



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 2

**SISTEMA AUTOMATIZADO PARA LA OBTENCIÓN DEL
PLAN DE VUELO DEL IL-96/300**

Tesis presentada en la opción al grado de Ingeniero en Ciencias Informáticas.

Autor: Leydis Esther Garzón Giro.

Yolexis Vega Pérez.

Tutor: Darian Horacio Grass Boada.

Luís Ramón González Páez.

Ciudad de La Habana, Junio 2007.

Año 49 de la Revolución.

DECLARACION DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Facultad 2 de la Universidad de las Ciencias Informáticas (UCI²⁴) a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los 29 días del mes de Junio del año 2007.

Firma del Autor

Leydis E. Garzón Giro

Firma del Autor

Yolexis Vega Pérez

Firma del Tutor

Darián Horacio Grass Boada

Firma del Tutor

Luis Ramón González Paet

OPINIÓN DEL TUTOR DEL TRABAJO DE DIPLOMA

Título: “Sistema Automatizado para la Obtención del Plan de Vuelo del IL-96/300”

Autores: Leydis E. Garzón Giro

Yolexis Vega Pérez

El tutor del presente Trabajo de Diploma considera que durante su ejecución el estudiante mostró las cualidades que a continuación se detallan:

Por todo lo anteriormente expresado considero que el estudiante está apto para ejercer como Ingeniero Informático; y propongo que se le otorgue al Trabajo de Diploma la calificación de _____.

Firma del Tutor

Darián Horacio Grass Boada

Firma del Tutor

Luis Ramón González Paet

Fecha

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado "Sistema Automatizado para la Obtención del Plan de Vuelo del IL-96/300", fue realizado en La Universidad de las Ciencias Informáticas. Esta entidad considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

Totalmente

Parcialmente en un ____ %

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes (cuantificar):

Como resultado de la implantación de este trabajo se reportará un efecto económico que asciende a _____.

Y para que así conste, se firma la presente a los ____ días del mes de _____ del año _____.

Representante de la entidad

Cargo

Firma

Cuño

DEDICATORIA

Leydis:

A mis padres, con todo el amor que les profesó, porque gracias a ellos he llegado hasta aquí.

A toda mi familia que siempre ha soñado con que llegara este día y hoy estoy haciendo realidad ese sueño.

Yolexis:

A mi mamá y a mi papá por todo su esfuerzo y la confianza depositada en mí.

A mis hermanas, a mi hermano y a ti Zenia, por saber darme el apoyo que siempre he necesitado.

A mis sobrinos Yoendris y Kevin para que les sirva de guía en el camino a seguir.

A todos aquellos que aunque no están presentes les hubiese gustado disfrutar como yo de este momento inolvidable.

AGRADECIMIENTOS

A la Revolución cubana por habernos dado la oportunidad de estudiar y superarnos, por habernos permitido formarnos como profesionales.

A la Tripulación del IL-96 300 por su apoyo y atención, por su comprensión y por darnos la posibilidad de serles útil.

A todos los compañeros de Cubana que nos ayudaron y nos atendieron para el desarrollo de nuestro trabajo.

A Kiuver por su ayuda y apoyo en el momento que más lo necesitamos.

A Sheila por su incansable apoyo, por dedicar tantas horas a nuestra tesis y habernos impulsado desde el inicio.

A Luís Ramón y Darián por habernos guiado en la preparación de la tesis.

De Yolexis:

A mi mamá por su incansable lucha y apoyo, por su confianza en mí, y por enseñarme el verdadero sacrificio para llegar al triunfo.

A mi papá por su apoyo incondicional y por sus enseñanzas.

A mis hermanas y hermano por darme su ayuda y por estar siempre ahí.

A mi querida novia Zenia por toda su ayuda durante estos últimos tiempos, por haberme comprendido, soportado y por brindarme lo mejor de sí para poder llegar a este día.

A toda mi familia.

A Paula y a Paver por su hospitalidad y ayuda.

A mis compañeros inseparables de carrera, y a los que se nos unieron durante todo este tiempo.

A todos mis queridos amigos.

A todos aquellos que de una forma u otra aportaron su granito de arena para que este sueño se hiciera realidad.

A todos los que contribuyeron con mi educación, a mis profesores y maestros durante toda mi trayectoria de estudiante.

De Leydis:

A mis padres por tener fe en mí y enseñarme a luchar contra la adversidad. Por estar siempre a mi lado apoyando cada una de mis decisiones.

A toda mi familia y amigos de adolescencia, que aunque lejos, han sabido brindarme el apoyo necesario para salir adelante.

A Juana e Isabel que junto a sus familias fueron de gran ayuda y apoyo en el transcurso de mi carrera aquí La Habana.

A todos mis amigos, con los que he compartido cada instante de mi vida universitaria, los que han estado a mi lado para apoyarme en cada recaída y en especial a Maybel, Adisley y Maidolys.

A Kiuver por haberme enseñado a ver la vida desde otro punto de vista, a quien le debo parte de mi fortalecimiento espiritual. Por haber estado a mi lado en momentos muy difíciles, contribuyendo en gran medida a la realización de este trabajo.

A mis profesores, por todos los conocimientos transmitidos.

A mis compañeros de proyecto, por haber empeñado todo su esfuerzo en el desarrollo esta tarea.

En fin a todos aquellos que de una forma u otra me han apoyado en la realización de este proyecto.

Actualmente la tripulación de Cubana de Aviación que opera los aviones IL-96 serie 300 presenta algunas dificultades en los cálculos necesarios para la realización de sus vuelos. Para llevar a cabo un vuelo se requiere de una previa planificación del mismo. Por ello resulta de suma importancia para la tripulación la elaboración de un plan de vuelo confiable, que brinde los resultados en un breve periodo de tiempo, asegurando así la efectividad de los cálculos, cuya exactitud garantizaría la seguridad del vuelo.

Hasta el momento la tripulación no cuenta con un sistema eficiente e independiente que les permita obtener de forma automatizada los planes de vuelo en el menor tiempo posible y con el mínimo margen de error. Esto nos condujo al diseño y elaboración de un sistema de automatización para la gestión de Planes de Vuelo para aviones IL-96/300.

Se realizó un análisis comparativo de las tecnologías candidatas seleccionándose las más adecuadas a los requerimientos de los clientes. Se diseñaron y elaboraron los módulos internos correspondientes a la gestión de los datos y a la gestión de usuarios utilizando la metodología RUP (*Rational Unified Process* por sus siglas en inglés).

El sistema desarrollado facilita los cálculos del Plan de Vuelo para un IL-96/300 dando cumplimiento al objetivo principal de este trabajo, obteniendo mayor autonomía y evitando los posibles errores que pudieran cometerse en la confección manual del plan de vuelo. Dicho sistema reportará beneficios significativos e importantes para la tripulación de los aviones IL-96-300, lo que indica que es factible la puesta en práctica de este sistema, ahorrando una suma considerable de dinero al país.

INTRODUCCION	1
Capítulo I: Fundamentación Teórica.....	7
Introducción	7
1.1 Estado del arte.....	7
SAPLAV	7
SITA ¹ (Flight Briefing Service)	8
AeroPlaner.com	9
1.2 Tecnología.....	11
1.2.1 Tecnología .Net.	11
1.2.2 Arquitectura Framework.NET.....	11
1.2.3 Propiedades de la plataforma .NET	13
1.3 Lenguaje de programación	13
1.3.1 Lenguaje C#.	13
1.3.2 Principales características que identifican al lenguaje C#.	14
1.4 Arquitectura.....	15
1.4.1 Programación Multicapas.....	15
1.5 Metodología	17
1.5.1 El proceso unificado de desarrollo de software RUP	17
1.5.2 Características del Proceso Unificado.....	18
1.6 Herramientas.....	20
1.6.1 Rational Rose	20
Conclusiones	20
Capítulo II: Características del Sistema.....	21
Introducción	21
2.1 Objeto de estudio.....	21
2.1.1 Problema y Situación Problemática.....	21
2.1.2 Análisis crítico del negocio.	22
2.1.3 Objeto de automatización	22

2.1.4 Información que se maneja	23
2.2 Propuesta del sistema.....	23
2.3 Modelo de negocio.....	24
2.3.1 Breve Descripción Del Negocio	24
2.3.2 Actores Del Negocio	25
2.3.3 Trabajadores Del Negocio	25
2.3.4 Diagrama de Casos de Uso del Negocio.....	26
2.3.5 Descripción de los Casos de Usos en formato expandido.	26
2.3.6 Diagramas de Actividad.....	26
2.3.7 Modelo de Objeto.	26
2.4 Relación de Requerimientos	27
2.4.1 Requerimientos funcionales.	27
2.4.2 Requerimientos no funcionales.	29
2.5 Modelo de Casos de Usos del Sistema	32
2.5.1 Definición de los actores del sistema.	32
2.5.2 Diagrama de Casos de Uso del Sistema.....	33
2.5.3 Descripción textual de los casos de uso.....	33
Conclusiones	34
Capítulo III: Análisis y Diseño del Sistema.....	35
Introducción	35
3.1 Análisis.....	36
3.1.1 Diagrama de Clases del Análisis.	36
3.2 Diseño	37
3.2.1 Diagrama de Clases del Diseño por CU ²¹	37
3.3 Diseño de la Base de Datos.....	40
3.3.1 Modelo Lógico de Datos.....	40
3.3.2 Modelo Físico de Datos.	41
Conclusiones	42

Capítulo IV: Implementación.....	43
Introducción.....	43
4.1 Diagrama de Despliegue.....	43
4.2 Diagrama de Componentes.....	44
4.2.1 Componentes Paquete Actualizar Avión.....	44
4.2.2 Componentes Paquete Actualizar Aeropuerto.....	45
4.2.3 Componentes Paquete Actualizar Punto de Control.....	45
4.2.4 Componentes Paquete Actualizar Ruta.....	46
4.2.5 Componentes Paquete Actualizar Ruta.....	46
Conclusiones.....	46
Capítulo V: Estudio de Factibilidad.....	47
Introducción.....	47
5.1 Planificación basada en casos de uso.....	47
5.2 Análisis de costos y beneficios.....	53
Conclusiones.....	53
RECOMENDACIONES.....	55
ANEXOS.....	56

INTRODUCCION

Durante el periodo comprendido entre el año 2005-2006, y por efecto de varios convenios establecidos por nuestro país con la República de RUSIA, se adquirieron varios aviones comerciales IL-96 serie 300, con el objetivo de mejorar la flota de la empresa Cubana de Aviación, y dentro del marco del perfeccionamiento de los servicios aeroportuarios de Cuba. Con la adquisición de estos nuevos aviones, los cuales se puede decir, de última generación, se entrenaron y adiestraron las tripulaciones que operarían las nuevas aeronaves.

Para la operatividad de los nuevos equipos se necesitan una serie de cálculos, datos, e informes que en la actualidad son obtenidos de empresas internacionales, y en el peor de los casos realizados manualmente. Por solo citar un ejemplo, para que los avión IL-96/300 puedan realizar vuelos, se deben solicitar previamente los Planes de Vuelo a SITA¹ (Sociedad Internacional de Telecomunicaciones Aeronáuticas, con sede en Atlanta), pagando por ello 6 dólares por cada plan de vuelo, más un valor agregado de aproximadamente 10 dólares por concepto de comunicación. De ahí que surja la siguiente situación problemática.

Situación Problemática:

En la actualidad la tripulación de los IL-96 de Cubana de Aviación solicita los planes de vuelo a SITA¹, este servicio implica un gasto monetario significativo para el país y en ocasiones no puede ser accedido desde ciertos lugares. Hasta el momento no existe un sistema alternativo completamente autónomo y con la profesionalidad requerida que permita obtener de forma automatizada los planes de vuelo de estos aviones en el menor tiempo posible. Además, en situaciones en que no puede solicitarse el mencionado servicio por falta de conexión o alguna otra razón, los cálculos se realizan manualmente, haciendo el proceso engorroso y extenso, permitiendo que se comentan errores humanos.

Problema científico:

¿Cómo crear un sistema automatizado de gestión, totalmente autónomo, que facilite la obtención de planes de vuelo para la tripulación de los aviones IL-96/300?

Objeto de estudio:

Procesos automatizados de gestión de vuelo para la tripulación de aviones IL-96/300.

Campo de acción:

Proceso automatizado de obtención de los planes de vuelo para la tripulación de aviones IL-96 serie 300 de la empresa Cubana de Aviación.

Objetivos Generales:

El objetivo principal de este trabajo es desarrollar un sistema automatizado que facilite los cálculos del Plan de Vuelo para un IL-96/300 y brinde distintos servicios de reportes e información a usuarios.

Objetivos Específicos:

Como objetivos específicos se tienen:

Elaborar un sistema automatizado que facilite los cálculos para la realización de un plan de vuelo para aviones IL-96/300, permitiendo:

- Obtener autonomía en el proceso de elaboración del plan de vuelo.
- Evitar los posibles errores que se puedan cometer al realizar este proceso manualmente.
- Brindar reportes en línea a los despachadores sobre los distintos planes de vuelo elaborados.

Posibles resultados:

Lograr un módulo, dentro del sistema integrador de gestión de vuelos para aviones IL-96/300, que facilite a la tripulación el control y gestión de los diferentes planes de vuelo a los que pueden estar sometidos, teniendo la posibilidad de tomas de decisión.

Entre los aportes prácticos se espera obtener una planificación óptima del vuelo mediante la introducción de rutas y elementos de *performance* de la aeronave, brindando un informe del combustible necesario para el vuelo, las distancias, consumos y tiempos de arribo a los diferentes puntos de control de la ruta y los niveles de vuelo más óptimos, además de ejecutar el resultado preciso del análisis de los aeropuertos alternativos.

Tareas de la investigación:

- Realizar un estudio de la situación actual y las principales tendencias del tema.
- Estudiar los principios que rigen el proceso de elaboración de un plan de vuelo para aviones IL-96/300.
- Identificar las necesidades de los clientes.
- Seleccionar una arquitectura acorde a las condiciones tecnológicas de los clientes.
- Especificar los procesos que se van a implementar en el primer ciclo de desarrollo.
- Seleccionar las tecnologías de software a utilizar.

Esta investigación se apoyó en procedimientos que permitieron estudiar las características del objeto de la investigación que no son observables directamente. Estos procedimientos son conocidos como: métodos científicos de la investigación y están clasificados en diferentes categorías.

En el desarrollo de nuestro estudio utilizamos los Métodos Teóricos y los Métodos Empíricos.

Métodos Teóricos:

Los Métodos Teóricos posibilitan el conocimiento del estado del arte del fenómeno, su evolución en una etapa determinada y su relación con otros fenómenos. Dentro de sus clasificaciones utilizamos:

- **Analítico–Sintético:** Permite identificar la esencia de los fenómenos, los rasgos que los caracterizan y distinguen. Su objetivo dentro de esta investigación es analizar las teorías, documentos, etc. relacionados con el objeto de estudio; permitiendo la extracción de los elementos más importantes. Es importante destacar que la revisión bibliográfica resultó de gran importancia para la concepción del proyecto de investigación y para apropiarse de los conocimientos relacionados con el tema.
- **Modelación:** Este método posibilitó la creación de modelos (propuestas, alternativas, estrategias, etc.) con vista a investigar la realidad. Donde un modelo no es más que una reproducción simplificada de la realidad, cumple una función heurística, que permite descubrir y estudiar nuevas relaciones y cualidades del objeto de estudio.

Métodos Empíricos:

Permiten extraer de los fenómenos analizados las informaciones que se necesitan sobre ellos a través de observaciones, del uso de técnicas opináticas y de la propia experimentación. Dentro de sus clasificaciones utilizamos:

- **Entrevista:** Es una conversación planificada para obtener información. Su uso constituyó un medio para el conocimiento de los fenómenos a investigar. Las entrevistas fueron técnicas aplicadas entre los desarrolladores y clientes que participan en el proceso, con el objetivo de diagnosticar determinados aspectos de vital importancia relacionados con el software para establecer los flujos de trabajo necesarios para la lógica del mismo.
- **Observación:** Permitió tener un registro visual de lo que ocurre en una situación real, clasificando y consignando los hechos y acontecimientos del proceso.

Estructuración del contenido:

El presente trabajo consta de 5 capítulos que abordarán todo lo referente al estudio y realización del sistema, quedando estructurados de la manera:

Capítulo 1: Contiene la fundamentación teórica del trabajo, en el se exponen las técnicas, tecnologías, metodologías y software utilizados en los que se apoya la solución propuesta del problema. En el mismo se realiza un análisis crítico y valorativo de las tendencias actuales y el estado del arte del tema abordado en el trabajo, tanto a nivel nacional como internacional.

Capítulo 2: Este capítulo presenta las características del sistema propuesto, describiendo como este debe funcionar, entre otros rasgos significativos del sistema se especifica el objeto de automatización y la información que se maneja. En este capítulo se describen de forma general los procesos del negocio propuesto y las mejoras que propone el negocio actual indicando cómo estas solucionarán los problemas que originaron la situación problemática. También incluyen una listado de los requerimientos funcionales y no funcionales del sistema.

Capítulo 3: En este capítulo se representa el modelo de análisis y diseño con sus diagramas de interacción y el modelo de datos. En el modelo de análisis representamos una vista interna del sistema a partir de refinar los requisitos y estructurarlos en base a clases y paquetes. En el diseño se reflejan los objetos que interactúan para cumplir los requisitos y las restricciones relacionadas con los lenguajes de programación, componentes reutilizables, entre otras restricciones relacionadas con el entorno de implementación.

Capítulo 4: Este capítulo incorpora los diagramas de despliegue y componentes que conforman lo que se conoce como un modelo de implementación al describir los componentes a construir y su organización, y la dependencia entre los nodos físicos en los que funcionará la aplicación.

Capítulo 5: En este capítulo se realiza un estudio de factibilidad del proyecto a través de la planificación del tiempo de desarrollo, la cantidad de desarrolladores, el esfuerzo a realizar y el costo necesario para llevarlo a cabo; para esto se tienen en cuenta los beneficios tangibles e intangibles y el análisis de costo.

En el contenido del trabajo aparecen palabras en fuente cursiva especificando que pertenecen a otro idioma. Aquellos vocablos que puedan dar al traste con la comprensión del trabajo tendrán en superíndice un número, el cual indica su posición en el glosario de términos.

Capítulo I: Fundamentación Teórica.

Introducción

En el presente capítulo se fundamentan cada una de las técnicas, herramientas, metodologías de desarrollo y software utilizados en los que se apoya la solución propuesta del problema. Se brinda una panorámica actual de las soluciones existentes de la problemática planteada, tanto a nivel nacional como internacional, realizando un análisis crítico y valorativo de las tendencias actuales y el estado del arte del tema abordado en el trabajo.

1.1 Estado del arte

Analizando las tendencias y tecnologías más modernas de los productos de software existentes en el mercado que dan solución a problemas similares encontramos varios sistemas de los que brindamos algunas características a continuación.

Nacional:

SAPLAV

SAPLAV es un sistema automatizado desarrollado en la CUJAE (Ciudad Universita José Antonio Echeverría), en 1993, que tiene como objetivo fundamental la obtención del plan de vuelo por una ruta determinada. Fue diseñado para aviones YAK-40, YAK-42, AN-26 e IL-62, es utilizado aun, a pesar de no cumplir con la totalidad de los requerimientos actuales. Este sistema requiere para sus cálculos tener archivadas en la computadora: las rutas, las tablas de *performance* del equipo en el cual se realizará el vuelo y los datos específicos del avión (Ej.: MTOW¹⁷, MLW, MZFW⁵, Capacidad máxima de combustible).

Entre sus principales opciones está el trabajo con rutas, dígame crear, modificar y eliminar, invertir, unir y transformar. También actualiza la base de puntos de control, permitiendo crear, modificar y eliminar un punto de control. Posibilita visualizar los equipos almacenados con su matrícula y datos específicos,

dando la opción de introducir un nuevo equipo o modificar-eliminar uno ya existente. El sistema capta los datos necesarios para confeccionar, inmediatamente después, el plan de vuelo, dando la opción al usuario de imprimirlo y archivarlo.

El acceso al sistema está dividido en 2 categorías y cada persona tiene una clave propia para autenticarse en el sistema. La primera categoría tiene acceso a todas las opciones y la segunda cumple con las siguientes restricciones:

- No puede crear ni modificar rutas fijas.
- No puede transformar rutas.
- No puede modificar los datos de *performance* de aviones.

Internacional:

SITA¹ (Flight Briefing Service)

En la actualidad, para los aviones YAK42, IL62M e IL96-300, Cubana de Aviación genera los Planes de Vuelo de forma automatizada a través de los servicios contratados de *Flight Briefing Service* por la empresa internacional SITA¹ INC (*Information Networking Computing*) quien es la principal proveedora a nivel mundial de aplicaciones para el transporte aéreo, comunicaciones e infraestructura IT.

Los servicios integrados de *Flight Briefing* ofrecen todas las herramientas requeridas para el planeamiento óptimo y exacto del vuelo. Propone la mejor ruta a seguir teniendo en cuenta el análisis del funcionamiento del avión, los datos más actuales del tiempo y las validaciones de la ruta según los requerimientos del ATC (Control de tránsito aéreo). Los clientes pueden también crear, manejar y mantener su propia base de datos de ruta según sus requisitos específicos. La aplicación de elementos de coste en el análisis conduce al proceso de planeamiento del vuelo a determinar con mayor exactitud el menor coste del plan de vuelo.

Teniendo en cuenta la variación de los precios de combustible en los diferentes aeropuertos del mundo, resulta fácil comprender porqué las líneas aéreas llevan a menudo combustible adicional para reducir al mínimo el costo de combustible en los aeropuertos costosos. Sin embargo, el combustible adicional significa peso adicional, y por consiguiente un aumento en el combustible de quemada¹¹, potencialmente erosionando cualquier ventaja. Por esto SITA¹ contiene un módulo de *tankering* donde determina dinámicamente los factores dominantes para determinar el coste verdadero de llevar combustible adicional, que junto a un resumen completo de ganancias y pérdidas, puede incluirse en el plan del vuelo.

Uno de los servicios más valiosos de SITA¹ es el *Surface Weather and NOTAM Service* el cual proporciona una base de datos del tiempo vigente en cada aeropuerto actualizada constantemente, unido a las advertencias de las condiciones meteorológicas (SIGMET), y la información de NOTAM³, además del acceso a una base de datos actualizada de más de 7.300 pistas de 2.600 aeropuertos de todo el mundo. El uso de las opciones avanzadas del filtro y la alineación exacta a la ruta volada permiten al piloto obtener solo los datos necesarios para su vuelo.

AeroPlaner.com

AeroPlaner.com es un servicio Web basado en el planeamiento del vuelo que requiere solamente de un Web *browser*. Puedes planear las rutas usando las funciones de *auto-routing*, o puedes describir cada punto de navegación basado en los aeropuertos, *navaids*²⁹, nombres de ciudad o apenas señalando un punto en el mapa. Todos los procedimientos terminales producidos para el vuelo por instrumento y las cartas de navegación son accesibles *on-line* con tu ruta dibujada.

Muchos servicios de AeroPlaner.com son libres o disponibles a un pequeño coste entre los que se incluyen el directorio de aeropuerto, los procedimientos terminales, las transmisiones de datos del tiempo, el planificador rápido del vuelo (*Quick Flight Planner*) y los materiales de referencia.

Recolecta algunos de los mejores *link* del tiempo para los pilotos en Internet. Toda la matemática dura se hace para el usuario. Información de FBO³² y precios de combustible por aeropuertos actualizada diariamente, ordenada por tipo de combustible, localización y precio.

AeroPlaner.com controla el acceso de los usuarios permitiéndoles autenticarse en el sitio accediendo solo a la información permitida para cada membresía. Los usuarios se encuentran distribuidos en 3 categorías:

- *Basic Membership*
- *Premium Membership*
- *Premium Plus Membership*

Entre las principales funciones que brinda AeroPlaner.com se encuentran:

Basic: Estudiar NOTAMS³ actual y las condiciones atmosféricas. Descargar las cartas completas o porciones (costo pequeño para cada transferencia directa).

Premium: Planear con *Pro Fligth Planner* que proporciona mejoras tales como perfeccionamiento en el plan DUATS, *auto-routing*, perfiles del terreno, recubrimiento del radar de tiempo, entre otras. Descargar e imprimir los libros de procedimientos terminales.

Premium Plus: Permite el acceso libre a los 3 planificadores de vuelo (*Basic Fligth Planner, Pro Fligth Planner, New Pro Fligth Planner*) y libres transferencias directas de cartas completas para SECTIONAL, WACs³³, IRF alto y IRF bajo.

Funcionalidades de los planificadores de vuelo:

Basic Fligth Planer. Permite que los usuarios hagan un plan de vuelo basado en aeropuertos, nav aids²⁹, o ciudades conocidas. Puedes utilizar las cartas de Sectionals o WACs³³. Calcula headings, los vientos, tiempo y el combustible. Solo debes saber los aeropuertos, VORS³¹ o aun el código postal a donde deseas volar. Notifica si haz trazado un curso con un área restringida o un MOA. Genera un pdf con el informe del registro de la navegación listo para imprimir.

Pro Flight Planer: Agrega mucho más funcionalidades al planificador básico. Permite que un usuario inserte o suprima puntos del centro de registro de la navegación y pone al día la lista automáticamente. Crear *waypoints* y anotarlos en la carta. Modificar el tamaño del mapa. Importa los planes de vuelo, o imprimir el plan y enviarlo por telefax. Utilizar cartas IFR²⁷ bajas y altas para tu planeamiento. Incluye la capacidad de exportar tu ruta a una variedad de formatos.

