

CAPITULO I

INTRODUCCIÓN

1.1 DESCRIPCIÓN DEL PROBLEMA

Hoy en día para que las Empresas de Transporte Aéreo cumplan con las Operaciones Aéreas que han programado, involucra una serie de actividades que deben ser cumplidos de acuerdo a estándares y reglamentos emitidos por las Autoridades Aeronáuticas.

Los Sistemas Informáticos en las Empresas cumplen un papel importante en la toma de decisiones, es por eso que en la actualidad se encuentra una serie de software que de una u otra forma apoyan en el cumplimiento de sus procesos.

A nivel mundial existen muchas Empresas Aeronáuticas que cumplen con la labor importante de transportar de un punto a otro al personal de pasajeros, y que al contar con una serie de factores de trascendencia tales como aviones sofisticados, personal capacitado, sistemas informáticos, entre otros, pueden encontrarse a un nivel competitivo. En nuestro país encontramos pocas empresas nacionales que se dedican al transporte aéreo, más aún tomando en cuenta que muchos de sus procesos operacionales todavía se cumplen en forma manual, nos vemos con la necesidad de realizar un estudio exhaustivo de sus tareas para una posible automatización y un mejoramiento continuo.

1.2 ANTECEDENTES

La Empresa de Transporte Aéreo TAME con el objetivo de mantener una plataforma de operaciones sólida, así como el cumplimiento de las Regulaciones Aeronáuticas de

Operaciones Aéreas punto fundamental en el AOC (Certificado de Operaciones Aérea) emitido por la Dirección de Aviación Civil, se ha propuesto automatizar los procesos involucrados en la emisión de la Orden de Vuelo.

La “Orden de Vuelo” es un documento en el que se encuentra toda la información relacionada a los vuelos que diariamente debe cumplir la Empresa. Estos datos son recopilados de áreas tales como el Centro de Control Operacional (CCO), Gerencia de Mantenimiento, Jefatura de Tripulación, Jefatura de Operaciones, Departamento de Entrenamiento de Tripulantes, entre otros, que luego de ser validados y verificando que cumplan con las normas establecidas por las autoridades, permitirán la creación de este documento.

1.3 SITUACIÓN ACTUAL

Actualmente la Empresa cumple al día con los procesos involucrados en la emisión de la Orden de Vuelo, pero por la falta de una automatización de los mismos, la labor se vuelve muchas veces interminable y con falta de precisión, los departamentos emiten información que al pasar de las horas son cambiados drásticamente y no permiten que el documento se imprima a la hora planificada, sin tomar en cuenta que día a día encontramos mucha información ingresada varias veces en hojas electrónicas, dando como resultado redundancia en sus datos y un desgaste innecesario de recursos humanos y materiales.

1.4 JUSTIFICACIÓN

La Empresa se encuentra en un proceso de mejoramiento continuo tal es el caso que ha realizado la compra de nuevas aeronaves, su personal se encuentra en una preparación diaria para el mejoramiento de sus tareas, los sistemas realizan el trabajo de soporte que el usuario

requiere, todo esto porque hoy en día los pasajeros requieren de un mejor servicio y atención por parte de las aerolíneas y si se quiere estar a un nivel competitivo, la Empresa deberá seguir cumpliendo la meta que se ha trazado. Calidad en su servicio y un mejoramiento continuo.

El que una aeronave recorra las rutas establecidas diariamente, involucra varios procesos a los cuales se debe agregar un mayor control y seguimiento, es por eso que para emitir una Orden de Vuelo se debe seguir primero por una larga trayectoria de recopilación de información en los diferentes departamentos de la Empresa, ahora si se lograría automatizar las tareas de cada usuario, obtendríamos ese control y se brindaría un servicio de calidad al personal de pasajeros.

La automatización de la “Orden de Vuelo” reflejará la interrelación que existe entre los diferentes departamentos de la Empresa que juntos apoyan a la toma de decisiones en función de resultados, acciones y operaciones en bien de la Empresa, los clientes internos y externos. Además, permitirá determinar la fiabilidad, operabilidad y confiabilidad en las operaciones aéreas.

1.5 OBJETIVOS

1.5.1 Objetivo General

Desarrollar el Sistema Computarizado “AeroFlex” que permita Gestionar y Controlar la Emisión de la Orden de Vuelo en la Empresa de Transporte Aéreo TAME, sobre un ambiente Windows con Plataforma .NET.

1.5.2 Objetivos Específicos

- Analizar los procesos involucrados en la emisión de las Órdenes de Vuelo.
- Aplicar una Metodología Orientada a Objetos para el desarrollo del proyecto.
- Analizar, Diseñar, Desarrollar e Implantar el Sistema Computarizado que permita Gestionar y Controlar la Orden de Vuelo.

1.5.3 Beneficios del Sistemas

- Contar con una herramienta informática que apoye en la toma de decisiones a la hora de ejecutar las Operaciones Aéreas.
- Integrar los procesos que se ejecutan en las diferentes Gerencias y Departamentos de la Empresa y que están involucrados directa o indirectamente en la emisión de la Orden de Vuelos.
- Apoyar al proceso de certificación ISO 9001 por medio de un Sistema Informático que cumpla con normas establecidas por la Empresa TAME y la Dirección de Aviación Civil.

1.6 ALCANCE

El Sistema Computarizado “AeroFlex” consta de la automatización de 4 procesos claramente identificados: Administración de Usuarios del Sistema, Programación de Itinerarios de Aeronaves, Programación de Itinerarios de Tripulantes y Programación de Entrenamiento de Tripulantes. La integración de estos permite la emisión de la Orden de Vuelo.

1.6.1 Administración de Usuarios del Sistema

Este módulo permite la creación y asignación de perfiles de usuarios con lo que se puede administrar la seguridad en el sistema. Esto con la finalidad de controlar el acceso a las diferentes opciones del sistema y el ingreso con el usuario y clave autorizados.

1.6.2 Programación de Itinerarios de Aeronaves

En el presente módulo se realiza la Programación de Itinerarios de Aeronaves recuperando información de los Vuelos de Itinerario que fueron creados desde el Sistema de Reservas perteneciente a la Empresa. Con estos vuelos y dependiendo de la fecha que se vaya a programar, se puede asignar las aeronaves que cumplirán las operaciones aéreas.

La disponibilidad de aeronaves es emitida desde la Gerencia de Mantenimiento y se dispone de esta lista sin tomar en cuenta si las aeronaves tienen o no mantenimiento programado, pudiendo afectar a la operación. Actualmente en la Empresa esta coordinación se la realiza en forma verbal entre las áreas involucradas y se lo mantendrá de la misma manera ya que su automatización no es motivo de este estudio.

1.6.3 Programación de Entrenamiento de Tripulantes

Para que un tripulante pueda cumplir con las operaciones aéreas debe permanentemente ser entrenado con la finalidad de mantenerse habilitado. En el presente módulo se administra la Programación de Entrenamiento de la Tripulación, permitiendo registrar la Malla Curricular, la Planificación Anual de Entrenamiento y la Asignación de Entrenamiento a cada Tripulante. El sistema no registra el cumplimiento del entrenamiento ya que esto es motivo de otro estudio.

1.6.4 Programación de Itinerarios de Tripulantes

Luego de asignar aeronaves a los diferentes vuelos que la Empresa debe cumplir diariamente, en este módulo se puede registrar su tripulación (Tripulación de Cabina y Tripulación de Pasajeros) de acuerdo al Equipo al que pertenezca (AirBus 320, Embraer, Boeing, Fokker). Además, el sistema controla que el número de tripulantes esté de acuerdo al equipo y lo establecido por las RDAC.

Se presenta un calendario en el que se refleja los vuelos, el entrenamiento, y otras actividades que puede cumplir cada tripulante en la Empresa. Actualizada diariamente esta información y junto con otros datos, el sistema permite la creación de la Orden de Vuelo, siendo éste el documento oficial en el que se visualiza las Operaciones Aéreas que la Empresa cumplirá.

CAPITULO II

MARCO TEORICO

Para el desarrollo del presente proyecto se ha tomado en cuenta una serie criterios e información que respalda las decisiones adoptadas en cuanto a la selección de Herramientas, Modelos de Procesos de Ingeniería, Metodología de Desarrollo, Lenguaje de Programación, Plataforma y Arquitectura de Desarrollo, entre otros.

2.1 PROCESO DE EMISIÓN DE LA ORDEN DE VUELO

La emisión de la Orden de Vuelo es una tarea fundamental a la hora de cumplir con las operaciones en la Empresa de Transporte Aéreo TAME. La Gerencia Comercial, Gerencia de Mantenimiento, Departamento de Entrenamiento de Tripulantes, Centro de Control Operacional, entre otros, se encuentran involucrados en este proceso, los cuales deben emitir información que en un momento determinado se integra y con una validación oportuna permiten la creación del documento “Orden de Vuelo”.

El proceso inicia con la emisión de los Itinerarios de Vuelo por parte de la Gerencia Comercial. Con esta información en la Gerencia de Operaciones y Mantenimiento deben coordinar para el cumplimiento de las operaciones, esto es, asignar las aeronaves disponibles a cada vuelo y su respectiva tripulación que la ejecutará.

Para asignar aeronaves y tripulación a un vuelo se debe tomar en cuenta factores tales como: Mantenimiento programador o no de las aeronaves, disponibilidad de los tripulantes según el departamento de recursos humanos y el departamento de entrenamiento, itinerarios de vuelo

emitidos con anticipación a los tripulantes, factores climáticos, además otros factores que de una u otra forma afectan el cumplimiento de las operaciones.

A continuación en la fig. 2.1 se presenta gráficamente los procesos involucrados en la Emisión de la Orden de Vuelo.

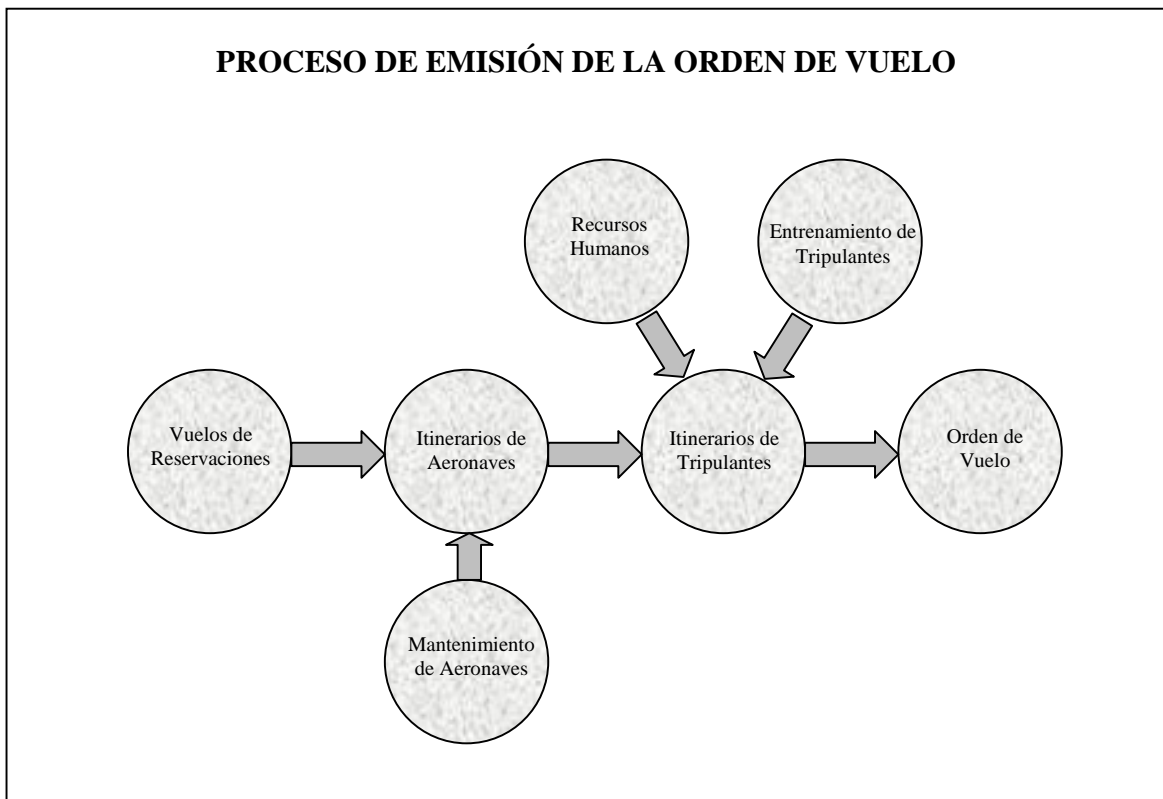


Figura 2.1 Proceso de Emisión de la Orden Vuelo

2.2 MODELOS DE PROCESOS DE INGENIERIA DE SOFTWARE

“...Para desarrollar software de calidad duradera, hay que idear una sólida base arquitectónica que sea flexible al cambio. Para desarrollar software rápida, eficiente y efectivamente, con el mínimo de desechos software y de trabajo repetido, hay que tener la gente apropiada, las

herramientas apropiadas y el enfoque apropiado. Para hacer todo esto de forma consistente y predecible, con una estimación de los costos del sistema en cada etapa de su vida, hay que disponer de un *Proceso de Desarrollo* sólido que pueda adaptarse a las necesidades cambiantes del problema en cuestión y de la tecnología”.¹

Un Modelo de Proceso de Software o también conocido como Modelo de Ciclo de Vida define un marco de trabajo que contiene áreas claves que deben ser tomadas en cuenta en la gestión de proyectos informáticos tales como: control de riesgos, administración de recursos humanos, control de calidad, costos, entre otros.

Independiente de la Metodología de Desarrollo de Sistemas que se adopte, todo Modelo de Proceso de Desarrollo debe contener fases tales como: Análisis, Diseño, Implementación, Pruebas, Implantación y Mantenimiento.

Según la Empresa, la naturaleza del proyecto, los empleados, los métodos y las herramientas a utilizarse se deberán seleccionar uno o varios Modelos de Procesos de Ingeniería de Software, dando como resultado, un modelo que contenga los aspectos y fortalezas más importantes del resto, que permita minimizar costos y riesgos que involucra el desarrollo de proyectos informáticos. A continuación se cita una serie de modelos:

- Modelo Lineal Secuencial o Cascada.
- Modelo en Espiral
- Modelo de Construcción de Prototipos.
- Modelo DRA.

¹ El Lenguaje Unificado de Modelado - Booch, Rumbaugh, Jacobson – Pág. 3

- Modelo Evolutivos de Proceso del Software.
 - Modelo Incremental.
 - Modelo Espiral.
 - Modelo Espiral WINWIN.
 - Modelo de Desarrollo Concurrente.
- Modelo Basado en Componentes.

Según la Metodología de Desarrollo de Sistemas adoptada para el cumplimiento del presente proyecto, se redacta con mayor detalle los Modelos de Ciclos de Vida a utilizarse.

2.2.1 Modelo de Ciclo de Vida en Cascada

En el presente modelo se necesita conocer es su totalidad los requerimientos del software a desarrollar, con la finalidad de cumplir con todas las tareas de una fase para pasar a otra. Al admitir iteraciones puede existir la posibilidad de encontrarse con modificaciones en una fase que exija volver a otra, esto se lo podrá realizar pero teniendo que recorrer nuevamente el resto de las fases.

Después de cada fase existe un control de tareas para ver si se puede o no pasar a la siguiente, obligando a trabajar con documentos que son revisados a la entrada y salida que certifican el trabajo cumplido.

- Ventajas:
 - La planificación es sencilla.
 - La calidad de producto es alta.
- Desventajas:

- Se debe conocer todos los requisitos al principio del proyecto.
- El producto se lo obtiene al final del proyecto.

2.2.2 Modelo de Ciclo de Vida Iterativo e Incremental

Combina los Modelos Lineal Secuencial o también conocido como Modelo en Cascada con la filosofía interactiva del Modelo de Construcción de Prototipos.

Se basa en la entrega de incrementos los cuales consta de “entregables” para el cliente y que son partes del producto final. Esto es, cada incremento entrega el software en partes pequeñas y utilizables y que son construidos sobre aquél que ya ha sido entregado.

Los incrementos afrontan las modificaciones del producto anterior a fin de cumplir con las necesidades del cliente, y la entrega de las nuevas funciones y características. Proporcionan al usuario la funcionalidad requerida y una plataforma de evaluación.

- Ventajas:
 - El producto es el requerido por el usuario.
 - En el transcurso del ciclo de vida, el usuario puede ir recibiendo partes del producto lo que le motivará permanentemente.
- Desventajas:
 - Si no se sigue una correcta metodología, el producto puede salir de los plazos establecidos, involucrando mayores costos.

2.2.3 Modelo de Ciclo de Vida en Espiral

Consiste en una serie de ciclos que se repiten hasta la entrega del producto. Estos ciclos pasan por todas las fases obteniendo en su finalización un producto más ampliado en relación al ciclo anterior. En esto es parecido al Modelo Incremental con la diferencia que toma en cuenta los riesgos (Requerimientos no comprendidos en su totalidad, diseños mal estructurados, errores encontrados en la implementación, etc.) .

Al terminar una iteración se verifica que cumpla con los requisitos establecidos y que funcione correctamente, dando la oportunidad de integrar en esta tarea al cliente final.

- Ventajas:
 - No necesita una definición completa de los requisitos para empezar a funcionar.
 - La validación de los requisitos se lo puede hacer al final de cada iteración.
 - Minimiza los riesgos en el proyecto, ya que si se encuentra algo errado solo se tendrá que revisar la iteración actual.
- Desventajas:
 - Es complicado evaluar los riesgos.
 - Se requiere de la participación continua del cliente, lo que muchas veces es difícil conseguirlo.

Este método puede ser aplicado a proyectos grandes donde es importante el factor riesgo y cuando no se pueda definir todos los requerimientos al principio.

2.3 METODOLOGIA DE DESARROLLO DE SISTEMAS

En décadas pasadas desde que se inició a desarrollar software se lo hacía en una forma primitiva, tomando en cuenta tan solo la codificación y la corrección de errores. No había una

especificación documentada del producto que se entregaba, cuando el sistema aparentemente cumplía con las especificaciones y parecía que no tenía errores inmediatamente era entregado, no se dedicaba tiempo en la planificación del proyecto, no existía una gestión de recursos humanos y económicos, etc.

Actualmente aún se encuentra ese tipo de sistemas informáticos que son desarrollados sin seguir una metodología dando como resultado que los costos sobrepasan lo planificado, el tiempo de desarrollo excede de lo previsto, el producto carece de un control de calidad, entre otros. Es por tales razones que se considera que es importante definir una Metodología de Desarrollo de Sistema que se adapte a la situación actual de la Empresa y del sistema a construir.

A continuación se nombra una serie de inconvenientes que se encuentran al desarrollar sistemas informáticos que no siguen una metodología que dicte lineamientos a seguir:

- No se soluciona satisfactoriamente las necesidades de las empresas.
- No se cumplen los proyectos en los tiempos planificados.
- Los costos proyectados no son los cumplidos.
- Los ajustes y modificaciones de los sistemas los vuelven dependientes de los desarrolladores.
- La solución informática no puede crecer conforme la organización lo requiere.

2.3.1 Definición de Metodología de Desarrollo de Software

Una Metodología de Desarrollo de Sistemas define el Proceso de Ciclo de Vida a seguir y la forma ordenada y efectiva en que se realiza las diferentes actividades en cada etapa. Especifica

las entradas que reciben y las salidas que producen cada fase. Dicta los lineamientos correctos para la gestión del proyecto.

2.3.2 Metodología de Desarrollo de Software Orientada a Objetos

Presenta un paradigma distinto a la Metodología Estructurada y no necesariamente mejor o peor. Por las ventajas que brinda como la reutilización de código, basada en componentes, fácil de mantenimiento, entre otros, las empresas actualmente han adoptado esta metodología y tomando en cuenta que los desarrolladores puede explotar el poder de los lenguajes orientados a objetos tales como clases, objetos, herencia, entre otros.

2.4 PROCESO UNIFICADO DE RATIONAL (RUP)

Es una Metodología de Desarrollo de Software Orientada a Objetos creada por Rational Software Corporation. Representa la unión de todas las metodologías antes presentadas hasta el año 1998, e incorpora el trabajo de los 3 autores (Ivar Jacobson, Grady Booch y James Rumbaugh) de UML, más las aportaciones fundamentales de empresas que han implementado el Proceso Unificado en su trabajo cotidiano.

Presenta las siguientes características:

- Basado en componentes.
- Usa el Lenguaje Unificado de Modelado (UML) para su notación.
- Dirigido por Casos de Uso.
- Centrado en la Arquitectura.
- El Ciclo de Vida que adoptado es Iterativo e Incremental.

El Proceso Unificado se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema de información. Cada ciclo representa una versión del producto que a su vez constan de 4 fases: Inicio, Elaboración, Construcción y Transición (fig. 2.2). Cada fase se subdivide en iteraciones con sus incrementos resultantes. Estas iteraciones terminan con un hito que son representados por la disponibilidad de un conjunto de artefactos (modelos y documentos).

En cada iteración se debe llevar a cabo los siguientes flujos de trabajo: Requisitos, Análisis, Diseño, Implementación y Pruebas.

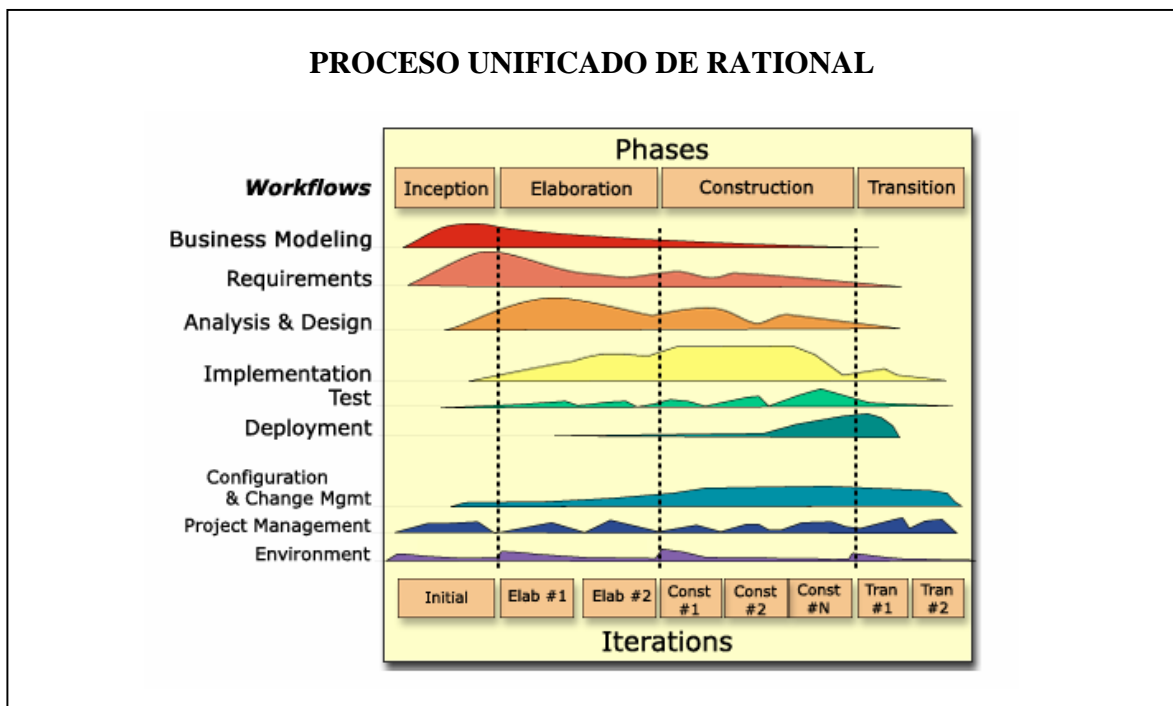


Figura 2.2 Proceso Unificado de Rational.²

² Internet

2.5 MICROSOFT SOLUTIONS FRAMEWORK (MSF)

Es un marco de trabajo que define una serie de principios, modelos y mejores prácticas para desarrollar, organizar e implementar Sistemas de Información Empresarial en un ambiente distribuido. Ofrece una guía para la organización del equipo de trabajo junto a una planificación, construcción y entrega de soluciones informáticas de alta calidad.

Se conforma de 2 modelos:

- Modelo de Equipos.
- Modelo de Proceso.

Apoyado en 3 disciplinas:

- Administración de Proyectos.
- Control de Riesgos.
- Manejo de Conocimientos.

2.5.1 Modelo de Equipos

- El equipo de trabajo mantiene una organización lineal y no basado en jerarquías.
- Existen roles independientes pero coordinados. Sus miembros tienen roles diferentes con su respectiva misión.
- Comparten la dirección del proyecto y el cumplimiento de las tareas basado en una comunicación continua.
- Siguiendo el modelo de procesos, tienen tareas que deben cumplir en el transcurso de todo el ciclo de vida del proyecto (fig. 2.3).

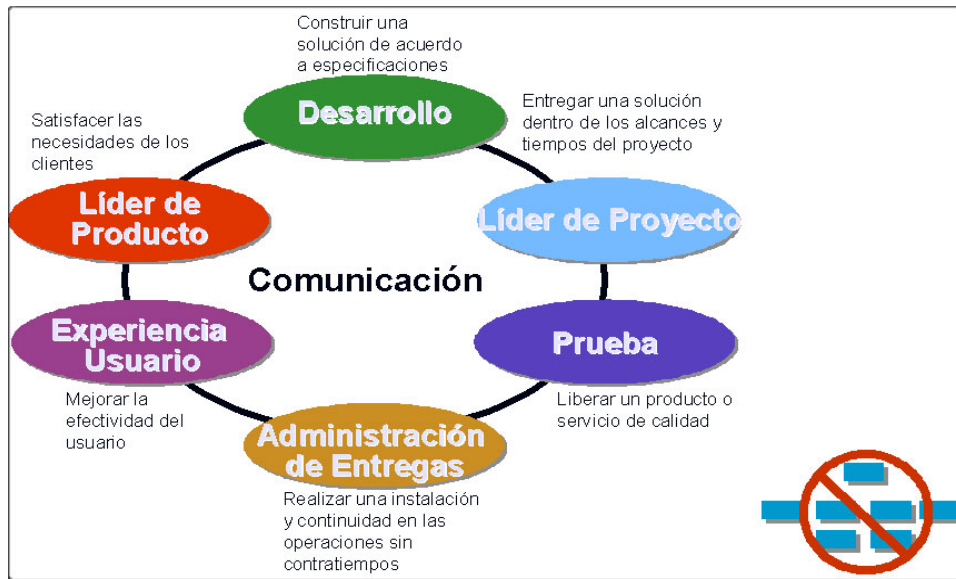


Figura 2.3 Modelo de Equipos MSF.³

El Equipo de Trabajo se encuentra conformado de la siguiente forma:

- Líder del Proyecto.
- Líder del Producto.
- Desarrollador / Arquitecto.
- Pruebas.
- Administrador de Entregas.
- Experiencia de Usuarios.

De acuerdo a las tareas que debe cumplir cada rol y al número de integrantes, en las figuras 2.4 y 2.5 podemos observar la integración de Pequeños y Grandes Equipos.

³ Internet

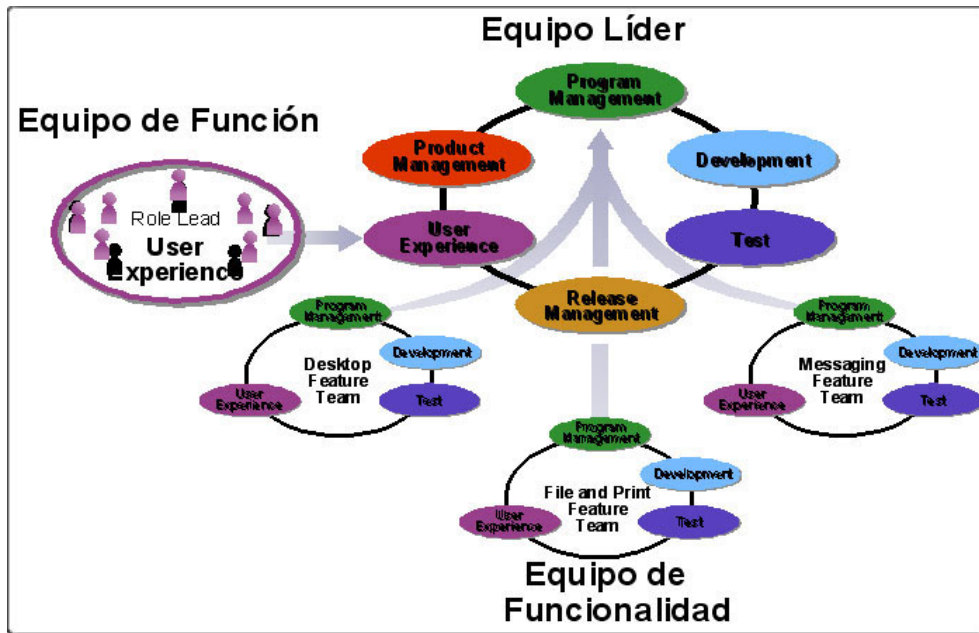


Figura 2.4 Equipo de trabajo para proyectos grandes.⁴

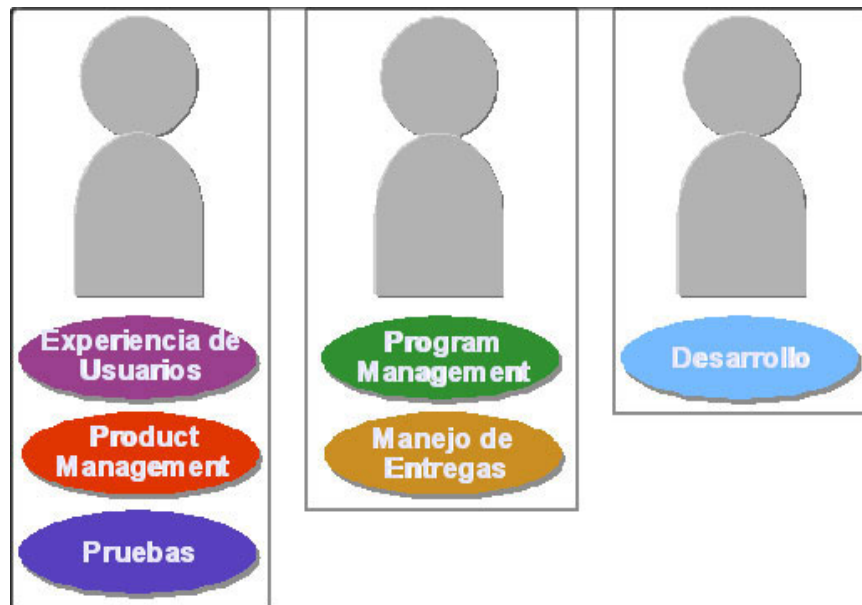


Figura 2.5 Equipo de trabajo para proyectos pequeños

⁴ Internet

A continuación en la fig. 2.6 se presenta las posibles combinaciones de funciones que puede cumplir cada rol en el Equipo MSF.

	Product Management	Program Management	Desarrollo	Pruebas	Experiencia de Usuarios	Manejo de Entregas
Product Management		N	N	P	P	U
Program Management	N		N	U	U	P
Desarrollo	N	N		N	N	N
Pruebas	P	U	N		P	P
Experiencia de Usuarios	P	U	N	P		U
Manejo de Entregas	U	P	N	P	U	

P Posible **U** Poco posible **N** No Recomendado

Figura 2.6 Combinaciones posibles de las funciones por Rol.⁵

2.5.2 Modelo de Proceso

El Modelo de Ciclo de Vida del MSF consta de 5 fases: Visión, Planificación, Desarrollo, Estabilización e Implantación. Al final de cada fase existen los “Entregables” que definen su culminación y el inicio de otra (fig. 2.7).

⁵ Internet

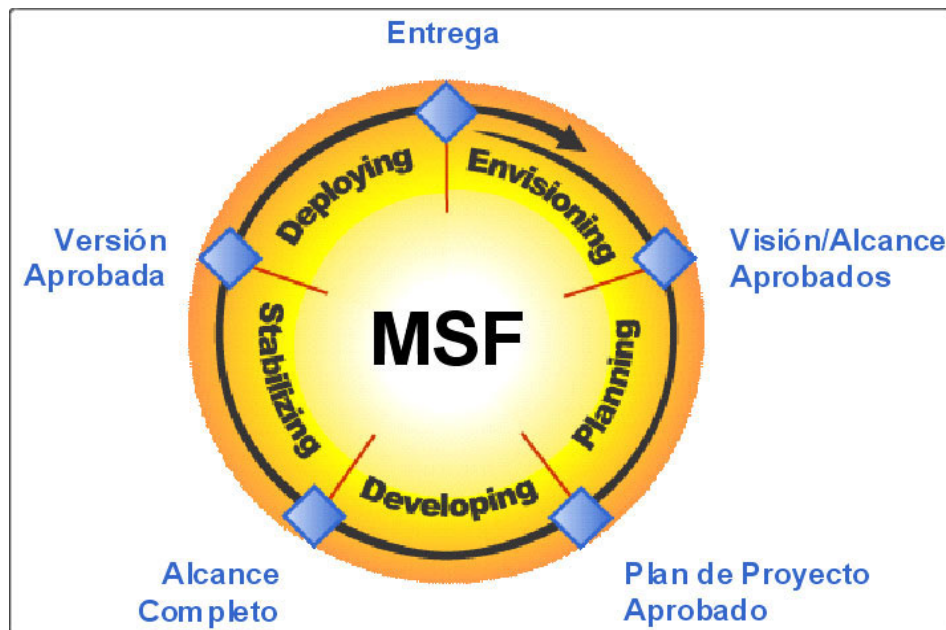


Figura 2.7 Fases del proceso MSF.⁶

Características del Modelo de Proceso MSF:

- El modelo de proceso de MSF es iterativo e incremental (fig. 2.8).
- Reduce los riesgos que puede correr el proyecto.
- Cada incremento se conforma de nuevas funcionalidades y mejoras de la iteración anterior.
- Cada fase se conforma de hitos principales (marcan la transición de una fase a otra) e intermedios (indican el progreso de una fase y dividen esfuerzos grandes en segmentos para un mejor control y cumplimiento de la fase).
- Cada rol del equipo tiene un entregable que debe cumplir en cada fase.

