1 Seleccionar Visor de reportes.

5.1.1 Mostrar una ventana con las siguientes opciones:

- Reportes.
- Formato del reporte.

5.1.2 Seleccionar ver el reporte “Listado de entidades que solicitan CUM” en los siguientes formatos.

- HTML.
- PDF.
- Excel.

5.1.3 Generar reporte “Listado de entidades que solicitan CUM” en el formato que se seleccionó con los siguientes datos:

- Organismo.
- Provincia.
- Entidad o programa.

5.2 Filtrar reporte “Listado de entidades que solicitan CUM”.

5.2.1 Mostrar las opciones:

5.2.1.1 Filtrar reporte:

- CUM\_nombre.
- Año.
- Organismo.
- Provincia.
- Municipio.
- Entidad.
- Demanda.

5.2.1.2 Tipo de filtrado:

- Por todos los filtros.
- Por al menos un filtro.

**RF6. Generar reporte “Demanda CUM agrupada por provincias”.**

6.1 Seleccionar Visor de reportes.

6.1.1 Mostrar las siguientes opciones:

- Reportes.
- Formato del reporte.

6.1.2 Seleccionar ver el reporte “Demanda CUM agrupada por provincias”.

6.1.3 Selecciona el formato en que se desea ver el reporte:

- HTML.
- PDF.
- Excel.

6.1.4 Generar reporte “Modelos de solicitudes” en el formato que se seleccionó con los siguientes datos:

- CUM\_nombre.
- OACE-CAP.
- Programa priorizado.
- Provincia.
- Municipio.
- Demanda.

6.2 Filtrar reporte “Demanda CUM agrupada por provincias”.

6.2.1 Mostrar las opciones:



6.2.1.1 Filtrar reporte:

- CUM\_nombre.
- Año.
- Organismo.
- Provincia.
- Municipio.
- Entidad.
- Demanda.

6.2.1.2 Tipo de filtrado:

- Por todos los filtros.
- Por al menos un filtro.

**RF7. Generar reporte “Demanda CUM agrupado por grupo de carreras”.**

7.1 Seleccionar Visor de reportes.

7.1.1 Mostrar las siguientes opciones:

- Reportes.
- Formato del reporte.

7.1.2 Seleccionar ver el reporte “Demanda CUM agrupado por grupo de carreras”.

7.1.3 Seleccionar el formato en que se desea ver el reporte:

- HTML.
- PDF.
- Excel.

7.1.4 Generar reporte “Demanda CUM agrupada por grupo de carreras” en el formato que se seleccionó con los siguientes datos:

- Provincia.
- Municipio.
- Grupo de carreras.
- Carrera.
- Demanda.
- Total.

7.2 Filtrar reporte “Demanda CUM agrupado por grupo de carreras”.

7.2.1 Mostrar las opciones:

7.2.1.1 Filtrar reporte:

- CUM\_nombre.
- Año.
- Organismo.
- Provincia.
- Municipio.
- Entidad.
- Demanda.

7.2.1.2 Tipo de filtrado:

- Por todos los filtros.
- Por al menos un filtro.

**RF8. Generar reporte “Plan CUM nivel MEP agrupado por grupo de carreras”.**

8.1 Seleccionar Visor de reportes.

8.1.1 Mostrar siguientes opciones:

- Reportes.
- Formato del reporte.

8.1.2 Seleccionar ver el reporte “Plan CUM nivel MEP agrupado por grupo de carreras”.

8.1.3 Seleccionar el formato en que se desea ver el reporte:

- HTML.
- PDF.
- Excel.

8.1.4 Generar reporte “Plan CUM nivel MEP agrupado por grupo de carreras” en el formato que se seleccionó con los siguientes datos:

- Provincia.
- Municipios.
- Grupo de carreras.
- Cantidad de plazas asignadas.

8.2 Filtrar reporte “Plan CUM nivel MEP agrupado por grupo de carreras”.

8.2.1 Mostrar las opciones:

8.2.1.1 Filtrar reporte:

- CUM\_nombre.
- Año.
- Organismo.
- Provincia.
- Municipio.
- Entidad.
- Demanda.

8.2.1.2 Tipo de filtrado:

- Por todos los filtros.
- Por al menos un filtro.

**RF9. Generar reporte “Plan CUM nivel MEP agrupado por provincias”.**

9.1 Seleccionar Visor de reportes.

9.1.1 Mostrar las siguientes opciones:

- Reportes.
- Formato del reporte.

9.1.2 Seleccionar ver el reporte “Plan CUM nivel MEP agrupado por provincias”.

9.1.3 Seleccionar el formato en que se desea ver el reporte:

- HTML.
- PDF.
- Excel.

9.1.4 Generar reporte “Plan CUM nivel MEP agrupado por provincias” en el formato que se seleccionó con los siguientes datos:

- Provincia.
- Municipio
- OACE-CAP-PP.
- Programas Priorizados.
- Cantidad de plazas asignadas.

9.2 Filtrar reporte “Plan CUM nivel MEP agrupado por provincias”.

9.2.1 Mostrar las opciones:

9.2.1.1 Filtrar reporte:

- CUM\_nombre.
- Año.
- Organismo.
- Provincia.
- Municipio.
- Entidad.
- Demanda.

9.2.1.2 Tipo de filtrado:

- Por todos los filtros.
- Por al menos un filtro.

**RF10. Generar reporte “Plan CUM municipal”.**

10.1 Seleccionar Visor de reportes.

10.1.1 Mostrar las siguientes opciones:

- Reportes.
- Formato del reporte.

10.1.2 Seleccionar ver el reporte “Plan CUM municipal”.

10.1.3 Seleccionar el formato en que se desea ver el reporte:

- HTML.
- PDF.
- Excel.

10.1.4 Generar reporte “Plan CUM municipal” en el formato que se seleccionó con los siguientes datos:

- CUM\_nombre.
- OACE-CAP.
- Programas Priorizados.
- Carreras que solicita.
- Propuesta.
- Demanda.

10.2 Filtrar reporte “Plan CUM municipal”.

10.2.1 Mostrar las opciones:

10.2.1.1 Filtrar reporte:

- CUM\_nombre.
- Año.
- Organismo.
- Provincia.
- Municipio.
- Entidad.
- Demanda.

10.2.1.2 Tipo de filtrado:

- Por todos los filtros.
- Por al menos un filtro.

**RF11. Generar reporte “Plan CUM provincial”.**

11.1 Seleccionar Visor de reportes.

11.1.1 Mostrar las siguientes opciones:

- Reportes.
- Formato del reporte.

11.1.2 Seleccionar ver el reporte “Plan CUM provincial”.

11.1.3 Seleccionar el formato en que se desea ver el reporte:

- HTML.
- PDF.
- Excel.

11.1.4 Generar reporte “Plan provincial” en el formato que se seleccionó con los siguientes datos:

- CUM\_nombre.
- OACE-CAP.
- Programas Priorizados.
- Carreras que solicita.
- Propuesta.
- Demanda.

11.2 Filtrar reporte “Plan CUM provincial”.

11.2.1 Mostrar las opciones:

11.2.1.1 Filtrar reporte:

- CUM\_nombre.
- Año.
- Organismo.
- Provincia.
- Municipio.
- Entidad.
- Demanda.

11.2.1.2 Tipo de filtrado:

- Por todos los filtros.
- Por al menos un filtro.

**RF12. Generar reporte “Plan CUM a nivel MEP”.**

12.1 Seleccionar Visor de reportes.

12.1.1 Mostrar las siguientes opciones:

- Reportes.
- Formato del reporte.

12.1.2 Seleccionar ver el reporte “Plan CUM a nivel MEP”.

12.1.3 Seleccionar el formato en que se desea ver el reporte:

- HTML.
- PDF.
- Excel.

12.1.4 Generar reporte “Plan CUM a nivel MEP” en el formato que se seleccionó con los siguientes datos:

- CUM\_nombre.
- OACE-CAP.
- Programas Priorizados.
- Carreras que solicita.
- Propuesta.
- Demanda.

12.2 Filtrar reporte “Plan CUM nacional”.

12.2.1 Mostrar las opciones:

12.2.1.1 Filtrar reporte:

- CUM\_nombre.
- Año.
- Organismo.
- Provincia.
- Municipio.
- Entidad.
- Demanda.

12.2.1.2 Tipo de filtrado:

- Por todos los filtros.
- Por al menos un filtro.

**RF13. Generar reporte “Plan CUM nivel municipal agrupado por grupo de carreras”.**

13.1 Seleccionar Visor de reportes.

13.1.1 Mostrar las siguientes opciones:

- Reportes.
- Formato del reporte.

13.1.2 Seleccionar ver el reporte “Plan CUM nivel municipal agrupado por grupo de carreras”.

13.1.3 Seleccionar el formato en que se desea ver el reporte:

- HTML.
- PDF.
- Excel.

13.1.4 Generar reporte “Plan CUM nivel municipal agrupado por grupo de carreras” en el formato que se seleccionó con los siguientes datos:

- Provincia.
- Municipio.
- Total de carreras propuestas por municipio.
- Carreras.

13.2 Filtrar reporte “Plan CUM nivel municipal agrupado por grupo de carreras”.

13.2.1 Mostrar las opciones:

13.2.1.1 Filtrar reporte:

- CUM\_nombre.
- Año.
- Organismo.
- Provincia.
- Municipio.
- Entidad.
- Demanda.

13.2.1.2 Tipo de filtrado:

- Por todos los filtros.
- Por al menos un filtro.

**RF14. Generar reporte “Plan CUM nivel provincial agrupado por grupo de carreras”.**

14.1 Seleccionar Visor de reportes.

14.1.1 Mostrar siguientes opciones:

- Reportes.
- Formato del reporte.

14.1.2 Seleccionar ver el reporte “Plan CUM nivel provincial agrupado por grupo de carreras”.

14.1.3 Seleccionar el formato en que se desea ver el reporte:

- HTML.
- PDF.
- Excel.

14.1.4 Generar reporte “Plan CUM nivel provincial agrupado por grupo de carreras” en el formato que se seleccionó con los siguientes datos:

- Provincia.
- Municipio.
- Total de carreras propuestas por municipio.
- Carreras.

14.2 Filtrar reporte “Plan CUM nivel provincial agrupado por grupo de carreras”.

14.2.1 Mostrar las opciones:



14.2.1.1 Filtrar reporte:

- CUM\_nombre.
- Año.
- Organismo.
- Provincia.
- Municipio.
- Entidad.
- Demanda.

14.2.1.2 Tipo de filtrado:

- Por todos los filtros.
- Por al menos un filtro.

## 2.12.2 Requerimientos no Funcionales

### Usabilidad

- ✚ A los administradores finales de la aplicación se les debe de dar un adiestramiento básico en el uso de la aplicación. Estas personas deben tener un nivel de acceso amplio en la aplicación para poder darle respuesta a cada incidente ocurrido.
- ✚ Los usuarios deben tener un conocimiento básico sobre informática para poder trabajar con el software a desarrollar.

### Fiabilidad

- ✚ Crear una copia que abarque toda la información que se necesite según el organismo que la use (versión MEP u organismo).
- ✚ Permitir la transferencia de información desde las distintas versiones (MEP u Organismo) hacia las otras.
- ✚ Permitir la transferencia de información de manera fragmentada y que después se pueda unir a la base de datos del MEP.
- ✚ El sistema deberá realizar el borrado de toda la información que se encuentre en la base de datos del MEP.

### Eficiencia

- ✚ El sistema deberá funcionar con un máximo rendimiento pero ajustado a bajas prestaciones de las computadoras debido que no todos los organismos poseen tecnología de punta.
- ✚ El sistema requiere de un buen rendimiento que se apoya en el mínimo acceso a base de datos y realización de consultas no redundantes.
- ✚ El sistema se apoya en una transferencia mínima de datos Cliente-Servidor.
- ✚ Para un funcionamiento óptimo de la aplicación se deben seguir las diferentes técnicas de elaboración en la web, que faciliten el rápido acceso a sus páginas.
- ✚ La eficiencia del producto debe estar determinada en gran medida por el aprovechamiento de los recursos que se disponen en el modelo Cliente-Servidor, y la velocidad de las consultas en la base de datos.
- ✚ La herramienta propuesta debe ser rápida y el tiempo de respuesta debe ser el mínimo posible, adecuado a la rapidez con que el cliente requiere la respuesta a su petición.

### **Soporte**

El sistema deberá presentar un manual de usuario, permitiendo con ello un correcto uso de sus funcionalidades y brindarle al usuario una mayor experiencia del trabajo con el mismo.

La documentación del sistema debe estar actualizada en todos los aspectos, fases de trabajo y ciclos de desarrollo del mismo, permitiendo con ello un respaldo tanto de ingeniería como legal del desarrollo de dicho sistema.

El sistema contará con un grupo de soporte y asesoría al cliente del producto destinado a brindar asesoría y soporte técnico al mismo.

### **Restricciones de diseño**

Lenguaje de programación: PHP 5.

Como gestor de base de datos se utilizará PostgreSQL 8.3 o superior.

Los marcos de trabajo que se utilizarán son: Zend Framework para la capa de negocio, ExtJS para la capa de presentación y Doctrine para la capa de acceso a datos.

Como Entorno Integrado de Desarrollo (IDE) se empleará NetBeans 6.8.

El servidor de aplicaciones será Apache-2.0 o superior.

El modelado UML se hará con Visual Paradigm 6.1.

Se utilizará como metodología de desarrollo de software RUP.

Las interfaces destinadas al usuario, se programarán en JavaScript.

### **Hardware**

Se necesitará una impresora, para la impresión de los reportes del sistema.

Las PC clientes deben tener las siguientes características: memoria RAM de 256 MB o superior, 20 GB o más de disco duro, 1.2 GHz o más de procesador.

Todas las PC clientes deben estar conectadas a la red para poder acceder a la aplicación.

Se requiere de un servidor para Bases de Datos con las siguientes características:

- ✚ Servidor Xeon a 3.0 GHz.
- ✚ 1 GB de memoria RAM.
- ✚ Dos discos duros de 36 y 250 GB, este último con dos particiones.

### **Software**

Para el cliente:

Sistema operativo Windows o GNU/Linux. En el caso de Windows las versiones compatibles con Windows 95 o superior.

Microsoft Excel 2003 o superior.

Adobe Reader 8.0 o superior.

Las computadoras cliente del sistema y ubicadas en el dominio de la organización, deben tener el navegador Mozilla Firefox.

Para el servidor:

Sistema operativo Windows Server 2000 o superior o Linux; Ubuntu Server 7.10 o superior.

Servidor Web: Apache 2.0 o superior.

Gestor de base de datos: PostgreSQL 8.3 o superior.

### **Requisitos para la documentación de usuarios en línea y ayuda al sistema**

- ✚ El usuario debe tener acceso al Manual de Usuario del producto final, en este debe documentarse la forma de utilizar el sistema.
- ✚ Independientemente de donde se encuentre el operador en la interfaz, este podrá obtener ayuda referente a las operaciones que puede realizar, al igual que una propuesta de las operaciones posibles.

### **Componentes comprados**

- ✚ Se utilizan sistemas existentes en el Ministerio de Economía y Planificación, tales como Sistema Matriz, Sistema Plan de Distribución, Módulo Administración.

### **Interfaz de usuario**

- ✚ El sistema debe poseer una interfaz amigable al usuario, basada en web, brindando facilidades que permitan interactuar con el mismo de forma fácil y rápida.