1.2 Tecnología

1.2.1 Tecnología .Net.

Microsoft.NET es el conjunto de tecnologías en las que Microsoft ha estado trabajando durante los últimos años, en realidad no es algo radicalmente nuevo, sino que es un conjunto de tecnologías dispersas, que en muchos casos ya existían. El objetivo de estas tecnologías es obtener una plataforma sencilla y potente para distribuir el software en forma de servicios, de forma tal que puedan ser suministrados remotamente, comunicándose y combinándose unos con otros totalmente independientes de la plataforma, lenguaje de programación y modelo de componentes con los que hayan sido desarrollados. Es sin duda alguna un nuevo tipo de servicios a los cuales se les puede definir como de tercera generación.

.NET ofrece un entorno de desarrollo de aplicaciones llamado *Visual Studio .NET* que consta de varios lenguajes de programación como Visual Basic .NET, Visual C#, Visual FoxPro y Visual C++ .NET. Estos lenguajes combinan las características de los lenguajes existentes con nuevas posibilidades para proporcionar un potente sistema de desarrollo. A continuación, se detallan algunas de las características de la Arquitectura .NET.

1.2.2 Arquitectura Framework.NET.

El objetivo de esta arquitectura es la de reducir la complejidad en el desarrollo de este tipo de aplicaciones, permitiendo a los desarrolladores centrarse en escribir la lógica específica del servicio a desarrollar.

La nueva tecnología de Microsoft ofrece soluciones a los problemas de programación actuales, como son la administración de código o la programación para Internet. Para aprovechar al máximo las características de .Net es necesario entender la arquitectura básica en la que esta implementada esta tecnología y así beneficiarse de todas las características que ofrece.

El *Framework* de .Net es una infraestructura sobre la que se reúne todo un conjunto de lenguajes y servicios que simplifican enormemente el desarrollo de aplicaciones. Mediante esta herramienta se ofrece un entorno de ejecución altamente distribuido, que permite crear aplicaciones robustas y escalables. Los principales componentes de este entorno son:

- Lenguajes de compilación
- Biblioteca de clases de .Net
- CLR (*Common Language Runtime*)

.NET framework: Es la parte más importante de la plataforma .NET. Incluye COM+, un entorno de ejecución común, un compilador JIT, y un conjunto de librerías de sistema que dan acceso a un amplio conjunto de servicios.

Servidores .NET: Son un conjunto de aplicaciones que pueden usarse en conjunción con el *.NET framework* para facilitar el desarrollo de aplicaciones empresariales. Como por ejemplo SQL Server 2000, Exchange 200 Server o BizTalk Server 2000.

La biblioteca de clases de *.Net Framework* incluye, entre otros, tres componentes claves:

- Visual C#.NET para construir aplicaciones y servicios Web.
- Windows Forms para desarrollar interfaces de usuario.
- ADO.NET para conectar las aplicaciones a bases de datos.

1.2.3 Propiedades de la plataforma .NET

Portabilidad

La portabilidad es la capacidad que tiene un sistema de ser utilizado en distintas plataformas, Microsoft continúa con su voluntad de apoyar a Windows, por lo que, por supuesto Microsoft .NET funciona únicamente en plataformas basadas en Win32.

Escalabilidad

La escalabilidad es la capacidad de un sistema de incrementar sus prestaciones en función del número de usuarios simultáneos que lo utilizan. El .NET ofrece métodos de escalabilidad como la carga balanceada que permite a un *cluster* de servidores colaborar y dar un servicio de forma simultánea.

Microsoft .NET ofrece una solución más barata, con mayor rendimiento, escalabilidad y más fácil de implantar. Tiene en su contra recibir el apoyo de una única empresa – aunque ésta sea Microsoft –, no tener la suficiente madurez en el mercado por lo que no está bastante probado y la falta de portabilidad.

1.3 Lenguaje de programación

1.3.1 Lenguaje C#.

Algunos de los lenguajes más utilizados en el campo de la ingeniería y la programación de sistemas son C y C++, pues proporcionan el nivel de abstracción preciso para construir una aplicación compleja, y ofrecen mecanismos de bajo nivel que permiten a los programadores hacer uso de las características más avanzadas de las plataformas sobre las que se ejecutan sus programas. Por el contrario, Java está lastrado por su compromiso con la portabilidad y es inherentemente ineficiente. Microsoft ha creado C# que combina algunas de las características más avanzadas de Java con algunas de las más potentes de C y C++, construido especialmente para adaptarse de manera natural al *framework*.

Ha sido diseñado específicamente para ser utilizado en la plataforma .NET, careciendo de elementos innecesarios en esta plataforma, por lo que programarla usando C# es mucho más sencillo e intuitivo que hacerlo con cualquiera de los otros lenguajes.

La sintaxis y estructuración de C# es muy similar a la C++, ya que la intención de Microsoft con C# es facilitar la migración de códigos escritos en estos lenguajes a C# y facilitar su aprendizaje a los desarrolladores habituados a ellos. Sin embargo, su sencillez y el alto nivel de productividad son equiparables a los de Visual Basic.

1.3.2 Principales características que identifican al lenguaje C#.

Sencillez: Elimina muchos elementos que otros lenguajes incluyen y que son innecesarios en .NET.

Modernidad: Incorpora en el lenguaje elementos que son muy útiles para el desarrollo de aplicaciones y que en otros lenguajes hay que simular.

Orientación a objetos: Es más puro pues no admiten funciones ni variables globales, todo el código y datos han de especificarse dentro de definiciones de tipos de datos, lo que reduce problemas por conflictos de nombres y facilita la legibilidad del código.

Orientación a componentes: Su sintaxis incluye elementos propios del diseño de componentes que otros lenguajes tienen que simular mediante construcciones más o menos complejas.

Seguridad de tipos: Incluye mecanismos que permiten asegurar que los accesos a tipos de datos siempre se realicen correctamente.

Eficiencia: El código incluye numerosas restricciones para alcanzar su seguridad y no permite el uso de punteros. A diferencia de Java, en el mismo es posible saltarse dichas restricciones manipulando objetos a través de punteros.

1.4 Arquitectura

1.4.1 Programación Multicapas.

Siguiendo la filosofía del modelo actual de desarrollo del software, para la realización del sistema se propone organizar los elementos de la aplicación en componentes independientes buscando alcanzar una mayor efectividad a la hora de administrarlos.

Para ello se seguirá los fundamentos de la programación en múltiples capas ya que esta además de facilitar una administración eficiente de los componentes que la integran, proporciona rapidez a todas las funcionalidades Cliente-Servidor y la magnitud de la aplicación lo exige. Tal y como plantea la arquitectura de esta tecnología, los componentes primarios de la aplicación serán divididos y programados por separados y en tiempo de ejecución serán unidos. De forma tal que si una de las capas definidas sufre cambios, no se vean afectados el resto de las capas ni el resultado final del producto.

Se definen tres capas (Figura 1), tal es el caso de: Presentación, Reglas del Negocio y Acceso a Datos.

Arquitecturas de tres capas

El principio que está detrás de las arquitecturas de multicapas (o de tres como caso particular) se puede resumir en la conocida frase 'divide y vencerás'. Está claro que la descomposición de un problema en subproblemas de menor complejidad facilita la solución del primero, pero del modo en que se efectúe esta descomposición dependerá el logro de los resultados esperados.

Una buena arquitectura de software debe facilitar los requerimientos de mantenimiento, reusabilidad, escalabilidad, y robustez del mismo. Al componer la solución de un problema como una serie de capas, cada capa debe ocuparse de un subconjunto de responsabilidades fuertemente acopladas y tener poca cohesión con las demás. Los cambios internos en cualquier capa deben ocasionar la menor cantidad posible de cambios en las restantes.

La arquitectura adoptada para este proyecto presenta las capas de: datos, negocio y presentación. Una ventaja evidente de este modelo es que la capa de presentación puede desarrollarse de variadas maneras simultáneas, digamos; cliente Web, aplicación Windows, aplicación para otro Sistema Operativo, etc.; en nuestro caso la capa de presentación se desarrolla como una aplicación Windows; mientras menos responsabilidades recaigan en esta capa tanto mayor será la facilidad de desarrollar múltiples versiones de la misma. Otra ventaja sería la posibilidad de migrar de servidor de bases de datos con un mínimo de cambios en el sistema, en tal caso, estos se concentrarían en la capa de datos, quizás hubiera que hacer pequeños ajustes en la capa de negocio, pero nunca en la capa de presentación.



Figura 1: Aplicaciones de tres capas

- Presentación

En esta capa se diseña todo lo que constituye la interfaz gráfica y la interacción del usuario con el sistema.

- Reglas del Negocio

Contiene todas las subrutinas creadas con el propósito de regular alguna acción del usuario.

- Acceso a Datos

En esta capa se programa todo lo que tiene que ver con el acceso a la base de datos. Esta capa queda encargada de tomar la información de la base de datos dada una petición de la capa de Reglas del Negocio, que a su vez es generada por la capa de presentación.

Ventajas del modelo

- Desarrollos paralelos (en cada capa)
- Aplicaciones más robustas debido al encapsulamiento
- Mantenimiento y soporte más sencillo (es más sencillo cambiar un componente que modificar una aplicación monolítica)
- Mayor flexibilidad (se pueden añadir nuevos módulos para dotar al sistema de nueva funcionalidad)
- Alta escalabilidad. La principal ventaja de una aplicación distribuida bien diseñada es su buen escalado, es decir, que puede manejar muchas peticiones con el mismo rendimiento simplemente añadiendo más hardware. El crecimiento es casi lineal y no es necesario añadir más código para conseguir esta escalabilidad.

1.5 Metodología

1.5.1 El proceso unificado de desarrollo de software RUP

El Proceso Unificado es 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 (Figura2). Más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, tipos de organizaciones, niveles de actitud y tamaños de proyecto. Está basado en componentes, lo cuál quiere decir que el sistema software en construcción está formado por componentes software interconectados a través de interfaces bien definidas. Utiliza el Lenguaje Unificado de Modelado (*Unified Modeling Language*, UML) para preparar todos los esquemas de un sistema software. Garantiza la elaboración de todas las fases de un producto de software orientado a objetos. UML es un lenguaje que permite modelar sistemas con tecnología orientada a objetos.



Figura 2: Un proceso de desarrollo de software.

1.5.2 Características del Proceso Unificado

Los verdaderos aspectos definitorios del Proceso Unificado, y que lo convierten en único, se resumen en tres frases clave - dirigido por casos de uso, centrado en la arquitectura, e iterativo e incremental.

- Dirigido por los casos de uso:

Teniendo en cuenta que la razón de ser de un sistema es brindar servicios a los usuarios, RUP¹ define caso de uso como el conjunto de acciones que debe realizar un sistema para dar un resultado de valor a un determinado usuario y los utiliza tanto para especificar los requisitos funcionales del sistema, como para guiar todos los demás pasos de su desarrollo, dígame diseño, implementación y prueba.

- Estar centrado en la arquitectura:

La arquitectura es una vista del diseño completo con las características más importantes, dejando a un lado los detalles. Esta no solo incluye las necesidades de los usuarios e inversores, sino también otros aspectos técnicos como el hardware, sistema operativo, sistema de gestión de base de datos, protocolos de red; con los que debe coexistir el sistema. En otras palabras, la arquitectura representa la forma del sistema, la cual va madurando en su interacción con los casos de uso hasta llegar a un equilibrio entre funcionalidad y características técnicas.

- Ser iterativo e incremental:

La alta complejidad de los sistemas actuales hace que sea factible dividir el proceso de desarrollo en varios mini-proyectos. Cada uno de estos mini-proyecto se les denomina iteración y pueden o no representar un incremento en el grado de terminación del producto completo. En cada iteración los desarrolladores seleccionan un grupo de casos de uso, los cuales se diseñan, implementan y prueban. La planificación de iteraciones hace que se reduzcan los riesgos de los costes de un solo incremento, no sacar al mercado un producto en el tiempo previsto, mantener la motivación del equipo pues puede ver avances claros a corto plazo y que el desarrollo pueda adaptarse a los cambios en los requisitos.

¹ *Rational Unified Process*

El alcance y complejidad de los sistemas informáticos que se desarrollan hoy en día, hace necesario el uso de una metodología de desarrollo que permita organizar y controlar los procesos de su producción y mantenimiento. En este sentido ha habido muchas propuestas, teniendo gran impacto en la actualidad el proceso unificado de desarrollo de software (RUP).

RUP, es una metodología basada en un pequeño grupo de principios claves: el equipo de un proyecto de software debe planificar el desarrollo; debe conocer hacia donde se dirige; debe documentar el proyecto de una manera perdurable y extensible. RUP además incorpora el concepto de "mejores prácticas" para la ingeniería de software, definido por cinco características fundamentales [Evans]:

1. Dirigido por casos de uso. El desarrollo está dirigido a satisfacer las necesidades de los usuarios del sistema expresadas en casos de uso.
2. Centrado en la arquitectura. El desarrollo se centra en una arquitectura bien definida, con relaciones claras entre sus distintos componentes.
3. Iterativo. El problema y la solución se organizan en pequeñas piezas, de manera que cada iteración se dirige específicamente al desarrollo de un conjunto de ellas.
4. Incremental. Cada iteración se construye sobre la base creada por las iteraciones anteriores, agregándole capacidades al sistema.
5. Controlado. El proceso se planifica y en cada momento está claro lo que debe hacerse.

1.6 Herramientas

Se hizo uso de las herramientas de la metodología RUP (*Rational Unified Process*) para facilitar el desarrollo del sistema.

1.6.1 Rational Rose

Es una herramienta para “modelado visual”, que forma parte de un conjunto más amplio de herramientas que juntas cubren todo el ciclo de vida del desarrollo de software. Rational Rose permite completar una gran parte de los flujos fundamentales del proceso unificado de Rational (RUP) como el modelado del negocio, la captura de requisitos en forma parcial, el análisis y diseño y parte de la implementación, así como el control de cambios. Incluye, además, un conjunto de herramientas de ingeniería inversa y generación de código a partir de modelos, y en sentido inverso, que allanan el camino hasta el producto final, admite notaciones como UML y Booch entre otras y la integración con modelado de datos, etc. Se apoya en UML v1.4, que no es la más actual pero cubre las necesidades de los clientes.

Conclusiones

En este capítulo se realizó un estudio del estado del arte del sistema donde, a través de una comparación de las soluciones existentes, se determinaron los aspectos que resultan indispensables en sistemas de este tipo. Se arribó a conclusiones sobre las características que distinguen nuestra propuesta diferenciándola del resto. En el se justifican las herramientas, metodologías de desarrollo, lenguajes, y tecnologías que son utilizadas, exponiendo sus característica y particularidades.

Capítulo II: Características del Sistema.

Introducción

Durante el presente capítulo, se mostrarán al usuario las características físicas que tendrá la aplicación, partiendo del análisis de los requisitos planteados por el cliente y el estudio de los procesos que intervienen en el cálculo y obtención del Plan de Vuelo para un avión IL-96/300, destacando las características reales del negocio, hasta quedar conformada la propuesta de la aplicación. Se especifican además los requisitos funcionales y no funcionales del sistema, así como la definición de casos de uso del sistema, definiendo los que serán objeto de automatización en una primera etapa de desarrollo.

2.1 Objeto de estudio

2.1.1 Problema y Situación Problemática

Para la transportación nacional e internacional de pasajeros, carga y correos de forma regular y charter¹⁹; se requiere de la realización de varios procesos previos al vuelo garantizando así su operatividad. Entre dichos procesos se encuentra la elaboración del Plan de Vuelo; el cual resulta imprescindible para la ejecución de un vuelo.

Actualmente el proceso de obtención del plan de vuelo se lleva a cabo de la siguiente manera:

Cuando se le asigna un vuelo a la tripulación esta requiere de un plan de vuelo para ejecutarlo, siendo el navegante el encargado de la realización del mismo. Normalmente el navegante solicita a SITA¹ el plan de vuelo; servicio que como se ha especificado anteriormente resulta costoso para el país; sin embargo en situaciones excepcionales el navegante se ve obligado a realizarlo manualmente debido a la imposibilidad de utilizar este servicio ya sea por falta de conexión, o alguna otra razón.

En estos casos el navegante realiza los cálculos necesarios partiendo de la información meteorológica, el peso de la carga de pago y teniendo en cuenta las restricciones impuestas por los NOTAMS³ y las vías del Atlántico disponibles en el día (NACKTRACK²). Una vez confeccionado el plan de vuelo se discute con el piloto al mando. Luego del análisis y aceptación del plan de vuelo se elabora la hoja de Autorización de Despacho con los datos de combustible necesarios para el centrado y balance del avión, la cual se envía a Operaciones ECASA⁴.

2.1.2 Análisis crítico del negocio.

La confección de un plan de vuelo manual implica la realización de grandes cálculos engorrosos y de mucha precisión, los cuales se ejecutan bajo situaciones de gran estrés y presión en las que se requieren los resultados en el menor tiempo posible y con la mayor exactitud. Todo esto propicia la posibilidad de introducir errores humanos al confeccionarlo.

Esto se debe a que actualmente no se cuenta con un sistema totalmente autónomo capaz de generar de forma automatizada un Plan de Vuelo. Sobre la base de estas dificultades el Sistema Automatizado para la Obtención del Plan de Vuelo del IL96/300, proporcionará al cliente un mejor funcionamiento en su empresa, dotando de mayor seguridad el proceso de obtención y gestión de planes de vuelo.

El análisis investigativo que fundamenta teóricamente este sistema parte del estudio de los procesos del negocio que actualmente soportan los cálculos para la obtención del Plan de Vuelo, y a partir de ahí se definen los actores, trabajadores y casos de uso del negocio.

2.1.3 Objeto de automatización

Serán objeto de automatización todos aquellos procesos requeridos para la confección del Plan de Vuelo de un IL_96 300, entre los que se incluye el registro de los puntos de control, que resulta de suma importancia en la confección de la ruta a seguir en el vuelo, estos procesos junto a la gestión de la información de los aviones y los aeropuertos permiten el cálculo y obtención del Plan de Vuelo.

En el proceso de cálculo del Plan de Vuelo se trabaja con información que se obtiene de tablas tabuladas entre las que se incluyen: Tablas de Ascenso, Descenso y Crucero, las que son específicas para el avión IL-96 300 según el manual de vuelo, además del análisis de las políticas de combustible y de despacho al alterno concebidas para la planificación del vuelo.

2.1.4 Información que se maneja

Para la planificación de vuelos de un IL-96/300 se utilizan las especificaciones para este tipo de avión contenidas en el Manual Básico de Operaciones y el Manual de Vuelo, siendo oportuno además el análisis de los datos meteorológicos y la información de los vientos que se brindan en las Cartas de tiempo significativos y las Cartas de vientos y temperaturas, así como la información referente a NOTAMS³ y NACKTRACKS².

2.2 Propuesta del sistema

El sistema que se propone requiere para su utilización de una previa autenticación, lo cual permitirá al usuario acceder solamente a los datos que le resultan realmente necesarios. Una vez dentro del módulo de Plan de Vuelo el usuario tendrá la posibilidad de: guardar, eliminar o modificar los datos de un avión; contará con la facilidad de manejar la información de los aeropuertos y crear, eliminar o modificar los puntos de control. El usuario tendrá la opción de crear nuevas rutas o visualizar las ya existentes, además de modificar o eliminar alguna de las que se encuentran almacenadas en la base de datos; igualmente el sistema posibilitará unificar, convertir o invertir una Ruta especificada; y a partir de la información obtenida de los Puntos de Control y la Ruta, unido a los restantes datos necesarios, se podrá generar el Plan de Vuelo.

2.3 Modelo de negocio.

2.3.1 Breve Descripción Del Negocio

Con la cantidad de pasajeros a transportar el navegante calcula el peso de equipaje según lo estipulado comercialmente (20 Kg. o 40 Kg. por pasajero), que unido a la información de la carga que debe trasladar la aeronave constituye el valor de la carga de pago. Conocida la información de las condiciones meteorológicas de los aeropuertos, el comportamiento de los vientos y las temperaturas en los diferentes niveles de vuelo y la carga de pago, el navegante confecciona el Plan de Vuelo, teniendo en cuenta las restricciones impuestas por los NOTAMS³ y las vías del Atlántico que estarán disponibles en el día.

Una vez confeccionado el Plan de Vuelo se discute con el piloto al mando. Luego del análisis y aceptación de este se elabora la hoja de Autorización de Despacho y se envía a Operaciones ECASA⁴ con los siguientes datos:

Imprescindibles:

- Combustible de quemada¹¹, el de reserva y el requerido.

Otros datos posibles

- Aeropuertos de origen y destino.
- Tipo de avión.
- Niveles de vuelo.
- Tiempos de vuelo

Como resultado de un profundo análisis se propone realizar una aplicación totalmente autónoma que automatice los cálculos necesarios para la elaboración de un plan de vuelo de un avión IL-96/300; dando solución a las principales necesidades de la empresa actualmente. Dicho sistema proporcionaría mayor seguridad en los cálculos y minimizaría el tiempo de ejecución de los mismos.

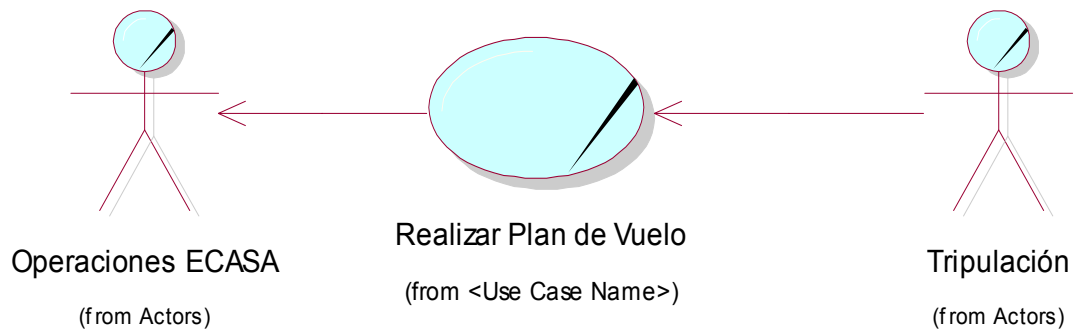
2.3.2 Actores Del Negocio

Actor	Descripción
Tripulación(Piloto al Mando)	Es el interesado en que se confeccione el plan de vuelo para conocer la ruta a seguir, los niveles de vuelo, el aeropuerto alterno, la cantidad de combustible, entre otros datos de importancia.
Operaciones ECASA	Interesado en obtener la hoja de autorización de despacho que se genera a partir del plan de vuelo, ya que conociendo la cantidad de combustible necesario puede balancear el avión.

2.3.3 Trabajadores Del Negocio

Trabajador	Descripción
Navegante	Integrante de la tripulación encargado de confeccionar el plan de vuelo.

2.3.4 Diagrama de Casos de Uso del Negocio



2.3.5 Descripción de los Casos de Usos en formato expandido.

En el modelo de negocio se describen los procesos del negocio (casos de uso del negocio) en términos de la descripción textual del caso de uso, donde se exponen los actores y su interacción con el negocio. Se describe el flujo de trabajo indicando todas las actividades del negocio que ocurren, el orden, el trabajador del negocio que las realiza y su relación con las entidades del negocio. Especificando los puntos intermedios en los que puede finalizar el proceso.

La descripción textual del negocio da una visión de las mejoras que aportará al proceso cuando algunas de sus actividades sean automatizadas. Ver [Anexo 1](#).

2.3.6 Diagramas de Actividad.

Los diagramas de actividad se utilizan para mostrar los diferentes flujos de trabajo o los procesos dentro del sistema, cómo se inician, los variados caminos alternativos que se pueden tomar desde el inicio hasta el fin y dónde puede ocurrir el procesamiento paralelo durante la ejecución.

En ellos no se modela el comportamiento exacto del sistema de software (como lo hace el diagrama de secuencia), sino los procesos y los flujos a un muy alto nivel. Las actividades generalmente son realizadas por uno o más casos de uso; la actividad describe el proceso que se desarrolla y tanto el caso de uso como un actor usará el sistema para realizar toda o parte de una actividad. Ver [Anexo 1](#)

2.3.7 Modelo de Objeto.

El modelo de objeto es un diagrama de clases, que como artefacto se construye para describir el modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y la relación entre ellos. Aunque se puede construir un único diagrama, se recomienda confeccionarlo para cada caso de uso de negocio para una mejor claridad.

Como todo diagrama de clases, se pueden representar, además de la asociación, los distintos tipos de relaciones entre las entidades de negocio (agregación, composición y generalización / especialización), la cardinalidad y navegabilidad de las relaciones, pero para efectos de su utilización posterior es suficiente con mostrar la relación entre los trabajadores y estos con las entidades. Ver [Anexo 1](#)

2.4 Relación de Requerimientos

2.4.1 Requerimientos funcionales.

1. Aeropuertos

1.1 *Introducir un nuevo aeropuerto:*

Permite incorporar las características de un nuevo aeropuerto y almacenarlo en la base de datos.

1.2 *Eliminar un aeropuerto:*

Permite eliminar un aeropuerto de la base de datos.

1.3 *Modificar un aeropuerto:*

Permite modificar la información de un aeropuerto almacenada en la base de datos.

2. Puntos de control

2.1 *Introducir un nuevo punto de control:*

Permite incorporar los datos correspondientes a un punto que no existe en la base de datos.

2.2 *Eliminar un punto de control:*

Permite eliminar un punto de control de la base de datos.

2.3 *Modificar un punto de control:*

Permite modificar la información de un punto de control almacenado en la base de datos.