⁶ Internet

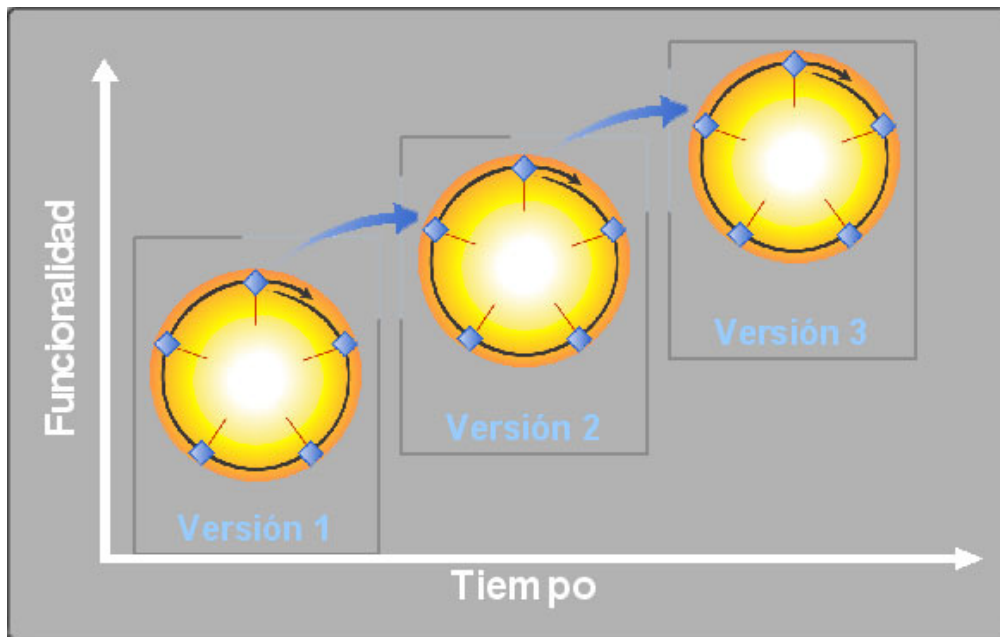


Figura 2.8 Proceso iterativo e incremental.⁷

⁷ Internet

2.5.2.1 Fase - Visión

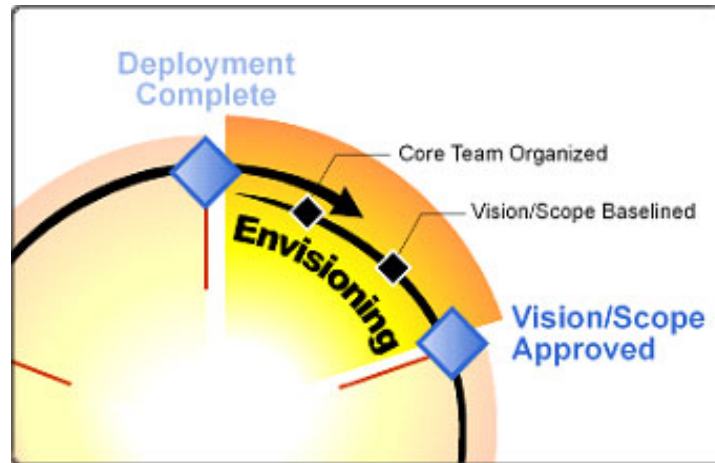


Figura 2.9 Fase - Visión.⁸

Objetivo.- Obtener una visión general de los propósitos del proyecto, su alcance y limitaciones, con la finalidad de cumplir con los requerimientos del usuario.

La Fase de Visión del Proyecto involucra:

- Identificar los requerimientos del usuario.
- Establecer bases para revisión y cambios en el proyecto.
- Definir el equipo de trabajo.
- Análisis de Riesgos.

Entregables de la Fase:

- Documento de Visión y Alcance, documento de Estructura de Proyecto y documento de Especificación de Riesgos.

⁸ Internet

2.5.2.2 Fase - Planificación

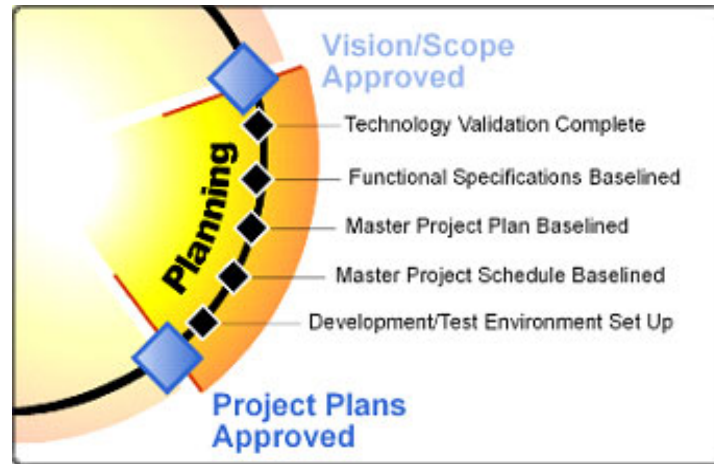


Figura 2.10 Fase - Planificación.⁹

Objetivo.- Definir la arquitectura y diseño de la solución.

La Fase de Planificación del Proyecto involucra:

- Definir con mayor exactitud los requerimientos funcionales del proyecto.
- Describir el diseño Conceptual, Lógico y Físico de la aplicación.
- Establecer con exactitud que se va a construir y como se lo hará.

Entregables de la Fase:

- Especificaciones Funcionales y Técnicas.
- Plan Maestro del Proyecto.
- Programación del Proyecto.

⁹ Internet

2.5.2.3 Fase - Desarrollo

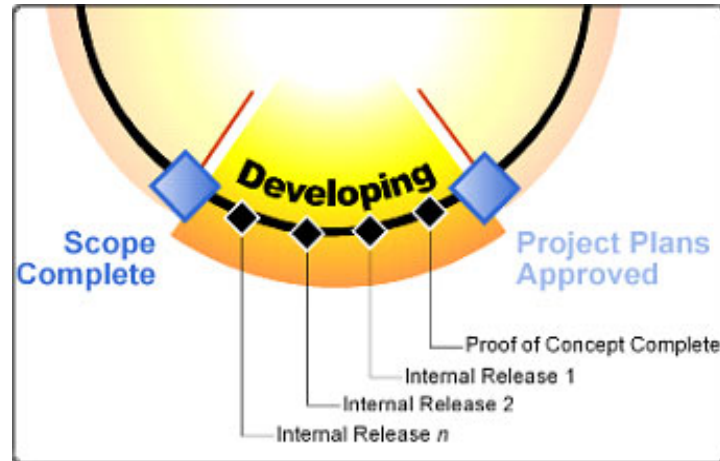


Figura 2.11 Fase - Desarrollo.¹⁰

Objetivo.- Desarrollar la aplicación propuesta.

La Fase de Desarrollo del Proyecto involucra:

- Definir la estructura para el desarrollo.
- Crear los scripts de desarrollo y proceso.
- Entregar el proyecto en forma incremental según la planificación.
- Alcanzar los criterios de calidad del producto.
- Realizar pruebas y liberaciones internas del producto.

Entregables de la Fase:

- Código fuente de la aplicación.
- Scripts de Instalación.

¹⁰ Internet

2.5.2.4 Fase - Estabilización

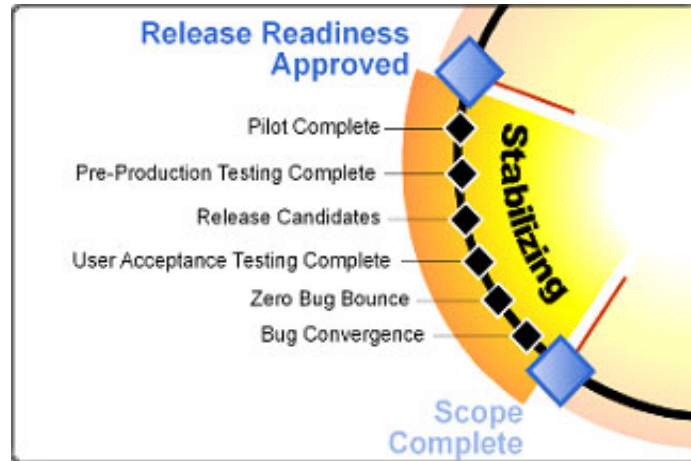


Figura 2.12 Fase - Estabilización.¹¹

Objetivo.- Mejorar la calidad del proyecto para alcanzar criterios de aceptación para la entrega final al usuario, en un ambiente de producción.

La Fase de Estabilización del Proyecto involucra:

- Mejorar la calidad del producto.
- Llevar la solución a un estado estable para la entrada a producción.
- Realizar varias pruebas antes de entregar el producto.

Entregables de la Fase:

- Código fuente y ejecutables.
- Manual de entrenamiento y ayuda para el usuario final.
- Reporte de Errores y Pruebas.

¹¹ Internet

2.5.2.5 Fase - Implantación

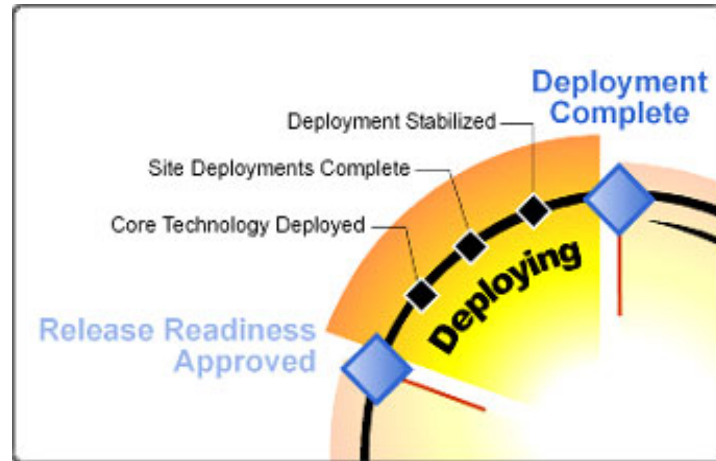


Figura 2.13 Fase - Implantación.¹²

Objetivo.- Instalar la aplicación en ambiente de producción.

La Fase de Implantación del Proyecto involucra:

- Facilitar la transferencia del producto desde el ambiente de desarrollo y pruebas al ambiente de producción.
- Asegurar la aprobación del usuario en la entrega del producto, de acuerdo a requerimientos funcionales.
- Capacitar al usuario para el uso del proyecto informático.

Entregables de la Fase:

- Reporte de cierre del proyecto.

¹² Internet

2.5.3 Disciplina de Administración de Proyectos

“El manejo de proyectos es la aplicación de conocimientos, destrezas, herramientas y técnicas a las actividades del proyecto para cumplir con sus requerimientos”.¹³

La Administración de Proyectos involucra:

- Planificación del proyecto.
- Gestión de costos.
- Gestión de recursos humanos.
- Gestión de alcances.
- Gestión del programa
- Gestión de riesgos.
- Control de Calidad.

2.5.4 Disciplina de Control de Riesgos

Riesgo.- Es una posible pérdida o daño que se puede ocasionar en el futuro.

El Control de Riesgos permite identificar los posibles riesgos que pueden ocasionarse en el transcurso del proyecto, y gestionar los que más amenacen al cumplimiento exitoso del mismo.

En la figura 2.14 se presenta las diferentes etapas que se cumplen en la Gestión de Riesgos de acuerdo al MSF e involucra:

¹³ Project Management Institute

- Distinguir los riesgos de los problemas existentes.
- Definir un proceso para identificar los riesgos, analizarlos y determinar planes de contingencia contra los mismos.
- Aprender de las experiencias de otros proyectos y aplicar al proyecto actual para minimizar el riesgo.
- Evaluar continuamente los riesgos y atacarlos en una forma proactiva.

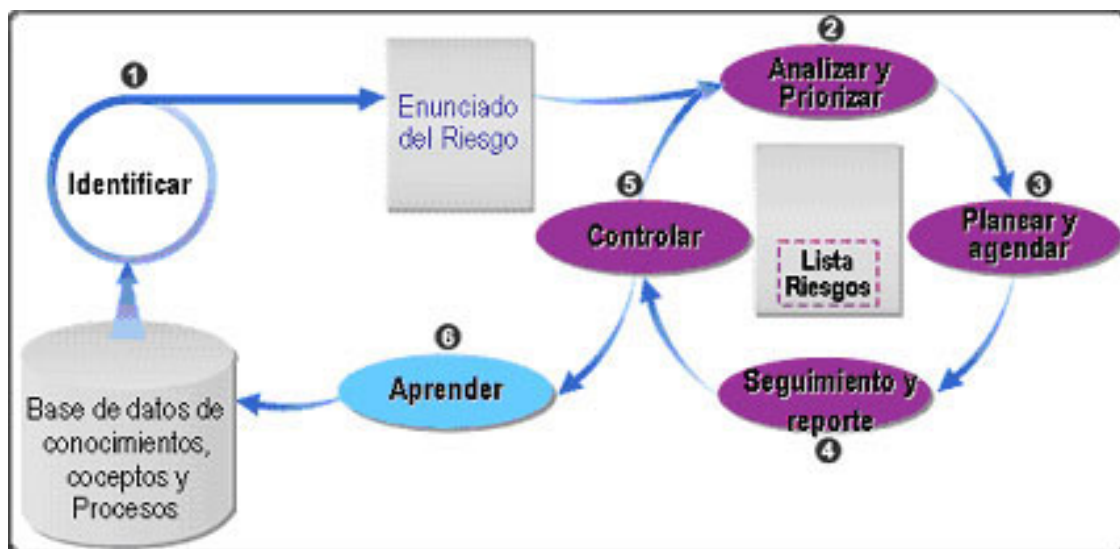


Figura 2.14 Gestión de Riesgos.¹⁴

2.5.5 Disciplina de Manejo de Conocimientos

Involucra el control de habilidades y competencias que el equipo de trabajo debe cumplir dependiendo del tipo de proyecto a desarrollar.

¹⁴ Internet

- Se debe conocer las habilidades y destrezas que tienen los integrantes del equipo.
- Preparar capacitación especializada al equipo si fuera necesaria.

2.6 LENGUAJE UNIFICADO DE MODELADO – UML

“Una empresa de software con éxito es aquella que produce de una manera consistente software de calidad que satisface las necesidades de los usuarios. Una empresa que puede desarrollar este software de forma predecible y puntual, con un uso eficiente y efectivo de recursos, tanto humanos como materiales, tiene un negocio sostenible.

El Lenguaje Unificado de Modelado es un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema con gran cantidad de software. UML proporciona una forma estándar de escribir los planos de un sistema, cubriendo tanto las cosas conceptuales, tales como procesos del negocio y funciones del sistema, como las cosas concretas, tales como las clases escritas en un lenguaje de programación específico, esquemas de bases de datos y componentes software reutilizables”¹⁵.

Los Lenguajes de Modelado Orientado a Objetos aparecieron entre los años 70 y 80 y a fines del año 1994 aparecieron aproximadamente 50 métodos Orientados a Objetos, pero quienes más destacaron fueron los de Booch, OOSE de Jacobson y OMT de Rumbaugh, que por sus grandes ventajas que presentaban fueron reconocidos como los tres principales métodos a nivel mundial.

En Octubre de 1994 los autores antes mencionados iniciaron esfuerzos para la creación de un nuevo lenguaje de modelado que permita unir sus mejores conocimientos en la orientación a

¹⁵ El Lenguaje Unificado de Modelado - Booch, Rumbaugh, Jacobson – Prólogo

objetos, es así como nació el Lenguaje Unificado de Modelado (UML). Entre las principales metas que buscaba este lenguaje podemos citar las siguientes:

- Modelar sistemas, desde el concepto hasta los artefactos ejecutables, utilizando técnicas orientadas a objetos.
- Cubrir las cuestiones relacionadas con el tamaño inherente a los sistemas complejos y críticos.
- Unificar los conocimientos de los autores para la obtención de estándares que apoyen a la creación del Lenguaje Unificado de Modelado.

Se publicaron varias versiones de UML que a continuación se detallan: Octubre 1995 UML 0.8, Junio 1996 UML 0.9, Enero 1997 UML 1.0, Julio 1997 UML 1.1 (Versión revisada para la estandarización), Noviembre 1997 UML 1.1 (Aceptada y adoptada por la OMG). A partir de la fecha OMG asumió el mantenimiento del UML. Junio de 1998 UML 1.2, Otoño de 1998 UML 1.3 (Versión con la cual se ha basado el desarrollo de la presente tesis).

“...Si construimos un sistema con la mirada de un desarrollador de bases de datos, probablemente nos centraremos en los modelos entidad-relación que trasladan el comportamiento a disparadores (triggers) y procedimientos almacenados. Si construimos un sistema con la mirada de un analista estructurado, probablemente se obtendrán modelos centrados en los algoritmos, con los datos fluyendo de proceso en proceso. Si construimos un sistema con la mirada de un desarrollador orientado a objetos, se obtendrá un sistema cuya arquitectura se centra en un mar de clases y los patrones de interacción que gobiernan cómo

trabajan juntas las clases. Cada visión del mundo conduce a un tipo de sistema diferente, con diferentes costes y beneficios¹⁶.

2.6.1 Arquitectura de un Sistema

La visualización, especificación, construcción y documentación de un sistema requiere que sea visto desde varias perspectivas. Usuarios tales como: desarrolladores, analistas, jefes de proyectos, entre otros, miran cada uno desde diferentes formas a los sistemas a lo largo del ciclo de vida del proyecto, es por eso que UML permite presentar a los sistemas desde 9 diagramas, los cuales se encuentran agrupados en 5 vistas (fig. 2.15):

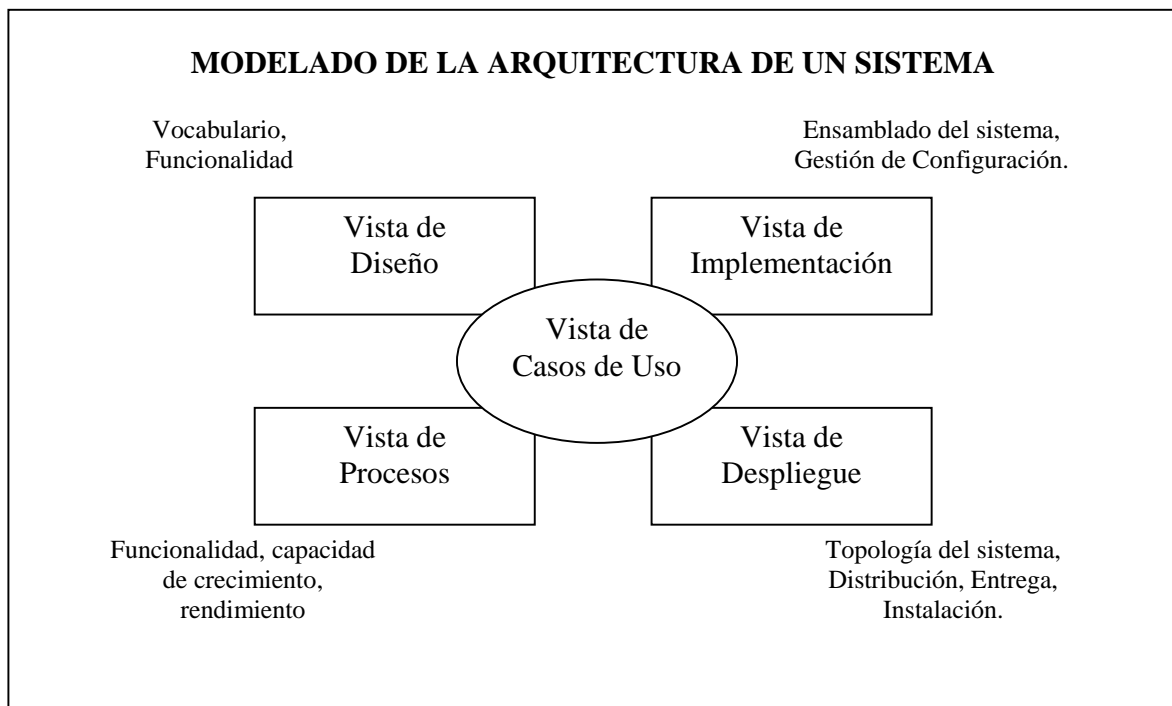


Figura 2.15 Modelo de la Arquitectura de un Sistema - RUP.

¹⁶ El Lenguaje Unificado de Modelado - Booch, Rumbaugh, Jacobson – Prólogo

- Vista de Casos de Uso.- Describe los requerimientos del sistema y su comportamiento desde el punto de vista de los usuarios finales, analistas y encargados de las pruebas.
 - Aspectos Estáticos (Diagrama de Casos de Uso).
 - Aspectos Dinámicos (Diagramas de Interacción, Diagrama de Estado y Diagramas de Actividades).
- Vista de Diseño.- Describe los requerimientos funcionales del sistema, entendiendo por ello los servicios que el sistema debería proporcionar a sus usuarios finales.
 - Aspectos Estáticos (Diagrama de Clases y Diagrama de Objetos).
 - Aspectos Dinámicos (Diagramas de Interacción, Diagrama de Estado y Diagramas de Actividades).
- Vista de Procesos.- Comprende los hilos y procesos que forman los mecanismos de sincronización y concurrencia del sistema. Cubre el funcionamiento, capacidad de crecimiento y rendimiento del sistema.
 - Aspectos Estáticos (Diagrama de Clases y Diagrama de Objetos).
 - Aspectos Dinámicos (Diagramas de Interacción, Diagrama de Estado y Diagramas de Actividades).
- Vista de Implementación.- Comprende los componentes y archivos que se utilizan para ensamblar el sistema.
 - Aspectos Estáticos (Diagrama de Componentes).
 - Aspectos Dinámicos (Diagramas de Interacción, Diagrama de Estado y Diagramas de Actividades)
- Vista de Despliegue.- Describe los nodos que forman la topología hardware sobre la que se ejecutará el sistema.

- Aspectos Estáticos (Diagrama de Despliegue).
- Aspectos Dinámicos (Diagramas de Interacción, Diagrama de Estado y Diagramas de Actividades).

Un sistema puede ser también observado desde 2 tipos de vistas: estática y dinámica.

- Vista Estática:

- Diagrama de Clases.- Presenta un conjunto de clases, interfaces y colaboraciones, y las relaciones entre ellas. Son los diagramas más comunes al modelar sistemas orientados a objetos.
- Diagrama de Objetos.- Presenta un conjunto de objetos y sus relaciones. Describe estructura de datos.
- Diagrama de Componentes.- Presenta un conjunto de componentes y sus relaciones. Un componente normalmente corresponde con una o más clases, interfaces o colaboraciones.
- Diagrama de Despliegue.- Presenta un conjunto de nodos y sus relaciones. Un nodo normalmente incluye uno o más componentes.

- Vista Dinámica:

- Diagrama de Casos de Uso.- Presenta un conjunto de casos de uso y sus actores. Son especialmente importantes para organizar y modelar el comportamiento de un sistema.
- Diagrama de Secuencia (Diagrama de Interacción).- Presenta un conjunto de objetos y los mensajes enviados y recibidos por ellos. Resalta la ordenación temporal de los mensajes.

- Diagrama de Colaboración (Diagrama de Interacción).- Presenta un conjunto de objetos, enlaces entre esos objetos y mensajes enviados y recibidos por esos objetos. Resalta la organización estructural de los objetos que envían y reciben mensajes.
- Diagrama de Estado.- Presenta una máquina de estados, constituida por estados, transiciones, eventos y actividades. Son importantes para modelar el comportamiento de una interfaz, una clase o una colaboración. Resalta el comportamiento dirigido por eventos de un objeto, lo que es muy útil para sistemas reactivos.
- Diagrama de Actividades.- Muestra el flujo de actividades, el flujo secuencial o ramificado de actividades, y los objetos que actúan y sobre los que actúan. Sirven para modelar la función de un sistema, así como para resaltar el flujo de control entre objetos.

La arquitectura del software no solo tiene que ver con la estructura y el comportamiento, sino también con su uso, funcionalidad, rendimiento, capacidad de adaptación, reutilización, capacidad de ser comprendido, restricciones económicas y tecnológicas.

2.7 MICROSOFT .NET

.NET representa un conjunto de tecnologías y conceptos que conforman una plataforma en la cual se puede desarrollar aplicaciones distribuidas alrededor del mundo.

.Net Framework ofrece 2 cosas primordiales:

- El entorno del motor de ejecución básico.- Es similar al sistema operativo ya que ofrece una capa intermedia entre los programas y resto del sistema, con lo que provee de servicios a las aplicaciones y simplifica el acceso a la funcionalidad de las capas intermedias.
- Un conjunto de clases fundamentales.- Ofrecen una gran funcionalidad, ya que envuelven y abstraen a tecnologías como los protocolos de Internet, acceso al sistema de archivos, manejo de XML, etc.

2.7.1 Ventajas de Microsoft .NET

- El lenguaje que se utilice es menos relevante en comparación con épocas anteriores. Para el desarrollo de una aplicación .NET se puede seleccionar cualquier lenguaje del IDE de Microsoft Visual Studio .NET, adaptándose así a sus necesidades y permitiendo cumplir con sus objetivos.
- La Plataforma .NET permite la creación de aplicaciones basadas en XML, lo cual facilita compartir información desde cualquier plataforma o dispositivo.
- Microsoft .NET permite crear aplicaciones sin pensar en 2 infraestructuras separadas: una aplicación web y otra aplicación de escritorio.
- Se puede crear aplicaciones basadas en Servicios Web XML, los cuales exponen sus funciones a través de Internet o Intranet, permitiendo así consumir sus servicios desde cualquier lugar del mundo.

2.8 APLICACIONES DISTRIBUIDAS

”Un sistema distribuido es aquel en que sus componentes localizados en computadoras, conectados en red, comunican y coordinan sus acciones únicamente mediante el paso de mensajes”.¹⁷

Características de las Aplicaciones Distribuidas:

- Heterogeneidad en sus componentes (variedad y diferencia en sus redes, hardware, sistemas operativos, lenguaje de programación, implementación de diferentes desarrolladores).
- Permiten añadir o reemplazar componentes.
- Alta seguridad y escalabilidad.
- Adaptación al incremento de usuarios en la aplicación.
- Buen tratamiento a los fallos que se produzcan.
- Concurrencia de sus componentes (cada objeto que presente un recurso compartido en un sistema distribuido debe responsabilizarse de garantizar que opera correctamente en un entorno concurrente).

¹⁷ Sistema Distribuidos – George Coulouris – Pág. 1

CAPITULO III

FASE I: VISIÓN

Siendo la primera fase del Ciclo de Vida de un proyecto y de acuerdo al Microsoft Soluciones Framework, se establece en forma general los objetivos del proyecto, su alcance y limitaciones, para dejar constancia del inicio del proyecto.

3.1 OBJETIVOS DEL PROYECTO

Tabla 3.1 Objetivos del Proyecto

OBJETIVO PRINCIPAL	
Desarrollar el Sistema Computarizado “AeroFlex” que permita Gestionar y Controlar la Emisión de la Orden de Vuelo en la Empresa de Transporte Aéreo TAME, sobre un ambiente Windows con Plataforma .NET	
OBJETIVOS DETALLADOS	
1	Emitir la Programación Semanal de las Aeronaves en la Empresa.
2	Emitir la Programación de Itinerarios de Vuelo de la Tripulación.
3	Crear la Malla Curricular de Entrenamiento.
4	Planificar el Entrenamiento Anual de la Tripulación.
5	Asignar vuelos a los Tripulantes para su posterior cumplimiento.
6	Generar la Orden de Vuelo.
7	Registrar los Históricos de la Orden de Vuelo
8	Integrar los procesos de Entrenamiento de Tripulación e Itinerarios de Vuelo.

3.2 AREAS IMPACTADAS

Tabla 3.2 Descripción de Perfiles en el Proyecto.

DESCRIPCIÓN DE LOS PERFILES INVOLUCRADOS		
Área	Perfil	Descripción
Centro de Control Operacional (CCO)	Tripulante	Es la persona a la cual se programará vuelos a cumplir, entrenamiento y otras actividades en la Empresa.
	Programador de Aeronaves	En coordinación con reservaciones podrá generar los itinerarios de vuelo de las aeronaves.
	Programador de Tripulantes	En coordinación con el programador de aeronaves y entrenamiento de tripulantes, podrá generar los itinerarios de vuelo de la tripulación.
Entrenamiento de Tripulantes	Programador de Entrenamiento	Presenta la planificación anual del entrenamiento que debe cumplir los tripulantes.
Gerencia Comercial (Reservaciones)	Sistema de Reservaciones	Es la representación de los datos que serán obtenidos de la base de datos de reservaciones, para la programación de itinerarios.

3.3 ALCANCE

Tabla 3.3 Matriz de Necesidades.

MATRIZ DE NECESIDADES	
Área / Perfil : CCO – TRIPULANTE	
Código	Necesidad
N01	Visualizar los itinerarios de vuelo a cumplir.
N02	Contar con el calendario de entrenamiento asignado.
Área / Perfil : CCO – PROGRAMADOR DE AERONAVES	
N03	Visualizar los vuelos programados en la Gerencia Comercial (Reservaciones)
N04	Contar con una interfaz gráfica para realizar la programación de itinerarios de vuelos de las aeronaves.
N05	El sistema deberá permitir programar los vuelos fuera de itinerario
Área / Perfil : CCO – PROGRAMADOR DE TRIPULANTES	
N06	La programación de tripulantes debe integrarse con la programación de aeronaves.
N07	Los cambios en las operaciones debe ser reflejado en la programación de tripulantes
N08	El sistema deberá permitir crear los Itinerarios de Vuelo de la Tripulación y la Orden de Vuelo diaria.
Área / Perfil: ENT. DE TRIPUL. – PROGRAMADOR DE ENTRENAMIENTO	
N09	La programación de entrenamiento de tripulantes deberá integrarse con los itinerarios de vuelo.

N10	El sistema deberá permitir registrar la planificación anual de entrenamiento.
N11	Deberá permitir la asignación de entrenamiento a cada tripulante.
Área / Perfil : RESERVACIONES – BASE DATOS DE RESERVACIONES	
N12	Presentar al Sistema de Operaciones (Control y Gestión de la Orden de Vuelo), los vuelos que deberá cumplir la Empresa diariamente.
N13	Los vuelos no podrán ser cambiados, deberán ser tan solo de lectura.
Área / Perfil : NA – ADMINISTRADOR DEL SISTEMA	
N14	Registrar a los usuarios que ingresarán al sistema.
N15	Validar el acceso al sistema por parte de los usuarios y asignar los permisos correspondientes dependiendo de su perfil.

3.4 DESCRIPCIÓN DEL PROCESO

En la emisión de la Orden de Vuelo se han identificados procesos que están en constante interacción y que pertenecen a diferentes áreas tales como Gerencia Comercial (Reservación de Vuelos), Departamento de Entrenamiento y Centro de Control Operacional.

La ejecución de los procesos (fig. 3.1) antes mencionados inicia desde el registro de los vuelos que debe cumplir la Empresa diariamente y que han sido informados a los futuros pasajeros. A estos vuelos se deberá asignar las distintas aeronaves disponibles en cada día, tomando en cuenta que la Gerencia de Mantenimiento es quien debe emitir esta disponibilidad y de acuerdo al mantenimiento planificado o no.

Teniendo los vuelos y las aeronaves que deben cumplir con su ruta a la hora planificada con anticipación, en el Centro de Control Operacional se crea los “Bloques de Vuelos” en constante interacción con el Programador de Tripulantes. Estos bloques de vuelos permiten que tanto la aeronave como los tripulantes a bordo puedan cumplir con las rutas en una secuencia lógica de origen – destino y horas establecidas.

Es importante tomar en cuenta que para emitir la disponibilidad de los tripulantes, hay diferentes factores que afectan a esta información tales como: vacaciones, permisos, entrenamiento, certificados médico, licencia, entre otros. De acuerdo a estos datos se podrá tener una información verás y con la cual se podrá formar los Itinerarios de Vuelo de la Tripulación.

