

Universidad de las Ciencias Informáticas



Facultad 2

**Sistema de Automatización para el Análisis de Pista del IL-96-300**

Trabajo de Diploma para optar por el título de Ingeniería en Ciencias Informáticas



**Autores:**

Laura Pereda Lamela

Omar Luis Garrido Jiménez

**Tutores:**

Lic. Darián Horacio Grass Boada

Ing. Luis Ramón González Paet

Ciudad de La Habana, Junio 2007

Año 49 de la Revolución

# *Agradecimientos*



El término de la enseñanza superior es uno de los momentos en que se transita de una etapa a otra en la vida. A diferencia de otros momentos similares, este tiene connotaciones especiales pues a partir de aquí se comienza a edificar el futuro profesional, una vida más independiente.

Este apartado, dentro de la tesis, tal vez sea el más complicado de escribir, pues es una sección que todo el mundo lee. Por eso, hoy quiero agradecer a todos los que han dejado una huella en mi corazón.

A mis padres por confiar en mí y esperar siempre lo mejor, por su paciencia y comprensión.

A mis dos hermanas que han actuado como mamá apoyándome en todo, por alentarme cada vez que pensaba que era el fin.

A mi tío Guille por estar disponible cada vez que sonaba el teléfono, por soportarme cada fin de semana.

A mis amigos Isel, Yuri y los muchachos del proyecto por dejarme entrar en sus vidas. En especial a Yadira y Cobo por sus consejos, por ser muchas veces mi paño de lágrimas y por la paciencia que han tenido conmigo. A Leansy por haberme servido de ayuda y apoyo durante mi carrera.

A toda la tripulación del IL-96 que nos han atendido, en especial a Raúl y Gómez por su total dedicación.

En fin, para no olvidar a nadie, a todos aquellos que me han permitido darme cuenta que la vida está llena de personas que te quieren, que te apoyan, que te toman de la mano y te regalan una sonrisa.

A ti, que aunque no estés presente en esta hoja sabes que te agradezco haberme regalado tantos momentos felices durante este último año de mi carrera...

Laura

# *Agradecimientos*



A mi madre porque sin ella nada sería posible.

A mi familia por apoyarme siempre.

A mi novia, Maidelis, por estar siempre a mi lado y apoyarme en todo lo que he necesitado.

A mis amigos, que me han apoyado en todas las esferas de la vida.

A Laura, que ha sido una magnífica compañera de Tesis.

A la familia de Maidelis que me han apoyado.

A todos los trabajadores de la Empresa Cubana de Aviación, que con paciencia nos han atendido y ayudado en todo, especialmente al ingeniero Raúl.

A Luis Ramón, Darián y Sheila, que han sido magníficos tutores, que nos han guiado y ayudado en la preparación de la tesis.

A todos los que no mencioné, pero saben que le estoy muy agradecido, gracias de todo corazón...

Omar

*Dedicatoria*



*A nuestros Padres...*

## *Resumen*



El principal objetivo en toda empresa de aviación es la seguridad de sus vuelos. El Análisis de Pista del IL-96 300 es un proceso de suma importancia dentro de este contexto. Actualmente, este flujo de información ocurre de manera manual, y son análisis complicados que deben ser llevados a cabo por los miembros de la tripulación, no estando exentos del error humano. Esto trae como consecuencia demoras, ineficiencia, gasto de recursos y vulnerabilidades en la seguridad.

La inexistencia de un sistema de soporte capaz de realizar dichos cálculos, de una forma óptima y libre de errores es lo que crea la necesidad de resolver la situación problemática que se plantea en este documento, en el cual se plasman los resultados del estudio realizado en la Empresa Cubana de Aviación.

El objetivo del presente trabajo es desarrollar una herramienta para la tripulación del IL-96 300 capaz de realizar los cálculos necesarios para crear los Análisis de Pista del IL-96 300 y de gestionar toda la información de los aeropuertos correspondientes a los Análisis de Pista.

El software soluciona los problemas que existían en el negocio, como la exactitud de los cálculos y la rapidez con la que se requerían si se toma en cuenta que muchas veces se realizaban bajo presión y en condiciones de estrés, posibilitando la inclusión de errores; por lo que la implementación e implantación del software es factible pues los beneficios tangibles e intangibles compensan el costo total del proyecto.



<b>Introducción</b> .....	1
<b>Capítulo 1: <i>Fundamentación Teórica</i></b> .....	4
1.1 Introducción.....	4
1.2 Estado del Arte.....	4
1.3 Tecnologías de software a conocer.....	5
1.3.1 Tecnología .Net. ....	5
1.3.2 Arquitectura Framework.NET ....	5
1.3.3 Lenguaje de programación.....	7
1.3.3 El proceso unificado de desarrollo de software RUP.....	10
1.3.4 SQL Server 2000.....	13
EndNote.....	14
1.4 Conclusiones.....	15
<b>Capítulo 2: <i>Características del Sistema</i></b> .....	16
2.1 Introducción.....	16
2.2 Objeto de Estudio.....	16
2.2.1 Problema y Situación Problémica.....	16
2.2.2 Objeto de Automatización.....	17
2.2.3 Información que se maneja. ....	18
2.3 Propuesta del sistema.....	18
2.4 Modelo del Negocio.....	18
2.4.1 Actores y Trabajadores del Negocio.....	20
2.4.2 Diagrama de Casos de Uso del Negocio.....	21
2.4.3 Especificación de los Casos de Uso. Ver Anexo 1.....	21
2.4.4 Diagramas de Actividades.....	21
2.4.6 Reglas generales del Negocio.....	22



2.5 Especificación de los Requisitos de software.....	22
2.6 Modelo de Casos de Uso del Sistema. ....	26
2.6.1 Definición de los actores.....	26
2.6.2 Diagrama de Casos de Uso del Sistema. ....	28
2.7 Descripción de los Casos de Uso.....	28
2.8 Conclusiones.....	29
<b>Capítulo 3: Análisis y Diseño del sistema. ....</b>	<b>30</b>
3.1 Introducción.....	30
3.2 Análisis.....	30
3.2.1 Diagramas de Clases del Análisis. ....	31
3.3 Diseño. ....	32
3.3.1 Diagramas de interacción. ....	32
3.3.2 Diagrama de Clases. ....	33
3.4 Diseño de la Base de Datos.....	36
3.4.1 Modelo Lógico de Datos.....	36
3.4.2 Modelo Físico de Datos.....	37
3.5 Conclusiones.....	38
<b>Capítulo 4: Implementación. ....</b>	<b>39</b>
4.1 Introducción.....	39
4.2 Diagrama de despliegue. ....	39
4.3 Diagrama de componentes. ....	40
4.4 Conclusiones.....	42
<b>Capítulo 5: Estudio de Factibilidad. ....</b>	<b>43</b>
5.1 Introducción.....	43
5.2 Planificación basada en Casos de Uso.....	43

# Índice



5.3 Beneficios tangibles e intangibles. ....	49
5.4 Análisis de costos y beneficios.....	49
5.5 Conclusiones.....	50
<b>Conclusiones. ....</b>	<b>51</b>
<b>Recomendaciones. ....</b>	<b>52</b>
<b>Bibliografía. ....</b>	<b>53</b>
<b>Anexos. ....</b>	<b>54</b>
<b>Glosario de Términos. ....</b>	<b>72</b>





## **Introducción.**

IL-96 300 es un avión de fabricación rusa comprado por el gobierno cubano para la empresa de Cubana de Aviación destinado al transporte de pasajeros, cargas y correo a largas distancias, de hasta 11 000 kilómetros. La producción en serie de dicho avión comenzó en 1992, de los cuales Cuba ha obtenido 3 hasta el momento.

Soporta un peso máximo de despegue y aterrizaje de 250 y 175 toneladas respectivamente, una velocidad de crucero de 850 Km./h a 900 Km./h y puede transportar hasta 262 pasajeros. La velocidad máxima permitida por las ruedas es de 178 nudos y el ángulo de inclinación de los Flaps/Slats en el despegue es de 25/25 grados y en el aterrizaje de 40/25 grados respectivamente.

Cuenta con una tripulación que está compuesta por el Capitán, Piloto, Copiloto, Ingeniero de Vuelos y el Despachador.

Casi todos los procesos que tienen lugar antes de la salida o llegada de un vuelo se hacen actualmente en nuestros aeropuertos de forma manual haciendo engorroso todo este trabajo que tiene como premisa fundamental la rapidez y seguridad. Si además se tiene en cuenta que los cálculos en ocasiones se realizan bajo condiciones extremas, aumenta la posibilidad real de introducir errores humanos y poner en peligro la vida de los pasajeros y de la tripulación. Todos estos aspectos motivaron la necesidad de crear un sistema que diera solución a estas dificultades.

El sistema debe ser capaz de realizar el Análisis de Pista tanto Completo como Operativo, mostrando para un rango de temperaturas seleccionado y todos los vientos el peso máximo de despegue y aterrizaje que permiten las condiciones ambientales imperantes, las características de la pista y el performance del avión. Además se mostrarán las velocidades para las distintas maniobras que realiza el avión en el despegue y para tomar altura. Debe permitir también la gestión de los aeropuertos con sus pistas, obstáculos y sensores. Con este sistema se elimina la posibilidad de introducción de errores humanos y se espera reducir el tiempo de realización de los cálculos.

# *Introducción*



Para este avión en específico solamente se tiene como fuente de información el Manual Básico de Operaciones y los gráficos del performance propios del Análisis de Pista.

El **objeto de estudio** para desarrollar el software satisfactoriamente son los procesos necesarios para realizar los vuelos en un IL-96 300 y nuestro **campo de acción** específicamente serán los procesos ejecutados en la elaboración de los Análisis de Pista para IL-96 300 que realiza la tripulación.

El **objetivo** de esta investigación es desarrollar un sistema para la tripulación del IL-96 300 que realice los cálculos para crear los Análisis de Pista del IL-96 300 y gestionar la información de los aeropuertos correspondientes a los Análisis de Pista.

Para dar cumplimiento al objetivo propuesto se trazaron las siguientes **tareas**:

- ❖ Desarrollar una investigación del estado del arte para la creación de los Análisis de Pista para el IL-96 300.
- ❖ Realizar un estudio de todos los cálculos y conceptos de la aviación, específicamente para la elaboración de los Análisis de Pista.
- ❖ Entrevistar al cliente para detallar documentación y un mejor entendimiento del negocio.
- ❖ Observar cómo se realiza el proceso de Análisis de Pista para refinar los resultados de la entrevista.
- ❖ Declarar los requisitos que debe cumplir el sistema.
- ❖ Analizar, diseñar, validar e implantar el software que se propone en esta investigación.

El trabajo ha sido organizado en 5 capítulos que abordarán todo lo referente al estudio y realización del sistema quedando estructurado de la siguiente manera:

**Capítulo 1:** Describe el estado del arte en la universidad, nacional e internacional para el Análisis de Pista del IL-96 300. Trata las distintas tecnologías, metodologías y lenguaje de programación a utilizar en el desarrollo de la aplicación, y se explican sus principales características.

# *Introducción*



**Capítulo 2:** Describe el flujo de los procesos involucrados en el campo de acción y aquellos que son objeto de automatización, además de la información que se maneja. Se hace referencia al flujo de trabajo Modelación del Negocio y se dan a conocer las características del sistema en general. Se describen los actores del negocio y del sistema y los trabajadores del negocio, así como los casos de usos encontrados. Se muestran los diagramas de casos de usos del negocio y del sistema, los diagramas de actividades y los modelos de objetos correspondientes a cada caso de uso. Se definen las funcionalidades y los requisitos no funcionales del sistema.

**Capítulo 3:** Describe el análisis y el diseño del sistema definiendo el diagrama de clases del análisis y el diagrama de clases del diseño con los diagramas de interacción correspondientes a cada caso de uso y escenario. Además se incorporan el diagrama de clases persistentes y el modelo de datos.

**Capítulo 4:** Se hace referencia al flujo de trabajo implementación, donde se muestran los diagramas de despliegue y componente.

**Capítulo 5:** Muestra un estudio de factibilidad para la construcción del sistema, observando los beneficios tangibles e intangibles y analizando los costos y beneficios del desarrollo de la propuesta.



## **Capítulo 1: Fundamentación Teórica.**

### **1.1 Introducción.**

En este capítulo se realizará un estudio sobre el estado del arte a nivel internacional, nacional y de la Universidad del sistema a construir, realizando una valoración crítica de los procedimientos utilizados actualmente para crear el Análisis de Pista para el avión IL- 96 300 sin dejar de abordar los principales conceptos y términos asociados al tema.

Se describen además las metodologías de desarrollo, lenguajes, herramientas y otros softwares utilizados en la solución e implementación del software, así como sus principales características por las que fueron seleccionados para la confección de esta aplicación.

### **1.2 Estado del Arte.**

En la actualidad no existe ningún software que realice el Análisis de Pista del IL-96 300 a nivel nacional y es la primera vez que se realizará en la Universidad de las Ciencias Informáticas un proyecto con la Empresa Cubana de Aviación, por lo que no existe la experiencia ni el conocimiento de las tendencias, técnicas, herramientas, y metodologías de desarrollo ideales para este tipo de sistema.

Se conoce mediante los tripulantes que en Rusia, lugar donde fueron construidos estos aviones, se creó una calculadora de mano que permite crear los Análisis de Pista para los aviones IL-96 300 en ese país, pero dada la confidencialidad del asunto, no se pudo hacer un estudio sobre el funcionamiento de esa herramienta.



## 1.3 Tecnologías de software a conocer.

### 1.3.1 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.(WIKIPEDIA 2006a), (MICROSOFT 2007b)

### 1.3.2 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.

## *Capítulo I: Fundamentación Teórica*



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.(ROBERT MONROE), (REYNOSO Junio de 2004), (KICILLOF 2004), (PROVENCIO 2003), (MICROSOFT, CORPORATION 2002)

### **Algunas 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 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. (GARLAN 1996; PERRY 1997)

### **1.3.3 Lenguaje de programación.**

#### **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. (U. 2004), (El lenguaje C# y la plataforma .NET), (SHAW 1984)

#### **Principales características que identifican al lenguaje C#:**

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

## *Capítulo I: Fundamentación Teórica*



**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.

### **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 pues 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.(ZAPATA 2004)



# *Capítulo I: Fundamentación Teórica*



## **Arquitecturas de tres capas.**

El principio que está detrás de las arquitecturas de multicapas (o de tres capas 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.

### **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.

# Capítulo I: Fundamentación Teórica



## 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.



Figura 1 Aplicaciones de tres capas.

## 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. (MCGLAUGHLIN 1991), (W. STEVENS 1974), (BAKER 1975), (MICROSOFT 2006).

### 1.3.3 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 la modelación de sistemas con tecnología orientada a objetos.



Figura 2. Un proceso de desarrollo de software.

## 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, RUP<sup>1</sup> 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, dígase 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

---

<sup>1</sup> Rational Unified Process

## *Capítulo I: Fundamentación Teórica*



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-proyectos 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, sacar al mercado un producto en el tiempo previsto, mantener la motivación del equipo pues pueden verse avances claros a corto plazo y que el desarrollo pueda adaptarse a los cambios en los requisitos.

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]:

- ❖ Dirigido por casos de uso. El desarrollo está dirigido a satisfacer las necesidades de los usuarios del sistema expresadas en casos de uso.
- ❖ Centrado en la arquitectura. El desarrollo se centra en una arquitectura bien definida, con relaciones claras entre sus distintos componentes.
- ❖ 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.
- ❖ Incremental. Cada iteración se construye sobre la base creada por las iteraciones anteriores, agregándole capacidades al sistema.
- ❖ Controlado. El proceso se planifica y en cada momento está claro lo que debe hacerse.