3. Aviones

3.1 Introducir un avión:

Permite incorporar los datos de un avión que no existe en la base de datos.

3.2 Eliminar un avión:

Permite eliminar un avión de la base de datos.

3.3 Modificar un avión:

Permite modificar la información almacenada de un avión en la base de datos.

4. Rutas

4.1 Rutas fijas

4.1.1 Crear una nueva ruta fija:

Permite introducir los datos de una nueva ruta fija si no existe en la base de datos.

4.1.2 Eliminar una ruta fija:

Permite eliminar una ruta fija de la base de datos.

4.1.3 Modificar ruta fija:

Permite modificar la información de una ruta fija almacenada en la base de datos.

4.2 Rutas provisionales

4.2.1 Crear una nueva ruta provisional:

Permite introducir los datos de una nueva ruta provisional si no existe en la base de datos.

4.2.2 Eliminar una ruta provisional:

Permite eliminar una ruta provisional de la base de datos.

4.2.3 Modificar una ruta provisional:

Permite modificar la información de una ruta provisional almacenada en la base de datos.

4.3 Unir rutas:

Permite unificar varias rutas almacenadas para trabajarlas como una sola.

4.4 Transformar ruta:

Permite transformar una ruta almacenada en la base de datos de fija a provisional o viceversa.

5. Plan de vuelo

5.1 Generar Plan de vuelo

Permite a usuario del sistema construir un plan de vuelo.

5.2 Gestionar Plan de vuelo

5.2.1 Imprimir plan de vuelo:

Permite imprimir el plan de vuelo obtenido.

5.2.2 Exportar plan de vuelo:

Permite exportar el plan de vuelo obtenido.

2.4.2 Requerimientos no funcionales.

Apariencia o interfaz externa.

El diseño de la interfaz externa del sistema debe ser profesional, sencillo, amigable, de fácil transición, familiar a los usuarios que han usado otras aplicaciones de escritorio en Windows, con el fin de lograr una eficiente interacción con el usuario, proporcionándole en todo momento una sensación de control sobre la aplicación. Para la construcción del sistema se deben seguir las normas convencionales de interfaz de usuario de Windows. Se debe mantener informado al usuario acerca de todo lo que sucede en la aplicación, los mensajes de esta deben estar dirigidos al usuario y por tanto, redactados en su idioma. La información se debe presentarse de forma clara.

Usabilidad.

El sistema está concebido para ser usado por la tripulación del avión IL_96 300 por lo tanto la dificultad dependerá del número de pasos, el conocimiento que el usuario debe tener del proceso y las decisiones que este debe tomar en cada paso. Para evitar errores, en los campos que se requiera el sistema brindará la opción de elegir el valor deseado en vez de que el usuario introduzca los datos.

Se seguirán las guías de Interfaz de Usuario para los menús, botones y los cuadros de diálogos, siempre que sea posible. Se debe informar al usuario en todo momento acerca de lo que sucede en la aplicación por lo que los mensajes deben ser evidentes y personalizados.

Rendimiento.

El sistema operará con grandes volúmenes de información, por tanto, se hacen necesarios tiempos de respuestas cortos, al igual que la velocidad de procesamiento de la información.

Soporte.

Debe ser de fácil instalación y mantenimiento. Se debe generar la inserción de nuevos módulos, sin negar lo realizado o afectar el buen funcionamiento del mismo. El sistema debe ser sometido a una etapa de prueba donde se realice el adiestramiento de los usuarios, permitiendo la familiarización con el software y la detección de posibles errores o cambios en las interfaces de manera que queden complacidos.

Portabilidad.

El sistema funcionará sobre plataforma Windows, porque es la exigida por el cliente debido a su conocimiento de esta plataforma, además es la utilizada en la facultad.

Seguridad

Confiabilidad: la información manejada por el sistema debe estar protegida de acceso no autorizado y divulgación.

- Integridad: la información manejada por el sistema debe ser objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma será considerada igual a la fuente o autoridad de los datos.
- Disponibilidad: se les garantizará el acceso a la información solo a los usuarios autorizados evitando que los dispositivos o mecanismos utilizados para lograr la seguridad oculten o retrasen a los usuarios en la obtención de los datos deseados en un momento dado.

Todo esto se logra a través de la creación de grupos de usuarios los cuales tendrán asignados permisos de acción sobre cada información manejada por el sistema, para lo cual se requiere la autenticación del usuario. Si no es un usuario reconocido por el sistema, no podrá acceder a ningún módulo. Detallando aun más, entre las acciones a tener en cuenta para garantizar la seguridad se encuentran las siguientes:

1. El acceso será controlado con nombres de usuario y contraseñas
2. Las contraseñas deberán tener de 4 a 10 caracteres de longitud
3. La información será manejada únicamente por quien tenga los permisos suficientes para acceder a ella.
4. El sistema contará con protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.
5. La aplicación tendrá implementada la verificación sobre acciones irreversibles (eliminaciones).

Interfaz de software:

- Se realizará una aplicación de escritorio.
- La base de datos será independiente a la aplicación.

Confiabilidad.

El sistema debe ser preciso en la información que le suministra al usuario para evitar cualquier tipo de error. Además estar bien documentado, para lograr que el tiempo de mantenimiento sea mínimo.

Ayuda y documentación.

El sistema contará con la documentación completa de todas las tareas y operaciones que realiza el software, el glosario de términos y las planillas que especifican toda la Ingeniería de Software.

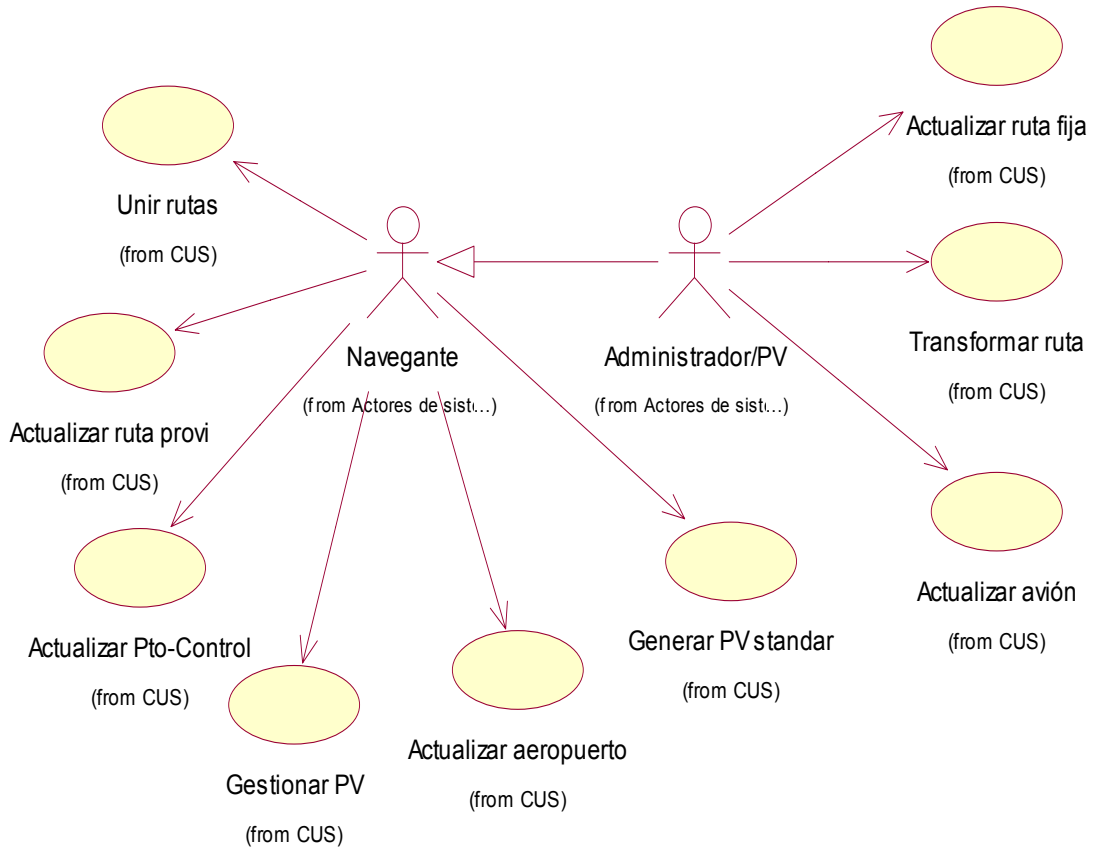
Además poseerá una ayuda que garantice el asesoramiento e información al usuario acerca de los contenidos tratados. En esta deben quedar claramente reflejadas las funcionalidades del sistema y su manipulación. También contendrá los datos del contacto para negociar futuros mantenimientos del software y solucionar los problemas que se puedan presentar. El sistema requiere de la construcción de un Manual de Usuario que describa pormenorizadamente sus características y uso.

2.5 Modelo de Casos de Usos del Sistema

2.5.1 Definición de los actores del sistema.

Actor	Descripción
Navegante	Responsable de manejar los datos e introducirlos en el software con antelación al despegue.
Administrador/PV	Administra la aplicación y a los usuarios de la misma. Tiene privilegios especiales.

2.5.2 Diagrama de Casos de Uso del Sistema.



2.5.3 Descripción textual de los casos de uso.

A través de la expansión de los casos de uso del sistema se describe paso a paso la secuencia de eventos que los actores utilizan para completar un proceso a través del sistema.

En el [Anexo 2](#) se ve la expansión de los casos de uso del sistema.

Conclusiones

En el modelo de negocio se describen los procesos del negocio (casos de uso del negocio) en términos de la descripción textual del caso de uso, donde se exponen los actores y su interacción con el negocio. Se describe el flujo de trabajo indicando todas las actividades del negocio que ocurren, el orden, el trabajador del negocio que las realiza y su relación con las entidades del negocio. Especificando los puntos intermedios en los que puede finalizar el proceso.

La descripción textual del negocio da una visión de las mejoras que aportará al proceso cuando algunas de sus actividades sean automatizadas.

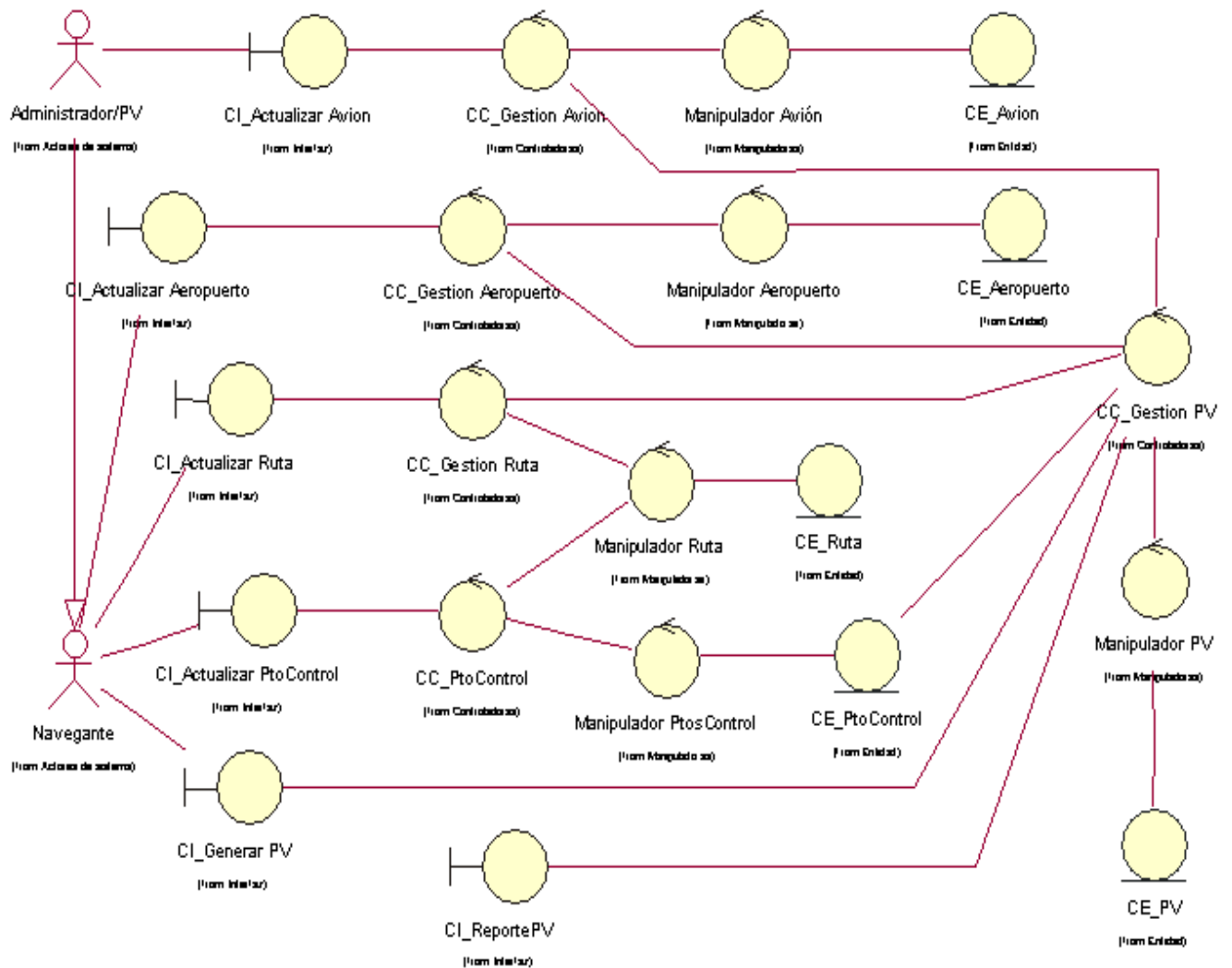
Capítulo III: Análisis y Diseño del Sistema.

Introducción

El análisis nos adentra en el problema a resolver, en él representamos una vista interna del sistema en la que, usando el lenguaje de los desarrolladores se analizan con mayor profundidad los requisitos, refinándolos y estructurándolos en base a clases y paquetes, proporcionando una visión general del sistema. Este proceso continúa en el diseño hasta obtener los objetos que interactúan para dar soporte a los requisitos funcionales y no funcionales definidos y adquirir una comprensión de los aspectos relacionados con las restricciones relacionadas con los lenguajes de programación, componentes reutilizables, sistemas operativos, tecnologías de distribución y concurrencia y tecnologías de interfaz de usuario. Los diagramas de interacción, por su parte, describen gráficamente la solución a partir de los objetos en interacción; muestran las decisiones referentes a la asignación de responsabilidades entre los objetos.

3.1 Análisis.

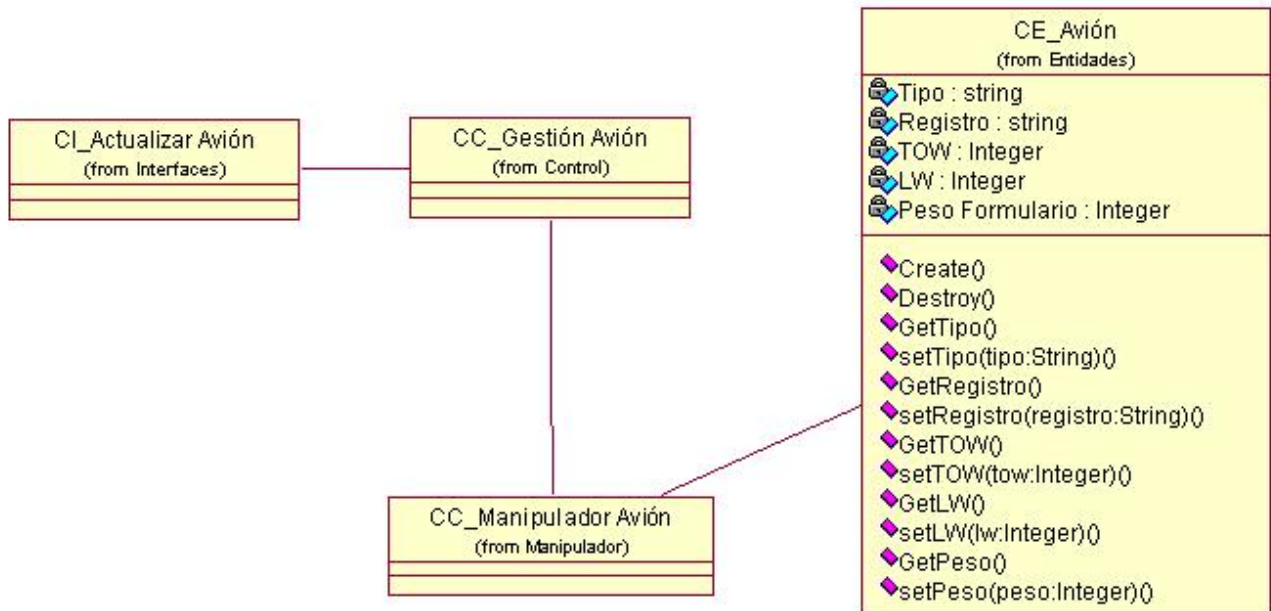
3.1.1 Diagrama de Clases del Análisis.