Por último, para la emisión de la Orden de Vuelo se toma en cuenta los Itinerarios de Vuelo Programados que al actualizarlos diariamente y con otros datos adicionales permiten crear la Orden de Vuelo Programada. A este documento se realiza varias actualizaciones permanentes y de acuerdo a la ejecución de la operación diaria.

La Orden de Vuelo Cumplida se emite diariamente habiendo con anticipación actualizado los posibles cambios en la operación y su previa autorización.

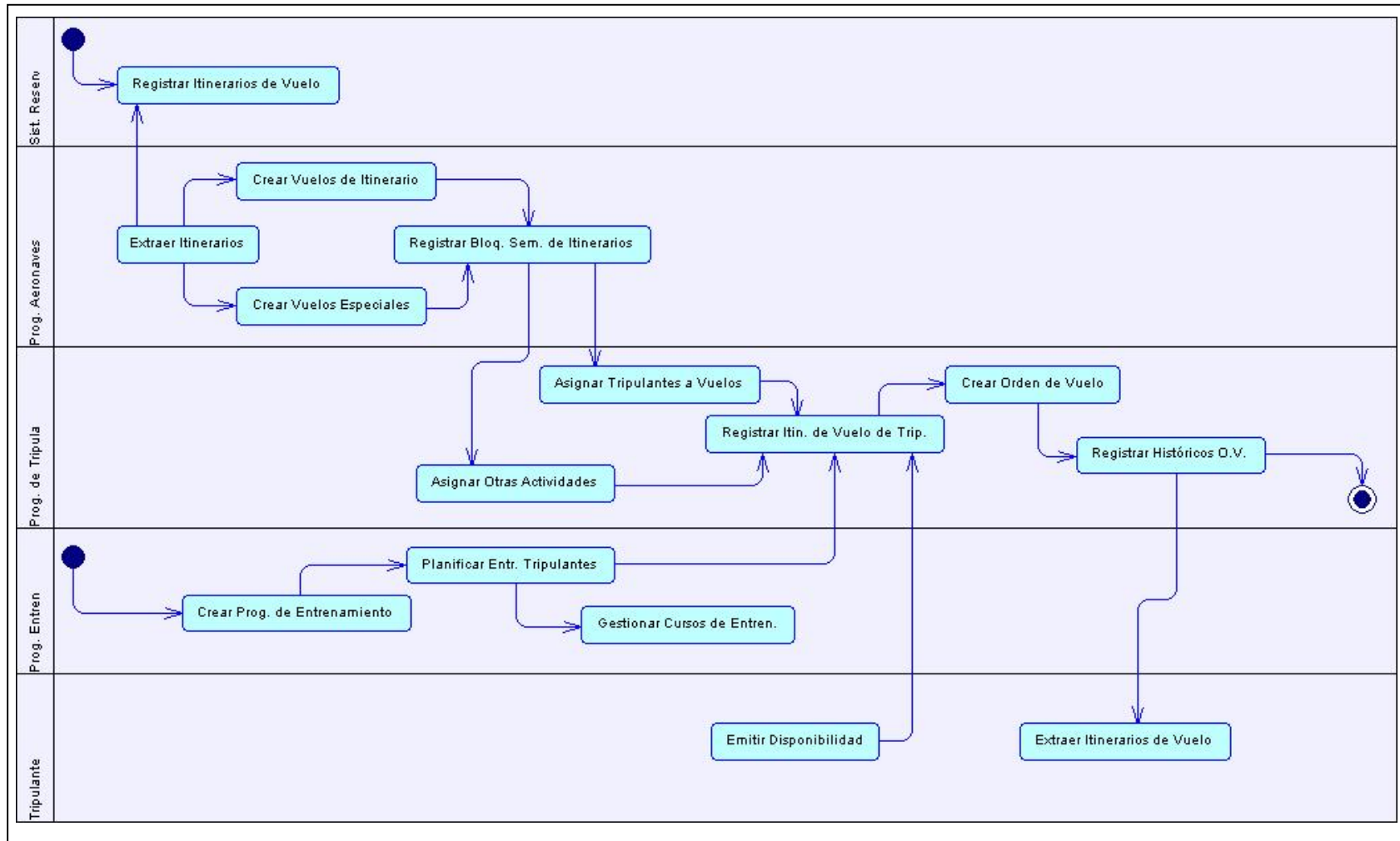


Figura 3.1 Procesos Integrados para la Emisión de la Orden de Vuelo.

3.5 MODELO DE LA SOLUCIÓN PROPUESTA

Para la automatización de la Orden de Vuelo se agrupará en 4 Módulos que permitirán claramente identificar las funciones a cumplir por el personal y departamentos involucrados. Estos módulos están plenamente enlazados a los procesos que se encuentran aprobados por las autoridades de la Empresa debido a la obtención de la Certificación ISO 9001.

3.5.1 Módulo de Seguridad

El Sistema AeroFlex contará con un módulo de seguridad para el control de acceso al sistema y la asignación de perfiles de usuario, por medio de los cuales se habilitará las opciones correspondientes según el grupo al que pertenezca.

3.5.2 Módulo de Programación de Aeronaves

El presente módulo extraer de la base de datos de Reservaciones los vuelos que son presentados a los pasajeros con sus respectivas rutas, horas de salida y que deberán ser cumplidos por la Empresa. A estos vuelos se asigna las aeronaves disponibles y de acuerdo a los horarios y tripulación. Tomando en cuenta que además se podrá programar vuelos fuera de itinerario.

Al existir cambios en las operaciones se pueden modificar las aeronaves o los Bloques de Vuelos a cumplir y en coordinación con la Programación de Tripulantes.

3.5.3 Módulo de Programación de Tripulantes

De acuerdo a los Bloques de Vuelos creados para el cumplimiento de las operaciones en la Empresa, el Programador de Tripulantes asigna a los vuelos la tripulación disponible según el

equipo al que este certificado, además siguiendo las normas que establece las RDAC. Esta programación debe estar integrada con el entrenamiento de tripulantes.

3.5.4 Módulo de Entrenamiento de Tripulantes

Para el cumplimiento de las operaciones el tripulante que es asignado a un vuelo debe siempre encontrarse certificado para esta tarea, por lo tanto la DAC dispone la forma como el tripulante deberá cumplir el entrenamiento y el cual tendrá que aprobarlo.

El presente módulo permite programar el entrenamiento que debe cumplir los tripulantes para poder continuar con las operaciones y se lo realiza en coordinación con la programación de tripulantes.

3.6 QUE NO CONTEMPLA LA VISIÓN DE LA SOLUCIÓN

- Programación automática de Itinerarios de Vuelo.
- Planificación de entrenamiento a un nivel de materias.
- Registro del cumplimiento del entrenamiento de la tripulación.
- Interacción con el sistema de la Gerencia de Mantenimiento lo que implica que no se reflejará la disponibilidad de aeronaves.

3.7 COSTOS DEL PROYECTO

A continuación en la fig. 3.2 se presenta los costos involucrados en el proyecto de acuerdo a las fases y tareas cumplidas:

Nombre de tarea	Costo total
PROYECTO INFORMATICO AEROFLEX	\$17,440.00
VISIÓN	\$3,100.00
Reunión con el equipo de trabajo	\$120.00
Crear cronograma de proyecto (preliminar)	\$60.00
Realizar relevamiento de información	\$600.00
Definir necesidades y alcance del proyecto	\$300.00
Elaborar documento de vision	\$180.00
Aprobar documento	\$0.00
PLANIFICACIÓN	\$2,100.00
Crear prototipo de la aplicación	\$300.00
Crear Plan Maestro	\$480.00
Definir plan de pruebas	\$120.00
Definir plan de riesgos	\$120.00
Definir plan de entrenamiento	\$120.00
Definir plan de implantación	\$120.00
Crear cronograma de proyecto definitivo	\$120.00
Elaborar documento de especificaciones funcionales	\$900.00
Elaborar documento de especificaciones técnicas	\$300.00
Aprobar documentos	\$0.00
DESARROLLO	\$9,780.00
Crear base de datos de la aplicación	\$300.00
Codificar la capa de acceso a datos	\$900.00
Codificar la capa de lógica de negocios	\$2,100.00
Codificar los servicios web	\$900.00
Codificar la capa de presentación	\$3,600.00
Coordinar integración con otros sistemas	\$900.00
Migrar datos de otros sistemas	\$900.00
Crear manual de instalación	\$180.00
Aprobar documentación	\$0.00
Aprobar el paso al ambiente de pruebas	\$0.00
ESTABILIZACIÓN	\$2,100.00
Instalar el ambiente de pruebas	\$120.00
Ejecutar plan de pruebas	\$900.00
Ejecutar plan de implantación	\$300.00
Crear manual de usuario	\$180.00
Ejecutar plan de entrenamiento	\$600.00
Aprobar el paso al ambiente de producción	\$0.00
IMPLANTACIÓN	\$360.00
Instalar el ambiente de producción	\$360.00
Firmar carta de cierre de proyecto	\$0.00

Figura 3.2 Costos por Tarea

En la fig. 3.3 se presenta los costos agrupados por recursos y según las tareas del proyecto:

Nombre del recurso	Costo
Jefe de Proyecto	\$15,600.00
<i>Reunión con el equipo de trabajo</i>	<i>\$120.00</i>
<i>Crear cronograma de proyecto (preliminar)</i>	<i>\$00.00</i>
<i>Realizar relevamiento de información</i>	<i>\$000.00</i>
<i>Definir necesidades y alcance del proyecto</i>	<i>\$300.00</i>
<i>Elaborar documento de visión</i>	<i>\$180.00</i>
<i>Aprobar documento</i>	<i>\$0.00</i>
<i>Crear prototipo de la aplicación</i>	<i>\$300.00</i>
<i>Definir plan de pruebas</i>	<i>\$120.00</i>
<i>Definir plan de riesgos</i>	<i>\$120.00</i>
<i>Definir plan de entrenamiento</i>	<i>\$120.00</i>
<i>Definir plan de implementación</i>	<i>\$120.00</i>
<i>Crear cronograma de proyecto definitivo</i>	<i>\$120.00</i>
<i>Elaborar documento de especificaciones funcionales</i>	<i>\$000.00</i>
<i>Elaborar documento de especificaciones técnicas</i>	<i>\$300.00</i>
<i>Aprobar documentos</i>	<i>\$0.00</i>
<i>Crear base de datos de la aplicación</i>	<i>\$300.00</i>
<i>Codificar la capa de acceso a datos</i>	<i>\$000.00</i>
<i>Codificar la capa de lógica de negocios</i>	<i>\$2,100.00</i>
<i>Codificar los servicios web</i>	<i>\$000.00</i>
<i>Codificar la capa de presentación</i>	<i>\$3,000.00</i>
<i>Coordinar integración con otros sistemas</i>	<i>\$000.00</i>
<i>Migrar datos de otros sistemas</i>	<i>\$000.00</i>
<i>Crear manual de instalación</i>	<i>\$180.00</i>
<i>Aprobar documentación</i>	<i>\$0.00</i>
<i>Aprobar el paso al ambiente de pruebas</i>	<i>\$0.00</i>
<i>Instalar el ambiente de pruebas</i>	<i>\$120.00</i>
<i>Ejecutar plan de pruebas</i>	<i>\$000.00</i>
<i>Ejecutar plan de implementación</i>	<i>\$300.00</i>
<i>Crear manual de usuario</i>	<i>\$180.00</i>
<i>Ejecutar plan de entrenamiento</i>	<i>\$000.00</i>
<i>Aprobar el paso al ambiente de producción</i>	<i>\$0.00</i>
<i>Instalar el ambiente de producción</i>	<i>\$300.00</i>
<i>Firmar carta de cierre de proyecto</i>	<i>\$0.00</i>
Software	\$1,300.00
<i>VISIÓN</i>	<i>\$1,300.00</i>
Utiles de Oficina	\$300.00
<i>VISIÓN</i>	<i>\$300.00</i>
Libros	\$240.00
<i>VISIÓN</i>	<i>\$240.00</i>

Figura 3.3 Costos por Recurso

3.8 PLANIFICACIÓN DEL PROYECTO

A continuación se presenta el cronograma de actividades que involucra el desarrollo del Sistema Computarizado “AeroFlex” (fig. 3.4):

Nombre de tarea	Duración	Comienzo	Fin
PROYECTO INFORMATICO AEROFLEX	260 días	lun 02/01/06	vie 29/12/06
VISIÓN	21 días	lun 02/01/06	lun 30/01/06
Reunión con el equipo de trabajo	2 días	lun 02/01/06	mar 03/01/06
Crear cronograma de proyecto (preliminar)	1 día	mié 04/01/06	mié 04/01/06
Realizar relevamiento de información	10 días	jue 05/01/06	mié 18/01/06
Definir necesidades y alcance del proyecto	5 días	jue 19/01/06	mié 25/01/06
Elaborar documento de vision	3 días	jue 26/01/06	lun 30/01/06
Aprobar documento	0 días	lun 30/01/06	lun 30/01/06
PLANIFICACIÓN	35 días	mar 31/01/06	lun 20/03/06
Crear prototipo de la aplicación	5 días	mar 31/01/06	lun 06/02/06
Crear Plan Maestro	8 días	mar 07/02/06	jue 16/02/06
Definir plan de pruebas	2 días	mar 07/02/06	mié 08/02/06
Definir plan de riesgos	2 días	jue 09/02/06	vie 10/02/06
Definir plan de entrenamiento	2 días	lun 13/02/06	mar 14/02/06
Definir plan de implantación	2 días	mié 15/02/06	jue 16/02/06
Crear cronograma de proyecto definitivo	2 días	vie 17/02/06	lun 20/02/06
Elaborar documento de especificaciones funcionales	15 días	mar 21/02/06	lun 13/03/06
Elaborar documento de especificaciones técnicas	5 días	mar 14/03/06	lun 20/03/06
Aprobar documentos	0 días	lun 20/03/06	lun 20/03/06
DESARROLLO	163 días	mar 21/03/06	jue 02/11/06
Crear base de datos de la aplicación	5 días	mar 21/03/06	lun 27/03/06
Codificar la capa de acceso a datos	15 días	mar 28/03/06	lun 17/04/06
Codificar la capa de lógica de negocios	35 días	mar 18/04/06	lun 05/06/06
Codificar los servicios web	15 días	mar 06/06/06	lun 26/06/06
Codificar la capa de presentación	60 días	mar 27/06/06	lun 18/09/06
Coordinar integración con otros sistemas	15 días	mar 19/09/06	lun 09/10/06
Migrar datos de otros sistemas	15 días	mar 10/10/06	lun 30/10/06
Crear manual de instalación	3 días	mar 31/10/06	jue 02/11/06
Aprobar documentación	0 días	jue 02/11/06	jue 02/11/06
Aprobar el paso al ambiente de pruebas	0 días	jue 02/11/06	jue 02/11/06
ESTABILIZACIÓN	35 días	vie 03/11/06	jue 21/12/06
Instalar el ambiente de pruebas	2 días	vie 03/11/06	lun 06/11/06
Ejecutar plan de pruebas	15 días	mar 07/11/06	lun 27/11/06
Ejecutar plan de implantación	5 días	mar 28/11/06	lun 04/12/06
Crear manual de usuario	3 días	mar 05/12/06	jue 07/12/06
Ejecutar plan de entrenamiento	10 días	vie 08/12/06	jue 21/12/06
Aprobar el paso al ambiente de producción	0 días	jue 21/12/06	jue 21/12/06
IMPLANTACIÓN	6 días	vie 22/12/06	vie 29/12/06
Instalar el ambiente de producción	6 días	vie 22/12/06	vie 29/12/06
Firmar carta de cierre de proyecto	0 días	vie 29/12/06	vie 29/12/06

Figura 3.4 Planificación del Proyecto

3.9 EQUIPO DEL PROYECTO

Tabla 3.4 Equipo del Proyecto.

ROL	CARACTERÍSTICAS	PERSONAL
<p>Líder de Proyecto</p>	<ul style="list-style-type: none"> • Entrega la solución dentro de lo planificado. • Gestiona el proyecto (Servicios administrativos, aseguramiento de procesos, arquitectura de la solución). • Coordina el trabajo entre los integrantes del equipo. • Es el enlace entre el área de negocio y el equipo del proyecto. • Verifica que la implementación del sistema este de acuerdo a los requerimientos. • Informa permanentemente los avances del proyecto. 	<p>Lenín Echeverría</p> <p>leninmau@hotmail.com</p> <p>09 5 201 268</p>
<p>Líder del Producto</p>	<ul style="list-style-type: none"> • Es el representante del Negocio dentro del equipo de trabajo. • Coordina con el equipo para estructurar la visión y los alcances del proyecto. • Define y aprueba las necesidades del negocio. Tiene la visión del negocio. 	<p>Elizabeth Gabela</p> <p>egabela@tame.com.ec</p>

	<ul style="list-style-type: none"> • Promociona el proyecto en la Empresa. 	
Desarrollador / Arquitecto	<ul style="list-style-type: none"> • Construye la aplicación de acuerdo a los requerimientos. • Encargado de la infraestructura. • Responsable de la arquitectura de implementación y diseño del sistema. 	<p>Lenín Echeverría</p> <p>leninmau@hotmail.com</p> <p>09 5 201 268</p>
Experiencia de Usuario	<ul style="list-style-type: none"> • Respaldo a los usuarios. • Coordina el diseño de la interfaz de usuario en la aplicación. • Trabajo en conjunto con el equipo para la implementación de los requerimientos. 	<p>Elizabeth Gabela</p> <p>egabela@tame.com.ec</p> <p>Hanns Heyman</p> <p>hheyman@tame.com.ec</p> <p>Jenny Pavón</p> <p>jpavon@tame.com.ec</p>
Pruebas	<ul style="list-style-type: none"> • Encargado del plan de pruebas. • Presenta un informe del plan de pruebas ejecutado. 	<p>Elizabeth Gabela</p> <p>egabela@tame.com.ec</p>
Administración de Entregables	<ul style="list-style-type: none"> • Responsable que el producto pase a producción cumpliendo estándares de la Empresa. • Verifica que la aplicación no genere conflictos con otros sistemas existentes en la Empresa. • Responsable del soporte y la operación del producto. 	<p>Lenín Echeverría</p> <p>leninmau@hotmail.com</p> <p>09 5 201 268</p>

	<ul style="list-style-type: none"> • Define los posibles riesgos y especifica los procedimientos de contingencia. • Establece los procedimientos para la operatividad y mantenimiento de producto. 	
--	--	--

3.10 RIESGOS DEL PROYECTO

Tabla 3.5 Riesgos del Proyecto.

ENUNCIADO DE RIESGO	PROB. DE OCURR.	IMPACTO	PLAN DE MITIGAC.	PLAN DE CONTINGENCIA	RESPONSA-BILIDAD
Cambios en el alcance del proyecto	5	ALTO	Analizar los requerimientos y emitir controles de cambios	Analizar impacto en tiempo y costo y si se autoriza se deberá añadir al cronograma	Lenín Echeverría Elizabeth Gabela
No se registra el entrenamiento de los tripulantes	3	BAJO	Se deberá actualizar permanentemente en el sistema el entrenamiento de los tripulantes	Publicar la Orden de Vuelo sin tomar en cuenta el entrenamiento de los tripulantes	Jenny Pavón Elizabeth Gabela
Los Itinerarios de Vuelo Programados no están actualizados	5	ALTO	Se deberá actualizar permanentemente los itinerarios de vuelo	Verificar la Orden de Vuelo Programada si es como se va o no a cumplir la operación	Elizabeth Gabela Hanns Heyman
La Orden de Vuelo programada no es la correcta	3	ALTO	Verificar que los itinerarios de vuelo han sido actualizados	Volver a verificar el proceso para actualizar los datos	Elizabeth Gabela
La Orden de Vuelo	3	ALTO	Los cambios en las operaciones	Cuando se ingrese la	Elizabeth Gabela

cumplida no es la correcta			deberán ser registrados todos los días a primera hora en el sistema.	Bitácora de Vuelo se encontrará el error y se deberá actualizar la Orden de Vuelo Cumplida	Patricio Sandoval
----------------------------	--	--	--	--	-------------------

CAPITULO IV

FASE II: PLANIFICACIÓN

En la Fase de Planificación del Proyecto se presenta la Arquitectura y Diseño de la solución propuesta, esto con la finalidad de dejar establecidas las bases arquitectónicas para el buen desarrollo y funcionamiento de la aplicación.

4.1 ESPECIFICACIONES FUNCIONALES

4.1.1 Requerimientos Funcionales

Tabla 4.1 Requerimientos Funcionales - Tripulante.

Área / Perfil: CCO – TRIPULANTE			
Nec.	Ref.	Descripción	Módulo
N01	RF1.0	Para un mejor cumplimiento de las operaciones, se requiere contar con la entrega oportuna de los itinerarios de vuelos a cumplir para cada tripulante.	Programación de Tripulantes
N01	RF2.0	Los itinerarios de vuelo deben presentar otras actividades que se puede cumplir en la Empresa, tales como entrenamiento, vacaciones, certificado médico, etc.	Programación de Tripulantes
N02	RF3.0	Se requiere contar con el calendario anual de entrenamiento de cada tripulante.	Entrenamiento de Tripulantes

Tabla 4.2 Requerimientos Funcionales – Programador de Aeronaves.

Área / Perfil: CCO – PROGRAMADOR DE AERONAVES			
Nec.	Ref.	Descripción	Módulo
N03	RF4.0	Se seleccionará el día que se vaya a programar las aeronaves y por medio del cual el sistema presentará la lista de vuelos a cumplir.	Programación de Aeronaves
N03	RF5.0	Para programar los vuelos el sistema permitirá seleccionar el equipo y las aeronaves pertenecientes al mismo. A partir de este podrá asignar los vuelos que vaya a cumplir.	Programación de Aeronaves
N04	RF6.0	Se requiere una interfaz gráfica para asignar vuelos a las aeronaves. En esta interfaz se podrá crear filas de vuelos que cumplirá la aeronave, y que se conformarán máximo de hasta 4 vuelos.	Programación de Aeronaves
N04	RF7.0	Las filas de vuelo se agruparán por aeronave conformando así los “Bloques de Vuelos” de cada aeronave.	Programación de Aeronaves
N05	RF8.0	El sistema también deberá permitir programar vuelos fuera de itinerario.	Programación de Aeronaves
N04	RF9.0	Para cada vuelo existirá la posibilidad de modificar sus respectivos datos que se han registrado con anticipación.	Programación de Aeronaves

Tabla 4.3 Requerimientos Funcionales – Programador de Tripulantes.

Área / Perfil: CCO – PROGRAMADOR DE TRIPULANTES			
Nec.	Ref.	Descripción	Módulo
N06	RF10.0	Para programar los itinerarios de vuelo de los tripulantes el sistema presentará la programación de las aeronaves en la que constarán los vuelos con la respectiva aeronave que lo cumplirá.	Programación de Tripulantes
N08	RF11.0	Para cada vuelo que se programará la tripulación, existirá una ventana como referencia indicando la lista de los tripulantes, informando si falta o no algún tripulante que programar en el vuelo. El número de tripulantes por cada vuelo dependerá de la aeronave seleccionada y las RDAC.	Programación de Tripulantes
N07	RF12.0	Cualquier cambio que se realice en la programación de aeronaves deberá ser reflejado en la ventana de programación de tripulantes.	Programación de Tripulantes
N08	RF13.0	Para presentar la lista de tripulantes a seleccionar, se lo realizará de acuerdo al Equipo seleccionado, Tripulación y Puesto.	Programación de Tripulantes
N08	RF14.0	El sistema deberá presentar un calendario en el cual se reflejarán los vuelos a cumplir y además las otras actividades que deberá cumplir.	Programación de Tripulantes
N08	RF15.0	Con la información almacenada en los itinerarios y	Programación

		otros datos adicionales, el sistema permitirá crear la Orden de Vuelo.	de Tripulantes
N08	RF16.0	Se podrá crear las Órdenes de Vuelo (Programa, Alcances y Cumplida). Además los Itinerarios de Vuelo (Programados y Cumplidos).	Programación de Tripulantes

Tabla 4.4 Requerimientos Funcionales – Programador de Entrenamiento.

Área / Perfil: ENT. DE TRIPUL. – PROGRAMADOR DE ENTRENAMIENTO			
Nec.	Ref.	Descripción	Módulo
N09	RF17.0	La programación registrada de cada tripulante deberá reflejarse en sus itinerarios de vuelo	Programación de Tripulantes
N10	RF18.0	El sistema debe permitir crear la malla curricular de entrenamiento de los tripulantes.	Entrenamiento de Tripulantes
N10	RF19.0	Al crear la malla curricular debe existir una opción en la que se seleccione que puesto (piloto, copiloto, auxiliar) podrá cumplir con cada segmento.	Entrenamiento de Tripulantes
N10	RF20.0	Creada la malla curricular se podrá planificar el entrenamiento anual que los tripulantes deben cumplir. Asignando a cada segmento una fecha y hora en la que se cumplirán.	Entrenamiento de Tripulantes
N11	RF21.0	Se podrá asignar a cada tripulante el entrenamiento necesario que debe cumplir para mantenerse habilitado	Entrenamiento de Tripulantes

		para el cumplimiento de las operaciones aéreas.	
--	--	---	--

Tabla 4.5 Requerimientos Funcionales – Base de Datos de Reservaciones.

Área / Perfil: RESERVACIONES – BASE DE DATOS DE RESERVACIONES			
Nec.	Ref.	Descripción	Módulo
N12	RF22.0	Los vuelos que fueron programados en el Sistema de Reservaciones de los cuales tienen permanentemente conocimiento el personal de pasajeros, deben ser presentados en Operaciones para que asignen aeronaves y tripulación para el cumplimiento de las operaciones	Programación de Aeronaves, Programación de Tripulantes
N13	RF23.0	Los vuelos que se presentarán en Operaciones solo serán de lectura y no podrán modificar sus datos. Estos cambios se los realizar desde el Sistema de Reservaciones y en coordinación con la Gerencia de Operaciones.	Programación de Aeronaves, Programación de Tripulantes

Tabla 4.6 Requerimientos Funcionales – Administrador del Sistema.

Área / Perfil: NA – ADMINISTRADOR DEL SISTEMA			
Nec.	Ref.	Descripción	Módulo
N14	RF24.0	Permitir registrar los datos de cada usuario que ingresará al sistema (Clave y contraseña).	Seguridad de Sistema

N14	RF25.0	Crear los grupos de acceso al sistema. Deberá existir previamente almacenada las diferentes opciones que permite el sistema acceder, las mismas que se asignarán a los grupos de usuario.	Seguridad de Sistema
N14	RF26.0	Luego de crear los grupos de usuarios, se permitirá asignar a cada usuario el grupo al que pertenezca. Recordando que un usuario puede pertenecer a más de un grupo.	Seguridad de Sistema
N15	RF27.0	El sistema validará el acceso al sistema por parte de los usuarios.	Seguridad de Sistema
N15	RF28.0	De acuerdo a los grupos a los que pertenezca cada usuario, el sistema le habilitará las opciones a las que puede ingresar.	Seguridad de Sistema

4.1.2 Escenarios de Casos de Uso

Luego de conocer los Requerimientos Funcionales del sistema es importante establecer los Casos de Uso que nos guiará en el cumplimiento de las tareas y el buen desarrollo de la aplicación (fig. 4.1).

A continuación se presenta los casos de uso y su respectiva descripción:

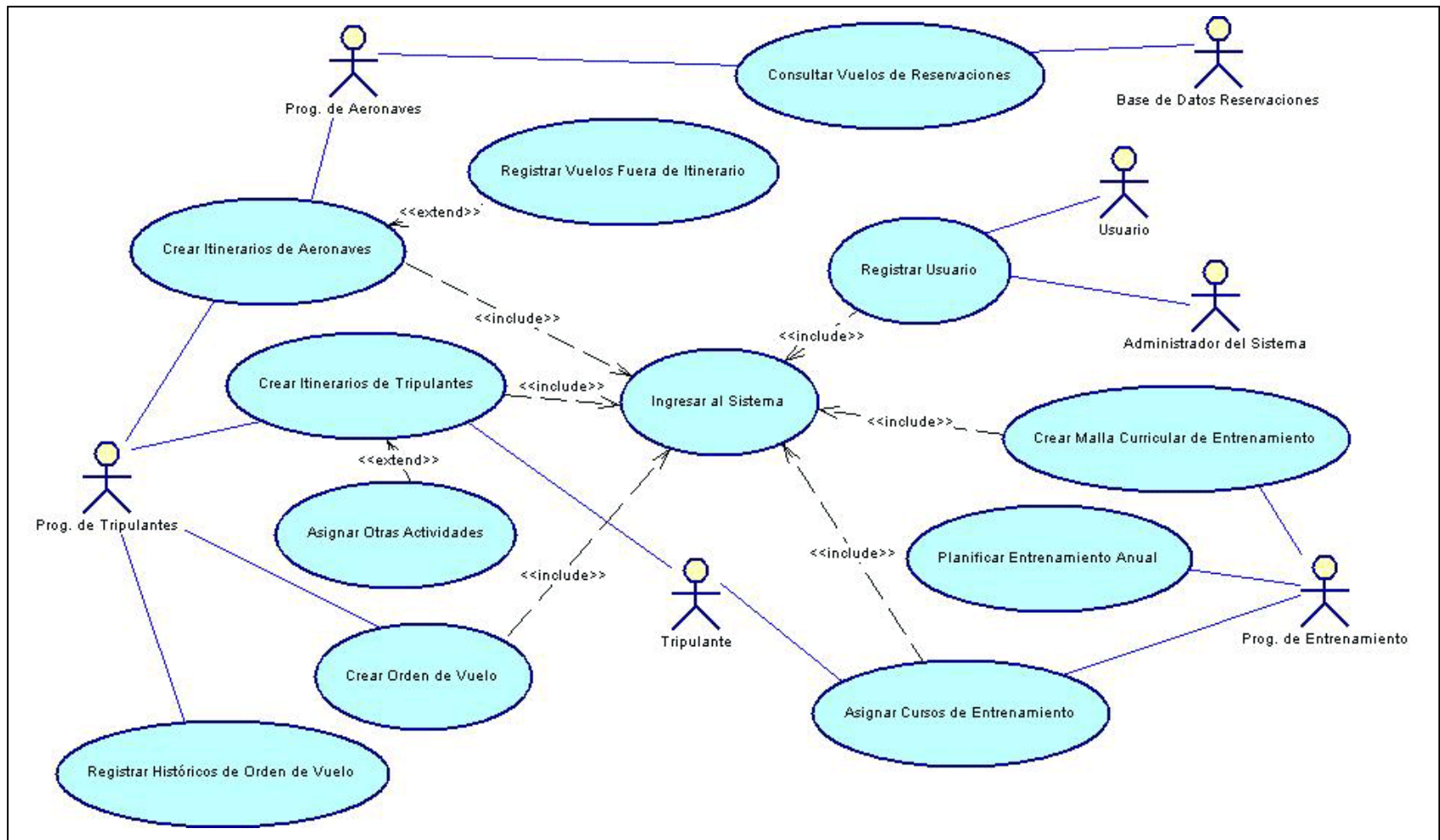


Figura 4.1 Casos de Uso del Sistema AeroFlex.

Tabla 4.7 Escenario de Caso de Uso – Registrar Usuario.

CASO DE USO		REGISTRAR USUARIO	
Código:	CU 1.0	Código RF Asociado:	RF 24.0, 25.0, 26.0
Diagrama de Caso de Uso Asociado:			
<pre> graph TD Admin[Administrador del Sistema] --- UC((Registrar Usuario)) UC --- User[Usuario] User < -- ProgTrip[Prog. de Tripulantes] User < -- ProgEnt[Prog. de Entrenamiento] User < -- ProgAer[Prog. de Aeronaves] </pre>			
Actores:	Usuario, Administrador del Sistema		
Tipo:	Básico		
Propósito:	Registrar al usuario en el sistema para su ingreso autorizado		
Resumen:	El caso de uso permitirá registrar al usuario en el sistema para posteriormente asignarle los perfiles a los que tendrá acceso.		
Precondiciones:	Deberá existir en el sistema las opciones del menú almacenadas en la base de datos.		
Post condiciones:	Ninguno.		
Flujo Principal:	Se presenta al administrador la pantalla para el registro de usuarios (PT-4.0). El administrador deberá asignarle los grupos a los cuales pertenecerá y que a su vez le asignará las opciones a las que tendrá acceso.		
Subflujos:	<p>S-1: Crear Grupos.- Se presentará al administrador una pantalla (PT-3.0) en la cual deberá crear los grupos de acceso al sistema. A estos grupos se les deberá asignar las diferentes opciones del menú principal y que están almacenadas en una base de datos.</p> <p>S-2: Crear Usuario.- Para registrar un usuario se le presentará al administrador la pantalla (PT-4.0) en la cual almacenará los datos personales del usuario. Posterior a la creación del usuario tendrá que asignarle los grupos a los que vaya a pertenecer.</p>		
Excepciones:	<p>E-1: El usuario ya existe.- Si el usuario ya existe en la base de datos, el sistema no permitirá registrarlo.</p> <p>E-2: Datos Incorrectos.- Si no se ingresa los datos que solicita el sistema, no podrá ser almacenado el usuario.</p>		

Tabla 4.8 Escenario de Caso de Uso – Ingresar al Sistema.