## *Capítulo I: Fundamentación Teórica*



Se hizo uso de las herramientas de la metodología RUP (*Rational Unified Process*) para facilitar el desarrollo del sistema. (MARTÍNEZ 2003), (WIKIPEDIA 2006b), (MOLPECERES 2003), (MOLPECERES 2002), (JACKSON 1975), (PRESSMAN 1998), (PALACIO 2005), (TEJADA 2002), (MICROSOFT 2007a), (LARMAN 1998),

### **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 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. (IBM\_RATIONAL\_SOFTWARE 2007; WIKIPEDIA 2006b), (CRUZ. and PEÑA. 2007)

### **1.3.4 SQL Server 2000.**

Debido a la naturaleza de los datos que se almacenarán en el sistema propuesto, se ha decidido utilizar Microsoft SQL Server 2000 como gestor de base de datos, por ser una aplicación poderosa, robusta, que permite gran seguridad, y ostenta marcas de referencia en cuanto a escalabilidad y confiabilidad, que son críticas para el éxito de bases de datos de gran tamaño. El SQL Server permite lograr una gran velocidad en el procesamiento de transacciones y agilidad en todas sus operaciones, además, es el utilizado en la Empresa Cubana de Aviación.

Es un potente motor de bases de datos de alto rendimiento capaz de soportar millones de registros por tabla con herramientas de desarrollo integradas como .NET, además incorpora un modelo de objetos totalmente programable (SQL-DMO) con el que podemos desarrollar cualquier aplicación que manipule

## *Capítulo I: Fundamentación Teórica*



componentes de SQL Server, es decir, hacer aplicación para crear bases de datos, tablas, DTS, backups, etc., todo lo que se puede hacer desde el administrador del SQL Server y podemos hacerlo no solo en Visual C++ sino también en Visual Basic, ASP y por supuesto en .NET.

El cliente determinó que debía usarse esta herramienta porque la base de datos del software a construir deberá ser compatible con la que ellos poseen en las oficinas de Cubana de Aviación. (LARMAN 1998), (SOLANO 2002)

### **EndNote.**

El EndNote es una herramienta de búsqueda online que proporciona la posibilidad de referenciar los documentos de una manera sencilla y correcta, el programa puede además importar ficheros de datos guardados desde diversos servicios online, CD-ROMs y bases de datos de bibliotecas.

Es una base de datos para referencias e imágenes cuya función principal es almacenar, gestionar y buscar referencias bibliográficas en una librería personal de referencias. Permite además organizar imágenes- incluyendo gráficos, tablas, figuras y ecuaciones asignando a cada imagen su propia leyenda y palabras clave.

EndNote elabora bibliografías y manuscritos, dando formato a citas, figuras y tablas en Word. Al ir introduciendo citas en el manuscrito, la lista de referencias, figuras y tablas se va actualizando. También es posible crear bibliografías empleando documentos RTF con otros procesadores de texto.

Dentro de la aplicación ya están predefinidos diferentes estilos de citas bibliográficas, para referenciar documentos, de forma tal que si se tiene que cambiar el formato de la referencia se puede realizar de forma automática.

Permite eliminar citas de forma automática, solamente eliminando el párrafo, oración o palabra que esté referenciado es eliminada la referencia.

## *Capítulo I: Fundamentación Teórica*



Muestra en una lista multi\_columna las referencias bibliográficas almacenadas de manera tal que se pueden revisar y observar antes de ponerlas en el documento.(OTERO 2007)

### **1.4 Conclusiones.**

En este capítulo se realizó un estudio sobre el estado del arte del sistema y las herramientas, metodologías, lenguajes y software que son utilizados en la oficina de Informática de Cubana de Aviación. Se exponen las características fundamentales de las herramientas, metodologías, lenguajes y otros softwares que han sido utilizados para el desarrollo del proyecto.



## **Capítulo 2: Características del Sistema.**

### **2.1 Introducción.**

En este capítulo se abordarán temas de suma importancia para entender como ocurren cada uno de los pasos que intervienen actualmente en el negocio. Se describirán además aquellos procesos que serán objeto de automatización y una comparación de otras soluciones existentes con la propuesta que aquí se expone. Se detallarán además cada información que se manipule en la solución.

Se hará referencia al flujo de trabajo Modelamiento del Negocio, describiendo las mejoras que se proponen para dar solución a la situación problemática. Se referirán los actores y trabajadores del negocio, el diagrama de casos de uso, los de actividades y el modelo de objeto, además de las descripciones textuales correspondientes a cada caso de uso. Se enumerarán las funcionalidades y los requisitos no funcionales del sistema.

Se hará un modelamiento del sistema definiendo sus actores, el diagrama de casos de uso y la descripción textual de los mismos.

### **2.2 Objeto de Estudio.**

#### **2.2.1 Problema y Situación Problemática.**

Uno de los principales objetivos que tiene la Empresa Cubana de Aviación es mantener la seguridad en sus vuelos, garantizando de esta forma oportunidades comerciales. Para ello realiza minuciosamente cada proceso que interviene en los cálculos del Análisis de Pista.

Cuando se planifica la salida de un vuelo la tripulación solicita al encargado dentro de la tripulación de esta actividad un Análisis de Pista Completo u Operativo. En ese análisis se especifica el peso que puede elevar el avión respecto a las características propias de la pista y las condiciones ambientales imperantes



## Capítulo II: Características del sistema



a la hora del vuelo. También se especifican las velocidades de cada una de las maniobras fundamentales que se deberán realizar en el despegue y en la toma de altura del avión.

El encargado en la tripulación de realizar estos Análisis de Pista analiza cada uno de los gráficos que brindan los datos necesarios para su confección.

Este proceso se realiza actualmente de forma manual y convergen una serie de elementos que los tornan sumamente engorrosos, posibilitando la inclusión de errores, debido fundamentalmente al nivel de concentración y análisis que llevan, máxime teniendo en cuenta la presión y rapidez con que se deben efectuar los mismos en momentos cruciales, muchas veces bajo condiciones extremas.

En consecuencia estos cálculos no son todo lo exacto que se quisieran pues dependen de la aproximación que dé la persona que lo esté realizando en ese momento. Esto no se hace de forma intencional, hay que tener en cuenta la cantidad de gráficos que hay que analizar y el número de factores que influyen en cada uno. La exactitud de estos cálculos garantiza como prioridad la seguridad del personal y de la carga, así como oportunidades comerciales.

### 2.2.2 Objeto de Automatización.

Serán objeto de automatización los procesos que intervienen en el cálculo de los Análisis de Pista, es decir todos los análisis de los gráficos que intervienen en el despegue para la obtención del peso máximo de despegue, y las distintas velocidades que lo componen.

Para lograr una mayor exactitud en los datos necesarios para la confección del Análisis de Pista se utiliza la siguiente fórmula de interpolación lineal, que permite obtener datos que se encuentran dentro de un rango para el cual si se tiene información.

$$L(x) = \frac{x - x_1}{x_0 - x_1} y_0 + \frac{x - x_0}{x_1 - x_0} y_1$$

## *Capítulo II: Características del sistema*



En Cubana de Aviación no existe ningún otro sistema automatizado para realizar los procesos del Análisis de Pista para el IL-96 300 con el cual establecer comparación.

### **2.2.3 Información que se maneja.**

Para la realización de este sistema se ha utilizado la información contenida en el Manual Básico de Operaciones y gráficos específicos del avión correspondiente a la confección de los Análisis de Pista. Toda esta información es confidencial, por lo que los detalles de la misma no se pueden reflejar en este documento.

### **2.3 Propuesta del sistema.**

Para acceder al sistema propuesto el usuario deberá autenticarse y de acuerdo a los permisos y privilegios que tenga accederá al módulo permitido. Si es el encargado de la tripulación de realizar los Análisis de Pista (Tripulación/AP) una vez dentro de la interfaz de Análisis de Pista tendrá la posibilidad de crear un nuevo aeropuerto con las características de todas sus pistas, además de los obstáculos y sensores de ruido que existan para cada una de ellas. También podrá gestionar toda esa información, entiéndase por gestionar adicionar, modificar o eliminar un aeropuerto, una pista, un obstáculo o un sensor de una pista en específico.

Podrá crear también un Análisis de Pista en el formato seleccionado (Completo u Operativo).

Si es otro miembro de la tripulación solamente podrá buscar Análisis de Pista que estén previamente elaborados y guardados.

Todas estas operaciones deberán ser realizadas en el menor tiempo posible y con la mayor exactitud para lograr datos confiables.

### **2.4 Modelo del Negocio.**

El primer paso en el proceso de desarrollo de software es alcanzar cierto nivel de conocimientos del problema a resolver. El Modelo del Negocio entra en la fase de Inicio, y es precisamente aquí donde se realiza un análisis y se determina si se continúa con el proyecto.

## *Capítulo II: Características del sistema*



Un sistema por pequeño que sea siempre es complicado y el modelamiento del negocio es la técnica más importante para la especificación de requisitos. Aquí resulta útil la creación de modelos que permitan abstraer las características esenciales de problemas reales que se vinculan con el sistema informático a construir. Para poder realizar el modelamiento del negocio es necesario tener bien definidos todos los procesos que intervienen en el mismo.

Los objetivos del modelo de negocio son: comprender la estructura y la dinámica de la organización en la cual se va a implantar el sistema, comprender los problemas actuales de la organización e identificar las mejoras potenciales. Asegurar que los consumidores, usuarios finales y desarrolladores tengan un entendimiento común de la organización y derivar los requerimientos del sistema que va a soportar la organización.

### **Breve Descripción del Negocio.**

El negocio actualmente ocurre de la siguiente manera: Cuando se planifica la salida de un vuelo la tripulación solicita un Análisis de Pista en cualquier formato Completo u Operativo al encargado dentro de la tripulación de realizar dichos análisis (Tripulación/AP). En ese análisis se especifica el peso que puede elevar el avión teniendo en cuenta las características propias de la pista y las condiciones ambientales imperantes a la hora del vuelo. También se especifican cada una de las velocidades que se necesitan alcanzar ante cualquier maniobra durante el despegue y la toma de altura para no perder la sustentación del avión.

El encargado en la tripulación de realizar estos Análisis de Pista, Tripulación/AP, analiza los siguientes gráficos que brindan los datos necesarios para su confección:

- ❖ Se analiza el gráfico 7.2-1 (Curva Watt).
- ❖ Se analiza el gráfico 7.4-9 (Tires Speed).
- ❖ Se analiza el gráfico 7.4-10 (Peso de Referencia por obstáculos).
- ❖ Se analiza el gráfico 7.4-11 (Peso máximo por obstáculo).
- ❖ Se analiza el gráfico 7.4-4 (TODA corregido).

## Capítulo II: Características del sistema



- ❖ Se analiza el gráfico 7.4-5<sup>a</sup>. (ASDA corregido).
- ❖ Se analiza el gráfico 7.4-5 (ASDA corregido nuevamente).
- ❖ Se analiza el gráfico 7.4-6 (Distancia Balanceada y relación V1/Vr).
- ❖ Se analiza el gráfico 7.4.7 (Peso máximo de despegue).
- ❖ Se analiza el gráfico 7.4-8 (V1).
- ❖ Se analiza el gráfico 7.4-1 (Vr y V2).

Existe una gran probabilidad de inclusión de errores en los resultados obtenidos pues todos estos procesos se ejecutan manualmente y se realizan varias aproximaciones durante su elaboración. También hay que tomar en cuenta la cantidad de factores que influyen en cada uno de los gráficos anteriores.

Con la creación de un sistema capaz de realizar todos los cálculos necesarios para el Análisis de Pista se eliminarán los posibles errores que se pudieran cometer además de lograr los resultados en menos tiempo, en fin se eliminarían los problemas del negocio actual.

### 2.4.1 Actores y Trabajadores del Negocio.

- **Actores del Negocio.**

Actor	Descripción
Tripulación	La tripulación es la responsable de estudiar y cumplir el análisis de pista entregado por el encargado de la tripulación de realizar dichos análisis.

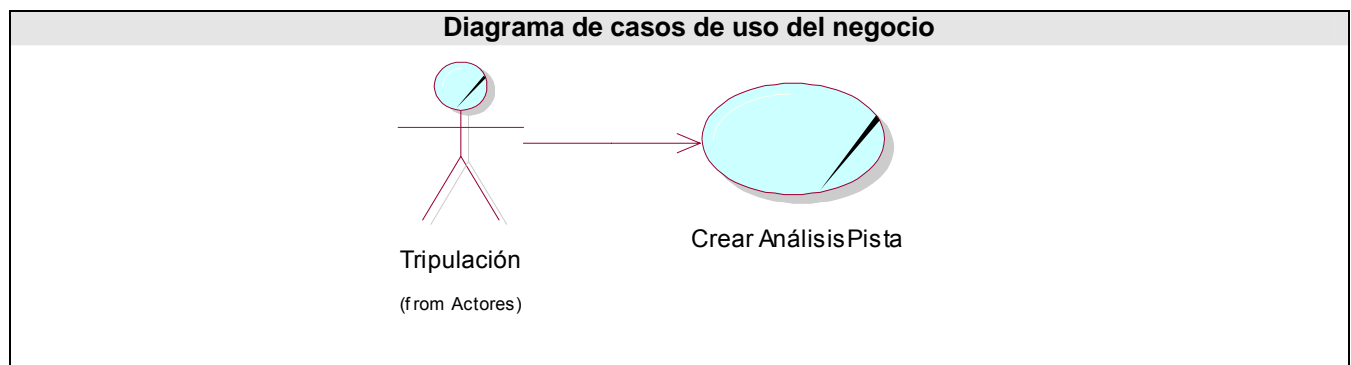
## Capítulo II: Características del sistema



- **Trabajadores del Negocio.**

Trabajador	Descripción
Tripulación/AP	Encargado de realizar los análisis de pista correspondientes tanto para el despegue como para el aterrizaje de cada aeropuerto que se necesite.

### 2.4.2 Diagrama de Casos de Uso del Negocio.



### 2.4.3 Especificación de los Casos de Uso. [Ver Anexo 1.](#)

### 2.4.4 Diagramas de Actividades. [Ver Anexo 2.](#)

### 2.4.5 Modelo de Objetos. [Ver Anexo 3.](#)



### **2.4.6 Reglas generales del Negocio.**

- ❖ La información solicitada podrá ser entregada en dos formatos diferentes: de Análisis (Completo) u Operativo.
- ❖ La V1 mínima debe ser de 130 nudos.
- ❖ Si el TODA sobrepasa 210 metros del TORA se toma el más pequeño.
- ❖ Sobrepasar zonas urbanas a 300 metros donde haya sensores. Analizar los decibeles de ruido y la distancia a la que se encuentran de la cabecera de la pista.
- ❖ El MLW es de 175 toneladas.
- ❖ El MTOW de 250 toneladas.
- ❖ La velocidad máxima permitida por las ruedas es de 178 nudos.
- ❖ El ángulo de inclinación de los Flaps/Slats en el despegue es de 25/25 grados y en el aterrizaje de 40/25 grados respectivamente.

### **2.5 Especificación de los Requisitos de software.**

#### **Requisitos Funcionales:**

1. Realizar Análisis de Pista Completo.
  - 1.1 Guardar Análisis de Pista Completo.
  - 1.2 Imprimir Análisis de Pista Completo.
2. Realizar Análisis de Pista Operativo.
  - 2.1 Guardar Análisis de Pista Operativo.
  - 2.2 Imprimir Análisis de Pista Operativo.
3. Gestionar Aeropuerto.
  - 3.1 Adicionar Datos.
  - 3.2 Modificar Datos.
  - 3.3 Eliminar Datos.

## *Capítulo II: Características del sistema*



### 4. Buscar Análisis de Pista.

#### **Requisitos 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 presentar 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 la Interfaz de Usuario para nombrar los menús, botones y las cajas de diálogo 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.

## *Capítulo II: Características del sistema*



### **Soporte:**

- ❖ Debe ser de fácil instalación y mantenimiento. Debe permitir 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 y 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:



## *Capítulo II: Características del sistema*



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

### **Interfaz:**

#### **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 de 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; donde quedarán 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 la construcción de un Manual de Usuario que describa pormenorizadamente sus características y uso.



## **2.6 Modelo de Casos de Uso del Sistema.**

### **2.6.1 Definición de los actores.**

<b>Actor</b>	<b>Descripción</b>
Tripulación.	Para acceder al sistema deben estar previamente autenticados. Reciben los Análisis de Pista para realizar sus vuelos. Pueden realizar búsquedas de Análisis de Pista que se hayan realizado con anterioridad.
Tripulación/AP	Son los responsable de crear nuevos aeropuertos así como de actualizar los datos que sufran modificaciones, como pistas, obstáculos, sensores, etc. Son los que introducen los datos al sistema para obtener los Análisis de Pista.