### **Interfaz Hardware**

- ✚ El sistema deberá poseer una interfaz que permita la creación y validación de consultas, permitiendo la creación de reportes sobre las operaciones.

### **Requisitos de Licencia**

- ✚ El proyecto utiliza la política de software libre donde todas las herramientas que se utilizan son software libre. Para la herramienta Visual Paradigm se utiliza la licencia que la universidad compró.

### **Requisitos legales, de derecho de autor y otros**

La Universidad de las Ciencias Informáticas tiene el derecho.

### **2.13 Definición de los casos de uso del sistema**

Un caso de uso es una técnica para la captura de requisitos potenciales de un nuevo sistema o una actualización de software. Cada caso de uso proporciona uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para conseguir un objetivo específico.

#### **2.13.1 Listado de los casos de uso del sistema**

- ✚ Gestionar Modelo 1.FTC-CUM.
- ✚ Gestionar propuesta del plan de ingreso municipal.

- ✚ Gestionar propuesta del plan de ingreso provincial.
- ✚ Gestionar plan de ingreso.
- ✚ Generar reporte “Listado de entidades que solicitan CUM”.
- ✚ Generar reporte “Demanda CUM agrupada por provincias”.
- ✚ Generar reporte “Demanda CUM agrupado por grupo de carreras”.
- ✚ Generar reporte “Plan CUM nivel MEP agrupado por grupo de carreras”.
- ✚ Generar reporte “Plan CUM nivel MEP agrupado por provincias”.
- ✚ Generar reporte “Plan CUM municipal”.
- ✚ Generar reporte “Plan CUM provincial”.
- ✚ Generar reporte “Plan CUM a nivel MEP”.
- ✚ Generar reporte “Plan CUM nivel municipal agrupado por grupo de carreras”.
- ✚ Generar reporte “Plan CUM nivel provincial agrupado por grupo de carreras”.

### **2.13.2 Diagrama de casos de uso del sistema**

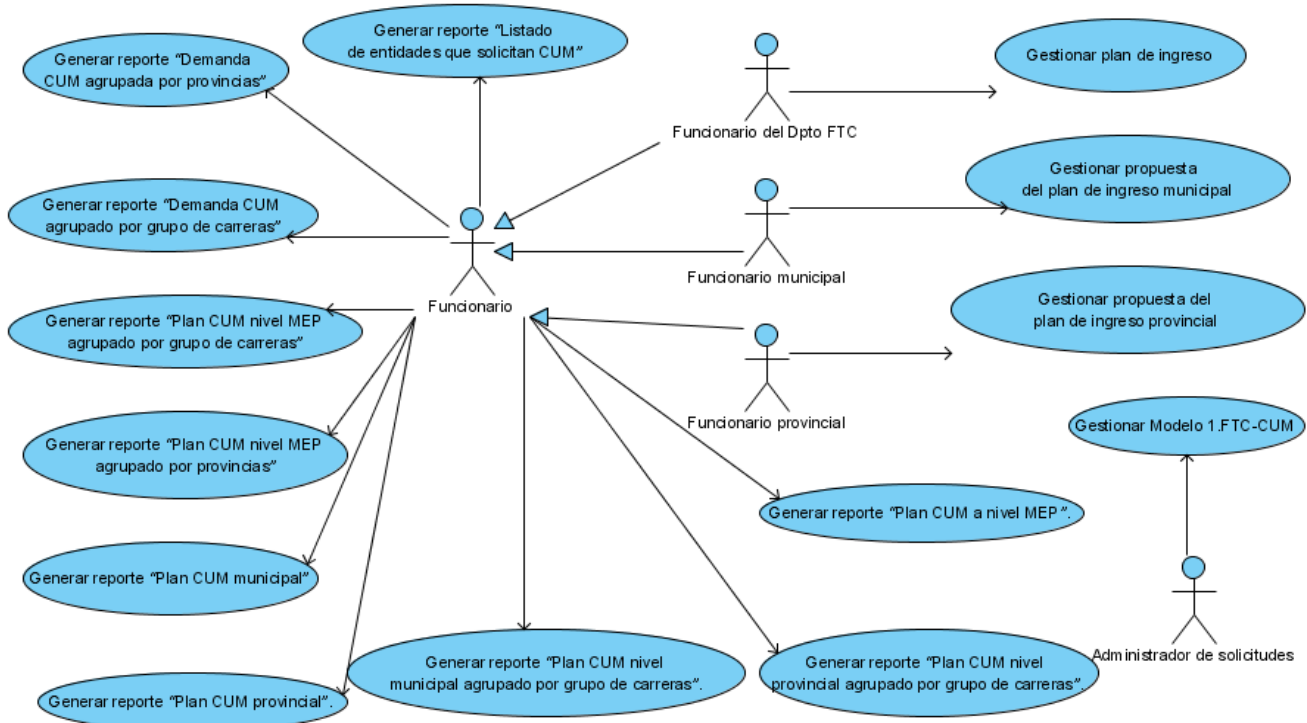


Figura # 3: Diagrama de casos de uso del sistema.

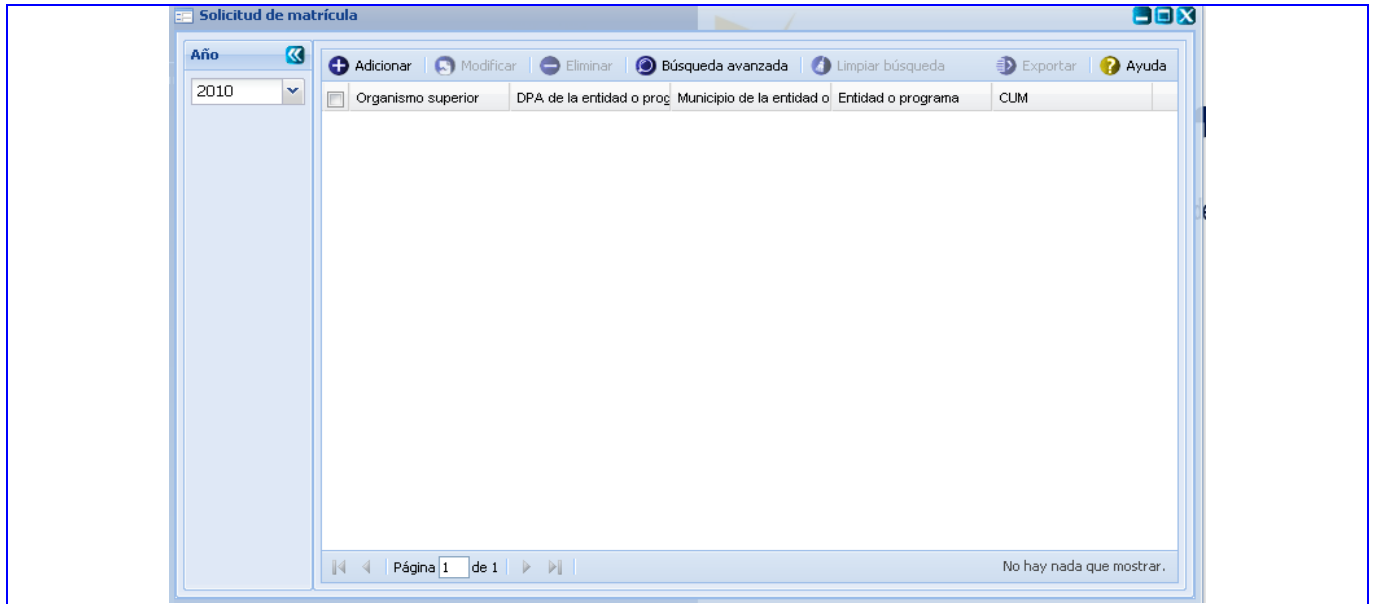
### 2.13.3 Descripción textual de los casos de uso del sistema

Mediante la descripción de los casos de uso del sistema se detalla la secuencia de eventos que los actores llevan a cabo para completar un proceso a través de la aplicación. A continuación se muestra la descripción del caso de uso Gestionar Modelo 1.FTC-CUM, en el Anexo # 5 se pueden encontrar las descripciones de los casos de uso más importantes.

Descripción del CU: Gestionar Modelo 1.FTC-CUM.

<b>Caso de Uso:</b>	Gestionar Modelo 1.FTC-CUM
<b>Actores:</b>	Administrador de solicitudes
<b>Resumen:</b>	El caso de uso comienza cuando el Administrador de solicitudes selecciona la opción Gestionar Modelo 1.FTC-CUM. Selecciona la opción que desea realizar como: agregar, modificar, eliminar, buscar o exportar. Introduce los datos necesarios y finaliza cuando el Administrador de solicitudes selecciona la opción salir.

<b>Precondiciones:</b>	El actor debe estar autenticado en el sistema y que se haya solicitado el caso de uso Gestionar Modelo 1.FTC-CUM.	
<b>Referencias</b>	RF 1	
<b>Prioridad</b>	Crítico	
<b>Flujo Normal de Eventos</b>		
<b>Sección “Gestionar Modelo 1.FTC-CUM”</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
1. Selecciona la opción de “Gestionar Modelo 1.FTC-CUM”.	2. Muestra todos los modelos creados hasta el momento del año actual con sus datos: <ul style="list-style-type: none"> <li>a) Organismo.</li> <li>b) Entidad / Programa.</li> <li>c) Provincia de la entidad o programa.</li> <li>d) Municipio de la entidad o programa.</li> <li>e) CUM donde estudiará el trabajador.</li> </ul> 3. Muestra las opciones para seleccionar el “Año”, para “Adicionar”, “Modificar”, “Eliminar”, “Búsqueda avanzada”, “Exportar” y “Limpiar búsqueda”.	
4. Ejecuta las siguientes acciones: <ul style="list-style-type: none"> <li>a) Solicita “Adicionar”.</li> <li>b) Selecciona uno de los modelos ya insertados y selecciona “Modificar”.</li> <li>c) Selecciona uno de los modelos ya insertados y selecciona “Eliminar”.</li> <li>d) Solicita “Búsqueda avanzada”.</li> <li>e) Selecciona un modelo y solicita “Exportar”.</li> <li>f) Solicita “Limpiar búsqueda”.</li> </ul>	5. Ejecuta las siguientes acciones: <ul style="list-style-type: none"> <li>a) Va a la sección “Adicionar Modelo 1.FTC-CUM”.</li> <li>b) Va a la sección “Modificar Modelo 1.FTC-CUM”.</li> <li>c) Va a la sección “Eliminar Modelo 1.FTC-CUM”.</li> <li>d) Va a la sección “Búsqueda Avanzada”.</li> <li>e) Va a la sección “Exportar”.</li> <li>f) Va a la sección “Limpiar Búsqueda”.</li> </ul>	
<b>Prototipo de Interfaz</b>		



**Sección “Adicionar Modelo 1.FTC-CUM”**

**Flujo Básico**

Acción del Actor	Respuesta del Sistema
	<ol style="list-style-type: none"> <li>1. Muestra los campos para introducir los siguientes datos:                             <ol style="list-style-type: none"> <li>a) DPA de la Entidad o programa.</li> <li>b) Municipio de la Entidad o programa.</li> <li>c) Organismo Superior.</li> <li>d) Entidad o programa.</li> <li>e) Municipio de la CUM.</li> <li>f) Código.</li> <li>g) Carrera.</li> <li>h) Cantidad de plazas.</li> </ol> </li> <li>2. Muestra las opciones de “Agregar fila”, “Eliminar fila” (Sólo se muestra si existe alguna fila), “Cancelar”, “Aceptar” y “Aplicar” (Realiza la operación y limpia el formulario).</li> </ol>

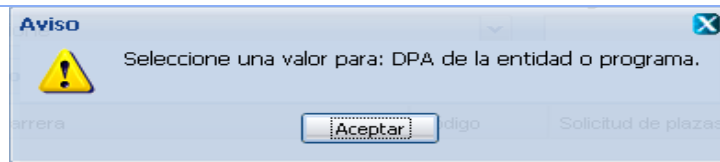


<p>3. Ejecuta una de las siguientes acciones:</p> <ul style="list-style-type: none"><li>a) Llena los datos y selecciona "Aceptar".</li><li>b) Llena los datos y selecciona "Aplicar".</li><li>c) Selecciona "Agregar fila".</li><li>d) Selecciona "Eliminar fila".</li><li>e) Selecciona "Cancelar".</li></ul>	<p>4. Ejecuta las siguientes acciones:</p> <ul style="list-style-type: none"><li>a) Verifica que todos los campos estén llenos y que sus datos estén introducidos correctamente, en caso contrario ir al flujo alterno 1.</li><li>b) Verifica que todos los campos estén llenos y que sus datos estén introducidos correctamente, si es así registra los datos y muestra un mensaje indicando que la solicitud ha sido creada satisfactoriamente ,limpia el formulario y vuelve a la acción 1 de flujo básico de la sección "Adicionar Modelo 1.FTC-CUM" ,en caso contrario ir al flujo alterno 1.</li><li>c) Agrega la fila si la cantidad de filas existentes es igual a la cantidad máxima de carreras el botón se oculta automáticamente.</li><li>d) Va al flujo alterno 2.</li><li>e) Va al paso 2 del flujo normal de eventos de la sección "Gestionar Modelo 1.FTC-CUM".</li></ul> <p>5.Registra los datos del nuevo modelo:</p> <ul style="list-style-type: none"><li>•DPA de la entidad o programa.</li><li>•Municipio de la entidad o programa.</li><li>•Organismo Superior.</li><li>•Entidad o programa.</li><li>•Municipio de la CUM.</li><li>•Código.</li><li>•Carrera.</li><li>•Cantidad de plazas.</li></ul> <p>6. Muestra un mensaje indicando que el modelo fue creado con éxito.</p> <p>7. Va a la acción 2 del flujo normal de eventos de la sección "Gestionar Modelo 1.FTC-CUM".</p>
<p style="text-align: center;"><b>Prototipo de Interfaz</b></p>	

**Flujo Alternativo 1**

Acción del Actor	Respuesta del Sistema
	<ol style="list-style-type: none"> <li>1. Ejecuta la siguiente acción:                             <ol style="list-style-type: none"> <li>a) Muestra un mensaje indicando el campo que se debe llenar.</li> </ol> </li> <li>2. Va a la acción 1 del flujo básico de la sección "Adicionar Modelo 1.FTC-CUM".</li> </ol>

**Prototipo de Interfaz**

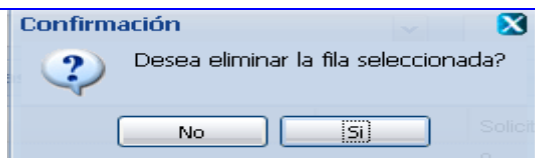


**Flujo Alternativo 2**

Acción del Actor	Respuesta del Sistema

1. Selecciona la cantidad de filas que desea eliminar.	2. Muestra un mensaje preguntando si desea eliminar las filas seleccionadas.
3. Ejecuta una de las siguientes acciones: a) Selecciona sí. b) Selecciona no.	4. Ejecuta las siguientes acciones: a) Elimina la fila seleccionada (si al eliminar las filas seleccionadas la cantidad de filas restantes es 0, entonces el botón eliminar filas se oculta automáticamente.) b) Va a la acción 1 del flujo básico de la sección "Adicionar Modelo 1.FTC-CUM".