CASO DE USO		INGRESAR AL SISTEMA	
Código:	CU 2.0	Código RF Asociado:	RF 27.0, 28.0
Diagrama de Caso de Uso Asociado:			
<pre> graph TD Usuario((Usuario)) --- RegistrarUsuario(Registrar Usuario) RegistrarUsuario -.-> <<include>> IngresarAlSistema(Ingresar al Sistema) style RegistrarUsuario fill:#add8e6 style IngresarAlSistema fill:#a9a9a9 </pre>			
Actores:	Usuario		
Tipo:	Inclusión		
Propósito:	Ingresar al Sistema		
Resumen:	Permite ingresar al sistema con el usuario y clave correctos.		
Precondiciones:	El usuario deberá encontrarse registrado en el sistema.		
Post condiciones:	Ninguno.		
Flujo Principal:	Se presenta al usuario la ventana (PT-2.0) para ingresar su “usuario” y “clave” correctos, y posteriormente ingresar al sistema.		
Subflujos:	S-1: Ingresar Datos de Usuario.- El usuario deberá ingresar el “usuario” y “clave” que se le fueron asignados. Si existe un dato mal ingresado el sistema le indicará que ingrese nuevamente. Posterior a ingresar los datos solicitados, el usuario ingresará y se habilitará las opciones a las que tiene acceso.		
Excepciones:	E-1: Datos Incorrectos.- Si no ingresa los datos solicitados correctamente.		

Tabla 4.9 Escenario de Caso de uso – Consultar Vuelos de Reservaciones.

CASO DE USO		CONSULTAR VUELOS DE RESERVACIONES	
Código:	CU 3.0	Código RF Asociado:	RF 4.0, 22.0, 23.0
Diagrama de Caso de Uso Asociado:			
Actores:	Programador de Aeronaves		
Tipo:	Básico		
Propósito:	Consultar los vuelos de itinerarios que se encuentran en la base de datos de reservaciones		
Resumen:	Para acceder a los vuelos de itinerarios, el sistema de reservaciones registrar con anticipación todos los vuelos de itinerarios. Posteriormente se podrá consultar para programar las aeronaves.		
Precondiciones:	El sistema de reservaciones que pertenece a la Empresa debe registrar los vuelos de itinerarios.		
Post condiciones:	Ninguno.		
Flujo Principal:	En la pantalla (PT-5.0) se debe ingresar la fecha de los vuelos que se irá a programar, para poder visualizarlos.		
Subflujos:			
Excepciones:	E-1: No se encuentran registrados los vuelos.- Si el sistema de reservaciones no registró los vuelos en la base de datos, no se podrá visualizar ni programar las aeronaves.		

Tabla 4.10 Escenario de Caso de uso – Crear Itinerarios de Aeronaves.

CASO DE USO		CREAR ITINERARIOS DE AERONAVES	
Código:	CU 4.0	Código RF Asociado:	RF 5.0, 6.0, 7.0, 9.0
Diagrama de Caso de Uso Asociado:			
Actores:	Programador de Aeronaves, Programador de Tripulantes		
Tipo:	Básico		
Propósito:	Programar los vuelos que deben cumplir diariamente las aeronaves		
Resumen:	El sistema permitirá programar a las aeronaves los vuelos de itinerario que se extrae de la base de datos de reservaciones. Asignando la aeronave y la fecha de vuelo.		
Precondiciones:	El sistema de reservaciones que pertenece a la Empresa debe registrar los vuelos de itinerarios. Además, para programar se debe seleccionar la semana a la que se asignará los vuelos y aeronaves.		
Post condiciones:	A los vuelos que fueron asignados aeronaves deberá programarse tripulantes para cumplir con las operaciones.		
Flujo Principal:	Se seleccionará el día que se vaya a programar los vuelos. Posteriormente se seleccionará la aeronave que vaya a cumplir con el bloque de vuelos a crear.		
Subflujos:	<p>S-1: Seleccionar día de vuelo y aeronave.- En la pantalla (PT-5.0) se seleccionará el día de vuelo que se vaya a programar. Posteriormente se seleccionará el equipo y la aeronave que cumplirá con los vuelos.</p> <p>S-2: Seleccionar vuelos y asignaran aeronave.- Se podrá seleccionar hasta 4 vuelos que conformarán un fila que la aeronave cumplirá. Podrá crear varias filas de vuelos en la pantalla (PT-7.0) dando como resultado el Bloque de Vuelos que cumplirá cada aeronave.</p> <p>S-3: Administrar Vuelos.- Si se desea posteriormente modificar los datos de un vuelo, el usuario podrá ingresar a la pantalla (PT-8.0).</p>		
Excepciones:	E-1: Se seleccionada más de 4 vuelos.- Si se selecciona más de 4 vuelos el sistema no permitirá crear la fila de vuelos para la aeronave.		

Tabla 4.11 Escenario de Caso de uso – Registrar Vuelos Fuera de Itinerario.

CASO DE USO		REGISTRAR VUELOS FUERA DE ITINERARIO	
Código:	CU 5.0	Código RF Asociado:	RF 8.0, 9.0
Diagrama de Caso de Uso Asociado:			
<pre> graph LR Actor[Prog. de Aeronaves] --- UC1((Crear Itinerarios de Aeronaves)) UC1 -.-> <<extend>> UC2((Registrar Vuelos Fuera de Itinerario)) </pre>			
Actores:	Programador de Aeronaves		
Tipo:	Extensión		
Propósito:	Registrar vuelos que se encuentran fuera de itinerario.		
Resumen:	En la Empresa también se cumplen vuelos que no fueron programados. Estos vuelos pueden ser Logísticos, Charter, Presidenciales, etc. El sistema debe permitir registrarlos.		
Precondiciones:	Se debe seleccionar la semana que se programa las aeronaves.		
Post condiciones:	A los vuelos que fueron asignados aeronaves deberá programarse tripulantes para cumplir con las operaciones.		
Flujo Principal:	En la pantalla (PT-9.0) el usuario podrá crear vuelos fuera de itinerario. Debe ingresar los datos que se le solicita y que son necesarios para crear la fila de vuelos. Posteriormente estos datos podrán ser modificados en la pantalla (PT-8.0).		
Subflujos:			
Excepciones:	E-1: Datos Incorrectos.- Si no se ingresa los datos que solicita el sistema, no podrán ser almacenados.		

Tabla 4.12 Escenario de Caso de uso – Crear Itinerarios de Tripulantes.

CASO DE USO		CREAR ITINERARIOS DE TRIPULANTES	
Código:	CU 6.0	Código RF Asociado:	RF 1.0, 2.0, 3.0, 10.0, 11.0, 12.0, 13.0, 14.0, 17.0, 22.0
Diagrama de Caso de Uso Asociado:			
Actores:	Programador de Tripulantes, Tripulante		
Tipo:	Básico		
Propósito:	Programar los vuelos que deben cumplir diariamente los tripulantes		
Resumen:	Luego que se asignaron aeronaves a los vuelos, se deberá programar tripulantes para el cumplimiento de las operaciones.		
Precondiciones:	Programar aeronaves a cada vuelo en la programación semanal.		
Post condiciones:	Publicar los itinerarios de vuelo.		
Flujo Principal:	Se seleccionara el día que se vaya a programar los itinerarios de los tripulantes. Posteriormente se deberá seleccionar el tripulante y para finalizar los vuelos que volará.		
Subflujos:	<p>S-1: Seleccionar los vuelos.- Se escogerá el día de vuelo (PT-13.0) y por medio del cual el sistema presentará los vuelos a cumplir por la Empresa.</p> <p>S-2: Seleccionar tripulante.- Se debe seleccionar el equipo del que se va a programar, la tripulación y el puesto. Se presentará la lista de tripulantes que coinciden con estos datos y se seleccionará un tripulante para programar sus vuelos. El sistema presenta un calendario mensual (PT-11.0) o semanal (PT-12.0) de todas las actividades que tiene por cumplir.</p> <p>S-3: Asignar vuelos: Se podrá seleccionar uno o varios vuelos del bloque de vuelos del equipo al que pertenece el tripulante (PT-13.0). Recordando que la interfaz gráfica presenta colores que guían e indica al usuario si un vuelo está o no completa la tripulación según dicta las RDAC.</p> <p>S-4: Visualizar la tripulación de vuelo: Por cada vuelo se podrá visualizar la lista de tripulantes que existe en cada vuelo (PT-14.0), con la finalidad de guiar al usuario en la programación.</p>		
Excepciones:	E-1: Cumplir otras actividades.- En el calendario del tripulante se presentarán varias actividades que puede cumplir en la Empresa. El sistema controlará que no coincidan 2 o más actividades en la misma hora programada.		

Tabla 4.13 Escenario de Caso de uso – Asignar Otras Actividades.

CASO DE USO		ASIGNAR OTRAS ACTIVIDADES	
Código:	CU 7.0	Código RF Asociado:	RF 2.0
Diagrama de Caso de Uso Asociado:			
<pre> graph TD Actor1[Prog. de Tripulantes] --- UC1((Crear Itinerarios de Tripulantes)) Actor2[Tripulante] --- UC1 UC1 -.-> <<extend>> UC2((Asignar Otras Actividades)) </pre>			
Actores:	Programador de Tripulantes, Tripulante		
Tipo:	Extensión		
Propósito:	Registrar otras actividades que cumple un tripulante		
Resumen:	El tripulante además de volar puede cumplir otras actividades en la Empresa como Permisos Médicos, Entrenamiento, etc. El sistema permitirá asignar este tipo de actividades.		
Precondiciones:	Visualizar el calendario del tripulante.		
Post condiciones:	Ninguno.		
Flujo Principal:	En la pantalla (PT-15.0) el sistema permitirá programar otro tipo de actividades que cumple el tripulante, tan solo con seleccionar la actividad e ingresar los datos requeridos según la seleccionada.		
Subflujos:	Ninguno.		
Excepciones:	E-1: Cumplir otras actividades.- En el calendario del tripulante se presentarán varias actividades que puede cumplir en la Empresa. El sistema controlará que no coincidan 2 o más actividades en la misma hora programada.		

Tabla 4.14 Escenario de Caso de uso – Crear Orden de Vuelo.

CASO DE USO		CREAR ORDEN DE VUELO	
Código:	CU 8.0	Código RF Asociado:	RF 15.0, 16.0
Diagrama de Caso de Uso Asociado:			
<pre> graph LR Actor[Prog. de Tripulantes] --- UC1((Crear Orden de Vuelo)) UC1 -.-> <<include>> UC2((Ingresar Al Sistema)) </pre>			
Actores:	Programador de Tripulantes		
Tipo:	Básico		
Propósito:	Crear la Orden de Vuelo		
Resumen:	Con los datos ingresados en los itinerarios de vuelo y otros adicionales, se puede publicar las Órdenes de Vuelo.		
Precondiciones:	Crear los itinerarios de vuelo.		
Post condiciones:	Ninguno.		
Flujo Principal:	En la pantalla (PT-17.0) se puede publicar las Órdenes de Vuelos tan solo con seleccionar el día e ingresar los datos solicitados.		
Subflujos:	<p>S-1: Publicar Orden Vuelo (Programada).- Un día antes de la operación se podrá grabar este documento y reflejará la operación programada. Se deberá seleccionar el día y crear una nueva orden de vuelo de este tipo e ingresar los datos solicitados.</p> <p>S-2 Publicar Orden Vuelo (Alcances).- Luego de grabar la OV-Programada se podrá realizar modificaciones (Alcances) a este documento y reflejará la operación programada con los cambios realizados. Se deberá seleccionar el día y crear una nueva orden de vuelo de este tipo e ingresar los datos solicitados.</p> <p>S-3: Publicar Orden de Vuelos (Cumplida).- Luego de finalizar el día y cumplida la operación se deberá actualizar este documento como se haya cumplido la operación, dando como resultado la OV-Cumplida. Se seleccionará el día y crear una nueva orden de vuelo de este tipo e ingresar los datos solicitados.</p>		
Excepciones:	E-1: No presenta datos.- Si no se almacenado con anticipación los itinerarios de vuelos, la Orden de Vuelo no presentará datos.		

Tabla 4.15 Escenario de Caso de uso – Registrar Históricos de Orden de Vuelo.

CASO DE USO		REGISTRAR HISTORICOS DE ORDEN DE VUELO	
Código:	CU 9.0	Código RF Asociado:	RF 16.0
Diagrama de Caso de Uso Asociado:			
<pre> graph LR Actor[Prog. de Tripulantes] --- UC(Registrar Históricos de Orden de Vuelo) </pre>			
Actores:	Programador de Tripulantes		
Tipo:	Básico		
Propósito:	Publicar los históricos de la Orden de Vuelo		
Resumen:	Por control y datos estadísticos, el sistema permitirá almacenar datos históricos de lo Programado y lo Cumplido.		
Precondiciones:	Registrar los itinerarios de vuelo.		
Post condiciones:	Ninguno.		
Flujo Principal:	Los históricos de la orden de vuelo se componen de los Itinerarios de Vuelo y las Órdenes de Vuelo. Estos datos se pueden clasificar en Programados, Alcances y Cumplidos. En las pantallas (PT-16.0) y (PT-17.0) se podrá almacenar y publicarlos		
Subflujos:	<p>S-1: Publicar Itinerarios de Vuelo.- En la pantalla (PT-16.0) el sistema permitirá registrar los itinerarios de vuelos programados (cuando se los entrega a los tripulantes) y los itinerarios de vuelos cumplidos (cumplida la operación). Se deberá ingresar los datos solicitados y seleccionar el tipo de Itinerario de Vuelo a publicar.</p> <p>S-2: Publicar Órdenes de Vuelo.- En la pantalla (PT-17.0) el sistema permitirá registrar las órdenes de vuelo programadas, alcances y cumplidas. Se ingresará los datos solicitados y seleccionar el tipo de orden de vuelo a publicar.</p>		
Excepciones:	E-1: No presenta datos.- Si no se almacenado con anticipación los itinerarios de vuelos, la Orden de Vuelo no presentará datos.		

Tabla 4.16 Escenario de Caso de uso – Crear Malla Curricular de Entrenamiento.

CASO DE USO		CREAR MALLA CURRICULAR DE ENTRENAMIENTO	
Código:	CU 10.0	Código RF Asociado:	RF 18.0, 19.0
Diagrama de Caso de Uso Asociado:			
<pre> graph LR Actor[Prog. de Entrenamiento] --- UC1((Crear Malla Curricular de Entrenamiento)) UC1 -.-> <<include>> UC2((Ingresar Al Sistema)) </pre>			
Actores:	Programador de Entrenamiento		
Tipo:	Básico		
Propósito:	Crear la malla curricular de entrenamiento para la tripulación		
Resumen:	El sistema permitirá crear la malla curricular de entrenamiento con la que se podrá programar el entrenamiento anual de los tripulantes		
Precondiciones:	Ninguno.		
Post condiciones:	Ninguno.		
Flujo Principal:	Se deberá registrar los segmentos de entrenamiento (PT-18.0) con los que se podrá programar las categorías (PT-19.0) y el puesto que cumplirá este entrenamiento.		
Subflujos:	S-1: Registrar Segmentos de Entrenamiento.- En la pantalla (PT-18.0) se registrará los datos solicitados para crear los segmentos de entrenamiento. S-2: Crear Malla Curricular.- En la pantalla (PT-19.0) se seleccionará la categoría y los segmentos que la conformarán, además los puestos que deberán cumplir.		
Excepciones:	Ninguno.		

Tabla 4.17 Escenario de Caso de uso – Planificar Entrenamiento Anual.

CASO DE USO		PLANIFICAR ENTRENAMIENTO ANUAL	
Código:	CU 11.0	Código RF Asociado:	RF 20.0
Diagrama de Caso de Uso Asociado:			
<pre> graph LR Actor[Prog. de Entrenamiento] --- UC((Planificar Entrenamiento Anual)) </pre>			
Actores:	Programador de Entrenamiento		
Tipo:	Básico		
Propósito:	Planificar el entrenamiento anual de la tripulación.		
Resumen:	En el sistema se tendrá que registrar el entrenamiento anual que debe cumplir la tripulación para encontrarse habilitados.		
Precondiciones:	Registrar la malla curricular.		
Post condiciones:	Asignar tripulación al entrenamiento planificado.		
Flujo Principal:	En la pantalla (PT-20.0) se seleccionará la categoría y segmento que se programará, adicional se deberá registrar la fecha y hora en el que se cumplirá.		
Subflujos:	Ninguno.		
Excepciones:	Ninguno.		

Tabla 4.18 Escenario de Caso de uso – Asignar Cursos de Entrenamiento.

CASO DE USO		ASIGNAR CURSOS DE ENTRENAMIENTO	
Código:	CU 12.0	Código RF Asociado:	RF 21.0
Diagrama de Caso de Uso Asociado:			
<pre> graph LR Actor1[Prog. de Entrenamiento] --- UC1(Asignar Cursos de Entrenamiento) Actor2[Tripulante] --- UC1 UC1 -.-> <<include>> UC2(Ingresar Al Sistema) </pre>			
Actores:	Programador de Entrenamiento, Tripulante		
Tipo:	Básico		
Propósito:	Asignar entrenamiento a la tripulación para encontrarse habilitado.		
Resumen:	Para que un tripulante se mantenga habilitado para cumplir las operaciones aéreas, debe cumplir cierto tipo de entrenamiento que le asigna la Empresa.		
Precondiciones:	Planificar entrenamiento para los tripulantes.		
Post condiciones:	Ninguno.		
Flujo Principal:	Se presenta la pantalla (PT-21.0) en la que se deberá seleccionar el entrenamiento al que se quiere asignar tripulación. Escogiendo el puesto se presenta la lista de tripulantes de los cuales se podrá seleccionar los que vayan a cumplir con el entrenamiento.		
Subflujos:	Ninguno.		
Excepciones:	Ninguno.		

4.1.3 Diagramas de Secuencia

De acuerdo a los flujos y subflujos de los casos de uso a continuación se presenta sus respectivos Diagramas de Secuencia que define el comportamiento de la aplicación:

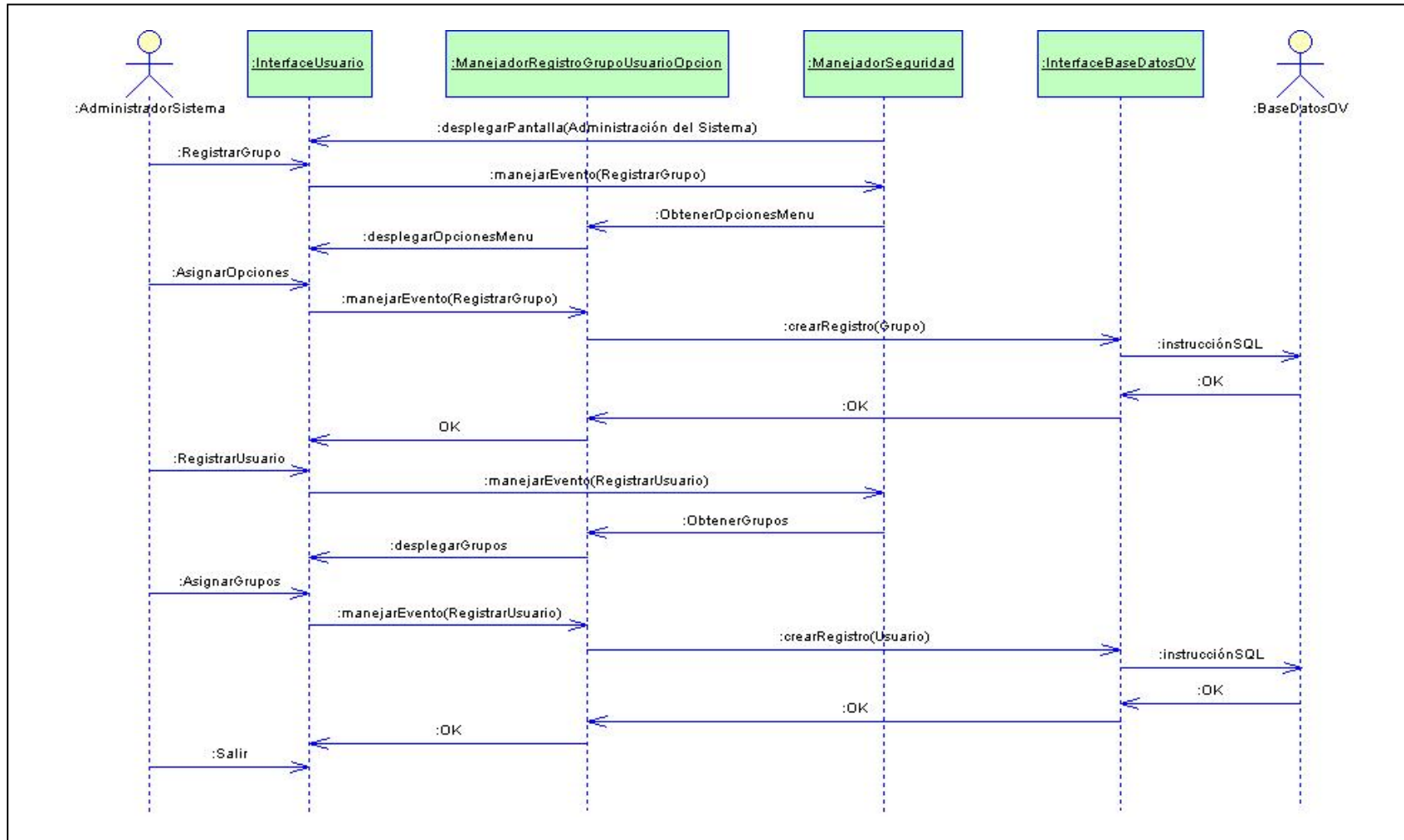


Figura 4.2 Diagrama de Secuencia – Registrar Usuario.

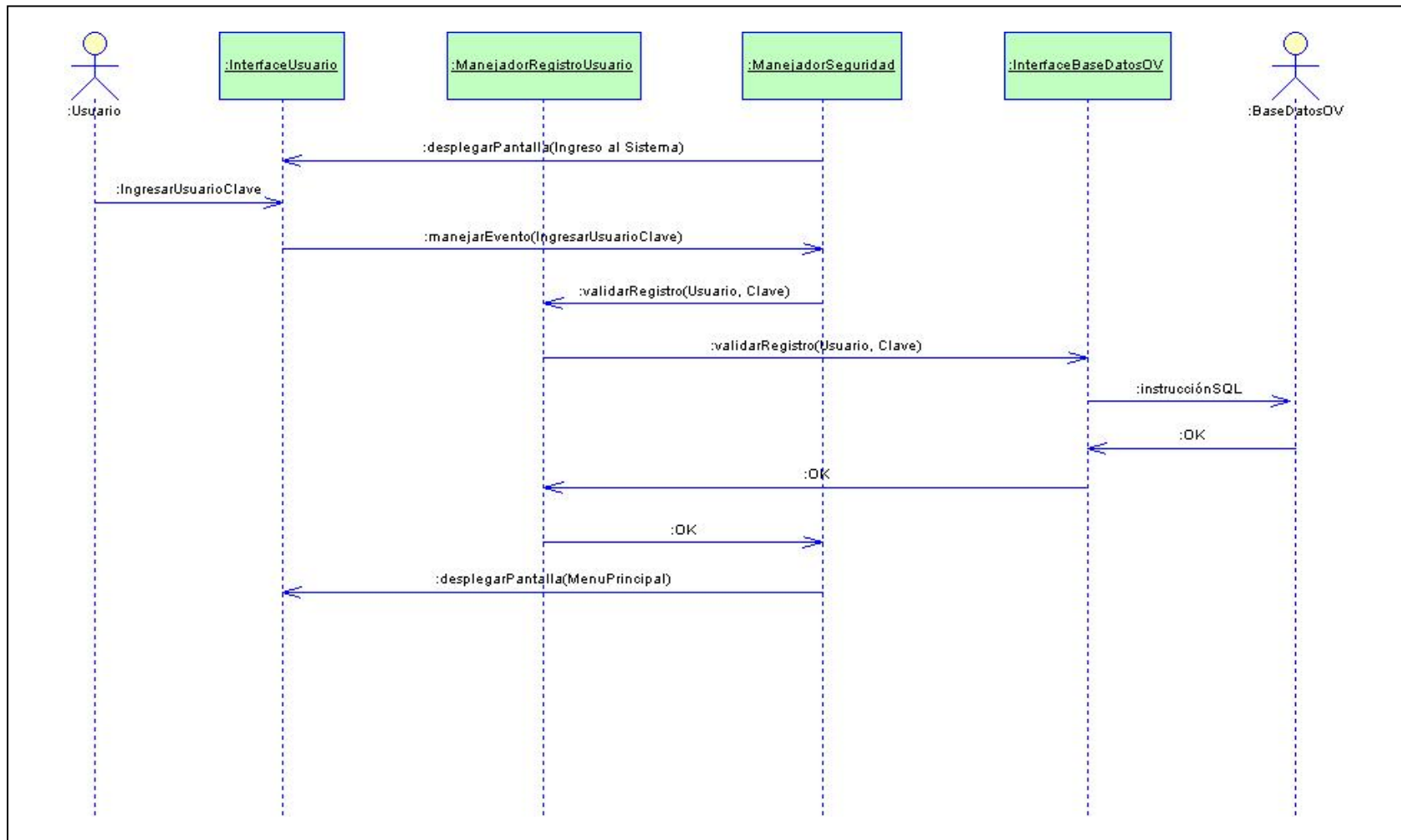


Figura 4.3 Diagrama de Secuencia – Ingresar al Sistema.

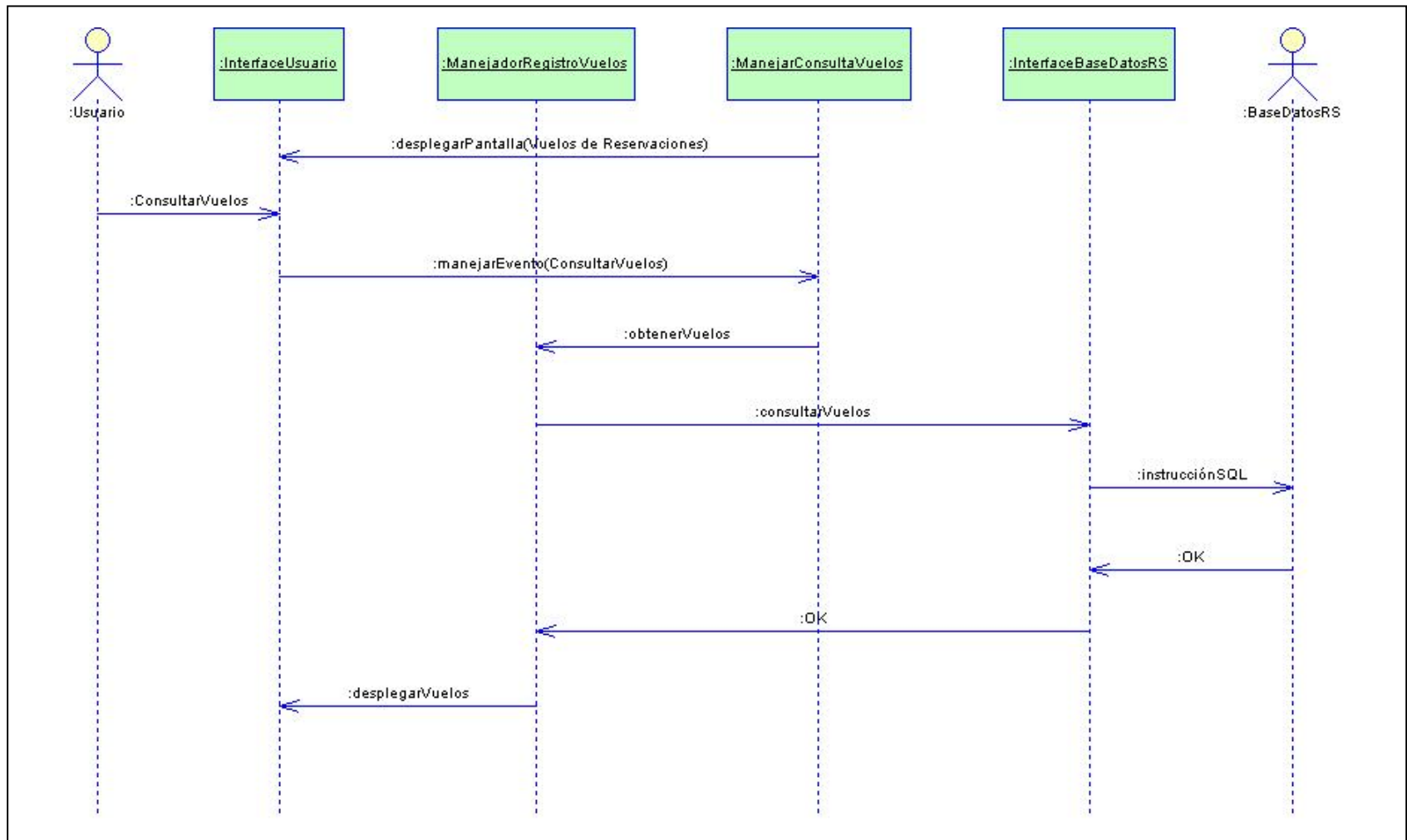


Figura 4.4 Diagrama de Secuencia – Consultar Vuelos de Reservaciones.

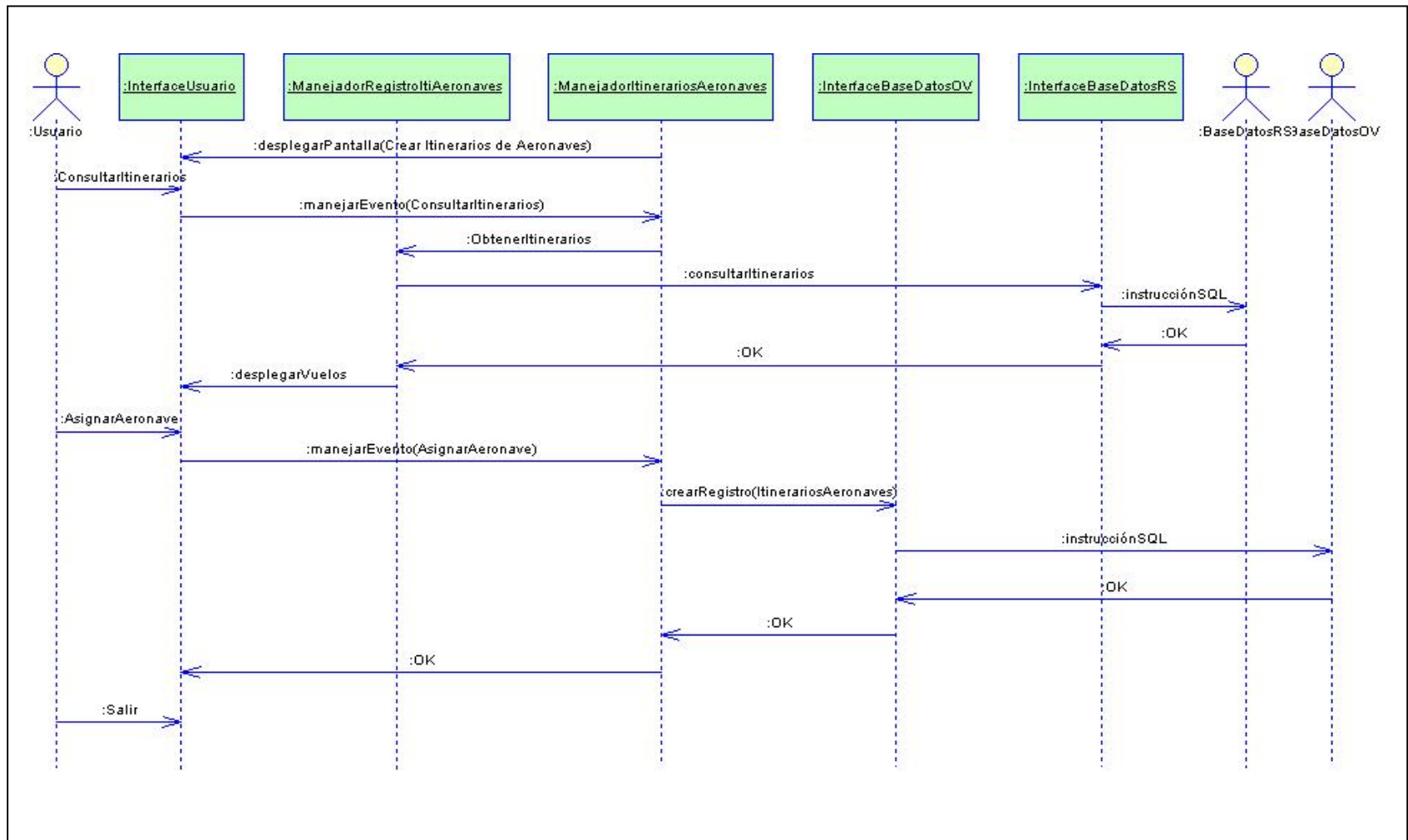


Figura 4.5 Diagrama de Secuencia – Crear Itinerarios de Aeronaves.