### **Listado de Casos de Uso.**

<b>CU - 1</b>	Crear AnálisisPista.
<b>Actor</b>	Tripulación/AP
<b>Descripción</b>	El encargado de la tripulación de elaborar los Análisis de Pista realiza los Análisis de Pista Completo u Operativo cuando la tripulación va a realizar un vuelo y lo solicita en un formato específico.
<b>Referencia</b>	R1, R1.1, R 1.2, R2, R2.1, R2.2

## Capítulo II: Características del sistema

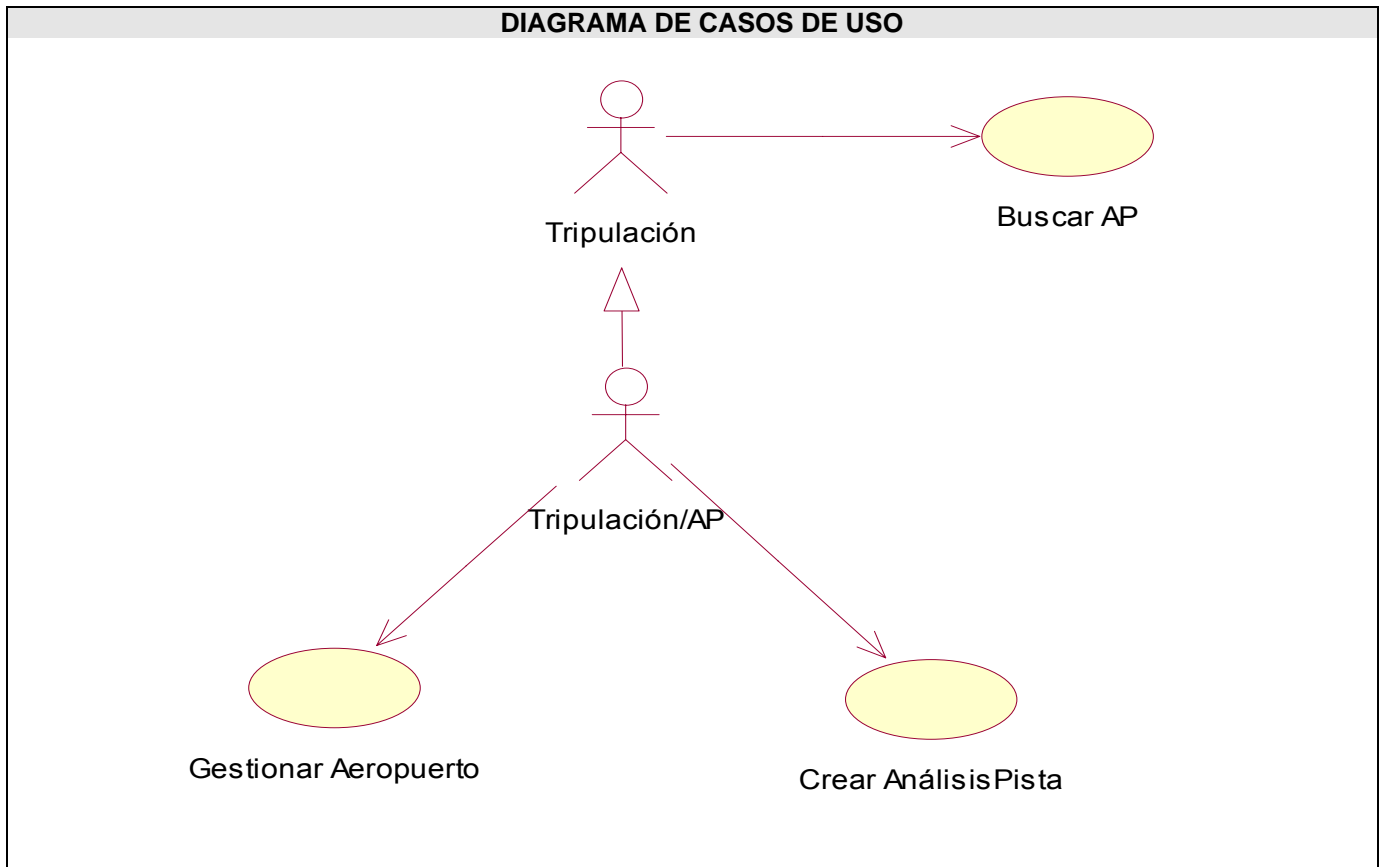


<b>CU - 2</b>	Gestionar Aeropuerto.
<b>Actor</b>	Tripulación/AP
<b>Descripción</b>	Pueden incluirse nuevos aeropuertos de destino para la tripulación del IL-96 300 por lo que será necesario incluir los datos específicos del aeropuerto, pero, los aeropuertos sufren modificaciones, pues algunas pistas pueden quedar deshabilitadas temporalmente o se crean nuevas pistas, así como pueden surgir nuevos obstáculos en el entorno, por lo que los datos del aeropuerto deben ser modificados para obtener un Análisis de Pista acorde a los datos reales y poder realizar vuelos seguros.
<b>Referencia</b>	R3, R3.1, R3.2, R3.3

<b>CU - 3</b>	Buscar AP.
<b>Actor</b>	Tripulación
<b>Descripción</b>	Cualquier miembro de la Tripulación, previamente autenticado, puede buscar Análisis Pista que estén previamente elaborados y guardados.
<b>Referencia</b>	R4



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



### 2.7 Descripción de los Casos de Uso. [Ver Anexo 4.](#)

## *Capítulo II: Características del sistema*



### **2.8 Conclusiones.**

En este capítulo se abordaron temas de suma importancia para lograr un mayor entendimiento del negocio actual pues se explicaron detalladamente el flujo de procesos que lo componen, además se abordó todo lo referente a la propuesta del sistema a construir y se enumeraron los requerimientos del sistema así como las funcionalidades que debe brindar.



## Capítulo 3: Análisis y Diseño del sistema.

### 3.1 Introducción.

El flujo de trabajo de Análisis y Diseño tiene un papel protagónico en la fase de Elaboración.

El objetivo del análisis es comprender perfectamente los requisitos del software y no precisar cómo se implementará la solución por esto no se toman en cuenta el lenguaje de programación a usar en la construcción, la plataforma en la que se ejecutará la aplicación, los componentes prefabricados o reusables de otras aplicaciones, entre otras características que afectan al sistema. El análisis consiste en obtener una visión del sistema que sólo se interesa por los requisitos funcionales.

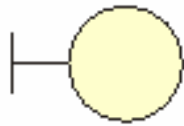
Por otro lado el diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, en definitiva cómo cumple el sistema sus objetivos. El grueso del esfuerzo del diseño está en las últimas iteraciones de la fase de elaboración y las primeras de construcción. El diseño debe ser suficiente para que el sistema pueda ser implementado sin ambigüedades.

### 3.2 Análisis.

En el Análisis los artefactos a utilizar serán el modelo de análisis y las clases de análisis que el RUP propone clasificar en:



Entidad: Modelan información que posee larga vida y que es a menudo persistente.



nombre\_interfaz

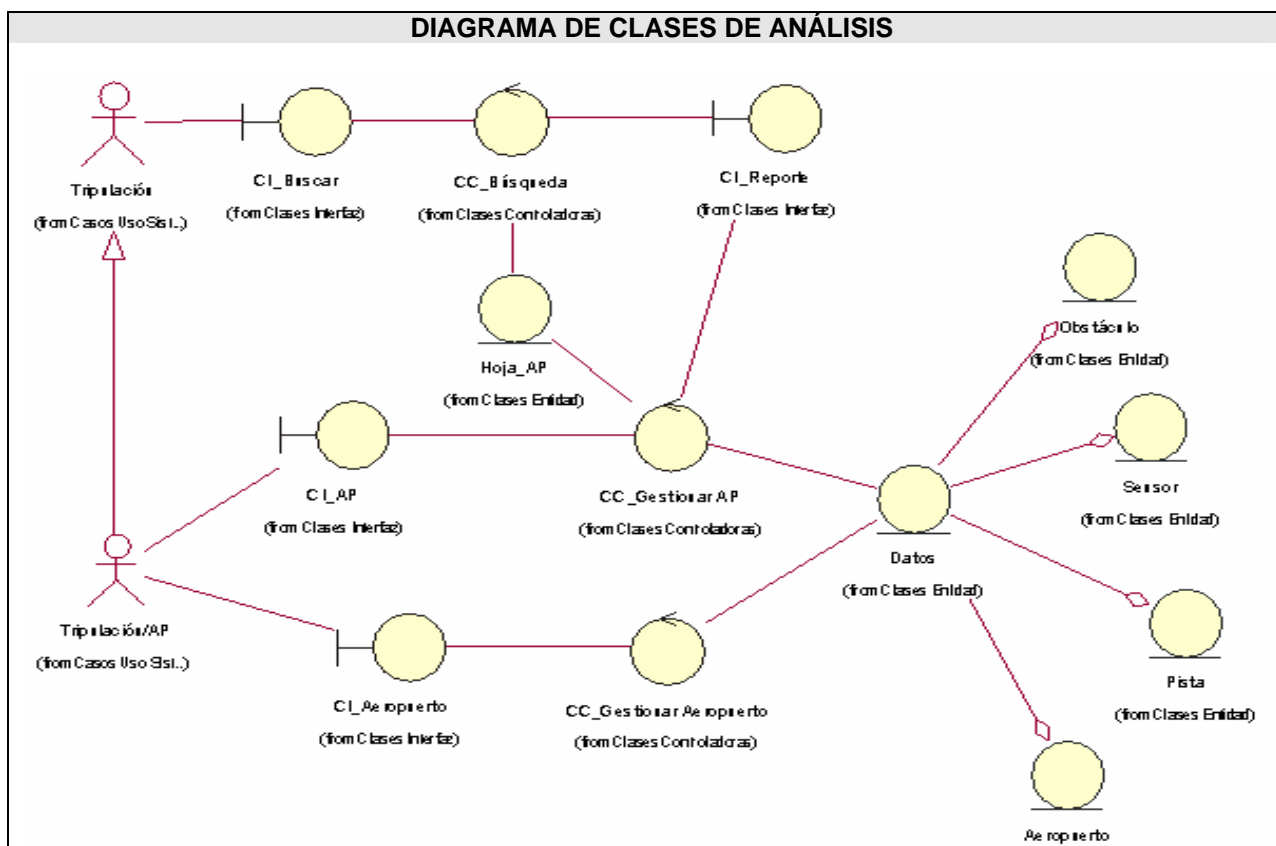
Interfaz: Modelan la interacción entre el sistema y sus actores.



nombre\_control

Control: Coordinan la realización de uno o unos pocos casos de uso así como las actividades de los objetos que implementan la funcionalidad del caso de uso.

### 3.2.1 Diagramas de Clases del Análisis.





### 3.3 Diseño.

En el Diseño los artefactos que aquí se expondrán son los diagramas de clases del diseño y los diagramas de interacción, específicamente los de secuencia.

#### Artefacto: Diagrama de clases

Los diagramas de clases son los más utilizados en el modelado de sistemas orientados a objetos. Un diagrama de clases muestra un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema.

#### Artefacto: Diagramas de interacción

Los diagramas de secuencia y los diagramas de colaboración (ambos llamados diagramas de interacción) son dos de los cinco tipos de diagramas de UML que se utilizan para modelar los aspectos dinámicos de los sistemas. Un diagrama de interacción muestra una interacción, que consiste en un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos.

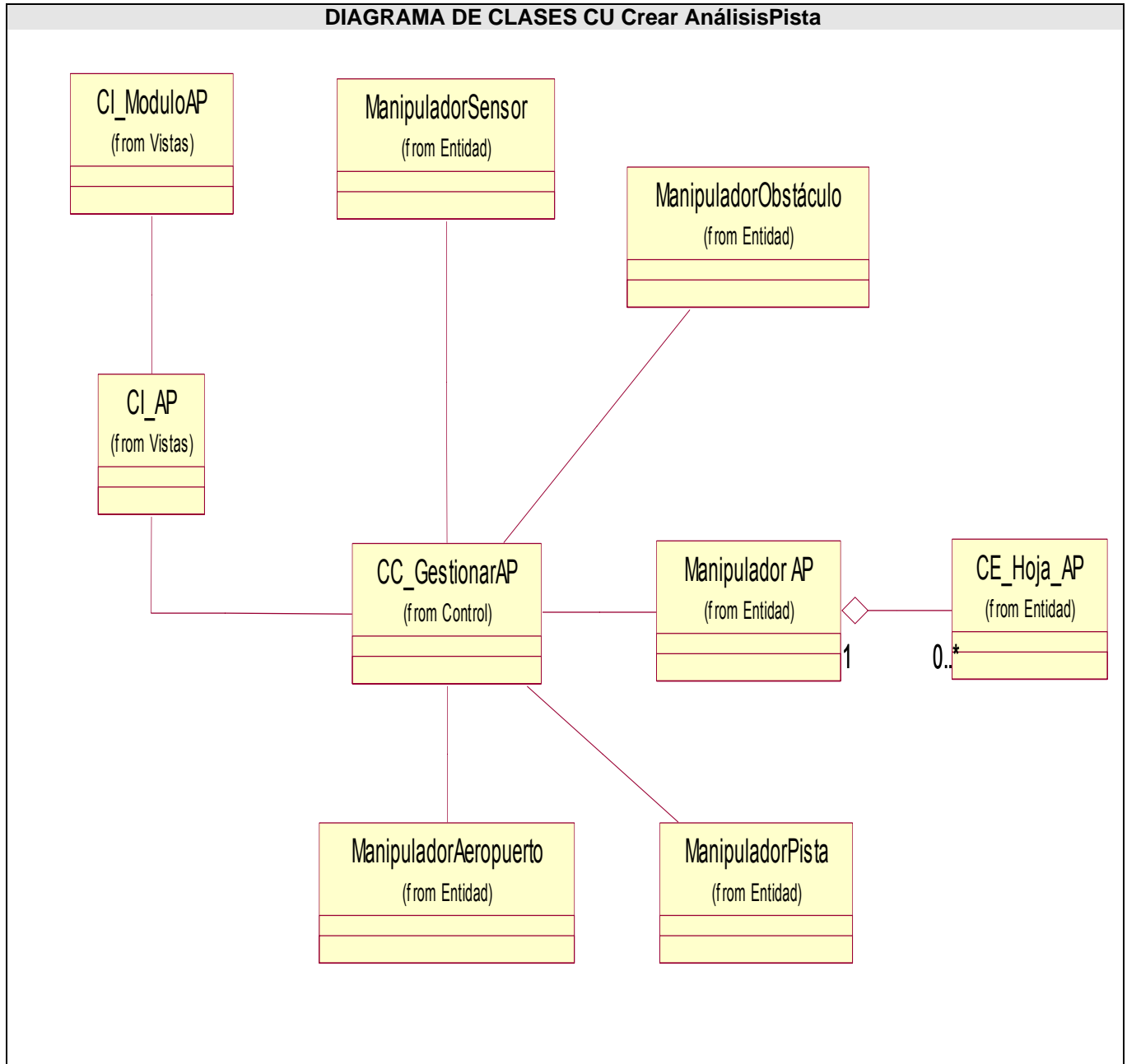
Un diagrama de secuencia es un diagrama de interacción que destaca la ordenación temporal de los Mensajes.