**Prototipo de Interfaz**



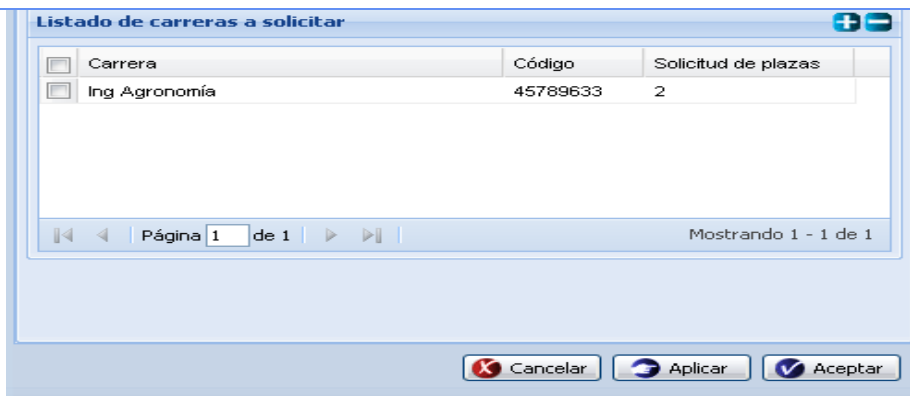
**Sección "Modificar Modelo 1.FTC-CUM"**

**Flujo Básico**

Acción del Actor	Respuesta del Sistema
	<ol style="list-style-type: none"> <li>Muestra los datos del modelo de solicitud seleccionado: <ul style="list-style-type: none"> <li>Carrera.</li> <li>Cantidad.</li> <li>Código.</li> </ul> </li> <li>Muestra las opciones de "Agregar fila", "Eliminar fila", "Cancelar", "Aceptar" y "Aplicar".</li> </ol>
<ol style="list-style-type: none"> <li>Ejecuta una de las siguientes acciones: <ol style="list-style-type: none"> <li>Modifica los datos y selecciona "Aceptar".</li> <li>Modifica los datos y selecciona "Aplicar".</li> <li>Selecciona "Agregar fila".</li> <li>Selecciona "Eliminar fila".</li> <li>Selecciona "Cancelar".</li> </ol> </li> </ol>	<ol style="list-style-type: none"> <li>Ejecuta las siguientes acciones: <ol style="list-style-type: none"> <li>Verifica que sus datos estén introducidos correctamente, en caso contrario ir al flujo alternativo 1.</li> <li>Verifica que sus datos estén introducidos correctamente, si es así registra los datos y muestra un mensaje indicando que la solicitud ha sido modificada satisfactoriamente, limpia el</li> </ol> </li> </ol>

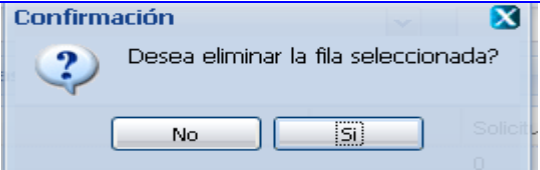
	<p>formulario y vuelve a la acción 1 de flujo básico de la sección “Modificar Modelo 1.FTC-CUM”, en caso contrario ir al flujo alterno 1.</p> <p>c) Agrega las filas necesarias y si la cantidad de filas existentes es igual a la cantidad máxima de carreras el botón se oculta automáticamente.</p> <p>d) Va al flujo alterno 2.</p> <p>e) Va al paso 2 del flujo normal de eventos de la sección “Gestionar Modelo 1.FTC-CUM”.</p> <p>5. Registra los nuevos datos del modelo:</p> <ul style="list-style-type: none"> <li>• Carrera.</li> <li>• Cantidad.</li> <li>• Código</li> </ul> <p>6. Muestra un mensaje indicando que el modelo fue modificado correctamente.</p> <p>7. Va a la acción 2 del flujo normal de eventos de la sección “Gestionar Modelo 1.FTC-CUM”.</p>
--	--

**Prototipo de Interfaz**



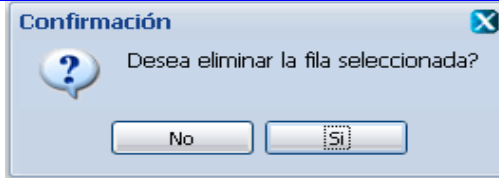
**Flujo Alterno 1**

Acción del Actor	Respuesta del Sistema
	<p>1. Ejecuta la siguiente acción:</p> <p>a) Muestra un mensaje indicando cuales campos están incorrectos.</p> <p>2. Va a la acción 1 del flujo básico de la sección</p>

"Modificar Modelo 1.FTC-CUM".	
<b>Flujo Alterno 2</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. Selecciona la cantidad de filas que desea eliminar.	2. Muestra un mensaje preguntando si desea eliminar las filas seleccionadas.
3. Ejecuta una de las siguientes acciones: a) Selecciona "sí". b) Selecciona "no".	4. Ejecuta las siguientes acciones: a) Elimina la fila seleccionada (si al eliminar las filas seleccionadas la cantidad de filas restantes es 0, entonces el botón eliminar filas se oculta automáticamente.) b) Va a la acción 1 del flujo básico de la sección "Adicionar Modelo 1.FTC-CUM".
<b>Prototipo de Interfaz</b>	
	
<b>Sección "Eliminar Modelo 1.FTC-CUM"</b>	
<b>Flujo Básico</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1. Muestra un mensaje de confirmación preguntando si está seguro de eliminar el modelo. 2. Muestra las opciones de "Aceptar" y "Cancelar".

<p>3. Realiza las siguientes acciones:</p> <ol style="list-style-type: none"> <li>Selecciona "Aceptar".</li> <li>Selecciona "Cancelar".</li> </ol>	<p>4. Ejecuta las siguientes acciones:</p> <ol style="list-style-type: none"> <li>Elimina completamente el modelo seleccionado.</li> <li>Va al paso 2 del flujo normal de eventos de la sección "Gestionar Modelo 1.FTC-CUM".</li> </ol> <p>5. Muestra un mensaje indicando que se eliminó completamente el modelo.</p> <p>6. Va a la acción 2 del flujo normal de eventos de la sección "Gestionar Modelo 1.FTC-CUM".</p>
--	--

**Prototipo de Interfaz**



**Sección "Búsqueda avanzada"**

**Flujo Básico**

Acción del Actor	Respuesta del Sistema
	<ol style="list-style-type: none"> <li>El sistema muestra los campos:                             <ol style="list-style-type: none"> <li>DPA de la Entidad o programa.</li> <li>Municipio de la entidad o programa.</li> <li>Organismo Superior.</li> </ol> </li> <li>Muestra las opciones de "Cancelar" y "Aceptar".</li> </ol>
<p>3. Realiza las siguientes acciones:</p> <ol style="list-style-type: none"> <li>Llena los campos y selecciona "Aceptar".</li> <li>Selecciona "Aceptar" sin llenar los campos.</li> <li>Selecciona "Cancelar".</li> </ol>	<p>4. Ejecuta las siguientes acciones:</p> <ol style="list-style-type: none"> <li>Muestra el resultado de la búsqueda, si no ir al flujo alterno 1.</li> <li>Muestra todos los modelos creados hasta el momento, si no hay modelos creados va al flujo alterno 1.</li> <li>Vuelve a la acción 1 del flujo básico de la sección "Gestionar Modelo 1.FTC-CUM".</li> </ol>

**Prototipo de Interfaz**

**Flujo Alterno 1**

Acción del Actor	Respuesta del Sistema
	<ol style="list-style-type: none"> <li>3. Ejecuta la siguiente acción:                             <ol style="list-style-type: none"> <li>b) Muestra un mensaje indicando que no hubo resultados.</li> </ol> </li> <li>4. Va a la acción 1 del flujo básico de la sección "Modificar Modelo 1.FTC-CUM".</li> </ol>

**Prototipo de Interfaz**

**Sección "Exportar"**

**Flujo Básico**

Acción del Actor	Respuesta del Sistema
	<ol style="list-style-type: none"> <li>1. Muestra las opciones para seleccionar el formato:                             <ul style="list-style-type: none"> <li>• Excel.</li> <li>• PDF.</li> </ul> </li> <li>2. Mostrar las opciones "Aceptar" y "Cancelar".</li> </ol>

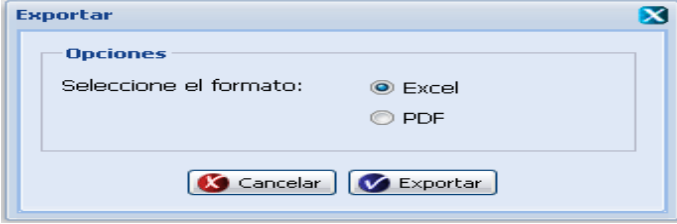
<p>3. Ejecuta las siguientes acciones:</p> <ul style="list-style-type: none"> <li>a) Selecciona el formato Excel.</li> <li>b) Selecciona el formato PDF.</li> </ul>	<p>4. Ejecuta las siguientes acciones:</p> <ul style="list-style-type: none"> <li>a) Genera un reporte en formato Excel.</li> <li>b) Genera un reporte en formato PDF.</li> </ul> <p>5. Vuelve a la acción 1 del flujo básico de la sección "Gestionar Modelo 1.FTC-CUM".</p>
<b>Prototipo de Interfaz</b>	
	
<b>Sección "Limpiar búsqueda".</b>	
<b>Flujo Básico</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	<ul style="list-style-type: none"> <li>1. Limpia todos los campos.</li> <li>2. Va a la acción 2 del flujo normal de eventos de la sección "Gestionar Modelo 1.FTC-CUM".</li> </ul>
<b>Poscondiciones</b>	Se actualizan todos los datos en la base de datos y quedan creados los modelos de solicitudes.

Tabla # 5: Descripción del CUN: Gestionar Modelo 1.FTC-CUM.

## 2.14 Conclusiones:

En este capítulo se abordó lo referente al modelo de negocio, sus objetivos y alcance, además se inició el desarrollo de la propuesta de solución, se obtuvo un listado de funcionalidades que debe tener el sistema, donde se adquirieron finalmente los casos de uso y los artefactos principales correspondientes a este flujo de trabajo: especificación de requisitos, diagramas de casos de uso y descripción de los casos de uso.



## Capítulo 3: Construcción de la solución propuesta

### 3.1 Introducción

En el presente capítulo se realizará el análisis y el diseño del Módulo Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales. Se muestran los diagramas de clases del análisis, con las interfaces, controladoras, entidades y las relaciones entre las mismas. Del diseño se presentan los diagramas de clases y de interacción correspondientes a cada caso de uso del sistema, mientras que de la Base de Datos se expone el diagrama de entidad relación.

### 3.2 Análisis

El flujo de trabajo análisis y diseño desempeña un papel fundamental en la fase de elaboración. El principal objetivo del análisis es comprender los requisitos funcionales con que cuenta el software para así estructurar el proyecto de forma clara y precisa, se profundiza en el dominio de la aplicación lo que permite una mayor comprensión del problema para modelar la solución.

#### 3.2.1 Diagramas de clases del análisis

El diagrama de clases del análisis es un diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Estos diagramas de clases son utilizados durante el proceso de análisis y el diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, los componentes que se encargan del funcionamiento y la relación entre uno y otro.

- ✚ Clase interfaz: se utilizan para modelar la interacción entre el sistema y los actores.
- ✚ Clase controladora: representan coordinación, secuencia, y control de otros objetos y se usan con frecuencia para encapsular el control de un caso de uso en concreto.
- ✚ Clase entidad: se utilizan para modelar información que posee una vida larga y que a menudo es persistente. Modelan la información y el comportamiento asociado de algún fenómeno o concepto, como una persona, un objeto, o un suceso del mundo real.

A continuación se muestra el diagrama de clase del análisis del caso de uso Gestionar plan de ingreso:

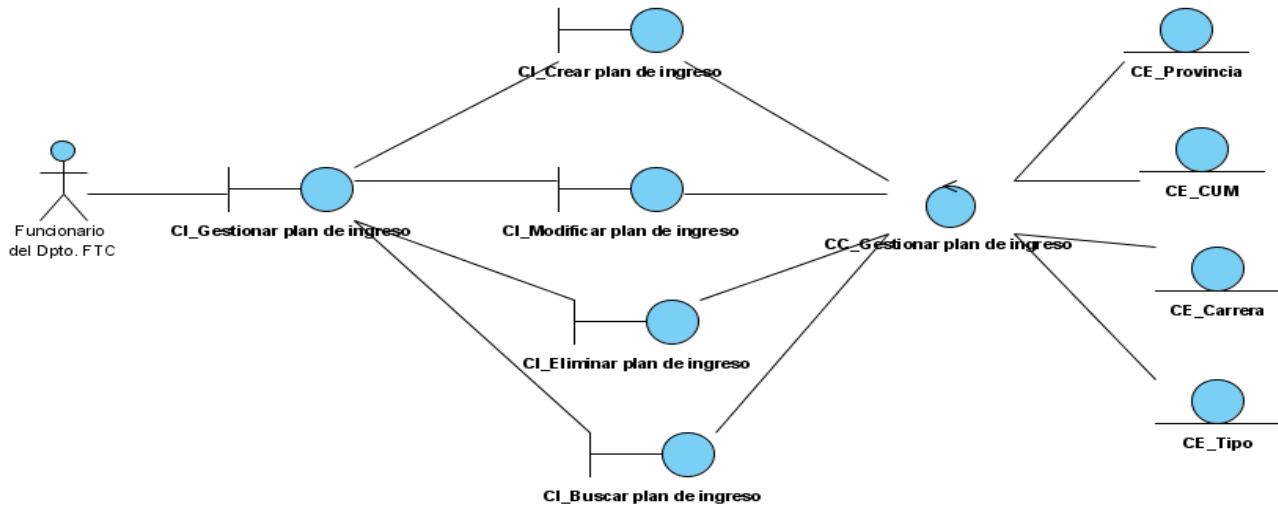


Figura # 4: Diagrama de clases del análisis.

En el Anexo # 6 se pueden encontrar los diagramas de clases del análisis para cada caso de uso del sistema.

### 3.2.2 Diagramas de Colaboración del análisis

Los diagramas de interacción representan una vista dinámica del sistema y se pueden clasificar en dos tipos, diagramas de colaboración y diagramas de secuencia. Un diagrama de interacción representa la secuencia de acciones que ocurren desde que el actor comienza el caso de uso, así como los mensajes que se envían desde cada una de las clases. En el análisis se usan los diagramas de colaboración, ya que el objetivo principal es identificar las funcionalidades de cada objeto y las responsabilidades sobre ellos.