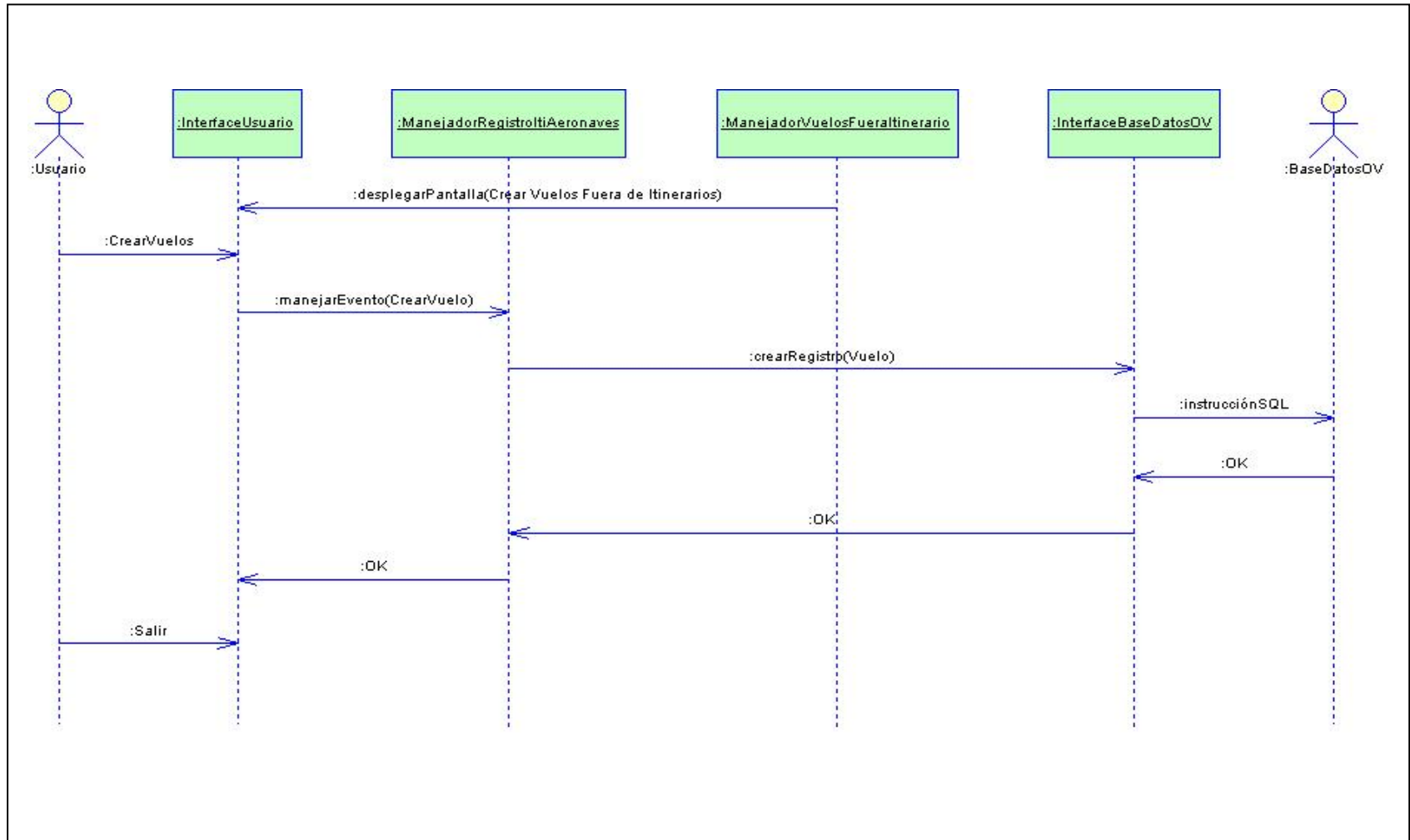


Figura 4.6 Diagrama de Secuencia – Registrar Vuelos Fuera de Itinerario.

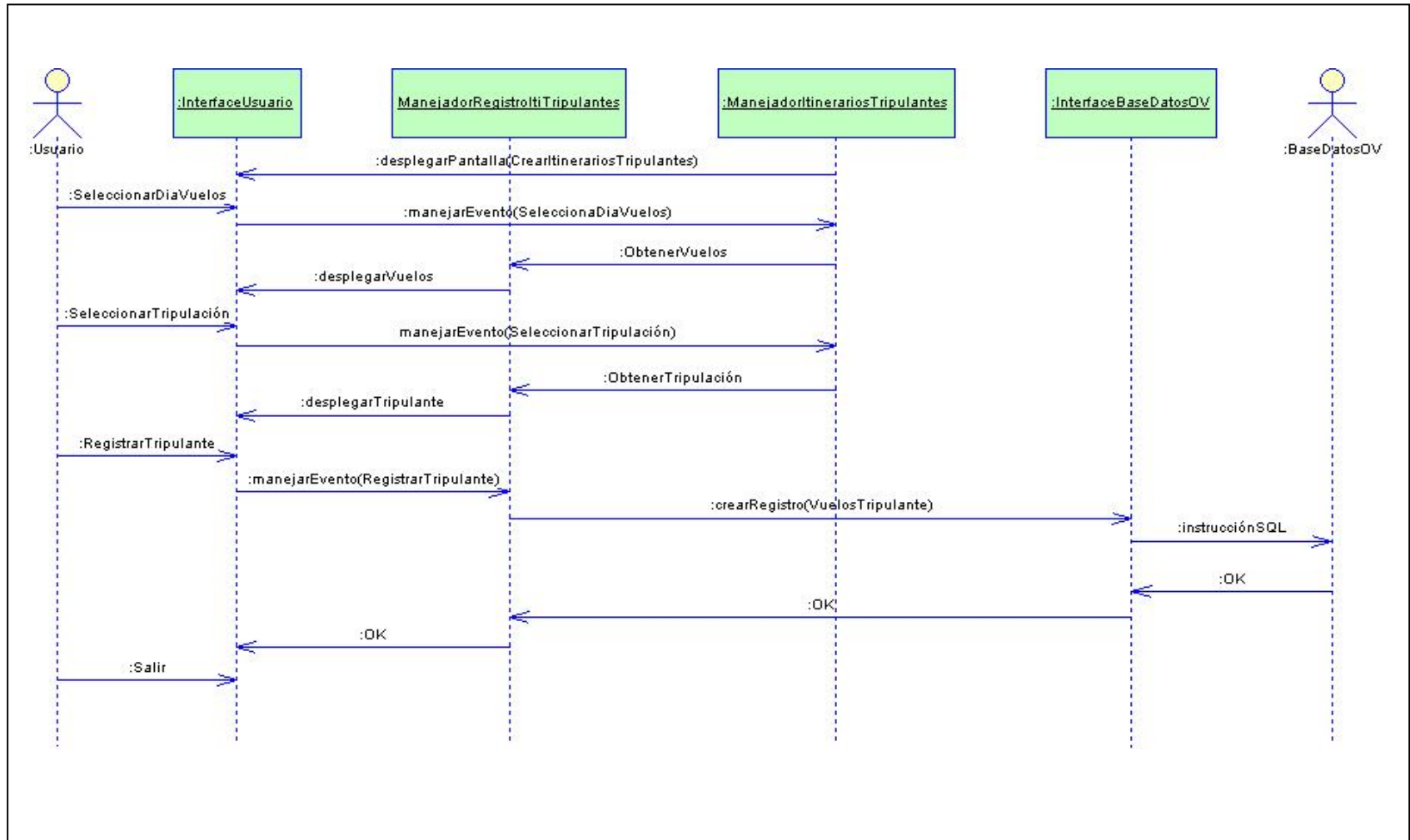


Figura 4.7 Diagrama de Secuencia – Crear Itinerarios de Tripulantes.

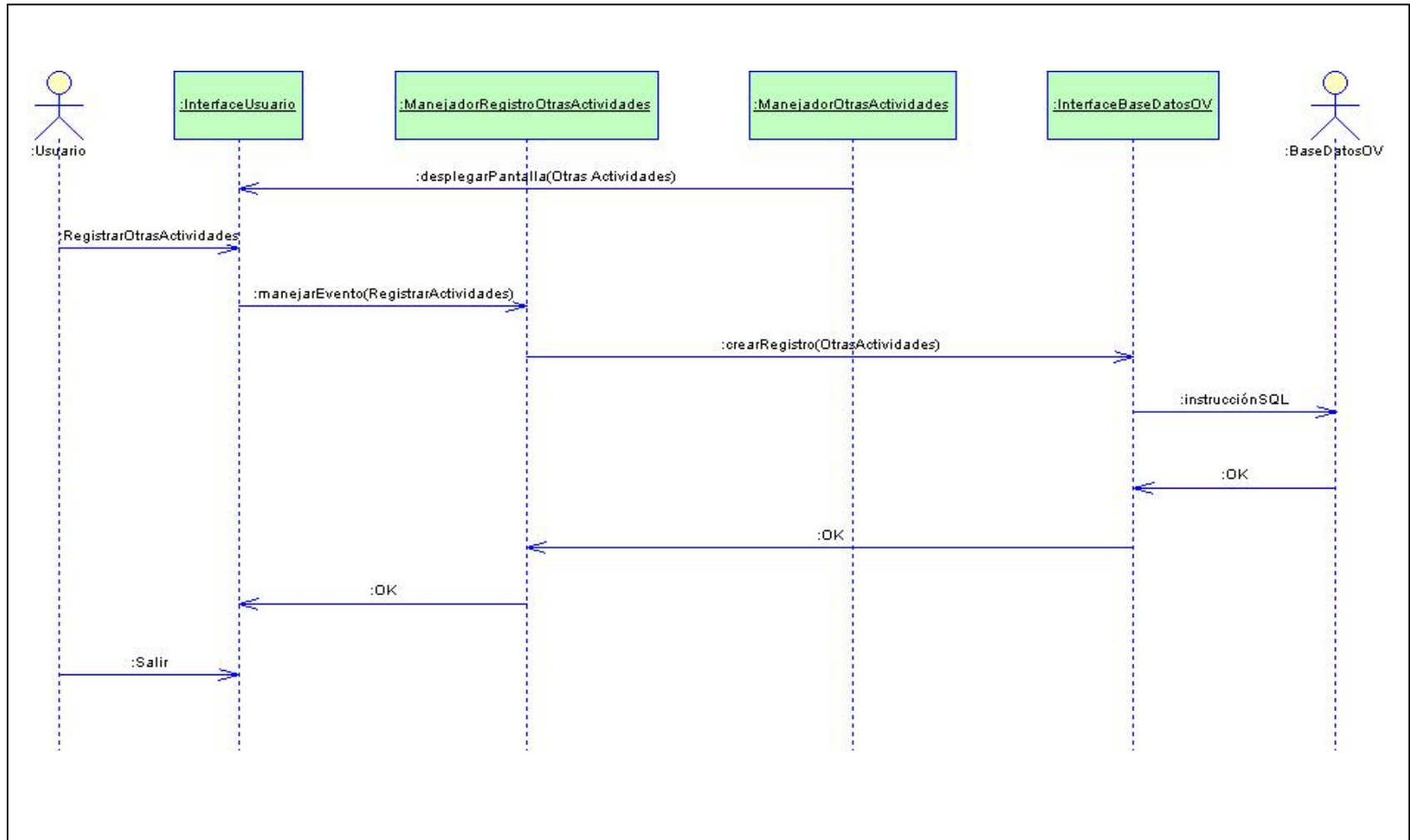


Figura 4.8 Diagrama de Secuencia – Asignar Otras Actividades.

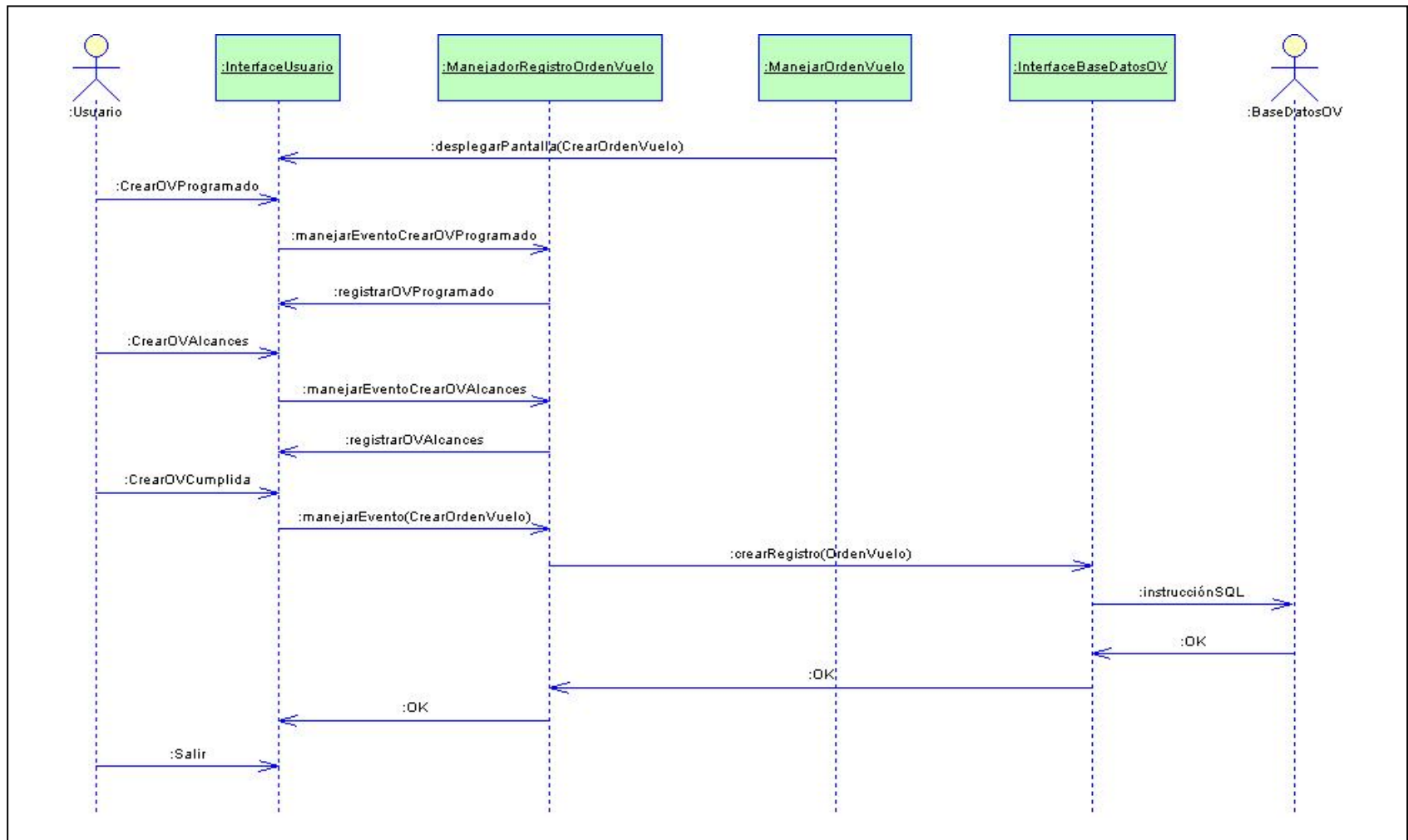


Figura 4.9 Diagrama de Secuencia – Crear Orden de Vuelo.

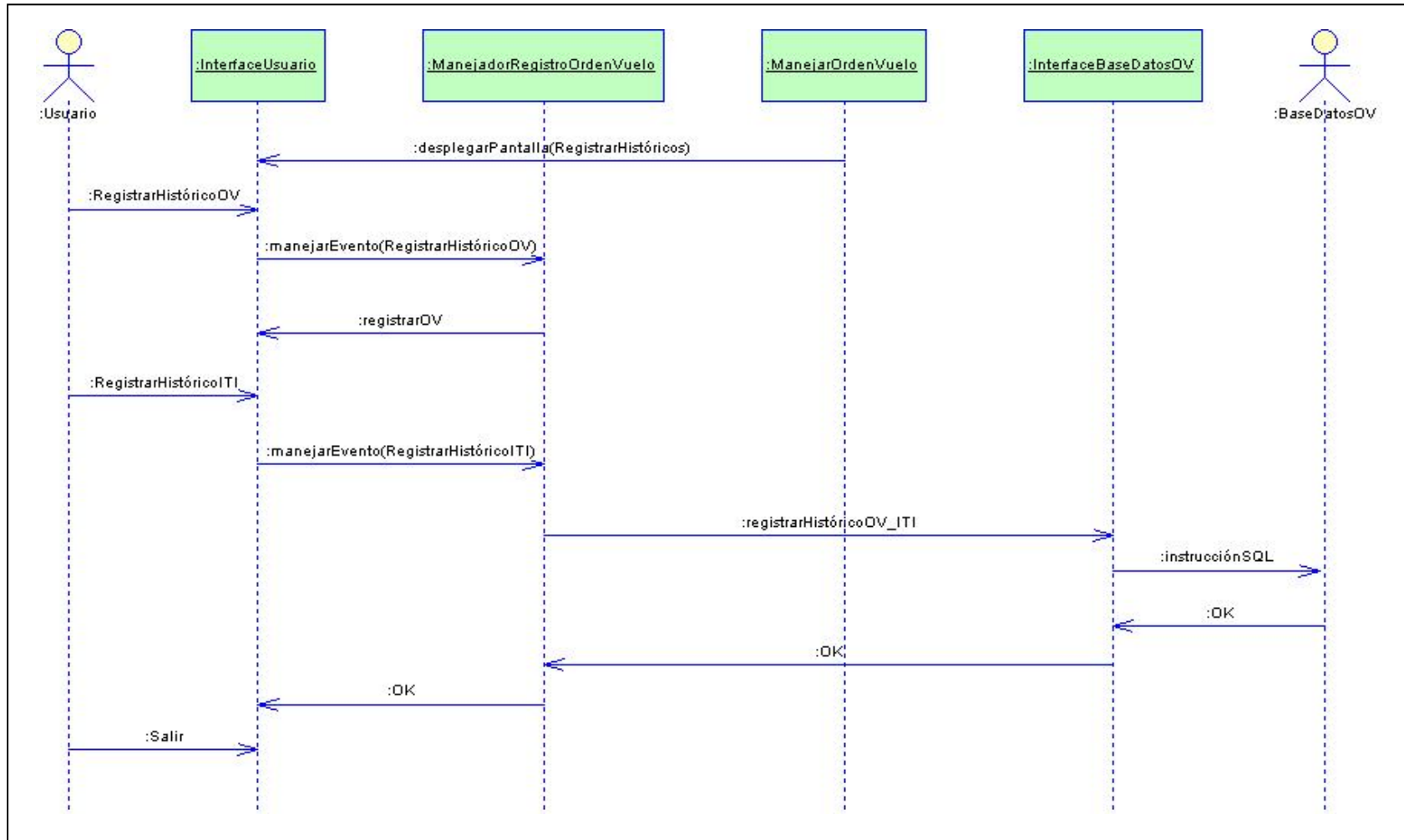


Figura 4.10 Diagrama de Secuencia – Registrar Históricos de Orden de Vuelo.

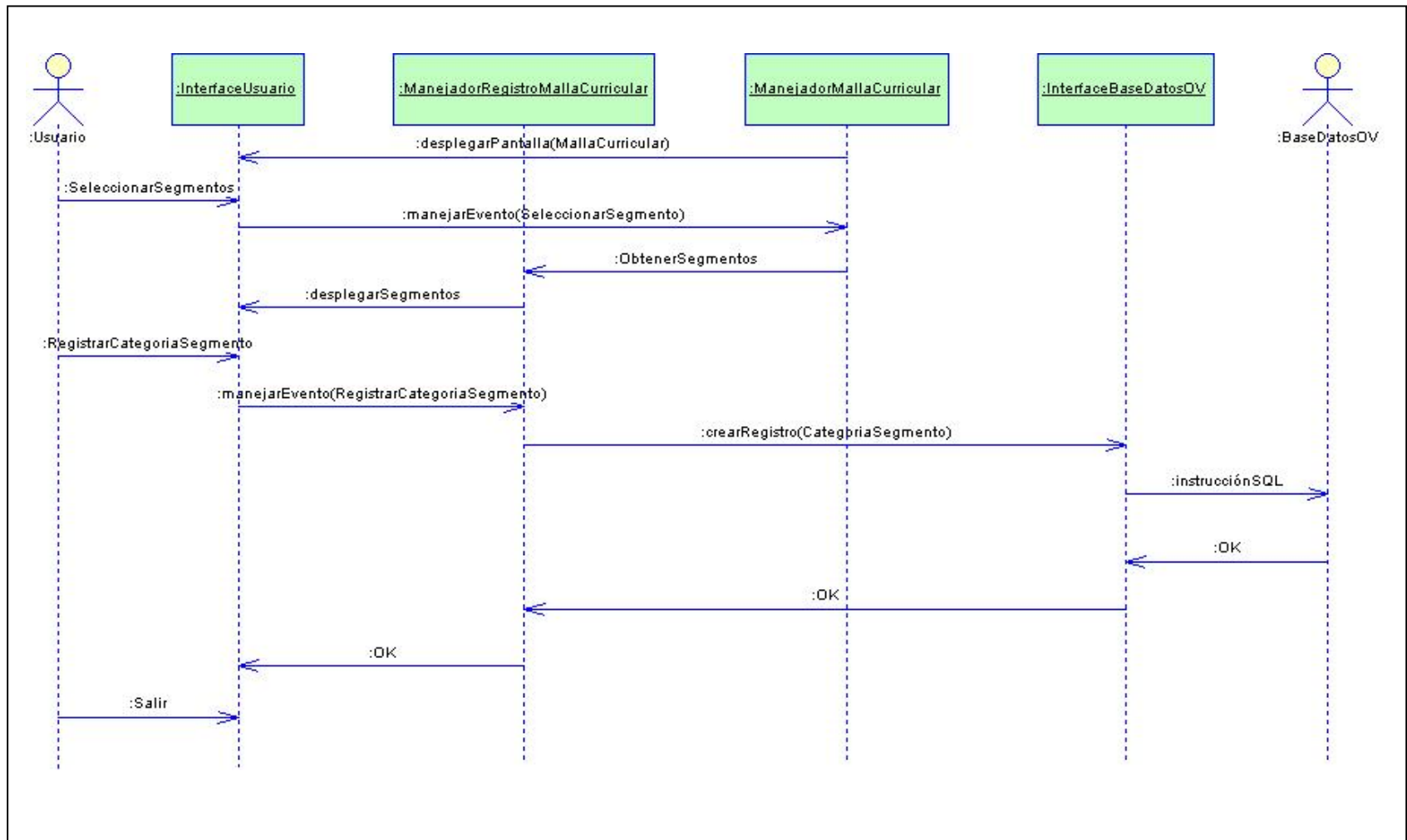


Figura 4.11 Diagrama de Secuencia – Crear Malla Curricular de Entrenamiento.

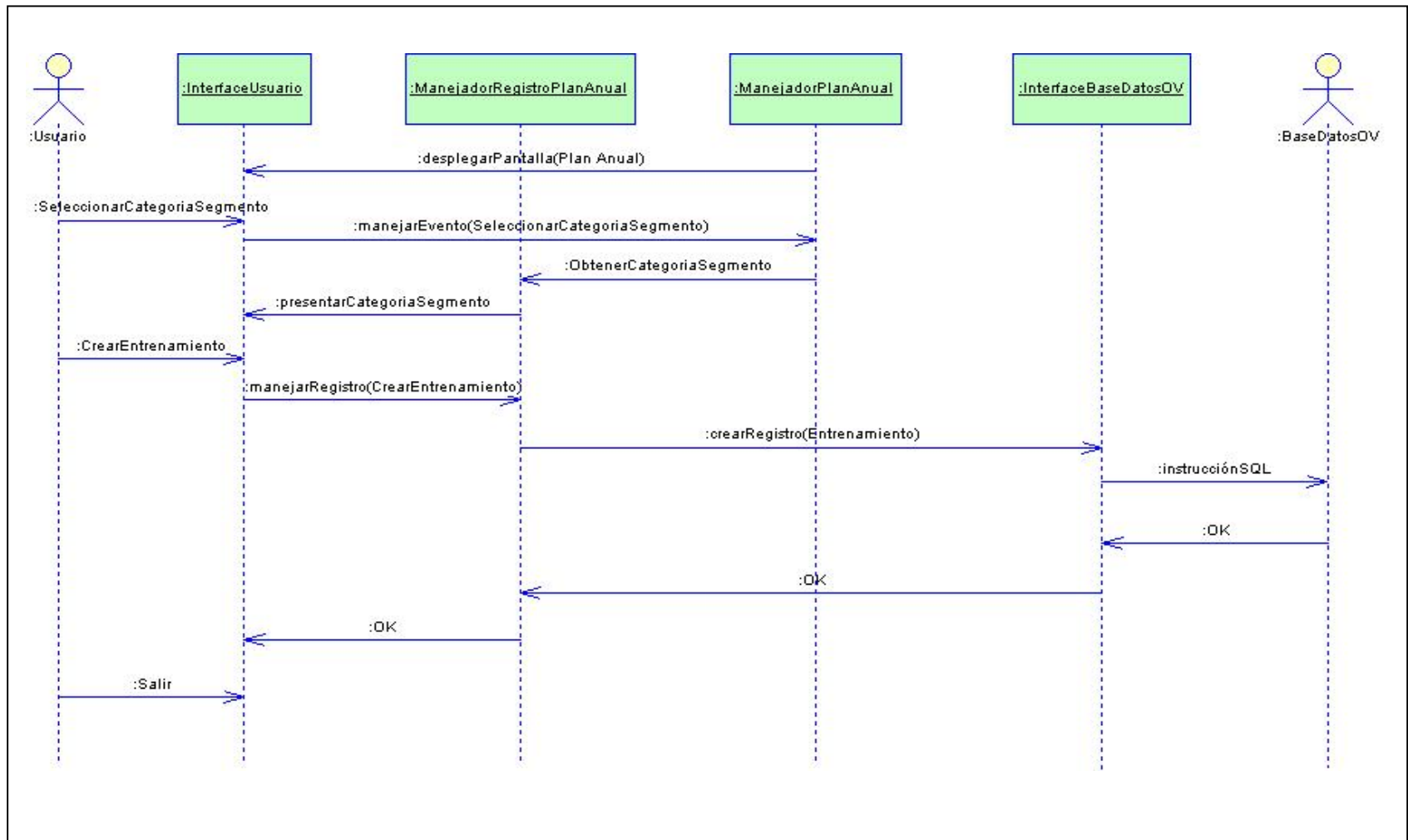


Figura 4.12 Diagrama de Secuencia – Planificar Entrenamiento Anual.

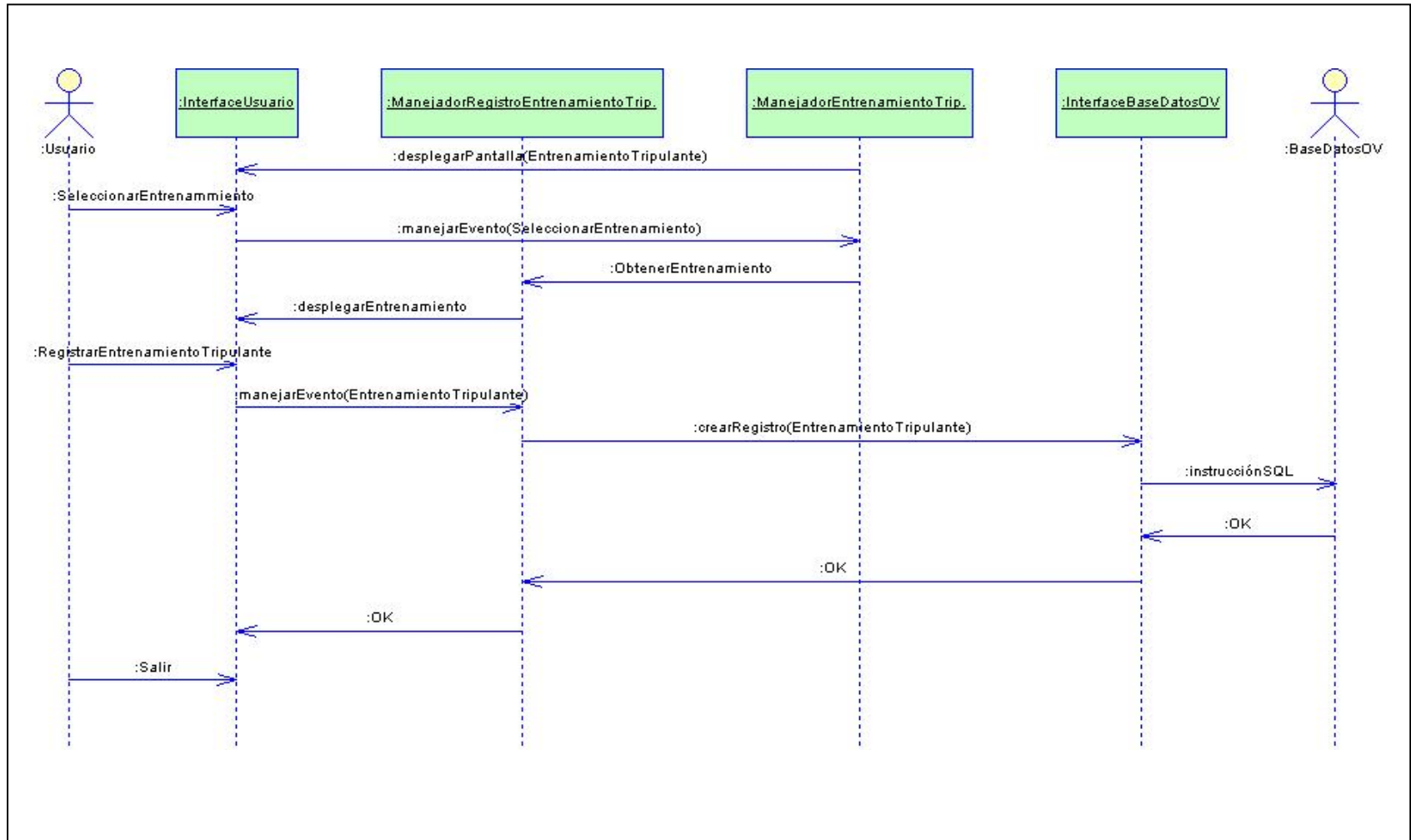


Figura 4.13 Diagrama de Secuencia – Asignar Cursos de Entrenamiento.

4.1.4 Prototipos de Pantallas

Con la finalidad de presentar una guía para el desarrollo de la aplicación., es importante definir el prototipo de las pantallas que la conformarán. Estos prototipos están de acuerdo a los requerimientos funcionales establecidos por el equipo de integrantes del proyecto.

Tabla 4.19 Prototipo – Menú del Sistema.

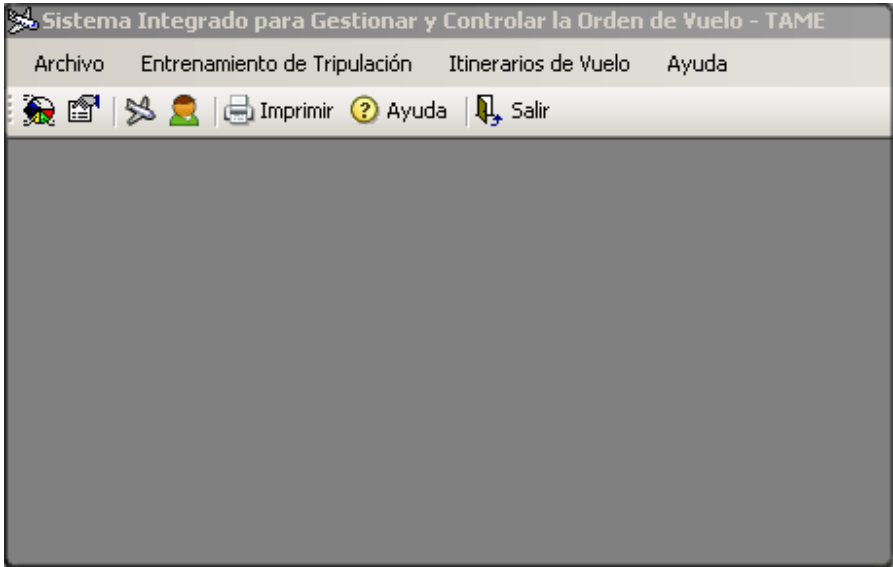
PROTOTIPO		MENU DEL SISTEMA	
Código:	PT 1.0	Caso de Uso Asociado:	
Descripción:	Permite acceder a las diferentes opciones que tiene el sistema.		
			

Tabla 4.20 Prototipo – Ingreso al Sistema.


PROTOTIPO		INGRESO AL SISTEMA	
Código:	PT 2.0	Caso de Uso Asociado:	CU 2.0
Descripción:	Permite el ingreso al sistema por medio del usuario y contraseña autorizados.		
			

Tabla 4.21 Prototipo – Administración de Grupos.

PROTOTIPO		ADMINISTRACIÓN DE GRUPOS	
Código:	PT 3.0	Caso de Uso Asociado:	CU 1.0
Descripción:	Permite administrar los grupos de usuarios. Para asignar a cada usuario el grupo al que pertenecerá, se debe primero crear estos grupos y asignarle las opciones a las que tendrán acceso.		

Tabla 4.22 Prototipo – Administración de Usuarios.