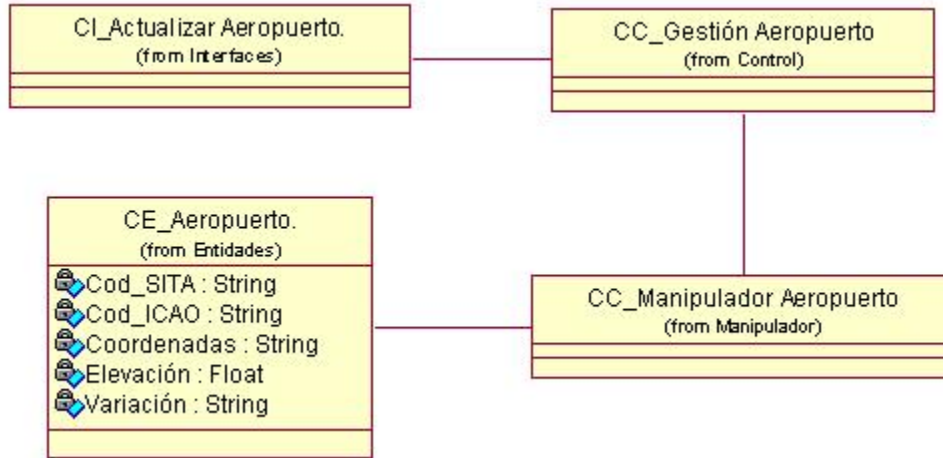
3.2 Diseño

3.2.1 Diagrama de Clases del Diseño por CU²¹.

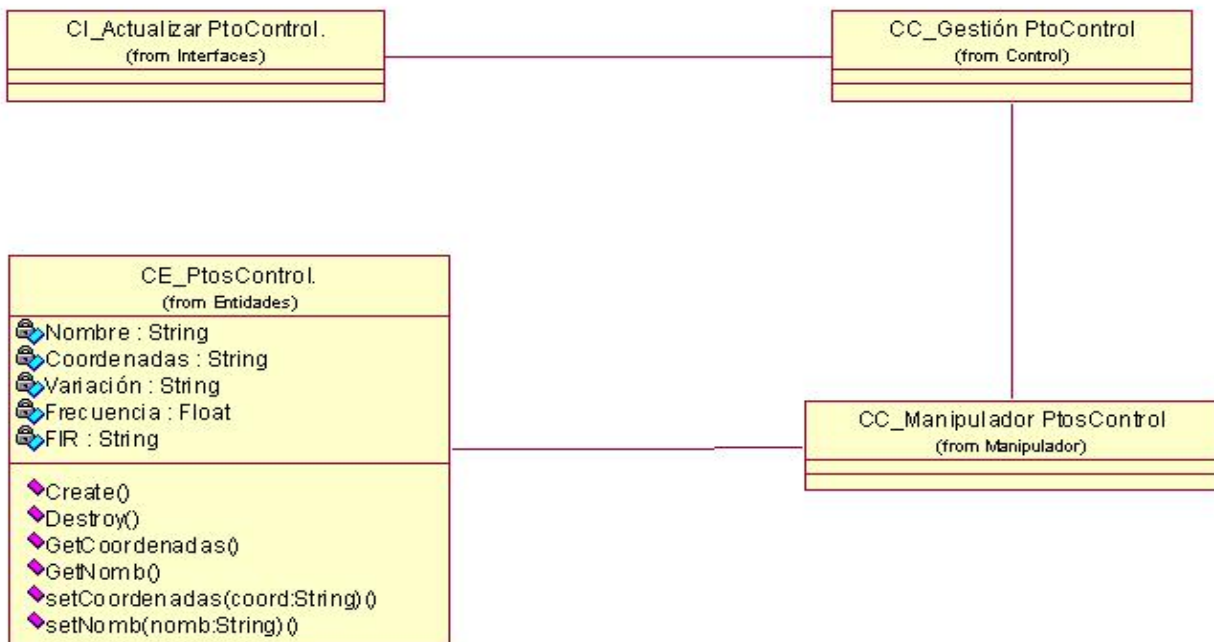
1. CU²¹ Actualizar Avión



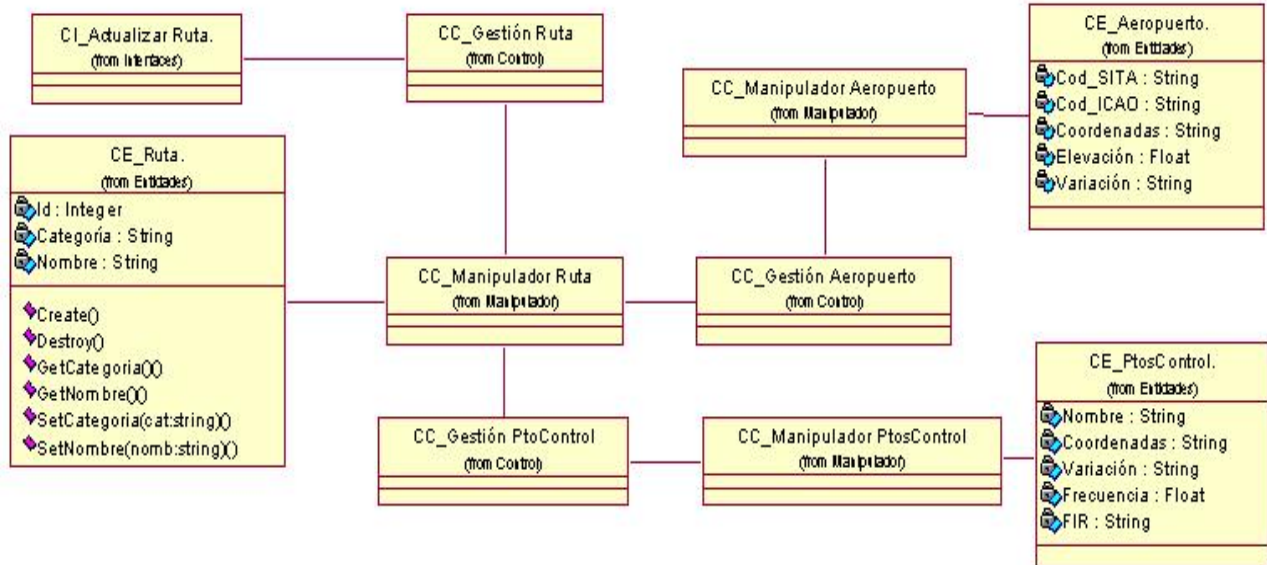
2. CU²¹ Actualizar Aeropuerto



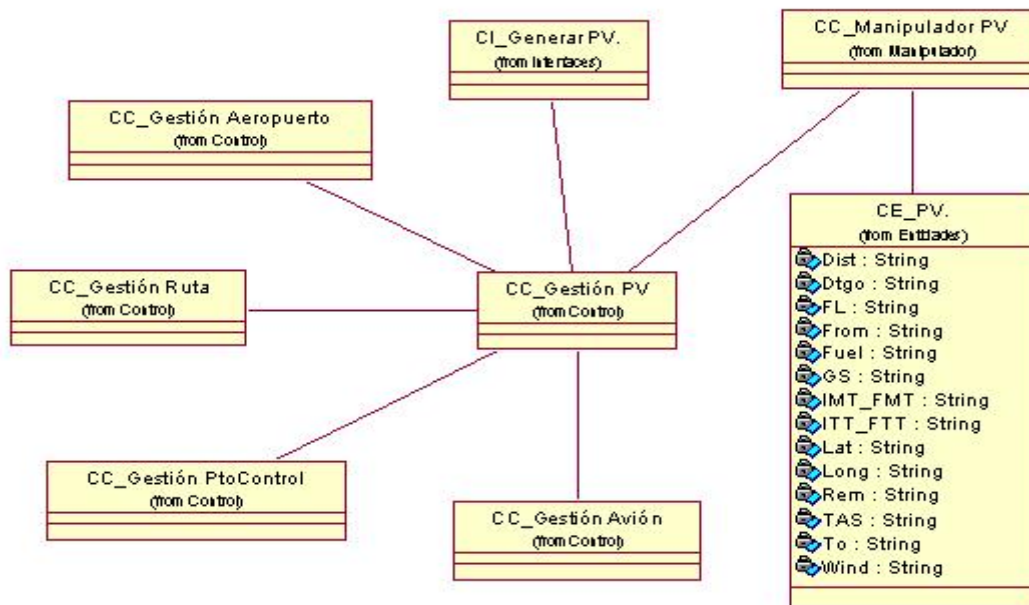
3. CU²¹ Actualizar Punto de Control



4. CU²¹ Actualizar Ruta

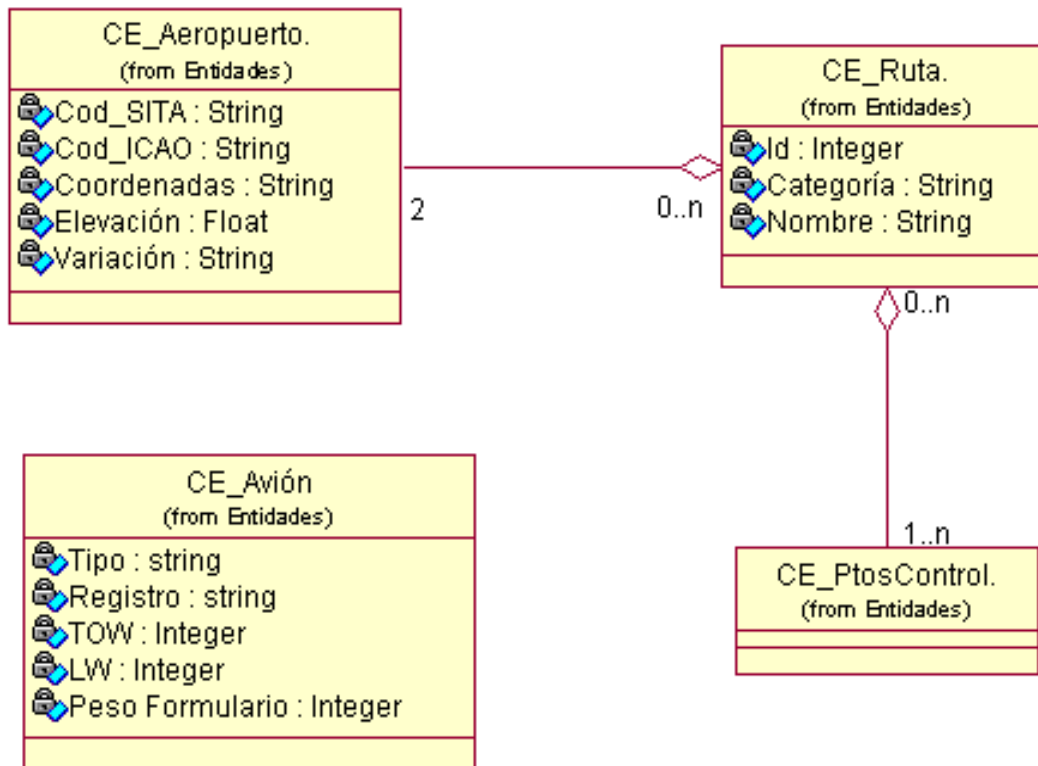


5. CU²¹ Plan de Vuelo

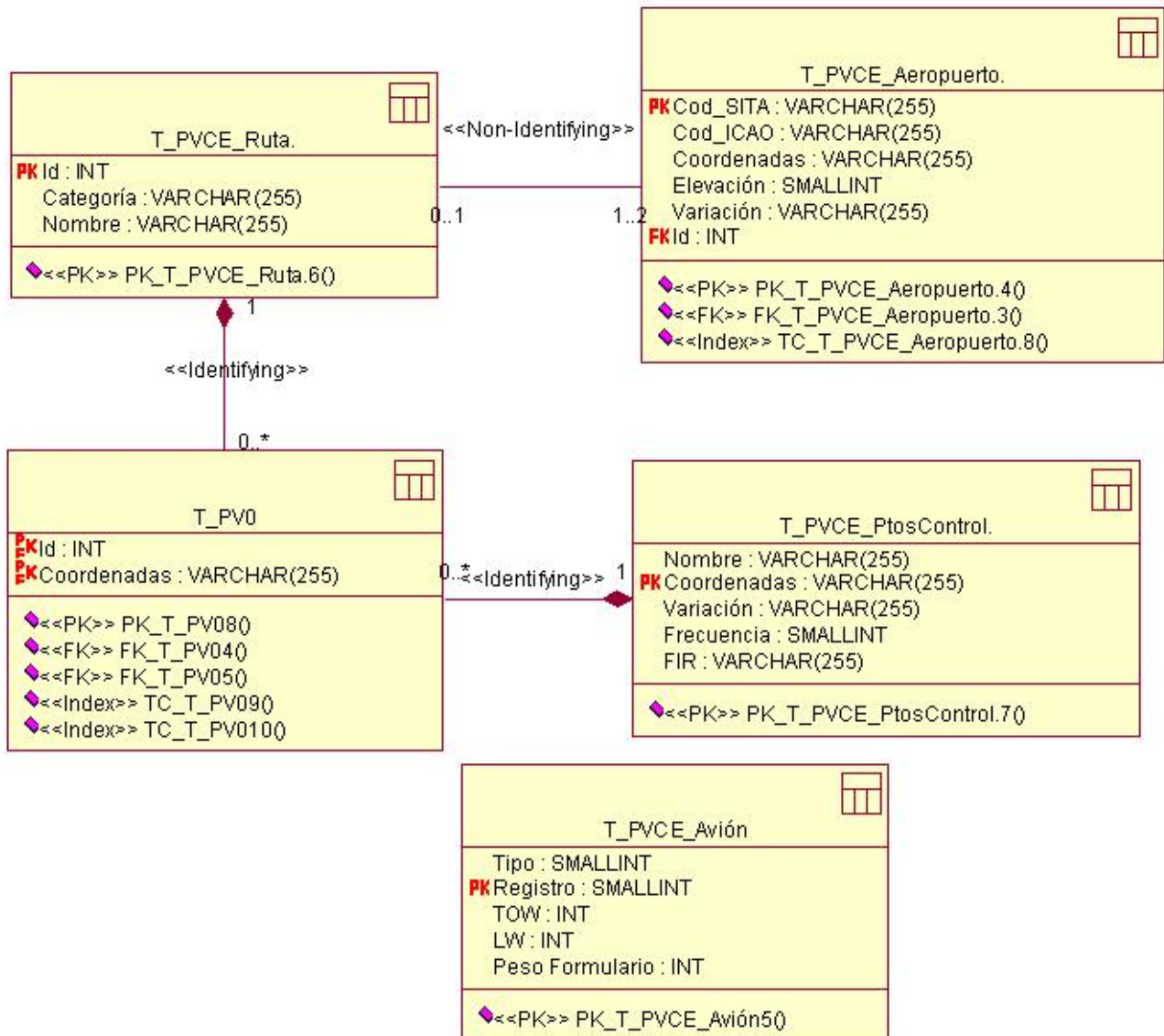


3.3 Diseño de la Base de Datos.

3.3.1 Modelo Lógico de Datos.



3.3.2 Modelo Físico de Datos.



Conclusiones

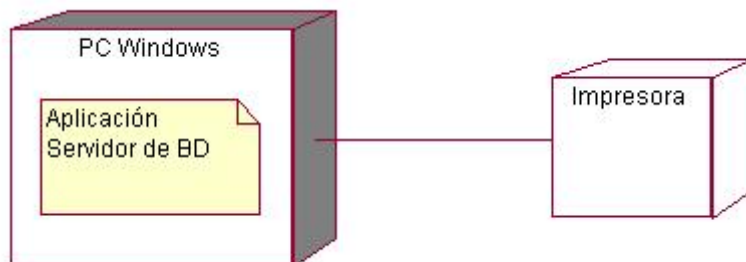
En el presente capítulo se mostraron los resultados de las etapas de análisis y diseño del sistema. Se desarrollaron los diagramas de clases del análisis, diagrama de clases del diseño por casos de uso, modelo lógico de datos, el modelo físico de la Base de Datos. Resultan de suma importancia para la comprensión del sistema los diagramas de interacción, específicamente los diagramas de secuencia, que se elaboraron de modo tal que se desarrollara un diagrama de interacción para cada una de las realizaciones de los casos de uso del sistema. Todos estos elementos obtenidos brindan una idea mucho más clara de las interfaces de usuario, subsistemas, algoritmos y demás elementos que influyen en el logro de una mejor herramienta.

Capítulo IV: Implementación.

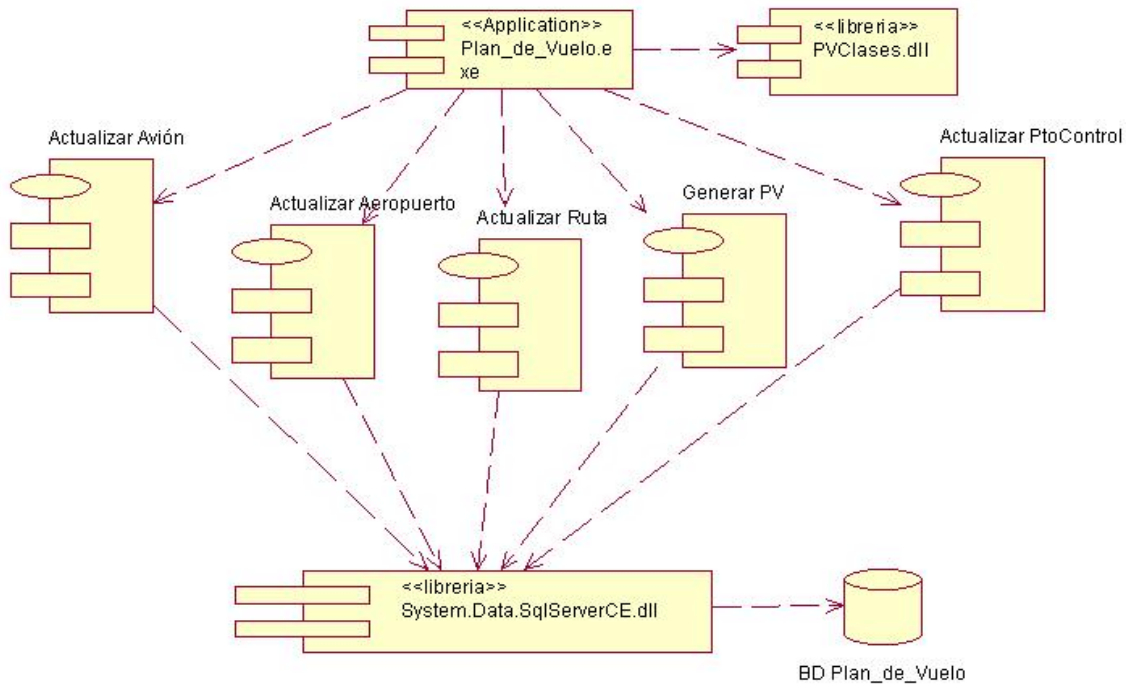
Introducción.

En el presente capítulo se modelan artefactos que apoyan el proceso de construcción del sistema propuesto. Mediante los diagramas de despliegue, los diagramas componentes de paquetes, unido a un buen uso de los estándares de interfaz y de codificación. El modelo de implementación describe cómo se implementan los elementos del modelo de diseño en términos de componentes y describe también como se organizan y se relacionan unos con otros, creándose una base sólida sobre la que el proceso de construcción de aplicaciones se optimiza y facilita.

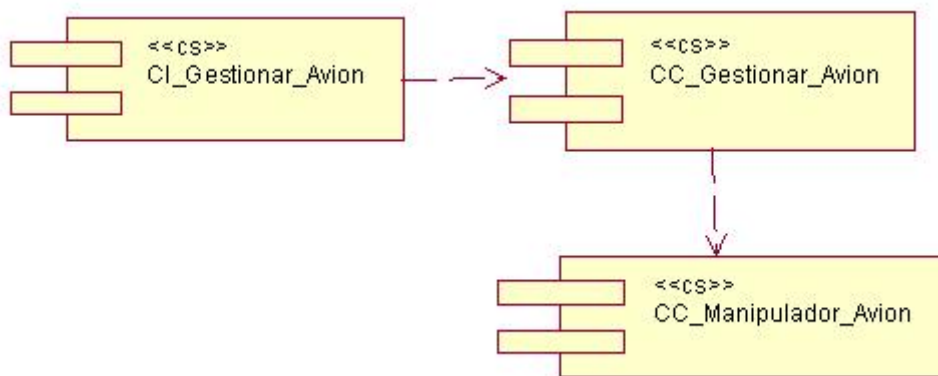
4.1 Diagrama de Despliegue.



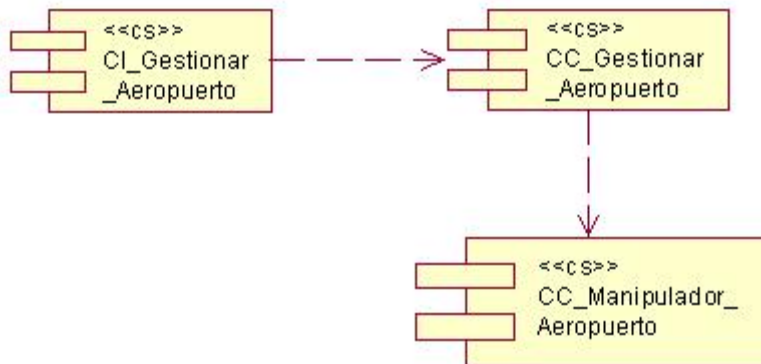
4.2 Diagrama de Componentes.



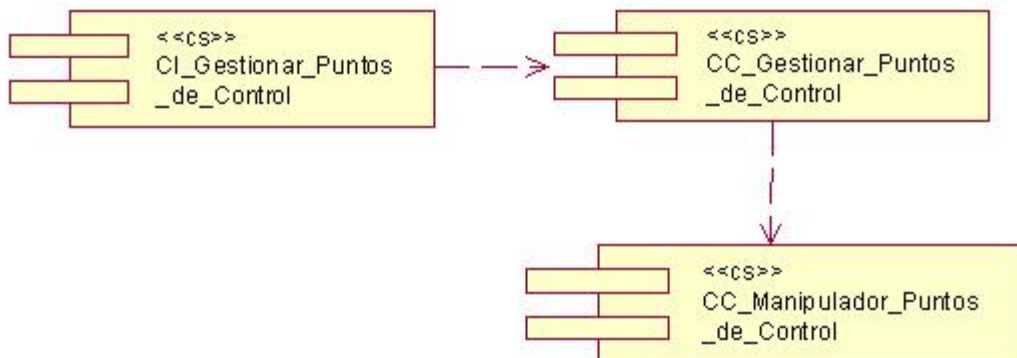
4.2.1 Componentes Paquete Actualizar Avión.



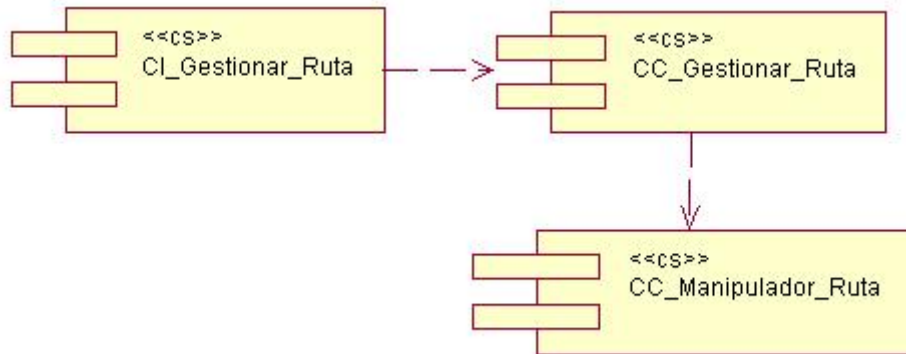
4.2.2 Componentes Paquete Actualizar Aeropuerto.



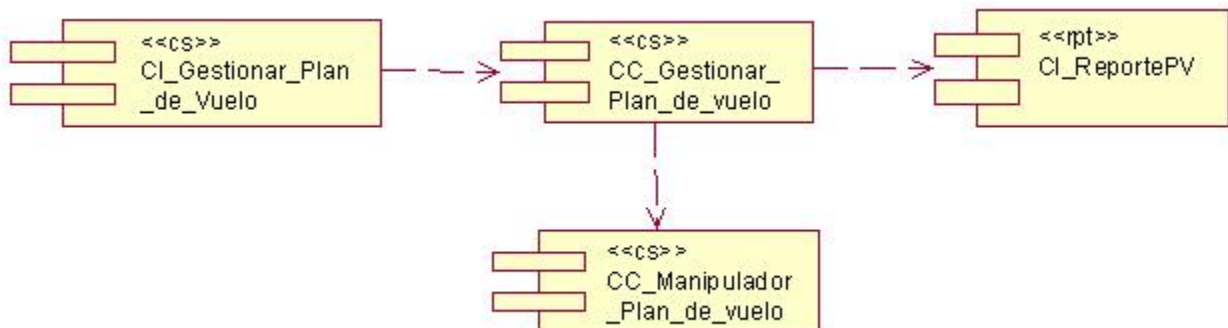
4.2.3 Componentes Paquete Actualizar Punto de Control.



4.2.4 Componentes Paquete Actualizar Ruta.



4.2.5 Componentes Paquete Actualizar Ruta.



Conclusiones

Con el desarrollo de este capítulo y basados en la modelación correspondiente para definir el diagrama de componente, el diagrama de despliegue y demás elementos necesarios para la implementación, se logró: una visión de la aplicación como entidad física y la definición de los componentes que la conforman; teniendo así una base sólida para un desarrollo eficiente de la solución propuesta.

Capítulo V: Estudio de Factibilidad.

Introducción

Este capítulo aborda el tema relacionado al estudio de factibilidad del producto, donde se presentan la planificación y los costos del proyecto, basados en la técnica de estimación de esfuerzo por punto de casos de uso, la que permitirá obtener una idea del esfuerzo necesario para llevar adelante el mismo. Este método proporciona una estimación del esfuerzo en horas-hombre contemplando sólo el desarrollo de la funcionalidad especificada en los casos de uso. Se analizan además los beneficios tangibles e intangibles obtenidos con el desarrollo del proyecto, realizando una evaluación de factibilidad mediante el análisis de costos y beneficios.

5.1 Planificación basada en casos de uso.

- a) Cálculo de los Puntos de casos de uso desajustados.

$$UUCP = UAW + UUCW$$

UUCP: Puntos de casos de uso sin ajustar.

UAW: Factor de peso de los actores sin ajustar.

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

Tipo de actor	Descripción	Factor de peso	Actores	Total
Simple	Sistema con sistema a través de interfaz de programación.	1	0	0
Medio	Sistema con sistema a través de interfaz basada en texto.	2	0	0

Complejo	Persona que interactúa con el sistema mediante interfaz gráfica	3	2	6
Total				6

Tabla 5.1 Factor de peso de los actores sin ajustar.

$$UAW = \sum \text{cant actores} * \text{peso}$$

$$UAW = 2*3=6$$

Tipo de CU	Descripción	Peso	Cantidad de CU	Total
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	3	15
Medio	El caso de uso tiene de 4 a 7 transacciones.	10	1	10
Complejo	El caso de uso tiene más de 8 transacciones.	15	5	75

Tabla 5.2 Factor de peso de los casos de uso sin ajustar.

$$UUCW = \sum \text{cant CU} * \text{Peso}$$

$$UUCW = 100$$

$$UUCP = 6+100$$

$$UUCP = 106$$

b) Cálculo de los Puntos de casos de uso ajustados.

$$UCP = UUCP * TCF * EF$$

UCP: Puntos de casos de uso ajustados.

UUCP: Puntos de casos de uso sin ajustar.

TCF: Factor de complejidad técnica.

EF: Factor de ambiente.

Calculándose el Factor de complejidad técnica (TCF) mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada factor se pondera con un valor desde 0 (aporte no considerable) hasta 5 (aporte muy apreciable).

Significado de los valores

0: No presente o sin influencia,

1: Influencia incidental o presencia incidental

2: Influencia moderada o presencia moderada

3: Influencia media o presencia media

4: Influencia significativa o presencia significativa

5: Fuerte influencia o fuerte presencia

Factor	Descripción	Peso	Valor asignado	Total
T1	Sistema distribuido	2	0	0
T2	Objetivos de <i>performance</i> o tiempo de respuesta	1	5	5
T3	Eficiencia del usuario final	1	5	5
T4	Procesamiento interno complejo	1	3	3
T5	El código debe ser reutilizable	1	5	5
T6	Facilidad de instalación	0.5	5	2.5
T7	Facilidad de uso	0.5	5	2.5
T8	Portabilidad	2	5	10
T9	Facilidad de cambio	1	5	5
T10	Concurrencia	1	1	1
T11	Incluye objetivos especiales de	1	5	5

	seguridad			
T12	Provee acceso directo a terceras partes	1	0	0
T13	Se requieren facilidades especiales de entrenamiento a los usuarios.	1	0	0
Total				44

Tabla 5.3 Factor de complejidad técnica.

$$TCF = 0.6 + 0.01 * \sum (\text{peso} * \text{valor asignado})$$

$$TCF = 0.6 + 0.01 * 44$$

$$TCF = 1.04$$

Factor de ambiente (EF) se relaciona directamente con las habilidades y entrenamiento del grupo que realiza el sistema. Cada factor se pondera con un valor desde 0 (aporte no considerable) hasta 5 (aporte muy apreciable).

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

Factor	Descripción	Peso	Valor asignado	Total
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	0	0
E2	Experiencia en la aplicación.	0.5	1	0.5
E3	Experiencia en la orientación a objetos	1	4	4
E4	Capacidad de analista líder.	0.5	3	1.5
E5	Motivación.	1	4	4
E6	Estabilidad de requerimientos.	2	3	6
E7	Personal Part-Time	-1	2	-2

E8	Dificultad del lenguaje de programación.	-1	3	-3
Total				11

Tabla 5.4 Factor de ambiente.

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

$$EF = 1.4 - 0.03 * 11$$

$$EF = 1.07$$

$$UCP = UUCP * TCF * EF$$

$$UCP = 106 * 1.04 * 1.07$$

$$UCP = 117.9568$$

c) Estimación de esfuerzo a través de los puntos de casos de uso.

$$E = UCP * CF$$

E: Esfuerzo estimado en horas hombres.

UCP: Punto de casos de usos ajustados.

CF: Factor de conversión.

Factor de convención (CF) se cuentan cuantos valores del factor ambiente están por debajo de la media (3) de E1 a E6, y cuántos están por encima de la media en E7 y E8. Si el total es 2 o menos se utiliza el factor de convención 20 HH/Puntos de CU²¹. Si el total es 3 o 4 se utiliza 28 HH/Puntos de CU²¹. Si el total es mayor a 5 se recomienda efectuar cambios en el proyecto ya que tiende a perecer en corto tiempo. En este caso se puede afirmar que:

$$CF = 20 \text{ HH/Puntos de CU}^{21}$$

$$E = 117.9568 * 20$$

$$E = 2359.136 \text{ Horas/Hombre}$$

Calcular esfuerzo de todo el proyecto.

Actividad	Porcentaje %	Horas/Hombre
Análisis	20%	1179.568 horas-hombre
Diseño	30%	1769.352 horas-hombre
Implementación	40%	2359.136 horas-hombre
Pruebas	0%	0 horas-hombre
Sobrecarga (otras actividades)	10%	589.784 horas-hombre
Total	100%	5897.84 horas-hombre

Tabla 5.5 Esfuerzo del proyecto.

Si $E_T = 5897.84$ Horas/ hombre y se estima que cada mes promediado 192 horas laborables, quedaría $E_T = 30.7179167$ mes/hombre.

d) Costo del Proyecto

En el caso del salario mensual es de \$100.00

CH: Cantidad de hombres

Tiempo: Tiempo total del Proyecto.

CH= 2 hombres.

CHM= 2 hombres * Salario.

CHM=200 \$/mes

Costo= CHM * E_T

Costo= 200 * 30.7179167

Costo= \$ 6143.58333

Tiempo total del Proyecto:

Tiempo = E_T / CH

Tiempo= 30.7179167 meses /2 hombres

Tiempo= 15.35895835 meses

Del resultado obtenido se interpreta que con 2 hombres el proyecto tiene un tiempo de duración de aproximadamente 15 meses con 9 días y el costo total que se estima es de \$ 6143.58.

5.2 Análisis de costos y beneficios.

El desarrollo de todo producto de software requiere de una inversión de capital, la que se dividen en gastos de personal, y de materiales necesarios para su elaboración, por lo que la viabilidad del proyecto depende de la medida en que los beneficios del mismo puedan saldar lo invertido en su desarrollo, ya sea, social o económicamente. En la realización de este proyecto se usará software propietario para el cual el cliente garantizó la licencia del producto por lo que este gasto no está reflejado en el estudio de factibilidad.

Al implantar el sistema se tendrá como resultado el aumento de la eficiencia del proceso del cálculo del Plan de Vuelo para el avión IL-96 300, disminuyendo el tiempo de realización de este proceso, logrando una mayor exactitud en dichos cálculos. Además contará nuestro país con una solución, pues anteriormente no existía un software que realizara los cálculos para Plan de Vuelo que cubriera las necesidades de la tripulación del IL-96 300; al mismo tiempo la puesta en práctica de este sistema ahorraría una suma considerable de dinero utilizado para el pago de estos servicios a SITA¹. A través del análisis del costo del proyecto y los numerosos beneficios que este reporta, se puede concluir que su implementación es realmente factible.

Conclusiones

En este capítulo se realizó el estudio de factibilidad y validación del sistema. Se obtuvo como resultado que con las dos personas que desarrollarán el sistema se requiere un tiempo de 15 meses y 9 días para la implementación del mismo, con un costo total del proyecto de \$ 6143.58. Se expusieron los beneficios del sistema y se efectuó un análisis entre estos y los costos, el cual brindó elementos para concluir que es factible el desarrollo de esta aplicación.

Se comprobó que la herramienta propuesta reportará beneficios significativos e importantes para la tripulación de los aviones IL-96-300.