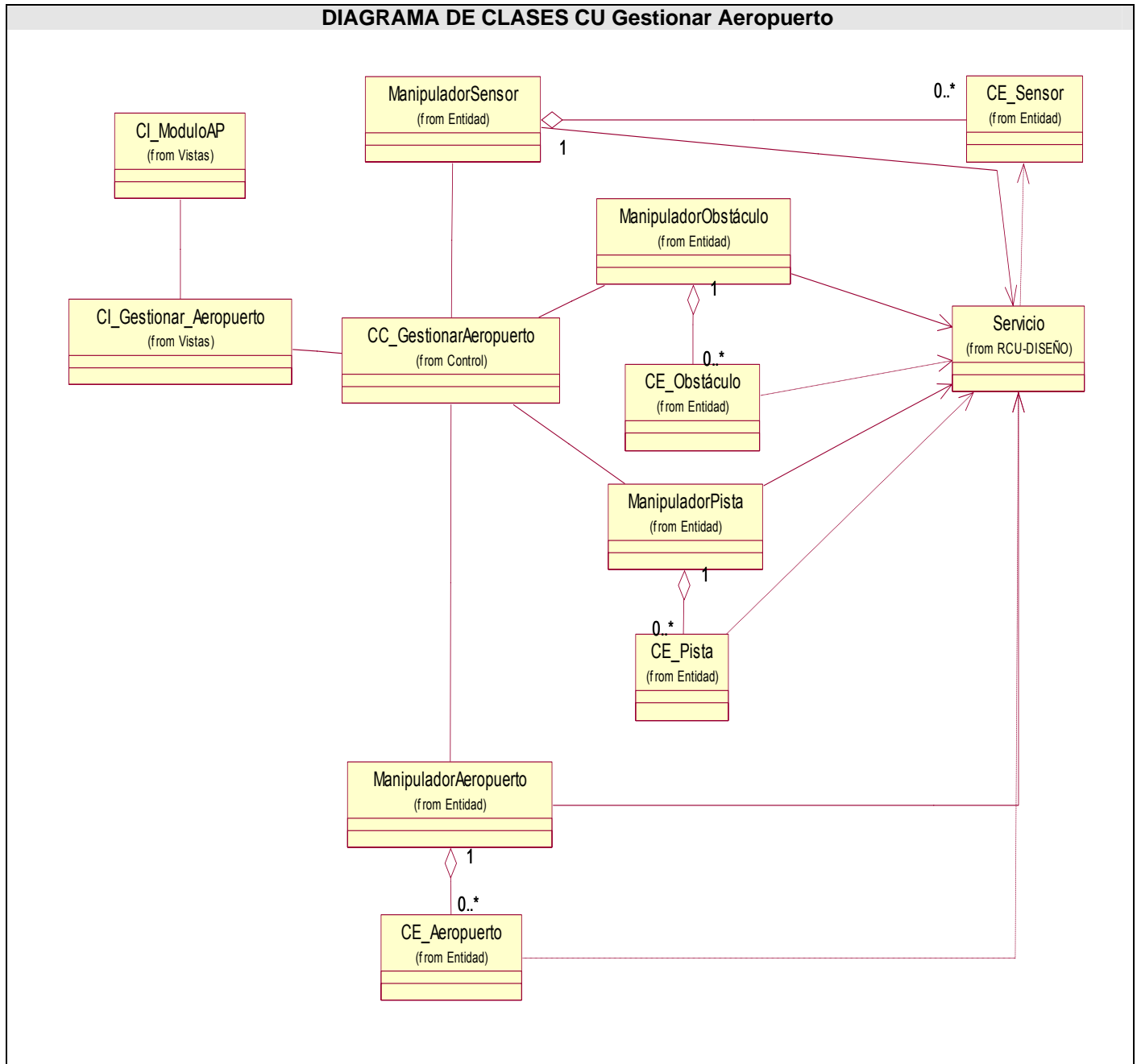
#### 3.3.1 Diagramas de interacción. [Ver Anexo 5.](#)

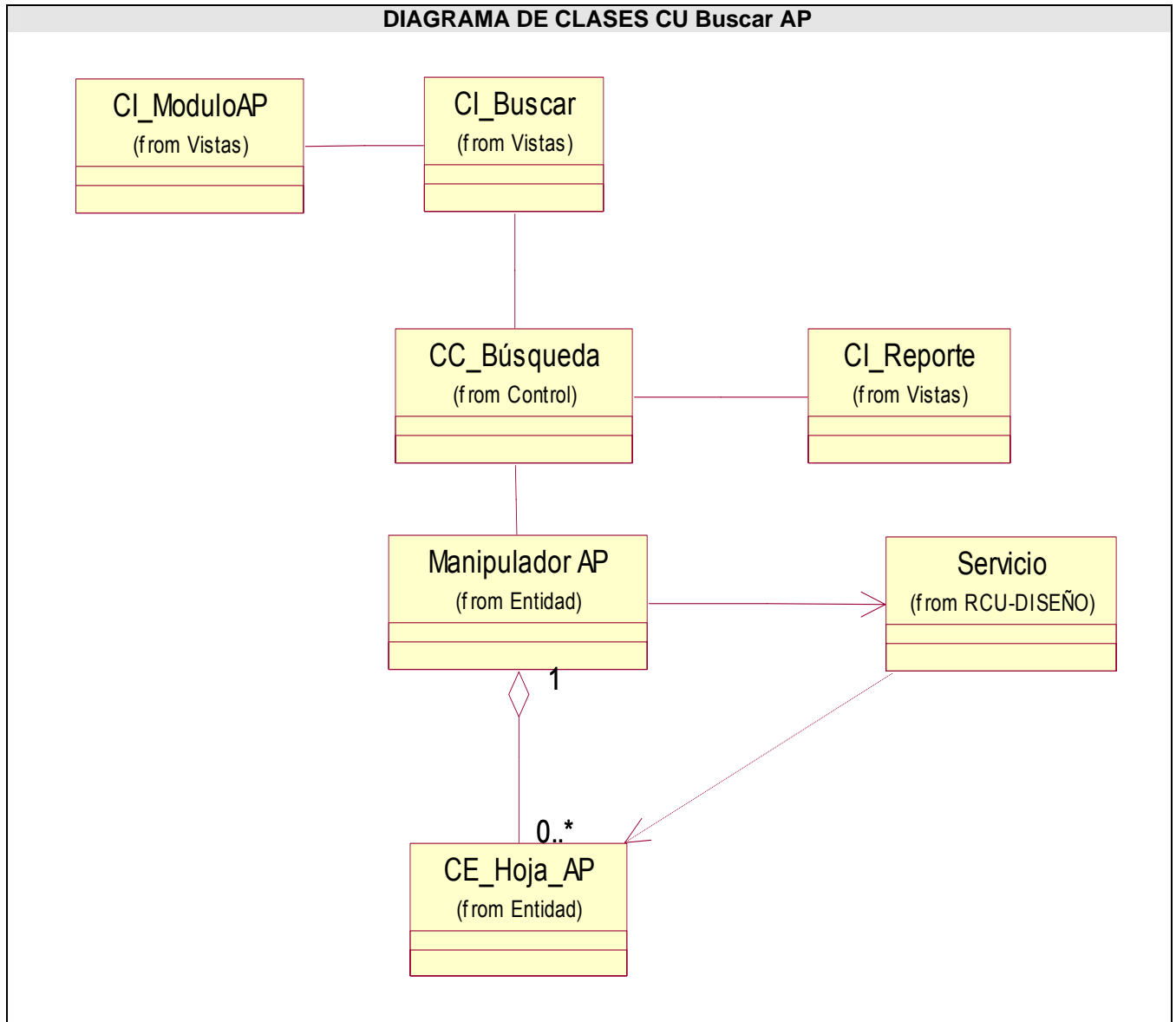




### 3.3.2 Diagrama de Clases.



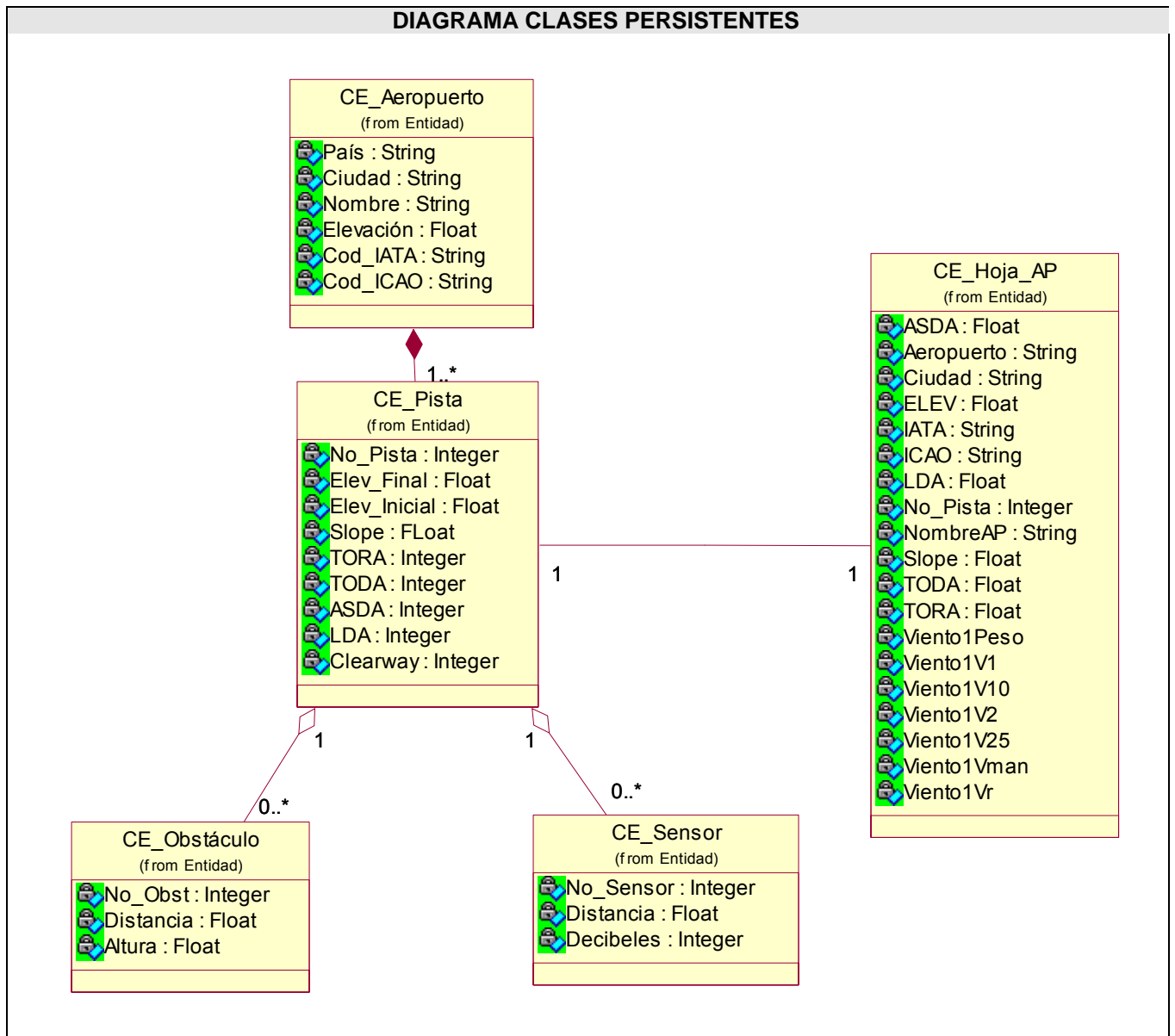






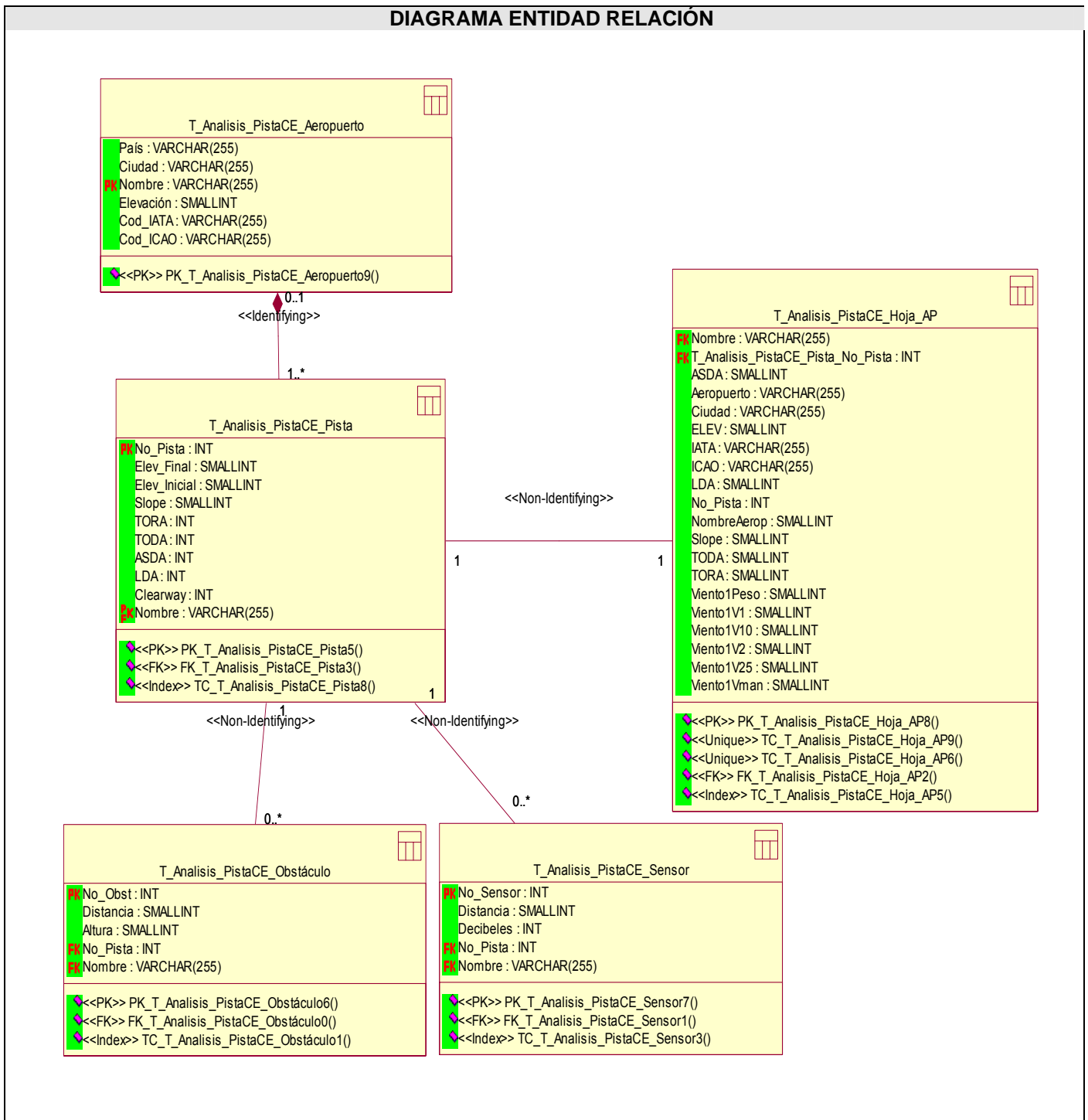
### 3.4 Diseño de la Base de Datos.

#### 3.4.1 Modelo Lógico de Datos.





## 3.4.2 Modelo Físico de Datos.





### **3.5 Conclusiones.**

En este capítulo se detalló a mayor profundidad la propuesta del sistema a través del análisis y el diseño. Es decir, se expuso que se quería que hiciera la aplicación mediante los artefactos del análisis, basándose en los requisitos del software y cómo el sistema cumple esos objetivos mediante los artefactos del diseño ya que refinan el análisis pues toman en cuenta los requisitos no funcionales.



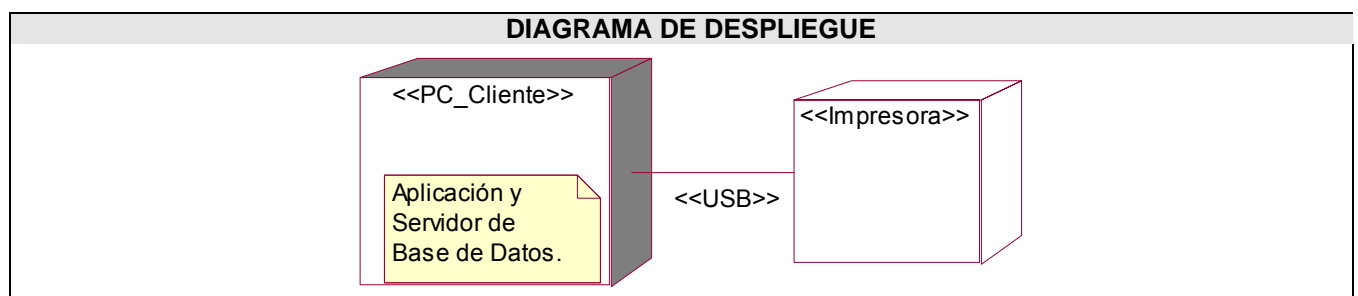
## Capítulo 4: Implementación.