PROTOTIPO		ADMINISTRACIÓN DE USUARIOS	
Código:	PT 4.0	Caso de Uso Asociado:	CU 1.0
Descripción:	La presente interfaz permite crear la lista de usuarios que accederán al sistema. A cada usuario se le deberá asignar los grupos a los que pertenecerá.		

Tabla 4.23 Prototipo – Vuelos de Reservaciones.

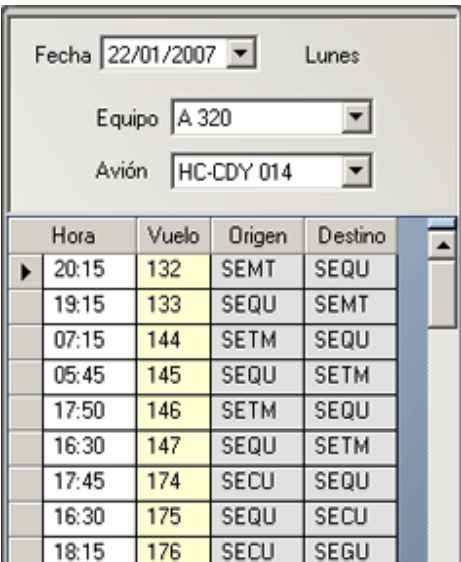
PROTOTIPO		VUELOS DE RESERVACIONES																																									
Código:	PT 5.0	Caso de Uso Asociado:	CU 3.0, 4.0																																								
Descripción:	Presenta la lista de vuelos del día seleccionado y que fueron creados en el Sistema de Reservaciones de la Empresa																																										
 <table border="1" style="margin: auto;"> <thead> <tr> <th>Hora</th> <th>Vuelo</th> <th>Origen</th> <th>Destino</th> </tr> </thead> <tbody> <tr><td>20:15</td><td>132</td><td>SEMT</td><td>SEQU</td></tr> <tr><td>19:15</td><td>133</td><td>SEQU</td><td>SEMT</td></tr> <tr><td>07:15</td><td>144</td><td>SETM</td><td>SEQU</td></tr> <tr><td>05:45</td><td>145</td><td>SEQU</td><td>SETM</td></tr> <tr><td>17:50</td><td>146</td><td>SETM</td><td>SEQU</td></tr> <tr><td>16:30</td><td>147</td><td>SEQU</td><td>SETM</td></tr> <tr><td>17:45</td><td>174</td><td>SECU</td><td>SEQU</td></tr> <tr><td>16:30</td><td>175</td><td>SEQU</td><td>SECU</td></tr> <tr><td>18:15</td><td>176</td><td>SECU</td><td>SEGU</td></tr> </tbody> </table>				Hora	Vuelo	Origen	Destino	20:15	132	SEMT	SEQU	19:15	133	SEQU	SEMT	07:15	144	SETM	SEQU	05:45	145	SEQU	SETM	17:50	146	SETM	SEQU	16:30	147	SEQU	SETM	17:45	174	SECU	SEQU	16:30	175	SEQU	SECU	18:15	176	SECU	SEGU
Hora	Vuelo	Origen	Destino																																								
20:15	132	SEMT	SEQU																																								
19:15	133	SEQU	SEMT																																								
07:15	144	SETM	SEQU																																								
05:45	145	SEQU	SETM																																								
17:50	146	SETM	SEQU																																								
16:30	147	SEQU	SETM																																								
17:45	174	SECU	SEQU																																								
16:30	175	SEQU	SECU																																								
18:15	176	SECU	SEGU																																								

Tabla 4.24 Prototipo – Bloque Matriz de Vuelos.

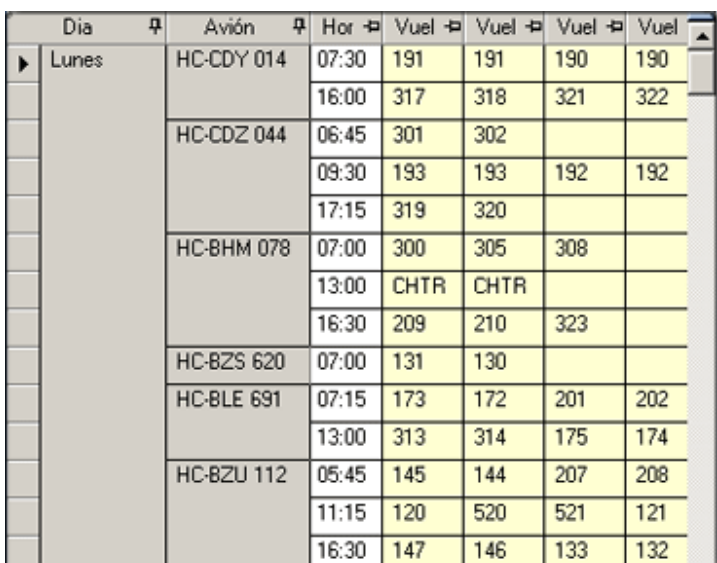
PROTOTIPO		BLOQUE MATRIZ DE VUELOS																																																																																													
Código:	PT 6.0	Caso de Uso Asociado:	CU 4.0																																																																																												
Descripción:	Presenta una Matriz de Vuelos con la que se podría semanalmente realizar una copia y modificarla dependiendo de la operación.																																																																																														
 <table border="1" style="margin: auto;"> <thead> <tr> <th>Día</th> <th>Avión</th> <th>Hor</th> <th>Vuel</th> <th>Vuel</th> <th>Vuel</th> <th>Vuel</th> </tr> </thead> <tbody> <tr><td rowspan="2">Lunes</td><td rowspan="2">HC-CDY 014</td><td>07:30</td><td>191</td><td>191</td><td>190</td><td>190</td></tr> <tr><td>16:00</td><td>317</td><td>318</td><td>321</td><td>322</td></tr> <tr><td rowspan="3"></td><td rowspan="3">HC-CDZ 044</td><td>06:45</td><td>301</td><td>302</td><td></td><td></td></tr> <tr><td>09:30</td><td>193</td><td>193</td><td>192</td><td>192</td></tr> <tr><td>17:15</td><td>319</td><td>320</td><td></td><td></td></tr> <tr><td rowspan="3"></td><td rowspan="3">HC-BHM 078</td><td>07:00</td><td>300</td><td>305</td><td>308</td><td></td></tr> <tr><td>13:00</td><td>CHTR</td><td>CHTR</td><td></td><td></td></tr> <tr><td>16:30</td><td>209</td><td>210</td><td>323</td><td></td></tr> <tr><td></td><td>HC-BZS 620</td><td>07:00</td><td>131</td><td>130</td><td></td><td></td></tr> <tr><td rowspan="2"></td><td rowspan="2">HC-BLE 691</td><td>07:15</td><td>173</td><td>172</td><td>201</td><td>202</td></tr> <tr><td>13:00</td><td>313</td><td>314</td><td>175</td><td>174</td></tr> <tr><td rowspan="3"></td><td rowspan="3">HC-BZU 112</td><td>05:45</td><td>145</td><td>144</td><td>207</td><td>208</td></tr> <tr><td>11:15</td><td>120</td><td>520</td><td>521</td><td>121</td></tr> <tr><td>16:30</td><td>147</td><td>146</td><td>133</td><td>132</td></tr> </tbody> </table>							Día	Avión	Hor	Vuel	Vuel	Vuel	Vuel	Lunes	HC-CDY 014	07:30	191	191	190	190	16:00	317	318	321	322		HC-CDZ 044	06:45	301	302			09:30	193	193	192	192	17:15	319	320				HC-BHM 078	07:00	300	305	308		13:00	CHTR	CHTR			16:30	209	210	323			HC-BZS 620	07:00	131	130				HC-BLE 691	07:15	173	172	201	202	13:00	313	314	175	174		HC-BZU 112	05:45	145	144	207	208	11:15	120	520	521	121	16:30	147	146	133	132
Día	Avión	Hor	Vuel	Vuel	Vuel	Vuel																																																																																									
Lunes	HC-CDY 014	07:30	191	191	190	190																																																																																									
		16:00	317	318	321	322																																																																																									
	HC-CDZ 044	06:45	301	302																																																																																											
		09:30	193	193	192	192																																																																																									
		17:15	319	320																																																																																											
	HC-BHM 078	07:00	300	305	308																																																																																										
		13:00	CHTR	CHTR																																																																																											
		16:30	209	210	323																																																																																										
	HC-BZS 620	07:00	131	130																																																																																											
	HC-BLE 691	07:15	173	172	201	202																																																																																									
		13:00	313	314	175	174																																																																																									
	HC-BZU 112	05:45	145	144	207	208																																																																																									
		11:15	120	520	521	121																																																																																									
		16:30	147	146	133	132																																																																																									

Tabla 4.25 Prototipo – Bloque Semanal de Vuelos.

PROTOTIPO		BLOQUE SEMANAL DE VUELOS																																														
Código:	PT 7.0	Caso de Uso Asociado:	CU 4.0																																													
Descripción:	Presenta el Bloque de Vuelos Semanal en el cual se indican los vuelos, rutas y horas que deberá cumplir cada aeronave asignada. Estos datos se visualizarán de acuerdo a la fecha seleccionada.																																															
<div style="border: 1px solid gray; padding: 10px; margin: 10px auto; width: 80%;"> <div style="display: flex; justify-content: space-around; border-bottom: 1px solid gray;"> <input type="checkbox"/> Origen - Destino <input type="checkbox"/> Horas de Vuelo </div> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th style="width: 10%;">Dia</th> <th style="width: 15%;">Avión</th> <th style="width: 10%;">Hor</th> <th style="width: 10%;">Vuel</th> <th style="width: 10%;">Vuel</th> <th style="width: 10%;">Vuel</th> <th style="width: 10%;">Vuel</th> </tr> </thead> <tbody> <tr> <td rowspan="2">Lunes</td> <td rowspan="2">HC-CDY 014</td> <td>05:45</td> <td>145</td> <td>147</td> <td style="background-color: #f08080;">146</td> <td></td> </tr> <tr> <td>14:00</td> <td>LOG</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Martes</td> <td>HC-CEH 228</td> <td>08:30</td> <td>305</td> <td>308</td> <td>311</td> <td></td> </tr> <tr> <td rowspan="2">Miercoles</td> <td rowspan="2">HC-CDY 014</td> <td>08:45</td> <td>207</td> <td>202</td> <td></td> <td></td> </tr> <tr> <td>14:00</td> <td style="background-color: #c0ffc0;">203</td> <td style="background-color: #c0ffc0;">204</td> <td></td> <td></td> </tr> <tr> <td>Sabado</td> <td>HC-CDY 014</td> <td>05:45</td> <td>145</td> <td>144</td> <td>147</td> <td>146</td> </tr> </tbody> </table> </div>				Dia	Avión	Hor	Vuel	Vuel	Vuel	Vuel	Lunes	HC-CDY 014	05:45	145	147	146		14:00	LOG				Martes	HC-CEH 228	08:30	305	308	311		Miercoles	HC-CDY 014	08:45	207	202			14:00	203	204			Sabado	HC-CDY 014	05:45	145	144	147	146
Dia	Avión	Hor	Vuel	Vuel	Vuel	Vuel																																										
Lunes	HC-CDY 014	05:45	145	147	146																																											
		14:00	LOG																																													
Martes	HC-CEH 228	08:30	305	308	311																																											
Miercoles	HC-CDY 014	08:45	207	202																																												
		14:00	203	204																																												
Sabado	HC-CDY 014	05:45	145	144	147	146																																										

Tabla 4.26 Prototipo – Administración de Vuelos.

PROTOTIPO		ADMINISTRACIÓN DE VUELOS	
Código:	PT 8.0	Caso de Uso Asociado:	CU 4.0, 5.0
Descripción:	Permite administrar los vuelos del bloque semanal		
<div style="border: 1px solid gray; padding: 10px; margin: 10px auto; width: 80%;"> <div style="display: flex; justify-content: space-between; border-bottom: 1px solid gray;"> <input type="checkbox"/> Vuelo Cancelado <input checked="" type="checkbox"/> Registrar Cambio </div> <div style="margin-bottom: 5px;"> Día: <input type="text" value="Martes"/> </div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> Equipo: <input type="text" value="F-28"/> Avión: <input type="text" value="HC-CEH 228"/> </div> <div style="margin-bottom: 5px;"> Vuelo: <input type="text" value="TRAS"/> <input style="background-color: #000080; color: white;" type="text" value="TRASLADO"/> </div> <div style="display: flex; justify-content: center; margin-bottom: 5px;"> <input checked="" type="radio"/> Nacional <input type="radio"/> Internacional </div> <div style="display: flex; justify-content: space-between; margin-bottom: 5px;"> Hora: <input type="text" value="08:30"/> Origen: <input type="text" value="SEGU"/> Destino: <input type="text" value="SEGU"/> Millas: <input type="text" value="180"/> Tiempo: <input type="text" value="00:30"/> </div> <div style="margin-bottom: 5px;"> Tipo de Operación: <input checked="" type="radio"/> Doméstica <input type="radio"/> No Regular <input type="radio"/> Internacional </div> <div style="margin-bottom: 5px;"> Empresa: <input type="text" value="EMBRAER"/> </div> <div style="margin-bottom: 5px;"> Observaciones: <input style="width: 100%;" type="text"/> </div> </div>			

Tabla 4.27 Prototipo – Vuelos Fuera de Itinerario.

PROTOTIPO		VUELOS FUERA DE ITINERARIO	
Código:	PT 9.0	Caso de Uso Asociado:	CU 5.0
Descripción:	Con esta interfaz se puede crear vuelos fuera de itinerario y asignarlos al bloque semanal.		
<p>The screenshot shows a form for creating a flight. It includes dropdown menus for 'Día' (Lunes), 'Equipo' (A 320), 'Avión' (HC-CDY 014), and 'Vuelo' (PRESIDENCIAL). There are radio buttons for 'Nacional' (selected) and 'Internacional'. Below, there are input fields for 'Hora' (14:00), 'Origen' (SEGU), 'Destino' (SEGU), 'Millas' (_180), and 'Tiempo' (00:30). At the bottom, there are radio buttons for 'Tipo de Operación' (Doméstica, No Regular (selected), Internacional) and a dropdown for 'Empresa' (D CAMERUM).</p>			

Tabla 4.28 Prototipo – Cuadro del Bloque Semanal.

PROTOTIPO		CUADRO DEL BLOQUE SEMANAL						
Código:	PT 10.0	Caso de Uso Asociado:	CU 4.0					
Descripción:	Presenta un cuadro con todos los vuelos que se deberá cumplir en la semana por parte de la Empresa.							
<p>The screenshot shows a weekly flight schedule interface. It has checkboxes for 'Origen - Destino' and 'Horas de Vuelo'. Below are checkboxes for each day of the week, all of which are checked: Lunes, Martes, Miércoles, Jueves, Viernes, Sábado, and Domingo. The main part of the interface is a table with columns for 'Avión', 'Hora', and 'Vuelo' for each day. The first row shows flights for Monday: HC-CDY 014 at 05:45 (flights 145, 147, 146), and HC-CDY 014 at 14:00 (flight LOG). The second row shows flights for Tuesday: HC-CEH 228 at 08:30 (flights 305, 308) and HC-CDZ 044. The bottom part of the interface is a large blue shaded area.</p>								

Tabla 4.29 Prototipo – Itinerarios de Vuelo Mensual.

PROTOTIPO	ITINERARIOS DE VUELOS MENSUAL																																																																	
Código:	PT 11.0	Caso de Uso Asociado:	CU 6.0																																																															
Descripción:	Cada tripulante cuenta con una calendario mensual en el cual esta reflejado los vuelos a cumplir, entrenamiento y otras actividades.																																																																	
<table border="1"> <thead> <tr> <th>lunes</th> <th>martes</th> <th>miércoles</th> <th>jueves</th> <th>viernes</th> <th>sábado</th> <th>domingo</th> </tr> </thead> <tbody> <tr> <td>8 de enero</td> <td>9</td> <td>10</td> <td>11</td> <td>12</td> <td>13</td> <td>14</td> </tr> <tr> <td>15</td> <td>16</td> <td>17</td> <td>18</td> <td>19</td> <td>20</td> <td>21</td> </tr> <tr> <td>145 - 147</td> <td>[A] LIBRE</td> <td>[E] Entrena</td> <td>[A] PERMIS</td> <td></td> <td></td> <td></td> </tr> <tr> <td>22</td> <td>23</td> <td>24</td> <td>25</td> <td>26</td> <td>27</td> <td>28</td> </tr> <tr> <td></td> <td>[A] CEMAC</td> <td>[A] LIBRE</td> <td>[A] LIBRE</td> <td></td> <td></td> <td></td> </tr> <tr> <td>29</td> <td>30</td> <td>31</td> <td>1 de feb de 07</td> <td>2</td> <td>3</td> <td>4</td> </tr> <tr> <td>5</td> <td>6</td> <td>7</td> <td>8</td> <td>9</td> <td>10</td> <td>11</td> </tr> <tr> <td>[E] Adoctrin</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>				lunes	martes	miércoles	jueves	viernes	sábado	domingo	8 de enero	9	10	11	12	13	14	15	16	17	18	19	20	21	145 - 147	[A] LIBRE	[E] Entrena	[A] PERMIS				22	23	24	25	26	27	28		[A] CEMAC	[A] LIBRE	[A] LIBRE				29	30	31	1 de feb de 07	2	3	4	5	6	7	8	9	10	11	[E] Adoctrin						
lunes	martes	miércoles	jueves	viernes	sábado	domingo																																																												
8 de enero	9	10	11	12	13	14																																																												
15	16	17	18	19	20	21																																																												
145 - 147	[A] LIBRE	[E] Entrena	[A] PERMIS																																																															
22	23	24	25	26	27	28																																																												
	[A] CEMAC	[A] LIBRE	[A] LIBRE																																																															
29	30	31	1 de feb de 07	2	3	4																																																												
5	6	7	8	9	10	11																																																												
[E] Adoctrin																																																																		

Tabla 4.30 Prototipo – Itinerarios de Vuelo Semanal.

PROTOTIPO	ITINERARIOS DE VUELO SEMANAL																	
Código:	PT 12.0	Caso de Uso Asociado: CU 6.0																
Descripción:	Es otra forma de visualizar los vuelos que debe cumplir un tripulante.																	
<table border="1"> <tbody> <tr> <td>lunes, 15 de enero</td> <td>jueves, 18 de enero</td> </tr> <tr> <td>145 - 147</td> <td>[A] PERMISO ESPECIAL</td> </tr> <tr> <td>martes, 16 de enero</td> <td>viernes, 19 de enero</td> </tr> <tr> <td>[A] LIBRE</td> <td></td> </tr> <tr> <td>miércoles, 17 de enero</td> <td>sábado, 20 de enero</td> </tr> <tr> <td>[E] Entrenamiento de Vuelo (14:00-16:00)</td> <td></td> </tr> <tr> <td></td> <td>domingo, 21 de enero</td> </tr> <tr> <td></td> <td></td> </tr> </tbody> </table>			lunes, 15 de enero	jueves, 18 de enero	145 - 147	[A] PERMISO ESPECIAL	martes, 16 de enero	viernes, 19 de enero	[A] LIBRE		miércoles, 17 de enero	sábado, 20 de enero	[E] Entrenamiento de Vuelo (14:00-16:00)			domingo, 21 de enero		
lunes, 15 de enero	jueves, 18 de enero																	
145 - 147	[A] PERMISO ESPECIAL																	
martes, 16 de enero	viernes, 19 de enero																	
[A] LIBRE																		
miércoles, 17 de enero	sábado, 20 de enero																	
[E] Entrenamiento de Vuelo (14:00-16:00)																		
	domingo, 21 de enero																	

Tabla 4.31 Prototipo – Asignación de Vuelos.


PROTOTIPO		ASIGNACION DE VUELOS	
Código:	PT 13.0	Caso de Uso Asociado:	CU 6.0
Descripción:	Permite asignar vuelos a los tripulantes.		
			

Tabla 4.32 Prototipo – Lista de Tripulación.

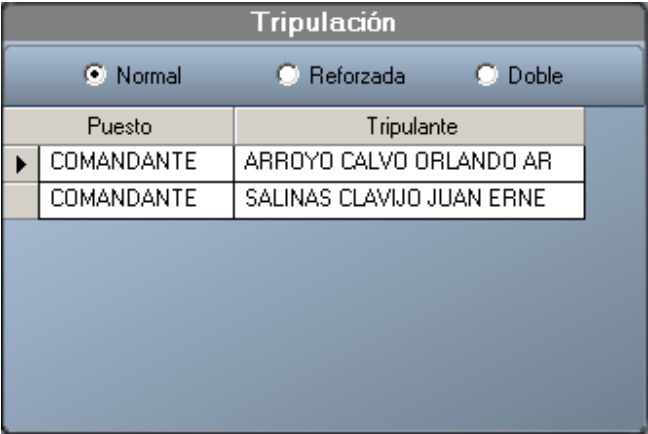
PROTOTIPO		LISTA DE TRIPULACION	
Código:	PT 14.0	Caso de Uso Asociado:	CU 6.0
Descripción:	Presenta la lista de tripulación mayor y menor de un vuelo.		
			

Tabla 4.33 Prototipo – Otras Actividades.

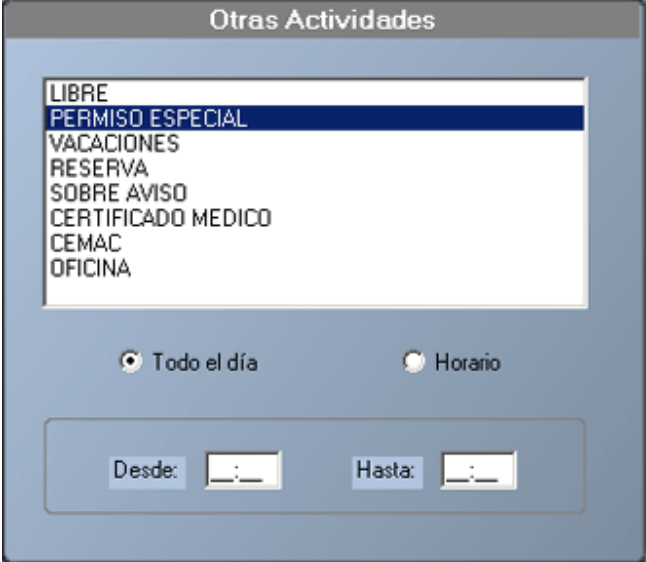
PROTOTIPO		OTRAS ACTIVIDADES	
Código:	PT 15.0	Caso de Uso Asociado:	CU 7.0
Descripción:	Permite asignar otro tipo de actividades que un tripulante puede cumplir en el transcurso del día.		
			

Tabla 4.34 Prototipo – Publicación de Itinerarios de Vuelo.

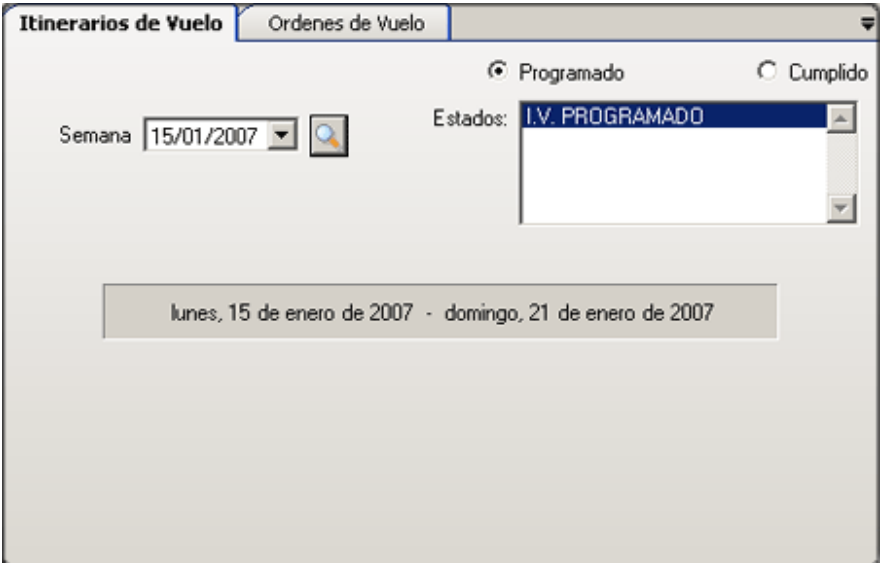
PROTOTIPO		PUBLICACION DE ITINERARIOS DE VUELO	
Código:	PT 16.0	Caso de Uso Asociado:	CU 9.0
Descripción:	Publica los Itinerarios de Vuelo semanal. Estos datos son registrados en un histórico de vuelos.		
			

Tabla 4.35 Prototipo – Publicación de Órdenes de Vuelo.

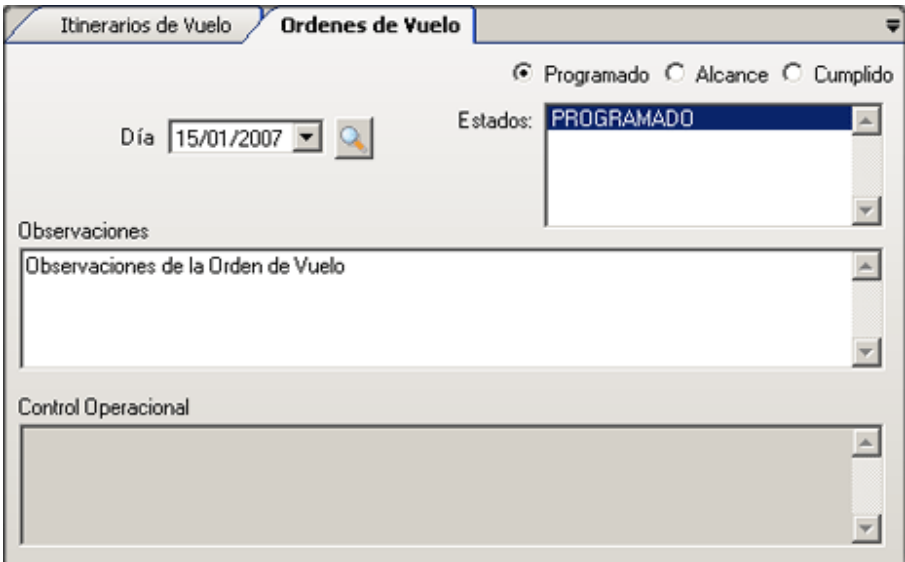
PROTOTIPO		PUBLICACION DE ORDENES DE VUELO	
Código:	PT 17.0	Caso de Uso Asociado:	CU 8.0, 9.0
Descripción:	Publica las Órdenes de Vuelo. Estos datos son registrados en un histórico de vuelos.		
			

Tabla 4.36 Prototipo – Segmentos de Entrenamiento.

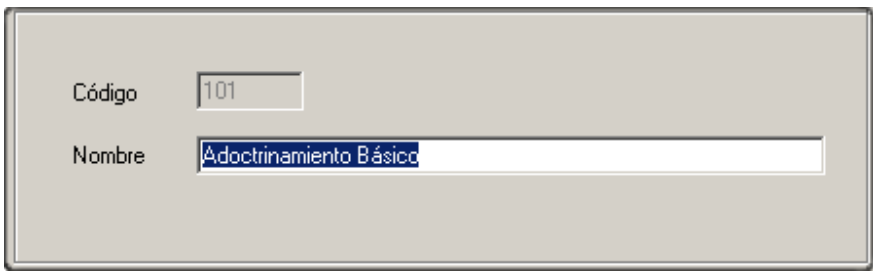
PROTOTIPO		SEGMENTOS DE ENTRENAMIENTO	
Código:	PT 18.0	Caso de Uso Asociado:	CU 10.0
Descripción:	La presente interfaz gráfica permite administrar los segmentos de entrenamiento de la tripulación.		
			

Tabla 4.37 Prototipo – Malla Curricular.

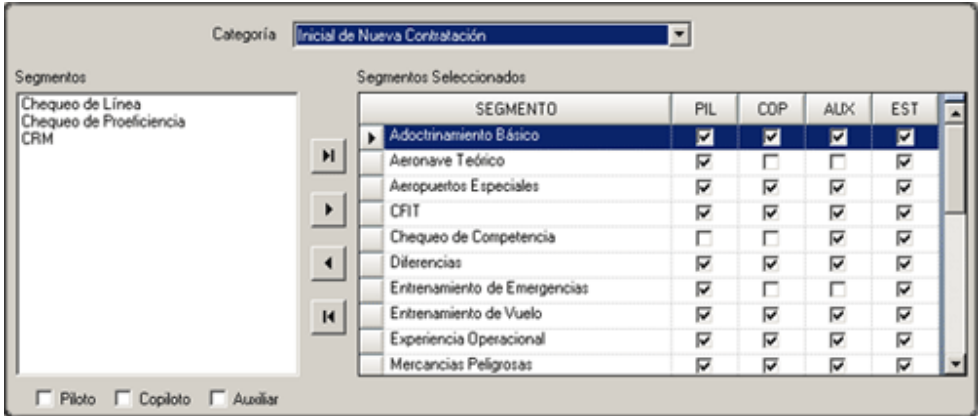
PROTOTIPO		MALLA CURRICULAR	
Código:	PT 19.0	Caso de Uso Asociado:	CU 10.0
Descripción:	En esta interfaz se crea la Malla Curricular de Entrenamiento que un tripulante debe cumplir para estar habilitado.		
			

Tabla 4.38 Prototipo – Planificación de Entrenamiento.

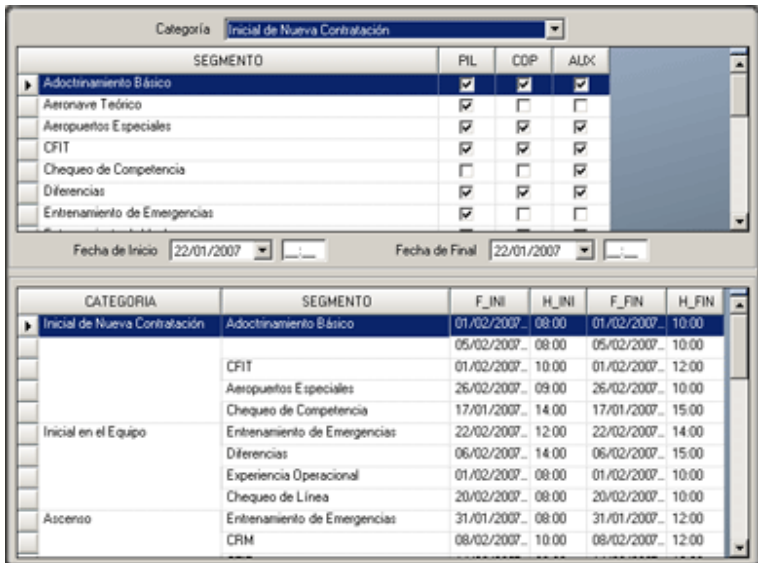
PROTOTIPO		PLANIFICACION DE ENTRENAMIENTO	
Código:	PT 20.0	Caso de Uso Asociado:	CU 11.0
Descripción:	Permite crear la Planificación Anual de Entrenamiento en la Empresa.		
			

Tabla 4.39 Prototipo – Asignación de Entrenamiento.

PROTOTIPO		ASIGNACION DE ENTRENAMIENTO			
Código:	PT 21.0	Caso de Uso Asociado:	CU 12.0		
Descripción:	A la planificación de entrenamiento se podrá asignar los tripulantes que lo cumplirán.				

CATEGORIA	SEGMENTO	F_INI	H_INI	F_FIN	H_FIN	PIL	COP	AUX
Inicial de Nueva Contrataci...	Adoctrinamiento Básico	01/02/2007...	08:00	01/02/2007...	10:00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
		05/02/2007...	08:00	05/02/2007...	10:00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	CFIT	01/02/2007...	10:00	01/02/2007...	12:00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Aeropuertos Especiales	26/02/2007...	09:00	26/02/2007...	10:00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Inicial en el Equipo	Chequeo de Competencia	17/01/2007...	14:00	17/01/2007...	15:00	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
	Entrenamiento de Emergencias	22/02/2007...	12:00	22/02/2007...	14:00	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	Diferencias	06/02/2007...	14:00	06/02/2007...	15:00	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Puesto: **Piloto**

Tripulante	Tripulante Seleccionado
ARROYO CALVO ORLANDO ARTURO	ARGUELLO CRUZ GUILLERMO
BAEZ AULESTIA CARLOS ALFREDO	CHICO ARIAS GALO GERMAN
CAMPOSANO MORLA JOSE ANTONIO	ESPINEL BURBANO CARLOS ALFONSO
CANO ANDRADE FERNANDO RUPERTO	GUERRERO RIVADENEIRA RUBEN ERNESTO
ESTRELLA VALLADARES MARCO RAMIRO	GUZMAN GUZMAN LEONARDO AUGUSTO
GARZON TRIVINO ESTEBAN	HELOU CEVALLOS JOSE MARIA
GUTIERREZ VARGAS JAIME RODRIGO	HEREDIA VILLAMAR HECTOR NELSON
LARRIVA DUENAS PATRICIO ALEJANDRO	HERNANDEZ RODRIGUEZ JORGE HERNAN
MATA VELASTEGUI ROBERTO MAURICIO	
PENAFIEL SANCHEZ LUIS GONZALO	
PENAHERRERA LIVE FAUSTO RAMIRO	
PUGA RUSSO CARLOS ENRIQUE	
REYES ZUNIGA AUGUSTO OLIVERIO	

Tabla 4.40 Prototipo – Reportes

PROTOTIPO		REPORTES	
Código:	PT 22.0	Caso de Uso Asociado:	
Descripción:	Presenta un ejemplo de los reportes que tendrá el sistema.		