A continuación se muestra el diagrama de colaboración del análisis del caso de uso Gestionar plan de ingreso (Sección crear):

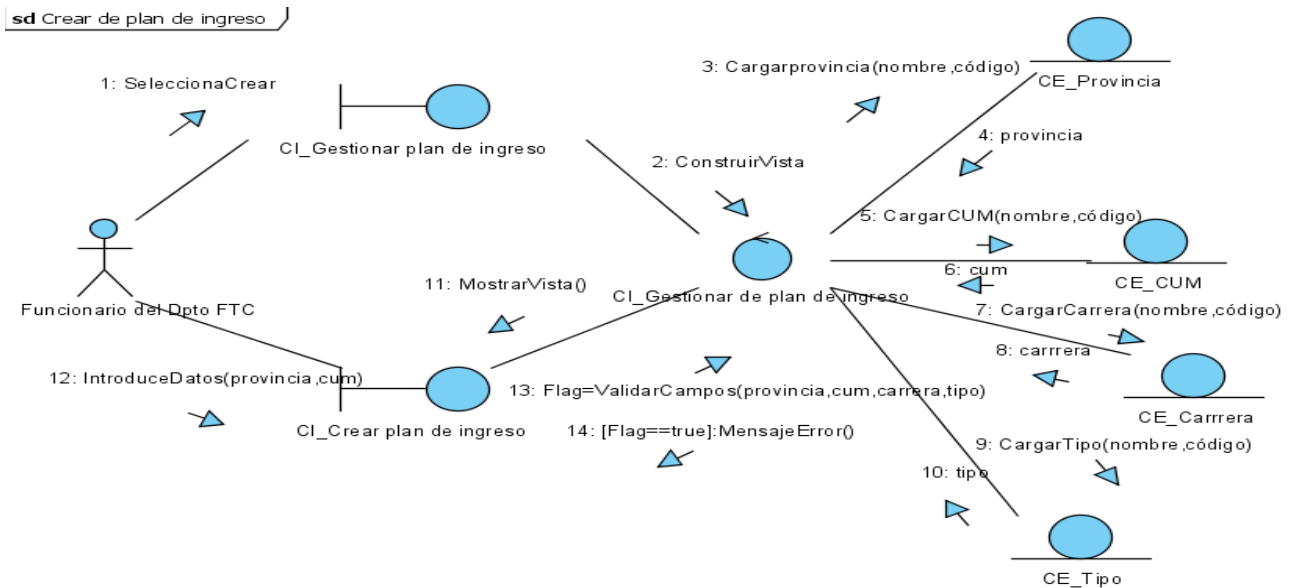


Figura # 5. Diagrama de colaboración del análisis.

En el Anexo # 7 se podrán apreciar los diagramas de colaboración para cada escenario de los casos de uso más significativos.

### 3.3 Diseño

En la fase de diseño se modela el sistema de manera que soporte todos los requisitos, tanto funcionales como no funcionales, creándose así una entrada apropiada para las actividades de implementación. El propósito del diseño es especificar una solución que trabaje y pueda ser fácilmente convertida en código fuente, y construir una arquitectura simple y fácilmente extensible.

#### 3.3.1 Arquitectura y patrones de diseño utilizado

Uno de los elementos base del proceso de desarrollo de software es diseñar la Arquitectura de Software. Esencialmente sobre ella se sustentan todos los mecanismos de diseño y representaciones de la estructura general de la aplicación a desarrollar. De la cohesión, utilidad y flexibilidad de los componentes de la arquitectura dependerán la calidad final y la utilidad del software.

El diseño del módulo se realizará teniendo en cuenta el patrón de arquitectura Modelo Vista Controlador (MVC).

### **Modelo Vista Controlador (MVC)**

Modelo Vista Controlador o MVC describe una forma, muy utilizada, de organizar el código de una aplicación separando los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos .

- ✚ Modelo: es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El modelo no tiene conocimiento específico de los controladores o de las vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene encomendada la responsabilidad de mantener enlaces entre el modelo y sus vistas, y notificar a las vistas cuando cambia el modelo.
- ✚ Vista: es el objeto que maneja la presentación visual de los datos representados por el modelo. Genera una representación visual del modelo y muestra los datos al usuario. Interactúa con el modelo a través de una referencia al propio modelo.
- ✚ Controlador: es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el modelo cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del modelo o por alteraciones de la vista. Interactúa con el modelo a través de una referencia al propio modelo. (13)

### **Patrones GOF:**

Decorator: este patrón permite añadir funcionalidad a una clase dinámicamente. Zend framework implementa dicho patrón en la clase `Zend_View`, encargada de asignarle responsabilidades a objetos de manera dinámica y configurarlos con nuevos atributos.

Singleton: este patrón garantiza la existencia de una única instancia para una clase y la creación de un mecanismo de acceso global a dicha instancia. Zend Framework tiene una instancia única del controlador frontal disponible mediante este patrón para lograr una vía de entrada única a las solicitudes.

Facade: este patrón simplifica los accesos a las clases de la capa de acceso a datos proporcionando un objeto que todas las clases de capas superiores utilizarán para acceder a las clases contenidas en la capa del modelo. Define una interface de más alto nivel que permite usar el sistema más fácil. El objetivo de la aplicación de este patrón es reducir la dependencia entre clases. Se utilizará una clase intermediaria entre las clases controladoras de Zend Framework y las de acceso a datos de Doctrine, la que brindará, de las

operaciones de acceso a datos, sólo las que necesiten los controladores para su funcionamiento, lo que reduce la dependencia de estos entre las múltiples clases de acceso a datos existentes en el sistema.

Factory: este patrón proporciona una interfaz para crear familias de objetos relacionados o dependientes sin especificar su clase concreta. Permite configurar en tiempo de ejecución un sistema con una familia u otra de objetos. Además, garantiza que un conjunto de clases se usen a la vez. Zend Framework provee una API para el acceso a datos conformada por un conjunto de clases que implementa dicho patrón.

**Patrones de acceso a datos:**

Active Record: este patrón representa de forma Orientada a Objetos los datos de una Base de Datos Relacional, definiendo interfaces sencillas para acceder y manipular esos datos. Es un enfoque al problema de acceder a los datos de una base de datos. Una fila en la tabla de la base de datos (o vista) se envuelve en una clase, de manera que se asocian filas únicas de la base de datos con objetos del lenguaje de programación usado. Cuando se crea uno de estos objetos, se añade una fila a la tabla de la base de datos. Cuando se modifican los atributos del objeto, se actualiza la fila de la base de datos. Doctrine usa este patrón para manejar la base de datos.

Row Data Gateway: este patrón asume el comportamiento de un objeto que actúa como puerta de enlace a una fila de una tabla de la base de datos. Tiene propiedades que reflejan las columnas de la tabla, y métodos de actualización en la base. Zend Framework provee una API para el acceso a datos conformada por un conjunto de clases que implementa dicho patrón.

Table Data Gateway: similar a Row Data Gateway, pero a diferencia de este define la estructura de acceso por registros en las entidades de la base de datos, especifica su acceso a nivel de tabla, proponiendo un objeto que se comporte como puerta de enlace a cada tabla de la base de datos. Contiene una interface que permite actualizar, buscar, borrar e insertar en la tabla y puede retornar un registro, un grupo de registro, y hasta un objeto del dominio. Zend Framework provee una API para el acceso a datos conformada por un conjunto de clases que implementa dicho patrón.

**Patrones de asignación de responsabilidades (GRASP):**

Controlador: este patrón se tiene en cuenta para realizar las asignaciones en cuanto al manejo de los eventos del sistema y definir sus operaciones. Zend Framework contribuye a la utilización de este patrón

ya que define un Controlador Frontal (Front Controller) que implica que todas las solicitudes son dirigidas a un único script PHP que se encarga de instanciar al controlador frontal y redirigir las llamadas.

Experto: este patrón propone como solución asignar la responsabilidad a la clase que cuenta con la información necesaria para cumplir la responsabilidad. Las clases que brinda el framework Ext JS se encargarán de visualizar las interfaces ya que cuentan con la información para crear los diferentes componentes visuales, las clases controladoras del Zend Framework manejarán las peticiones del cliente, y las clases que genera y utiliza el Doctrine serán las encargadas del acceso a datos pues contienen y representan los datos que manejará el sistema. Permitiendo que se conserve el encapsulamiento, soportando un bajo acoplamiento y una alta cohesión.

Creador: este patrón se tiene en cuenta para la asignación de responsabilidades a las clases relacionadas con la creación de objetos, de forma tal que una instancia de un objeto sólo pueda ser creada por el objeto que contiene la información necesaria para ello. Existe un único script PHP que se encarga de instanciar al controlador frontal, este último es el encargado de instanciar las clases controladoras y estas, a su vez, instancian objetos de la clase Zend\_View. El uso de este patrón permite crear las dependencias mínimas necesarias entre las clases, favoreciendo al mantenimiento del sistema.

Bajo Acoplamiento: este patrón brinda como solución asignar responsabilidades de manera que las clases no dependan fuertemente de otras. Ofreciendo como beneficio que son fáciles de entender por separadas, fáciles de reutilizar y no se afectan por cambios de otros componentes. Dicho patrón se tiene en cuenta debido a la importancia de realizar un diseño de clases independientes que soporten los cambios.

Alta cohesión: este patrón propone asignar la responsabilidad de manera que la complejidad se mantenga dentro de límites manejables asumiendo solamente las responsabilidades que deben manejar, evadiendo un trabajo excesivo. Su utilización mejora la claridad y facilidad con que se entiende el diseño, simplifica el mantenimiento y las mejoras de funcionalidad, generan un bajo acoplamiento, soporta mayor capacidad de reutilización.

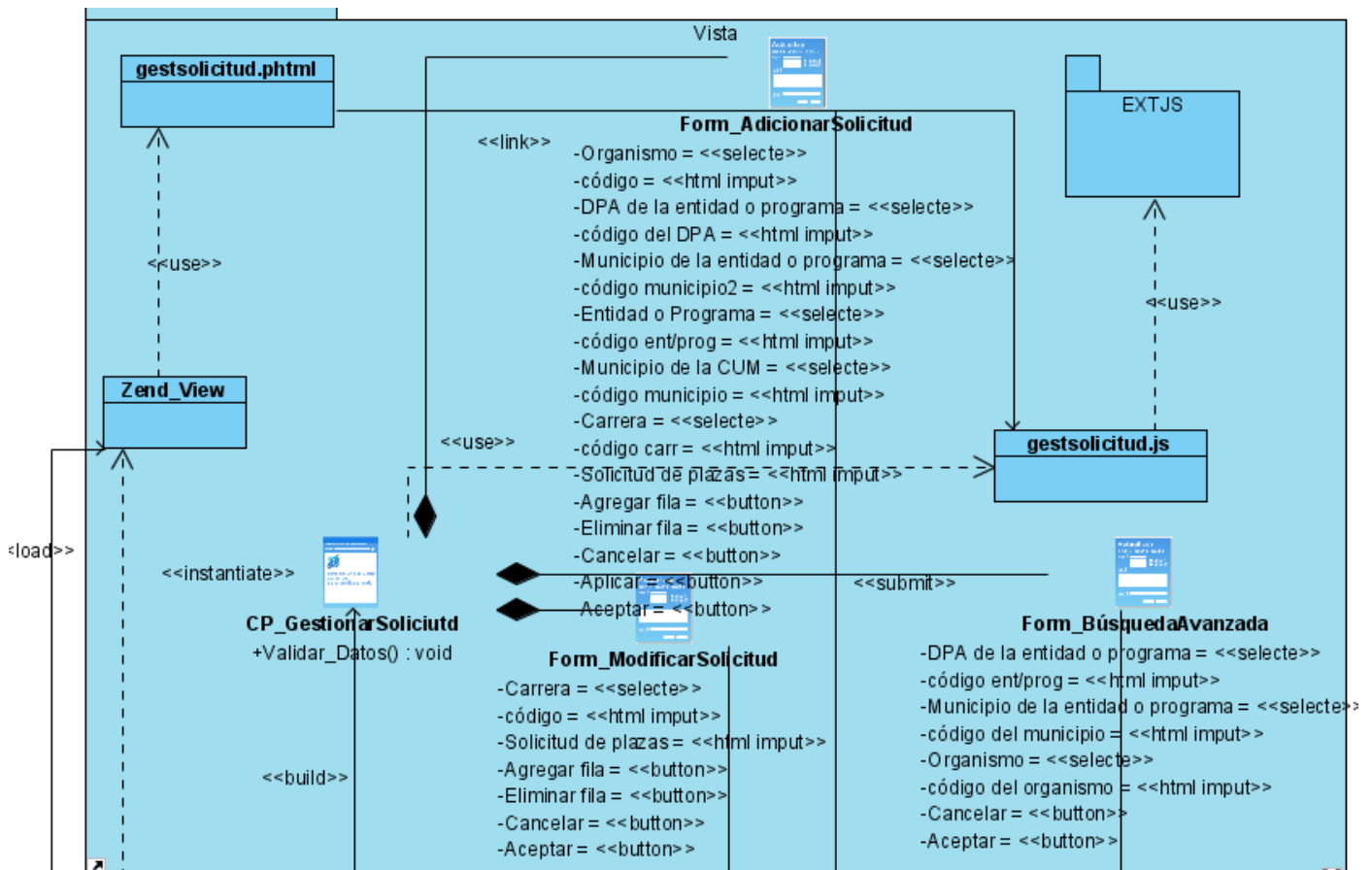
### **3.3.2 Modelo de diseño**

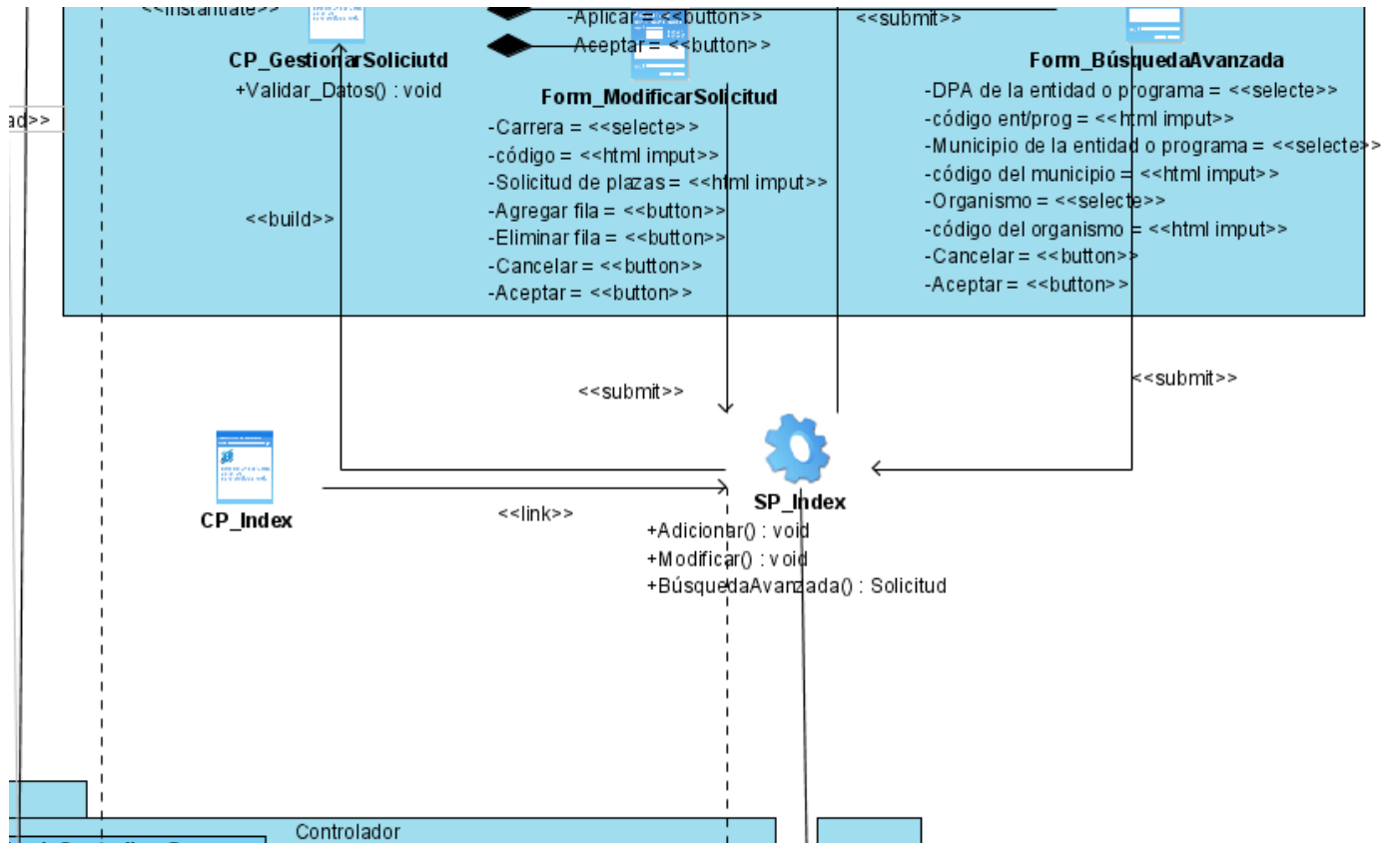
En el flujo de trabajo de Análisis y Diseño, definido por RUP, se describe cómo se debe implementar el sistema, como resultado de un análisis realizado a los requerimientos no funcionales del mismo. El diseño no debe tener ambigüedades para que el modelo final que se obtenga sea suficiente para la

implementación. El modelo de diseño es el resultado más importante de este flujo de trabajo, el cual consiste en colaboraciones de clases, las que pueden ser agrupadas en paquetes y subsistemas, además describe la realización de los casos de uso.