CONCLUSIONES

Con la implantación exitosa del sistema, se dará solución a los problemas actuales existentes en la empresa Cubana de Aviación para la tripulación que opera los aviones IL96/300, lo cual implicaría un mejoramiento en las condiciones de trabajo de los usuarios logrando eliminar los errores en los resultados obtenidos y una reducción de costos relacionados con los servicios contratados a SITA1 que se utilizan.

Para la realización del software se transitó por los flujos de trabajo de negocio, análisis, diseño, administración del proyecto e implementación respondiendo al proceso iterativo e incremental RUP, realizándose una modelación del negocio con vista a entender el contexto donde se va a implantar el sistema y se presentándose un grupo de artefactos para los flujos de trabajo de requisitos, análisis y diseño e implementación que propone dicho proceso.

Como resultado de la selección de las herramientas se determinó utilizar para la implementación del software el lenguaje de programación C#, la plataforma .NET, el SQL Server 2000 como servidor de base de datos, y el Rational Rose para el modelado visual del sistema.

Con los beneficios expuestos, tangibles e intangibles y el análisis de costos efectuado, se determinó que resulta factible el desarrollo de la aplicación, siendo incuestionable las mejoras que producirá la implantación de este servicio en la Empresa Cubana de Aviación.

RECOMENDACIONES

Se recomienda a las personas que darán continuidad a este trabajo:

- Concluir con la etapa de prueba del primer ciclo, e incorporar nuevas funcionalidades al sistema, entre ellas la elaboración de un plan de vuelo resumido e importar un plan de vuelo almacenado en el disco duro, dar la opción de introducir los datos de vientos como componentes de viento y la temperatura en formato ISA.
- Implementar el módulo de seguridad de la aplicación.
- Realizar la integración de los módulos de Análisis de Pista, Peso y Balance y Plan de Vuelo.
- Desarrollar la ayuda del sistema.
- Desarrollar e implementar la Base de Datos del sistema.
- Utilizar el material como bibliografía para investigaciones futura en esta área de desarrollo.
- Instruir al personal que operará el sistema.
- Mantener sobre el sistema un estricto cumplimiento del proceso de mantenimiento y actualización periódica, logrando así que se mantenga la fiabilidad y funcionamiento óptimo del sistema y de la información que se gestiona a través de el.

ANEXOS

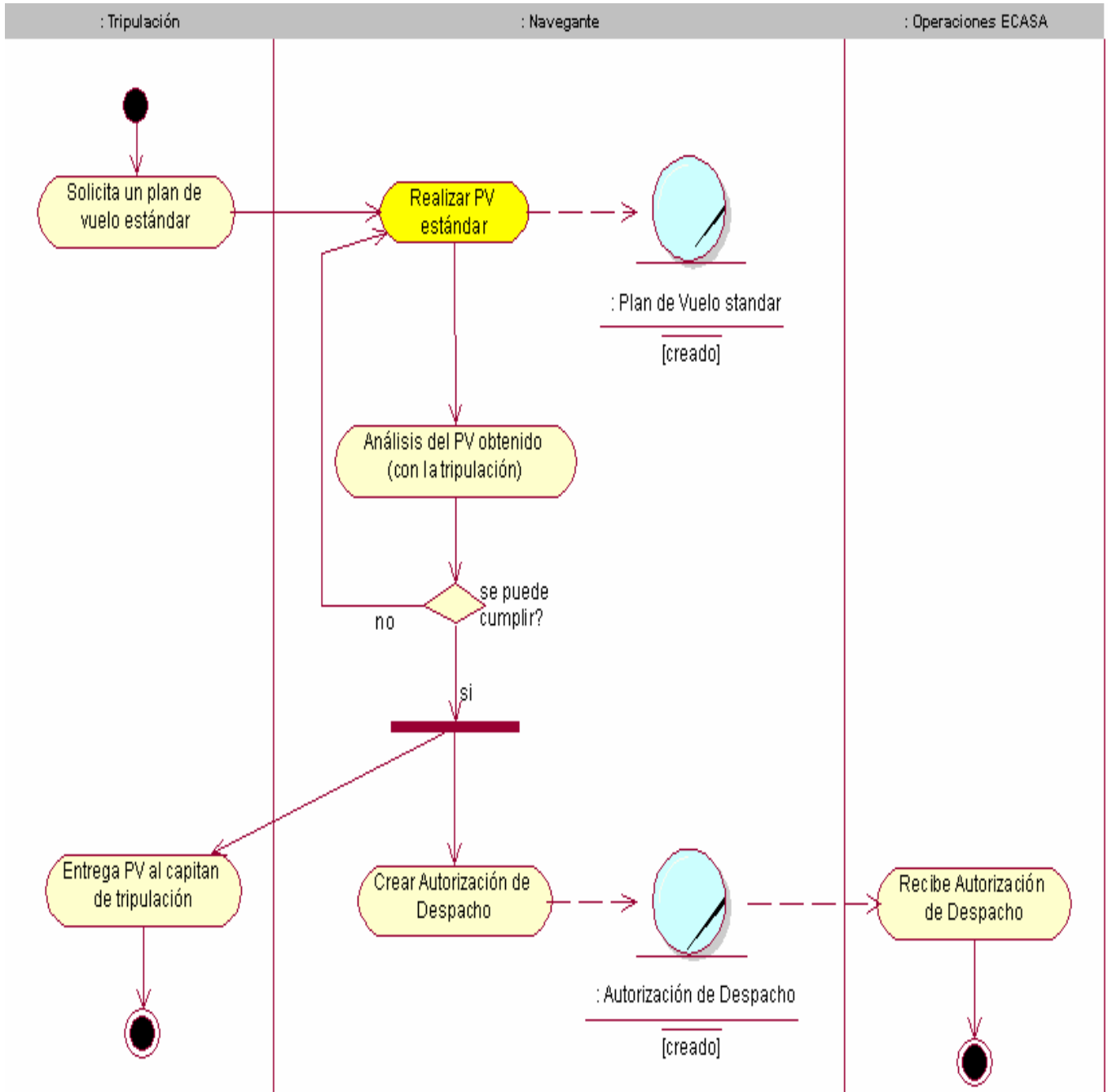
Anexo 1

1.1 Descripción de Casos de Uso <Realizar Plan de Vuelo>

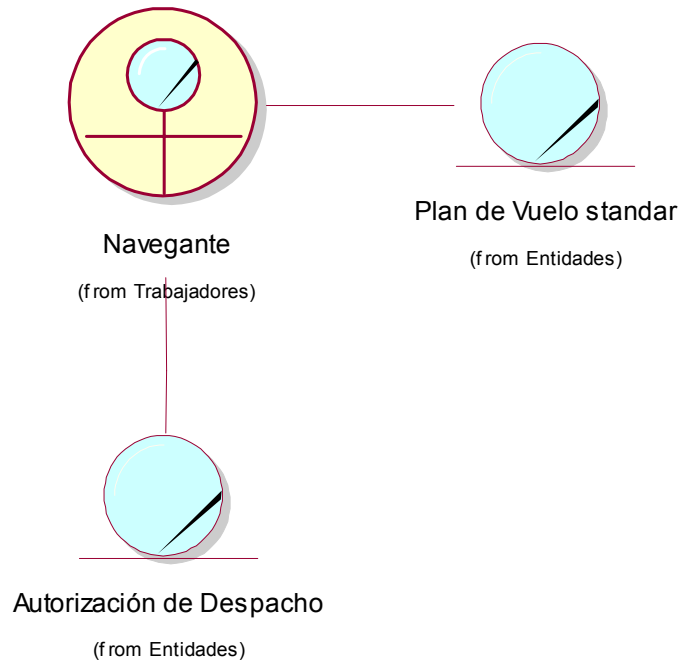
Caso de Uso:	Realizar Plan de Vuelo	
Actores:	Tripulación, Operaciones ECASA	
Trabajadores:	Navegante	
Resumen:	El caso de uso se inicia cuando se le asigna un vuelo a la tripulación para el cual necesita un plan de vuelo y no tienen conexión con SITA ¹ , por lo que es necesario hacerlo manualmente. El caso de uso finaliza con la elaboración del Plan de Vuelo.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. La tripulación tiene asignado un vuelo.		
	2. El navegante confecciona el plan de vuelo con los datos de la carga de pago que se debe transportar, la información meteorológica y la ruta a seguir. Teniendo en cuenta la información de los aeropuertos y las restricciones de NOTAMS ³ y los NACKTRACK ² .	
	3. El navegante entrega el plan de vuelo al piloto al mando y juntos hacen un análisis de los resultados.	
5. El piloto al mando de la tripulación recibe el plan de vuelo y lo analiza teniendo en cuenta las condiciones reales (las condiciones ambientales y técnicas).		
6. Si el plan de vuelo elaborado se puede cumplir bajo		

estas condiciones se realiza el vuelo.		
		7. Se le entrega a Operaciones ECASA ⁴ la hoja de Autorización de despacho y finaliza el caso de uso.
Flujos Alternos		
Acción del Actor	Respuesta del Negocio	
6. De no poderse cumplir el plan de vuelo elaborado se solicita al navegante que realice un nuevo plan de vuelo.		
	7. Se pasa al punto 2	
Mejoras	El registro de la información sobre los planes de vuelo aumentará el control y facilitará el trabajo. La automatización del proceso de realización del Plan de vuelo reduciría el tiempo de ejecución del mismo, permitiendo a los trabajadores mejorar su gestión y evitar cálculos engorrosos y errores. También dotaría al sistema de la autonomía necesaria.	
Prioridad	Responde al principal objetivo de automatización al resolver gran parte de los problemas que se presentan.	

1.2 Diagrama de Actividades < Realizar Plan de Vuelo >



1.3 Modelo de Objetos del Negocio



Anexo 2

1. Definición de los casos de uso.

1.1. Definición de Casos de Uso <Actualizar Punto de Control>

Caso de Uso:	Actualizar Pto-Control
Actores:	Navegante
Resumen:	Dada la necesita de actualizar la información de un punto de control, este caso de uso permite incorporar nuevos punto de control al sistema. Igualmente el navegante puede eliminar puntos existentes, visualizar la información de un punto específico o realizar alguna modificación a dicha información.
Referencia:	Requerimientos 2.1 ; 2.2 ; 2.3

1.2 Definición de Casos de Uso <Actualizar Aeropuerto>

Caso de Uso:	Actualizar Aeropuerto
Actores:	Navegante
Resumen:	En busca de la actualización de la información de aeropuertos, se permite introducir un nuevo aeropuerto al sistema. Igualmente el navegante puede eliminar un aeropuerto existente, visualizar la información de un aeropuerto dado y realizar alguna modificación a dicha información.
Referencia:	Requerimientos 1.1 ; 1.2 ; 1.3

1.3 Definición de Casos de Uso <Actualizar Avión>

Caso de Uso:	Actualizar Avión
Actores:	Administrador/PV
Resumen:	El administrador de plan de vuelo necesita actualizar la información de los aviones, funcionalidad que se implementa en este caso de uso, permitiendo entrar datos de nuevos aviones en el sistema. Igualmente el administrador puede eliminar o modificar la información de aviones ya existentes.
Referencia:	Requerimientos 3.1 ; 3.2 ; 3.3

1.4 Definición de Casos de Uso <Actualizar ruta fija>

Caso de Uso:	Actualizar ruta fija
Actores:	Administrador/PV
Resumen:	Permite crear una ruta fija almacenándola en el sistema. Igualmente el administrador de PV puede eliminar una ruta fija y visualizar una ruta específica permitiendo modificarla.
Referencia:	Requerimientos 4.1.1 ; 4.1.2 ; 4.1.3

1.5 Definición de Casos de Uso <Actualizar ruta provi>

Caso de Uso:	Actualizar ruta provi
Actores:	Navegante
Resumen:	Permite al navegante crear, modificar o eliminar una ruta provisional, respondiendo así a la necesidad de tener actualizada las rutas provisionales de la compañía.
Referencia:	Requerimientos 4.2.1 ; 4.2.2 ; 4.2.3

1.6 Definición de Casos de Uso <Unir ruta>

Caso de Uso:	Unir ruta
Actores:	Navegante
Resumen:	El navegante necesita construir una ruta a partir de las rutas almacenadas en el sistema, por lo que este caso de uso le brinda la opción de unir varias rutas.
Referencia:	Requerimientos 4.3

1.7 Definición de Casos de Uso <Transformar ruta>

Caso de Uso:	Transformar ruta
Actores:	Administrador/PV
Resumen:	Permite transformar una ruta fija en provisional y viceversa.
Referencia:	Requerimientos 4.4

1.8 Definición de Casos de Uso <Generar Plan de Vuelo>

Caso de Uso:	Generar Plan de Vuelo
Actores:	Navegante
Resumen:	Responde a la necesidad del navegante de obtener un plan de vuelo estándar.
Referencia:	Requerimientos 5.1

1.9 Definición de Casos de Uso <Gestionar PV >

Caso de Uso:	Gestionar PV
Actores:	Navegante
Resumen:	Después de que se haya generado previamente el plan de vuelo permite al navegante variadas opciones de trabajo con plan de vuelo, dígase imprimir y exportar, por ejemplo.
Referencia:	Requerimientos 5.2.1 ; 5.2.2

2 Casos de uso por ciclos

Cód	Nombre de caso de uso	Justificación
1	Actualizar aeropuerto	Representan aquellos procesos que sirven de soporte a las funcionalidades principales que serán implementadas en el siguiente ciclo.
	Actualizar avión	
	Actualizar puntos de control	
	Actualizar ruta provisional	
	Actualizar ruta fija	
2	Generar PV estándar	Representan los procesos primarios de la aplicación.
	Gestionar PV	
3	Unir ruta	No forman parte de las funciones prioritarias del sistema.
	Transformar ruta	

3 Casos de uso expandidos

3.1 Descripción de Casos de Uso <Actualizar Punto de Control>

Caso de Uso:	Actualizar Pto-Control
Actores:	Navegante
Resumen:	Permite incorporar un nuevo punto de control al sistema. Igualmente el navegante puede eliminar puntos ya existentes, visualizar la información de un punto específico o realizar alguna modificación de dicha información.
Referencia:	Requerimientos 2.1 ; 2.2 ; 2.3
CU asociados:	
Precondiciones:	El usuario está registrado en el sistema
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El navegante necesita registrar, eliminar y modificar los datos de los puntos de control.	
	1.1 El sistema ejecuta una de las siguientes acciones: 1 Si decide registrar un punto de control, ir a la sección "Registrar Punto de Control" 2 Si decide modificar la información de un punto de control, ir a la sección "Modificar Punto de Control". 3 Si decide eliminar un punto de control, ir a la sección "Eliminar Punto de Control".
Sección "Registrar Punto de Control"	
Acción del Actor	Respuesta del Sistema
2. El navegante entra los datos necesario.	
	2.1 El sistema valida los datos entrados
	2.2 El sistema verifica que no exista ya un punto en las coordenadas especificadas.
	2.3 El nuevo punto se almacena en el sistema.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema

	1. Se muestra un mensaje de error y vuelve a la acción 2.
	2. Si existe se muestra un mensaje informativo.
Sección “Modificar Punto de Control”	
Acción del Actor	Respuesta del Sistema
2. El navegante elige la opción de modificar puntos de control.	
	2.1 El sistema carga los puntos de control existentes en la base de datos.
3. El navegante selecciona el punto a modificar.	
	3.1 El sistema muestra los datos del punto seleccionado y brinda la posibilidad de modificarlos.
4. El navegante realiza las modificaciones pertinentes.	
	4.1 El sistema valida los datos entrados.
	4.2 Se actualiza y almacena la información del punto.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	4.1 Se muestra un mensaje de error y vuelve a la acción 3.
Sección “Eliminar Punto de Control”	
Acción del Actor	Respuesta del Sistema
2. El navegante elige la opción de eliminar puntos de control.	
	2.1 El sistema carga los puntos de control existentes en la base de datos.
3. El navegante selecciona el punto a eliminar.	
	3.1 El sistema pide confirmación.
	3.2 Elimina el punto del sistema.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	3.1 Se muestra un mensaje informativo y vuelve a la acción 2.
Poscondiciones:	- Si se introdujo un nuevo punto de control, se creó el punto de control y se

	<p>almacenó en la base de datos.</p> <p>- De haberse modificado o eliminado la información de un punto de control, el cambio se actualizó en las <i>Rutas</i> que lo contienen.</p>
Prioridad:	Media

3.2 Descripción de Casos de Uso <Actualizar Aeropuerto>

Caso de Uso:	Actualizar Aeropuerto	
Actores:	Navegante	
Resumen:	Permite introducir un nuevo aeropuerto al sistema. Igualmente el navegante puede eliminar un aeropuerto existente, visualizar la información de un aeropuerto dado o realizar alguna modificación a dicha información.	
Referencia:	Requerimientos 1.1 ; 1.2 ; 1.3	
CU asociados:		
Precondiciones:	El usuario está registrado en el sistema.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El navegante precisa registrar, eliminar o modificar aeropuertos.		
	<p>1.1 El sistema ejecuta una de las siguientes acciones:</p> <p style="margin-left: 40px;">2 Si decide registrar un aeropuerto, ir a la sección "Registrar Aeropuerto"</p> <p style="margin-left: 40px;">3 Si decide modificar un aeropuerto, ir a la sección "Modificar Aeropuerto".</p> <p style="margin-left: 40px;">4 Si decide eliminar un aeropuerto, ir a la sección "Eliminar Aeropuerto".</p>	
Sección "Registrar Aeropuerto"		
Acción del Actor	Respuesta del Sistema	
2. El navegante entra los datos necesarios.		
	2.1 El sistema valida los datos entrados.	
	2.2 El sistema verifica la existencia del aeropuerto.	

	2.3 Se almacena el nuevo aeropuerto en el sistema.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	2.1 Se muestra un mensaje de error y vuelve a la acción 2.
	2.2 Si existe se muestra un mensaje informativo.
Sección "Modificar Aeropuerto"	
Acción del Actor	Respuesta del Sistema
2. El navegante elige la opción de modificar aeropuerto.	
	2.1 El sistema carga los aeropuertos almacenados en la base de datos.
3. El navegante selecciona el aeropuerto a modificar.	
	3.1 El sistema muestra los datos del aeropuerto seleccionado brindando la posibilidad de modificarlo.
4. El navegante realiza las actualizaciones deseadas.	
	4.1 El sistema valida los datos entrados.
	4.2 Se actualiza y almacena la información del aeropuerto.
Sección "Eliminar Aeropuerto"	
Acción del Actor	Respuesta del Sistema
2. El navegante elige la opción de eliminar aeropuerto.	
	2.1 El sistema carga los aeropuertos almacenados en la base de datos.
2. El navegante selecciona el aeropuerto a eliminar.	
	2.2 El sistema pide confirmación para eliminar el aeropuerto.
	2.3 Elimina el aeropuerto del sistema.

Poscondiciones:	- De tratarse de un nuevo aeropuerto, se creó el <i>Aeropuerto</i> y se almacenó en la base de datos. - Si se eliminó un <i>Aeropuerto</i> , se actualizaron las <i>Rutas</i> que lo contienen.
Prioridad:	Media

3.3 Descripción de Casos de Uso <Actualizar Avión>

Caso de Uso:	Actualizar Avión	
Actores:	Administrador/PV	
Resumen:	Permite introducir un avión en el sistema. Igualmente el administrador de plan de vuelo puede eliminar un avión existente, visualizar la información de un avión específico o realizar alguna modificación a dicha información.	
Referencia:	Requerimientos 3.1 ; 3.2 ; 3.3	
CU asociados:		
Precondiciones:	El usuario está registrado en el sistema como administrador de plan de vuelo.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El administrador/PV precisa registrar, eliminar o modificar la información referente a un avión.		
	1.1 El sistema ejecuta una de las siguientes acciones: a) Si decide registrar un avión, ir a la sección "Registrar Avión" b) Si decide modificar un avión, ir a la sección "Modificar Avión". c) Si decide eliminar un avión, ir a la sección "Eliminar Avión".	
Sección "Registrar Avión"		
Acción del Actor	Respuesta del Sistema	
2. El administrador de PV entra los datos necesarios.		
	2.1 El sistema valida los datos entrados	
	2.2 El sistema verifica la existencia del avión.	

	2.3 Se almacena el nuevo avión en el sistema.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	1.1 Se muestra un mensaje de error y vuelve a la acción 2.
	1.2 Si existe se muestra un mensaje informativo.
Sección "Modificar Avión"	
Acción del Actor	Respuesta del Sistema
2. El administrador de PV selecciona la acción de modificar.	
	2.1 El sistema carga los aviones existentes.
2. El administrador de PV selecciona el avión a modificar.	
	2.1 El sistema muestra los datos del avión seleccionado brindando la posibilidad de modificarlo.
3. El administrador de PV realiza las actualizaciones deseadas.	
	3.1 El sistema valida los datos entrados.
	3.2 Se actualiza y almacena la información del avión.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	3.1 Se muestra un mensaje de error y vuelve a la acción 3.
Sección "Eliminar Avión"	
Acción del Actor	Respuesta del Sistema
2. El administrador elige la opción de eliminar.	
	2.1 El sistema carga los aviones existentes en la base de datos.
3. El administrador de PV selecciona el avión a eliminar.	
	3.1 El sistema pide confirmación para eliminar.
	3.2 Elimina el avión seleccionado.

Poscondiciones:	Si se trata de un nuevo avión, se creó el <i>Avión</i> , de ser modificarse o eliminarse la información del mismo esta se actualizó en el sistema.
Prioridad:	Media

3.4 Descripción de Casos de Uso <Actualizar ruta fija>

Caso de Uso:	Actualizar ruta fija	
Actores:	Administrador/PV	
Resumen:	Permite crear una ruta fija almacenándola en el sistema. Igualmente el administrador de PV puede eliminar una ruta fija y visualizar una ruta específica permitiendo modificarla.	
Referencia:	Requerimientos 4.1.1 ; 4.1.2 ; 4.1.3	
CU asociados:		
Precondiciones:	El usuario está registrado en el sistema como administrador de plan de vuelo. El sistema tiene registrados los puntos de control y las aerovías.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El administrador de PV requiere registrar, eliminar o modificar una ruta fija.		
	1.1 El sistema ejecuta una de las siguientes acciones: <ul style="list-style-type: none"> a) Si decide registrar una ruta fija, ir a la sección "Registrar Ruta Fija" b) Si decide modificar una ruta fija, ir a la sección "Modificar Ruta Fija". c) Si decide eliminar una ruta fija, ir a la sección "Eliminar Ruta Fija". 	
Sección "Registrar Ruta Fija"		
Acción del Actor	Respuesta del Sistema	
2. El administrador de PV entra los datos.		
	2.1 El sistema carga los puntos de control.	
	2.2 Verifica los datos entrados	

	2.2 El sistema verifica la existencia de la ruta fija.
	2.4 Se almacena la nueva ruta fija en el sistema.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	1.1 Se muestra un mensaje de error y vuelve a la acción 2.
	1.2 Si existe se muestra un mensaje informativo.
Sección "Modificar Ruta Fija"	
Acción del Actor	Respuesta del Sistema
2. El administrador opta por la opción de modificar una ruta de fija.	
	2.1 El sistema carga las rutas fijas.
	2.1 El sistema carga los puntos de control existentes en la aplicación.
3. El administrador selecciona la ruta fija a modificar.	
	2.2 El sistema muestra los datos de la ruta fija seleccionada brindando la posibilidad de modificarlos.
3. El administrador introduce los nuevos datos.	
	3.1 El sistema valida los datos entrados.
	3.2 Se actualiza y almacena la información de la ruta fija.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	3.1 Se muestra un mensaje de error y vuelve a la acción 3.
Sección "Eliminar Ruta Fija"	
Acción del Actor	Respuesta del Sistema
2. Selecciona la ruta fija a eliminar.	
	2.2 El sistema pide confirmación para eliminarla.
	2.3 Elimina la ruta fija especificada.

Poscondiciones:	<ul style="list-style-type: none"> - En caso de una nueva ruta fija, se creó la <i>Ruta fija</i> y se almacenó en la base de datos. Se asociaron varias instancias de <i>Puntos de Control</i> a la <i>Ruta fija</i> creada. - De haberse modificado o eliminado una <i>Ruta fija</i>, la modificación se hizo efectiva en la asociación con los <i>Puntos de Control</i> correspondientes.
Prioridad:	Media

3.5 Descripción de Casos de Uso <Actualizar ruta provi>

Caso de Uso:	Actualizar ruta provi	
Actores:	Navegante	
Resumen:	Permite al navegante crear, modificar o eliminar una ruta provisional.	
Referencia:	Requerimientos 4.2.1 ; 4.2.2 ; 4.2.3	
CU asociados:		
Precondiciones:	El usuario está registrado en el sistema como administrador de plan de vuelo. El sistema tiene registrados los puntos de control y las aerovías.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El navegante necesita crear, eliminar o modificar una ruta provisional.		
	1.1 El sistema ejecuta una de las siguientes acciones: <ul style="list-style-type: none"> a) Si decide registrar una ruta provisional, ir a la sección "Registrar Ruta Provisional" b) Si decide modificar una ruta provisional, ir a la sección "Modificar Ruta Provisional". c) Si decide eliminar una ruta provisional, ir a la sección "Eliminar Ruta Provisional". 	
Sección "Registrar Ruta Provisional"		
Acción del Actor	Respuesta del Sistema	
2. El navegante entra los datos necesarios.		
	2.1 El sistema verifica los datos entrados.	
	2.2 El sistema carga los puntos de control existentes	

	en la aplicación.
	2.3 El sistema verifica la existencia de la ruta provisional.
	2.4 Se almacena la nueva ruta provisional en el sistema.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	2.1 Se muestra un mensaje de error y vuelve a la acción 2.
	3.6 Si existe se muestra un mensaje informativo.
Sección "Modificar Ruta Provisional"	
Acción del Actor	Respuesta del Sistema
2. El usuario elige modificar una ruta provisional.	
	2.1 El sistema carga las rutas provisionales existentes.
3. Selecciona la ruta provisional a modificar.	
	3.1 El sistema carga los puntos de control.
	3.2 El sistema muestra los datos de la ruta provisional seleccionada dando la opción de modificarlos.
4. El navegante introduce los nuevos datos.	
	4.1 El sistema valida los datos entrados.
	4.2 Se actualiza y almacena la información de la ruta provisional.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	4.1 Muestra mensaje de error y vuelve a acción 3.
Sección "Eliminar Ruta Provisional"	
Acción del Actor	Respuesta del Sistema
2. Selecciona la opción de eliminar.	
	2.1 Carga las rutas provisionales existentes.