### 4.1 Introducción.

En la implementación se comienza a trabajar con el resultado del diseño y se implementa el sistema en términos de componentes, es decir, ficheros de código fuente, scripts, ficheros de código binario, ejecutables y similares y cómo estos se organizan de acuerdo a los nodos específicos en el modelo de despliegue.

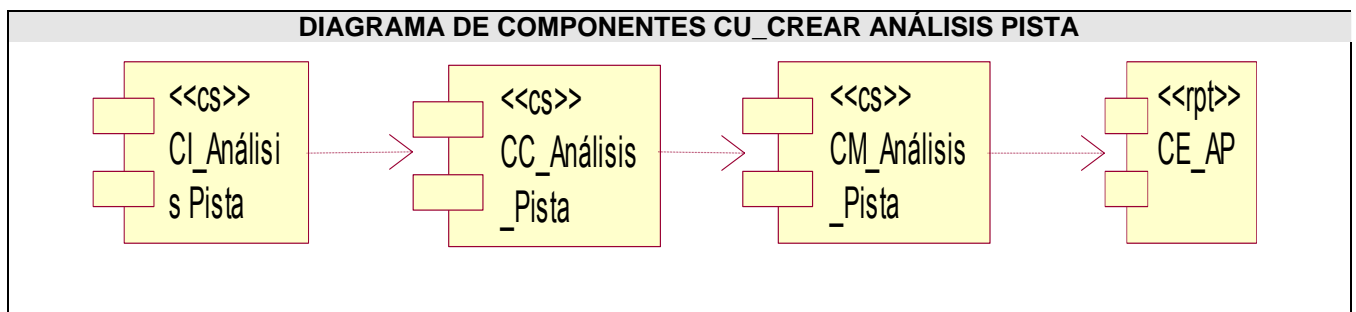
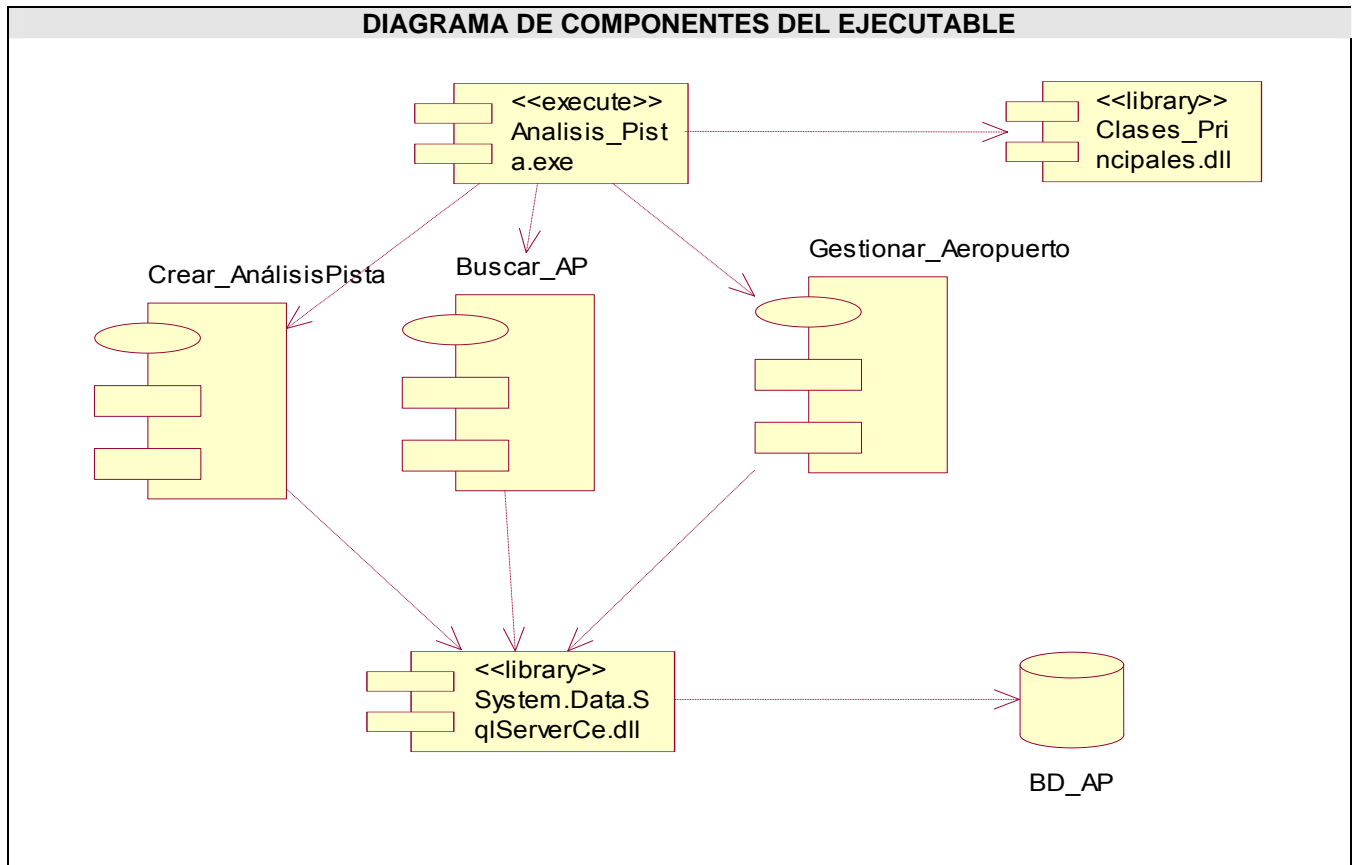
Los diagramas de despliegue y componentes, que son artefactos generados en este flujo de trabajo conforman lo que se conoce como un modelo de implementación al describir los componentes a construir y su organización y dependencia entre nodos físicos en los que funcionará la aplicación. Afortunadamente, la mayor parte de la arquitectura del sistema es capturada durante el diseño, siendo el propósito principal de la implementación el desarrollar la arquitectura y el sistema como un todo.

### 4.2 Diagrama de despliegue.

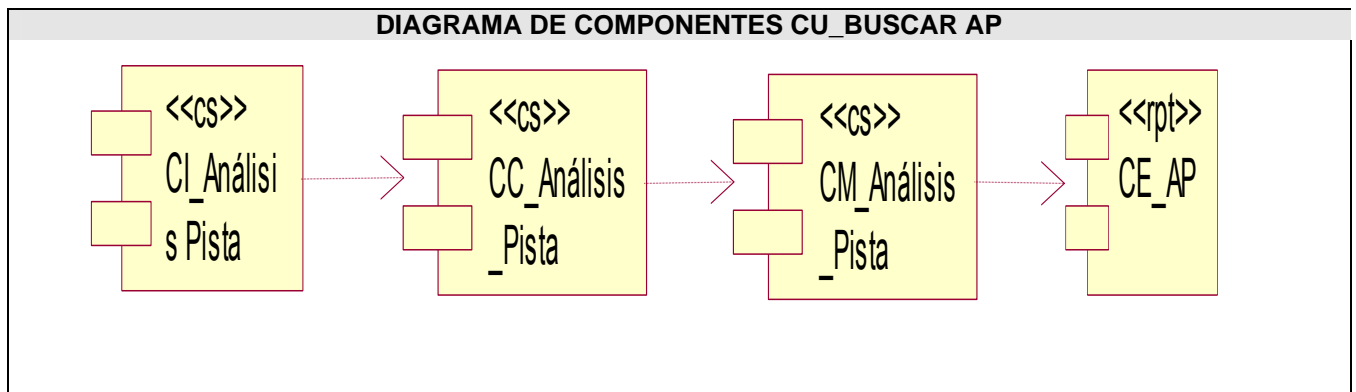
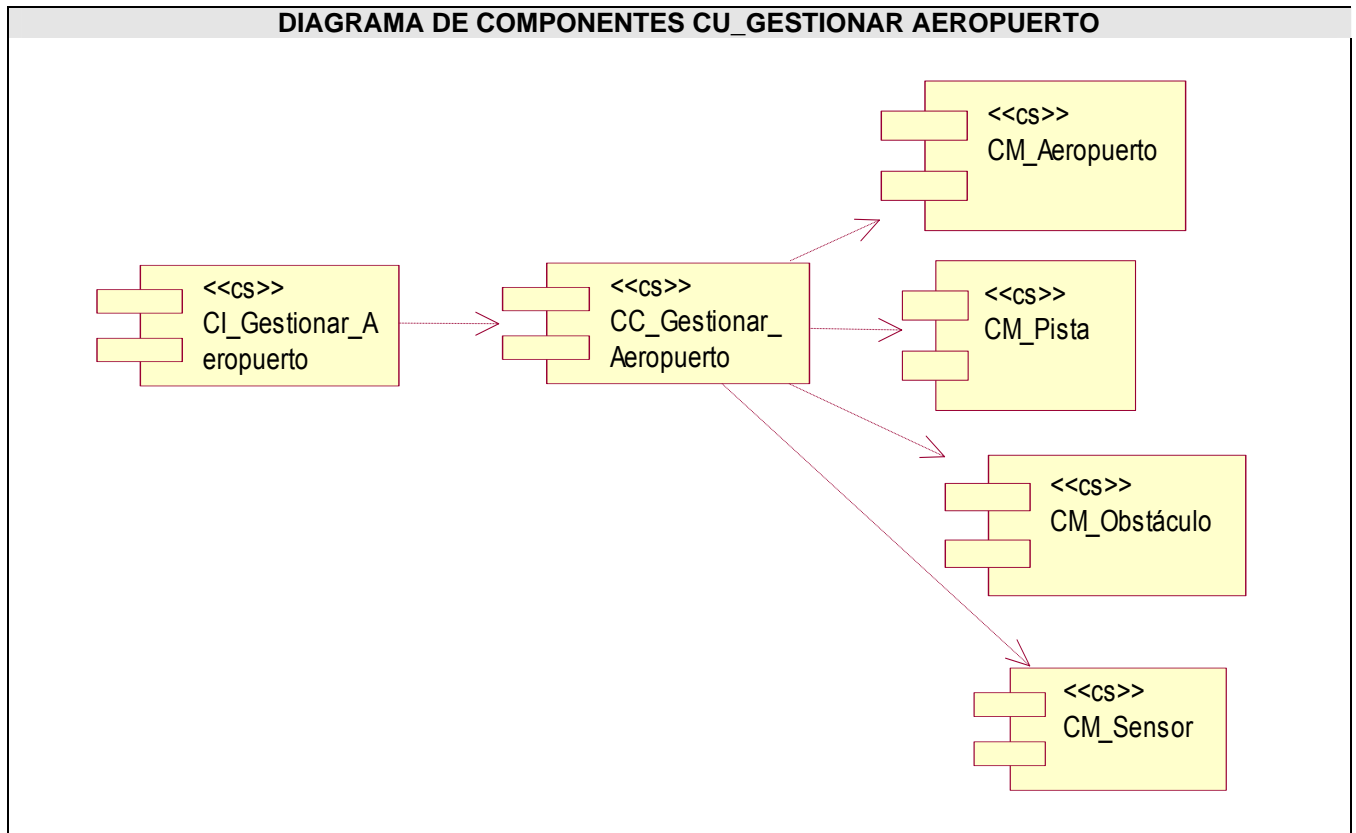




### 4.3 Diagrama de componentes.







## *Capítulo IV: Implementación*



### **4.4 Conclusiones.**

En este capítulo se expusieron los artefactos que componen el modelo de implementación permitiendo desarrollar la arquitectura y el sistema como un todo, pues se definieron los componentes a construir así como su organización y dependencia.



## Capítulo 5: Estudio de Factibilidad.

### 5.1 Introducción.

El estudio de factibilidad de un proyecto es de suma importancia, pues se obtiene el costo y el tiempo de duración del proyecto, analizándose si es conveniente continuar desarrollándolo.

La estimación mediante el análisis de Puntos de Casos de Uso es un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

### 5.2 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	2	0	0

## Capítulo V: Estudio de Factibilidad



	de interfaz basada en texto.			
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	1	5
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	1	15

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

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

$$UUCW = 30$$

$$UUCP = 6+30$$

$$UUCP = 36$$

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

$$UCP = UUCP * TCF * EF$$

## Capítulo V: Estudio de Factibilidad



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 performance 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	<b>2.5</b>
T8	Portabilidad	2	5	10

## Capítulo V: Estudio de Factibilidad



T9	Facilidad de cambio	1	5	5
T10	Concurrencia	1	1	1
T11	Incluye objetivos especiales de seguridad	1	5	5
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

## Capítulo V: Estudio de Factibilidad



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.

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

$$EF = 1.4 - 0.03 * 11$$

$$EF = 1.07$$

$$UCP = UUCP * TCF * EF$$

$$UCP = 36 * 1.04 * 1.07$$

$$UCP = 40.0608$$

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 CU. Si el total es 3 o 4 se utiliza 28 HH/Puntos de CU. Si el total es mayor a 5 se recomienda efectuar cambios en el proyecto ya que tiende a perecer en corto tiempo.

## Capítulo V: Estudio de Factibilidad



En este caso se puede afirmar que:

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

$$E = 40.0608 * 20$$

$$E = 801.216 \text{ Horas/Hombre}$$

Calcular esfuerzo de todo el proyecto.

Actividad	Porcentaje %	Horas/Hombre
Análisis	20%	400.608 horas-hombre
Diseño	30%	600.912 horas-hombre
Implementación	40%	801.216 horas-hombre
Pruebas	0%	0 horas-hombre
Sobrecarga (otras actividades)	10%	200.304 horas-hombre
Total	100%	2003.04 horas-hombre

Tabla 5.5 Esfuerzo del proyecto.

Si  $E_T = 2003.04$  Horas/ hombre y se estima que cada mes promediado 192 horas laborables, quedaría

$$E_T = 10.4325 \text{ mes/hombre.}$$

-Costo del Proyecto

En el caso del salario mensual es de \$100.00

CH: Cantidad de hombres

Tiempo: Tiempo total del Proyecto.

$$CH = 2 \text{ hombres.}$$

$$CHM = 2 \text{ hombres} * \text{Salario.}$$

$$CHM = 200 \text{ \$/mes}$$

$$\text{Costo} = CHM * E_T$$

$$\text{Costo} = 200 * 10.4325$$

$$\text{Costo} = \$ 2086.5$$

Tiempo total del Proyecto:



## *Capítulo V: Estudio de Factibilidad*



$$\text{Tiempo} = E_T / CH$$

$$\text{Tiempo} = 10.4325 \text{ meses} / 2 \text{ hombres}$$

$$\text{Tiempo} = 5.21625 \text{ meses}$$

Del resultado obtenido se interpreta que con 2 hombres el proyecto tiene un tiempo de duración de 5.21625 meses y el costo total que se estima es de \$ 2086.

### **5.3 Beneficios tangibles e intangibles.**

Esta aplicación tiene como beneficio fundamental agilizar los procesos de realización de los cálculos necesarios en la creación de los Análisis de Pista, erradicando así las posibles inclusiones de errores humanos, pues estos cálculos son sumamente engorrosos y en ocasiones se realizan en situaciones extremas, donde el factor tiempo es crucial.

Con la creación del sistema se pueden apreciar los siguientes beneficios:

- ❖ Disminuirá considerablemente el tiempo empleado en la elaboración de los Análisis de Pista, pues estas operaciones se realizaban de forma manual, logrando realizar varios análisis en menor tiempo.
- ❖ Contará con una interfaz agradable a la vista, organizada y sencilla, que permitirá una rápida y fácil utilización del mismo. Facilitará la gestión de los datos de los aeropuertos para los cuales se desea el Análisis de Pista.
- ❖ Permitirá la disminución de los gastos en materiales de oficina.
- ❖ Ofrecerá la posibilidad de crear los Análisis de Pista en el formato deseado, Completo u Operativo.
- ❖ Agilizará la búsqueda de Análisis de Pista previamente elaborados pues se tendrán almacenados.