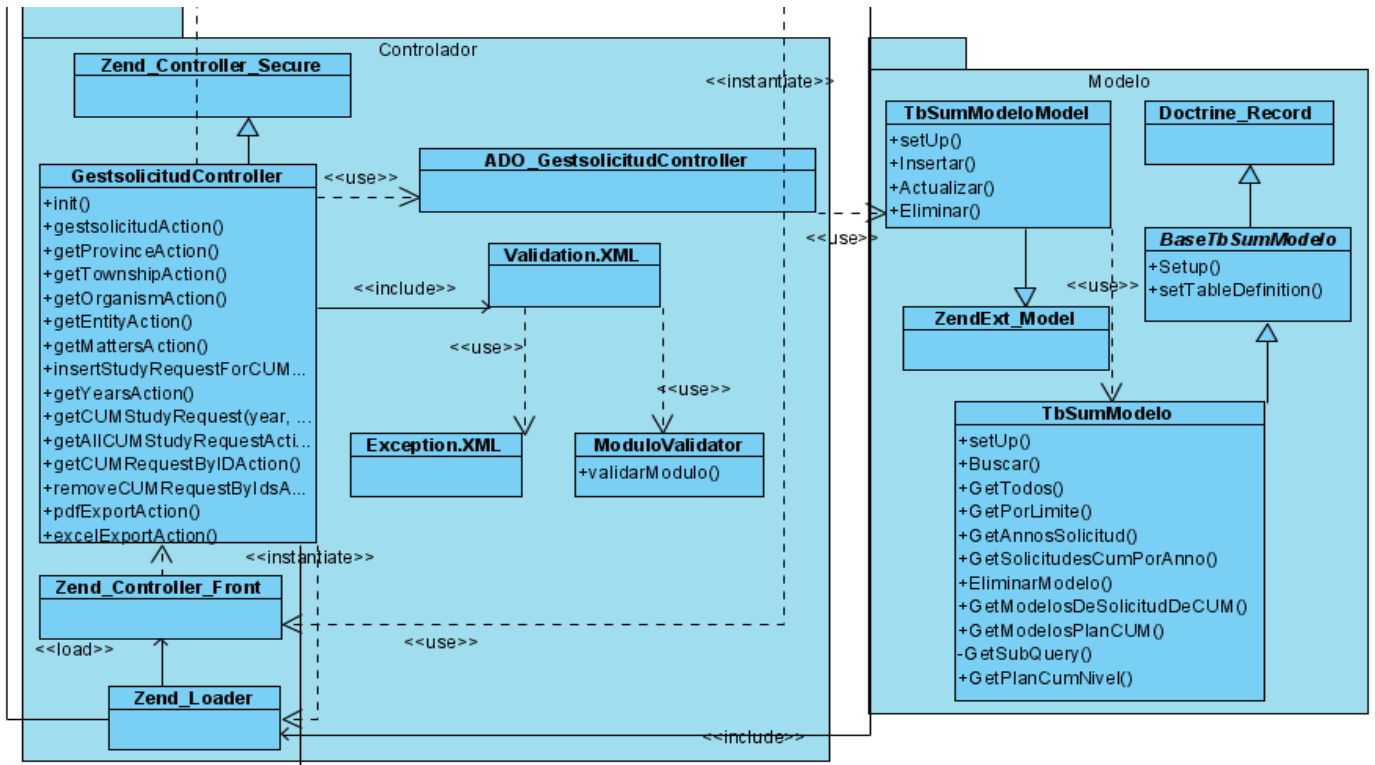
### Realización de los diagramas de clases del diseño

A continuación se muestra el diagrama de diseño del caso de uso Gestionar Modelo 1.FTC-CUM:









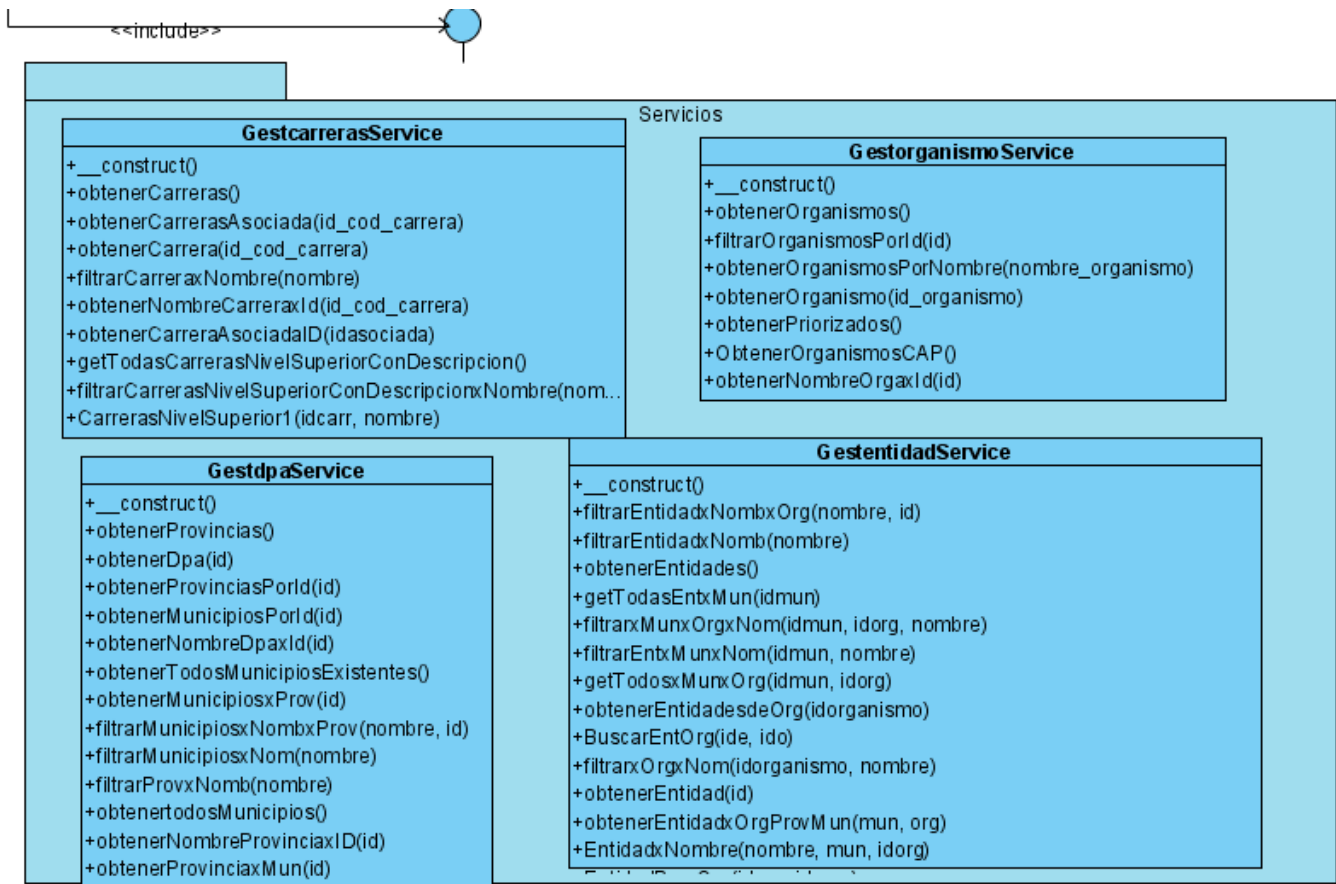


Figura # 6: Diagrama de diseño.

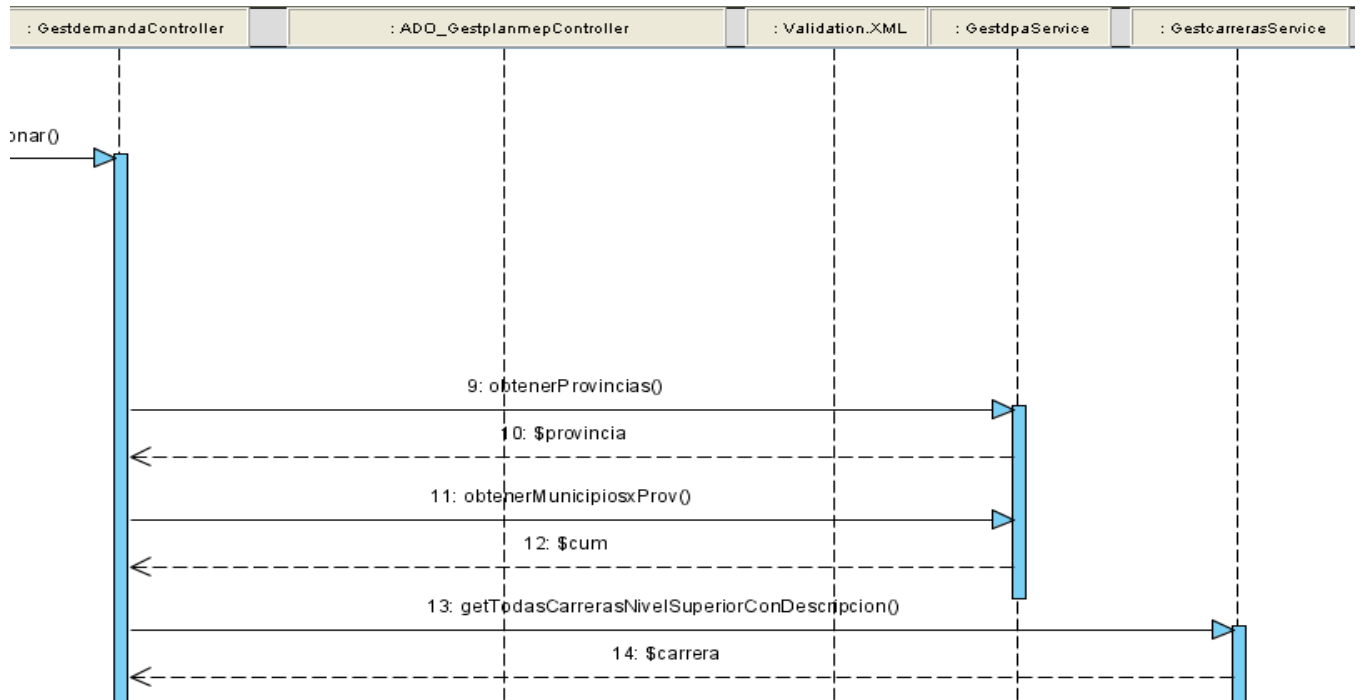
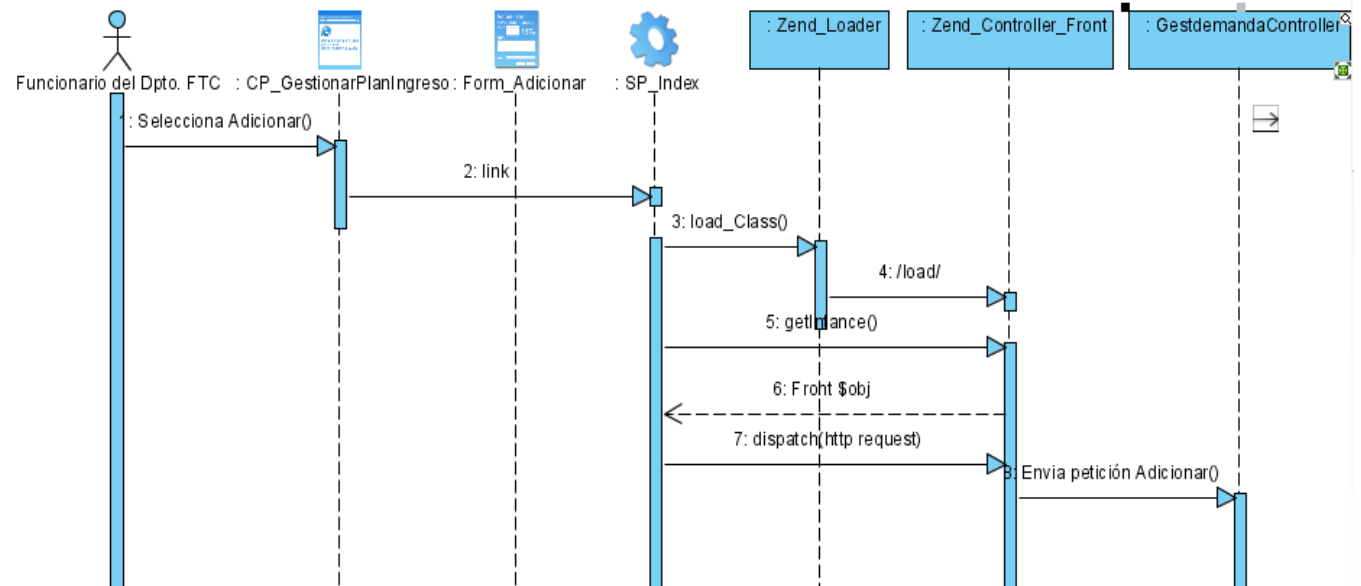
En el Anexo # 8 se encuentran los Diagramas del diseño de los casos de usos más significativos.