3. Selecciona la ruta provisional a eliminar.	
	3.1 El sistema pide confirmación para eliminarla.
	3.2 Elimina la ruta provisional especificada.
Poscondiciones:	<ul style="list-style-type: none"> - En caso de una nueva ruta provisional, se creó la <i>Ruta provisional</i> y se almacenó en la base de datos. Se asociaron varias instancias de <i>Puntos de Control</i> a la <i>Ruta provisional</i> creada. - De haberse modificado o eliminado una <i>Ruta provisional</i>, la modificación se hizo efectiva en la asociación con los <i>Puntos de Control</i> correspondientes.
Prioridad:	Media

3.6 Descripción de Casos de Uso <Unir rutas>

Caso de Uso:	Unir rutas
Actores:	Navegante
Resumen:	Brinda al navegante la opción de unir dos rutas existentes en la base de datos y trabajar el resultado de esta unión como una nueva ruta.
Referencia:	Requerimientos 4.3
CU asociados:	
Precondiciones:	<p>El usuario está registrado en el sistema.</p> <p>El sistema tiene almacenadas las rutas a unificar.</p>
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
2. El navegante escoge unificar varias rutas.	
	2.1 Carga las rutas existentes
3. Selecciona las rutas que desea unir.	
	3.1 El sistema une las rutas seleccionadas.
	3.2 Se guarda el resultado de la unión como una ruta provisional
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	2.2 En caso de ser un administrador se puede almacenar la nueva ruta como cualquiera de los 2

	tipos (provisional o fija).
Poscondiciones:	<ul style="list-style-type: none"> - Se creó una nueva <i>Ruta fija</i> o <i>provisional</i> según el caso. Se estableció la asociación entre la nueva Ruta y los <i>Puntos de Control</i> contenidos en las <i>Rutas</i> unificadas. - De haberse eliminado o modificado la <i>Ruta</i>, las modificaciones actualizaron la asociación con los <i>Puntos de Control</i> de la <i>Ruta</i>.
Prioridad:	Baja

3.7 Descripción de Casos de Uso <Transformar ruta>

Caso de Uso:	Transformar ruta
Actores:	Administrador/PV
Resumen:	Permite transformar una ruta fija en provisional y viceversa.
Referencia:	Requerimientos 4.4
CU asociados:	
Precondiciones:	<p>El usuario está registrado en el sistema como administrador.</p> <p>El sistema tiene almacenada la ruta.</p>
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. Escoge la opción de transformar una ruta	
	1.1 Carga todas las rutas.
2 El usuario especifica la ruta a transformar.	
	1.2 Transforma la categoría de la ruta especificada.
Poscondiciones:	Se le otorgó una nueva categoría a la ruta.
Prioridad:	Baja

3.8 Descripción de Casos de Uso <Generar PV >

Caso de Uso:	Generar PV
Actores:	Navegante

Resumen:	Permite al navegante del sistema construir un plan de vuelo.
Referencia:	Requerimientos 5.1
CU asociados:	
Precondiciones:	El usuario está registrado en el sistema, por lo que se conoce el nombre del navegante que va a general el plan de vuelo. El sistema tiene almacenadas la ruta, el avión y los aeropuertos (de origen y destino) y tiene en cuenta las políticas de combustible y de alternos.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1- El usuario selecciona la opción de generar Plan de Vuelo.	
	1.1 Carga todo los datos necesarios de la base de datos.
	1.2 El sistema llena los campos necesarios.
2- El usuario selecciona los datos y llenar los restantes campos.	
	2.1 El sistema valida los datos.
	2.2 El sistema realiza los cálculos y genera el plan de vuelo.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
	2.1 Levanta un mensaje de error y vuelve al paso 1.
Poscondiciones:	Se generó un <i>Plan de Vuelo Operativo</i> y se asocia con la <i>Ruta</i> a volar, los <i>Aeropuertos</i> (de origen y destino) y el <i>Avión</i> que se utilizará para el vuelo.
Prioridad:	Alta

3.9 Descripción de Casos de Uso <Gestionar PV >

Caso de Uso:	Gestionar PV
Actores:	Navegante
Resumen:	Brinda al navegante variadas opciones de trabajo con plan de vuelo.

Referencia:	Requerimientos 5.2.1 ; 5.2.2
CU asociados:	
Precondiciones:	El usuario debe estar registrado en el sistema. El plan de vuelo debe haber sido generado previamente.
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El navegante necesita exportar o imprimir un plan de vuelo.	
	1.1 El sistema ejecuta una de las siguientes acciones: 2 Si decide exportar un plan de vuelo, ir a la sección "Exportar Plan de Vuelo". 3 Si decide imprimir plan de vuelo, ir a la sección "Imprimir Plan de Vuelo".
Sección "Exportar Plan de Vuelo"	
Acción del Actor	Respuesta del Sistema
2. El navegante decide exportar el plan de vuelo.	
	2.1 Exporta el plan de vuelo a otro formato.
Sección "Imprimir Plan de Vuelo"	
Acción del Actor	Respuesta del Sistema
2. El navegante decide imprimir un plan de vuelo.	
	2.1 El sistema imprime el plan de vuelo.
Flujos Alternos	
Acción del Actor	Respuesta del Sistema
Poscondiciones:	
Prioridad:	Alta

Anexo 3

Diagrama de Secuencia por Realización de CU²¹

Insertar Avión

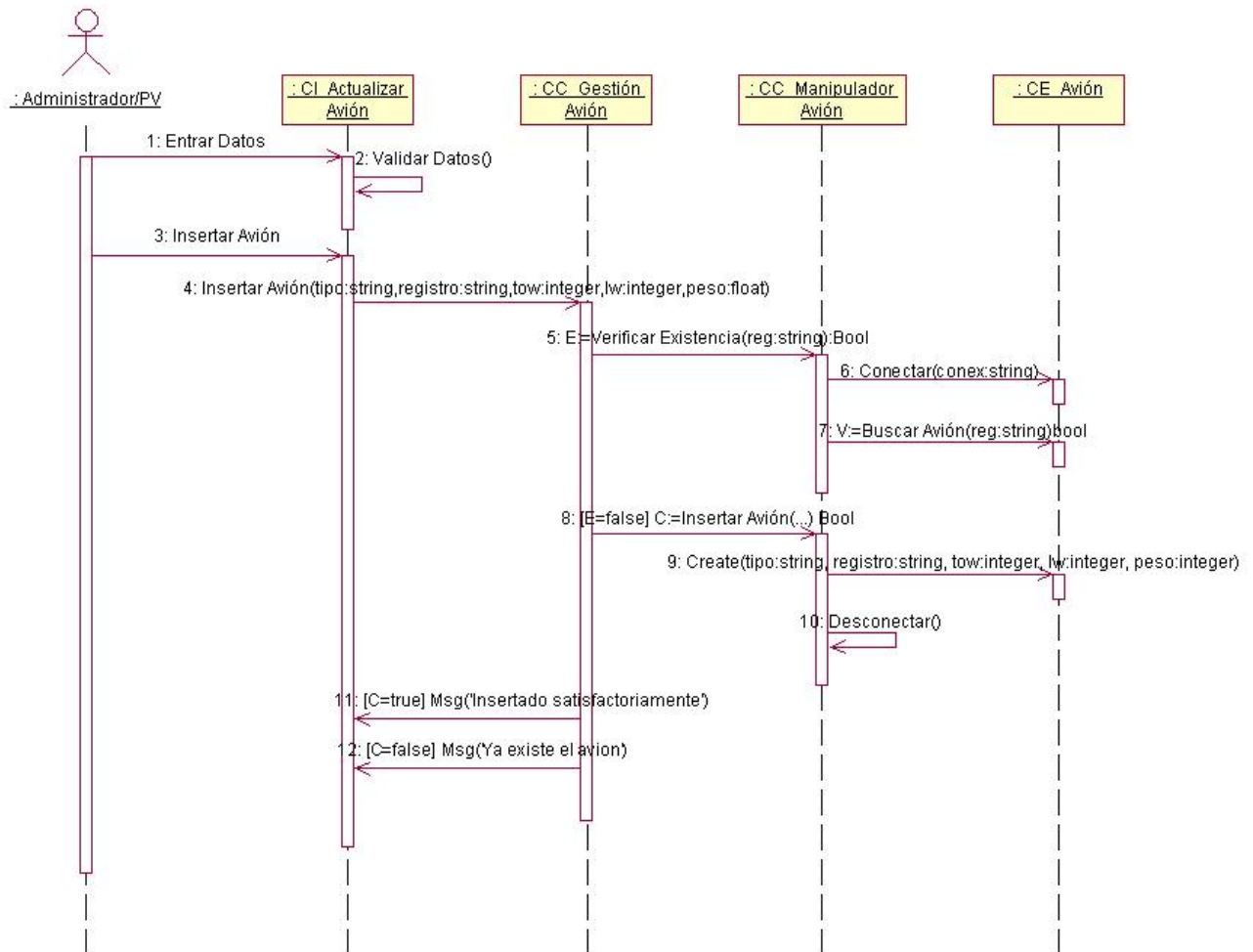


Imagen 1: Insertar Avión.

The image shows a software window titled "Sistema para la Gestión del Plan de vuelo IL-96 300". Inside the window, there is a section labeled "Opciones" (Options). This section contains a form for adding an aircraft, titled "Avion". The form includes five text input fields: "Tipo", "Registro", "TOW", "LW", and "PBO". Below these fields are two buttons: "Crear" (Create) and "Cancelar" (Cancel).

Modificar Avión

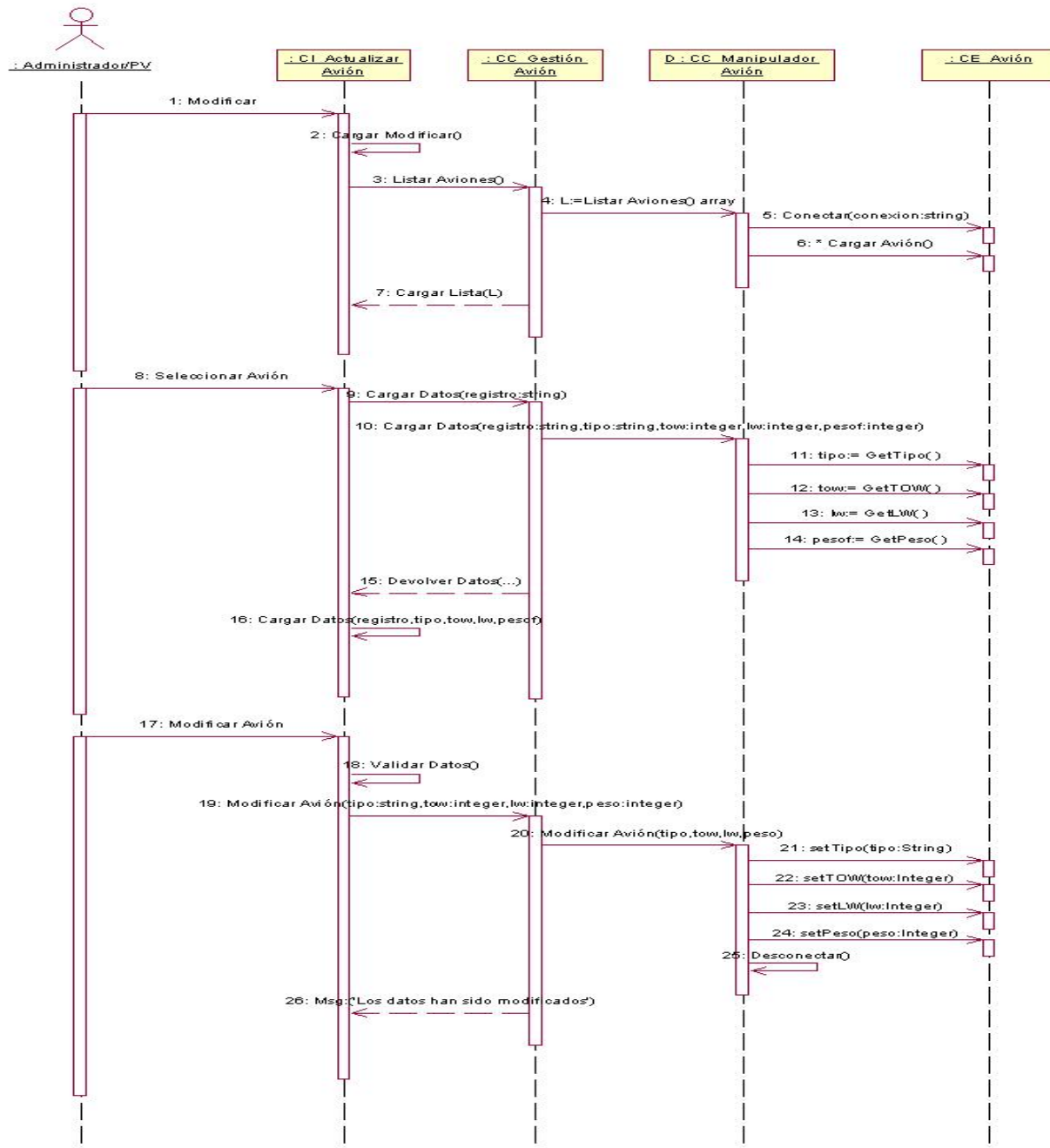


Imagen 2: Modificar Avión.

The image shows a software window titled "Sistema para la Gestion del Plan de vuelo IL-96 300". Inside the window, there is a sub-dialog titled "Opciones" with the following elements:

- A label "Modificar/Eliminar Avion" above a dropdown menu containing "CU6988".
- Two radio buttons: "Modificar" (checked) and "Eliminar" (unchecked).
- Text input fields for "Tipo" (containing "IL96"), "Registro" (containing "CU6988"), "TOW" (containing "159000"), "LW" (containing "174000"), and "PBO" (containing "4000").
- Two buttons at the bottom: "Guardar Cambios" and "Cancelar".

Eliminar Avión

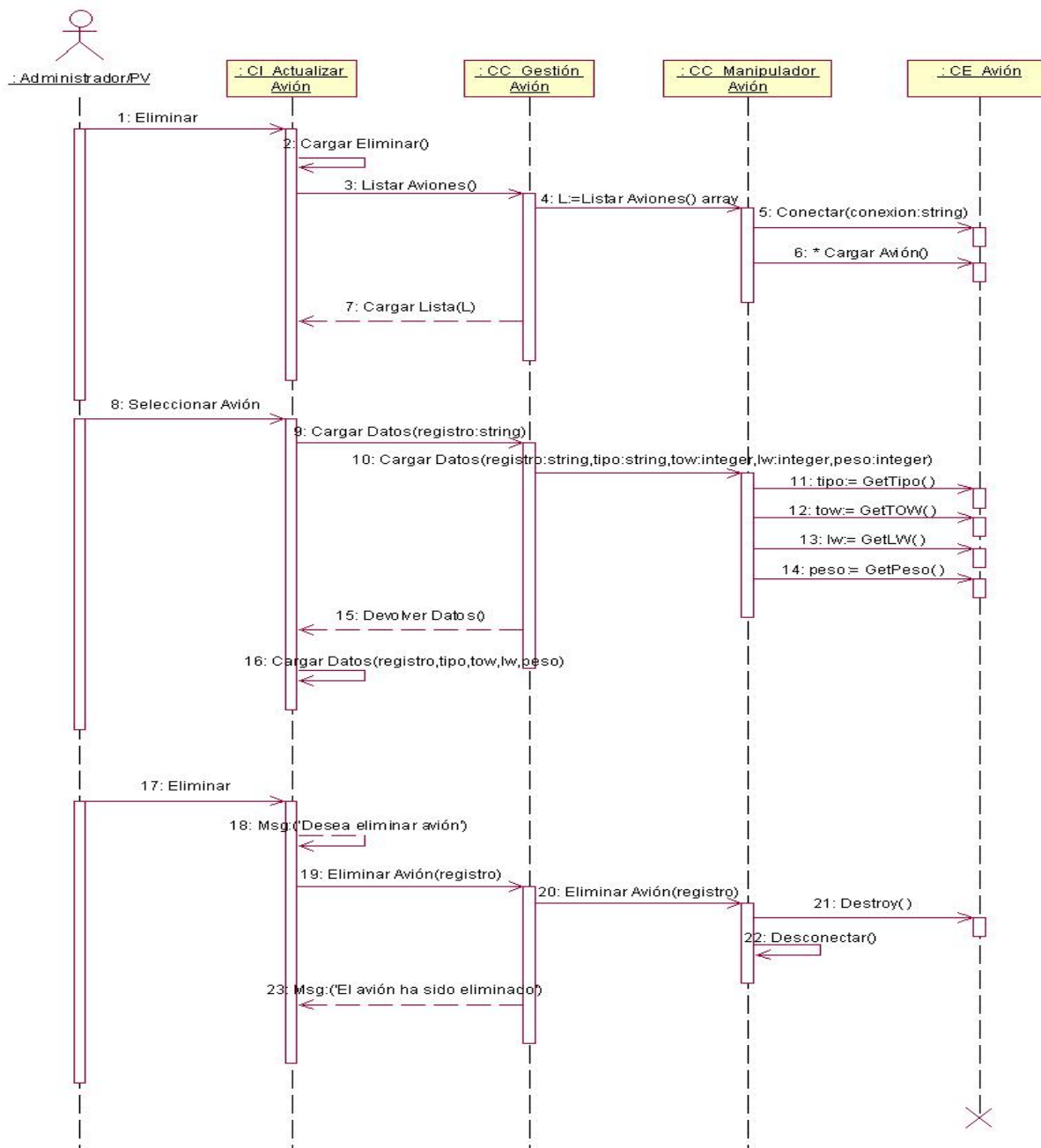


Imagen 3: Eliminar Avión

Sistema para la Gestion del Plan de vuelo IL-96 300

Opciones

Modificar/Eliminar Avion

Registro

Modificar Eliminar

Tipo

Registro

TDW

LW

PBO

Insertar Aeropuerto

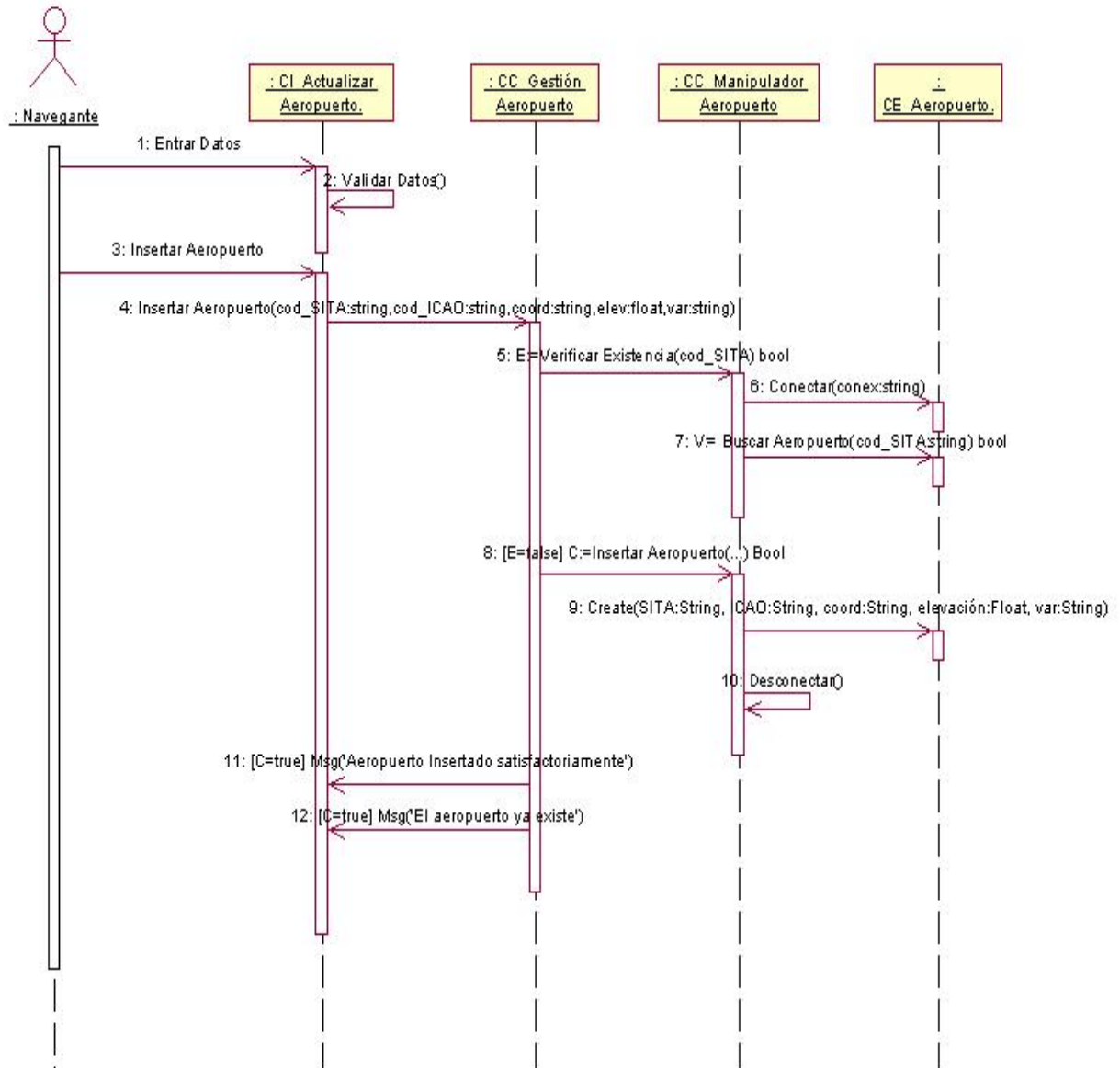


Imagen 4: Insertar Aeropuerto.

The image shows a screenshot of a software application window titled "Sistema para la Gestion del Plan de vuelo IL-96 300". Inside the window, there is a sub-window titled "Opciones". The "Opciones" window contains a form for adding an airport, with the following fields and buttons:

- Aeropuerto
- Nombre (SITA)
- Nombre (OASI)
- Latitud
- Longitud
- Variacion
- Elevacion
- Crear
- Cancelar

Modificar Aeropuerto

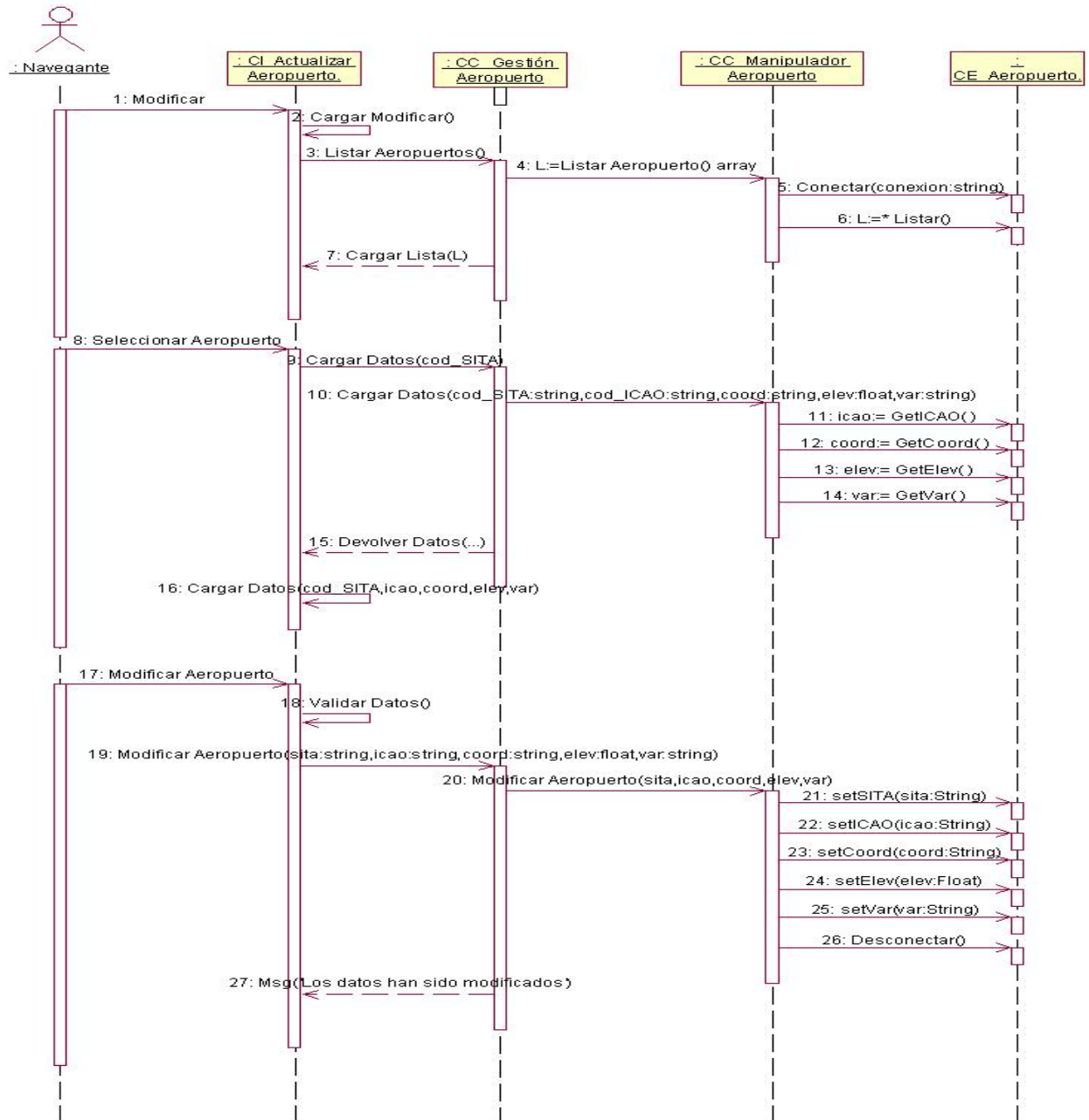


Imagen 5: Modificar Aeropuerto.