Malla Curricular de Entrenamiento de Tripulantes

SEGMENTO	PILOTO	CO
Ascenso		
CFIT	*	
Chequeo de Competencia	*	
Chequeo de Línea	*	
Chequeo de Proficiencia	*	
CRM	*	
Diferencias	*	
Entrenamiento de Emergencias	*	
Inicial de Nueva Contratación		
Adoctrinamiento Básico	*	
Aeronave Teórico	*	
Aeropuertos Especiales	*	
CFIT	*	
Chequeo de Competencia	*	

Current Page No.: 1 | Total Page No.: 1+ | Zoom Factor: 100%

4.2 ESPECIFICACIONES TECNICAS

4.2.1 Herramientas Utilizadas

Las herramientas a utilizar en el proyecto esta de acuerdo a normas establecidas por la empresa TAME y son las siguientes:

- Para el desarrollo de la aplicación se utilizará el siguiente software:
 - Microsoft Visual Studio .NET 2005
 - Microsoft Enterprise Library 2005
 - Microsoft Source Safe 2005
 - Componentes Infragistics 2005
 - Base de Datos: DB2, SQL Server 2005 (Para la presentación del proyecto)
- Para creación de la documentación se utilizará:
 - Microsoft Office 2003

4.2.2 Diagramas de la Arquitectura

En todo proyecto informático es importante definir la arquitectura de la aplicación con la finalidad de establecer una base para la posterior codificación.

4.2.2.1 Arquitectura de la Aplicación

En las figuras 4.14 y 4.15 se describen los Diagramas de Despliegue y Componentes que integran la Aplicación Distribuida desarrollada en Microsoft Visual Studio .NET 2005.

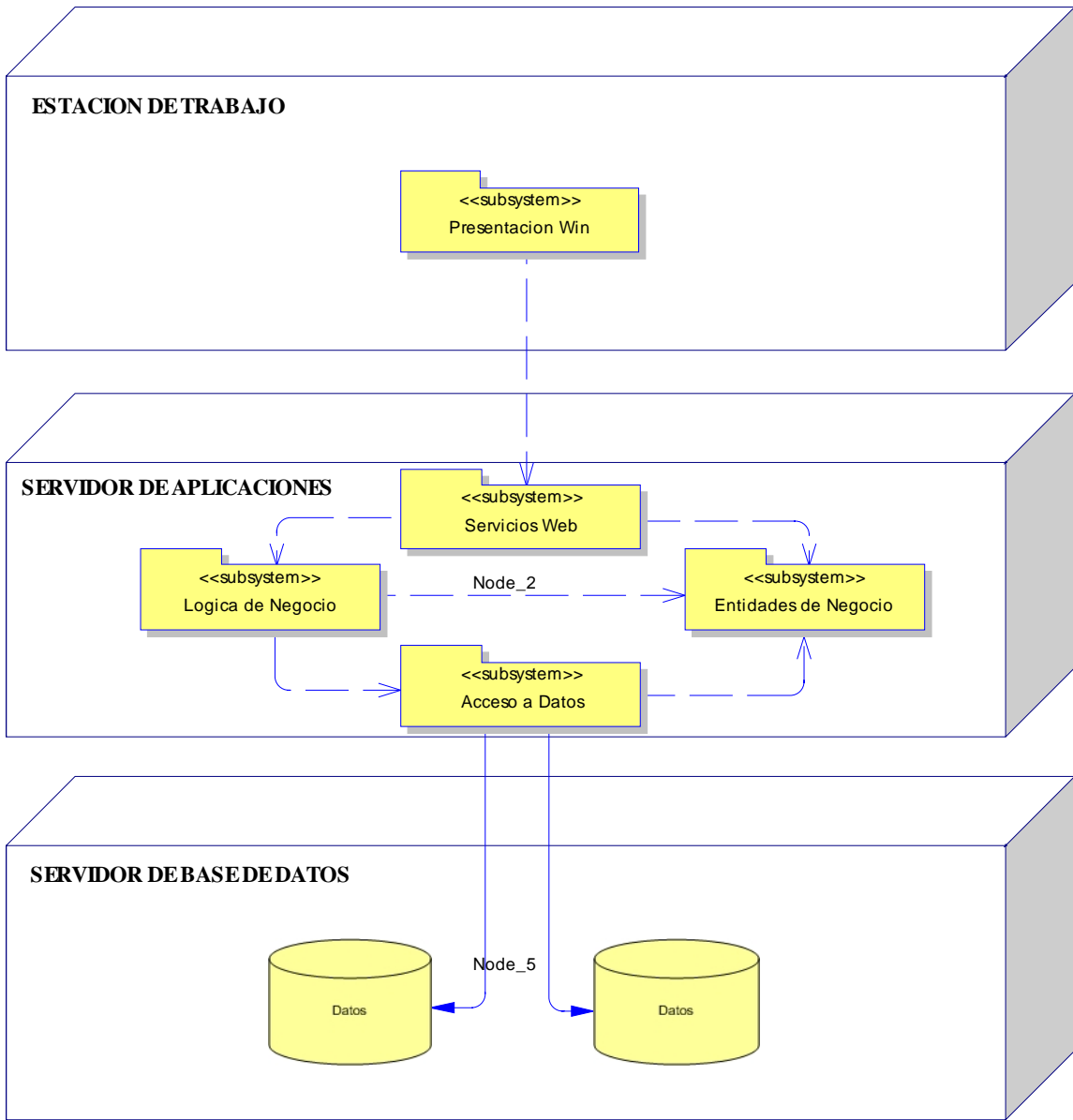


Figura 4.14 Diagrama de Despliegue.

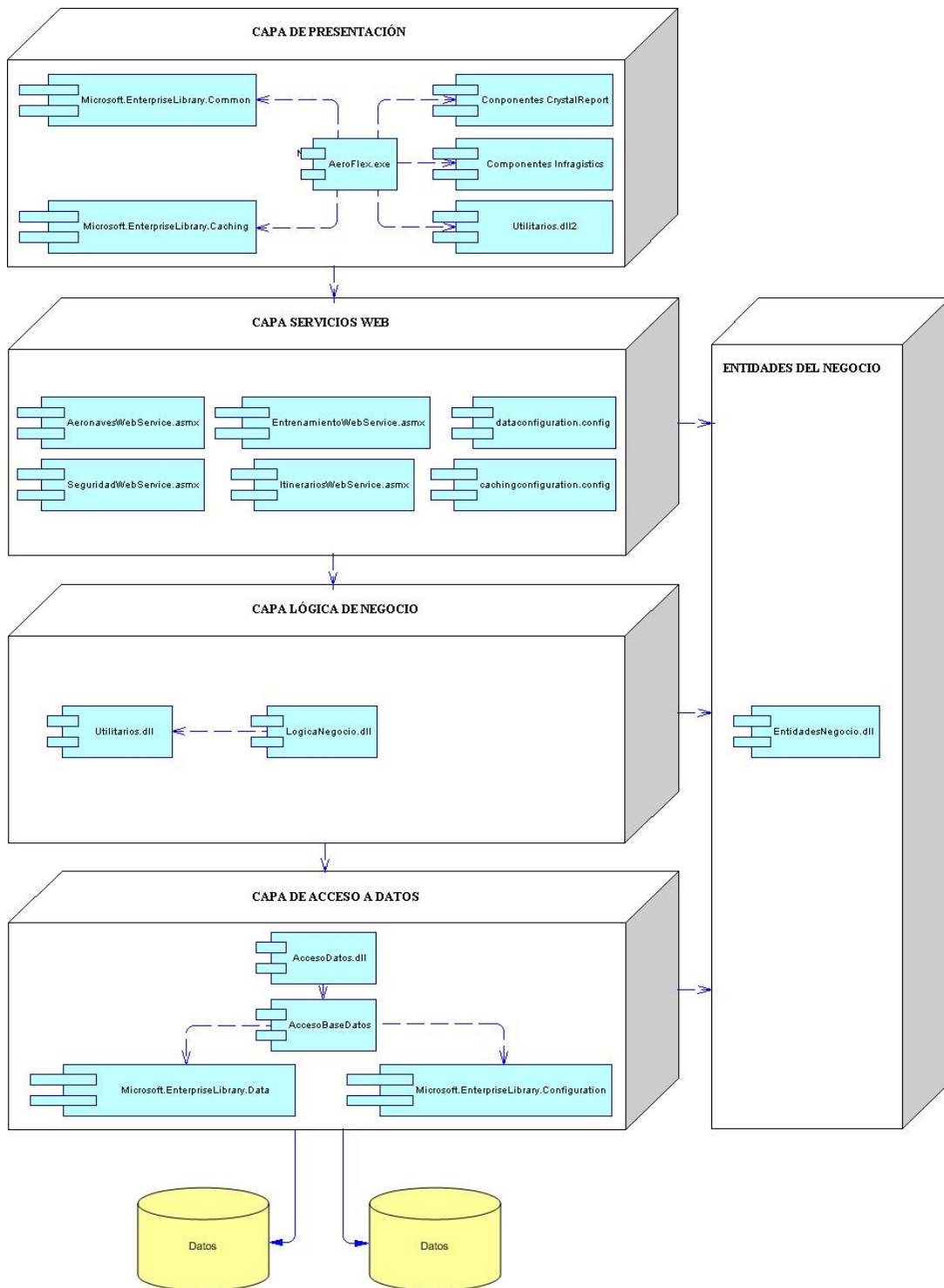


Figura 4.15 Diagrama de Componentes.

4.2.2.2 Arquitectura Orientada a Servicios

“AeroFlex” es una Aplicación Distribuida orientada a servicios como se indica en la figura 4.16. Esto permite la creación de Servicios Web con la finalidad de exponer la Lógica de Negocio por medio del Internet y que sean consumidos desde cualquier plataforma o dispositivo.

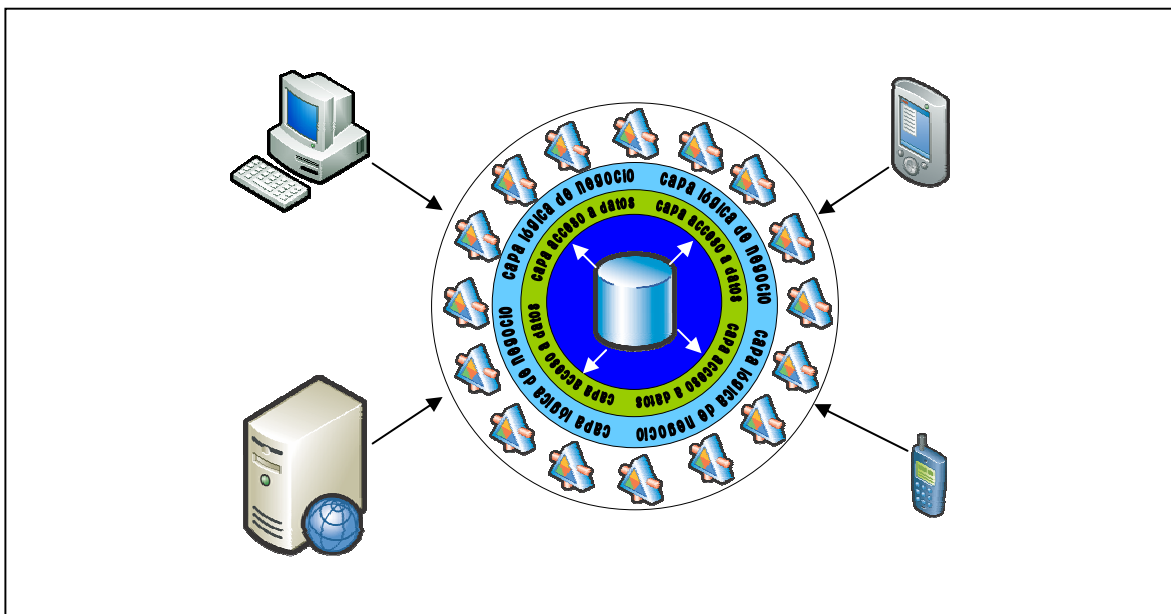


Figura 4.16 Arquitectura Orientada a Servicios.

4.2.2.3 Arquitectura Interna de un Módulo

Cada módulo de la aplicación consta de las siguientes capas: Presentación, Servicios Web, Lógica de Negocio, Acceso a Datos y Entidades de Negocio como se indica en la figura 4.17.

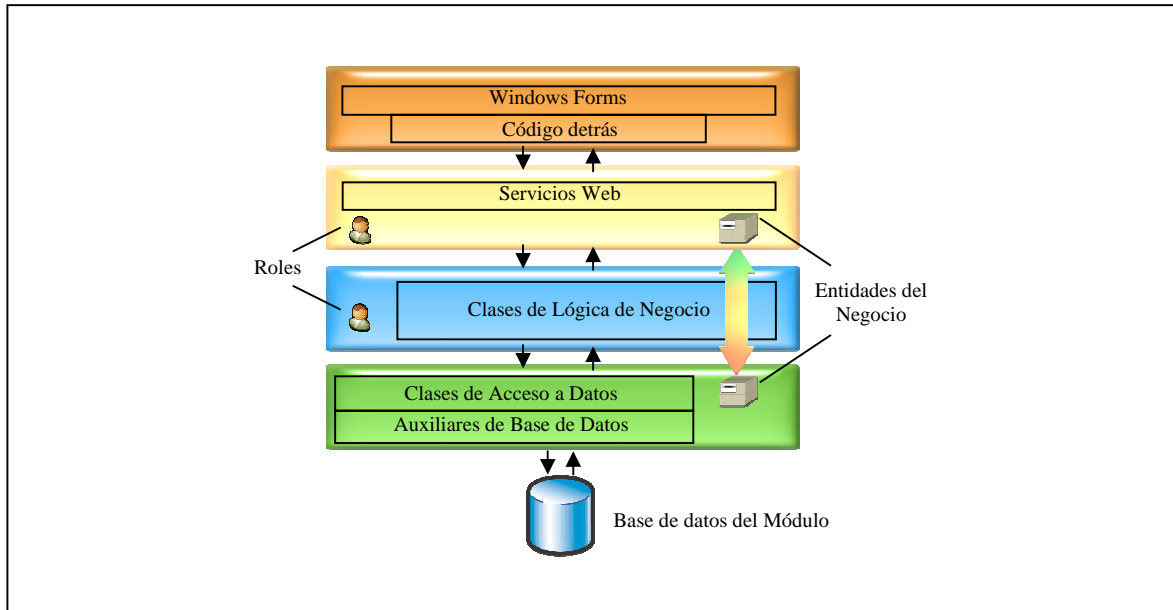


Figura 4.17 Arquitectura Interna de un Módulo.

4.2.3 Infraestructura Física

Según los diferentes servidores de Base de Datos y de Aplicaciones con los que cuenta la Empresa, en las figuras 4.18 y 4.19 se presenta la infraestructura física, indicando la conexión a los servidores y la comunicación en su intranet.

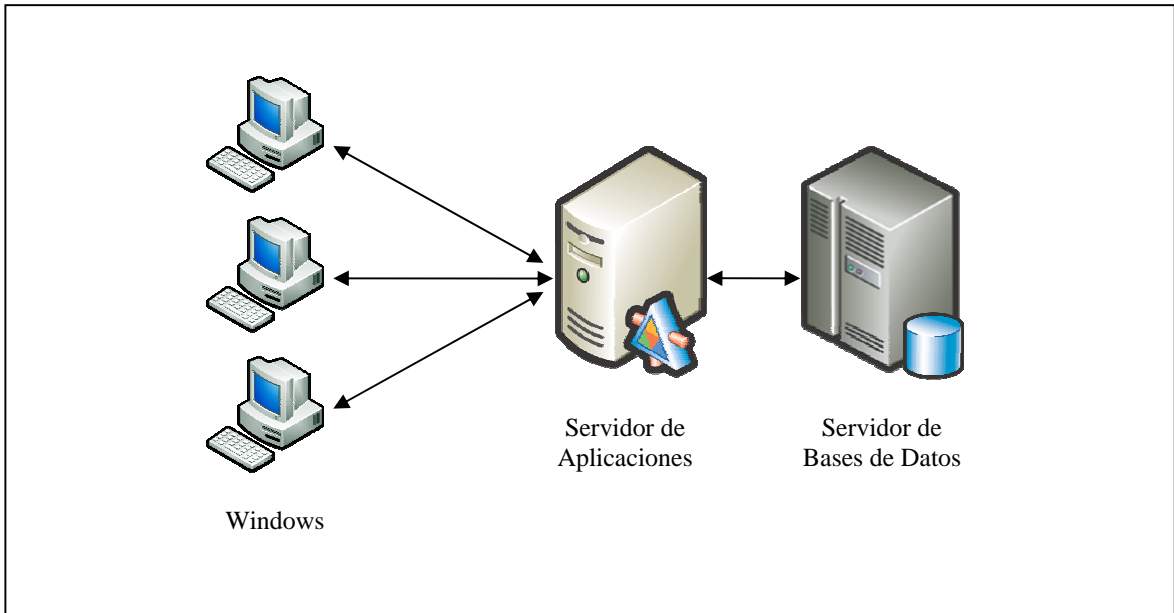


Figura 4.18 Infraestructura Física 1.

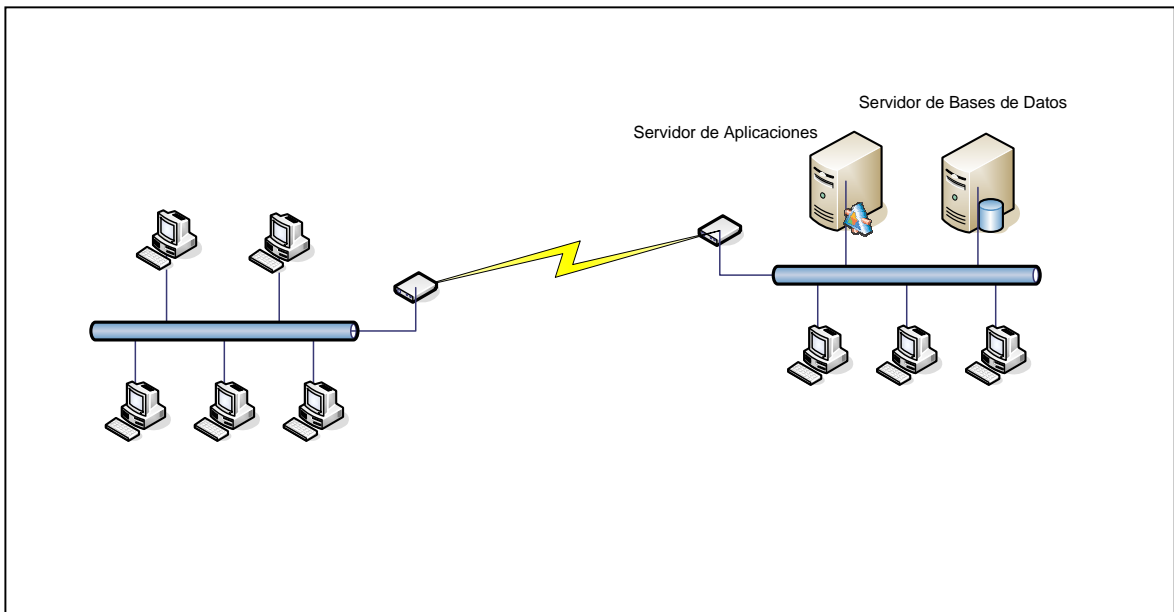


Figura 4.19 Infraestructura Física 2.

4.2.4 Diseño Lógico y Físico

Para dejar establecido las diferentes Clases que integran cada Capa de la aplicación, a continuación se describe los Diagramas de Clases y Entidades:

4.2.4.1 Capa de Servicios

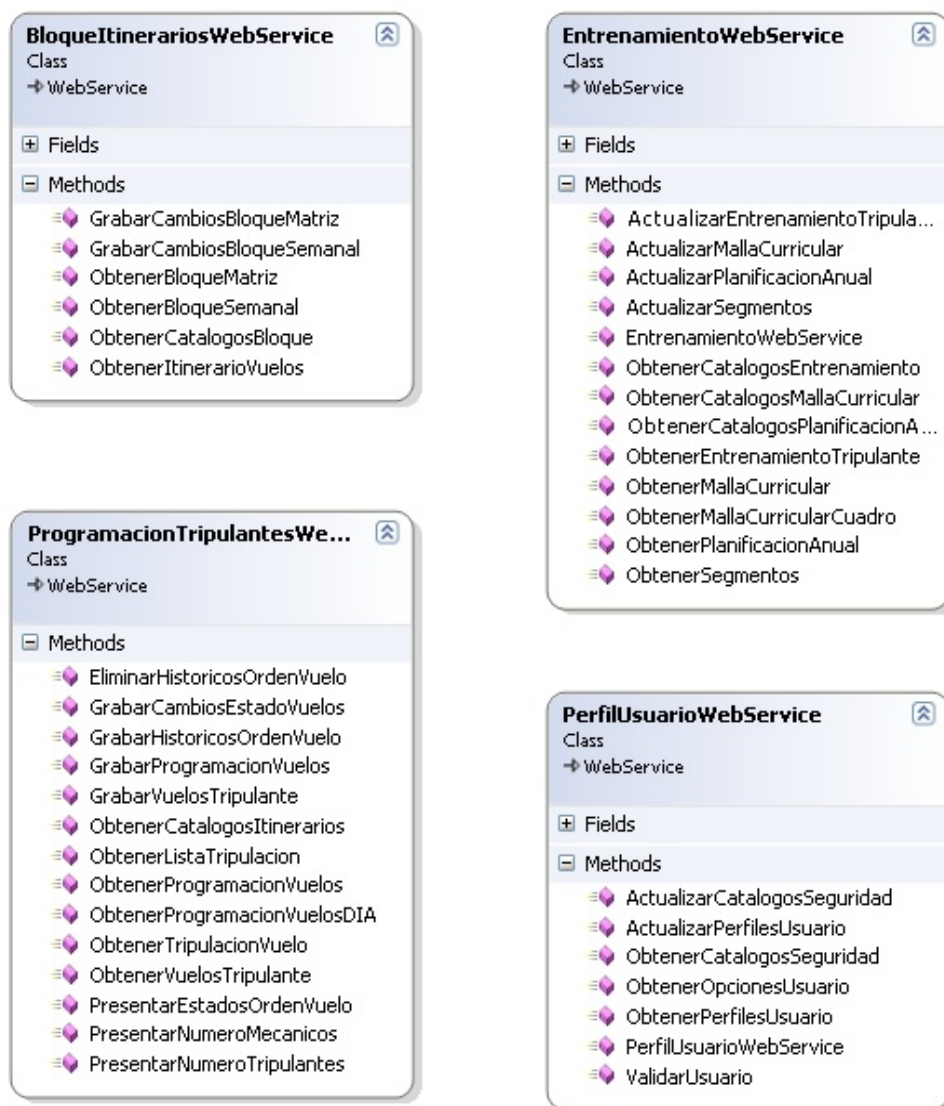


Figura 4.20 Diagrama de Clases - Capa de Servicios

4.2.4.2 Capa de Lógica de Negocio



Figura 4.21 Diagrama de Clases - Capa de Lógica de Negocio

4.2.4.3 Capa de Acceso a Datos

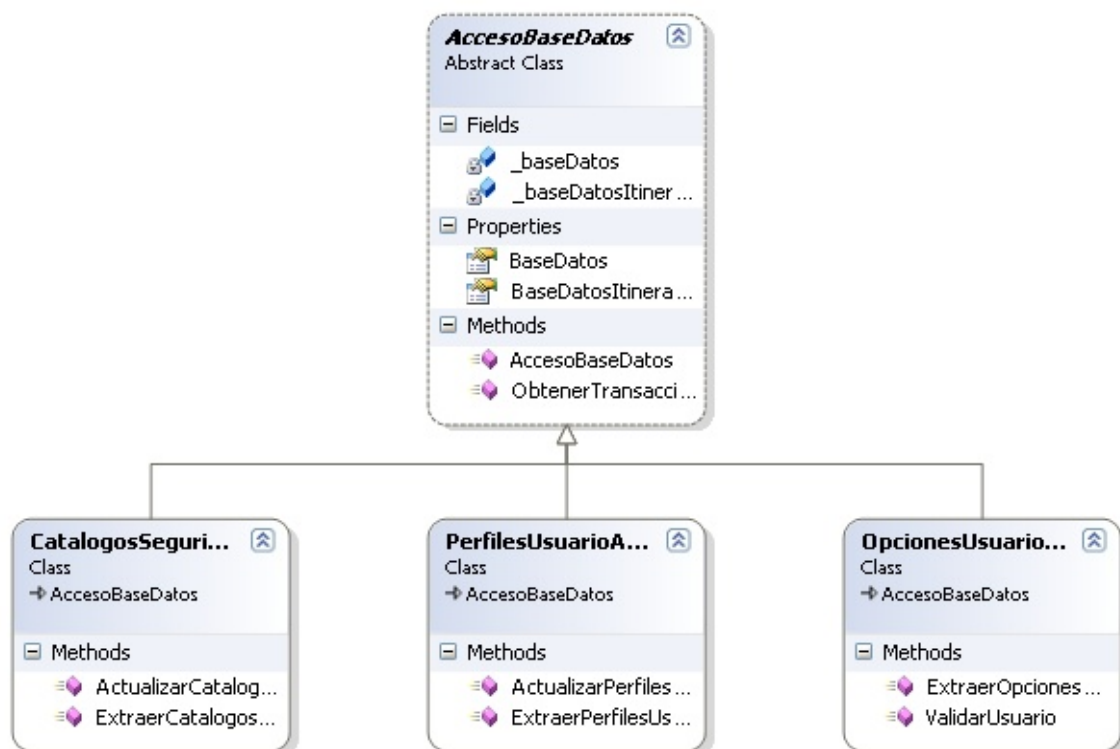


Figura 4.22 Diagrama de Clases - Capa de Acceso a Datos – Seguridad

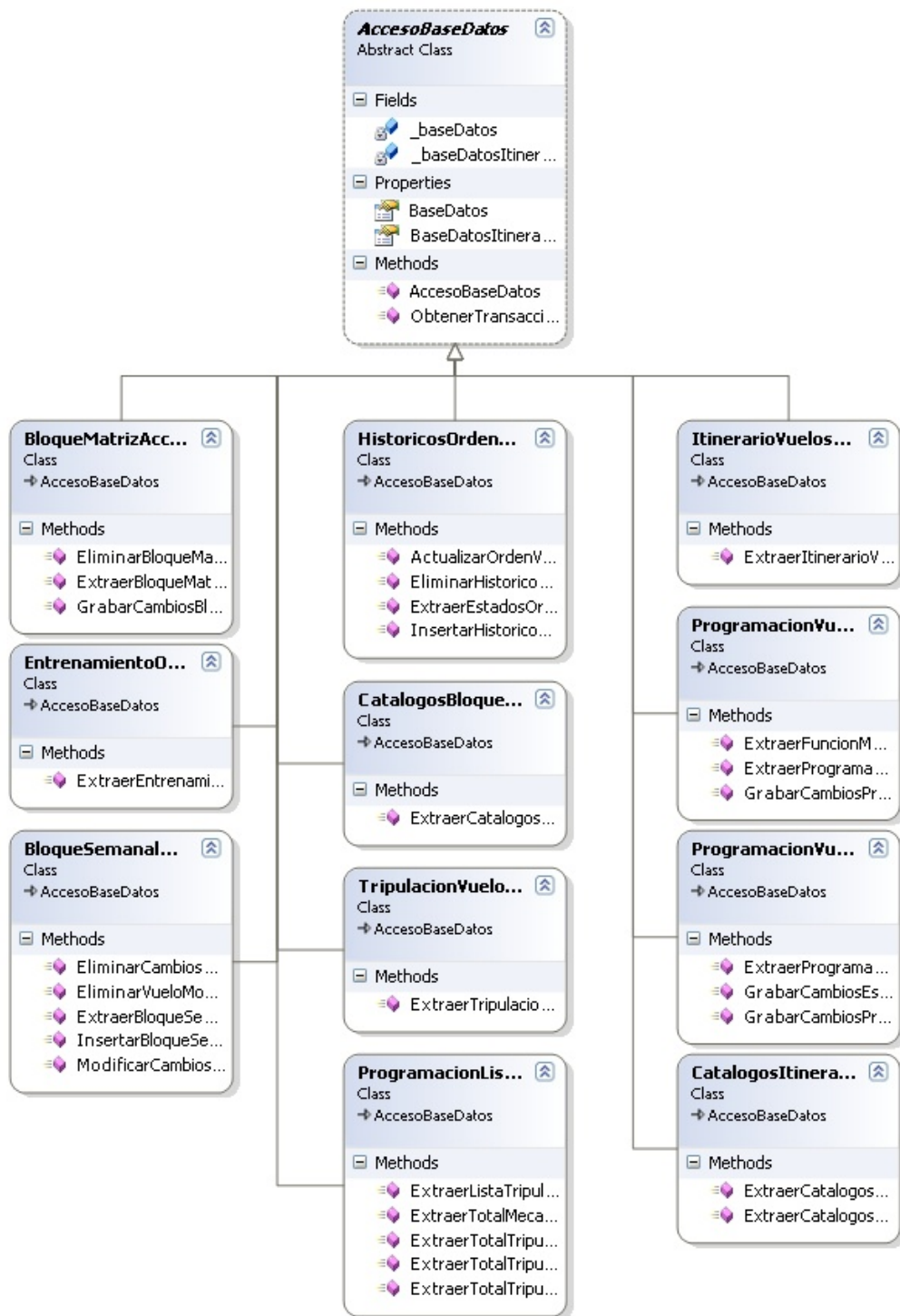


Figura 4.23 Diagrama de Clases - Capa de Acceso a Datos – Itinerarios de Vuelo.