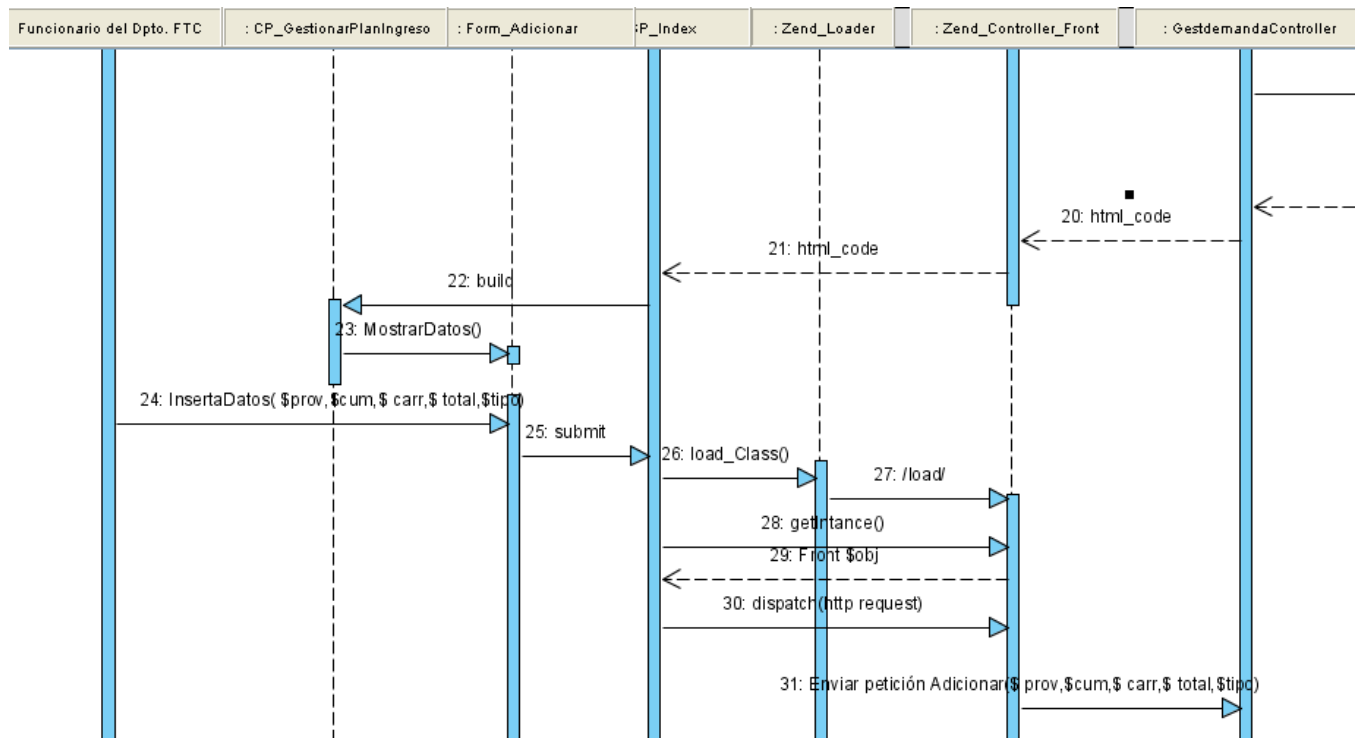
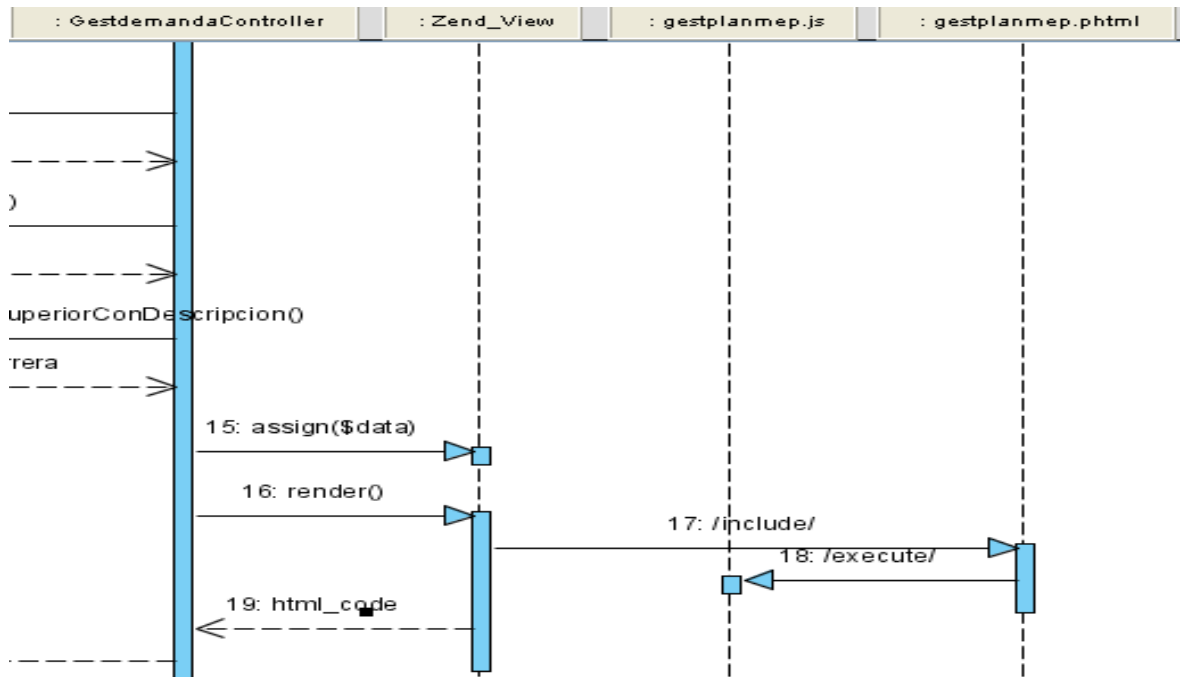
### 3.3.3 Diagramas de secuencias

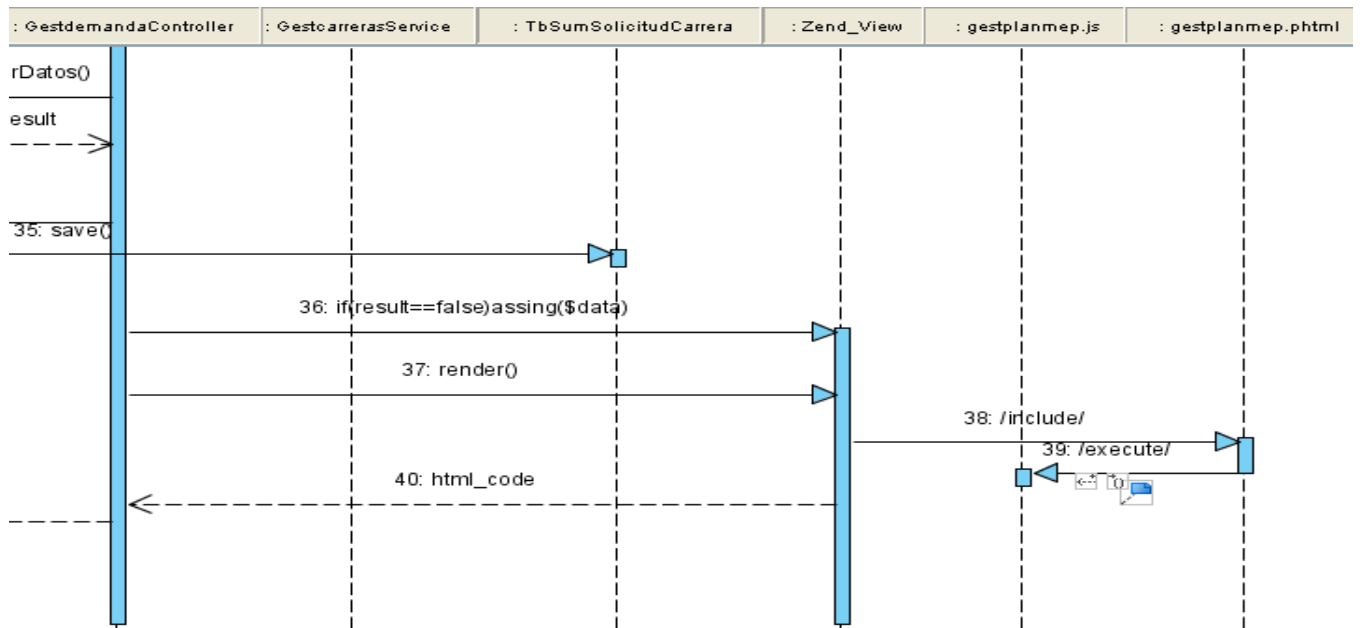
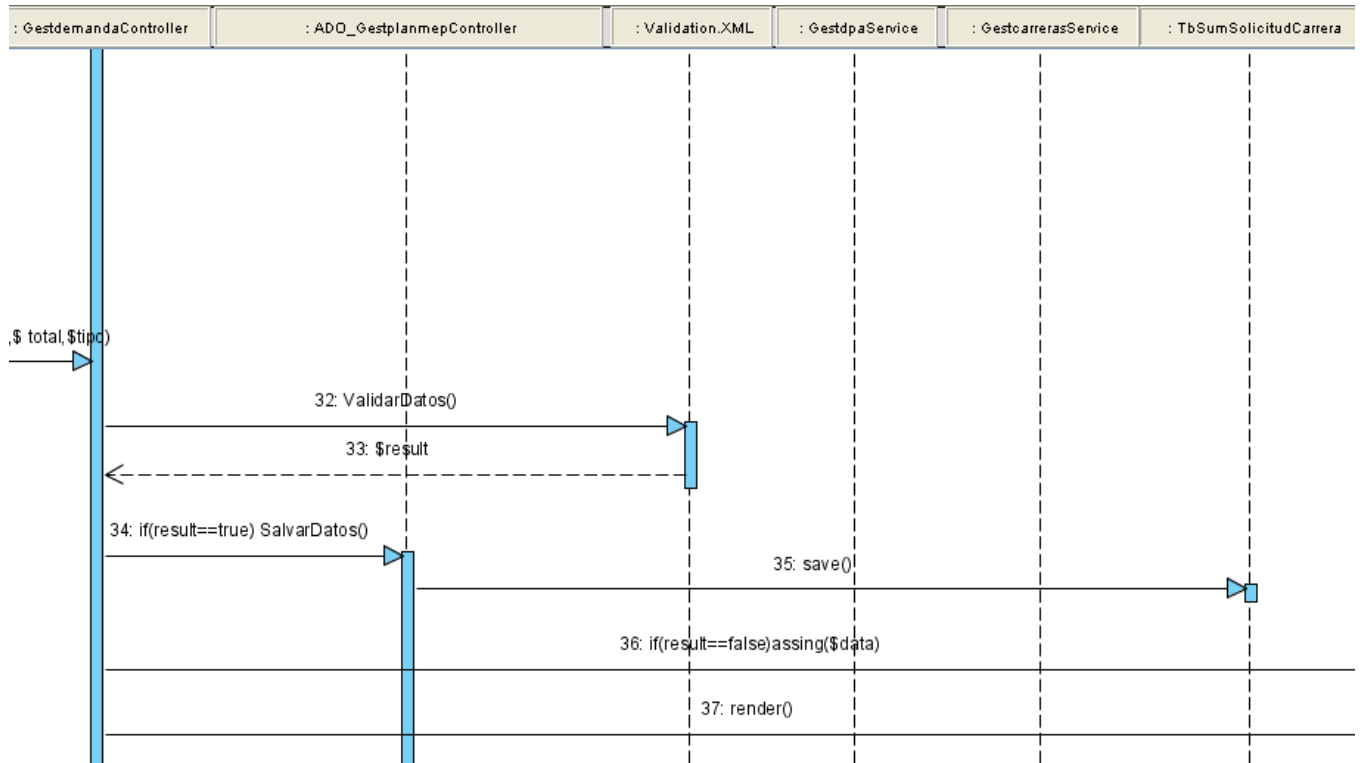
Los diagramas de secuencia son un tipo de diagrama de interacción. Estos diagramas muestran cómo se relacionan los objetos entre ellos mediante secuencias de mensajes que se envían entre sí.

A continuación se muestra el diagrama de secuencia del caso de uso Gestionar plan de ingreso (Sección Adicionar).

gestionar plan de ingreso (sección adicional)







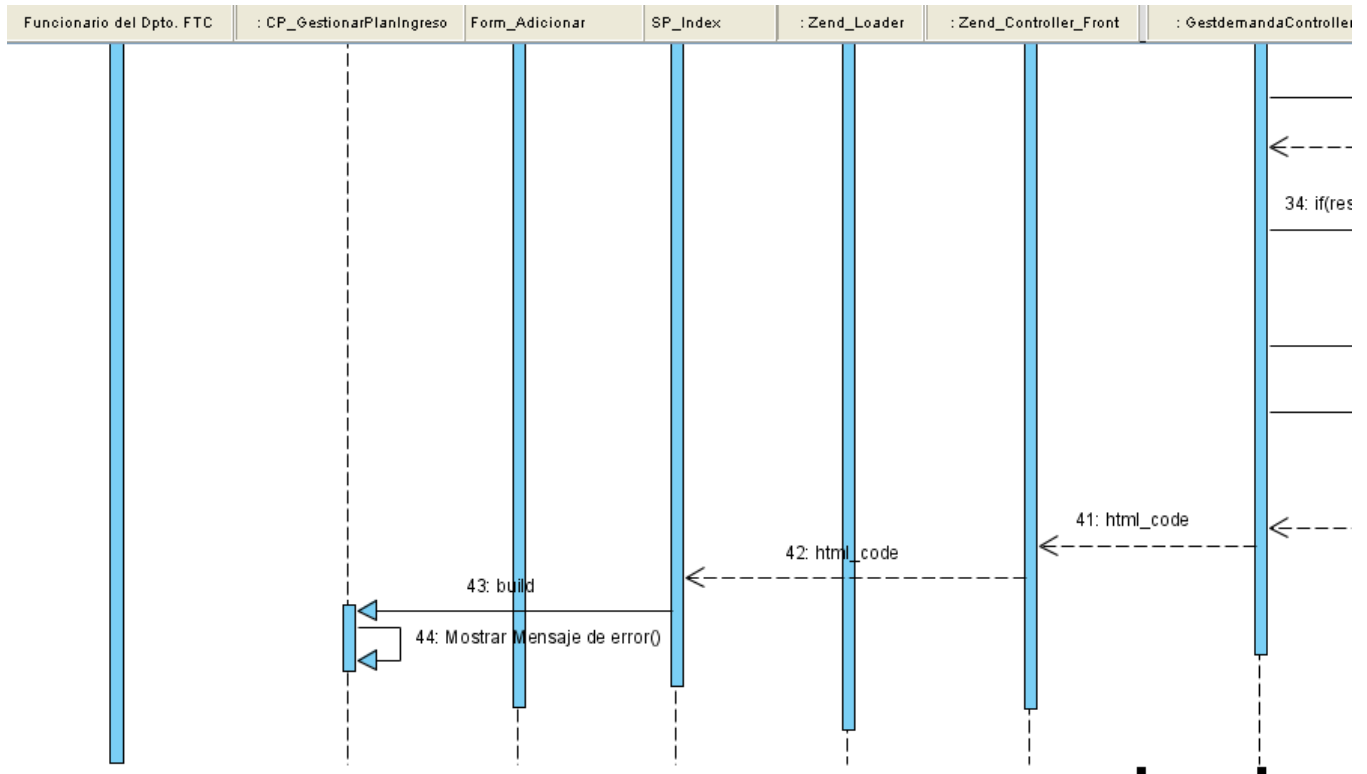


Figura # 7: Diagrama de secuencia del diseño.

En el Anexo # 9 se encuentran los Diagramas de secuencias de las secciones modificar, eliminar y crear.