The screenshot shows a software window titled "Sistema para la Gestion del Plan de vuelo IL-96 300" with a sub-dialog box titled "Opciones". The dialog box is titled "Modificar/Eliminar Aeropuerto" and contains the following elements:

- A dropdown menu for "Nombre (SITA)" with "MUHA" selected.
- Two checkboxes: "Modificar" (checked) and "Eliminar" (unchecked).
- Text input fields for:
 - "Nombre (SITA)" containing "MUHA"
 - "Nombre (DASI)" containing "HVA"
 - "Latitud" containing "N2259.3"
 - "Longitud" containing "W08224.6"
 - "Variacion" containing "56E"
 - "Elevacion" containing "23"
- Two buttons at the bottom: "Guardar Cambios" and "Cancelar".

Eliminar Aeropuerto

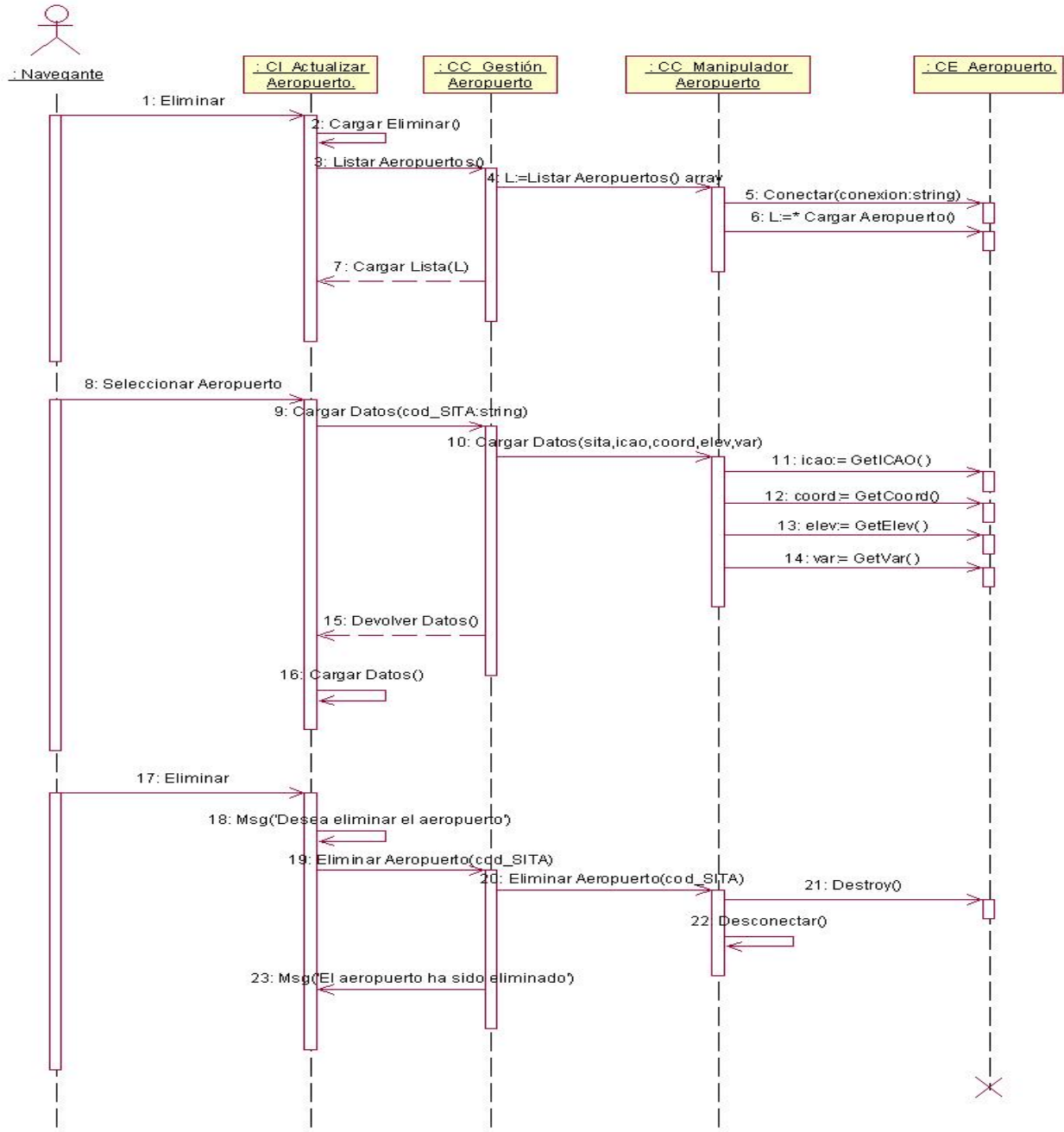
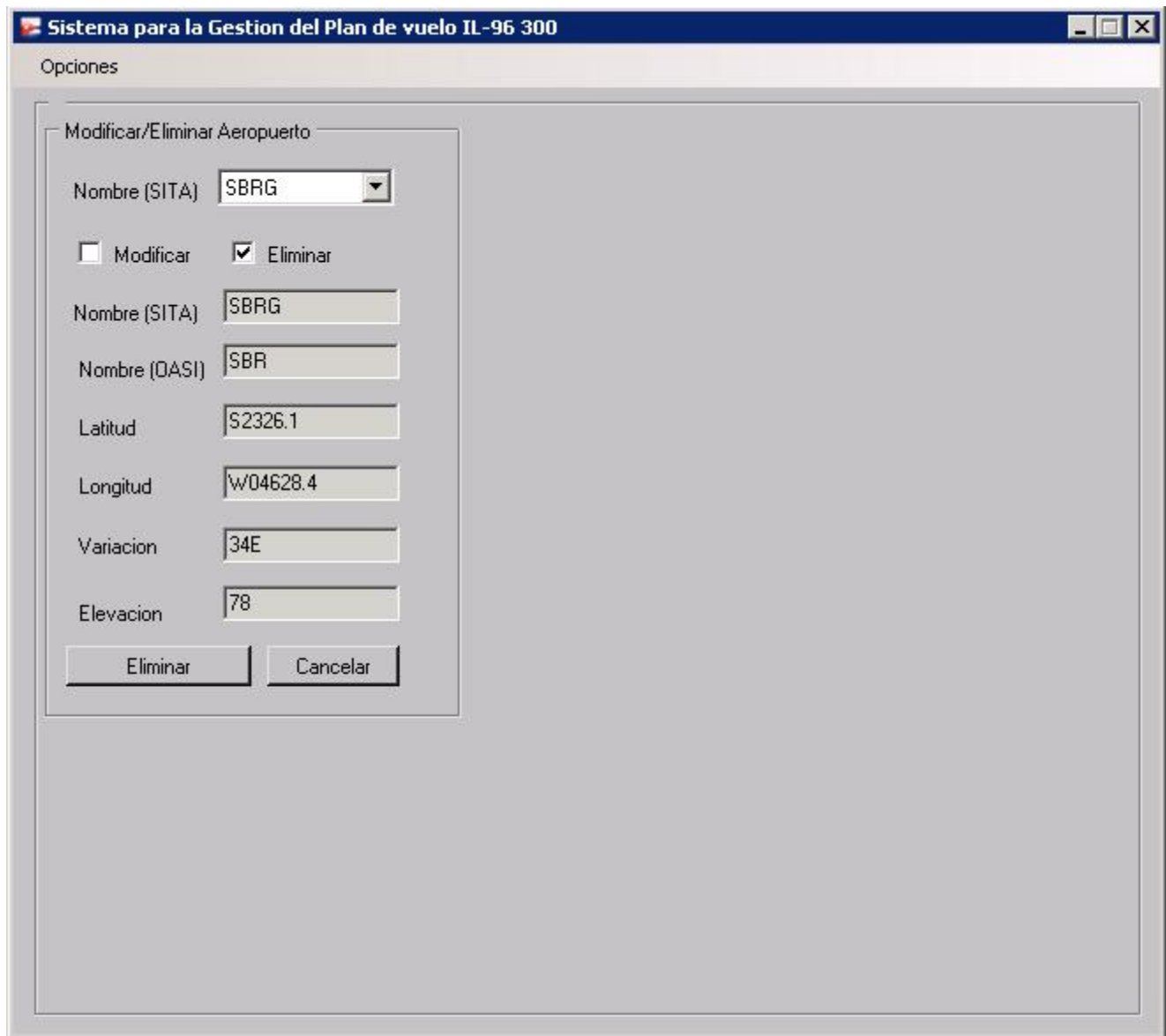


Imagen 6: Eliminar Aeropuerto.



Insertar Puntos de Control

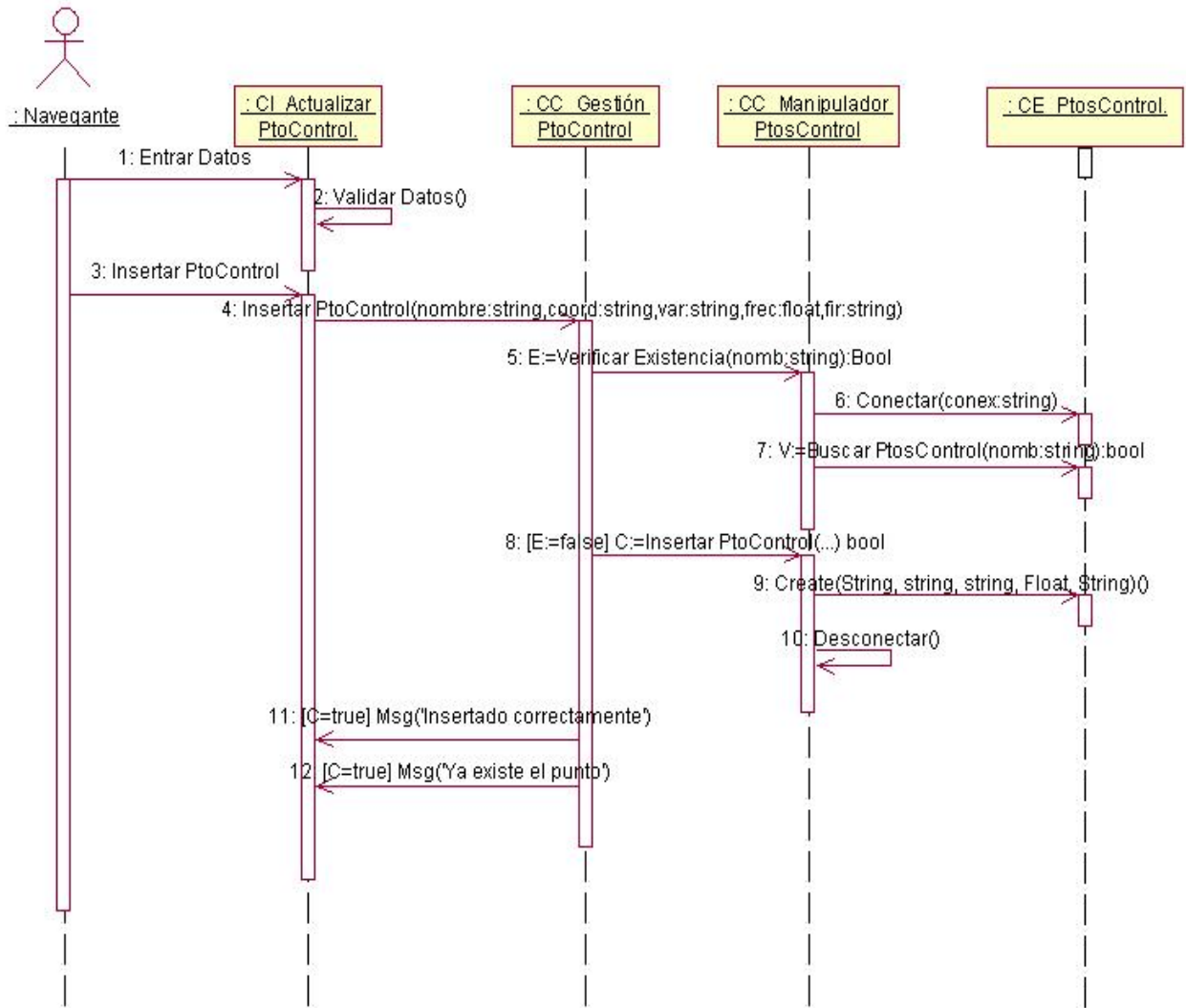


Imagen 7: Insertar Puntos de Control.

The image shows a software window titled "Sistema para la Gestion del Plan de vuelo IL-96 300". Inside the window, there is a dialog box titled "Opciones". The dialog box contains a section titled "Puntos de Control" with the following fields and buttons:

- Nombre:
- Latitud:
- Longitud:
- Variacion:
- Frecuencia:
- FIR:
- Crear:
- Cancelar:

Modificar Punto de Control

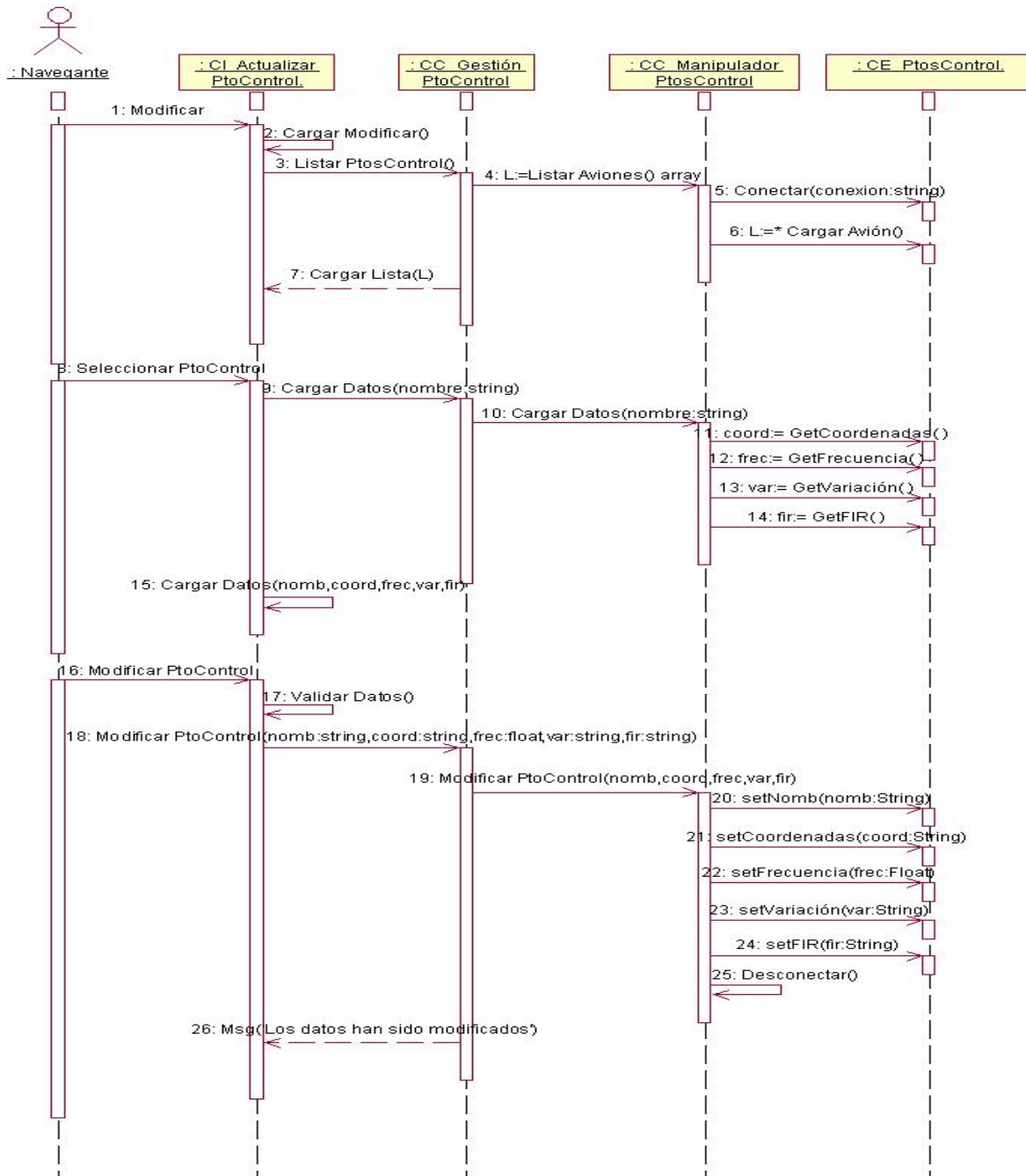


Imagen 8: Modificar Punto de Control.

The image shows a software window titled "Sistema para la Gestion del Plan de vuelo IL-96 300" with a sub-window titled "Opciones". The "Opciones" window contains a section titled "Modificar/Eliminar Puntos de Control".

Within this section, there are the following elements:

- A dropdown menu for "Nombre" with "PLUMA" selected.
- Two radio buttons: "Modificar" (checked) and "Eliminar" (unchecked).
- Text input fields for:
 - "Nombre": PLUMA
 - "Latitud": N2444.3
 - "Longitud": W07805.8
 - "Variacion": 23E
 - "Frecuencia": 4654
 - "FIR": (empty)
- Two buttons at the bottom: "Guardar Cambios" and "Cancelar".

Eliminar Punto de Control

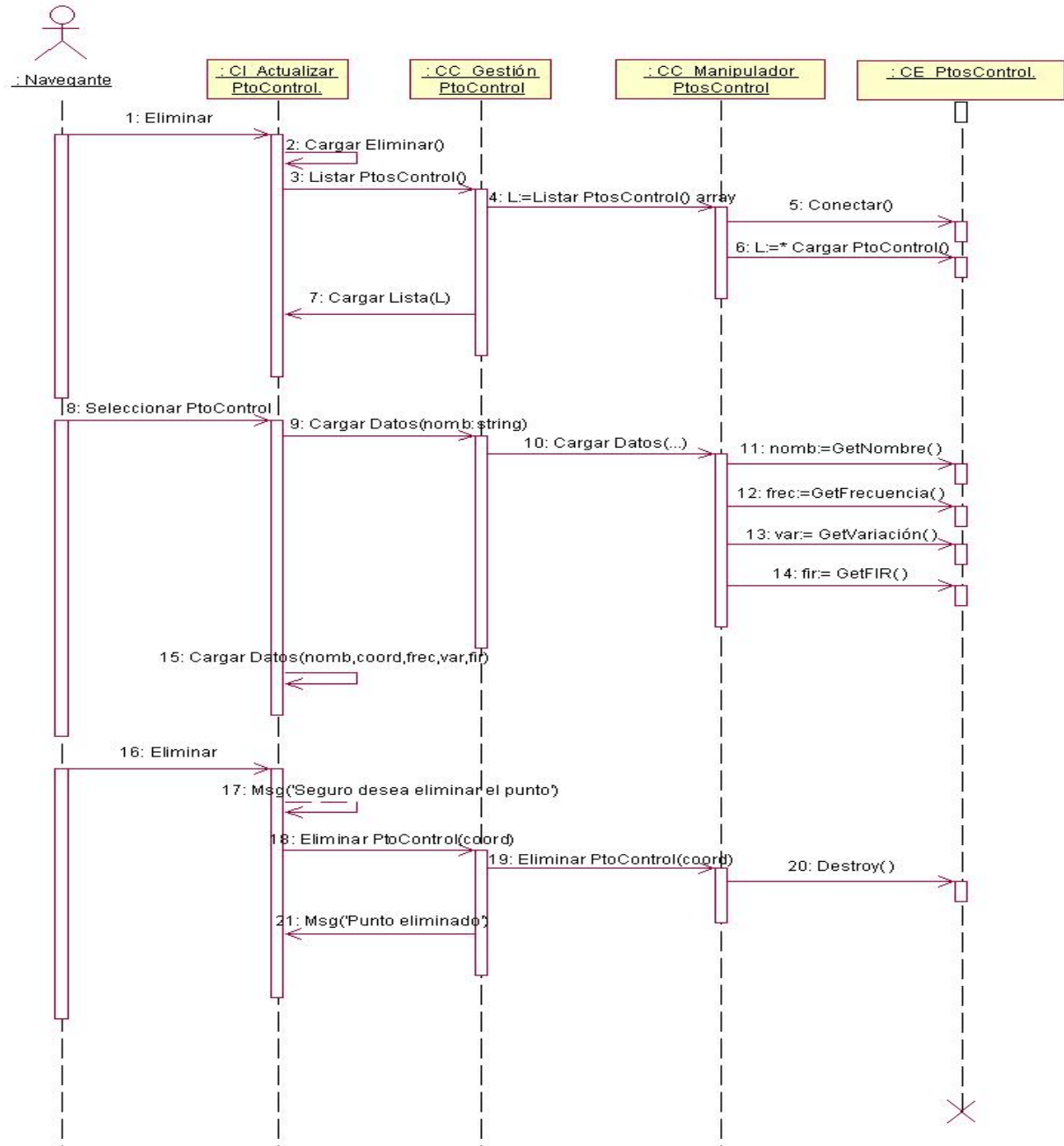


Imagen 9: Eliminar Punto de Control.

The image shows a software window titled "Sistema para la Gestion del Plan de vuelo IL-96 300" with a sub-dialog box titled "Opciones". The dialog box contains a section for "Modificar/Eliminar Puntos de Control".

Fields and options in the dialog:

- Nombre: PLUMA (dropdown menu)
- Modificar Eliminar
- Nombre: PLUMA (text field)
- Latitud: N2444.3 (text field)
- Longitud: W07805.8 (text field)
- Variacion: 23E (text field)
- Frecuencia: 4654 (text field)
- FIR: (empty text field)
- Buttons: Eliminar, Cancelar

Insertar Ruta

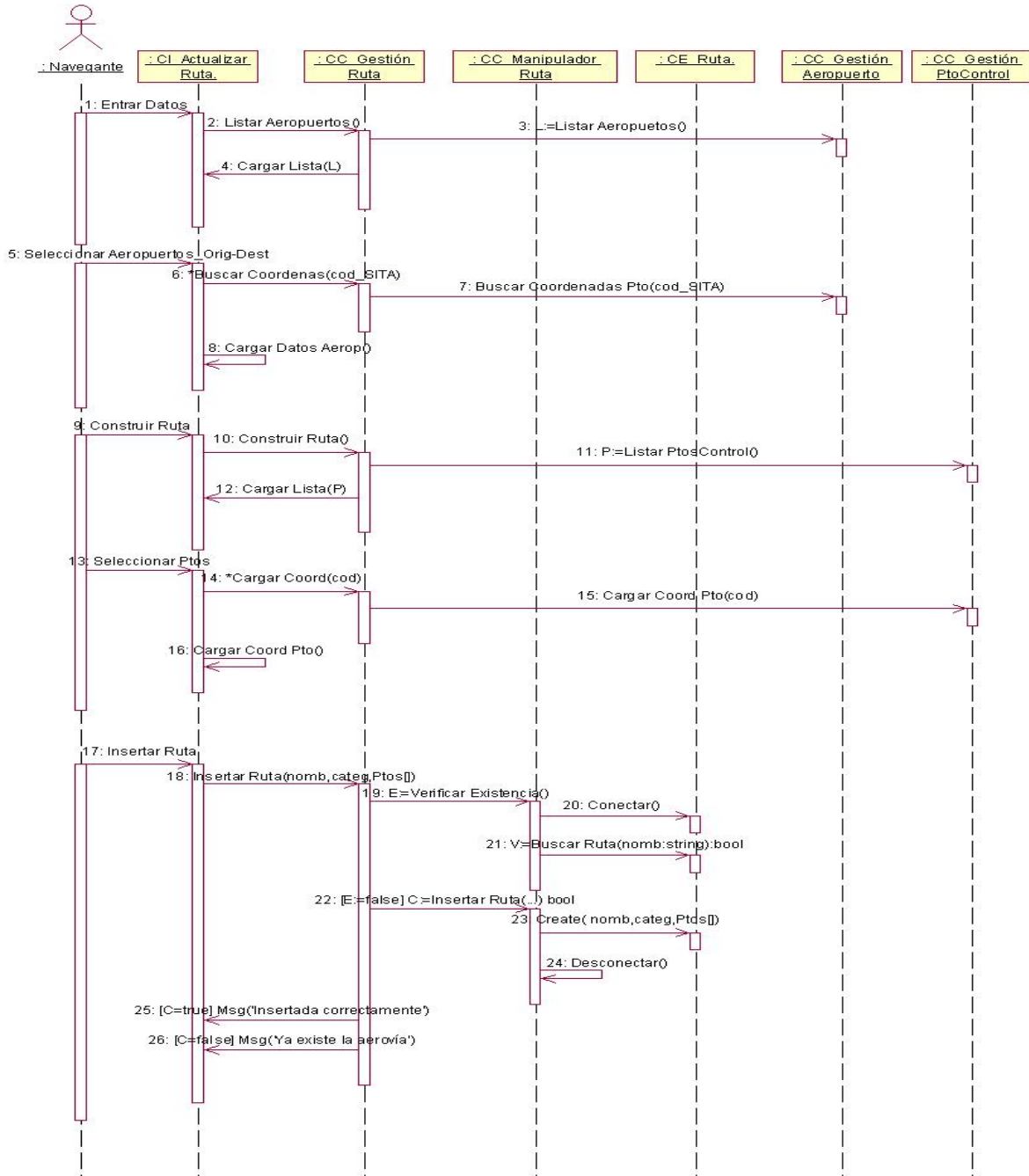


Imagen 10: Insertar Ruta.