### **5.4 Análisis de costos y beneficios.**

El desarrollo de todo producto de software cuesta dinero, esos costos se dividen en gastos de personal, y gastos de materiales necesarios para su elaboración, por lo que retribuir ese costo depende directamente

## *Capítulo V: Estudio de Factibilidad*



de los beneficios que traería su implementación, monetarios y de tipo social. 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 de cálculo en los Análisis de Pista 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 los Análisis de Pista que cubriera las necesidades de la tripulación del IL-96 300. Analizando el costo del proyecto y los numerosos beneficios que reporta, se puede concluir que su implementación es realmente factible.

### **5.5 Conclusiones.**

La herramienta propuesta reportará beneficios significativos e importantes para la Empresa Cubana de Aviación, contribuyendo a mejorar los servicios que se prestan en esa empresa ante la salida de un vuelo, lo que demuestra la factibilidad de implementar la herramienta propuesta.

## *Conclusiones.*



## **Conclusiones.**

Con la implantación exitosa del sistema, se dará solución a los problemas actuales existentes en las oficinas de Cubana de Aviación, 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 recursos materiales que se utilizaban.

Se ha puesto en práctica una solución con un análisis, diseño e implementación fácil de manipular por los futuros programadores que puedan darle mantenimiento al sistema.

El desarrollo del sistema se ha basado en el Proceso Unificado de Desarrollo de Software, realizándose una modelación del negocio con vista a entender el contexto donde se va a implantar el sistema y se presentó un grupo de artefactos para los flujos de trabajo de requisitos, análisis y diseño e implementación que propone dicho proceso.

En el proceso de selección de las herramientas se determinó utilizar para la implementación del software el lenguaje de programación C#, la plataforma .NET, el SQL Server 2000, y el Rational Rose para la parte de modelamiento del sistema.

Con los beneficios expuestos, tangibles e intangibles, se determinó que el desarrollo de la aplicación es realmente factible, siendo indudable la revolución que experimentará este servicio en la Empresa Cubana de Aviación.



## **Recomendaciones.**

El software creado tiene implementada cada una de sus funcionalidades pero para lograr un mejor funcionamiento del mismo se reflejan las siguientes recomendaciones:

- ❖ Poner a prueba el sistema durante un período de tiempo significativo para comprobar los resultados obtenidos pues es un factor importante para mantener la seguridad en los vuelos.
- ❖ 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.
- ❖ Crear la Base de Datos final que se le entregará al usuario con toda la información contenida para un completo funcionamiento.
- ❖ Investigar la posibilidad de realizar una versión de esta aplicación que realice los Análisis de Pista de todos los modelos de aviones que utiliza la Empresa Cubana de Aviación.



## Bibliografía.

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 Holguin, 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*
8. 2004. [Disponible en:
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.mspx>, 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](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](http://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](http://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*, 2003.
23. Manual Básico de Operaciones.



## Anexos.

### Anexo 1:

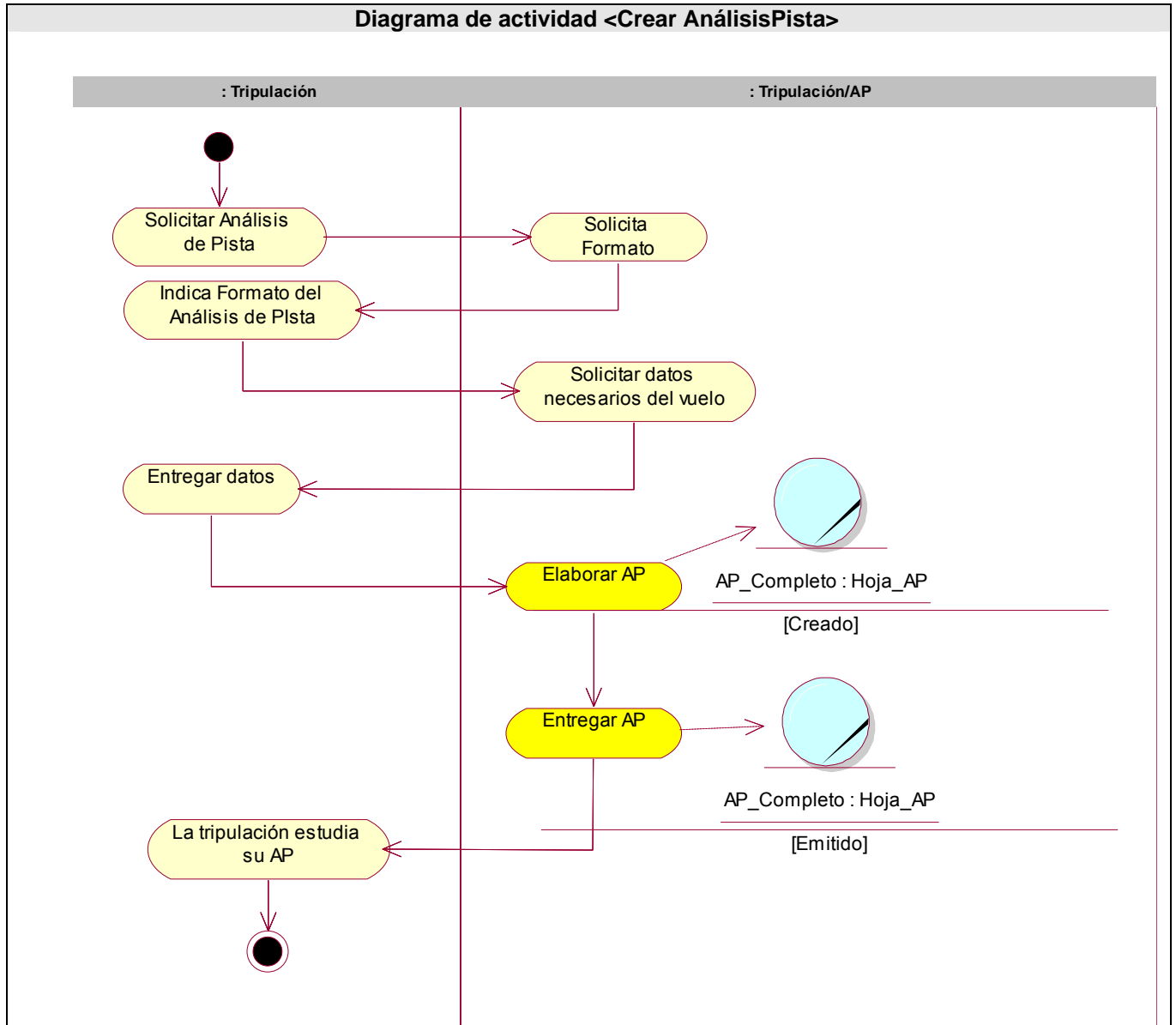
#### Descripción del Caso de Uso Crear AnálisisPista.

<b>Caso de Uso:</b>	<b>Crear AnálisisPista</b>	
<b>Actores:</b>	<b>Tripulación</b>	
<b>Trabajadores:</b>	<b>Tripulación/AP</b>	
<b>Resumen:</b>	<p>El caso de uso se inicia cuando la tripulación va a realizar un vuelo y necesita un análisis de pista para el mismo, este puede ser un Análisis de Pista Completo (para todas las temperaturas) u Operativo (sólo para cuatro temperaturas, una por debajo y dos por encima de la solicitada). El encargado en la tripulación de elaborar el Análisis de Pista examina cada uno de los gráficos que intervienen en este flujo. El caso de uso finaliza cuando la tripulación obtiene el Análisis de Pista correspondiente.</p>	
<b>Flujo Normal de Eventos</b>		
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>	
1- La tripulación solicita un Análisis de Pista en uno de los dos formatos.	2- Tripulación/AP analiza cada uno de los gráficos que intervienen en el flujo de trabajo realizando los cálculos necesarios.	
	3- Le entrega una copia a la Tripulación del Análisis de Pista solicitado.	
<b>Flujos Alternos</b>		
<b>Acción del Actor</b>	<b>Respuesta del Negocio</b>	
<b>Mejoras</b>		
<b>Prioridad</b>	<b>Crítica</b>	



Anexo 2:

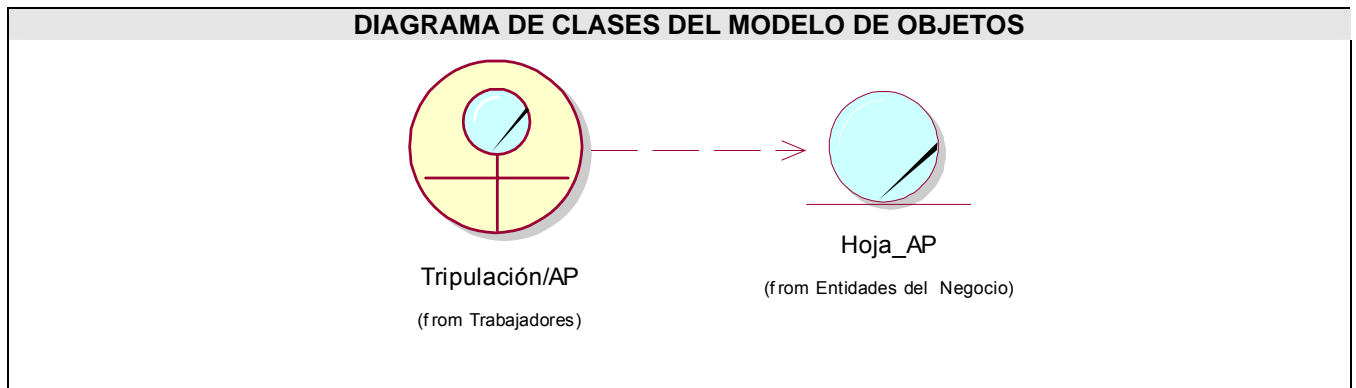
Diagrama de Actividades <Crear AnálisisPista>





**Anexo 3:**

**Diagrama de clases del Modelo de Objetos.**







**Anexo 4:**

**Casos de Uso Extendidos.**

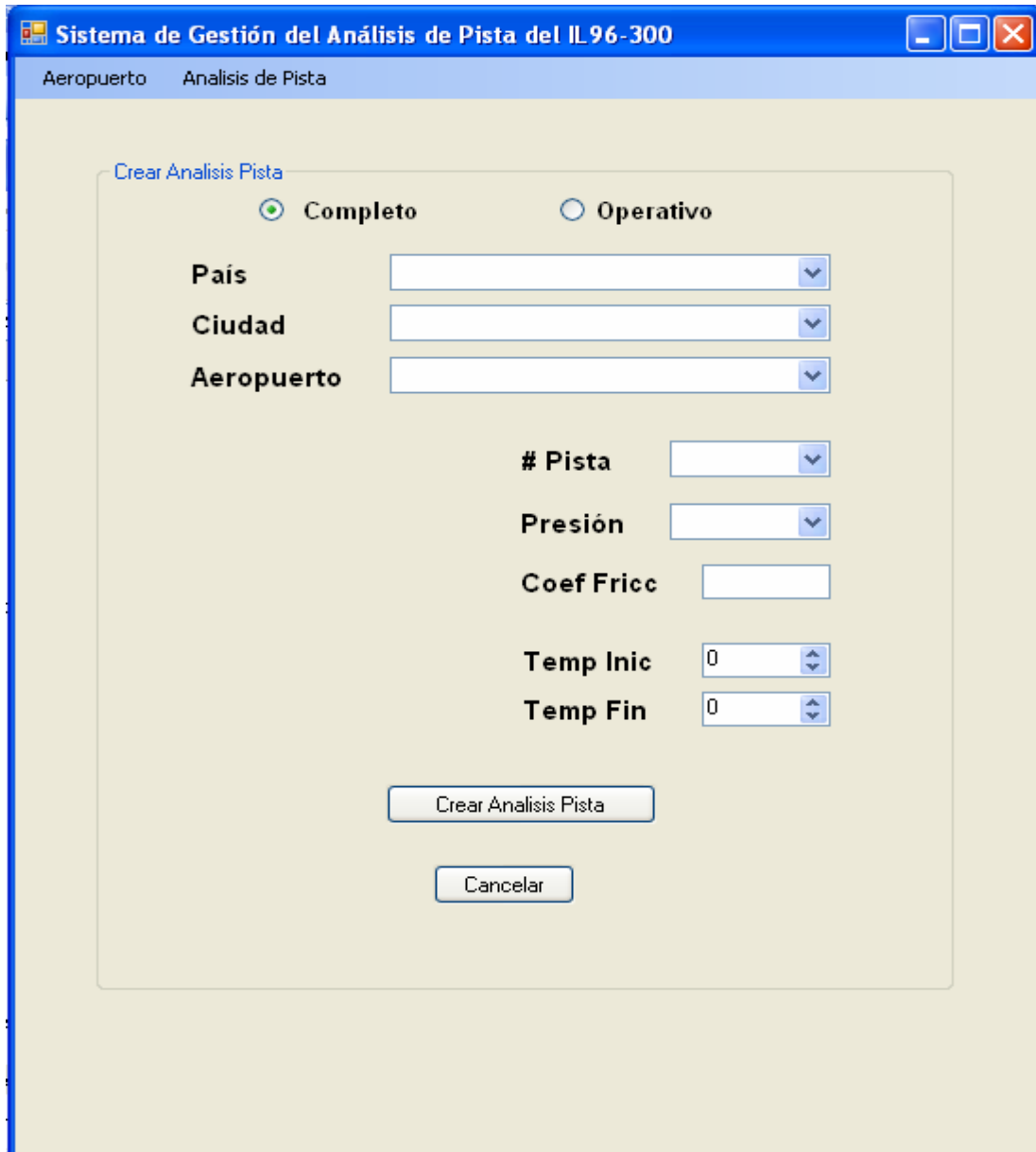
**Descripción de Casos de Uso <Crear AnálisisPista>**

<b>Caso de Uso:</b>	Crear AnálisisPista	
<b>Actores:</b>	Tripulación/AP	
<b>Resumen:</b>	El caso de uso se inicia cuando el encargado de la tripulación necesita realizar un Análisis de Pista en el formato solicitado. Introduce los datos necesarios para ello y termina con la devolución del Análisis pedido.	
<b>Referencia:</b>	R1, R1.1,R1.2, R2, R2.1,R2.2	
<b>CU asociados:</b>		
<b>Precondiciones:</b>	El Aeropuerto donde está la pista para la que se requiere el Análisis debe estar creado con antelación.	
<b>Flujo Normal de Eventos</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
1- El usuario selecciona dentro de la interfaz de Análisis de Pista la opción de Crear Análisis de Pista y selecciona el formato, Completo u Operativo.	2- El sistema lee los valores de la Base de Datos.	
	3- El sistema llena los campos necesarios.	
4- El usuario selecciona los datos para llenar sus campos.	5 El sistema realiza los cálculos y devuelve el Análisis de Pista solicitado.	
	6- El sistema guarda el Análisis de Pista elaborado.	
<b>Flujos Alternos</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	

# Anexo 4



<b>Poscondiciones:</b>	
<b>Prioridad:</b>	Crítica
<b>Especificaciones Complementaria:</b>	



**Sistema de Gestión del Análisis de Pista del IL96-300**

Aeropuerto Analisis de Pista

Crear Analisis Pista

Completo  Operativo

**País** [dropdown]  
**Ciudad** [dropdown]  
**Aeropuerto** [dropdown]

**# Pista** [dropdown]  
**Presión** [dropdown]  
**Coef Fricc** [text]  
**Temp Inic** [dropdown]  
**Temp Fin** [dropdown]

Crear Analisis Pista  
Cancelar

Fig. 1 Interfaz Crear Análisis de Pista.



**Descripción de Casos de Uso <Gestionar Aeropuerto>**

<b>Caso de Uso:</b>	Gestionar Aeropuerto	
<b>Actores:</b>	Tripulación/AP	
<b>Resumen:</b>	El encargado de la tripulación de gestionar los aeropuertos puede adicionar, modificar o eliminar un aeropuerto con toda su información, una pista, un obstáculo o un sensor dentro de un aeropuerto específico.	
<b>Referencia:</b>	R3, R3.1, R3.2, R3.3	
<b>CU asociados:</b>		
<b>Precondiciones:</b>	Si lo que se desea es modificar o eliminar, el aeropuerto sobre el que se va a trabajar debe estar creado con anterioridad.	
<b>Flujo Normal de Eventos</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
1- El usuario necesita adicionar, modificar o eliminar los datos de un aeropuerto, una pista, un obstáculo o un sensor dentro de un aeropuerto específico.	1.1- El sistema ejecuta alguna de las siguientes acciones: <ul style="list-style-type: none"> <li>• Si decide adicionar un aeropuerto, una pista, un obstáculo o un sensor, ir a la sección "Adicionar"</li> <li>• Si decide modificar los datos de un aeropuerto, una pista, un obstáculo o un sensor, ir a la sección "Modificar".</li> <li>• Si decide eliminar un aeropuerto, una pista, un obstáculo o un sensor, ir a la sección "Eliminar".</li> </ul>	
<b>Sección "Adicionar"</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
2- El Administrador del Sistema introduce los datos del aeropuerto, la pista, el obstáculo o el sensor que desea adicionar.	3- El sistema verifica que todos los campos estén llenos.	