### 3.4 Diseño de la base de datos

En el desarrollo de un sistema informático, el diseño de la BD es de gran importancia, ya que en ella se almacenan todos los datos que son necesarios en la modelación del problema que se desea resolver, además ésta es la fuente de obtención de toda la información que se quiera recuperar del sistema. Las bases de datos necesitan de una definición de su estructura que le permitan almacenar datos, reconocer el contenido, y recuperar la información.

El Modelo físico se puede encontrar en el Anexo # 10.

### Modelo lógico

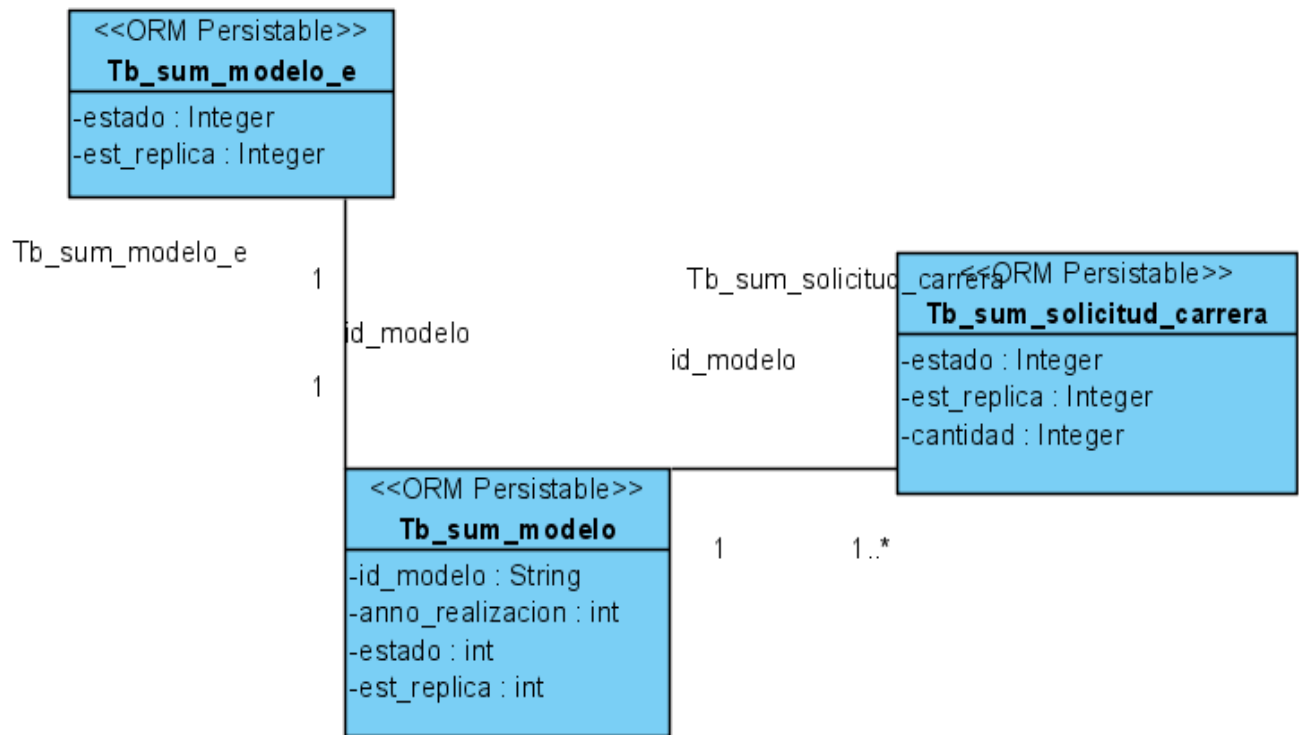


Figura # 8: Modelo Lógico de Datos

### 3.5 Conclusiones

En este capítulo se abordaron los aspectos básicos relacionados con el análisis y diseño del sistema, además se mostraron los diagramas correspondientes: los de clases del análisis y de clases del diseño así como los diagramas de colaboración y secuencia para cada una de las funcionalidades descritas. Se describe la arquitectura y los patrones que se tuvieron en cuenta para el diseño de los casos de uso. Además, se muestra el modelo lógico de la base de datos que es de gran importancia para la implementación del sistema.

## **Capítulo 4: Implementación y prueba**

### **4.1 Introducción**

Este capítulo muestra la construcción de los artefactos esenciales de los flujos de trabajo Implementación y Prueba. Se muestra el diagrama de despliegue, los diagramas de componentes y el modelo de prueba con sus respectivos casos de prueba.

### **4.2 Modelo de implementación**

El Modelo de implementación constituye la vista de implementación de la arquitectura, y como tal, guía las labores de construcción del sistema. Este contiene fundamentalmente los subsistemas de implementación, incluyendo las dependencias y otras informaciones necesarias para su utilización.

#### **4.2.1 Modelo despliegue**

Los diagramas de despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos los cuales están unidos por conexiones de comunicación. En la siguiente figura se muestra la distribución física del proyecto “GeForza” teniendo en cuenta la arquitectura del software y del hardware.



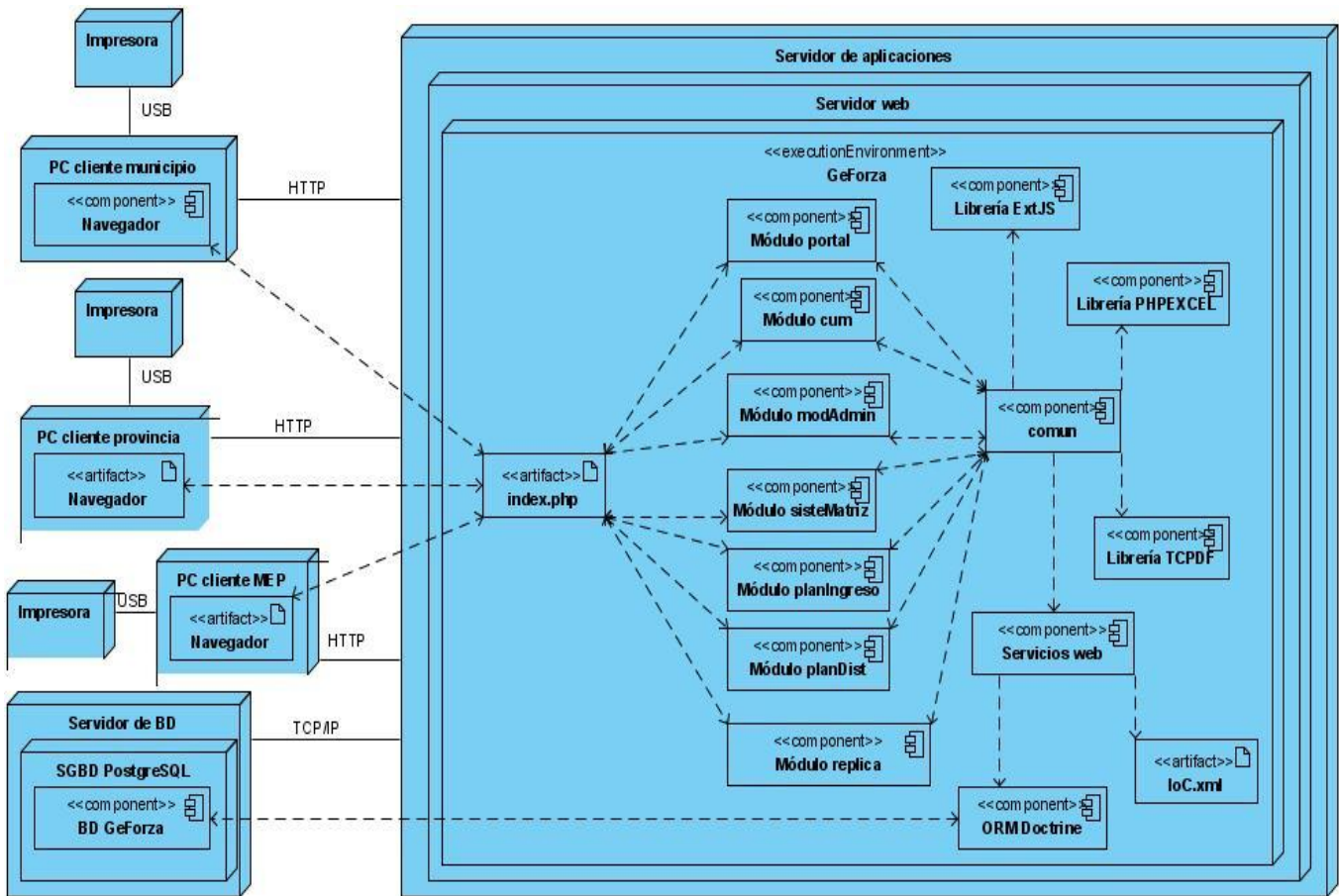


Figura # 9: Modelo de despliegue.

#### 4.2.2 Diagrama de componentes

Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes, además de ayudar a entender mejor el modelo de implementación. Los diagramas de componentes contienen componentes, interfaces y relaciones entre ellos. También puede contener paquetes utilizados para agrupar elementos del modelo. A continuación se podrá apreciar el diagrama de componentes del caso de uso del sistema Gestionar plan de ingreso, los demás se podrán ver en el Anexo # 10.

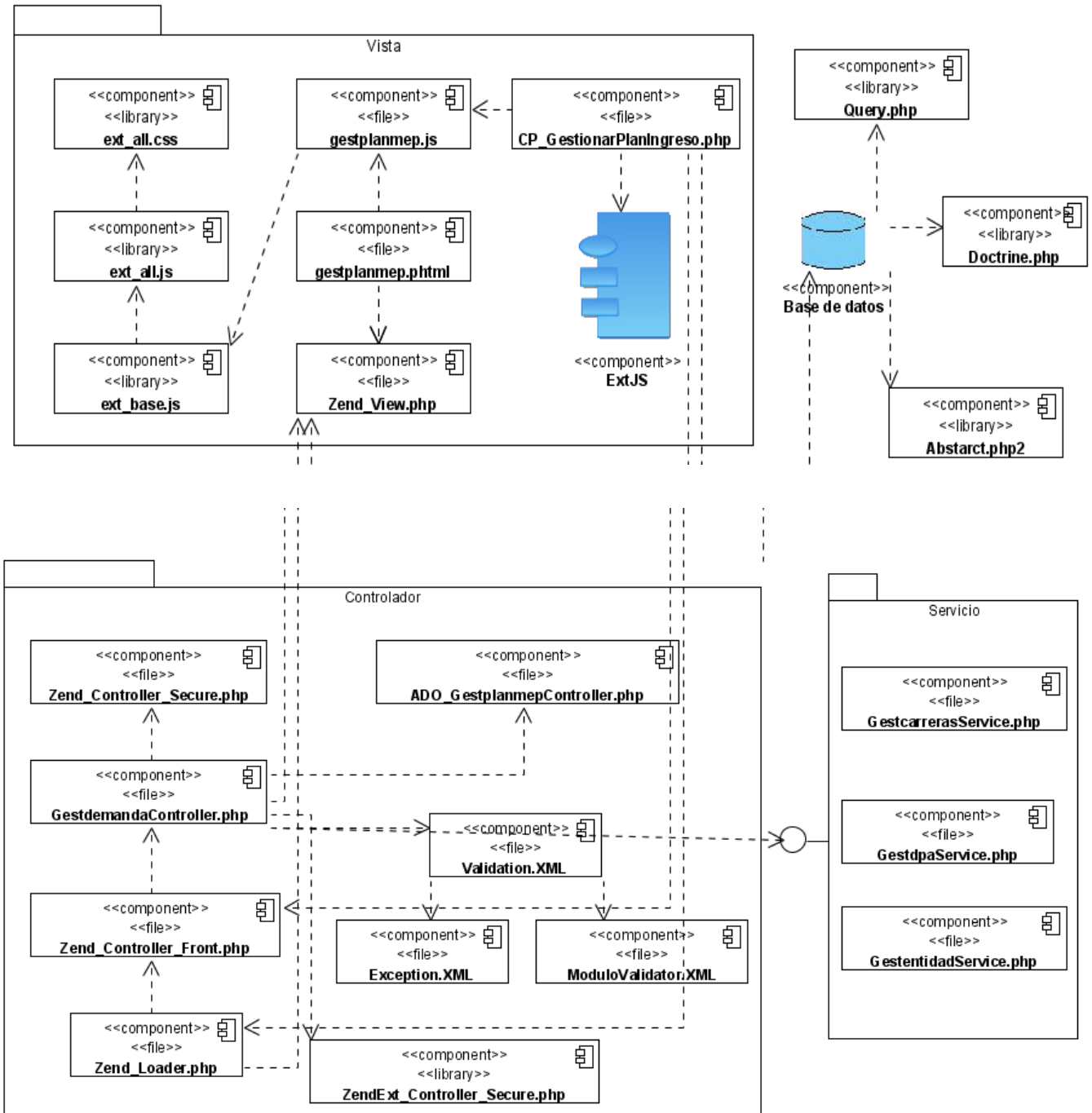


Figura # 10: Modelo de componentes.

### 4.3 Estándares de implementación

#### En la Vista

Para esta capa se definen como estándares los siguientes:

- ✚ En el nombre de las clases debe comenzar con gest en el caso de los casos de uso gestionar, seguido por el nombre del caso de uso. Ejemplo: gestsolicitud, gestplanmep.
- ✚ Teniendo en cuenta el componente que se utilizará el nombre del mismo estará definido de la siguiente forma:
  - Los botones se nombrarán: btn+Nombre. Ejemplo: btnAddSolicitud.
  - Los stores se nombrarán: st+Nombre. Ejemplo: stProv stOrg.
  - Los combobox se nombrarán: cb+Nombre. Ejemplo: cbProvincia.
  - Los texfields se nombrarán: txf+Nombre. Ejemplo: txfOrganismo.
  - Los gridpanels se nombrarán: gp+Nombre. Ejemplo: gpSolicitudes.
  - Las ventanas se nombrarán: win+Nombre. Ejemplo: winAddSolicitud.

#### En el Negocio

Para esta capa se define:

- ✚ El nombre de la clase comenzará con Gest seguida por el nombre si el caso de uso es un gestionar y la palabra Controller. Ejemplo: GestsolicitudController.
- ✚ El nombre de los atributos será en letra minúscula y en caso de ser más de una palabra se escribirá sin separación. Ejemplo: datosfinal.
- ✚ El nombre de las funciones será nombre seguido por la palabra Action. Ejemplo: cargarcomboprovinciaAction.
- ✚ Los servicios se nombrarán Gest seguido del nombre y la palabra Service. Ejemplo: GestorganismoService.
- ✚ Las validaciones se nombrarán Gest seguido del nombre y la palabra Validator. Ejemplo: GestorganismoValidator.

### En el Modelo

Para esta capa se utilizan:

- ✚ El nombre de las entidades se define por tb seguido de las siglas del módulo y el nombre separando cada una de las palabras por un “\_”. Ejemplo: tb\_ma\_organismos.
- ✚ El nombre de los atributos será en letra minúscula y en caso de ser más de una palabra serán separadas por un “\_”. Ejemplo: id\_código.

## 4.4 Pruebas

Las pruebas de software son un elemento crítico para la garantía de calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación. (13)

RUP define cuatro niveles de prueba: unidad, integración, sistema y aceptación.