Sistema para la Gestion del Plan de vuelo IL-96 300

Opciones

Crear Ruta

Nombre de la Ruta
 MUHA UZG DARLO SAEZ

Categoría
 Ruta Fija Ruta Provisional

Aeropuertos

Origen
 MUHA
 W08224.6;N2259.3

Destino
 SAEZ
 W05832.2;S3449.3

Agregar Puntos de Control

Puntos existentes		Puntos de la ruta
PLUMA (W07805.8;N2444.3)		UZG (W08202.3;N2256.1)
MENDL (W07840.7;N2427.2)		UCL (W08132.0;N2136.3)
ZQA (W07728.2;N2502.4)		IBSEN (W08014.3;N2031.0)
BARTS (W07730.4;N2559.1)		UVA (W08127.2;N2301.5)
ABACO (W07734.2;N2700.0)	7 >>	ZOLLA (W07906.1;N2414.6)
NUCAR (W07738.0;N2807.6)	<<	DARLO (W05522.1;S0512.5)
HOOKY (W07728.4;N2812.9)		
LOUIZ (W07700.0;N2828.4)		
SUMRS (W07633.5;N2842.7)		
BURTT (W07313.0;N3140.1)		

Ruta MUHA UZG UCL IBSEN UVA ZOLLA DARLO SAEZ

Guardar Cancelar

Modificar Ruta

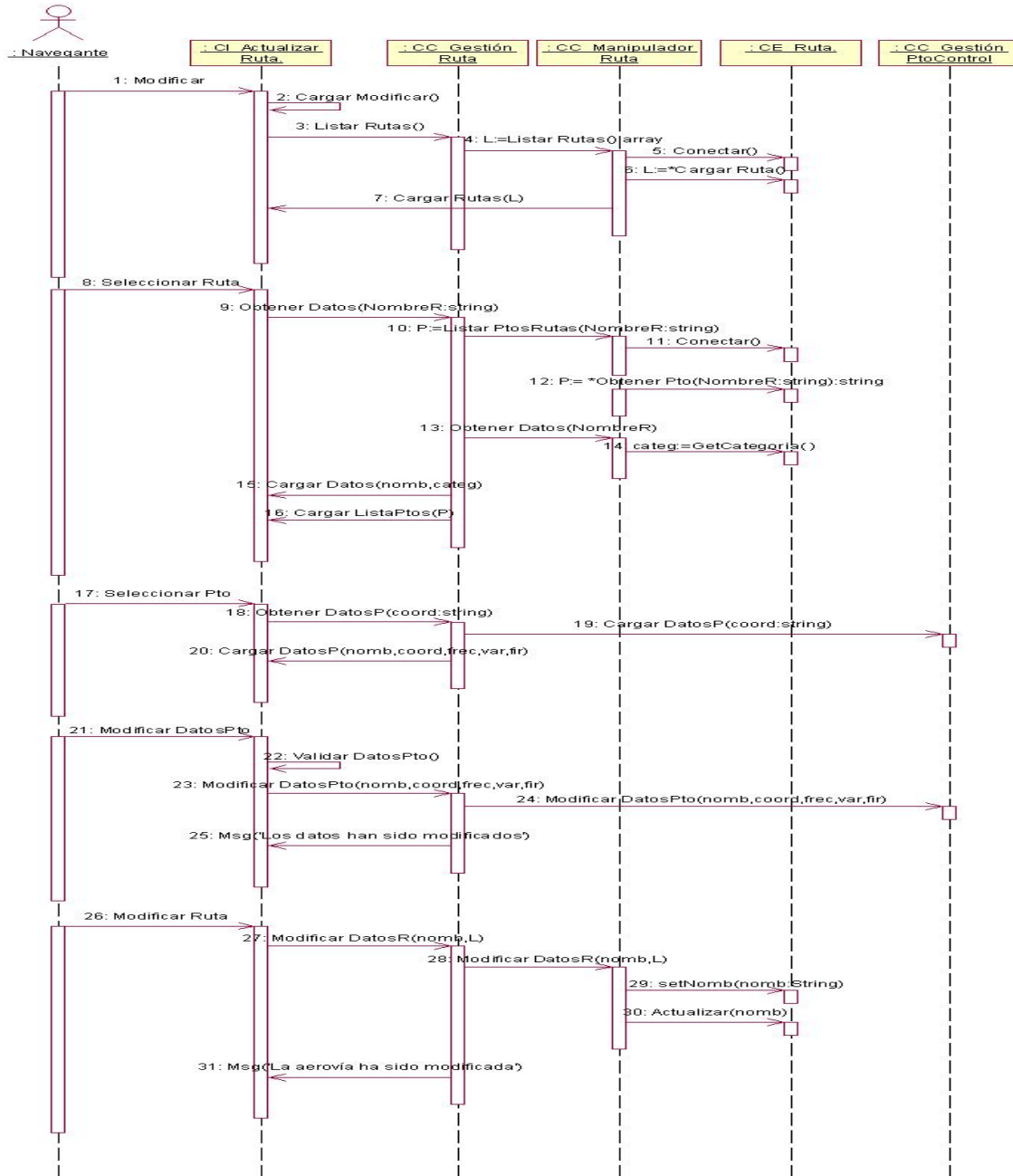


Imagen 11: Modificar Ruta.

Sistema para la Gestion del Plan de vuelo IL-96 300

Opciones

Modificar Ruta

Aeropuertos

Origen: MUHA
Destino: CYQX

W08224.6;N2259.3 W05434.8;N4856.2

Nombre de la Ruta: MUHA ODEAL HOOKY CYQX

Categoría: Ruta Fija Ruta Provisional

Agregar Puntos de Control

Puntos existentes		Puntos de la ruta
UZG (w08202.3;N2256.1)	6 >>	ODEAL (w07128.9;N3325.9)
UCL (w08132.0;N2136.3)		BURTT (w07313.0;N3140.1)
IBSEN (w08014.3;N2031.0)	<<	SUMRS (w07633.5;N2842.7)
UVA (w08127.2;N2301.5)		LOUIZ (w07700.0;N2828.4)
ZOLLA (w07906.1;N2414.6)		HOOKY (w07728.4;N2812.9)
DARLO (w05522.1;S0512.5)		
PLUMA (w07805.8;N2444.3)		
MENDL (w07840.7;N2427.2)		
ZQA (w07728.2;N2502.4)		
BARTS (w07730.4;N2559.1)		

Ruta: MUHA ODEAL BURTT SUMRS LOUIZ HOOKY CYQX

Guardar Cancelar

Eliminar Ruta

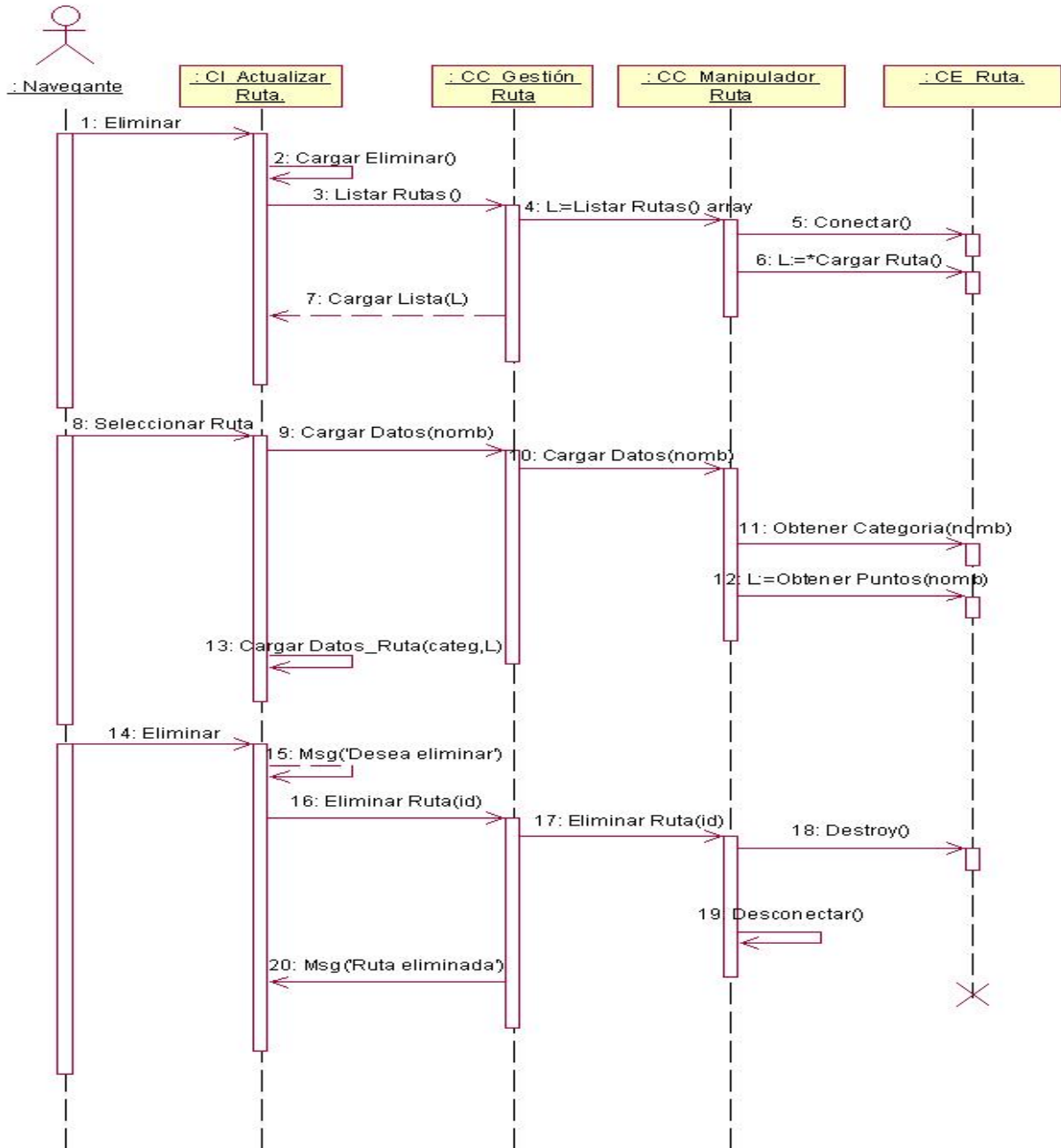


Imagen 12: Eliminar Ruta.

Sistema para la Gestion del Plan de vuelo IL-96 300

Opciones

Eliminar Rutas

Origen (ICAO): Destino (ICAO):

Rutas Fijas

Ruta
<input type="checkbox"/> MUHA ZOLLA PLUMA CYQX
<input checked="" type="checkbox"/> MUHA MENDL ZQA CYQX

Rutas Provisionales

Ruta
<input type="checkbox"/> MUHA ODEAL HOOKY CYQX

Plan de Vuelo

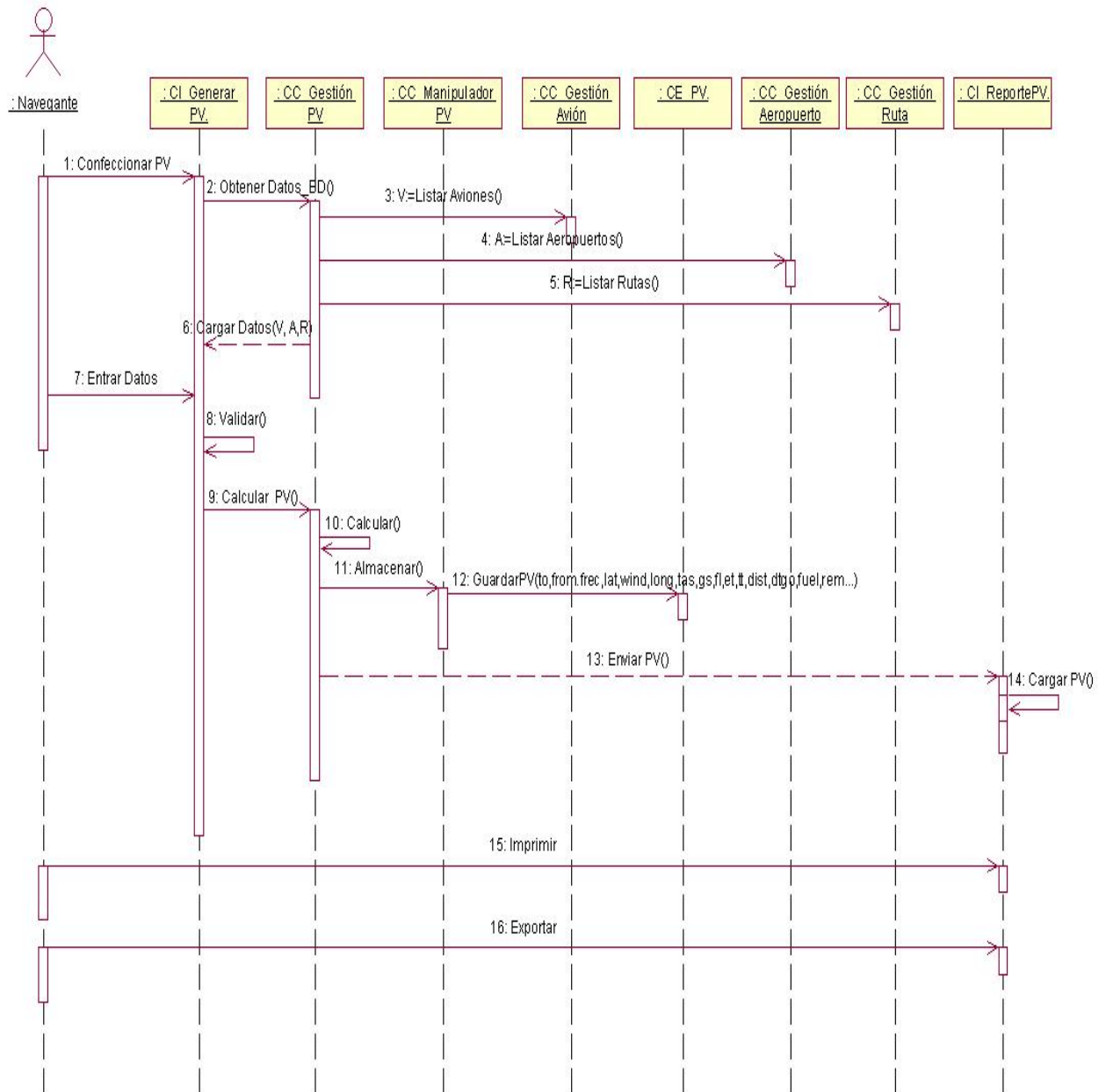


Imagen 13: Plan de Vuelo (Seleccionar Ruta).

Sistema para la Gestion del Plan de vuelo IL-96 300

Opciones

Fecha: /2007 11:34:13 PM Origen (ICAO): SAEZ Destino (ICAO): VARC
 Registro: CU6988 Origen (SITA): EZA Destino (SITA): VAR
 FLT: 23654 Alterno: CMGA ALTN DIST: 350
 ETD: 32456 Data Met: 6/6/7 Mach: 0.78
 DOW: 451233 ZPW: 156000 PLD: 342561

Limitantes
 Máx. FL: 43000
 TOW Rest: 198000
 LW Rest: 165000
 Extra Fuel: 2000
 Extra Fuel Max

Ruta: Viento-Temperatura

Mostrar Rutas

Rutas Fijas

Ruta
<input type="checkbox"/> SAEZ IBSEN UVA VARC
<input checked="" type="checkbox"/> SAEZ UZG VARC
<input type="checkbox"/> SAEZ DARLO UCL VARC

Rutas Provisionales

Ruta
<input type="checkbox"/> CMGA HOOKY VARC
<input type="checkbox"/> SAEZ LOUIZ MENDL VARC

Invertir Cancelar

Elaborado por: _____ Generar PLN Cancelar

Imagen 14: Plan de Vuelo (Viento-Temperatura).

Sistema para la Gestion del Plan de vuelo IL-96 300

Opciones

Fecha: /2007 11:34:13 PM Origen (ICAO): SAEZ Destino (ICAO): VARC
 Registro: CU6988 Origen (SITA): EZA Destino (SITA): VAR
 FLT: 23654 Alterno: CMGA ALTN DIST: 350
 ETD: 32456 Data Met: 6/6/7 Mach: 0.78
 DDW: 451233 ZPW: 156000 PLD: 342561

Limitantes
 Máx. FL: 43000
 TOW Rest: 198000
 LW Rest: 165000
 Extra Fuel: 2000
 Extra Fuel Max

Ruta: Viento-Temperatura

Ascenso (CLB)
 WC: 23
 OAT: 54

Descenso (DCT)
 WC: 45
 OAT: 23

Alterno (ALTN)
 WC: 44 Frente Cola
 OAT: 75

Crucero

	<input checked="" type="checkbox"/> 1_Tramo	<input checked="" type="checkbox"/> 2_Tramo	<input type="checkbox"/> 3_Tramo	<input type="checkbox"/> 4_Tramo
Pto. Inicial	UZG	UVA		
Pto. Final	UVA	DARLO		
WC	78	23		
OAT	36	85		

Aceptar Cancelar

Elaborado por: _____ Generar PLN Cancelar

Glosario de Términos:

1. SITA: Sociedad Internacional de Telecomunicaciones Aeronáuticas.
2. NACKTRACK (North Atlantic Track): son coordenadas geográficas unidas con un nombre específico, utilizadas para sobrevolar el mar en el espacio aéreo del Atlántico Norte y que no son las mismas diariamente.
3. NOTAMS: Acrónimo de **Notices To Airmen** (Avisos a los aviadores). Publicados en forma de boletines con el objetivo de informar a los pilotos los cambios existentes en cuanto a instalaciones aeronáuticas, servicios, procedimientos, o peligros, ya sean temporales o permanentes.
4. ECASA: Empresa Cubana de Aeropuertos y Servicios Aeronáuticos.
5. MZFW (Maximum Zero Fuel Weight): máximo peso total del avión sin combustible, compuesto por el DOW y el PLD.
6. PLD (Pay Load): Es el peso correspondiente a los pasajeros, el equipaje, la carga y el correo.
7. DOW (Dry Operating Weight): Es el peso correspondiente al PBO, comisaría, extracrew, contenedores de carga, y técnica extra: entendiéndose por esta última instrumentos, gomas de repuestos, etc.
8. PBO (Peso Básico de Operaciones): Es el peso de fábrica del avión, correspondiente al avión vacío.
9. TOF (Take off Fuel): Peso correspondiente al combustible de quemada¹¹ unido al de reserva.
10. TAXI: Combustible necesario para ubicarse en la cabecera de la pista.
11. Trip Fuel (Quemada): Peso del combustible que se emplea en la realización del vuelo, es decir el combustible quemado en la trayectoria.
12. MissAPP (Miss Approach): Combustible necesario para hacer holding y luego dirigirse al alterno.
13. Route Res: 3 por ciento del combustible de quemada.
14. ALTN: aeropuerto alterno.
15. HOLD: combustible requerido para hacer holding en el aeropuerto alterno por 30 min. en caso de alguna eventualidad.
16. Diff to min: valor que representa la diferencia entre el valor de MissApp y su valor mínimo (8000), este valor se le añade al MissApp cuando es menor que 8000.

17. MTOW (Maximum Take off weight): máximo peso real de despegue.
18. FL (Flight Level): Superficie de presión atmosférica constante relacionada con una determinada referencia de presión, 1 013.2 hectopascales y que está separada de otras superficies análogas, por determinados intervalos de presión.
19. Charter: Compañía aérea dedicada a la realización de vuelos a la demanda, sin calendario ni itinerario prefijados. Se denominan también "suplementales
20. Math: Número que representa la relación de la velocidad real y la velocidad del sonido. Ej: 0.78 representa el 78% de la velocidad del sonido.
21. CU (Caso de Uso del Negocio): es una secuencia de acciones, realizadas en el negocio, que producen un resultado de valor observable para ciertos actores del negocio.
22. CUS (Caso de Uso del Sistema): son artefactos narrativos que describen el comportamiento del sistema desde el punto de vista del usuario. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades (requisitos) que debe cumplir el sistema.
23. MBO: Acrónimo de Manual Básico de Operaciones.
24. UCI: Siglas que representan a la Universidad de las Ciencias Informáticas.
25. FAA - Federal Aviation Administration
26. IFP - Instrument Flight Procedure (Procedimiento del vuelo por instrumento)
27. IFR - Instrument Flight Rules (REGLAS DE VUELO POR INSTRUMENTOS)
28. NACO - National Aeronautical Charting Office (Oficina nacional de Cartas aeronáuticas)
29. NAVAIDS - Navigational Aids (Ayudas navegacionales) las ayudas a la navegación de tipo navaid (NDB, VOR³¹, ILS)
30. VFR - Visual Flight Rules (REGLAS DE VUELO VISUAL): Son reglas que garantizan que los vuelos VFR se realicen de forma que la aeronave vuele en condiciones de visibilidad y de distancia de las nubes que sean iguales o superiores a las indicadas.
31. VOR - Very high frequency Omni directional Range (Gama direccional Omni de alta frecuencia).
32. FBO: En la industria de la aviación este término es conocido como una base fija de operación, y no es más que un centro de servicio en un aeropuerto que pueda ser una empresa privada o un departamento del municipio al que el aeropuerto sirve.
33. WACs (World Aeronautical Charts): Cartas aeronáuticas mundiales

BIBLIOGRAFIA

1. BAKER, F. T. Structured programming in a production programming environment. ACM Press, 1975. 172-185 p.
2. CRUZ., I. O. A. and I. J. R. D. L. PEÑA. Sistema de administración de las capacidades. Ciencias Holguín, 2007.
3. El lenguaje C# y la plataforma .NET. en: PROGRAMACIÓN C#. 20.p.
4. GARLAN, M. S. Y. D. Software Architecture: Perspectives on an emerging discipline. Upper Saddle River, Prentice Hall. 1996. p.
5. IBM_RATIONAL_SOFTWARE. Rational Rose Enterprise, <http://www-306.ibm.com/software/awdtools/developer/rose/enterprise/>, 2007.
6. JACKSON, M. A. Principles of Program Design. Academic Press, 1975. p.
7. KICILLOF, C. R. N. Estilos y Patrones en la Estrategia de Arquitectura de Microsoft 2004.
9. LARMAN, C. UML y Patrones. Introducción al análisis y diseño orientado a objetos. España, 1998. 195 a 250 p. 0-13-748880-7
10. MARTÍNEZ, I. G. Un Modelo para la Toma de Decisiones usando Razonamiento Basado en Casos en condiciones de Incertidumbre. Departamento de Ciencias de la Computación, Facultad de Matemática-Física y Computación. Santa Clara, Universidad Central Marta Abreu de las Villas, 2003. 116. p.
11. MCGLAUGHLIN, R. Some Notes on Program Design. Software Engineering Notes, 1991. 53-54 p.
12. MICROSOFT. Diseño de software, <http://www.microsoft.com/spanish/MSDN/estudiantes/ingsoft/ingenieria/disenio.asp>., 2007a.
13. ---. Microsoft Solutions Framework, <http://www.microsoft.com/technet/solutionaccelerators/msf/default.aspx>, 2006.
14. ---. Visual Studio Team System <http://msdn2.microsoft.com/en-us/teamsystem/default.aspx>, 2007b.
15. MICROSOFT, C. Cambio de aplicaciones a .NET, 2002. [Disponible en: http://www.microsoft.com/spanish/msdn/articulos/archivo/130902/voices/dotnet_MovingJavaApps.asp

16. MOLPECERES, A. Procesos de desarrollo: RUP, XP y FDD, www.willydev.net/descargas/articulos/general/cualxpfdrrup.PDF, 2003.
17. Procesos de desarrollo: RUP, XP y FDD, 2002. 1: 14.
18. OTERO, A. S. Gestor de Referencias Bibliográficas. 2007. 55 p.
19. PALACIO, J. Gestión y procesos en empresas de software, www.navegapolis.net/files/articulos/gestion_y_procesos.pdf 2005.
20. PERRY, D. Software Architecture and its relevance for Software Engineering. Coord, 1997. p.
21. PRESSMAN, R. S. Ingeniería de software. Un enfoque practico. Mc Graw Hill, 1998. 614 p.
22. PROVENCIO, F. R. Y. D. .Net Framework, 200.
23. AeroPlanner, www.aeroplanner.com.
24. Flight Briefing Services, www.sita.com.