	4- El sistema verifica que este aeropuerto, pista, obstáculo o sensor no exista.
	5- El aeropuerto, pista, obstáculo o sensor se almacena en el sistema.
	6- Se muestra un mensaje informándole al usuario que la operación ha sido realizada satisfactoriamente y finaliza el caso de uso.
<b>Flujos Alternos</b>	
	4- Se emite un mensaje para que llene los campos obligatorios.
	5- Si el aeropuerto, pista, obstáculo o sensor existe se muestra un mensaje informativo y finaliza el caso de uso.
<b>Sección "Modificar"</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
2- El usuario selecciona si va a modificar un aeropuerto, una pista, un obstáculo o un sensor.	3- El sistema lee los datos de la Base de Datos y llena los campos de la interfaz de usuario.
4- El usuario realiza las modificaciones deseadas.	5- Se verifica que los campos obligatorios estén llenos.
	6- Se actualiza la información y finaliza el caso de uso.
<b>Flujos Alternos</b>	
	6- Se emite un mensaje para que llene los campos obligatorios.
<b>Sección "Eliminar"</b>	

## Anexo 4



Acción del Actor		Respuesta del Sistema
2- El usuario selecciona si va a eliminar un aeropuerto, una pista, un obstáculo o un sensor.		3- El sistema verifica la existencia del objeto seleccionado para eliminar.
		4- El sistema pide confirmación
5- El usuario confirma su acción.		6- El sistema elimina el aeropuerto, la pista, el obstáculo o el sensor.
Flujos Alternos		
Acción del Actor		Respuesta del Sistema
5- El usuario cancela la acción.		6- El sistema desecha la solicitud.
<b>Poscondiciones:</b>		
<b>Prioridad:</b>	Crítica	
<b>Especificaciones Complementaria:</b>		



The screenshot shows a software window titled "ANALISIS DE PISTA" with a menu bar containing "Aeropuerto" and "Analisis de Pista". The main area is titled "Gestionar Aeropuerto" and contains three dropdown menus for "País", "Ciudad", and "Aeropuerto". Below these are three radio buttons: "Adicionar" (selected), "Modificar", and "Eliminar".

Underneath, there is a section titled "Adicionar" with three radio buttons: "Pistas" (selected), "Obstáculos", and "Sensores".

Inside the "Adicionar Pistas" section, there is a sub-dialog box titled "Adicionar Pistas" containing the following fields and buttons:

- # Pista: A spinner box showing the value "0".
- Elev Inic: A text input field.
- Elev Fin: A text input field.
- TORA: A text input field.
- ASDA: A text input field.
- TODA: A text input field.
- LDA: A text input field.
- Buttons: "Cancelar" and "Aceptar".

Fig. 2 Interfaz Adicionar Pista.

The image shows a screenshot of a software application window titled 'ANALISIS DE PISTA'. The window has a blue title bar with standard Windows window controls (minimize, maximize, close). Below the title bar, there is a menu bar with 'Aeropuerto' and 'Analisis de Pista'. The main content area is light beige and contains several sections. The first section is 'Gestionar Aeropuerto', which includes three dropdown menus for 'País', 'Ciudad', and 'Aeropuerto'. Below these are three radio buttons: 'Adicionar', 'Modificar' (which is selected), and 'Eliminar'. The second section is 'Modificar', which contains four radio buttons: 'Aeropuerto', 'Pistas', 'Sensores', and 'Obstáculos' (which is selected). Below this is a sub-section titled 'Modificar Obstáculos' containing four input fields: '# Pista' (a dropdown menu), '# Obstáculo' (a dropdown menu), 'Distancia' (a text input field), and 'Altura' (a text input field). At the bottom of this sub-section are two buttons: 'Cancelar' and 'Adicionar'.

Fig. 3 Interfaz Modificar Obstáculo.



The screenshot shows a software window titled "ANALISIS DE PISTA" with a sub-tab "Aeropuerto Analisis de Pista". The main area is titled "Gestionar Aeropuerto" and contains three dropdown menus for "País", "Ciudad", and "Aeropuerto". Below these are three radio buttons: "Adicionar", "Modificar", and "Eliminar" (which is selected). A dialog box titled "Eliminar" is open, containing four radio buttons: "Aeropuerto", "Pistas", "Obstáculos", and "Sensores". Below these are three dropdown menus for "# Pista", "# Sensor", and "# Obstáculo". At the bottom of the dialog are "Cancelar" and "Aceptar" buttons.

Fig. 4 Interfaz Eliminar.





**Descripción de Casos de Uso <Buscar AP>**

<b>Caso de Uso:</b>	Buscar AP	
<b>Actores:</b>	Tripulación	
<b>Resumen:</b>	El caso de uso se inicia cuando la tripulación desea buscar un análisis de pista previamente elaborado y termina cuando el sistema le muestra el Análisis de Pista solicitado.	
<b>Referencia:</b>	R4	
<b>CU asociados:</b>		
<b>Precondiciones:</b>	El Análisis de Pista solicitado debe estar previamente elaborado.	
<b>Flujo Normal de Eventos</b>		
	<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	1- El usuario selecciona dentro de la interfaz de Análisis de Pista la opción de Buscar Análisis de Pista y especifica el formato de los Análisis de que desea buscar.	2- El sistema lee los valores de la Base de Datos y llena los campos.
	3- El Usuario selecciona los datos necesarios para la búsqueda, como son el País, la Ciudad, el nombre del aeropuerto y el número de la pista.	4- El sistema le muestra los Análisis que existan de esa pista específica.
	5- El usuario selecciona el Análisis de Pista que desea ver.	6- El sistema muestra el Análisis de Pista solicitado.
<b>Flujos Alternos</b>		
	<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
<b>Poscondiciones:</b>		
<b>Prioridad:</b>	Secundario	
<b>Especificaciones Complementaria:</b>		

The image shows a software window titled "Sistema de Gestión del Análisis de Pista del IL96-300". The window has a blue title bar with standard minimize, maximize, and close buttons. Below the title bar, there is a light blue header area with the text "Aeropuerto" and "Análisis de Pista". The main content area is a light beige color and contains a search form titled "Buscar Análisis Pista". The form includes four dropdown menus: "País", "Ciudad", "Aeropuerto", and "# Pista". Below the dropdown menus are two buttons: "Generar Reporte" and "Cancelar".

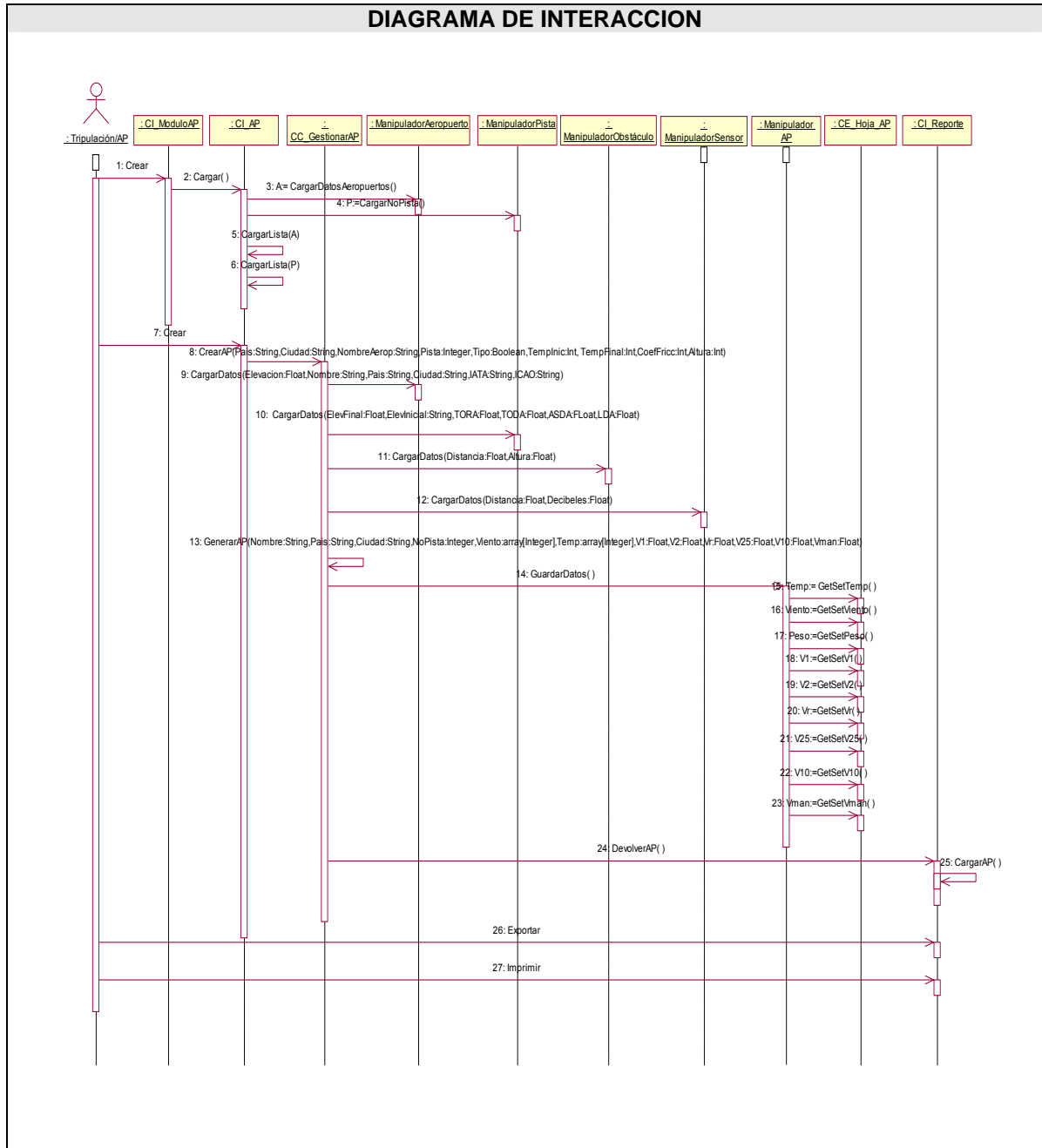
Fig. 5 Interfaz Buscar Análisis de Pista.

# Anexo 5:



## Anexo 5:

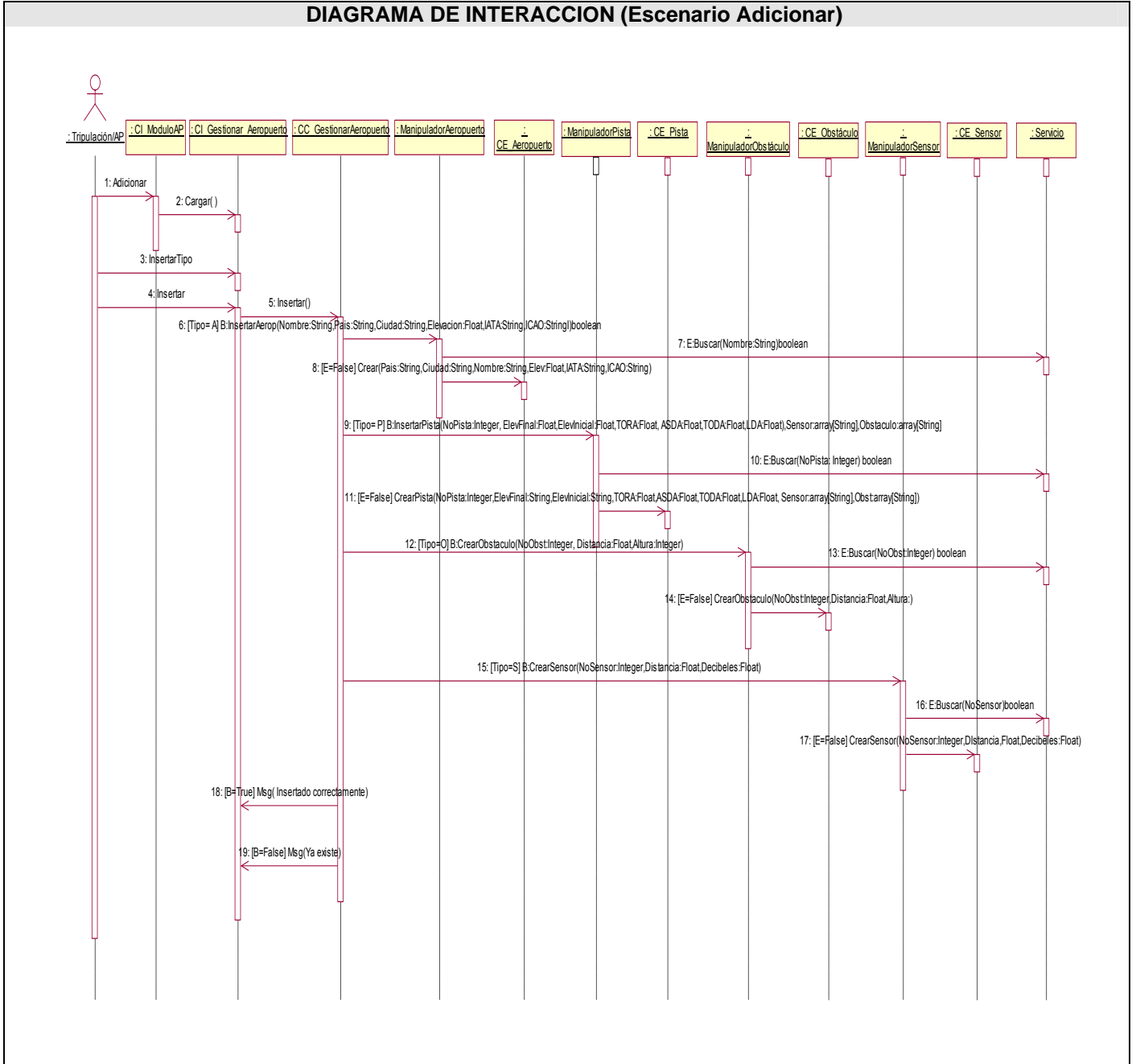
### Caso de Uso Crear AnálisisPista.