Para la realización de este trabajo se usaran las pruebas unidad, ya que estas se realizan durante la fase de construcción, específicamente en el flujo de trabajo de implementación; las cuales se basan en probar los componentes implementados como unidades individuales.

### 4.4.1 Modelos de pruebas

Existen distintos métodos de pruebas, entre los cuales se encuentra Prueba de Caja Negra; prueba que se centra principalmente en los requisitos funcionales del software. Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la estructura de control, concentrándose en los requisitos funcionales del sistema y ejercitándolos. Se refiere a las pruebas que se llevan a cabo sobre la interfaz del software, por lo que los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce una salida correcta, así como que la integridad de la información externa se mantiene. Esta prueba examina algunos aspectos del modelo fundamentalmente del sistema sin tener mucho en cuenta la estructura interna del software.

Las secciones de prueba de los casos de uso más importantes se pueden encontrar en el Anexo # 12.

## Resultados de las pruebas

Luego de realizadas tres iteraciones de pruebas, en las cuales se detectó un total de 25 no conformidades, las cuales fueron resueltas en su totalidad al término de la tercera iteración, se puede concluir que las pruebas fueron exitosas, el sistema está apto para su utilización y cumple con todas las funcionalidades para las que fue concebido.

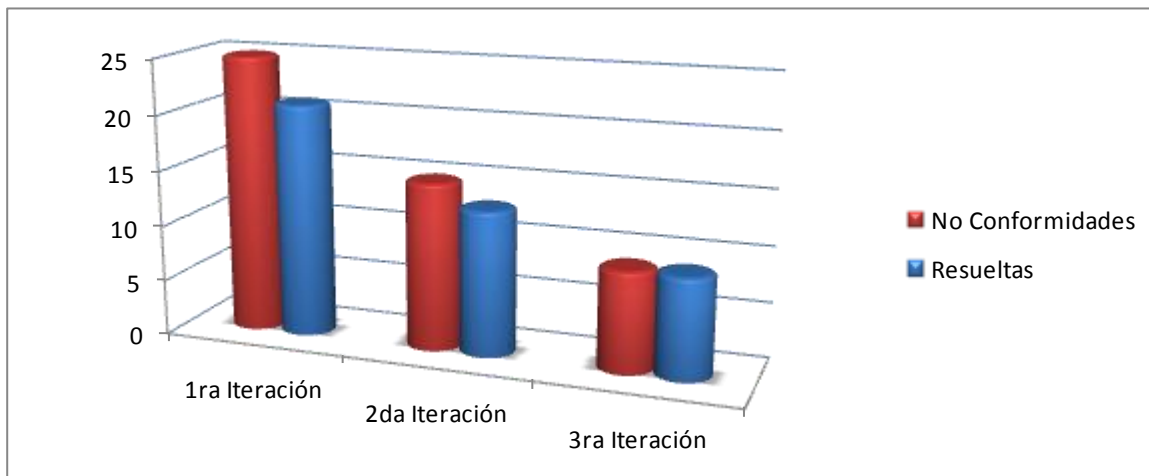


Figura # 11: Iteraciones de pruebas.

## 4.5 Conclusiones

En este capítulo se realizan los artefactos principales de los flujos de trabajo Implementación y Prueba: diagrama de despliegue, diagrama de componentes y modelo de prueba con sus respectivos casos de prueba, los cuales permitieron chequear cada una de las funcionalidades del sistema.

## Conclusiones

Con el desarrollo de este trabajo se arribaron a las siguientes conclusiones:

- ✚ El estudio de los procesos de ingreso a la Educación Superior en los Centros Universitarios Municipales en el Departamento de Fuerza de Trabajo Calificada del Ministerio de Economía y Planificación, permitió identificar los diferentes problemas que afronta el mismo en este sentido, lo cual validó la necesidad de la investigación y la actualidad del problema.
- ✚ Con el estudio de sistemas de gestión de la información existentes en Cuba y en el mundo, se pudo apreciar que ninguno brinda una solución a los problemas encontrados, ya que están desarrollados según las características de la organización a la que pertenecen, demostrándose la necesidad de implementar un nuevo sistema.
- ✚ La realización del análisis y diseño del módulo permitió que se obtuvieran todos los artefactos necesarios para la implementación del mismo, logrando así un mayor entendimiento de los procesos a la hora de ser implementados y servirá como base para futuras iteraciones del producto.
- ✚ Con la implementación de la solución propuesta, se obtuvo una versión entregable del módulo Plan de Ingreso a los CUM para el sistema GeForza, que permite realizar de forma automatizada los procesos de ingreso, a través de una interfaz agradable y fácil de utilizar, agilizando la gestión de la información generada en los mismos. Además, se desarrolló según las tendencias actuales del desarrollo de software, lo que lo hace más eficiente, seguro y de fácil mantenimiento y soporte.
- ✚ Luego de un conjunto de pruebas realizadas a la aplicación, se corrigieron las no conformidades detectadas, por lo que esta se encuentra lista para su liberación y posterior entrega al cliente.

Por todo lo anteriormente expuesto se concluye que la investigación fue completada con éxito cumpliéndose con el objetivo general de la misma.

## Recomendaciones

Al concluir esta investigación y teniendo en cuenta las experiencias obtenidas en la misma, se recomienda al Departamento Fuerza de Trabajo Calificada:

- ✚ Que se integre al Módulo Plan de Ingreso a la Educación Superior en los Centros Universitarios Municipales, la funcionalidad de consultar mediante algún servicio web brindado por una aplicación del Ministerio de Educación Superior, los resultados de las pruebas de ingreso realizadas por los solicitantes a los Centros Universitarios Municipales, con el objetivo de que con esta información se puedan elaborar reportes y estadísticas que ayuden a los especialistas del Ministerio de Economía y Planificación a confeccionar los planes de ingreso de cada uno de los Centros Universitarios Municipales con más exactitud.
- ✚ Que se actualice el generador de reportes dinámico utilizado en el sistema GeForza, ya que el mismo actualmente no genera los reportes de forma correcta en los formatos Excel o pdf. En caso de que no sea posible actualizar el generador de reportes, lo ideal sería utilizar el generador de reportes Agatha, el cual es una aplicación libre que es considerada a nivel internacional como la versión libre del generador de reportes Crystal Report de la empresa Microsoft.
- ✚ Actualizar las librerías y marcos de trabajo de la aplicación Geforza a las últimas versiones disponibles (ExtJS 3.2, Zend Framework, 1.10 y Doctrine 1.4), con el fin de lograr mejoras sustanciales en el rendimiento del sistema, así como la corrección de algunos errores que presentan las versiones en uso.

## Referencia bibliográfica

1. planificación, Ministerio de economía y. Indicaciones para la elaboración del Ingreso a los Centros Universitarios Municipales. 2008.
2. slideshare. [En línea] <http://www.slideshare.net/cindyjvera/conceptos-basicos-de-administracion-de-servicios-de-informacion>.
3. DICCIONARIO INFORMÁTICO. [En línea] <http://www.lawebdelprogramador.com/diccionario/mostrar.php?letra=l&pagina=3#>.
4. Giner, F. Javier Muñoz. dspace. [En línea] junio de 2004. <http://dspace.upv.es/xmlui/bitstream/handle/10251/2625/tesisUPV2175.pdf?sequence=3>.
5. La influencia de las tecnologías de la información y las comunicaciones en las Pymes del Sector Textil. [En línea] junio de 2004. <http://dspace.upv.es/xmlui/bitstream/handle/10251/2625/tesisUPV2175.pdf?sequence=3>.
6. CSI Centro de Servicios Informáticos. [En línea] 2008. <http://www.csi.espol.edu.ec/ui/es/content/sistema/sistema.aspx?op=toshow&id=105>.
7. Juan Antonio González García, J. SISTEMA INTEGRADO DE PLANIFICACIÓN ESTADÍSTICA (SINPLE). [En línea] 2008. [www.icane.es/.../ANDALUCIA/SISTEMA\\_INTEGRADO\\_DE\\_PLANIFICACION\\_ESTADISTICA\\_SIMPLE.d](http://www.icane.es/.../ANDALUCIA/SISTEMA_INTEGRADO_DE_PLANIFICACION_ESTADISTICA_SIMPLE.d)  
[OC](http://www.icane.es/.../ANDALUCIA/SISTEMA_INTEGRADO_DE_PLANIFICACION_ESTADISTICA_SIMPLE.d).
8. Zambrano, Ing. Donel Vázquez. Desarrollo de software con plazos de tiempo variables. [En línea] 1 de 9 de 2009. [www.gestiopolis.com/...estrategia/desarrollo-software-ciclo-de-vida.pdf](http://www.gestiopolis.com/...estrategia/desarrollo-software-ciclo-de-vida.pdf).
8. mozilla Sitio Web de la Comunidad Paraguaya. [En línea] 2009. <http://www.mozillaparaguay.org/?p=57>.
10. Vallecillo, Lidia Fuentes y Antonio. *FuentesUna Introducción a los Perfiles UML*. España: s.n., 2003.
11. Jacobson, James. *El Lenguaje Unificado de Modelado*. Madrid: s.n., 200.
12. Herramientas Case. [En línea] 2010. [www.slideshare.net/.../herramientas-case](http://www.slideshare.net/.../herramientas-case).
13. Pressman, Roger S. *Ingeniería del Software. Un enfoque práctico*. 2005.
14. Kernel Error . [En línea] 2009. [http://kernelerror.net/author/fecr\\_88/](http://kernelerror.net/author/fecr_88/).
15. rubenmauricio.com. [En línea] 18 de septiembre de 2007. <http://rubenmauricio.com/page/5/>
16. Quevedo, Juan Carlos Ramírez. [En línea] [www.eqaula.org/eva/file.php](http://www.eqaula.org/eva/file.php)
17. ATIX Numero 8. [En línea] 2009. [www.scribd.com/doc/13112003/ATIX-Numero-8](http://www.scribd.com/doc/13112003/ATIX-Numero-8)
18. Arzion. [En línea] 9 de diciembre de 200. [www.arzion.com/.../Instalacion-de-Doctrine-ORM-en-](http://www.arzion.com/.../Instalacion-de-Doctrine-ORM-en-)



19. [En línea] [www.perseo.cs.buap.mx/bellatrix/tesis/TES808.pdf](http://www.perseo.cs.buap.mx/bellatrix/tesis/TES808.pdf)
20. Aplicaciones Wb. [En línea] 2010. [www.blog.aplicacionesweb.cl/php.../php-version.../introduccion-a-php-5/](http://www.blog.aplicacionesweb.cl/php.../php-version.../introduccion-a-php-5/).
21. Taringa. [En línea] 2 de marzo de 2009. [www.taringa.net/posts/ebooks](http://www.taringa.net/posts/ebooks)
22. Programación Web y Tecnologías Informáticas. [En línea] 8 de marzo de 2008. [www.zenkius.blogspot.com/.../tecnologas-del-lado-del-cliente.html](http://www.zenkius.blogspot.com/.../tecnologas-del-lado-del-cliente.html) .

## Bibliografía

1. Arquitectura de Aplicaciones para .NET. [En línea] 10 de enero de 2009.  
<http://www.scribd.com/doc/209675/Arquitectura-de-Aplicaciones-para-NET>.
2. ExtJS 3.2.1 API Documentation . [En línea] 2009. <http://www.extjs.com/deploy/dev/docs/>.
3. García, Autores: Juan Antonio González. Ponencia: Sistema Informático Estadístico de planificación Estadística (SINPLE). [En línea] 2008.  
[www.icanes.es/.../SISTEMA\\_INTEGRADO\\_DE\\_PLANIFICACION\\_ESTADISTICA\\_SINPLE.doc](http://www.icanes.es/.../SISTEMA_INTEGRADO_DE_PLANIFICACION_ESTADISTICA_SINPLE.doc).
4. James Jacobson, y otros. El Lenguaje Unificado de Modelado. Madrid 2000: s.n., 2000.
5. Manual introducción a Javascript en Español . [En línea] noviembre de 2009.  
<http://www.taringa.net/posts/downloads/4036001/Manual-Introduccion-a-Javascript-en-Espa%C3%B1ol.html>.
6. Manual Online de Doctrine PHP. [En línea] 2008 de abril de 26.  
<http://www.phpdoctrine.org/documentation/manual>
8. Paradigm, Visual. [En línea] 10 de enero de 2009. <http://www.visual-paradigm.com/product/vpum/>.
10. Pressman, Roger S. *Ingeniería del Software. Un enfoque práctico*. 2005.
11. Qué es Javascript . [En línea] enero 2009 de 2009]. <http://www.desarrolloweb.com/articulos/25.php>.
12. Sanchez, Maria A. Mendoza. Metodologías De Desarrollo De software. [En línea] 2004.  
<http://kernelerror.net/programacion/php/arquitectura-3-capas/>
13. Sevilla), Juan Antonio González García (Universidad de. Ponencia: Sistema Informático de Planificación Estadística . [En línea] 2008.  
[www.icanes.es/.../SISTEMA\\_INTEGRADO\\_DE\\_PLANIFICACION\\_ESTADISTICA\\_SINPLE.doc](http://www.icanes.es/.../SISTEMA_INTEGRADO_DE_PLANIFICACION_ESTADISTICA_SINPLE.doc).
14. Revista de Software libre . [En línea] <http://www.scribd.com/doc/13112003/ATIX-Numero-8>
15. Sistema de Administración Académica Postgrado - SAAC. [En línea] 2008.  
<http://www.csi.espol.edu.ec/ui/es/content/sistema/sistema.aspx?op=toshow&id=105>.
16. Sitio oficial del IDE netbeans. [En línea] <http://netbeans.org/>
17. Torossi, Gustavo. *El Proceso Unificado de Software*.
18. Zambrano, Ing. Donel Vázquez. *Desarrollo de software con plazos de tiempo variables*. Ciudad de la Habana: s.n., 2009.
19. ZendFramework Tutorial. [En línea] <http://www.zftutorials.com/>.

## Glosario de términos

OACE: Organismos de Administración Central del Estado.

CAP: Consejo de Administración Provincial.

PP: Programa priorizado.

GC: Grupo de Carrera.

MEP: Ministerio de Economía y Planificación.

FTC: Fuerza de Trabajo Calificada.

GeForza: Sistema Unificado de la Gestión de la Fuerza de Trabajo Calificada.

RF: Requerimiento funcional.

RNF: Requerimiento no funcional.

DPA: División Político Administrativa.

MVC: Modelo Vista Controlador.

IDE: Entorno Integrado de Desarrollo.

SGBD: Sistema Gestor de Base de Datos.

UML: Lenguaje Unificado de Modelado.

UCI: Universidad de las Ciencias Informáticas

MAC: Mandatory Access Control. Control de Acceso Obligatorio por sus siglas en inglés.

DAC: Discretionary Access Control. Control de Acceso Discrecional por sus siglas en inglés.

RBAC: Role-based Access Control. Control de Acceso Basado en Roles.

Depto. FTC: Departamento de Fuerza de Trabajo Calificada.

BD: Base de Datos.

RUP: Proceso Unificado de Desarrollo.

ORM: Mapeador Relacional de Objetos.

POO: Programación Orientada a Objetos.

RUP: Proceso Unificado de Rational.

CUM: Centros Universitarios Municipales.

PI: Plan de Ingreso.

DOM: Document Object Model (una traducción al español no literal, pero apropiada, podría ser Modelo en Objetos para la representación de Documentos).