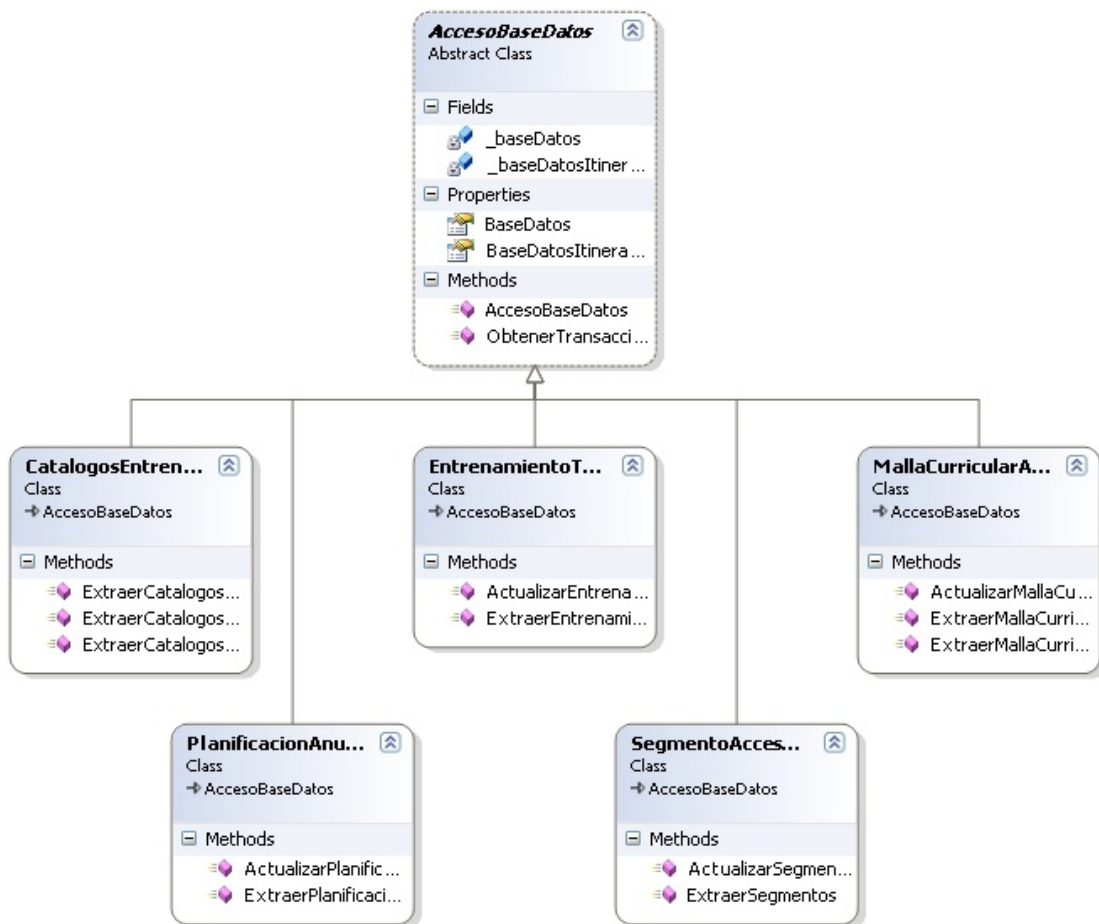


Figura 4.24 Diagrama de Clases - Capa de Acceso a Datos – Entrenamiento de Tripulantes

4.2.4.4 Entidades del Negocio

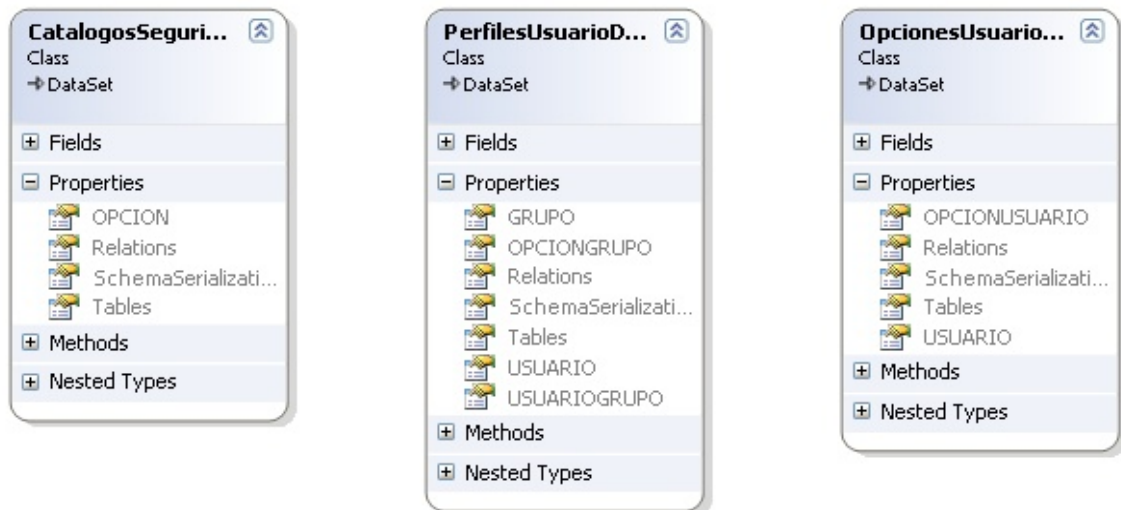


Figura 4.25 Diagrama de Clases – Entidades del Negocio – Seguridad

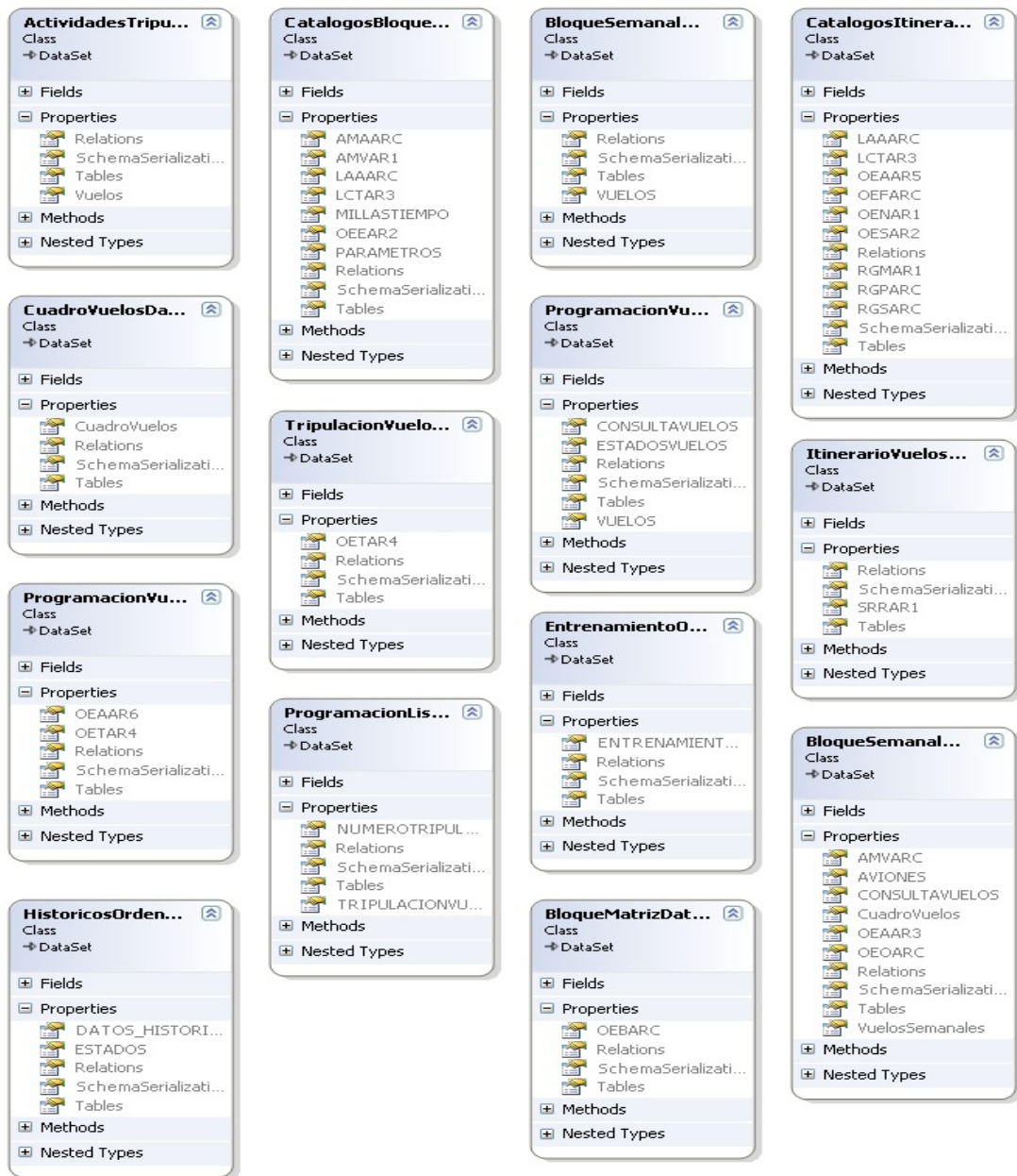


Figura 4.26 Diagrama de Clases – Entidades del Negocio – Itinerarios de Vuelo

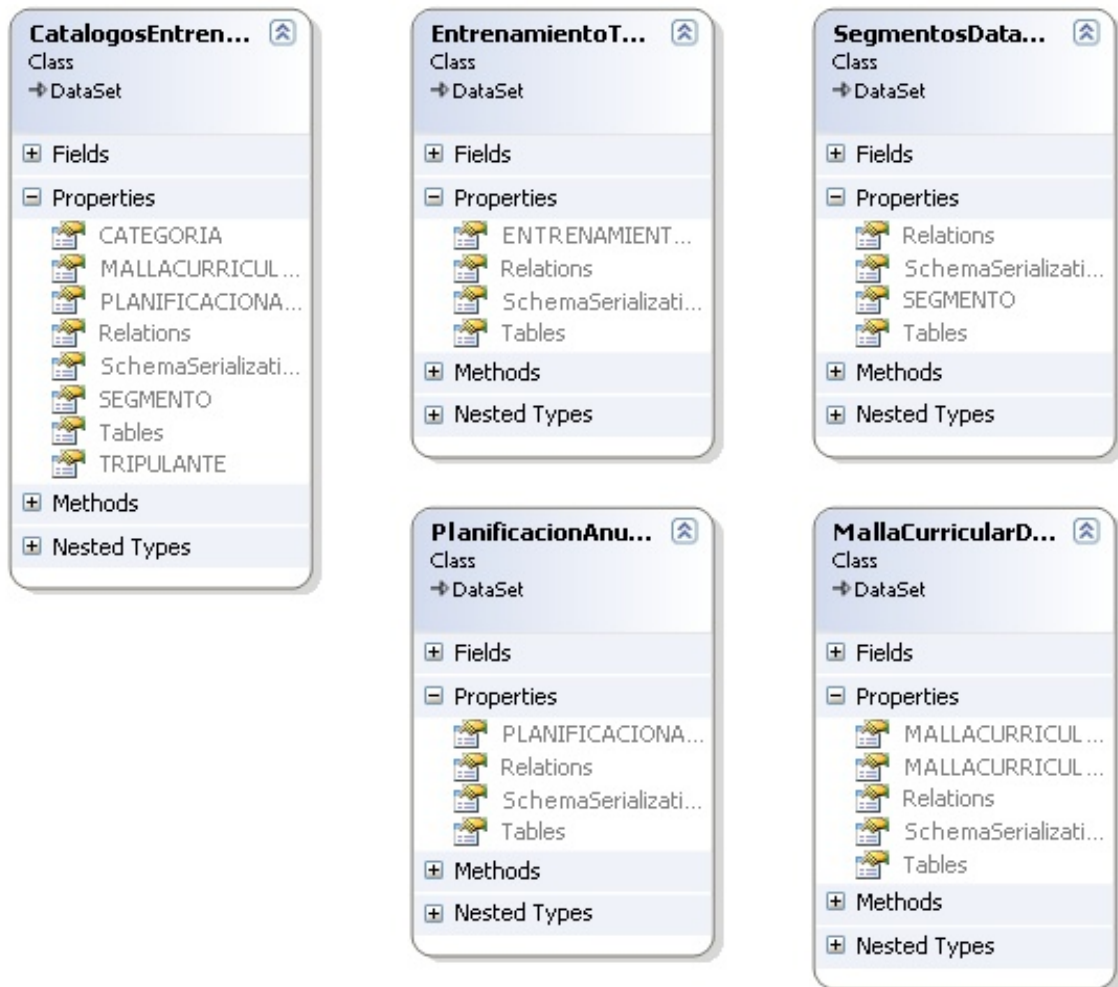


Figura 4.27 Diagrama de Clases – Entidades del Negocio – Entrenamiento de Tripulantes

4.2.4.5 Capa de Base de Datos

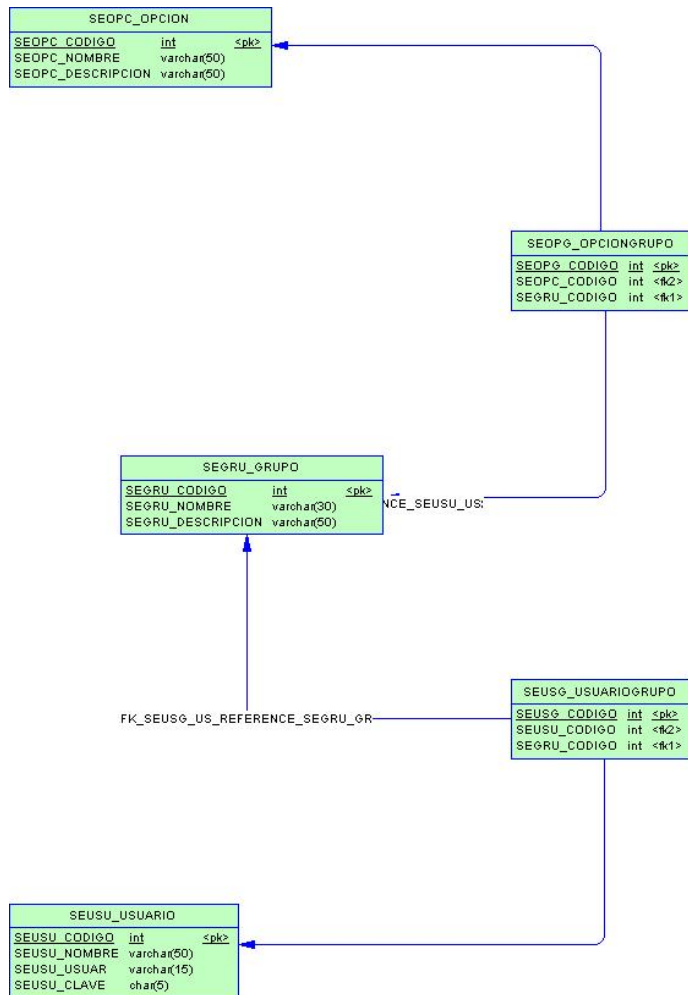


Figura 4.28 Base de Datos – Modelo de Seguridad

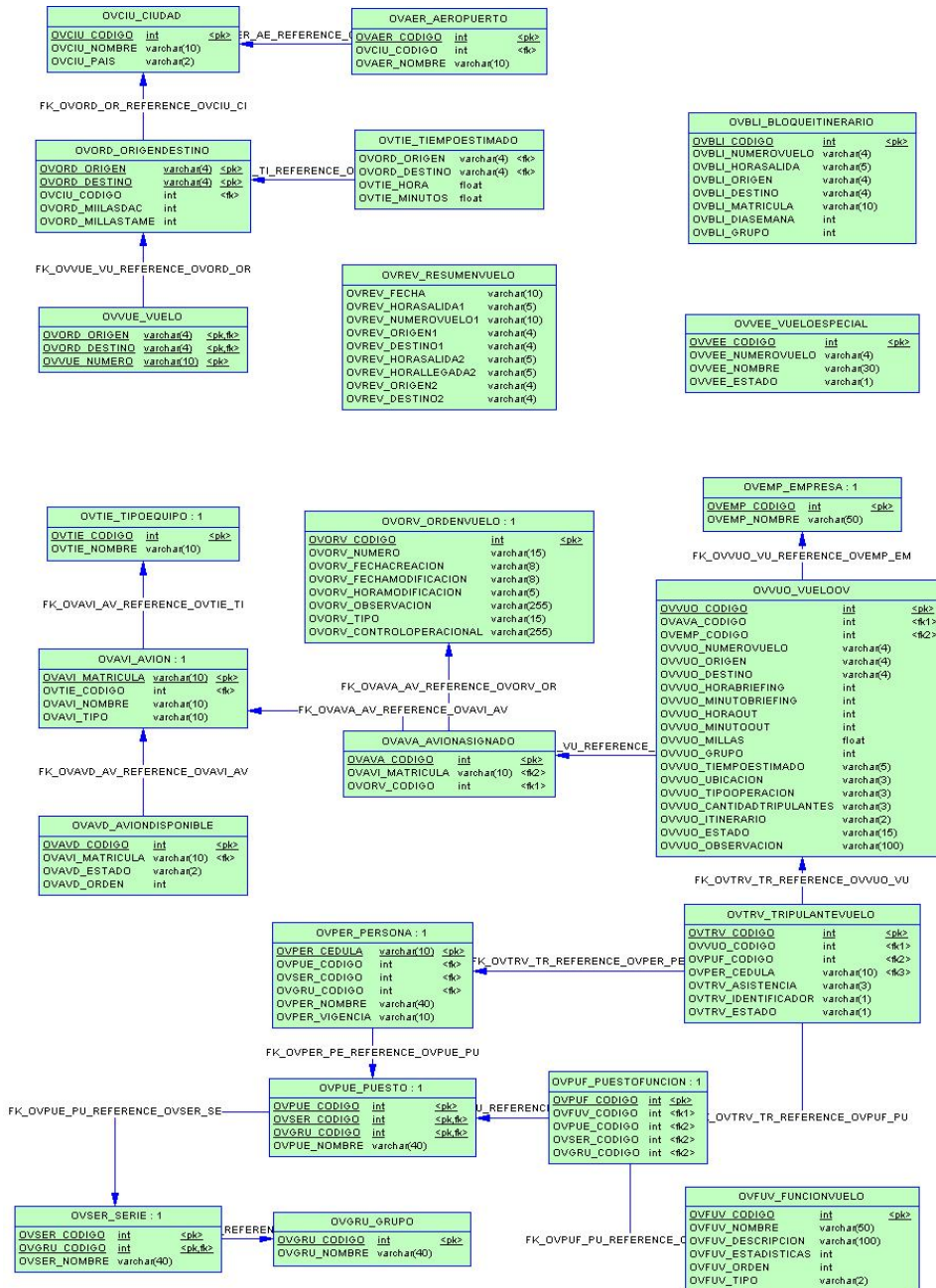


Figura 4.29 Base de Datos – Modelo de Itinerarios de Vuelo 1.

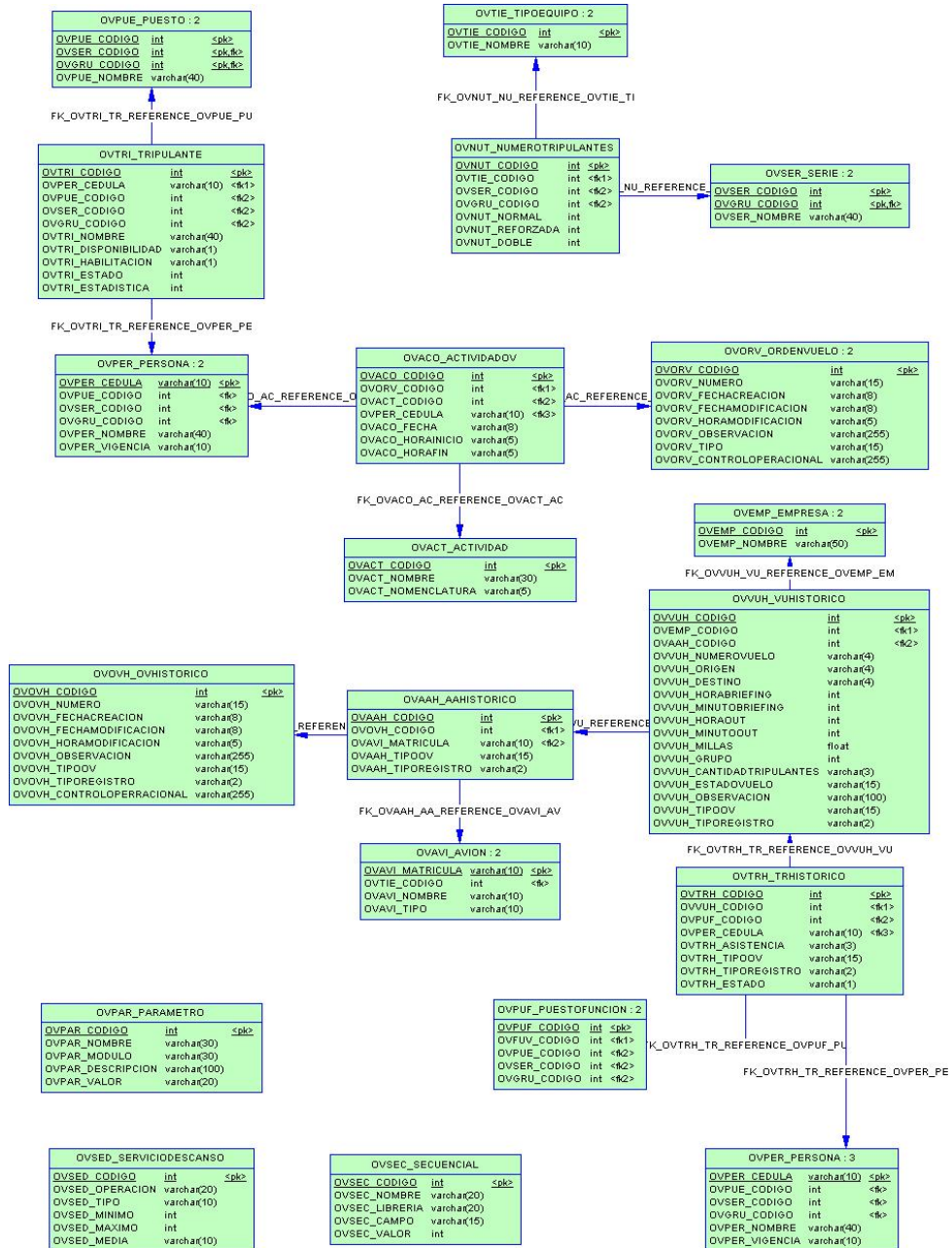


Figura 4.30 Base de Datos – Modelo de Itinerarios de Vuelo 2.

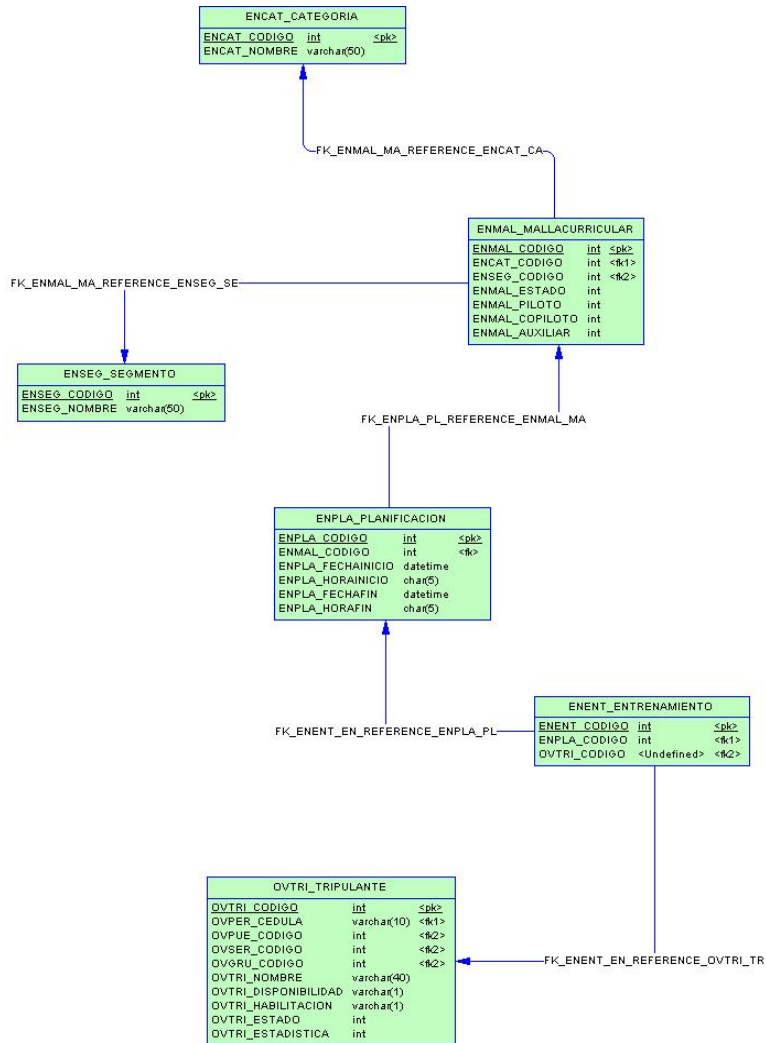


Figura 4.31 Base de Datos – Modelo de Entrenamiento.

4.3 PLAN MAESTRO

El objetivo de crear un Plan Maestro es el tener documentado los diferentes procedimientos que se adoptarán en el cumplimiento de ciertas tareas fundamentales para el cumplimiento adecuado del proyecto.

4.3.1 Plan de Entrenamiento

4.3.1.1 Objetivo

Capacitar a los jefes, técnicos informáticos y usuarios en el buen uso de la aplicación para obtener un mejor rendimiento del mismo.

4.3.1.2 Descripción del Plan

El Plan de Entrenamiento o capacitación del personal se lo realizara de acuerdo a las siguientes características:

- Duración de la capacitación: 3 horas * 10 días
- Áreas involucradas: Jefes, Informática, CCO, Entrenamiento, Programación de Aeronaves y Tripulantes.
- Lugar: Aulas de capacitación de tripulantes.

4.3.2 Plan de Pruebas

4.3.2.1 Objetivo

El Plan de Pruebas contiene las especificaciones de las pruebas que se realizarán con la finalidad de garantizar el buen funcionamiento del sistema en un ambiente de producción.

Además, verificar el cumplimiento de estándares acordados, arquitectura definida y el cumplimiento de las especificaciones funcionales establecidas junto a los usuarios.

4.3.2.2 Descripción del Plan

Con la finalidad de verificar el buen funcionamiento del sistema, se utilizarán las siguientes

Técnicas de Pruebas:

- Pruebas de Operación.
- Pruebas de Sobrecarga.
- Pruebas basadas en requisitos Pruebas de Casos de Uso.
- Pruebas de Aceptación o de Validación.

Las pruebas se las realiza en los siguientes niveles:

- Prueba de Unidad.
- Prueba de Integración.
- Prueba de Sistema.

CAPITULO V

FASE III, IV: IMPLEMENTACIÓN Y ESTABILIZACIÓN

5.1 IMPLEMENTACIÓN

En la Fase de Implementación el equipo de trabajo es responsable de crear la aplicación de acuerdo a normas establecidas en las primeras fases y siguiente el proceso de Ciclo de Vida que persigue el Microsoft Solutions Framework.

En el transcurso del proyecto y siguiendo con el cronograma de actividades, se va entregando incrementos que representan parte de la funcionalidad del sistema, más las modificaciones del anterior. Además, se cumple con los hitos de control señalados en el proyecto para una permanente evaluación, mejorando el producto y minimizando los riesgos.

La codificación del producto se deberá también realizar en base a estándares tales como:

- Colocar en cada Clase, Variable, Método, etc. comentarios de guía.
- Establecer normas para nombrar las variables, métodos, etc.
- Registrar en un documento los siguientes objetos con la finalidad de ser reutilizados:
 - Clases y funciones genéricas.
 - Procedimientos almacenados
 - Controles reutilizables.
 - Servicios Web genéricos, entre otros.

5.1.1 Script de Instalación de la Aplicación

Para la instalación de la aplicación se deberá seguir los siguientes pasos:

- Servidor de Base de Datos:
 - Instalar la base de datos de la aplicación
 - Dar los permisos necesarios en la base de datos para acceder desde la aplicación.
- Servidor de Aplicaciones:
 - Instalar el Servidor Web IIS.
 - Instalar los componentes de Acceso a Datos.
 - Instalar los componentes de Entidades de Negocio.
 - Instalar los componentes de Lógica de Negocio.
 - Crear el Sitio Web e instalar los Web Services.
 - Configurar la autenticación del sitio web.
 - Configurar los archivos web.config y dataconfiguration.config.
- Clientes:
 - Instalar la aplicación en cada cliente.
 - Verificar el acceso a los servicios web del servidor de aplicaciones.
 - Para cada actualización de la aplicación se podrá utilizar ClickOne.

5.2 ESTABILIZACIÓN

En todo proyecto informático la Fase de Estabilización o Pruebas debe ser realizada con mucho cuidado y con personal seleccionado de acuerdo a la metodología adoptada. Esto nos garantizará la entrega de un producto que cumpla con la funcional requerida.

5.2.1 Técnicas de Pruebas

5.2.1.1 Pruebas de Operación.

Para medir la confiabilidad del sistema, la presente aplicación luego del ser instalada en el Ambiente de Producción tuvo 3 meses antes de firmar el Acta de Entrega y Recepción, dando los siguientes resultados:

- Existieron cambios pero que nunca afectaron a la funcionalidad requerida al inicio del proyecto.
- El sistema cumple con las necesidades de la Empresa.

5.2.1.2 Pruebas de Sobrecarga

Al sistema se realizó pruebas de sobrecarga en horas pico, dando como resultados los siguientes:

- En un principio del proyecto al existir bastante tráfico en la red, la aplicación se volvía lenta, por tal motivo se mejoró la red contratando mayor ancho de banda y adquiriendo mejores equipos de transmisión.
- Se comprobó que al colocar los Servicios Web en el servidor ubicado en el Aeropuerto, la aplicación se volvía lenta, por tanto se adoptó por utilizar el Servidor de Aplicaciones de la Matriz, mejorando notablemente el rendimiento de la aplicación ya que son computadoras potentes.

5.2.1.3 Pruebas basadas en requisitos o Pruebas de Casos de Uso.

Se trata de verificar que la aplicación este desarrolla de acuerdo a los requerimientos de la Empresa.

Según la metodología adoptada las pruebas de validación se apoyo en los Casos de Uso diseñados, con los que permanentemente se evaluó la funcionalidad del sistema, y de acuerdo a este se obtuvo el siguiente resultado:

- El sistema cumple con las especificaciones funcionales requeridas.

5.2.1.4 Prueba de Aceptación o de Validación

De acuerdo al Acta de Entrega – Recepción que fue firmada por las Autoridades de la Empresa, el Sistema Computarizado para Gestionar y Controlar la Orden de Vuelo en la Empresa de Transporte Aéreo TAME, cumple con los requerimientos establecidos en el proyecto y las normas señaladas por las Autoridades.

5.2.2 Niveles de Pruebas

Las pruebas se las realizaron en los siguientes niveles: Prueba de Unidad, Prueba de Integración y Prueba de Sistema, obteniendo los siguientes resultados:

- El sistema esta apto para trabajar en forma local o en red con uno o varios usuarios a la vez.
- Se integran sin inconvenientes los procesos automatizados en las Gerencias y Departamentos involucrados en la emisión de la Orden de Vuelo.
- Al ser un proyecto con una visión muy amplia, esta apto para integrarse a otros sistemas informáticos de la Empresa por su arquitectura y herramientas utilizadas.

CAPITULO VI

CONCLUSIONES Y RECOMENDACIONES

6.1 CONCLUSIONES

- La automatización de la Orden de Vuelo en la Empresa de Transporte Aéreo TAME involucró el trabajo y conocimientos de varias personas, que con su apoyo y dedicación permitieron el mejoramiento en los procesos involucrados y la exitosa finalización del Sistema Informático.
- El Sistema de Gestión y Control de la Orden de Vuelo, es actualmente una de las herramientas que apoya notablemente al cumplimiento de las Operaciones Aéreas y a la toma de decisiones por parte de las autoridades.
- El seguir una Metodología de Desarrollo de Sistemas basado en el MSF permitió el cumplimiento organizado y bien definido de las diferentes fases pertenecientes al ciclo de vida del sistema. Además, con el uso de formatos y matrices, se logró definir claramente los requerimientos del usuario, arquitectura del sistema y el uso del Lenguaje de Modelado Unificado.
- El desarrollo de la aplicación se encuentra basado en la arquitectura y herramientas de Microsoft Visual Studio .NET, permitiendo así la construcción de una Aplicación Distribuida estable y apta para la implantación en la Empresa.

6.2 RECOMENDACIONES

- Para que el Sistema Informático apoye de mejor forma a los procesos que se ejecutan día a día en el Centro de Control Operacional de la Empresa, es importante tomar en cuenta el cumplimiento de procedimientos y horarios que se logró establecer en todo el tiempo transcurrido para la finalización del proyecto.
- Para el desarrollo y finalización del sistema se realizó una serie de estudios en el campo del Desarrollo de Sistemas Informáticos y Operaciones Aéreas, es por eso que al ser un proyecto con buenas bases y conocimientos, se recomienda dar continuidad e integración de otras Gerencias y Departamentos de la Empresa.
- La automatización de los procesos que integran a una empresa, es una tarea de mucha dedicación y cuidado, por lo tanto es importante siempre seguir una serie de procedimientos y tareas que dictan las Metodologías de Desarrollo de Sistemas, ya que estas son un conjunto de conocimientos, experiencias y estudios realizados por otros autores.
- Al ser un sistema integrado a otros procesos automatizados en la Empresa TAME, se recomienda que las seguridades sean gestionadas a través de controles con mayor capacidad y seguridad en la información, tales como algoritmos que permitan encriptar la información que emite los Servicios Web del sistema o con el uso del protocolo SSL que ayuda a proteger la información a medida que es transmita por la red.

BIBLIOGRAFIA

REFERENCIA BIBLIOGRÁFICAS DE TEXTOS IMPRESOS

- Pressman, Roger. Ingeniería de Software. Mc. Graw Hill. México 2005
- Pressman, Roger. Ingeniería de Software. Mc. Graw Hill. España 2002.
- Weitzenfeld, Alfredo. Ingeniería de Software Orientada a Objetos con UML Java Internet. Thomson. México 2005.
- Schildt, Herbert. Manual de Referencia C#. Mc. Graw Hill. España 2003
- Riordan, Rebecca. ADO NET. Mc. Graw Hill .España 2002
- Jacobson, Booch, Rumbauch. El Proceso Unificado de Desarrollo de Software. Addison Wesley. España. 2000.
- Jacobson, Booch, Rumbauch. El Lenguaje Unificado de Modelado. Addison Wesley. España. 1999
- Microsoft. Analyzing Requirements and Defining Microsoft .NET Solution Arquitecturas.

REFERENCIA BIBLIOGRÁFICAS DE DOCUMENTOS EN INTERNET

- <http://www.microsoft.com/latam/technet/articulos/199911/art03/>
- <http://www.willydev.net/descargas/articulos/general/MSF.aspx>
- <http://www.upt.edu.pe/EPIS/RevistaTI/AnaDisFisNet.asp>
- http://www.mentores.net/articulos/intro_microsoft_sol_frame.htm
- http://www.willydev.net/descargas/WillyDEV_Especializacion_MSf.pdf
- <http://www.microsoft.com/spanish/MSDN/estudiantes/ingsoft/planificacion/msf.asp>

GLOSARIO DE TÉRMINOS

AOC.- Certificado de Operaciones Aéreas.

DAC.- Dirección de Aviación Civil.

RDAC.-Regulaciones de la Dirección de Aviación Civil.

CCO.- Centro de Control Operacional.

MSF. - Microsoft Solutions Framework.

UML.- Lenguaje Unificado de Modelado.

RUP.- Proceso Unificado de Rational

Framework.- Marco de Trabajo.

SSL.- Secure Sockets Layer.

IIS.- Internet Information Server.

XML.- Lenguaje de Marcas Extendido.