# Anexo 5:



## Caso de Uso Gestionar Aeropuerto.

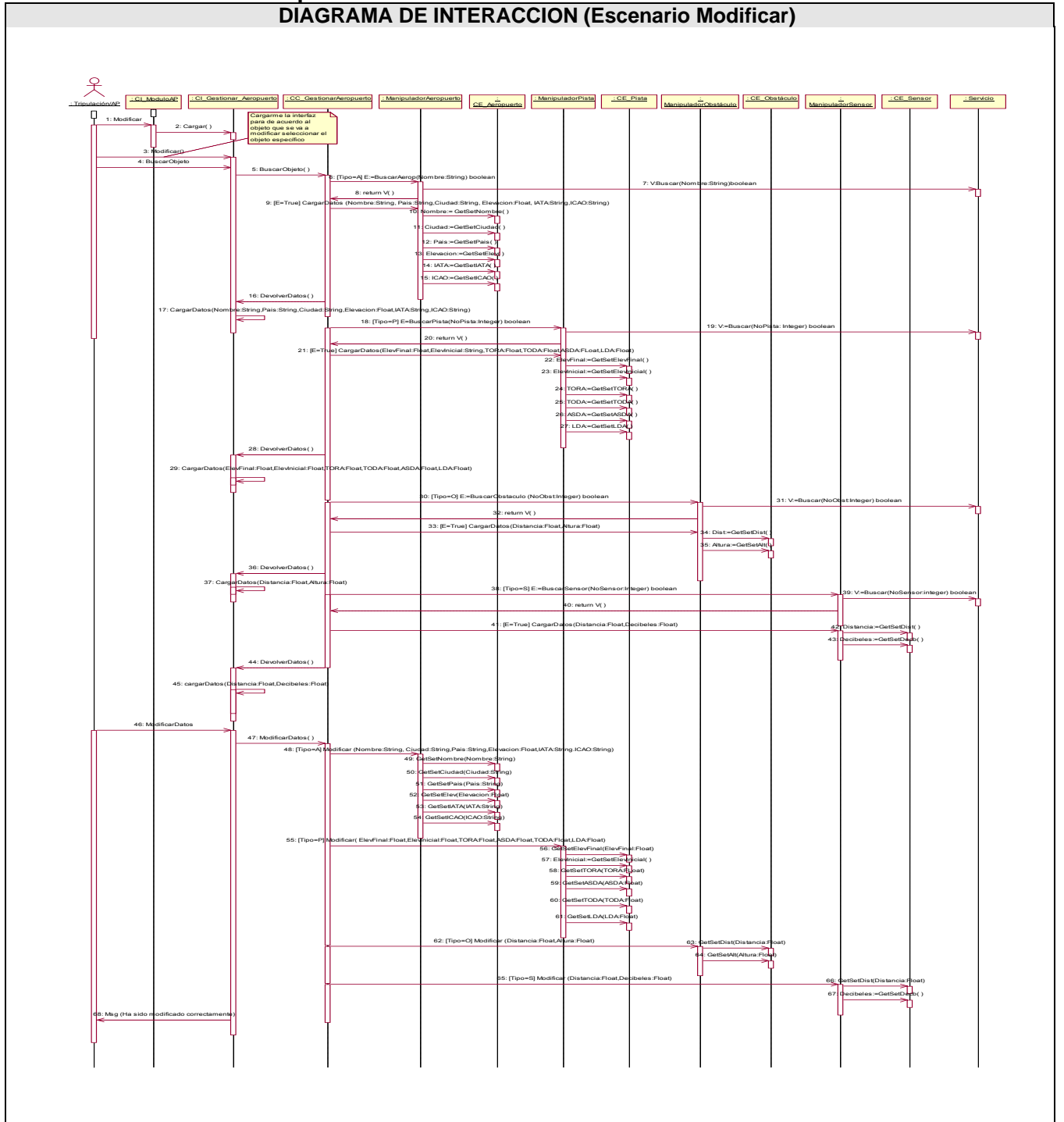


# Anexo 5:



## Caso de Uso Gestionar Aeropuerto.

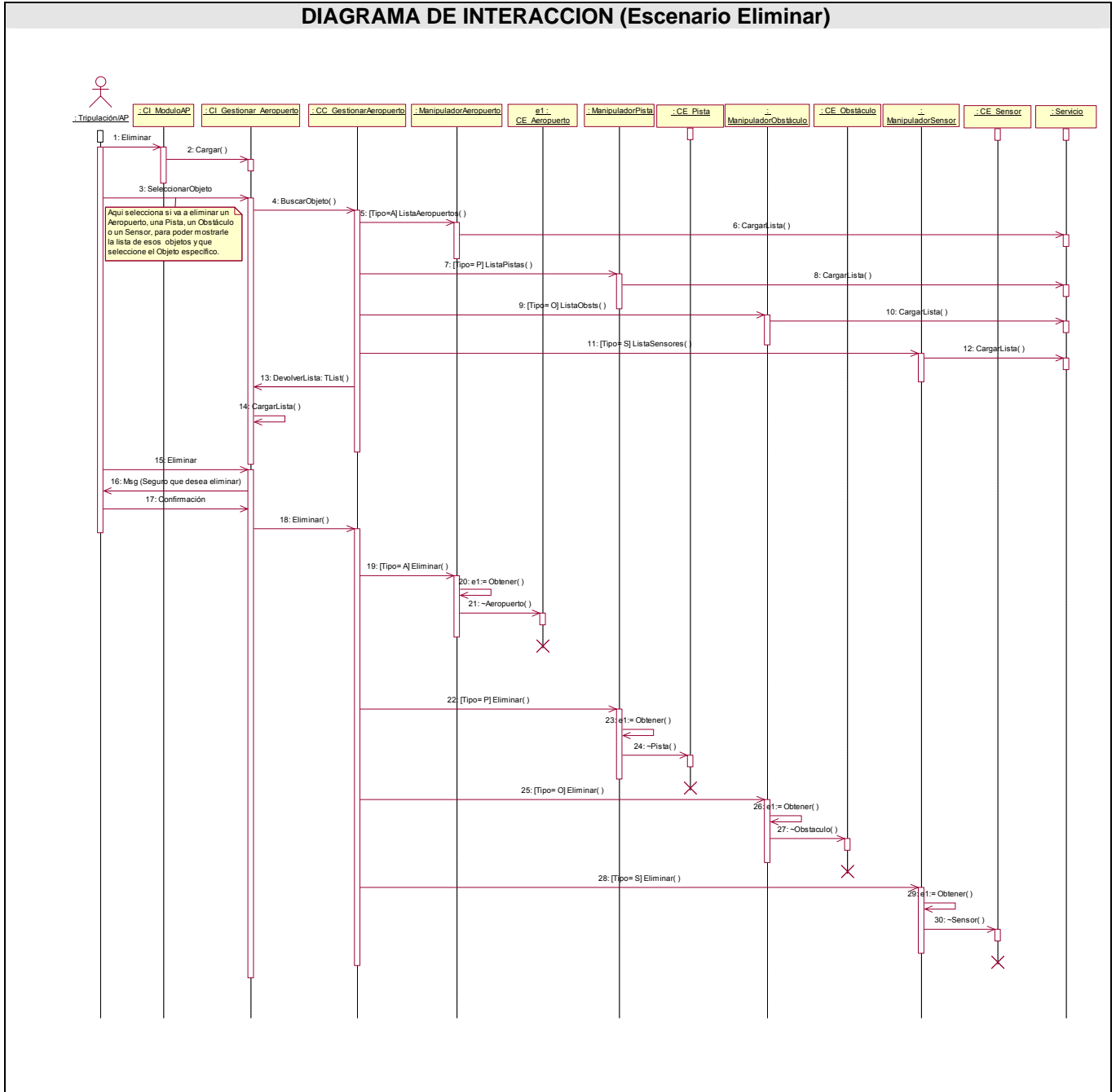
### DIAGRAMA DE INTERACCION (Escenario Modificar)



# Anexo 5:

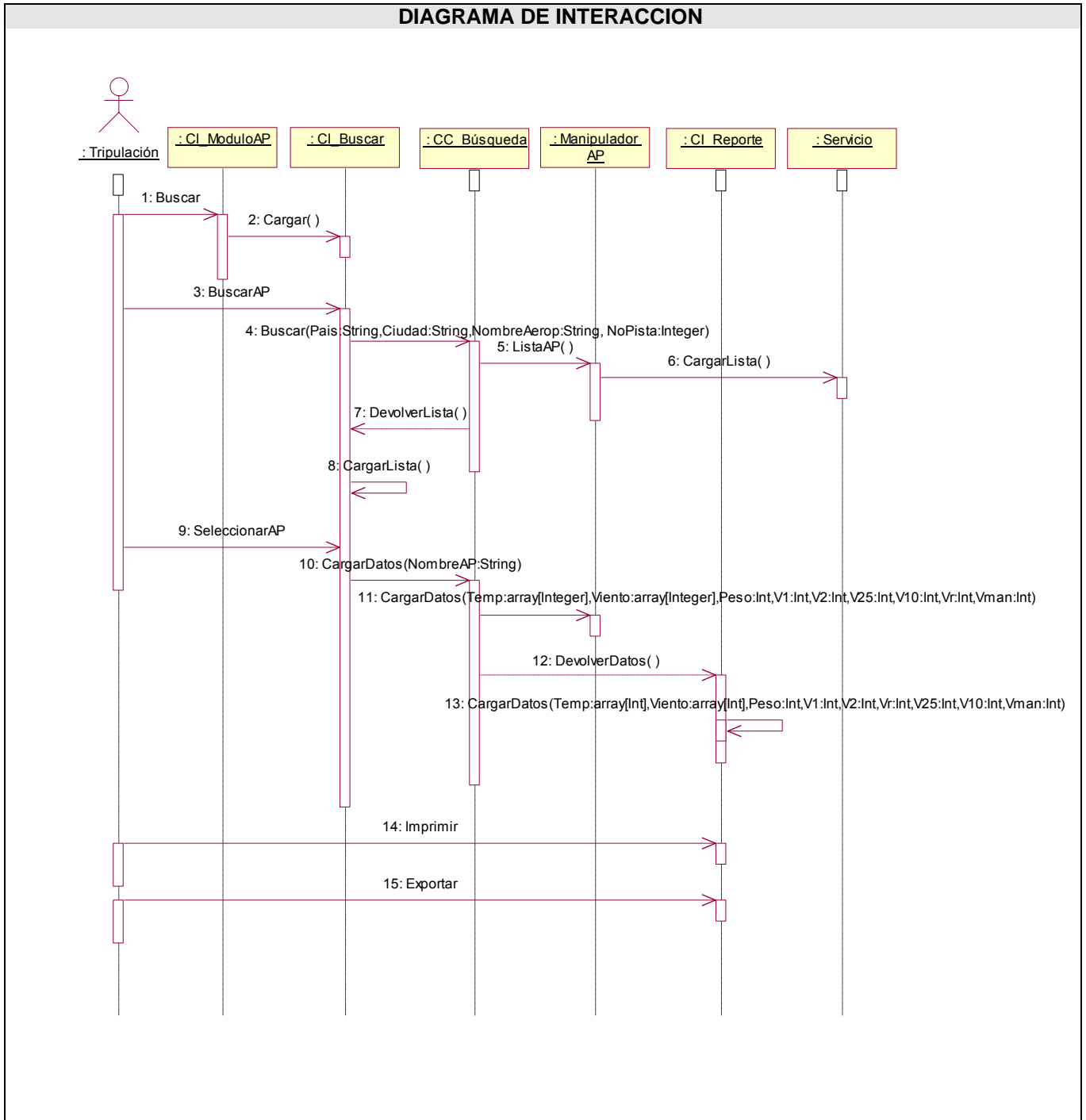


## Caso de Uso Gestionar Aeropuerto.





Caso de Uso Buscar AP





## Glosario de Términos.

1. **Velocidad de crucero:** La velocidad de crucero es la velocidad estable a la que se mueve el avión.
2. **Tripulación:** La tripulación son aquellas personas que se encargan del manejo del avión y de la atención a los pasajeros. En este caso esta compuestas por el Capitán, Piloto, Copiloto, Ingeniero de vuelo, Navegante, Despachador y Aeromozas.
3. **Tripulación/AP:** Es el encargado en la tripulación de realizar los Análisis de Pista y es el responsable de gestionar toda la información en la Base de Datos.
4. **Framework.NET:** La arquitectura .NET (.NET Framework) es el modelo de programación de la plataforma .NET para construir y ejecutar los servicios .NET.
5. **Biblioteca de clases de .Net:** La biblioteca de clases de .NET Framework es una biblioteca de clases, interfaces y tipos de valor que se incluye en Microsoft .NET Framework. Esta biblioteca brinda acceso a la funcionalidad del sistema y es la base sobre la que se crean las aplicaciones, los componentes y los controles de .NET Framework.
6. **COM+(Component Object Model +):** No es más que la integración de la arquitectura COM y MTS (Microsoft Transaction Server). A diferencia de MTS, esta nueva capa de ejecución no es opcional, COM+ es parte por defecto de la instalación del sistema operativo Windows 2000. Como COM, COM+, es basado sobre componentes binarios y programación basada en interfaces. Los Componentes COM+ pueden ser actualizados y extendidos una vez que estén en explotación sin afectar a las aplicaciones clientes que los usan en la producción. De este modo, la combinación de la tecnología COM+ junto con las técnicas de programación orientada a objeto, nos ofrece una importante simplificación en el proceso de desarrollo de aplicaciones informáticas.
7. **CLR (Common Language Runtime):** El CLR es el núcleo de la plataforma .NET. Es el motor encargado de gestionar la ejecución de las aplicaciones para ella desarrolladas y a las que ofrece numerosos servicios que simplifican su desarrollo y favorecen su fiabilidad y seguridad.





8. **Compilador JIT:** El compilador JIT nos brinda la posibilidad de verificar y garantizar que el programa no realice ninguna operación que pueda poner en peligro la seguridad ni la integridad del sistema.
9. **SQL-DMO (Distributed Management Objects):** SQL-DMO (objetos distribuidos de administración) proporciona un interfaz programable al servidor del SQL. Es una gran herramienta para automatizar las tareas rutinarias más flexibles.
10. **Performance:** Desempeño con respecto al rendimiento, es decir, medida o cuantificación de los componentes del avión.
11. **TORA:** Takeoff Run Available por sus siglas en inglés, es la distancia disponible dentro de la pista para el despegue.
12. **ASDA:** Accelerate-Stop Distance Available por sus siglas en inglés, es la distancia con que cuenta el piloto para acelerar o frenar dentro de la pista de forma segura antes de levantar vuelo.
13. **TODA:** Takeoff Distance Available por sus siglas en inglés, es la distancia disponible para el despegue.
14. **LDA:** Landing Distance Available por sus siglas en inglés, es la distancia disponible para el aterrizaje.
15. **MTW:** Peso Máximo que resiste el avión en Rampa y Rodaje. Que en este caso del IL-96 300 permite 250 toneladas.
16. **MTOW:** Máximo Peso de Despegue del avión, limitado estructuralmente y por las normas de pilotaje.
17. **MLW:** Máximo Peso de Aterrizaje que permite el avión limitado estructuralmente y por las normas de pilotaje.
18. **Tires Speed:** Velocidad permitida por las ruedas. Para este avión esa velocidad es igual a 178 nudos.
19. **ICAO:** Es un código alfanumérico de 4 dígitos que identifica a cada aeropuerto alrededor del mundo. Estos códigos son definidos por la organización internacional de la aviación civil (ICAO). Estos códigos son únicos. La primera letra es asignada generalmente pro el continente y representa un país o un grupo de países dentro de ese continente. La segunda letra



representa un país dentro de esa región, y las dos restantes se utilizan para identificar cada aeropuerto.

20. **IATA:** Es un código de tres letras que designa a cada aeropuerto en el mundo. Estos códigos son decididos por la organización internacional para el transporte aéreo (International Air Transport Association) IATA. Los códigos no son únicos: 323 de los aproximadamente 20.000 códigos son usados por más de un aeropuerto. Las letras mostradas claramente en las etiquetas de equipaje usadas en las mesas de embarque de los aeropuertos son una muestra del uso de estos códigos.
21. **Slope:** Describe la medida de inclinación o gradiente de una pista, también se le conoce como gradiente.
22. **Coefficiente de Fricción:** Determina el nivel de contaminación de la pista por los fenómenos del tiempo tales como lluvia o agua, nieve, aguanieve o hielo. Todas estas condiciones pueden reducir la capacidad de aceleración del avión aumentando la fricción.
23. **Distancia Balanceada:** Igual distancia entre el inicio de la pista y el punto donde se alcanza  $V_1$  y de  $V_1$  hasta el final de la pista.
24. **Flaps:** Dispositivos hipersustentadores.
25.  **$V_1$ :** Velocidad de decisión, cuando se alcanza esta velocidad el piloto decide si despegue o se queda en la pista.
26.  **$V_2$ :** Velocidad segura de despegue.
27.  **$V_{f125}$ :** Velocidad mínima para la retracción de flaps de 25 grados a 10 grados.
28.  **$V_{f10}$ :** Velocidad mínima para la retracción de flaps 10 grados a 0 grado.
29.  **$V_{man}$ :** Velocidad mínima de maniobra con el avión limpio de mecanización.
30.  **$V_r$ :** Velocidad de rotación, a la cual se comienza a despegar el avión de la superficie de soporte para llevarlo al aire.
31. **Análisis de Pista Completo:** Es un tipo de formato en el que se devolverán los datos, contiene toda la información necesaria para realizar el Despegue y Aterrizaje para todas las temperaturas.
32. **Análisis de Pista Operativo:** Es un formato más pequeño, solo se devuelven los datos para un rango de temperaturas especificado.
33. **MBO:** Manual Básico de Operaciones.